Math-for-industry
Education & Research Hub

ISSN2188-1200

IMI Workshop of the Joint Usage Research Projects

Construction of Mathematical Basis for Realizing Data Rating Service

Editors: Katsuki Fujisawa, Shizuo Kaji, Toru Ishihara, Masaaki Kondo, Yuji Shinano, Takuji Tanigawa, Naoko Nakayama

九州大学マス・フォア・インダストリ研究所

MI Lecture Note Vol.91 : Kyushu University

IMI Workshop of the Joint Usage Research Projects

Construction of Mathematical Basis for Realizing Data Rating Service

■**Editors:**

Katsuki Fujisawa, Shizuo Kaji, Toru Ishihara, Masaaki Kondo,
Yuji Shinano, Takuji Tanigawa, Naoko Nakayama

About MI Lecture Note Series

The Math-for-Industry (MI) Lecture Note Series is the successor to the COE Lecture Notes, which were published for the 21st COE Program "Development of Dynamic Mathematics with High Functionality," sponsored by Japan's Ministry of Education, Culture, Sports, Science and Technology (MEXT) from 2003 to 2007. The MI Lecture Note Series has published the notes of lectures organized under the following two programs: "Training Program for Ph.D. and New Master's Degree in Mathematics as Required by Industry," adopted as a Support Program for Improving Graduate School Education by MEXT from 2007 to 2009; and "Education-and-Research Hub for Mathematics-for-Industry," adopted as a Global COE Program by MEXT from 2008 to 2012.

In accordance with the establishment of the Institute of Mathematics for Industry (IMI) in April 2011 and the authorization of IMI's Joint Research Center for Advanced and Fundamental Mathematics-for-Industry as a MEXT Joint Usage / Research Center in April 2013, hereafter the MI Lecture Notes Series will publish lecture notes and proceedings by worldwide researchers of MI to contribute to the development of MI.

October 2022
Kenji Kajiwara
Director, Institute of Mathematics for Industry

IMI Workshop of the Joint Usage Research Projects

Construction of Mathematical Basis for Realizing Data Rating Service

MI Lecture Note Vol.91, Institute of Mathematics for Industry, Kyushu University
ISSN 2188-1200
Date of issue: December 27, 2022
Editors: Katsuki Fujisawa, Shizuo Kaji, Toru Ishihara, Masaaki Kondo, Yuji Shinano,
Takuji Tanigawa, Naoko Nakayama
Publisher:
Institute of Mathematics for Industry, Kyushu University
Graduate School of Mathematics, Kyushu University
Motooka 744, Nishi-ku, Fukuoka, 819-0395, JAPAN
Tel +81-(0)92-802-4402, Fax +81-(0)92-802-4405
URL https://www.imi.kyushu-u.ac.jp/

Preface

In November 2020, the Institute of Mathematics for Industry (IMI) of Kyushu University, SoftBank Corporation and MAMEZOU Corporation began joint research on the realization of a "data rating" system that will use mathematical theory to objectively determine the quality of various types of digital data ("data") accumulated by companies, local governments, educational and research institutions, etc. The three parties will use the "data rating" system to clarify the quality of data held by industry, government, and academia. By clarifying the quality of data held by industry, government, and academia through "data rating," the three parties aim to promote the mutual use of data and revitalize the data distribution market.

This IMI Workshop of the Joint Usage Research Projects “Construction of Mathematical Basis for Realizing Data Rating Service” held on September 21st and 22, 2022. And this workshop was held jointly with the following international workshop.

The 6th RIKEN-IMI-ISM-NUS-ZIB-MODAL-NHR Workshop on Advances in Classical and Quantum Algorithms for Optimization and Machine Learning

RIKEN, IMI, The Institute of Statistical Mathematics (ISM), the National University of Singapore (NUS), the Zuse Institute Berlin (ZIB), and the NHR Center at ZIB hold the sixth workshop on mathematical optimization and related fields. This workshop was held at the University of Tokyo from September 16 to 19 and at Kyushu University from September 21 to 22, 2022. The workshop also discussed methodologies for establishing a new mathematical foundation (algorithm) for "data rating," building theory, and conducting empirical experiments. This lecture note contains the materials of the lectures given at the workshop, and the Japanese-language lectures were edited in a separate volume. For more information about this workshop, please refer to the website below [1].

November 2022.

Editors
Katsuki Fujisawa, Shizuo Kaji (Kyushu University)
Toru Ishihara (Nagoya University)
Masaaki Kondo (Keio University)
Yuji Shinano (Zuse Institute Berlin)
Takuji Tanigawa (SoftBank Corp.)
Naoko Nakayama (MAMEZOU Corp.)

2022年度 九州大学マス・フォア・インダストリ研究所 共同利用・共同研究 女性研究者活躍支援研究-研究集会(I)

データ格付けサービス実現のための数理基盤の構築

2022.9.21(水)–22(木)

九州大学 伊都キャンパス ウエスト1号館 C棟
2階教育情報システム室 (W1-C-201)
Computer Lab (W1-C-201), West Zone 1,
Ito campus, Kyushu University

Construction of Mathematical Basis for Realizing Data Rating Service

参加無料
事前申込制

Program

■9月21日(水)

The 6th RIKEN-IMI-ISM-NUS-ZIB-MODAL-NHR Workshop on Advances in Classical and Quantum Algorithms for Optimization and Machine Learning

09:30–11:30 Session 13
- Thorsten Koch (ZIB)
- João Doriguello (NUS)
- Ralf Borndörfer (ZIB)

11:00–11:20 Coffee Break

11:20–12:20 Session 14
- Pierre-Louis Poirion (RIKEN)
- Akifumi Okuno (ISM / RIKEN)

12:20–13:50 Lunch Break

13:50–15:10 Session 15
- Niels Lindner (ZIB)
- Inci Yüksel-Ergün (ZIB)
- Jaap Pedersen (ZIB)

15:10–15:30 Break

15:30–16:50 Session 16
- Uwe Gotzes (Open Grid Europe GmbH)
- Ying Chen (NUS)
- Osamu Saeki (IMI)
- Katsuki Fujisawa (IMI)

■9月22日(木)

10:30–11:10
株式会社豆蔵 執行役員 C.D.O. (DX事業推進統括)
安井 昌男

11:15–12:00
株式会社Fixstars Amplify 代表取締役社長 CEO
平岡 卓爾

12:00–13:30 昼休憩

13:30–14:00
株式会社豆蔵 デジタル戦略支援事業 AI-Techチーム コンサルタント
林 沛萱

14:15–15:00
立教大学 大学院人工知能科学研究科 特任准教授
株式会社豆蔵
デジタル戦略支援事業 AI-Techチームリーダー チーフ・コンサルタント
石川 真之介

15:15–16:00
横河電機株式会社 常務執行役員(CIO)
デジタル戦略本部長 兼 デジタルソリューション本部 DX−Platformセンター長
舩生 幸宏

16:15–16:50
株式会社豆蔵 チーフ・コンサルタント
中山 尚子

[研究代表者]
中山 尚子(株式会社豆蔵)

[組織委員]
谷川 拓司(ソフトバンク株式会社)
品野 勇治(Zuse Institute Berlin)
近藤 正章(慶応大学)
石原 亨 (名古屋大学)
鍛冶 静雄(九州大学)
藤澤 克樹(九州大学)

[共催機関]
理化学研究所(RIKEN)
九州大学マス・フォア・インダストリ研究所(IMI)
統計数理研究所(ISM)
National University of Singapore(NUS)
Zuse Institute Berlin (ZIB)
NHR Center at ZIB

最新情報
最新情報は、下記のホームページをご覧ください
URL:https://sites.google.com/view/6th-riken-imi-ism-zib-workshop/program

参加申込
参加にあたっては、下記の参加申込をお願いします
URL:https://forms.gle/H3rA1D7EmHMGKQtC6
【申込フォーム】

E-mail:imikyoten@jimu.kyushu-u.ac.jp
https://joint.imi.kyushu-u.ac.jp/post-6137/

Institute of Mathematics for Industry
Kyushu University

Joint Research Center for Advanced and Fundamental Mathematics-for-Industry
文部科学大臣認定「産業数学の先進的・基礎的共同研究拠点」
九州大学マス・フォア・インダストリ研究所

Program

September 17th

09:20--11:30 Session 01

- 09:20--09:30 **Akiko Takeda** (RIKEN / Univ Tokyo) "Opening Remarks"
- 09:30--10:00 **Masashi Sugiyama** (RIKEN / Univ Tokyo) "Recent Advances in Machine Learning from Noisy Labels"
- 10:00--10:30 **Patrick Gelß** (ZIB) "Low-rank tensor representations of quantum circuits"
- 10:30--11:00 **Shintaro Momose** (NEC) "Aurora Vector Annealing to Solve Social Issues and Acceleration by NEC's Supercomputer, SX-Aurora TSUBASA"

11:00--11:20 Coffee Break

11:20--12:20 Session 02

- 11:20--11:50 **Kazuma Tsuji** (MUFG Bank) "Pairwise Conditional Gradients without Swap Steps and Sparser Kernel Herding"
- 11:50--12:20 **Christoph Spiegel** (ZIB) "Proofs in Extremal Combinatorics through Optimization"
- 12:20--13:50 Lunch Break
- 13:50--15:20 Ice Breaking

15:20--16:20 Session 03

- 15:20--15:50 **Yuji Shinano** (ZIB) "The UG framework version 1.0: An update"
- 15:50--16:20 **Junko Hosoda** (Hitachi) "A parallel algorithm combining relaxation and heuristic for the integrated long-haul and local vehicle routing problem on an adaptive transportation network"

16:20--16:40 Break

16:40--17:40 Session 04

- 16:40--17:10 **Koichi Fujii** (NTT DATA MSI) "Solving Large Scale QAPs by Massively Parallel DNN-based Branch-and-bound Method"
- 17:10--17:40 **Elias Wirth** (ZIB) "Approximate Vanishing Ideal Computations at Scale"

September 18th

09:30--11:30 Session 05

- 09:30--10:00 **Katsuki Fujisawa** (Kyushu Univ) "Mobility Optimization Engine and its Real-world Applications"
- 10:00--10:30 **Hiroki Ishikura** (Kyushu Univ) "Towards an optimal operation of automated storage and retrieval system with multiple machines"
- 10:30--11:00 **Nozomi Hata** (Kyushu Univ) "Theoretical Analysis for Representation Learning Methods of Graph-Structured Data"

11:00--11:20 Coffee Break

11:20--12:20 Session 06

- 11:20--11:50 **Mark Turner** (TU Berlin) "Adaptive Cut Selection in Mixed-Integer Linear Programming"
- 11:50--12:20 **Ryohei Yokoyama** (Osaka Metro Univ) "A Quadratic Programming Approach for Performance Analysis of Energy Systems"

12:20--13:50 Lunch Break

13:50--15:50 Ice Breaking

15:50--16:50 Session 07

- 15:50--16:20 **Shizuo Kaji** (Kyushu Univ) "Geometric Learning of Ranking Distributions"
- 16:20--16:50 **Akiko Takeda** (RIKEN / Univ Tokyo) "Bilevel Optimization for Machine Learning Problems"

September 19th

09:30--11:30 Session 08

- 09:30--10:00 **Sebastian Pokutta** (ZIB) "Convex integer optimization with Frank-Wolfe methods"
- 10:00--10:30 **Shota Takahashi** (SOKENDAI / ISM) "Bregman Proximal DC Algorithms and Their Application to Blind Deconvolution with Nonsmooth Regularization"
- 10:30--11:00 **Akira Tanaka** (NICT) "Port Set Clustering for Internet-Wide Scanner"

11:00--11:20 Coffee Break

11:20--12:20 Session 09

- 11:20--11:50 **Atsushi Miyauchi** (Univ Tokyo) "Finding densest *k*-connected subgraphs"
- 11:50--12:20 **Antoine Deza** (McMaster Univ) "Worst-case constructions for linear optimization"

12:20--13:50 Lunch Break

13:50--15:20 Session 10

- 13:50--14:20 **Xun Shen** (Tokyo Inst Tech) "Approximate Methods for Solving Chance Constrained Linear Programs in Probability Measure Space"
- 14:20--14:50 **Jun-ya Gotoh** (Chuo Univ) "Knot Selection of B-Spline Regression via Trimmed Regularizer"
- 14:50--15:20 **Keisuke Yano** (ISM) "Minimum information dependence modeling and its application"

15:20--15:40 Break

15:40--17:10 Session 11

- 15:40--16:10 **Naoki Marumo** (NTT / Univ Tokyo) "A generalized Levenberg–Marquardt method for large-scale composite minimization"
- 16:10--16:40 **Shunji Umetani** (Osaka Univ) "BIPSOL: A metaheuristic solver for large-scale binary integer programs"

- 16:40--17:10 **Masahiro Nakao** (RIKEN) "Performance of the supercomputer Fugaku for Graph500 benchmark"

September 21st

09:30--11:30 Session 12

- 09:30--10:00 **Thorsten Koch** (ZIB) "Notes on Solving QUBOs and Quantum Computing"
- 10:00--10:30 **João Doriguello** (NUS) "Quantum algorithm for stochastic optimal stopping problems with applications in finance"
- 10:30--11:00 **Ralf Borndörfer** (ZIB) "Multicriteria Shortest Path Algorithms"

11:00--11:20 Coffee Break

11:20--12:20 Session 13

- 11:20--11:50 **Pierre-Louis Poirion** (RIKEN) "Randomized subspace regularized Newton method for unconstrained non-convex optimization"
- 11:50--12:20 **Akifumi Okuno** (ISM / RIKEN) "Minimax Analysis for Inverse Risk in Nonparametric Invertible Regression"

12:20--13:50 Lunch Break

13:50--15:10 Session 14

- 13:50--14:20 **Niels Lindner** (ZIB) "On the geometry of periodic timetables in public transport"
- 14:20--14:50 **Inci Yüksel-Ergün** (ZIB) "Improving data quality in the presence of superhuman complexity in data errors"
- 14:50--15:10 **Jaap Pedersen** (ZIB) "Optimal discrete pipe sizing for tree-shaped CO2 networks"

15:10--15:30 Break

15:30--16:50 Session 15

- 15:30--16:00 **Uwe Gotzes** (OGE) "Spotlights on success stories of public-private partnership"
- 16:00--16:30 **Ying Chen** (NUS) "Deep Switching State Space Model (DS3M) for Nonlinear Time Series Forecasting with Regime Switching"
- 16:30--16:40 **Osamu Saeki** (Kyushu Univ) "Institute of Mathematics for Industry: its uniqueness, strength and prospects"
- 16:40--16:50 **Katsuki Fujisawa** (Kyushu Univ) "Closing Remarks"


The 6th RIKEN-IMI-ISM-NUS-ZIB-MODAL-NHR
Workshop on Advances in Classical and Quantum

Table of contents

On the geometry of periodic timetables in public transport
Niels LINDNER (Zuse Institute Berlin, Germany) · · · · · · · · · · · · · · · · · 501

Improving Data Quality in the Presence of Superhuman Complexity in Data Errors
Inci YÜKSEL-ERGÜN (Applied Algorithmic Intelligence Methods, Zuse Institute Berlin, Germany) · · · · · · · · · · · · · · · · ·

Recent Advances in Machine Learning from Noisy Labels

Masashi SUGIYAMA

RIKEN Center for Advanced Intelligence Project/
Graduate School of Frontier Sciences, The University of Tokyo.
sugi@k.u-tokyo.ac.jp

Supervised learning from noisy output is one of the classical problems in machine learning. While this task is relatively straightforward in regression since independent additive noise cancels with big data, classification from noisy labels is still a challenging research topic. Recently, it has been shown that when the noise transition matrix which specifies the label flipping probability is available, the bias caused by label noise can be canceled by appropriately correcting the loss function. However, when the noise transition matrix is unknown, which is often the case in practice, its estimation only from noisy labels is not straightforward due to its non-identifiability. In this talk, I will give an overview of recent advances in classification from noisy labels, including joint estimation of the noise transition matrix and a classifier, analysis of identifiability conditions, and extension to instance-dependent noise.

References

[1] Patrini, G., Rozza, A., Menon, A. K., Nock, R., & Qu, L. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR2017), pp. 1944-1952, 2017.

[2] Zhang, Y., Niu, G., & Sugiyama, M. Learning noise transition matrix from only noisy labels via total variation regularization. In Proceedings of 38th International Conference on Machine Learning (ICML2021), pp. 12501-12512, 2021.

[3] Li, X., Liu, T., Han, B., Niu, G., & Sugiyama, M. Provably end-to-end label-noise learning without anchor points. In Proceedings of 38th International Conference on Machine Learning (ICML2021), pp. 6403-6413, 2021.

[4] Xia, X., Liu, T., Han, B., Wang, N., Gong, M., Liu, H., Niu, G., Tao, D., & Sugiyama, M. Parts-dependent label noise: Towards instance-dependent label noise. In Advances in Neural Information Processing Systems 33, pp. 7597-7610, 2020.

[5] Berthon, A., Han, B., Niu, G., Liu, T., & Sugiyama, M. Confidence scores make instance-dependent label-noise learning possible. In Proceedings of 38th International Conference on Machine Learning (ICML2021), pp. 825-836, 2021.

[6] Cheng, D., Liu, T., Ning, Y., Wang, N., Han, B., Niu, G., Gao, X., & Sugiyama, M. Instance-dependent label-noise learning with manifold-regularized transition matrix estimation. In Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR2022), pp. 16630-16639, 2022.

[7] Han, B., Yao, Q., Yu, X., Niu, G., Xu, M., Hu, W., Tsang, I., & Sugiyama, M. Co-teaching: Robust training deep neural networks with extremely noisy labels. In Advances in Neural Information Processing Systems 31, pp. 8527-8537, 2018.

[8] Han, B., Niu, G., Yu, X., Yao, Q., Xu, M., Tsang, I., & Sugiyama, M. SIGUA: Forgetting may make learning with noisy labels more robust. In Proceedings of 37th International Conference on Machine Learning (ICML2020), pp. 4006-4016, 2020.

[9] Sugiyama, M., Bao, H., Ishida, T., Lu, N., Sakai, T., & Niu, G. Machine Learning from Weak Supervision: An Empirical Risk Minimization Approach, MIT Press, 2022.

The 6th RIKEN-IMI-ISM-NUS-ZIB-MODAL-NHR Workshop
on Advances in Classical and Quantum Algorithms
for Optimization and Machine Learning

Sep. 17, 2022

Recent Advances in Machine Learning from Noisy Labels

Masashi Sugiyama

RIKEN Center for Advanced Intelligence Project/
The University of Tokyo

http://www.ms.k.u-tokyo.ac.jp/sugi/

RIKEN-AIP

- MEXT Advanced Intelligence Project (2016-2025):
  - 130 employed researchers (36% international, 23% female)
  - 200 visiting researchers, 100 domestic students
  - 140 international interns (total)
- Missions:
  - Develop new AI technology (ML, Opt, math)
  - Accelerate scientific research (cancer, material, genomics)
  - Solve socially critical problems (disaster, elderly healthcare)
  - Study of ELSI in AI (ethical guidelines, personal data)
  - Human resource development (researchers, engineers)

Main office in the heart of Tokyo

Distributed offices across Japan

Standard Supervised Learning

3

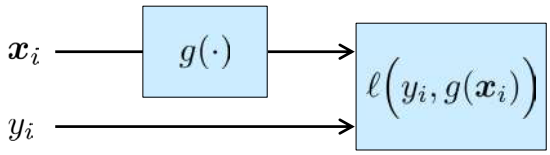
- Goal: Learn a predictor $y = f(\boldsymbol{x})$ from input-output training data $\{(\boldsymbol{x}_i, y_i)\}_{i=1}^n$.

  $\boldsymbol{x} \in \mathbb{R}^d$: Input y : Output

- Approach: Training error minimization (a.k.a. empirical risk minimization or maximum likelihood estimation)

$$\hat{f} = \underset{g}{\operatorname{argmin}} \left[\frac{1}{n} \sum_{i=1}^{n} \ell\Big(y_i, g(\boldsymbol{x}_i)\Big) \right]$$

$\ell(\cdot,\cdot)$: Loss

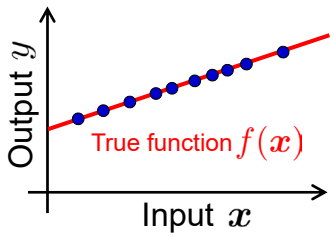


Without Output Noise

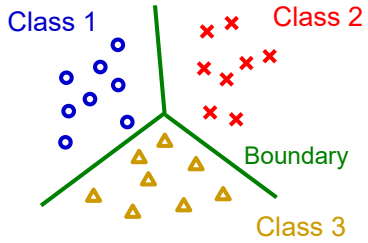
4

- Suppose there is no noise in output: $y_i = f(\boldsymbol{x}_i)$

Regression $y \in \mathbb{R}$



Classification $y \in \{1, \ldots, c\}$



- Training error minimization is statistically consistent:

$$\hat{f} = \underset{g}{\operatorname{argmin}} \left[\frac{1}{n} \sum_{i=1}^{n} \ell\Big(y_i, g(\boldsymbol{x}_i)\Big) \right]$$

  - When $n \to \infty$, $\hat{f}$ converges to true f. $\{(\boldsymbol{x}_i, y_i)\}_{i=1}^n$

- Big data helps!

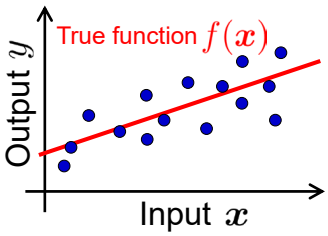
With Output Noise

5

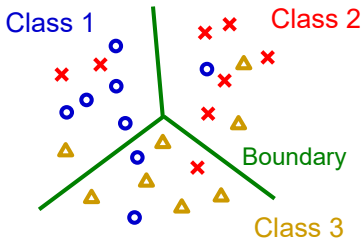
- Output y_i is often corrupted by noise:
  - Due to sensor errors, human errors, etc.

$$y_i \to \bar{y}_i$$

Regression



Classification

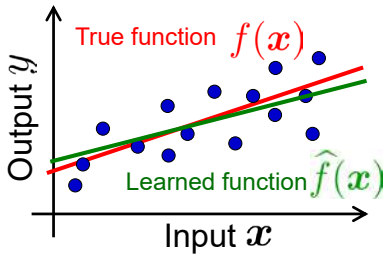


- In the noisy case, does big data still help?

Noisy-Output Regression

6

- Standard noise assumptions:
  - Additive: $\bar{y}_i = f(\boldsymbol{x}_i) + \varepsilon_i$
  - Input-independent: $\boldsymbol{x}_i \perp\!\!\!\perp \varepsilon_i$
  - Zero-mean: $\mathbb{E}[\varepsilon_i] = 0$



- Noisy training error minimization is still statistically consistent.

$$\hat{f} = \operatorname*{argmin}_{g} \left[\frac{1}{n} \sum_{i=1}^{n} \ell\Big(\bar{y}_i, g(\boldsymbol{x}_i)\Big) \right]$$

- Naïve use of big data still helps!

Noisy-Output Classification

7

- Standard assumptions:
  - Class-conditional noise (input-independent flip):

$$\bar{y}_i \sim \bar{p}(\bar{y}|\boldsymbol{x}) = \sum_{j=1}^{C} p(\bar{y}|y=j)p(y=j|\boldsymbol{x})$$

- Noisy training error minimization is not always statistically consistent:

$$\hat{f} = \operatorname*{argmin}_{g} \left[\frac{1}{n} \sum_{i=1}^{n} \ell\Big(\bar{y}_i, g(\boldsymbol{x}_i)\Big) \right]$$

- Need to explicitly remove the influence of label noise in learning!

Generic Approach (1)

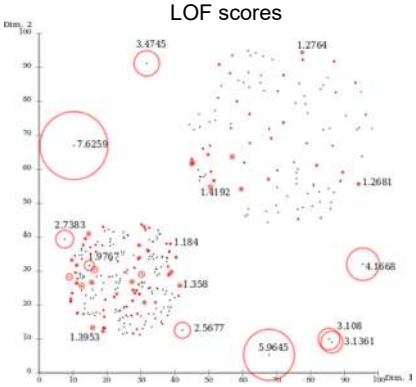
8

- Unsupervised noisy data removal:
  - Hotelling's T^2 statistics
  - k-means clustering
  - local outlier factor (LOF)

Breunig et al. (SIGMOD2000)

https://en.wikipedia.org/wiki/Harold_Hotelling

LOF scores



https://en.wikipedia.org/wiki/Local_outlier_factor

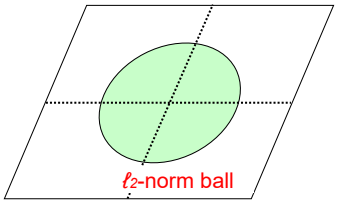
- Easy to use, but this is completely heuristic and no supervision is used.

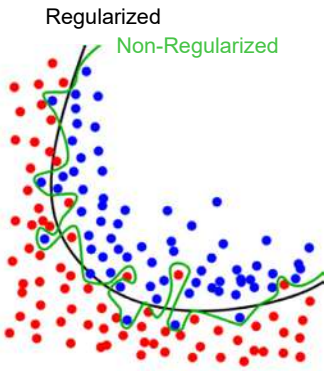
Generic Approach (2)

9

- Regularization: keeping the norm of the model parameters small for preventing overfitting.
  - Tikhonov regularization

https://en.wikipedia.org/wiki/Andrey_Nikolayevich_Tikhonov





https://en.wikipedia.org/wiki/Overfitting

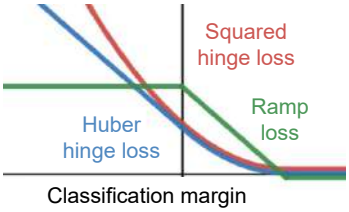
- Nice theory, but smoothing is not enough to cope with strong label noise.

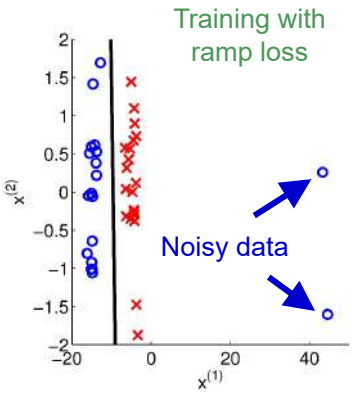
Generic Approach (3)

10

- Robust statistics: suppressing the influence of noisy data by a gentle loss.
  - Huber loss
  - Ramp loss

https://en.wikipedia.org/wiki/Peter_J._Huber





- Nice theory for regression (additive noise), but not very robust in classification (flipping noise).

Goal of Noisy-Label Classification

11

- These generic approaches were not specifically designed for handling label noise.
- In this talk, I review recent advances in noisy-label multi-class classification that explicitly handle noisy supervision:
  - Forward/backward loss correction.
  - Noise transition estimation.
  - Coping with non-identifiability.
  - Input-dependent label noise.

Contents

12

Formulation

13

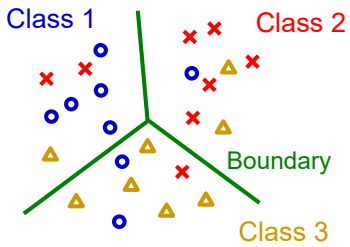
- Clean training data: $\{(\boldsymbol{x}_i, y_i)\}_{i=1}^n \overset{\text{i.i.d.}}{\sim} p(\boldsymbol{x}, y)$
- Noisy training data: $\{(\boldsymbol{x}_i, \bar{y}_i)\}_{i=1}^n \overset{\text{i.i.d.}}{\sim} \bar{p}(\boldsymbol{x}, \bar{y})$

$\boldsymbol{x} \in \mathbb{R}^d$: Input instance
$y \in \{1, \ldots, c\}$: Clean class label
$\bar{y} \in \{1, \ldots, c\}$: Noisy class label

- Probabilistic classifier in simplex: $\boldsymbol{h}(\boldsymbol{x}) \in \Delta^{c-1}$
  - Each element approximates the class-posterior probability.

$$h_y(\boldsymbol{x}) \approx p(y|\boldsymbol{x})$$

- Loss: $\ell(y, \boldsymbol{h}(\boldsymbol{x})) \in \mathbb{R}$



Modeling Class-Conditional Noise

14

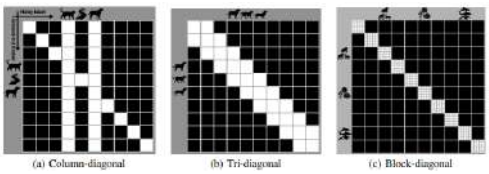
- Noise transition matrix: $T_{y,\bar{y}} = \bar{p}(\bar{y}|y)$
  - Probability of flipping y to $\bar{y}$.

y

1	0	0
0.1	0.8	0.1
0.5	0.5	0

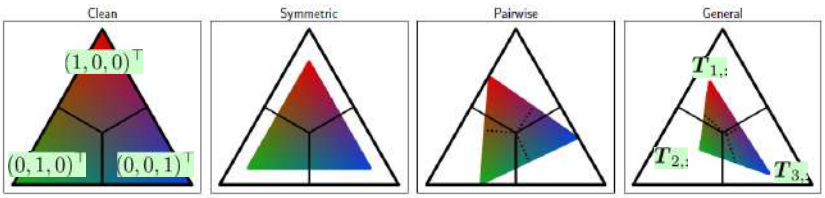
$\bar{y}$

- We may encode human-cognitive bias:



Han, Yao, Niu, Zhou, Tsang, Zhang & Sugiyama (NeurIPS2018)

- Visualization as a simplex: Zhang, Niu & Sugiyama (ICML2021)



Loss Correction

15

Patrini, Rozza, Menon, Nock & Qu (CVPR2017)

- Forward correction: Add noise by $\boldsymbol{T}^\top$
  - $\boldsymbol{\ell}^{\rightarrow}(\boldsymbol{h}(\boldsymbol{x})) = \boldsymbol{\ell}(\boldsymbol{T}^\top \boldsymbol{h}(\boldsymbol{x}))$ $\quad \ell_y(\boldsymbol{h}(\boldsymbol{x})) = \ell(y, \boldsymbol{h}(\boldsymbol{x}))$

Classifier-consistency

$$\operatorname*{argmin}_{\boldsymbol{h}} \mathbb{E}_{\bar{p}(\boldsymbol{x},\bar{y})}[\ell^{\rightarrow}(y, \boldsymbol{h}(\boldsymbol{x}))] = \operatorname*{argmin}_{\boldsymbol{h}} \mathbb{E}_{p(\boldsymbol{x},y)}[\ell(y, \boldsymbol{h}(\boldsymbol{x}))]$$

- Backward correction: Remove noise by $\boldsymbol{T}^{-1}$
  - $\boldsymbol{\ell}^{\leftarrow}(\boldsymbol{h}(\boldsymbol{x})) = \boldsymbol{T}^{-1}\boldsymbol{\ell}(\boldsymbol{h}(\boldsymbol{x}))$

Classifier-consistency

$$\operatorname*{argmin}_{\boldsymbol{h}} \mathbb{E}_{\bar{p}(\boldsymbol{x},\bar{y})}[\ell^{\leftarrow}(y, \boldsymbol{h}(\boldsymbol{x}))] = \operatorname*{argmin}_{\boldsymbol{h}} \mathbb{E}_{p(\boldsymbol{x},y)}[\ell(y, \boldsymbol{h}(\boldsymbol{x}))]$$

Risk-consistency

$$\forall \boldsymbol{x},\ \mathbb{E}_{\bar{p}(\bar{y}|\boldsymbol{x})}[\ell^{\leftarrow}(y, \boldsymbol{h}(\boldsymbol{x}))] = \mathbb{E}_{p(y|\boldsymbol{x})}[\ell(y, \boldsymbol{h}(\boldsymbol{x}))]$$

- If $\boldsymbol{T}$ is given, consistency can be guaranteed!

Identifiability of Noise Transition

16

- In practice, we need to estimate $\boldsymbol{T}$ from noisy training data $\{(\boldsymbol{x}_i, \bar{y}_i)\}_{i=1}^n$.

- However, $\boldsymbol{T}$ is non-identifiable in general:
  - $\boldsymbol{T}$ can be decomposed as $\boldsymbol{T} = \boldsymbol{U}\boldsymbol{V}$, where $\boldsymbol{U}, \boldsymbol{V}$ are some transition matrices.
  - Then $\bar{\boldsymbol{p}}_{\boldsymbol{x}} = \boldsymbol{T}^\top \boldsymbol{p}_{\boldsymbol{x}}$
    $= \boldsymbol{V}^\top(\boldsymbol{U}^\top \boldsymbol{p}_{\boldsymbol{x}})$

$$T_{y,\bar{y}} = \bar{p}(\bar{y}|y)$$
$$[\bar{\boldsymbol{p}}_{\boldsymbol{x}}]_{\bar{y}} = \bar{p}(\bar{y}|\boldsymbol{x})$$
$$[\boldsymbol{p}_{\boldsymbol{x}}]_y = p(y|\boldsymbol{x})$$

- Let's use anchor points (100%-certain samples):

$$\{\boldsymbol{x}^y \mid p(y|\boldsymbol{x}^y) = 1\}_{y=1}^c$$

Estimation of Noise Transition with Anchor Points

17

- Given anchor points $\{\boldsymbol{x}^y \mid p(y|\boldsymbol{x}^y) = 1\}_{y=1}^c$, $T_{y,\bar{y}} = \bar{p}(\bar{y}|y)$ can be naïvely estimated as

$$T_{y,\bar{y}} = \sum_{y'=1}^{c} p(\bar{y}|y')p(y'|\boldsymbol{x}^y) = \bar{p}(\bar{y}|\boldsymbol{x}^y) \approx \bar{h}_{\bar{y}}(\boldsymbol{x}^y)$$

  - $\bar{\boldsymbol{h}}(\boldsymbol{x})$ is a probabilistic classifier learned from noisy training data $\{(\boldsymbol{x}_i, \bar{y}_i)\}_{i=1}^n$.

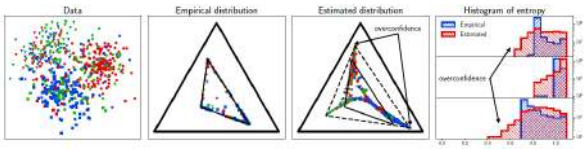
- Even if anchor points are unknown, as long as they exist in noisy training data, we may find them as $\boldsymbol{x}^y \leftarrow \boldsymbol{x}_i \text{ s.t. } \bar{h}_y(\boldsymbol{x}_i) \approx 1$.

Further Improvements

18

$$\boldsymbol{x}^y \leftarrow \boldsymbol{x}_i \text{ s.t. } \bar{h}_y(\boldsymbol{x}_i) \approx 1$$

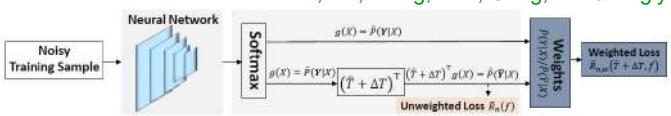
- We typically use deep learning to obtain $\bar{\boldsymbol{h}}(\boldsymbol{x})$:
  - Then it is often over-confident and unreliable.



Zhang, Niu & Sugiyama (ICML2021)

- Estimated $\boldsymbol{T}$ is revised during classifier training:

Xia, Liu, Wang, Han, Gong, Niu & Sugiyama (NeurIPS2019)



- Instead of explicitly finding anchor points, latent labels are utilized: $y_i' = \mathrm{argmax}_{y'} \bar{h}_{y'}(\boldsymbol{x}_i)$

Yao, Liu, Han, Gong, Deng, Niu, Sugiyama & Tao (NeurIPS2020)

Contents

19

Challenge

20

- Current approaches are in two-step:
  1. Estimate transition matrix $\boldsymbol{T}$.
  2. Use estimated $\boldsymbol{T}$ to train a classifier $\boldsymbol{h}(\boldsymbol{x})$.
- Step 1 is done without regard to Step 2:
  - Estimation error of $\boldsymbol{T}$ in Step 1 can be magnified in Step 2.
- We want to estimate $\boldsymbol{T}$ and $\boldsymbol{h}(\boldsymbol{x})$ simultaneously in one-step.

Naïve Solution

21

- Naively, we may learn the noise transition and classifier at the same time as

$$\min_{\boldsymbol{U},\boldsymbol{h}} \mathbb{E}_{\bar{p}(\boldsymbol{x},\bar{y})}[\ell(\bar{y}, \boldsymbol{U}^\top \boldsymbol{h}(\boldsymbol{x}))]$$

- However, the solution is not unique:
  - With any invertible transition matrix $\boldsymbol{Q}$, any $(\widehat{\boldsymbol{U}}, \widehat{\boldsymbol{h}}) = (\boldsymbol{Q}^{-1}\boldsymbol{T}, \boldsymbol{Q}^\top \boldsymbol{p}_{\boldsymbol{x}})$ are solutions.

$$T_{y,\bar{y}} = \bar{p}(\bar{y}|y) \qquad [\boldsymbol{p}_{\boldsymbol{x}}]_y = p(y|\boldsymbol{x})$$

- We need a certain constraint to obtain the right solution: $(\widehat{\boldsymbol{U}}, \widehat{\boldsymbol{h}}) = (\boldsymbol{T}, \boldsymbol{p}_{\boldsymbol{x}})$

Total Variation Regularization

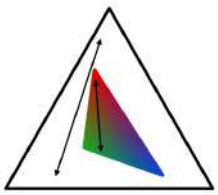
22

Zhang, Niu & Sugiyama (ICML2021)

- Noise transition $\boldsymbol{p}_{\boldsymbol{x}} \to \boldsymbol{U}^\top \boldsymbol{p}_{\boldsymbol{x}}$ is contraction in total variation distance:

$$\|\boldsymbol{U}^\top \boldsymbol{p}_{\boldsymbol{x}} - \boldsymbol{U}^\top \boldsymbol{p}_{\boldsymbol{x}'}\|_1 \leq \|\boldsymbol{p}_{\boldsymbol{x}} - \boldsymbol{p}_{\boldsymbol{x}'}\|_1$$

$$[\boldsymbol{p}_{\boldsymbol{x}}]_y = p(y|\boldsymbol{x})$$



  - Cleaner class-posteriors have a larger total variation distance!

- Let's use this knowledge as a regularizer:

$$\min_{\boldsymbol{U},\boldsymbol{h}} \left[\mathbb{E}_{\bar{p}(\boldsymbol{x},\bar{y})}[\ell(\bar{y}, \boldsymbol{U}^\top \boldsymbol{h}(\boldsymbol{x}))] - \lambda \mathbb{E}_{p(\boldsymbol{x}),p(\boldsymbol{x}')}\|\boldsymbol{h}(\boldsymbol{x}) - \boldsymbol{h}(\boldsymbol{x}')\|_1\right]$$

  - Under the anchor point assumption, $\lambda > 0$ the empirical solution has statistical consistency.

Contents

23

Challenges

24

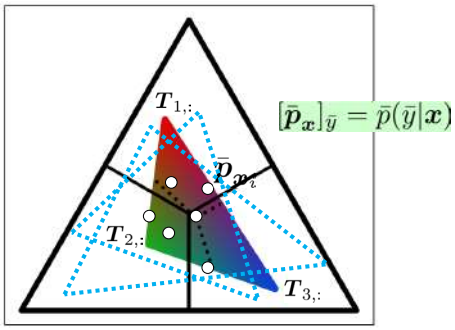
$$\{\boldsymbol{x}^y \mid p(y|\boldsymbol{x}^y) = 1\}_{y=1}^{c}$$

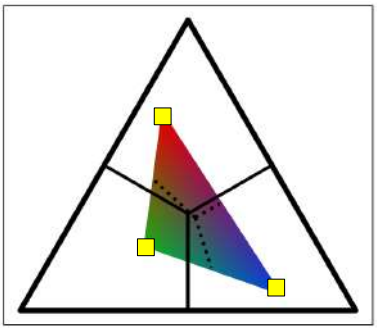
- To overcome the non-identifiability of $\boldsymbol{T}$:
  - Anchor points are explicitly used.
- This condition has been relaxed to:
  - Only the existence of anchor points is assumed
- Can we further relax this assumption?

Non-identifiability of $\boldsymbol{T}$

25

- $\boldsymbol{T}$ can be visualized as a simplex, containing all training data.
- Generally, such a simplex is not unique.
- Anchor points are vertices of the true simplex.
  - Explicitly using anchor points naively recovers $\boldsymbol{T}$.

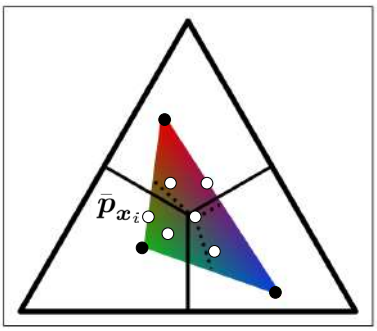


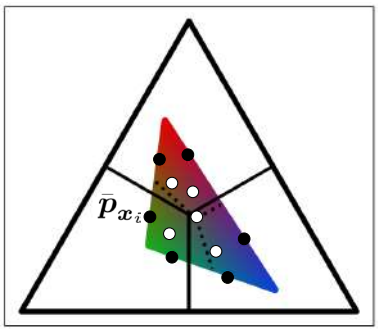


Non-identifiability of $\boldsymbol{T}$ (cont.)

26

- Only the existence of anchor points still guarantees the identifiability of $\boldsymbol{T}$.
- Even without anchor points, "sufficiently scattered" training data can guarantee the consisntency (with the next algorithm).





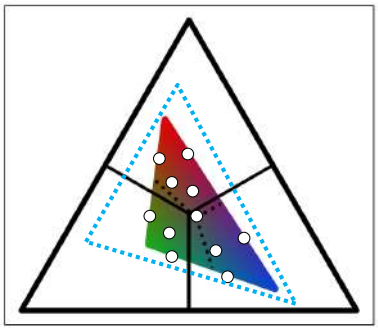
Volume Minimization

27

Li, Liu, Han, Niu & Sugiyama (ICML2021)

- Under the "sufficiently scattered" assumption, minimizing the volume of the transition matrix guarantees consistency!

$$\min_{\boldsymbol{U},\boldsymbol{h}} \left[\mathbb{E}_{\bar{p}(\boldsymbol{x},\bar{y})}[\ell(\bar{y}, \boldsymbol{U}^\top \boldsymbol{h}(\boldsymbol{x}))] + \lambda \log\det(\boldsymbol{U}) \right] \quad \lambda > 0$$



Contents

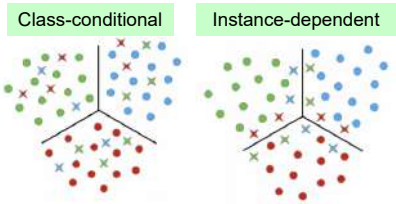
28

1. Introduction
2. Technical background
3. Single-step approach
4. Beyond anchor points
5. Further challenges

Beyond Class-Conditional Noise

29

- Instance-independence in class-conditional noise is restrictive.



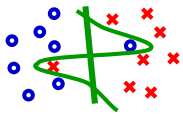
- Instance-dependent noise: $T_{y,\bar{y}}(\boldsymbol{x}) = \bar{p}(\bar{y}|y, \boldsymbol{x})$
  - Extremely challenging problem!

- Various heuristic solutions:
  - Parts-based estimation — Xia, Liu, Han, Wang, Gong, Liu, Niu, Tao & Sugiyama (NeurIPS2020)
  - Use of additional confidence scores — Berthon, Han,Niu, Liu & Sugiyama (ICML2021)
  - Manifold regularization — Cheng, Liu, Ning, Wang, Han, Niu, Gao & Sugiyama (CVPR2022)

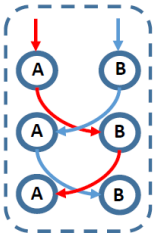
Co-teaching

30

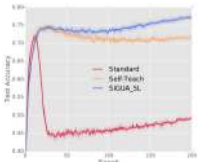
- Memorization of neural nets: Arpit et al. (ICML2017) Zhang et al. (ICLR2017)
  - Stochastic gradient descent fits clean data faster.
  - However, naïve early stopping does not work well.



- "Co-teaching" between two neural nets:
  - Teach small-loss data each other. Han, Yao, Yu, Niu, Xu, Hu, Tsang & Sugiyama (NeurIPS2018)
  - Teach only disagreed data. Yu, Han, Yao, Niu, Tsang & Sugiyama (ICML2019)
  - Gradient ascent for large-loss data. Han, Niu, Yu, Yao, Xu, Tsang & Sugiyama (ICML2020)



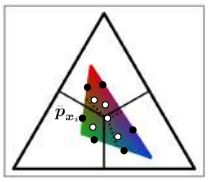
- No theory but very robust in experiments:
  - Works well even if 50% random label flipping!



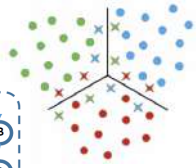
Summary

- Classification requires explicit treatment of label noise:
  - Loss correction by noise transition is promising.

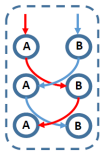
$$T_{y,\bar{y}} = \bar{p}(\bar{y}|y)$$



- However, noise transition is generally non-identifiable.
  - Recent development allows its consistent estimation under mild assumptions.
- Real-world noise is often instance-dependent:
  - Heuristic solutions have been developed.



- Super-robustness by co-teaching:
  - Heuristic solutions have been developed.



Low-Rank Tensor Representations of Quantum Circuits

Patrick Gelß

Zuse Institute Berlin, Germany
gelss@zib.de

Quantum computing is arguably one of the most revolutionary and disruptive technologies of this century. Due to the ever-increasing number of potential applications as well as the continuing rise in complexity, the development, simulation, optimization, and physical realization of quantum circuits is of utmost importance for designing novel algorithms. We show how matrix product states (MPSs) and matrix product operators (MPOs) can be used to express not only the state of the system but also quantum gates and entire quantum circuits as low-rank tensors. This allows us to analyze and simulate complex quantum circuits on classical computers and to gain insight into the underlying structure of the system. We present different examples to demonstrate the advantages of MPO formulations and provide a new perspective on the construction of quantum algorithms.

Low-rank tensor decompositions of quantum circuits

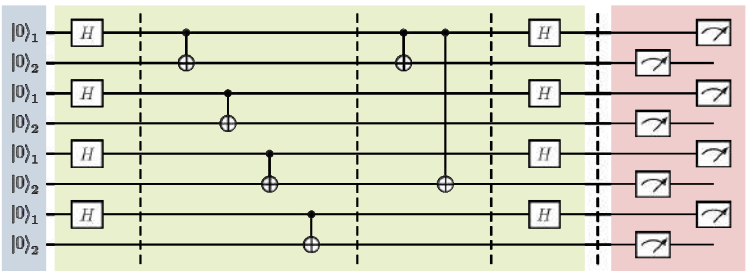
Patrick Gelß
joint work with Stefan Klus, Zarin Shakibaei, and Sebastian Pokutta

Zuse Institute Berlin
University of Surrey
Freie Universität Berlin
Technische Universität Berlin

6th RIKEN-IMI-ISM-NUS-ZIB-MODAL-NHR Workshop
Tokyo, Japan
September 17, 2022

Motivation

QUANTUM SIMULATION



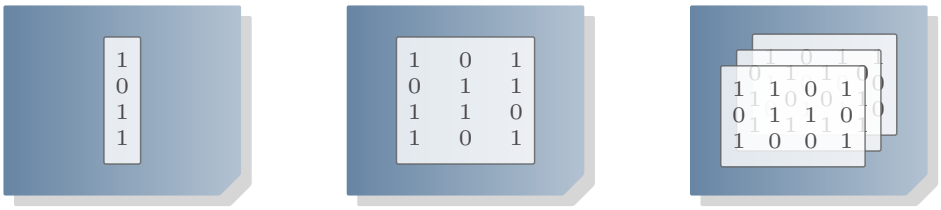
- tensor product representation already been used
- in particular, MPS/TT format
- reduce storage consumption and computational effort

- **AIM:** construct compact expressions of quantum circuits in form of MPOs
- MPS-MPO contraction to represent wave functions

- direct insight into network structure
- reduce CPU time for quantum simulations on classical computers

→ towards QOPT, QML ...

Tensors



- tensor: $\mathbf{T} \in \mathbb{C}^D = \mathbb{C}^{d_1 \times d_2 \times \cdots \times d_n}$
  - $(\mathbf{T} + \mathbf{U})_{x_1,\ldots,x_n} = \mathbf{T}_{x_1,\ldots,x_n} + \mathbf{U}_{x_1,\ldots,x_n}$
  - $(\alpha \cdot \mathbf{T})_{x_1,\ldots,x_n} = \alpha \cdot \mathbf{T}_{x_1,\ldots,x_n}$

$\mathbb{C}^D$ is the linear space of tensors with modes $d_1, \ldots, d_n$

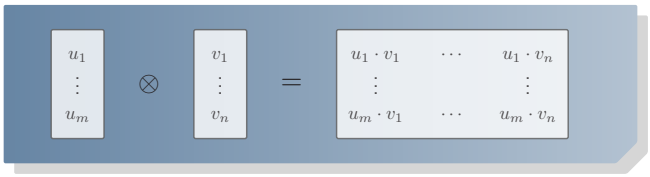
- tensor operator: $\mathbf{L} \in \mathbb{C}^{D \times D} = \mathbb{C}^{d_1 \times \cdots \times d_n \times d_1 \times \cdots \times d_n}$
  - $(\mathbf{L} \cdot \mathbf{T})_{x_1,\ldots,x_n} = \sum_{y_1,\ldots,y_n} \mathbf{L}_{x_1,\ldots x_n, y_1,\ldots,y_n} \cdot \mathbf{T}_{y_1,\ldots,y_n}$

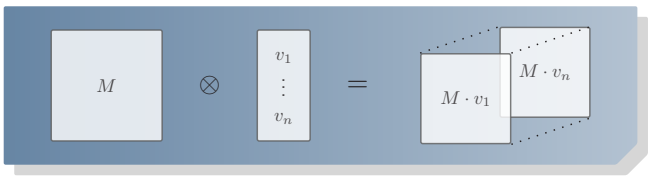
$\mathbb{C}^{D \times D}$ is the space of linear maps from $\mathbb{C}^D$ to $\mathbb{C}^D$

Tensor Product

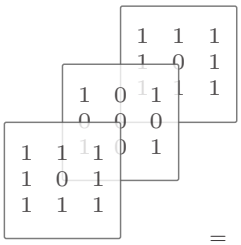
- given tensors $\mathbf{T} \in \mathbb{C}^{d_1 \times \cdots \times d_n}$ and $\mathbf{U} \in \mathbb{C}^{d'_1 \times \cdots \times d'_m}$, the **tensor product** $\mathbf{T} \otimes \mathbf{U}$ is defined by

$$(\mathbf{T} \otimes \mathbf{U})_{x_1,\ldots,x_n,y_1,\ldots,y_m} = \mathbf{T}_{x_1,\ldots,x_c} \cdot \mathbf{U}_{y_1,\ldots,y_d}$$





Tensor Product



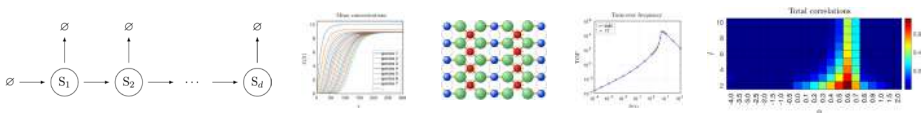
$$= \begin{bmatrix} 1 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 1 \end{bmatrix} \otimes \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix} + \begin{bmatrix} 1 & 0 & 1 \\ 0 & 0 & 0 \\ 1 & 0 & 1 \end{bmatrix} \otimes \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}$$

$$= \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} \otimes \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix} \otimes \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix} + \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix} \otimes \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} \otimes \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix} + \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix} \otimes \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix} \otimes \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}$$

$$= \left[\!\!\left[\begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix} \right]\!\!\right] \otimes \left[\!\!\left[\begin{matrix} \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix} & \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \\ \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} & \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix} \end{matrix} \right]\!\!\right] \otimes \left[\!\!\left[\begin{matrix} \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix} \\ \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} \end{matrix} \right]\!\!\right]$$

Matrix Product States aka Tensor Trains

chemical reaction networks and catalytic systems



system identification

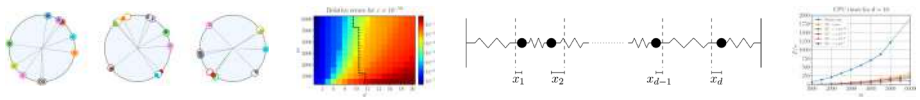
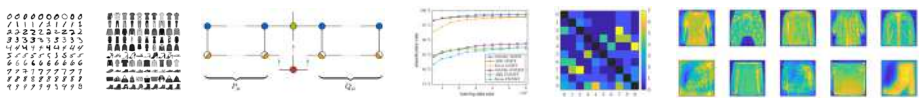


image classification



Matrix Product States aka Tensor Trains

Definition

Affleck, 1987 / Oseledets, 2009

A tensor $\mathbf{T} \in \mathbb{C}^{d_1 \times \cdots \times d_n}$ is said to be in the **MPS/TT format** if

$$\mathbf{T} = \sum_{k_0=1}^{r_0} \sum_{k_1=1}^{r_1} \cdots \sum_{k_d=1}^{r_n} \mathbf{T}^{(1)}_{k_0,:,k_1} \otimes \mathbf{T}^{(2)}_{k_1,:,k_2} \otimes \cdots \otimes \mathbf{T}^{(n)}_{k_{n-1},:,k_n}$$

The tensors $\mathbf{T}^{(1)}, \ldots, \mathbf{T}^{(n)}$ with $\mathbf{T}^{(i)} \in \mathbb{C}^{r_{i-1} \times d_i \times r_i}$ are called **cores** and the numbers r_i are **bond dimensions** or **ranks**. It holds that $r_0 = r_n = 1$ and $r_i \geq 1$ for $i = 1, \ldots, n-1$.

Idea: represent high-dimensional systems, e.g., quantum registers, in MPS/TT format in order to mitigate the curse of dimensionality

Matrix Product States aka Tensor Trains

- **explicit notation**

$$\mathbf{T} = \sum_{k_0=1}^{r_0} \sum_{k_1=1}^{r_1} \cdots \sum_{k_n=1}^{r_n} \mathbf{T}^{(1)}_{k_0,:,k_1} \otimes \mathbf{T}^{(2)}_{k_1,:,k_2} \otimes \cdots \otimes \mathbf{T}^{(n)}_{k_{n-1},:,k_n}$$

$$\Longleftrightarrow$$

$$\mathbf{T}_{x_1,x_2,\ldots,x_n} = \mathbf{T}^{(1)}_{:,x_1,:} \cdot \ldots \cdot \mathbf{T}^{(n)}_{:,x_n,:}$$

- **core notation**

$$\left[\!\left[\mathbf{T}^{(1)}_{1,:,1} \cdots \mathbf{T}^{(1)}_{1,:,r_1} \right]\!\right] \otimes \begin{bmatrix} \mathbf{T}^{(2)}_{1,:,1} & \cdots & \mathbf{T}^{(2)}_{1,:,r_2} \\ \vdots & & \vdots \\ \mathbf{T}^{(2)}_{r_1,:,1} & \cdots & \mathbf{T}^{(2)}_{r_1,:,r_2} \end{bmatrix} \otimes \cdots \otimes \begin{bmatrix} \mathbf{T}^{(n-1)}_{1,:,1} & \cdots & \mathbf{T}^{(n-1)}_{1,:,r_{n-1}} \\ \vdots & & \vdots \\ \mathbf{T}^{(n-1)}_{r_{n-2},:,1} & \cdots & \mathbf{T}^{(n-1)}_{r_{n-2},:,r_{n-1}} \end{bmatrix} \otimes \begin{bmatrix} \mathbf{T}^{(n)}_{1,:,1} \\ \vdots \\ \mathbf{T}^{(n)}_{r_{n-1},:,1} \end{bmatrix}$$

Matrix Product States aka Tensor Trains

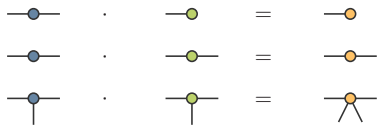
- **graphical notation**
  - depict a tensor $\mathbf{T} \in \mathbb{R}^{d_1 \times \cdots \times d_n}$ as a circle with n arms:

  - tensor contraction:

$matrix \cdot vector$:

$matrix \cdot matrix$:

$TT\ core \cdot TT\ core$:



  - tensor-train representation:

Matrix Product States aka Tensor Trains

- **conversion into TT format**

Initial tensor

Isolate first mode

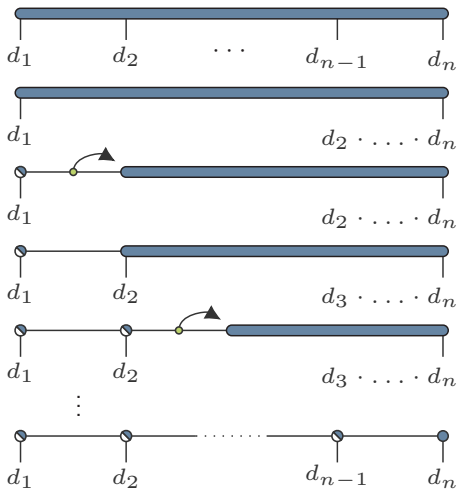
Apply SVD ($M = U\Sigma V^T$)

Isolate second mode

Apply SVD

⋮

Tensor-train approximation

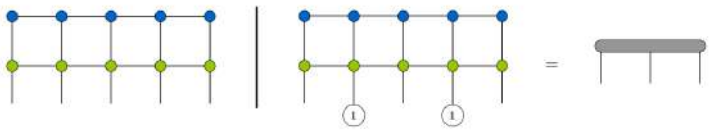


Matrix Product States aka Tensor Trains

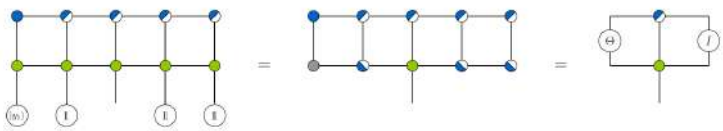
- **examples**
  - $|\Phi^{\pm}\rangle = \frac{1}{\sqrt{2}}(|00\rangle \pm |11\rangle) = \frac{1}{\sqrt{2}} \begin{bmatrix} |0\rangle & \pm|1\rangle \end{bmatrix} \otimes \begin{bmatrix} |0\rangle \\ |1\rangle \end{bmatrix}$
  - $|\Psi^{\pm}\rangle = \frac{1}{\sqrt{2}}(|01\rangle \pm |10\rangle) = \frac{1}{\sqrt{2}} \begin{bmatrix} |0\rangle & \pm|1\rangle \end{bmatrix} \otimes \begin{bmatrix} |1\rangle \\ |0\rangle \end{bmatrix}$
  - $|\mathrm{GHZ}\rangle = \frac{1}{\sqrt{2}}(|0\ldots0\rangle + |1\ldots1\rangle)$
    $= \frac{1}{\sqrt{2}} \begin{bmatrix} |0\rangle & |1\rangle \end{bmatrix} \otimes \begin{bmatrix} |0\rangle & \\ & |1\rangle \end{bmatrix} \otimes \cdots \otimes \begin{bmatrix} |0\rangle & \\ & |1\rangle \end{bmatrix} \otimes \begin{bmatrix} |1\rangle \\ |0\rangle \end{bmatrix}$
  - $|\mathrm{W}\rangle = \frac{1}{\sqrt{n}}(|10\ldots0\rangle + \cdots + |0\ldots01\rangle)$
    $= \frac{1}{\sqrt{n}} \begin{bmatrix} |1\rangle & |0\rangle \end{bmatrix} \otimes \begin{bmatrix} |0\rangle & \\ |1\rangle & |0\rangle \end{bmatrix} \otimes \cdots \otimes \begin{bmatrix} |0\rangle & \\ |1\rangle & |0\rangle \end{bmatrix} \otimes \begin{bmatrix} |0\rangle \\ |1\rangle \end{bmatrix}$

Matrix Product States aka Tensor Trains

- **generative sampling** — Ferris et al., 2012 / Han et al., 2018
  - suppose the wave function is given in form of an MPS $\mathbf{\Psi} \in \mathbb{C}^{2^{\times n}}$, then it is possible to directly sample from the probability distribution given by $\mathbf{P}_{x_1,\ldots,x_n} = \left|\mathbf{\Psi}_{x_1,\ldots,x_n}\right|^2 / Z$
  - $\mathbf{P}_{x_1,\ldots,x_n} = \overline{\mathbf{\Psi}_{x_1,\ldots,x_n}}\mathbf{\Psi}_{x_1,\ldots,x_n} \iff \mathbf{P} = \mathrm{diag}(\overline{\mathbf{\Psi}})\mathbf{\Psi}$

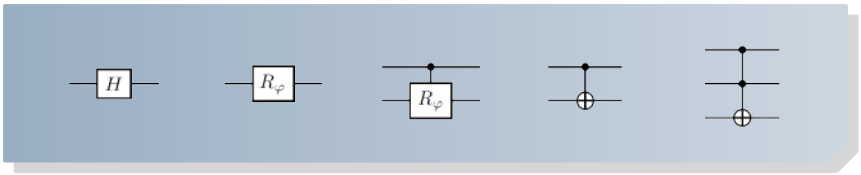


  - conditional probabilities for sampling: $P^{i_k} = \sum_{x_{i_{k+1}},\ldots,x_{i_m}} \mathbf{P}^{I}_{y_{i_1},\ldots,y_{i_{k-1}},:,x_{i_{k+1}},\ldots,x_{i_m}}$



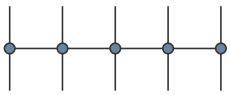
Quantum Circuits in MPO Format

- **Q:** How to represent quantum gates and circuits?



- **A:** Express them as matrix product operators.

$$\mathbf{T}^{(i)} \in \mathbb{C}^{r_{i-1}\times d_i \times r_i}$$
$$\downarrow$$
$$\mathbf{G}^{(i)} \in \mathbb{C}^{r_{i-1}\times d_i \times d_i \times r_i}$$



$$\begin{bmatrix} \mathbf{G}^{(i)}_{1,:,:,1} & \cdots & \mathbf{G}^{(i)}_{1,:,:,r_i} \\ \vdots & & \vdots \\ \mathbf{G}^{(i)}_{r_{i-1},:,:,1} & \cdots & \mathbf{G}^{(i)}_{r_{i-1},:,:,r_i} \end{bmatrix}$$

Quantum Circuits in MPO Format

- **MPO representations of controlled gates**
  - controlled NOT gate (CNOT):

$$\mathrm{CNOT} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix} \cong \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} \otimes I + \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix} \otimes \sigma_x = I^{\otimes 2} + C \otimes (\sigma_x - I) = \begin{bmatrix} I & C \end{bmatrix} \otimes \begin{bmatrix} I \\ \sigma_x - I \end{bmatrix}$$

  - controlled-controlled NOT gate (CCNOT):

$$\mathrm{CCNOT} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix} \cong I^{\otimes 3} + C \otimes C \otimes (\sigma_x - I) = \begin{bmatrix} I & C \end{bmatrix} \otimes \begin{bmatrix} I & \\ & C \end{bmatrix} \otimes \begin{bmatrix} I \\ \sigma_x - I \end{bmatrix}$$

Quantum Circuits in MPO Format

- **MPO representations for multi-qubit systems**
  - single-qubit gates (acting on n-qubit system):

$$\mathbf{G} = I^{\otimes(p-1)} \otimes A \otimes I^{\otimes(n-p)} = I \otimes \cdots \otimes I \otimes \underbrace{A}_{\text{position } p} \otimes I \otimes \cdots \otimes I$$

  - controlled gates (e.g., CPHASE, CNOT):

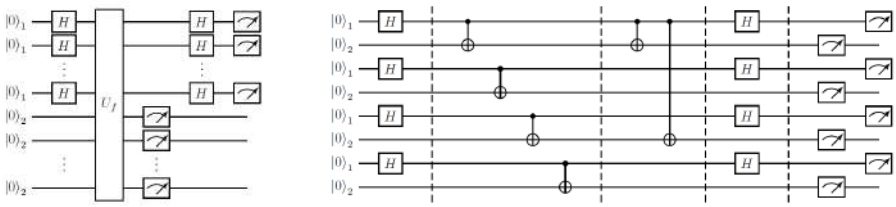
$$\begin{aligned}\mathbf{G} &= I^{\otimes n} + I^{\otimes(p-1)} \otimes C \otimes I^{\otimes(q-p-1)} \otimes (A - I) \otimes I^{\otimes(n-q)} \\ &= [\![I]\!] \otimes \cdots \otimes [\![I]\!] \otimes \underbrace{[\![I \quad C]\!]}_{\text{position } p} \otimes \begin{bmatrix} I & \\ & I \end{bmatrix} \otimes \cdots \otimes \begin{bmatrix} I & \\ & I \end{bmatrix} \otimes \underbrace{\begin{bmatrix} I \\ A - I \end{bmatrix}}_{\text{position } q} \otimes [\![I]\!] \otimes \cdots \otimes [\![I]\!]\end{aligned}$$

  - controlled-controlled gates (e.g., CCNOT):

$$\begin{aligned}\mathbf{G} = {} & [\![I]\!] \otimes \cdots \otimes [\![I]\!] \otimes \underbrace{[\![I \quad C]\!]}_{\text{position } p_1} \otimes \begin{bmatrix} I & \\ & I \end{bmatrix} \otimes \cdots \otimes \begin{bmatrix} I & \\ & I \end{bmatrix} \otimes \underbrace{\begin{bmatrix} I & \\ & C \end{bmatrix}}_{\text{position } p_2} \otimes \cdots \\ & \otimes \begin{bmatrix} I & \\ & I \end{bmatrix} \otimes \cdots \otimes \begin{bmatrix} I & \\ & I \end{bmatrix} \otimes \underbrace{\begin{bmatrix} I \\ A - I \end{bmatrix}}_{\text{position } q} \otimes [\![I]\!] \otimes \cdots \otimes [\![I]\!],\end{aligned}$$

Quantum Circuits in MPO Format

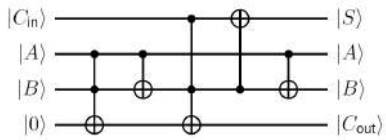
- **Simon's algorithm**
  - determine hidden bitstring b with $f(x) = f(y) \Leftrightarrow x = y \oplus b$



  - $\mathbf{G} = [\![A \quad B]\!] \otimes \begin{bmatrix} I & \\ & I \end{bmatrix} \otimes \begin{bmatrix} A & B & & \\ & & A & B \end{bmatrix} \otimes \begin{bmatrix} I & & & \\ \sigma_x & & & \\ & & I & \\ & & & \sigma_x \end{bmatrix} \otimes \begin{bmatrix} A & B \\ B & A \end{bmatrix} \otimes \begin{bmatrix} I \\ \sigma_x \end{bmatrix} \otimes [\![A \quad B]\!] \otimes \begin{bmatrix} I \\ \sigma_x \end{bmatrix}$ with $A = \frac{1}{2}\begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}$ and $B = \frac{1}{2}\begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}$
  - $\mathbf{G}|0\rangle = \frac{1}{4} [\![|+\rangle \quad |-\rangle]\!] \otimes \begin{bmatrix} |0\rangle & \\ & |0\rangle \end{bmatrix} \otimes \begin{bmatrix} |+\rangle & |-\rangle & & \\ & & |+\rangle & |-\rangle \end{bmatrix} \otimes \begin{bmatrix} |0\rangle & & & \\ |1\rangle & & & \\ & & |0\rangle & \\ & & |1\rangle & \end{bmatrix} \otimes \begin{bmatrix} |+\rangle & |-\rangle \\ |-\rangle & |+\rangle \end{bmatrix} \otimes \begin{bmatrix} |0\rangle \\ |1\rangle \end{bmatrix} \otimes [\![|+\rangle \quad |-\rangle]\!] \otimes \begin{bmatrix} |0\rangle \\ |1\rangle \end{bmatrix} \quad \Rightarrow \quad \underline{b = 1010}$

Quantum Circuits in MPO Format

- **Quantum full adder**
  - adds input qubits $|A\rangle$, $|B\rangle$, $|C_{\text{in}}\rangle$, produces sum $|S\rangle$ and carry-out qubit $|C_{\text{out}}\rangle$



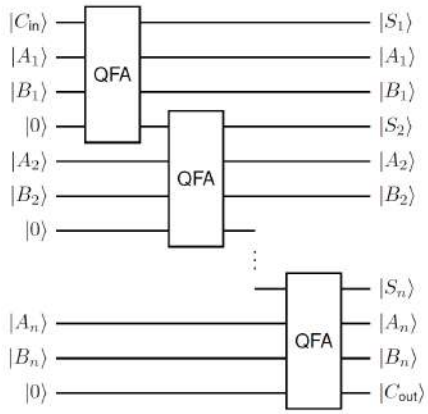
  - $\mathbf{G} = \begin{bmatrix} \sigma_x C_0 & I & \sigma_x C_1 \end{bmatrix} \otimes \begin{bmatrix} C_0 & C_1 & 0 & 0 \\ 0 & C_0 & C_1 & 0 \\ 0 & 0 & C_0 & C_1 \end{bmatrix} \otimes \begin{bmatrix} C_1 & 0 \\ C_0 & 0 \\ 0 & C_1 \\ 0 & C_0 \end{bmatrix} \otimes \begin{bmatrix} I \\ \sigma_x \end{bmatrix}$

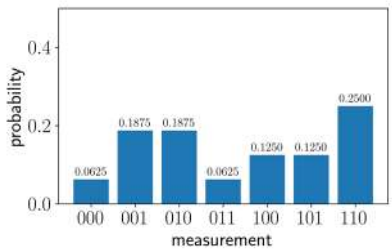
    with $C_0 = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}$ and $C_1 = C = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}$

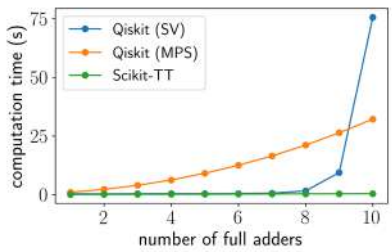
Numerical Results

scikit tt
Tensor-train computations in Python

- open-source tensor-train library for Python using NumPy and SciPy
- simulation and analysis of systems with high-dimensional state spaces
- reduce memory consumption and computational costs
- includes model building, tensor-based solvers, and data-driven methods
- possible application areas: Markovian master equations, nearest-neighbor interactions, nonlinear dynamical systems, quantum simulation, etc.







NUMERICAL RESULTS

- **Quantum full adder network**

- construct network of quantum full adders by concatenating MPOs:

$$\begin{aligned}\mathbf{G} = {} & \left[\!\left[\mathbf{G}^{(1)}_{\text{QFA}}\right]\!\right] \otimes \left[\!\left[\mathbf{G}^{(2)}_{\text{QFA}}\right]\!\right] \otimes \left[\!\left[\mathbf{G}^{(3)}_{\text{QFA}}\right]\!\right] \otimes \left[\!\left[\mathbf{G}^{(4)}_{\text{QFA}}\right]\!\right] \otimes [\![I]\!] \otimes \cdots \otimes [\![I]\!] \\ & \cdot [\![I]\!] \otimes [\![I]\!] \otimes [\![I]\!] \otimes \left[\!\left[\mathbf{G}^{(1)}_{\text{QFA}}\right]\!\right] \otimes \left[\!\left[\mathbf{G}^{(2)}_{\text{QFA}}\right]\!\right] \otimes \left[\!\left[\mathbf{G}^{(3)}_{\text{QFA}}\right]\!\right] \otimes \left[\!\left[\mathbf{G}^{(4)}_{\text{QFA}}\right]\!\right] \otimes [\![I]\!] \otimes \cdots \otimes [\![I]\!] \\ & \cdot \ldots \\ & \cdot [\![I]\!] \otimes \cdots \otimes [\![I]\!] \otimes \left[\!\left[\mathbf{G}^{(1)}_{\text{QFA}}\right]\!\right] \otimes \left[\!\left[\mathbf{G}^{(2)}_{\text{QFA}}\right]\!\right] \otimes \left[\!\left[\mathbf{G}^{(3)}_{\text{QFA}}\right]\!\right] \otimes \left[\!\left[\mathbf{G}^{(4)}_{\text{QFA}}\right]\!\right]\end{aligned}$$

- define initial quantum state:

$$\boldsymbol{\Psi} = \begin{bmatrix}1\\0\end{bmatrix} \otimes \begin{bmatrix}\frac{1}{\sqrt{2}}\\\frac{1}{\sqrt{2}}\end{bmatrix} \otimes \begin{bmatrix}\frac{1}{\sqrt{2}}\\\frac{1}{\sqrt{2}}\end{bmatrix} \otimes \begin{bmatrix}1\\0\end{bmatrix} \otimes \begin{bmatrix}\frac{1}{\sqrt{2}}\\\frac{1}{\sqrt{2}}\end{bmatrix} \otimes \begin{bmatrix}\frac{1}{\sqrt{2}}\\\frac{1}{\sqrt{2}}\end{bmatrix} \otimes \begin{bmatrix}1\\0\end{bmatrix} \otimes \begin{bmatrix}\frac{1}{\sqrt{2}}\\\frac{1}{\sqrt{2}}\end{bmatrix} \otimes \begin{bmatrix}\frac{1}{\sqrt{2}}\\\frac{1}{\sqrt{2}}\end{bmatrix} \otimes \ldots$$

- compute final probability distribution of $|S_1, \ldots, S_n, C_{\text{out}}\rangle$ by generative sampling on MPSs

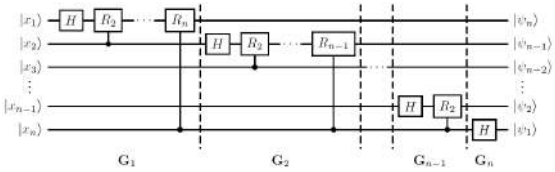
---

NUMERICAL RESULTS

- compute 10^6 samples of the output for different numbers of QFAs in the network
- same distributions as obtained with Qiskit
- CPU time of MPO-based approach depends only linearly on the number of QFAs
- for instance, for $n_{\text{QFA}} = 100$, the MPO simulation needs about $30\,$s only

Numerical Results

- **Quantum Fourier transform**
  - map quantum states between computational and Fourier basis



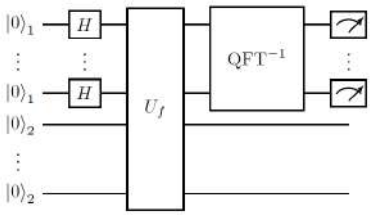
$$\mathbf{G}_i = \frac{1}{\sqrt{2}} \left[\!\left[I^{\otimes(i-1)} \right]\!\right] \otimes \left[\!\left[\begin{bmatrix} 1 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} 0 & 0 \\ 1 & -1 \end{bmatrix} \right]\!\right]$$

$$\otimes \begin{bmatrix} I & 0 \\ 0 & R_2 \end{bmatrix} \otimes \cdots \otimes \begin{bmatrix} I & 0 \\ 0 & R_{n-i} \end{bmatrix} \otimes \left[\!\left[\begin{matrix} I \\ R_{n-i+1} \end{matrix} \right]\!\right]$$

		$n = 16$	$n = 32$	$n = 64$	$n = 128$
Qiskit	$s = 10^2$	0.07 ± 0.08	0.12 ± 0.02	0.33 ± 0.05	1.09 ± 0.11
	$s = 10^4$	0.16 ± 0.01	0.27 ± 0.03	0.57 ± 0.05	1.57 ± 0.10
	$s = 10^6$	6.65 ± 0.03	15.12 ± 0.05	25.72 ± 0.10	48.35 ± 0.11
Scikit-TT	$s = 10^2$	0.04 ± 0.00	0.15 ± 0.00	0.59 ± 0.00	2.31 ± 0.01
	$s = 10^4$	0.07 ± 0.00	0.20 ± 0.00	0.66 ± 0.00	2.43 ± 0.01
	$s = 10^6$	5.34 ± 0.15	7.82 ± 0.18	11.51 ± 0.16	20.39 ± 0.20

Numerical Results

- **Shor's algorithm**



- given $M \in \mathbb{N}$, choose $1 < a < M$ (coprime)
- initialize input register with $2n$ qubits and target register with n qubits, where $2^n > M$
- apply Hadamard gates and modular exponentiation circuit U_f with $f(x) = a^x \bmod M$
- use inverse QFT to calculate period of f and find factors of M

- consider $M = 15$, $a \in \{2, 4, 7, 8, 11, 13, 14\}$:
  - U_f can be constructed as MPO with ranks bounded by either 2 or 4
  - orthonormalize MPO cores between applications of QFT^{-1} gate groups

a	y	p	(M_1, M_2)
$2, 7, 8, 13$	0	1	$\varnothing$
	64	4	$(3, 5)$
	128	2	$(3, 1)$
	192	4	$(3, 5)$
$4, 11, 14$	0	1	$\varnothing$
	128	2	$(3, 1)$

Thanks for your attention

Links

Scikit-TT on GitHub: http://github.com/PGelss/scikit_tt

Collaborations

Dr. Stefan Klus, *Department of Mathematics, University of Surrey*

Zarin Shakibaei, *Telekom Innovation Laboratories, TU Berlin*

Prof. Sebastian Pokutta, *AI in Society, Science, and Technology, ZIB*

Publications

- P. Gelß, S. Klus, Z. Shakibaei, S. Pokutta. *Low-rank tensor decompositions of quantum circuits*, in submission
- F. Nüske, P. Gelß, S. Klus, C. Clementi. *Tensor-based computation of metastable and coherent sets*, Physica D, 2021
- P. Gelß, S. Klus, J. Eisert, C. Schütte. *Multidimensional approximation of nonlinear dynamical systems*, J. Comput. Nonlinear Dynam., 2019
- S. Klus, P. Gelß, S. Peitz, C. Schütte. *Tensor-based dynamic mode decomposition*, Nonlinearity, 2018
- P. Gelß, S. Klus, S. Matera, C. Schütte. *Nearest-neighbor interaction systems in the tensor-train format*, J. Comput. Phys., 2017

NEC's Quantum Computing Technologies

Shintaro MOMOSE

Quantum Computing Business Department

Advanced Platform Division
NEC Corporation
s-momoseak@nec.com

This presentation consists of two parts, discussing SX-Aurora TSUBASA vector supercomputer and introducing digital annealer working on SX-Aurora TSUBASA called Aurora Vector Annealer. The first half of the presentation shows the vector architecture of SX-Aurora TSUBASA, especially its latest vector processors having the highest-level memory bandwidth. Sustained performance and power efficiency are also discussed, as well as NEC's future plans and roadmap. The second half of the presentation shows NEC's quantum computing strategies and their products to provide higher sustained performance in the annealing/optimization fields. NEC developed the Aurora Vector Annealer as a digital annealer and has a strong business relationship with D-Wave providing a quantum annealer. NEC aims at solving various social issues by using the quantum/digital annealing technologies and by developing a hybrid platform with supercomputer and quantum/digital annealer to provide much higher sustained performance.

\Orchestrating a brighter world NEC

The 6th RIKEN-IMI-ISM-NUS-ZIB-MODAL-NHR Workshop
September 17th, 2022. Tokyo Japan

NEC's Quantum Computing Technologies

Shintaro MOMOSE, Ph.D. (Director)

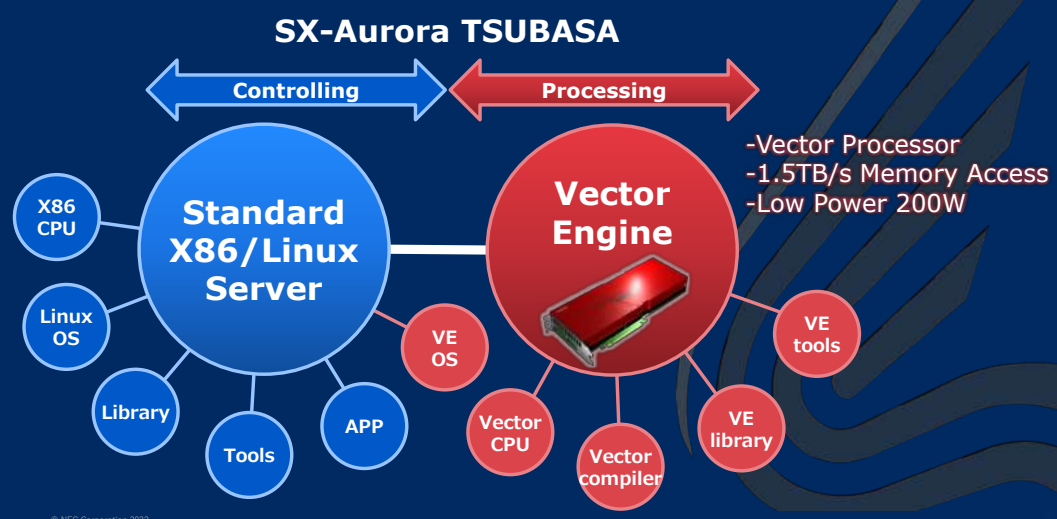
Quantum Computing Business Department
Advanced Platform Division
NEC Corporation

Contents

- NEC's Strategy for Quantum Computing
- Vector Annealing on SX-Aurora TSUBASA
- Case Study

\Orchestrating a brighter world NEC

Architecture of SX-Aurora TSUBASA

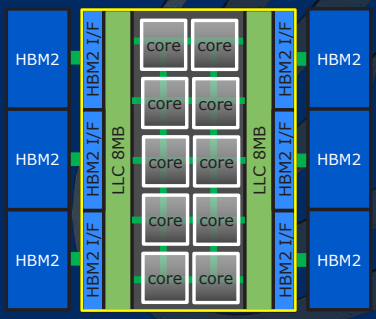


-Vector Processor
-1.5TB/s Memory Access
-Low Power 200W

VE20 Processor

VE20 Specifications

Processor Version	Type 20A	Type 20B
Cores/processor	10	8
Core performance	307GF (DP) 614GF (SP)	
Processor performance	3.07TF (DP) 6.14TF (SP)	2.45TF (DP) 4.91TF (SP)
Cache capacity	16MB	
Cache bandwidth	3TB/s	
Cache Function	Software Controllable	
Memory capacity	48GB	
Memory bandwidth	1.53TB/s	
Power	~300W (TDP) ~200W (Application)	



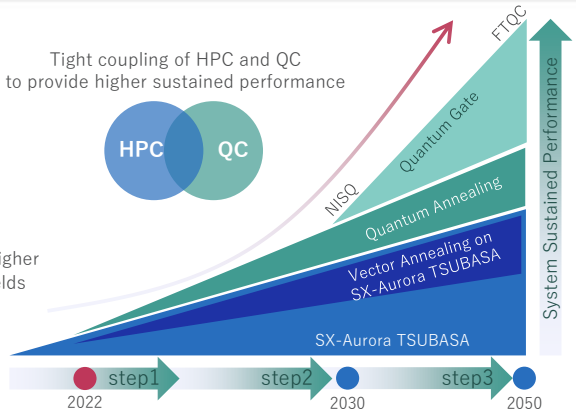
Why is NEC Focusing on Quantum Computing?

- Both HPC and Quantum technologies will be used for higher sustained performance
- NEC develops HPC, Simulated Annealing on Aurora, Quantum Annealer and Quantum Gate

For higher sustained performance, NEC continuously combine HPC and new cutting edge technologies

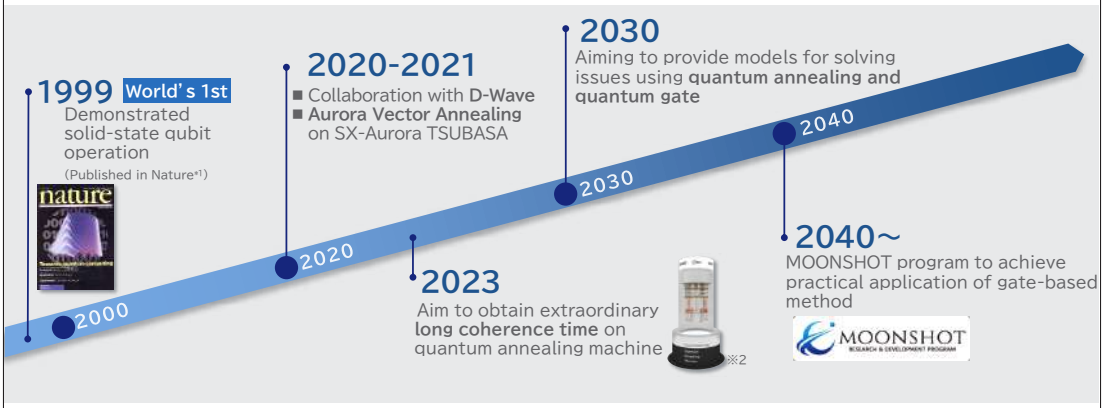
- **Step1: Annealing on HPC resource**
  - Vector Annealing on SX-Aurora TSUBASA
  - Using Quantum Annealer to accelerate
- **Step2: HPC/QC Hybrid Computing**
  - Tight coupled HPC/QC hybrid system to reach higher sustained performance in scientific/industrial fields
- **Step3: Introducing "QC Gate" as new era**
  - NISC type: Around 2030
  - FTQC type: Around 2050

NISQ
Noisy Intermediate-Scale QC
FTQC
Fault Tolerant QC



NEC's Initiative in Quantum Computing

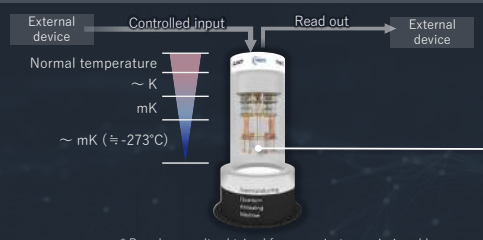
Since succeeding in the world's first demonstration of solid-state qubit operation, NEC has been working towards the social implementation of quantum computing.



*1: Y. Nakamura et al., Nature 398, 786 (1999)
*2: Based on results obtained from a project commissioned by the New Energy and Industrial Technology Development Organization (NEDO).

NEC is leading the development of quantum annealing devices to enable practical use of superconducting quantum annealing machine in 2023

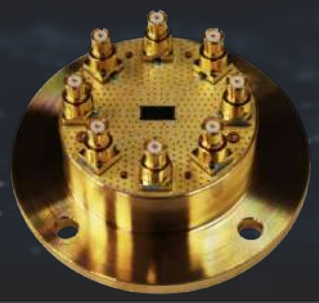
Superconducting quantum annealing machine (mockup)



* Based on results obtained from a project commissioned by the New Energy and Industrial Technology Development Organization (NEDO).

Support	Participating organizations
NEDO	NEC (Deputy rep. facilities responsible for operationalization and practical application) AIST (Rep.), Tokyo Institute of Technology, Waseda Univ., Yokohama National Univ., and others

Quantum annealing device that NEC has been working on
(The heart part of the machine)



In addition to focusing on the quantum annealing method to address society's optimization needs, NEC is also promoting research and development toward practical application of the gate-based method.

Quantum Computing

(Broadly defined to include quantum behavior)

Annealing method, etc.
Solves combinatorial optimization problems by means of the Ising model or other statistical physics model

- Quantum Annealing Superconducting Circuits
  - D-Wave (product)
  - NEC (R&D)
  - AIST
- Digital Circuits
  - NEC (product)
  - Hitachi
  - Fujitsu
  - Toshiba
- Optical Parametric Oscillator
  - NTT

Quantum gate method
Performs calculations by replacing classical computer bits with qubits

- NEC (R&D)
- RIKEN UTokyo
- IBM
- Google
- Rigetti
- Intel

*Based on NEC's survey. (Due to limited space, not all institutions are covered.)

Vector Annealing on SX-Aurora TSUBASA

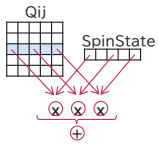
\Orchestrating a brighter world NEC

NEC Vector Annealing

VA Performance is provided by:
Matrix operation acceleration by VE, large and fast memory, and optimized algorithm for VE

Vector operation on VE

Energy calculation is matrix operation



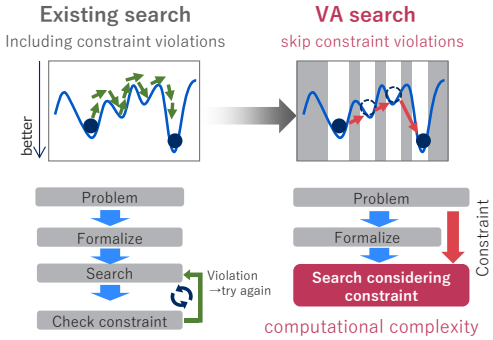

```
for(i=0; i<numSpins; i++)
  DeltaEnergy +=
      Qij[FlipSpin][i] x SpinState[i]
```

Effect from neighbor spin | 0 or 1

Full connect 100k qbits/VE and high memory bandwidth

- 48GB memory capacity and 1.5TB/s memory bandwidth
- Multi card supports larger number of qbits (100k qbits x n)$^{1/2}$

Avoiding Redundant Search and Optimized algorithm for VE



computational complexity reduction

\Orchestrating a brighter world NEC

Case Study

\Orchestrating a brighter world NEC

Solving Social Issues Using Quantum Computing

NEC is trying to apply QC technologies for practical use with partners

Development with Co-creation Partners

SMBC Group/ JRI / NEC Platforms / NEC Fielding etc.

Advertisement Infrastructure	Manufacturing	Traffic/Logistics	Financial	Material/Drug
• Matching/ Recommendation • Com. base station • Surveillance sensor	• Production plan • Parts ordering plan	• Crew shift • Delivery plan • Load placement	• Card fraud detection • Monte Carlo simulation • Risk calculation	• Screening • Experimental • parameter search

※研究中、顧客実証～実用に至るものが含まれています

Leap Quantum Cloud Service

NEC Vector Annealing Service

17

\Orchestrating a brighter world NEC

Real Time TSUNAMI Disaster Simulation & Real Time Proposing Optimal Evacuation Routes

"Next Generation Supercomputing Platform assisted by Quantum Annealing"
R&D with Tohoku University

Evacuation Route (Annealing)

Tohoku University

Information Science
Prof. Kobayashi, Prof. Ozeki

Disaster Science
Prof. Koshimura
Associate Prof. Erick

Riken

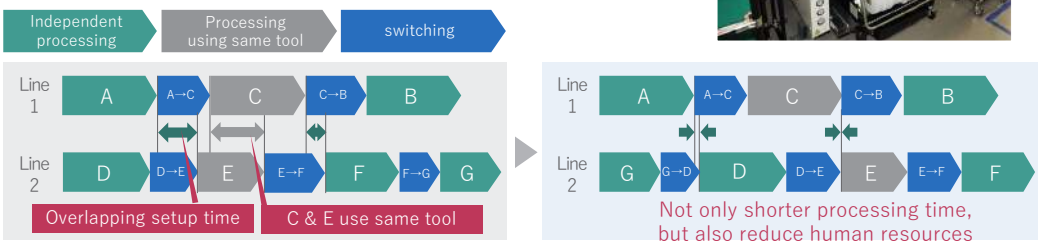
Associate Prof. Ota

Use Case: **Production Planning Optimization**

Optimizing complex planning for multi-product manufacturing lines

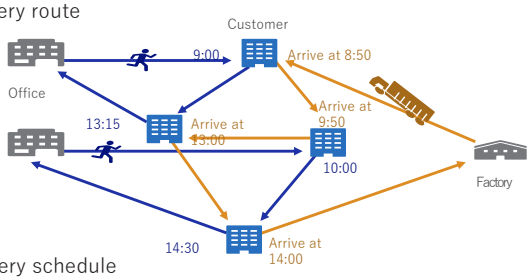
Higher versatile processing equipment needs
highly optimized product planning for higher efficiency

- Switching processed product makes idling time of equipment
- Production planning can reduce the idling time and also reduce human resources



Use Case: **Delivery Route and Schedule Optimization**
for reducing costs, time, energy, CO_2, etc.

delivery route



delivery schedule

EX. Delivery of parts and dispatch of Engineers

- Parts are delivered by truck
- Engineers move by car/train
- Have to consider skills of each engineer

Pairwise Conditional Gradients without Swap Steps and Sparser Kernel Herding

Kazuma Tsuji

MUFG Bank, Ltd., Tokyo, Japan
takahashi.shota@ism.ac.jp

The Pairwise Conditional Gradients (PCG) algorithm [1] is a powerful extension of the Frank-Wolfe algorithm leading to particularly sparse solutions, which makes PCG very appealing for problems such as sparse signal recovery, sparse regression, and kernel herding. Unfortunately, PCG exhibits so-called swap steps that might not provide sufficient primal progress. The number of these bad steps is bounded by a function in the dimension and as such known guarantees do not generalize to the infinite-dimensional case, which would be needed for kernel herding. We propose a new variant of PCG, the so-called Blended Pairwise Conditional Gradients (BPCG) which is a combination of Blended Conditional Gradients [2] and PCG, and BPCG does not exhibit swap steps. The convergence rate of BPCG is basically that of PCG if no drop steps would occur and as such is no worse than PCG but improves and provides new rates in many cases. Moreover, we observe in the numerical experiments that BPCG's solutions are much sparser than those of PCG. We apply BPCG to the kernel herding setting, where we derive nice quadrature rules and provide numerical results demonstrating the performance of our method.

References

[1] S. Lacoste-Julien and M. Jaggi. On the global linear convergence of Frank-Wolfe optimization variants. In C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, editors, *Advances in Neural Information Processing Systems 28*, pages 496–504. Curran Associates, Inc., 2015. URL http://papers.nips.cc/paper/ 5925-on-the-global-linear-convergence-of-frank-wolfe-optimi pdf.

[2] G. Braun, S. Pokutta, D. Tu, and S. Wright. Blended conditional gradients: the unconditioning of conditional gradients. In *Proceedings of the 36th International Conference on Machine Learning (PMLR)*, volume 97, pages 735–743, 2019.

Pairwise Conditional Gradients without Swap Steps and Sparser Kernel Herding

Kazuma Tsuji (MUFG bank)

Coresearchers:
Ken'ichiro Tanaka (The University of Tokyo, PRESTO)
Sebastian Pokutta (AISST, ZIB)

2022/09/17

Outline of today's talk

- The main topic of today's talk is Conditional Gradients methods.
- We propose a new variant of Conditional Gradients which is called Blended Pairwise Conditional Gradients (BPCG).
- BPCG algorithm works well in high dimensional cases and outputs highly sparse solutions practically.

The contents of today's talk are written in Tsuji et al. (2022) in detail.

CG algorithm

Conditional Gradients (Levitin and Polyak, 1966) are in an important class of first-order methods for constrained convex minimization, i.e., solving

$$\min_{x \in C} f(x) \quad (f : \text{convex}, C \subset \mathbb{R}^d : \text{convex compact region}).$$

- CG algorithm is an iterative first-order method.
- The solution of CG algorithm is represented as a convex combination of the vertices of C:

$$\xi_t = \sum_{i=1}^{n} c_i v_i \quad \left(\{v_i\}_{i=1}^{n} \subset V_C, C = \mathrm{conv}(V_C), \sum_{i=1}^{n} c_i = 1 \right)$$

.

Algorithm

1. $w_i = \mathrm{argmax}_{v \in V_C} \langle -\nabla f(\xi_i), v \rangle \quad (C = \mathrm{conv}(V_C))$
2. determine the step-size α_i $(0 \leq \alpha_i \leq 1)$
3. $\xi_{i+1} = \xi_i + \alpha_i(w_i - \xi_i) = (1 - \alpha_i)\xi_i + \alpha_i w_i$

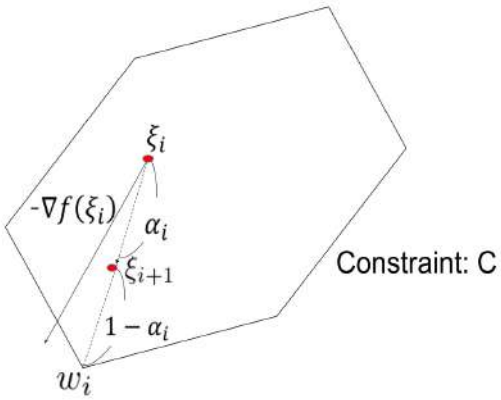


Figure: CG Algorithm

Variants of CG algorithm

There are many variants of CG algorithm to achieve better performance (faster convergence, computational efficiency, sparser solutions, etc.).

We explain the following two variants of CG algorithm:

- Pairwise CG algorithm (Lacoste-Julien and Jaggi, 2015)
- Blended CG algorithm (Braun et al., 2019)

Pairwise CG method

Pairwise CG method (Lacoste-Julien and Jaggi, 2015) uses the direction $w_t - a_t$ instead of $w_t - \xi_t$ for the update of current solutions:

$$d_t = \underbrace{(w_t - \xi_t)}_{\text{FW}} + \underbrace{(\xi_t - a_t)}_{\text{Away}} = w_t - a_t \text{ (Pairwise direction)}$$

$$\xi_{t+1} = \xi_t + \alpha_t d_t$$

$$\left(\begin{array}{l} w_t = \mathrm{argmax}_{v \in V_C} \langle v, -\nabla f(\xi_t) \rangle \\ a_t = \mathrm{argmin}_{v \in S_t} \langle v, -\nabla f(\xi_t) \rangle \ (\xi_t \in \mathrm{conv}(S_t) \subset V_C) \\ S_t : \text{vertices set that construct the convex combination of } \xi_t \end{array} \right)$$

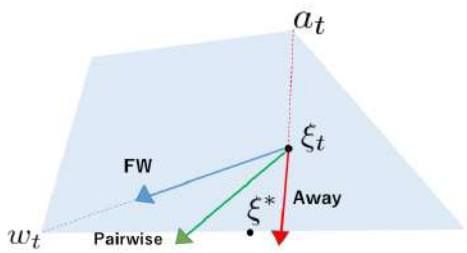


Figure: Pairwise CG

Blended CG

Blended Conditional Gradients (Braun, Pokutta, Tu, and Wright, 2019)

$\xi_t = \sum_{i=1}^k c_i v_i$ ($\sum_{i=1}^k c_i = 1, c_1, \ldots, c_k \geq 0, S_t = \{v_i\}_{i=1}^k \subset V_C$).
$a_t \leftarrow \mathrm{argmin}_{v \in S_t} \langle -\nabla f(\xi_t), v \rangle$, $s_t \leftarrow \mathrm{argmax}_{v \in S_t} \langle -\nabla f(\xi_t), v \rangle$
$w_t \leftarrow \mathrm{argmax}_{v \in V_C} \langle -\nabla f(\xi_t), v \rangle$

$A = \langle \nabla f(\xi_t), a_t - s_t \rangle$ (local pairwise gap) : upper bound of local error $f(\xi_t) - \min_{x \in \mathrm{conv}(S_t)} f(x)$
$B = \langle \nabla f(\xi_t), \xi_t - w_t \rangle$ (dual gap) : upper bound of global error $f(\xi_t) - \min_{x \in C} f(x)$

Algorithm
1. $A \geq B$
optimize the convex coefficients $\{c_i\}_{i=1}^k$ by SiGD which is an optimization method on a simplex.
2. $A < B$
$\xi_{t+1} = \xi_t + \alpha_t(w_t - \xi_t), S_{t+1} \leftarrow S_t \cup \{w_t\}$ (vanilla CG update).

Algorithm Blended Conditional Gradients

for $t = 0$ to $T - 1$ **do**
  $a_t \leftarrow \mathrm{argmin}_{v \in S_t} \langle -\nabla f(\xi_t), v \rangle$
  $s_t \leftarrow \mathrm{argmax}_{v \in S_t} \langle -\nabla f(\xi_t), v \rangle$
  $w_t \leftarrow \mathrm{argmax}_{v \in V_C} \langle -\nabla f(\xi_t), v \rangle$
  if $\langle \nabla f(\xi_t), a_t - s_t \rangle \geq \langle \nabla f(\xi_t), \xi_t - w_t \rangle$ **then**
    optimize the convex coefficients $\{c_i\}_{i=1}^k$.
  else
    $\xi_{t+1} = \xi_t + \alpha_t(w_t - \xi_t)$ {FW step}
  end if
end for

- By the structure of algorithm, new vertices are added only when convex coefficients are sufficiently optimized. Therefore, BCG outputs highly sparse solutions.

Convergence speed of PCG and BCG

PCG and BCG achieve faster convergence rates than CG algorithm:

Table: Theoretical convergence rates (finite-dimensional cases)

	L-smooth	Strongly convex and polytope
CG	$O(\frac{1}{T})$	$O(\frac{1}{T})$
PCG	$O(\frac{1}{T})$	$\exp(-c_{\mathrm{P}}T)$
BCG	$O(\frac{1}{T})$	$\exp(-c_{\mathrm{B}}T)$

However, both algorithms suffer in high-dimensional cases. In particular, we cannot guarantee convergence in infinite-dimensional cases !

Bottleneck of BCG and PCG

PCG

- Swap Step:
  - Swap Step means the step in which a_t is swapped by w_t.
  - Swap steps affect theoretical analysis and a dimension-dependent constant appears in convergence rates.

BCG

- Simplex Gradient Descent (SiGD):
  - SiGD is the coefficients optimization method in BCG.
  - SiGD is the optimization method on the simplex

$$\left\{(c_1, \ldots, c_k) \in \mathbb{R}^k \,\middle|\, \sum_{i=1}^{k} c_i = 1, c_i \geq 0\right\}.$$

  - The convergence rate of SiGD includes the dimension of polytope and therefore BCG includes a dimension-dependent term in convergence rate.

BPCG algorithm (proposed algorithm)

We propose the following BPCG algorithm. The framework uses that of BCG and the difference is the *local Pairwise step*.

Algorithm Blended Pairwise Conditional Gradients

for $t = 0$ to $T - 1$ **do**
  $a_t \leftarrow \operatorname{argmin}_{v \in S_t} \langle -\nabla f(\xi_t), v \rangle$
  $s_t \leftarrow \operatorname{argmax}_{v \in S_t} \langle -\nabla f(\xi_t), v \rangle$
  $w_t \leftarrow \operatorname{argmax}_{v \in V_C} \langle -\nabla f(\xi_t), v \rangle$
  if $\langle \nabla f(\xi_t), a_t - s_t \rangle \geq \langle \nabla f(\xi_t), \xi_t - w_t \rangle$ **then**
    $\xi_{t+1} = \xi_t + \alpha_t (s_t - a_t)$ {local pairwise step}
  else
    $\xi_{t+1} = \xi_t + \alpha_t (w_t - \xi_t)$ {FW step}
  end if
end for

The moving direction of BPCG is $d_t = s_t - a_t$ or $d_t = w_t - \xi_t$.

Local pairwise steps

In local Pairwise steps, the direcion

$$d_t = s_t - a_t$$

$$\left(a_t = \operatorname*{argmin}_{v \in S_t} \langle -\nabla f(\xi_t), v \rangle, s_t = \operatorname*{argmax}_{v \in S_t} \langle -\nabla f(\xi_t), v \rangle \right)$$

is used.

Properties of local pairwise steps:

- By the definition of s_t and a_t, local pairwise updates are equivalent to the implementation of the PCG over S_t.
- Only the two coefficients that correspond to s_t and a_t are changed.
- We do not mind swap steps in local pairwise steps.

Analysis of BPCG for L-smooth functions

Roughly speaking, for L-smooth convex functions, we have

$$h_t - h_{t+1} \geq \frac{\langle \nabla f(\xi_t), d_t \rangle^2}{2LD^2},$$

where $h_t = f(\xi_t) - f(\xi^*)$ and d_t is the moving direction of BPCG.

Case(A) $\langle \nabla f(\xi_t), a_t - s_t \rangle \geq \langle \nabla f(\xi_t), \xi_t - v_t \rangle$

$$h_t - h_{t+1} \geq \frac{\langle \nabla f(\xi_t), s_t - a_t \rangle^2}{2LD^2} \geq \frac{\langle \nabla f(\xi_t), \xi_t - v_t \rangle^2}{2LD^2} \geq \frac{h_t^2}{2LD^2}$$

Case(B) $\langle \nabla f(\xi_t), a_t - s_t \rangle \leq \langle \nabla f(\xi_t), \xi_t - v_t \rangle$

$$h_t - h_{t+1} \geq \frac{\langle \nabla f(\xi_t), \xi_t - v_t \rangle^2}{2LD^2} \geq \frac{h_t^2}{2LD^2}$$

Theoretical analysis: general smooth case

Theorem

P : convex feasible domain with diameter D (dim P can be ∞)
f : convex and L-smooth.
Let $\{\xi_i\}_{i=0}^T \subset P$ be the sequence given by the BPCG algorithm. Then, it holds that

$$f(\xi_T) - f(\xi^*) \leq \frac{4LD^2}{T}.$$

Since the constant factor $4LD^2$ does not depend on the dimension of the domain, we can apply this result to infinite-dimensional cases!

Analysis of BPCG for strongly convex functions

Roughly speaking, for L-smooth convex functions, we have

$$h_t - h_{t+1} \geq \frac{\langle \nabla f(\xi_t), d_t \rangle^2}{2LD^2}$$

. We use the following two Lemmas:

Lemma (Lacoste-Julien and Jaggi (2015), Inequalities (23) and (28))

Assume that f is μ-strongly convex and P is a polytope with pyramidal width δ. Then,

$$h_t \leq \frac{\langle \nabla f(\xi_t), a_t - w_t \rangle^2}{2\mu\delta^2}.$$

Lemma

For each step t, an inequality $2\langle \nabla f(\xi_t),\, d_t \rangle \geq \langle \nabla f(\xi_t),\, a_t - w_t \rangle$ holds.

Theoretical analysis: polytopes and strongly convex case

Theorem

P : finite-dimensional polytope with pyramidal width δ and diameter D
f : μ-strongly convex and L-smooth
Consider the sequence $\{\xi_i\}_{i=0}^T \subset P$ obtained by the BPCG algorithm. Then, it holds that

$$f(\xi_T) - f(\xi^*) \leq (f(\xi_0) - f(\xi^*)) \exp\left(-c_{f,P}\, T\right),$$

where $c_{f,P} := \frac{1}{2}\min\{\frac{1}{2},\, \frac{\mu\delta^2}{4LD^2}\}$.

Compare the constant factor in convergece rate

The convergence rate for finite-dimensional polytope case:

- BPCG

$$\exp(-c_f T)$$
$$c_f = \frac{1}{2}\min\{\frac{1}{2}, \frac{\mu\delta^2}{4LD^2}\}$$

- PCG

$$\exp(-ck(T))$$
$$k(T) \geq T/(3|V_C|! + 1)$$

BPCG bounds the constant factor better than PCG.

Compare BPCG to other variants

- BPCG ensures $O(\frac{1}{T})$ convergence in infinite-dimensional cases.
- BPCG ensures linear convergence for strongly convex and polytope cases.
- Moreover, BPCG outputs highly sparse solutions since BPCG inherits the framewrok of BCG.

Table: Theoretical convergence rate

	L-smooth infinite-dimensional domain	Strongly convex, finite-dimensional polytope
CG	$O(\frac{1}{T})$	$O(\frac{1}{T})$
PCG	✗	$\exp(-c_{\mathrm{P}}T)$
BCG	✗	$\exp(-c_{\mathrm{B}}T)$
BPCG	$O(\frac{1}{T})$	$\exp(-c_{\mathrm{BP}}T)$

Lazified Version of BPCG

In BPCG, we need to compute the dual gap

$$\max_{v \in V_C} \langle \nabla f(\xi_t), \xi_t - v \rangle$$

in each iteration and we need $|V_C|$ times access.

To reduce computational cost, we employ the *lazification* techique (Braun et al., 2017).

The lazification means the follwoing estimation

$$\Phi_t \approx \max_{v \in V_C} \langle \nabla f(\xi_t), \xi_t - v \rangle$$

Lazified Version of BPCG

Algorithm Lazified Blended Pairwise Conditional Gradients

for $t = 0$ to $T - 1$ **do**
  $a_t \leftarrow \mathrm{argmin}_{v \in S_t} \langle -\nabla f(\xi_t), v \rangle$
  $s_t \leftarrow \mathrm{argmax}_{v \in S_t} \langle -\nabla f(\xi_t), v \rangle$
  $w_t \leftarrow \mathrm{argmax}_{v \in V_C} \langle -\nabla f(\xi_t), v \rangle$
  if $\langle \nabla f(\xi_t), a_t - s_t \rangle \geq \Phi_t$ **then**
    $\xi_{t+1} = \xi_t + \alpha_t (s_t - a_t)$ {local Pairwise step}
  else
    if $\langle \nabla f(\xi_t), \xi_t - w_t \rangle \geq \Phi_t / J$ **then**
      $\xi_{t+1} = \xi_t + \alpha_t (w_t - \xi_t)$ {FW step}
    else
      $\Phi_{t+1} = \Phi_t / 2$
    end if
  end if
end for

Theoretical analysis: general smooth case

Theorem

P : convex feasible domain with diameter D
f : convex and L-smooth.
$\{\xi_i\}_{i=0}^T \subset P$: output of the Lazified BPCG algorithm .

Case (A) If f is μ-strongly convex and P is a polytope with pyramidal width $\delta > 0$, we have

$$f(\xi_t) - f(\xi^*) = O\left(\exp\left(-cT\right)\right) \quad (T \to \infty)$$

for a constant $c > 0$ independent of T.

Case (B) If f is only convex and L-smooth, we have

$$f(\xi_T) - f(\xi^*) = O\left(\frac{1}{T}\right) \quad (T \to \infty).$$

Numerical experiments for finite-dimensional problems

We confirm the effectiveness of BPCG through numerical experiments. BPCG and Lazified BPCG are compared with CG, ACG(Wolfe, 1970) and PCG.

Problem 1 : Convex optimization over probability simplex

$$\min_{x \in \mathbb{R}^n} \|x - x_0\|_2^2$$
$$\text{s.t. } x \in \Delta(n),$$

Here, $\Delta(n) := \{x \in \mathbb{R}^n \mid \sum_{i=1}^n x_i = 1, x_i \geq 0 \ (i = 1, \ldots, n)\}$ and $x_0 \in \Delta(n)$.

Problem 2 : ℓ_p norm ball

$$\min_{x \in \mathbb{R}^n} \|x - x_0\|_2^2$$
$$\text{s.t. } \|x\|_p \leq 1$$

$\|\cdot\|_p$ means the $_p$ norm.

Problem 1 ($n = 200$) : iterations and computational time

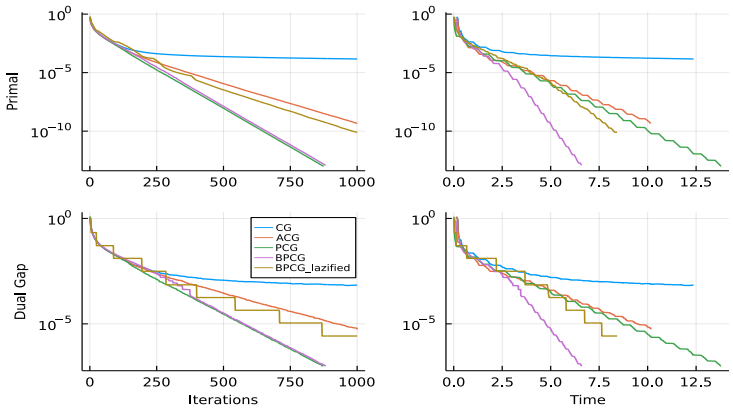


Figure: Problem 1 (x-axis: left: iterations , right: computational time y-axis: top: primal gap, bottom: dual gap)

Convergence speed of BPCG

Iterations: Competitive with PCG.

Computational time: BPCG is the fastest.

Problem 1 ($n = 500$) : sparsity of solutions

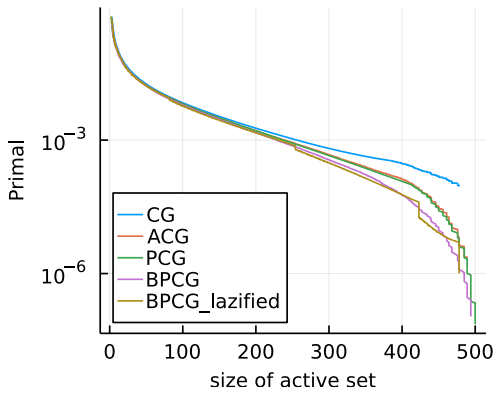


Figure: Problem 1: Convergence of the primal gap for the number of vertices that are the members of convex combination of a solution.

BPCG and Lazified BPCG output sparse solutions.

Problem 2 ($p = 5, n = 1000$) : sparsity of solutions

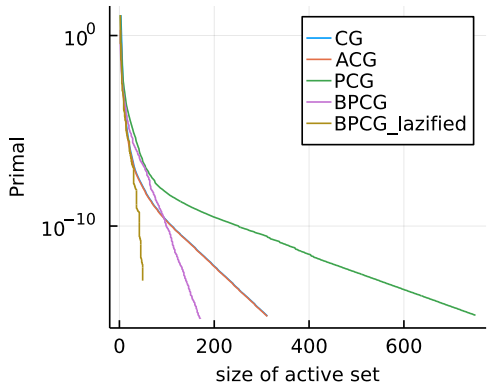


Figure: Problem 2: Convergence of primal gap for the number of vertices that are the members of convex combination of a solution.

BPCG and Lazified BPCG output much sparser solutions.

Numerical experiments (Kernel Herding)

$\mathcal{P}(\Omega)$: all probability measures on $\Omega \in \mathbb{R}^d$

$\mathrm{MMD}(\cdot, \cdot)$: distance between probability measures measured in a Reproducing Kernel Hilbert Space (RKHS) on Ω

Kernel Herding solves the following minimization problem over infinite-dimensional domain $\mathcal{P}(\Omega)$ using a CG manner:

$$\operatorname*{argmin}_{\xi \in \mathcal{P}(\Omega)} \mathrm{MMD}^2(\mu, \xi) \quad (\mu \in \mathcal{P}(\Omega)).$$

The output of Kernel Herding is a discrete measure

$$\xi = \sum_{i=1}^{n} \omega_i \delta_{x_i} \quad (\{\omega_i\}_{i=1}^n \subset \mathbb{R}, \{x_i\}_{i=1}^d \subset \mathbb{R}^d).$$

Using an efficient CG method, we want to derive ξ that approximates μ with small number of nodes n. That is, we want to derive nice sparse soltions.

BPCG for kernel herding

Domain : $\Omega = [-1, 1]^2$, Kernel : Matérn kernel with $\nu = \frac{3}{2}, \frac{5}{2}$.
Optimal rates of the convergence of MMD is $n^{-\frac{5}{4}}, n^{-\frac{7}{4}}$, respectively.

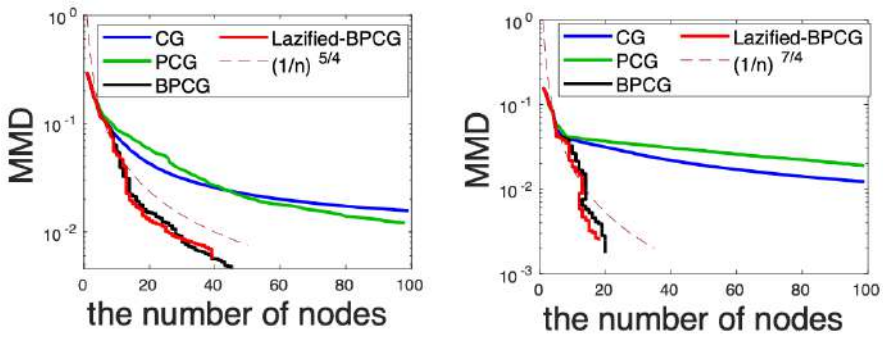


Figure: Matérn kernel ($\nu = 3/2$) (left) and Matérn kernel ($\nu = 5/2$) (right)

Summary

- To overcome the difficulties of PCG and BCG, we proposed the BPCG algorithm.
- We showed that for BPCG we can ensure $O(\frac{1}{T})$ convergence even if the dimension of convex constraints is infinite. For strongly convex and polytope cases, we can guarantee the linear rate.
- Through numerical experiments, we showed the practical effectiveness of BPCG. In particular, sparsity of solutions is notable.

Reference I

G. Braun, S. Pokutta, and D. Zink. Lazifying conditional gradient algorithms. In *Proceedings of the 34th International Conference on Machine Learning*, pages 566–575, 2017.

G. Braun, S. Pokutta, D. Tu, and S. Wright. Blended conditional gradients: the unconditioning of conditional gradients. In *Proceedings of the 36th International Conference on Machine Learning (PMLR)*, volume 97, pages 735–743, 2019.

S. Lacoste-Julien and M. Jaggi. On the global linear convergence of Frank-Wolfe optimization variants. In C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, editors, *Advances in Neural Information Processing Systems 28*, pages 496–504. Curran Associates, Inc., 2015. URL `http://papers.nips.cc/paper/5925-on-the-global-linear-convergence-of-frank-wolfe-optimi` `pdf`.

Reference II

E. S. Levitin and B. T. Polyak. Constrained minimization methods. *USSR Computational Mathematics and Mathematical Physics*, 6(5):1–50, 1966.

K. K. Tsuji, K. Tanaka, and S. Pokutta. Pairwise conditional gradients without swap steps and sparser kernel herding. In *International Conference on Machine Learning*, pages 21864–21883. PMLR, 2022.

P. Wolfe. Convergence theory in nonlinear programming. In *Integer and Nonlinear Programming*, pages 1–36. North-Holland, Amsterdam, 1970.

Proofs in Extremal Combinatorics through Optimization

Christoph SPIEGEL

Zuse Institute Berlin, Germany
spiegel@zib.de

We present a fully computer-assisted proof system for solving a particular family of problems in Extremal Combinatorics. Existing techniques using Flag Algebras have proven powerful in the past, but have so far lacked a computational counterpart to derive matching constructive bounds. We demonstrate that common search heuristics are capable of finding constructions far beyond the reach of human intuition. Additionally, the most obvious downside of such heuristics, namely a missing guarantee of global optimality, can often be fully eliminated in this case through lower bounds and stability results coming from the Flag Algebra approach.

To illustrate the potential of this approach, we study two related and well-known problems in Extremal Graph Theory that go back to questions of Erdős from the 60s. Most notably, we present the first major improvement in the upper bound of the Ramsey multiplicity of K_4 in 25 years, precisely determine the first off-diagonal Ramsey multiplicity number, and settle the minimum number of independent sets of size four in graphs with clique number strictly less than five.

Proofs in Extremal Combinatorics through Optimization

6th RIKEN-IMI-ISM-NUS-ZIB-MODAL-NHR Workshop

Christoph Spiegel

17th of September 2022

Results are joint work with...

Olaf Parczyk
Freie Universität Berlin

Sebastian Pokutta
Zuse Institute Berlin

Tibor Szabó
Freie Universität Berlin

Research partially funded through Math+ project EF1-12

Proofs in Combinatorics through Optimization

1. The Ramsey Multiplicity Problem
2. Search Heuristics for Upper Bounds
3. Flag Algebras for Lower Bounds
4. A Related Problem

The Ramsey Multiplicity of triangles

Theorem (Ramsey 1930)

For any $t \in \mathbb{N}$ there exists $R(t) \in \mathbb{N}$ such that any 2-edge-coloring of the complete graph of order at least $R(t)$ contains a monochromatic clique of size t.

A well-known question: Can we determine $R(t)$?

A related question: How many cliques do we need to have? That means, letting $k_t(G)$ denote the fraction of all possible t-cliques in G, what is

$$c_t = \lim_{n\to\infty} \min\{k_t(\overline{G}) + k_t(G) : G \text{ graph of order } n\}?$$

Theorem (Goodman 1959)

$c_3 = 1/4$.

$\rightarrow$ Same as Erdős-Rényi random graph! $\rightarrow$

Conjecture (Erdős 1962)

$c_t = 2^{1-\binom{t}{2}}$.

1. The Ramsey Multiplicity Problem

Ramsey Multiplicity beyond triangles

Theorem (Thomason 1989 / 1997)

$c_4 \leq 0.970 \cdot 2^{-5}$ *and* $c_5 \leq 0.881 \cdot 2^{-9}$.

Theorem (Even-Zohar and Linial '15)

$c_4 \leq 0.969 \cdot 2^{-5}$.

Erdős conjecture was false! But what about lower bounds?

Theorem (Giraud 1976)

$c_4 \geq 0.695 \cdot 2^{-5}$.

Theorem (Sperfeld / Nieß'11)

$c_4 \geq 0.914 \cdot 2^{-5}$.

Theorem (Grzesik et al. '20)

$c_4 \geq 0.947 \cdot 2^{-5}$.

Both the best upper and lower bounds heavily rely on computer-assistence!

Theorem (Parczyk, Pokutta, S., and Szabó 2022+)

$c_4 \leq 0.964 \cdot 2^{-5}$ and $0.780 \cdot 2^{-9} \leq c_5 \leq 0.874 \cdot 2^{-9}$.

How can we use Optimization to formulate mathematical proofs?

Proofs in Combinatorics through Optimization

Graph blow-ups

We want constructive bounds that are 'finitely describable'. Random graphs are one source for such constructions. Another natural deterministic one are graph blow-ups.

Definition

The *m-fold blow-up* $C[m]$ of a graph C is given by replacing each vertex in C with an independent set of size m. Two vertices are adjacent if the originals were.

Using blow-ups, we can derive an upper bounds for c_t from **any** graph C through

$$c_t \leq \lim_{m\to\infty} k_t(\overline{C[m]}) + k_t(C[m]). \tag{1}$$

This is in fact efficiently computable since

$$\lim_{m\to\infty} k_t(C[m]) = n^{\underline{t}}\, k_t(C)\ /n^t \quad \text{and} \quad \lim_{m\to\infty} k_t(\overline{C[m]}) = \sum_{j=1}^{t} S(t,j) n^{\underline{j}}\, k_j(\overline{C})/n^t. \tag{2}$$

Constructing graphs through search heuristics

For fixed n and $\mathbf{s} \in \{0,1\}^{\binom{n}{2}}$ let $C_{\mathbf{s}} = \left([n], \{ij : i < j, s_{\binom{j-1}{2}+i} = 1\}\right)$ and consider

$$\min_{\mathbf{s}\in\{0,1\}^{\binom{n}{2}}} \sum_{j=1}^{s} \frac{S(t,j) n^{\underline{j}}\, k_j(\overline{C_{\mathbf{s}}})}{n^t} + \frac{n^{\underline{t}}\, k_t(C_{\mathbf{s}})}{n^t}.$$

So we have found our optimization problem! How to solve it?

For $n \lessapprox 40$ we can use Search Heuristics.

Unfortunately even $n = 40$ is much too small for c_4 and c_5, barely disproving Erdős' original conjecture. **Can we use combinatorial insights to bias the search space?**

Constructing Cayley graphs through search heuristics

Thomason's constructions are based on computing the values of XOR-graph-products. The results are in fact Cayley graphs in $C_3^{\times 2} \times C_2^{\times 5}$ and $C_3 \times C_2^{\times 6}$.

Definition

Given an abelian group G and set $S \subseteq G^\star$ satisfying $S^{-1} = S$, the associated *Cayley graph* has vertex set G and $g_1, g_2 \in G$ are adjacent if and only if $g_1^{-1} g_2 \in S$.

Idea. Why not directly search Cayley graph constructions?

The binary vector **s** now represents the generating set S. Since $|G|/2 < |S| < |G|$ the number of variables is therefore linear (instead of quadratic) in the number of vertices!

The groups $C_3 \times C_2^{\times 8}$ and $C_3 \times C_2^{\times 6}$ give the improved upper bounds for c_4 and c_5.

Proofs in Combinatorics through Optimization

A trivial computational lower bound

The Flag Algebra SDP approach can be seen as (i) a formalized Cauchy-Schwarz-type argument and **(ii) an improvement over a trivial computational lower bound.**

Let $d_H(G)$ denote the probability that $v(H)$ vertices chosen uniformly at random in G induce a copy of H. Writing $c_t(G) = k_t(G) + k_t(\overline{G})$, basic double counting gives us

$$c_t(G) = \sum_{\substack{H \text{ graph} \\ v(H)=N}} d_H(G)\, c_t(H) \tag{3}$$

for $t \leq N \leq v(G)$. For any $N \geq t$ this implies a trivial lower bound of

$$c_t \geq \min_{\substack{H \text{ graph} \\ v(H)=N}} c_t(H). \tag{4}$$

The Flag Algebras SDP approach

Razborov (2007) introduced *Flag Algebras* in order to study this type of problem. One important observation is that for any $Q \succeq 0$ the coefficients $a_H = \langle Q, D_H \rangle$ satisfy

$$\sum_{\substack{H \text{ graph} \\ v(H)=N}} d_H(G)\, a_H \leq O(1/v(G)) \tag{5}$$

for any graph G. Through (3) this implies the (hopefully improved) bound

$$c_t \geq \min_{\substack{H \text{ graph} \\ v(H)=N}} c_t(H) - a_H. \tag{6}$$

This approach gives the best current lower bounds for c_4 and c_5. The biggest bottleneck for further improvements consists of finding Q for larger N.

Proofs in Combinatorics through Optimization

Off-diagonal Ramsey Multiplicity

Question. Determining c_3 is easy, but even c_4 has been unresolved for over 60 years, so can we say more when studying the off-diagonal variant

$$c_{s,t} = \lim_{n\to\infty} \min\{k_s(\overline{G}) + k_t(G) : |G| = n\}?$$

A famous result of Reiher from 2016 implies that $c_{2,t} = 1/(t-1)$.

Theorem (Parczyk, Pokutta, S., and Szabó 2022+)

$c_{3,4} = 689 \cdot 3^{-8}$ and any large enough graph G admits a strong homomorphism into the Schläfli graph after changing at most $O(k_3(\overline{G}) + k_4(G) - c_{3,4})\, v(G)^2$ edges.

The fact that we can show stability proves that the search heuristic found a unique global optimum over all graphs of order 27!

ZUSE INSTITUTE BERLIN

Thank you for your attention!

The UG framework version 1.0: An update

Yuji Shinano

Zuse Institute Berlin, Germany
shinano@zib.de

The Ubiquity Generator Framework (UG) version 1.0 was released last year. It was designed to parallelize powerful state-of-the-art branch-and-bound based solvers externally in order to exploit their powerful performance. We call the underlying solvers ``base solvers''; originally, a base solver is a branch-and-bound based solver, but in the current release, it is redefined as any solver that is being parallelized by UG, since, in version 1.0, it was generalized to be a software framework for high-level task parallelization. In this talk, we present the concept of high-level task parallelization and its flexibility. We will show a few recent success stories of the instantiated parallel solvers by UG version 1.0.

References

[1] Y. Shinano, T. Achterberg, T. Berthold, S. Heinz, T. Koch and M. Winkler, "Solving Open MIP Instances with ParaSCIP on Supercomputers Using up to 80,000 Cores," 2016 IEEE International Parallel and Distributed Processing Symposium (IPDPS), 2016, pp. 770-779, doi: 10.1109/IPDPS.2016.56.

[2] G. Gamrath, T. Koch, S. Maher, D. Rehfeldt, and Y. Shinano, “SCIP-Jack—a solver for STP and variants with parallelization extensions,”Mathematical Programming Computation, vol. 9, no. 2, pp. 231–296,2017.

[3] Y. Shinano, D. Rehfeldt, and T. Koch, “Building optimal steiner trees on supercomputers by using up to 43,000 cores,” in Integration of Constraint Programming, Artificial Intelligence, and Operations Research.CPAIOR 2019, vol. 11494, 2019, pp. 529–539.

[4] Y. Shinano, D. Rehfeldt, and T. Gally, “An easy way to build parallel state-of-the-art combinatorial optimization problem solvers: A com-putational study on solving steiner tree problems and mixed integer semidefinite programs by using ug[scip-*,*]-libraries,” in2019 IEEE International Parallel and Distributed Processing Symposium Workshops(IPDPSW), 2019, pp. 530–541.

[5] K. Fujii, N. Ito, S. Kim, M. Kojima, Y. Shinano, AND K.-C. Toh, Solving challenging large scale QAPs, Tech. Rep. 21-02, ZIB, Takustr. 7, 14195 Berlin,2021.

[6] N.Tateiwa, Y.Shinano, S.Nakamura, A.Yoshida, S.Kaji, M.Yasuda, and K.Fujisawa. Massive parallelization for finding shortest lattice vectors based on ubiquity generator framework. In SC20: International Conference for High Performance Computing, Networking, Storage and Analysis, pages 1-15. IEEE, 2020.

Ubiquity Generator (UG) Framework Version 1.0: An update

Yuji Shinano
Zuse Institute Berlin

17.09.2022

The 6th RIKEN-IMI-ISM-NUS-ZIB-MODAL-NHR Workshop on Advances in Classical and Quantum Algorithms for Optimization and Machine Learning, Japan

Outline

- What is Ubiquity Generator (UG) Framework
  - Basic concept
  - Success stories
- Ubiquity Generator (UG) Framework – **version 1.0**
  - Concept of high level task parallelization
  - What we could do with UG ver. 1.0
  - What comes with UG ver.1.0
- Summary

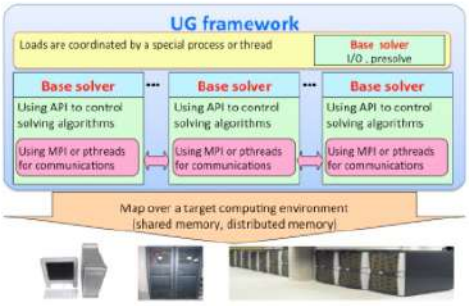
What is Ubiquity Generator (UG) Framework

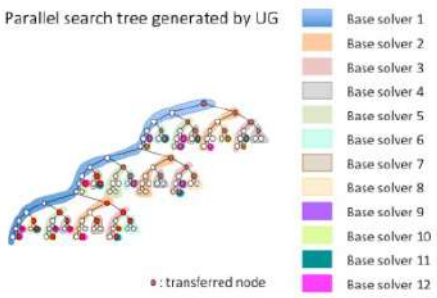
➢ Current UG web page (https://ug.zib.de) says:

UG is a generic framework to parallelize branch-and-bound based solvers (e.g., MIP, MINLP, ExactIP) in a distributed or shared memory computing environment.

"Generic": written in C++

- Exploits powerful performance of state-of-the-art "base solvers", such as SCIP, CPLEX, etc.
- Without the need for base solver parallelization





Base solvers and communication libraries are abstracted within UG. A parallel solver instantiated by UG framework is named:

- ug[Base solver, Communication libaray]

What is Ubiquity Generator (UG) Framework

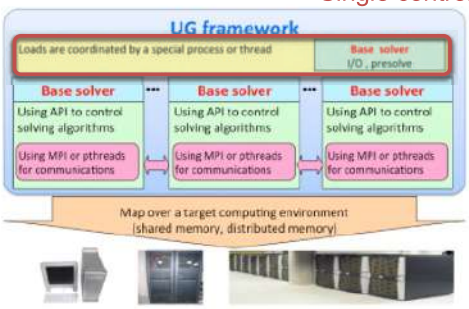
➢ Current UG web page (https://ug.zib.de) says:

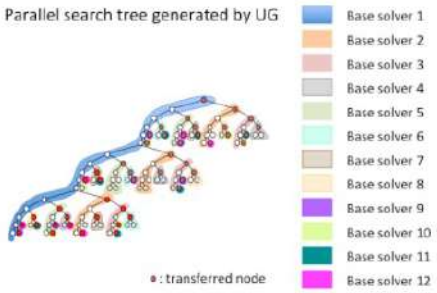
UG is a generic framework to parallelize branch-and-bound based solvers (e.g., MIP, MINLP, ExactIP) in a distributed or shared memory computing environment.

"Generic": written in C++

- Exploits powerful performance of state-of-the-art "base solvers", such as SCIP, CPLEX, etc.
- Without the need for base solver parallelization

Single controller: LoadCoordinator





Base solvers and communication libraries are abstracted within UG. A parallel solver instantiated by UG framework is named:

- ug[Base solver, Communication libaray]

What is Ubiquity Generator (UG) Framework

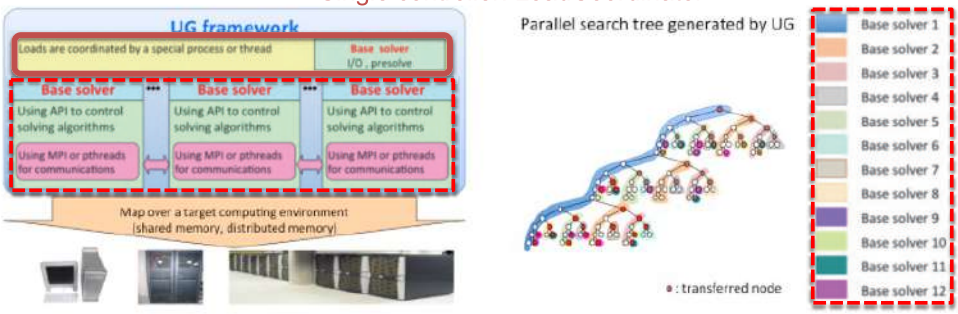
➢ Current UG web page (https://ug.zib.de) says:

UG is a generic framework to parallelize branch-and-bound based solvers (e.g., MIP, MINLP, ExactIP) in a distributed or shared memory computing environment.

"Generic": written in C++ Parallelize Base solver externally

- Exploits powerful performance of state-of-the-art "base solvers", such as SCIP, CPLEX, etc.
- Without the need for base solver parallelization

Single controller: LoadCoordinator



Base solvers and communication libraries are abstracted within UG. A parallel solver instantiated by UG framework is named:

- ug[Base solver, Communication libaray]

What is Ubiquity Generator (UG) Framework

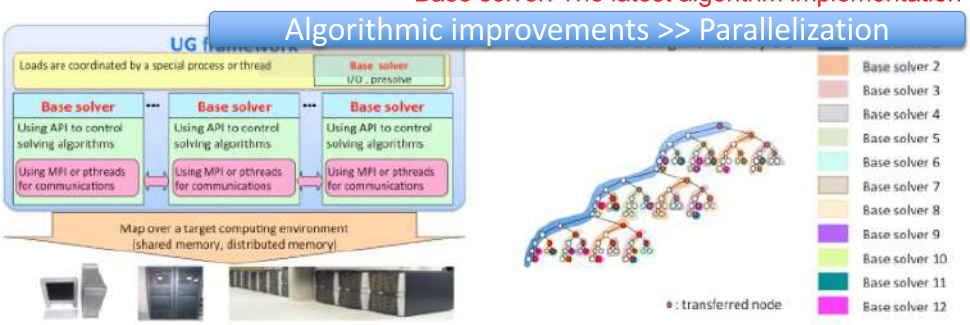
➢ Current UG web page (https://ug.zib.de) says:

UG is a generic framework to parallelize branch-and-bound based solvers (e.g., MIP, MINLP, ExactIP) in a distributed or shared memory computing environment.

- Exploits powerful performance of state-of-the-art "base solvers", such as SCIP, CPLEX, etc.
- Without the need for base solver parallelization

Base solver: The latest algorithm implementation

Algorithmic improvements >> Parallelization



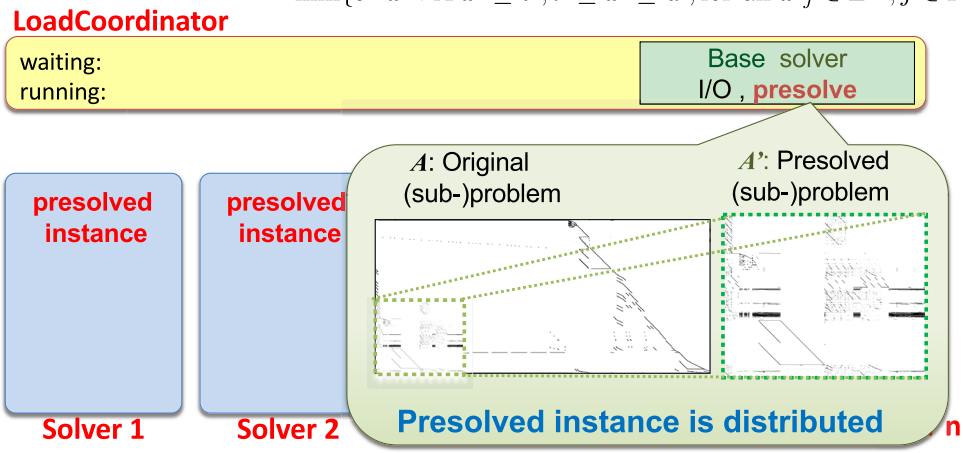
Base solvers and communication libraries are abstracted within UG. A parallel solver instantiated by UG framework is named:

- ug[Base solver, Communication libaray]

How UG initializes all solvers

[Initialization] Original $\min\{c^\top x : Ax \le b, l \le x \le u, \text{for all } x_j \in \mathbb{Z}^n, j \in I\}$

$\min\{c'^\top x' : A'x' \le b', l' \le x' \le u', \text{for all } x'_j \in \mathbb{Z}^{n'}, j \in I'\}$

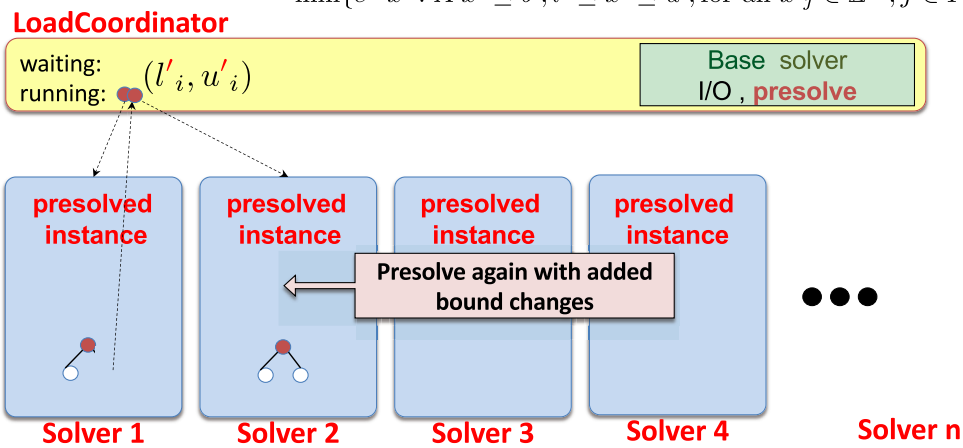


All Solvers keep the **presolved instance**

How UG does ramp-up (start with a single tree search)

[Normal Ramp-up] Original $\min\{c^\top x : Ax \le b, l \le x \le u, \text{for all } x_j \in \mathbb{Z}^n, j \in I\}$

$\min\{c'^\top x' : A'x' \le b', l' \le x' \le u', \text{for all } x'_j \in \mathbb{Z}^{n'}, j \in I'\}$



This procedure last until all solvers become busy

How UG does ramp-up (start with a single tree search)

[Normal Ramp-up] Original $\min\{c^{\mathsf{T}}x : Ax \le b, l \le x \le u, \text{for all } x_j \in \mathbb{Z}^n, j \in I\}$

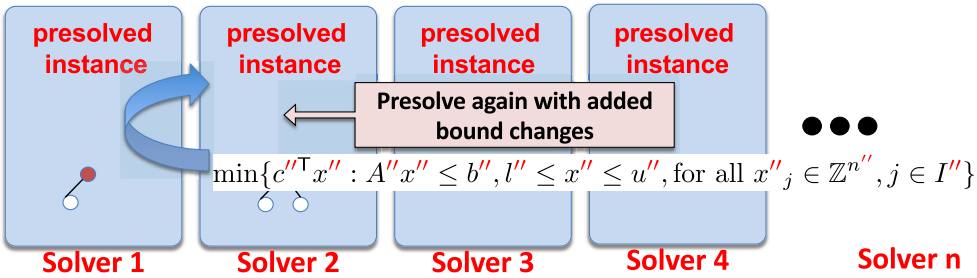
$\min\{c'^{\mathsf{T}}x' : A'x' \le b', l' \le x' \le u', \text{for all } x'_j \in \mathbb{Z}^{n'}, j \in I'\}$

LoadCoordinator

waiting: (l'_i, u'_i)

running:

Base solver
I/O , presolve

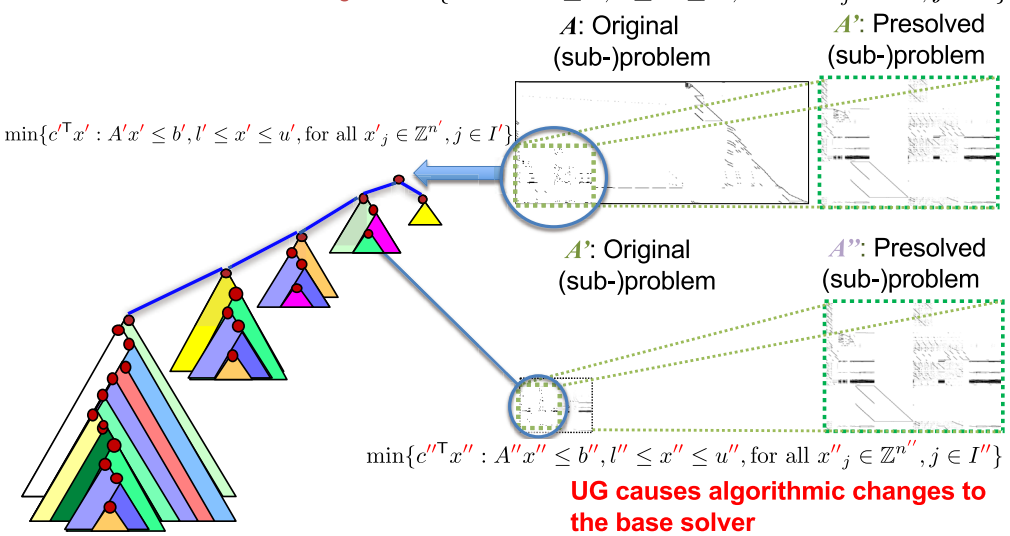


All transfer data need to be converted back for the presolved instance

All feasible solutions need to be converted back for the original instance

Layered presolving

Original $\min\{c^{\mathsf{T}}x : Ax \le b, l \le x \le u, \text{for all } x_j \in \mathbb{Z}^n, j \in I\}$



UG causes algorithmic changes to the base solver

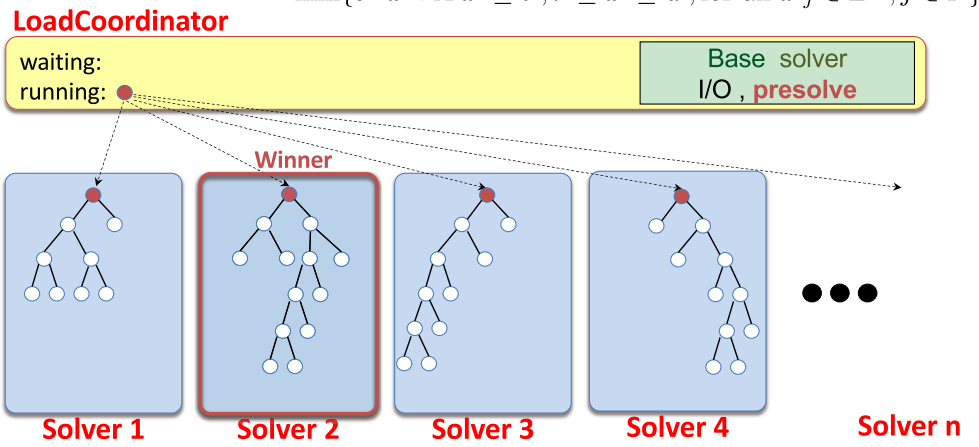
Global view of tree search

How UG does ramp-up (start with multiple trees search)

[Ramp-up(Racing)]

Original $\min\{c^{\mathsf{T}}x : Ax \le b, l \le x \le u, \text{for all } x_j \in \mathbb{Z}^n, j \in I\}$

$\min\{c'^{\mathsf{T}}x' : A'x' \le b', l' \le x' \le u', \text{for all } x'_j \in \mathbb{Z}^{n'}, j \in I'\}$



All Solvers start solving immediately, trying to generate different search trees

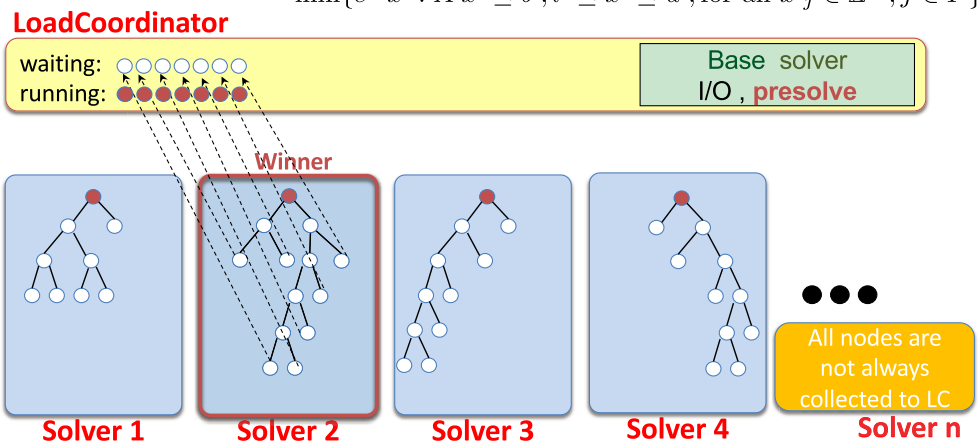
ug[SCIP,*]: work with distributed domain propagation

How UG does ramp-up (start with multiple trees search)

[Ramp-up(Racing)]

Original $\min\{c^{\mathsf{T}}x : Ax \le b, l \le x \le u, \text{for all } x_j \in \mathbb{Z}^n, j \in I\}$

$\min\{c'^{\mathsf{T}}x' : A'x' \le b', l' \le x' \le u', \text{for all } x'_j \in \mathbb{Z}^{n'}, j \in I'\}$



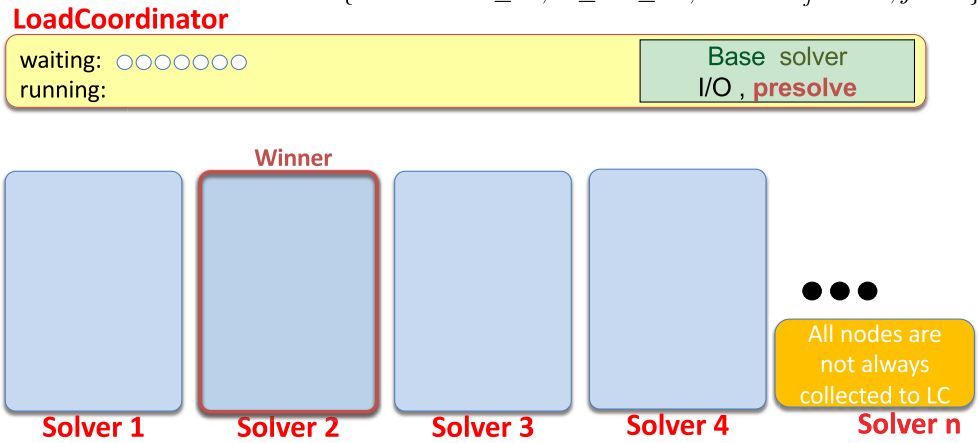
When the racing terminated without enough open nodes, automatically the ramp-up is continued with normal ramp-up

How UG does ramp-up (start with multiple trees search)

[Ramp-up(Racing)]

Original $\min\{c^\top x : Ax \le b, l \le x \le u, \text{for all } x_j \in \mathbb{Z}^n, j \in I\}$

$\min\{c'^\top x' : A'x' \le b', l' \le x' \le u', \text{for all } x'_j \in \mathbb{Z}^{n'}, j \in I'\}$



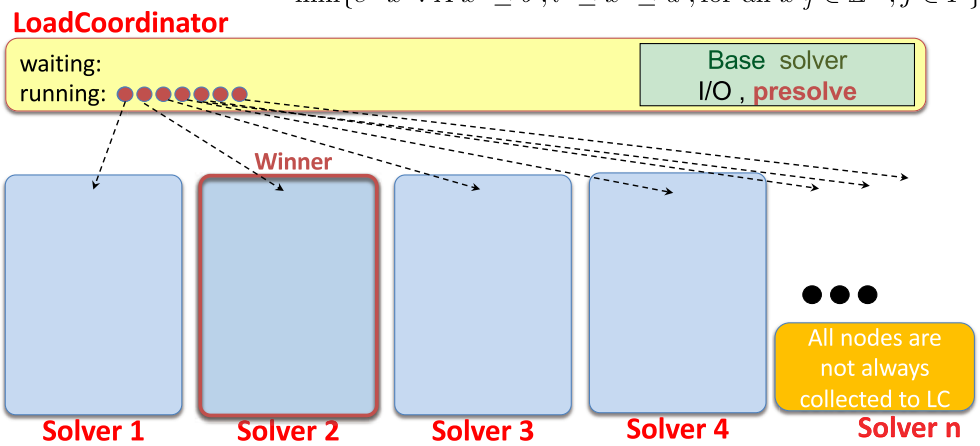
When the racing terminated without enough open nodes, automatically the ramp-up is continued with normal ramp-up

How UG does ramp-up (start with multiple trees search)

[Ramp-up(Racing)]

Original $\min\{c^\top x : Ax \le b, l \le x \le u, \text{for all } x_j \in \mathbb{Z}^n, j \in I\}$

$\min\{c'^\top x' : A'x' \le b', l' \le x' \le u', \text{for all } x'_j \in \mathbb{Z}^{n'}, j \in I'\}$



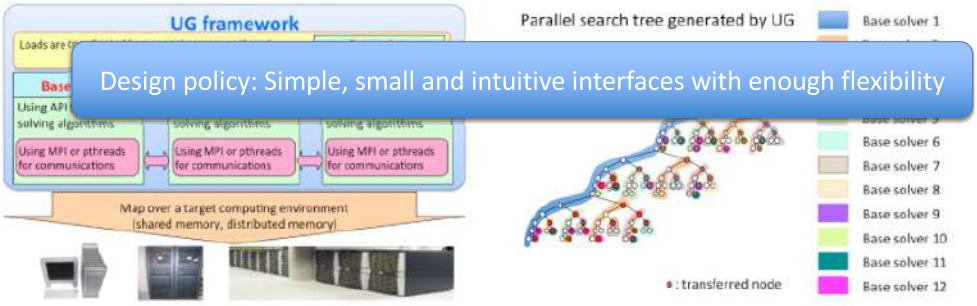
When the racing terminated without enough open nodes, automatically the ramp-up is continued with normal ramp-up

What is Ubiquity Generator (UG) Framework

➢ Current UG web page (https://ug.zib.de) says:

UG is a generic framework to parallelize branch-and-bound based solvers (e.g., MIP, MINLP, ExactIP) in a distributed or shared memory computing environment.

- Exploits powerful performance of state-of-the-art "base solvers", such as SCIP, CPLEX, etc.
- Without the need for base solver parallelization



Design policy: Simple, small and intuitive interfaces with enough flexibility

Base solvers and communication libraries are abstracted within UG. A parallel solver instantiated by UG framework is named:

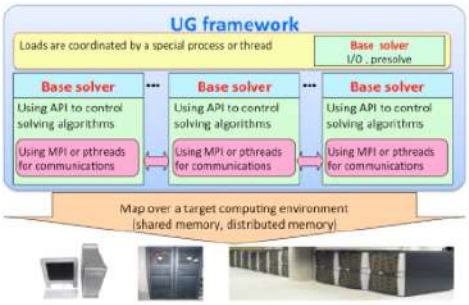
- ug[Base solver, Communication libaray]

What is Ubiquity Generator (UG) Framework

➢ Current UG web page (https://ug.zib.de) says:

UG is a generic framework to parallelize branch-and-bound based solvers (e.g., MIP, MINLP, ExactIP) in a distributed or shared memory computing environment.

- Exploits powerful performance of state-of-the-art "base solvers", such as SCIP, CPLEX, etc.
- Without the need for base solver parallelization



Base solver can be

- Single threaded solver
- Multi threaded solver
- Distributed memory parallel solver

Maximum number of base solvers parallelized so far:

103,584 solvers (MPI processes) on HLRN IV (Note: initial target was 10,000)

Base solvers and communication libraries are abstracted within UG. A parallel solver instantiated by UG framework is named:

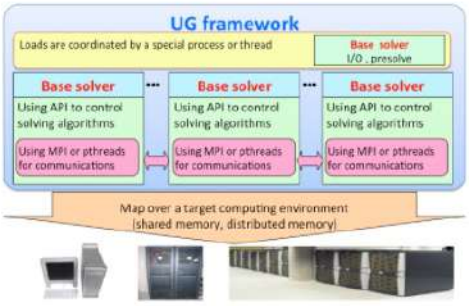
- ug[Base solver, Communication libaray]

What is Ubiquity Generator (UG) Framework

➢ Current UG web page (https://ug.zib.de) says:

UG is a generic framework to parallelize branch-and-bound based solvers (e.g., MIP, MINLP, ExactIP) in a distributed or shared memory computing environment.

- Exploits powerful performance of state-of-the-art "base solvers", such as SCIP, CPLEX, etc.
- Without the need for base solver parallelization



Provide a **systematic way** to develop a large scale distributed memory solver

1. debugging base solver itself
2. debugging the shared memory version (ug[Base solver, C++11]) on a PC
3. debugging the distributed memory version (ug[Base solver, MPI]) on a PC cluster or on a supercomputer

Base solvers and communication libraries are abstracted within UG. A parallel solver instantiated by UG framework is named:

- ug[Base solver, Communication libaray] Solver and Communication parts are abstracted

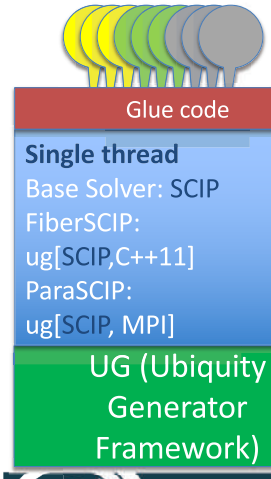
Instantiated parallel solvers by UG (B&B based)

Mixed Integer Programming Problem(MIP) Solvers: Keep solving open instances using up to 103,680 cores (103,584 MPI processes) ParaXpress potentially could handle over a million CPU cores		Stochastic MIP Solver: Potentially could handle over a million CPU cores	Traveling Salesman Problem (TSP) Solver (Experimental)
Single thread Base Solver: SCIP FiberSCIP: ug[SCIP,C++11] ParaSCIP: ug[SCIP, MPI]	**Multi threaded** Base Solver: Xpress FiberXpress: ug[Xpress,C++11] ParaXpress: ug[Xpress, MPI]	**Distributed memory** Base Solver: PIPS-SBB ug[PISP-SBB, MPI]	Base Solver: Concorde FiberConcorde: ug[Concorde,C++11] ParaConcorde: ug[Concorde, MPI]
UG (Ubiquity Generator Framework)			

Experimental: Before UG 1.0 cannot handle the 'base solver', that is, UG framework itself was modified to realize it

What is ug[SCIP-*,*]-libraries

- SCIP (Solving Constraint Integer Programs)
  - A software framework for LP based B&B algorithms
  - plugin based design



Customized SCIP solvers (ex. SCIP-Jack) are developed as a set of plugins plus main function

Software libraries

Parallel Branch-and-Cut framework is provided

ug[SCIP-*,*]-libraries are **general purpose parallel B&B libraries**

Instantiated parallel solvers by UG (**Non**-B&B based)

- **MAP-SVP:** Massively Parallel Solver for Shortest Vector Problem (SVP)
  - Lattices and SVP
    - An n-dimensional **lattice** is the discrete set

$$\mathcal{L}(\mathbf{B}) = \left\{ \sum_{i=1}^{n} x_i \mathbf{b}_i;\ x_i \in \mathbb{Z} \right\}$$

where $\mathbf{B} = (\mathbf{b}_1, \dots, \mathbf{b}_n)$ are linearly independent vectors. (This $\mathbf{B}$ is called a "*basis*".)

    - The **Shortest Vector Problem (SVP)** asks to find a *shortest non-zero vector* in the lattice:
      - minimize $\|\mathbf{v}\|$
      - subject to $\mathbf{v} \in \mathcal{L}(\mathbf{B}) \setminus \{\mathbf{0}\}$

the security of many cryptosystems is based on the hardness of an approximate variant of the SVP

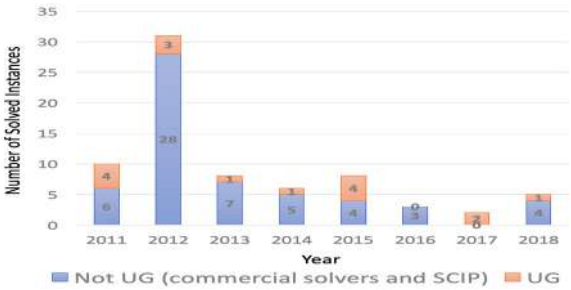
  - Experimental: Before UG 1.0 cannot handle the 'base solver', that is, UG framework itself was modified to realize it

- **CMAP-LAP**: Configurable Massively Parallel Solver **Framework** for Lattice Problems
  - This solver has developed on UG version 1.0 candidate
    - UG version 1.0 is a generalized UG

Success stories - 1

- **ParaSCIP** = **ug[SCIP, MPI]**: Massively parallel solver for MIP
  - Solved 21 previously unsolved MIP instances from MIPLIB for the first time by using up to 80,000 cores
    (It is running using 103,680 cores on HLRN IV, currently)

Y. Shinano, T. Achterberg, T. Berthold, S. Heinz, T. Koch and M. Winkler, "Solving Open MIP Instances with ParaSCIP on Supercomputers Using up to 80,000 Cores," 2016 IEEE International Parallel and Distributed Processing Symposium (IPDPS), 2016, pp. 770-779, doi: 10.1109/IPDPS.2016.56.



https://miplib.zib.de/

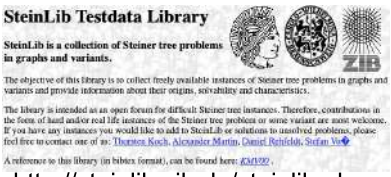
Success stories - 2

- **ug[SCIP-Jack,MPI]**: Massively parallel solver for Steiner Tree Problems in Graphs and their relatives
  - Solved 5 open instances from SteinLib by using up to 43,000 cores

G. Gamrath, T. Koch, S. Maher, D. Rehfeldt, and Y. Shinano, "SCIP-Jack—a solver for STP and variants with parallelization extensions,"Mathematical Programming Computation, vol. 9, no. 2, pp. 231–296,2017.

Y. Shinano, D. Rehfeldt, and T. Koch, "Building optimal steiner trees on supercomputers by using up to 43,000 cores," in Integration of Constraint Programming, Artificial Intelligence, and Operations Research.CPAIOR 2019, vol. 11494, 2019, pp. 529–539.

Y. Shinano, D. Rehfeldt, and T. Gally, "An easy way to build parallel state-of-the-art combinatorial optimization problem solvers: A com-putational study on solving steiner tree problems and mixed integer semidefinite programs by using ug[scip-*,*]-libraries," in2019 IEEE International Parallel and Distributed Processing Symposium Workshops(IPDPSW), 2019, pp. 530–541.



http://steinlib.zib.de/steinlib.php

Success stories - 3

- **ParaXpress** = **ug[Xpress, MPI]**: Massively parallel solver for MIP
  - Solved 3 previously unsolved MIP instances from MIPLIB for the first time (2017, 2018, no paper, yet)
  - Generated good feasible solutions for ITC 2021

Organization Rules Format Q&A Instances Validation

ITC 2021: International Timetabling Competition on Sports Timetabling

How to cite?

Van Bulck, D., Goossens, D., Beliën, J. & Davari, M. International Timetabling Competition 2021: Sports Timetabling – website. itc2021.ugent.be

Van Bulck, D., Goossens, D., Beliën, J., and Davari, M. (2021). The Fifth International Timetabling Competition (ITC 2021): Sports Timetabling. Proceedings of MathSport International 2021 Conference, MathSport, pp. 117-122.

- **5th place** (236 points)
  Team MODAL consisting of Thorsten Koch, Timo Berthold, and Yuji Shinano.
  Research Campus MODAL, Zuse Institute Berlin

Success stories - 4

- **ParaQapNB** = **ug[QapNB, MPI]**: Massively parallel solver for Quadratic Assignment Problem (QAP)
  - Solved 3 previously unsolved QAP instances from QAPLIB using up to 5,184 cores

Miguel Anjos, PhD, FCAE, FEUROPT, SMIEEE

Miguel.F.Anjos@ed.ac.uk

Home Bio Events Recognition Research Consulting Service Supervision Other

QAPLIB is a Quadratic Assignment Problem Library.

Four previously unsolved QAPLIB instances have been solved to optimality in recent years: se[illegible] tai30a, [illegible]35b, tai40b, an[illegible] sko42.

Tho40 was solved !

https://www.miguelanjos.com/qaplib

K. FUJII, N. ITO, S. KIM, M. KOJIMA, Y. SHINANO, AND K.-C. TOH, Solving challenging large scale QAPs, Tech. Rep. 21-02, ZIB, Takustr. 7, 14195 Berlin,2021

Koichi Fuji,
Solving Large Scale Open QAPs by Massively Parallel DNN-based Branch-and-bound Method

Success stories - 5

- **MAP-SVP:** Massively Parallel Solver for Shortest Vector Problem (SVP)
  - Achieved new records for **104**, **111**, **121** and **127**dimensions in SVP Challenge using up to 103,680 cores

    (Experimental)

N.Tateiwa, Y.Shinano, S.Nakamura, A.Yoshida, S.Kaji, M.Yasuda, and K.Fujisawa. Massive parallelization for finding shortest lattice vectors based on ubiquity generator framework. In SC20: International Conference for High Performance Computing, Networking, Storage and Analysis, pages 1-15. IEEE, 2020.

■ **Lattices and SVP**

- An n-dimensional **lattice** is the discrete set

$$\mathcal{L}(\mathbf{B}) = \left\{ \sum_{i=1}^{n} x_i \mathbf{b}_i;\ \ x_i \in \mathbb{Z} \right\}$$

  where $\mathbf{B} = (\mathbf{b}_1, \dots, \mathbf{b}_n)$ are linearly independent vectors. (This $\mathbf{B}$ is called a "*basis*".)
- The **Shortest Vector Problem (SVP)** asks to find a *shortest non-zero vector* in the lattice:
  - minimize $\|\mathbf{v}\|$
  - subject to $\mathbf{v} \in \mathcal{L}(\mathbf{B}) \setminus \{\mathbf{0}\}$

https://www.latticechallenge.org/svp-challenge/

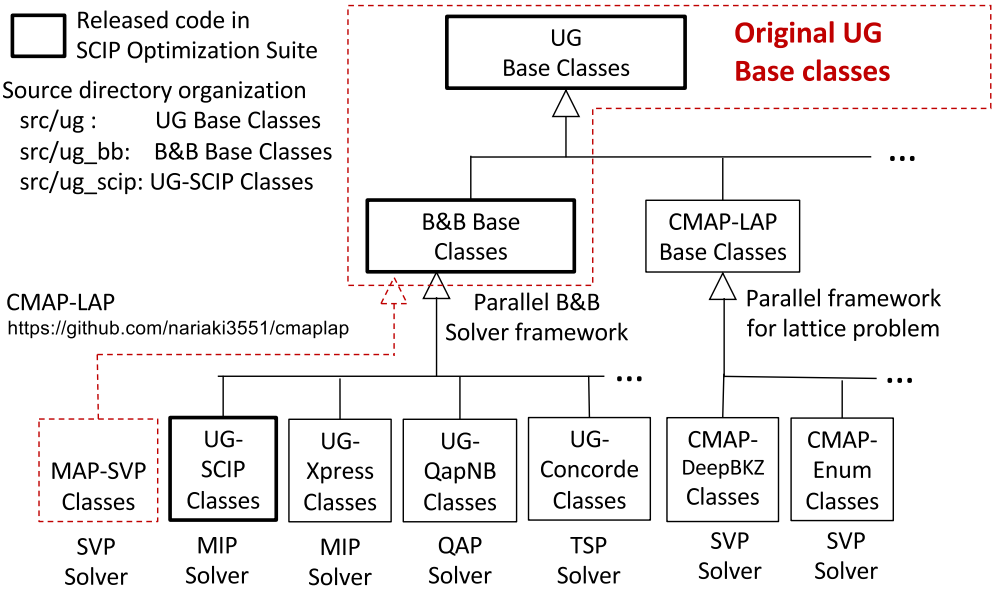
What is Ubiquity Generator (UG) Framework– **Ver.1.0**

- UG is a **high-level task parallelization framework**
  - Can parallelize any kind of solver
    - which needs to share some data among running solvers
    - which needs to share them very flexibly

⇒ **All experimental parallel solvers can be handled with a single unified framework UG version 1.0**

The framework was changed internally!

Classes hierarchy and source code directory organization



What is Ubiquity Generator (UG) Framework– Ver.1.0

- UG is a **high-level task parallelization framework**
  - Can parallelize any kind of solver
    - which needs to share some data among running solvers
    - which needs to share them very flexibly

All experimental parallel solvers can be handled with a single unified framework UG version 1.0

The framework was changed internally!

Current status:

- ug[SCIP,*] : FiberSCIP and ParaSCIP → working on UG version 1.0
- ug[SCIP-Jack,*] → working on UG version 1.0
- ug[Xpress, *] → working on UG version 1.0
- ug[qapNB, *] → working on UG version 1.0
- ug[PIPS-SBB, MPI] → out of development
- ug[Concorde, *] → under development on UG version 1.0
- MAP-SVP→ out of development, moved to CMAP-LAP
- CMPAP-LAP → working on UG version 1.0

What is Ubiquity Generator (UG) Framework– **Ver.1.0**

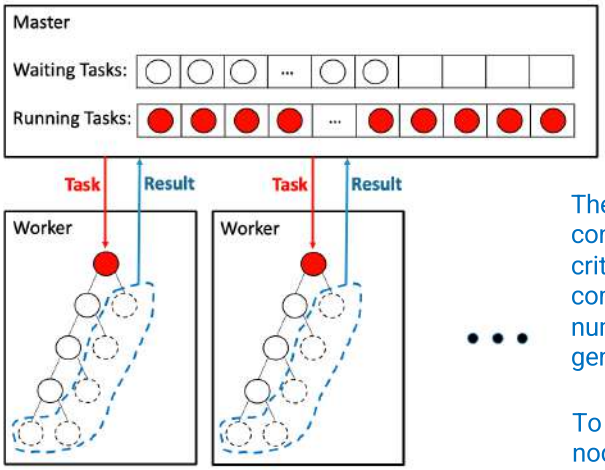
- UG is a **high-level task parallelization framework**
  - Can parallelize any kind of solver
    - which needs to share some data among running solvers
    - which needs to share them very flexibly
  - **Can parallelize Branch-and-bound based solvers,**
    - can parallelize
      fully branch-and-cut-and-price programs
      (Could not parallelize before, but ver. 1.0 can parallelize it)
    - can customize parallelization mechanism depending on
      - base solver used
      - purpose to have a solution
        (prove feasibility, prove optimality etc.)

Concept of UG's high-level task parallelization framework

- **Master-Worker paradigm : UG is NOT Master-Worker**
  - one of the most famous high-level task parallelization
  - **Task**: represents an operation that is running on a Worker
  - For parallel B&B, key message:
    - (new) Task: a (sub-)problem representation and current best incumbent value
    - Result: the best incumbent solution and all open nodes

Concept of UG's high-level task parallelization framework

- **Master-Worker paradigm : UG is NOT Master-Worker**



All open B&B nodes are managed by the Master

The **granularity** of a Task: controlled by a termination criterion for a (sub-)problem computation, such as the number of open nodes generated.

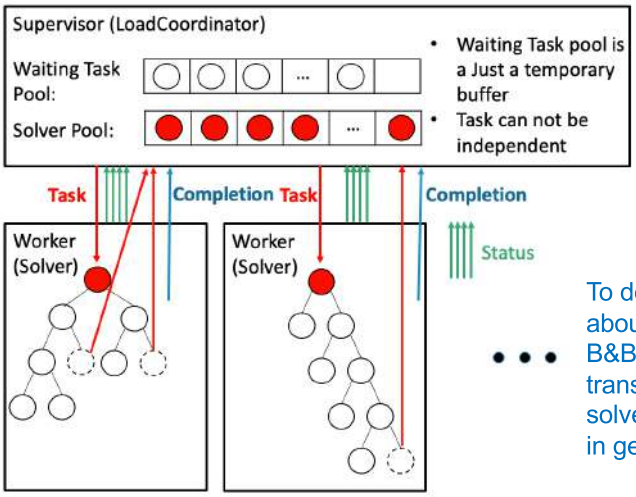
To reduce the number of open nodes, usually depth-first search is used in the Worker side.

Concept of UG's high-level task parallelization framework

- **Supervisor-Worker paradigm: UG's high-level task parallelization**
  - **Can define a very flexible message passing protocol**
  - For parallel B&B, key message:
    - **(new) Task**: a (sub-)problem representation, and it indicates the beginning of the Task computation
    - **Status**: the Task computation status and the notification frequency to Supervisor, which can be specified at run-time.
    - **Completion**: the termination of the task computation
  - In between the Task and Completion messages, any message passing protocol can be defined, for example
    - **Solution** : the best incumbent solution. Then, the solution can be shared whenever a single Solver found a new one.
    - **InCollecting** : indicates that the Supervisor needs new Tasks. This message allows to collect (sub-)problems on demand.
    - **OutCollecting** : indicates that the Supervisor does not need new Tasks.
    - **Interrupt** : indicates the current executing Task can be interrupted
    - etc.

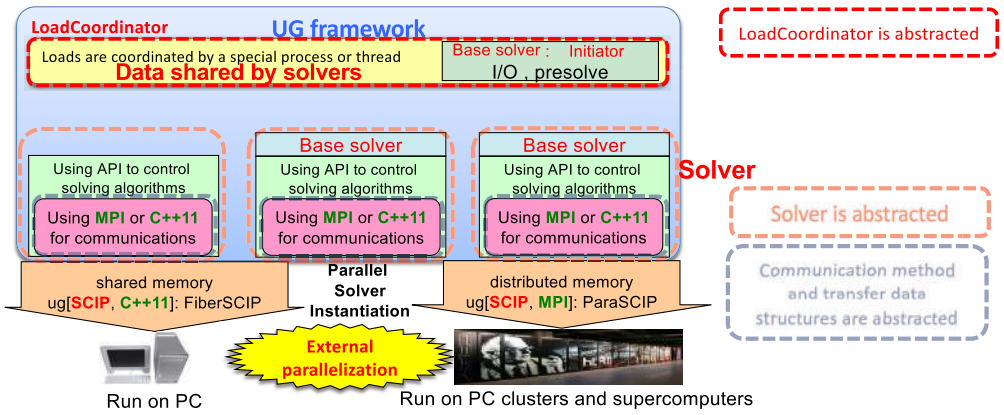
Concept of UG's high-level task parallelization framework

- **Supervisor-Worker paradigm: UG's high-level task parallelization**



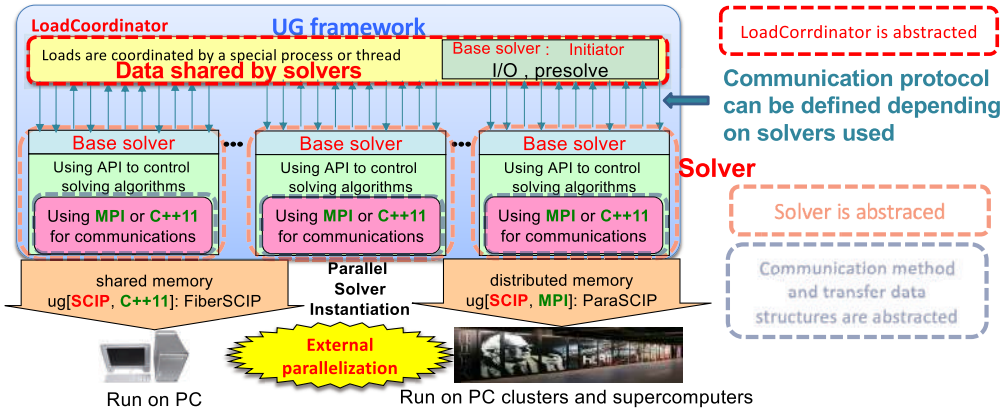
To do dynamic load balancing, about **1% (at most 2%)** of B&B nodes are transferred to the other solver during computation in general

LoadCoordinator is abstracted in UG ver.1.0



Allows to have data in LoadCoordinator(LC) that are shared by solvers, depending on the base solver used

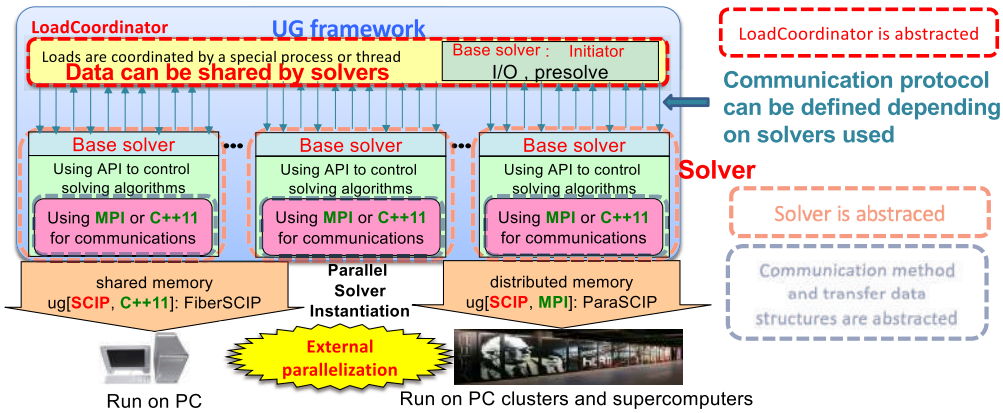
LoadCoordinator is abstracted in UG ver.1.0



Allows to have data in LoadCoordinator(LC) that are shared by solvers, depending on the base solver used

Can communicate the data in LoadCoordinator flexibly, by defining the communication protocol between LC and Solvers

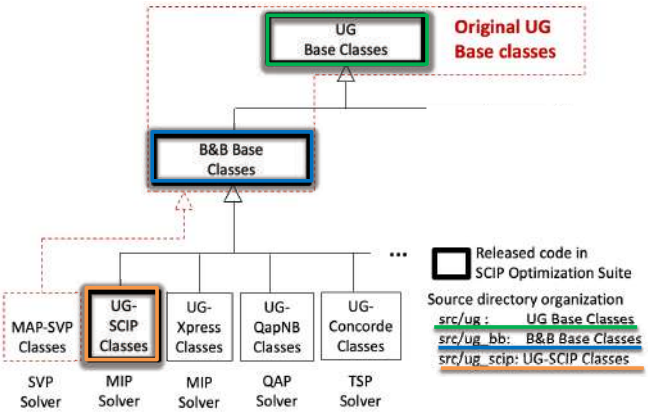
LoadCoordinator is abstracted in UG ver.1.0



See below how flexibly the messages can be defined:

N. Tateiwa, et al., "CMAP-LAP: Configurable Massively Parallel Solver for Lattice Problems," in 2021 IEEE 28th International Conference on High Performance Computing, Data, and Analytics (HiPC), Bengaluru, India, 2021 pp. 42-52.

Can define the communication protocols in each layer



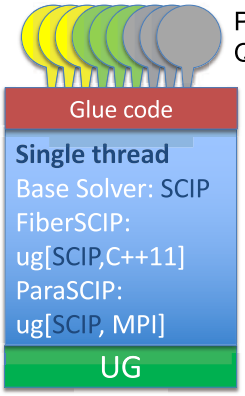
Communication protocols and parameter set for UG base

Communication protocols and parameter set for B&B base

Communication protocols and parameter set for UG-SCIP

Parallelization of the latest parallel QUBO solver

- Initial parallelization used ug[SICIP-*,*]-libraries



Plugins for QUBO solver

```
>> cloc ug/ug_scip_applications/QUBO/src/*
       1 text file.
       1 unique file.
       0 files ignored.

github.com/AlDanial/cloc v 1.92  T=0.00 s (231.2 files/s, 39538.4 lines/s)
-------------------------------------------------------------------------------
Language                     files          blank        comment           code
-------------------------------------------------------------------------------
C++                              1             22             36            113
-------------------------------------------------------------------------------
```

Faster exact solution of sparse MaxCut and QUBO problems
Daniel Rehfeldt, Thorsten Koch, Yuji Shinano
doi: 10.48550/arXiv.2202.02305

Parallelization of the latest parallel QUBO solver

- The last slide in Daniel Rehfeldt's talk at OR 2022

So how about Quantum annealers?

From the article
Quantum Annealing versus Digital Computing: An Experimental Comparison, Jünger et al., ACM J. Exp. Algorithmics, 2021:

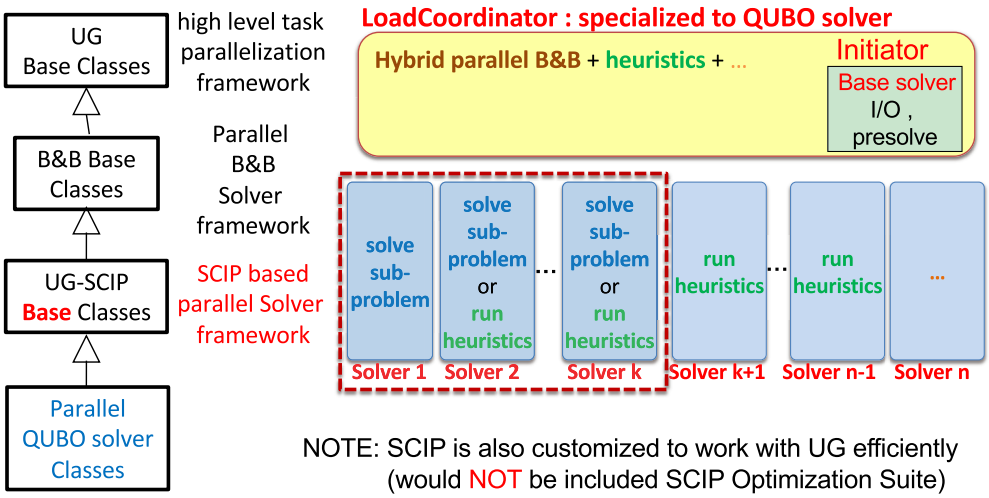
"However, we should stress the fact that exact optimization requires a lot of time to prove optimality, and thus it is not fair to compare their times with the heuristic times, but even with this additional burden, the exact algorithms are faster than D-Wave on a large portion of the sample".

- McSparse is faster than the (sparse) solver used by Jünger et. al.
- Our solver is faster than McSparse

→ If you need to solve a QUBO, better save the money for a D-Wave machine and use our solver instead!

Parallelization of the latest parallel QUBO solver

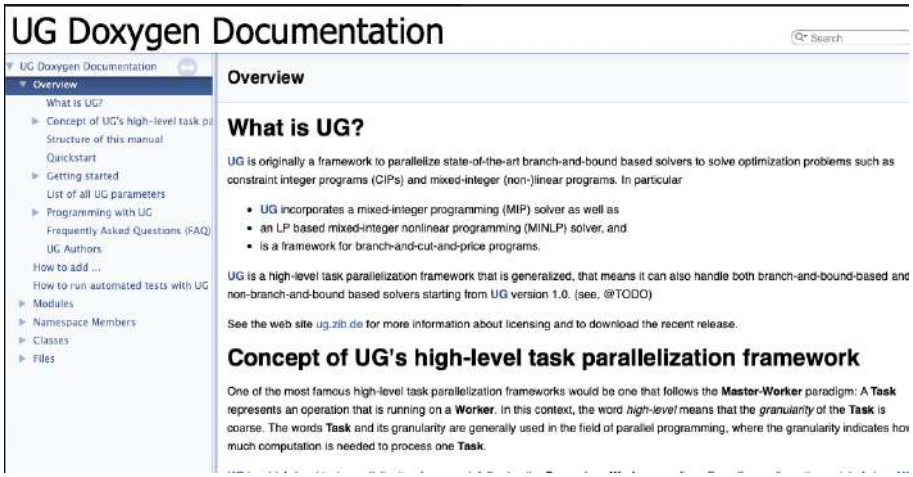
- A specialized parallel QUBO solver based on UG version 1.0



NOTE: SCIP is also customized to work with UG efficiently (would NOT be included SCIP Optimization Suite)

What comes with UG ver.1.0

- Documentation



- Cmake build system

Summary

- UG version 1.0 is a generalized UG
  - a high level task parallelization framework
    - can parallelize any kind of "base solver" (not only B&B based)
    - can parallelize branch-and-cut-and-price programs
  - can do special treatment for shared data and communication protocol depending on the "base solver" used

Included in SCIP Optimization Suite 8.0

A Parallel Algorithm Combining Relaxatino and heuristic for the integrated long-haul and local vehicle routing problem

Junko HOSODA*1, Stephen J. MAHER*2, Yuji SHINANO*3 and Honas Christoffer VILLUMSEN*4

*1 Controls and Robotics Innovation Center, Hitachi Ltd., Japan
junko.hosoda.dp@hitachi.com

*2 University of Exeter, United Kingdom *3 Zuse Institute Berlin, Germany *4 Hitachi Europe Ltd., Denmark

Commodity consolidation and vehicle route coordination are fundamental features of the logistics problem. This problem is called the supply chain service network design problem (SCSNDP); the SCSNDP includes three problems: the warehouse consolidation problem (WCP), the service network design problem (SNDP), and the pickup and delivery problem (PDP). To obtain high-quality solutions, a combined relaxation and heuristic algorithm is proposed[1]. The relaxation solver sets the boundaries of the solution space by considering the trend of the solution space. The heuristic solver finds a high-quality solution that satisfies all the constraints within the bounded solution space; using the UG framework, the relaxation and heuristic solvers are executed in parallel. The results show that the parallel execution of the relaxation and heuristic influences the quality of the SCSNDP solution.

References

[1] Junko Hosoda, Stepehn J. Maher, Yuji Shinano, and Jonas Christoffer Villumsen, “Location, transshipment and routing: An adaptive transportation network integrating long-haul and local vehicle routing”, EURO Journal on Transportation and Logistics, 100091, 2022, https://doi.org/10.1016/j.ejtl.2022.100091

HITACHI
Inspire the Next

A parallel branch-and-bound heuristic for the integrated long-haul and local vehicle routing problem on an adaptive transportation network

Junko Hosoda*1, Stephen J. Maher*2, Yuji Shinano*3, and Jonas Christoffer Villumsen *4

1 Hitachi Ltd., Japan
2 University of Exeter, United Kingdom
3 Zuse Institute Berlin, Germany
4 Hitachi Europe Ltd., Denmark

HITACHI
Inspire the Next

Contents

HITACHI
Inspire the Next

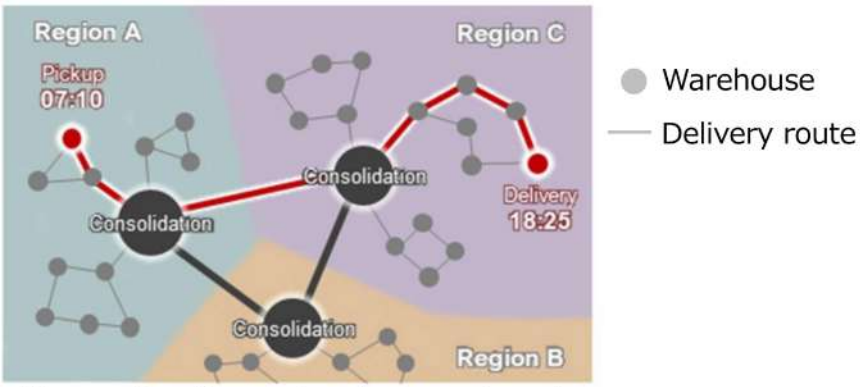
Contents

1-1. Supply chain network

HITACHI
Inspire the Next

[Target] Supply chain network for delivery business

- Commodities are picked up and delivered to.
- Commodities are delivered by land transportation.
- Commodities are delivered within 48 hours.



Example of supply chain network

1-2. Purpose and issues

HITACHI
Inspire the Next

[Purpose]

Deliver all commodities on time at **lower delivery costs**

[Issues]

Design supply chain network and determine vehicle routes to reduce delivery costs.

- Supply chain network design includes determining
  - Number of Regions
  - Warehouse included in each region
  - Consolidation location for each region
- Vehicle routes are determined integrating long-haul and local deliveries

 4

HITACHI
Inspire the Next

Contents

 5

2-1. Definition of the SCSNDP

HITACHI
Inspire the Next

[SCSNDP]

The supply chain service network design problem

[Definition of the SCSNDP]

Given items: A **collection of warehouses and commodities** to be distributed between warehouses

Decision items: **Warehouse clusters and consolidation locations**

Objective: **To minimise the cost** of synchronised intra- and inter-cluster routes

Constraints: **To satisfy all pickup and delivery requests**

 6

2-2. Aim of project

HITACHI
Inspire the Next

[Previous work]

Developed an **iterative heuristic and multi-armed bandit algorithm** to find solutions to the SCSNDP.

[Aims]

- Develop a mathematical programming problem to model the integrated SCSNDP
- **Find lower bounds** for the SCSNDP to assess solution quality
- Using a mathematical programming-based approach, **develop a parallel algorithm** that can find higher quality solutions

 7

2-3. Mathematical modelling - WCP

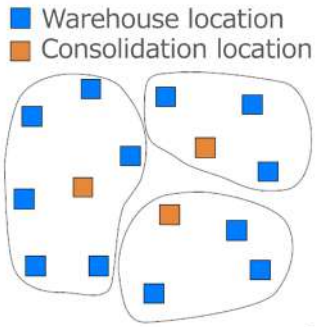
HITACHI
Inspire the Next

The SCSNDP is the integration of three separate problems.

[Problems]

(a) Warehouse clustering problem (WCP)

Identify clusters of warehouses to minimise a distance function



 8

2-4. Mathematical modelling - SNDP

HITACHI
Inspire the Next

The SCSNDP is the integration of three separate problems.

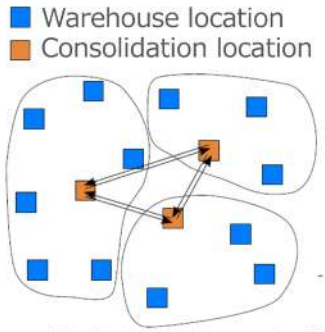
[Problems]

(a) Warehouse clustering problem (WCP)

Identify clusters of warehouses to minimise a distance function

(b) Service network design problem (SNDP)

Identify a transportation schedule on inter-cluster routes



 9

2-5. Mathematical modelling - PDP

HITACHI
Inspire the Next

The SCSNDP is the integration of three separate problems.

[Problems]

(a) Warehouse clustering problem (WCP)

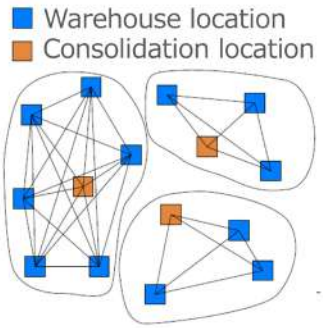
Identify clusters of warehouses to minimise a distance function

(b) Service network design problem (SNDP)

Identify a transportation schedule on inter-cluster routes

(c) Pickup and delivery problem (PDP)

Identify intra-cluster vehicle routes.



 10

2-6. Mathematical modelling - Integration

HITACHI
Inspire the Next

The SCSNDP is the integration of three separate problems.

[Policies for problems integration]

- Unable to model the complete integrated problem
- Modelled a MIP relaxation of the SCSNDP
- Comprising: Complete WCP and SNDP, PDP relaxed to a packing problem
- Relaxation is the basis of the parallel heuristic design.

 11

2-7. MIP relaxation model – Decision variables

HITACHI Inspire the Next

Complete WCP and SNDP, PDP relaxed to a packing problem

[Decision variables]

$\bar{T}_i^v$: Departure time from consolidation location i of vehicle v

$\hat{T}_i^v$: Arrival time at consolidation location i of vehicle v

u_{ki}^v: Binary variable
It is 1 if the pickup or delivery of commodity k is assigned to vehicle v that departs from consolidation location

α_i^v: Binary variable
It is 1 if vehicle v departs from consolidation location

T_i^v: Arrival time at location i of vehicle v

h_k^v: Binary variable
It is 1 if vehicle v picks up commodity k

 12

2-8. MIP relaxation model – Arrival time

HITACHI Inspire the Next

Complete WCP and SNDP, PDP relaxed to a packing problem

[Arrival time constraint]

[PDP]

$$\left(T_i^v + g_i + tt_{ij}\right)x_{ij}^v \le T_j^v \quad \forall (i,j) \in A, \forall v \in V$$

$$T_j^v \le T^v \quad \forall j \in N, \forall v \in V$$

A:Paths
V:Vehicles
g_i:Processing time
tt_{ij}:Travel time
T^v:Work time

[PDP relaxed to a packing problem]

$$\bar{T}_i^v + \sum_{k \epsilon K} \hat{tt}_k u_{ki}^v \le \hat{T}_j^v \quad \forall i \in N, \forall v \in V$$

$$\sum_{k \epsilon K} \hat{tt}_k u_{ki}^v \le T^v \alpha_i^v \quad \forall i \in N, \forall v \in V$$

K:Commodities
N:Locations
V:Vehicles
$\hat{tt}_k$:Shortest travel time

 13

2-9. MIP relaxation model – Load capacity

HITACHI
Inspire the Next

Complete WCP and SNDP, PDP relaxed to a packing problem

[Load capacity constraint]

[PDP]

$$Q_i^v + \sum_{k \epsilon K, o_k = i} q_k h_k^v - \sum_{k \epsilon K, d_k = i} q_k h_k^v \leq Q_j^v \quad \forall (i,j) \in A, \forall v \in V$$

$$0 \leq Q_i^v \leq Q^v \quad \forall i \in N, \forall v \in V$$

A:Paths, V:Vehicles, q_k:Weight

[PDP relaxed to a packing problem]

$$\sum_{k \epsilon K} q_k u_{ki}^v \leq Q^v \alpha_i^v \quad \forall i \in N, \forall v \in V$$

N:Locations
V:Vehicles
q_k:Weight
Q^v:Load capacity

 14

HITACHI
Inspire the Next

Contents

 15

3-1. Solution algorithm

HITACHI
Inspire the Next

Two algorithms are developed.

[Sequential algorithm]

- Solve relaxation as a MIP
- Using callback functions, execute iterative heuristic to find feasible solutions to the SCSNDP

[Parallel algorithm - 2 modes]

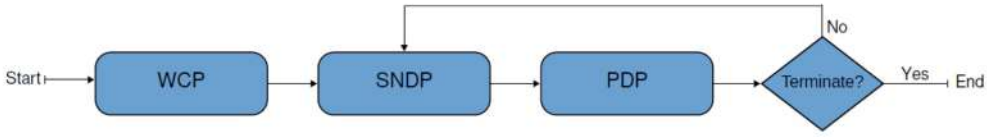
- **Racing mode** - execute the sequential algorithm on n solvers using different random seeds
- **Parallel heuristic mode** - solve the relaxation on one or more solvers, execute heuristic on different solvers

 16

3-2. Iterative heuristic

HITACHI
Inspire the Next

Input: (partial) fixing of consolidation locations



Solve WCP to find warehouse clustering

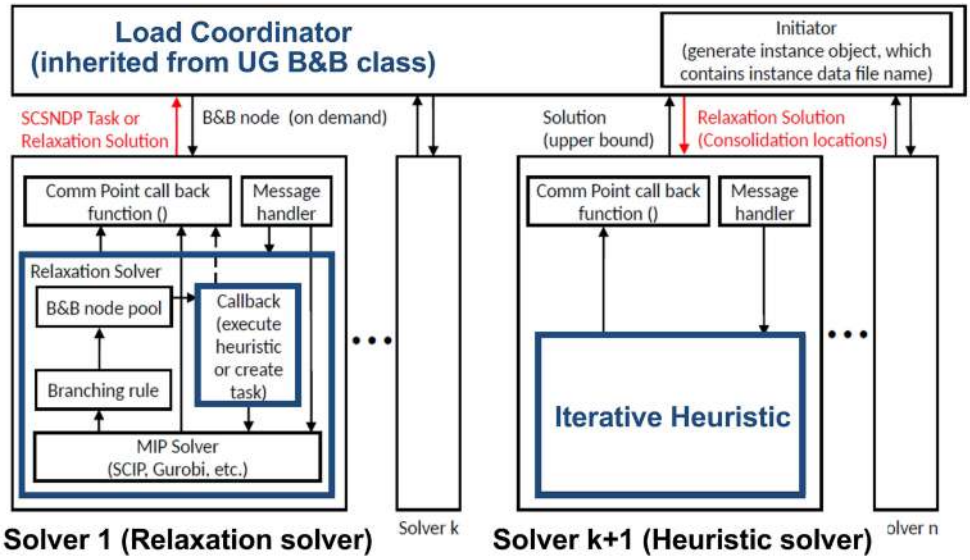
Iterate between SNDP and PDP to identify synchronised intra- and inter-cluster routes.

 17

3-3. Basic parallel architecture

HITACHI
Inspire the Next

Each solver is fixed as Relaxation solver or Heuristic solver.

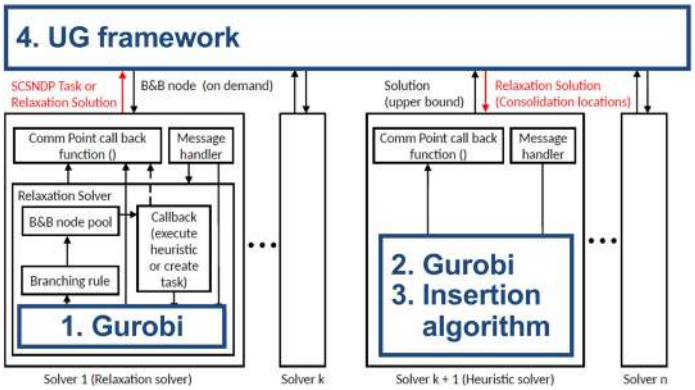


 18

3-4. Overview of implementation

HITACHI
Inspire the Next

1. Relaxation is solved by Gurobi
2. WCP and SNDP are solved directly as MIP, using Gurobi
3. A purpose-built insertion algorithm used to solve the PDP
4. The UG framework is used to implement the parallel architecture

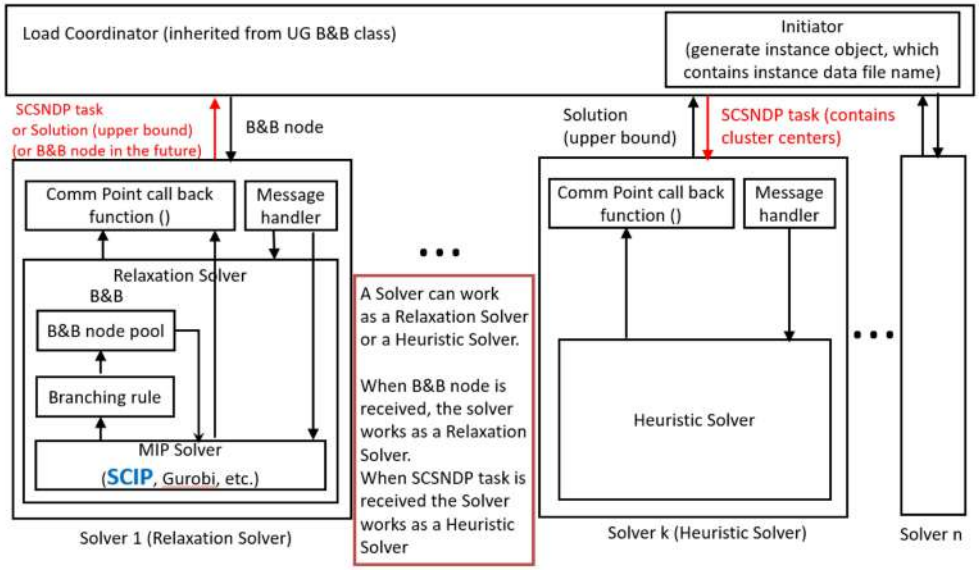


 19

3-5. Improved parallel architecture

HITACHI
Inspire the Next

A solver works as a Relaxation solver or Heuristic solver.



 20

3-6. Parallel architecture for 2 stages

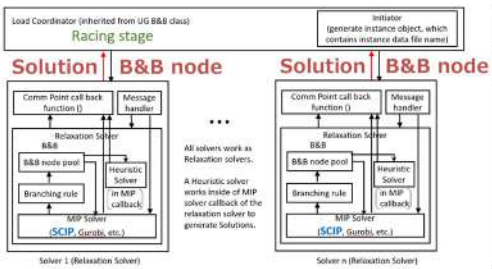
HITACHI
Inspire the Next

[Racing stage]

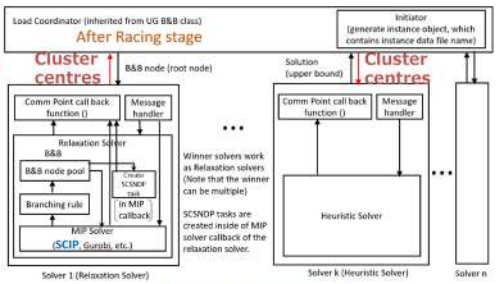
Execute the sequential algorithm on n solvers using different random seeds

[After racing stage]

Solve the relaxation on one or more solvers, execute heuristic on different solvers



All solvers work as Racing solver.



Solvers work as Racing solver or Heuristic solver.

 21

HITACHI
Inspire the Next

Contents

4-1. Problem instances

HITACHI
Inspire the Next

- A set of 135 randomly generated instances used for computational experiments
- Instances referenced by (N,K,C).

  N:Number of warehouses $N \in \{5,10,25\}$

  K:Number of commodities $K \in \{N, 2N, 4N\}$

  C:Number of subregions for N=5,$C \in \{1,2\}$ for N=10, $C \in \{1,2,3\}$ for N=25,$C \in \{1,2,5\}$
- Homogeneous set of vehicle

Parameter Type	Description
Planning horizon	2 days.
Business hours	6:00 until 20:00 each day.
Warehouse locations	Either randomly within a square or within subregions.
Pickup time window	selected uniformly at random between 6:00 and 22:00, with a length between 2 and 18 hours.
Delivery time window	selected uniformly at random between 6:00 and 44:00, with a length between 2 and 18 hours.

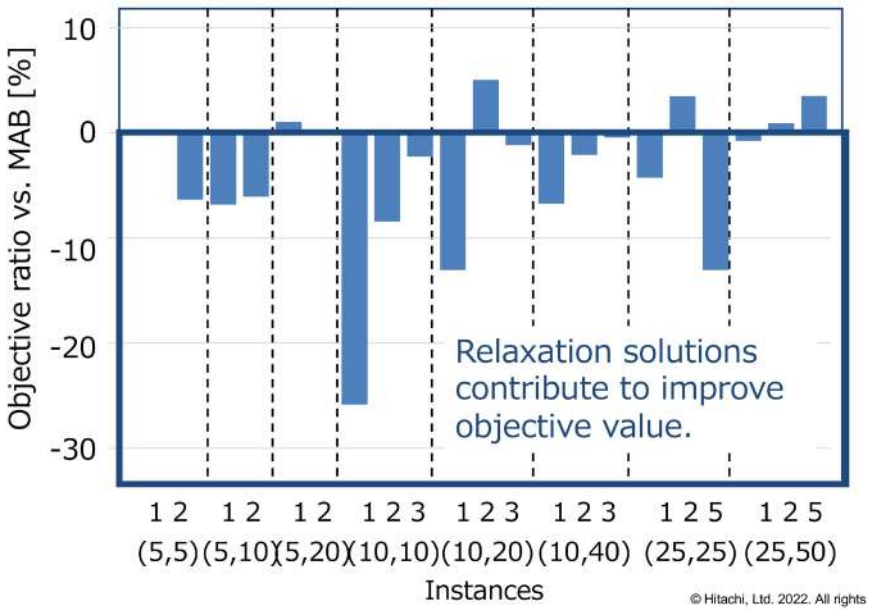
4-2. Computational conditions

HITACHI
Inspire the Next

- Parallel algorithm uses **8 threads**.
- Racing solvers are provided with different random seeds.
- Two parallel algorithms: Racing and Parallel heuristic.
- Racing terminates after all solvers execute **5 heuristics**, **3 winners** are selected in the Parallel heuristic algorithm.
- Benchmark is multi-armed bandit algorithm from previous work.
- Multi-armed bandit, sequential and parallel algorithm have a **time limit of 7,200 seconds**.

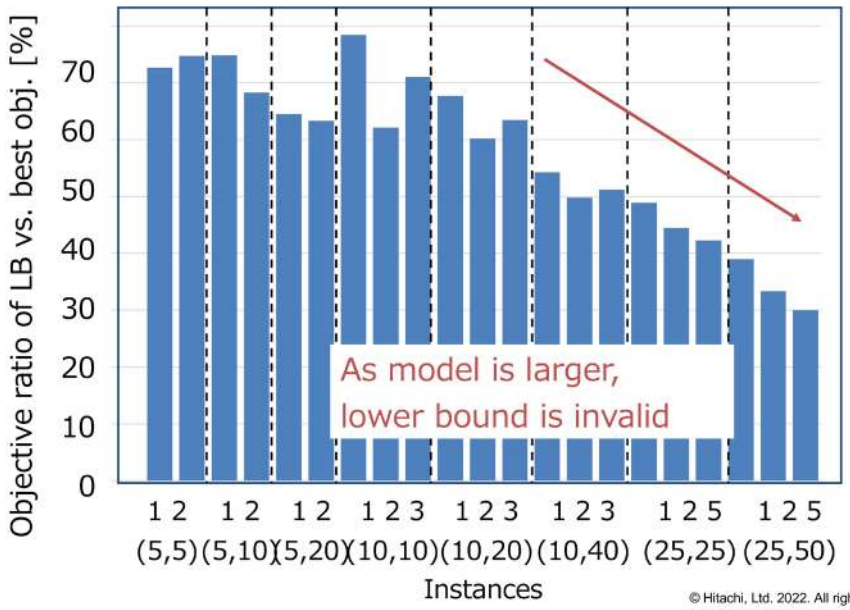
4-3. Objective improvement – Sequential algorithm

HITACHI
Inspire the Next



4-4. Lower bound – Sequential algorithm

HITACHI
Inspire the Next

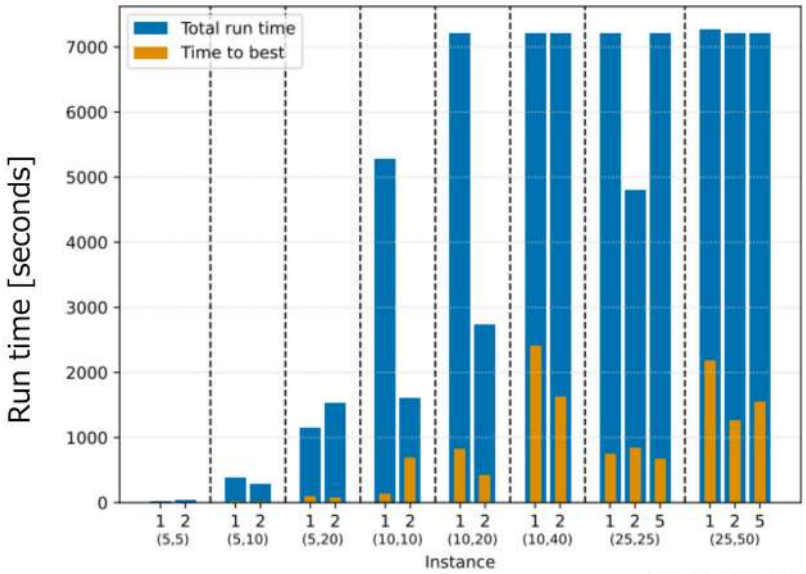


 26

4-5. Run time – Sequential algorithm

HITACHI
Inspire the Next

Time to best is within 2,400 sec.

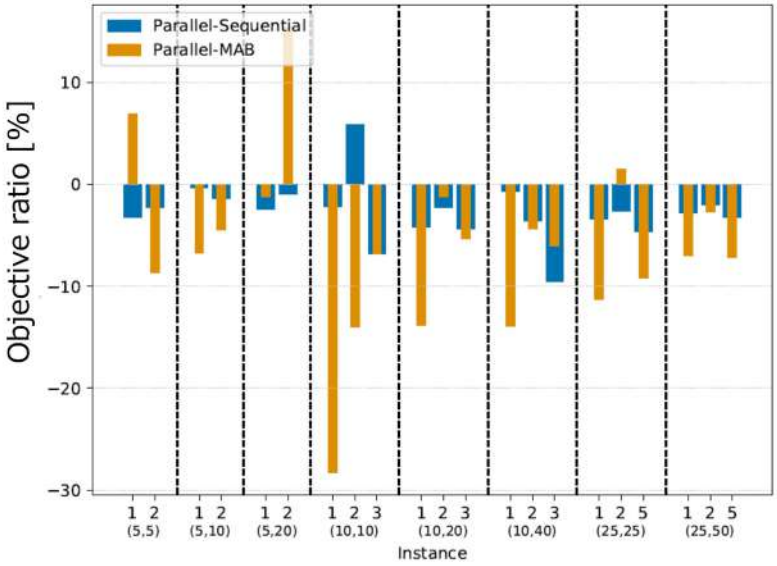


 27

4-6. Objective improvement – Parallel racing algorithm

HITACHI
Inspire the Next

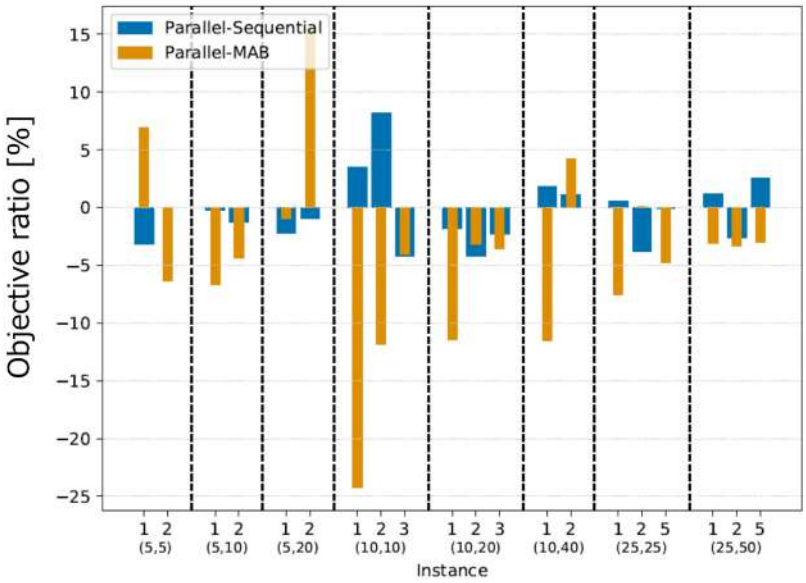
Racing finds better solution than Sequential algorithm



 28

4-7. Objective improvement – Parallel heuristic algorithm

HITACHI
Inspire the Next



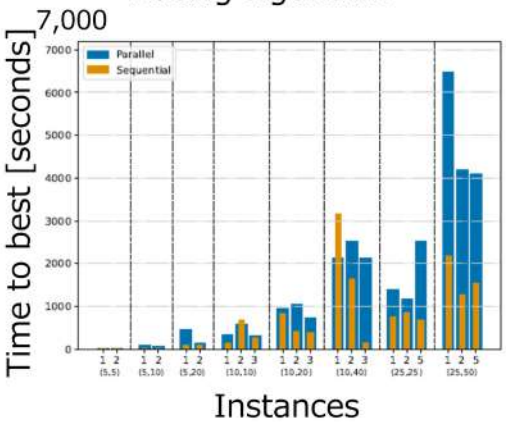
 29

4-8. Time to best
- Racing and Parallel heuristic algorithm

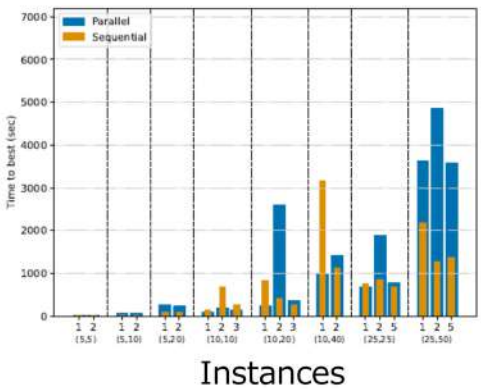
HITACHI
Inspire the Next

Parallel heuristic tends to be shorten time to best than Racing

Racing algorithm



Parallel heuristic algorithm



HITACHI
Inspire the Next

Contents

5. Conclusions

HITACHI
Inspire the Next

- Investigate the combination of intra- and inter-cluster transportation for movement of commodities between warehouses

- Developed a mathematical programming model for adaptive transportation network combining clustering, transshipment and routing.

- Sequential and parallel algorithms developed to find high quality solutions to the SCSNDP.

- Demonstrated the benefits of using a mathematical programming approach for guiding the heuristic algorithm.

 32

Solving Large Scale QAPs by Massively Parallel DNN-based Branch-and-bound Method

Koichi FUJII

NTT DATA Mathematical Systems Inc. Japan
fujii@msi.co.jp

We report our progress on the project for solving large scale quadratic assignment problems (QAPs). Our main approach to solve QAPs is a parallel branch-and-bound method efficiently implemented on a powerful computer system, using the Ubiquity Generator Framework (UG) which can utilize more than 100,000 cores ([1]). Newton-bracketing method, the method we utilize to solve Lagrangian doubly nonnegative (DNN) relaxation of subproblems of QAPs, gives strong lower bounds, but it requires more computational time ([2]) which makes difficult to scale in parallelization. We have added some new features to UG such as Enhanced Checkpoint or Huge Checkpoint File Split to overcome these obstacles. In this talk, we describe the details of new features of UG for solving QAPs and present some preliminary numerical results of solving large QAPs on supercomputers ([3]).

References

[1] Shinano, Yuji. "The ubiquity generator framework: 7 years of progress in parallelizing branch-and-bound." Operations Research Proceedings 2017. Springer, Cham, 2018. 143-149.

[2] Kim, Sunyoung, Masakazu Kojima, and Kim-Chuan Toh. "A Newton-bracketing method for a simple conic optimization problem." Optimization Methods and Software 36.2-3 (2021): 371-388.

[3] Fujii, Koichi, Naoki Ito, Sunyoung Kim, Masakazu Kojima, Yuji Shinano, and Kim-Chuan Toh. "Solving challenging large scale qaps." arXiv preprint arXiv:2101.09629 (2021).

The 6th RIKEN-IMI-ISM-NUS-ZIB-MODAL-NHR Workshop on Advances in Classical and Quantum Algorithms for Optimization and Machine Learning
September 17 2022
Tokyo Japan

NTT DATA
NTT DATA Mathematical Systems Inc.

Solving Large Scale QAPs by Massively Parallel DNN-based Branch-and-bound Method

Koichi Fujii, Naoki Ito, Sunyoung Kim, Masakazu Kojima, Hans Mittelmann, Yuji Shinano, Kim-Chuan Toh

Summary: DNN-based Branch-and-bound for the Quadratic Assignment Problem

- **Motivation**
  - Quadratic assignment problems (QAPs) remain as one of the most difficult combinatorial problems
  - Recent conic relaxation technique updates the lower bounds of open QAP instances.
- **Goal**
  - Solve all open instances of QAPLIB
- **Our Results**
  - Our DNN-based branch-and-bound solver solved three open instances(tai30a, sko42, tho40).

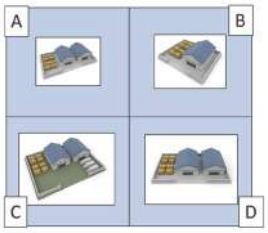
NTT DATA
NTT DATA Mathematical Systems Inc.

Agenda

- Overview
- QapNB:
  Newton-bracketing method + branch-and-bound
- ParaQapNB :
  Newton-bracketing method + parallelized branch-and-bound
- New features of UG for ParaQapNB
- Computational Results

Overview: Quadratic Assignment Problem

- Problem Description
  - f_{ij} : flow from facility i to facility j
  - d_{kl} : distance from location k to location l
  - Assign each facility to each location to minimize flow × distance



$n = 4$, Flow f_{ij}				
	j			
i	1	2	3	4
1	0	5	4	2
2	4	0	2	5
3	2	3	0	4
4	1	5	3	0

Distance $d_{k\ell}$				
	ℓ			
k	1	2	3	4
1	0	6	7	8
2	6	0	9	6
3	7	9	0	8
4	8	6	8	0

Assignment				
i	1	2	3	4
k	k_1	k_2	k_3	k_4
	‖	‖	‖	‖
	4	1	3	2

$$\min_{\sigma \in S_n} \sum_{i,j} f_{ij} d_{\sigma_i \sigma_j}$$

n! feasible solutions
40! = 8.2e47 --- too large combination

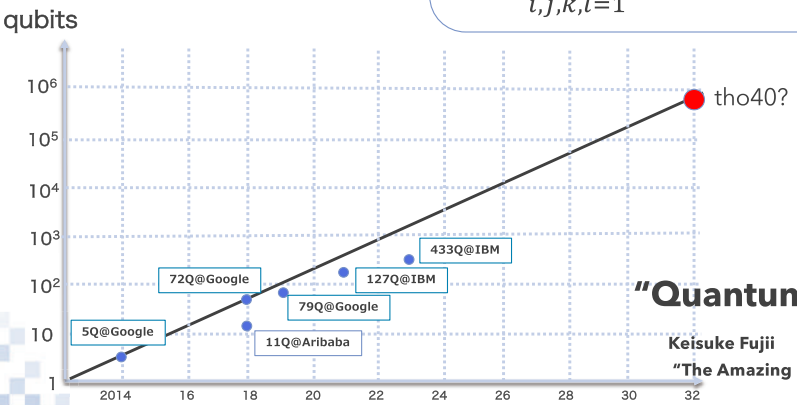
Overview: Quadratic Assignment Problem

$$\min_{\sigma \in S_n} \sum_{i,j} f_{ij} d_{\sigma_i \sigma_j}$$

QUBO formulation

size = n^2 ; $O(n^4)$ physical qubits

$$\min_{x} \sum_{i,j,k,l=1}^{n} Q_{(i,j)(k,l)} x_{(i,j)} x_{(k,l)}$$



"Quantum" Moore law

Keisuke Fujii
"The Amazing Quantum Computer"(2020)

NTT DATA
NTT DATA Mathematical Systems Inc.

Overview: DNN relaxation and Newton-bracketing method

QAP(01-QOP)

$$\zeta^* := \min \left\{ \langle \boldsymbol{B} \otimes \boldsymbol{A},\ \boldsymbol{x}\boldsymbol{x}^T \rangle \;\middle|\; \begin{array}{l} \boldsymbol{x} \in \{0,1\}^m \\ (\boldsymbol{I} \otimes \boldsymbol{e}^T)\boldsymbol{x} = (\boldsymbol{e}^T \otimes \boldsymbol{I})\boldsymbol{x} = \boldsymbol{e} \end{array} \right\}.$$

$$\Downarrow$$

Lagrangian DNN relaxation of QAP

$$\text{P: } \eta^\lambda = \min \left\{ \langle \boldsymbol{Q}^\lambda,\ \boldsymbol{X} \rangle : \ \boldsymbol{X} \in \mathbb{K}_1 \cap \mathbb{K}_2, \langle \boldsymbol{H}^0,\ \boldsymbol{X} \rangle = 1 \ (X_{00} = 1) \right\}.$$

$$\mathbb{K}_1 \quad : \quad \text{semidefinite cone} \tag{1}$$

$$\mathbb{K}_2 \quad := \quad \left\{ \boldsymbol{X} \in \mathbb{S}^{1+m} : \begin{array}{l} \boldsymbol{X}_{\alpha\beta} \geq 0 \ (\text{nonnegative}) \\ \boldsymbol{X}_{0\alpha} = \boldsymbol{X}_{\alpha 0} = \boldsymbol{X}_{\alpha\alpha} \end{array} \right\} \tag{2}$$

NTT DATA
NTT DATA Mathematical Systems Inc.

Overview: DNN relaxation and Newton-bracketing method

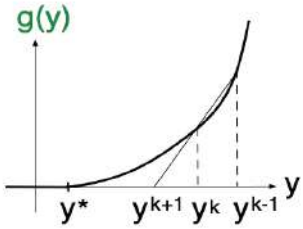
- ζ^* : The optimal value of QAP
- y^* : The optimal value of Lagrangian DNN relaxation; $y^* \leq \zeta^*$.

Newton-bracketing method

Lagrangian DNN relaxation problem can be regarded as minimization problem of convex function $g : \mathbb{R} \to \mathbb{R}_+$.

- Newton method (+ secant method)
  $y^* \leftarrow y^{r+1} \leq y^r \leq \cdots \leq y^0$.
- (Valid) lower bound
  $\ell^1 \leq \cdots \leq \ell^r \leq \ell^{r+1} \to y^*$

$y^* \in [\ell^r, y^r]$, $|y^r - \ell^r| \to 0$.



[Sunyoung, Kojima, and Toh. (2021)]
"A Newton-bracketing method for a simple conic optimization problem."

Overview: DNN relaxation and Newton-bracketing method

Miguel Anjos, PhD, FCAE, FEUROPT, SMIEEE

Miguel.F.Anjos@ed.ac.uk

Home | Bio | Events | Recognition | Research | Consulting | Service | Supervision | Other

QAPLIB is a Quadratic Assignment Problem Library

Four previously unsolved QAPLIB instances have been solved to optimality in recent years: see tai30a, tai35b, tai40b, and sko42.

We present below the 28 instances that have not yet been solved to optimality; recent developments are highlighted. Have you found a better solution or bound? Send me an email!

Thonemann and Bölte (1994)	Feasible solution	Bound	Gap	Software
Tho40 (n = 40)	240516	228079	5.17%	NewtBracket
Tho150 (n = 150)	8133398	7854894	3.42%	NewtBracket

Wilhelm and Ward (1987)	Feasible solution	Bound	Gap	Software
Wil50 (n = 50)	48816	48245	1.17%	NewtBracket
Wil100 (n = 100)	273038	268955	1.50%	BBCPOP

https://www.miguelanjos.com/qaplib

Overview: Computation Results

	problem	Opt.val	#node	time(sec)	No. of CPU cores used
	nug30	6,124	26,181	3.14e3	1,728
2021	tai30a	1,818,146	34,000,579	5.81e5 ≈ 6.8 days	1,728
	tai35b	283,315,445	2,620,547	2.49e5	1,728
	tai40b	637,250,948	278,465	1.05e5	1,728
2021	sko42	15,812	6,019,419	5.12e5 ≈ 5.9 days	5,184
2022	tho40	240,516	139,077,975	1.66e6 ≈ 19.1 days	24,528

Table: Computational results on large scale QAPs

- HPE SGI 8600 (384 nodes, 13,824 cores)
- HLRN-IV System (1270 nodes, 230,000 cores)

Newton-bracketing method

QAP(01-QOP)

$$\zeta^* := \min \left\{ \langle \boldsymbol{B} \otimes \boldsymbol{A},\ \boldsymbol{x}\boldsymbol{x}^T \rangle \;\middle|\; \begin{array}{l} \boldsymbol{x} \in \{0,1\}^m \\ (\boldsymbol{I} \otimes \boldsymbol{e}^T)\boldsymbol{x} = (\boldsymbol{e}^T \otimes \boldsymbol{I})\boldsymbol{x} = \boldsymbol{e} \end{array} \right\}.$$

$$\Downarrow$$

Lagrangian DNN relaxation of QAP

$$\text{P: } \eta^\lambda = \min \left\{ \langle \boldsymbol{Q}^\lambda,\ \boldsymbol{X} \rangle : \ \boldsymbol{X} \in \mathbb{K}_1 \cap \mathbb{K}_2, \langle \boldsymbol{H}^0,\ \boldsymbol{X} \rangle = 1\ (X_{00} = 1) \right\}.$$

$$\mathbb{K}_1 \quad : \quad \text{semidefinite cone} \tag{1}$$

$$\mathbb{K}_2 \quad := \quad \left\{ \boldsymbol{X} \in \mathbb{S}^{1+m} : \begin{array}{l} \boldsymbol{X}_{\alpha\beta} \geq 0 \text{ (nonnegative)} \\ \boldsymbol{X}_{0\alpha} = \boldsymbol{X}_{\alpha 0} = \boldsymbol{X}_{\alpha\alpha} \end{array} \right\} \tag{2}$$

Newton-bracketing method

P: $\eta^\lambda = \min\{\langle \boldsymbol{Q}^\lambda, \boldsymbol{X}\rangle : \boldsymbol{X} \in \mathbb{K}_1 \cap \mathbb{K}_2, \langle \boldsymbol{H}^0, \boldsymbol{X}\rangle = 1\ (X_{00} = 1)\}$

$\Updownarrow$ Strong duality

D: $y^* = \max\{y \in \mathbb{R} : \underbrace{\boldsymbol{Q}^\lambda - \boldsymbol{H}^0 y}_{\boldsymbol{G}(y)} \in (\mathbb{K}_1 \cap \mathbb{K}_2)^* \equiv \mathbb{K}_1^* + \mathbb{K}_2^*\} = \eta^\lambda$,

Newton-bracketing method

Step 0: Start with upper bound $y^0 > y^*$.

Step 1: Obtain $(\boldsymbol{X}, \boldsymbol{Y}_1, \boldsymbol{Y}_2) \in (\mathbb{K}_1 \cap \mathbb{K}_2) \times \mathbb{K}_1^* \times \mathbb{K}_2^*$ satisfying KKT condition by APGR method.

Step 2: Update y^k by Newton iteration $y^{k+1} := y^k - \frac{g(y^k)}{g'(y^k)}$ or secant iteration.

Step 3: $\mu \leftarrow$ min eig. of $\boldsymbol{G}(y^k) - \boldsymbol{Y}_2$, $\ell^k \leftarrow \max\{y^k + \rho\mu, \ell^k\}$

Step 4: Obtain $[\ell^k, y^k](\ni \eta^\lambda)$, $k \leftarrow k+1$, goto Step1.

Newton-bracketing method

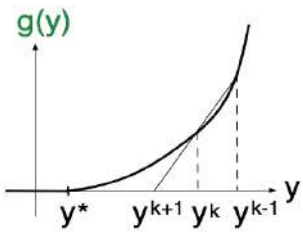
- ζ^* : The optimal value of QAP
- y^* : The optimal value of Lagrangian DNN relaxation; $y^* \le \zeta^*$.

Newton-bracketing method

Lagrangian DNN relaxation problem can be regarded as minimization problem of convex function $g : \mathbb{R} \to \mathbb{R}_+$.

- Newton method (+ secant method)
  $y^* \leftarrow y^{r+1} \le y^r \le \cdots \le y^0$.
- (Valid) lower bound
  $\ell^1 \le \cdots \le \ell^r \le \ell^{r+1} \to y^*$

$y^* \in [\ell^r, y^r]$, $|y^r - \ell^r| \to 0$.



[Sunyoung, Kojima, and Toh. (2021)]
"A Newton-bracketing method for a simple conic optimization problem."

QapNB:Newton-bracketing method+branch-and-bound

QAP

$$\zeta^* := \min \left\{ \langle \boldsymbol{B} \otimes \boldsymbol{A},\ \boldsymbol{x}\boldsymbol{x}^T \rangle \;\middle|\; \begin{array}{l} \boldsymbol{x} \in \{0,1\}^m \\ (\boldsymbol{I} \otimes \boldsymbol{e}^T)\boldsymbol{x} = (\boldsymbol{e}^T \otimes \boldsymbol{I})\boldsymbol{x} = e \end{array} \right\}.$$

(1) Incumbent solution

- Apply robust tabu search method（[Tailard 1991]）to obtain global upper bound $\hat{\zeta}$.

(2) Stop condition of Newton-bracketing method

[Condition 1] Stop if lower bound of DNN ℓ^k exceeds $\hat{\zeta}$
$\rightarrow$ prune the node

[Condition 2] Stop if upper bound of DNN y^k gets below $\hat{\zeta}$
$\rightarrow$ branch on the node

(3) Branching strategy (Polytomic branching)

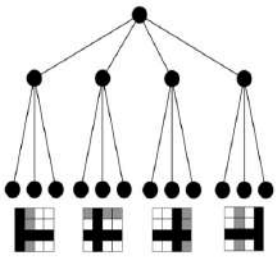
QapNB:Newton-bracketing method+branch-and-bound

QAP

$$\zeta^* := \min \left\{ \langle \boldsymbol{B} \otimes \boldsymbol{A},\ \boldsymbol{x}\boldsymbol{x}^T \rangle \;\middle|\; \begin{array}{l} \boldsymbol{x} \in \{0,1\}^m \\ (\boldsymbol{I} \otimes \boldsymbol{e}^T)\boldsymbol{x} = (\boldsymbol{e}^T \otimes \boldsymbol{I})\boldsymbol{x} = e \end{array} \right\}.$$

(3) Branching strategy (Polytomic branching)

1. Select facility $f \rightarrow$ fix $x_{f,l} = 1$ for $\forall \ell$.
2. Select location $\ell \rightarrow$ fix $x_{f,l} = 1$ for $\forall f$.



QapNB:Newton-bracketing method+branch-and-bound

Branching strategy

Estimate the score $\varphi(f,\ell)$ for unfixed facility f and location ℓ.

average branching

$\varphi(f,\ell) :=$ avg. obj. value of feasible solutions in $QAP(F \cup f, L \cup \ell)$

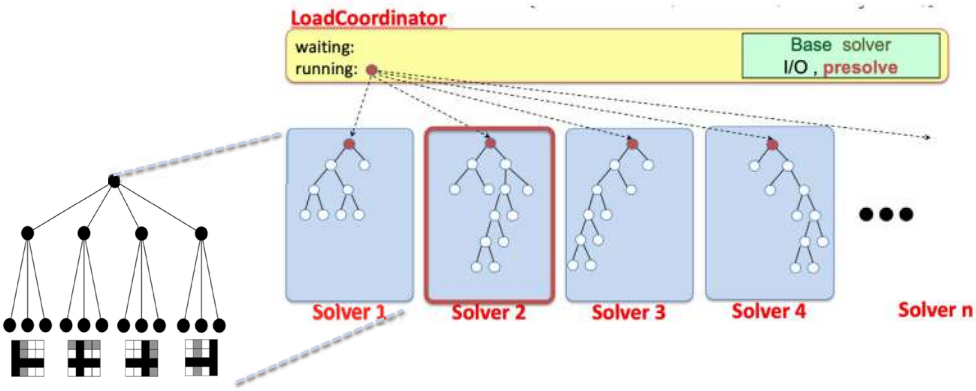
primal DNN branching

Step1. Obtain primal feasible solution of DNN Lagrangian relaxation $\widehat{\boldsymbol{X}} := \boldsymbol{X}/\langle \boldsymbol{H}^0,\ \boldsymbol{X}\rangle$.

Step2. Project primal solution $\widehat{\boldsymbol{X}}$ onto subproblem space : $\widehat{\boldsymbol{X}} \mapsto \widehat{\boldsymbol{X}}(f,\ell)$

Step3. $\varphi(f,\ell) := \langle \boldsymbol{Q}^0((F \cup f),(L \cup \ell)),\ \widehat{\boldsymbol{X}}(f,\ell)\rangle$.

ParaQapNB: Newton-bracketing method + parallelized branch-and-bound

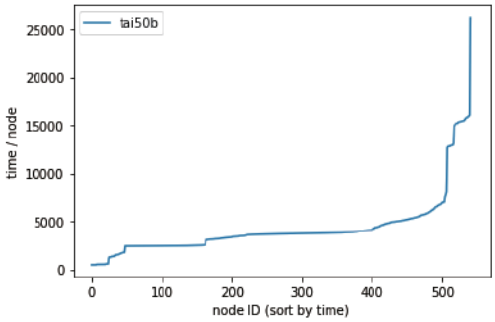


- DNN (doubly nonnegative cone) -based branch-and-bound
- Special criteria of Newton-bracketing method
- Primal heuristics
- Branching rule

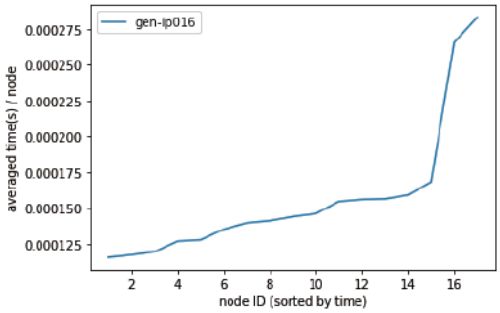
Koichi Fujii, Naoki Ito, Sunyoung Kim, Masakazu Kojima, Yuji Shinano, Kim-Chuan Toh "Solving Challenging Large Scale QAPs." (2021). [arXiv:2101.09629]

ParaQapNB: Newton-bracketing method + parallelized branch-and-bound

- Time/node of DNN-based branch-and-bound >> Time/node of LP-based branch-and-bound
  - hard to parallelize
  - hard to debug



time per node of tai50b (QAP) by ParaQapNB

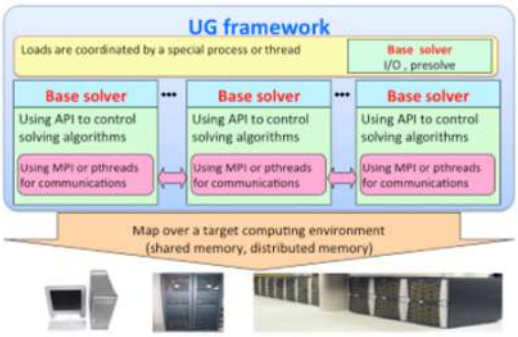


(averaged) time per node of gen-ip016(IP) by ParaNUOPT

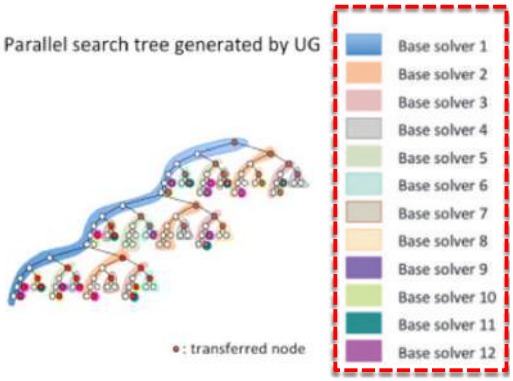
UG: Ubiquity Generator (UG) Framework

- C++ parallel branch-and-bound framework
- Based solvers and communication libraries are abstracted
- NuOpt, SCIP and Xpress is parallelized with UG
- Many attractive features: checkpoint, racing, self-split, ... et al.

Single controller: LoadCoordinaor



Parallelize Base solver externally



New features of UG

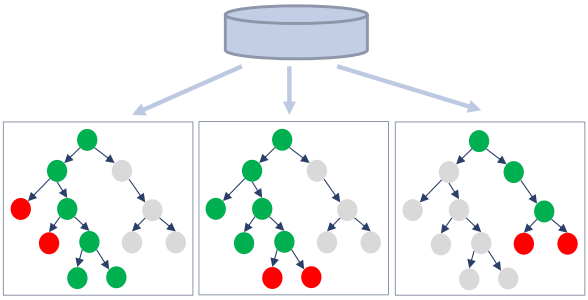
- Self-Split (+ heuristics)
- Enhanced checkpoint
- Huge checkpoint split
- Notification Offset

New features of UG

- Self-Split (+ heuristics)
- Enhanced checkpoint
- Huge checkpoint split
- Notification Offset

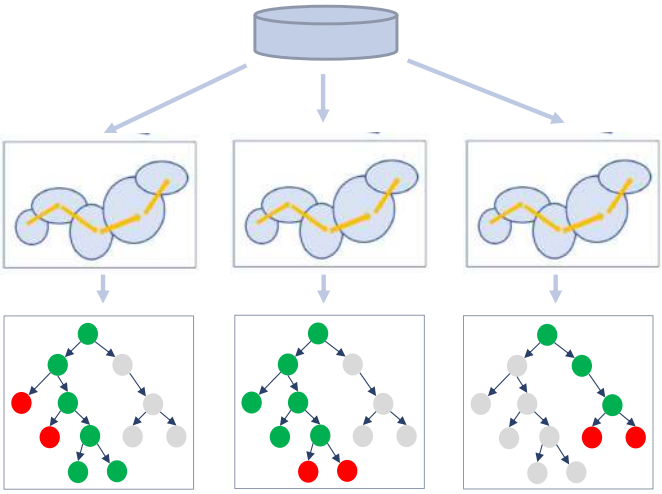
UG: Self-splitting

- Self-splitting ([Laursen 1994] [Fischetti 2014])
  - The same enumeration trees are initially built by all workers.
  - Each worker solves the assigned nodes after sampling phase.



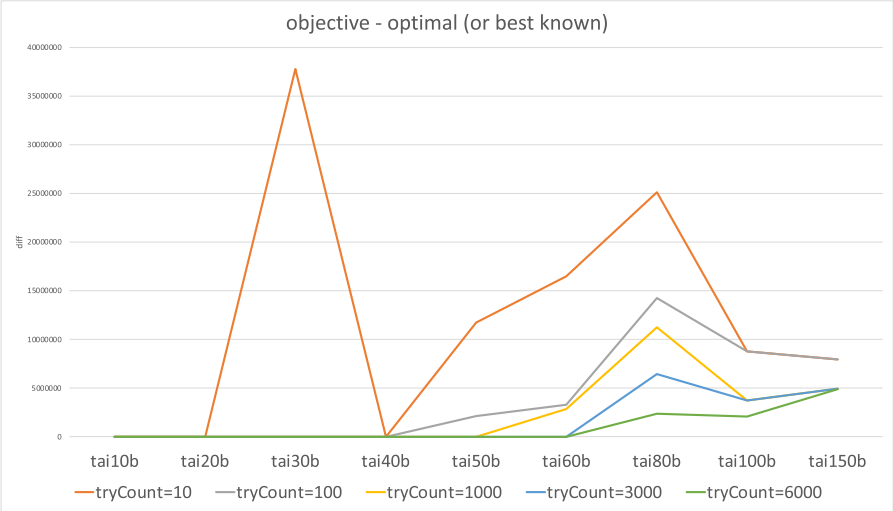
UG: Self-splitting (+heuristics)

- Self-splitting (+heuristics)
  - Robust Tabu Search [Taillard 1991] + randomization



UG: Self-splitting (+heuristics)

- Self-splitting (+heuristics)



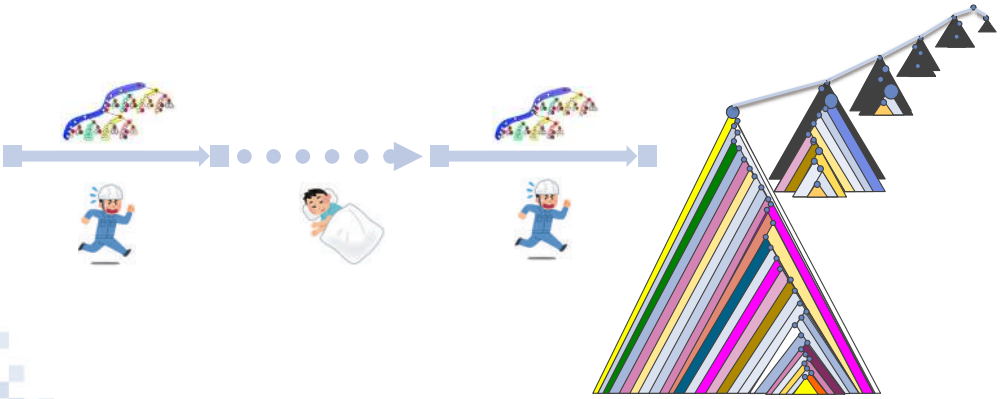
- iteration=100000
- AMD Ryzen 9 5900X 12-Core Processor

New features of UG

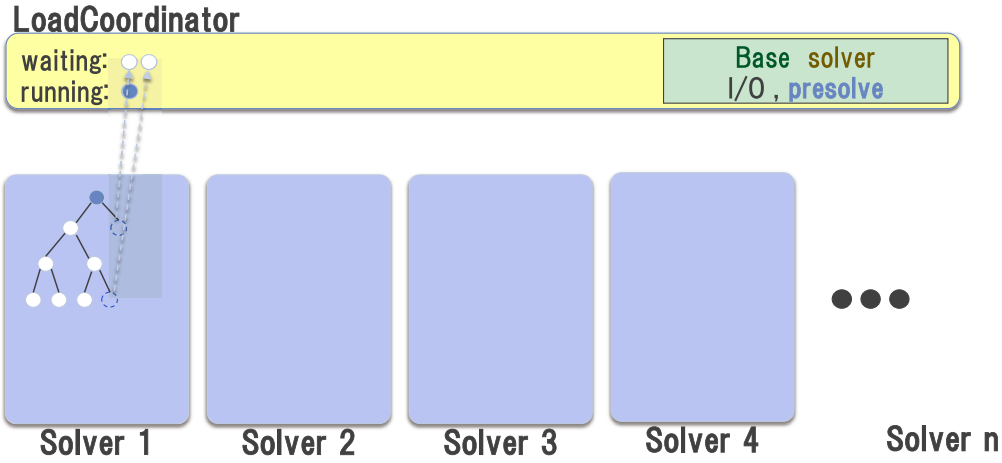
- Self-Split (+ heuristics)
- Enhanced checkpoint
- Huge checkpoint split
- Notification Offset

UG: Checkpointing mechanism

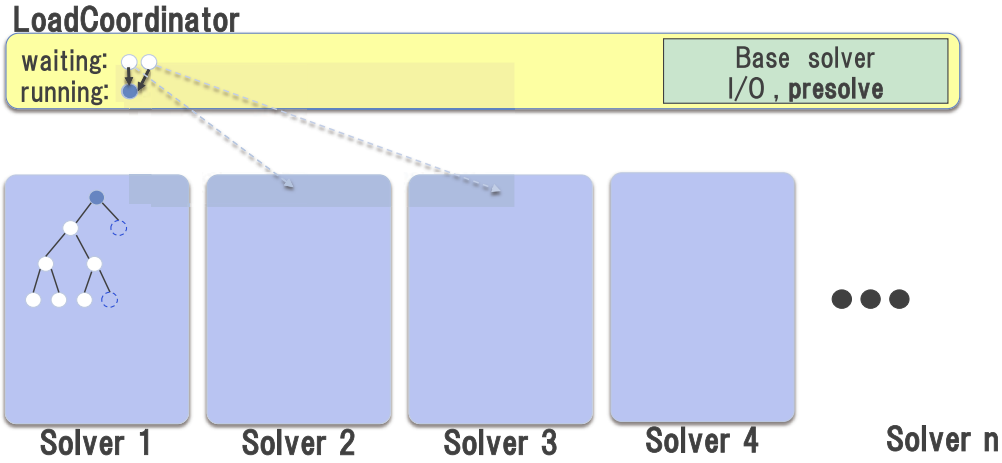
- Since usage of supercomputer is limited, checkpointing mechanism (b&b tree storage) is indispensable
- Only the essential nodes are saved into checkpoint files



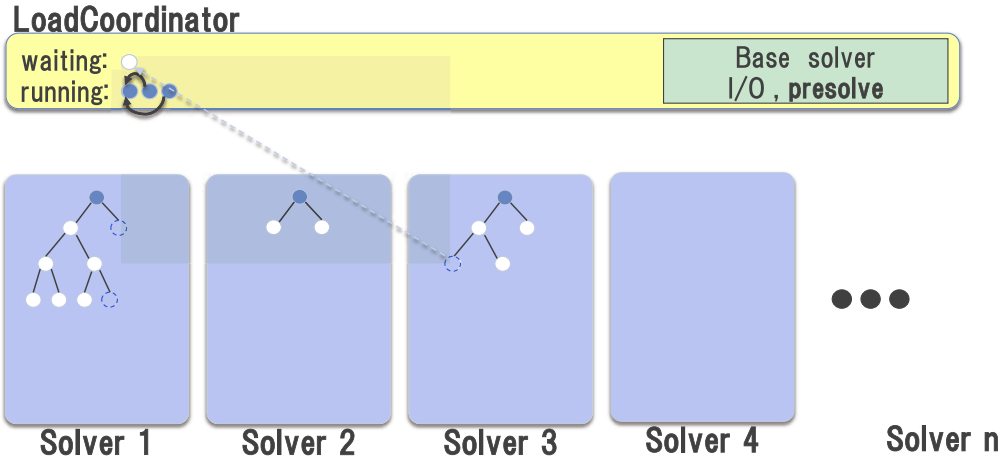
UG: Checkpoint/Restart mechanism



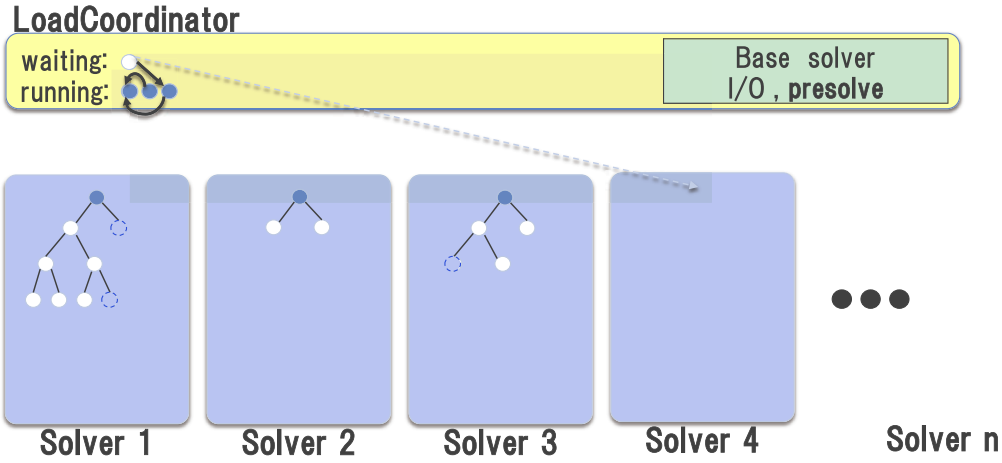
UG: Checkpoint/Restart mechanism



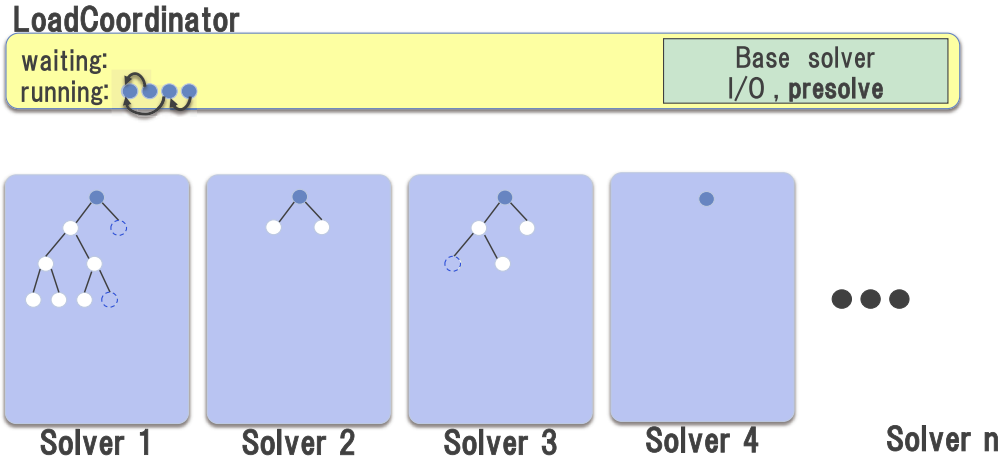
UG: Checkpoint/Restart mechanism



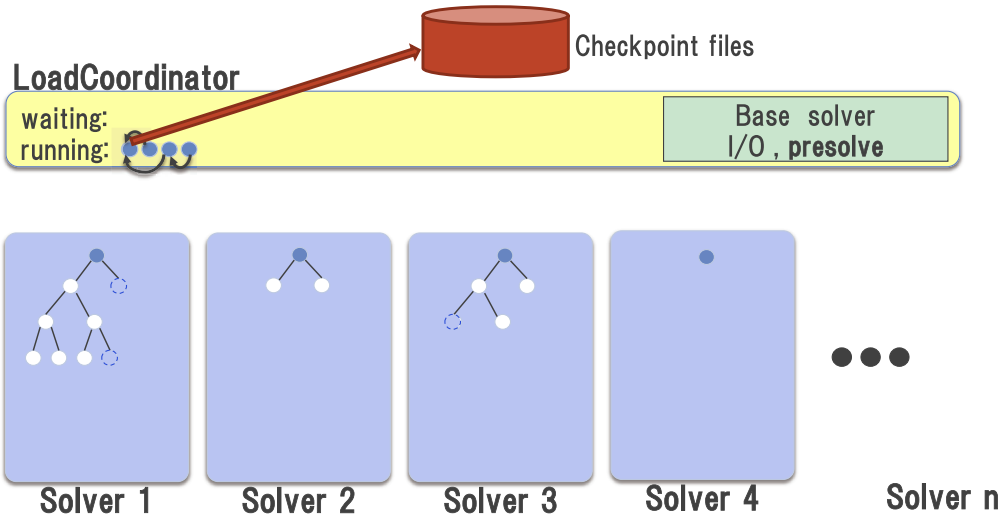
UG: Checkpoint/Restart mechanism



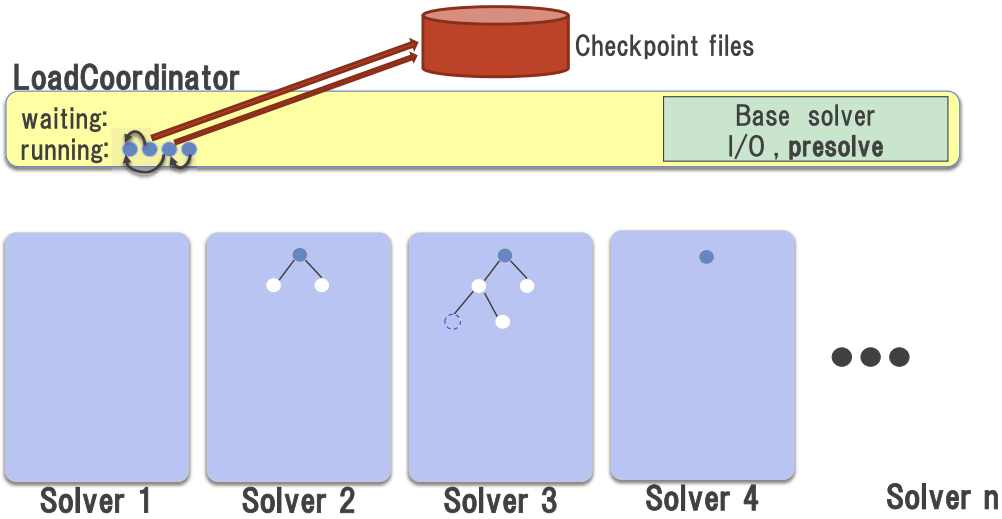
UG: Checkpoint/Restart mechanism



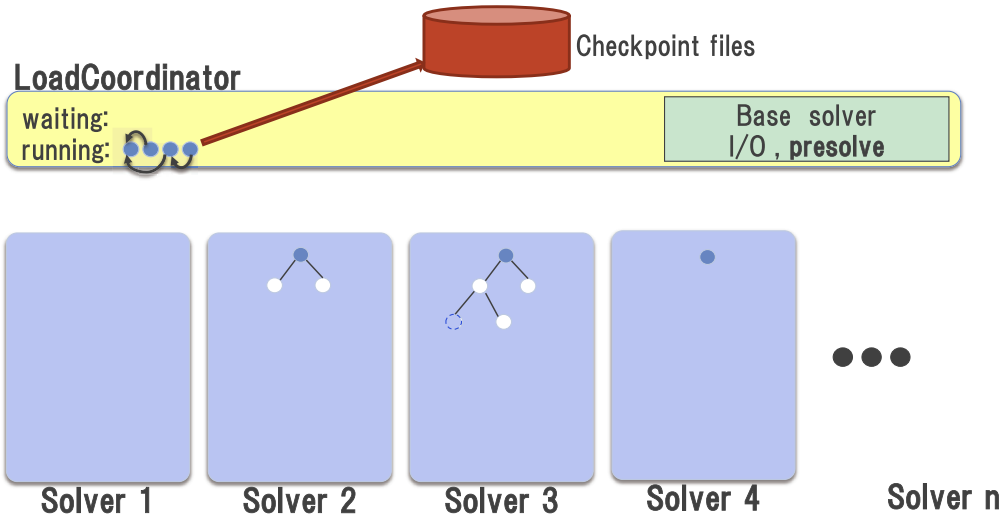
UG: Checkpoint/Restart mechanism



UG: Checkpoint/Restart mechanism

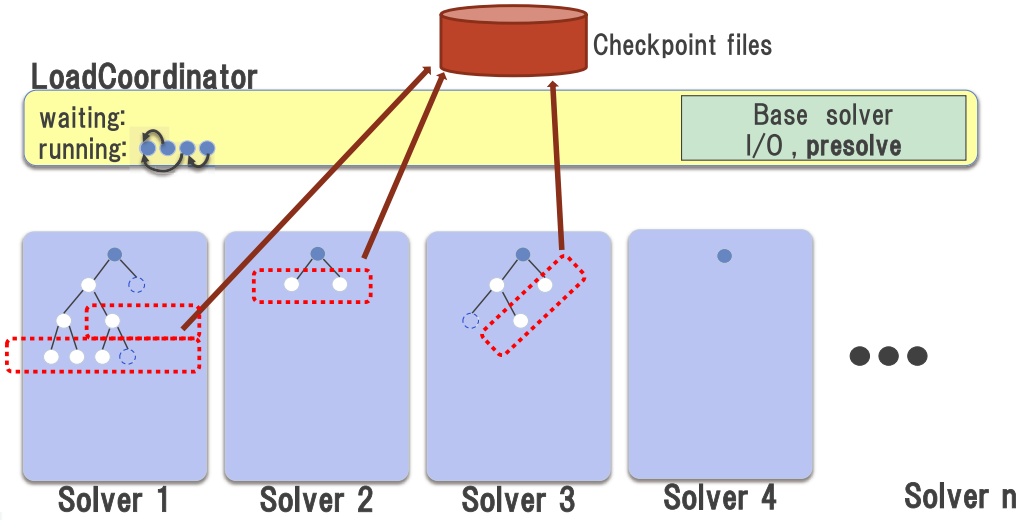


UG: Checkpoint/Restart mechanism



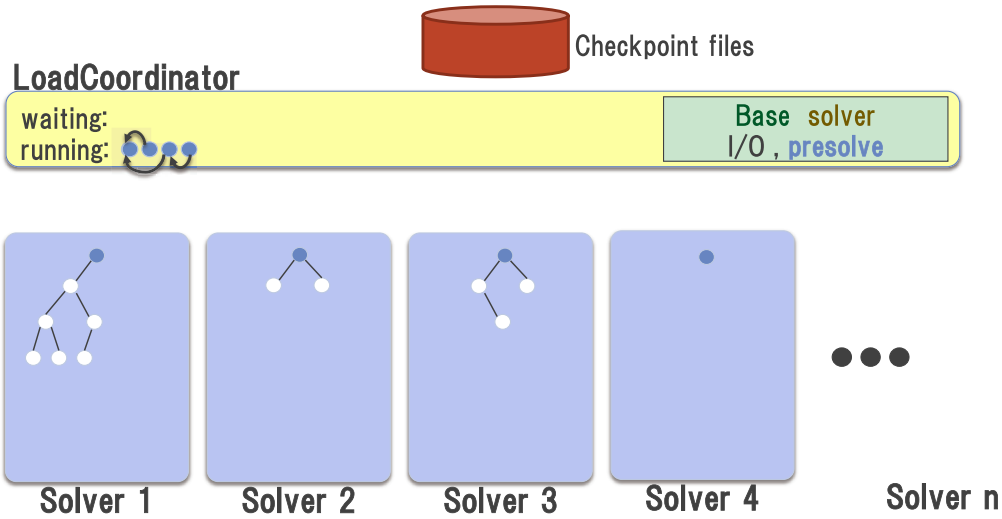
UG: Enhanced Checkpoint

[collects all nodes one solver by one solver]



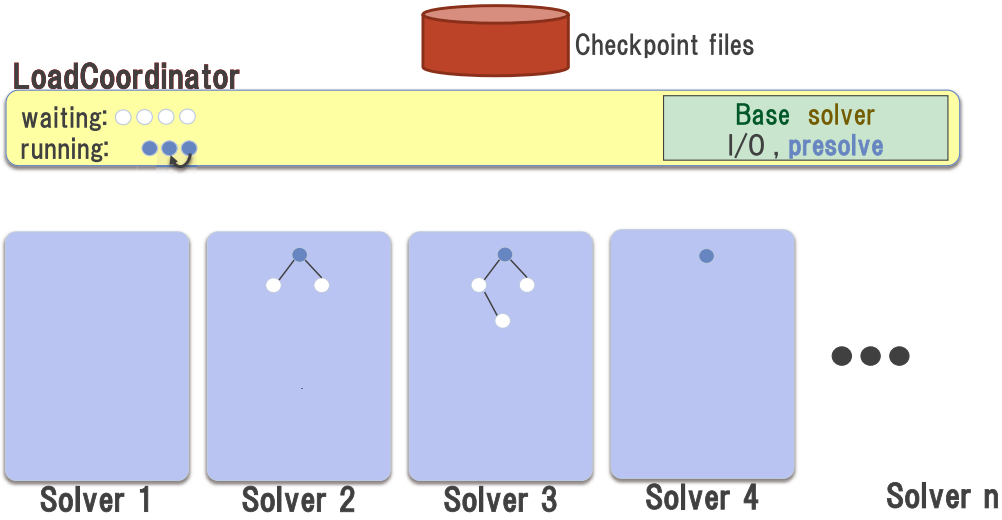
UG: Enhanced Checkpoint

[collects all nodes one solver by one solver]



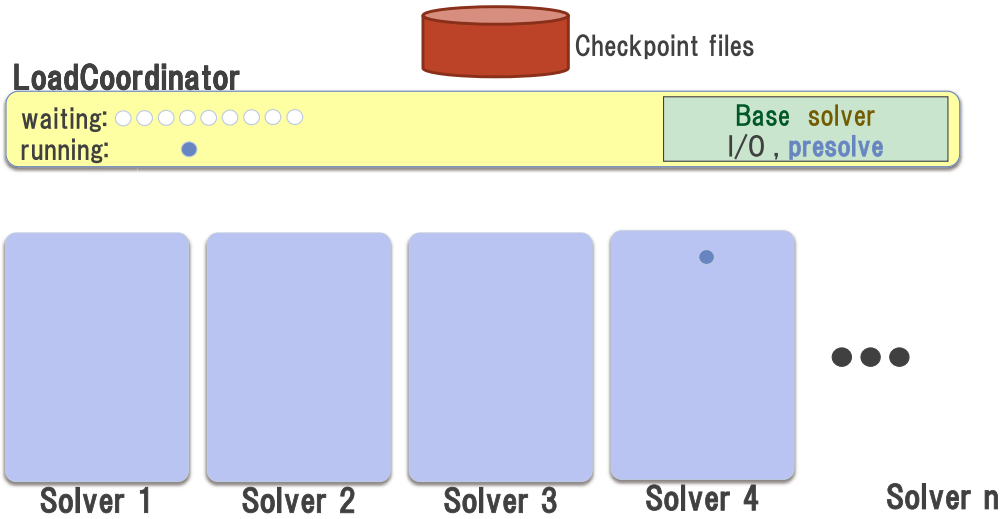
UG: Enhanced Checkpoint

[collects all nodes one solver by one solver]



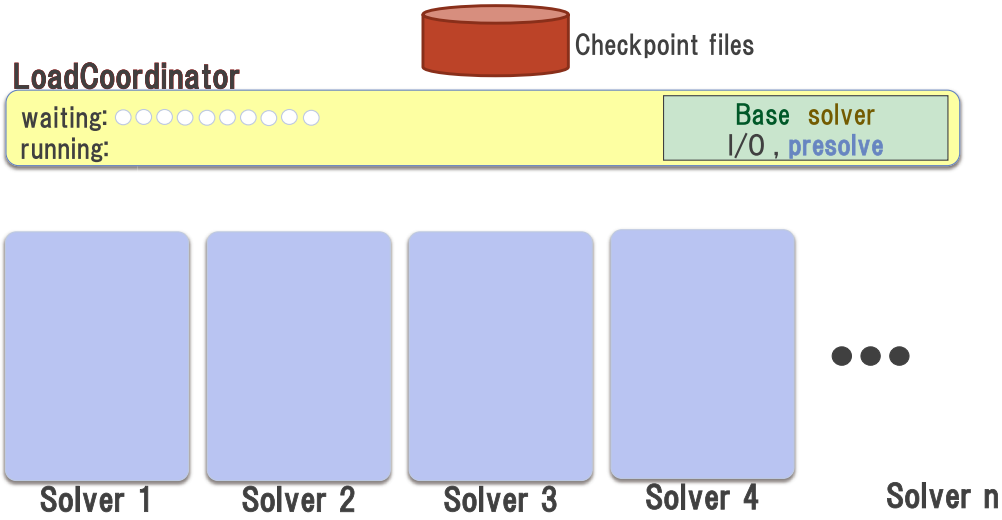
UG: Enhanced Checkpoint

[collects all nodes one solver by one solver]



UG: Enhanced Checkpoint

[collects all nodes one solver by one solver]



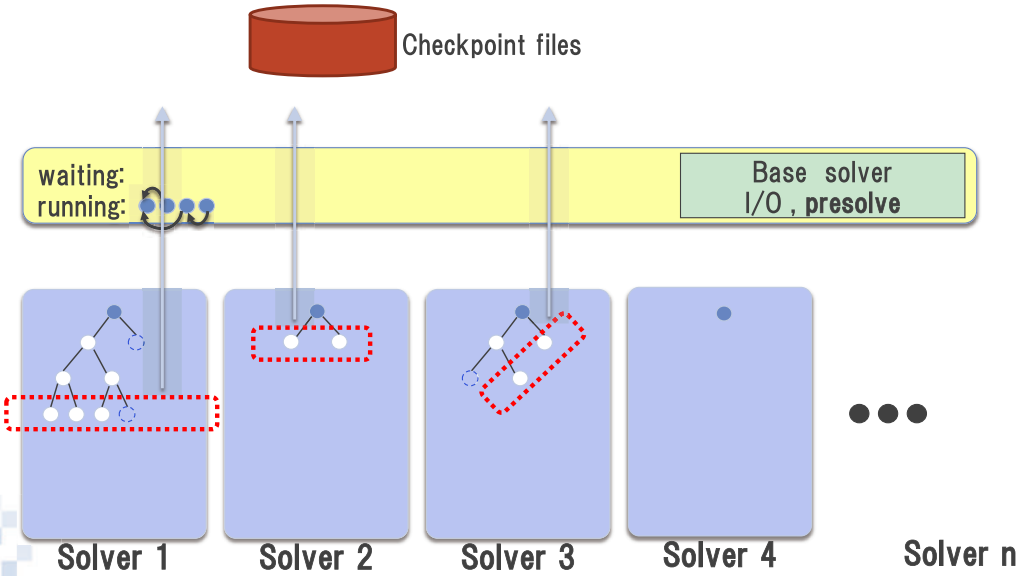
UG: Enhanced Checkpoint

- UG is based on "Supervised/Worker" mechanism (≠ Primary/Replica mechanism)
  cf. Kurt Anstreicher et al "Solving large quadratic assignment problems on computational grids"
- Enhanced check point enables to collect all open nodes, working carefully to be compatible with existing check point mechanism in UG.

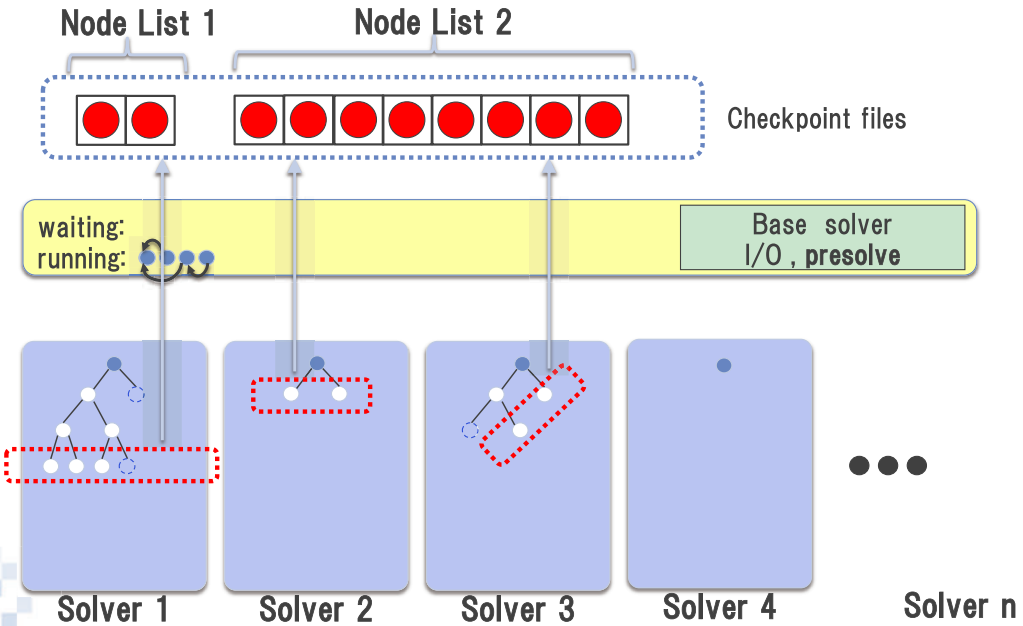
New features of UG

- Self-Split (+ heuristics)
- Enhanced checkpoint
- Huge checkpoint split
- Notification Offset

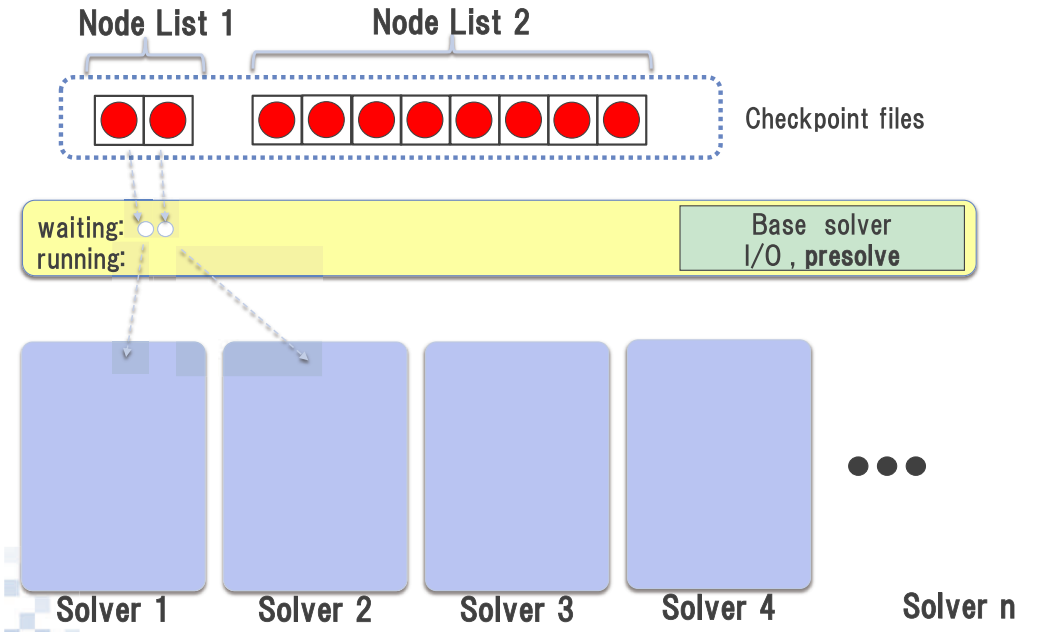
UG: Huge Checkpoint Split



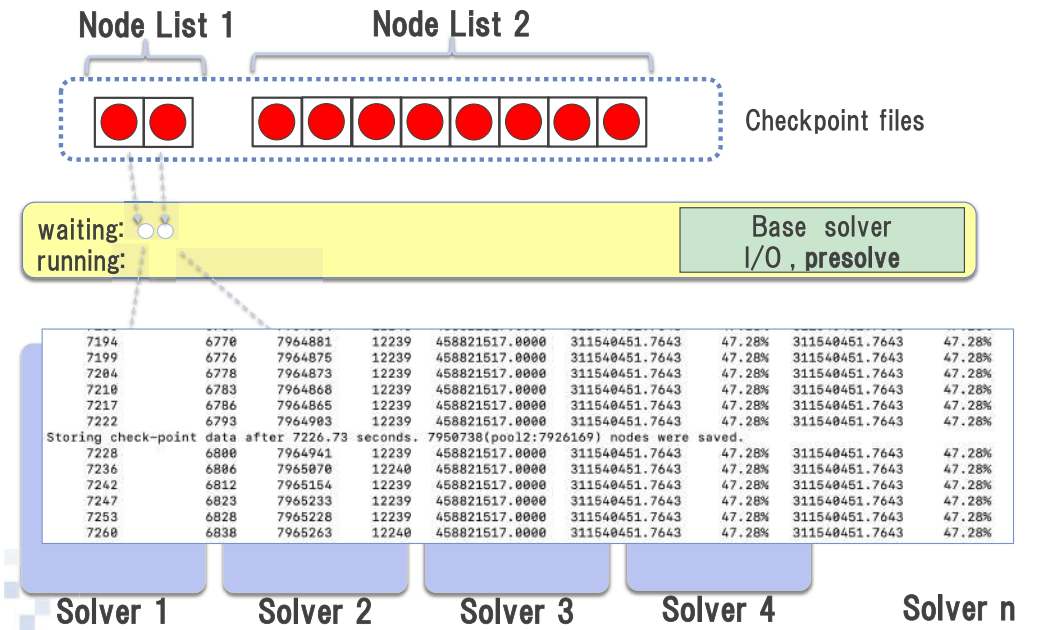
UG: Huge Checkpoint Split



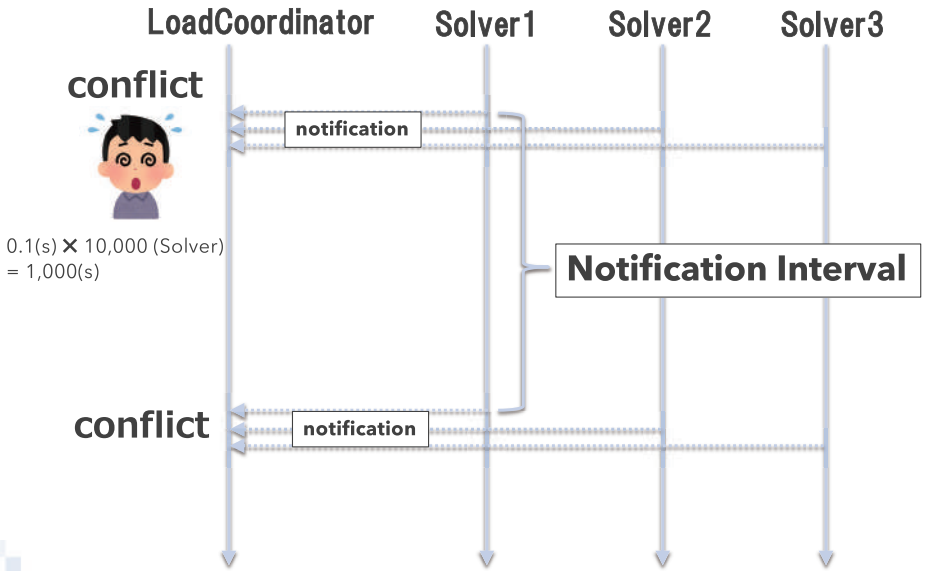
UG: Huge Checkpoint Split



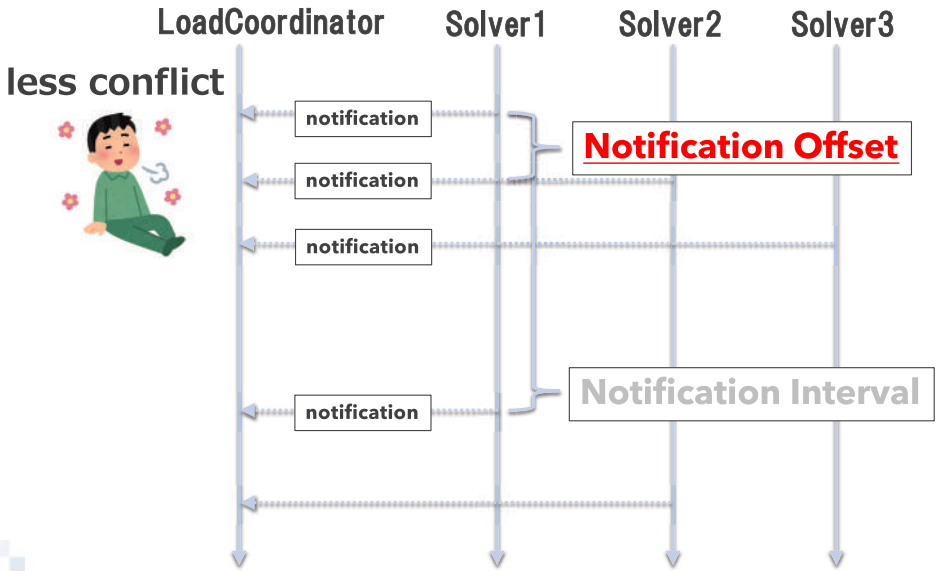
UG: Huge Checkpoint Split



UG: Notification Offset



UG: Notification Offset



UG: Huge checkpoint split / Notification Offset

- UG is now possible to "split" checkpoint files
  - avoid computational cost to work on large checkpoint file
- New feature "Notification Offset" is useful if processing status messages (notification) is time consuming.

Experimental Results of ParaQapNB

	solver=1(+LC=1)		solver=5(+LC=1)		solver=11(+LC=1)	
problem	#node	time(s)	#node	time(s)	#node	time(s)
nug17	50	100.09	50	52.27	50	48.49
nug18	104	170.58	104	68.57	104	37.67
nug20	685	1974.14	685	407.14	685	236.71
nug21	312	1047.18	312	239.59	312	156.45
nug22	429	1968.21	429	412.47	429	240.47
nug24	1245	9225.15	1245	2014.35	1245	1217.50
tai17a	364	290.59	364	70.85	364	44.72
tai20a	2464	5185.60	2464	1087.38	2464	577.78
tai20b	166	223.01	166	186.08	166	187.90
geomean		843.16		252.45		166.08

Table: computational results of ParaQapNB on medium instances

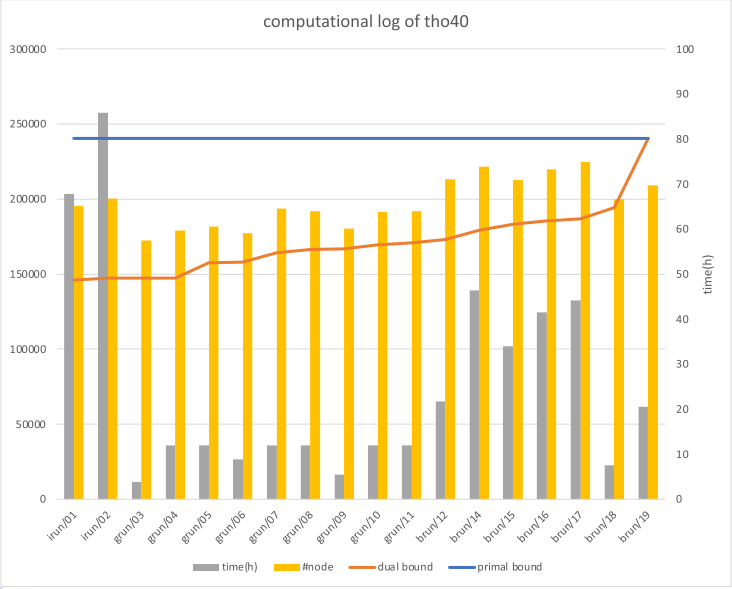
Experimental Results of ParaQapNB

	problem	Opt.val	#node	time(sec)	No. of CPU cores used
	nug30	6,124	26,181	3.14e3	1,728
2021	tai30a	1,818,146	34,000,579	5.81e5 ≈ 6.8 days	1,728
	tai35b	283,315,445	2,620,547	2.49e5	1,728
	tai40b	637,250,948	278,465	1.05e5	1,728
2021	sko42	15,812	6,019,419	5.12e5 ≈ 5.9 days	5,184
2022	tho40	240,516	139,077,975	1.66e6 ≈ 19.1 days	24,528

Table: Computational results on large scale QAPs

- HPE SGI 8600 (384 nodes, 13,824 cores)
- HLRN-IV System (1270 nodes, 230,000 cores)

Experimental Results of ParaQapNB



irun:
HPE SGI 8600
@ISM(Japan)

grun:
HLRN-IV System
@ Göttingen

brun:
HLRN-IV System
@ Berlin

Summary & Future work

- ParaQapNB: massively parallel DNN-based branch-and-bound solver for QAP
  - successfully solved open instances tai30a(2021), sko42 (2021), tho40(2022)
  - still have chance to solve some more open instances

- New features of UG contribute to process large & heavy tree of DNN-based branch-and-bound

Mobility Optimization Engine and its Real-world Applications

Katsuki FUJISAWA
Institute of Mathematics for Industry, Kyushu University, Japan
fujisawa@imi.kyushu-u.ac.jp

Various efforts have been made to realize a so-called super-smart society recently. Our project team builds services to create new industries and other services through corporate collaboration [1,2,3]. We have utilized large-scale computing infrastructures and developed the Cyber-Physical System Mobility Optimization Engine (CPS-MOE) that provides various functions, including creating new industries. It can reduce cost and industrial waste and constructing services to calculate the optimum control schedule of transportation agencies. The latest research results and industry-academia collaborative projects using CPS-MOE will be presented in this talk.

References

[1] Akihiro Yoshida, Tatsuru Higurashi, Masaki Maruishi, Nariaki Tateiwa, Nozomi Hata, Akira Tanaka, Takashi Wakamatsu, Kenichi Nagamatsu, Akira Tajima, and Katsuki Fujisawa, "New Performance Index "Attractiveness Factor" for Evaluating Websites via Obtaining Transition of Users' Interests", Data Science and Engineering, Volume 5, Issue 1, pp. 48-64, March 2020, https://doi.org/10.1007/s41019-019-00112-1

[2] Akihiro Yoshida, Yosuke Yatsushiro, Nozomi Hata, Tatsuru Higurashi, Nariaki Tateiwa, Takashi Wakamatsu, Akira Tanaka, Kenichi Nagamatsu, and Katsuki Fujisawa, "Practical End-to-End Repositioning Algorithm for Managing Bike-Sharing System", The proceedings of the IEEE BigData2019, 2019, https://doi.org/10.1109/BigData47090.2019.9005986

[3] Nozomi Hata, Takashi Nakayama, Akira Tanaka, Takashi Wakamatsu, Akihiro Yoshida, Nariaki Tateiwa, Yuri Nishikawa, Jun Ozawa, and Katsuki Fujisawa, "Mobility Optimization on Cyber Physical System via Multiple Object Tracking and Mathematical Programming", the Fifth International Workshop on High Performance Big Graph Data Management, Analysis, and Mining (BigGraphs 2018), to be held in conjunction with the 2018 IEEE International Conference on Big Data (IEEE BigData 2018), 2018, https://doi.org/10.1109/BigData.2018.8622146

Institute of Mathematics for Industry
Kyushu University

九州大学
KYUSHU UNIVERSITY

Cyber-physical System and Industrial Applications on Large-scale Computing Infrastructure

Katsuki Fujisawa

Professor, Institute of Mathematics for Industry, Kyushu University

The 6th RIKEN-IMI-ISM-NUS-ZIB-MODAL-NHR Workshop on Advances in Classical and Quantum Algorithms for Optimization and Machine Learning
September 18th, 2022

1

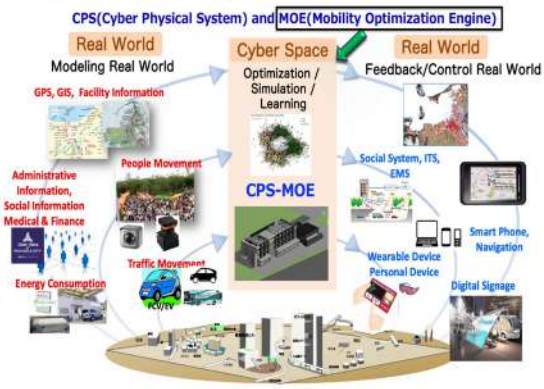
The battle for supremacy over data, the "oil of the 21st century

The main battlefield will shift from

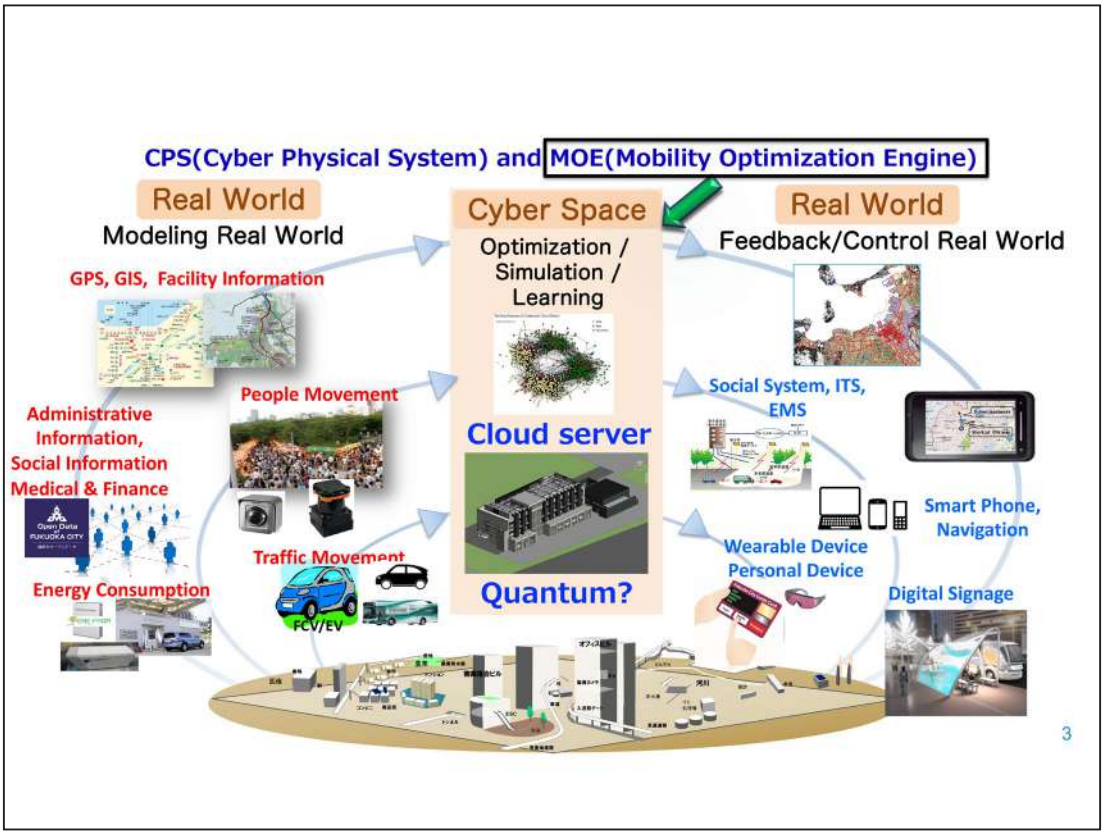
cyberspace to cyber-physical space

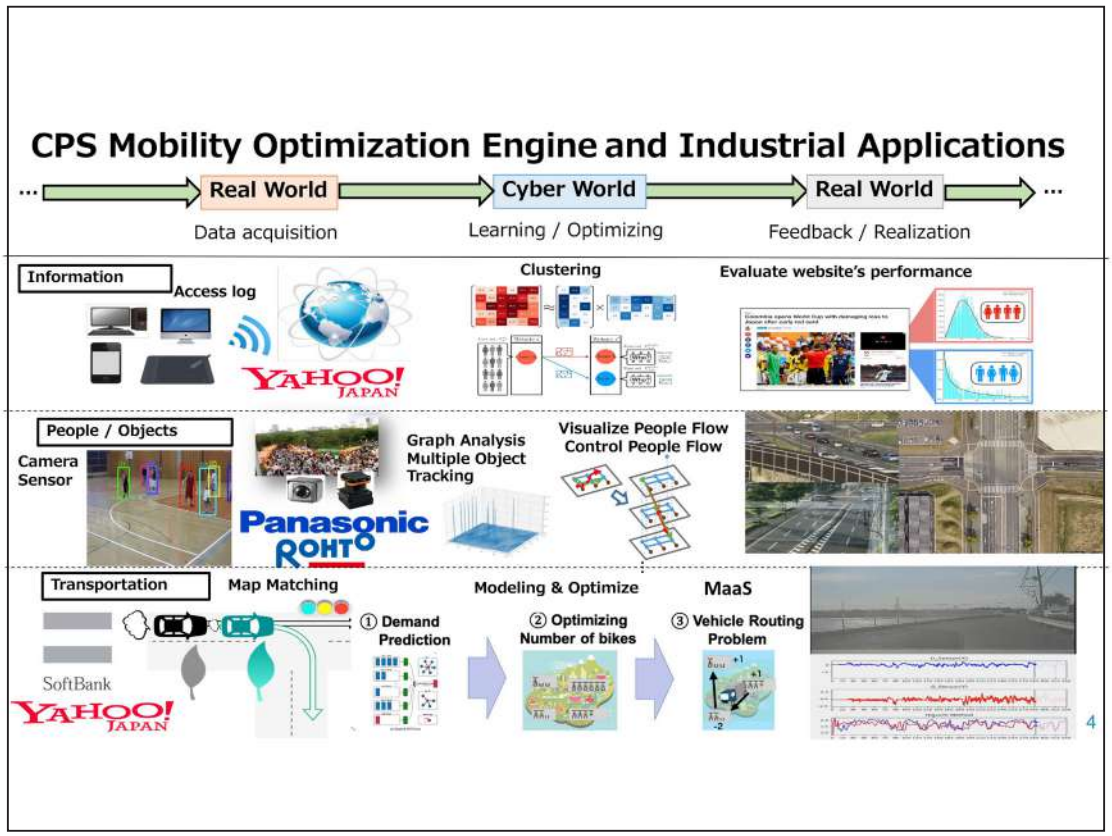
The world's most valuable resource is no longer oil, but data

The data economy demands a new approach to antitrust rules



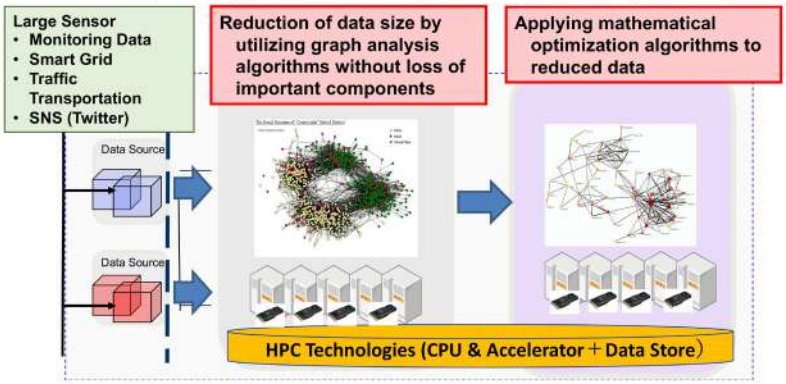
2


CPS(Cyber Physical System) and MOE(Mobility Optimization Engine)
Real World
Modeling Real World
GPS, GIS, Facility Information
People Movement
Administrative Information, Social Information Medical & Finance
Traffic Movement
Energy Consumption
FCV/EV
Cyber Space
Optimization / Simulation / Learning
Cloud server
Quantum?
Real World
Feedback/Control Real World
Social System, ITS, EMS
Smart Phone, Navigation
Wearable Device Personal Device
Digital Signage
3


CPS Mobility Optimization Engine and Industrial Applications
Real World
Cyber World
Real World
Data acquisition
Learning / Optimizing
Feedback / Realization
Information
Access log
YAHOO! JAPAN
Clustering
Evaluate website's performance
People / Objects
Camera Sensor
Panasonic
ROHTO
Graph Analysis Multiple Object Tracking
Visualize People Flow Control People Flow
Transportation
Map Matching
SoftBank
YAHOO! JAPAN
Modeling & Optimize
① Demand Prediction
② Optimizing Number of bikes
MaaS
③ Vehicle Routing Problem
4

Algorithm Layers of CPS Mobility Optimization Engine

- First layer ：Graph and network analysis algorithms
  - **Dijkstra algorithm (Single source shortest path problem) ,BFS (Breath first search algorithm)** : Shortest path, Centrality(BC etc.), Clustering problem
- Second layer ：AI and Mathematical Optimization algorithms
  - **ML/DL, MILP(Mixed Integer Linear Problem), SDP** : Facility location problem, Set covering (partitioning) problem, Scheduling, Evacuation Program, Object Tracking



5

Graph500 (https://graph500.org/)

Graph Search Based Benchmarks for Ranking Supercomputers

- **Graph500** is a competition for evaluating performance of large-scale graph processing
- The performance metric is a traversed edges per second (TEPS)
  - 1GTEPS : Search 1 billion edges per second
- Graph500 list is updated twice a year (June and November in BoFs of ISC and SC)
- An artificial graph called the "Kronecker graph" is used
  - A good approximation of Social network graph (small world and scale free)

In 2014 to 2019, Our project team has been a winner at the eighth, and 10th to 18th Graph500 benchmark.
K computer (RIKEN, Japan)

In 2020 to 2021, Our project team has been a winner at 20th to 22nd Graph500 benchmark.
Fugaku super computer (RIKEN, Japan)

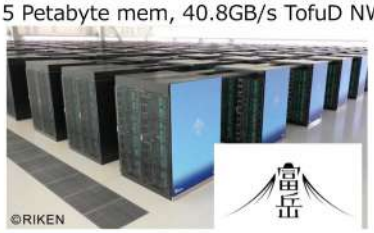
Future direction --> Collaborative research (RIKEN (R-CCS), ZIB, IMI)

6

Fugaku & K Computer #1 : Graph500 Benchmark

Graph Search Based Benchmarks for Ranking Supercomputers https://graph500.org/

158,976 nodes, 7,630,848 CPU cores
5 Petabyte mem, 40.8GB/s TofuD NW



88,000 nodes,
660,000 CPU Cores
1.3 Petabyte mem
20GB/s Tofu NW

K computer

Effective x13 performance c.f. Linpack

List	Rank	GTEPS	Implementation
June 2014	1	17977.05	Efficient hybrid
June 2015 ~ Nov 2018	1	38621.40	Hybrid + Node Compression
June 2020	1	70,980.17	Hybrid + Node Compression
June 2021	1	102,955.45	Hybrid + Node Compression Load Balancing

*Problem size is weak scaling "Brain-class" graph

LLNL-IBM Sequoia
1.6 million CPUs
1.6 Petabyte mem

TaihuLight
10 million CPUs
1.3 Petabyte mem

IBM

Example of CPS Application

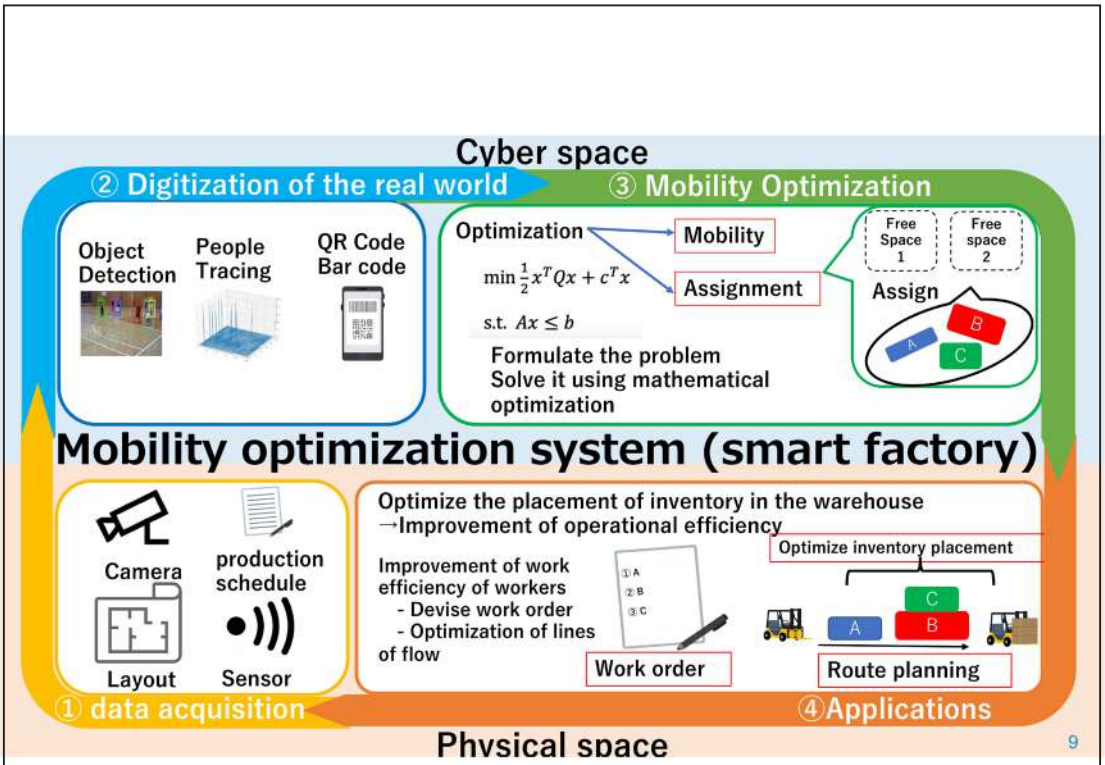
Smart Factory : ROHTO Pharmaceutical Co.,Ltd., Ueno City, Japan

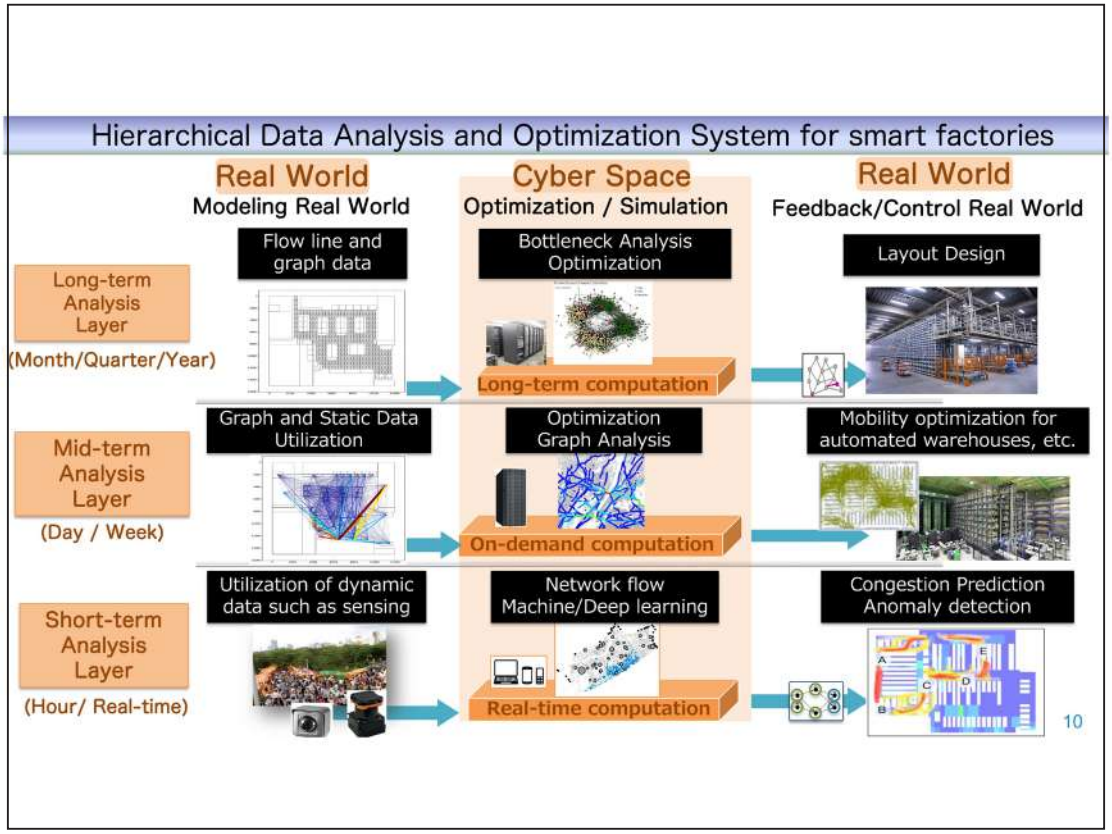
ROHTO

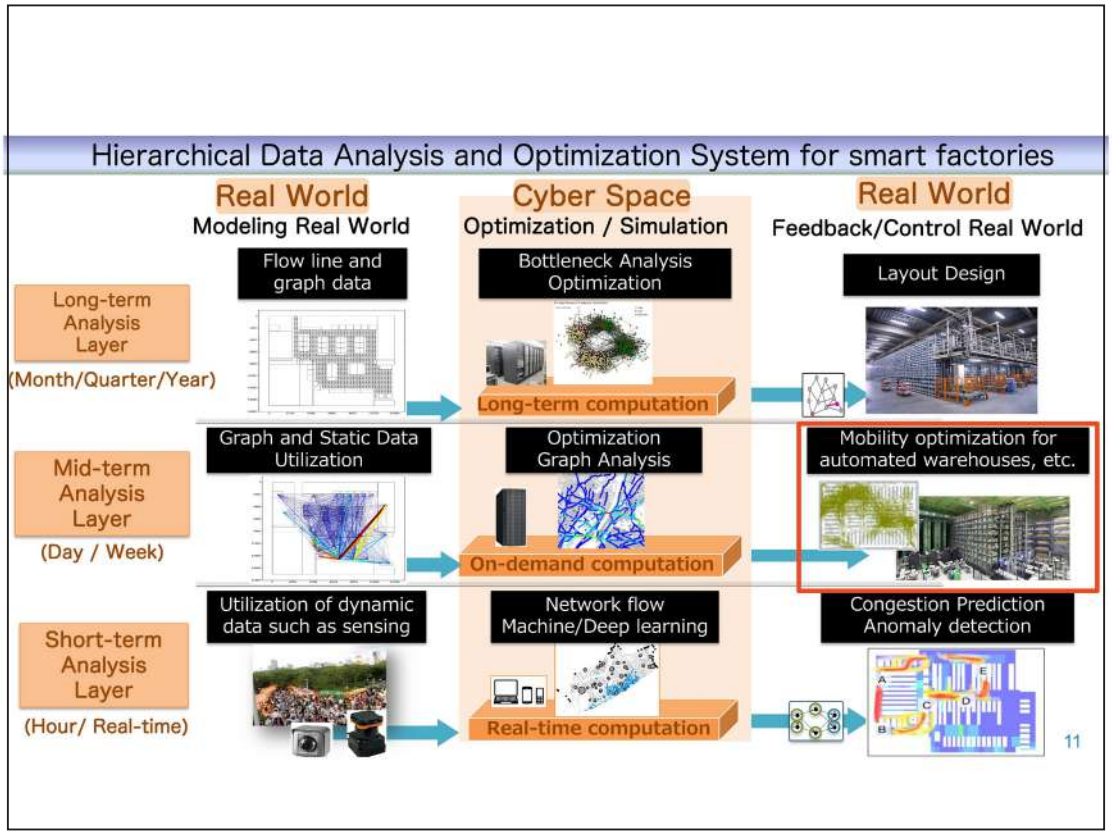
- Main Products of this Plant
  - Eye drops, Cosmetics
    - **One of the world's leading eye drop production lines**

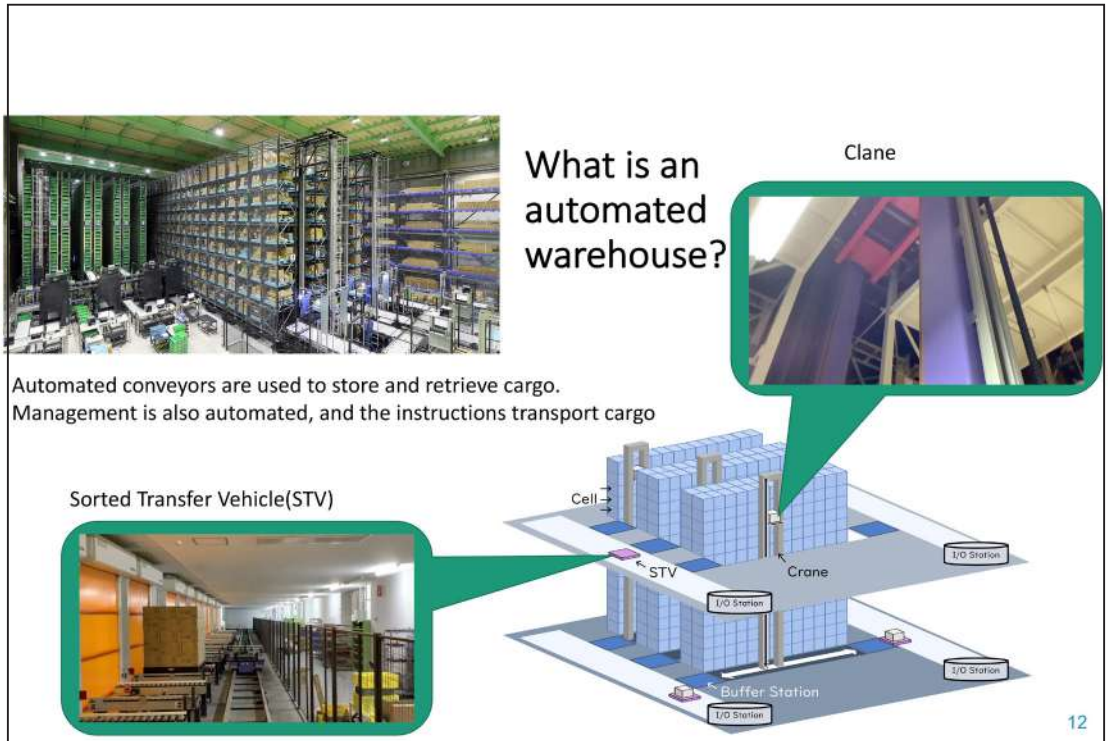
Ueno City
Famous as a village of ninja

https://www.youtube.com/watch?v=fESTa-QZKTE


Cyber space
② Digitization of the real world
③ Mobility Optimization
Object Detection
People Tracing
QR Code Bar code
Optimization
Mobility
Assignment
$\min \frac{1}{2}x^T Qx + c^T x$
s.t. $Ax \leq b$
Formulate the problem
Solve it using mathematical optimization
Free Space 1
Free space 2
Assign
A
B
C
Mobility optimization system (smart factory)
Camera
production schedule
Layout
Sensor
Optimize the placement of inventory in the warehouse
→Improvement of operational efficiency
Improvement of work efficiency of workers
- Devise work order
- Optimization of lines of flow
Work order
Optimize inventory placement
Route planning
① data acquisition
④Applications
Physical space
9


Hierarchical Data Analysis and Optimization System for smart factories
Real World
Modeling Real World
Cyber Space
Optimization / Simulation
Real World
Feedback/Control Real World
Long-term Analysis Layer
(Month/Quarter/Year)
Flow line and graph data
Bottleneck Analysis Optimization
Layout Design
Long-term computation
Mid-term Analysis Layer
(Day / Week)
Graph and Static Data Utilization
Optimization Graph Analysis
Mobility optimization for automated warehouses, etc.
On-demand computation
Short-term Analysis Layer
(Hour/ Real-time)
Utilization of dynamic data such as sensing
Network flow Machine/Deep learning
Congestion Prediction Anomaly detection
Real-time computation
10


Hierarchical Data Analysis and Optimization System for smart factories
Real World
Modeling Real World
Cyber Space
Optimization / Simulation
Real World
Feedback/Control Real World
Long-term Analysis Layer
(Month/Quarter/Year)
Flow line and graph data
Bottleneck Analysis Optimization
Layout Design
Long-term computation
Mid-term Analysis Layer
(Day / Week)
Graph and Static Data Utilization
Optimization Graph Analysis
Mobility optimization for automated warehouses, etc.
On-demand computation
Short-term Analysis Layer
(Hour/ Real-time)
Utilization of dynamic data such as sensing
Network flow Machine/Deep learning
Congestion Prediction Anomaly detection
Real-time computation
11


What is an automated warehouse?
Clane
Automated conveyors are used to store and retrieve cargo.
Management is also automated, and the instructions transport cargo
Sorted Transfer Vehicle(STV)
Cell
STV
Crane
I/O Station
Buffer Station
12

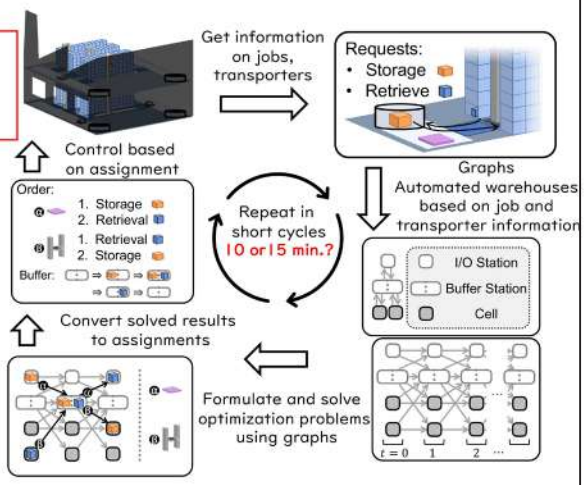
Automated warehouse operation optimization system

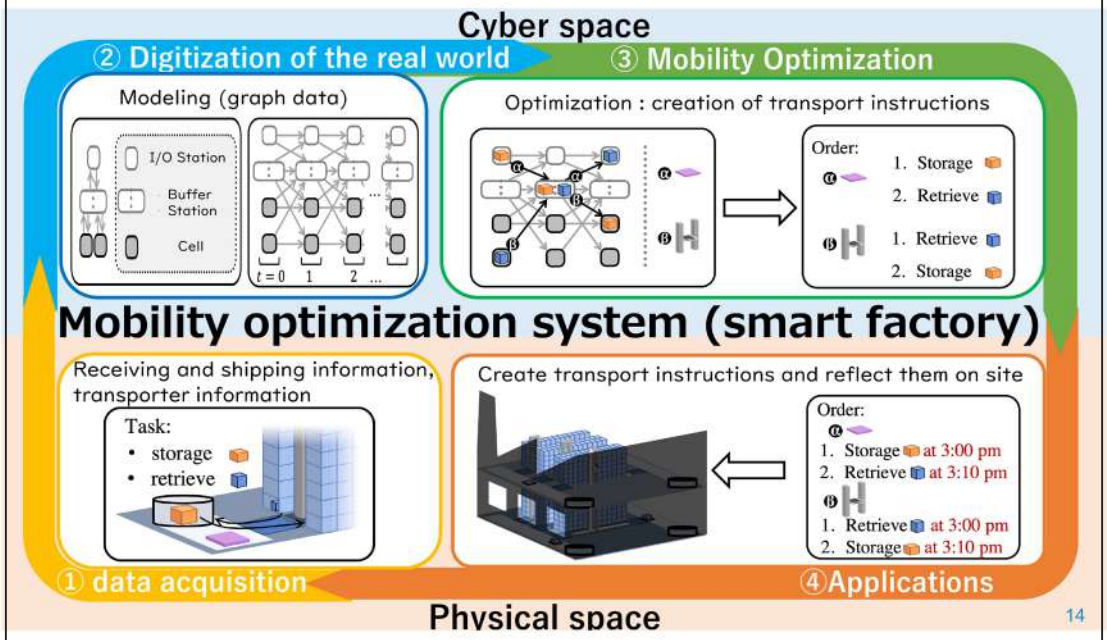
Three algorithms for generating job assignments
- Heuristic method (rule-based)
- Exact solution method (optimization problem)
- Deep reinforcement learning

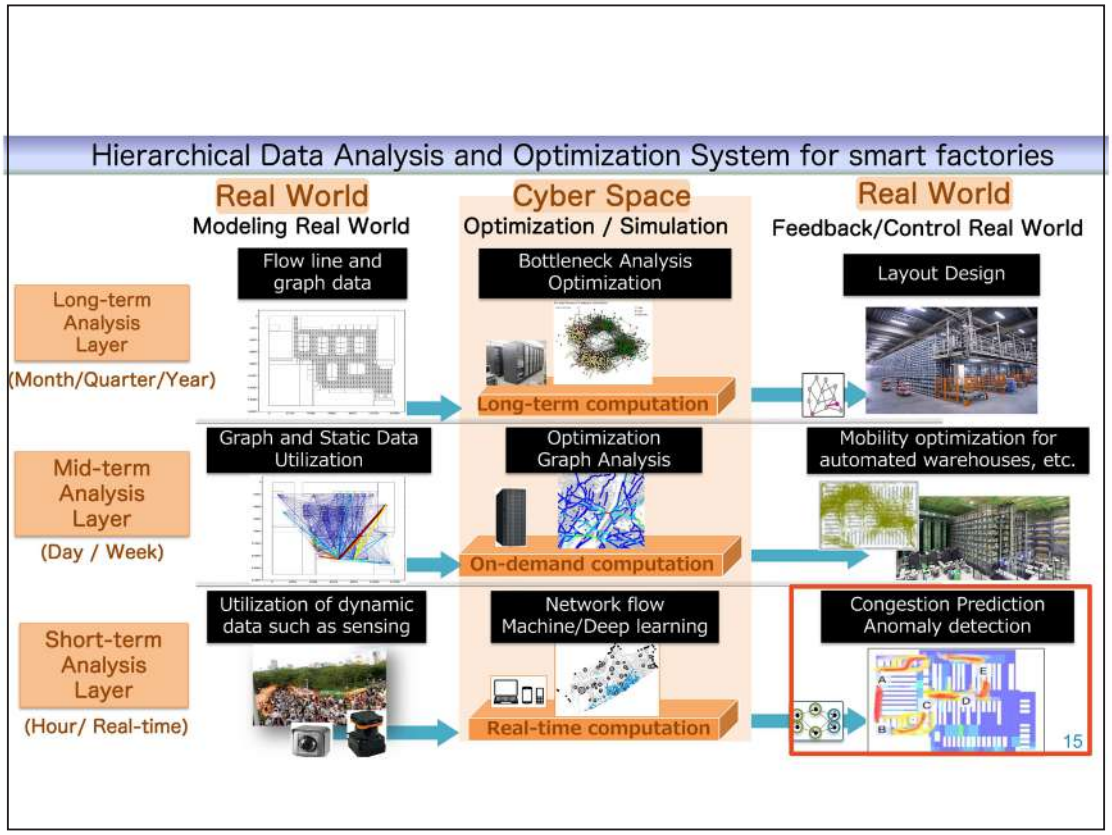
Perform calculations based on formulation

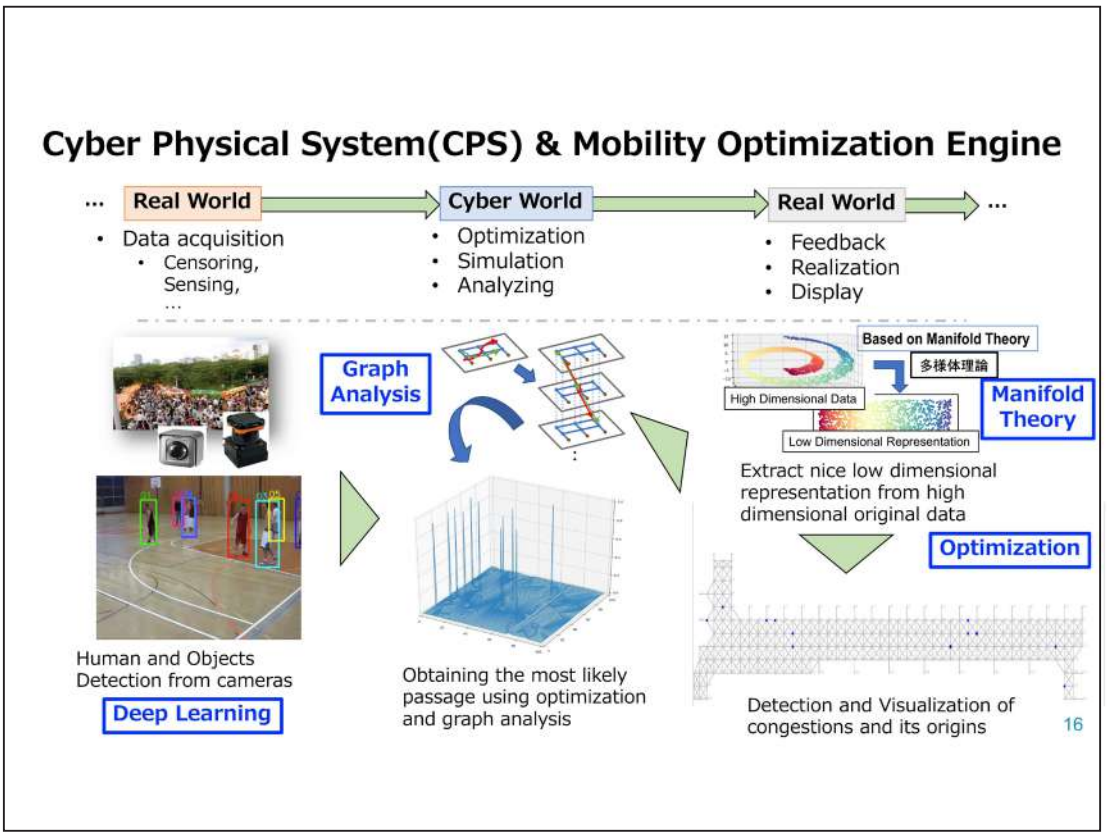
- Features (compared to optimization problems)
- Solution accuracy: Optimality guaranteed
- Computation time: Depends on the number of jobs, etc. It may take more than 1 hour

Optimize "job and transporter assignment" and "the order in which each transporter processes jobs.
Optimize the order in which each transporter processes jobs.






Hierarchical Data Analysis and Optimization System for smart factories
Real World
Modeling Real World
Cyber Space
Optimization / Simulation
Real World
Feedback/Control Real World
Long-term Analysis Layer
(Month/Quarter/Year)
Flow line and graph data
Bottleneck Analysis Optimization
Long-term computation
Layout Design
Mid-term Analysis Layer
(Day / Week)
Graph and Static Data Utilization
Optimization Graph Analysis
On-demand computation
Mobility optimization for automated warehouses, etc.
Short-term Analysis Layer
(Hour/ Real-time)
Utilization of dynamic data such as sensing
Network flow Machine/Deep learning
Real-time computation
Congestion Prediction Anomaly detection
15


Cyber Physical System(CPS) & Mobility Optimization Engine
...
Real World
Cyber World
Real World
...
• Data acquisition
• Censoring, Sensing, ...
• Optimization
• Simulation
• Analyzing
• Feedback
• Realization
• Display
Graph Analysis
Based on Manifold Theory
多様体理論
High Dimensional Data
Low Dimensional Representation
Manifold Theory
Extract nice low dimensional representation from high dimensional original data
Optimization
Human and Objects Detection from cameras
Deep Learning
Obtaining the most likely passage using optimization and graph analysis
Detection and Visualization of congestions and its origins
16

Optimization of flow lines of people and objects

Motion line tracking and optimization using images captured by fisheye and box cameras

【 Objective 】

- Understanding the work-flow line
- Understanding areas and duration of stay

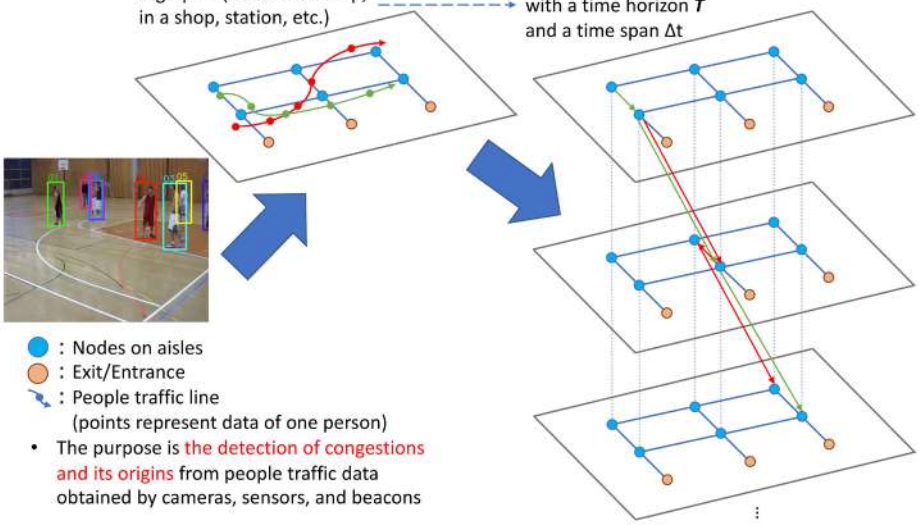
17

People Congestion Analyzing System Using Time-Expanded Graph

– Step 1: Mapping People Traffic Lines to Graph Data –

A graph ***G***(based on a map; in a shop, station, etc.)

Its time-expanded graph G_T with a time horizon ***T*** and a time span Δt



: Nodes on aisles

: Exit/Entrance

: People traffic line (points represent data of one person)

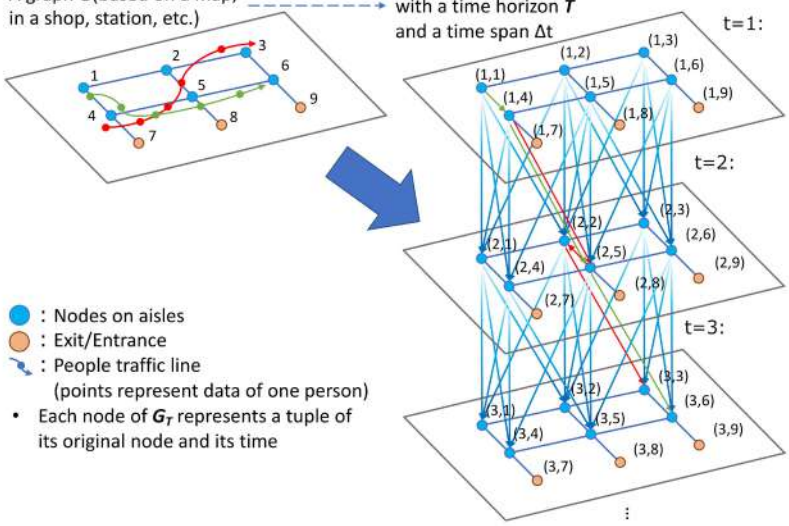
- The purpose is the detection of congestions and its origins from people traffic data obtained by cameras, sensors, and beacons

18

People Congestion Analyzing System Using Time-Expanded Graph
– Step 2: Construction of Time-Expanded Graph -

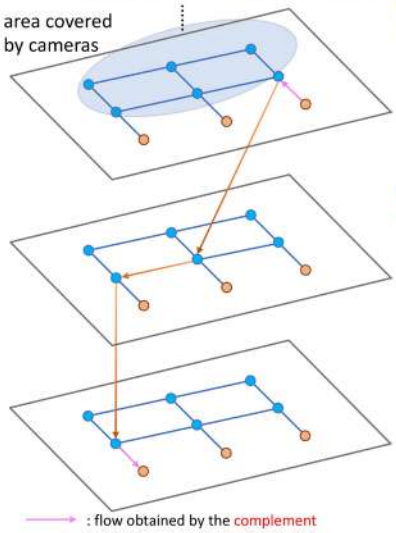
A graph ***G***(based on a map; in a shop, station, etc.)

Its time-expanded graph G_T with a time horizon T and a time span Δt



- : Nodes on aisles
- : Exit/Entrance
- : People traffic line (points represent data of one person)
- Each node of G_T represents a tuple of its original node and its time

People Congestion Analyzing System Using Time-Expanded Graph
– Step 3: Data Complement via Network Flow Algorithm (as LP) -



: flow obtained by the complement

: flow obtained by the original data

The LP(Linear Programming) formulation:

$$\text{minimize} \quad \sum_{e \in A_T} F(e)$$

$$\text{subject to} \quad \sum_{e \in \delta(v,t)} F(e) = \sum_{e \in \rho(v,t)} F(e)$$

flow conservation $(\forall v \in V \setminus S,\ t \in \{2, \cdots, T-1\})$

Keeping original flow $F(e) = r(e) \quad (\forall e \in R)$

$$F(e) + F(rev(e)) \leq U \quad (\forall e \in A_T)$$

where

- $G = (V, A)$
- $G_T = (V_T, A_T)$
- $S \subset V$: the set of entrance/exit nodes
- $R \subset A_T$: the set of edges with people traffic data
- $r : R \to \mathbb{N}$: the number of people on each edge
- $\delta(v,t) \subset A_T$: the set of edges from (v,t)
- $\rho(v,t) \subset A_T$: the set of edges to (v,t)
- U : an upper bound of people passing through an edge
- $rev(e) := ((w,t),(v,t'))$ for $e = ((v,t),(w,t'))$
- $F : A_T \to \mathbb{Z}_{\geq 0}$: a complemented flow

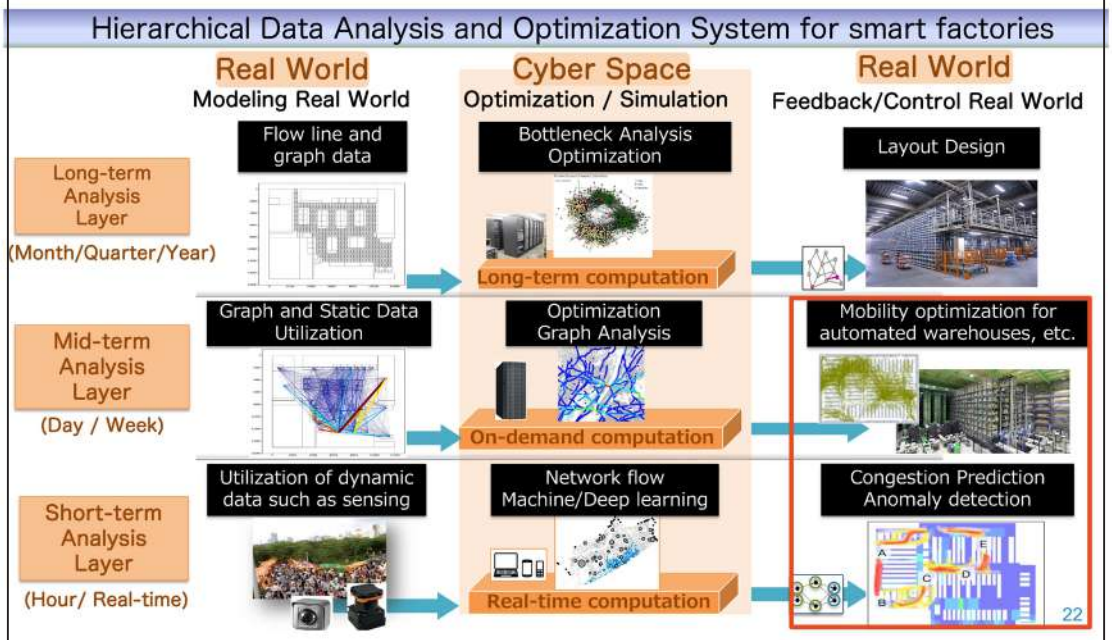
Data Complement via Network Flow Algorithm Computational results

Nodes = 907,200 : # Edges = 12,367,882 (time-expanded graph : one hour)
LP : 6138118 rows, 12367882 columns and 27549816 nonzeros

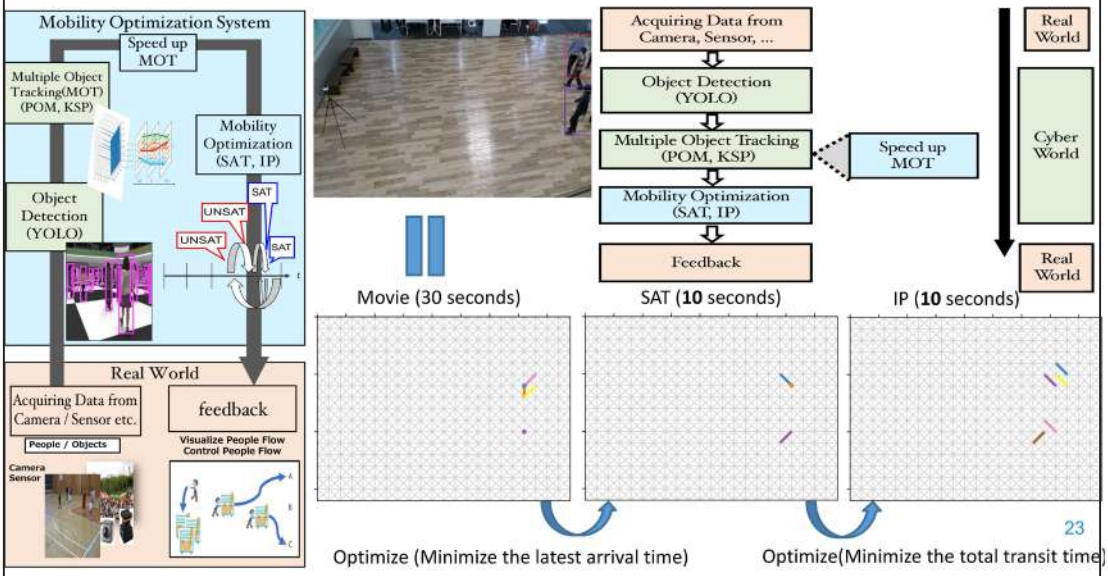
Instance : flow_completion.mps (LP)
Software : Computation Time
- ○ **CBC 2.10.5 : 52s**
- ○ **CPLEX 22.1.0.0 : 72s**
- ○ **GUROBI 9.5.2 : 54s**

Computational Server
CPU : Intel(R) Core(TM) i9-7940X CPU @ 3.10GHz x 14 cores
Memory : 128GB
OS : CentOS 7.9

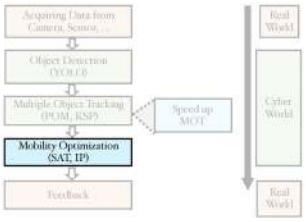
21



Cyber Physical System(CPS) & Mobility Optimization Engine



Mobility Optimization – Formulation (SAT)



$$\varphi_{T_{end}}((x^{v,w}_{k,t})_{k,t,v,w}) := \left(\bigwedge_{\substack{v \in V_{opt} \setminus \{a_k\} \\ w,k,t}} A_1(k,t,v,w) \right) \wedge \left(\bigwedge_{\substack{v \in V_{opt} \setminus \{b_k\} \\ u,k,t}} A_2(k,t,u,v) \right) \wedge \left(\bigwedge_{k=1}^{K} B_1(k) \right) \wedge \left(\bigwedge_{k=1}^{K} B_2(k) \right) \wedge \left(\bigwedge_{\substack{k,l \in \{1,\dots,K\}, k \neq l \\ t \in \tau_{k,l}, v \in V_{opt}}} C(k,l,t,v) \right)$$

$$A_1(k,t,v,w) := (\neg x^{v,w}_{k,t}) \vee \bigvee_{u \in \mathcal{N}_{opt}(v)} x^{u,v}_{k,t-1} \quad (\forall v \in V_{opt} \setminus \{a_k\}, \forall w,k,t)$$

$$A_2(k,t,u,v) := (\neg x^{u,v}_{k,t-1}) \vee \bigvee_{w \in \mathcal{N}_{opt}(v)} x^{v,w}_{k,t} \quad (\forall v \in V_{opt} \setminus \{b_k\}, \forall u,k,t)$$

Flow conservation

$$B_1(k) := \bigvee_{t=S_k}^{T_k-1} \bigvee_{w \in \mathcal{N}_{opt}(a_k)} x^{a_k,w}_{k,t} \quad (k = 1,\dots,K)$$

$$B_2(k) := \bigvee_{t=S_k}^{T_k-1} \bigvee_{v \in \mathcal{N}_{out}(b_k)} x^{v,b_k}_{k,t} \quad (k = 1,\dots,K)$$

Departure / Arrival time

$$C(k,l,t,v) := \begin{cases} \bigwedge_{w_1,w_2 \in \mathcal{N}_{opt}(v)} (\neg x^{w_1,v}_{k,t}) \vee (\neg x^{w_2,v}_{l,t}) & (v \in \beta) \\ \bigwedge_{w_1,w_2 \in \mathcal{N}_{opt}(v)} (\neg x^{v,w_1}_{k,t}) \vee (\neg x^{v,w_2}_{l,t}) & (v \notin \beta) \end{cases} \quad (\forall k,l,\forall t \in \tau_{k,l})$$

Capacity

β: the set of terminal nodes of the transits

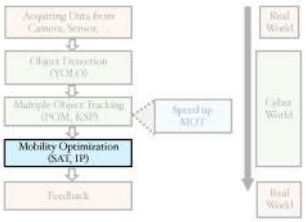
$$T_k := \min\{T_{end}, L_k\}$$

$$\tau_{k,l} = \{S_k, \dots, T_k\} \cap \{S_l, \dots, T_l\}$$

$\varphi_{T_{end}}$ is satisfiable

$\Longleftrightarrow$ all transits can be finished by T_{end}

Mobility Optimization – Formulation (IP)



find the minimum value of T_{end}

⇩

- fix T_{end} as a timelimit
- minimize the total transit time by IP

$$\text{minimize} \sum_{k=1}^{K} \sum_{t=S_k}^{T_k} \sum_{v \in \mathcal{N}_{opt}(b_k)} (t - S_k - d_k) x_{k,t}^{v,b_k}$$

subject to

Flow conservation

$$\sum_{w \in \mathcal{N}_{opt}(v)} x_{k,t}^{v,w} = \sum_{u \in \mathcal{N}_{opt}(v)} x_{k,t-1}^{u,v} (\forall v, k, t)$$

Departure / Arrival time

$$\sum_{t=S_k}^{T_k} \sum_{w \in \mathcal{N}_{opt}(a_k)} x_{k,t}^{a_k,w} = 1 (\forall k)$$

Capacity

$$\sum_{t=S_k}^{T_k} \sum_{v \in \mathcal{N}_{opt}(b_k)} x_{k,t}^{v,b_k} = 1 (\forall k)$$

$$x_{k,t}^{v,w_1} + x_{l,t}^{v,w_2} \leq 1$$
$$(\forall v \in V_{opt} \setminus \beta, \forall w_1, w_2 \in \mathcal{N}_{opt}(v), \forall k, l, \forall t \in \tau_{k,l})$$
$$x_{k,t}^{w_1,v} + x_{l,t}^{w_2,v} \leq 1$$
$$(\forall v \in \beta, \forall w_1, w_2 \in \mathcal{N}_{opt}(v), \forall k, l, \forall t \in \tau_{k,l})$$

β: the set of terminal nodes of the transits

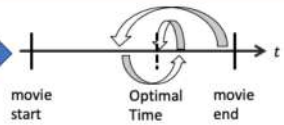
$$T_k := \min\{T_{end}, L_k\}$$
$$\tau_{k,l} = \{S_k, \ldots, T_k\} \cap \{S_l, \ldots, T_l\}$$

Mobility Optimization : Computational results

Solver
○ SAT (Satisfiability problem) : Minisat Ver. 2.2.0
○ IP (Integer optimization problem) : Gurobi Ver. 9.5.2

Computation Time

○ **SAT : 16m50s (Binary search)**
○ **IP : 1m59s**

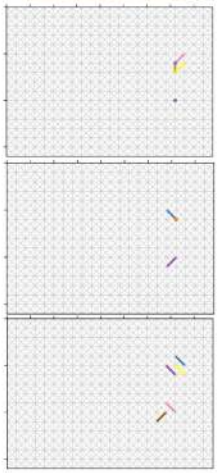


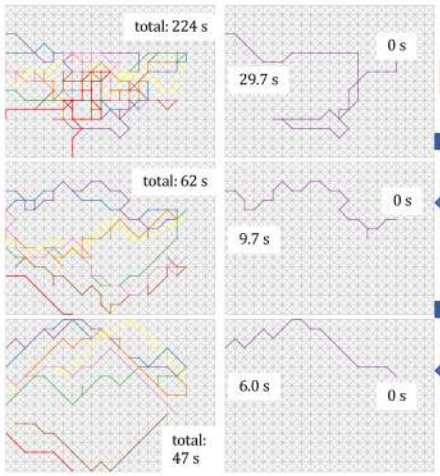
Computational Server
CPU : Intel(R) Core(TM) i9-7940X CPU @ 3.10GHz x 14 cores
Memory : 128GB
OS : CentOS 7.9

26

Mobility Optimization – Experiments

SAT : Optimize (Minimize the latest arrival time)
IP : Optimize(Minimize the total transit time)





THE DEPARTURE AND ARRIVAL TIME OF EACH ID[S]

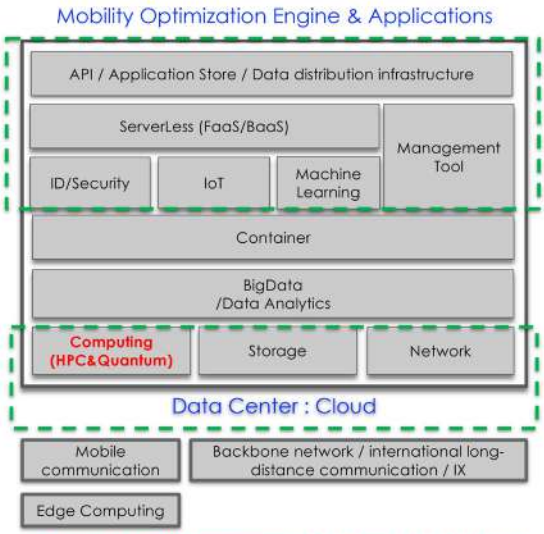
ID	Original	SAT	IP
1	0~27.3	1.3~10	0~6.3
2	0~28.7	5~9	2.7~10
3	0~29.7	0~9.7	0~6
4	0~29.7	0~8.7	2.7~10
5	0~27.3	10~9.7	0~6.3
6	0~29.7	3~10	0~5.3
7	0~29.7	0~10	0~6
8	7.7~29.7	7.7~10	7.7~10
total	224	62	47
latest	29.7	10	10

- The SAT solution suggests all transits can be finished in 10 seconds (~3x faster)
- The IP solution finally achieved **4.7x faster transit** w.r.t. total transit time **(224s → 47s)**

Overall architecture | Overview

28

- **API / Application Store / Data distribution infrastructure:** Platform functions that can use various APIs, and online store functions that users can find and use software and services they need
- **ServerLess(FaaS/BaaS) :**
  - Function-based execution environment and back-end functions
- **ID/Security**
  - ID account management, firewall, etc.
- **IoT :**
  - Essential for industries that handle equipment data such as manufacturing and power / traffic
- **Machine Learning :**
  - Provides large-scale model construction and learning processing using GPU etc.
- **BigData/Data Analytics/DB :**
  - Supports large-scale data analysis. Provided for large-scale data providers such as financial companies, and for real-time-oriented companies such as manufacturing
- **Computing:**
  - Provides machine accelerator function with TPU (Tensorflow Processing Unit) configured with GPU or ASIC
- **Storage:**
  - Provides various types according to access frequency, access speed, and reliability
- **Data center:**
  - Provide facilities / operational maintenance



※ It is assumed that the implementation time / function for each function is determined by service requirements

28

Towards an optimal operation of automated storage and retrieval system with multiple machines

Hiroki ISHIKURA

Kyushu University, Japan
tomomi_ishikura@kyudai.jp

We aim to improve the efficiency of a new type of automated storage and retrieval systems (AS/RSs) called multi-control automated storage and retrieval systems (MC-AS/RSs). MC-AS/RSs have multiple storage/retrieval (S/R) machines that operate independently according to storage and retrieval requests. Consequently, MC-AS/RSs can transport loads farther without using human labor, thereby requiring fewer human resources than conventional AS/RSs. However, the structure and control method of AS/RSs are complex because multiple S/R machines must be controlled simultaneously. Therefore, when operating an MC-AS/RS, many factors must be considered, such as the sequence and transport timing. We propose an optimization method using a time-expanded network (TEN) to solve these problems and generate optimal operational methods. First, our method models an AS/RS with a TEN to calculate the optimal sequence and conveyance timing while considering the movements of multiple S/R machines. Second, we formulate the operational efficiency of the MC-AS/RS as a problem of minimizing the sum of execution times of requests on the TEN. Finally, we generate the request order necessary for practical use based on the results. The mechanisms implemented to achieve include a generator, optimizer, and scheduler. Our experiments confirm that this method reduces the total execution time of requests compared with other rule-based methods. This method enables us to propose an efficient operation method for AS/RSs with a complex structure of multiple carriers.

Towards an optimal operation of automated storage and retrieval system with multiple machines

Hiroki Ishikura*, Katsuki Fujisawa

Kyushu university, Fukuoka, Japan

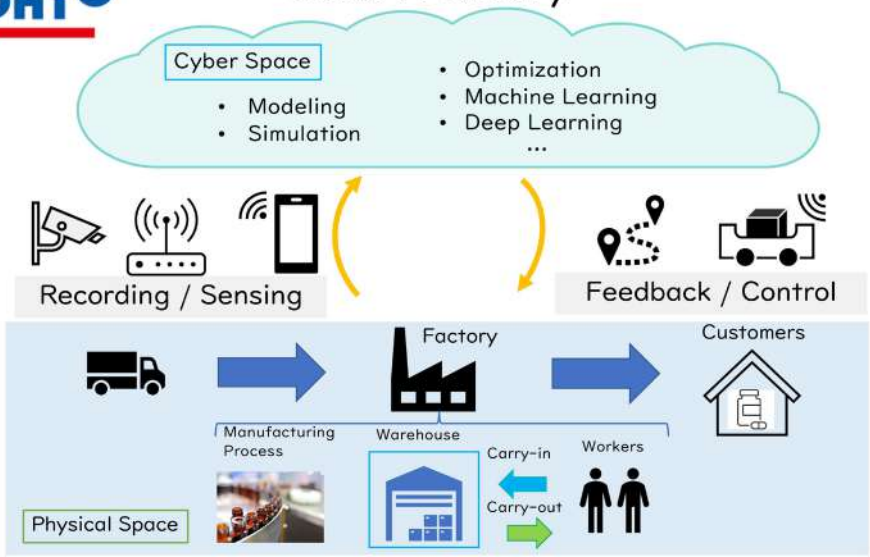
The 6th RIKEN-IMI-ISM-NUS-ZIB-MODAL-NHR Workshop
on Advances in Classical and Quantum Algorithms
for Optimization and Machine Learning

18/9/2022

Table of contents

Table of contents

Smart Factory



4

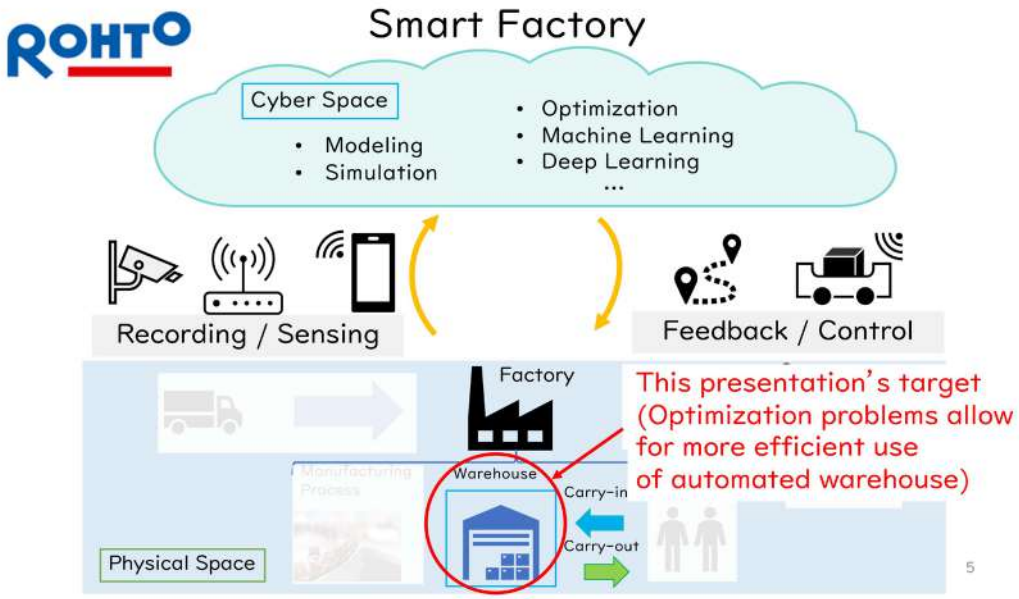
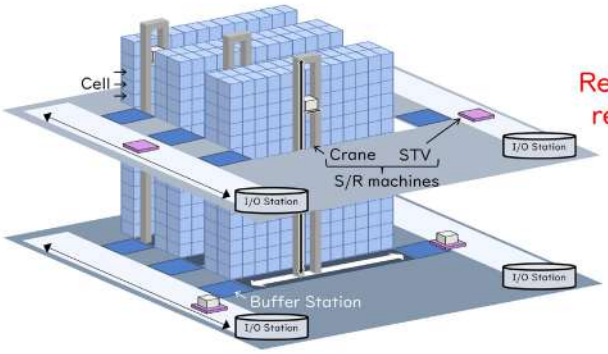


Table of contents

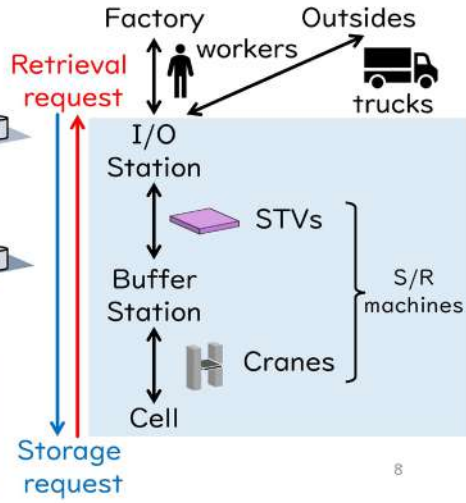
Automated Warehouse

7

Mobility in Automated Warehouses

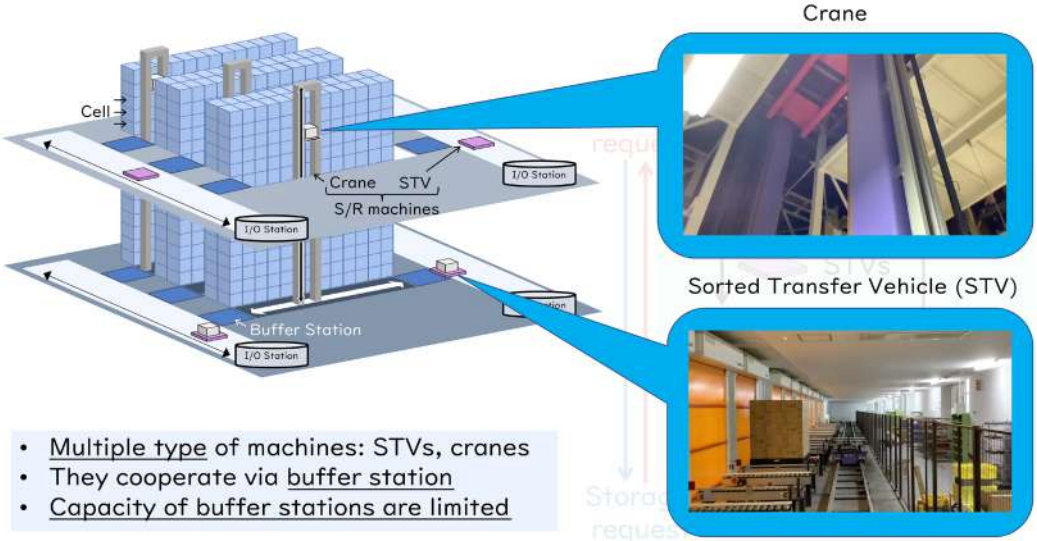


- S/R machines can store and retrieve loads without human labor
- Loads is transported in accordance with the requests.



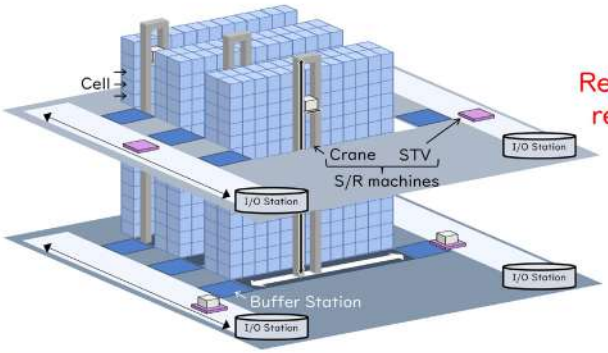
8

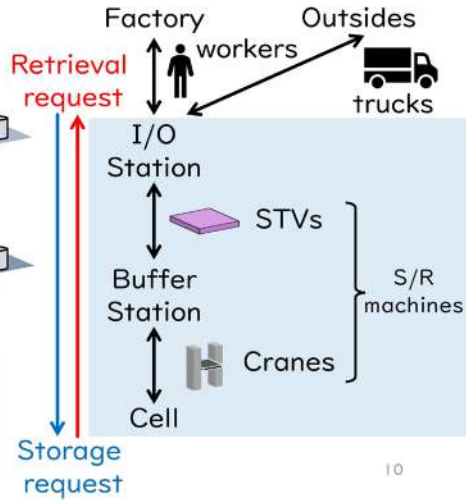
Mobility in Automated Warehouses



- Multiple type of machines: STVs, cranes
- They cooperate via buffer station
- Capacity of buffer stations are limited

Mobility in Automated Warehouses



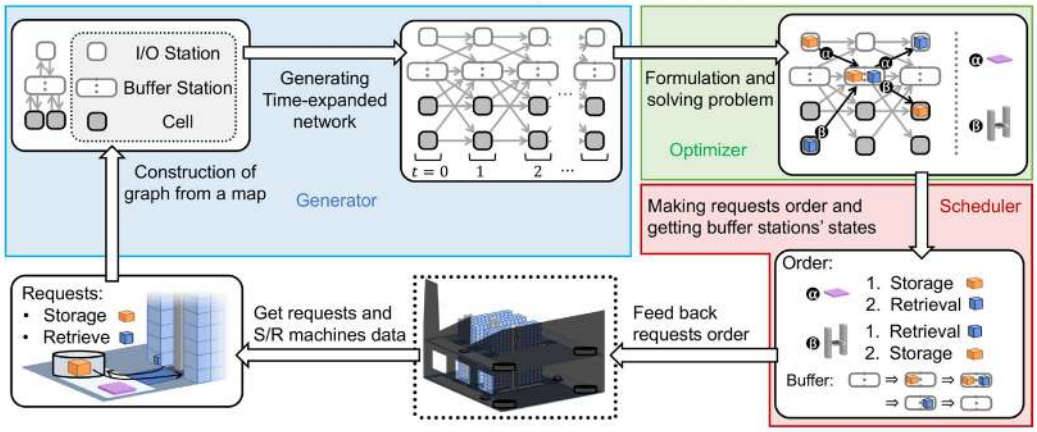


Optimization Target:
How do S/R machines process requests?
→ Minimize the total time to execute all requests

10

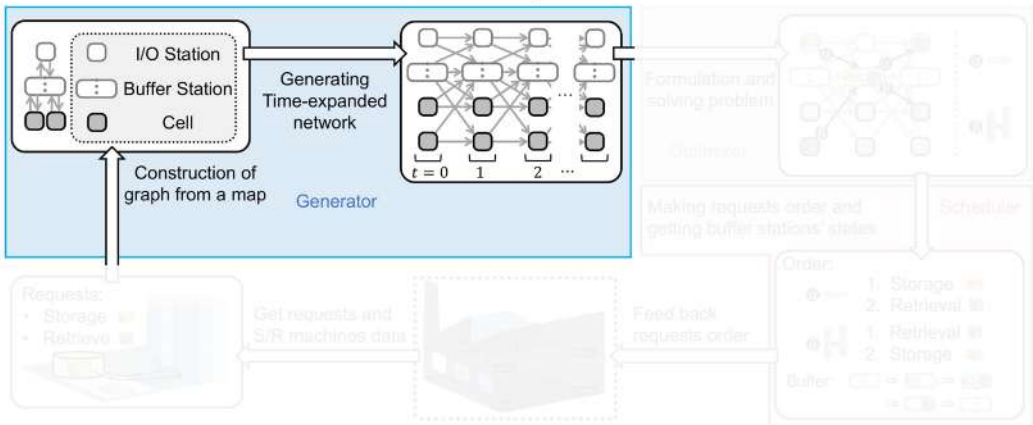
Table of contents

Overview of Proposed Method



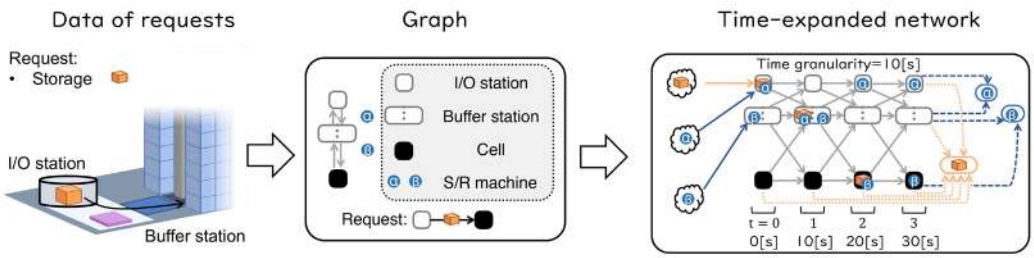
- Movements of S/R machines and loads → paths on time-expand network
- Capacity of Buffer station → capacity of node
- Optimal control → optimization problem on time-expanded network

Overview of Proposed Method



- Movements of S/R machines and loads → paths on time-expand network
- Capacity of Buffer station → capacity of node
- Optimal control → optimization problem on time-expanded network

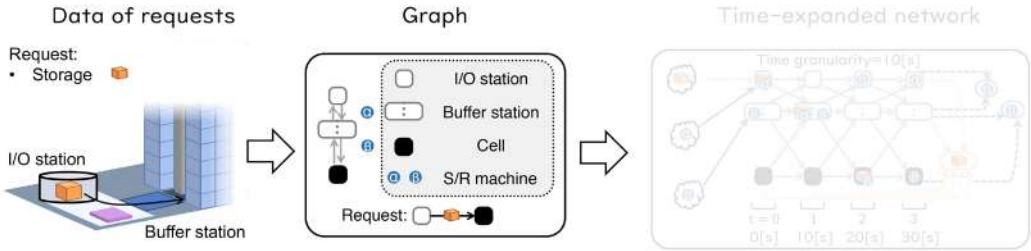
Construction of graph and time-expanded network



- Construction of a graph based on the location information used by the request
- Determination of time granularity based on the error between discretized travel time and actual travel time

14

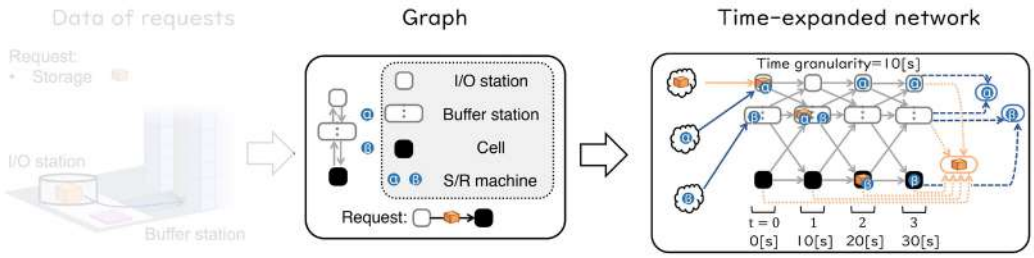
Construction of graph and time-expanded network



- Construction of a graph based on the location information used by the request
- Determination of time granularity based on the error between discretized travel time and actual travel time

15

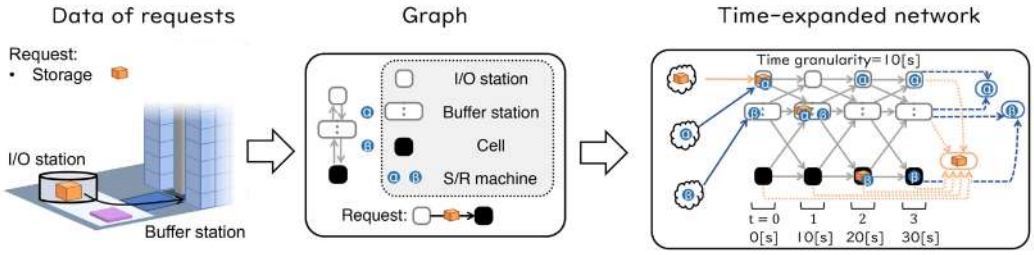
Construction of graph and time-expanded network



- Construction of a graph based on the location information used by the request
- Determination of time granularity based on the error between discretized travel time and actual travel time

16

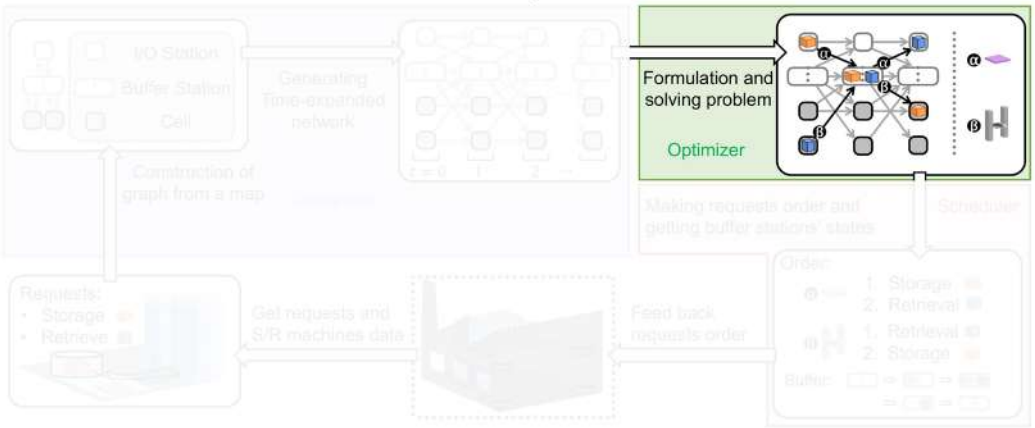
Construction of graph and time-expanded network



- Construction of a graph based on the location information used by the request
- Determination of time granularity based on the error between discretized travel time and actual travel time

17

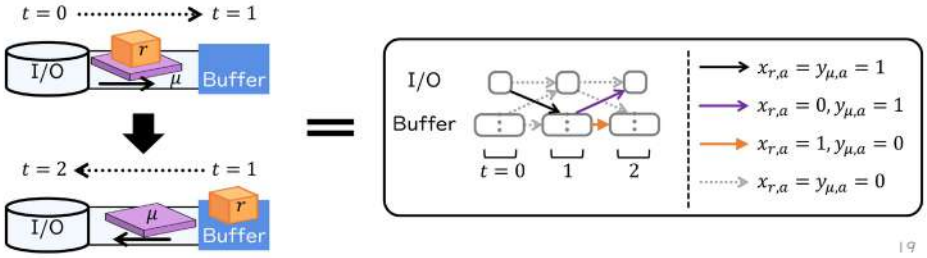
Overview of Proposed Method



- Movements of S/R machines and loads → paths on time-expand network
- Capacity of Buffer station → capacity of node
- Optimal control → optimization problem on time-expanded network

Objective function and variables

Variables:
- $x_{r,a} \in \{0,1\}$: 1 if the loads of request r passes arc a, otherwise 0
- $y_{\mu,a} \in \{0,1\}$: 1 if S/R machine μ passes arc a, otherwise 0



Objective function and variables

$$\underset{x,y}{\text{minimize}} \left[\sum_{r \in R} (f_r(x) - \tau_r) + U \sum_{r \in R} (1 - g_r(x)) \right]$$

Variables:
$x_{r,a} \in \{0,1\}, y_{\mu,a} \in \{0,1\}$

- Objective: Minimize the total time to execute all requests
  - $f_r(x) - \tau_r$: execution time to finish request r from r occurs (if r is not finished, $f_r(x) = 0$)
  - $g_r(x)$: 1 if r is finished, otherwise 0
    → If it is impossible to execute request r, we set penalty U

- $f_r(x) = \sum_{v \in V_r^{\mathcal{T}}} t(v) \times x_{r,(v,l_r')}$
- $g_r(x) = \sum_{v \in V_r^{\mathcal{T}}} x_{r,(v,l_r')}$

$V_r^{\mathcal{T}}$: The set of request r's destination nodes on time-expanded network
l_r': Request r's super sink, $t(v)$: Time of node $v \in V_r^{\mathcal{T}}$

- Path for the request r → $\{a \in A^{\mathcal{T}} \mid x_{r,a} = 1\}$
- $x_{r,a} = y_{\mu,a} = 1 \Rightarrow$ the load of request r is carried by S/R machine μ

Constraints

$x_{r,a}, y_{r,a}$ must form disjoint path

$$\sum_{r \in R} x_{r,a} \leq 1 \qquad ({}^\forall a \in A^{\mathcal{T}} \setminus H^{\mathcal{T}})$$

$$\sum_{\mu \in M} y_{\mu,a} \leq 1 \qquad ({}^\forall a \in A^{\mathcal{T}})$$

$$\sum_{a \in \delta_+(n)} x_{r,a} - \sum_{a \in \delta_-(n)} x_{r,a} = p(l_r, l_{r'}, n) \; ({}^\forall r, n)$$

$$\sum_{a \in \delta_+(n)} y_{\mu,a} - \sum_{a \in \delta_-(n)} y_{\mu,a} = p(l_\mu, l_{\mu'}, n) \; ({}^\forall \mu, n)$$

$$p(n_1, n_2, n) = \begin{cases} -1 & (n = n_1) \\ 1 & (n = n_2) \\ 0 & (\text{otherwise}) \end{cases}$$

buffer station has capacity

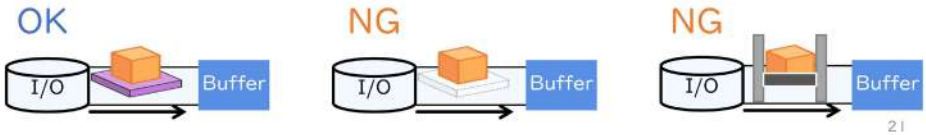
$$\sum_{r \in R} \sum_{a \in \delta_+(b(t))} x_{r,a} \leq w(b) \qquad ({}^\forall b \quad \in B^{\mathcal{T}})$$

each load must move with S/R machines

$$\sum_{r \in R} x_{r,a} \leq \sum_{\mu \in M} y_{\mu,a} \qquad ({}^\forall a \in A_G^{\mathcal{T}})$$

S/R machine has limited space to move

$$y_{\mu,a} = 0 \qquad ({}^\forall \mu \in M, {}^\forall a \in A^{\mathcal{T}} \setminus A_\mu^{\mathcal{T}})$$



21

Constraints

$x_{r,a}, y_{r,a}$ must form disjoint path

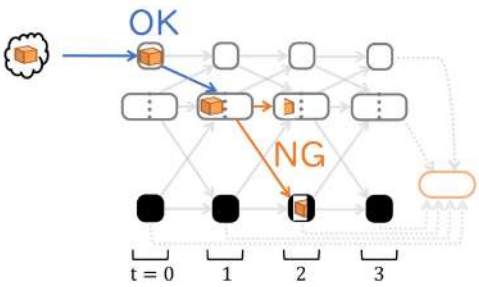
$$\sum_{r \in R} x_{r,a} \leq 1 \qquad ({}^\forall a \in A^{\mathcal{T}} \setminus H^{\mathcal{T}})$$

$$\sum_{\mu \in M} y_{\mu,a} \leq 1 \qquad ({}^\forall a \in A^{\mathcal{T}})$$

$$\sum_{a \in \delta_+(n)} x_{r,a} - \sum_{a \in \delta_-(n)} x_{r,a} = p(l_r, l_{r'}, n) \; ({}^\forall r, n)$$

$$\sum_{a \in \delta_+(n)} y_{\mu,a} - \sum_{a \in \delta_-(n)} y_{\mu,a} = p(l_\mu, l_{\mu'}, n) \; ({}^\forall \mu, n)$$

$$p(n_1, n_2, n) = \begin{cases} -1 & (n = n_1) \\ 1 & (n = n_2) \\ 0 & (\text{otherwise}) \end{cases}$$

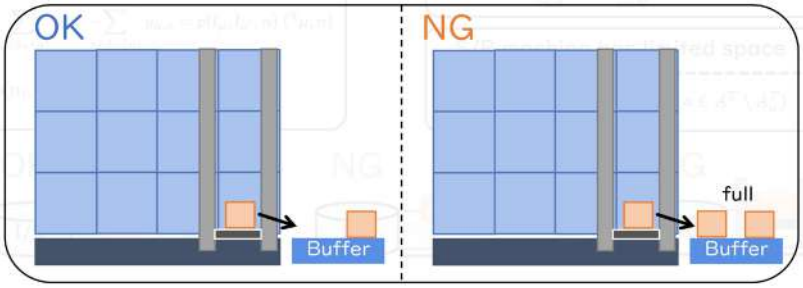


22

Constraints

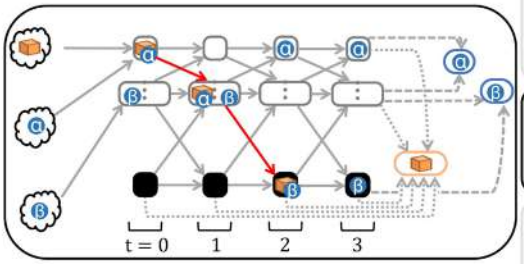
buffer station has capacity

$$\sum_{r \in R} \sum_{a \in \delta_+(b(t))} x_{r,a} \leq w(b) \quad (^\forall b \in B^T)$$



23

Constraints



buffer station has capacity

$$\sum_{r \in R} \sum_{a \in \delta_+(b(t))} x_{r,a} \leq w(b) \quad (^\forall b \in B^T)$$

each load must move with S/R machines

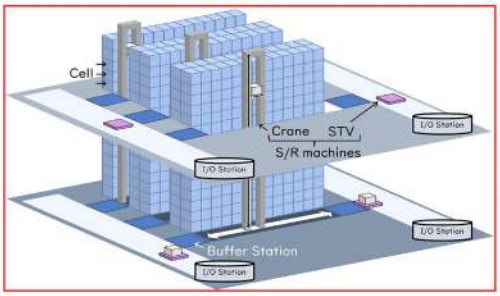
$$\sum_{r \in R} x_{r,a} \leq \sum_{\mu \in M} y_{\mu,a} \quad (^\forall a \in A_G^T)$$

S/R machine has limited space to move

$$y_{\mu,a} = 0 \quad (^\forall \mu \in M, ^\forall a \in A^T \setminus A_\mu^T)$$

24

Constraints



buffer station has capacity

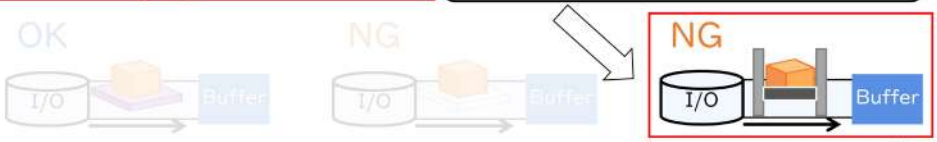
$$\sum_{r \in R} \sum_{a \in A_r(b)} x_{r,a} \le w(b) \quad ({}^\forall b \in B^{\mathcal{T}})$$

each load must move with S/R machines

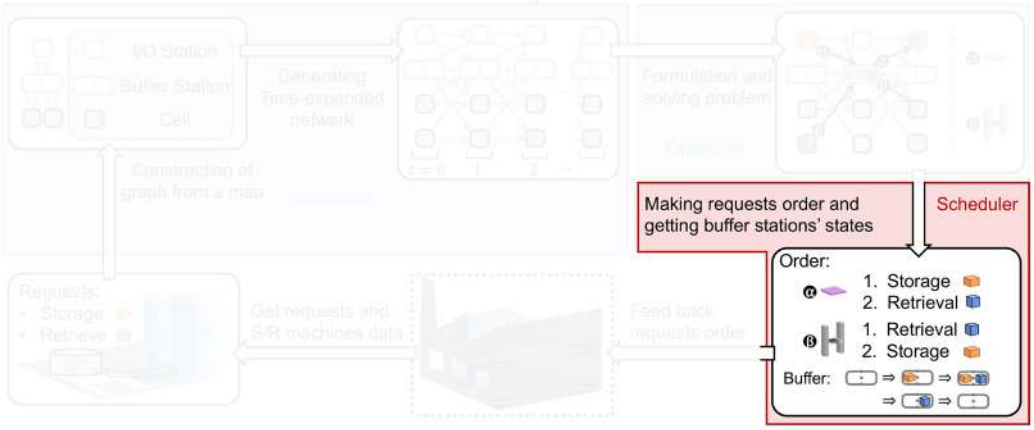
$$\sum_{r \in R} x_{r,a} \le \sum_{\mu \in M} y_{\mu,a} \quad ({}^\forall a \in A_{\mu}^{\mathcal{T}})$$

S/R machine has limited space to move

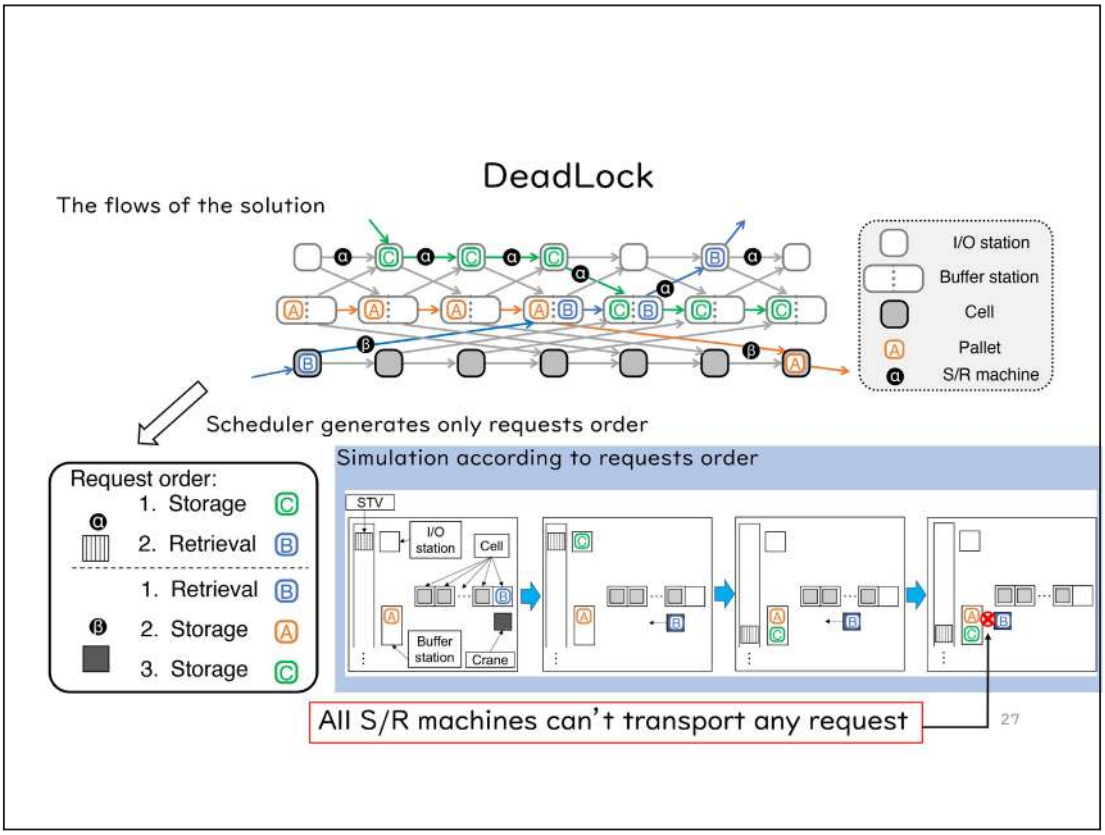
$$y_{\mu,a} = 0 \quad ({}^\forall \mu \in M, {}^\forall a \in A^{\mathcal{T}} \setminus A_{\mu}^{\mathcal{T}})$$



Overview of Proposed Method



- Movements of S/R machines and loads → paths on time-expand network
- Capacity of Buffer station → capacity of node
- Optimal control → optimization problem on time-expanded network


DeadLock
The flows of the solution
I/O station
Buffer station
Cell
Pallet
S/R machine
Scheduler generates only requests order
Request order:
1. Storage
2. Retrieval
1. Retrieval
2. Storage
3. Storage
Simulation according to requests order
STV
I/O station
Cell
Buffer station
Crane
All S/R machines can't transport any request
27

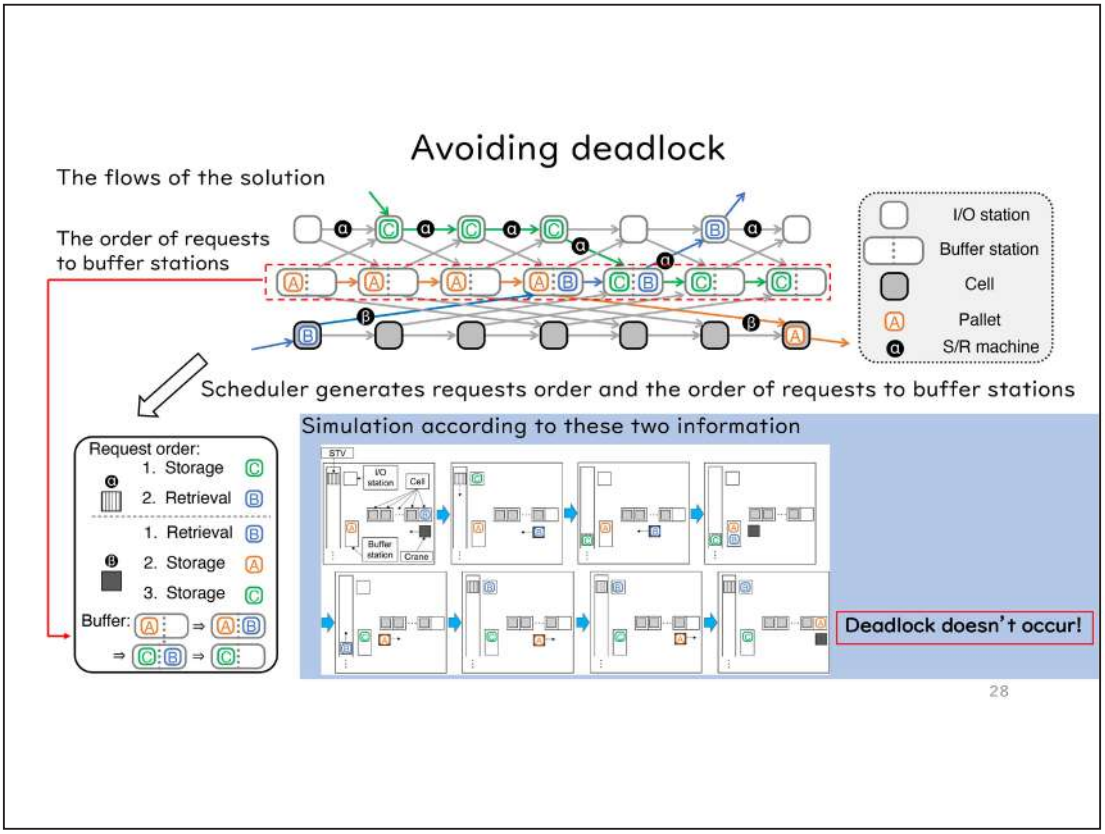
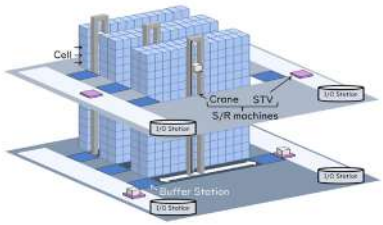

Avoiding deadlock
The flows of the solution
The order of requests to buffer stations
I/O station
Buffer station
Cell
Pallet
S/R machine
Scheduler generates requests order and the order of requests to buffer stations
Request order:
1. Storage
2. Retrieval
1. Retrieval
2. Storage
3. Storage
Buffer:
Simulation according to these two information
STV
Deadlock doesn't occur!
28

Table of contents

Numerical Experiments – Overview

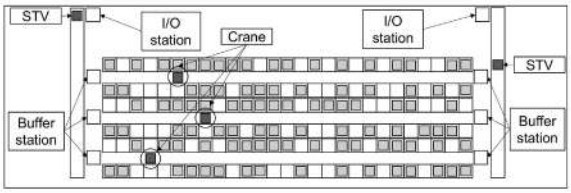
【Evaluation】
Total execution time to process all requests

【Baseline】
Nearest-Neighbor method(Details later)



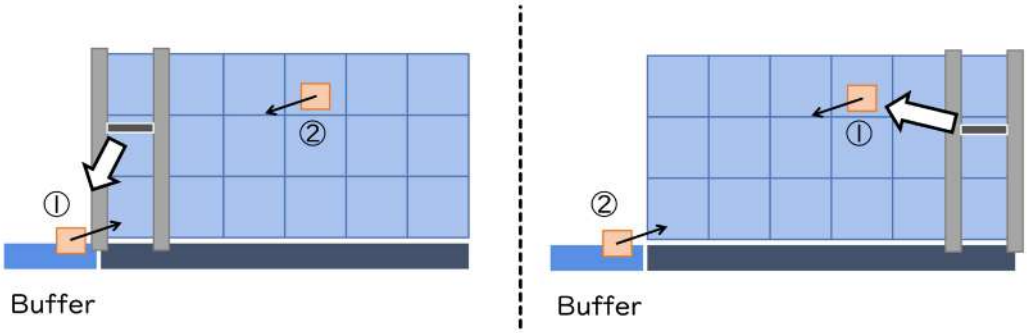
【Setting】
Time granularity : 10[s]
Number of requests : 10, 20, 30, 40
Solver : GUROBI 9.5.1
Warehouse

- 4 I/O Stations
- 12 Buffer Stations
- 4 STVs
- 3 Cranes
- 1296 cells



30

Numerical Experiments – Baseline



Nearest-Neighbor method (NN):
each S/R machine chooses its nearest request

OK calculated quickly

NG cannot consider capacity

31

Proposed method vs. baseline (Nearest Neighbor)

summary

#requests	Execution time to process all requests: NN	Ours
10	2,036.0(± 3.29)	1,904.8(± 12.35)
20	4,743.2(±131.34)	4,399.2(±16.76)
30	9,629.6(±56.06)	8,844.6(±10.50)
40	14,695.0(±76.65)	13,353.2(±54.68)

- randomly generated requests
- Evaluation by the total execution time to process all requests
- 6~10% faster operation

detail (5 calculation per data)

#requests	data	NN	Ours(Proposed)	diff
10	A	2,036.0(±3.29)	**1,904.8**(±12.35)	-131.2
	B	1,732.6(±32.33)	**1,551.2**(±9.60)	-181.4
	C	1,559.4(±36.44)	**1,480.8**(±2.40)	-78.6
	D	1,876.0(±7.35)	**1,756.2**(±0.98)	-119.8
	E	1,921.8(±19.69)	**1,640.4**(±1.96)	-281.4
20	F	4,743.2(±131.34)	**4,399.2**(±16.76)	-344.0
	G	4,974.2(±34.63)	**4,270.0**(±11.40)	-704.2
	H	5,316.2(±23.56)	**4,466.4**(±10.78)	-849.8
	I	4,926.4(±20.20)	**4,357.0**(±7.46)	-569.4
	J	4,886.6(±62.79)	**4,433.6**(±5.54)	-453.0
30	K	9,629.6(±56.06)	**8,844.6**(±10.50)	-785.0
	L	8,967.6(±139.99)	**8,563.6**(±4.63)	-404.0
	M	8,736.0(±84.53)	**7,916.2**(±19.45)	-819.8
	N	9,835.2(±136.96)	**8,711.6**(±12.14)	-1,123.6
	O	10,523.4(±144.19)	**9,936.6**(±24.65)	-586.8
40	P	14,695.0(±76.65)	**13,353.2**(±54.68)	-1,341.8
	Q	14,522.6(±123.26)	**13,505.0**(±49.67)	-1,017.6
	R	15,221.2(±270.75)	**13,279.8**(±26.03)	-1,941.4
	S	16,416.8(±121.63)	**14,944.4**(±54.75)	-1,472.4
	T	15,572.4(±206.49)	**13,896.6**(±43.23)	-1,675.8

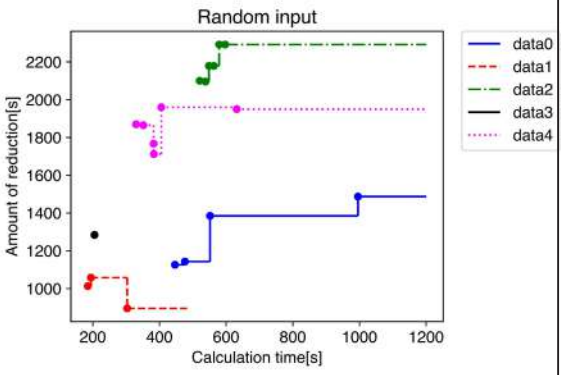
32

Difference in performance between NN and our method

- The number of requests: 40
- Amount of reduction
  $\rightarrow NN - Ours$ [s]
- Data 1 and 3 allowed
  the solution to be calculated

If we continue the calculation
until we get the solution …

Data 0 → 8,215.45(±1,413.75)[s]
Data 2 → 4,090.23(±272.11)[s]
Data 4 → 3,286.66(±362.79)[s]



Computational environment
CPU: Intel (R) Core (TM) i9-7940X@3.10GHz ×10

33

Table of contents

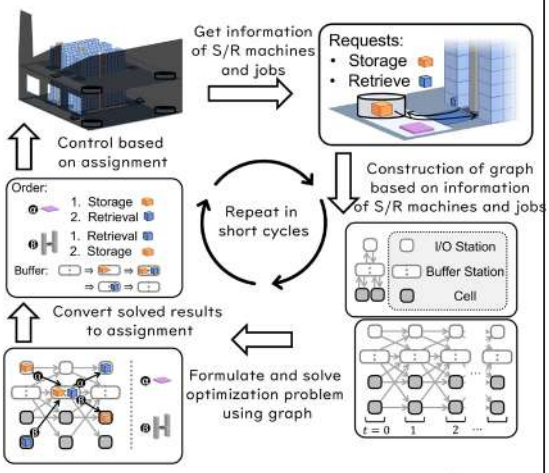
Conclusion

Optimization problems allow for more efficient use of automated warehouse

【Next Step】
Systematized to be applicable to actual factories

【Future Work】
Sometimes it takes too long to calculate

- Conjunction with a method to find assignments at high speed
  ex.) Nearest-Neighbor method
- Development of a method to find fast and efficient assignments
  ex.) reinforcement learning



35

Adaptive Cut Selection in Mixed-Integer Linear Programming

Mark Turner

Chair of Software and Algorithms for Discrete Optimization,
Institute of Mathematics, Technische Universität Berlin,
Straße des 17. Juni 135, 10623 Berlin, Germany
turner@zib.de

Cut selection is a subroutine used in all modern mixed-integer linear programming solvers with the goal of selecting a subset of generated cuts that induce optimal solver performance. These solvers have millions of parameter combinations, and so are excellent candidates for parameter tuning. Cut selection scoring rules are usually weighted sums of different measurements, where the weights are parameters. We present a parametric family of mixed-integer linear programs together with infinitely many family-wide valid cuts. Some of these cuts can induce integer optimal solutions directly after being applied, while others fail to do so even if an infinite amount are applied. We show for a specific cut selection rule, that any finite grid search of the parameter space will always miss all parameter values, which select integer optimal inducing cuts in an infinite amount of our problems. We propose a variation on the design of existing graph convolutional neural networks, adapting them to learn cut selection rule parameters. We present a reinforcement learning framework for selecting cuts, and train our design using said framework over MIPLIB 2017. Our framework and design show that adaptive cut selection does substantially improve performance over a diverse set of instances, but that finding a single function describing such a rule is difficult.

Adaptive Cut Selection in Mixed-Integer Linear Programming

M. Turner
T. Koch, F. Serrano, M. Winkler

18.09.22

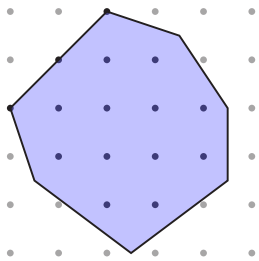
6th RIKEN-IMI-ISM-NUS-ZIB-MODAL-NHR Workshop – Tokyo

The definition of a MILP

This is a Mixed-Integer Linear Program (MILP):

$$\underset{\mathbf{x}}{\operatorname{argmin}}\{\mathbf{c}^\intercal\mathbf{x} \mid \mathbf{A}\mathbf{x} \leq \mathbf{b},\ \ \mathbf{l} \leq \mathbf{x} \leq \mathbf{u},\ \ \mathbf{x} \in \mathbb{Z}^{|\mathcal{J}|} \times \mathbb{R}^{n-|\mathcal{J}|}\}$$

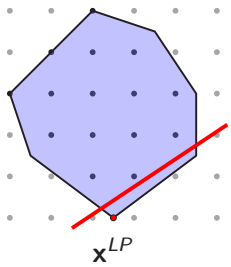
- $\mathbf{c} \in \mathbb{R}^n$ - Objective coefficient vector
- $\mathbf{A} \in \mathbb{R}^{m \times n}$ - Constraint matrix
- $\mathbf{b} \in \mathbb{R}^m$ - RHS constraint vector
- $\mathbf{l}, \mathbf{u} \in \{\mathbb{R}, -\infty, \infty\}^n$ - Lower and upper variable bound vectors
- $\mathcal{J} \subseteq \{1, \ldots, n\}$ - Set of indices of integer variables



The definition of a cut

A cut is a constraint that does not remove any feasible solutions when added.
We restrict ourselves to linear cuts.
A cut $\boldsymbol{\alpha} = (\alpha_0, \cdots, \alpha_n) \in \mathbb{R}^{n+1}$ is valid, where the set of feasible solutions is $\mathcal{I}_{\mathcal{X}}$ and:

$$\sum_{i=1}^{n} \alpha_i x_i \leq \alpha_0, \quad \forall x \in \mathcal{I}_{\mathcal{X}}, \quad \text{where } \mathbf{x} = (x_1, \cdots, x_n)$$



Cut selection

The purpose of cuts is to tighten the linear programming (LP) relaxation.
Cuts sometimes called separators, as they're often generated by separating specific points ($\mathbf{x}^{LP}$).

$$\sum_{i=1}^{n} \alpha_i x_i^{LP} > \alpha_0, \quad \text{where } \mathbf{x}^{LP} = (x_1^{LP}, \ldots, x_n^{LP})$$

Given the set of generated cuts $\mathcal{S}' = \{\boldsymbol{\alpha}_1, \cdots, \boldsymbol{\alpha}_{|\mathcal{S}'|}\}$, find a subset $\mathcal{S} \subseteq \mathcal{S}'$ to add to the formulation. That is the cut selection subproblem.

Con of adding all cuts: Large computational burden when solving larger LPs at each node
Con of adding no cuts: Substantially more nodes needed to solve to optimality

Existing literature for cut selection

Older comprehensive computational experiments on cut selection:

- Constraint integer programming - Achterberg, 2007, PhD Thesis
- Embedding {0, 1/2}-Cuts in a Branch-and-Cut Framework: A Computational Study - Giuseppe et al, 2007, doi 10.1287/ijoc.1050.0162
- Implementing cutting plane management and selection techniques - Wesselmann et al, 2011

Summary: Cut selection best as cheap heuristic. Filtering parallel cuts is most important.

Recent machine learning experiments on cut selection:

- Reinforcement learning for integer programming: Learning to cut - Y. Tang et al, 2020, MLR Press
- Learning to Select Cuts for Efficient Mixed-Integer Programming - Z. Huang et al, 2021, https://doi.org/10.1016/j.patcog.2021.108353
- Adaptive Cut Selection in Mixed-Integer Linear Programming - M. Turner et al, 2022

Summary: Improvement is possible, but non-trivial and difficult to generalise.

Cut selection rule in SCIP

Cuts are scored as a weighted sum of measurements.
Efficacy - Distance between cut and $\mathbf{x}^{LP}$.
Directed cutoff distance - Distance between cut and $\mathbf{x}^{LP}$ in the direction of some primal solution $\hat{\mathbf{x}}$.
Integer support - Ratio of non-zero coefficients that are for integer valued variables.
Objective parallelism - Absolute cosine similarity measure between cut and objective.

$$\lambda_1 * \texttt{eff} + \lambda_2 * \texttt{dcd} + \lambda_3 * \texttt{isp} + \lambda_4 * \texttt{obp}, \quad \lambda_1 + \lambda_2 + \lambda_3 + \lambda_4 = 1,$$
$$\lambda_i \geq 0 \quad \forall i \in \{1, 2, 3, 4\}, \quad \boldsymbol{\lambda} = [\lambda_1, \lambda_2, \lambda_3, \lambda_4]$$

Algorithmic Idea: Add highest scoring cut. Filter all parallel cuts. Repeat until no cuts.

Example of worst-case performance

Questions:

- Can a grid-search of the parameter space miss all cut selector parameter values that would quickly solve an instance?
- Can this be extended to an infinite family of instances, all of which only solve quickly for values outside the grid-search?
- Can this infinite family be made to, no matter the grid-search, have an infinite subset of instances that do not solve quickly?

Answer: Yes. By choosing a simplified cut selection rule, and disabling all other solver settings (e.g. branching), we manage to prove this.

Is this useful: It's incredibly cool. It motivates the need for adaptive cut selection if these fringe cases are to be handled. Modern solvers are so interconnected however, that proving this for practical solving processes is impossible.

Learn mapping from MILP instances to $\boldsymbol{\lambda}$

That is, can we learn a function that outputs values $\{\lambda_1, \lambda_2, \lambda_3, \lambda_4\}$, which induce optimal solver performance for the input instance.

Computational setup:

- Force 50 rounds of cuts
- Select exactly 10 cuts per round
- Stop solve process after the 50 rounds of cuts
- Enable all cut generators
- Test-set MIPLIB 2017. Filter out all numerically troubling instances. Root solve must take at most 20s and be non-optimal with standard parameters.
- Use MIPLIB solution as primal

Default constant SCIP 8.0 parameters: [1.0, 0.0, 0.1, 0.1] (`eff`, `dcd`, `isp`, `obp`)

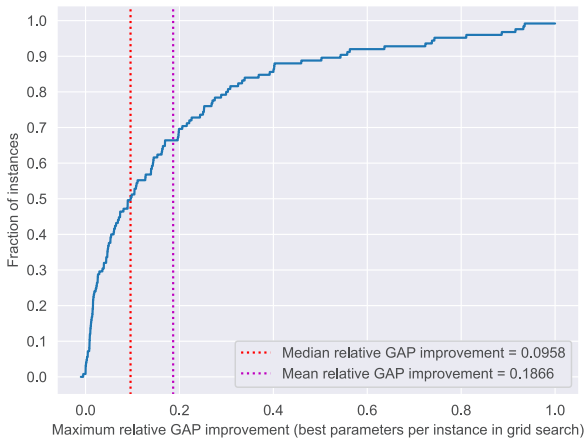
Parameter sweep (Experiment)

Improvement measure: Relative improvement of gap closed compared to run using standard cut selector parameters.
Experiment: Get the best improvement per instance with parameters from the following grid-search:

$$\sum_{i=1}^{4} \lambda_i = 1, \quad \text{where} \quad \lambda_i = \frac{\beta_i}{10}, \quad \beta_i \in \mathbb{N}, \quad \forall i \in \{1, 2, 3, 4\}$$

Aim: Provide a lower bound on the potential improvement gain. In a perfect world, our learnt function will be at-least as good as a grid-search approach.

Parameter sweep (Results)



Phrase problem using reinforcement learning

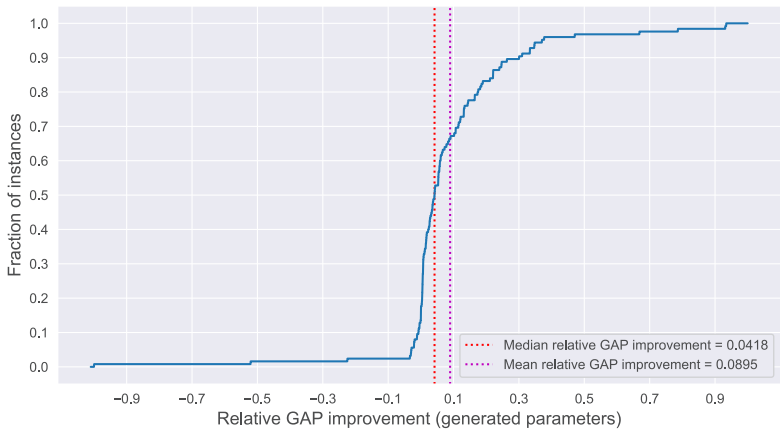
We formulate our problem as a single step Markov decision process.

Initial state: MILP instance post-presolve, but before any cuts are added.
Terminal state: MILP instance after 50 rounds of cuts.
Action: Decision of cut selector parameter values followed by applying 50 round of cuts.
Reward: Relative gap improvement compared to standard cut selector parameter values.

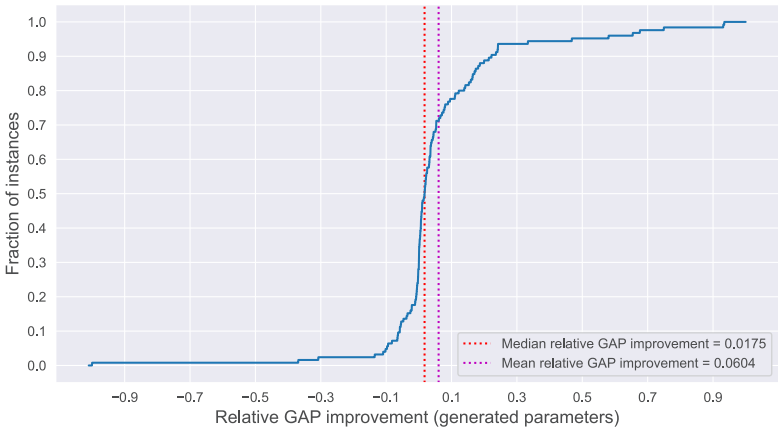
Additional Info:

- Actions drawn from a normal distribution, modelled as a graph neural network.
- Only use static features. All information is available before first LP solver.
- Approximate instance distribution using sample average approximation on MIPLIB.
- Evaluate trained network using mean of distribution.

One model per instance (Overfitting results)



Single model for all instances (Results)



Thanks for listening!

If interested and would like more details, please feel free to email me at turner@zib.de or read the paper this presentation was based on:

Adaptive Cut Selection in Mixed-Integer Linear Programming, 2022
M. Turner, T. Koch, F. Serrano, M. Winkler
Preprint: https://arxiv.org/abs/2202.10962

Outlook / Future Research

- Explore non-linear cut selection rules
- Explore different set of cut measurements
- Directly rank cuts with learned model
- Learn additional parameters in combination with cut selector parameters
- Define better standards of improved solver performance

A Quadratic Programming Approach for Performance Analysis of Energy Systems

Ryohei YOKOYAMA
Department of Mechanical Engineering, Osaka Metropolitan University, Japan
ryokoyama@omu.ac.jp

The optimization of energy systems for their design and operation necessitates the analysis of their performances under many different conditions. To analyze static (steady) and dynamic (unsteady) performances, it is necessary to solve nonlinear algebraic equations and nonlinear differential algebraic equations, respectively [1, 2]. Nonlinear equations have been solved conventionally by the Newton-Raphson method, where the solution of linearized equations is repeated until convergence. On one hand, however, the Jacobian matrices may not be regular because of network structures and operating conditions of systems. On the other hand, they may not be calculated because of violated restrictions on variables used for equations. It is a burden for analysts to take account of avoiding these situations in modeling systems. Thus, an alternative approach is necessary to reduce the burden. The singular value decomposition approach, which derives least squares and minimum norm solutions, can resolve the former situation, but cannot resolve the latter situation. In this work, a quadratic programming approach will be proposed to derive least squares and minimum norm solutions under restrictions on variables. Some examples will be presented to show the effectiveness of the proposed approach.

References

[1] Ryohei Yokoyama, Shinsuke Takeuchi, and Koichi Ito, "Thermoeconomic Analysis and Optimization of a Gas Turbine Cogeneration Unit by a Systems Approach," Proceedings of the ASME Turbo Expo 2005, Paper No. GT2005-68392, pp. 1-7, 2005, https://doi.org/10.1115/GT2005-68392

[2] Ryohei Yokoyama, "Performance Analysis and Optimization of a CO_2 Heat Pump Water Heating System," Xin-Rong Zhang and Hiroshi Yamaguchi (Eds.), Transcritical CO_2 Heat Pump: Fundamentals and Applications, Chapter 9, pp. 249-282, 2021, John Wiley & Sons, https://doi.org/10.1002/9781118380055.ch9

A Quadratic Programming Approach for Performance Analysis of Energy Systems

6th RIKEN-IMI-ISM-ZIB-MODAL-NHR Workshop on Advances in
Classical and Quantum Algorithms for Optimization and Machine Learning

September 16–19, 2022; University of Tokyo, Tokyo, Japan
September 21–22, 2022; Kyushu University, Fukuoka, Japan

Ryohei Yokoyama

Department of Mechanical Engineering
Osaka Metropolitan University
Sakai, Osaka, Japan

Introduction (1)

- **Background**
  - Importance of optimizing design and operation of energy systems to save energy and global environment
  - Necessity of grasping performance of energy systems under many different conditions
  - Necessity of solving nonlinear algebraic equations and nonlinear differential algebraic equations for static (steady) and dynamic (unsteady) performances, respectively
  - Newton-Raphson method used conventionally to solve nonlinear algebraic equations causes troubles

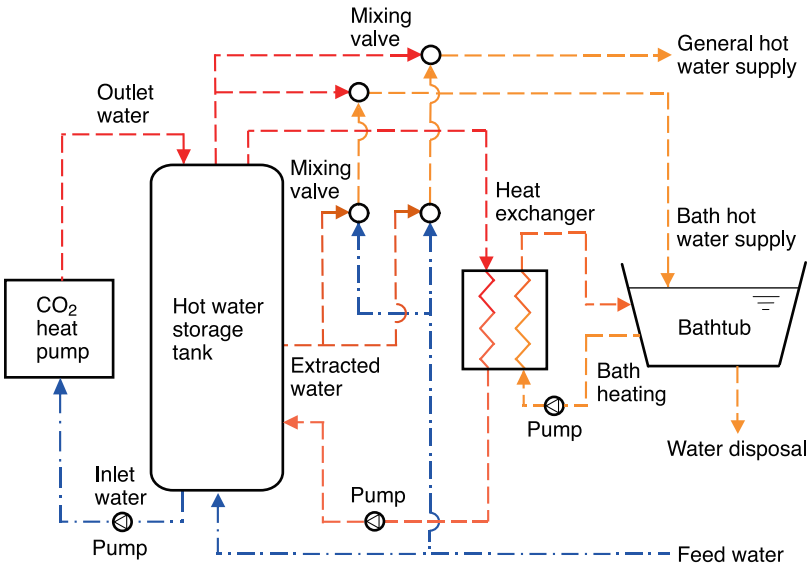
Introduction (2)

- **Singular value decomposition method can avoid some of troubles**

❍ **Objectives**

- **Proposal of quadratic programming method in place of singular value decomposition method for Newton-Raphson method in performance analysis**
- **Presentation of some fundamental examples to show effectiveness of proposed method**

Example of energy systems: CO_2 heat pump water heating system (ECO CUTE)



5

Fundamental equations for performance analysis

- Component equations
  - Mass, pressure, and energy balances
  - Other performance characteristics
- Connecting conditions
  - Mass, pressure, and energy balances
- Boundary and ambient conditions
- Operation and control conditions
- Initial conditions (only for dynamic analysis)

6

Static (steady) analysis of energy systems

- Formulation by nonlinear algebraic equations

$$\left.\begin{aligned} \boldsymbol{x} &= (x_1,\ x_2,\ \cdots,\ x_n)^{\mathrm{T}} \\ \boldsymbol{f} &= (f_1,\ f_2,\ \cdots,\ f_n)^{\mathrm{T}} \end{aligned}\right\}$$

$$\boldsymbol{f}(\boldsymbol{x}) = \boldsymbol{0}$$

- Solution by Newton-Raphson method

$$\boldsymbol{x}_{(l+1)} = \boldsymbol{x}_{(l)} - \left[\frac{\partial \boldsymbol{f}}{\partial \boldsymbol{x}}(\boldsymbol{x}_{(l)})\right]^{-1} \boldsymbol{f}(\boldsymbol{x}_{(l)})$$

Dynamic (unsteady) analysis of energy systems (1)

- **Formulation by nonlinear differential algebraic equations**

$$\left.\begin{aligned}
&\boldsymbol{x}(t) = (x_1(t),\ x_2(t),\ \cdots,\ x_n(t))^{\mathrm{T}} \\
&\boldsymbol{y}(t) = (y_1(t),\ y_2(t),\ \cdots,\ y_m(t))^{\mathrm{T}} \\
&\boldsymbol{f} = (f_1,\ f_2,\ \cdots,\ f_{n+m})^{\mathrm{T}} \\
&\boldsymbol{X} = (X_1,\ X_2,\ \cdots,\ X_n)^{\mathrm{T}}
\end{aligned}\right\}$$

$$\left.\begin{aligned}
&\boldsymbol{f}(\boldsymbol{x}(t),\ \dot{\boldsymbol{x}}(t),\ \boldsymbol{y}(t),\ t) = \boldsymbol{0} \\
&\boldsymbol{x}(t_0) = \boldsymbol{X}
\end{aligned}\right\}$$

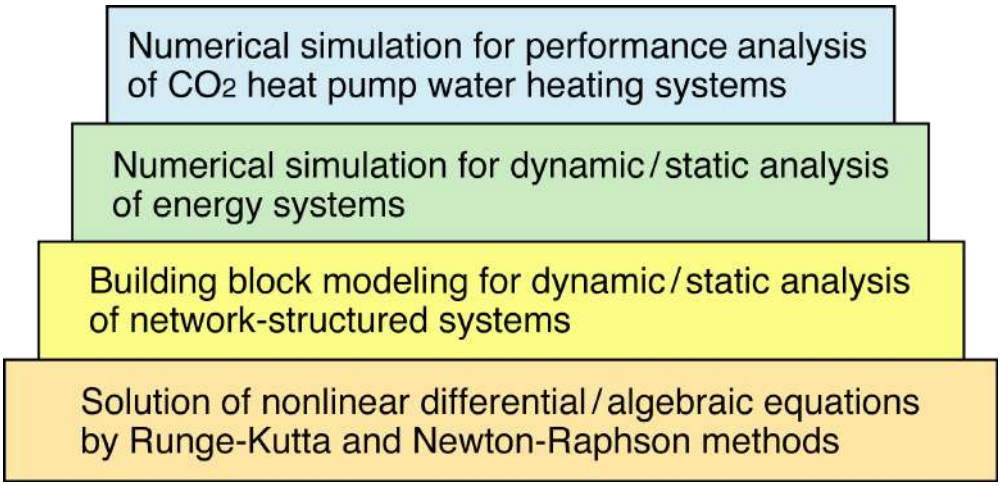
Dynamic (unsteady) analysis of energy systems (2)

- **Solution by hierarchical combination of Runge-Kutta and Newton-Raphson methods**

$$\left.\begin{aligned}
&\boldsymbol{f}(\boldsymbol{x}(t) + \dot{\boldsymbol{x}}_{[r]}k_{[r+1]}\Delta t,\ \underline{\dot{\boldsymbol{x}}_{[r+1]}},\ \underline{\boldsymbol{y}_{[r+1]}},\ t + k_{[r+1]}\Delta t) = \boldsymbol{0} \\
&\begin{Bmatrix} \underline{\dot{\boldsymbol{x}}_{(l+1)}} \\ \underline{\boldsymbol{y}_{(l+1)}} \end{Bmatrix} = \begin{Bmatrix} \dot{\boldsymbol{x}}_{(l)} \\ \boldsymbol{y}_{(l)} \end{Bmatrix} \\
&\qquad - \left[\frac{\partial \boldsymbol{f}}{\partial \dot{\boldsymbol{x}}}(\boldsymbol{x}(t) + \dot{\boldsymbol{x}}_{[r]}k_{[r+1]}\Delta t,\ \dot{\boldsymbol{x}}_{(l)},\ \boldsymbol{y}_{(l)},\ t + k_{[r+1]}\Delta t),\right. \\
&\qquad\quad \left. \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{y}}(\boldsymbol{x}(t) + \dot{\boldsymbol{x}}_{[r]}k_{[r+1]}\Delta t,\ \dot{\boldsymbol{x}}_{(l)},\ \boldsymbol{y}_{(l)},\ t + k_{[r+1]}\Delta t) \right]^{-1} \\
&\qquad \times \boldsymbol{f}(\boldsymbol{x}(t) + \dot{\boldsymbol{x}}_{[r]}k_{[r+1]}\Delta t,\ \dot{\boldsymbol{x}}_{(l)},\ \boldsymbol{y}_{(l)},\ t + k_{[r+1]}\Delta t) \\
&\qquad\qquad (r = 1,\ 2,\ \cdots\ ;\ l = 1,\ 2,\ \cdots)
\end{aligned}\right\}$$

9

General-purpose tool for performance analysis



10

Possible troubles in numerical computation

- **Trouble 1: Singularity of Jacobian matrices**
  - **Circulating fluid flows ⇨ Redundant mass balances**
  - **Products of mass flow rates and temperatures as heat flow rates in energy balances ⇨ Indefinite temperatures with zero mass flow rates**
- **Trouble 2: Violation of physical and mathematical limits for values of variables**
  - **Initial values of variables far from those after convergence ⇨ drastic changes in values of variables**
  - **Changes of operational modes ⇨ drastic changes in values of variables**

Measures to avoid troubles in numerical computation

11

- Modeling without troubles
  - Trouble 1: Removal of unnecessary variables and equations ⇨ Increase in burden for modeling and difficulty in systematic way
  - Trouble 2: Setting of appropriate initial values of variables close to those after convergence ⇨ Increase in burden for modeling and difficulty in systematic way
- Solution without troubles
  - Replacement of solution method ⇨ No burden for modeling and easiness in systematic way

Singular value decomposition (SVD) method (for static analysis)

12

- Equations to be solved

$$\left[\frac{\partial \boldsymbol{f}}{\partial \boldsymbol{x}}(\boldsymbol{x}_{(l)})\right]\left(\boldsymbol{x}_{(l+1)} - \boldsymbol{x}_{(l)}\right) = -\boldsymbol{f}(\boldsymbol{x}_{(l)})$$

- Least squares and minimum norm solution

$$\left.\begin{aligned} \left[\frac{\partial \boldsymbol{f}}{\partial \boldsymbol{x}}(\boldsymbol{x}_{(l)})\right] &= \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^{\mathrm{T}} \\ \boldsymbol{x}_{(l+1)} - \boldsymbol{x}_{(l)} &= -\left[\boldsymbol{V}(\boldsymbol{V}^{\mathrm{T}}\boldsymbol{V})^{-1}\right]\boldsymbol{\Sigma}^{-1}\left[(\boldsymbol{U}^{\mathrm{T}}\boldsymbol{U})^{-1}\boldsymbol{U}^{\mathrm{T}}\right]\boldsymbol{f}(\boldsymbol{x}_{(l)}) \\ &= -\boldsymbol{V}\boldsymbol{\Sigma}^{-1}\boldsymbol{U}^{\mathrm{T}}\boldsymbol{f}(\boldsymbol{x}_{(l)}) \end{aligned}\right\}$$

Quadratic programming (QP) method (for static analysis)

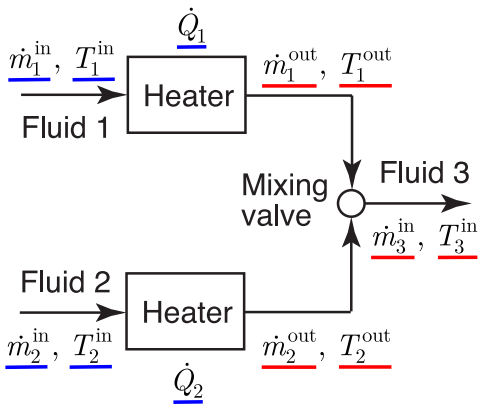
- **Least squares solution**

$$\left.\begin{array}{ll} \text{min.} & ||\boldsymbol{r}_{(l+1)}||_2^{\,2} \\ \text{sub. to} & \left[\dfrac{\partial \boldsymbol{f}}{\partial \boldsymbol{x}}(\boldsymbol{x}_{(l)})\right]\left(\boldsymbol{x}_{(l+1)} - \boldsymbol{x}_{(l)}\right) - \boldsymbol{r}_{(l+1)} = -\boldsymbol{f}(\boldsymbol{x}_{(l)}) \\ & \underline{\boldsymbol{r}} \le \boldsymbol{r}_{(l+1)} \le \overline{\boldsymbol{r}} \\ & \underline{\boldsymbol{x}} \le \boldsymbol{x}_{(l+1)} \le \overline{\boldsymbol{x}} \end{array}\right\}$$

- **Minimum norm solution**

$$\left.\begin{array}{ll} \text{min.} & ||\boldsymbol{x}_{(l+1)} - \boldsymbol{x}_{(l)}||_2^{\,2} \\ \text{sub. to} & \left[\dfrac{\partial \boldsymbol{f}}{\partial \boldsymbol{x}}(\boldsymbol{x}_{(l)})\right]\left(\boldsymbol{x}_{(l+1)} - \boldsymbol{x}_{(l)}\right) = -\boldsymbol{f}(\boldsymbol{x}_{(l)}) + \boldsymbol{r}_{(l+1)} \\ & \underline{\boldsymbol{x}} \le \boldsymbol{x}_{(l+1)} \le \overline{\boldsymbol{x}} \end{array}\right\}$$

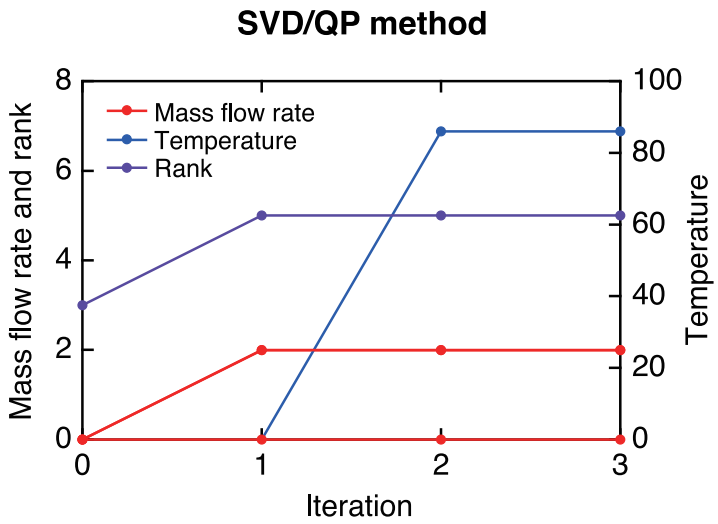
Example 1: Mixing of fluids (Problem)



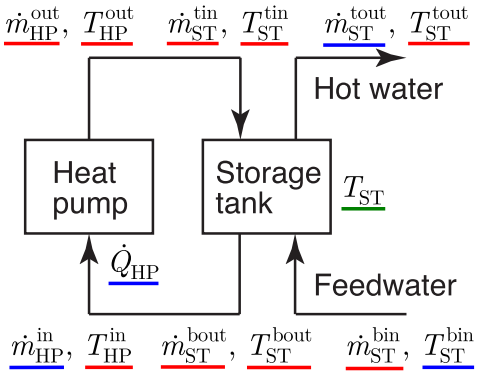
$$\left.\begin{array}{l} \dot{m}_1^{\text{in}} = \dot{m}_1^{\text{out}} \\ \dot{m}_1^{\text{in}} c T_1^{\text{in}} + \dot{Q}_1 = \dot{m}_1^{\text{out}} c T_1^{\text{out}} \\ \dot{m}_2^{\text{in}} = \dot{m}_2^{\text{out}} \\ \dot{m}_2^{\text{in}} c T_2^{\text{in}} + \dot{Q}_2 = \dot{m}_2^{\text{out}} c T_2^{\text{out}} \\ \dot{m}_1^{\text{out}} + \dot{m}_2^{\text{out}} = \dot{m}_3^{\text{in}} \\ \dot{m}_1^{\text{out}} c T_1^{\text{out}} + \dot{m}_2^{\text{out}} c T_2^{\text{out}} = \dot{m}_3^{\text{in}} c T_3^{\text{in}} \end{array}\right\}$$

- **Number of variables without derivatives = 6**
- **Number of equations = 6**

Example 1: Mixing of fluids (Results)



Example 2: Water heating, storage, and supply (Problem)

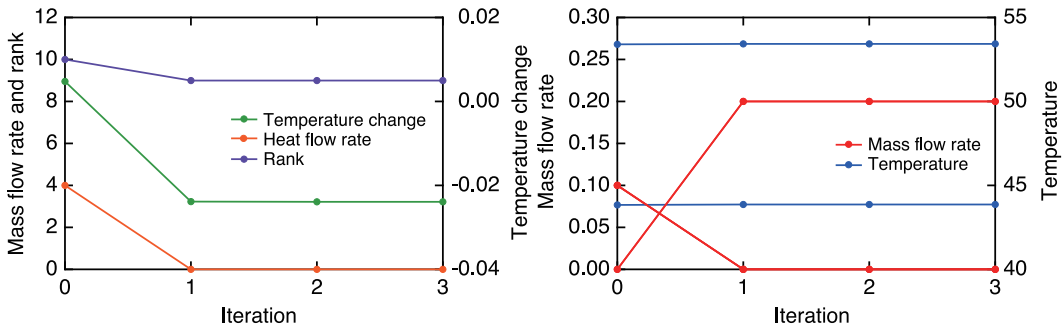


- **Number of variables with derivatives = 1**
- **Number of variables without derivatives = 9**
- **Number of equations = 11**

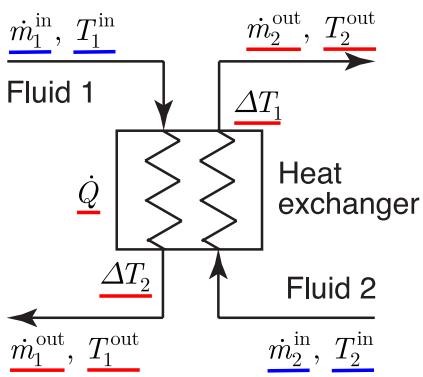
$$
\left.
\begin{aligned}
\dot{m}_{HP}^{in} &= \dot{m}_{HP}^{out} \\
\dot{m}_{HP}^{in} c T_{HP}^{in} + \dot{Q}_{HP} &= \dot{m}_{HP}^{out} c T_{HP}^{out} \\
\dot{m}_{ST}^{tin} &= \dot{m}_{ST}^{bout} \\
\dot{m}_{ST}^{bin} &= \dot{m}_{ST}^{tout} \\
m_{ST} c (dT_{ST}/dt) &= \dot{m}_{ST}^{tin} c T_{ST}^{tin} - \dot{m}_{ST}^{bout} c T_{ST}^{bout} \\
&\quad + \dot{m}_{ST}^{bin} c T_{ST}^{bin} - \dot{m}_{ST}^{tout} c T_{ST}^{tout} \\
&\quad - U_{ST} A_{ST} (T_{ST} - T_{amb}) \\
T_{ST}^{bout} &= T_{ST} \\
T_{ST}^{tout} &= T_{ST} \\
\dot{m}_{HP}^{out} &= \dot{m}_{ST}^{tin} \\
T_{HP}^{out} &= T_{ST}^{tin} \\
\dot{m}_{ST}^{bout} &= \dot{m}_{HP}^{in} \\
T_{ST}^{bout} &= T_{HP}^{in}
\end{aligned}
\right\}
$$

Example 2: Water heating, storage, and supply (Results)

SVD/QP method



Example 3: Heat exchange of fluids (Problem)

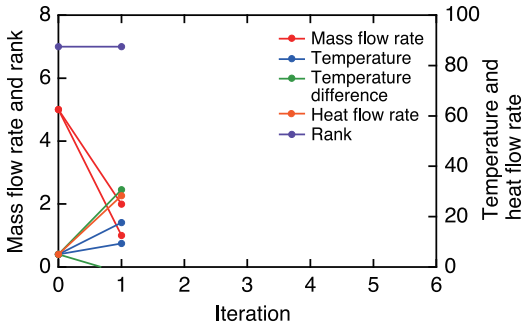


$$\left.\begin{aligned}
\dot{m}_1^{\text{in}} &= \dot{m}_1^{\text{out}} \\
\dot{m}_1^{\text{in}} c T_1^{\text{in}} + \dot{Q} &= \dot{m}_1^{\text{out}} c T_1^{\text{out}} \\
\dot{m}_2^{\text{in}} &= \dot{m}_2^{\text{out}} \\
\dot{m}_2^{\text{in}} c T_2^{\text{in}} - \dot{Q} &= \dot{m}_2^{\text{out}} c T_2^{\text{out}} \\
\Delta T_1 &= T_2^{\text{out}} - T_1^{\text{in}} \\
\Delta T_2 &= T_2^{\text{in}} - T_1^{\text{out}} \\
\dot{Q} &= UA\left(\frac{2}{3}\sqrt{\Delta T_1 \Delta T_2} + \frac{1}{3}\frac{\Delta T_1 + \Delta T_2}{2}\right)
\end{aligned}\right\}$$

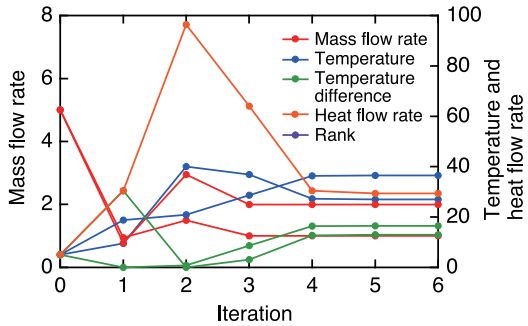
- **Number of variables without derivatives = 7**
- **Number of equations = 7**

Example 3: Heat exchange of fluids (Results)

SVD method



QP method



Conclusions (1)

- **Main results**
  - **Proposal of QP method in place of SVD method for Newton-Raphson method in performance analysis of energy systems**
  - **Applicability of SVD method in many cases except in case that values of variables violate their limits**
  - **Applicability of QP method in place of SVD method in all cases**
  - **Unsuitability of QP method for dynamic analysis because of longer computation time**
  - **Recommendation of applying SVD method primarily and QP method conditionally**

Conclusions (2)

- Future work
  - Incorporation of proposed method into general-purpose tool for performance analysis
  - Application of proposed method to static and dynamic analysis of several energy systems

Geometric Learning of Ranking Distributions"

Shizuo KAJI

Institute of Mathematics for Industry, Kyushu University, Japan
skaji@imi.kyushu-u.ac.j

Given a finite set X of n items, a complete order (permutation) of the items is called a ranking of X. A ranking distribution over X is a collection of rankings of X. Most existing models are classified into two groups; distance based and utility based. The former assumes the probability of a ranking depends on the distance from the central ranking, while the latter assumes the existence of the global utility value for each item which is independent of raters. We introduce a high-fidelity model of a ranking distribution utilising a novel geometric idea based on the hyperplane arrangement. We will also discuss efficient learning and sampling algorithms ([1]).

References

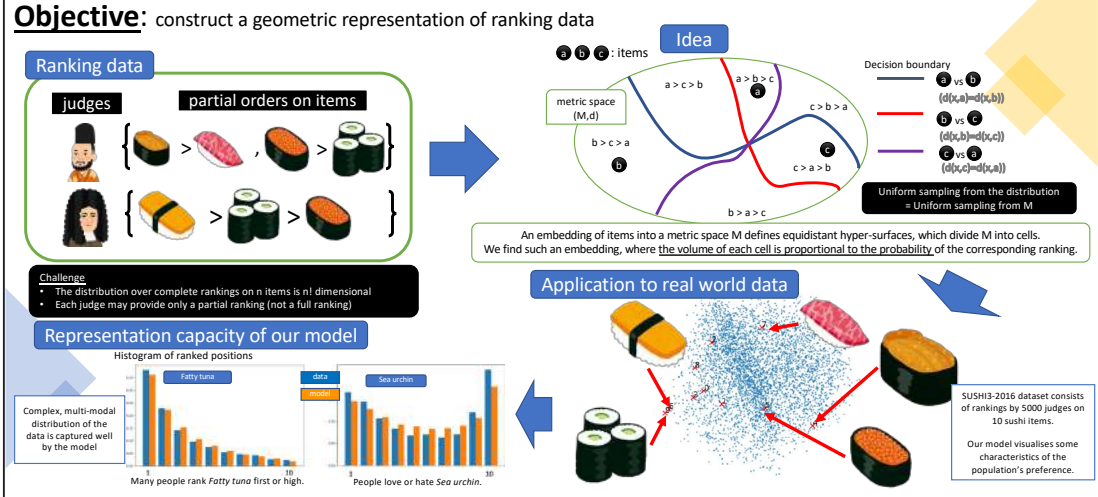
[1] Shizuo Kaji, Akira Horiguchi, Takuro Abe, Yohsuke Watanabe, “A Hyper-surface Arrangement Model of Ranking Distributions”, KDD '21: Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining, 796--804, 2021, https://doi.org/10.1145/3447548.3467253

Modelling Preference with Hyperplane Arrangement

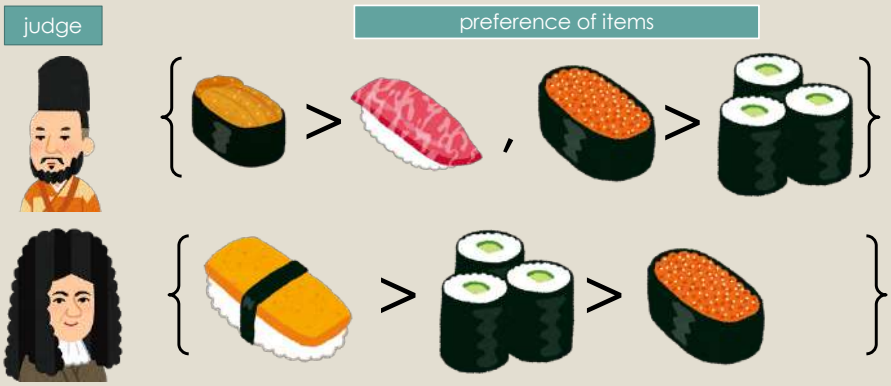
Shizuo Kaji
joint with A. Horiguchi, T. Abe, Y. Watanabe

Summary

Objective: construct a geometric representation of ranking data

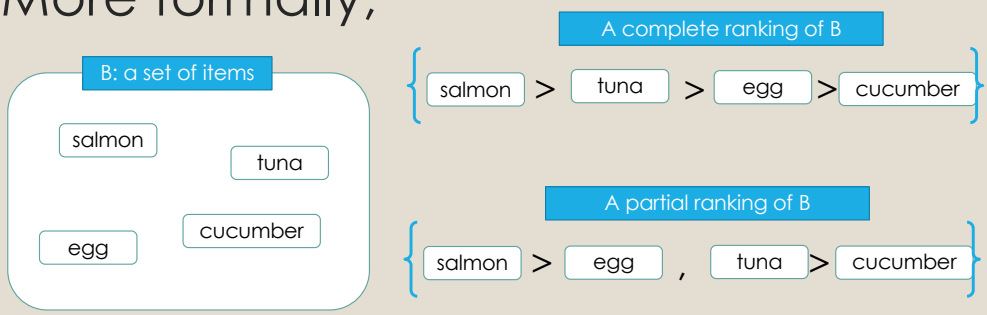


Preference



- Each judge has his or her own preference
- A judge may not know (or reveal) preference among some items

More formally,



- Preference among the set of items can be modelled by an ***order*** on B
- An order can be partial;
  e.g., in the second case above, we do not know if egg > tuna or tuna < egg
- Preference depends on judges
  (A judge is not necessarily a human but a criterion such as price and nutrition)
- Ranking data consists of rankings by many judges

Definition of Order (Ranking)

- A *partial order* on a set B satisfies
  - a < b and b > a do not occur at the same time
  - a < b and b < c imply a < c
- A *total order* on B further satisfies
  - a > b or b > a or a = b for any a,b ∈ B
- An order is described by a set of *pairwise comparisons*
  ex. { *a>b, b>c, a>d* }
- remark: a partial order = transitive closure of an acyclic directed graph

In this talk:
complete (or full) ranking = total order
incomplete (or partial) ranking = partial order

Ranking data given by a collection of pairwise comparisons

judge	higher	lower
0	a	b
0	b	d
0	a	c
1	d	b
1	d	a
2	c	a
2	b	c

Ranking data
=
rankings by many judges
=
Pairwise comparisons associated with judges

Remark: Various problems occur as special instances of ranking inference

Classification

Input

Output

Penguin > Puffin > Pigeon > … > Cat

Probability/confidence

Classifier + Data give rise to a ranking data

MODELLING RANKING DISTRIBUTION

Goal Today

Constructing a geometric model for ranking data (a set of partial rankings)

Assumptions:

- Judges are indistinguishable (i.e., the data is just a set of partial rankings)
- Each incomplete ranking can be completed
  (i.e., each judge has an unknown complete ranking)

These mean that the data is represented by a probability distribution over the permutation group S_n of n items (| B | =n).
We call such a distribution a *ranking distribution on n items*

Two main difficulties:
(1) A probability distribution over S_n is (n!-1) dimensional!
(2) How can we complete an incomplete ranking?

Existing models for ranking distribution

Two popular and basic models are

- Mallows' φ model
  - $P(w) \propto \exp\left(-\theta d_K(w, w_0)\right) \quad w \in S_n$
  - Parameters: $\theta > 0, w_0 \in S_n$
  - d_K is a distance on S_n (e.g. the Kendal tau distance)

It has the mode at w_0 and the probability decrease as the distance from w_0 increases.

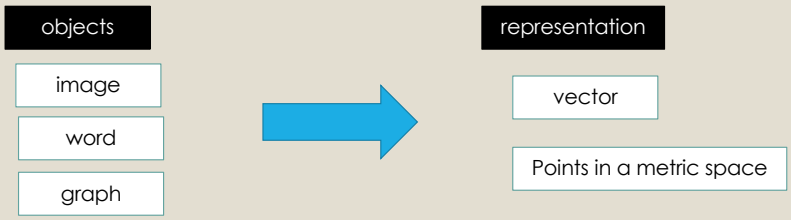
- Plackett-Luce's model
  - $P(w) \propto \prod_{i=1}^{n} \frac{\alpha(w_i)}{\sum_{j=i+1}^{n} \alpha(w_j)} \qquad w \in S_n$
  - Parameters: $\alpha: X \to R_{>0}$

Each item has a utility value and selected one by one according to the value

Reference: Marden, Analyzing and Modeling Rank Data, 1996

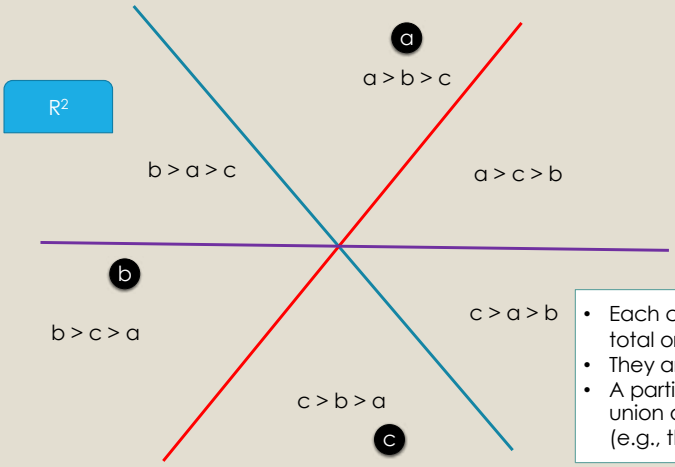
Geometric model in general

- *Representation learning* is a field of research to find a representation of (implicitly) structured objects.
- The representation space often admits geometric structures. (e.g., Euclidean space and other metric spaces)



Ex. Graph representation translates a combinatorial structure (adjacency) to a geometric one (metric)

Equidistant Line Arrangement on plane

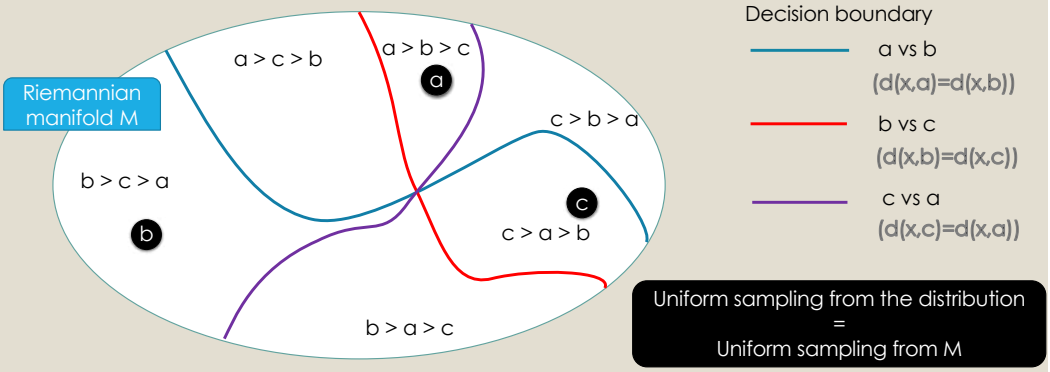


Decision boundary

- a vs b ($|x-a|=|x-b|$)
- b vs c ($|x-b|=|x-c|$)
- c vs a ($|x-c|=|x-a|$)

- Each compartment corresponds to a total ordering
- They are subdivision of Voronoi cells
- A partial ordering corresponds to the union of compartments (e.g., the Voronoi of a = {a>b and a>c})

Equidistant Hyper-surface Arrangement



Our model is an equidistant hyper-surface arrangement such that <u>the volume of each cell is proportional to the probability</u> of the corresponding ranking.

Learning Algorithm

Optimising coordinates in M of **<u>both</u>** judges and items by minimising

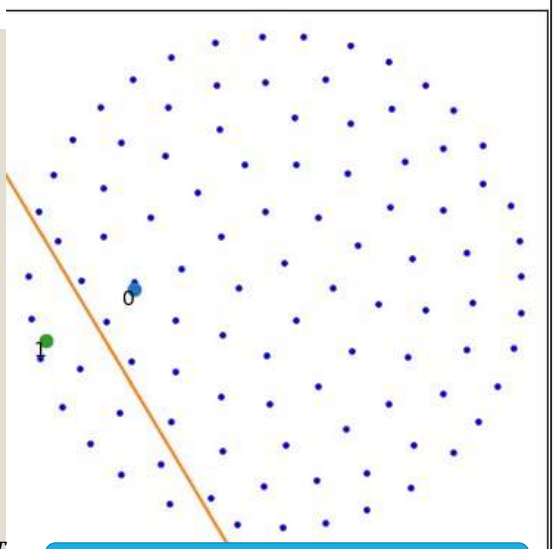
Triplet loss

$$L_T = \sum_{u \in X} \sum_{b_i >_u b_j} \max(d(b_i, u) - d(b_j, u), -\epsilon)$$

Coulomb's potential

$$L_R = \sum_{i \neq j} \frac{1}{d(u_i, u_j)^2}$$

Target function: $L = L_T + \lambda(t) L_R$
First, place judges in the right cells by L_T
Later, adjust volume by L_R



90 judges prefer item 0, 10 judges item 1
M = disk

Dimension constraints

Not all arrangements can be realised as equidistant arrangement.
(e.g., three equidistant lines of a triangle meet at the circumcentre)

Theorem
An equidistant arrangement in D^m of (generic) n points, we have n! cells if and only if $n-2 < m$.

- This gives a clue on the choice of the embedding dimension m.
- Note that the dimension of the representation space is mn.

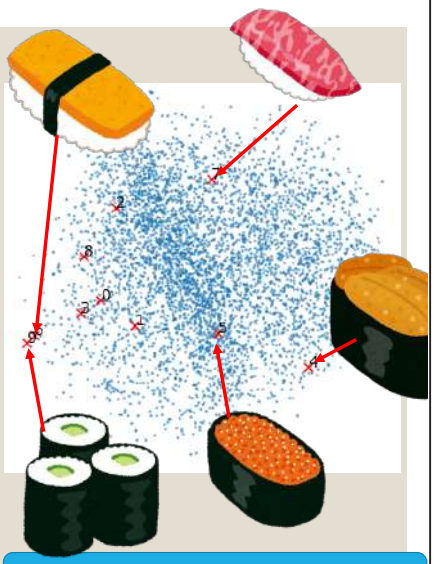
EXPERIMENTS WITH SUSHI DATASET

The SUSHI3-2016 dataset

- Consists of complete rankings of **10 items** by **5000 judges**
- Collected by T. Kamishima et al.
- Available at http://www.kamishima.net/sushi/

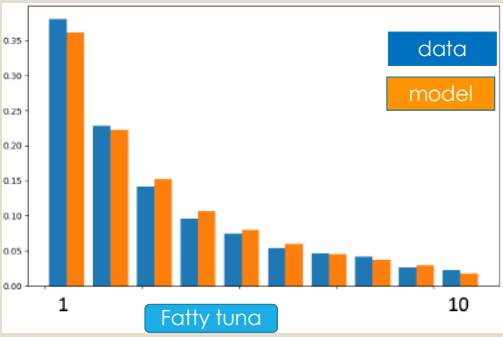
Learning the model

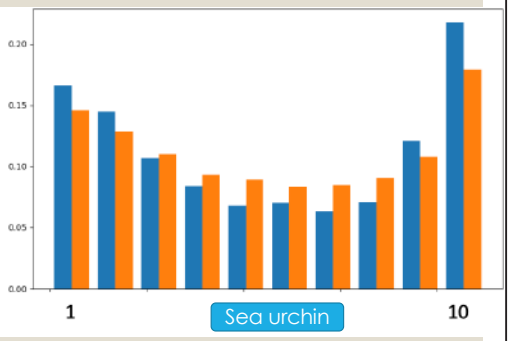
- We chose $M = D^9$ as the metric space.
  - The theorem suggests dim = | B | -1.
  - The results was almost same for $M = D^{10}$
- It took about two hours on Ryzen 2990WX.
- Visualisation was interpretable:
  - Fatty tuna has a large Voronoi region
  - Egg and Cucumber are similar and not popular
  - Sea urchin and Salmon roe are similar and distinctive



PCA projection of D^9

Representation capacity: ranking position



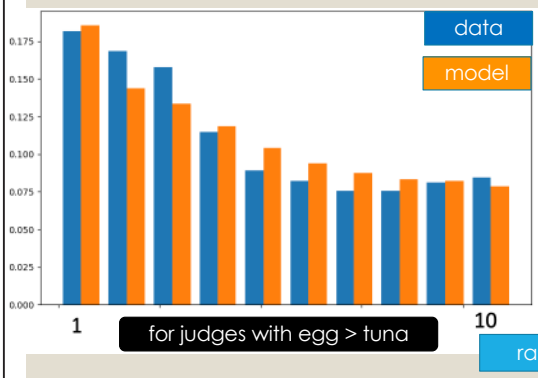


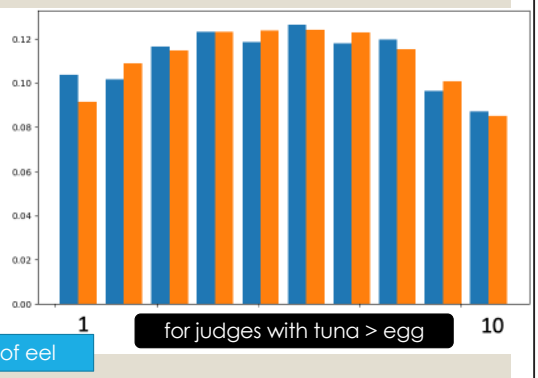
Many people rank Fatty tuna first or high.

People love or hate Sea urchin.

Complex, multi-modal distribution is captured well by the model

Representation capacity: conditional probability





rank of eel

The ranked position of eel (non-raw) depends heavily on judges' preference between tuna and egg, which the model successfully captures.

Generalisation capacity

Experimental setting

Not necessarily top-4

Input: Each judge reveals his or her ranking of, say, four items.

Output: The distribution for all ten items

input

Judge u: a > c > b > d

Judge v: e > g > d > b

...

output

The probability of a>b>c>d>e>f>g>h>i>j is 0.01

The probability of b>f>e>i>j>a>g>h>d>c is 0.02

...

This can be used for prediction:
For example, "what is the conditional probability of g being ranked in the first by a judge with a>c>b?"

Our model

Evaluation metrics for top-4 rankings

Comparison target: a = 10

α	2	4	6	8	10
Corr	0.165	0.795	0.840	0.868	0.886
sKL	0.370	0.154	0.150	0.142	0.121
W	2.502	0.832	0.770	0.666	0.457

model	Uniform	Plackett-Luce
Corr	0.025($\pm$0.058)	0.436
sKL	0.388($\pm$0.001)	0.283
W	2.51($\pm$0.110)	2.03

Each judge revealed a partial ranking among α items out of 10.

Evaluated for the top-4 ranking with

- Correlation of probability (higher is better)
- symmetrised KL-divergence (lower is better)
- Wasserstein distance (lower is better)

How to interpret:
If you are given the ranking of $\alpha = 4$ random items from each judge (the choice of four items varies for each judge), you can tell the population's top-4 ranking distribution fairly well (corr = 0.795).

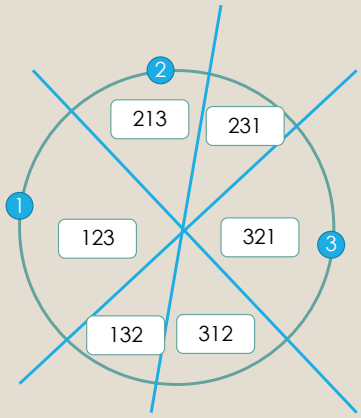
Conclusion

A model for ranking data which

- has high representation capacity
- utilises a geometric structure as regularisation
- is relatively low dimensional
- is easy to sample from
- is mathematically interesting
- comes with an implementation: https://github.com/shizuo-kaji/rankLearning

Open problem

- Parametrisation of arrangements
- Efficient sampling under a condition
- Stricter bounds for embedding dimension
- Choice of the ambient manifold
- Combinatorics of arrangement as a topological invariant (c.f. the homotopy type of the configuration space can sometimes distinguish homeo types)
- Optimal transport on S_n (e.g., how the combinatorics helps computation?)
- Generalisation to partial rankings
- Evaluation by myself by eating a lot of sushi



Toy example of S^1

Always antipodal

Convex integer optimization with Frank-Wolfe methods

Sebastian POKUTTA

Zuse Institute Berlin and TU Berlin
pokutta@zib.de

Mixed-integer nonlinear optimization is a broad class of problems that feature combinatorial structures and nonlinearities. Typical exact methods combine a branch-and-bound scheme with relaxation and separation subroutines. We investigate the properties and advantages of error-adaptive first-order methods based on the Frank-Wolfe algorithm for this setting, requiring only a gradient oracle for the objective function and linear optimization over the feasible set. In particular, we will study the algorithmic consequences of optimizing with a branch-and-bound approach where the subproblem is solved over the convex hull of the mixed-integer feasible set thanks to linear oracle calls, compared to solving the subproblems over the continuous relaxation of the same set. This novel approach computes feasible solutions while working on a single representation of the polyhedral constraints, leveraging the full extent of Mixed-Integer Programming (MIP) solvers without an outer approximation scheme.

(joint work with Deborah Hendrych, Hannah Troppens, Mathieu Besançon)

References

[1] Hendrych, D., Troppens, H., Besançon, M., and Pokutta, S. (2022). Convex integer optimization with Frank-Wolfe methods. Preprint. https://arxiv.org/abs/2208.11010

Convex integer optimization with Frank-Wolfe methods

Sebastian Pokutta

joint work with: Mathieu Besançon, Deborah Hendrych, and Hannah Troppens

Technische Universität Berlin
and
Zuse Institute Berlin

pokutta@math.tu-berlin.de
@spokutta

6th RIKEN-IMI-ISM-NUS-ZIB-MODAL-NHR Workshop on
Advances in Classical and Quantum Algorithms for Optimization
and Machine Learning

September 19th, 2022 · Tokyo, Japan

Berlin Mathematics Research Center

What is this talk about?

Introduction

A mixed-integer convex optimization method based on conditional gradients.

What is this talk about?

Introduction

A mixed-integer convex optimization method based on conditional gradients.

Why? Conditional gradients generate *sparse* iterates, leading to lower fractionality, and hence less branching.

What is this talk about?

Introduction

A mixed-integer convex optimization method based on conditional gradients.

Why? Conditional gradients generate *sparse* iterates, leading to lower fractionality, and hence less branching.

Today: A brief overview of approach and solver.

What is this talk about?

Introduction

A mixed-integer convex optimization method based on conditional gradients.

Why? Conditional gradients generate *sparse* iterates, leading to lower fractionality, and hence less branching.

Today: A brief overview of approach and solver.

Outline

- Recap: Conditional Gradients a.k.a. the Frank-Wolfe algorithm
- Mixed-Integer Conditional Gradients
- Julia Package Boscia.jl

(Hyperlinked) References are not exhaustive; check references contained therein.

Conditional Gradients a.k.a. the Frank-Wolfe algorithm

—The Basics—

The basic problem

Conditional Gradients a.k.a. the Frank-Wolfe algorithm

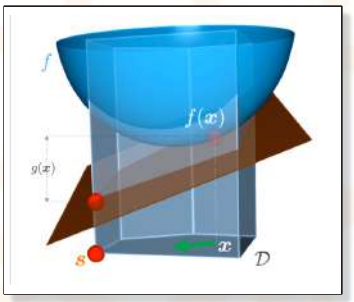
Given a smooth and convex function f and a polytope P, solve optimization problem:

The basic problem

Conditional Gradients a.k.a. the Frank-Wolfe algorithm

Given a smooth and convex function f and a polytope P, solve optimization problem:

$$\min_{x \in P} f(x) \qquad \text{(baseProblem)}$$



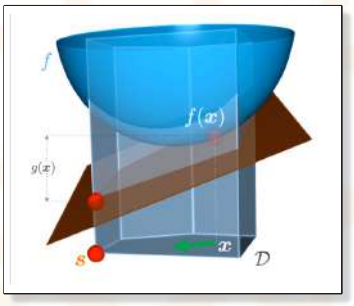
Source: [Jaggi, 2013]

The basic problem

Conditional Gradients a.k.a. the Frank-Wolfe algorithm

Given a smooth and convex function f and a polytope P, solve optimization problem:

$$\min_{x \in P} f(x) \qquad \text{(baseProblem)}$$



Source: [Jaggi, 2013]

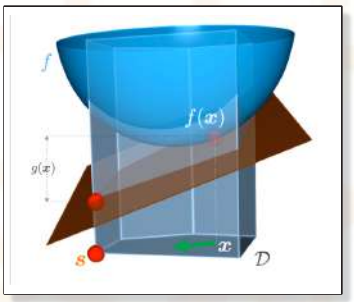
1. Very versatile model
2. Can use various types of information about both f and P
3. Works very well in (continuous) real-world applications
4. At the core of many (all?) learning algorithms (albeit mostly non-convex case)

The basic problem

Conditional Gradients a.k.a. the Frank-Wolfe algorithm

Given a smooth and convex function f and a polytope P, solve optimization problem:

$$\min_{x \in P} f(x) \qquad \text{(baseProblem)}$$



Source: [Jaggi, 2013]

Our setup.

1. Access to P. Linear Minimization Oracle (LMO): Given linear objective c return

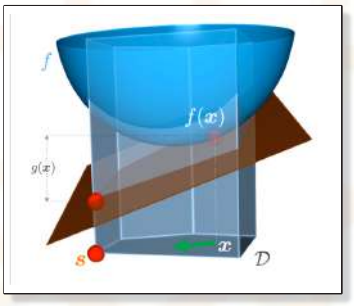
$$x \leftarrow \arg\min_{v \in P} c^T v.$$

The basic problem

Conditional Gradients a.k.a. the Frank-Wolfe algorithm

Given a smooth and convex function f and a polytope P, solve optimization problem:

$$\min_{x \in P} f(x) \qquad \text{(baseProblem)}$$



Source: [Jaggi, 2013]

Our setup.

1. Access to P. Linear Minimization Oracle (LMO): Given linear objective c return

$$x \leftarrow \arg\min_{v \in P} c^T v.$$

2. Access to f. First-Order Oracle (FO): Given x return

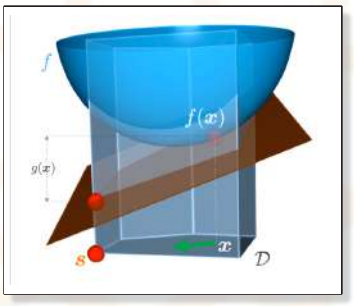
$$\nabla f(x) \quad \text{and} \quad f(x).$$

The basic problem

Conditional Gradients a.k.a. the Frank-Wolfe algorithm

Given a smooth and convex function f and a polytope P, solve optimization problem:

$$\min_{x \in P} f(x) \qquad \text{(baseProblem)}$$



Source: [Jaggi, 2013]

Our setup.

1. Access to P. Linear Minimization Oracle (LMO): Given linear objective c return

$$x \leftarrow \arg\min_{v \in P} c^T v.$$

2. Access to f. First-Order Oracle (FO): Given x return

$$\nabla f(x) \quad \text{and} \quad f(x).$$

$\Rightarrow$ *Complexity of convex optimization relative to LO/FO oracle*

Interlude: why LMOs?

Conditional Gradients a.k.a. the Frank-Wolfe algorithm

LMO model has many advantages.

1. Includes explicit formulation via constraints
2. Some problems do not posess 'small' formulations but have efficient LMOs.
   Example: Matching Polytope [Rothvoss, 2014, Braun and Pokutta, 2015a,b, Braun et al., 2015, 2017a]
3. Allows modeling of compact convex constraints as long as we have an LMO.
   Example: SDP cone
4. Often much faster than projection.
   Example: nuclear norm. Largest singular vector (Lanczos method) vs. full SVD
5. LMO is a black box for the algorithms
6. For many LMOs of interest close form solutions available.
   Example: ℓ_1-ball for LASSO regression.

For an overview see: [Combettes and Pokutta, 2021]

The basic problem

Conditional Gradients a.k.a. the Frank-Wolfe algorithm

Basic notions. Let $f : \mathbb{R}^n \to \mathbb{R}$ be a differentiable function.

The basic problem

Conditional Gradients a.k.a. the Frank-Wolfe algorithm

Basic notions. Let $f : \mathbb{R}^n \to \mathbb{R}$ be a differentiable function.

Definition (Convexity)

For all x, y it holds:

$$f(y) - f(x) \geq \langle \nabla f(x), y - x \rangle .$$

In particular, all local minima are global minima.

The basic problem

Conditional Gradients a.k.a. the Frank-Wolfe algorithm

Basic notions. Let $f : \mathbb{R}^n \to \mathbb{R}$ be a differentiable function.

Definition (Convexity)

For all x, y it holds:

$$f(y) - f(x) \geq \langle \nabla f(x), y - x \rangle .$$

In particular, all local minima are global minima.

Definition (L-Smoothness)

For all x, y it holds:

$$f(y) - f(x) \leq \langle \nabla f(x), y - x \rangle + \frac{L}{2} \|y - x\|^2.$$

The basic problem

Conditional Gradients a.k.a. the Frank-Wolfe algorithm

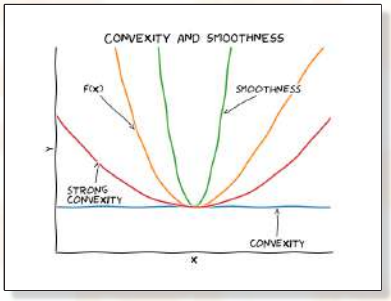
Basic notions. Let $f : \mathbb{R}^n \to \mathbb{R}$ be a differentiable function.

Definition (Convexity)

For all x, y it holds:

$$f(y) - f(x) \geq \langle \nabla f(x), y - x \rangle .$$

In particular, all local minima are global minima.



Definition (L-Smoothness)

For all x, y it holds:

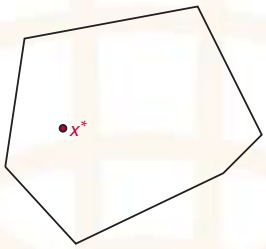
$$f(y) - f(x) \leq \langle \nabla f(x), y - x \rangle + \frac{L}{2}\|y - x\|^2.$$

The Frank-Wolfe Algorithm

Conditional Gradients a.k.a. the Frank-Wolfe algorithm

Algorithm Frank-Wolfe Algorithm (FW)

1: $x_0 \in P$
2: **for** $t = 0$ **to** $T - 1$ **do**
3: $\quad v_t \leftarrow \arg\min_{v \in P} \langle \nabla f(x_t), v \rangle$
4: $\quad x_{t+1} \leftarrow x_t + \gamma_t (v_t - x_t)$
5: **end for**



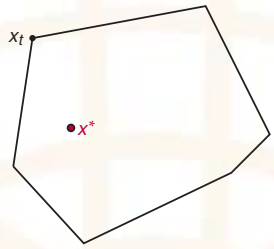
[Frank and Wolfe, 1956, Levitin and Polyak, 1966]

The Frank-Wolfe Algorithm

Conditional Gradients a.k.a. the Frank-Wolfe algorithm

Algorithm Frank-Wolfe Algorithm (FW)

$x_0 \in P$
for $t = 0$ **to** $T - 1$ **do**
  $v_t \leftarrow \arg\min_{v \in P} \langle \nabla f(x_t), v \rangle$
  $x_{t+1} \leftarrow x_t + \gamma_t (v_t - x_t)$
end for



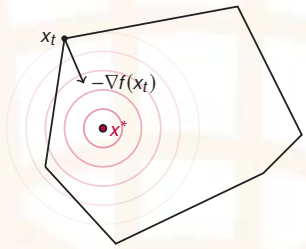
[Frank and Wolfe, 1956, Levitin and Polyak, 1966]

The Frank-Wolfe Algorithm

Conditional Gradients a.k.a. the Frank-Wolfe algorithm

Algorithm Frank-Wolfe Algorithm (FW)

$x_0 \in P$
for $t = 0$ **to** $T - 1$ **do**
  $v_t \leftarrow \arg\min_{v \in P} \langle \nabla f(x_t), v \rangle$
  $x_{t+1} \leftarrow x_t + \gamma_t (v_t - x_t)$
end for



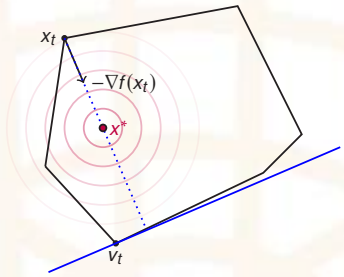
[Frank and Wolfe, 1956, Levitin and Polyak, 1966]

The Frank-Wolfe Algorithm

Conditional Gradients a.k.a. the Frank-Wolfe algorithm

Algorithm Frank-Wolfe Algorithm (FW)

1: $x_0 \in P$
2: **for** $t = 0$ **to** $T - 1$ **do**
3: $\quad v_t \leftarrow \arg\min_{v \in P} \langle \nabla f(x_t), v \rangle$
4: $\quad x_{t+1} \leftarrow x_t + \gamma_t (v_t - x_t)$
5: **end for**



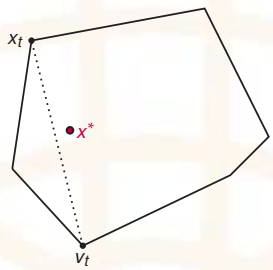
[Frank and Wolfe, 1956, Levitin and Polyak, 1966]

The Frank-Wolfe Algorithm

Conditional Gradients a.k.a. the Frank-Wolfe algorithm

Algorithm Frank-Wolfe Algorithm (FW)

1: $x_0 \in P$
2: **for** $t = 0$ **to** $T - 1$ **do**
3: $\quad v_t \leftarrow \arg\min_{v \in P} \langle \nabla f(x_t), v \rangle$
4: $\quad x_{t+1} \leftarrow x_t + \gamma_t (v_t - x_t)$
5: **end for**



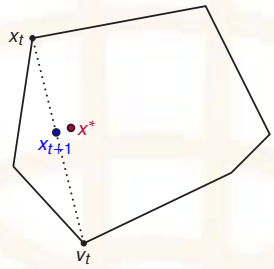
[Frank and Wolfe, 1956, Levitin and Polyak, 1966]

The Frank-Wolfe Algorithm

Conditional Gradients a.k.a. the Frank-Wolfe algorithm

Algorithm Frank-Wolfe Algorithm (FW)

1: $x_0 \in P$
2: **for** $t = 0$ **to** $T - 1$ **do**
3: $\quad v_t \leftarrow \arg\min_{v \in P} \langle \nabla f(x_t), v \rangle$
4: $\quad x_{t+1} \leftarrow x_t + \gamma_t (v_t - x_t)$
5: **end for**



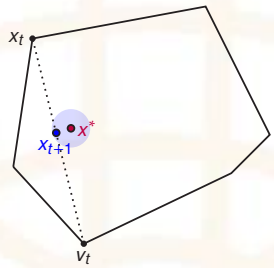
[Frank and Wolfe, 1956, Levitin and Polyak, 1966]

The Frank-Wolfe Algorithm

Conditional Gradients a.k.a. the Frank-Wolfe algorithm

Algorithm Frank-Wolfe Algorithm (FW)

1: $x_0 \in P$
2: **for** $t = 0$ **to** $T - 1$ **do**
3: $\quad v_t \leftarrow \arg\min_{v \in P} \langle \nabla f(x_t), v \rangle$
4: $\quad x_{t+1} \leftarrow x_t + \gamma_t (v_t - x_t)$
5: **end for**



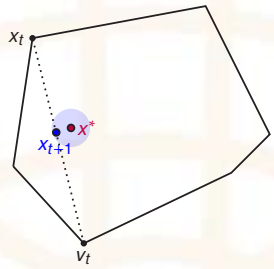
[Frank and Wolfe, 1956, Levitin and Polyak, 1966]

The Frank-Wolfe Algorithm

Conditional Gradients a.k.a. the Frank-Wolfe algorithm

Algorithm Frank-Wolfe Algorithm (FW)

1: $x_0 \in P$
2: **for** $t = 0$ **to** $T - 1$ **do**
3: $\quad v_t \leftarrow \arg\min_{v \in P} \langle \nabla f(x_t), v \rangle$
4: $\quad x_{t+1} \leftarrow x_t + \gamma_t (v_t - x_t)$
5: **end for**



[Frank and Wolfe, 1956, Levitin and Polyak, 1966]

Advantages:

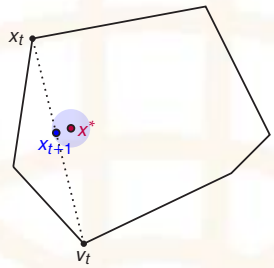
- Extremely simple and robust: no complicated data structures to maintain
- Easy to implement: requires only the two oracles
- Projection-free: feasibility convex combination and LO oracle.
- Sparsity: optimal solution is a convex combination of (usually) vertices.

The Frank-Wolfe Algorithm

Conditional Gradients a.k.a. the Frank-Wolfe algorithm

Algorithm Frank-Wolfe Algorithm (FW)

1: $x_0 \in P$
2: **for** $t = 0$ **to** $T - 1$ **do**
3: $\quad v_t \leftarrow \arg\min_{v \in P} \langle \nabla f(x_t), v \rangle$
4: $\quad x_{t+1} \leftarrow x_t + \gamma_t (v_t - x_t)$
5: **end for**



[Frank and Wolfe, 1956, Levitin and Polyak, 1966]

Advantages:

- Extremely simple and robust: no complicated data structures to maintain
- Easy to implement: requires only the two oracles
- Projection-free: feasibility convex combination and LO oracle.
- Sparsity: optimal solution is a convex combination of (usually) vertices.

Disadvantages:

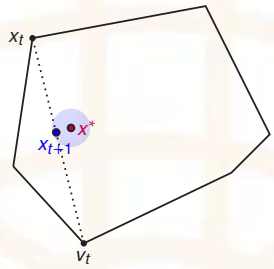
- Suboptimal convergence rate of $O(1/T)$

The Frank-Wolfe Algorithm

Conditional Gradients a.k.a. the Frank-Wolfe algorithm

Algorithm Frank-Wolfe Algorithm (FW)

1: $x_0 \in P$
2: **for** $t = 0$ **to** $T - 1$ **do**
3: $\quad v_t \leftarrow \arg\min_{v \in P} \langle \nabla f(x_t), v \rangle$
4: $\quad x_{t+1} \leftarrow x_t + \gamma_t (v_t - x_t)$
5: **end for**



[Frank and Wolfe, 1956, Levitin and Polyak, 1966]

Advantages:

- Extremely simple and robust: no complicated data structures to maintain
- Easy to implement: requires only the two oracles
- Projection-free: feasibility convex combination and LO oracle.
- Sparsity: optimal solution is a convex combination of (usually) vertices.

Disadvantages:

- Suboptimal convergence rate of $O(1/T)$

$\Longrightarrow$ *Despite (theoretically) suboptimal rate heavily used in applications due to simplicity.*

Simple Convergence Proof

Conditional Gradients a.k.a. the Frank-Wolfe algorithm

Theorem (Convergence rate of the vanilla Frank-Wolfe Algorithm)

Let f be L-smooth convex, P be polytope with diameter D. With choice $\gamma_t \doteq \frac{2}{t+3}$:

$$f(x_t) - f(x^*) \leq \frac{2LD^2}{t+3}.$$

Simple Convergence Proof

Conditional Gradients a.k.a. the Frank-Wolfe algorithm

Theorem (Convergence rate of the vanilla Frank-Wolfe Algorithm)

Let f be L-smooth convex, P be polytope with diameter D. With choice $\gamma_t \doteq \frac{2}{t+3}$:

$$f(x_t) - f(x^*) \leq \frac{2LD^2}{t+3}.$$

Proof Sketch.

By smoothness:

$$f(x_{t+1}) - f(x_t) \leq \langle \nabla f(x_t), x_{t+1} - x_t \rangle + \frac{L}{2}\|x_{t+1} - x_t\|^2 = \gamma_t \langle \nabla f(x_t), v_t - x_t \rangle + \frac{L\gamma_t^2}{2}\|v_t - x_t\|^2.$$

Simple Convergence Proof

Conditional Gradients a.k.a. the Frank-Wolfe algorithm

Theorem (Convergence rate of the vanilla Frank-Wolfe Algorithm)

Let f be L-smooth convex, P be polytope with diameter D. With choice $\gamma_t \doteq \frac{2}{t+3}$:

$$f(x_t) - f(x^*) \leq \frac{2LD^2}{t+3}.$$

Proof Sketch.

By smoothness:

$$f(x_{t+1}) - f(x_t) \leq \langle \nabla f(x_t), x_{t+1} - x_t \rangle + \frac{L}{2}\|x_{t+1} - x_t\|^2 = \gamma_t \langle \nabla f(x_t), v_t - x_t \rangle + \frac{L\gamma_t^2}{2}\|v_t - x_t\|^2.$$

LP maximality and convexity: $\langle \nabla f(x_t), v_t - x_t \rangle \leq \langle \nabla f(x_t), x^* - x_t \rangle \leq f(x^*) - f(x_t)$. Moreover, $\|v_t - x_t\| \leq D$.

Simple Convergence Proof

Conditional Gradients a.k.a. the Frank-Wolfe algorithm

Theorem (Convergence rate of the vanilla Frank-Wolfe Algorithm)

Let f be L-smooth convex, P be polytope with diameter D. With choice $\gamma_t \doteq \frac{2}{t+3}$:

$$f(x_t) - f(x^*) \leq \frac{2LD^2}{t+3}.$$

Proof Sketch.

By smoothness:

$$f(x_{t+1}) - f(x_t) \leq \langle \nabla f(x_t), x_{t+1} - x_t \rangle + \frac{L}{2}\|x_{t+1} - x_t\|^2 = \gamma_t \langle \nabla f(x_t), v_t - x_t \rangle + \frac{L\gamma_t^2}{2}\|v_t - x_t\|^2.$$

LP maximality and convexity: $\langle \nabla f(x_t), v_t - x_t \rangle \leq \langle \nabla f(x_t), x^* - x_t \rangle \leq f(x^*) - f(x_t)$. Moreover, $\|v_t - x_t\| \leq D$.

Thus:

$$f(x_{t+1}) - f(x^*) \leq (1 - \gamma_t)(f(x_t) - f(x^*)) + \gamma_t^2 \frac{LD^2}{2}.$$

Simple Convergence Proof

Conditional Gradients a.k.a. the Frank-Wolfe algorithm

Theorem (Convergence rate of the vanilla Frank-Wolfe Algorithm)

Let f be L-smooth convex, P be polytope with diameter D. With choice $\gamma_t \doteq \frac{2}{t+3}$:

$$f(x_t) - f(x^*) \leq \frac{2LD^2}{t+3}.$$

Proof Sketch.

By smoothness:

$$f(x_{t+1}) - f(x_t) \leq \langle \nabla f(x_t), x_{t+1} - x_t \rangle + \frac{L}{2}\|x_{t+1} - x_t\|^2 = \gamma_t \langle \nabla f(x_t), v_t - x_t \rangle + \frac{L\gamma_t^2}{2}\|v_t - x_t\|^2.$$

LP maximality and convexity: $\langle \nabla f(x_t), v_t - x_t \rangle \leq \langle \nabla f(x_t), x^* - x_t \rangle \leq f(x^*) - f(x_t)$. Moreover, $\|v_t - x_t\| \leq D$.

Thus:

$$f(x_{t+1}) - f(x^*) \leq (1 - \gamma_t)(f(x_t) - f(x^*)) + \gamma_t^2 \frac{LD^2}{2}.$$

By Induction (plugging in the guarantee + definition of γ_t):

$$f(x_{t+1}) - f(x^*) \leq \left(1 - \frac{2}{t+3}\right)\frac{2LD^2}{t+3} + \frac{4}{(t+3)^2} \cdot \frac{LD^2}{2} = \frac{2LD^2(t+2)}{(t+3)^2} \leq \frac{2LD^2}{t+4},$$

by $(t+2)(t+4) \leq (t+3)^2$.

Significant progress over the recent years (incomplete list)

Conditional Gradients a.k.a. the Frank-Wolfe algorithm

1. Strongly convex case [Garber and Hazan, 2013, Lacoste-Julien and Jaggi, 2015, Lan and Zhou, 2016, Garber and Meshi, 2016]
2. Non-convex case [Lacoste-Julien, 2016]
3. Online case [Hazan and Kale, 2012]
4. Stochastic variants and adaptive gradients [Hazan and Luo, 2016, Reddi et al., 2016, Combettes et al., 2020]
5. Sharp functions and sharp regions [Kerdreux et al., 2019, 2021a,b]
6. Acceleration [Diakonikolas et al., 2020, Bach, 2020, Carderera et al., 2021]
7. Specialized variants [Freund et al., 2017, Braun et al., 2017b, 2019b,a]

Conditional Gradients very competitive: simple, robust, real-world performance.

For more background etc see our survey! [Braun et al., 2022]

Mixed-Integer Conditional Gradients

—The Framework—

Problem setting

Mixed-Integer Conditional Gradients

Basically. Smooth convex objective over MIPs.

Problem setting

Mixed-Integer Conditional Gradients

Basically. Smooth convex objective over MIPs.

Slightly more general:

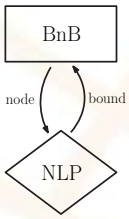
$$\begin{aligned} \min_{x,y} \; & f(x,y) \\ \text{s.t. } & x \in \mathcal{X} \\ & x_j \in \mathbb{Z} \quad \forall j \in J \\ & y \in \mathcal{Y} \end{aligned}$$

with LMO over $(\mathcal{X} \cap \text{bounds}) \times \mathcal{Y}$.

Three main algorithmic frameworks for MINLPs

Mixed-Integer Conditional Gradients

Diamond blocks represent nodal relaxations in the given framework.

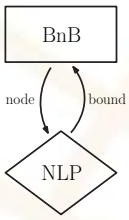


Standard BnB framework on top of NLP relaxations.

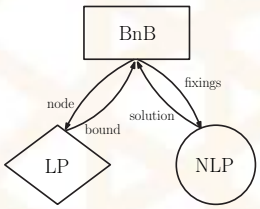
Three main algorithmic frameworks for MINLPs

Mixed-Integer Conditional Gradients

Diamond blocks represent nodal relaxations in the given framework.



Standard BnB framework on top of NLP relaxations.

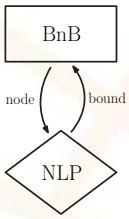


LP-based MINLP frameworks and outer approximations

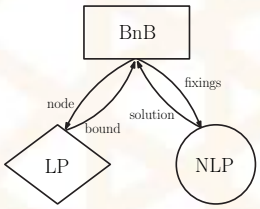
Three main algorithmic frameworks for MINLPs

Mixed-Integer Conditional Gradients

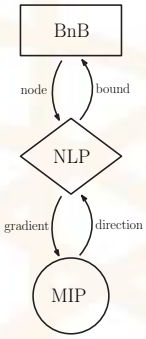
Diamond blocks represent nodal relaxations in the given framework.



Standard BnB framework on top of NLP relaxations.



LP-based MINLP frameworks and outer approximations

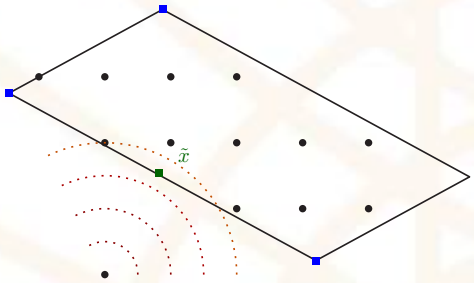


Our approach. Linearized models solved as MIPs within the Frank-Wolfe algorithm on top of which we branch

Tree of trees or forest ⟶ Boscia (Corsican) = Forest.

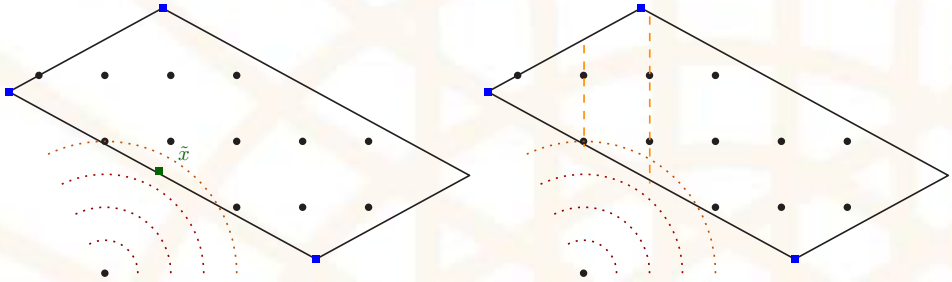
Branching: continuous relaxation (usual approach)

Mixed-Integer Conditional Gradients



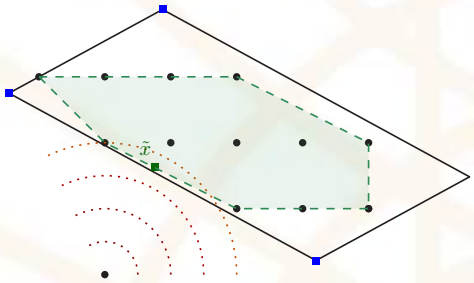
Branching: continuous relaxation (usual approach)

Mixed-Integer Conditional Gradients



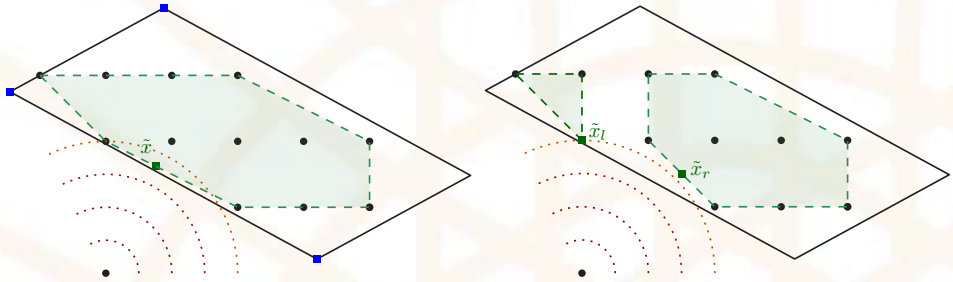
Branching: mixed-integer hull (our approach)

Mixed-Integer Conditional Gradients



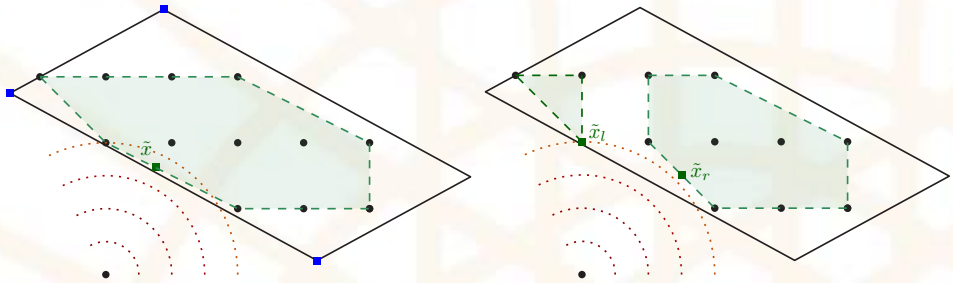
Branching: mixed-integer hull (our approach)

Mixed-Integer Conditional Gradients



Branching: mixed-integer hull (our approach)

Mixed-Integer Conditional Gradients



Open question.

Can we define adaptive criteria to choose relaxation?

(E.g., geometry of the feasible set, conditioning of the function)

Reducing number of MIP oracle calls

Mixed-Integer Conditional Gradients

We use Blended Pairwise Conditional Gradients (BPCG) [Tsuji et al., 2022]

Reducing number of MIP oracle calls

Mixed-Integer Conditional Gradients

We use Blended Pairwise Conditional Gradients (BPCG) [Tsuji et al., 2022]

- Lazification. aggressively reuse old solutions

Reducing number of MIP oracle calls

Mixed-Integer Conditional Gradients

We use Blended Pairwise Conditional Gradients (BPCG) [Tsuji et al., 2022]

- Lazification. aggressively reuse old solutions
- Blending. perform local steps for sparsity $\longrightarrow$ low fractionality

Reducing number of MIP oracle calls

Mixed-Integer Conditional Gradients

We use Blended Pairwise Conditional Gradients (BPCG) [Tsuji et al., 2022]

- Lazification. aggressively reuse old solutions
- Blending. perform local steps for sparsity $\longrightarrow$ low fractionality
- Active set. branching means simply splitting convex combination

Reducing number of MIP oracle calls

Mixed-Integer Conditional Gradients

We use Blended Pairwise Conditional Gradients (BPCG) [Tsuji et al., 2022]

- Lazification. aggressively reuse old solutions
- Blending. perform local steps for sparsity $\longrightarrow$ low fractionality
- Active set. branching means simply splitting convex combination
- Discarded set. reuse solutions from previous nodes

Reducing number of MIP oracle calls

Mixed-Integer Conditional Gradients

We use Blended Pairwise Conditional Gradients (BPCG) [Tsuji et al., 2022]

- Lazification. aggressively reuse old solutions
- Blending. perform local steps for sparsity $\longrightarrow$ low fractionality
- Active set. branching means simply splitting convex combination
- Discarded set. reuse solutions from previous nodes
- Incomplete resolution and warmstarts. Less work per node

Does it help?

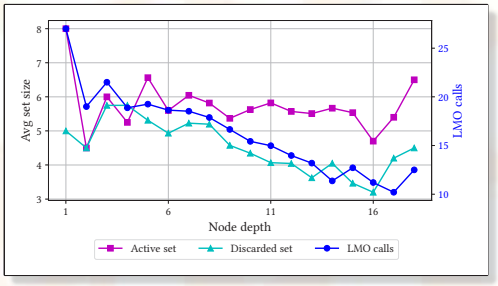
Reducing number of MIP oracle calls

Mixed-Integer Conditional Gradients

We use Blended Pairwise Conditional Gradients (BPCG) [Tsuji et al., 2022]

- Lazification. aggressively reuse old solutions
- Blending. perform local steps for sparsity ⟶ low fractionality
- Active set. branching means simply splitting convex combination
- Discarded set. reuse solutions from previous nodes
- Incomplete resolution and warmstarts. Less work per node

Does it help?



On average something like 7 to 10 sub-MIPs per node.

Reducing cost for each MIP

Mixed-Integer Conditional Gradients

Subproblems are MIPs. Leverage MIP advances:

- Cutting-planes
- Domain propagation
- Presolving
- Primal heuristics
- etc.

Reducing cost for each MIP

Mixed-Integer Conditional Gradients

Subproblems are MIPs. Leverage MIP advances:

- Cutting-planes
- Domain propagation
- Presolving
- Primal heuristics
- etc.

Moreover, we can reuse information across solves heavily:

- MIP solver called with different objectives within node
- Identical polyhedron with updated bounds solved across nodes
- All found primal solutions are valid for main problem

Reducing cost for each MIP

Mixed-Integer Conditional Gradients

Subproblems are MIPs. Leverage MIP advances:

- Cutting-planes
- Domain propagation
- Presolving
- Primal heuristics
- etc.

Moreover, we can reuse information across solves heavily:

- MIP solver called with different objectives within node
- Identical polyhedron with updated bounds solved across nodes
- All found primal solutions are valid for main problem

Question of MIP reoptimization: [Gamrath et al., 2015]

Which information should be (conditionally) transferred across instances?

Boscia.jl

—The Code—

The package

Boscia.jl

- Julia package
- Based on `Bonobo.jl` (BnB package) and `FrankWolfe.jl` (our FW package)
- Via `MOI` can use basically any MIP solver; some features specific to `SCIP`
- Includes other features such as hybrid branching
- Available under MIT license

Example: Code

Boscia.jl

```julia
using Boscia
using FrankWolfe
using Random
using SCIP
using LinearAlgebra
import MathOptInterface
const MOI = MathOptInterface

n = 6

const diffw = 0.5 * ones(n)
o = SCIP.Optimizer()

MOI.set(o, MOI.Silent(), true)

x = MOI.add_variables(o, n)

for xi in x
MOI.add_constraint(o, xi, MOI.GreaterThan(0.0))
MOI.add_constraint(o, xi, MOI.LessThan(1.0))
MOI.add_constraint(o, xi, MOI.ZeroOne())
end

lmo = FrankWolfe.MathOptLMO(o)

function f(x)
return sum(0.5*(x.-diffw).^2)
end

function grad!(storage, x)
@. storage = x-diffw
end

x, _, result = Boscia.solve(f, grad!, lmo, verbose = true)
```

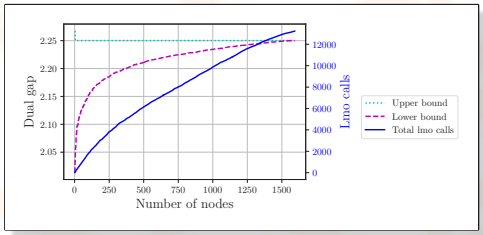

Example: planted solution in high-dimensional space

Boscia.jl

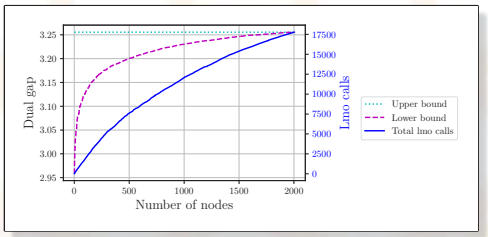
```
julia> include("examples/low_dim_in_high_dim.jl")

Boscia Algorithm.

Parameter settings.
    Tree traversal strategy: Move best bound
    Branching strategy: Most infeasible
    Absolute dual gap tolerance: 1.000000e-06
    Relative dual gap tolerance: 1.000000e-02
    Frank-Wolfe subproblem tolerance: 1.000000e-05
    Total number of varibales: 12
    Number of integer variables: 0
    Number of binary variables: 12

Iteration    Open        Bound     Incumbent      Gap (abs)      Gap (rel)      Time (s)    Nodes/sec       FW (ms)     LMO (ms)  LMO (calls c)   FW (Its)  #ActiveSet  Discarded
-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
*         1         2  -2.297891e+03  -1.977958e+03  3.199328e+02  1.617490e-01  9.150000e-01  3.278689e+00         758           3             11      10001           1          0
*         2         3  -2.297891e+03  -2.238338e+03  5.955322e+01  2.660600e-02  1.442000e+00  3.467406e+00         526          11             46      10001          17          0
*         4         5  -2.292074e+03  -2.239976e+03  5.209869e+01  2.325860e-02  2.500000e+00  3.600000e+00         531           6            101      10001          11          0
*         5         6  -2.292074e+03  -2.242301e+03  4.977302e+01  2.219729e-02  3.024000e+00  3.637566e+00         523          10            133      10001          12          0
*         6         7  -2.292074e+03  -2.242301e+03  4.977302e+01  2.219729e-02  3.544000e+00  3.668172e+00         519          11            164      10001           4          0
*        16        17  -2.282246e+03  -2.243023e+03  3.922226e+01  1.748634e-02  8.726000e+00  3.781802e+00         524           8            439      10001           6          2
*        21        22  -2.280549e+03  -2.243325e+03  3.722381e+01  1.659314e-02  1.131100e+01  3.801609e+00         514           5            564      10001           7          1
*        29        30  -2.279719e+03  -2.244814e+03  3.490433e+01  1.554887e-02  1.544900e+01  3.819017e+00         522           8            761      10001           3          1
*        66        67  -2.271953e+03  -2.245231e+03  2.672273e+01  1.190200e-02  3.451800e+01  3.853062e+00         517           8           1616      10001           1          2
        100       101  -2.268603e+03  -2.245231e+03  2.337210e+01  1.040967e-02  5.204200e+01  3.862265e+00         516           6           2357      10001           2          1
        119       120  -2.267387e+03  -2.245231e+03  2.215595e+01  9.868008e-03  6.184900e+01  3.864250e+00         530           6           2778      10001           4          0
-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Solution Statistics.
    Solution Status: Optimal (tolerance reached)
    Primal Objective: -2245.23067557406
    Dual Bound: -2267.3866295644566
    Dual Gap (relative): 0.009968007876175864

Search Statistics.
    Total number of nodes processed: 239
    Total number of lmo calls: 2782
    Total time (s): 61.853
    LMO calls / sec: 44.9776082000873
    Nodes / sec: 3.8640001293389163
    LMO calls / node: 11.640167364016737
```

Example: computational results: Sparse Regression

Boscia.jl



High-dimensional sparse regression problem with ℓ_0-constraints



High-dimensional sparse regression problem over mixed-integer feasible region.

Thank you!

Preprint: `arxiv.org/abs/2208.11010`,
Convex integer optimization with FW methods [Hendrych et al., 2022]
Package available at `github.com/ZIB-IOL/Boscia.jl`

In a nutshell. Minimize smooth convex objective over any MIP. Applications in engineering, sparse prediction models, statistics, and relaxation of combinatorial problems.

References I

F. Bach. On the effectiveness of Richardson extrapolation in machine learning. *arXiv preprint 2002.02835v3*, July 2020.

G. Braun and S. Pokutta. The matching polytope does not admit fully-polynomial size relaxation schemes. *Proceeedings of SODA*, 2015a.

G. Braun and S. Pokutta. The matching polytope does not admit fully-polynomial size relaxation schemes. *IEEE Transactions on Information Theory*, 61(10):1–11, 2015b.

G. Braun, S. Pokutta, and D. Zink. Inapproximability of combinatorial problems via small LPs and SDPs. *Proceeedings of STOC*, 2015.

G. Braun, R. Jain, T. Lee, and S. Pokutta. Information-theoretic approximations of the nonnegative rank. *Computational Complexity*, 26(1):147–197, 2017a.

G. Braun, S. Pokutta, and D. Zink. Lazifying Conditional Gradient Algorithms. *Proceedings of the International Conference on Machine Learning (ICML)*, 2017b.

G. Braun, S. Pokutta, D. Tu, and S. Wright. Blended Conditional Gradients: the unconditioning of conditional gradients. *Proceedings of ICML*, 2019a.

G. Braun, S. Pokutta, and D. Zink. Lazifying Conditional Gradient Algorithms. *Journal of Machine Learning Research (JMLR)*, 20(71):1–42, 2019b.

G. Braun, A. Carderera, C. W. Combettes, H. Hassani, A. Karbasi, A. Mokthari, and S. Pokutta. Conditional gradient methods. *preprint*, 8 2022.

A. Carderera, J. Diakonikolas, C. Y. Lin, and S. Pokutta. Parameter-free Locally Accelerated Conditional Gradients. *to appear in Proceedings of ICML*, 2 2021.

C. W. Combettes and S. Pokutta. Complexity of Linear Minimization and Projection on Some Sets. *to appear in Operations Research Letters*, 1 2021.

C. W. Combettes, C. Spiegel, and S. Pokutta. Projection-Free Adaptive Gradients for Large-Scale Optimization. *preprint*, 10 2020.

J. Diakonikolas, A. Carderera, and S. Pokutta. Locally Accelerated Conditional Gradients. *Proceedings of AISTATS*, 2020.

M. Frank and P. Wolfe. An algorithm for quadratic programming. *Naval Research Logistics Quarterly*, 3(1-2):95–110, 1956.

R. M. Freund, P. Grigas, and R. Mazumder. An extended Frank-Wolfe method with "in-face" directions, and its application to low-rank matrix completion. *SIAM Journal on Optimization*, 27(1):319–346, 2017.

G. Gamrath, B. Hiller, and J. Witzig. Reoptimization techniques for mip solvers. In *International Symposium on Experimental Algorithms*, pages 181–192. Springer, 2015.

D. Garber and E. Hazan. Playing non-linear games with linear oracles. In *2013 IEEE 54th Annual Symposium on Foundations of Computer Science*, pages 420–428. IEEE, 2013.

D. Garber and O. Meshi. Linear-memory and decomposition-invariant linearly convergent conditional gradient algorithm for structured polytopes. In D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems 29*, pages 1001–1009. Curran Associates, Inc., 2016. URL `http://papers.nips.cc/paper/6115-linear-memory-and-decomposition-invariant-linearly-convergent-conditional-gradient-algorithm-for-structured-pdf`.

E. Hazan and S. Kale. Projection-free online learning. In *Proceedings of the 29th International Conference on Machine Learning*, 2012.

E. Hazan and H. Luo. Variance-reduced and projection-free stochastic optimization. In *International Conference on Machine Learning*, pages 1263–1271, 2016.

D. Hendrych, H. Troppens, M. Besançon, and S. Pokutta. Convex integer optimization with frank-wolfe methods. *preprint*, 8 2022.

References II

M. Jaggi. Revisiting Frank-Wolfe: projection-free sparse convex optimization. In *Proceedings of the 30th International Conference on Machine Learning*, pages 427–435, 2013.

T. Kerdreux, A. d'Aspremont, and S. Pokutta. Restarting Frank-Wolfe. *Proceedings of AISTATS*, 2019.

T. Kerdreux, A. d'Aspremont, and S. Pokutta. Projection-Free Optimization on Uniformly Convex Sets. *to appear in Proceedings of AISTATS*, 1 2021a.

T. Kerdreux, A. d'Aspremont, and S. Pokutta. Local and Global Uniform Convexity Conditions. *preprint*, 2 2021b.

S. Lacoste-Julien. Convergence rate of Frank-Wolfe for non-convex objectives. *arXiv preprint arXiv:1607.00345*, 2016.

S. Lacoste-Julien and M. Jaggi. On the global linear convergence of Frank-Wolfe optimization variants. In C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, editors, *Advances in Neural Information Processing Systems 28*, pages 496–504. Curran Associates, Inc., 2015. URL `http://papers.nips.cc/paper/5925-on-the-global-linear-convergence-of-frank-wolfe-optimization-variants.pdf`.

G. Lan and Y. Zhou. Conditional gradient sliding for convex optimization. *SIAM Journal on Optimization*, 26(2):1379–1409, 2016.

E. S. Levitin and B. T. Polyak. Constrained minimization methods. *USSR Computational Mathematics and Mathematical Physics*, 6(5):1–50, 1966.

S. J. Reddi, S. Sra, B. Póczos, and A. Smola. Stochastic Frank-Wolfe methods for nonconvex optimization. In *2016 54th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, pages 1244–1251. IEEE, 2016.

T. Rothvoss. The matching polytope has exponential extension complexity. In *Symposium on Theory of Computing*, pages 263–272, 2014.

K. Tsuji, K. Tanaka, and S. Pokutta. Pairwise Conditional Gradients without Swap Steps and Sparser Kernel Herding. *To Appear in Proceedings of ICML*, 5 2022.

Bregman Proximal DC Algorithms and Their Application to Blind Deconvolution with Nonsmooth Regularization

Shota Takahashi
The Graduate University for Advanced Studies, Japan
takahashi.shota@ism.ac.jp

Blind deconvolution is a technique to recover an original signal without knowing a convolving filter. It is naturally formulated as a minimization of a quartic objective function under some assumption. Because its differentiable part does not have a Lipschitz continuous gradient, existing first-order methods are not theoretically supported. In this presentation, we reformulate the objective function as a difference of convex (DC) functions and add nonsmooth regularization. Then, we apply the Bregman proximal DC algorithm (BPDCA) and the BPDCA with extrapolation (BPDCAe), whose convergences are theoretically guaranteed under the L-smooth adaptable (L-smad) property. BPDCAe outperformed other existing algorithms in image deburring applications. This talk is based on [1] and [2].

References

[1] Shota Takahashi, Mituhiro Fukuda, and Mirai Tanaka, "New Bregman proximal type algorithms for solving DC optimization problems", Computational Optimization and Applications, online, September 2022, https://doi.org/10.1007/s10589-022-00411-w
[2] Shota Takahashi, Mituhiro Fukuda, and Shiro Ikeda, "Blind deconvolution with non-smooth regularization via Bregman proximal DCAs", Signal Processing, Volume 202, 108734, January 2023, https://doi.org/10.1016/j.sigpro.2022.108734

Bregman Proximal DC Algorithms and Their Application to Blind Deconvolution with Nonsmooth Regularization

The 6th RIKEN-IMI-ISM-NUS-ZIB-MODAL-NHR Workshop on Advances in Classical and Quantum Algorithms for Optimization and Machine Learning

Shota Takahashi[1]

September 19th, 2022

[1]Department of Statistical Science, The Graduate University for Advanced Studies (SOKENDAI)

Takahashi, S., Fukuda, M., and Tanaka, M. (2022).
New Bregman proximal type algorithms for solving DC optimization problems.
Computational Optimization and Applications, online.

Takahashi, S., Tanaka, M., and Ikeda, S. (2023).
Blind deconvolution with non-smooth regularization via Bregman proximal DCAs.
Signal Processing, 202:108734.

Difference of convex functions (DC) optimization

Definition (DC optimization problem with a regularization term)

Given convex functions $F_1, F_2, G : \mathbb{R}^d \to (-\infty, +\infty]$, consider the following DC optimizaion problem:

$$\min_{\boldsymbol{z} \in \operatorname{cl} C} \quad F_1(\boldsymbol{z}) - F_2(\boldsymbol{z}) + G(\boldsymbol{z}), \tag{1}$$

where F_1 is $\mathcal{C}^1$, and $C \subset \mathbb{R}^d$ is a nonempty open convex set.

Exisiting algorithms ($\boldsymbol{\xi}^k \in \partial F_2(\boldsymbol{z}^k)$ is a subgradient of F_2 at $\boldsymbol{z}^k$):

- **DC algorithm (DCA)**: $\boldsymbol{z}^{k+1} \in \operatorname{argmin}_{\boldsymbol{z} \in \operatorname{cl} C}\{F_1(\boldsymbol{z}) - \langle \boldsymbol{\xi}^k, \boldsymbol{z}\rangle + G(\boldsymbol{z})\}$.
  - Its subproblem is computationally demanding unless F_1 and G have simple structures.
- **Proximal DCA**: $\boldsymbol{z}^{k+1} = \operatorname{argmin}_{\boldsymbol{z} \in \operatorname{cl} C}\{\langle \nabla F_1(\boldsymbol{z}^k) - \boldsymbol{\xi}^k, \boldsymbol{z}\rangle + G(\boldsymbol{z}) + \frac{1}{2\lambda}\|\boldsymbol{z} - \boldsymbol{z}^k\|_2^2\}$, $\lambda \in (0, \frac{1}{L})$.
  - To guarantee its global convergence, it requires F_1 is L-smooth (∇F_1 is Lipschitz continuous).
  - When F_1 is not L-smooth (L cannot be defined), it is **not practical**.
- **Bregman proximal gradient (BPG)**: $\boldsymbol{z}^{k+1} = \operatorname{argmin}_{\boldsymbol{z} \in \operatorname{cl} C}\{\langle \nabla F(\boldsymbol{z}^k), \boldsymbol{z}\rangle + G(\boldsymbol{z}) + \frac{1}{\lambda} D_H(\boldsymbol{z}, \boldsymbol{z}^k)\}$.
  - The **Bregman distance** $D_H(\boldsymbol{z}, \boldsymbol{w}) := H(\boldsymbol{z}) - H(\boldsymbol{w}) - \langle \nabla H(\boldsymbol{w}), \boldsymbol{z} - \boldsymbol{w}\rangle$, where H is $\mathcal{C}^1$ and convex.
  - **Does not require L-smoothness** of $F = F_1 - F_2$ (when F_2 is also $\mathcal{C}^1$).

Today's contents

Definition (DC optimization problem)

Given convex functions $F_1, F_2, G : \mathbb{R}^d \to (-\infty, +\infty]$, consider the following DC optimizaion problem:

$$\min_{\boldsymbol{z} \in \mathrm{cl}\, C} \quad F_1(\boldsymbol{z}) - F_2(\boldsymbol{z}) + G(\boldsymbol{z}), \tag{1}$$

where F_1 is $\mathcal{C}^1$, and $C \subset \mathbb{R}^d$ is a nonempty open convex set.

Overview

- Introduce proximal DCA using the Bregman distance.
  - The **Bregman distance** $D_H(\boldsymbol{z}, \boldsymbol{w}) := H(\boldsymbol{z}) - H(\boldsymbol{w}) - \langle \nabla H(\boldsymbol{w}), \boldsymbol{z} - \boldsymbol{w} \rangle$, where H is $\mathcal{C}^1$ and convex.
  - The proposed methods converge to a stationary point of (1) under the L-smooth adaptable property instead of L-smoothness.
- Application to **blind deconvolution**.
  - For BPG, finding an appropriate H for F is difficult. **Using DC decoposition, it is easier.**

Proposed method: Bregman proximal DC algorithm

Bregman proximal DC algorithm (BPDCA) [Takahashi et al., 2022]

Input: $\boldsymbol{z}^0 \in C$, $\lambda > 0$, and a convex and $\mathcal{C}^1$ function H.

for $k = 0, 1, 2, \ldots, :$

Compute $\boldsymbol{\xi}^k \in \partial F_2(\boldsymbol{z}^k)$ and

$$\boldsymbol{z}^{k+1} = \operatorname*{argmin}_{\boldsymbol{z} \in \mathrm{cl}\, C} \left\{ \langle \nabla F_1(\boldsymbol{z}^k) - \boldsymbol{\xi}^k, \boldsymbol{z} \rangle + G(\boldsymbol{z}) + \frac{1}{\lambda} D_H(\boldsymbol{z}, \boldsymbol{z}^k) \right\}. \tag{2}$$

BPDCA is a method combined with **BPG** [Bolte et al., 2018] and **proximal DCA** [Wen et al., 2017].

- Minimizes a first-order approximation of the objective function,

$$F_1(\boldsymbol{z}) - F_2(\boldsymbol{z}) + G(\boldsymbol{z}) \simeq \langle \nabla F_1(\boldsymbol{z}^k) - \boldsymbol{\xi}^k, \boldsymbol{z} - \boldsymbol{z}^k \rangle + \underbrace{F_1(\boldsymbol{z}^k) - F_2(\boldsymbol{z}^k)}_{\text{const.}} + G(\boldsymbol{z}), \tag{3}$$

  with the Bregman proximality $\frac{1}{\lambda} D_H(\boldsymbol{z}, \boldsymbol{z}^k) = \frac{1}{\lambda} \left(H(\boldsymbol{z}) - H(\boldsymbol{z}^k) - \langle \nabla H(\boldsymbol{z}^k), \boldsymbol{z} - \boldsymbol{z}^k \rangle \right)$ as (2).
- Does not require the differentiability of G.
- When G has a sufficiently simple structure, such as $G = \| \cdot \|_1$, (2) is solved in a closed form.

Proposed method: BPDCA with extraporation

The accelerated version of BPDCA.

BPDCA with extrapolation (BPDCAe) [Takahashi et al., 2022]

Input: $\boldsymbol{z}^{-1} = \boldsymbol{z}^0 \in C$, $\lambda > 0$, $t_{-1} = t_0 = 1$, and a convex and $\mathcal{C}^1$ function H.

for $k = 0, 1, 2, \ldots,$:

Set $\boldsymbol{w}^k = \boldsymbol{z}^k + \frac{t_{k-1}-1}{t_k}(\boldsymbol{z}^k - \boldsymbol{z}^{k-1})$ and $t_{k+1} = \frac{1+\sqrt{1+4t_k^2}}{2}$.

Compute $\boldsymbol{\xi}^k \in \partial F_2(\boldsymbol{z}^k)$ and

$$\boldsymbol{z}^{k+1} = \operatorname*{argmin}_{\boldsymbol{z} \in \operatorname{cl} C} \left\{ \langle \nabla F_1(\boldsymbol{w}^k) - \boldsymbol{\xi}^k, \boldsymbol{z} \rangle + G(\boldsymbol{z}) + \frac{1}{\lambda} D_H(\boldsymbol{z}, \boldsymbol{w}^k) \right\}. \tag{4}$$

Reset $t_{k-1} = t_k = 1$ and $\boldsymbol{w}^k = \boldsymbol{z}^k$ if either of the following conditions holds:

- $k \equiv 0 \pmod N$, $N \in \mathbb{N}$ is given.
- $D_H(\boldsymbol{z}^k, \boldsymbol{w}^k) > \rho D_H(\boldsymbol{z}^{k-1}, \boldsymbol{z}^k)$, where $\rho \in [0, 1)$ is given.
- $\boldsymbol{w}^k \notin C$.

Convergence analysis: Decreasing property

Definition (L-smooth adaptable (L-smad) [Bolte et al., 2018])

Let $F, H : \mathbb{R}^d \to (-\infty, +\infty]$ be $\mathcal{C}^1$ and H be convex. The pair (F, H) is called L-smooth adaptable (L-smad) if there exists $L > 0$ such that **$LH - F$ and $LH + F$ are convex.**

The L-smad property is a generalization of L-smoothness.
When $H = \frac{1}{2}\|\cdot\|_2^2$, the L-smad property corresponds to L-smoothness.

If the pair (F_1, H) is L-smad, the sequence of the objective function value $\Psi := F_1 - F_2 + G$ generated by BPDCA is decreasing.

Lemma (Decreasing property [Takahashi et al., 2022])

Let $\{\boldsymbol{z}^k\}_k$ be a sequence generated by BPDCA. Then, it holds that

$$\lambda \Psi(\boldsymbol{z}^{k+1}) \leq \lambda \Psi(\boldsymbol{z}^k) - (1 - \lambda L) D_H(\boldsymbol{z}^{k+1}, \boldsymbol{z}^k). \tag{5}$$

In particular, the decreasing property in the objective function value Ψ is ensured with $0 < \lambda L < 1$.

Convergence analysis: Global convergence

When the objective function is a Kurdyka-Łojasiewicz (KL) function, the following theorems holds.

Theorem (Global convergence [Takahashi et al., 2022])

Let $\{\boldsymbol{z}^k\}_k$ be a sequence generated by BPDCA. Assume that the objective function is a KL function. Then, $\{\boldsymbol{z}^k\}_k$ **converges to a stationary point** $\tilde{\boldsymbol{z}}$.

Theorem (Rate of convergence [Takahashi et al., 2022])

Let $\{\boldsymbol{z}^k\}_k$ be a sequence generated by BPDCA and assume that $\{\boldsymbol{z}^k\}_k$ converges to a stationary point $\tilde{\boldsymbol{z}}$. Assume that the objective function is a KL function with $\phi(s) = cs^{1-\theta}$ for some $\theta \in [0, 1)$ and $c > 0$. Then, the following statements hold:

- If $\theta = 0$, there exists $k_0 > 0$ such that $\boldsymbol{z}^k$ is constant for $k > k_0$ (**finite**);
- If $\theta \in (0, \frac{1}{2}]$, there exist $c_1, k_1 > 0$, and $\eta \in (0, 1)$ such that $\|\boldsymbol{z}^k - \tilde{\boldsymbol{z}}\|_2 < c_1\eta^k$ for $k > k_1$ (**linear**);
- If $\theta \in (\frac{1}{2}, 1)$, there exist $c_2 > 0$ and $k_2 > 0$ such that $\|\boldsymbol{z}^k - \tilde{\boldsymbol{z}}\|_2 < c_2 k^{-\frac{1-\theta}{2\theta-1}}$ for $k > k_2$ (**sublinear**).

For BPDCAe, similar convergence results hold.

Blind deconvolution

Definition (Blind deconvolution)

Consider the convolution of a filter $\boldsymbol{f} \in \mathbb{R}^m$ and a signal $\boldsymbol{g} \in \mathbb{R}^m$, given by

$$\tilde{\boldsymbol{y}} = \boldsymbol{f} * \boldsymbol{g}, \tag{6}$$

where $*$ denotes the convolution. **Our goal is to recover $\boldsymbol{g}$ from $\tilde{\boldsymbol{y}}$ without knowing $\boldsymbol{f}$.**

- Non-blind deconvolution: Recover $\boldsymbol{g}$ with known $\boldsymbol{f}$.
- Blind deconvolution: Recover $\boldsymbol{g}$ and $\boldsymbol{f}$ simultaneously. It is a charenging problem.

Application: Astronomy, communication engineering, and image processing.

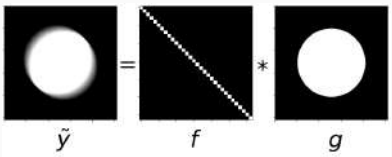


Figure 1: Recover the original image $\boldsymbol{g}$ from the blurred image $\tilde{\boldsymbol{y}}$ by removing the blurring kernel $\boldsymbol{f}$.

Optimization problem for blind deconvolution

For the linear operators $\tilde{\boldsymbol{B}}$ and $\tilde{\boldsymbol{A}}$, assume that there exist the true $(\boldsymbol{h}^\circ, \boldsymbol{x}^\circ) \in \mathbb{R}^{d_1} \times \mathbb{R}^{d_2}$ such that

$$\boldsymbol{f} = \tilde{\boldsymbol{B}}\boldsymbol{h}^\circ, \quad \boldsymbol{g} = \tilde{\boldsymbol{A}}\boldsymbol{x}^\circ. \tag{7}$$

Applying the discrete Fourier transform $\sqrt{m}\boldsymbol{F} \in \mathbb{C}^{m\times m}$ to both sides of $\tilde{\boldsymbol{y}} = \boldsymbol{f} * \boldsymbol{g}$,

$$\boldsymbol{y} = \boldsymbol{B}\boldsymbol{h}^\circ \odot \overline{\boldsymbol{A}\boldsymbol{x}^\circ}, \tag{8}$$

where $\boldsymbol{y} := \frac{1}{\sqrt{m}}\boldsymbol{F}\tilde{\boldsymbol{y}}$, $\boldsymbol{B} := \boldsymbol{F}\tilde{\boldsymbol{B}}$, and $\overline{\boldsymbol{A}} := \boldsymbol{F}\tilde{\boldsymbol{A}}$, and $\odot$ denotes the Hadamard (elementwise) product, and $\bar{\cdot}$ denotes the complex conjugate.

Definition (Optimization problem for blind deconvolution)

$$\min_{(\boldsymbol{h},\boldsymbol{x})\in \mathrm{cl}\, C} \quad \underbrace{\frac{1}{2}\|\boldsymbol{B}\boldsymbol{h} \odot \overline{\boldsymbol{A}\boldsymbol{x}} - \boldsymbol{y}\|_2^2}_{=:F(\boldsymbol{h},\boldsymbol{x})} + G(\boldsymbol{h},\boldsymbol{x}). \tag{9}$$

- F is nonconvex and **not** L**-smooth**.
- $G : \mathbb{R}^{d_1} \times \mathbb{R}^{d_2} \to (-\infty, +\infty]$ is convex and <u>non-smooth</u> (not differentiable) as a sparse regularizer.
  - Use sparse regularization (for example, ℓ_1 norm) when $\boldsymbol{h}$ or $\boldsymbol{x}$ have sparse structures.
  - [Li et al., 2019] used a smooth G using ℓ_2 norm.

DC decomposition for blind deconvolution

Because F has a bilinear term $2\,\mathrm{Re}\langle \boldsymbol{B}\boldsymbol{h} \odot \overline{\boldsymbol{A}\boldsymbol{x}}, \boldsymbol{y}\rangle$, it is difficult to find H satisfying the L-smad property.
$\to$ BPG [Bolte et al., 2018] is not applicable.

DC decomposition for blind deconvolution

Our optimization problem for blind deconvolution:

$$\min_{(\boldsymbol{h},\boldsymbol{x})\in \mathrm{cl}\, C} \quad \underbrace{\frac{1}{2}\|\boldsymbol{B}\boldsymbol{h} \odot \overline{\boldsymbol{A}\boldsymbol{x}} - \boldsymbol{y}\|_2^2}_{=:F(\boldsymbol{h},\boldsymbol{x})} + G(\boldsymbol{h},\boldsymbol{x}), \tag{9}$$

F has a DC decomposition $F = F_1 - F_2$ for two convex functions F_1 and F_2:

$$F_1(\boldsymbol{h},\boldsymbol{x}) = \frac{1}{4}\|\boldsymbol{B}\boldsymbol{h}\|_4^4 + \frac{1}{4}\|\boldsymbol{A}\boldsymbol{x}\|_4^4 + \frac{1}{2}(\|\boldsymbol{B}\boldsymbol{h} \odot \boldsymbol{A}\boldsymbol{x}\|_2^2 + \|\boldsymbol{y} \odot \boldsymbol{B}\boldsymbol{h}\|_2^2 + \|\boldsymbol{A}\boldsymbol{x}\|_2^2 + \|\boldsymbol{y}\|_2^2),$$

$$F_2(\boldsymbol{h},\boldsymbol{x}) = \frac{1}{4}\|\boldsymbol{B}\boldsymbol{h}\|_4^4 + \frac{1}{4}\|\boldsymbol{A}\boldsymbol{x}\|_4^4 + \frac{1}{2}\|\bar{\boldsymbol{y}} \odot \boldsymbol{B}\boldsymbol{h} + \overline{\boldsymbol{A}\boldsymbol{x}}\|_2^2.$$

Problem (9) is equivalent to the following DC optimization problem:

$$\min_{(\boldsymbol{h},\boldsymbol{x})\in \mathrm{cl}\, C} \quad F_1(\boldsymbol{h},\boldsymbol{x}) - F_2(\boldsymbol{h},\boldsymbol{x}) + G(\boldsymbol{h},\boldsymbol{x}). \tag{10}$$

L-smad property for blind deconvolution

Consider the L-smad property of (F_1, H) to apply BPDCA(e).

Theorem (L-smad property [Takahashi et al., 2023])

Let H be defined by

$$H(\boldsymbol{h}, \boldsymbol{x}) = \frac{1}{4}\left(\|\boldsymbol{h}\|_2^2 + \|\boldsymbol{x}\|_2^2\right)^2 + \frac{1}{2}\left(\|\boldsymbol{h}\|_2^2 + \|\boldsymbol{x}\|_2^2\right). \tag{11}$$

By denoting $\boldsymbol{b}_j$ and $\boldsymbol{a}_j$ be the jth column vectors of $\boldsymbol{B}^{\mathrm{H}}$ and $\boldsymbol{A}^{\mathrm{H}}$, respectively, then for any L satisfying

$$L \geq \sum_{j=1}^{m}\left(3\|\boldsymbol{b}_j\|_2^4 + 3\|\boldsymbol{a}_j\|_2^4 + \|\boldsymbol{b}_j\|_2^2\|\boldsymbol{a}_j\|_2^2 + |y_j|^2\|\boldsymbol{b}_j\|_2^2 + \|\boldsymbol{a}_j\|_2^2\right), \tag{12}$$

the pair (F_1, H) is L-smad.

BPDCA(e) converges to a stationary point of (9).

Adjuste L in our numerical experiments.
Backtracking can be applied to BPDCA(e) but its calculations are sometimes expensive.

Numerical experiments: Blind deconvolution

Setting

- The blurring kernel $\boldsymbol{h} \in \mathbb{R}^{d_1}$ has its elements in $\sqrt{d_1} \times \sqrt{d_1}$ pixels ($\sqrt{d_1} = 48$).
- The wavelet coefficients $\boldsymbol{x} \in \mathbb{R}^{d_2}$ ($d_2 = 256^2$).
- $\tilde{\boldsymbol{B}}$ is an operator reshaping $\boldsymbol{h}$, and $\tilde{\boldsymbol{A}}$ is an inverse discrete wavelet transform operator.
  - Therefore, $\boldsymbol{f} = \tilde{\boldsymbol{B}}\boldsymbol{h}^\circ$, $\boldsymbol{g} = \tilde{\boldsymbol{A}}\boldsymbol{x}^\circ$. The pixcels of the original image is 512×512.
- The regularizer $G(\boldsymbol{h}, \boldsymbol{x}) = \theta\|\boldsymbol{h}\|_1$.
- The feasible region $\mathrm{cl}\, C = \{(\boldsymbol{h}, \boldsymbol{x}) \in \mathbb{R}^{d_1} \times \mathbb{R}^{d_2} \mid \boldsymbol{h} \geq \boldsymbol{0}, \boldsymbol{x} \geq \boldsymbol{0}\}$.

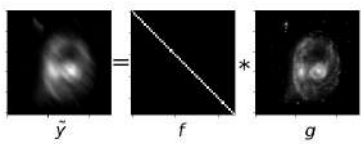


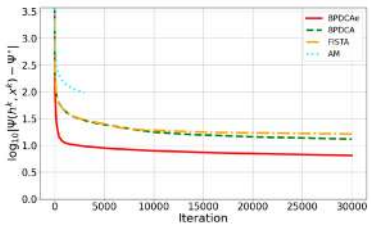
Figure 2: Recover $(\boldsymbol{h}^\circ, \boldsymbol{x}^\circ)$ from $\boldsymbol{y} = \frac{1}{\sqrt{m}}\boldsymbol{F}\tilde{\boldsymbol{y}}$.

Comparisons of algorithms

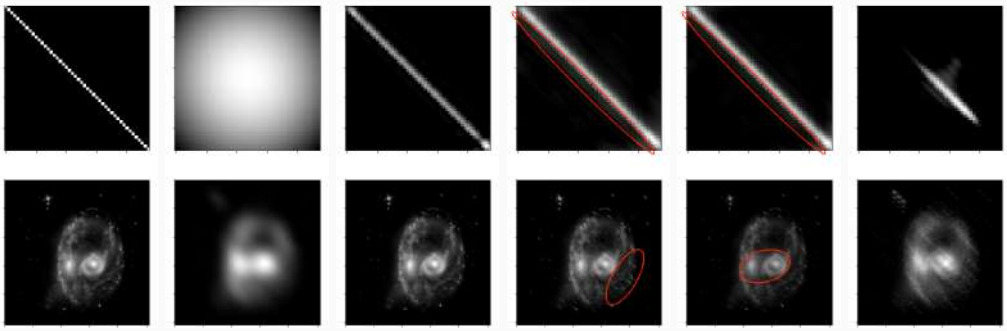
Algorithms

- **BPDCAe** [Takahashi et al., 2022]: The accelerated version of BPDCA.
- **BPDCA** [Takahashi et al., 2022]
- **FISTA** [Beck and Teboulle, 2009]: Adjusts the step size by backtracking.
- **Alternating minimization (AM)**: Minimizes Ψ with respect to $\boldsymbol{h}$ and $\boldsymbol{x}$ alternately (its subproblems are solved by FISTA (10 iterations); the number of the maximum iteration is 3,000).

- $\Psi(\boldsymbol{h}, \boldsymbol{x}) = \frac{1}{2}\|\boldsymbol{B}\boldsymbol{h} \odot \overline{\boldsymbol{A}\boldsymbol{x}} - \boldsymbol{y}\|_2^2 + \theta\|\boldsymbol{h}\|_1$.
- $(\boldsymbol{h}^\circ, \boldsymbol{x}^\circ)$ is the ground truth, $\Psi^\circ = \Psi(\boldsymbol{h}^\circ, \boldsymbol{x}^\circ)$.
- $\theta = 0.01$.



Comparisons of the recovered images



(a) Ground truth **(b)** Intial point **(c)** BPDCAe **(d)** BPDCA **(e)** FISTA **(f)** AM

Figure 3: The upper row: the recovered $\boldsymbol{h}$s. The lower row: the recovered $\boldsymbol{x}$s.

Summary and future work

Summary

- Introduce BPDCA(e) for solving a DC optimization problem without L-smoothness.
- Reformulate blind deconvolution as a DC optimization problem with non-smooth regularization and apply it to BPDCA(e).

Future work

- How to choose the Bregman distance D_H.
- Application to self-calibration in radio interferometric imaging.

Takahashi, S., Fukuda, M., and Tanaka, M. (2022).
New Bregman proximal type algorithms for solving DC optimization problems.
Computational Optimization and Applications, online.

Takahashi, S., Tanaka, M., and Ikeda, S. (2023).
Blind deconvolution with non-smooth regularization via Bregman proximal DCAs.
Signal Processing, 202:108734.

References

References i

[Beck and Teboulle, 2009] Beck, A. and Teboulle, M. (2009).
A fast iterative shrinkage-thresholding algorithm for linear inverse problems.
SIAM Journal on Imaging Sciences, 2(1):183–202.

[Bolte et al., 2018] Bolte, J., Sabach, S., Teboulle, M., and Vaisbourd, Y. (2018).
First order methods beyond convexity and Lipschitz gradient continuity with applications to quadratic inverse problems.
SIAM Journal on Optimization, 28(3):2131–2151.

[Li et al., 2019] Li, X., Ling, S., Strohmer, T., and Wei, K. (2019).
Rapid, robust, and reliable blind deconvolution via nonconvex optimization.
Applied and Computational Harmonic Analysis, 47(3):893–934.

[Takahashi et al., 2022] Takahashi, S., Fukuda, M., and Tanaka, M. (2022).
New Bregman proximal type algorithms for solving DC optimization problems.
Computational Optimization and Applications, online.

References ii

[Takahashi et al., 2023] Takahashi, S., Tanaka, M., and Ikeda, S. (2023).
Blind deconvolution with non-smooth regularization via Bregman proximal DCAs.
Signal Processing, 202:108734.

[Wen et al., 2017] Wen, B., Chen, X., and Pong, T. K. (2017).
A proximal difference-of-convex algorithm with extrapolation.
Computational Optimization and Applications, 69(2):297–324.

Theoretical Analysis for Representation Learning Methods of Graph-Structured Data

Nozomi HATA
Institute of Mathematics for Industry, Kyushu University, Japan
n.hata@kyudai.jp

Graph-structured data is one of the representative discrete data structures, while continuous data is usually represented as vectors. Continuous representation of graph-structured data refers to the assignment of vectors to nodes in the graph-structured data so that the relationship between two nodes can be recovered using these vectors. By representing graph-structured data in continuous space, we can apply various algorithms in continuous space to real-world applications such as link prediction, attribute prediction, information retrieval, and question answering while preserving combinatorial characteristics of graph-structured data. In this presentation, we focus on the theoretical representational power of representation methods. Some representation methods, such as [1] or [2], can represent any inputs accurately. Such a property is called full expressiveness. We theoretically proved that some representation methods which are not fully expressive are, in fact, almost fully expressive. This presentation introduces almost fully expressive models and shows numerical results for link prediction tasks.

References

[1] Ivana Balažević, Carl Allen, and Timothy M. Hospedales, “TuckER: Tensor Factorization for Knowledge Graph Completion”, Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP). 2019.
[2] Ralph Abboud, Ismail Ceylan, Thomas Lukasiewicz, and Tommaso Salvatori, "BoxE: A Box Embedding Model for Knowledge Base Completion." Advances in Neural Information Processing Systems 33 (2020): 9649-9661.

Theoretical Analysis for Representation Learning Methods of Graph-Structured Data

Nozomi Hata*, Katsuki Fujisawa

Kyushu University, Japan

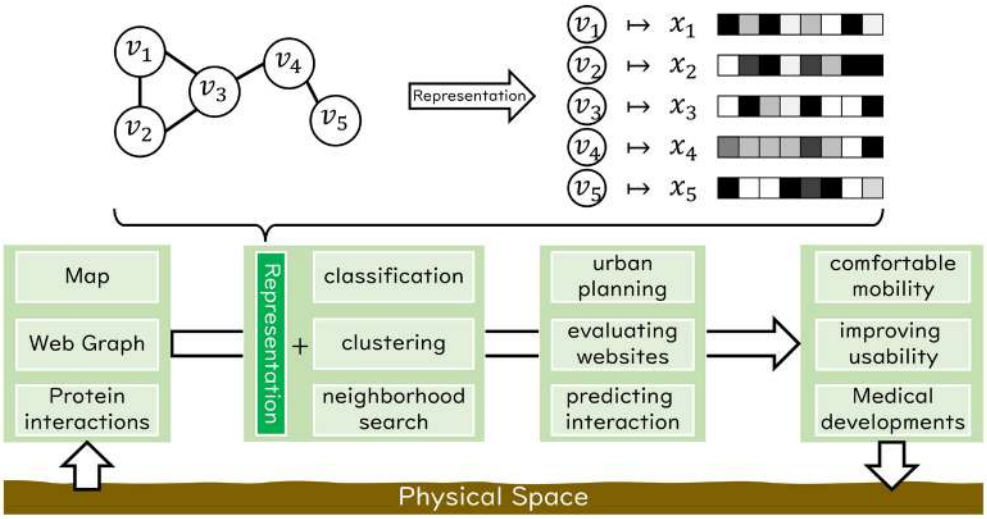
The 6th RIKEN-IMI-ISM-NUS-ZIB-MODAL-NHR
Workshop on Advances in Classical and Quantum
Algorithms for Optimization and Machine Learning

September 18th, 2022

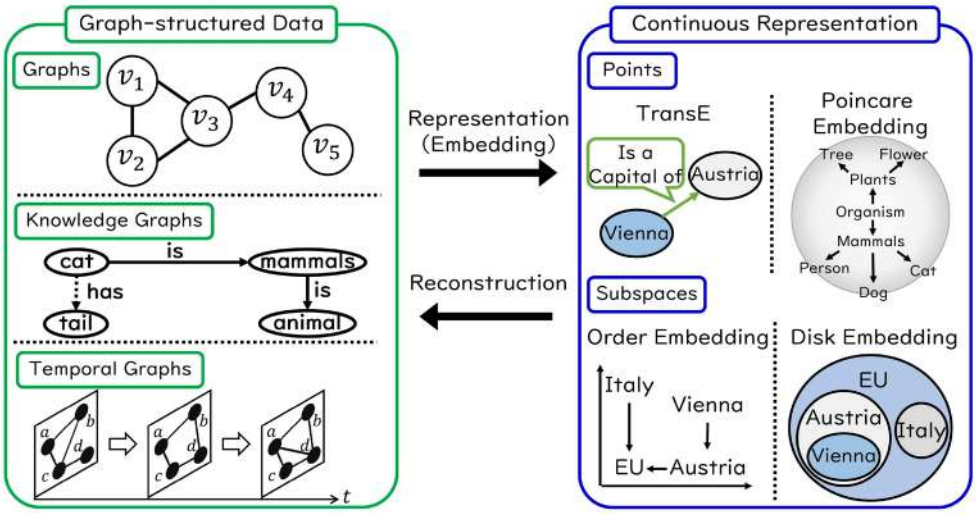
Table of Contents

Table of Contents

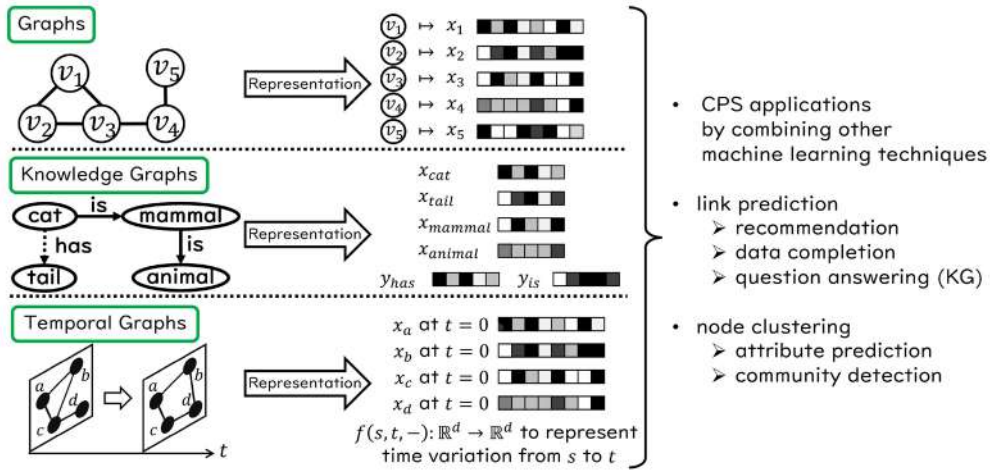
Applications of graph representation



Graph-structured Data and its representation



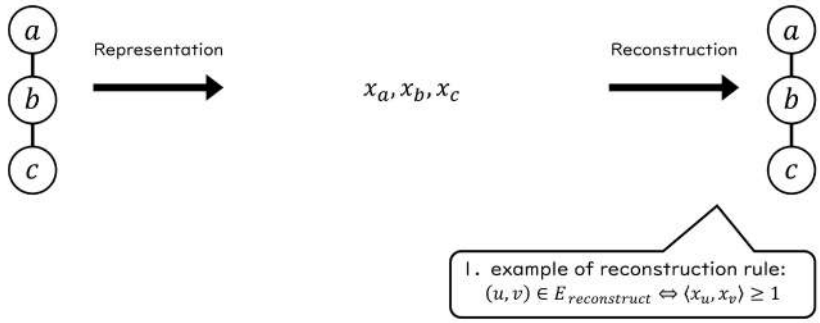
Which objects are vectorized?



Our interest: representational capacity

Some representational methods ignore some information, and some do not

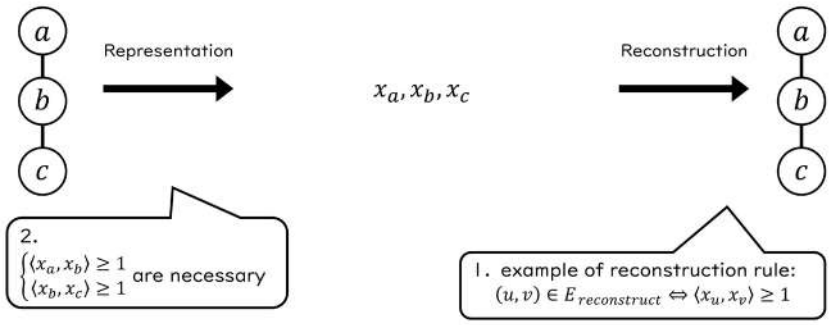
Example: inner product-based representation for undirected graphs



Our interest: representational capacity

Some representational methods ignore some information, and some do not

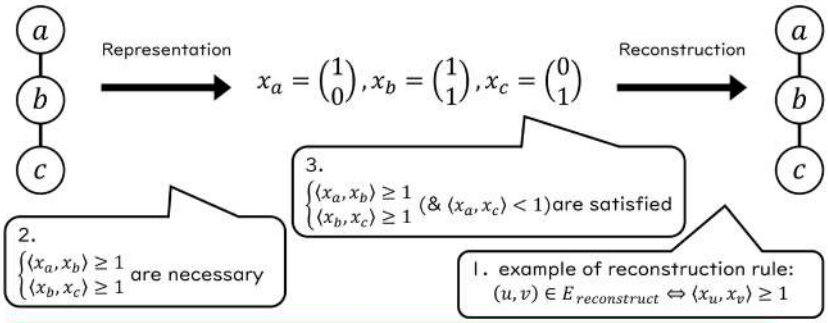
Example: inner product-based representation for undirected graphs



Our interest: representational capacity

Some representational methods ignore some information, and some do not

Example: inner product-based representation for undirected graphs



$1 \leq \langle x_a, x_b\rangle \leq \max\{\|x_a\|_2^2, \|x_b\|_2^2\} \Rightarrow$ fail to represent the non-existence of self-loop

Our interest: representational capacity

Some representational methods ignore some information, and some do not

Example: inner product-based representation for undirected graphs

What information is represented or ignored?

↓

we can select appropriate model according to their characteristics

2.
$\begin{cases}\langle x_a, x_b\rangle \geq 1 \\ \langle x_b, x_c\rangle \geq 1\end{cases}$ are necessary

1. example of reconstruction rule:
$(u, v) \in E_{reconstruct} \Leftrightarrow \langle x_u, x_v\rangle \geq 1$

$1 \leq \langle x_a, x_b\rangle \leq \max\{\|x_a\|_2^2, \|x_b\|_2^2\} \Rightarrow$ fail to represent the non-existence of self-loop

Definition: full expressiveness

<u>Def</u> (full expressiveness)

A representation model $\mathcal{M}$ is fully expressive for class $\mathcal{C}$

$\Leftrightarrow$ For any input $\mathcal{G}$ in $\mathcal{C}$, there exists a "good" representation $\mathcal{X}_{\mathcal{G}}$ s.t. $\mathcal{M}$ can recover $\mathcal{G}$ from $\mathcal{X}_{\mathcal{G}}$

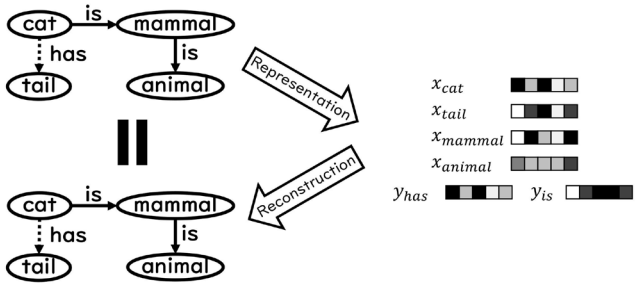
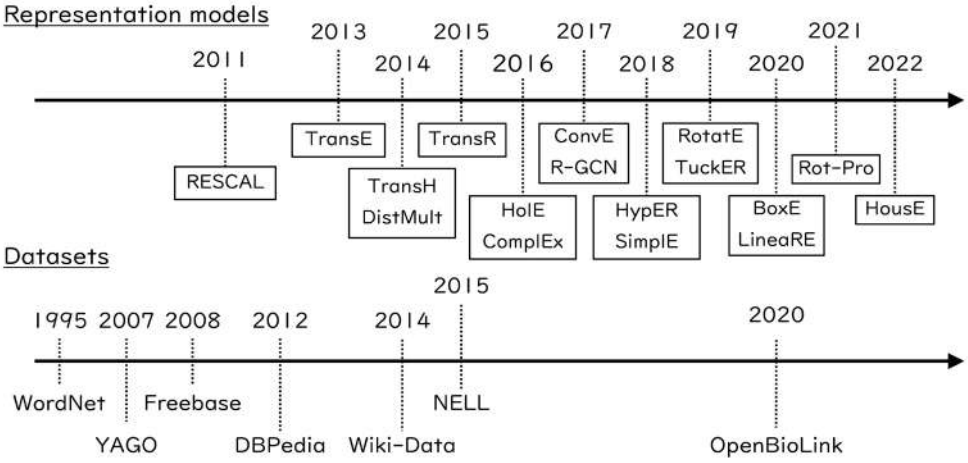


Table of Contents

Datasets and Models



Target: Knowledge Graph Representation

Geometric models

Relations between entities

→ geometric relation between vectors

ex) TransE, RotatE, BoxE

Bilinear models

KGs → 3rd-order tensor

representation = dimensionality reduction

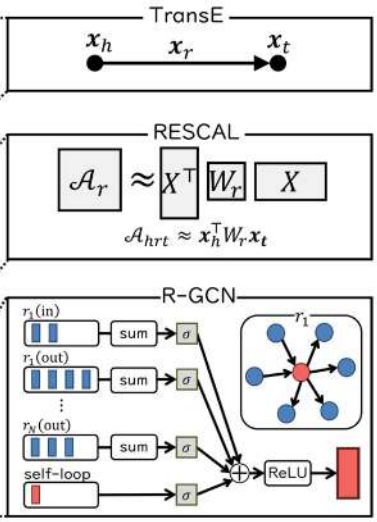
ex)RESCAL, ComplEx, TuckER

Other models

Deep learning-based models,

probabilistic models, ⋯

ex) R-GCN, GAATs, CapsE



Target: Knowledge Graph Representation

Geometric models

Relations between entities
→ geometric relation between vectors
ex) TransE, RotatE, BoxE

⇒ reconstruction by distance of vectors

Bilinear models

KGs → 3rd-order tensor
representation = dimensionality reduction
ex)RESCAL, ComplEx, TuckER

⇒ reconstruction by inner product of vectors

Other models

Deep learning-based models,
probabilistic models, ⋯
ex) R-GCN, GAATs, CapsE

⇒ reconstruction by method-specific decoders
e.g. neural networks

Target: Knowledge Graph Representation

Geometric models

Relations between entities
→ geometric relation between vectors
ex) TransE, RotatE, BoxE

Bilinear models

KGs → 3rd-order tensor
representation = dimensionality reduction
ex)RESCAL, ComplEx, TuckER

Other models

Deep learning-based models,
probabilistic models, ⋯
ex) R-GCN, GAATs, CapsE

Most of them have high
(or theoretically perfect)
representational capacity
based on

- matrix factorization techniques
- universal approximation theorem

Target: Knowledge Graph Representation

Geometric models

Relations between entities
→ geometric relation between vectors
ex) TransE, RotatE, BoxE

Bilinear models

KGs → 3rd-order tensor
representation = dimensionality reduction
ex)RESCAL, ComplEx, TuckER

Other models

Deep learning-based models,
probabilistic models, ⋯
ex) R-GCN, GAATs, CapsE

Most methods do not focus on the theoretical representational power, and focus on the pattern recognition on KGs

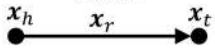
How about theoretical differences?

Target: Knowledge Graph Representation

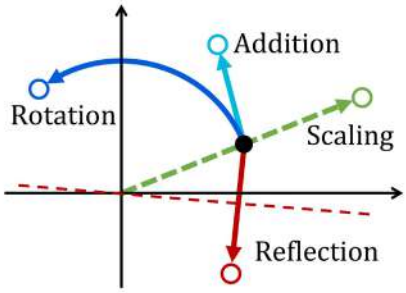
Geometric models

Relations between entities
→ geometric relation between vectors
ex) TransE, RotatE, BoxE

TransE



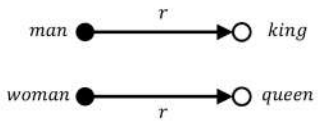
- In experiment, distance-based reconstruction combining these operations performs well
- Most papers explain that the performance is mainly because the operation can recognize implicit patterns in KGs



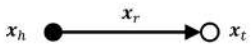
Example: TransE

TransE translates fact (h, r, t) into an equation $x_h + x_r = x_t$

- inspired by word2vec
- only uses addition
- When we set the reconstruction rule as
  $$(h, r, t) \in F_{predict} \Leftrightarrow x_h + x_r = x_t$$
- TransE is said to have low representation capacity because of the simplicity



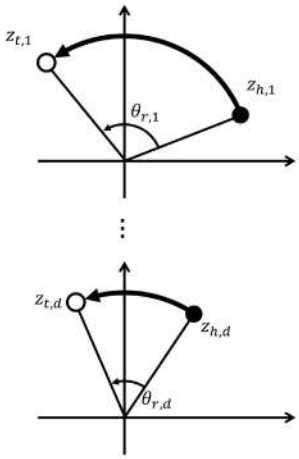
$$x_{king} - x_{man} + x_{woman} \approx x_{queen}$$
$$\rightarrow x_{king} - x_{man} \approx x_{queen} - x_{woman}$$
$$\Rightarrow \exists y\ s.t. \begin{cases} x_{man} + y \approx x_{king} \\ x_{woman} + y \approx x_{queen} \end{cases}$$



Example: RotatE

RotatE translates fact (h, r, t) into an equation $z_h \circ e^{i\theta_r} = z_t \left(z_h, z_t \in \mathbb{C}^d, \theta_r \in \mathbb{R}^d\right)$

- only uses 2D rotation and uses d planes
- RotatE has higher representation capacity than TransE
  - TransE cannot distinguish two symmetric relations while RotatE can
- We will show that <u>RotatE can represent any KGs</u> by setting reconstruction rule as
  $$\|e^{i\theta_r} \circ z_h - z_t\|_\infty \leq \varepsilon\ (\varepsilon > 0: const.)$$



Example: BoxE

BoxE translates fact (h, r, t) into an element-wise inequality in $\mathbb{R}^d$

$$\begin{cases} \left| x_h + y_t - c_r^{(1)} \right| \leq w_r^{(1)} \\ \left| y_h + x_t - c_r^{(2)} \right| \leq w_r^{(2)} \end{cases}$$

- Since inequality can be rewritten as

$$\left| \begin{pmatrix} x_h \\ y_h \end{pmatrix} + \begin{pmatrix} O & I \\ I & O \end{pmatrix} \begin{pmatrix} x_t \\ y_t \end{pmatrix} - \begin{pmatrix} c_r^{(1)} \\ c_r^{(2)} \end{pmatrix} \right| \leq \begin{pmatrix} w_r^{(1)} \\ w_r^{(2)} \end{pmatrix},$$

  BoxE utilizes reflection
- The first geometric model which is proven to be fully expressive

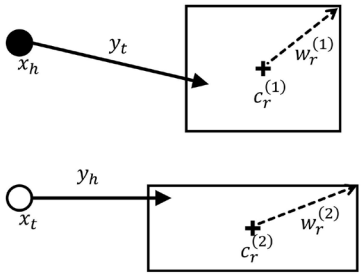


Table of Contents

Theoretical Analysis

Consider the following model $\mathcal{M}$:

- Representation:
  $V \ni v \mapsto z_v \in K^d$
  $R \ni r \mapsto f_r: K^d \to K^d$ (geometric operator)
- Reconstruction rule:
  $(h, r, t) \in F_{predict} \Leftrightarrow \|f_r(z_h) - z_t\|_\infty \leq \varepsilon$

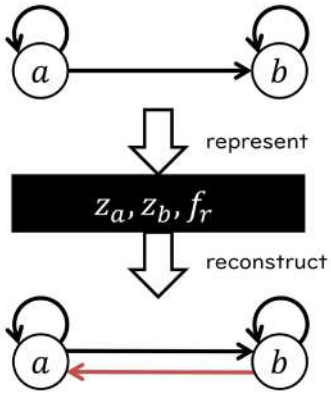
where $\varepsilon \geq 0$: constant and $K = \mathbb{R}$ or $\mathbb{C}$.

1. When $\varepsilon = 0$, i.e.,
   $(h, r, t) \in F_{predict} \Leftrightarrow f_r(z_h) = z_t$,
   there exists a KG which $\mathcal{M}$ cannot reconstruct.

To represent the graph on the right figure,

$$\begin{cases} f_r(z_a) = z_a, f_r(z_a) = z_b, f_r(z_b) = z_b \\ f_r(z_b) \neq z_a \end{cases}$$

requires, however, $f_r(z_b) = z_b = f_r(z_a) = z_a$.



Theoretical Analysis

Consider the following model $\mathcal{M}$:

- Representation:
  $V \ni v \mapsto z_v \in K^d$
  $R \ni r \mapsto f_r: K^d \to K^d$
- Reconstruction rule:
  $(h, r, t) \in F_{predict} \Leftrightarrow \|f_r(z_h) - z_t\|_\infty \leq \varepsilon$

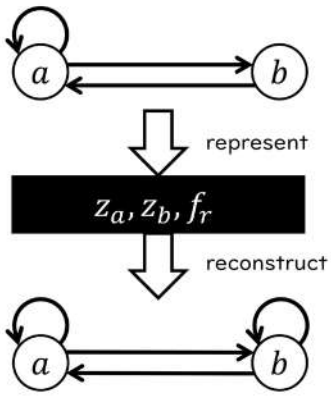
where $\varepsilon \geq 0$: constant and $K = \mathbb{R}$ or $\mathbb{C}$.

2. (Addition) When $\varepsilon > 0, K = \mathbb{R}$ and
   f_r can be written as $f_r(z_h) = z_h + z_r$ $(z_r \in \mathbb{R}^d)$,
   there exists a KG which $\mathcal{M}$ cannot reconstruct.

$\|z_a + z_r - z_a\|_\infty = \|z_r\|_\infty \leq \varepsilon$ is required

to represent $a \to a$, which causes that $\mathcal{M}$ cannot

represent the inexistence of $b \to b$.



Theoretical Analysis

Consider the following model $\mathcal{M}$:

- Representation:

$$V \ni v \mapsto z_v \in K^d$$
$$R \ni r \mapsto f_r : K^d \to K^d$$

- Reconstruction rule:

$$(h, r, t) \in F_{predict} \Leftrightarrow \|f_r(z_h) - z_t\|_\infty \leq \varepsilon$$

where $\varepsilon \geq 0$: constant and $K = \mathbb{R}$ or $\mathbb{C}$.

3. (Reflection) When $\varepsilon > 0, K = \mathbb{C}$ and

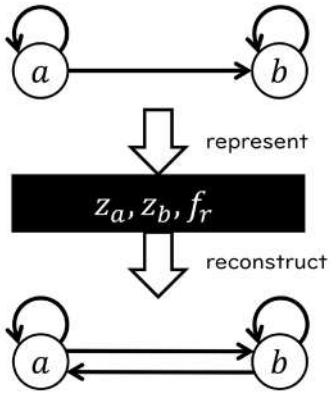
$$f_r(z_h) = w_r \circ \bar{z}_h \ (w_r \in \mathbb{C}^d, |w_r| = 1),$$

there exists a KG which $\mathcal{M}$ cannot reconstruct.

Since $\|f_r(z_a) - z_b\|_\infty = \|f_r(z_b) - z_a\|_\infty$,

reflection operator cannot represent

directed edges.



Theoretical Analysis

Consider the following model $\mathcal{M}$:

- Representation:

$$V \ni v \mapsto z_v \in K^d$$
$$R \ni r \mapsto f_r : K^d \to K^d$$

- Reconstruction rule:

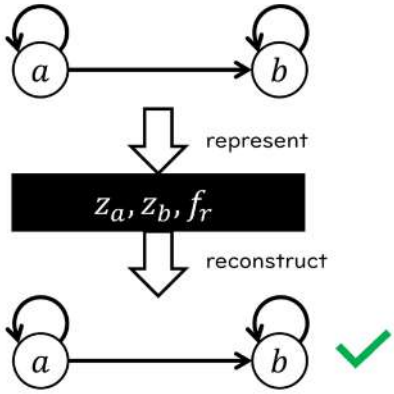
$$(h, r, t) \in F_{predict} \Leftrightarrow \|f_r(z_h) - z_t\|_\infty \leq \varepsilon$$

where $\varepsilon \geq 0$: constant and $K = \mathbb{R}$ or $\mathbb{C}$.

4. (Rotation) When $\varepsilon > 0, K = \mathbb{C}$ and

$$f_r(z_h) = e^{i\theta_r} \circ z_h \ (\theta_r \in \mathbb{R}^d),$$

$\mathcal{M}$ can represent any KGs
with sufficiently large dimension d.



*$e^{i\theta_r}$: element-wise exponential function
**$z[k]$: k-th component of vector z

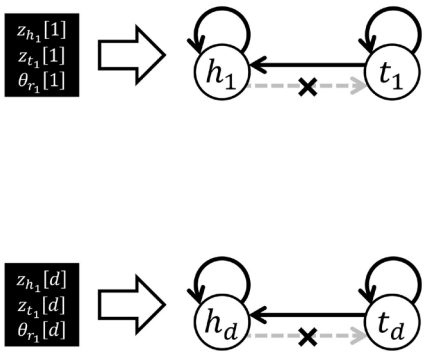
Theoretical Analysis

4. (Rotation) When $\varepsilon > 0, K = \mathbb{C}$ and
$$f_r(z_h) = e^{i\theta_r} \circ z_h \ (\theta_r \in \mathbb{R}^d),$$
$\mathcal{M}$ can represent any KGs
with sufficiently large dimension d under
the reconstruction rule
$$(h, r, t) \in F_{predict} \Leftrightarrow \|f_r(z_h) - z_t\|_\infty \leq \varepsilon.$$

Idea of proof:

- Using uniform distance:
  $(h, r, t) \in F_{predict} \Leftrightarrow |f_r(z_h)[j] - z_t[j]| \leq \varepsilon \ (\forall j).$
  $(h, r, t) \notin F_{predict} \Leftrightarrow \exists j \ s.t. \ |f_r(z_h)[j] - z_t[j]| > \varepsilon.$

1. Index $V \times R \times V \setminus F = \{(h_j, r_j, t_j)\}_{j=1}^{d}$
2. j-th component of vectors
represent only $(h_j, r_j, t_j) \notin F$



*$e^{i\theta_r}$: element-wise exponential function
**$z[j]$: j-th component of vector z

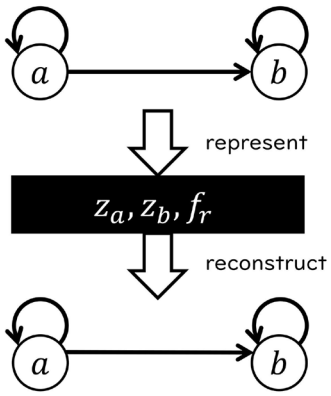
Theoretical Analysis

Consider the following model $\mathcal{M}$:

- Representation:
$$V \ni v \mapsto z_v \in K^d$$
$$R \ni r \mapsto f_r : K^d \to K^d$$
- Reconstruction rule:
$$(h, r, t) \in F_{predict} \Leftrightarrow \|f_r(z_h) - z_t\|_\infty \leq \varepsilon$$

where $\varepsilon \geq 0$: constant and $K = \mathbb{R}$ or $\mathbb{C}$.

4. (Scaling) When $\varepsilon > 0, K = \mathbb{R}$ and
$$f_r(z_h) = z_r \circ z_h \ (z_r \in \mathbb{R}^d),$$
$\mathcal{M}$ can represent any KGs
with sufficiently large dimension d.



Proof

$$\boldsymbol{x}_v[j] = \begin{cases} \varepsilon & (h_j = t_j = v), \\ \varepsilon/2 & (v = h_j \neq t_j), \\ -\varepsilon/5 & (h_j \neq t_j = v), \\ 0 & \text{(otherwise)}. \end{cases} \qquad \boldsymbol{x}_r[j] = \begin{cases} -1 & (r = r_j, h_j = t_j), \\ 2 & (r = r_j, h_j \neq t_j) \\ 0 & (r \neq r_j). \end{cases}$$

$$\boldsymbol{z}_v[j] = \begin{cases} \varepsilon & (h_j = t_j = v), \\ (-1+i)\varepsilon/2 & (v = h_j \neq t_j), \\ (1+i)\varepsilon/2 & (h_j \neq t_j = v), \\ 0 & \text{(otherwise)}. \end{cases} \qquad e^{i\boldsymbol{\theta}_r[j]} = \begin{cases} i & (r = r_j), \\ 1 & (r \neq r_j). \end{cases}$$

			$\lvert \boldsymbol{x}_r[j]\boldsymbol{x}_h[j] - \boldsymbol{x}_t[j] \rvert$		$\lvert e^{i\boldsymbol{\theta}_r[j]}\boldsymbol{z}_h[j] - \boldsymbol{z}_t[j] \rvert$	
			$r = r_j$	$r \neq r_j$	$r = r_j$	$r \neq r_j$
$h_j = t_j$	$h = h_j$	$t = h_j$	$\boxed{2\varepsilon}$	ε	$\boxed{\sqrt{2}\varepsilon}$	0
		$t \neq h_j$	ε	0	ε	ε
	$h_j \neq t_j$	$t = t_j$	ε	ε	ε	ε
		$t \neq t_j$	0	0	0	0
$h_j \neq t_j$	$h = h_j$	$t = h_j$	$\varepsilon/2$	$\varepsilon/2$	ε	0
		$t = t_j$	$\boxed{6\varepsilon/5}$	$\varepsilon/5$	$\boxed{\sqrt{2}\varepsilon}$	ε
		$t \neq h_j, t_j$	ε	0	$\varepsilon/\sqrt{2}$	$\varepsilon/\sqrt{2}$
	$h = t_j$	$t = h_j$	$9\varepsilon/10$	$\varepsilon/2$	0	ε
		$t = t_j$	$\varepsilon/5$	$\varepsilon/5$	ε	0
		$t \neq h_j, t_j$	$2\varepsilon/5$	0	$\varepsilon/\sqrt{2}$	$\varepsilon/\sqrt{2}$
	$h \neq h_j, t_j$	$t = h_j$	$\varepsilon/2$	$\varepsilon/2$	$\varepsilon/\sqrt{2}$	$\varepsilon/\sqrt{2}$
		$t = t_j$	$\varepsilon/5$	$\varepsilon/5$	$\varepsilon/\sqrt{2}$	$\varepsilon/\sqrt{2}$
		$t \neq h_j, t_j$	0	0	0	0

$x_v[j], x_r[j]$ for scaling / $z_v[j], \theta_r[j]$ for rotation

Table of Contents

Experimental Setup: Learning Method

Training

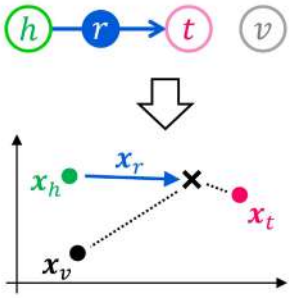
1. Define score function $S_r(h,t)$ according to model
2. Train embedding as $S_r(h,t) \rightarrow \begin{cases} \text{large } \big((h,r,t) \in F\big) \\ \text{small } (o/w) \end{cases}$

Ex) TransE learns embedding as

$$(h,r,t) \in T \Leftrightarrow \boldsymbol{x}_h + \boldsymbol{x}_r = \boldsymbol{x}_t$$

$\rightarrow S_r(h,t) := -\|\boldsymbol{x}_h + \boldsymbol{x}_r - \boldsymbol{x}_t\|$

(h,r,t) is regarded as a fact if $S_r(h,t)$ is sufficiently large



model	Entity embedding	Relation embedding	expected condition	$S_r(h,t)$
TransE	$x_v \in \mathbb{R}^d$	$x_r \in \mathbb{R}^d$	$x_h + x_r = x_t$	$-\|x_h + x_r - x_t\|$
RotatE	$z_v \in \mathbb{C}^d$	$\theta_r \in \mathbb{R}^d$	$z_h \circ e^{i\theta_r} = z_t$	$-\|z_h \circ e^{i\theta_r} - z_t\|$
BoxE	$x_v, y_v \in \mathbb{R}^d$	$c_r^{(1)}, c_r^{(2)}, w_r^{(1)}, w_r^{(2)} \in \mathbb{R}^d$	$\begin{cases} \lvert x_h + y_t - c_r^{(1)} \rvert \le w_r^{(1)} \\ \lvert y_h + x_t - c_r^{(2)} \rvert \le w_r^{(2)} \end{cases}$	(see original paper)

Experimental Setup: Evaluation Metrics

Training

1. Define score function $f_r(h,t)$ according to model
2. Train embedding as $f_r(h,t) \rightarrow \begin{cases} \text{large } \big((h,r,t) \in F_{data}\big) \\ \text{small } (o/w) \end{cases}$

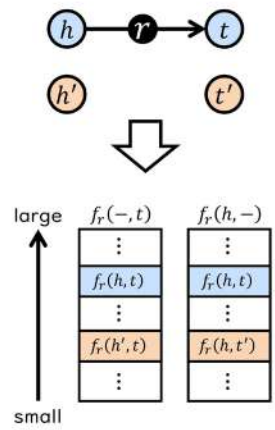
Evaluation

For test triple (h,r,t),

1. Sort $P_{r,t} = \{(h',r,t) \mid h' : \text{entity}\}, Q_{h,r} = \{(h,r,t') \mid t' : \text{entity}\}$ according to the score function*
2. rank of (h,r,t) in P or Q is used for evaluation

Mean Reciprocal Rank (MRR): mean of $1/rank(h,r,t)$.

HITS@k: percentage of test triples with $rank(h,r,t)$.



*If (h',r,t) or (h,r,t') is in the training / valid / test dataset, these are excluded for evaluation (filtered setting).

Experimental Setup: Datasets

Dataset:

- FB15k-237 (⊂ FreeBase): KG of general facts / various relations
- WN18RR (⊂ WordNet): KG of words / hierarchy & similarity
- YAGO3-10 (⊂ YAGO3): KG of people, countries, …… / largest

dataset	FB15k-237	WN18RR	YAGO3-10
#entity	14,541	40,943	123,182
#relation	237	11	37
#training	272,115	86,835	1,079,040
#valid	17,535	3,034	5,000
#test	20,466	3,134	5,000

Experimental Results

Model	f_r	**scoring**	**FB15k-237**		**WN18RR**		**YAGO3-10**	
			MRR	H@10	MRR	H@10	MRR	H@10
TransE	Addition	$\mathbb{R}^{1,000}, L_1$	.323	.522	.232	.539	.439	.645
-	Scaling	$\mathbb{R}^{1,000}, L_1$	.335	**.527**	.473	.573	.497	.663
RotatE	Rotation	$\mathbb{C}^{500}, L_1$	.330	.521	.480	.580	.452	.641
-	Reflection	$\mathbb{C}^{500}, L_1$	.325	.517	.448	.533	.455	.642
MuRE*	Scaling + Addition	$\mathbb{R}^{1,000}, L_1$	**.337**	.525	.477	.563	**.509**	**.676**
RotE*	Rotation + Addition	$\mathbb{C}^{500}, L_1$	.336	.522	**.484**	**.582**	.481	.658
RefE*	Reflection + Addition	$\mathbb{C}^{500}, L_1$	.334	.520	.472	.568	.505	.665
BoxE**	$\pi/2$-Reflection + Addition	using boxes	.318	.505	.443	.541	.480	.650

- 30 times hyperparameter search using Optuna
- 1,000 parameters per entity
- Theoretical difference appears to the link prediction task
- Combination of addition slightly improves the results

Table of Contents

Conclusion & Future Work

Theoretical analysis:

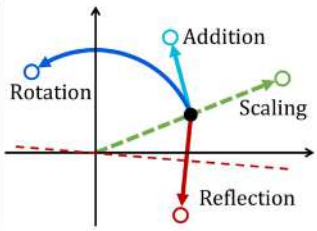
rotation & scaling operators have high representational capacity compared to addition & reflection

Numerical experiment (link prediction):

- results reflected theoretical analysis
- combining multiple operator improves representational power

Future work:

- to analyze other types of representation models
  e.g., for weighted graphs, temporal graphs, ⋯
- to find good representation model
  - with high representational power
  - while reducing the number of parameters



Port Set Clustering for Internet-Wide Scanner

Akira TANAKA, Chansu Han, Takeshi Takahashi

National Institute of Information and Communications Technology, Japan
tanaka.akira@nict.go.jp

Indiscriminate IoT attacks have increased in recent years. Adversaries confirm if vulnerable destination ports are open as a preliminary step of the attack, and this procedure is called port scan. The darknet, also known as a network telescope, is used for observing such port scan activities. It passively monitors network traffic with an unreachable dark IP address block; thus, it receives not regular network traffic but Internet-wide scans for attack or investigation. Our goal is to specify scan activities for attack purposes by focusing on the destination ports of packets collected from a darknet. We treat each source IP address as a multiset made from the pairs of the destination port and the number of packets. We create a distance on the multisets and perform clustering using the distance. Multisets contribute to more fine clustering than clustering using port sets or the number of packets. We also propose the speedup technique for clustering based on the property of the distance.

1

Port Set Clustering for Internet-Wide Scanner

Akira Tanaka, Chansu Han, Takeshi Takahashi

National Institute of Information and Communications Technology

The 6th RIKEN-IMI-ISM-NUS-ZIB-MODAL-NHR Workshop on Advances in Classical and Quantum Algorithms for Optimization and Machine Learning
September 19th, 2022

NICTER
Network Incident analysis Center for Tactical Emergency Response

2

Contribution

- We propose a metric space on multiset and prove that the metric satisfies well-known properties. (positivity, symmetry, and triangle inequality)

Triangle inequality $d(x,z) \leq d(x,y) + d(y,z)$

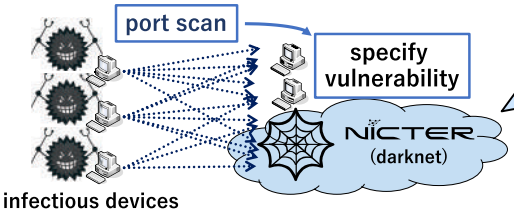
- We propose the fast DBSCAN for the metric space.
  - The output of the fast DBSCAN is the same as the original DBSCAN.
  - The fast DBSCAN reduces 99.5% computation cost compared to the original DBSCAN in our experiments.

NICTER
Network Incident analysis Center for Tactical Emergency Response

Contents

Background

- Indiscriminate attacks have been increasing in the IP address space.
- Scan activities derived from malware (dangerous software) are observed by darknets.



- passively monitors network traffic with an unreachable dark IP address block →receive port scans from malware or an Internet investigation institute
- observe only first contact
  - Intention of a scanner is unclear. (we guess it from a destination port)

Our Goal	We want to grasp emerging threats

Contents

5

- Background
- → **Malware/Botnet behavior**
- → **Research direction**
- Multiset
- Approach
- Metric on multiset
- DBSCAN
- Characteristic of our data
- Fast DBSCAN
  - Partition of data
  - Ball-based DBSCAN
- Experiment

NICT National Institute of Information and Communications Technology

NICTER
Network Incident analysis Center for Tactical Emergency Response

Malware/Botnet behavior①

6

1. Looking for targets
   They send packets to random IP addresses to specify hosts that run vulnerable service/software (port scans).

2. Infection
   They exploit the vulnerability and gain control of the host.
   – They try to authenticate via a set of known default credentials.

3. Attack
   Infectious devices are used for DDoS attacks, email spam, and cryptocurrency mining.

NICT National Institute of Information and Communications Technology

NICTER
Network Incident analysis Center for Tactical Emergency Response

Malware/Botnet behavior②

7

1. Looking for targets
   They send packets to random IP addresses to specify hosts that run vulnerable service/software (port scans).

- Each malware has its target services/destination ports.
- A vulnerable service/software has a default port, and malware checks if the port is open (port scans).
- Darknet can monitor port scans.

Do we detect malware from the destination ports collected by the darknet?

NICTER
Network Incident analysis Center for Tactical Emergency Response

Malware/Botnet behavior③

8

- Botnet/malware has its target destination ports.
- Infected devices are chronologically synchronized and suspected to send about the same number of packets to the target destination ports.

Because

- C&C server commands the infected devices to perform port scans all at once.
- Port scan script is open access and reused by many attackers.

NICTER
Network Incident analysis Center for Tactical Emergency Response

Research direction

9

- Assumption
  Infected devices are chronologically synchronized and suspected to send about the same number of packets to the target destination ports.

- Research direction
  We perform clustering so that if two IP addresses are in the same cluster, the two IP addresses send about the same number of packets to the target destination ports.

NICT National Institute of Information and Communications Technology

NICTER
Network Incident analysis Center for Tactical Emergency Response

Contents

10

- Background
- Malware/Botnet behavior
- Research direction
- → **Multiset**
- Approach
- Metric on multiset
- DBSCAN
- Characteristic of our data
- Fast DBSCAN
  - Partition of data
  - Ball-based DBSCAN
- Experiment

NICT National Institute of Information and Communications Technology

NICTER
Network Incident analysis Center for Tactical Emergency Response

Multiset

Definition

A collection of unordered elements, where every element occurs a finite number of times

$m_A(1) = 3, m_A(2) = 1, m_A(3) = 3,$

Example

$A = \{2, 1,1,1,3,3\}$ ← Another expression → multiplicity of 1 is three

$$m_A(x) = \begin{cases} 3 & \text{if } x = 1 \\ 1 & \text{if } x = 2 \\ 2 & \text{if } x = 3 \\ 0 & \text{otherwis} \end{cases}$$

multiplicity function

Interpretation of the multiset in our problem

$\text{Supp}(A) = \{1,2,3\}$

The IP address A sent one packet to port 2, three packets to port 1, and two packets to port 3.

NICT National Institute of Information and Communications Technology

NICTER Network Incident analysis Center for Tactical Emergency Response

Contents

- Background
- Malware/Botnet behavior
- Research direction
- Multiset
- → **Approach**
- Metric on multiset
- DBSCAN
- Characteristic of our data
- Fast DBSCAN
  - Partition of data
  - Ball-based DBSCAN
- Experiment

NICT National Institute of Information and Communications Technology

NICTER Network Incident analysis Center for Tactical Emergency Response

Approach

1. We propose a metric on multiset that satisfies
   - the metric becomes small the if multiplicity of each element is almost the same.

multiplicity of 1 is three

$A = \{2, 1,1,1,3,3\}$

2. We perform clustering (DBSCAN) based on the metric.
   - We speed up the DBSCAN.

A cluster may represent a malware/botnet.

NICTER
Network Incident analysis Center for Tactical Emergency Response

Contents

- Background
- Malware/Botnet behavior
- Research direction
- Multiset
- Approach
- → **Metric on multiset**
- DBSCAN
- Characteristic of our data
- Fast DBSCAN
  - Partition of data
  - Ball-based DBSCAN
- Experiment

NICTER
Network Incident analysis Center for Tactical Emergency Response

Metric on a multiset

- If $x \in \mathrm{supp}(A)\,\Delta\,\mathrm{supp}(B) \Rightarrow$ $+1$ to metric
- If $x \in \mathrm{supp}(A) \cap \mathrm{supp}(B) \Rightarrow$ $+t(0 \le t < 1)$ to metric, where t is based on the multiplicity difference($t = 0$ if multiplicities equal)

Min-max normalization term

$$d(A,B) := \frac{1}{|\mathrm{Supp}(A \cup B)|} \sum_{x \in \mathrm{Supp}(A \cup B)} \frac{m_{A\Delta B}(x)}{m_{A \cup B}(x)}$$

$$= \frac{1}{3}\left(\frac{1-1}{1} + \frac{3-2}{3} + \frac{2-0}{2}\right)$$

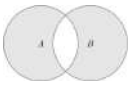
$A = \{2, 1, 1, 1, \quad\}$

$B = \{2, \quad 11, 3, 3\}$

$A \cup B = \{2, 1, 1, 1, 3, 3\}$

$A \Delta B = \{ \qquad 1, 3, 3\}$

- Support
  $\mathrm{Supp}(A) = \{x \mid m_A(x) > 0\}$
- Union
  $m_{A \cup B}(x) := \max\{m_A(x), m_B(x)\}$
- Symmetric difference
  $m_{A\Delta B}(x) := \max\{m_A(x), m_B(x)\} - \min\{m_A(x), m_B(x)\}$



NICTER
Network Incident analysis Center for Tactical Emergency Response

Properties of metric

$$d(A,B) := \frac{1}{|\mathrm{Supp}(A \cup B)|} \sum_{x \in \mathrm{Supp}(A \cup B)} \frac{m_{A\Delta B}(x)}{m_{A \cup B}(x)}$$

- The metric is small if the multiplicity of each element is almost the same.
- We prioritize support differences over multiplicity differences.
- $0 \le d(A,B) \le 1$
- The function d satisfies the following well-known metric properties;
  - $d(A,A) = 0$
  - (Positivity) $d(A,B) > 0$ if $A \ne B$
  - (Symmetry) $d(A,B) = d(B,A)$
  - (Triangle inequality) $d(A,B) \le d(A,B) + d(B,C)$

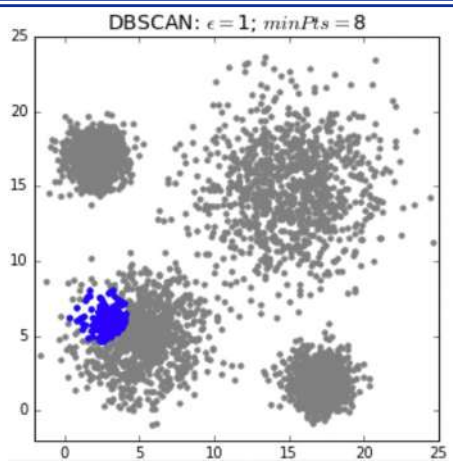
NICTER
Network Incident analysis Center for Tactical Emergency Response

Contents

- Background
- Malware/Botnet behavior
- Research direction
- Multiset
- Approach
- Metric on multiset
- → **DBSCAN**
- Characteristic of our data
- Fast DBSCAN
  - Partition of data
  - Ball-based DBSCAN
- Experiment

NICTER
Network Incident analysis Center for Tactical Emergency Response

DBSCAN①



Abstract

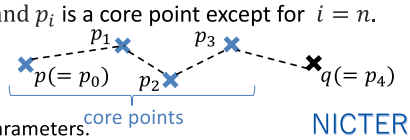
DBSCAN puts together points close to each other (distance $\leq \varepsilon$). It also marks as outliers the points that are in low-density regions.

Core point

A point is a called *core point* if at least minPts points are within distance ε.

Clustering rule (q is reachable from p)

A point q is the same cluster as a core point p if and only if there exists $p_0, p_1, \cdots, p_n$ such that $p_0 = p, p_n = q, d(p_i, p_{i+1}) \leq \varepsilon\ (0 \leq i \leq n-1)$, and p_i is a core point except for $i = n$.



Note : ε and minPts are both DBSCAN parameters.

NICTER
Network Incident analysis Center for Tactical Emergency Response

DBSCAN②

19

Core point

A point is a called *core point* if at least minPts points are within distance ε.

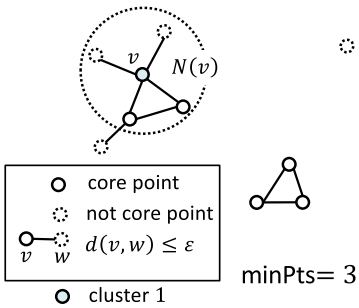
Neighborhood

Neighborhood of v is $N(v)$ is defined as

$$N(v) := \{w \in \mathcal{M} : d(v,w) \le \varepsilon\}$$

DBSCAN Algorithm

1. $j \leftarrow 1$
2. We arbitrarily pick up a core point v not been visited until now.
3. We label v as cluster j.
4. For each $w \in N(v)$. If w is a core point, go to step 3 (we replace v with w in step 3). Otherwise, we label w as j and end this step.
5. If we finish the step 4 for all points, we update $j \leftarrow j + 1$, and we go to step 2.



minPts= 3

Note : ε and minPts are both DBSCAN parameters.

NICTER
Network Incident analysis Center for Tactical Emergency Response

DBSCAN③

20

Core point

A point is a called *core point* if at least minPts points are within distance ε.

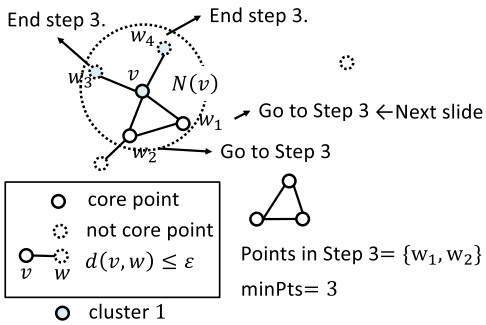
Neighborhood

Neighborhood of v is $N(v)$ is defined as

$$N(v) := \{w \in \mathcal{M} : d(v,w) \le \varepsilon\}$$

DBSCAN Algorithm

1. $j \leftarrow 1$
2. We arbitrarily pick up a core point v not been visited until now.
3. We label v as cluster j.
4. For each $w \in N(v)$, if w is a core point and not been visited, go to step 3 (we replace v with w in step 3). Otherwise, we label w as j and end this step.
5. If we finish the step 4 for all points, we update $j \leftarrow j + 1$, and we go to step 2.



Points in Step 3= $\{w_1, w_2\}$

minPts= 3

Note : ε and minPts are both DBSCAN parameters.

NICTER
Network Incident analysis Center for Tactical Emergency Response

DBSCAN④

Core point

A point is a called *core point* if at least minPts points are within distance ε.

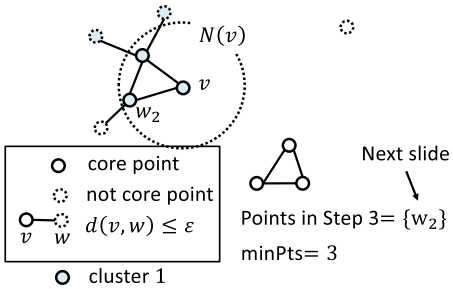
Neighborhood

Neighborhood of v is $N(v)$ is defined as

$N(v) := \{w \in \mathcal{M} : d(v,w) \le \varepsilon\}$

DBSCAN Algorithm

1. $j \leftarrow 1$
2. We arbitrarily pick up a core point v not been visited until now.
3. We label v as cluster j.
4. For each $w \in N(v)$, if w is a core point and not been visited, go to step 3 (we replace v with w in step 3). Otherwise, we label w as j and end this step.
5. If we finish the step 4 for all points, we update $j \leftarrow j + 1$, and we go to step 2.



Note : ε and minPts are both DBSCAN parameters.

NICTER
Network Incident analysis Center for Tactical Emergency Response

DBSCAN⑤

Core point

A point is a called *core point* if at least minPts points are within distance ε.

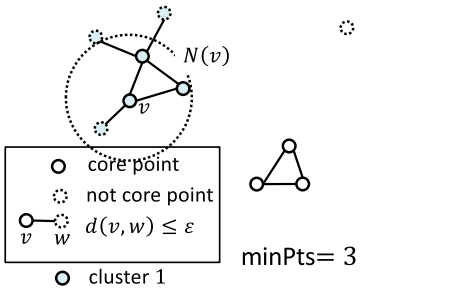
Neighborhood

Neighborhood of v is $N(v)$ is defined as

$N(v) := \{w \in \mathcal{M} : d(v,w) \le \varepsilon\}$

DBSCAN Algorithm

1. $j \leftarrow 1$
2. We arbitrarily pick up a core point v not been visited until now.
3. We label v as cluster j.
4. For each $w \in N(v)$, If w is a core point and not been visited, go to step 3 (we replace v with w in step 3). Otherwise, we label w as j and end this step.
5. If we finish the step 4 for all points, we update $j \leftarrow j + 1$, and we go to step 2.



Note : ε and minPts are both DBSCAN parameters.

NICTER
Network Incident analysis Center for Tactical Emergency Response

DBSCAN⑥

Core point

A point is a called *core point* if at least minPts points are within distance ε.

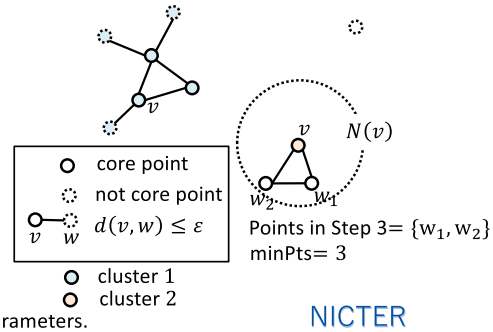
Neighborhood

Neighborhood of v is $N(v)$ is defined as

$$N(v) := \{w \in \mathcal{M} : d(v,w) \le \varepsilon\}$$

DBSCAN Algorithm

1. $j \leftarrow 1$
2. We arbitrarily pick up a core point v not been visited until now.
3. We label v as cluster j.
4. For each $w \in N(v)$, If w is a core point and not been visited, go to step 3 (we replace v with w in step 3). Otherwise, we label w as j and end this step.
5. If we finish the step 4 for all points, we update $j \leftarrow j + 1$, and we go to step 2.



Note : ε and minPts are both DBSCAN parameters.

NICTER
Network Incident analysis Center for Tactical Emergency Response

DBSCAN⑦

Core point

A point is a called *core point* if at least minPts points are within distance ε.

Neighborhood

Neighborhood of v is $N(v)$ is defined as

$$N(v) := \{w \in \mathcal{M} : d(v,w) \le \varepsilon\}$$

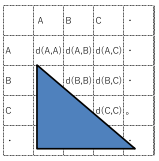
DBSCAN Algorithm

Problem

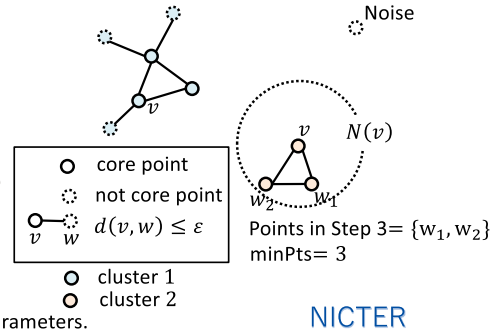
We calculate distances for all pairs of points

Because
- to obtain neighborhood of a point
- to judge whether a point is core point

distance matrix→



We propose an algorithm to reduce calculations of distance.



Note : ε and minPts are both DBSCAN parameters.

NICTER
Network Incident analysis Center for Tactical Emergency Response

Contents

NICT National Institute of Information and Communications Technology

NICTER Network Incident analysis Center for Tactical Emergency Response

Characteristic of our data

The distribution of supports

ratio	#multiset	support
31.7%	102,042	445
11.4%	36,689	23:80:8080
10.3%	33,079	23
3.0%	9,787	5555
2.6%	8,292	50382
2.6%	8,291	50390
2.4%	7,742	22
2.1%	6,622	0.976388889
1.5%	4,750	1433
1.5%	4,712	23:81
1.4%	4,600	23:80:81:1023:2323:5555:7574:8080:8443:37215:49152:52869

11.4% of multisets have the support $\{23,80,8080\}$.

Support

$A = \{1,1,1,2,3,3\} \Rightarrow \mathrm{supp}(A) = \{1,2,3\}$

Several high-density regions exist in support of multisets.

We take advantage of the characteristic for speeding up the DBSCAN.

NICT National Institute of Information and Communications Technology

NICTER Network Incident analysis Center for Tactical Emergency Response

Contents

27

- Background
- Malware/Botnet behavior
- Research direction
- Multiset
- Approach
- Metric on multiset
- DBSCAN
- Characteristic of our data
- → **Fast DBSCAN**
  - **Partition of data**
  - Ball-based DBSCAN
- Experiment

NICTER
Network Incident analysis Center for Tactical Emergency Response

Fast DBSCAN for multiset

28

1. Based on the support of multisets, we partition the data(the set of multisets) such that multisets in different subsets are different clusters.

2. We perform ball-based DBSCAN for each subset.
   1. We make balls from each support.
   2. We connect or divide balls.
   3. We obtain clusters from a ball graph.

NICTER
Network Incident analysis Center for Tactical Emergency Response

Fast DBSCAN for multiset

29

1. Based on the support of multisets, we partition the data(the set of multisets) such that multisets in different subsets are different clusters.

2. We perform ball-based DBSCAN for each subset.
   1. We make balls from each support.
   2. We connect or divide balls.
   3. We obtain clusters from a ball graph.

NICTER
Network Incident analysis Center for Tactical Emergency Response

Partition of data①

30

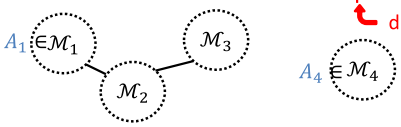
Lemma 1

- Let $\bigcup_{j=1}^{k} \mathcal{M}_j \ (= \mathcal{M})$ be a partition of data (set of multisets).
- We consider an undirected graph $G = (V, E)$ such that
  - $V = \{\mathcal{M}_1, \mathcal{M}_2, \cdots, \mathcal{M}_k\}$
  - $E = \{(\mathcal{M}_j, \mathcal{M}_{j'})$: a lower bound of $d(\mathcal{M}_j, \mathcal{M}_{j'}) \leq \varepsilon\}$

$$d(\mathcal{M}_j, \mathcal{M}_{j'}) := \min_{A_j \in \mathcal{M}_j, A_j \in \mathcal{M}_{j'}} d(A_j, A_{j'})$$

Multisets in different connected components belong to different clusters.

desirable partition



$\Rightarrow$ A_1 and A_4 are different cluster.

We give a partition and roughly estimate the distance between subsets.

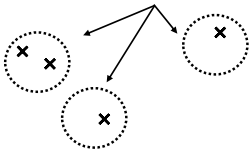
NICTER
Network Incident analysis Center for Tactical Emergency Response

Partition of data②

1. Based on the support of multisets, we partition the data(the set of multisets) such that multisets in **different subsets are different clusters.**

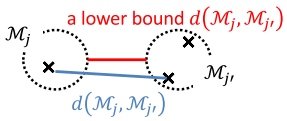
Step 1

We give a partition.



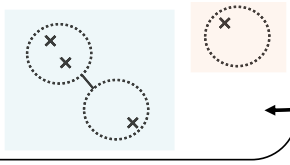
Step 2

We create a lower bound of distance between subsets.



Step 3

We make a graph and obtain connected components.



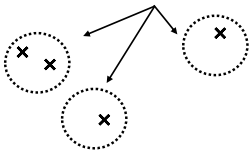
NICTER
Network Incident analysis Center for Tactical Emergency Response

Partition of data③

1. Based on the support of multisets, we partition the data(the set of multisets) such that multisets in **different subsets are different clusters.**

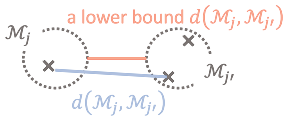
Step 1

We give a partition.



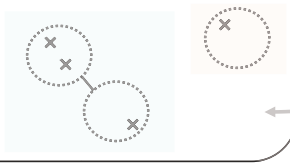
Step 2

We create a lower bound of distance between subsets.



Step 3

We make a graph and obtain connected components.



NICTER
Network Incident analysis Center for Tactical Emergency Response

Partition of data②

- Partition
  - We define the following equivalent relation $\sim$ on multisets $\mathcal{M}$
    $A \sim B \Leftrightarrow \mathrm{supp}(A) = \mathrm{supp}(B)$
  - The quotient set $\mathcal{M}/\sim$ is a partition of $\mathcal{M}$

$A_1 = \{1,2,3,3\}$
$A_2 = \{1,1,1,2,2,3\}$

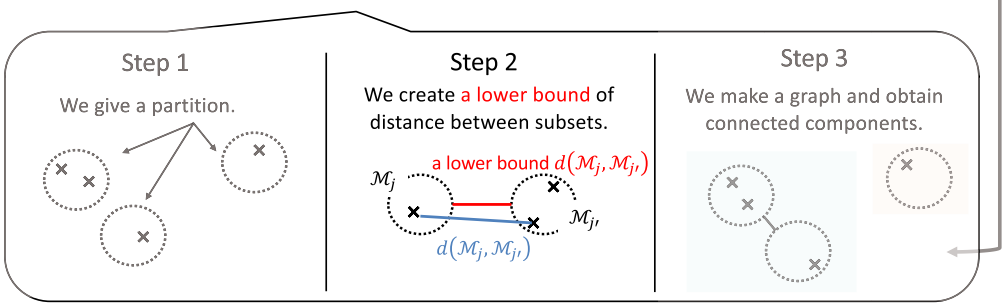
$B_1 = \{1,1,2\}$
$B_2 = \{1,1,1,2,2\}$
$B_3 = \{1,2\}$

The equivalent class of A_1 is $[A_1] := \{A \in \mathcal{M} : \mathrm{supp}(A) = \mathrm{supp}(A_1) = \{1,2,3\} = \{A_1, A_2\}$
The quotient set $\mathcal{M}/\sim = \{[A_1], [B_1]\}$

NICTER
Network Incident analysis Center for Tactical Emergency Response

Partition of data③

1. Based on the support of multisets, we partition the data(the set of multisets) such that multisets in **different subsets are different clusters.**



NICTER
Network Incident analysis Center for Tactical Emergency Response

Partition of data③

The lower bounds bounds of the distance between equivalent class $[A]$ and $[B]$

Lemma 2

$\widetilde{d_2}([A],[B]) \leq \widetilde{d_1}([A],[B]) \leq d(A,B)$ for any $A \in [A], B \in [B]$

$A = \{2, 1, 1, 1, \quad 4\}, B = \{2, \quad 11, 3, 3\}$

Distance or upper bounds of the distance	what compares	value
$\widetilde{d_2}([A],[B]) := 1 - \frac{\min(\lvert\mathrm{Supp}(A)\rvert, \lvert\mathrm{Supp}(B)\rvert)}{\max(\lvert\mathrm{Supp}(A)\rvert, \lvert\mathrm{Supp}(B)\rvert)}$	cardinality of the support	0
$\widetilde{d_1}([A],[B]) := \frac{\lvert\mathrm{Supp}(A)\Delta\mathrm{Supp}(B)\rvert}{\lvert\mathrm{Supp}(A)\cup\mathrm{Supp}(B)\rvert}$	support	$\frac{2}{4} = \frac{1}{2}$
$d(A,B) := \frac{1}{\lvert\mathrm{Supp}(A\cup B)\rvert} \sum_{x\in\mathrm{Supp}(A\cup B)} \frac{m_{A\Delta B}(x)}{m_{A\cup B}(x)}$	multisets	$\frac{7}{12}$

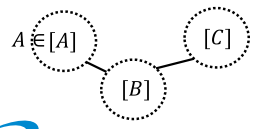
NICTER
Network Incident analysis Center for Tactical Emergency Response

Partition of data⑤

Procedure

1. We makes the quotient set $\mathcal{M}/\sim$ with the equivalent relation
$$A \sim B \Leftrightarrow \mathrm{supp}(A) = \mathrm{supp}(B)$$
2. We makes an undirected graph $G = (V, E)$ such that
$V = \mathcal{M}/\sim$ and $E = \{([A],[B]) : \widetilde{d_2}([A],[B])\} \leq \varepsilon \wedge \widetilde{d_1}([A],[B])\} \leq \varepsilon\}$

Multiset in different connected components belong to different clusters.



$D \in [D]$ $\Rightarrow$ A and D are different clusters.

NICTER
Network Incident analysis Center for Tactical Emergency Response

Contents

37

- Background
- Malware/Botnet behavior
- Research direction
- Multiset
- Approach
- Metric on multiset
- DBSCAN
- Characteristic of our data
- Fast DBSCAN
  - Partition of data
  - → **Ball-based DBSCAN**
- Experiment

NICTER
Network Incident analysis Center for Tactical Emergency Response

Fast DBSCAN for multiset

38

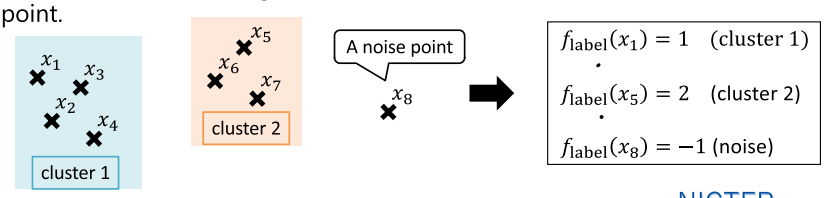
1. Based on the support of multisets, we partition the data(the set of multisets) such that multisets in different subsets are different clusters.

2. We perform ball-based DBSCAN for each subset.
   1. We make balls from each support.
   2. We connect or divide balls.
   3. We obtain clusters from a ball graph.

NICTER
Network Incident analysis Center for Tactical Emergency Response

Ball-based DBSCAN①

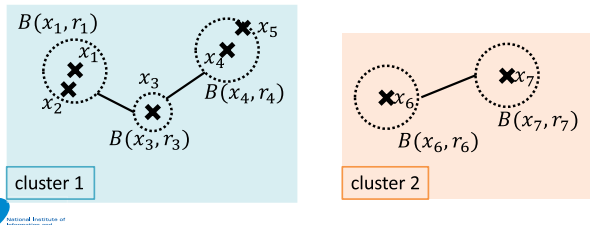
- Input
  - $\mathcal{M}$: a finite number of multisets. We admit one multiset appears several times in $\mathcal{M}$.
  - $\varepsilon\ (>0)$: a radius parameter of DBSCAN
  - minPts : a density threshold of DBSCAN
  - $r_{\mathrm{init}}(\leq \varepsilon/2)$: an upper bound of initial radius
  - $\alpha\ (0<\alpha<1)$: an attenuation rate
- Output
  - $f_{\mathrm{label}} : \mathcal{M} \to \mathbb{N} \cup \{-1\}$, the clustering result of DBSCAN. $f_{\mathrm{label}}(x) = k$ means that a multiset x belongs to cluster k if $k \neq -1$; otherwise ($k = -1$), x is a noise point.



NICTER
Network Incident analysis Center for Tactical Emergency Response

Ball-based DBSCAN②

1. We make a disjoint closed-ball cover of $\mathcal{M}$ whose radius is less than or equal to r_0.
   $\mathcal{M} = \bigcup_{j=1}^{m} B(x_j, r_j),\ r_j \leq r_0, B(x_j, r_j) \cap B(x_k, r_k) = \emptyset,$
   $B(x_j, r_j) = \{x \in \mathcal{M} : d(x, x_j) \leq r_0\}$
2. We build a graph by connecting or dividing the closed balls.
3. We obtain connected components of the graph; each component corresponds to a cluster.

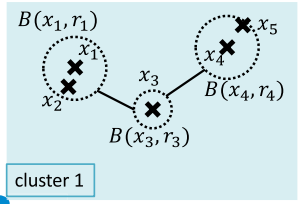


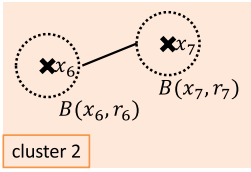
cluster1 : $x_1, x_2, \cdots, x_5$
cluster2 : x_6, x_7

NICTER
Network Incident analysis Center for Tactical Emergency Response

Ball-based DBSCAN②

1. We make a disjoint closed-ball cover of $\mathcal{M}$ whose radius is less than or equal to r_0.
   $\mathcal{M} = \bigcup_{j=1}^{m} B(x_j, r_j)$, $r_j \le r_0$, $B(x_j, r_j) \cap B(x_k, r_k) = \emptyset$,
   $B(x_j, r_j) = \{x \in \mathcal{M} : d(x, x_j) \le r_0\}$
2. We build a graph by connecting or dividing the closed balls.
3. We obtain connected components of the graph; each component corresponds to a cluster.





cluster1 : $x_1, x_2, \cdots, x_5$
cluster2 : x_6, x_7

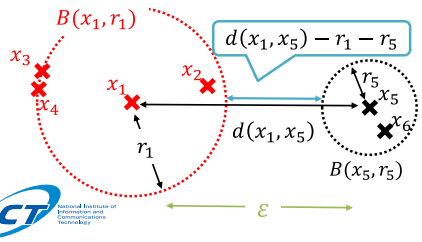
NICTER
Network Incident analysis Center for Tactical Emergency Response

Build graph①

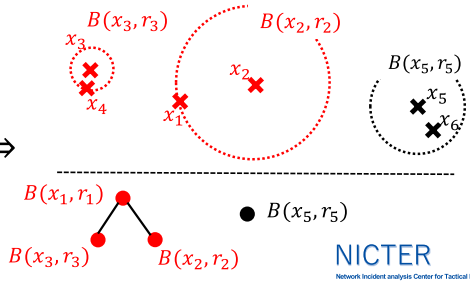
For any pair of balls $B(x_j, r_j)$, $B(x_{j'}, r_{j'})$ (suppose $r_j \ge r_{j'}$)

- $d(x_j, x_{j'}) - r_j - r_{j'} > \varepsilon$ ⇒ do nothing
- $d(x_j, x_{j'}) \le \varepsilon$ ⇒ connect $B(x_j, r_j)$ to $B(x_{j'}, r_{j'})$
- Otherwise
  ⇒ Divide the ball $B(x_j, r_j) = \bigcup_{x,r} B(x, r)$ where $r \le \alpha r_j$
  For each $B(x, r)$, we connect $B(x, r)$ to $B(x_j, r_j)$ and do this step for $B(x, r)$ and $B(x_j, r_j)$.

the slide after next slide
next slide
- $B(x_3, r_3)$ and $B(x_5, r_5)$
- $B(x_2, r_2)$ and $B(x_5, r_5)$



⇒



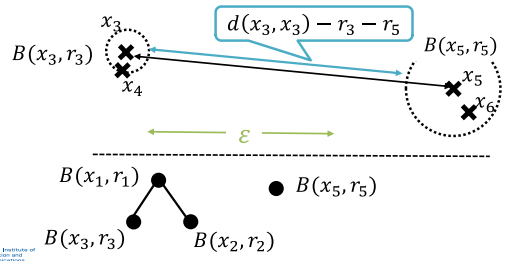
NICTER
Network Incident analysis Center for Tactical Emergency Response

Build graph②

For any pair of balls $B(x_j, r_j)$, $B(x_{j'}, r_{j'})$ (suppose $r_j \geq r_{j'}$)

- $d(x_j, x_{j'}) - r_j - r_{j'} > \varepsilon$ ⇒ do nothing
- $d(x_j, x_{j'}) \leq \varepsilon$ ⇒ connect $B(x_j, r_j)$ to $B(x_{j'}, r_{j'})$
- Otherw
  ⇒ Divid
  For each $B(x,r)$, we connect $B(x,r)$ to $B(x_j, r_j)$ and do this step for $B(x,r)$ and $B(x_j, r_j)$.

$$d\left(B(x_j, r_j), B(x_{j'}, r_{j'})\right) = \min_{\substack{x_j \in B(x_j, r_j) \\ x_{j'} \in B(x_{j'}, r_{j'})}} d(x_j, x_{j'}) > \varepsilon \text{ satisfies.}$$

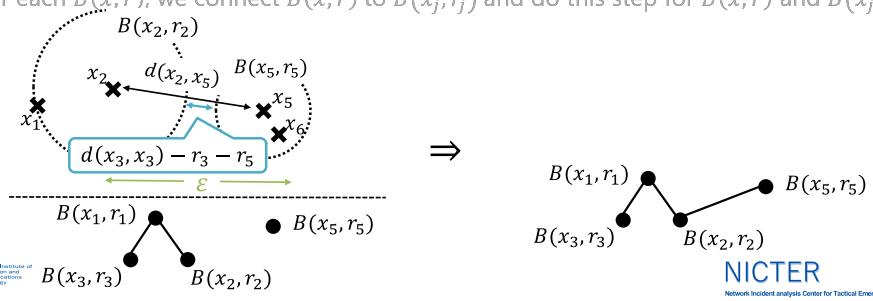


NICTER
Network Incident analysis Center for Tactical Emergency Response

Build graph③

For any pair of balls $B(x_j, r_j)$, $B(x_{j'}, r_{j'})$ (suppose $r_j \geq r_{j'}$)

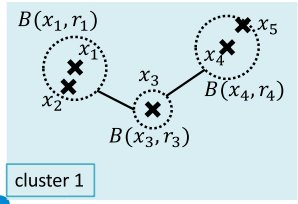
- $d(x_j, x_{j'}) - r_j - r_{j'} > \varepsilon$ ⇒ do nothing
- $d(x_j, x_{j'}) \leq \varepsilon$ ⇒ connect $B(x_j, r_j)$ to $B(x_{j'}, r_{j'})$
- Otherwise
  ⇒ Divide the ball $B(x_j, r_j) = \cup_{x,r} B(x,r)$ where $r \leq \alpha r_j$
  For each $B(x,r)$, we connect $B(x,r)$ to $B(x_j, r_j)$ and do this step for $B(x,r)$ and $B(x_j, r_j)$.

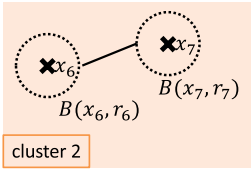


NICTER
Network Incident analysis Center for Tactical Emergency Response

Ball-based DBSCAN③

1. We make a disjoint closed-ball cover of $\mathcal{M}$ whose radius is less than or equal to r_0.
   $\mathcal{M} = \bigcup_{j=1}^{m} B(x_j, r_j)$, $r_j \leq r_0, B(x_j, r_j) \cap B(x_k, r_k) = \emptyset$,
   $B(x_j, r_j) = \{x \in \mathcal{M} : d(x, x_j) \leq r_0\}$
2. We build a graph by connecting or dividing the closed balls.
3. We obtain connected components of the graph; each component corresponds to a cluster.



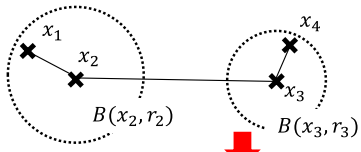


cluster1 : $x_1, x_2, \cdots, x_5$
cluster2 : x_6, x_7

NICTER
Network Incident analysis Center for Tactical Emergency Response

Obtain clusters

- For simplicity of explanation, we suppose all multisets are core points.
- Finally, we obtain the undirected graph $G = (V, E)$.
  - V = the set of closed balls $B(x_\lambda, r_\lambda)$ $\left(r_\lambda \leq \frac{\varepsilon}{2}\right)$ with $\bigcup_\lambda B(x_\lambda, r_\lambda)$
    $\Rightarrow$ Multisets in a closed ball belongs to the same cluster because distance $\leq \varepsilon$
  - An edge $(B(x_\lambda, r_\lambda), B(x_{\lambda'}, r_{\lambda'}))$ exists if $d(x_\lambda, x_{\lambda'}) \leq \varepsilon$
    $\Rightarrow B(x_\lambda, r_\lambda)$ and $B(x_{\lambda'}, r_{\lambda'})$ are the same cluster
    because any element in $B(x_\lambda, r_\lambda)$ is reachable from any element in $B(x_{\lambda'}, r_{\lambda'})$.



x_4 is reachable from x_1
because there exists the path
$\{(x_1, x_2), (x_2, x_3), (x_3, x_4)\}$ and $d(x_i, x_{i+1}) \leq \varepsilon$

Each connected component of $G = (V, E)$ corresponds to one cluster.

NICTER
Network Incident analysis Center for Tactical Emergency Response

Contents

NICTER
Network Incident analysis Center for Tactical Emergency Response

Experimental settings

- Our darknet received packets from 487,761 IP addresses (ip.src) on September 1st.
- We removed noise ip.src (#packet $\leq$ 6) and then obtained 321,435 ip.srcs.
- For each ip.src, we created a multiset whose support is destination ports and whose multiplicity is the number of packets.
- We conducted the fast DBSCAN and calculated the computation cost.
- We used the Julia language.

#darknet's IP addresses	#ip.src (multisets)	day
298,280	321,435	September 1st 2021

NICTER
Network Incident analysis Center for Tactical Emergency Response

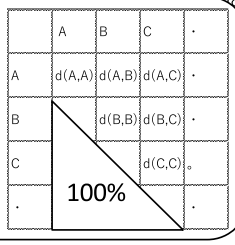
Computation Cost①

$$\#\text{total } d = \#d + \alpha \#\widetilde{d_1}$$

where α is the computation cost ratio of $\widetilde{d_1}$ to d.

Algorithm	eps	minPts	$\#\tilde{d}_1$	$\#d$	$\#d$(%)	#total d	#total d(%)
fast DBSCAN	0.15	10	9,709,900	110,539,156	0.2	128,648,994	**0.25%**
	0.15	20	9,70	[illegible]		[illegible]088	**0.25%**
	0.25	10	17	[illegible]		[illegible]	**0.37%**
	0.25	20	17	[illegible]		[illegible]	**0.36%**
	0.35	10	23	[illegible]		[illegible]	**0.41%**
	0.35	20	23	[illegible]		[illegible]	**0.42%**
Original DBSCAN	-	-	[illegible]	[illegible]		[illegible]	100.00%

Calculation percentage to distance matrix

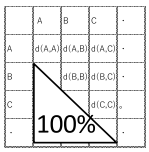


NICTER
Network Incident analysis Center for Tactical Emergency Response

Computation Cost②

Algorithm	eps	minPts	$\#\tilde{d}_1$	$\#d$	$\#d$(%)	#total d	#total d(%)
fast DBSCAN	0.15	10	9,70	[illegible]		[illegible]	**0.25%**
	0.15	20	9,70	[illegible]		[illegible]	**0.25%**
	0.25	10	17,30	[illegible]		[illegible]	**0.37%**
	0.25	20	17,30	[illegible]		[illegible]	**0.36%**
	0.35	10	23,49	[illegible]		[illegible]	**0.41%**
	0.35	20	23,498,478	175,406,957	0.34%	[illegible]9,233,735	**0.42%**
Original DBSCAN	-	-	-	51,660,068,895	100.00%	51,660,068,895	100.00%

Calculation percentage to distance matrix



NICTER
Network Incident analysis Center for Tactical Emergency Response

Computation Cost④

- The fast DBSCAN calculated less than 0.5% of distances.
- Calculation cost was not much affected by the DBSCAN's parameters.

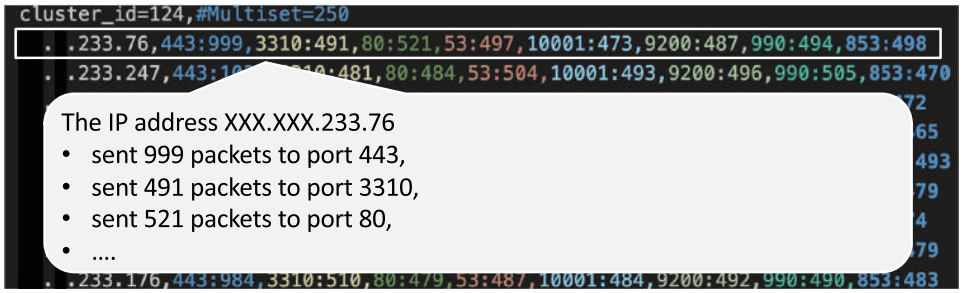
Algorithm	eps	minPts	$\#\tilde{d}_1$	$\#d$	$\#d$(%)	#total d	#total d(%)
fast DBSCAN	0.15	10	9,709,900	110,539,156	0.21%	128,648,994	**0.25%**
	0.15	20	9,709,900	110,996,250	0.21%	129,106,088	**0.25%**
	0.25	10	17,309,806	157,003,986	0.30%	189,288,333	**0.37%**
	0.25	20	17,309,806	155,472,637	0.30%	187,756,984	**0.36%**
	0.35	10	23,498,478	168,235,838	0.33%	212,062,616	**0.41%**
	0.35	20	23,498,478	175,406,957	0.34%	219,233,735	**0.42%**
Original DBSCAN	-	-	-	51,660,068,895	100.00%	51,660,068,895	100.00%

Note: We do not calculate fast DBSCAN in parallel after a partition of data

NICTER
Network Incident analysis Center for Tactical Emergency Response

Multi-target cluster①

The below cluster targeted 8 destination ports.



※eps=0.15, minPts=10

NICTER
Network Incident analysis Center for Tactical Emergency Response

Multi-target cluster②

The below cluster targeted 8 destination ports.

```
cluster_id=124,#Multiset=250
  . .233.76,443:999,3310:491,80:521,53:497,10001:473,9200:487,990:494,853:498
  . .233.247,443:1022,3310:481,80:484,53:504,10001:493,9200:496,990:505,853:470
  . .233.60,443:979,3310:492,80:488,53:485,10001:493,9200:485,990:503,853:472
  . .233.140,443:952,3310:477,80:497,53:519,10001:482,9200:517,990:488,853:465
  . .233.226,443:1019,3310:481,80:510,53:510,10001:495,9200:483,990:479,853:493
  . .233.206,443:959,3310:519,80:515,53:500,10001:511,9200:529,990:479,853:479
  . .233.23,443:973,3310:468,80:520,53:448,10001:470,9200:480,990:478,853:474
  . .233.179,443:994,3310:511,80:485,53:507,10001:530,9200:533,990:486,853:479
  . .233.176,443:984,3310:510,80:479,53:487,10001:484,9200:492,990:490,853:483
```

※eps=0.15, minPts=10

The multiplicity of each element is similar to each other.

NICTER
Network Incident analysis Center for Tactical Emergency Response

Contribution

- We propose a metric space on a multiset and prove that the metric satisfies well-known properties. (positivity, symmetry, and triangle inequality)

Triangle inequality $d(x,z) \leq d(x,y) + d(y,z)$

- We propose the fast DBSCAN for the metric space.
  - The output of our algorithm is the same as the original one.
  - Our algorithm reduces 95.5% computation cost compared to the original DBSCAN in our experiments.

NICTER
Network Incident analysis Center for Tactical Emergency Response

55

Acknowledgement

This research was conducted under a contract of “Research and development on IoT malware removal / make it non-functional technologies for effective use of the radio spectrum” among “Research and Development for Expansion of Radio Wave Resources (JPJ000254)”, which was supported by the Ministry of Internal Affairs and Communications, Japan.

NICT National Institute of Information and Communications Technology

NICTER
Network Incident analysis Center for Tactical Emergency Response

Finding Densest *k*-Connected Subgraphs

Atsushi MIYAUCHI

Graduate School of Information Science and Technology,
University of Tokyo, Japan
miyauchi@mist.i.u-tokyo.ac.jp

Dense subgraph discovery is an important graph-mining primitive with a variety of real-world applications. One of the most well-studied optimization problems for dense subgraph discovery is the densest subgraph problem, where given an edge-weighted undirected graph, we are asked to find a subgraph that maximizes the average degree. Although this problem can be solved exactly in polynomial time and well-approximately in almost linear time, a densest subgraph has a structural drawback, namely, the subgraph may be disconnected by removing only a few vertices/edges within it. In this talk, we propose an algorithmic framework to find a dense subgraph that is well-connected in terms of vertex/edge connectivity. This talk is based on joint work [1] with Francesco Bonchi (CENTAI), David García-Soriano (ISI Foundation), and Charalampos E. Tsourakakis (Boston University).

References

[1] Francesco Bonchi, David García-Soriano, Atsushi Miyauchi, and Charalampos E. Tsourakakis, "Finding Densest *k*-Connected Subgraphs," *Discrete Applied Mathematics* **305**, pp. 34–47, 2021, https://doi.org/10.1016/j.dam.2021.08.032.

Finding Densest k-Connected Subgraphs

Discrete Applied Mathematics **305**, pp. 34–47, 2021

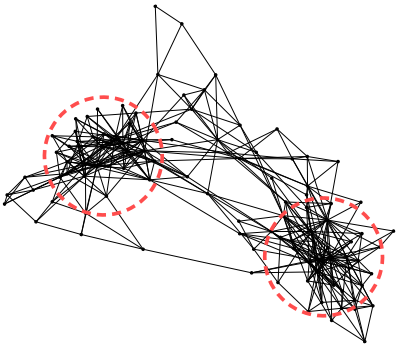
Francesco Bonchi[1]
David García-Soriano[2]
Atsushi Miyauchi[3]
Charalampos E. Tsourakakis[2,4]

[1]CENTAI [2]ISI Foundation [3]University of Tokyo [4]Boston University

Sep. 19, 2022

Introduction

Dense subgraph discovery



Dense subgraphs in real-world:

- social groups
  in friendship networks
- communities & spam link farms
  in Web graphs
- molecular complexes
  in protein interaction networks

Dense subgraph discovery is a fundamental task in graph mining

How to detect dense subgraphs

The most common way is to utilize **optimization theory**:

Step 1: Introduce a **quality function**

Step 2: Define an optimization model (with or without **constraints**)

Step 3: Solve the model exactly or approximately

Many optimization models and algorithms have been developed

Densest subgraph problem

Let $G = (V, E, w)$ be an edge-weighted graph ($w : E \to \mathbb{Q}_{>0}$)

Problem (Densest subgraph)

Input: $G = (V, E, w)$

Output: $S \subseteq V$ that maximizes $d(S) := \dfrac{w(S)}{|S|}$ (**density**)

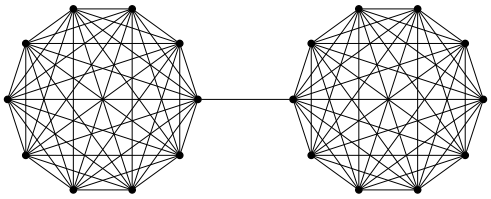
$w(S)$: sum of edge weights in the induced subgraph $G[S]$

- **Polynomial-time exact algorithms**
  - LP-based algorithm [Charikar '00]
  - flow-based algorithm [Goldberg '84]
- **Almost-linear-time 1/2-approximation algorithm** [Charikar '00]

An optimal solution is referred to as a **densest subgraph**

Drawback of densest subgraphs

Densest subgraphs are **not necessarily well-connected**



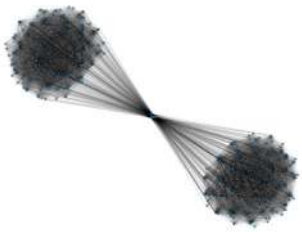
Barbell graph

- The entire graph is the densest subgraph
- Removing only one edge or two vertices separates it

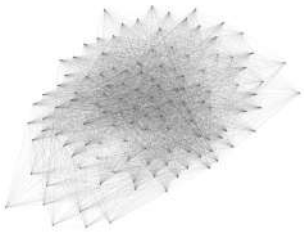
Densest subgraphs may **not be robust** to vertex/edge failure

Not only in theory...

Densest subgraphs in real-world Web graphs:

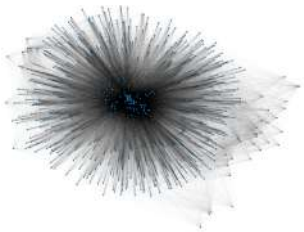


web-BerkStan



web-Google

web-NotreDame



web-Stanford

Terminology

Let $G = (V, E)$ be a graph

- $S \subset V$ is called a **vertex separator** if its removal divides G into at least two nonempty subgraphs between which there are no edges
- The **vertex connectivity** $\kappa(G)$ is the smallest cardinality of a vertex separator of G if G is not a clique and $|V| - 1$ otherwise
- G is said to be ***k*-vertex-connected** if $\kappa(G) \geq k$

Table: Statistics of the four densest subgraphs $S^{\mathrm{DS}} \subseteq V$

Graph	$\lvert S^{\mathrm{DS}}\rvert$	$\lvert E(S^{\mathrm{DS}})\rvert$	$d(S)$	$\kappa(G[S^{\mathrm{DS}}])$	min-deg$(G[S^{\mathrm{DS}}])$
web-BerkStan	392	40,535	103.41	**12**	201
web-Google	123	3,449	28.04	30	30
web-NotreDame	1,367	107,526	78.66	**1**	155
web-Stanford	597	35,456	59.39	60	60

Note: $\kappa(G[S^{\mathrm{DS}}]) \leq$ min-deg$(G[S^{\mathrm{DS}}])$

Our contribution

Problem (Densest k-vertex-connected subgraph)

Input: $G = (V, E, w)$ and $k \in \mathbb{Z}_{>0}$

Output: $S \subseteq V$ that maximizes $d(S) = \dfrac{w(S)}{|S|}$ under $\kappa(G[S]) \geq k$

- Generalization of **Mader's theorem**
- Algorithm for finding a Mader subgraph
- Bicriteria approximation algorithm
- Approximation algorithm

The edge-connectivity counterparts are obtained but omitted

Mader's theorem & Mader subgraph

Mader's theorem

Theorem [Mader '72]

Let $G = (V, E)$ be a graph and let $d \in \mathbb{Z}_{>0}$

If $d(V) \geq d$, then G has a $(\lfloor d/2 \rfloor + 1)$-vertex-connected subgraph whose minimum degree of vertices is greater than d

This theorem says that

- dense graphs contain well-connected subgraphs
- (although dense graphs are not necessarily well-connected)

Generalization to edge-weighted graphs

Theorem [This work]

Let $G = (V, E, w)$ be an edge-weighted graph and let $d \in \mathbb{R}_{>0}$

If $d(V) \geq d$, then G has a $(\lfloor \lceil d/w_{\max} \rceil / 2 \rfloor + 1)$-vertex-connected subgraph whose minimum weighted degree of vertices is greater than d

Proof strategy is the same as that of the original theorem

A subgraph whose existence is guaranteed by our theorem is referred to as a **Mader subgraph**

Algorithm for finding Mader subgraph

Let G^* be a Mader subgraph of $G = (V, E, w)$

Strategy

- Our algorithm keeps a family $\mathcal{H}$ of subgraphs of G
- Exactly one subgraph in $\mathcal{H}$ contains G^* as its subgraph
- In each iteration, our algorithm tests whether a subgraph in $\mathcal{H}$ is a Mader subgraph or not

  YES: Output the subgraph
  NO: Divide the subgraph into strictly smaller pieces

This is based on the algorithm for finding the most highly connected subgraph in terms of vertex connectivity [Matula '78]

Algorithm for finding Mader subgraph

An important subprocedure:

`Peel`(G, d)

Input: $G = (V, E, w)$ and $d \in \mathbb{R}_{>0}$
Output: $G[S]$ or `Null`

$S \leftarrow V$;
while $S \neq \emptyset$
  $v_{\min} \leftarrow \operatorname{argmin}_{v \in S} \deg_S(v)$;
    // $\deg_S(v)$ is the weighted degree of v in $G[S]$
  if $\deg_S(v_{\min}) > d$ **then return** $G[S]$;
  $S \leftarrow S \setminus \{v_{\min}\}$;
return `Null`;

- Often used for dense subgraph discovery (e.g., [Charikar '00])
- This algorithm runs in $O(|E| + |V| \log |V|)$ time

Algorithm for finding Mader subgraph

`Mader_subgraph`(G)

Input: $G = (V, E, w)$
Output: $G[S]$

$H \leftarrow$ `Peel`$(G, d(V))$;
$\tau \leftarrow \left\lfloor \frac{\lceil d(V)/w_{\max} \rceil}{2} \right\rfloor + 1$; // vertex connectivity guaranteed by our theorem
$\mathcal{H} \leftarrow$ family of the connected components of H that have at least $\tau + 1$ vertices;
if there exists a clique K in $\mathcal{H}$ **then return** K;

Algorithm for finding Mader subgraph

`Mader_subgraph`(G)

Input: $G = (V, E, w)$
Output: $G[S]$

$H \leftarrow$ `Peel`$(G, d(V))$;
$\tau \leftarrow \left\lfloor \frac{\lceil d(V)/w_{\max} \rceil}{2} \right\rfloor + 1$; // vertex connectivity guaranteed by our theorem
$\mathcal{H} \leftarrow$ family of the connected components of H that have at least $\tau + 1$ vertices;
if there exists a clique K in $\mathcal{H}$ **then return** K;
while `True`
  $H' \leftarrow$ an arbitrary element of $\mathcal{H}$;
  $C \leftarrow$ the minimum vertex separator of H';
  if $|C| \geq \tau$ **then return** H';
  $\mathcal{S} \leftarrow$ family of the connected components of $G[V(H') \setminus C]$;
    // $V(H')$ denotes the vertex set of H'
  $\mathcal{H}' \leftarrow \emptyset$;
  for each $S \in \mathcal{S}$
    if $T :=$ `Peel`$(G[S \cup C], d(V))$ has at least $\tau + 1$ vertices **then** $\mathcal{H}' \leftarrow \mathcal{H}' \cup \{T\}$;
  if there exists a clique K in $\mathcal{H}'$ **then return** K;
  $\mathcal{H} \leftarrow (\mathcal{H} \setminus \{H'\}) \cup \mathcal{H}'$;

Analysis

Theorem

`Mader_subgraph`(G) outputs a Mader subgraph of G in poly time

Proof (sketch):

- It suffices to show the while-loop terminates in polynomial time
- The time complexity of each iteration is dominated by computing the minimum vertex separator (i.e., polynomial)
- **The number of iterations is bounded by $|V|$**

Note: The actual time complexity is $O(|V|^{19/4})$

Bicriteria approximation algorithm

Algorithm (with parameter $\gamma \in [1, 2]$)

Input: $G = (V, E, w)$ and $k \in \mathbb{Z}_{>0}$
Output: $S \subseteq V$ or `INFEASIBLE`
Find the family of maximal k-vertex-connected subgraphs $\{G[S_1], \ldots, G[S_p]\}$; // Use the algorithm by [Makino '88]
if there is no k-vertex-connected subgraph found **then**
  return `INFEASIBLE`;
for $i = 1, \ldots, p$
  $S_i^* \leftarrow S_i$;
  Find a densest subgraph S_i^{DS} (without any constraint) in $G[S_i]$;
  if $k \leq \gamma \left(\left\lfloor \frac{\lceil d(S_i^{\mathrm{DS}})/w_{\max} \rceil}{2} \right\rfloor + 1 \right)$ **then**
    $S_i^* \leftarrow$ The vertex set of `Mader_subgraph`$(G[S_i^{\mathrm{DS}}])$;
return $S \in \operatorname{argmax}_{S \in \{S_1^*, \ldots, S_p^*\}} d(S)$;

Analysis

Theorem

Our algorithm is a polynomial-time $\left(\frac{\gamma}{4} \cdot \frac{w_{\min}}{w_{\max}}, 1/\gamma\right)$-bicriteria approximation algorithm ($\forall \gamma \in [1, 2]$)

Let $S \subseteq V$ be the output

- $d(S) \geq \frac{\gamma}{4} \cdot \frac{w_{\min}}{w_{\max}} \cdot \mathrm{OPT}$ (OPT: optimal value of the original problem)
- $G[S]$ is (k/γ)-vertex-connected

Note:

- We can get $\left(\frac{1}{4} \cdot \frac{w_{\min}}{w_{\max}}\right)$-approximation by setting $\gamma = 1$
- The time complexity is $O(|V|(|V|^{19/4} + T_{\mathrm{DS}}))$
  (T_{DS}: time complexity of computing a densest subgraph in G)

Analysis

Theorem

Our algorithm is a polynomial-time $\left(\frac{\gamma}{4} \cdot \frac{w_{\min}}{w_{\max}}, 1/\gamma\right)$-bicriteria approximation algorithm ($\forall \gamma \in [1, 2]$)

Proof (sketch): Let $S \subseteq V$ be the output

- **$G[S]$ is (k/γ)-vertex-connected**
  - It suffices to show $G[S_i^*]$ is (k/γ)-vertex-connected ($\forall i = 1, \ldots, p$)
  - If $k \leq \gamma \left(\left\lfloor \frac{\lceil d(S_i^{\mathrm{DS}})/w_{\max} \rceil}{2} \right\rfloor + 1 \right)$ does not hold, S_i^* is given by S_i; so OK
  - Otherwise S_i^* is the vertex set of `Mader_subgraph`$(G[S_i^{\mathrm{DS}}])$
  - Apply the generalized Mader's theorem:
    S_i^* is $\left(\left\lfloor \frac{\lceil d(S_i^{\mathrm{DS}})/w_{\max} \rceil}{2} \right\rfloor + 1 \right)$-vertex-connected; so $\frac{k}{\gamma}$-vertex-connected

Analysis

Theorem

Our algorithm is a polynomial-time $\left(\frac{\gamma}{4} \cdot \frac{w_{\min}}{w_{\max}}, 1/\gamma\right)$-bicriteria approximation algorithm ($\forall \gamma \in [1, 2]$)

- $d(S) \geq \frac{\gamma}{4} \cdot \frac{w_{\min}}{w_{\max}} \cdot \mathrm{OPT}$
  - Let OPT_i be the optimal value of the original problem on $G[S_i]$
  - It suffices to show $d(S_i^*) \geq \frac{\gamma}{4} \cdot \frac{w_{\min}}{w_{\max}} \cdot \mathrm{OPT}_i$
  - If $k \leq \gamma \left(\left\lfloor \frac{\lceil d(S_i^{\mathrm{DS}})/w_{\max} \rceil}{2} \right\rfloor + 1 \right)$ does not hold, S_i^* is given by S_i;
    so every vertex in $G[S_i^*]$ has weighted degree of at least
    $w_{\min} k > \cdots > \frac{\gamma}{2} \cdot \frac{w_{\min}}{w_{\max}} \cdot \mathrm{OPT}_i$, implying $d(G[S_i^*]) \geq \frac{\gamma}{4} \cdot \frac{w_{\min}}{w_{\max}} \cdot \mathrm{OPT}_i$
  - Otherwise S_i^* is the vertex set of `Mader_subgraph`$(G[S_i^{\mathrm{DS}}])$
  - Apply the generalized Mader's theorem:
    S_i^* has the minimum weighted degree of at least $d(S_i^{\mathrm{DS}})$;
    so $d(S_i^*) \geq d(S_i^{\mathrm{DS}})/2 \geq \mathrm{OPT}_i/2$ (irrespective of edge weights)

Approximation algorithm

Algorithm

Input: $G = (V, E, w)$ and $k \in \mathbb{Z}_{>0}$
Output: $S \subseteq V$ or `INFEASIBLE`
$S \leftarrow \operatorname{argmax}\{\kappa(G[S]) \mid S \subseteq V\}$; // Use the algorithm by [Matula '78]
if $\kappa(G[S]) \geq k$ **then return** S;
return `INFEASIBLE`;

This runs in $O(|V|^2(\kappa(G)^2 \cdot \min\{|V|^{3/4}, \kappa(G)^{3/2}\} + \kappa(G)|V|))$ time

Analysis

Theorem

Our algorithm is a polynomial-time $\left(\frac{6}{19} \cdot \frac{w_{\min}}{w_{\max}}\right)$-approximation algorithm

Note: This is better than the previous $\left(\frac{1}{4} \cdot \frac{w_{\min}}{w_{\max}}\right)$-approximation

Proof (sketch):

- **Theorem by [Bernshteyn & Kostochka '16]:**
  Let $G = (V, E)$ be a graph and $t \in \mathbb{Z}$ with $t \geq 2$
  If G satisfies $|V| \geq \frac{5}{2}t$ and $|E| > \frac{19}{12}t(|V| - t)$, then G has a $(t+1)$-vertex-connected subgraph
- Use the theorem as in the analysis of the bicriteria approximation

Conclusion

Summary

Problem (Densest k-vertex-connected subgraph)

Input: $G = (V, E, w)$

Output: $S \subseteq V$ that maximizes $d(S)$ under $\kappa(G[S]) \geq k$

- Generalization of **Mader's theorem**
- Algorithm for finding a Mader subgraph
- $\left(\frac{\gamma}{4} \cdot \frac{w_{\min}}{w_{\max}}, 1/\gamma\right)$-bicriteria approximation algorithm ($\gamma \in [1, 2]$)
- $\left(\frac{6}{19} \cdot \frac{w_{\min}}{w_{\max}}\right)$-approximation algorithm

The edge-connectivity counterparts are obtained but omitted

Future work

- Design better (bicriteria or ordinary) approximation algorithms
- Conduct experiments to investigate practical performance
- Analyze the computational complexity (NP-hardness etc.)

Thank you!

Worst-case constructions for linear optimization

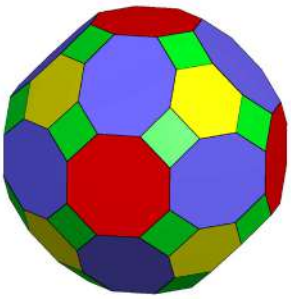
Antoine DEZA
McMaster University, Canada
deza@mcmaster.ca

Worst-case constructions have helped providing a deeper understanding of how the structural properties of the input affect the computational performance of linear optimization. Recent examples include the construction of Allamigeon et al. for which the interior point method performs an exponential number of iterations, and thus is not strongly polynomial. In a similar spirit, recent lower bounds on the number of simplex pivots required in the worst-case to perform linear optimization over a lattice polytope will be presented, as well as the first worst-case instances for geometric scaling methods that solve integer optimization problems by primal augmentation steps.

References

[1] Antoine Deza, Mingfei Hao, and Lionel Pournin: *Sizing the White Whale*. Fields Institute Communications Series on Data Science and Optimization (to appear).
[2] Antoine Deza, Sebastian Pokutta, and Lionel Pournin: *The complexity of geometric scaling*. arXiv:2205.04063 (2022).
[3] Antoine Deza and Lionel Pournin: *Primitive point packing*. Mathematika 68 (2022) 979 – 1007.
[4] Antoine Deza, Lionel Pournin, and Noriyoshi Sukegawa: *The diameter of lattice zonotopes*. Proceedings of the American Mathematical Society 148 (2020) 3507 – 3516.
[5] Anna Deza, Antoine Deza, Zhongyan Guan, and Lionel Pournin: *Distances between vertices of lattice polytopes*. Optimization Letters 14 (2020) 309 – 326.
[6] Antoine Deza, Asaf Levin, Syed Meesum, and Shmuel Onn: Optimization over degree sequences. SIAM Journal on Discrete Mathematics 32 (2018) 2067 – 2079.
[7] Antoine Deza, George Manoussakis, and Shmuel Onn: *Primitive zonotopes*. Discrete and Computational Geometry 60 (2018) 27 – 39.

Worst-case constructions for linear optimization



Antoine Deza, McMaster

based on joint works with:
Shmuel Onn, Technion, **Sebastian Pokutta**, ZIB, **Lionel Pournin**, Paris XIII

Linear optimization

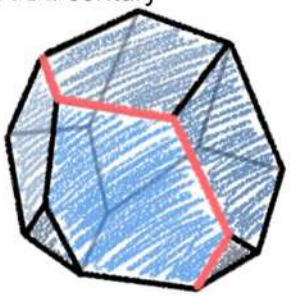
Given an ***n***-dimensional vector ***b*** and an ***n*** x ***d*** matrix ***A***
find, in any, a ***d***-dimensional vector ***x*** such that :

$\boldsymbol{Ax} = \boldsymbol{b}$ $\qquad$ $\boldsymbol{Ax} \leq \boldsymbol{b}$

linear algebra $\qquad$ **linear optimization**

"Can linear optimization be solved in ***strongly polynomial*** time?"
is listed by Smale as one of the top problems for the XXI century

Strongly polynomial *:* algorithm ***independent*** from
the ***input data length*** and polynomial in ***n*** and ***d***.

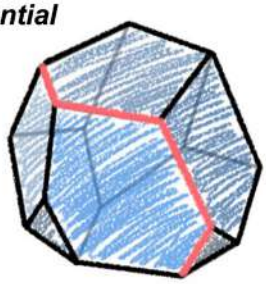


Linear optimization algorithms
simplex methods

Given an ***n***-dimensional vector ***b*** and an ***n*** x ***d*** (full row-rank) matrix ***A*** and a ***d***-dimensional cost vector ***c***, solve : max { $\boldsymbol{c}^{\mathrm{T}}\boldsymbol{x}$: $\boldsymbol{Ax} = \boldsymbol{b}$, $\boldsymbol{x} \geq \boldsymbol{0}$ }

Simplex methods (Dantzig 1947): pivot-based, combinatorial, ***not proven to be polynomial***, efficient in practice

- start from a ***feasible basis***
- use a ***pivot rule***
- find an optimal solution after a ***finite number*** of iterations
- most known pivot rules are known to be ***exponential*** (worst case); ***efficient*** implementations exist

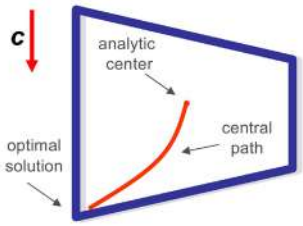


Linear optimization algorithms
(central path following) interior point methods

Given an ***n***-dimensional vector ***b*** and an ***n*** x ***d*** (full row-rank) matrix ***A*** and a ***d***-dimensional cost vector ***c***, solve : max { $\boldsymbol{c}^{\mathrm{T}}\boldsymbol{x}$: $\boldsymbol{Ax} = \boldsymbol{b}$, $\boldsymbol{x} \geq \boldsymbol{0}$ }

Interior Point Methods :
path-following, ***polynomial***, efficient in practice

- start from the ***analytic center***
- follow the ***central path***
- converge to an optimal solution in $O(\sqrt{n}L)$ iterations (***L***: input data length)



$$\max \quad c^{\mathrm{T}}x - \mu \sum_i \ln(b - Ax)_i$$

μ : central path parameter
$x \in \boldsymbol{P}$: $Ax \leq b$

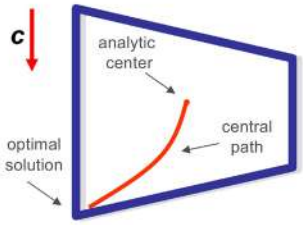
Linear optimization diameter and curvature

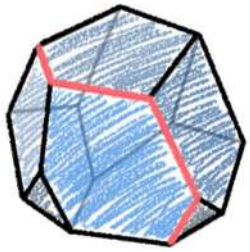
Diameter (*of a polytope*) :

lower bound for the number of iterations for *pivoting* **simplex methods**

Curvature (*of the central path associated to a polytope*) :

large curvature indicates large number of iterations for *path following* **interior point methods**





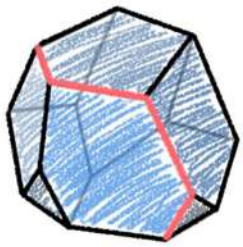
Linear optimization

Given an ***n***-dimensional vector ***b*** and an ***n*** x ***d*** matrix ***A***
find, in any, a ***d***-dimensional vector ***x*** such that :

$\boldsymbol{Ax} = \boldsymbol{b}$	$\boldsymbol{Ax} \leq \boldsymbol{b}$
linear algebra	**linear optimization**

"Can linear optimization be solved in ***strongly polynomial*** time?"
is listed by Smale as one of the top I problems for the XXI century

- [Allamigeon, Benchimol, Gaubert, Joswig 2018] (logarithmic barrier) ***Interior point methods*** are ***not strongly polynomial***
- [Allamigeon, Gaubert, Vandame 2022] (self-concordant barrier) ***Interior point methods*** are ***not strongly polynomial***



(tropical counterexample to continuous Hirsch conjecture [Deza-Terlaky-Zinchenko 2008])

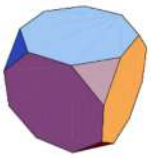
Lattice polytopes with large diameter

lattice (d,k)-polytope : convex hull of points drawn from $\{0,1,\ldots,k\}^d$

diameter $\delta(P)$ of polytope P : smallest number such that **any two vertices** of P can be connected by a **path with at most** $\delta(P)$ **edges**

$\delta(d,k)$: largest diameter over all **lattice** (d,k)-polytopes

ex. $\delta(3,3) = 6$ and is achieved by a ***truncated cube***



- $\delta(d,k)$: lower bound on the number of simplex pivots required in the worst case to perform linear optimization on a lattice polytope
- [Del Pia-Michini 2018] *preprocessing* and *scaling algorithm* yielding simplex paths that are ***short relative*** to $\delta(d,k)$

Lattice polytopes with large diameter

lattice (d,k)-polytope : convex hull of points drawn from $\{0,1,\ldots,k\}^d$

diameter $\delta(P)$ of polytope P : smallest number such that **any two vertices** of P can be connected by a **path with at most** $\delta(P)$ **edges**

$\delta(d,k)$: largest diameter over all **lattice** (d,k)-polytopes

- $\delta(P)$: lower bound for the worst-case number of iterations required by *pivoting methods* (simplex) to optimize a linear function over P
- *Hirsch conjecture* : $\delta(P) \leq n - d$ (n number of inequalities) was ***disproved*** [Santos 2012]

$\delta(P) \leq (n - d)^{\log d - \ldots}$ [Kalai-Kleitman 1992, Todd 2014, Sukegawa 2019]

❖ ***no polynomial upper bound*** known for $\delta(P)$

Lattice polytopes with large diameter

$\delta(d,k)$: largest ***diameter*** of a convex hull of points drawn from $\{0,1,\ldots,k\}^d$

upper bounds :

$\delta(d,1) \le d$		[Naddef 1989]
$\delta(2,k) = O(k^{2/3})$		[Balog-Bárány 1991]
$\delta(2,k) = 6(k/2\pi)^{2/3} + O(k^{1/3} \log k)$		[Thiele 1991] [Acketa-Žunić 1995]
$\delta(d,k) \le kd$		[Kleinschmid-Onn 1992]
$\delta(d,k) \le kd - \lceil d/2 \rceil$	for $k \ge 2$	[Del Pia-Michini 2016]
$\delta(d,k) \le kd - \lceil 2d/3 \rceil - (k - 3)$	for $k \ge 3$	[Deza-Pournin 2018]

Lattice polytopes with large diameter

$\delta(d,k)$: largest ***diameter*** of a convex hull of points drawn from $\{0,1,\ldots,k\}^d$

lower bounds :

$\delta(d,1) \ge d$	[Naddef 1989]
$\delta(d,2) \ge \lfloor 3d/2 \rfloor$	[Del Pia-Michini 2016]
$\delta(d,k) = \Omega(k^{2/3}\, d)$	[Del Pia-Michini 2016]
$\delta(d,k) \ge \lfloor (k+1)d/2 \rfloor$ for $k < 2d$	[Deza-Manoussakis-Onn 2018]
$\delta(d,k) = \Omega(k^{d/d+1})$ for fixed d	[Deza-Pournin-Sukegawa 2020]

- Lower bound of $\Omega(k^{d/d+1})$ obtained by counting primitive points within simplex and cross polytope blown up by an integer factor

  [Manecke-Sanyal 2020]: primitive Ehrhart theory

Lattice polytopes with large diameter

$\delta(d,k)$		k								
		1	2	3	4	5	6	7	8	9
d	2	**2**	**3**	**4**	**4**	**5**	**6**	**6**	**7**	**8**
	3	**3**	**4**	6	**7**	**9**	**10**			
	4	**4**	**6**	**8**						
	5	**5**	**7**	**10**						

$\delta(d,1) = d$	[Naddef 1989]
$\delta(2,k)$: close form	[Thiele 1991] [Acketa-Žunić 1995]
$\delta(d,2) = \lfloor 3d/2 \rfloor$	[Del Pia-Michini 2016]
$\delta(4,3)=8$, $\delta(3,4)=7$, $\delta(3,5)=9$	[Deza-Pournin 2018], [Chadder-Deza 2017]
$\delta(5,3)=10$, $\delta(3,6)=10$	[Deza-Deza-Guan-Pournin 2019]

Lattice polytopes with large diameter

$\delta(d,k)$		k								
		1	2	3	4	5	6	7	8	9
d	2	2	3	4	4	5	6	6	7	8
	3	3	4	6	7	9	10	11+	12+	13+
	4	4	6	8	10+	12+	14+	16+	17+	18+
	5	5	7	10	12+	15+	17+	20+	22+	25+

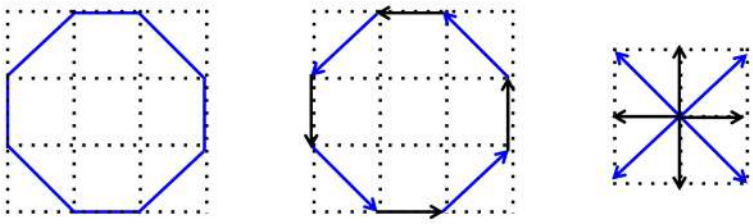
- Conjecture [Deza-Manoussakis-Onn 2018] $\delta(d,k) \leq \lfloor (k+1)d/2 \rfloor$

and $\delta(d,k)$ is achieved, up to translation, by a *Minkowski sum of primitive lattice vectors*. The conjecture holds for all known entries of $\delta(d,k)$

Lattice polygons with large diameter

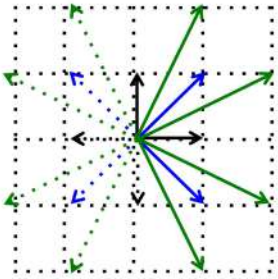
Q. What is $\delta(2,k)$: largest diameter of a polygon which vertices are drawn form the k x k grid?

A polygon can be associated to a set of vectors (*edges*) *summing up to zero*, and *without a pair of positively multiple vectors*



$\delta(2,3) = 4$ is achieved by the 8 vectors : $(\pm 1,0)$, $(0,\pm 1)$, $(\pm 1,\pm 1)$

Primitive polygons



$||x||_1 \leq p$

$H_1(2,p)$: Minkowski sum generated by $\{x \in \mathbb{Z}^2 : ||x||_1 \leq p, \gcd(x)=1, x \succeq 0\}$

$H_1(2,p)$ has diameter $\delta(2,k) = 2\sum_{i=1}^{p} \varphi(i)$ for $k = \sum_{i=1}^{p} i\varphi(i)$

Ex. $H_1(2,2)$ generated by (1,0), (0,1), (1,1), (1,-1) (fits, *up to translation*, in 3x3 grid)

$\varphi(p)$: ***Euler totient function*** counting positive integers less or equal to p relatively prime with p
$\varphi(1) = \varphi(2) = 1$, $\varphi(3) = \varphi(4) = 2,\ldots$ $x \succeq 0$: first nonzero coordinate of x is nonnegative

Primitive zonotopes

$H_q(\boldsymbol{d},\boldsymbol{p})$: Minkowski ($x \in \mathbf{Z}^d$: $||x||_q \leq \boldsymbol{p}$, gcd($x$)=1, $x \succeq 0$)

$x \succeq 0$: first nonzero coordinate of x is nonnegative

Given a set G of $\boldsymbol{m}$ vectors (generators),
Minkowski (G) : convex hull of all the $2^{\boldsymbol{m}}$ ***subsums*** of the $\boldsymbol{m}$ vectors in G

- ***Primitive zonotopes***: Minkowski sum generated by ***short integer*** vectors which are ***pairwise linearly independent***

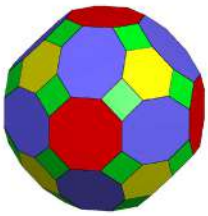
  - *Note*: convex hull of all the ***signed*** subsums of the vectors of $H_q(\boldsymbol{d},\boldsymbol{p})$ *is a generalization of the permutahedron of type* B_d

Primitive zonotopes

$H_q(\boldsymbol{d},\boldsymbol{p})$: Minkowski ($x \in \mathbf{Z}^d$: $||x||_q \leq \boldsymbol{p}$, gcd($x$)=1, $x \succeq 0$)

$x \succeq 0$: first nonzero coordinate of x is nonnegative

- $H_q(\boldsymbol{d}, 1)$: $[0, 1]^d$ cube for $\boldsymbol{q} \neq \infty$
- $H_1(\boldsymbol{d},2)$: permutahedron of type B_d *(up to a homothety)*
- $H_1(\boldsymbol{3},2)$: great rhombicuboctahedron
- $H_\infty(\boldsymbol{3},1)$: truncated small rhombicuboctahedron



Primitive zonotopes

❖ lattice polytopes with *large diameter*

$$H_q(d,p) : \text{Minkowski } (x \in \mathbf{Z}^d : ||x||_q \le p,\ \gcd(x)=1,\ x \succeq 0)$$

$x \succeq 0$: first nonzero coordinate of x is nonnegative

➢ For $k < 2d$, Minkowski sum of a subset of the generators of $H_1(d,2)$ is, up to translation, a lattice (d,k)-polytope with diameter $\lfloor (k+1)d/2 \rfloor$

Positive primitive zonotopes

$$H_q(d,p) : \text{Minkowski } (x \in \mathbf{Z}^d : ||x||_q \le p,\ \gcd(x)=1,\ x \succeq 0)$$

$x \succ 0$: first nonzero coordinate of x is nonnegative

$$H_q(d,p)^+ : \text{Minkowski } (x \in \mathbf{Z}_+^d : ||x||_q \le p,\ \gcd(x)=1)$$

➢ $H_1(d,2)^+$: Minkowski sum permutahedron + unit cube *(graphical zonotope)*

➢ $H_\infty(d,1)^+$: ***white whale*** *(hypergraphical zonotope)*

$$a(d) = |H_\infty(d,1)^+|$$

number $a(d)$ of generalized retarded functions in quantum field theory is equal to the number of vertices *of* $H_\infty(d,1)^+$

Berlin Mathematical School

BMS Friday Colloquium

Friday 27 April 2018 at 14:15

Tea & Cookies starting at 13:00

BMS Loft, Urania, An der Urania 17, 10787 Berlin

Louis J. Billera

(Cornell University)

© Louis J. Billera

In pursuit of a white whale: On the real linear algebra of vectors of zeros and ones

We are interested in the real linear relations (the real *matroid*) on the set of all 0-1 n-vectors. This fundamental combinatorial object is behind questions arising over the past 50 years in a variety of fields, from economics, circuit theory and integer programming to quantum physics, and has connections to an 1893 problem of Hadamard. Yet there has been little real progress on some of the most basic questions.

Some applications seek the number of regions in $\mathbf{R}^n$ that are determined by the $2^n - 1$ linear hyperplanes having 0–1 normals. This number, asymptotically 2^{n^2}, can be obtained exactly from the characteristic polynomial of the geometric lattice of all real subspaces spanned by these 0–1 vectors. These polynomials are known only through $n = 7$, while the number of regions is known through $n = 8$.

Discrete optimization and theoretical physics

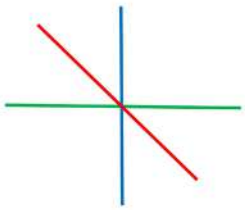
- Ising model (spin glasses)
  maxcut, cut and metric polytopes [Deza-Laurent 1997]

- a(***d***) : number of generalized retarded functions in quantum field theory
  (number of real-time Green functions) [Evans 1994]

a(***d***) = number of regions of the arrangement formed by the 2^d -1 hyperplanes with {0,1}-valued normals in dimension ***d***

d = 2 2^d -1 = 3 hyperplanes

(0,1)
(1,0)
(1,1)



6 regions

- a(**2**)=6

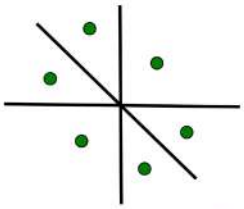
Discrete optimization and theoretical physics

- Ising model (spin glasses)
  maxcut, cut and metric polytopes [Deza-Laurent 1997]

- a(***d***) : number of generalized retarded functions in quantum field theory
  (number of real-time Green functions) [Evans 1994]

a(***d***) = number of regions of the arrangement formed by the 2^d -1 hyperplanes with {0,1}-valued normals in dimension ***d***

- is a(***d***) ≥ ***d*** ! [question by Evans]
- a(***d***) determined till ***d*** = 9
- how to estimate a(***d***) ?



a(***d***) regions <=> a(***d***) vertices

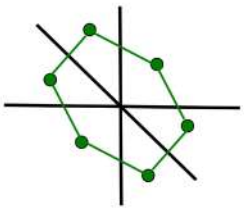
Discrete optimization and theoretical physics

- Ising model (spin glasses)
  maxcut, cut and metric polytopes [Deza-Laurent 1997]

- a(***d***) : number of generalized retarded functions in quantum field theory
  (number of real-time Green functions) [Evans 1994]

a(***d***) = number of regions of the arrangement formed by the 2^d -1 hyperplanes with {0,1}-valued normals in dimension ***d***

- is a(***d***) ≥ ***d*** ! [question by Evans]
- a(***d***) determined till ***d*** = 9
- how to estimate a(***d***) ?
- a(***d***) vertices of the *white whale*



a(**2**) = 6

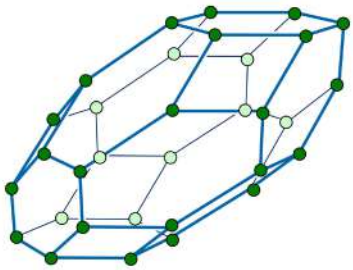
Discrete optimization and theoretical physics

- Ising model (spin glasses)
  maxcut, cut and metric polytopes [Deza-Laurent 1997]

- a(***d***) : number of generalized retarded functions in quantum field theory
  (number of real-time Green functions) [Evans 1994]

a(***d***) = number of regions of the arrangement formed by the 2^d -1 hyperplanes with {0,1}-valued normals in dimension ***d***

- is a(***d***) ≥ ***d*** ! [question by Evans]
- a(***d***) determined till ***d*** = 9
- how to estimate a(***d***) ?
- a(***d***) vertices of the *white whale*



a(**3**) = 32

Vertices of primitive zonotopes

Sloane OEI sequences

$H_\infty(d,1)^+$ vertices : A034997 = number of generalized retarded functions in quantum Field theory (determined till ***d*** = 9)

$H_\infty(d,1)$ vertices : A009997 = number of regions of hyperplane arrangements with {-1,0,1}-valued normals in dimension ***d*** (determined till ***d*** = 7)

Estimating the number of vertices of $H_\infty(d,1)^+$ (*white whale*)

$d(d-1)/2 \le \log_2 |H_\infty(d,1)^+| \le d^2$ [Billera et al 2012]

$d(d-1)/2 \le \log_2 |H_\infty(d,1)^+| \le d(d-3)$ [Deza-Pournin-Rakotonarivo 2021]

$d^2(1-\varepsilon_d) \le \log_2 |H_\infty(d,1)^+| \le d(d-3)$ [Gutekunst, Mészáros, Petersen 2021]

(root resonance arrangement, maximal unbalanced families...)

Sizing the White Whale

d	$a(d)$	
2	6	[Evans 1995]
3	32	[Evans 1995]
4	370	[Evans 1995]
5	11 292	[Evans 1995, van Eijck 1995]
6	1 066 044	[Evans 1995, van Eijck 1995]
7	347 326 352	[van Eijck 1995, Kamiya, Takemura, Terao 2011]
8	419 172 756 930	[Evans 2011]
9	1 955 230 985 997 140	[Brysiewicz, Eble, Kühne 2021] [Chroman-Singhal 2021] [Deza-Hao-Pournin 2021]

Generating and **counting** the vertices of the White Whale

- [Deza-Hao-Pournin 2021] : Generating all the edges of White While till $d = 9$, and exhibiting a family of White While vertices of degree roughly 2^d

Vertices of primitive zonotopes

Sloane OEI sequences

$H_\infty(d,1)^+$ vertices : A034997 = number of generalized retarded functions in quantum Field theory (determined till $d = 9$)

$H_\infty(d,1)$ vertices : A009997 = number of regions of hyperplane arrangements with {-1,0,1}-valued normals in dimension d (determined till $d = 7$)

Estimating the number of vertices of $H_\infty(d,1)$ (*matroid optimization*)

$d \leq \log_3 |H_\infty(d,1)| \leq d(d-1)$ [Melamed-Onn 2014]

$d \log d \leq \log_3 |H_\infty(d,1)| \leq d(d-1)$ [Deza-Onn-Manoussakis 2018]

$d(d-1)/2 \leq \log_3 |H_\infty(d,1)| \leq d(d-2)$ [Deza-Pournin-Rakotonarivo 2021]

Convex matroid optimization

The optimal solution of max { **f**(**W**x) : $x \in$ **S**} is attained at a vertex of the projection integer polytope in $\mathbf{R}^d$: conv(**WS**) = **W**conv(**S**)

S : set of feasible point in $\mathbf{Z}^n$ (in the talk **S** $\in \{0,1\}^n$)
W : integer d x n matrix (**W** is $\{0,1,\ldots, p\}$-valued)
f : convex function from $\mathbf{R}^d$ to $\mathbf{R}$

Q. What is the maximum number **v**(d,n) of vertices of conv(**WS**) when **S** $\in \{0,1\}^n$ and **W** is a {0,1}-valued d x n matrix ?

obviously $v(d,n) \leq |\mathbf{WS}| = O(n^d)$
in particular $v(2,n) = O(n^2)$, and $v(2,n) = \Omega(n^{0.5})$

- [Hunkenschröder, Pokutta, Weismantel 2022] : min { **g**(**W**x) : $x \in \{0,1\}^n$

  Machine Learning setting with **W** unknow, but $||\mathbf{W}||_\infty$ and the number of rows $m \ll n$ are revealed, some conditions on **g** such as having Lipschitz continuous gradients

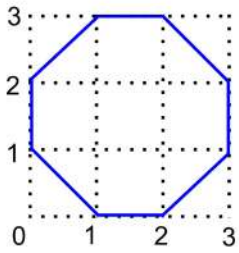
Convex matroid optimization

[Melamed-Onn 2014] Given matroid **S** of order n and $\{0,1,\ldots,p\}$-valued d x n matrix **W**, the maximum number **m**(d,p) of vertices of conv(**WS**) is independent of n and **S**

Ex: maximum number **m**(2,1) of vertices of a planar projection conv(**WS**) of matroid **S** by a binary matrix **W** is attained by the following matrix and uniform matroid of rank 3 and order 8:

$$\mathbf{W} = \begin{pmatrix} 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{pmatrix}$$

$$\mathbf{S} = U(3,8) = \begin{pmatrix} 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \end{pmatrix}$$



conv(**WS**)

Convex matroid optimization

The optimal solution of max { **f**(**W**x) : $x \in$ **S**} is attained at a vertex of the projection integer polytope in $\mathbf{R}^d$: conv(**WS**) = **W**conv(**S**)

S : set of feasible point in $\mathbf{Z}^n$ (in the talk **S** $\in \{0,1\}^n$)
W : integer d x n matrix (**W** is mostly {0,1,…, p}-valued)
f : convex function from $\mathbf{R}^d$ to $\mathbf{R}$

Q. What is the maximum number **v**(d,n) of vertices of conv(**WS**) when **S** $\in \{0,1\}^n$ and **W** is a {0,1}-valued d x n matrix ?

$$v(d,n) \le |\mathbf{WS}| = O(n^d)$$
$$v(2,n) = O(n^2), \quad \text{and } v(2,n) = \Omega(n^{0.5})$$

[Melamed-Onn 2014] Given matroid **S** of order n and {0,1,…,p}-valued d x n matrix **W**, the maximum number **m**(d,p) of vertices of conv(**WS**) is independent of n and **S**

Convex matroid optimization

[Melamed-Onn 2014] Given matroid **S** of order n and {0,1,…,p}-valued d x n matrix **W**, the maximum number **m**(d,p) of vertices of conv(**WS**) is independent of n and **S**

[Deza-Manoussakis-Onn 2018] Given matroid **S** of order n, {0,1,…,p}-valued d x n matrix **W**, maximum number **m**(d,p) of vertices of conv(**WS**) is equal to the number of vertices of $H_\infty(d,p)$

$$\mathbf{m}(d,p) = |\, H_\infty(d,p) \,|$$

[Melamed-Onn 2014]

$$d\, 2^d \le \mathbf{m}(d,1) \le 2\sum_{i=0}^{d-1}\binom{(3^d-3)/2}{i}$$

$24 \le \mathbf{m}(3,1) \le 158$
$64 \le \mathbf{m}(4,1) \le 19840$

$\mathbf{m}(2,1) = 8$

[Deza-Pournin-Rakotonarivo 2021]

$$3^{d(d-1)/2} \le \mathbf{m}(d,1) \le 3^{d(d-2)}$$

$\mathbf{m}(3,1) = 96$
$\mathbf{m}(4,1) = 5376$

$$\mathbf{m}(2,p) = 8\sum_{i=1}^{p}\varphi(i)$$

Geometric scaling

(IP) integer optimization max $\{\boldsymbol{c}^T\boldsymbol{x} : \boldsymbol{x} \in \boldsymbol{P} \cap \{0,1\}^d\}$

[Schultz-Weismantel-Ziegler 1995] optimization and augmentation are equivalent (***bit scaling***)

[Schulz-Weismantel 2002] ***geometric scaling*** solves (IP) by $O(\boldsymbol{d} \log \boldsymbol{d}\, ||c||_\infty)$ augmentation oracle calls

[Le Bodic-Pavelka-Pfetsch-Pokutta 2018] ***geometric scaling*** solves (IP) by $O(\boldsymbol{d} \log ||\boldsymbol{c}||_\infty)$ augmentation oracle calls

[Deza-Pournin-Pokutta 2022] ***geometric scaling*** may require $\boldsymbol{d} + \log ||\boldsymbol{c}||_\infty + 1$ iterations over a simplex

[Le Bodic-Pavelka-Pfetsch-Pokutta 2018] tight upper and lower bound for ***bit scaling***

Maximum ratio augmentation based geometric scaling

(IP) integer optimization max $\{\boldsymbol{c}^T\boldsymbol{x} : \boldsymbol{x} \in \boldsymbol{P} \cap \{0,1\}^d\}$

Input: $\boldsymbol{P}$, $\boldsymbol{c} \in \mathbb{Z}^d$, vertex $\boldsymbol{x^o} \in \boldsymbol{P}$, $\mu_o \geq ||\boldsymbol{c}||_\infty$
Output: vertex $\boldsymbol{x}^*$ maximizing $\boldsymbol{c}^T\boldsymbol{x}$

1. $\mu \leftarrow \mu_o$, $\boldsymbol{x}^* \leftarrow \boldsymbol{x^o}$
2. repeat
3. compute vertex $\boldsymbol{x^+} \in \boldsymbol{P}$ **maximizing** $\boldsymbol{c}^T(\boldsymbol{x^+} - \boldsymbol{x}^*) / ||\boldsymbol{x^+} - \boldsymbol{x}^*||_1$
4. if $\boldsymbol{x^+} = \boldsymbol{x}^*$ or $\boldsymbol{c}^T(\boldsymbol{x^+} - \boldsymbol{x}^*) < \mu ||\boldsymbol{x^+} - \boldsymbol{x}^*||_1$ then $\mu \leftarrow \mu/2$ (halving step)
5. else $\boldsymbol{x}^* \leftarrow \boldsymbol{x^+}$ (augmenting step)
6. end
7. until $\mu < 1/\boldsymbol{d}$
8. return $\boldsymbol{x}^*$

$\boldsymbol{P}$ = convex hull $(v^0, v^1, \ldots, v^d)$ where $v^i = (0,\ldots0,1\ldots1)$ with i ones
$\boldsymbol{c} = (1,2,3,\ldots, \boldsymbol{d})$, $x^0 = v^0$

- requires $\boldsymbol{d}$ augmenting steps and $\log ||\boldsymbol{c}||_\infty + 1$ halving steps

Feasibility test based geometric scaling

(IP) integer optimization max $\{\boldsymbol{c}^T\boldsymbol{x} : \boldsymbol{x} \in \boldsymbol{P} \cap \{0,1\}^d\}$

Input: $\boldsymbol{P}$, $\boldsymbol{c} \in \mathbb{Z}^d$, vertex $\boldsymbol{x^o} \in \boldsymbol{P}$, $\mu_o \geq \|\boldsymbol{c}\|_\infty$
Output: vertex $\boldsymbol{x}^*$ maximizing $\boldsymbol{c}^T\boldsymbol{x}$

1. $\mu \leftarrow \mu_o$, $\boldsymbol{x}^* \leftarrow \boldsymbol{x^o}$
2. repeat
3. compute **a vertex** $\boldsymbol{x}^+ \in \boldsymbol{P}$ such that $\boldsymbol{c}^T(\boldsymbol{x}^+ - \boldsymbol{x}^*) > \mu \| \boldsymbol{x}^+ - \boldsymbol{x}^* \|_1$
4. if there is no such vertex then $\mu \leftarrow \mu/2$ (halving step)
5. else $\boldsymbol{x}^* \leftarrow \boldsymbol{x}^+$ (augmenting step)
6. end
7. until $\mu < 1/d$
8. return $\boldsymbol{x}^*$

$\boldsymbol{P}$ = convex hull $(v^0, v^1, \ldots, v^d)$ where $v^i = (0,\ldots0,1\ldots1)$ with i ones
$\boldsymbol{c} = (2,4,8,\ldots,2^d)$, $x^0 = v^0$

- requires $d/3$ augmenting steps and $\log \|\boldsymbol{c}\|_\infty + 1$ halving steps

Feasibility test based geometric scaling

(IP) integer optimization max $\{\boldsymbol{c}^T\boldsymbol{x} : \boldsymbol{x} \in \boldsymbol{P} \cap \{0,1\}^d\}$

Input: $\boldsymbol{P}$, $\boldsymbol{c} \in \mathbb{Z}^d$, vertex $\boldsymbol{x^o} \in \boldsymbol{P}$, $\mu_o \geq \|\boldsymbol{c}\|_\infty$
Output: vertex $\boldsymbol{x}^*$ maximizing $\boldsymbol{c}^T\boldsymbol{x}$

1. $\mu \leftarrow \mu_o$, $\boldsymbol{x}^* \leftarrow \boldsymbol{x^o}$
2. repeat
3. compute **a vertex** $\boldsymbol{x}^+ \in \boldsymbol{P}$ such that $\boldsymbol{c}^T(\boldsymbol{x}^+ - \boldsymbol{x}^*) > \mu \| \boldsymbol{x}^+ - \boldsymbol{x}^* \|_1$
4. if there is no such vertex then $\mu \leftarrow 3\mu/4$ (halving step)
5. else $\boldsymbol{x}^* \leftarrow \boldsymbol{x}^+$ (augmenting step)
6. end
7. until $\mu < 1/d$
8. return $\boldsymbol{x}^*$

$\boldsymbol{P}$ = convex hull $(v^0, v^1, \ldots, v^d)$ where $v^i = (0,\ldots0,1\ldots1)$ with i ones
$\boldsymbol{c} = (2,4,8,\ldots,2^d)$, $x^0 = v^0$

- requires d augmenting steps and $\log \|\boldsymbol{c}\|_\infty + 1$ halving steps

Primitive zonotopes

(degree sequences)

$\boldsymbol{D_d}$: convex hull of the degree sequences of all hypergraphs on $\boldsymbol{d}$ nodes

$$\boldsymbol{D_d} = H_\infty(\boldsymbol{d},1)^+$$

$\boldsymbol{D_d}(\boldsymbol{k})$: convex hull of the degree sequences of all $\boldsymbol{k}$-uniform hypergraphs on $\boldsymbol{d}$ nodes

Q: check whether $x \in \boldsymbol{D_d}(\boldsymbol{k}) \cap \mathbb{Z}^d$ is the degree sequence of a $\boldsymbol{k}$-uniform hypergraph. Necessary condition: sum of the coordinates of x is multiple of $\boldsymbol{k}$.

[Erdős-Gallai 1960]: for $\boldsymbol{k} = 2$ (graphs) necessary condition is sufficient

[Liu 2013] exhibited counterexamples (holes) for $\boldsymbol{k} = 3$ (Klivans-Reiner **Q**.)

➢Answer to Colbourn-Kocay-Stinson **Q.** (1986)
Deciding whether a given integer sequence is the degree sequence of a **3**-uniform hypergraph is NP-complete [Deza-Levin-Meesum-Onn 2018]

(reduction to 3-partition problem)

Primitive zonotopes, convex matroid optimization, and degree sequences of hypergraphs

$\delta(\boldsymbol{d},\boldsymbol{k})$: *largest diameter* over all lattice $(\boldsymbol{d},\boldsymbol{k})$-polytopes

➢Conjecture : $\delta(\boldsymbol{d},\boldsymbol{k}) \leq \lfloor(\boldsymbol{k}+1)\boldsymbol{d}/2\rfloor$ and $\delta(\boldsymbol{d},\boldsymbol{k})$ is achieved, up to translation, by a *Minkowski sum* of primitive lattice vectors (holds for all known $\delta(\boldsymbol{d},\boldsymbol{k})$)

$$\Rightarrow \delta(\boldsymbol{d},\boldsymbol{k}) = \lfloor(\boldsymbol{k}+1)\boldsymbol{d}/2\rfloor \text{ for } \boldsymbol{k} < 2\boldsymbol{d}$$

➢$\mathbf{m}(\boldsymbol{d},\boldsymbol{p}) = |\, H_\infty(\boldsymbol{d},\boldsymbol{p})\,|$ (*convex matroid optimization complexity*)

➢tightening of the *bounds* for $\mathbf{m}(\boldsymbol{d},\mathbf{1}) = |\, H_\infty(\boldsymbol{d},\mathbf{1})\,|$

➢tightening of the *bounds* for $\mathrm{a}(\boldsymbol{d}) = |\, H_\infty(\boldsymbol{d},\mathbf{1})^+\,|$ (*white whale*)

➢Answer to [Colbourn-Kocay-Stinson 1986] question:
Deciding whether a given integer sequence is the *degree sequence* of a 3-hypergraph is *NP-complete* [Deza-Levin-Meesum-Onn 2018]

- ✓ Deza, Pournin: *Primitive point packing.*
  Mathematika (2022)

- ✓ Deza, Pokutta, Pournin: *The complexity of geometric scaling*.
  arXiv (2022)

- ✓ Deza, Hao, Pournin: *Sizing the White Whale.*
  arXiv (2022)

- ✓ Deza, Pournin, Rakotonarivo: *Vertices of primitive zonotopes.*
  Contemporary Mathematics (2021)

- ✓ Deza, Pournin, Sukegawa: *The diameter of lattice zonotopes*.
  Proceedings of the American Mathematical Society (2020)

- ✓ Deza, Deza, Guan, Pournin: *Distances between vertices of lattice polytopes*.
  Optimization Letters (2019)

- ✓ Deza, Levin, Meesum, Onn: *Optimization over degree sequences*.
  SIAM Journal on Discrete Mathematics (2018)

- ✓ Deza, Manoussakis, Onn: *Primitive zonotopes.*
  Discrete and Computational Geometry (2018)

Approximate Methods for Solving Chance Constrained Linear Programs in Probability Measure Space

Xun Shen

Tokyo Institute of Technology, Japan
shen.x.ad@m.titech.ac.jp

Many risk-aware decision-making problems can be formulated as a chance constrained linear program in probability measure space, which is NP-hard and unsolvable directly. In this talk, we introduce approximate methods to address this NP-hard problem. In the proposed methods, the original problem is approximated by two kinds of solvable optimization problems in finite-dimension space. We show the convergence of the approximations and give numerical experiments including a stochastic control problem for validation. Two numerical examples are presented to show the effectiveness of the proposed method.

Approximate Methods for Solving Chance Constrained Linear Programs in Probability Measure Space

Xun Shen, Satoshi Ito

Sep. 19th 2022
The 6th RIKEN-IMI-ISM-NUS-ZIB-MODAL-NHR Workshop

Overview

- Background and Motivation
- Problem Formulation
- Proposed Method
- Numerical Examples
- Conclusion and Future Work

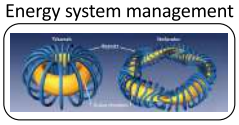
2

Background and Motivation

- Chance constrained optimal control: applications

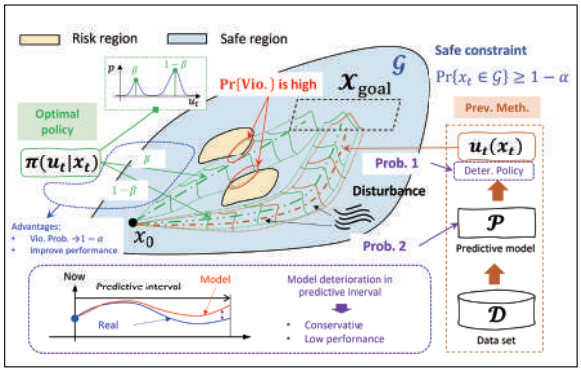
Planetary rover and self-driving vehicle

Energy system management



Magnetic control of tokamak plasmas

Problem formulation



Background and Motivation

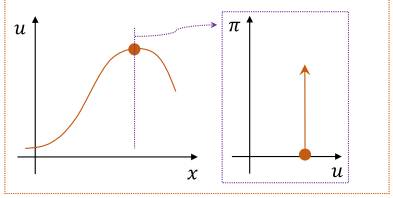
- Chance constrained optimal control: stochastic policy

Time index: t State variable: $x_t \in \mathbb{R}^n$ Input variable: $u_t \in \mathbb{R}^m$ Stochastic variable: $\delta_t \in \Delta$

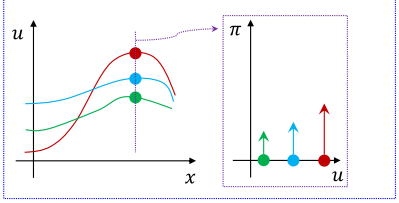
Let $M(\mathbb{R}^m)$ be the space of probability measure on $\mathbb{R}^m$.

Let $\pi : \mathbb{R}^n \to M(\mathbb{R}^m)$ be stochastic policy. For a given x_t, $u_t \sim \pi(u_t|x_t)$. The space of $\pi : \Pi$.

- Deterministic policy



- Stochastic policy



Background and Motivation

- Chance constrained optimal control: how to obtain optimal policy (closed loop)

Time index: t State variable: $x_t \in \mathbb{R}^n$ Input variable: $u_t \in \mathbb{R}^m$ Stochastic variable: $\delta_t \in \Delta$

$$(\mathcal{P}_{\mathrm{cl}}) \quad \min_{\pi \in \Pi} \ \mathbb{E}\{\gamma^T \ell_T(x_T) + \sum_{t=0}^{T-1} \gamma^t \ell_t(x_t, u_t)\}, \quad \gamma \in (0,1], \qquad \text{Objective function}$$

$$\text{s.t.} \ \ x_{t+1} = f(x_t, u_t, \delta_t), \ \ u_t \sim \pi(\cdot|x_t), \ \ t = 0, 1, ..., T-1, \qquad \text{Dynamics}$$

NP hard

$$\Pr \left\{ \begin{array}{c} g_1(x_1) \le 0 \\ ... \\ g_t(x_t) \le 0 \\ ... \\ g_{T-1}(x_{T-1}) \le 0 \\ g_T(x_T) \le 0, \end{array} \right\} \ge 1 - \alpha, \qquad \text{Chance constraint}$$

Probability level $\alpha \in [0,1]$

$$u_t \in [u_{\min}, u_{\max}], t = 0, 1, ..., T-1. \qquad \text{Normal constraint}$$

5

Background and Motivation

- Chance constrained optimal control: how to obtain optimal policy (open loop)

Time index: t State variable: $x_t \in \mathbb{R}^n$ Input variable: $u_t \in \mathbb{R}^m$ Stochastic variable: $\delta_t \in \Delta$

$$(\mathcal{P}_{\mathrm{ol}}) \quad \min_{\pi \in \Pi} \ \mathbb{E}\{\gamma^T \ell_T(x_T) + \sum_{t=0}^{T-1} \gamma^t \ell_t(x_t, u_t)\}, \quad \gamma \in (0,1], \qquad \text{Objective function}$$

$$\text{s.t.} \ \ x_{t+1} = f(x_t, u_t, \delta_t), \ \ \boxed{u_t \sim \pi_t(\cdot|x_0)}, \ \ t = 0, 1, ..., T-1, \qquad \text{Dynamics}$$

$$\boxed{U = \begin{bmatrix} u_0 \\ ... \\ u_{T-1} \end{bmatrix} \sim \pi_{\mathrm{ol}}(\cdot|x_0) \quad \begin{array}{l} \pi_{\mathrm{ol}}(\cdot|x_0) \in M(\mathbb{R}^{m_{\mathrm{ol}}}) \\ m_{\mathrm{ol}} = mT \end{array}}$$

NP hard

$$\Pr \left\{ \begin{array}{c} g_1(x_1) \le 0 \\ ... \\ g_t(x_t) \le 0 \\ ... \\ g_{T-1}(x_{T-1}) \le 0 \\ g_T(x_T) \le 0, \end{array} \right\} \ge 1 - \alpha, \qquad \text{Chance constraint}$$

Probability level $\alpha \in [0,1]$

$$u_t \in [u_{\min}, u_{\max}], t = 0, 1, ..., T-1. \qquad \text{Normal constraint}$$

6

Problem Formulation

- Chance Constrained Linear Programs in Probability Measure Space (CCLP in PMS)

Let $\mathcal{Z} \subset \mathbb{R}^{n_z}$ be a compact set with the infinite norm defined by $\|z\|_\infty = \max_{i=1,\dots,n_z} |z_i|,\ z \in \mathcal{Z}$.

Diameter of $\mathcal{Z}$: $D = \sup\{\|z - z'\|_\infty : z, z' \in \mathcal{Z}\}$.

Random vector: $\delta \in \Delta \subseteq \mathbb{R}^s$ Probability density function: $p(\delta)$

$$(\mathcal{P}_{\mu,\alpha}) \qquad \min_{\mu \in M(\mathcal{Z})} \int_{\mathcal{Z}} J(z) \mathrm{d}\mu(z)$$

$$\text{s.t.} \quad \int_{\mathcal{Z}} \mathrm{d}\mu(z) = 1,$$

$$\int_{\mathcal{Z}} \int_{\Delta} \boxed{\mathbb{I}\{h(z,\delta)\}} \mathrm{d}p(\delta) \mathrm{d}\mu(z) \geq 1 - \alpha$$

$J : \mathcal{Z} \to \mathbb{R}$ continuous and differentiable

$\mathbb{I}\{y\} = 1,$ if $y \leq 0$

$\mathbb{I}\{y\} = 0,$ if $y > 0$

$h : \mathcal{Z} \times \Delta \to \mathbb{R}^m$

- For every δ, $h(\cdot, \delta)$ is continuous and differentiable;
- For every z, $h(z, \delta)$ has a continuous PDF.

7

Problem Formulation

- Chance Constrained Linear Programs in Probability Measure Space (CCLP in PMS)

$$(\mathcal{P}_{\mu,\alpha}) \qquad \min_{\mu \in M(\mathcal{Z})} \int_{\mathcal{Z}} J(z) \mathrm{d}\mu(z)$$

$$\text{s.t.} \quad \int_{\mathcal{Z}} \mathrm{d}\mu(z) = 1,$$

$$\int_{\mathcal{Z}} \int_{\Delta} \mathbb{I}\{h(z,\delta)\} \mathrm{d}p(\delta) \mathrm{d}\mu(z) \geq 1 - \alpha$$

Feasible region: $M_\alpha(\mathcal{Z}) = \{\mu \in M(\mathcal{Z}) : \int_{\mathcal{Z}} \int_{\Delta} \mathbb{I}\{h(z,\delta)\} \mathrm{d}p(\delta) \mathrm{d}\mu(z) \geq 1 - \alpha\}$

Optimal objective function: $\bar{J}_{\mu,\alpha} := \min_{\mu \in M_\alpha(\mathcal{Z})} \int_{\mathcal{Z}} J(z) d\mu(z)$

Optimal solution set: $A_{\mu,\alpha} := \{\mu \in M_\alpha(\mathcal{Z}) : \int_{\mathcal{Z}} J(z) \mathrm{d}\mu(z) = \bar{J}_{\mu,\alpha}\}$

Optimal Measure: $\bar{\mu}_{\mu,\alpha} \in A_{\mu,\alpha}$

8

Proposed Method

- CCLP in PMS v.s. chance constrained optimization (CCO) in finite space

$$(\mathcal{P}_{z,\alpha}) \qquad \min_{z\in\mathcal{Z}} \ J(z)$$
$$\text{s.t.} \ \ \Pr\{h(z,\delta)\le 0\}\ge 1-\alpha.$$

Feasible region:

$\mathcal{Z}_\alpha := \{z\in\mathcal{Z} : \Pr\{h(z,\delta)\le 0\}\ge 1-\alpha\}.$

Optimal solution set: $A_{z,\alpha} := \{z\in\mathcal{Z}_\alpha : J(z)=\bar{J}_{z,\alpha}\}.$

Optimal objective function:

$\bar{J}_{z,\alpha} := \min\{J(z) : z\in\mathcal{Z}_\alpha\}.$

Optimal solution: $\bar{z}_\alpha \in A_{z,\alpha}.$

Theorem $(\mathcal{P}_{\mu,\alpha})$ and $(\mathcal{P}_{z,\alpha})$

The optimal objective values of $(\mathcal{P}_{\mu,\alpha})$ and $(\mathcal{P}_{z,\alpha})$ satisfies

$$\bar{J}_{\mu,\alpha}\le \bar{J}_{z,\alpha},\ \ \forall\alpha\in[0,1].$$

Besides, if $\alpha=0$, with probability 1, we have

$$\bar{J}_{\mu,0}=\bar{J}_{z,0} \ \text{ and } \ A_{\mu,0}=\{\mu\in M(\mathcal{Z}) : \mu(\mathcal{Z})=\mu(A_{z,0})\}\ .$$

9

Proposed Method

- Sample-based approximation

$$(\mathcal{P}^S_{\mu,\alpha}) \quad \min_{\mu\in\mathbb{R}^S} \ \sum_{i=1}^S J(z^{(i)})\mu(i)$$

Randomly extracted z : $\mathcal{Z}_S=\{z^{(1)},\dots,z^{(S)}\}.$

$$\text{s.t.} \ \ \sum_{i=1}^S \mu(i)=1,\ \mu(i)\ge 0,\ \forall i=1,\dots,S, \qquad \int_\Delta \sum_{i=1}^S \mu(i)\mathbb{I}\{h(z^{(i)},\delta)\}p(\delta)\mathrm{d}\delta\ge 1-\alpha.$$

Feasible region:

$\mathcal{F}^S_{\mu,\alpha} := \{\mu\in\mathbb{R}^S : \int_\Delta \sum_{i=1}^S \mu(i)\mathbb{I}\{h(z^{(i)},\delta)\le 0\}\mathrm{d}p(\delta)\ge 1-\alpha\}.$

Optimal solution set: $A^S_{\mu,\alpha} := \{\mu\in\mathcal{F}^S_{\mu,\alpha} : \sum_{i=1}^S J(z^{(i)})\mu(i)=\bar{J}^S_{\mu,\alpha}\}.$

Optimal objective function:

$\bar{J}^S_{\mu,\alpha} := \min_{\mu\in\mathcal{F}^S_{\mu,\alpha}} \sum_{i=1}^S J(z^{(i)})\mu(i).$

Optimal solution: $\bar{\mu}^S_{\mu,\alpha}\in A^S_{\mu,\alpha}.$

Theorem $(\mathcal{P}_{\mu,\alpha})$ and $(\mathcal{P}^S_{\mu,\alpha})$

As $S\to\infty$, with probability 1,

(a) For any discrete probability measure $\mu^S\in\mathcal{F}^S_{\mu,\alpha}$, we have $\mu^S\in M_\alpha(\mathcal{Z})$;

(b) We have

$$\lim_{S\to\infty}\inf \bar{J}^S_{\mu,\alpha}=\bar{J}_{\mu,\alpha}.$$

10

Proposed Method

- Sample-based approximation

Randomly extracted z: $\mathcal{Z}_S = \{z^{(1)}, ..., z^{(S)}\}$.
Randomly extracted δ: $\Delta_N = \{\delta^{(1)}, ..., \delta^{(N)}\}$.

$$(\mathcal{P}^{S,N}_{\mu,\alpha}) \quad \min_{\mu \in \mathbb{R}^S} \sum_{i=1}^{S} J(z^{(i)})\mu(i)$$

$$\text{s.t.} \quad \sum_{i=1}^{S} \mu(i) = 1, \ \mu(i) \geq 0, \ \forall i = 1, ..., S, \quad \sum_{i=1}^{S} \mu(i) \frac{1}{N} \sum_{j=1}^{N} \mathbb{I}\{h(z^{(i)}, \delta^{(j)})\} \geq 1 - \alpha.$$

Feasible region:

$$\mathcal{F}^{S,N}_{\mu,\alpha} := \{\mu \in \mathbb{R}^S : \sum_{i=1}^{S} \mu(i) \frac{1}{N} \sum_{j=1}^{N} \mathbb{I}\{h(z^{(i)}, \delta^{(j)})\} \geq 1 - \alpha\}.$$

Optimal objective function:

$$\bar{J}^{S,N}_{\mu,\alpha} := \min_{\mu \in \mathcal{F}^{S,N}_{\mu,\alpha}} \sum_{i=1}^{S} J(z^{(i)})\mu(i).$$

Optimal solution set: $A^{S,N}_{\mu,\alpha} := \{\mu \in \mathcal{F}^{S,N}_{\mu,\alpha} : \sum_{i=1}^{S} J(z^{(i)})\mu(i) = \bar{J}^{S,N}_{\mu,\alpha}\}$.

Optimal solution: $\bar{\mu}^{S,N}_{\mu,\alpha} \in A^{S,N}_{\mu,\alpha}$.

Theorem $(\mathcal{P}^{S}_{\mu,\alpha})$ and $(\mathcal{P}^{S,N}_{\mu,\alpha})$

As $N \to \infty$, with probability 1,

$$\bar{J}^{S,N}_{\mu,\alpha} \to \bar{J}^{S}_{\mu,\alpha}, \text{ and } A^{S,N}_{\mu,\alpha} \to A^{S}_{\mu,\alpha}.$$

11

Proposed Method

- Sample-based approximation

$$\boxed{(\mathcal{P}^{S,N}_{\mu,\alpha})} \xrightarrow{N \to \infty,} (\mathcal{P}^{S}_{\mu,\alpha}) \xrightarrow{S \to \infty,} (\mathcal{P}_{\mu,\alpha})$$

Solvable | NP hard | NP hard

Theorem $(\mathcal{P}_{\mu,\alpha})$ and $(\mathcal{P}^{S,N}_{\mu,\alpha})$

As $S, N \to \infty$, with probability 1,

$$\lim_{S,N \to \infty} \inf \bar{J}^{S,N}_{\mu,\alpha} = \bar{J}_{\mu,\alpha}, \text{ and } A^{S,N}_{\mu,\alpha} \subset M_\alpha(\mathcal{Z}).$$

12

Proposed Method

- Gaussian mixture model(GMM) based approximation

$$(\mathcal{P}^{\theta,N}_{\mu,\alpha}) \quad \min_{\theta\in\Theta} \int_{\mathcal{Z}} J(z)p_\theta(z)\mathrm{d}z$$
$$\text{s.t.} \quad \int_{\mathcal{Z}} \sum_{j=1}^{N} \frac{1}{N}\mathbb{I}\{h(z,\delta^{(i)})\}p_\theta(z)\mathrm{d}z \geq 1-\alpha.$$

Randomly extracted δ: $\Delta_N = \{\delta^{(1)},\ldots,\delta^{(N)}\}$.

About PDF of GMM:

$$p_\theta(z) = \sum_{i=1}^{L} \omega_i \phi(z, m_i, \Sigma_i). \qquad \Theta = \{\theta \in \mathbb{R}^{n_\theta} : \sum_{i=1}^{L} \omega_i = 1, \omega_i \geq 0\}.$$

Feasible region:

$$\Theta^N_\alpha = \{\theta \in \Theta : \int_{\mathcal{Z}} \sum_{j=1}^{N} \frac{1}{N}\mathbb{I}\{h(z,\delta^{(i)})\}p_\theta(z)\mathrm{d}z \geq 1-\alpha\}.$$

Optimal objective function:

$$\bar{J}^{\theta,N}_{\mu,\alpha} := \min_{\theta\in\Theta^N_\alpha} \int_{\mathcal{Z}} J(z)p_\theta(z)dz.$$

Optimal solution set: $A^{\theta,N}_{\mu,\alpha} = \{\theta \in \Theta^N_\alpha : \int_{\mathcal{Z}} J(z)p_\theta(z)dz = \bar{J}^{\theta,N}_{\mu,\alpha}$. Optimal solution: $\bar{\theta}^N_\alpha \in A^{\theta,N}_{\mu,\alpha}$.

13

Proposed Method

- Gaussian mixture model(GMM) based approximation

$(\mathcal{P}^{\theta,N}_{\mu,\alpha})$

Solvable

$$\xrightarrow[p_\theta(z) = \sum_{i=1}^{L} \omega_i \phi(z, m_i, \Sigma_i).]{N\to\infty, \quad L\to\infty.}$$

$(\mathcal{P}_{\mu,\alpha})$

NP hard

Theorem $(\mathcal{P}_{\mu,\alpha})$ and $(\mathcal{P}^{\theta,N}_{\mu,\alpha})$

As $N, L \to \infty$, with probability 1,

$$\lim_{L,N\to\infty} \inf \bar{J}^{\theta,N}_{\mu,\alpha} = \bar{J}_{\mu,\alpha},$$
$$\bar{\mu}_{\bar{\theta}^N_\alpha} \in M_\alpha(\mathcal{Z}).$$
$$\bar{\mu}_{\bar{\theta}^N_\alpha} := \int_Z p_{\bar{\theta}^N_\alpha}(z)\mathrm{d}z. \ \forall Z \subseteq \mathcal{Z}.$$

14

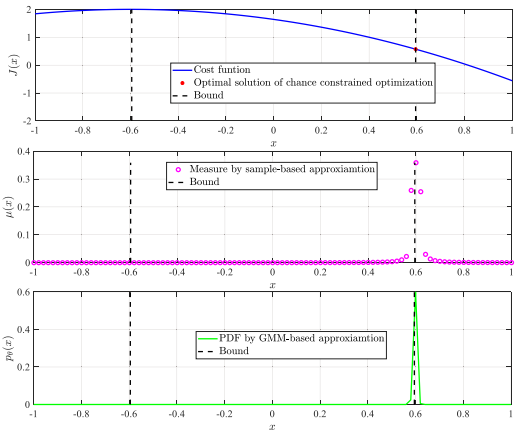
Numerical Examples

- Example 1: one dimension example

(a)

(b)

(c)



$$J(z) = -(z+0.6)^2 + 2$$

$$\mathcal{Z} = \{z \in \mathbb{R} : z \in [-1, 1]\}.$$

$$h(z, \delta) = z^2 + \delta - 2$$

$$\delta \sim \mathcal{N}(0, 1)$$

$$\bar{J}_{z,0.05} = 0.572$$

$$\bar{J}^{200,2000}_{\mu,0.05} = 0.5602$$

$$\bar{J}^{\theta,2000}_{\mu,0.05} = 0.5615$$

15

Numerical Examples

- Quadrotor system control

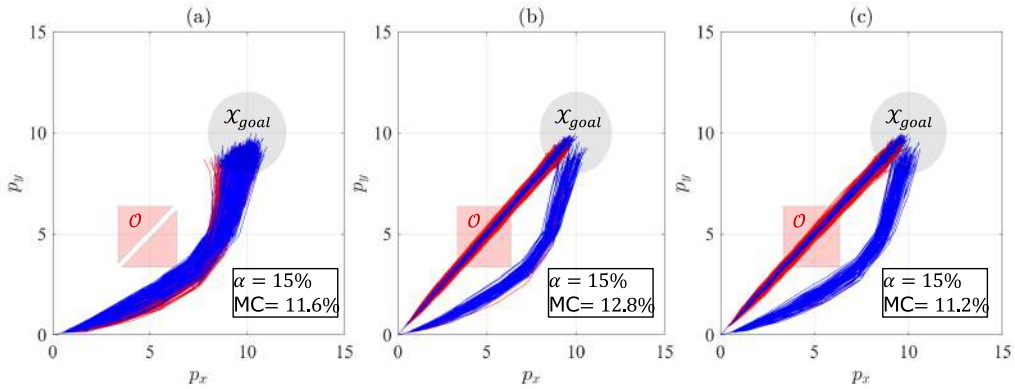
$$\begin{aligned} \min_{\mu \in M(\mathcal{U}^{\mathcal{N}})} \quad & \mathbb{E}\{\ell^x(x) + \ell^u(u)\} \\ \text{s.t.} \quad & x_{t+1} = Ax_t + B(m)u_t + d(x_t, \varphi) + \omega_t, \ u \sim M(\mathcal{U}^{\mathcal{N}}), \\ & t = 0, 1, \ldots, N-1, \\ & \Pr\{(\wedge_{t=1}^{N-1} x_t \notin \mathcal{O}_t) \wedge (x_N \in \mathcal{X}_{\text{goal}})\} \geq 1 - \alpha, \end{aligned} \qquad (P_{\text{QSC}})$$

$$A = \begin{bmatrix} 1 & \Delta t & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & \Delta t \\ 0 & 0 & 0 & 1 \end{bmatrix}, \ B(m) = \frac{1}{m} \begin{bmatrix} \Delta t^2/2 & 0 \\ \Delta t & 0 \\ 0 & \Delta t^2/2 \\ 0 & \Delta t \end{bmatrix}, \ d(x_t, \varphi) = -\varphi \begin{bmatrix} \Delta t^2 |v_x| v_x/2 \\ \Delta t |v_x| v_x \\ \Delta t^2 |v_y| v_y/2 \\ \Delta t |v_y| v_y \end{bmatrix}.$$

16

Numerical Examples

- Quadrotor system control



17

Conclusion and Future Work

Conclusion

- Formulate CCLS in PMS from chance constrained optimal control (CCOC)
- Proposed sample-based approximate problem of CCLS in PMS
- Proposed GMM-based approximate problem of CCLS in PMS

Future work

- To overcome the dimension curse of sample-based approximation
- Fast algorithm to obtain parameters in GMM-based approximation
- Establish closed loop feedback stochastic optimal policy for CCOC

18

Tokyo Tech

Thanks for Your Kind Attentions!

A generalized Levenberg–Marquardt method for large-scale composite minimization

Naoki MARUMO

University of Tokyo / NTT, Japan
marumo-naoki@g.ecc.u-tokyo.ac.jp

We propose a new generalized Levenberg–Marquardt method for minimizing the sum of a smooth composite function and a convex function. The method enjoys three theoretical guarantees: iteration complexity bound, oracle complexity bound, and local convergence under an error bound condition [1]. Numerical results show that the proposed method performs well for some large-scale problems.

References

[1] N. Yamashita and M. Fukushima. On the rate of convergence of the Levenberg–Marquardt method. In G. Alefeld and X. Chen, editors, Topics in Numerical Analysis, pages 239–249, Vienna, 2001. Springer Vienna.

Sep 19, 2022

6th RIKEN-IMI-ISM-NUS-ZIB-MODAL-NHR Workshop

A generalized Levenberg–Marquardt method for large-scale composite minimization

Naoki Marumo (NTT / UTokyo, Japan)
joint work with Takayuki Okuno and Akiko Takeda

Outline: Problem setting & contribution

Composite minimization

$$\min_{x \in \mathbb{R}^d} \quad g(x) + h(c(x))$$

- $g : \mathbb{R}^d \to \mathbb{R} \cup \{+\infty\}$: convex
  $c : \mathbb{R}^d \to \mathbb{R}^n$: smooth (Lipschitz Jacobian)
  $h : \mathbb{R}^n \to \mathbb{R}$: convex & smooth (Lipschitz gradient)
- Nonconvex optimization problem with many applications
- **Levenberg–Marquardt (LM) method:** efficient and widely used (e.g., implemented in `MATLAB` and `SciPy`)
- We propose a new LM method with both an **oracle complexity** bound and a **local quadratic** convergence guarantee ☺

Application: Empirical risk minimization in ML

$$\min_{x \in \mathbb{R}^d} \quad \frac{1}{N} \sum_{i=1}^{N} \ell(\phi_x(a_i), b_i)$$

- $(a_i, b_i) \in \mathbb{R}^p \times \mathbb{R}^q$: training data $(i = 1, \ldots, N)$
- $\phi_x : \mathbb{R}^p \to \mathbb{R}^q$: machine learning model with parameter x
- $\ell : \mathbb{R}^q \times \mathbb{R}^q \to \mathbb{R}$: loss function

- $c(x) = (\phi_x(a_1), \ldots, \phi_x(a_N)) \in \mathbb{R}^{q \times N}$
- $h(y) = h(y_1, \ldots, y_N) = \frac{1}{N} \sum_{i=1}^{N} \ell(y_i, b_i)$

$$\min_{x \in \mathbb{R}^d} \quad h(c(x))$$

Levenberg–Marquardt (LM) method

- Iterative method that **uses the composite structure** of the problem
- Construct a subproblem for the k-th iterate x^k.
  Set x^{k+1} to be an (approximate) solution to the subproblem

Subproblem for LM

linear approx. of $c(x)$ damping term $(\mu > 0)$

$$\min_{x \in \mathbb{R}^d} \quad g(x) + h\Big(c(x^k) + \nabla c(x^k)(x - x^k)\Big) + \frac{\mu}{2}\|x - x^k\|_2^2$$

cf. original problem

$$\min_{x \in \mathbb{R}^d} \quad g(x) + h(c(x))$$

- $\nabla c(x^k) \in \mathbb{R}^{n \times d}$: Jacobian matrix
- The subproblem is strongly convex and much easier than the original

Levenberg–Marquardt (LM) method

- Introduced for least-squares problems $(g(\cdot) = 0,\ h(\cdot) = \frac{1}{2}\|\cdot\|_2^2)$
  [Levenberg, 1944, Marquardt, 1963]

- Extended to general g, h
  [Nesterov, 2007, Lewis and Wright, 2016, Drusvyatskiy and Lewis, 2018]

- Many other LM methods
  [Osborne, 1976, Yamashita and Fukushima, 2001, Dan et al., 2002, Kanzow et al., 2004, Ueda and Yamashita, 2010, Behling and Fischer, 2012, Drusvyatskiy and Paquette, 2019, Bergou et al., 2020, Marumo et al., 2020]...

Main differences between LMs

Assumptions on g, h

How to set the damping parameter μ

Algorithm for subproblems

Theoretical guarantees

Features of our LM

Assumptions on g, h

$\rightarrow$ g: convex, h: smooth convex. Not restricted to least squares

How to set the damping parameter μ

$\rightarrow$ Adaptively

Algorithm for subproblems

$\rightarrow$ **Accelerated proximal gradient** with a particular termination cond.

Theoretical guarantees

$\rightarrow$ **Iteration** complexity, **oracle** complexity,
local **quadratic** convergence (under additional assumptions)

First LM to achieve both oracle complexity and local quadratic conv.

cf.: iteration and oracle complexity

Iteration complexity

The number of iterations required to find an ε-stationary point

- Used when we, at each iteration,

  ❶ compute the Jacobian $\nabla c(x^k) \in \mathbb{R}^{n \times d}$ and

  ❷ solve the subproblem using it,

  and the cost of ❶ is dominant

- Then,

  (Tocal cost) $\simeq$ **(Iteration complexity)** $\times$ **(Cost for $\nabla c(x^k)$)**

- Mainly for small- or medium-scale problems $(d, n \lesssim 10^3)$

cf.: iteration and oracle complexity

Oracle complexity

The number of **oracle calls** required to find an ε-stationary point

Oracles assumed in this work

- $c(x),\ \nabla c(x)u,\ \nabla c(x)^\top v$ $\qquad (x, u \in \mathbb{R}^d,\ v \in \mathbb{R}^n)$
- $h(y),\ \nabla h(y)$ $\qquad (y \in \mathbb{R}^n)$
- $\mathrm{prox}_{\eta g}(x) := \operatorname*{argmin}_{z \in \mathbb{R}^d} \{\eta g(z) + \frac{1}{2}\|x - z\|^2\}$ $\qquad (x \in \mathbb{R}^d,\ \eta > 0)$

- Used when we, at each iteration,
  - do not compute $\nabla c(x^k) \in \mathbb{R}^{n \times d}$ explicitly and
  - solve the subproblem using $\nabla c(x^k)u,\ \nabla c(x^k)^\top v$
- Mainly for medium- or large-scale problems $(d, n \gtrsim 10^3)$

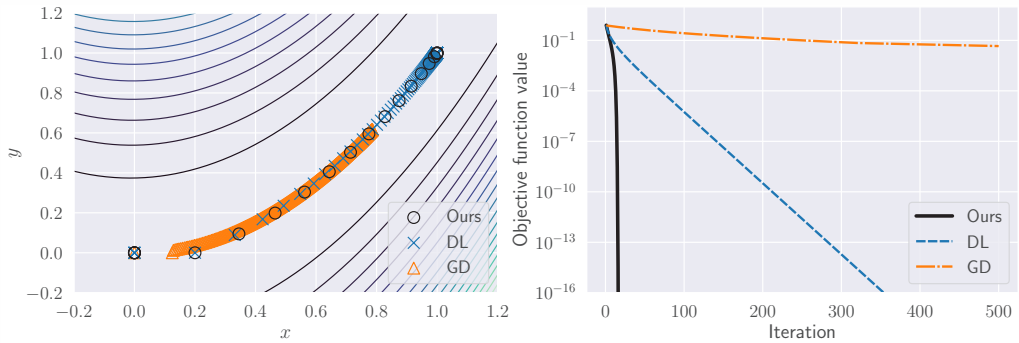
Comparison of theoretical guarantees

	General g, h	#Iteration	#Oracle	Local
Existing LM 1 [Yamashita and Fukushima, 2001]				✓
2 [Ueda and Yamashita, 2010]		$O(\sqrt{\Delta}\kappa)$		
3 [Marumo et al., 2020]		$O(\sqrt{\Delta}\kappa)$	$O(\sqrt{\Delta}\kappa)$	
4 [Marumo et al., 2020]		$O(\sqrt{\Delta}\kappa)$		✓
5 [Drusvyatskiy and Paquette, 2019]	✓	$O(K_h)$	$\tilde{O}(K_h\sqrt{\kappa'})$	
Our LM	✓✓	$O(\sqrt{\Delta})$	$\tilde{O}(\sqrt{\Delta}\sqrt{\kappa})$	✓
cf.: Prox. grad. 1	✓	$O(K_h\kappa')$	$O(K_h\kappa')$	
2	✓✓	$O(\sqrt{\Delta}\kappa)$	$O(\sqrt{\Delta}\kappa)$	

- $\kappa, \kappa' \geq 1$: constants like a condition number
- K_h, L_h: Lipschitz constants of $h, \nabla h$. Normalized to $L_h = 1$
- $\Delta := g(x_0) + h(c(x_0)) - \left(\min_{x \in \mathbb{R}^d} g(x) + \min_{y \in \mathbb{R}^n} h(y)\right)$

Numerical example: Rosenbrock function

$$\min_{(x,y)\in\mathbb{R}^2} \quad (x-1)^2 + 100(x^2-y)^2$$



Our LM converges faster than **existing LM** and **Gradient Descent** ☺

Previous parts of this talk:

- Problem setting
- LM method
- Our contribution & comparison with existing work

Remaining parts:

- Details of the proposed LM and theoretical guarantees
- Numerical experiments with large-scale problems

Proposed LM method

k-th iteration of our LM

❶ $\mu_k := \rho\sqrt{F(x^k) - (g^* + h^*)}$ (ρ: constant)

❷ $x^{k+1} \in \left\{ x \in \mathbb{R}^d \;\middle|\; \bar{\omega}_{k,\mu_k}(x) \leq \frac{\mu_k}{2}\|x - x^k\|_2 \right\}$

- $F(x) := g(x) + h(c(x)), \quad g^* := \min_{x \in \mathbb{R}^d} g(x), \quad h^* := \min_{y \in \mathbb{R}^n} h(y)$
- $\bar{\omega}_{k,\mu_k}(x)$: (sub)gradient norm for the subproblem
- ❷ is computed by **accelerated proximal gradient**

Key point of the proposed LM method

k-th iteration of our LM

❶ $\mu_k := \rho\sqrt{F(x^k) - (g^* + h^*)}$ (ρ: constant)

❷ $x^{k+1} \in \left\{ x \in \mathbb{R}^d \;\middle|\; \bar{\omega}_{k,\mu_k}(x) \leq \frac{\mu_k}{2}\|x - x^k\|_2 \right\}$

Difficulty on μ:

- small μ $\rightarrow$ worsen cond. number of subproblem $\rightarrow$ increase #oracle
- large μ $\rightarrow$ fail quadratic convergence

Difficulty on the accuracy of subproblem's solution:

- accurate $\rightarrow$ increase costs per iteration $\rightarrow$ increase #oracle
- inaccurate $\rightarrow$ fail quadratic convergence & increase #iteration

❶ and ❷ strike a good balance!

Theoretical guarantee: Complexity (Main result 1)

Complexity for an ε-stationary point

- **Iteration complexity:** $O\left(\frac{L_c\sqrt{L_h\Delta}}{\varepsilon^2}(F(x_0)-F^*)\right)$
- **Oracle per iteration:** $O(\sqrt{\kappa}\log\kappa)$
- **Oracle complexity:** $O\left(\frac{L_c\sqrt{L_h\Delta}}{\varepsilon^2}(F(x_0)-F^*)\sqrt{\kappa}\log\kappa\right)$

- ε-stationary point: a point $x\in\mathbb{R}^d$ s.t. $\omega(x)\le\varepsilon$

$$\omega(x) := \min_{p\in\partial g(x)}\left\|p+\nabla c(x)^\top\nabla h(c(x))\right\|_2$$

- L_c, L_h: Lipschitz constants of $\nabla c, \nabla h$, $\quad \kappa := 1+\frac{\sqrt{L_h}}{L_c\sqrt{\Delta}}\sup_{k\in\mathbb{N}}\|\nabla c(x^k)\|_{\mathrm{op}}^2$

- $\Delta := F(x_0)-(g^*+h^*)$, $\quad F^* := \min_{x\in\mathbb{R}^d}F(x)$

Theoretical guarantee: Local quad. conv. (Main result 2)

Assume:

- **Zero-residual:** $F(x^*)=g^*+h^*$ for some $x^*\in\mathbb{R}^d$,
- x^0 is sufficiently close to x^*,
- **Error bound condition**.

Then, $(F(x^k))$ converges to $F_\infty := g^*+h^*$. Futhermore, for some $C>0$,

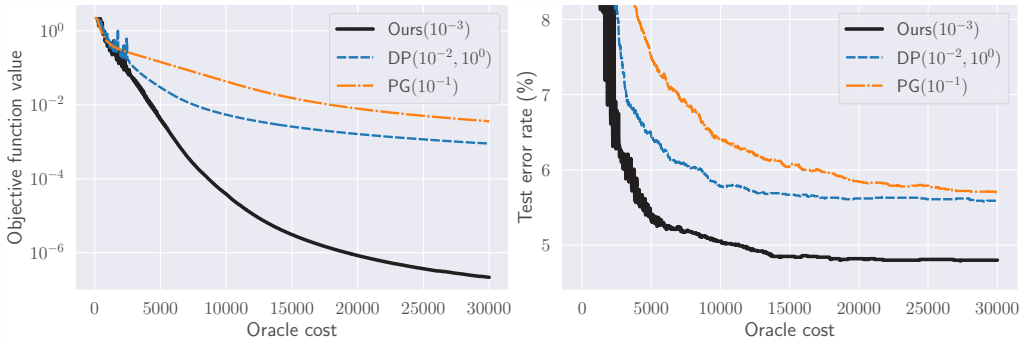
$$F(x^{k+1})-F_\infty \le C\big(F(x^k)-F_\infty\big)^2, \quad \forall k\ge 0.$$

Def.: Error bound condition

For some $\gamma>0$, $\quad \frac{\gamma}{2}\operatorname{dist}(x^k,X^*)^2 \le F(x^k)-(g^*+h^*), \quad \forall k\ge 0.$

- $\operatorname{dist}(x^k,X^*)$: distance between x^k and the optimal solution set X^*
- Weaker than strong convexity

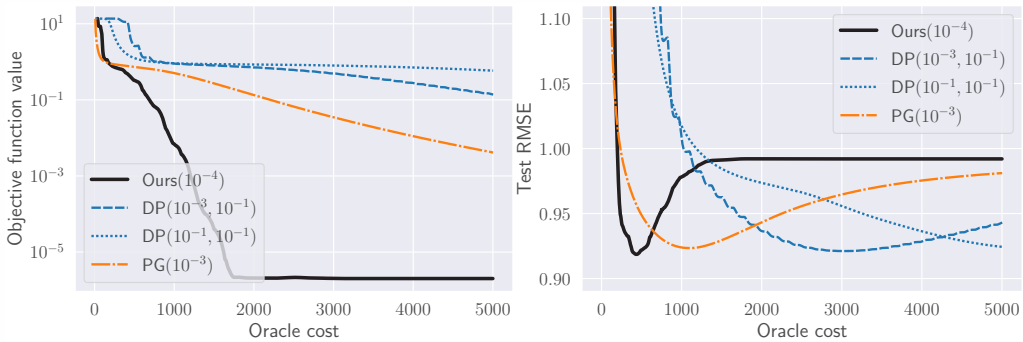
Numerical result 1: Classification with NN (MNIST)



$$\min_{x \in \mathbb{R}^d} \quad -\frac{1}{N} \sum_{i=1}^{N} \langle b_i, \mathrm{LogSoftmax}(\phi_x(a_i)) \rangle$$

- $(a_i, b_i) \in [0,1]^{784} \times \{0,1\}^{10}$: training data. Pair of image and label
- $\phi_x : \mathbb{R}^{784} \to \mathbb{R}^{10}$: neural network with parameter x
- $d = 104{,}938, \quad n = 10N = 60{,}000$

Numerical result 2: Nonneg. Matrix Factr. (MovieLens)



$$\min_{U \in \mathbb{R}_+^{p \times r},\, V \in \mathbb{R}_+^{q \times r}} \quad \frac{1}{|\Omega|} \sum_{(i,j,s) \in \Omega} \big(\langle u_i, v_j \rangle - s\big)^2 + \lambda\big(\|U\|_{\mathrm{F}}^2 + \|V\|_{\mathrm{F}}^2\big)$$

- Ω: training dataset. $(i,j,s) \in \Omega$: user i rated movie j as $s \in \{1, \ldots, 5\}$
- $\lambda = 10^{-10}$: regularization parameter
- $d = pr + qr = 1{,}312{,}500, \quad n = |\Omega| = 80{,}000$

Conclusion

$$\min_{x \in \mathbb{R}^d} \quad g(x) + h(c(x))$$

Our LM has **iteration** and **oracle** complexity bounds and a **local quadratic** convergence guarantee

k-th iteration of our LM

1. $\mu_k := \rho\sqrt{F(x^k) - (g^* + h^*)}$ (ρ: constant)
2. $x^{k+1} \in \left\{ x \in \mathbb{R}^d \;\middle|\; \bar{\omega}_{k,\mu_k}(x) \leq \frac{\mu_k}{2}\|x - x^k\|_2 \right\}$

- Subproblems are solved by an accelerated proximal gradient method
- Parameter μ and the accuracy of subproblem's solution are carefully set
- Our LM is practical for large-scale problems ($d \simeq 10^5$–10^6)

Appendix: Lemmas for complexity bounds

Iteration complexity is derived from Lemma 1, and
oracle complexity is derived from Lemmas 1 and 2

Lemma 1 (this work)

$$F(x^{k+1}) \leq F(x^k) - \frac{\mu_k}{4}\|x^{k+1} - x^k\|_2^2, \quad \forall k \geq 0$$

Lemma 2 (this work)

Oracle complexity for 2 is

$$O(\sqrt{\kappa_k}\log\kappa_k), \quad \kappa_k := 1 + \frac{L_h}{\mu_k}\|\nabla c(x^k)\|_{\mathrm{op}}^2$$

Appendix: Example of zero-residual problem

- $f : \mathbb{R}^d \to \mathbb{R}^n$: given smooth & nonlinear function, $n < d$
- $x^* \in \mathbb{R}^d$: unknown signal, **sparse**
- $y := f(x^*)$: observation

Goal: Recover the sparse signal x^* from the low-dimensional observation y

Nonlinear compressed sensing

$$\min_{x \in \mathbb{R}^d} \ \|y - f(x)\|_2^2, \qquad \text{s.t. } \|x\|_1 \le r$$

- The constraint $\|x\|_1 \le r$ enhances the sparsity of x
- This problem is composite minimization and **zero-residual**

References I

R. Behling and A. Fischer. A unified local convergence analysis of inexact constrained Levenberg–Marquardt methods. *Optimization Letters*, 6(5):927–940, 2012.

E. H. Bergou, Y. Diouane, and V. Kungurtsev. Convergence and complexity analysis of a Levenberg–Marquardt algorithm for inverse problems. *Journal of Optimization Theory and Applications*, 185(3):927–944, 2020.

H. Dan, N. Yamashita, and M. Fukushima. Convergence properties of the inexact Levenberg–Marquardt method under local error bound conditions. *Optimization Methods and Software*, 17(4):605–626, 2002.

D. Drusvyatskiy and A. S. Lewis. Error bounds, quadratic growth, and linear convergence of proximal methods. *Mathematics of Operations Research*, 43(3):919–948, 2018.

D. Drusvyatskiy and C. Paquette. Efficiency of minimizing compositions of convex functions and smooth maps. *Mathematical Programming*, 178(1-2):503–558, 2019.

C. Kanzow, N. Yamashita, and M. Fukushima. Levenberg–Marquardt methods with strong local convergence properties for solving nonlinear equations with convex constraints. *Journal of Computational and Applied Mathematics*, 172(2):375–397, 2004.

K. Levenberg. A method for the solution of certain non-linear problems in least squares. *Quarterly of Applied Aathematics*, 2(2):164–168, 1944.

References II

A. S. Lewis and S. J. Wright. A proximal method for composite minimization. *Mathematical Programming*, 158(1):501–546, 2016.

D. W. Marquardt. An algorithm for least-squares estimation of nonlinear parameters. *Journal of the Society for Industrial and Applied Mathematics*, 11(2):431–441, 1963.

N. Marumo, T. Okuno, and A. Takeda. Majorization-minimization-based Levenberg–Marquardt method for constrained nonlinear least squares. *arXiv preprint arXiv:2004.08259*, 2020.

Y. Nesterov. Modified Gauss–Newton scheme with worst case guarantees for global performance. *Optimisation Methods and Software*, 22(3):469–483, 2007.

M. R. Osborne. Nonlinear least squares—the Levenberg algorithm revisited. *The Journal of the Australian Mathematical Society. Series B. Applied Mathematics*, 19(3): 343–357, 1976.

K. Ueda and N. Yamashita. On a global complexity bound of the Levenberg–Marquardt method. *Journal of Optimization Theory and Applications*, 147(3):443–453, 2010.

N. Yamashita and M. Fukushima. On the rate of convergence of the Levenberg–Marquardt method. In G. Alefeld and X. Chen, editors, *Topics in Numerical Analysis*, pages 239–249, Vienna, 2001. Springer Vienna.

BIPSOL: A metaheuristic solver for large-scale binary integer programs

Shunji UMETANI

Graduate School of Information Science and Technology,
Osaka University, Japan
umetani@ist.osaka-u.ac.jp

Metaheuristics have proven to a comprehensive approach to attain good solutions for hard combinatorial optimization problems. However, they are usually based on specific characteristics of the problem to be solved, which makes them hard to develop efficient general purpose solvers for such as the mixed integer programs (MIPs) and the constraint satisfaction problems (CSPs). In designing metaheuristics for combinatorial optimization problems, the quality of solutions typically improves if larger and sophisticated neighborhoods are used, while computation time of searching the neighborhood also increases rapidly. BIPSOL is a metaheuristic solver for large-scale binary integer programs (BIPs) that introduces a generalized technique of the neighbor-list used for traveling salesman problem (TSP) to generate smaller and structured neighborhoods automatically [1,2]. We incorporate an efficient incremental evaluation of solutions and a dynamic control mechanisms of penalty weights into BIPSOL. In this talk, we show some progress of development in BIPSOL and future directions.

References

[1] Shunji Umetani, Exploiting variable associations to configure efficient local search algorithm in large-scale binary integer programs, European Journal of Operational Research, 263 (2017), 72-81. https://doi.org/10.1016/j.ejor.2017.05.025

[2] Shunji Umetani, Exploiting variable associations to configure efficient local search algorithm in large-scale set partitioning problems, Proceedings of 29th AAAI Conference on Artificial Intelligence (AAAI-15), 1226-1232.

BIPSOL: A metaheuristic solver for large-scale binary integer programs

Shunji Umetani
Osaka University, JAPAN

6th RIKEN-IMI-ISM-NUS-ZIB-MODAL-NHR Workshop
on Advances in Classical and Quantum Algorithms for Optimization and Machine Learning

Overview

- We have developed a metaheuristic solver called BIPSOL for large-scale binary integer program up to several millions of variables.
  - ✓ S. Umetani, Exploiting variable associations to configure efficient local search algorithm in large-scale binary integer programs, European Journal of Operational Research, **263** (2017), 72-81. (open access)
  - ✓ S. Umetani, Exploiting variable associations to configure efficient local search algorithm in large-scale set partitioning problems, Proc. of 29th AAAI Conference on Artificial Intelligence (AAAI-15), 1226-1232.
- The proposed algorithm was implemented as the WLS solver in the Nuorium Optimizer, a mathematical optimization package of NTT DATA Mathematical Systems, Inc.
- We review the concept and implementation of the proposed metaheristic solver and discuss some current technical issues.

2

Binary integer program

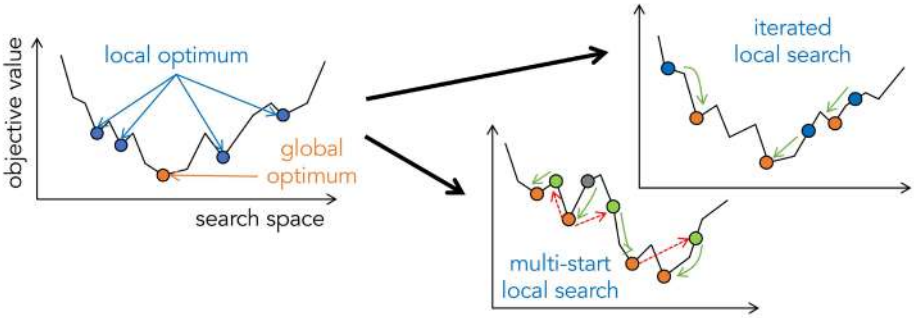
- The binary integer program (BIP) is a general model having many real-world applications, such as crew scheduling, vehicle routing, facility location, and so on.
- Based on linear programming (LP) techniques, many exact and heuristic algorithms have been developed.
- Many large-scale BIP instances still remain due to a large gap between the lower and upper bounds of the optimal values.

$$
\begin{aligned}
\text{min.} \quad & z(\mathbf{x}) = \sum_{j \in N} c_j x_j \\
\text{s.t.} \quad & \sum_{j \in N} a_{ij} x_j \le b_i, && i \in M_L, \\
& \sum_{j \in N} a_{ij} x_j \ge b_i, && i \in M_G, \\
& \sum_{j \in N} a_{ij} x_j = b_i, && i \in M_E, \\
& x_j \in \{0, 1\}, && j \in N.
\end{aligned}
$$

3

Metaheuristics

- Metaheuristics can be considered as the collection of ideas on designing heuristic algorithms for optimization problems.
- The ideas of metaheuristics give us to a systematic view by incorporating them into greedy and local search algorithm.
- However, we often need to extract special features of individual problems to achieve high performance in practice.

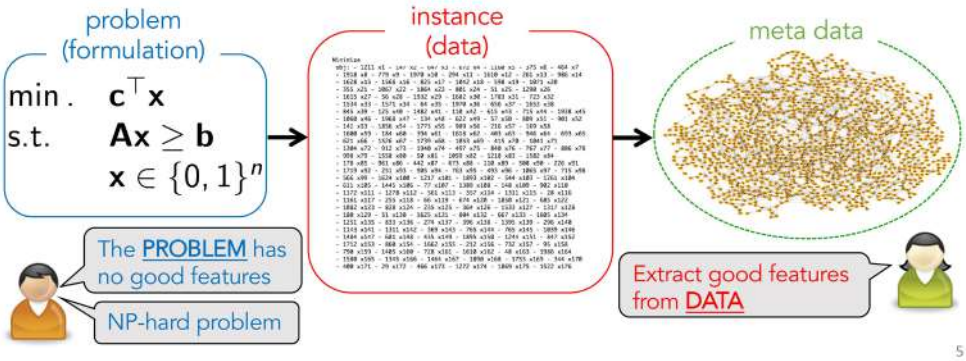


How to develop general purpose metaheuristics?

4

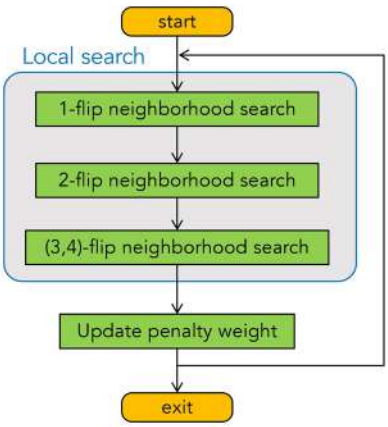
Approach overview

- BIP is a general-purpose optimization problem but having few good features to improve efficiency of algorithms.
- We consider extracting good features from the instance to be solved (not the problem formulation) to improve efficiency of local search (LS) algorithms.
- We introduce a simple data mining approach for reducing the search space of LS algorithms for BIP.



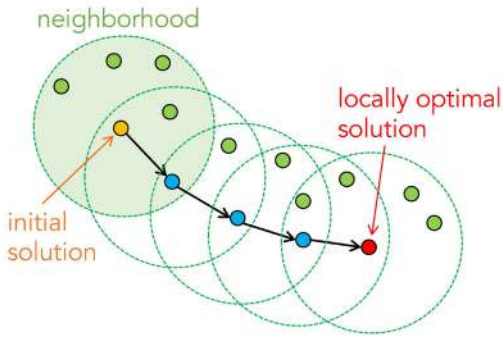
Outline of the proposed algorithm

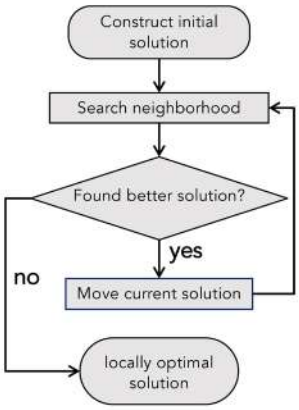
- The main components of the proposed algorithm:
  - ✓ Exploiting variable associations to reduce the search space
  - ✓ Adaptive control of penalty weights
  - ✓ Efficient incremental evaluation



Local search algorithm

- Start from an initial solution $\mathbf{x}$, and repeats replacing $\mathbf{x}$ with a better solution $\mathbf{x}'$ in its neighborhood $\mathrm{NB}(\mathbf{x})$ until no better solution is found in $\mathrm{NB}(\mathbf{x})$.
- Searching the larger neighborhood increases chance to find better solutions but very time consuming.



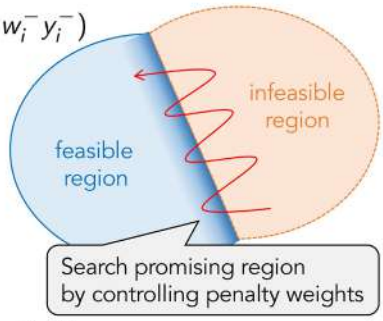


7

Search space and evaluation

- Even the problem only to find a feasible solution is NP-complete.
- Allow excess y_i^+ and shortage y_i^- of the i-th constraint and introduce penalty functions with weights w_i^+ and w_i^- (adaptively controlled in the search).

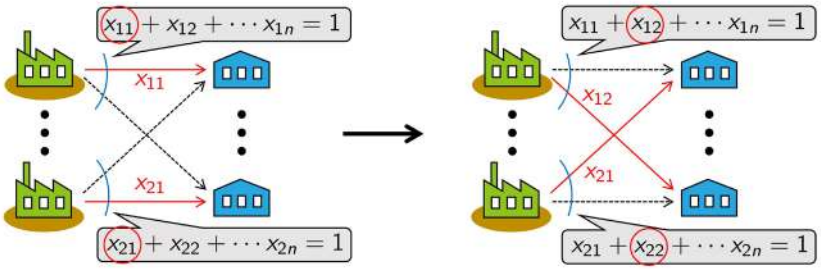
$$
\begin{array}{lll}
\text{min.} & \tilde{z}(\mathbf{x}) = \sum_{j \in N} c_j x_j + \sum_{i \in M_L \cup M_G \cup M_E} (w_i^+ y_i^+ + w_i^- y_i^-) & \\
\text{s.t.} & \sum_{j \in N} a_{ij} x_j - y_i^+ \le b_i, & i \in M_L, \\
 & \sum_{j \in N} a_{ij} x_j + y_i^- \ge b_i, & i \in M_G, \\
 & \sum_{j \in N} a_{ij} x_j - y_i^+ + y_i^- = b_i, & i \in M_E, \\
 & x_j \in \{0, 1\}, & j \in N, \\
 & y_i^+, y_i^- \ge 0, & i \in M_L \cup M_G \cup M_E.
\end{array}
$$



8

Neighborhood structure

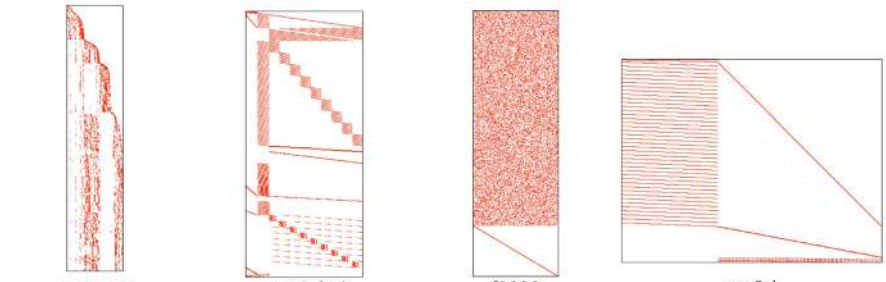
- The k-flip neighborhood $\mathrm{NB}(\mathbf{x})$ for BIP is defined by
  $\mathrm{NB}_k(\mathbf{x}) = \{\mathbf{x}' \in \{0,1\}^n \mid d(\mathbf{x},\mathbf{x}') \leq k\} \quad (d(\mathbf{x},\mathbf{x}') = |\{j \in N \mid x_j \neq x'_j\}|)$
- The 4-flip operation includes representative operations such as exchanging assignments between two agents.
- However, the 4-flip neighborhood is too large to search efficiently in large-scale instances.



How to define and search large neighborhood?

How to extract features in BIP

- MIP instances often include characteristic parts of special constraints, and extracting them may improve the performance of MIP solvers.
- However, extracting characteristic parts of constraint matrix is basically very hard (e.g., extracting a maximum totally unimodular matrix is NP-hard).

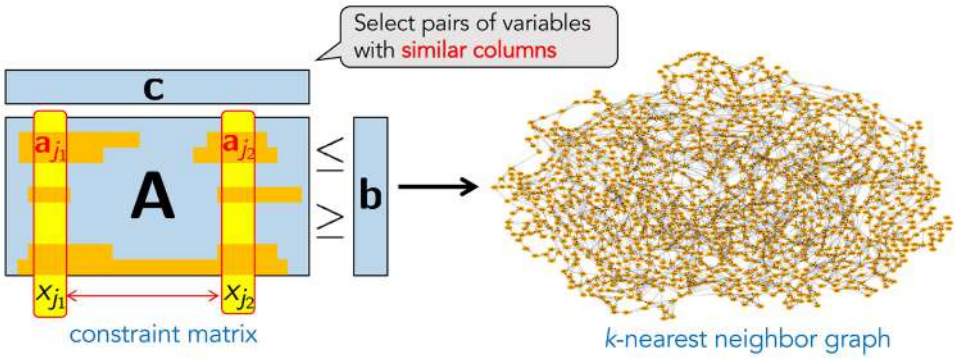


seymour (#cst:4944,#var:1372) acc-tight4 (#cst:3052,#var:1339) f2000 (#cst:10500,#var:4000) stp3d (#cst:159488,#var:204880)

ds (#cst:656,#var:67732)

Exploiting variable associations

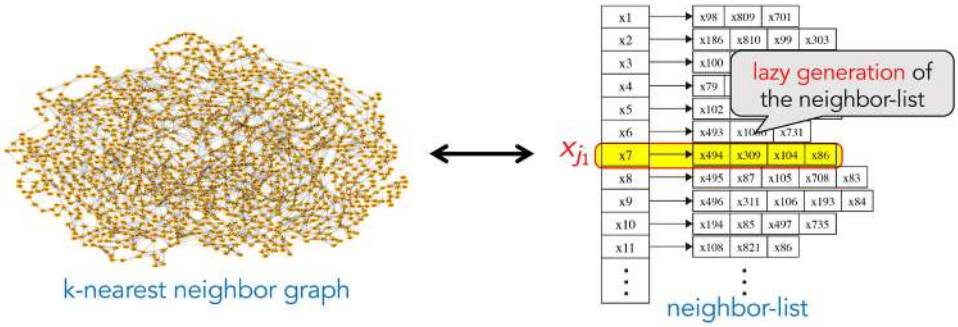
- Find promising pairs of variables in $NB_2(\mathbf{x})$ from the instance to be solved for reducing the search space.
- Keep pairs of variables x_{j_1} and x_{j_2} having similar columns $\mathbf{a}_{j_1}$ and $\mathbf{a}_{j_2}$ in the k-nearest neighbor graph.
- Search candidate solutions only for edges in the k-nearest neighbor graph.



11

Neighbor-list implementation

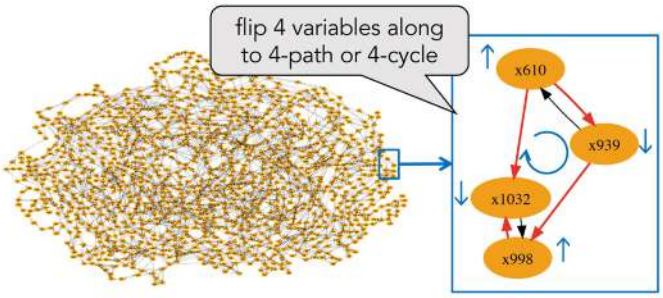
- For each variable x_{j_1}, keep a number of variables x_{j_2} with the largest $\mathbf{a}_{j_1}^{\top}\mathbf{a}_{j_2}$ into the j_1-th row of the neighbor-list.
- However, constructing the whole neighbor-list is still very time consuming for large-scale instances.
- Starting from empty list, we incrementally compute the corresponding rows of neighbor-list when flipping variable x_{j_1} in the 2-flip neighborhood search.



12

Large neighborhood search

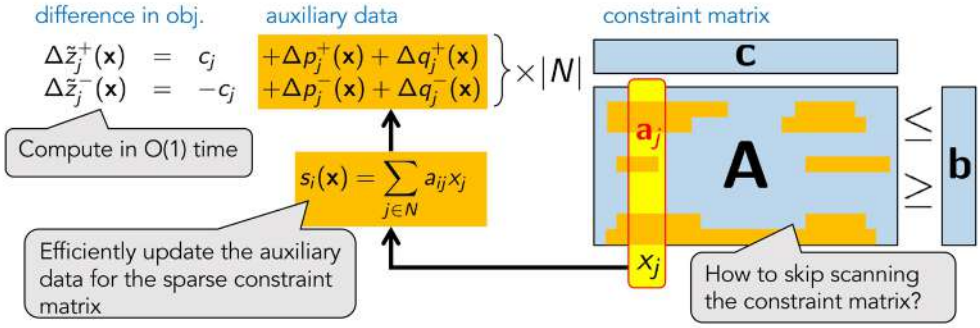
- Using k-nearest neighbor graph, we extend the 2-flip neighborhood search to search a set of promising solution in larger neighborhood.
- We develop the 4-flip neighborhood search for an improving solution by flipping four variables alternately along 4-paths or 4-cycles in the k-nearest neighbor graph.



13

Incremental evaluation

- Naive implementation requires to scan the constraint matrix to compute the evaluation function $\tilde{z}(\mathbf{x})$.
- Using the auxiliary data, we compute the increase of the evaluation function in O(1) time when flipping a variable.
- The auxiliary data are updated only when updating the current solution $\mathbf{x} \to \mathbf{x}' \in \mathrm{NB}(\mathbf{x})^*$.



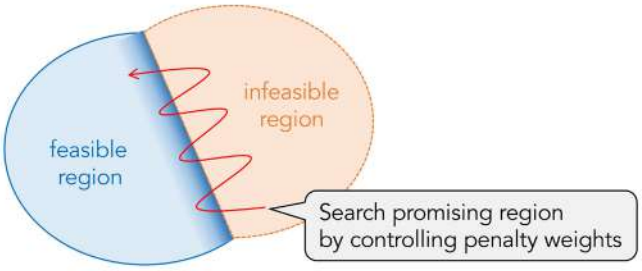
*# of evaluating neighbors $\mathbf{x}' \in \mathrm{NB}(\mathbf{x})$ is much larger than # of updating the current solution.

14

Adaptive control of penalty weights

- We incorporate an adaptive control of penalty weights w_i^+ and w_i^- to escape poor locally optimal solutions and intensify the search around the boundary of feasible region.
- If $\tilde{z}(\mathbf{x}) \geq z(\mathbf{x}^*)$ holds, decrease the penalty weights w_i^+ and w_i^- uniformly; otherwise, increase them by the following formula.

$$w_i \leftarrow w_i + \frac{z(\mathbf{x}^*) - \tilde{z}(\mathbf{x})}{\sum_{k \in M}(y_k^{+2} + y_k^{-2})} y_i, \quad i \in M,$$



* $\mathbf{x}^*$ is the incumbent solution obtained so far.

Test instances of SCP

- We have tested our algorithm on benchmark instances of the set covering problem (SCP) and the set partitioning problem (SPP).
- Run in a single thread on MacBook Pro (Intel Core i7, 2.7GHz, 16GB mem).

			original		presolved		
instance	z_{LP}	z_{best}	#cst.	#var.	#cst.	#var.	time limit
⋆G.1–5 (5)	149.48	166.4	1000.0	10000.0	1000.0	10000.0	600 s
⋆H.1–5 (5)	45.67	59.6	1000.0	10000.0	1000.0	10000.0	600 s
⋆I.1–5 (5)	138.97	158.0	1000.0	50000.0	1000.0	49981.0	1200 s
⋆J.1–5 (5)	104.78	129.0	1000.0	100000.0	1000.0	99944.8	1200 s
⋆K.1–5 (5)	276.67	313.2	2000.0	100000.0	2000.0	99971.0	1800 s
⋆L.1–5 (5)	209.34	258.0	2000.0	200000.0	2000.0	199927.6	1800 s
⋆M.1–5 (5)	415.78	549.8	5000.0	500000.0	5000.0	499988.0	3600 s
⋆N.1–5 (5)	348.93	503.8	5000.0	1000000.0	5000.0	999993.2	3600 s
RAIL507	172.15	∗174	507	63009	440	20700	600 s
RAIL516	182.00	∗182	516	47311	403	37832	600 s
RAIL582	209.71	∗211	582	55515	544	27427	600 s
RAIL2536	688.40	∗689	2536	1081841	2001	480597	3600 s
⋆RAIL2586	935.92	947	2586	920683	2239	408724	3600 s
⋆RAIL4284	1054.05	1064	4284	1092610	3633	607884	3600 s
⋆RAIL4872	1509.64	1530	4872	968672	4207	482500	3600 s

Test instances of SPP

- We have tested our algorithm on benchmark instances of the set covering problem (SCP) and the set partitioning problem (SPP).
- Run in a single thread on MacBook Pro (Intel Core i7, 2.7GHz, 16GB mem).

			original		presolved		
instance	z_{LP}	z_{best}	#cst.	#var.	#cst.	#var.	time limit
aa01–06 (6)	40372.75	∗40588.83	675.3	7587.3	478.7	6092.7	600 s
us01–04 (4)	9749.44	∗9798.25	121.3	295085.0	65.5	85772.5	600 s
v0415–0421 (7)	2385764.17	∗2393303.71	1479.3	30341.6	263.9	7277.0	600 s
v1616–1622 (7)	1021288.76	∗1025552.43	1375.7	83986.7	1171.9	51136.7	600 s
t0415–0421 (7)	5199083.74	5453475.71	1479.3	7304.3	820.7	2617.4	600 s
⋆t1716–1722 (7)	121445.76	157516.29	475.7	58981.3	475.7	13193.6	3600 s
⋆ds	57.23	187.47	656	67732	656	67730	3600 s
⋆ds-big	86.82	731.69	1042	174997	1042	173026	3600 s
⋆ivu06-big	135.43	166.02	1177	2277736	1177	2197774	3600 s
⋆ivu59	884.46	1878.83	3436	2569996	3413	2565083	3600 s

Computational results for SCP (2016)

- We compare other variations with the proposed algorithm: w/o neighbor-list, w/o incremental evaluation, w/o 4-flip.
- We show the relative gap $\frac{z(\mathbf{x})-z_{\text{best}}}{z(\mathbf{x})} \times 100$ (%) of the obtained feasible solution.

instance	no-list	no-inc	2-FNLS	proposed
⋆G.1–5 (5)	0.00%	0.12%	0.00%	0.00%
⋆H.1–5 (5)	0.31%	0.31%	0.31%	0.00%
⋆I.1–5 (5)	1.24%	0.86%	0.50%	0.50%
⋆J.1–5 (5)	2.42%	1.67%	1.68%	1.53%
⋆K.1–5 (5)	2.12%	1.69%	1.32%	1.26%
⋆L.1–5 (5)	3.44%	3.51%	2.35%	2.05%
⋆M.1–5 (5)	10.97%	8.33%	2.79%	2.65%
⋆N.1–5 (5)	19.11%	22.06%	4.76%	5.47%
RAIL507	0.00%	0.57%	0.00%	0.00%
RAIL516	0.00%	0.00%	0.00%	0.00%
RAIL582	0.47%	0.47%	0.47%	0.00%
RAIL2536	2.68%	2.27%	1.29%	0.72%
⋆RAIL2586	2.57%	2.87%	2.27%	1.56%
⋆RAIL4284	5.42%	5.17%	2.74%	2.12%
⋆RAIL4872	4.43%	3.47%	2.36%	1.80%
avg. (all)	4.55%	4.42%	1.65%	1.56%
avg. (with stars)	4.89%	4.75%	1.77%	1.69%

15-20 times faster than the naive algorithm with a simple incremental evaluation

Computational results for SPP (2016)

- We compare other variations with the proposed algorithm: w/o neighbor-list, w/o incremental evaluation, w/o 4-flip.
- We show the relative gap $\frac{z(\mathbf{x})-z_{\text{best}}}{z(\mathbf{x})} \times 100$ (%) of the obtained feasible solution.
- We also show # of instances for which the algorithm obtained feasible solutions within the time limit.

instance	no-list	no-inc	2-FNLS	proposed
aa01–06 (6)	2.33%(6)	2.26%(6)	2.07%(6)	1.60%(6)
us01–04 (4)	0.04%(4)	1.16%(4)	0.63%(4)	0.04%(4)
v0415–0421 (7)	0.00%(7)	0.00%(7)	0.00%(7)	0.00%(7)
v1616–1622 (7)	0.62%(7)	0.17%(7)	0.09%(7)	0.09%(7)
t0415–0421 (7)	1.46%(5)	1.30%(6)	0.29%(7)	0.92%(6)
⋆t1716–1722 (7)	5.46%(7)	4.33%(7)	5.71%(7)	2.45%(7)
⋆ds	36.03%	33.80%	24.13%	0.00%
⋆ds-big	29.11%	0.00%	40.75%	0.00%
⋆ivu06-big	5.31%	3.83%	2.25%	0.00%
⋆ivu59	15.75%	11.39%	16.01%	0.00%
avg. (all)	3.76%(40/42)	2.60%(41/42)	3.35%(42/42)	0.81%(41/42)
avg. (with stars)	10.37%(12/13)	6.61%(12/13)	9.48%(13/13)	1.43%(12/13)

19

Computational results for SCP (2016)

- We compare the proposed algorithm with the recent solvers.
- We show the relative gap $\frac{z(\mathbf{x})-z_{\text{best}}}{z(\mathbf{x})} \times 100$ (%) of the obtained feasible solution.

instance	CPLEX12.6	Gurobi5.6.3	SCIP3.1	LocalSolver3.1	Yagiura et al.	proposed
⋆G.1–5 (5)	0.37%	0.49%	0.24%	45.80%	0.00%	0.00%
⋆H.1–5 (5)	1.92%	2.28%	1.93%	61.54%	0.00%	0.00%
⋆I.1–5 (5)	2.81%	2.72%	1.85%	41.38%	0.00%	0.50%
⋆J.1–5 (5)	8.37%	4.30%	3.59%	58.40%	0.00%	1.53%
⋆K.1–5 (5)	4.77%	4.38%	2.55%	51.22%	0.00%	1.26%
⋆L.1–5 (5)	9.57%	8.44%	3.52%	57.79%	0.00%	2.05%
⋆M.1–5 (5)	18.43%	10.10%	30.71%	71.08%	0.00%	2.65%
⋆N.1–5 (5)	33.13%	12.49%	42.32%	75.63%	0.00%	5.47%
RAIL507	0.00%	0.00%	0.00%	5.43%	0.00%	0.00%
RAIL516	0.00%	0.00%	0.00%	3.19%	0.00%	0.00%
RAIL582	0.00%	0.00%	0.00%	5.80%	0.00%	0.00%
RAIL2536	0.00%	0.00%	0.86%	3.50%	0.29%	0.72%
⋆RAIL2586	2.27%	2.17%	2.27%	5.39%	0.00%	1.56%
⋆RAIL4284	5.34%	1.57%	30.55%	6.50%	0.00%	2.12%
⋆RAIL4872	1.73%	1.73%	2.67%	5.61%	0.00%	1.80%
avg. (all)	8.64%	4.92%	10.00%	49.99%	0.01%	1.56%
avg. (with stars)	9.45%	5.38%	10.91%	54.22%	0.00%	1.69%

20

Computational results for SPP (2016)

- We compare the proposed algorithm with the recent solvers.
- We show the relative gap $\frac{z(\mathbf{x})-z_{\text{best}}}{z(\mathbf{x})} \times 100$ (%) of the obtained feasible solution.
- We also show # of instances for which the algorithm obtained feasible solutions within the time limit.

instance	CPLEX12.6	Gurobi5.6.3	SCIP3.1	LocalSolver3.1	proposed
aa01–06 (6)	0.00%(6)	0.00%(6)	0.00%(6)	13.89%(1)	1.60%(6)
us01–04 (4)	0.00%(4)	0.00%(4)	0.00%(3)	11.26%(2)	0.04%(4)
v0415–0421 (7)	0.00%(7)	0.00%(7)	0.00%(7)	0.05%(7)	0.00%(7)
v1616–1622 (7)	0.00%(7)	0.00%(7)	0.00%(7)	4.60%(7)	0.09%(7)
t0415–0421 (7)	0.66%(7)	0.60%(7)	1.61%(6)	— (0)	0.92%(6)
⋆t1716–1722 (7)	8.34%(7)	16.58%(7)	3.51%(7)	37.08%(1)	2.45%(7)
⋆ds	8.86%	55.61%	40.53%	85.17%	0.00%
⋆ds-big	62.16%	24.03%	72.01%	92.69%	0.00%
⋆ivu06-big	20.86%	0.68%	17.90%	52.54%	0.00%
⋆ivu59	28.50%	4.36%	37.84%	48.95%	0.00%
avg. (all)	4.37%(42/42)	4.88%(42/42)	5.06%(40/42)	17.52%(22/42)	0.81%(41/42)
avg. (with stars)	14.10%(13/13)	15.66%(13/13)	15.07%(13/13)	63.29%(5/13)	1.43%(12/13)

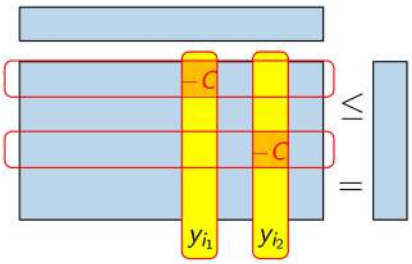
21

Current technical issues

- BIPSOL misses pairs of variables to be swapped for fixed charge cost type problems such as the bin packing problem and the facility location problem.
- The variable y_i illustrates the usage of i-th bin.
- We often prefer to swap a selected bin and a non-selected one.
- Any pair of variables y_{i_1} and y_{i_2} never appears in the same constraint.
- 2-FNLS never swaps the values of y_{i_1} and y_{i_2} simultaneously.

bin packing problem

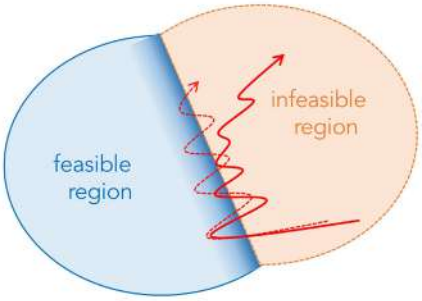
$$
\begin{aligned}
\text{min.} \quad & \sum_{i \in M} y_i \\
\text{s.t.} \quad & \sum_{j \in N} w_j x_{ij} - C y_i \le 0, && i \in M, \\
& \sum_{i \in M} x_{ij} = 1, && j \in N, \\
& x_j \in \{0,1\}, && j \in N, \\
& y_i \in \{0,1\}, && i \in M.
\end{aligned}
$$



22

Current technical issues (cont.)

- We expect BIPSOL searches around the boundary of the feasible region, but ...
- BIPSOL often dives the infeasible region because of the undesirable behavior of the adaptive control of penalty weights.
- The local search algorithm sometimes stops at an infeasible solution even though increasing penalty weights.
- BIPSOL then decreases penalty weights despite of having some violating constraints, it makes to dive the infeasible region deeper.



23

Conclusion and future direction

- We introduce a simple data mining approach to reduce the search space of local search algorithms by extracting the instance to be solved.
- We construct parts of a *k*-nearest neighbor graph on demand that identifies promising pairs of flipping variables in the 2-flip neighborhood search.
- We develop the 4-flip neighborhood search using the *k*-nearest neighbor graph.
- We also introduce the adaptive control of penalty weights and fast incremental evaluation.
- We discuss current some technical issues on BIPSOL.

- We now plan to extend BIPSOL to perform more general optimziation problems, such as the general integer programs (IP), the mixed integer programs (MIP), the binary quadratic programs with constraints (BQP), the constraint satisfaction problem (CSP), etc.

24

Reference

- S. Umetani, Exploiting variable associations to configure efficient local search algorithm in large-scale binary integer programs, European Journal of Operational Research, **263** (2017), 72-81. (open access)
- S. Umetani, Exploiting variable associations to configure efficient local search algorithm in large-scale set partitioning problems, Proc. of 29th AAAI Conference on Artificial Intelligence (AAAI-15), 1226-1232.


European Journal of Operational Research 263 (2017) 72–81

Contents lists available at ScienceDirect

European Journal of Operational Research

journal homepage: www.elsevier.com/locate/ejor

Discrete Optimization

Exploiting variable associations to configure efficient local search algorithms in large-scale binary integer programs☆

Shunji Umetani

Graduate School of Informatics Science and Technology, Osaka University, 1-5 Yamadaoka, Suita, Osaka 565-0871, Japan

25

Performance of the supercomputer Fugaku for Graph500 benchmark

Masahiro Nakao

RIKEN Center for Computational Science, Japan
masahiro.nakao@riken.jp

We present the performance of the supercomputer Fugaku for breadth-first search (BFS) in the Graph500 benchmark, which is known as a ranking benchmark used to evaluate large-scale graph processing performance on supercomputer systems. Fugaku is a huge-scale Japanese exascale supercomputer that consists of 158,976 nodes. We evaluate the BFS performance for a large-scale graph consisting of about 2.2 trillion vertices and 35.2 trillion edges using the whole Fugaku system, and achieve 102,955 giga-traversed edges per second, resulting in the first position of Graph500 BFS ranking[1, 2].

References

[1] Masahiro Nakao, Koji Ueno, Katsuki Fujisawa, Yuetsu Kodama, Mitsuhisa Sato. ``Performance of the Supercomputer Fugaku for Breadth-First Search in Graph500 Benchmark.", ISC 2021, Jun. 2021, https://doi.org/10.1007/978-3-030-78713-4_20

[2] https://graph500.org

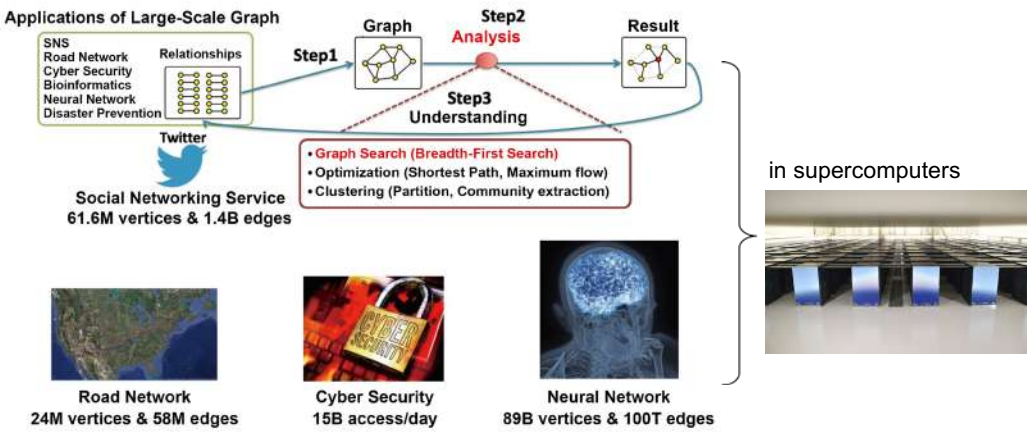
Performance of the supercomputer Fugaku for Graph500 benchmark

†Masahiro Nakao

Joint work with ‡Koji Ueno, *Katsuki Fujisawa, †Yuetsu Kodama, †Mitsuhisa Sato

† RIKEN Center for Computational Science
‡ Fixstars Corporation
* Institute of Mathematics for Industry, Kyushu University

Background



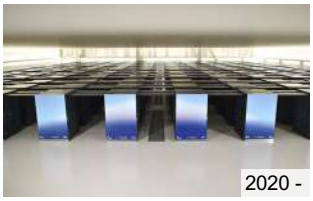
http://opt.imi.kyushu-u.ac.jp/lab/jp/activities.html

Parallelize BFS and SSSP algorithms on supercomputers

- Parallelize fundamental graph algorithms on supercomputers
  - Breadth First Search (BFS) in cooperation with IMI
  - Single-Source Shortest Path (SSSP) in cooperation with ZIB and IMI
- Large scale supercomputers
  - Graph algorithm that runs efficiently on such systems is a challenging research

The K computer : 82,944 nodes[1]

Supercomputer Fugaku : 158,976 nodes[2]



[1] Koji Ueno et al: Efficient breadth-first search on massively parallel and distributed-memory machines. Data Science and Engineering, 2016

[2] Masahiro Nakao et al: Performance of the Supercomputer Fugaku for Breadth-First Search in Graph500 Benchmark." ISC 2021

Graph500

https://graph500.org

- Graph500 is a competition for evaluating performance of large-scale graph processing
- The performance unit is a traversed edges per second (TEPS)
  - 1GTEPS : Search 1 billion edges per second
- Graph500 list is updated twice a year (June and November in ISC and SC)
  - BFS : The K computer ranked first 10 times from 2014 to 2019
  - BFS : Supercomputer Fugaku ranks first from 2020 to now
  - SSSP : Supercomputer Fugaku ranks 2nd in June 2022
- In graph500, an artificial graph called the “Kronecker graph” is used
  - Some vertices are connected to many other vertices while numerous others are connected to only a few vertices
  - Social network is known to have a similar property

Objective

- This presentation describes the performance tuning of BFS for the Graph500 submission and experimental evaluation results conducted on Fugaku

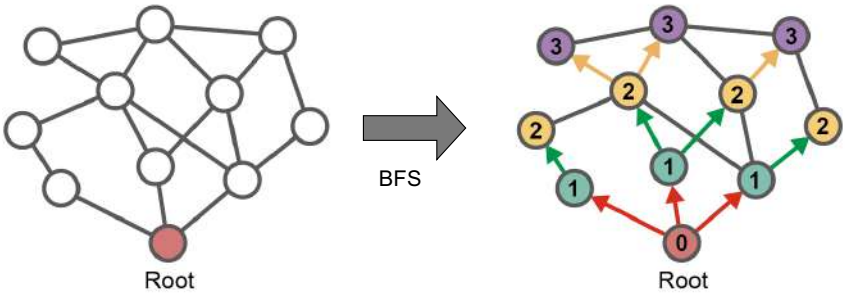
- Summary
  - Use a large-graph with 2.2 trillion vertices and 35.2 trillion edges (SCALE=41)
  - Archive 102,955 GTEPS
  - The performance of Fugaku is 3.3 times better than that of the K computer

	June 2019			November 2020		
	NAME	SCALE	GTEPS	NAME	SCALE	GTEPS
1st	**K computer**	**40**	**31,302**	**Supercomputer Fugaku**	**41**	**102,955**
2nd	Sunway TaihuLight	40	23,756	Sunway TaihuLight	40	23,756
3rd	Sequoia	41	23,751	TOKI-SORA	36	10,813
4th	Mira	40	14,982	Summit	40	7,666
5th	SuperMUC-NG	39	6,279	SuperMUC-NG	39	6,279

Outline

- BFS in Graph500 Benchmark
- The supercomputer Fugaku
- Tuning BFS on the supercomputer Fugaku
- Full node evaluation

Overview of BFS in Graph500

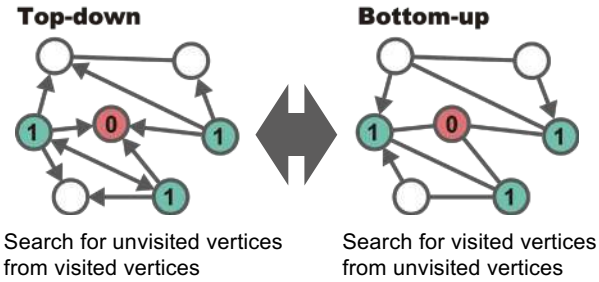


Input : Kronecker graph and root vertex **Output : BFS tree**

- Repeat BFS 64 times from different root vertex
- The harmonic mean of the performance in 64 trials is used as the final performance

Hybrid-BFS [Beamer, 2012] Scott Beamer et al. Direction-optimizing breadth-first search

- Hybrid-BFS runs while switching between **Top-down** and **Bottom-up**



- In the middle of BFS, the number of vertices being visited increases explosively, so it is inefficient in only **Top-down**
- Hybrid-BFS switches between **Top-down** and **Bottom-up** on the situation

Number of Vertex Checks

- SCALE = 26
  - Graph with 2^{SCALE} vertices and 2^{SCALE+4} edges
  - 67.1 million vertices and 1.1 billion edges

	Top-down	Bottom-up	Hybrid-BFS
0	**2**	2,103,840,895	**2**
1	**66,206**	1,766,587,029	**66,206**
2	346,918,235	**52,667,691**	**52,667,691**
3	1,727,195,615	**12,820,854**	**12,820,854**
4	29,557,400	**103,184**	**103,184**
5	82,357	**21,467**	**21,467**
6	**221**	21,240	**221**
Total	2,103,820,036	3,936,062,360	**65,679,625**
Rate	100.00%	187.09%	**3.12%**

Hybrid-BFS decides when to switch between Top-down and Bottom-up from information such as the number of vertices being searched.

2D Hybrid-BFS

[Beamer, 2013] Scott Beamer, et. al. Distributed Memory Breadth-First Search Revisited: Enabling Bottom-Up Search. IPDPSW '13.

- Adjacency matrix is distributed to a 2D process grid (R x C)

$$A = \left(\begin{array}{c|c|c} A_{1,1} & \cdots & A_{1,C} \\ \hline \vdots & \ddots & \vdots \\ \hline A_{R,1} & \cdots & A_{R,C} \end{array}\right)$$

- Communication only within the column processes and row processes
  - Allgatherv, Alltoallv, point-to-point (isend/irecv/wait)
- The closer the R and C values are, the smaller the total communication size
- **Based on this 2D Hybrid-BFS, we implemented BFS with various ideas to improve performance[1]**

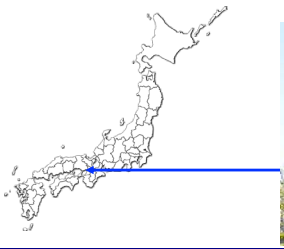
[1] Koji Ueno et al: Efficient breadth-first search on massively parallel and distributed-memory machines. Data Science and Engineering, (2016)

Outline

- BFS in Graph500 Benchmark
- The supercomputer Fugaku
- Tuning BFS on the supercomputer Fugaku
- Full node evaluation

The supercomputer Fugaku

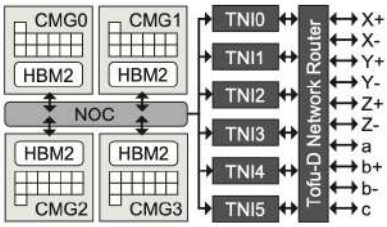
- The supercomputer Fugaku, which is developed jointly by RIKEN and Fujitsu Limited based on Arm technology
- Located in RIKEN Center for Computational Science in Kobe, Hyogo, Japan
- 158,976 compute nodes
- Start sharing in March 2021



Specification of Computer Node

CPU	A64FX, 48+2/4cores, 2.0/2.2GHz, L2 8MB 3,072/3,379GFlops (double precision)
Memory	HBM2, 32GB, 1,024GB/s
Network	TofuD, 0.49 to 0.54μs (Latency) 6.8GB/s (Bandwidth)

CPU (A64FX)



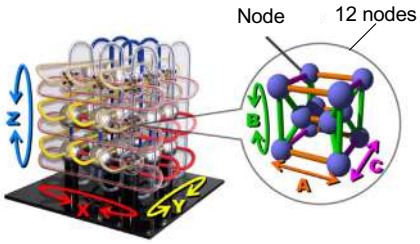
L2 Cache Coherent control between CMGs

- Each node has a single CPU
- Each CPU has 48 compute cores and 2/4 assistant cores. The assistant cores handle the interrupts OS and communication
- **2.0 GHz or 2.2 GHz for each job**
- Each CPU consists of 4 CMGs
  - Each CMG consists of 12 + 1 cores and 8GiB HBM2
  - **It is recommended that the number of processes per CPU is a divisor of 4**
- Each CPU has 10 network cables

CMG : Core Memory Group
NOC : Network on Chip
TNI: Tofu Network Interface

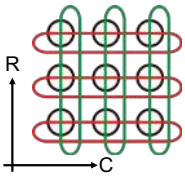
Network Topology of Fugaku

- 6D mesh/torus : XYZabc-axis
  - The size of abc is (2,3,2)
  - The size of XYZ is (24,23,24)
    so it has 24*23*24*2*3*2 = 158,976 nodes



https://pr.fujitsu.com/jp/news/2020/04/28.html

- Process Mapping
  - Discrete assignment
  - 1D torus or mesh
  - **2D torus** or mesh
  - 3D torus or mesh



$$A = \begin{pmatrix} A_{1,1} & \cdots & A_{1,C} \\ \vdots & \ddots & \vdots \\ A_{R,1} & \cdots & A_{R,C} \end{pmatrix}$$

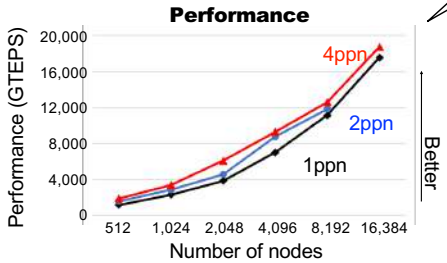
Outline

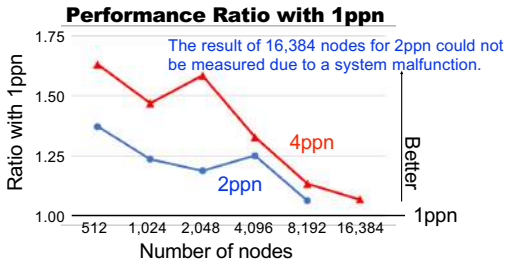
- BFS in Graph500 Benchmark
- The supercomputer Fugaku
- Tuning BFS on the supercomputer Fugaku
- Full node evaluation

Number of processes per node (1/2)

- Process per node (ppn)
  - 1 process 48 threads per node (1ppn)
  - 2 processes 24threads per node (2ppn)
  - 4 processes 12threads per node (4ppn)

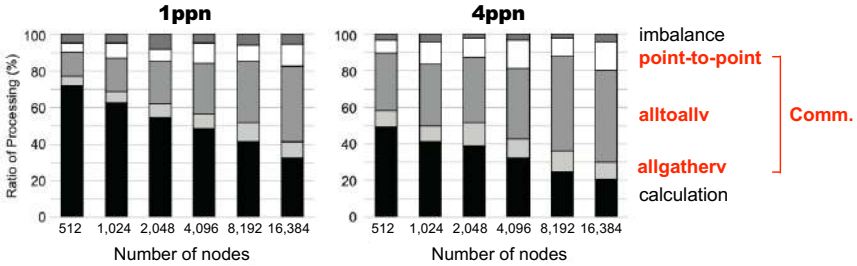
In the cases of 1ppn and 2ppn, the cache hit rate decreases because the memory accesses by threads cross the CMG.





- The larger the number of nodes, the smaller the performance difference

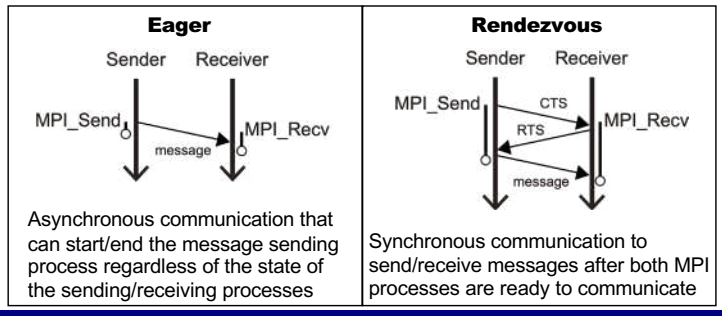
Number of processes per node (2/2)



- As the number of nodes increases, the rate of communication increases
- 1ppn has a smaller rate of communication than 4ppn
- If the number of nodes is increased further, the communication ratio will increase.
- **Thus, we select 1ppn, which can bring out the full communication performance**

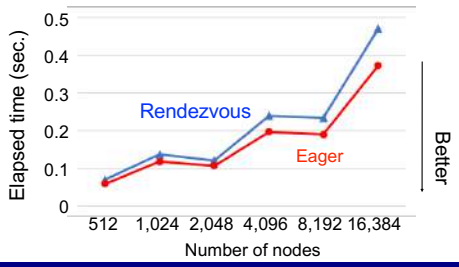
Use of Eager method (1/2)

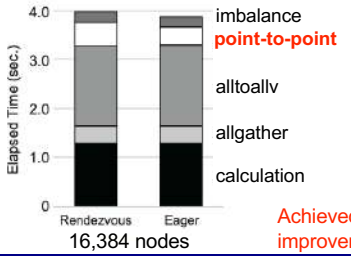
- In the point-to-point communication of most MPI implementations, the Eager and Rendezvous methods are implemented
- Although most MPI implementations switch the Eager and Rendezvous methods automatically depending on message size, optimal message size depends on application



Use of Eager method (2/2)

- In the default setting, Rendezvous was selected for all send/recv communication of BFS
- Fujitsu MPI library on Fugaku can set the threshold for switching between Eager and Rendezvous methods
  - We change the threshold to 512 Kbytes from default value to use Eager method
  - Since Fugaku's compute node has 32 Gbytes memory, the buffer size is relatively small





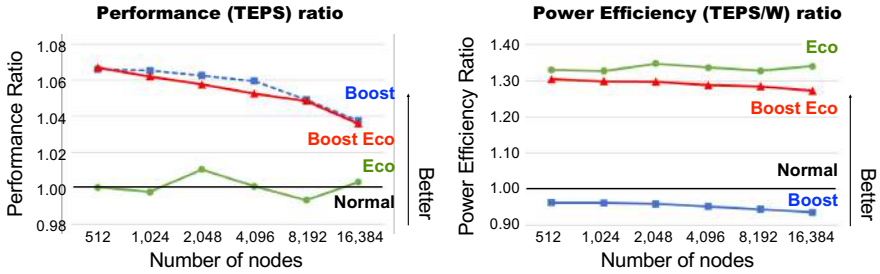
Achieved a performance improvement of 2.3%

Power Management (1/2)

- User can specify CPU frequency for each job
  - **Normal mode : 2.0 GHz**
  - **Boost mode : 2.2 GHz**
- **Eco mode : Two** floating-point arithmetic pipelines of A64FX are limited to **one**, and power control is performed according to the maximum power
  - Since BFS does not perform floating-point arithmetic, the use of Eco mode can be expected to reduce power consumption without affecting performance

- **Normal :** 2.0 GHz, **two** floating-point arithmetic pipelines (in previous evaluations)
- **Boost :** 2.2 GHz, **two** floating-point arithmetic pipelines
- **Normal Eco :** 2.0 GHz, **one** floating-point arithmetic pipeline
- **Boost Eco :** 2.2 GHz, **one** floating-point arithmetic pipeline

Power Management (2/2)



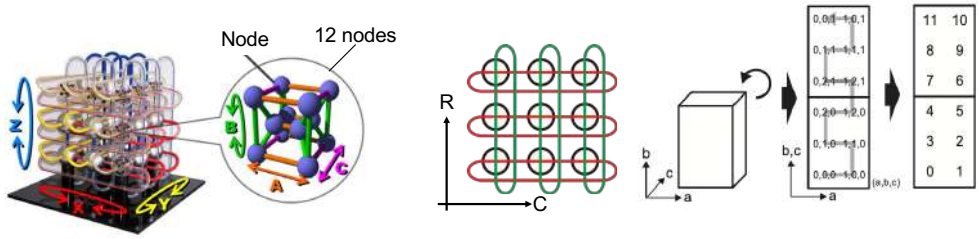
- Boost Eco mode has a good balance between performance and power efficiency
- The performance in Boost Eco mode is 3.6 % better than that in Normal mode
- The power efficiency in Boost Eco mode is 27.2 % better than that in Normal mode
- The results of Boost Eco mode for 16,384 nodes are 18,607 GTEPS and 1,408 kW

Outline

- BFS in Graph500 Benchmark
- The supercomputer Fugaku
- Tuning BFS on the supercomputer Fugaku
- Full node evaluation

Six-dimensional Process Mapping (1/2)

- The size of six axes in Fugaku network is (X, Y, Z, a, b, c) = (23, 24, 23, 2, 3, 2)
- It is desirable that the values of R and C process grid of BFS are close
- We assign the processes to (R, C) = (XY, Zabc) = (552, 288)
- Since neighborhood communication occurs in BFS, we assign the processes physically next to each other in row/column dimension

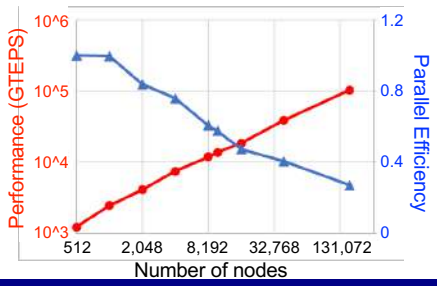


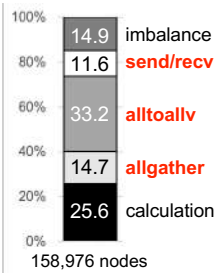
Six-dimensional Process Mapping (2/2)

- We evaluate the BFS performance for a large-scale graph consisting of about 2.2 trillion vertices and 35.2 trillion edges using the whole Fugaku system (158,976 nodes)
- Boost Eco mode
- **Performance: 102,956 GTEPS, Power: 14,961 kW, Efficiency: 6.9 MTEPS/W**

- Performance is 3.3 times that of the K computer (82,944 nodes), and power efficiency is 1.9 times that of IBM Sequoia (Blue Gene/Q)

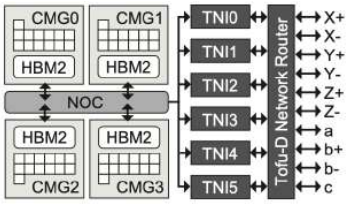
The K computer did not measure power. At IBM Sequoia, the graph is the same size as Fugaku.





Summary

- Tune performance of BFS in Graph500 on Fugaku
- We evaluate the BFS performance for a large-scale graph consisting of about 2.2 trillion vertices and 35.2 trillion edges using the whole Fugaku system
- Achieve 102,955 GTEPS, resulting in the first position of Graph500 lists in from 2020 to now



- Future works
  - Develop SSSP in cooperation with ZIB and IMI
  - Some ideas to improve performance of BFS

25

Notes on Solving QUBOs and Quantum Computing

Prof. Dr. Thorsten KOCH
Zuse Institute Berlin and Technical University of Berlin,
Germany
koch@zib.de

It is regularly claimed that quantum computers will bring breakthrough progress in solving challenging combinatorial optimization problems relevant in practice. In particular, Quadratic Unconstraint Binary Optimization (QUBO) problems are said to be the model of choice for use in (adiabatic) quantum systems. Combinatorial Optimization searches for an optimum object in a finite but usually vast collection of objects. This approach can be used for many practical purposes, like efficient allocation of limited resources, network planning, and hundreds of other applications in almost all fields, e.g., finance, production, scheduling, and inventory control. However, many combinatorial optimization problems are known to be NP-hard. This theoretical statement about worst-case runtime complexity is often translated simplistically as "intractable"; however, the practical side looks different. In many cases, it is possible to solve such problems to proven global optimality. We explain some of the meaning and implications, review the state of affairs, the potential of quantum computing, and give new computational results regarding solving OUBOs.

References

[1] Kamil Bradler, Hugo Wallner, "Certain properties and applications of shallow bosonic circuits", 2021, https://arxiv.org/abs/2112.09766
[2] Catherine C. McGeoch, "Adiabatic Quantum Computation and Quantum Annealing", Synthesis Lectures on Quantum Computing, Springer, 2014, https://link.springer.com/book/10.1007/978-3-031-02518-1
[3] Feng Pan, Keyang Chen, Pan Zhang, "Solving the Sampling Problem of the Sycamore Quantum Circuits", Physical Review Letters, Vol. 129, Issue 9, 2022 https://journals.aps.org/prl/abstract/10.1103/PhysRevLett.129.090502
[4] Richard M. Karp, "Reducibility among Combinatorial Problems", Complexity of Computer Computations, The IBM Research Symposia Series, pp. 85-103, 1972
[5] Leo Liberti, "Undecidability and hardness in mixed-integer nonlinear programming", RAIRO Operations Research, Volume 53, pp. 81-109, 2019, https://hal.archives-ouvertes.fr/hal-02104836
[6] Robin Brown, David E. Bernal Neira, Davide Venturelle, Marco Pavone, "Copositive programming for mixed-binary quadratic optimization via Ising Solvers", 2022 https://arxiv.org/abs/2207.13630
[7] Fred Glover, Gary Kochenberger, Yu Du, „Quantum Bridge Analytics I: A Tutorial on Formulating and Using QUBO Models", 2022, https://arxiv.org/abs/1811.11538
[8] Daniel Bienstock, Oktay Günlük, "Computational experience with a difficult mixed-integer multicommodity flow problem", Mathematical Programming, Volume 68, pp. 213-237, 1995, https://doi.org/10.1007/BF01585766
[9] Daniel Rehfeldt, Thorsten Koch, Yuji Shinano, "Faster exact solution of sparse MaxCut and QUBO problems", 2022, https://arxiv.org/abs/2202.02305
[10] Michael Jünger, Elisabeth Lobe, Petra Mutzel, Gerhard Reinelt, Franz Rendl, Giovanni

Rinaldi, „Quantum Annealing versus Digital Computing: An Experimental Comparison“, ACM Journal of Experimental Algorithmics, Volume 26, pp. 1-30, 2021, https://doi.org/10.1145/3459606

[11] Scott Aaronson, “NP-complete Problems and Physical Reality”, 2005 https://arxiv.org/abs/quant-ph/0502072

[12] Giacomo Nannicini, “An Introduction to Quantum Computing, Without the Physics”, 2017, https://arxiv.org/abs/1708.03684

[13] Pradeep Niroula, “Conquering the challenge of quantum optimization”, Physics World, 2022, https://physicsworld.com/a/conquering-the-challenge-of-quantum-optimization

[14] Lennart Bittel, Martin Kliesch, “Training Variational Quantum Algorithms Is NP-Hard”, Physical Review Letters, Volume 127, 2021, https://journals.aps.org/prl/abstract/10.1103/PhysRevLett.127.120502

[15] “Twenty Questions for Donald Knuth”, InformIT, 2014 https://www.informit.com/articles/article.aspx?p=2213858&WT.mc_id=Author_Knuth_20Questions

News: The Ministry of Defence has procured the government's first quantum computer.

https://quantumzeitgeist.com/british-ministry-of-defence-procures-governments-first-quantum-computer
They'll work with Orca Computing, a British company, to investigate and apply quantum technology in military defense.
Quantum computer manufacturers claim their devices can solve complex problems that classical computers cannot solve.

About Orca I find there: **British Quantum Computing Firm ORCA, Claims Breakthrough in Quantum Computers**
https://quantumzeitgeist.com/british-quantum-computing-firm-orca-claims-breakthrough-in-quantum-computers
The **major challenge** identified by one of the UK's leading quantum computing experts Prof Morton, **was the ability of the Orca computer to scale up** quickly.

"Scaling up is very important for the computer to be able to serve its purpose by performing complex, highly scientific and experimental tasks such as combating climate changes, accelerating artificial intelligence, ship navigation or even drug development.
These tasks require millions of qubits to be successful when in reality, the Orca computer has just four qubits which is very far from ideal. "
However, the company assured that they will scale up in the next two years.

Looking at the Orca Webpage I found then:
https://www.orcacomputing.com/news/orca-computing-provides-uk-mod-with-first-quantum-computer-i6pns-rzbhh
ORCA discovers new algorithm for solving QUBO problems with near-term, 'shallow' quantum computers
On the webpage Orca claims to solve binary knapsack problems with 70 variables faster than any other QC and classical methods.

As a reference to the conclusions a scientific article published by the Orca scientists was given: https://arxiv.org/abs/2112.09766

Unfortunately, there is nothing about knapsacks, but:

2.8. Scalability and other challenges.
How plausible is it to scale the presented variational method?
The first problem the reader can point out is the fact that despite the parity map effectively coarse graining the measurement results of an M-mode circuit it is not a sustainable strategy since the size of an M-qubit Hilbert space grows exponentially. [...]
But here we tacitly assume that the measurement outcomes are uniform which is not the case. We observed that randomly chosen parameters of the studied shallow circuits typically result in a small set of measurement patterns to have high probability enabling it to be sampled with confidence with a bounded number of repeated measurements.
The caveat is, of course, that even if this trend continues as M grows it inevitably means that the ratio of such reliably estimated measurements with respect to all possible patterns decreases exponentially.
On the one hand, this still enables us to use the proposed variational algorithm. However, the odds of getting stuck in a local minimum most likely increase. How exactly it affects the ability to reach a global solution is a matter of a more detailed study.

Notes on Solving QUBOs and Quantum Computing

Thorsten Koch
Daniel Rehfeldt
Yuji Shinano

Chair of
Software and Algorithms
for Discrete Optimization

Department:
Applied Algorithmic
Intelligence Methods

"In theory, there is no difference between theory and practice; but in practice, there is."

— unknown (not Einstein, or Feynman)

News: The Ministry of D ... ter.

https://quantumzeitgeist.com/british-ministry-of-def

They'll work with Orca Computing, a ... article published by the Orca ... 12.09766

apply quantum technology in milita

Quantum computer manufacturers ... cks, but:

problems that classical computers c

About Orca I find there: **British Qua ... iational method?**

Breakthrough in Quantum Comput ... the fact that despite the parity

https://quantumzeitgeist.com/british-quantum-comp ... ment results of an M-mode

The **major challenge** identified by o ... e size of an M-qubit Hilbert

computing experts Prof Morton, **wa**

scale up quickly.

"Scaling up is very important for the ... ment outcomes are uniform

by performing complex, highly scien ... domly chosen parameters of the

combating climate changes, accelen ... mall set of measurement

navigation or even drug developme ... to be sampled with confidence

These tasks require millions of qub ... ements.

Orca computer has just four qubits ... d continues as M grows it

However, the company assured tha ... ly estimated measurements

... es exponentially.

Looking at the Orca Webpage I four ... the proposed variational

https://www.orcacomputing.com/news/orca-computing-prov ... k in a local minimum most likely

ORCA discovers new algorithm for ... reach a global solution is a

'shallow' quantum computers

On the webpage Orca claims to solv

variables faster than any other QC a

ELSEVIER

European Journal of Operational Research

Volume 2, Issue 6, November 1978, Pages 420-428

An algorithm and efficient data structures for the binary knapsack problem

Uwe Suhl *

Show more

+ Add to Mendeley Share Cite

https://doi.org/10.1016/0377-2217(78)90137-6 Get rights and content

Abstract

A branch-and-bound algorithm for the binary knapsack problem is presented which uses a combined stack and deque for storing the tree and the corresponding LP-relaxation. A reduction scheme is used to reduce the problem size. The algorithm was implemented in FORTRAN. Computational experience is based on 600 randomly generated test problems with up to 9000 zero-one variables. The average solution times (excluding an initial sorting step) increase linearly with problem size and compare favorably with other codes designed to solve binary knapsack problems.

Being Deterministic

https://xkcd.com/221

```
int getRandomNumber()
{
    return 4;  // chosen by fair dice roll.
               // guaranteed to be random.
}
```

"Insanity Is Doing the Same Thing Over and Over Again and Expecting Different Results."

— unknown

An algorithm is deterministic, if **given a particular input, it will always produce the same output**, with the underlying machine always passing through the same sequence of states. https://en.wikipedia.org/wiki/Deterministic_algorithm

A deterministic algorithm computes a mathematical function; a function has a unique value for any input in its domain, and the algorithm is a process that produces this particular value as output.

Non-determinism can result, for example, from:

- use of an external state other than the input, such as a hardware timer value.
- if multiple processors writing to the same data at the same time. In this case, the precise order in which each processor writes its data will affect the result.

While digital computers are thought of being deterministic,
Quantum computers a probabilistic, i.e., non-deterministic by definition.

What does "solving" an NP-hard problem typically mean?

	Being able to …
Theoretical Computer Scientist	… compute **proven optimal solutions to every instance** of this problem class with at most this effort
Applied Discrete Mathematician	… practically compute within numerical tolerances **proven optimal solutions to these particular (relevant) instances** of the problem class in reasonable time
Physicist, Quantum Computing Researcher	… compute reasonably **good solutions to these (selected) particular instances** of the problem class in very short time

Setting the stage

"Combinatorial Optimization searches for an optimum object in a finite collection of objects. Typically, the collection has a concise representation, while the number of objects is huge --- more precisely, grows exponentially in the size of the representation. So scanning all objects one by one and selecting the best one is not an option."

— Alexander Schrijver, Combinatorial Optimization, 2003, Page 1.

$$\min_{x \in X} \boldsymbol{f}(\boldsymbol{x}) \text{ with } \boldsymbol{X} = \left\{\boldsymbol{x}, \boldsymbol{b}, \underline{\boldsymbol{l}}, \overline{\boldsymbol{u}} \in \mathbb{Z}^{\boldsymbol{n}} : \boldsymbol{g}(\boldsymbol{x}) \leq \boldsymbol{b}, \underline{\boldsymbol{l}} \leq \boldsymbol{x} \leq \overline{\boldsymbol{u}}\right\}$$

For the rest of the talk, we assume: $f: X \to \mathbb{Z}$ is a linear or quadratic function, i.e., $f(x) = c^T x + x^T Q x, c \in \mathbb{Z}^n, Q \in \mathbb{Z}^{n \times n}$, and $g: X \to \mathbb{Z}^n$ is a linear function, i.e., $g(x) = Ax, A \in \mathbb{Z}^{n \times n}$.

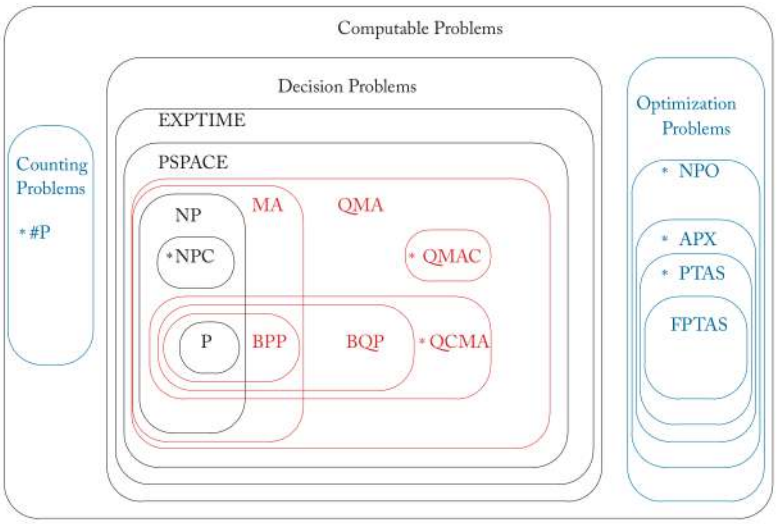
Note: $\operatorname{argmin} f(x) = \operatorname{argmax} -f(x)$ and $g(x) + s = b, s \geq 0 \iff g(x) \leq 0$, and similar for $\geq$.

We defined everything using integer numbers. If we would use rational numbers, we could then scale them by the least common multiple of all denominators to make everything integer.

Disclaimer

For the rest of this talk,
unless noted otherwise,
everything I say,
will refer (only) to
Combinatorial Optimization Problems.

The Bigger (complexity) Picture



From Catherine C. McGeoch: Adiabatic Quantum Computation and Quantum Annealing

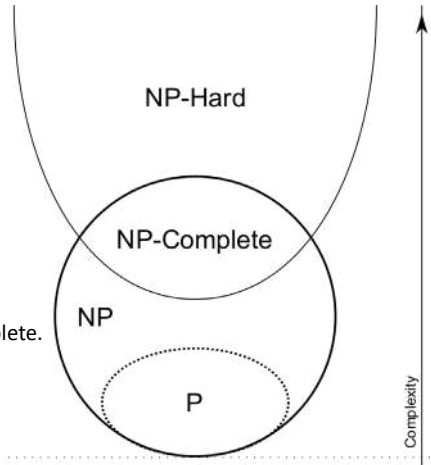
BQP is bounded-error quantum polynomial time. A quantum model of computation uses an arrangement of quantum circuits (in the gate model) operating on qubits. These problems can be solved by a quantum computer in polynomial time with an error probability of at most 1/3.

QMA stands for Quantum Merlin-Arthur. This is the set of decision problems for which yes-instances can be verified by a quantum computer in polynomial time, and no-instances rejected in polynomial time, with error probability at most 1/3. This is the quantum analog of NP, related to BQP as NP is related to P. QMAC is the set of complete problems for this class.

QCMA stands for Quantum-classical Merlin-Arthur. This class is similar to QMA but the witness for a yes-instance must be a classical string.

Complexity classes

P *"easy"*	Solvable in polynomial time. A running time of $10^{20} \cdot n^{1000}$ for input size n is poly time.
NP	"Yes" answer can be checked in poly time. It does not matter where the answer comes from.
NP-Complete	Hardest problems in NP. If you could solve one of them in poly time, then you can solve them all.
NP-Hard *"intractable"*	As least as hard as the hardest problems in NP. Need not to be in NP.



By Behnam Esfahbod, CC BY-SA 3.0, https://commons.wikimedia.org/w/index.php?curid=3532181

Integer factorization is in NP.

Current assumption: it is **not** in P and **not** NP-hard => not NP-complete.

Integer Programming/Optimization is in NP.

Proven to be NP-hard and therefore NP-complete

A proof of P=NP might not be constructive and include a huge constant.

$P \neq NP$ — Open

Quantum Supremacy

https://www.quantamagazine.org/john-preskill-explains-quantum-supremacy-20191002

In 2012, I proposed the term "quantum supremacy" to describe the point where quantum computers can do things that classical computers can't, regardless of whether those tasks are useful.

https://scottaaronson.blog/?p=4317

Scott's Supreme Quantum Supremacy FAQ!

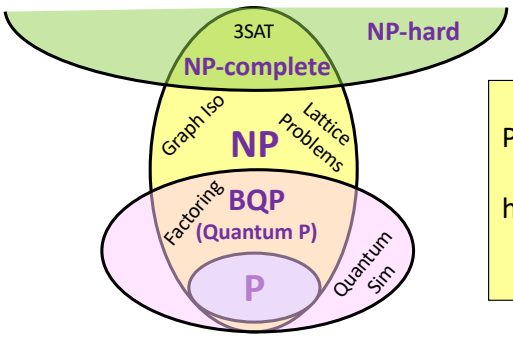
Q1. What is quantum computational supremacy?

Often abbreviated to just "quantum supremacy," the term refers to the use of a quantum computer to solve ***some*** well-defined set of problems that would take orders of magnitude longer to solve with any currently known algorithms running on existing classical computers—and not for incidental reasons, but for reasons of asymptotic quantum complexity. The emphasis here is on being as sure as possible that the problem *really was* solved quantumly and *really is* classically intractable, and ideally achieving the speedup *soon* (with the noisy, non-universal QCs of the present or very near future). If the problem is also *useful* for something, then so much the better, but that's not at all necessary. The Wright Flyer and the Fermi pile weren't useful in themselves.

Slide by Scott Aaronson: https://www.scottaaronson.com/talks/speedup.ppt

An Inconvenient Truth

If we set aside **NP**-complete problems, there just aren't that many compelling candidates left for exponential quantum speedups! **(And for many of those, we *do* have exponential speedups, and for many of the rest we have polynomial ones)**



P≠BQP, NP⊄BQP: Plausible conjectures, which we have no hope of proving given the current state of complexity theory

Integer Linear Program (ILP)

$$\min_{x \in X} f(x) \text{ with } X = \{x \in \mathbb{Z}^n : Ax \leq b, \underline{l} \leq x \leq \overline{u}\}, f(x) = c^T x$$

Solving this problem is in general NP-hard, even if have $x \in \{0,1\}$. However, is important to note that without the integrality requirement, i.e., for $x \in \mathbb{R}^n$ the problem can be solved in polynomial time.

What does NP-hard mean:

(1) If we get **some** x, we can check in polynomial time whether it belongs to X and compute $f(x)$.

(2) Finding the minimum x might, in the worst case, be as difficult to solve as any other problem in NP.

(3) As of today, no algorithm is known that has better than exponential worst case runtime complexity to do so. And there is little hope to find one. Note: Integer factorization is not NP-hard!

What does NP-hard not necessarily mean:

(1) Finding some $x \in X$ is difficult. Might be difficult, but doesn't have to.

(2) Finding the minimum $x \in X$ is difficult.
Finding it might actually be easy, proving it is the minimum is the hard part.

(3) Large size instances of NP-hard problems are intractable or unsolvable in principle.

https://arxiv.org/abs/2111.03011

Solving the sampling problem of the Sycamore quantum supremacy circuits

Feng Pan, Keyang Chen, Pan Zhang

We study the problem of generating independent samples from the output distribution of Google's Sycamore quantum circuits with a target fidelity, which is believed to be beyond the reach of classical supercomputers and has been used to demonstrate quantum supremacy. We propose a new method to classically solve this problem by contracting the corresponding tensor network just once, and is massively more efficient than existing methods in obtaining a large number of uncorrelated samples with a target fidelity. For the Sycamore quantum supremacy circuit with 53 qubits and 20 cycles, we have generated one million uncorrelated bitstrings {s} which are sampled from a distribution P^(s)=|ψ^(s)|2, where the approximate state ψ^ has fidelity F≈0.0037. The whole computation has cost about 15 hours on a computational cluster with 512 GPUs. The obtained one million samples, the contraction code and contraction order are made public. If our algorithm could be implemented with high efficiency on a modern supercomputer with ExaFLOPS performance, we estimate that ideally, the simulation would cost a few dozens of seconds, which is faster than Google's quantum hardware.

https://journals.aps.org/prl/abstract/10.1103/PhysRevLett.129.090502

Notes on NP-hard

Subset-sum: Given $x \in \mathbb{Z}^n$ and $b \in \mathbb{Z}$ exists a subset $S \subseteq \{1, \dots, n\}$ such that $\sum_{i \in S} x_i = b$,
Subset-sum is NP-hard (even with $x_i \in \mathbb{N}$)
However: Subset-sum bounded by a constant $C > x_i \in \mathbb{N}$ is in P
On a Computer x_i is always bounded if we restrict ourselves to, say 64-bit integers.

For a mathematician nearly all numbers are larger than 2^{64}. (there are only finite many exceptions)

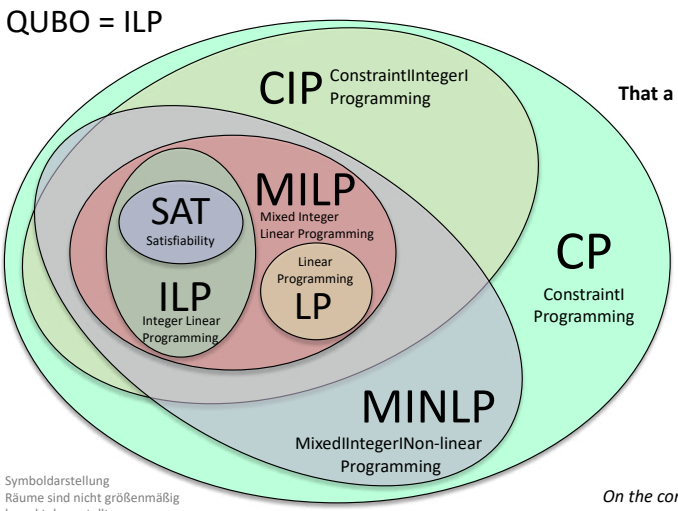
But in real-life even the US dept in pennies is just $100 \cdot 30 \cdot 10^{12}$
2^{64} = 18.446.744.073.709.551.616
$3 \cdot 10^{15}$ = 3.000.000.000.000.000

Adding numbers up to this is O(1)

NP-hard problems are extremely difficult as a class and in theory. In practice, it depends.

Problem classes

QUBO = ILP



Symboldarstellung
Räume sind nicht größenmäßig korrekt dargestellt.

SAT is NP-complete, so is ILP and everything that encompasses it.

That a class of problems is NP-complete should not stop you from solving instances of this class to proven optimality

Stephen Cook (1971)
The Complexity of Theorem Proving Procedures
doi:10.1145/800157.805047

Richard M. Karp (1972)
Reducibility Among Combinatorial Problems
doi:10.1007/978-1-4684-2001-2_9

Leo Liberti (2019)
Undecidability and Hardness in MINLP
https://doi.org/10.1051/ro/2018036

Matthias Köppe (2010)
On the complexity of nonlinear mixed-integer optimization
arXiv:1006.4895v1

Examples for finding some $x \in X$ is difficult. Might be, but doesn't have to.

We can write our problem as a decision problem (and minimize by binary search):

$$\boldsymbol{X_k} \neq \emptyset \text{ ? with } \boldsymbol{X_k} = \left\{\boldsymbol{x} \in \mathbb{Z}^{\boldsymbol{n}}: \boldsymbol{Ax} \leq \boldsymbol{b}, \underline{\boldsymbol{l}} \leq \boldsymbol{x} \leq \overline{\boldsymbol{u}} \ \wedge \boldsymbol{c}^{\boldsymbol{T}}\boldsymbol{x} = \boldsymbol{k}\right\}$$

In this case finding some x is equivalent to solving the problem.

Or, using some suitable big constant M, we can move the constraints into the objective:

$$\min_{\boldsymbol{x} \in \boldsymbol{X}} \boldsymbol{f}(\boldsymbol{x}) \text{ with } \boldsymbol{X} = \left\{\boldsymbol{x} \in \mathbb{Z}^{\boldsymbol{n}}: \underline{\boldsymbol{l}} \leq \boldsymbol{x} \leq \overline{\boldsymbol{u}}\right\}, \boldsymbol{f}(\boldsymbol{x}) = \boldsymbol{c}^{\boldsymbol{T}}\boldsymbol{x} + \boldsymbol{M}(\boldsymbol{b} - \boldsymbol{Ax})$$

now it is obviously trivial to find some $x \in X$.

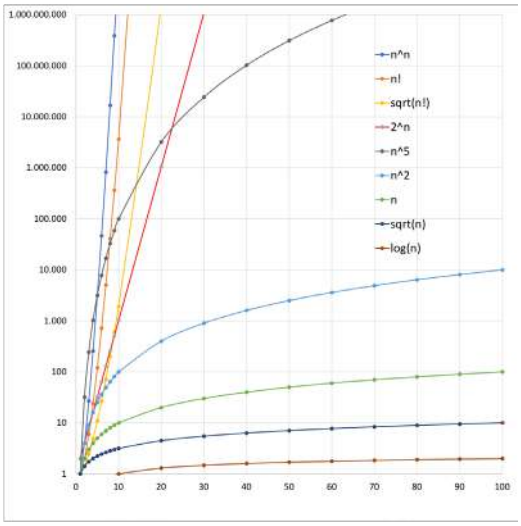
This difference in difficulty is one reason why people believe $P \neq NP$

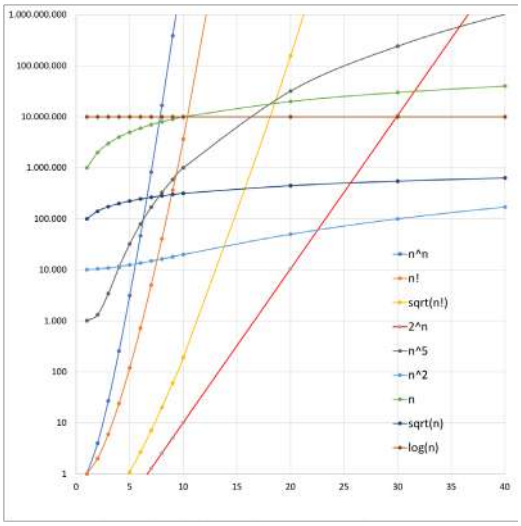
Note: To solve an ILP, i.e., to optimality two things must be done:

(1) $\exists$:Find the minimum $x^* \in X$.

(2) $\forall$:Prove there exists no $x^* \in X$ with $f(x) < f(x^*)$.
This is equivalent to showing: $\left\{\boldsymbol{x} \in \mathbb{Z}^{\boldsymbol{n}}: \boldsymbol{Ax} \leq \boldsymbol{b}, \underline{\boldsymbol{l}} \leq \boldsymbol{x} \leq \overline{\boldsymbol{u}} \ \wedge \boldsymbol{f}(\boldsymbol{x}) < \boldsymbol{f}(\boldsymbol{x}^*)\right\} = \emptyset$

What $\mathcal{O}$ = *Order of* means: both pictures are equally correct





What are QUBOs?

QUBO : Quadratic Unconstraint Binary Optimization
UBQP : Unconstrained Binary Quadratic Program
(BIQ : Binary Integer Quadratic problem)

$$\min_{x \in \{0,1\}^n} x^T Q x$$

- x is a vector of binary variables, Q is a square $n \times n$ matrix of constants
- Since QUBOs are unconstraint, any 0/1 vector is a feasible solution
- All QUBOs can be brought to the form where Q is symmetric or upper triangular
- Solving QUBO (in general) is NP-hard
- Since x is binary, $x_i = x_i^2$ holds $\Longrightarrow$ The coefficients of the linear terms of the objective function correspond to the diagonal entries of Q

https://arxiv.org/abs/2207.13630

Copositive programming for mixed-binary quadratic optimization via Ising solvers

Robin Brown, David E. Bernal Neira, Davide Venturelli, Marco Pavone

Recent years have seen significant advances in quantum/quantum-inspired technologies capable of approximately searching for the ground state of Ising spin Hamiltonians.

The promise of leveraging such technologies to accelerate the solution of difficult optimization problems has spurred an increased interest in exploring methods to integrate Ising problems as part of their solution process, with existing approaches ranging from direct transcription to hybrid quantum-classical approaches rooted in existing optimization algorithms. Due to the heuristic and black-box nature of the underlying Ising solvers, many such approaches have limited optimality guarantees.

While some hybrid algorithms may converge to global optima, their underlying classical algorithms typically rely on exhaustive search, making it unclear if such algorithmic scaffolds are primed to take advantage of speed-ups that the Ising solver may offer.

In this paper, we propose a framework for solving mixed-binary quadratic programs (MBQP) to global optimality using black-box and heuristic Ising solvers. We show the exactness of a convex copositive reformulation of MBQPs, which we propose to solve via a hybrid quantum-classical cutting-plane algorithm. The classical portion of this hybrid framework is guaranteed to be polynomial time, suggesting that when applied to NP-hard problems, the complexity of the solution is shifted onto the subroutine handled by the Ising solver.

BIP to QUBO

BIP

$$\min_{x \in \{0,1\}^n} c^T x$$
$$\text{s.t. } Ax \leq b$$

QUBO

$$\min_{x \in \{0,1\}^n} c^T x^2 + P(Ax + Is - b)^T (Ax + Is - b)$$

=>

$$\min_{x \in \{0,1\}^n} x^T Q x$$

where $Q \in \mathbb{R}^{n \times n}$ and symmetric

BIPs can be reformulated as QUBOs by putting the constraints into the objective with a penalty term P. The penalty should be zero if and only if the constraint is fulfilled.

Glover, Kochenberger, Du (2019):
A Tutorial on Formulating and Using QUBO Models
arXiv:1811.11538

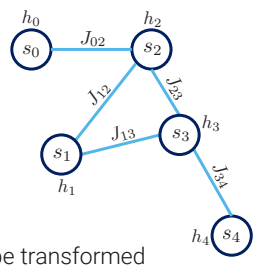
From *QUBO transformation*, E. Rodríguez-Heck, Gurobi Days Digital May 03,2022

Ising Hamiltonian

- Quantum annealing is a special-purpose device that finds the minimum energy of an Ising Hamiltonian heuristically

Linear terms (biases, local fields) | Quadratic terms (couplers) | Spins

$$H_{\text{Ising}} = \sum_{i \in V} h_i s_i + \sum_{\{i,j\} \in E} J_{ij} s_i s_j, \quad s_i \in \{-1, +1\}$$



- A Quadratic Unconstrained Binary Optimization (QUBO) problem can be transformed into an Ising Hamiltonian with a simple algebraic manipulation:

$$x^T Q x \xrightarrow[s_i \in \{-1,1\}]{x_i = \frac{s_i + 1}{2}} H_{\text{Ising}}$$

Max-Cut

Graph formulation $G = (V, E, w)$

$$\max_{S,T} \sum_{i \in S, j \in T} w_{ij} \quad \text{with } S \subset V, T \subset V, S \cap T = \emptyset, S \cup T = V$$

Ising formulation

$$\max \frac{1}{4} \sum_{i=1}^{n} \sum_{j=1}^{n} w_{ij}(1 - x_i x_j), x_k \in \{1, -1\} \text{ for } k \in \{1, \dots, n\}$$

Binary Linear Programing formulation:

$$\max \sum_{i=1}^{n} \sum_{j=1}^{n} w_{ij} z_{ij}$$
$$z_{ij} \leq x_i + x_j$$
$$z_{ij} \leq 2 - (x_i + x_j)$$
$$x_k \in \{0,1\}$$
$$z_{ij} \in \{0,1\}$$

$x_i = x_j = 0 \Rightarrow z_{ij} = 0$

$x_i = x_j = 1 \Rightarrow z_{ij} = 0$

Binary Quadratic Programing formulation:

$$\max_{x \in \{0,1\}^n} \sum_{i=1}^{n} \sum_{j=1}^{n} w_{ij}(x_i + x_j - 2x_{ij})$$

Can be written as:

$$\min_{x \in \{0,1\}^n} x^T Q x$$

Independent (stable) Set

Given a graph $G = (V, E)$, find the maximum size independent set of nodes

$$\max_{S \subseteq V} |S| \text{ with } u, v \in S \Longrightarrow (u, v) \notin E$$

Classical Constraint	Equivalent Penalty
$x + y \leq 1$	$P(xy)$
$x + y \geq 1$	$P(1 - x - y + xy)$
$x + y = 1$	$P(1 - x - y + 2xy)$
$x \leq y$	$P(x - xy)$
$x_1 + x_2 + x_3 \leq 1$	$P(x_1x_2 + x_1x_3 + x_2x_3)$
$x = y$	$P(x + y - 2xy)$

Table of a few Known constraint/penalty pairs

Binary Linear Programing formulation:

$$\max_{x \in \{0,1\}^{|V|}} \sum_{v \in V} x_v \quad \text{subject to} \quad x_u + x_v \leq 1 \quad \text{for all } (u, v) \in E$$

Unconstraint Quadratic Binary Programing formulation:

$$\min_{x \in \{0,1\}^{|V|}} -\sum_{v \in V} x_v^2 + P \cdot \underbrace{\sum_{(u,v) \in E} x_u \cdot x_v}_{=0}$$

Some more notes on QUBOs

- In theory possible to model all ILP and SAT, but not MILP
- **Number of constraints is not important, only the number of Variables!**
- Constraint with large support result in dense Q, cardinality constraint is worst
- While most available software works on dense instances this limits the problem size dramatically
- General $P(Ax-b)^T(Ax-b)$. Beware numerical trouble!
- QUBO is unconstraint and pure binary, nearly all heuristic ideas work nicely
- Preprocessing is limited compared to ILP/SAT
- Impossible to distinguish between feasibility and optimization. When to stop?
- Best for problems where there is some "natural" quadratic formulation. (But, e.g., QAP is dense)
- On the primal side hard to win against problem specific heuristic approaches
- How to get good lower bounds? (LP/SDP/Newton-Bracket)

Clause #	Clause	Quadratic penalty
1	$x_1 \vee x_2$	$1 - x_1 - x_2 + x_1x_2$
2	$x_1 \vee \bar{x}_2$	$x_2 - x_1x_2$
3	$\bar{x}_1 \vee x_2$	$x_1 - x_1x_2$
4	$\bar{x}_1 \vee \bar{x}_2$	x_1x_2
5	$\bar{x}_1 \vee x_3$	$x_1 - x_1x_3$
6	$\bar{x}_1 \vee \bar{x}_3$	x_1x_3
7	$x_2 \vee \bar{x}_3$	$x_3 - x_2x_3$
8	$x_2 \vee x_4$	$1 - x_2 - x_4 + x_2x_4$
9	$\bar{x}_2 \vee x_3$	$x_3 - x_2x_3$
10	$\bar{x}_2 \vee \bar{x}_3$	x_2x_3
11	$x_3 \vee x_4$	$1 - x_3 - x_4 + x_3x_4$
12	$\bar{x}_3 \vee \bar{x}_4$	x_3x_4

Glover, Kochenberger, Du (2019):
A Tutorial on Formulating and Using QUBO Models
arXiv:1811.11538

Transformations

Modelling a linear relationship quadratic:

$$x = x^2, x \in \{0,1\}$$

Modelling multiplication of binary variables $y = x_1 \cdot x_2 = x_1 \wedge x_2$, for $x_1, x_2 \in \{0,1\}$:

$$y \leq x_1$$
$$y \leq x_2$$
$$y \geq x_1 + x_2 - 1$$
$$y \in \{0,1\}$$

Modelling general Integer variables from Binary variables:

$$\sum_{i=0}^{n-1} 2^i x_i \leq N, x_i \in \{0,1\} \Leftrightarrow z \in \{0, \dots, \min(2^n - 1, N)\}$$

Alternatively:

$$\sum_{i=1}^{N} i \cdot x_i = z \quad \text{and} \quad \sum_{i=1}^{N} x_i = 1 \quad \text{for } x_i \in \{0,1\}, z \in \{1, \dots, N\}$$

Once upon a time

Mathematical Programming 68 (1995) 213–237

Computational experience with a difficult mixed-integer multicommodity flow problem☆

D. Bienstock*, O. Günlük

Department of Industrial Engineering and Operations Research, Columbia University, New York, NY 10027, USA

Received 27 April 1993; revised manuscript received 17 June 1994

Reminder

Solving an optimization problem means

$$\min_{x \in X} f(x) \text{ with } X = \{x, b, \underline{l}, \overline{u} \in \mathbb{Z}^n : g(x) \leq b, \underline{l} \leq x \leq \overline{u}\}$$

(1) finding a feasible solution (often trivial, but not always)

(2) Proving it is the best one

Solving a decision (feasibility) Problem means

$$X \neq \emptyset \Leftrightarrow \min_{x \in X} 0 \text{ with } X = \{x, b, \underline{l}, \overline{u} \in \mathbb{Z}^n : g(x) \leq b, \underline{l} \leq x \leq \overline{u}\}$$

(1) Finding a solution

A QUBO by definition is an unconstraint optimization problem

$$\min_{x \in \{0,1\}^n} x^t Q x$$

i.e., finding a solution is absolutely trivial (e.g., $x = 0$)

We can transform between the 3, so be careful what is done exactly and how a solution translates back.

Best solutions so far

	ID	Objective	Exact	Int. Viol	Cons. Viol	Obj. Viol	Submitter	Date	Description
3	4	665.5714	665.5714	0	0	0	Yuji Shinano	2020-04-16	Obtained with ParaSCIP in 2014
1	3	667.5577		0	0	0	Edward Rothberg	2019-12-13	Obtained with Gurobi 9.0
4	2	676.5630		0	0	0	Robert Ashford and Alkis Vazacopoulus	2019-12-18	Found using ODH|CPlex
2	1	691.8961	691.8961	0	0	0	-	2018-10-12	Solution found during MIPLIB2017 problem selection.

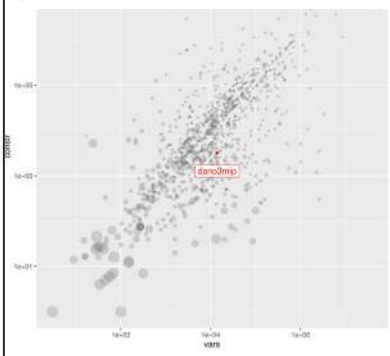
dano3mip

variable_bound | cardinality | mixed_binary

Submitter	Variables	Constraints	Density	Status	Group	Objective	MPS File
Daniel Bienstock	13873	3202	1.79317e-0	open	dano	665.571428571428*	dano3mip.mps.gz

Telecommunications applications

Imported from MIPLIB2010.



Variables	13873
Constraints	3202
Binaries	552
Integers	0
Continuous	13321
Implicit Integers	0
Fixed Variables	0
Nonzero Density	0.00179317
Nonzeroes	79655

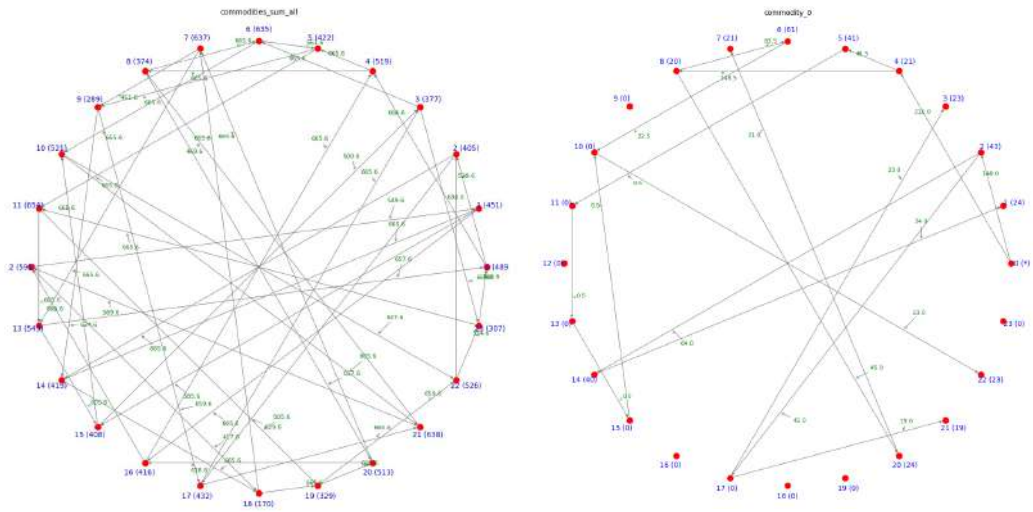
MIPLIB 3.0

January 1996

: (To download in Netscape, click while pressing the SHIFT key)

- The MIPLIB 3.0 Problem Set as a compressed tar file
- MIPLIB Problem Set:
  - 10teams
  - air03
  - air04
  - air05
  - arki001
  - bell3a
  - bell5
  - blend2
  - cap6000
  - dano3mip
  - danoint
  - dcmulti

Best know solution



ICONG(2)

$$\min z \qquad k, i, j \in \{1, \dots, 24\}$$

$$\text{s.t.} \quad \sum_{j \neq i} x_{ij} = 2 \quad \text{for all } i \qquad (1)$$

$$\sum_{j \neq i} x_{ji} = 2 \quad \text{for all } i \qquad (2)$$

$$f_{kij} \leq M \cdot x_{ij} \quad \text{for all } k, i \neq j \qquad (3)$$

$$\sum_{j \neq i} f_{kij} - \sum_{j \neq i} f_{kij} = s_i^k \quad \text{for all } \mathrm{i}, k \qquad (4)$$

$$\sum_{k} f_{kij} \leq z \quad \text{for all } i \neq j \qquad (5)$$

$$f_{kij} \geq 0 \quad \text{for all } k, i \neq j \qquad (6)$$

$$x_{ij} \in \{0,1\} \quad \text{for all } i \neq j \qquad (7)$$

```
Root relaxation objective:  30.28571
dano3mip root relaxation : 576.2316

Best known solution:       665.5714
```

$$M = 666$$

	30.28571	95.4%
	350.63706	47.3%
	425.04748	36.1%
	436.46971	34.4%
Gurobi 9.5.1 root cuts	439.96568	33.9%
original model	577.82468	–
with some cuts removed	577.88492	–

Make it all integer

$$\min z$$

s.t.

$$\sum_{j \neq i} x_{ij} = 2 \quad \text{for all } i$$

$$\sum_{j \neq i} x_{ji} = 2 \quad \text{for all } i$$

$$f_{kij} \leq M \cdot x_{ij} \quad \text{for all } k, i \neq j$$

$$\sum_{j \neq i} f_{kij} - \sum_{j \neq i} f_{kij} = \mathbf{1000} \cdot s_i^k \quad \text{for all i}, k$$

$$\sum_{k} f_{kij} \leq z \quad \text{for all } i \neq j$$

$$f_{kij} \in \{\mathbf{0}, \ldots, \mathbf{1000}\} \quad \text{for all } k, i \neq j$$

$$x_{ij} \in \{0,1\} \quad \text{for all } i \neq j$$

```
Scale by 1000 for 3 decimal digits

Root relaxation objective 3028.571
```


Transforming ICONG(2) into a QUBU (so we can solve on a QC)

Todo:

1. Make it all integer
2. Make it equality ($Ax = b$)
3. Make it binary
4. Put the constraints into the objective

Making it binary

Modelling general Integer variables $z \in \{0, \dots, N\}$ using Binary variables $x_i \in \{0,1\}$:

$\sum_{i=1}^{N} x_i$	$\mathcal{O}(N)$	representation is not unique
$\sum_{i=1}^{N} i \cdot x_i \quad \wedge \quad \sum_{i=1}^{N} x_i \leq 1$	$\mathcal{O}(N)$	we need an extra constraint, which quadratic looks like $\sum_{i,j \in \{1,\dots N\}, i \neq j} x_i \cdot x_j$, and is dense
$\sum_{i=1}^{n:=\lceil\sqrt{N}\rceil} i \cdot x_i + \sum_{i=1, i \in R_1}^{n-1} i \cdot x_{n+i}$	$\mathcal{O}(\sqrt{N})$	representation is not unique
$\sum_{i=0}^{n:=\lfloor\log_2 N\rfloor} 2^i \cdot x_i + \sum_{i=1, 2^i \in R_2}^{n-1} 2^i \cdot x_{n+i} N$	$\mathcal{O}(\log_2 N)$	switching from $2^n - 1$ to 2^n changes all involved variables, only unique for powers of 2

with $R_1 := \{ i \in \{1, \dots, n-1\} | \sum_{i \in R} i = N - \sum_{i=1}^{n:=\lceil\sqrt{N}\rceil} i \wedge |R| \text{ minimal}\}$, $R_2 := \{ i \in \{1, \dots, n-1\} | \sum_{i \in R} i = N - \sum_{i=1}^{n:=\lfloor\log_2 N\rfloor} 2^i \wedge |R| \text{ minimal}\}$

Transforming an ILP to a system of equations

$$\min_{x \in \mathbb{Z}} -2x_1 - x_2 \quad \text{subject to}$$

$$\begin{array}{rcrcr}
80x_1 & - & 10x_2 & \leq & -7 \\
x_1 & + & 20x_2 & \leq & 120 \\
-5x_1 & - & 20x_2 & \leq & -32 \\
7x_1 & + & 2x_2 & \leq & 48 \\
-2x_1 & + & 2x_2 & \leq & -7 \\
3x_1 & + & 4x_2 & \geq & 5 \\
3x_1 & + & 4x_2 & \leq & 5 \\
x & & & \leq & l \\
x & & & \geq & u
\end{array}
\quad \Longleftrightarrow \quad
\begin{array}{rcrcrcr}
\multicolumn{7}{l}{\min_{x \in \mathbb{Z}} -2x_1 - x_2 \quad \text{subject to}} \\
80x_1 & - & 10x_2 & + & s_1 & = & -7 \\
x_1 & + & 20x_2 & + & s_2 & = & 120 \\
5x_1 & + & 20x_2 & - & s_3 & = & 2 \\
7x_1 & + & 2x_2 & + & s_4 & = & 48 \\
2x_1 & - & 2x_2 & - & s_5 & = & 7 \\
3x_1 & + & 4x_2 & & & = & 5 \\
 & & l & \leq & x & \leq & u \\
 & & & & s & \geq & 0
\end{array}$$

$$\begin{aligned}
&\min c^T x \\
&\text{s.t.} \quad Ax \leq b \\
&\qquad l \leq x \leq u \\
&\qquad\quad x \in \mathbb{Z}^m
\end{aligned}
\quad \Longleftrightarrow \quad
\begin{aligned}
&\min c^T x \\
&\text{s.t.} \quad Ax + I_m s = b \\
&\qquad l \leq x \leq u \\
&\qquad\quad x \in \mathbb{Z}^m \\
&\qquad\quad s \geq 0 \\
&\qquad\quad s \in \mathbb{R}^m
\end{aligned}$$

Converting to equation form adds m variables.

$$A \in \mathbb{R}^{m \times n}, c \in \mathbb{R}^n, b, l, u \in \mathbb{R}^m$$

Faster than a Quantum computer

MAX-Reaper:
An exact solver for QUBO and Max-Cut

Faster exact solution of sparse MaxCut and QUBO problems

Daniel Rehfeldt, Thorsten Koch, Yuji Shinano

doi: 10.48550/arXiv.2202.02305

Putting the constraints into the objective

$$\min z$$

$$\text{s.t.} \quad \sum_{j \neq i} x_{ij} = 2 \quad \text{for all } i$$

$$\sum_{j \neq i} x_{ji} = 2 \quad \text{for all } i$$

$$f_{kij} + s_{kij} - M \cdot x_{ij} = 0 \quad \text{for all } k, i \neq j$$

$$\sum_{j \neq i} f_{kij} - \sum_{j \neq i} f_{kij} - M \cdot s_i^k = 0 \quad \text{for all } i, k$$

$$\sum_{k} f_{kij} + s_{ij} - z = 0 \quad \text{for all } i \neq j$$

$$x_{ij} \in \{0,1\} \quad \text{for all } i \neq j$$

$$f_{kij} \in \{0, \dots, M\} \quad \text{for all } k, i \neq j$$

$$s_{kij} \in \{0, \dots, M\} \quad \text{for all } k, i \neq j$$

$$s_{ij} \in \{0, \dots, M\} \quad \text{for all } i \neq j$$

$$\min_{x \in \{0,1\}^n} c^T x^2 + P(Ax + Is - b)^T (Ax + Is - b)$$

Nodes, commodities: $k, i, j \in \{1, \dots, 24\}$

Integer to binary: $M = 666 \rightarrow \lfloor \log_2 666 \rfloor + 4 = 13$

for all $i \neq j \rightarrow 24 \times 23 = 552$

for all $k, i \neq j \rightarrow 24 \times 552 = 13248$

Binary variables total:

$$552 + 13 \times (13248 + 13248 + 552) = 352{,}176$$

$$\Longrightarrow Q \in \mathbb{Z}^{352176 \times 352176}$$

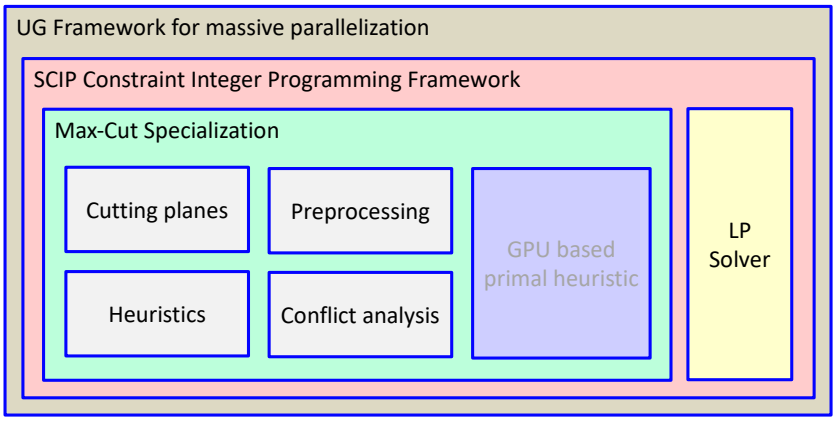
i.e., we need at least 352,176 qubits

The range of the coefficients in Q is at least up to

$$666^3 = 295{,}408{,}296$$

And Q will not be particular sparse!

Overall Architecture



The final code will be able to completely utilize existing HPC resources:
Run multicore on many nodes employing GPU accelerators in parallel.

Branch-and-cut algorithm for sparse QUBO and Max-Cut problems

Main components:

- Presolving
- Simple Domain Propagation
- Problem-specific cutting planes (optimized implementation)
- Primal heuristics
- Parallel branch-and-bound search via UG framework (still experimental).

©2016 Two Strings LLC

About 15,000 lines of code + 1M lines for SCIP and LP solver.

Already faster than existing QUBO/Max-Cut solvers on sparse QUBO and Max-Cut benchmarks in many cases.

Preprocessing

Whenever one tries to solve real-world (and other) problem instances, the first thing to do is to implement some preprocessing. We found some new techniques to improve performance.

As one can see, the IsingChain and the K64-chimera instances practically vanish.

		base preprocessing		+new techniques		relative change	
Test-set	#	|V| [%]	|E| [%]	|V| [%]	|E| [%]	|V| [%]	|E| [%]
IsingChain	30	6.1	0.8	1.1	<0.05	**-82.0**	**<-93.8**
K64-chimera	80	3.1	4.6	3.1	4.6	0.0	0.0
Kernel	14	24.1	30.1	16.4	20.6	**-32.0**	**-31.6**
Mannino	4	64.1	69.3	63.2	68.7	**-1.4**	**-0.9**
Torus	18	80.6	87.5	78.5	85.2	**-2.6**	**-2.6**
$W01_{100}$	10	99.1	94.8	99.1	94.8	0.0	0.0
DIMACS	4	97.0	98.9	96.9	98.9	**-0.1**	0.0
$PM1s_{100}$	10	99.7	99.9	99.7	99.9	0.0	0.0

Selected Benchmarks (there are more)

Name	#	|V|	|E|	Description
DIMACS	4	512-3375	1,536-10,125	Instances from 7th DIMACS Challenge
IsingChain	30	100-300	4,950-44,850	Max-Cut instances from physics applications
QBLIB	22	120-1,225	602-34,876	QUBOs from QPLIB instances
Mannino	4	48-487	1,128-8,511	Frequency assignment problems
I64-dwave	80	2,049	8,064	Max-Cut instances from D-Wave Chimera graphs
Paintshop	30	10-1,000	22-2,498	QUBO Instances modelling the binary paintshop probnlem
Torus	18	100-343	200-1,029	Max-Cut instances from physics applications
Kernel	14	33-2,888	91-2,981	Instances from various sources
$GKA_{a\text{-}d}$	35	20-125	204-7,788	Randomly generated

|V|: Number of vertices (Max-Cut), or n of matrix $Q \in \mathbb{R}^{\{n \times n\}}$ (QUBO).

|E|: Number of edges (Max-Cut), or number of non-zero entries in Q (QUBO)

Quantum Annealing versus Digital Computing: An Experimental Comparison

M. Jünger, E. Lobe, P. Mutzel, G. Reinelt, F. Rendl, G., T. Stollenwerk. 2021.
ACM J. Exp. Algorithmics 26, Article 1.9, doi: 10.1145/3459606

This is paper makes a very detailed and precise comparison with the following conclusion:

> *"However, we should stress the fact that exact optimization requires a lot of time to prove optimality, and thus it is not fair to compare their times with the heuristic times, but even with this additional burden, the exact algorithms are faster than D-Wave on a large portion of the sample.*
>
> *[…]*
>
> *It may well be (and we hope) that the exciting new quantum computer technology will make leaps in the future, but in our experiments, we have certainly not observed superior performance of quantum annealing in comparison to "classical" methods."*

Comparison of the new solver MAX-Reaper (new) and Gurobi 9.5 MIQP (Grb)

		# solved		mean time (sh. geo. mean)			maximum time		
Test-set	#	Grb	new	Grb [s]	new [s]	speedup	Grb [s]	new [s]	speedup
$PM1s_{100}$	10	10	10	192.3	20.9	**9.20**	303.3	48.4	**6.27**
$W01_{100}$	10	10	10	44.1	3.1	**14.23**	97.1	21.5	**4.52**
Kernel	14	14	14	0.7	0.1	**7.00**	14.3	1.1	**13.00**
IsingChain	30	30	30	1.3	<0.05	**>26.00**	41.0	<0.05	**>820.00**
Torus	18	18	18	3.8	0.4	**9.50**	628.0	7.6	**82.63**
K64-chimera	80	80	80	90.1	1.5	**60.07**	195.4	6.0	**32.57**
QPLIB	22	8	**13**	687.4	165.5	**4.15**	3600	3600	1.00
BQP100	10	10	10	0.1	0.1	1.00	0.2	0.3	0.67
BQP250	10	0	**7**	3600	610.6	**5.90**	3600	3600	1.00
BE120.3	10	9	**10**	265.6	50.1	**5.30**	3600	525.1	> **6.86**
BE250	10	0	**8**	3600	571.8	**6.30**	3600	3600	1.00
$GKA_{a\text{-}d}$	35	29	**31**	6.5	6.1	**1.07**	3600	3600	1.00

Time limit: 1 h , single-threaded, Intel Xeon Gold 5122 3.60 GHz, 96 GB, Sep. 2022

Running Parallel

Results with non-negated weights for torusg3/pm3

88 core E7-8880v4@2.20GHz

Name	gap [%]	new primal	previous primal
torusg3-15	**opt**	**286626481**	282534518
toruspm3-15-50	1.3	3010	2968
QPLIB_3693	1.3	-1152	-1148
QPLIB_3850	1.7	-1194	-1192

	primal-dual gap [%]				run time [s]			
Name	Grb-T1	Grb-T8	new-T1	new-T8	Grb-T1	Grb-T8	new-T1	new-T8
torusg3-8	0.0	0.0	0.0	0.0	1494.2	1178.5	8.5	9.3
toruspm3-8-50	1.8	1.8	0.5	0.0	>3600	>3600	>3600	1415.8
torusg3-15	6.8	3.4	1.3	0.4	>3600	>3600	>3600	>3600
toruspm3-15-50	9.5	12.2	2.3	2.3	>3600	>3600	>3600	>3600
mannino_k487a	0.0	0.0	0.0	0.0	3.5	10.7	1.1	1.3
mannino_k487b	0.0	0.0	0.0	0.0	9.2	80.5	2.9	2.8
mannino_k487c	0.1	0.0	0.1	0.0	>3600	3176.7	>3600	398.2
mannino_k48	0.0	0.0	0.0	0.0	0.1	0.4	2.7	3.8

Comparing to Mc-Sparse

			# B&B nodes		run time	
Name	|V|	|E|	MS	new	MS [s]	new [s]
pm1s_100.3	100	495	341	741	48.2	48.0
pw01_100.0	100	495	171	179	20.0	8.5
mannino_k487b	487	5391	1	15	167.3	4.3
bio-diseasome	516	1188	1	1	9.5	0.6
ca-netscience	379	914	1	1	0.1	0.0
g000981	110	188	1	1	0.0	0.0
imgseg_138032	12736	23664	1	1	30.5	3.9

Name	n	nnz	MS	new	MS [s]	new [s]
bqp250-3	250	3092	25	17	414.1	84.1
gka2c	50	813	1	1	0.5	0.3
gka4d	100	2010	129	9	219.6	43.7
gka5c	80	721	1	1	0.1	0.1
gka7a	30	241	1	1	0.0	0.0
be120.3.5	120	2248	111	15	257.7	46.6
be250.3	250	3277	107	47	841.0	150.7

MS data from: Charfreitag, Jünger, Mallach, Mutzel, ALENEX 2022, doi:10.1137/1.9781611977042.5

Mc-Sparse: Exact solutions of sparse maximum cut and sparse unconstrained binary quadratic optimization problems.

MC was used in the comparison with the Quantum Annealers

On the previous slide.

We see still substantial room for performance improvement on solving QUBOs on digital computers.

Experience shows that this improvement will happen esp. on those instances who are now difficult.

https://blogs.nvidia.com/blog/2021/11/09/cuquantum-world-record/

NVIDIA Sets W puting Simulation With cuQuantu

Theoretical computer scientist mean something different by "efficiently"
+ by this definition just finding a solution is **not** "solving" it.

Not the optimal solution.
We need **0.2 s** on a workstation to find this quality solution.
2 days to find the optimal solution, 3 more days to prove optimality.

Driving towa ed the largest ever simulation solving the M antum, our SDK for acceleratin GPU.

In the math w rld, MaxCut is often cited as an example of an optimization no known computer can solve efficiently. MaxCut algorithms are used to design large compute orks, find the optimal layout of chips with billions of silicon pathways and explore the field of ical physics.

[...]

We used the cuTensorNet library in cuQuantum running on NVIDIA' house supercomputer, Selene, to simulate a quantum algorithm to solve the MaxCut problem. Usin 96 GPUs to simulate 1,688 qubits, we were able to solve a graph with a whopping 3,375 vertices. T t's 8x more qubits than the previous largest quantum simulation.

Our solution was also highly accurate, reaching 96% of the best-known answer. We set this new record with an algorithm developed by NVIDIA researchers and an open-source framework.

https://blogs.nvidia.com/blog/2021/11/09/cuquantum-world-record/

NVIDIA Sets World Record for Quantum Computing Simulation With cuQuantum Running on DGX SuperPOD

Driving toward that future, NVIDIA created the largest ever simulation of a quantum algorithm for solving the MaxCut problem using cuQuantum, our SDK for accelerating quantum circuit simulations on a GPU.

In the math world, MaxCut is often cited as an example of an optimization problem no known computer can solve efficiently. MaxCut algorithms are used to design large computer networks, find the optimal layout of chips with billions of silicon pathways and explore the field of statistical physics.

[...]

We used the cuTensorNet library in cuQuantum running on NVIDIA's in-house supercomputer, Selene, to simulate a quantum algorithm to solve the MaxCut problem. Using 896 GPUs to simulate 1,688 qubits, we were able to solve a graph with a whopping 3,375 vertices. That's 8x more qubits than the previous largest quantum simulation.

Our solution was also highly accurate, reaching 96% of the best-known answer. We set this new record with an algorithm developed by NVIDIA researchers and an open-source framework.

Looking back

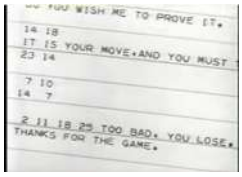
What people through about computers in the early 1960s

The Thinking machines (MIT 1961)

https://www.youtube.com/watch?v=aygSMgK3BEM

https://www.youtube.com/watch?v=5YBIrc-6G-0

If you check the movie, you will find, that many of the ideas mentioned became true.
Only in the movie they mention "within 10 years" while in practice its now 60 years later.



Current state in Quantum Computing

- ▷ Still trying to figure out how to build a QC
- ▷ Doing gate level algorithms
  This is comparable to digital computers in the 50s
- ▷ QC is about where the Wright bros were with planes.
  Getting a 747 took 67 more years.
- ▷ There is much to be done in Quantum computing.
  It will take time and effort.

There is still a lot missing (or waiting to come)

SCIP-Jack is faster than problem specific state-of-the-art solvers for several well-known Steiner problem variants. Example: The Rooted Prize-Collecting Steiner-Tree-Problem (RPC-STP is NP-hard)

Given an undirected graph $G = (V, E)$, a root $r \in V$, edge-weights $c : E \to \mathbb{Q}_{\geq 0}$, and node-weights $p : V \to \mathbb{Q}_{\geq 0}$, a tree $S = (V_S, E\_S)$ in G is required such that $r \in S$ and

$$P(S) := \sum_{e \in E_S} c_e + \sum_{v \in V \setminus V_S} p_v$$

is minimized.

DIMACS fiber network instances, hard instances (>20,000 edges):

- ▷ first publication (Ljubic '04): > 4,000 s*
- ▷ SCIP-Jack at DIMACS (1st) > 100 s
- ▷ best other solver: > 300 s
- ▷ current SCIP-Jack: < 1 s

Rehfeldt, Koch (2021)
Implications, conflicts, and reductions for Steiner trees
doi: 10.1007/978-3-030-73879-2_33

https://physicsworld.com/a/conquering-the-challenge-of-quantum-optimization

Quantum computers are often touted as the solution to all our problems. They are expected to cure disease, alleviate world hunger and even help mitigate the effects of climate change. Fuelled by this enthusiasm, a number of quantum computing firms have started joining established markets. However, despite this interest, there is still a lot of uncertainty around the near-term uses of quantum computers. A crucial question facing quantum researchers today, in both academia and industry, is a pretty fundamental one: what problems are best solved with these devices?

[…]

There is, nevertheless, one point on which everyone seems to agree: it is very likely that some problems exist where quantum optimization is provably superior to classical methods, but these problems will likely occur in the realm of physics and not in finance or industrial operations. *"Nature is quantum. If nature can solve a problem, so should quantum computers,"* says França, who is confident about problems involving molecules or quantum materials like superconductors. *"The strongest case for variational algorithms,"* Aaronson says, *"seems to be on problems that are themselves quantum."*

Training Variational Quantum Algorithms Is NP-Hard

https://journals.aps.org/prl/abstract/10.1103/PhysRevLett.127.120502

Looking forward

"We tend to be too optimistic about the short run, too pessimistic about the long run."

— J. Preskill

- Practical MILP solving on digital computers got arguably faster at least 42% every year (combined hard+software) during the last 40 years. This is an exponential speed-up.
  Progress in Mathematical Programming Solvers from 2001 to 2020, K., Berthold, Pedersen, Vanaret, ZR-21-20
- Regarding our QUBO solver, there are still plenty algorithmic improvements possible. Additionally, we will add GPU-based heuristics and distributed memory parallelization to able to run up to 1 million cores.
- QC likely will evolve for some very specific applications, first likely around Quantum Simulation. This is the original application of QC. It has some strong inherit advantage compared to classical computers.

Recommended further Reading:

NP-complete Problems and Physical Reality, Scott Aaronson, https://arxiv.org/abs/quant-ph/0502072

An Introduction to Quantum Computing, without the Physics, Giacomo Nannicini, https://arxiv.org/abs/1708.03684

https://www.scientificamerican.com/article/will-quantum-computing-ever-live-up-to-its-hype

https://www.linkedin.com/pulse/quantum-computing-hype-bad-science-victor-galitski-1c?trk=public_post-content_share-article_title

https://www.scottaaronson.com/talks/speedup.ppt

https://scottaaronson.blog/?p=5387 (and the rest of his Quantum blog entries)

https://physicsworld.com/a/conquering-the-challenge-of-quantum-optimization

https://m-malinowski.github.io/2022/03/11/forecasting-future-of-qc.html

https://www.technologyreview.com/2022/03/28/1048355/quantum-computing-has-a-hype-problem

3 reasons why a question in industrial optimization might be difficult to answer

1. The question is not well defined,
   i.e., the modeling is intricate. Very often, in industry, problems are involved and multi-layered. Determining a precise definition of the problem, the input and output data, and mapping this to a mathematically well-defined computable optimization problem can be challenging.
2. The data needed to solve the problem is not fully available.
   Many companies struggle hard to consolidate their IT. Getting out precise numbers is often surprisingly hard. One fundamental reason is decomposition, which has been necessary, at least in the past, to counter complexity. As a result, everyone only sees either a very little or very simplified part of the whole picture, and it is very hard to impossible to collect and the data into a coherent set.
3. The resulting problem is computationally hard to solve.
   Since the complexity class of discrete optimization problems often is NP-hard, this is not surprising. However, experience shows, that solving particular instances works surprisingly well and that usually, the main reason for the inability to solve a problem is its size. For example, the likes of SAP, Amazon, Google, Huawei all have extremely large-scale supply-chain-type problems at hand. But not so many others. And there are surprisingly few small challenging real-world problems unless the time allowed for solving is very short.

More Links

Twenty Questions for Donald Knuth

17. Andrew Binstock, Dr. Dobb's: At the ACM Turing Centennial in 2012, you stated that you were becoming convinced that P = N P. Would you be kind enough to explain your current thinking on this question, how you came to it, and whether this growing conviction came as a surprise to you?

https://www.informit.com/articles/article.aspx?p=2213858&WT.mc_id=Author_Knuth_20Questions

Quantum algorithm for stochastic optimal stopping problems with applications in finance

João F. DORIGUELLO

Centre for Quantum Technologies, National University of Singapore, Singapore
joaofd@nus.edu.sg

The famous least squares Monte Carlo (LSM) algorithm [1,2,3] combines linear least square regression with Monte Carlo simulation to approximately solve problems in stochastic optimal stopping theory. In this work, we propose a quantum LSM based on quantum access to a stochastic process, on quantum circuits for computing the optimal stopping times, and on quantum techniques for Monte Carlo. For this algorithm, we elucidate the intricate interplay of function approximation and quantum algorithms for Monte Carlo. Our algorithm achieves a nearly quadratic speedup in the runtime compared to the LSM algorithm under some mild assumptions. Specifically, our quantum algorithm can be applied to American option pricing and we analyze a case study for the common situation of Brownian motion and geometric Brownian motion processes.

References

[1] F. A. Longstaff and E. S. Schwartz. "Valuing American options by simulation: a simple least-squares approach". In: The review of financial studies 14.1 (2001), pp. 113–147. https://doi.org/10.1093/rfs/14.1.113

[2] E. Clément, D. Lamberton, and P. Protter. "An analysis of a least squares regression method for American option pricing". In: Finance and Stochastics 6.4 (2002), pp. 449–471. https://doi.org/10.1007/s007800200071

[3] D. Z. Zanger. "Convergence of a least-squares Monte Carlo algorithm for bounded approximating sets". In: Applied Mathematical Finance 16.2 (2009), pp. 123–150. https://doi.org/10.1080/13504860802516881

Quantum algorithm for stochastic optimal stopping problems with applications in finance

João F. Doriguello

October 10, 2022

arXiv:2111.15332

NUS National University of Singapore

CQT Centre for Quantum Technologies

Collaborators

Alessandro Luongo

Patrick Rebentrost

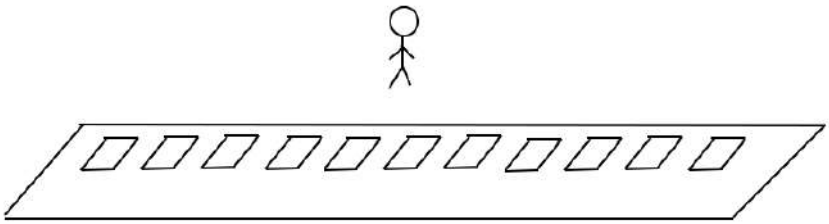
Jinge Bao

Miklos Santha

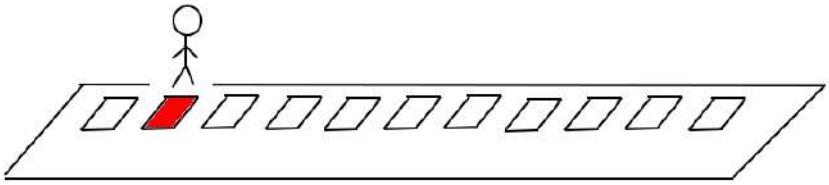
Introduction

Consider the following game:

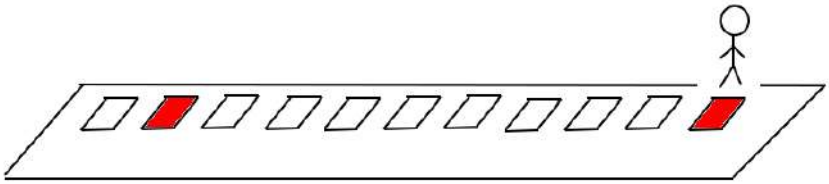
Introduction



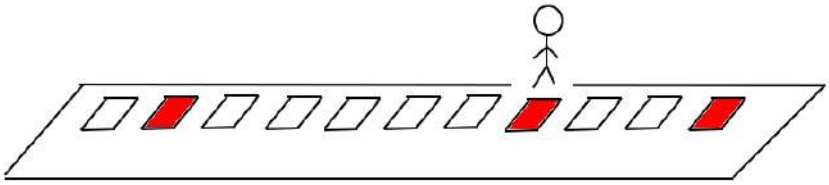
Introduction



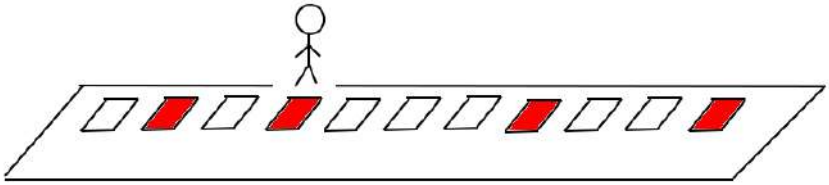
Introduction



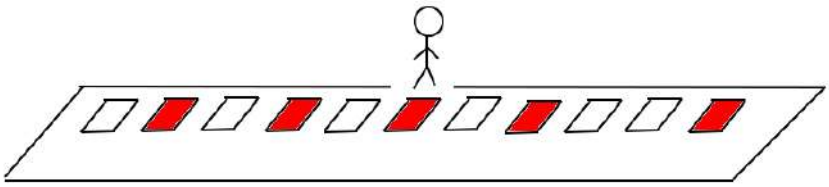
Introduction



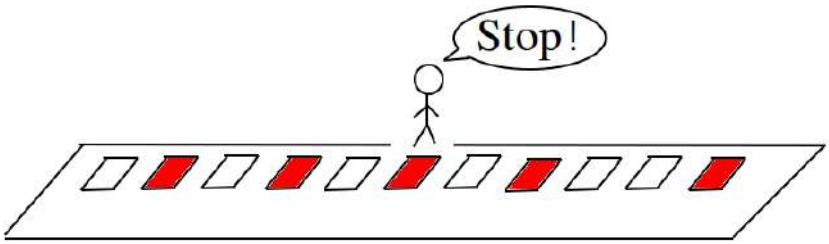
Introduction



Introduction



Introduction



American Options

- Contract that allows the holder to buy/sell an asset at a specific price (strike price).
- It can be exercised at any time until an expiration date.
- Payoff: $\max\{S - K, 0\}$, where K is the strike price and S is the asset's price.

American Options

Problem Statement

Problem Statement: Markov chain and payoff process

- Probability space $(\Omega, \mathcal{F}, \mathbb{P})$.
- Markovian discrete-time stochastic process $(X_t)_{t=0}^T$ with state space $(E, \mathcal{E})$, $E \subseteq \mathbb{R}^d$.
- $X_0 = x_0$ is known.
- Given $z_t \in L^2(E, \rho_t)$, let $(Z_t)_{t=0}^T$ be a non-negative discrete-time stochastic process such that $Z_t = z_t(X_t)$.
- $(X_t)_{t=0}^T$: asset's price; $(Z_t)_{t=0}^T$: American option's price.

Problem Statement: stopping time

- Stopping time: random variable $\tau : \Omega \to \{0, 1, \ldots, T\}$.
- The payoff obtained with τ is $Z_\tau(\omega) := Z_{\tau(\omega)}(\omega)$.

Problem Statement

For $\varepsilon > 0$, approximate

$$\sup_{\tau \text{ stopping time}} \mathbb{E}[Z_\tau]$$

up to additive accuracy ε.

Dynamic program

- It is known[1]

$$\boxed{\sup_{\tau \text{ stopping time}} \mathbb{E}[Z_\tau] = \mathbb{E}[Z_{\tau_0}] = \max\{Z_0, \mathbb{E}[Z_{\tau_1}]\}.}$$

Dynamic program

$$\begin{cases} \tau_T = T, \\ \tau_t = t\mathbf{1}\{Z_t \geq \mathbb{E}[Z_{\tau_{t+1}}|X_t]\} + \tau_{t+1}\mathbf{1}\{Z_t < \mathbb{E}[Z_{\tau_{t+1}}|X_t]\}, \quad 0 \leq t \leq T-1. \end{cases}$$

$\mathbb{E}[Z_{\tau_{t+1}}|X_t]$ are called *continuation values*.

[1] E. Clément, D. Lamberton, and P. Protter. "An analysis of a least squares regression method for American option pricing". (2002)

The LSM algorithm

$$\begin{cases} \tau_T = T, \\ \tau_t = t\mathbf{1}\{Z_t \geq \mathbb{E}[Z_{\tau_{t+1}}|X_t]\} + \tau_{t+1}\mathbf{1}\{Z_t < \mathbb{E}[Z_{\tau_{t+1}}|X_t]\}, \quad 0 \leq t \leq T-1. \end{cases}$$

- Not known how to solve exactly.
- Least squares Monte Carlo (LSM) algorithm by Longstaff and Schwartz in 2001.[2]
  - Approximates $\mathbb{E}[Z_{\tau_{t+1}}|X_t]$ by a set of functions.
  - Evaluates the functions from such set by a Monte Carlo procedure.

[2] F. A. Longstaff and E. S. Schwartz. "Valuing American options by simulation: a simple least-squares approach". (2001)

The classical LSM algorithm

The LSM algorithm: 1st approximation

- $\mathscr{H}_t \subseteq L^2(E, \rho_t)$, $t \in \{0, \dots, T\}$: hypothesis class.
- Approximate $\mathbb{E}[Z_{\tau_{t+1}} | X_t]$ by $f_t \in \mathscr{H}_t$.
- Which f_t? Use **linear least-square regression**.
- For a set $\{e_k : E \to \mathbb{R}\}_{k=1}^m$ of m real functions and $\mathscr{H}_t = \text{span}\{e_k : E \to \mathbb{R}\}_{k=1}^m$,

$$\mathbb{E}[Z_{\tau_{t+1}} | X_t] \approx \alpha_t \cdot \vec{e}(X_t) = \sum_{j=1}^{n} (\alpha_t)_j e_j(X_t),$$

where $\vec{e}(\cdot) := (e_1(\cdot), \dots, e_m(\cdot))^\top$ and

$$\alpha_t = \arg\min_{a \in \mathbb{R}^m} \mathbb{E}\big[(Z_{\tau_{t+1}} - a \cdot \vec{e}(X_t))^2\big].$$

The LSM algorithm: 1st approximation

- If $\{e_k : E \to \mathbb{R}\}_{k=1}^m$ are linearly independent, then α_t has a closed formula:

$$\boxed{\alpha_t = A_t^{-1} b_t \text{ where } b_t = \mathbb{E}[Z_{\tau_{t+1}} \vec{e}(X_t)]}$$

and $A_t \in \mathbb{R}^{m \times m}$ has coefficients

$$\boxed{(A_t)_{k,l} = \mathbb{E}[e_k(X_t) e_l(X_t)].}$$

The LSM algorithm: 2nd approximation

- Evaluate f_t using Monte Carlo sampling.
- Sample N independent paths $(X_t^{(1)})_{t=0}^T, \ldots, (X_t^{(N)})_{t=0}^T$.
- Denote by $Z_t^{(n)} = z_t(X_t^{(n)})$ the associated payoffs.
- For each path solve the dynamic programming: obtain N sampled stopping times $\tau_t^{(n)}$.
- $\mathbb{E}[Z_{\tau_{t+1}}|X_t] \approx \alpha_t \cdot \vec{e}(X_t)$ where

$$\alpha_t = \arg\min_{a \in \mathbb{R}^m} \frac{1}{N} \sum_{n=1}^{N} \left(Z^{(n)}_{\tau^{(n)}_{t+1}} - a \cdot \vec{e}(X_t^{(n)})\right)^2.$$

The LSM algorithm: 2nd approximation

- For linearly independent functions,

$$\boxed{\alpha_t = A_t^{-1} b_t \text{ where } b_t = \frac{1}{N} \sum_{n=1}^{N} Z^{(n)}_{\tau^{(n)}_{t+1}} \vec{e}(X_t^{(n)})}$$

  and $A_t \in \mathbb{R}^{m \times m}$ has coefficients

$$\boxed{(A_t)_{k,l} = \frac{1}{N} \sum_{n=1}^{N} e_k(X_t^{(n)}) e_l(X_t^{(n)}).}$$

- At the end of the dynamic program, obtain $\{\tau_1^{(n)}\}_{n=1}^N$ and output (remember $\sup_\tau \mathbb{E}[Z_\tau] = \max\{Z_0, \mathbb{E}[Z_{\tau_1}]\}$)

$$\mathcal{U}_0 = \max\left\{Z_0, \frac{1}{N} \sum_{n=1}^{N} Z^{(n)}_{\tau_1^{(n)}}\right\}.$$

The LSM algorithm

1: Sample N independent paths $(X_t^{(1)}, \ldots, X_t^{(N)})_{t=0}^T$.
2: Compute the associated payoffs $(Z_t^{(1)}, \ldots, Z_t^{(N)})_{t=0}^T$ and values $(e_k(X_t^{(1)}), \ldots, e_k(X_t^{(N)}))_{t\in[T],k\in[m]}$.
3: Estimate the matrices $\{A_t\}_{t=1}^{T-1}$ and inverses $\{A_t^{-1}\}_{t=1}^{T-1}$.
4: Set $\tau_T^{(n)} = T$ for $n \in [N]$.
5: **for** $t = T-1$ to 1 **do**
6: Calculate the vector $\alpha_t = A_t^{-1} \frac{1}{N} \sum_{n=1}^N Z_{\tau_{t+1}^{(n)}}^{(n)} \vec{e}(X_t^{(n)})$.
7: Calculate, for $n \in [N]$, $\tau_t^{(n)} = t1\{Z_t^{(n)} \geq \alpha_t \cdot \vec{e}(X_t^{(n)})\} + \tau_{t+1}^{(n)} 1\{Z_t^{(n)} < \alpha_t \cdot \vec{e}(X_t^{(n)})\}$.
8: **end for**
9: Output $\mathcal{U}_0 := \max\left\{Z_0, \frac{1}{N}\sum_{n=1}^N Z_{\tau_1^{(n)}}^{(n)}\right\}$.

Our quantum LSM algorithm

Our quantum LSM algorithm

- Main tool: use quantum Monte Carlo[3] to approximate $b_t = \mathbb{E}[Z_{\tau_{t+1}} \vec{e}(X_t)]$ and $A_t = \mathbb{E}[\vec{e}(X_t)\vec{e}(X_t)^\top]$.

Montanaro (2015)

Given $\varepsilon \in (0,1)$ and random variable X with $\mu = \mathbb{E}[X]$ and $\mathrm{Var}(X) \leq \sigma^2$, there is a quantum algorithm that runs in time $\widetilde{O}(\sigma/\varepsilon)$ and outputs $\widetilde{\mu}$ such that $|\widetilde{\mu} - \mu| \leq \varepsilon$ with high probability.

- Recent results of Cornelissen, Hamoudi and Jerbi[4].

[3] A. Montanaro. "Quantum speedup of Monte Carlo methods". (2015)

[4] A. Cornelissen, Y. Hamoudi, and S. Jerbi. "Near-Optimal Quantum Algorithms for Multi-variate Mean Estimation". (2021)

Our quantum LSM algorithm: input model

- Quantum sampling access to the Markov chain $(X_t)_{t=0}^T$:

$$U_{\mathbb{P}}|\vec{0}\rangle = \sum_{x \in E^T} \sqrt{p(x)}|x\rangle$$

where $p(x) = \mathbb{P}[X_1 = x_1] \prod_{t=1}^{T-1} \mathbb{P}[X_{t+1} = x_{t+1} | X_t = x_t]$.

- Quantum access to functions $h : E \to \mathbb{R}$:

$$V_h|x\rangle|\vec{0}\rangle = |x\rangle|h(x)\rangle.$$

Our quantum LSM algorithm: high-level analysis

- Use quantum Monte Carlo on $b_t = \mathbb{E}[Z_{\tau_{t+1}} \vec{e}(X_t)]$ and $A_t = \mathbb{E}[\vec{e}(X_t)\vec{e}(X_t)^\top]$.
- Compute $\alpha_t = A_t^{-1} b_t$ classically (m is small) and use to compute next stopping time τ_t.
- Difficulty: τ_t is computed in superposition and we don't have direct access to τ_u for $t+1 \leq u \leq T$.
- Main idea: recompute all τ_u for $t+1 \leq u \leq T$ recursively in superposition in order to obtain τ_t.

Our quantum LSM algorithm: computing τ_t

- Want $C_t^{(k)}$, $t \in [T]$ and $k \in [m]$, such that

$$C_t^{(k)} |x\rangle|\vec{0}\rangle = |x\rangle|z_{\tau_t(x)} e_k(x_{t-1})\rangle.$$

- Construct, with knowledge of α_t, W_t such that[5]

$$\begin{cases} W_t|x\rangle|\tau_{t+1}(x)\rangle|\vec{0}\rangle = |x\rangle|\tau_{t+1}(x)\rangle|\tau_t(x)\rangle & \text{if } t \neq T, \\ W_t|x\rangle|\vec{0}\rangle = |x\rangle|T\rangle & \text{if } t = T. \end{cases}$$

- Construct $V_t^{(k)}$ such that $V_t^{(k)}|x\rangle|\tau_t(x)\rangle|\vec{0}\rangle = |x\rangle|\tau_t(x)\rangle|z_{\tau_t(x)} e_k(x_{t-1})\rangle$.
- $$C_t^{(k)} := W_T^\dagger \dots W_{t+1}^\dagger W_t^\dagger V_t^{(k)} W_t W_{t+1} \dots W_T.$$

[5] $\tau_t(x) = t\mathbf{1}\{z_t(x_t) \geq \alpha_t \cdot \vec{e}(x_t)\} + \tau_{t+1}(x)\mathbf{1}\{z_t(x_t) < \alpha_t \cdot \vec{e}(x_t)\}$

Our quantum LSM algorithm

Approximate the matrices $\{A_t\}_{t=1}^{T-1}$ by QMC.
Compute the inverses $\{A_t^{-1}\}_{t=1}^{T-1}$ classically.
Construct $W_T : |x\rangle|0\rangle \mapsto |x\rangle|T\rangle$.
for $t = T$ to 2 **do**
  if $t \neq T$ **then**
    Construct $W_t : |x\rangle|\tau_{t+1}(x)\rangle|0\rangle \mapsto |x\rangle|\tau_{t+1}(x)\rangle|\tau_t(x)\rangle$.
  end if
  Construct $V_t^{(k)} : |x\rangle|\tau_t(x)\rangle|0\rangle \mapsto |x\rangle|\tau_t(x)\rangle|z_{\tau_t(x)} e_k(x_{t-1})\rangle$, $k \in [m]$.
  Approximate b_{t-1} using QMC with $W_T^\dagger \dots W_t^\dagger V_t^{(k)} W_t \dots W_T$, $k \in [m]$ (remember $b_{t-1} = \mathbb{E}[Z_{\tau_t} \vec{e}(X_{t-1})]$).
  Compute the vector $\alpha_{t-1} = A_{t-1}^{-1} b_{t-1}$ classically.
end for
Construct the unitary $W_1 : |x\rangle|\tau_2(x)\rangle|0\rangle \mapsto |x\rangle|\tau_2(x)\rangle|\tau_1(x)\rangle$.
Construct the unitary $V_1 : |x\rangle|\tau_1(x)\rangle|0\rangle \mapsto |x\rangle|\tau_1(x)\rangle|z_{\tau_1(x)}\rangle$.
Approximate $\mathbb{E}[Z_{\tau_1}]$ using QMC with $W_T^\dagger \dots W_1^\dagger V_1 W_1 \dots W_T$.
Output $\mathcal{U}_0 := \max\{Z_0, \mathbb{E}[Z_{\tau_1}]\}$.

Error analysis and complexity

Classical vs quantum LSM algorithm

Let $\varepsilon \in (0,1)$. Let $\sigma_{\min} \leq \min_{t\in[T-1]} \sigma_{\min}(A_t)$ and $N = \Theta\left(\frac{m^4}{\varepsilon^2 \sigma_{\min}^4}\right)$. LSM outputs $\mathcal{U}_0$ with high probability such that

$$|\mathcal{U}_0 - \mathbb{E}[Z_{\tau_0}]| \leq 5^T \left(\varepsilon + \max_{0\leq t<T} \min_{a\in\mathbb{R}^m} \|a \cdot \vec{e}(X_t) - \mathbb{E}[Z_{\tau_{t+1}}|X_t]\|_{L^2(\rho_t)}\right)$$

using classical $\widetilde{O}\left(\frac{Tm^6}{\varepsilon^2 \sigma_{\min}^4}\right)$ or quantum $\widetilde{O}\left(\frac{T^2 m^4}{\varepsilon \sigma_{\min}^2}\right)$ time.

Error analysis and complexity

$$|\mathcal{U}_0 - \mathbb{E}[Z_{\tau_0}]| \leq 5^T \left(\varepsilon + \max_{0 \leq t < T} \min_{a \in \mathbb{R}^m} \|a \cdot \vec{e}(X_t) - \mathbb{E}[Z_{\tau_{t+1}} | X_t]\|_{L^2(\rho_t)} \right)$$

- ε comes from Monte Carlo approximation.
- $\max_{0 \leq t < T} \min_{a \in \mathbb{R}^m} \|a \cdot \vec{e}(X_t) - \mathbb{E}[Z_{\tau_{t+1}} | X_t]\|_{L^2(\rho_t)}$: known as *approximation error*.
- Classical $\widetilde{O}\left(\frac{Tm^6}{\varepsilon^2 \sigma_{\min}^4}\right)$ vs quantum $\widetilde{O}\left(\frac{T^2 m^4}{\varepsilon \sigma_{\min}^2}\right)$.

Special cases

- Bound the approximation error and $\sigma_{\min}$?
- $\mathscr{H}_t$ = polynomials of degree at most q.
- Underlying Markov process:
  - Brownian motion;
  - Geometric Brownian motion.

NISQ/Fault-Tolerant Quantum Computers

- Quantum LSM algorithm suitable for NISQ devices?
  - Probably not.
- Require fault-tolerant quantum computers to observe some advantage.

Summary and open problems

- Proposed a quantum version of the LSM algorithm.
- Obtained a near quadratic improvement on the complexity for a few scenarios.

- Can we improve the T^2 time dependence?
- Perform a full quantum algorithm without intermediary classical steps?
- Other algorithms for the optimal stopping problem?

Multicriteria Shortest Path Algorithms

Ralf BORNDÖRFER
Zuse Institute Berlin & Freie Universität Berlin, Berlin, Germany
borndoerfer@zib.de

The optimization of paths subject to different criteria such as length, duration, cost, etc. comes up in all kinds of route planning applications; they lead to the Multiobjective Shortest Path Problem (MOSP) of computing the Pareto front of efficient solutions. We propose a new "Multiobjective Dijkstra" label-setting algorithm [1,2] that computes a minimum complete set of Pareto optimal paths; it is based on a lexicographic organization of the label exploration process. In this way, the main data structure, a priority queue, can be kept small, holding at most one label per node of the underlying graph, and all extracted labels are guaranteed to be efficient. The resulting algorithm improves the best know complexity bounds in this area. It gives rise to an FPTAS approximation variant [3], it can be generalized to a time dependent setting (in the FIFO case), it is parallelizable, and it works in practical implementations for more than two objectives.

References

[1] Pedro Maristany de las Casas, Ralf Borndörfer, Luitgard Kraus, Antonio Sedeñ-Noda, "An Improved Multiobjective Shortest Path Algorithm", Computers & Operations Research, Volume 135,105424,November 2021, https://doi.org/10.1016/j.cor.2021.105424 2021

[2] Pedro Maristany de las Casas, Luitgard Kraus, Ralf Borndörfer, Antonio Sedeñ-Noda, "Targeted Multiobjective Dijkstra Algorithm", ZIB Report, 2021, arXiv:2110.10978, https://arxiv.org/abs/2110.10978

[3] Pedro Maristany de las Casas, Ralf Borndörfer, Luitgard Kraus, Antonio Sedeñ-Noda, "An FPTAS for Dynamic Multiobjective Shortest Path Problems", Algorithms, Volume 14, Number 2, pages 1-22, 2021, https://doi.org/https://doi.org/10.3390/a14020043

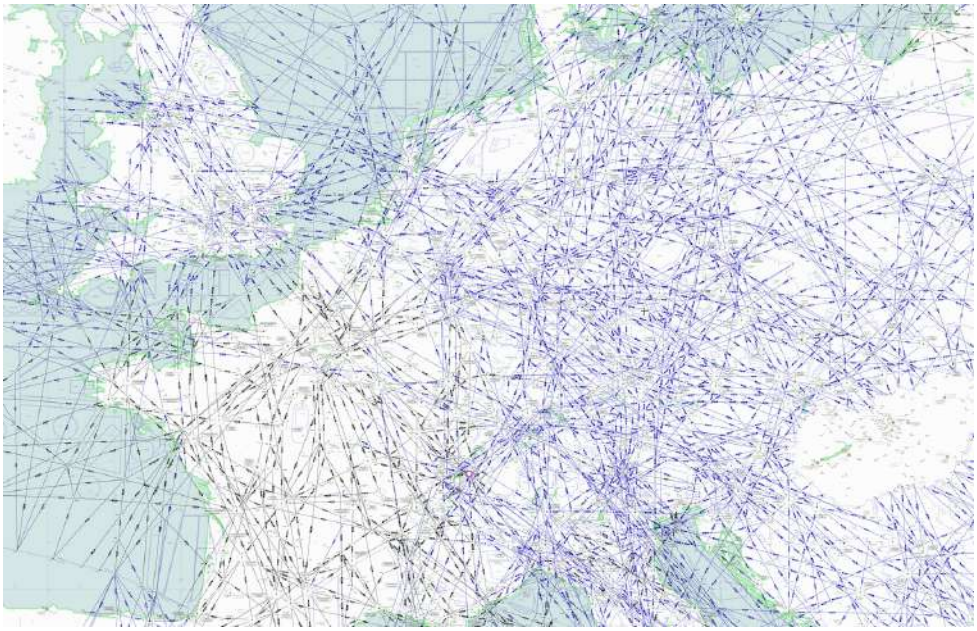
Multiobjective Shortest Path Algorithms

Ralf Borndörfer Pedro Maristany Luitgard Kraus Antonio Sedeño-Noda
Freie Universität Berlin – Zuse Institute Berlin – MODAL MobilityLab

6th RIKEN-IMI-ISM-NUS-ZIB-MODAL-NHR Workshop on Advances in Classical and Quantum Algorithms for Optimization and Machine Learning

21.09.2022

The Airway Network (2D)



The Time-dependent 2D Flight Planning Problem

Definition (Time-dependent 2D Flight Planning Problem).

Input:

- Directed graph $D = (V, A)$ (embedded on the Earth)
- Source and target nodes $s, t \in V$
- Departure time τ_0
- Travel time functions (TTFs) $t_a \colon \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ for each arc $a \in A$, mapping starting time τ to traversal time $t_a(\tau)$

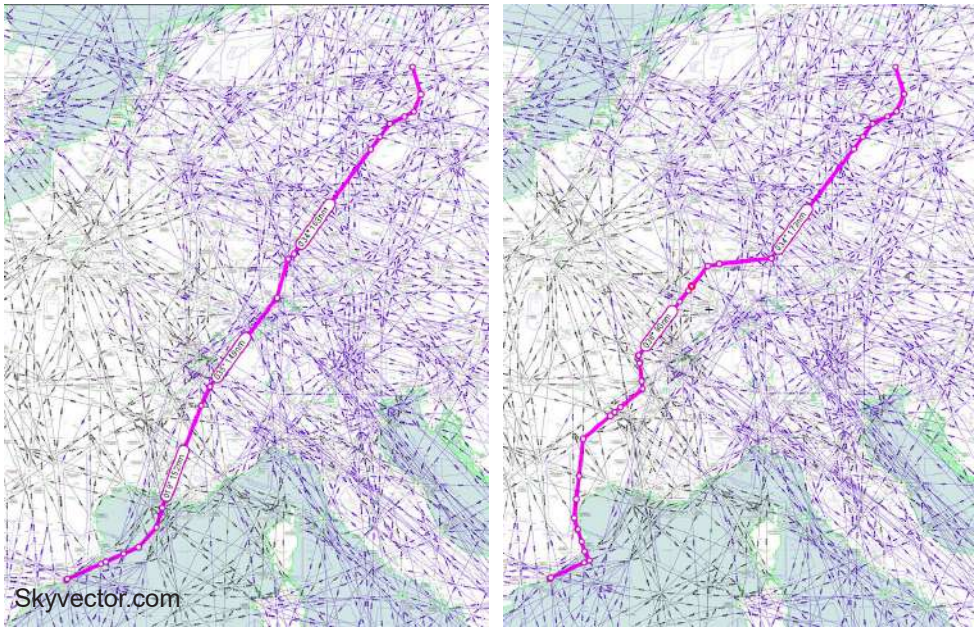
Output:

- st-path path $p = \{v_0, \dots, v_n\}$ minimizing

$$t(p) := \sum_{i=0}^{n-1} t_{(v_i, v_i+1)}(\tau_i) \quad \text{s.t.} \quad \tau_i = \tau_{i-1} + t_{(v_{i-1}, v_i)}(\tau_{i-1})$$

FRA-SFO: Min. Fuel vs Min. Distance Track

FRA-MRS: MFT vs. Min. Overflight Cost Track



The Static 2D Flight Planning Problem

Definition (Static 2D Flight Planning Problem).

Input:

- Directed graph $D = (V, A)$ (embedded on the Earth)
- Source and target nodes $s, t \in V$
- Departure time τ_0
- Cost $c_a \in \mathbb{N}_0$ for each arc $a \in A$

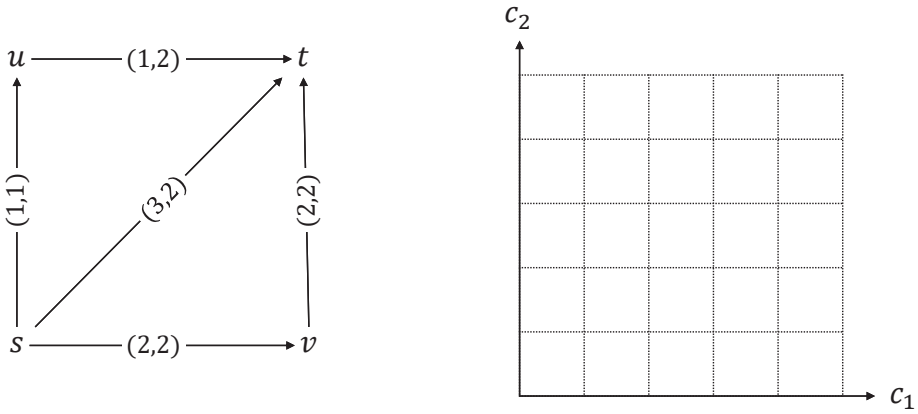
Geben Sie hier eine Formel ein.

Output:

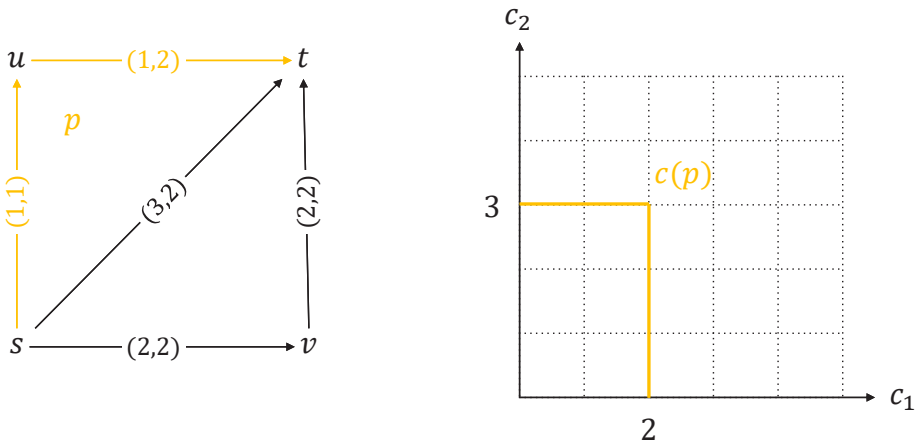
- st-path path $p = \{v_0, \dots, v_n\}$ minimizing

$$c(p) := \sum_{i=0}^{n-1} c_{(v_i, v_i+1)}$$

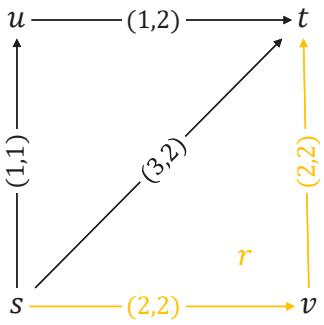
The Multiobjective Shortest Path Problem

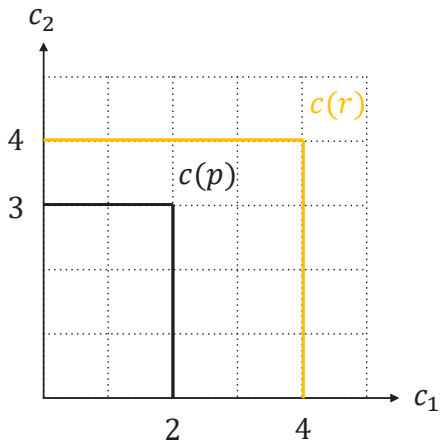


The Multiobjective Shortest Path Problem

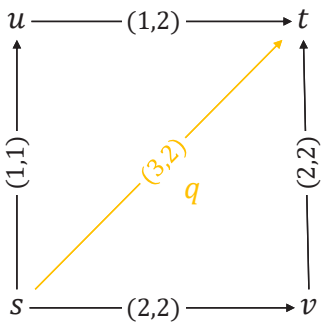


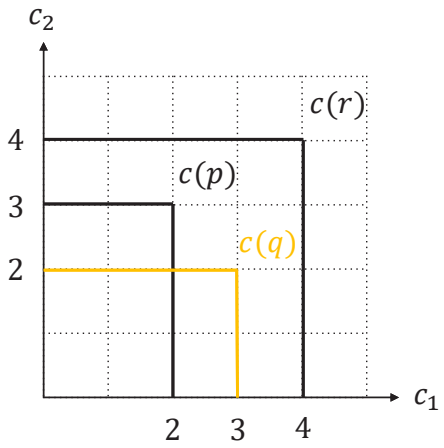
The Multiobjective Shortest Path Problem



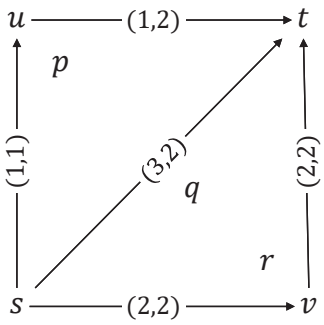


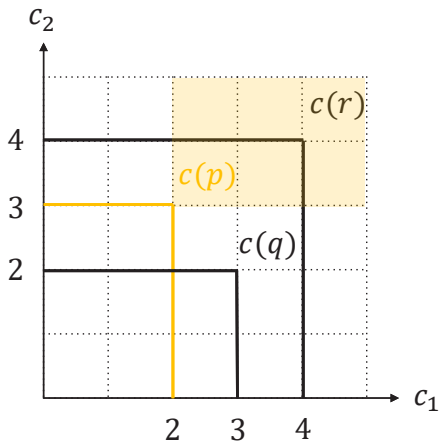
The Multiobjective Shortest Path Problem





The Multiobjective Shortest Path Problem

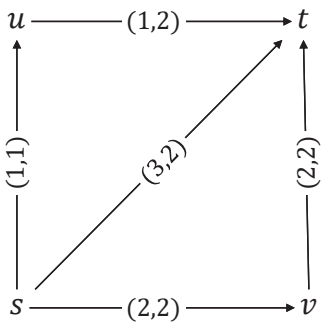


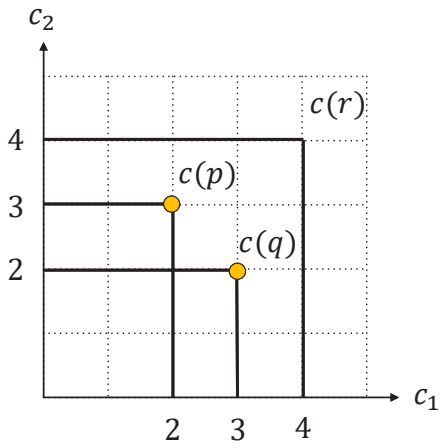


Definition (Dominance). Let p, r be sv-paths in $D = (V, A)$, $c \in \mathbb{R}_{\geq 0}^{A \times d}$. Then p **dominates** r $:\Leftrightarrow p \preccurlyeq r :\Leftrightarrow c(p) \lneqq c(r)$.

Definition (Efficiency). An undominated sv-path is **efficient**.

The Multiobjective Shortest Path Problem





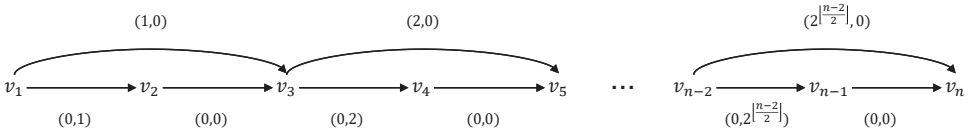
Definition (Multiobjective Shortest Path Problem (MOSP)).

Input: Digraph $D = (V, A, c), c \in \mathbb{N}_0^{A \times d}, s \in V$.

Output: "Minimally complete" set of efficient sv-paths for all $v \in V$.

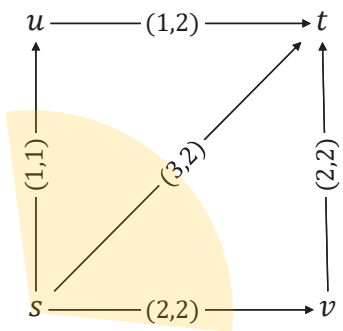
Note: Only one path for each multidimensional objective.

The Intractability of MOSP



Example & Observation (Intractability of MOSP, Hansen [1979]). Even for only two objectives, a MOSP can have an exponential number of efficient paths.

Multiobjective Shortest Path Algorithms



Definition (Explored & Permanent Paths). Let p be an sv-path in a digraph $D(V, A, c), c \in \mathbb{N}_0^d$, $s, v \in V$.

a) p **explored** $:\Leftrightarrow$ p has been seen but is not known to be efficient

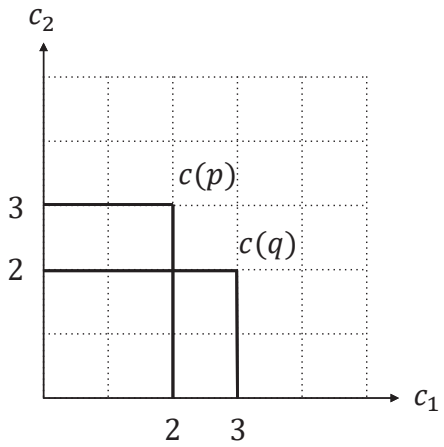
b) p **permanent** $:\Leftrightarrow$ p is guaranteed to be (and remain) efficient

Explored sv-paths are stored in a priority queue Q, and permanent sv-paths in a set P_v.

Multiobjective Shortest Path Algorithms

$p \nleq q$ and $q \nleq p$

$p \prec_{lex} q$



Definition (Lexicographic Order). Let p, q be sv-paths in $D = (V, A)$, $c \in \mathbb{N}_0^{A \times d}$. Then

$$p \prec_{lex} q \;:\Leftrightarrow\; c_i(p) < c_i(q) \text{ for the first } i \in [d] \text{ s.t. } c_i(p) \neq c_i(q).$$

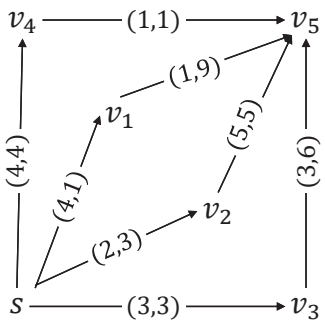
Martins's Algorithm (Martins [1984])

Input: MOSP (D, c, s)

Output: $P_v, v \in V$

1. forall $v \in V$ do $P_v \leftarrow \emptyset$ endforall
2. $p_{\text{init}} \leftarrow (s), Q \leftarrow \{p_{\text{init}}\}$
3. while $Q \neq \emptyset$ do
4. $\quad p \leftarrow Q.\text{extract\_min}()$, $v \leftarrow \text{head}(p)$
5. $\quad P_v \leftarrow P_v \cup \{p\}$
6. $\quad$ forall $w \in \delta^+(v)$ do
7. $\quad\quad$ if $P_w \nleq (p, w)$ then
8. $\quad\quad\quad Q \leftarrow Q \cup \{(p, w)\}$
9. $\quad\quad\quad Q \leftarrow \text{clean\_heap}$
10. $\quad\quad$ endif
11. endwhile
12. return $P_v, v \in V$

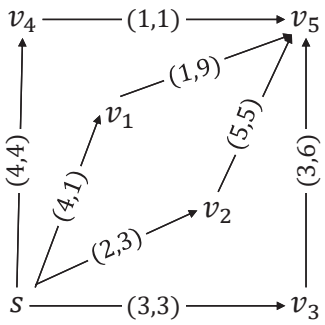
Martins's Algorithm



priority queue Q

s	v_1	v_2	v_3	v_4	v_5
(0,0)					

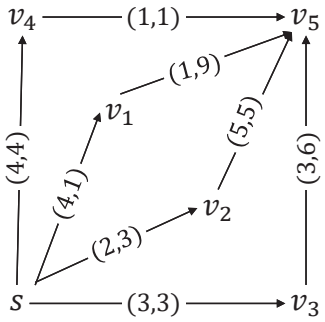
Martins's Algorithm



priority queue Q

s	v_1	v_2	v_3	v_4	v_5
	(4,1)	(2,3)	(3,3)	(4,4)	

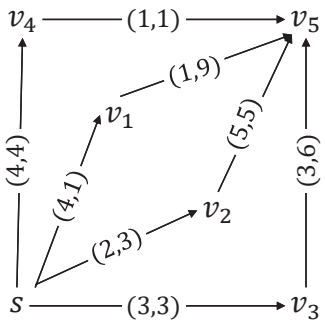
Martins's Algorithm



priority queue Q

s	v_1	v_2	v_3	v_4	v_5
	(4,1)		(3,3)	(4,4)	(7,8)

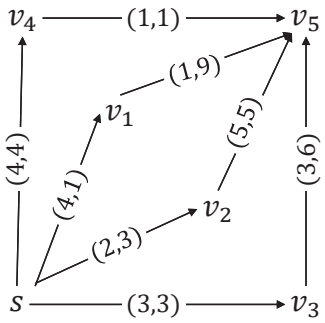
Martins's Algorithm



priority queue Q

s	v_1	v_2	v_3	v_4	v_5
	(4,1)			(4,4)	(7,8) (6,9)

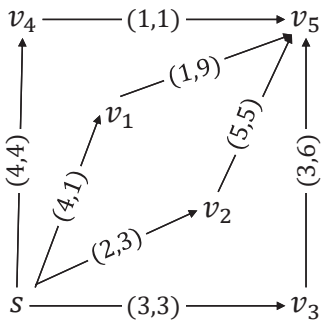
Martins's Algorithm



priority queue Q

s	v_1	v_2	v_3	v_4	v_5
				(4,4)	(7,8)
					(6,9)
					(5,10)

Martins's Algorithm



priority queue Q

s	v_1	v_2	v_3	v_4	v_5
					(5,5)

Martins's Algorithm

Input: MOSP (D, c, s)

Output: $P_v, v \in V$

forall $v \in V$ do $P_v \leftarrow \emptyset$ endforall
$p_{\text{init}} \leftarrow (s), Q \leftarrow \{p_{\text{init}}\}$
while $Q \neq \emptyset$ do
  $p_v^* \leftarrow Q.\text{extract\_min}()$, $v \leftarrow \text{head}(p_v^*)$
  $P_v \leftarrow P_v \cup \{p_v^*\}$
  forall $w \in \delta^+(v)$ do
    if $P_w \nleq (p_v^*, w)$ then
      $Q \leftarrow Q \cup \{(p_v^*, w)\}$
      $Q \leftarrow \text{clean\_heap}$
    endif
endwhile
return $P_v, v \in V$

a) Q can contain exponentially many sv-paths for any $v \in V$.
b) clean_heap must access them all and remove the dominated ones.
c) Heap properties have to be restored after deletions.

Multiobjective Dijkstra Algorithm (BKMS [2021])

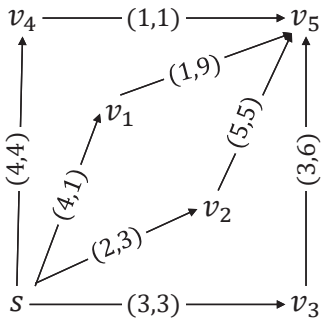
Input: MOSP (D, c, s)

Output: $P_v, v \in V$

forall $v \in V$ do $P_v \leftarrow \emptyset$ endforall
$p_{\text{init}} \leftarrow (s), Q \leftarrow \{p_{\text{init}}\}$
while $Q \neq \emptyset$ do
$p_v^* \leftarrow Q.\text{extract\_min}()$, $v \leftarrow \text{head}(p_v^*)$
$P_v \leftarrow P_v \cup \{p_v^*\}$
$p_v^{\text{new}} \leftarrow \text{arglexmin}_{p_u \in P_u : u \in \delta^-(v)} \{(p_u, v) : P_v \nleq (p_u, v)\}$ // next cand. label
if $p_v^{\text{new}} \neq \text{nil}$ then $Q \leftarrow Q \cup \{p_v^{\text{new}}\}$ endif
forall $w \in \delta^+(v)$ do
  if $P_w \nleq (p_v^*, w)$ and $(p_v^*, w) \prec_{\text{lex}} Q.w$ then $Q.\text{decrease\_key}(Q, w, (p_v^*, w))$ endif
endforall
endwhile
return $P_v, v \in V$

a) Paths extracted from Q are efficient.
b) Q contains at most one path per node: the lex-smallest undominated path at P_v.

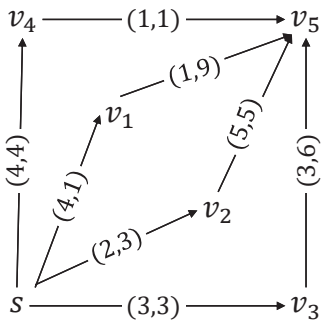
Multiobjective Dijkstra Algorithm



priority queue Q

s	v_1	v_2	v_3	v_4	v_5
(0,0)					

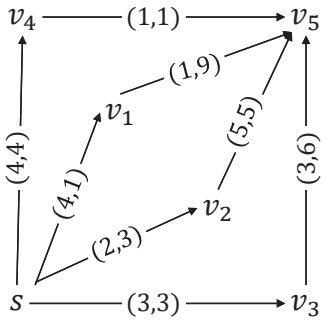
Multiobjective Dijkstra Algorithm



priority queue Q

s	v_1	v_2	v_3	v_4	v_5
	(4,1)	(2,3)	(3,3)	(4,4)	

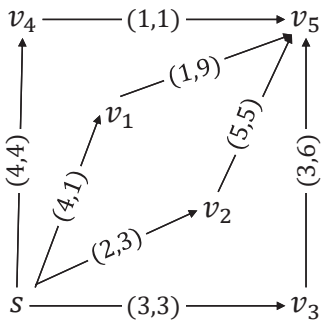
Multiobjective Dijkstra Algorithm



priority queue Q

s	v_1	v_2	v_3	v_4	v_5
	(4,1)		(3,3)	(4,4)	(7,8)

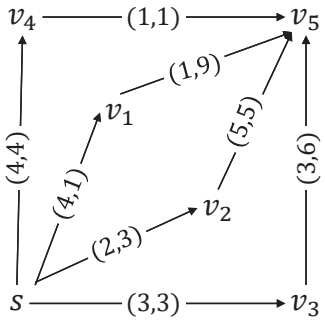
Multiobjective Dijkstra Algorithm



priority queue Q

s	v_1	v_2	v_3	v_4	v_5
	(4,1)			(4,4)	(6,9)

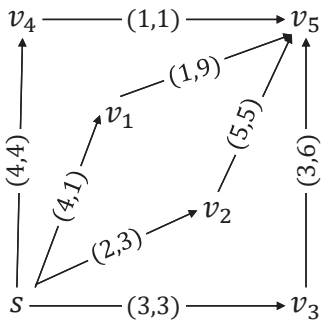
Multiobjective Dijkstra Algorithm



priority queue Q

s	v_1	v_2	v_3	v_4	v_5
				(4,4)	(5,10)

Multiobjective Dijkstra Algorithm



priority queue Q

s	v_1	v_2	v_3	v_4	v_5
					(5,5)

Multiobjective Dijkstra Algorithm

Theorem (Complexity of the MDA). Let (D, c, s) be a MOSP and let

- n number of nodes
- m number of arcs
- N total number of efficient paths
- $N_{\max}$ maximal number of efficient paths at a single node.

Then the complexities of Martins's Algorithm and the MDA are

Algorithm	Martins's Algorithm	MDA
Run time	$O(dN^2 n)$	$O(dN \log n + dN_{max}^2\, m)$.

Breugem et. al. [2017]

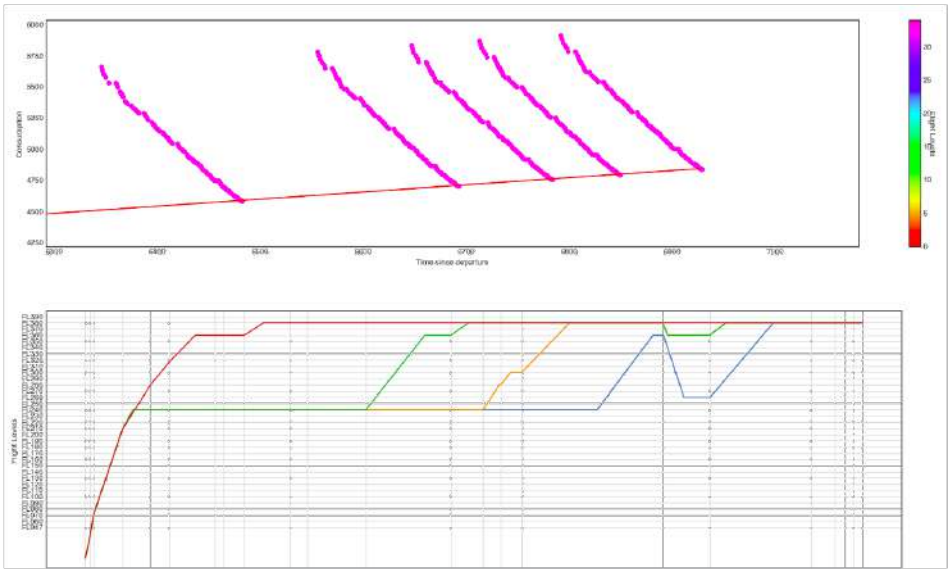
Proof (Sketch):

a) $N \log n$ = # iterations × complexity of $\mathrm{extract\_min}$.

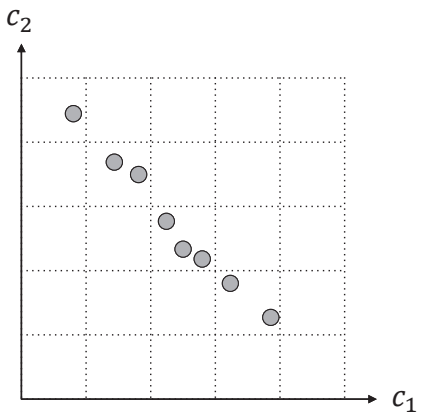
b) $N_{max}^2\, m$ = complexity of building paths and checking dominance

Note: $P_v \preccurlyeq p$ takes $O(d|P_v|) \leq O(d\, N_{\max})$.

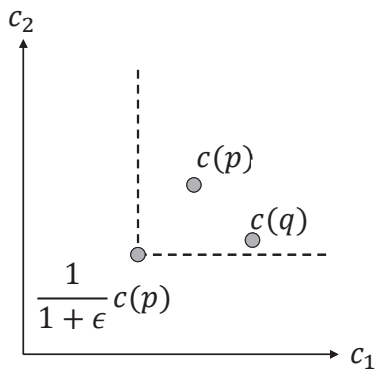
Many efficient flight paths exist (time vs fuel).



Approximating the Set of Efficient Solutions



Approximating the Set of Efficient Solutions



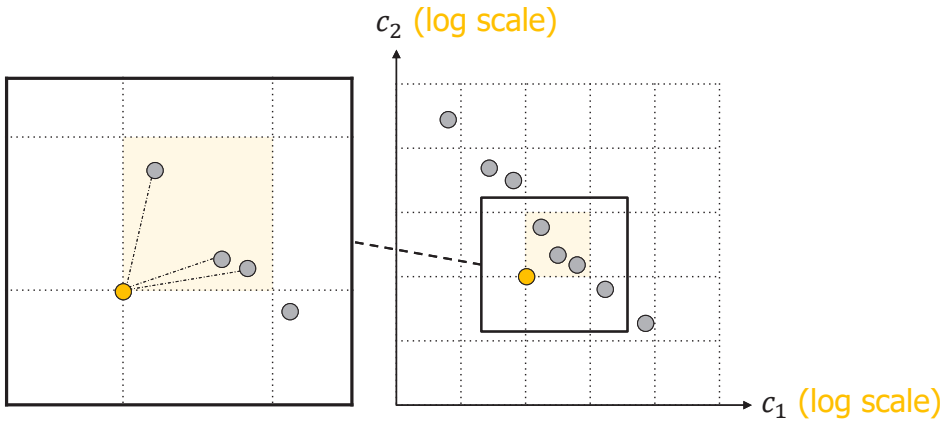
Definition (ϵ-Dominance). Let p, q be sv-paths, $\epsilon > 0$. Then

$$p \ \boldsymbol{\epsilon}\textbf{-dominates}\ q :\Leftrightarrow p \preccurlyeq_\epsilon q :\Leftrightarrow c(p) \leq (1+\epsilon)c(q).$$

Definition (ϵ-Cover). P_ϵ is an **ϵ-cover** for a MOSP (D, c, s) if for every sv-path p there is an sv-path $p' \in P_\epsilon$ s.t. $p' \preccurlyeq_\epsilon p$.

Definition (FPTAS). An **FTPAS** computes for any $\epsilon > 0$ and any MOSP (D, c, s) an ϵ-cover in time polynomial in $\langle D, c, s\rangle$ and $1/\epsilon$.

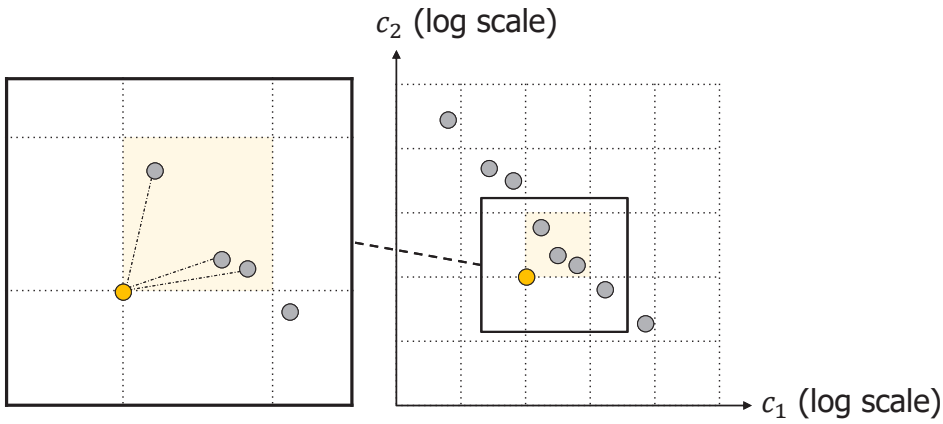
Approximating the Set of Efficient Solutions



Definition (pos-function, Tsaggouris & Zaroliagis [2006]). The lower left corner of the cell of path p is assigned grid coordinates

$$\mathrm{pos}_i(p) := \begin{cases} 0, & c_i(p) = 0, \\ 1 + \left\lfloor \frac{\log c_i(p)}{\log r} \right\rfloor, & c_i(p) \neq 0, \end{cases} \quad i \in [d], r = (1+\epsilon)^{\frac{1}{n-1}}.$$

Approximating the Set of Efficient Solutions



Observation (Cells and Coverage). Let p, q be sv-paths. Then

$$\mathrm{pos}(p) \leq \mathrm{pos}(q) \Longrightarrow p \preccurlyeq_\epsilon q.$$

Approx. Multiobjective Dijkstra Alg. (BKMS [2021])

Input: MOSP (D, c, s)

Output: $P_v, v \in V$

1. forall $v \in V$ do $P_v \leftarrow \emptyset$ endforall
2. $p_{\text{init}} \leftarrow (s), Q \leftarrow \{p_{\text{init}}\}$
3. while $Q \neq \emptyset$ do
4. $p_v^* \leftarrow Q.\text{extract\_min}()$, $v \leftarrow \text{head}(p_v^*)$
5. $P_v \leftarrow P_v \cup \{p_v^*\}$
6. $p_v^{\text{new}} \leftarrow \text{arglexmin}_{p_u \in P_u : u \in \delta^-(v)} \{(p_u, v) : P_v \not\preceq_\epsilon (p_u, v)\}$
7. if $p_v^{\text{new}} \neq \text{nil}$ then $Q \leftarrow Q \cup \{p_v^{\text{new}}\}$ endif
8. forall $w \in \delta^+(v)$ do
9. if $P_w \not\preceq_\epsilon (p_v^*, w)$ and $(p_v^*, w) \prec_{\text{lex}} Q.w$ then $Q.\text{decrease\_key}(Q, w, (p_v^*, w))$ endif
10. endforall
11. endwhile
12. return $P_v, v \in V$

Approximative Multiobjective Dijkstra Algorithm

Lemma (Correctness of the Approximate MDA). Let p be an efficient sv-path with k arcs. Then the output of the Approximate MDA contains an sv-path p' s.t.

$$c(p') \leq r^k c(p).$$

Corollary (Correctness of the Approximate MDA). Under the conditions of the above Lemma,

$$c(p') \leq (1 + \epsilon) c(p).$$

Proof. $k \leq n - 1$ and $r = (1 + \epsilon)^{\frac{1}{n-1}}$.

Corollary (ϵ-Cover). The Approximate MDA computes an ϵ-cover.

Approximative Multiobjective Dijkstra Algorithm

Theorem (Complexity of the Approximate MDA). The run time of the Approximate MDA is

$$O(d\mathcal{C}n \log n + d\mathcal{C}^2 m),$$

where $C := \max_{i\in[d], a\in A} c_i(a)$.

Proof. The pos-function takes at most

$$\mathcal{C} := \left(\left\lfloor \frac{n}{\epsilon} \log(nC) \right\rfloor\right)^d$$

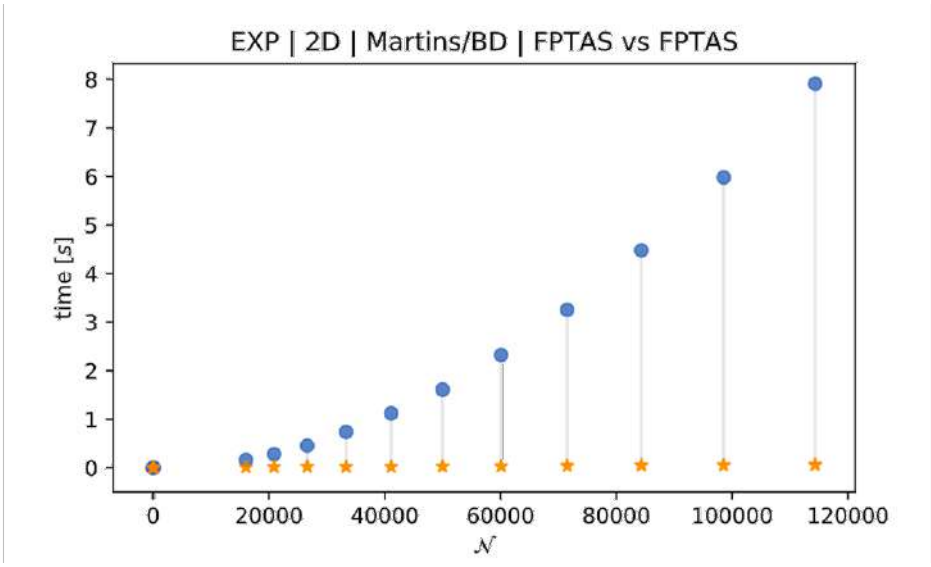
values. Hence, every node can have a path in at most $\mathcal{C}$ cells, and the number of output paths is at most $n\mathcal{C}$. As

$$O(dN \log n + dN_{max}^2\, m) \overset{N\leq nN_{\max}}{\leq} O(dN_{\max} n \log n + dN_{max}^2\, m),$$

the claim follows from $N_{\max} = \mathcal{C}$.

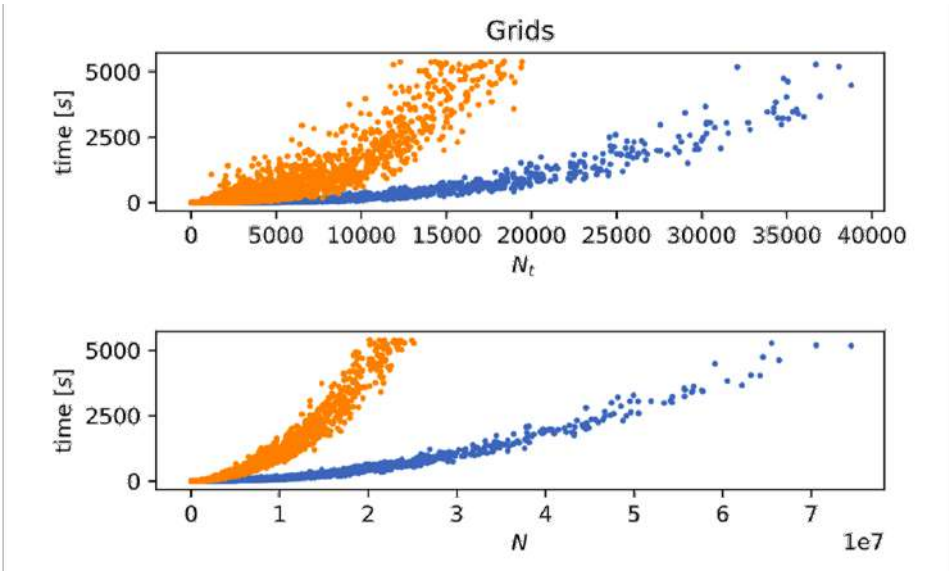
Corollary (FPATS). The Approximate MDA is an FPTAS for MOSP.

Computational Results on Exponential Graphs



Orange: MDA, Blue: Martins

Computational Results on Grid Graphs



Orange: MDA, Blue: Martins

Back to the 2D Flight Planning Problem

Definition (Time-dependent 2D Flight Planning Problem).
Input:

- Directed graph $D = (V, A)$ (embedded on the Earth)
- Source and target nodes $s, t \in V$
- Travel time functions (TTFs) $t_a: \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ for each arc $a \in A$, mapping starting time τ to traversal time $t_a(\tau)$
- Departure time τ_0

Output:

- st-path path $p = \{v_0, \dots, v_n\}$ minimizing

$$t(p) := \sum_{i=0}^{n-1} t_{(v_i, v_i+1)}(\tau_i) \quad \text{s.t.} \quad \tau_i = \tau_{i-1} + t_{(v_{i-1}, v_i)}(\tau_{i-1})$$

Back to the 2D Flight Planning Problem

Observation (Dynamic Arc Costs). Let $c: A \times \mathbb{R}^d_{\geq 0} \to \mathbb{R}^d_{\geq 0}$ be a **dynamic arc cost function**. Then the cost of an sv-path (p, a) is defined recursively as

$$c(s) := 0, \qquad c(p, a) := c(p) + c\big(a, c(p)\big).$$

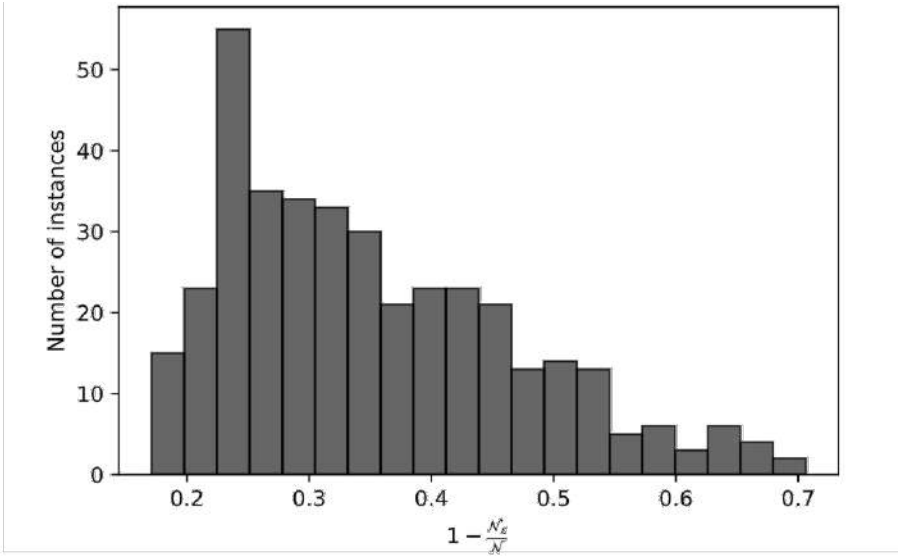
Proposition & Definition (Dynamic MDA & FIFO Property). If a dynamic arc cost function satisfies the **FIFO property**

$$x \leq y \Longrightarrow x + c(a, x) \leq y + c(a, y) \qquad \forall x, y, a,$$

MDA s.t. dynamic arc costs is correct.

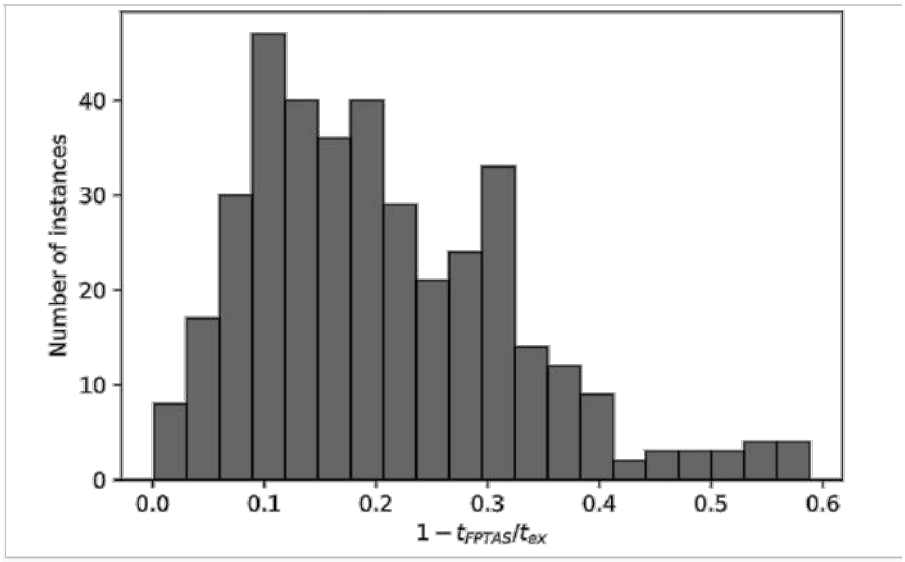
Proposition (Dynamic FPTAS). If a dynamic arc cost function is piecewise linear with positive intercepts, the Approximate MDA is correct.

Back to the 2D Flight Planning Problem



Labels saved by Dynamic FPTAS on 2D flight planning instances.

Back to the 2D Flight Planning Problem



Time saved by Dynamic FPTAS on 2D flight planning instances.

Summary and 注目してくれてありがとう

- Dynamic and Static 2D Flight Planning Problem
- Multiobjective Shortest Path Problem (MOSP)
- Martins's Algorithm
- Multiobjective Dijkstra Algorithm (MDA)
- Approximate MDA (FPTAS)
- Dynamic Arc Cost Versions
- A* Versions
- Papers @ https://www.zib.de/projects/flight-trajectory-optimization-airway-networks

Pedro Maristany de las Casas, Ralf Borndörfer, Luitgard Kraus, Antonio Sedeño-Noda	An FPTAS for Dynamic Multiobjective Shortest Path Problems	*Algorithms*, 14(2), pp. 1-22, 2021 (preprint available as ZIB-Report 20-31)	PDF (ZIB-Report) BibTeX DOI
Pedro Maristany de las Casas, Antonio Sedeno-Noda, Ralf Borndörfer	An Improved Multiobjective Shortest Path Algorithm	*Computers & Operations Research*, Vol.135, 2021 (preprint available as ZIB-Report 20-26)	PDF (ZIB-Report) BibTeX DOI
Pedro Maristany de las Casas, Luitgard Kraus, Ralf Borndörfer, Antonio Sedeno-Noda	Targeted Multiobjective Dijkstra Algorithm	2021 (under review)	BibTeX arXiv

Randomized subspace regularized Newton method for unconstrained non-convex optimization

Pierre-Louis POIRION

RIKEN-AIP, Tokyo, Japan
pierre-louis.poirion@riken.jp

In this talk we present a randomized subspace regularized Newton method for a non-convex function. We show that our method has global convergence under appropriate assumptions and its convergence rate is the same as that of the full regularized Newton method. Furthermore, we can obtain a local linear convergence rate, under some additional assumptions, and prove that this rate is the best we can hope when using random subspace.

Randomized Subspace Newton Method for Unconstrained Non-Convex Optimization

The 6th RIKEN-IMI-ISM-NUS-ZIB-MODAL-NHR Workshop

Pierre-Louis Poirion (RIKEN-AIP)
joint work with Terunari Fuji and Akiko Takeda

November 2, 2022

Overview

The gist

Non-convex unconstrained minimization

$$\min_{x \in \mathbb{R}^n} f(x),$$

where $f : \mathbb{R}^n \to \mathbb{R}$ is twice differentiable

The gist

Non-convex unconstrained minimization

$$\min_{x \in \mathbb{R}^n} f(x),$$

where $f : \mathbb{R}^n \to \mathbb{R}$ is twice differentiable

Subspace optimization

$$\min_{u \in \mathbb{R}^s} f(x + P^\top u),$$

where $P \in \mathbb{R}^{s \times n}$ is a random matrix.

The gist

Non-convex unconstrained minimization

$$\min_{x \in \mathbb{R}^n} f(x),$$

where $f : \mathbb{R}^n \to \mathbb{R}$ is twice differentiable

Subspace optimization

$$\min_{u \in \mathbb{R}^s} f(x + P^\top u),$$

where $P \in \mathbb{R}^{s \times n}$ is a random matrix.

- Can we speed up the computation time?
- Global and local convergence properties?

Previous works

Random Subspace Newton (RSN) [Gower et al., 2019](f is convex)

By computing the Newton direction on the function $u \mapsto f(x_k + P_k^\top u_k)$, we obtain $u_k = -(P_k \nabla^2 f(x_k) P_k^\top)^{-1} P_k \nabla f(x_k)$, hence

$$x_{k+1} = x_k - t_k P_k^\top (P_k \nabla^2 f(x_k) P_k^\top)^{-1} P_k \nabla f(x_k).$$

They prove global sub-linear convergence and local linear convergence if f is strongly convex.

Previous works

Random Subspace Newton (RSN) [Gower et al., 2019](f is convex)

By computing the Newton direction on the function $u \mapsto f(x_k + P_k^\top u_k)$, we obtain $u_k = -(P_k \nabla^2 f(x_k) P_k^\top)^{-1} P_k \nabla f(x_k)$, hence

$$x_{k+1} = x_k - t_k P_k^\top (P_k \nabla^2 f(x_k) P_k^\top)^{-1} P_k \nabla f(x_k).$$

They prove global sub-linear convergence and local linear convergence if f is strongly convex.

- [Hanzely et al., 2020]: Cubically-regularized subspace Newton method.
- [Kovalev et al., 2020]: random subspace version of the BFGS method.
- [Roberts and Royer, 2022]: probabilistic direct-search method in reduced random spaces (non-convex problems). The authors prove sub-linear convergence.

Our work

Based on regularized Newton method (RNM) for the unconstrained non-convex optimization [Ueda and Yamashita, 2010], we propose the randomized subspace regularized Newton method (RS-RNM):

$$\begin{aligned} d_k &= -P_k^\top (P_k \nabla^2 f(x_k) P_k^\top + \eta_k I_s)^{-1} P_k \nabla f(x_k), \\ x_{k+1} &= x_k + t_k d_k, \end{aligned}$$

where η_k is defined to ensure $P_k \nabla^2 f(x_k) P_k^\top + \eta_k I_s \succ 0$ and t_k satisfies Armijo's rule.

Our work

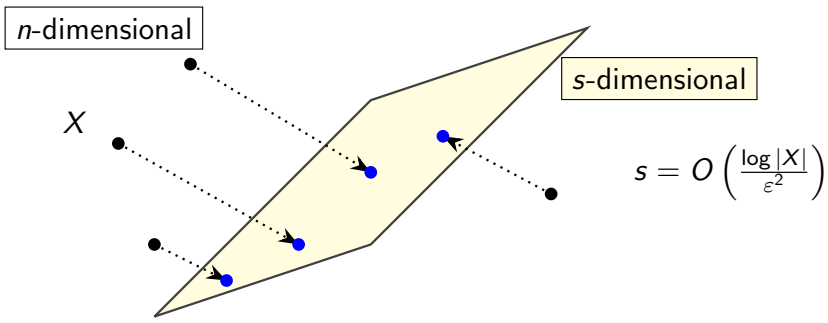
Based on regularized Newton method (RNM) for the unconstrained non-convex optimization [Ueda and Yamashita, 2010], we propose the randomized subspace regularized Newton method (RS-RNM):

$$d_k = -P_k^\top (P_k \nabla^2 f(x_k) P_k^\top + \eta_k I_s)^{-1} P_k \nabla f(x_k),$$
$$x_{k+1} = x_k + t_k d_k,$$

where η_k is defined to ensure $P_k \nabla^2 f(x_k) P_k^\top + \eta_k I_s \succ 0$ and t_k satisfies Armijo's rule.

- In [Ueda and Yamashita, 2010] the authors prove global sub-linear convergence and local quadratic convergence under local-error bound condition.
- Can we extend these results to the random subspace setting ?

What is Random Projection



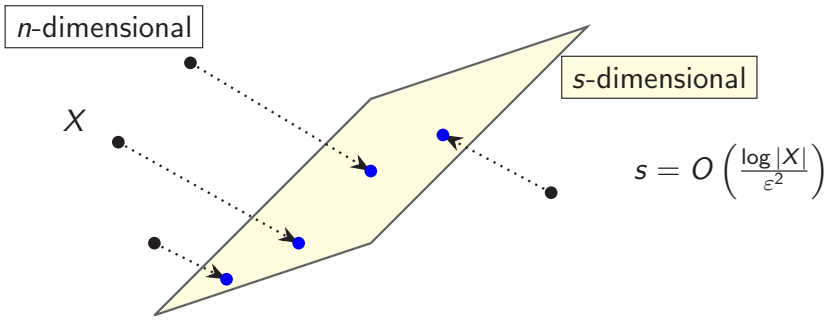
Random Projection

Lemma JLL

Let $P \in \mathbb{R}^{d\times n}$, $P_{ij} \sim \mathrm{N}(0, 1/s)$, i.i.d..
Then for any $x \in \mathbb{R}^n$ and $\varepsilon \in (0,1)$, we have

$$\mathrm{Prob}\ [(1-\varepsilon)\|x\|_2^2 \le \|Px\|_2^2 \le (1+\varepsilon)\|x\|_2^2] \ge 1 - 2\exp(-\mathcal{C}\varepsilon^2 s),$$

where $\mathcal{C}$ is an absolute constant.



Algorithm 1 Randomized subspace regularized Newton method (RS-RNM)

input: $x_0 \in \mathbb{R}^n$, $\gamma \ge 0, c_1 > 1, c_2 > 0, \alpha, \beta \in (0,1)$

1: $k \leftarrow 0$
2: **repeat**
3: sample a random matrix: $P_k \sim$ Gaussian matrix $\mathcal{N}(0, 1/s)^{s\times n}$
4: compute the regularized sketched hessian: $M_k = P_k \nabla^2 f(x_k) P_k^\top + c_1 \Lambda_k I_s + c_2 \|\nabla f(x_k)\|^\gamma I_s$, where $\Lambda_k = \max(0, -\lambda_{\min}(P_k \nabla^2 f(x_k) P_k^\top))$
5: compute the search direction: $d_k = -P_k^\top M_k^{-1} P_k \nabla f(x_k)$
6: apply the backtracking line search with Armijo's to compute $l_k \ge 0$ such that (1) holds. Set $t_k = \beta^{l_k}$, $x_{k+1} = x_k + t_k d_k$ and $k \leftarrow k+1$
7: **until** the stopping criteria is satisfied
8: **return** the last iterate x_k

$$f(x_k) - f(x_k + \beta^{l_k} d_k) \ge -\alpha \beta^{l_k} g_k^\top d_k. \tag{1}$$

Global convergence

Assumption (1)

The level set of f at the initial point x_0 is compact, i.e.,
$\Omega := \{\mathbb{R}^n : f(x) \leq f(x_0)\}$ *is compact.*

Global convergence

Assumption (1)

The level set of f at the initial point x_0 is compact, i.e.,
$\Omega := \{\mathbb{R}^n : f(x) \leq f(x_0)\}$ *is compact.*

Assumption (2)

1. $\gamma \leq 1/2$,
2. $\alpha \leq 1/2$,
3. *There exists $L_H > 0$ such that*

$$\|\nabla^2 f(x) - \nabla^2 f(y)\| \leq L_H \|x - y\|, \quad {}^\forall x, y \in \Omega + B(0, r_1),$$

where $r_1 := \dfrac{\mathcal{C} U_g^{1-\gamma} n}{c_2 s}$, *and* $\|\nabla f(x_k)\| \leq U_g$.

Global convergence

Let

$$t_{\min} = \min\left(1, \frac{\beta c_2^2 s^2}{\mathcal{C}^2 L_H U_g^{1-2\gamma} n^2}\right)$$

Theorem

Suppose that Assumptions (1) and (2) hold. Let

$$p = \frac{\alpha t_{\min}}{2\mathcal{C}(1+c_1)\frac{n}{s}U_H + 2c_2 U_g^{\gamma}}.$$

Then, with probability at least $1 - 2m\left(\exp(-\frac{\mathcal{C}_0}{4}s) - \exp(-s)\right)$, we have

$$\sqrt{\frac{f(x_0) - f^*}{mp}} \geq \min_{k=0,1,\ldots,m-1} \|\nabla f(x_k)\|.$$

This global $O(\varepsilon^{-2})$ complexity is the same as that obtained in

Local convergence

Assume that $\{x_k\}$ converge to a strict local minima $\bar{x}$. We show that

- the sequence $\{f(x_k)\}$ converges locally linearly to $f(\bar{x})$
- when f is strongly convex, we cannot aim at local super-linear convergence using random subspace.

Local convergence: assumptions

Assumption (2')

In a neighborhood of $\bar{x}$, we have

$$\|\nabla^2 f(x) - \nabla^2 f(y)\| \leq L_H \|x - y\|.$$

Assumption (3)

We have that $s = o(n)$, that is, $\lim_{n \to +\infty} \frac{s}{n} = 0$.

Local convergence: assumptions

Assumption (2')

In a neighborhood of $\bar{x}$, we have

$$\|\nabla^2 f(x) - \nabla^2 f(y)\| \leq L_H \|x - y\|.$$

Assumption (3)

We have that $s = o(n)$, that is, $\lim_{n \to +\infty} \frac{s}{n} = 0$.

Assumption (4)

We assume that

1. *There exists $\sigma \in (0, 1)$ such that $r = rank(\nabla^2 f(\bar{x})) \geq \sigma n$*
2. *There exists $\rho \in (0, 3)$ and $\tilde{C}$ such that in a neighborhood of $\bar{x}$, $f(x_k) - f(\bar{x}) \geq \tilde{C} \|x_k - \bar{x}\|^\rho$ holds.*

Proposition 1

Let $0 < \varepsilon_0 < 1$. Then under Assumptions (3) and (4.1) there exists $n_0 \in \mathbb{N}$ (which depends only on ε_0 and σ) and a neighborhood $B^* \subseteq \bar{B}$ such that if $n \geq n_0$, for any $x \in B^*$,

$$P\nabla^2 f(x)P^\top \succeq \frac{(1-\varepsilon_0)^2 n}{2s}\sigma^2\bar{\lambda}I_s$$

holds with probability at least $1 - 6\exp(-s)$.

Proposition 1

Let $0 < \varepsilon_0 < 1$. Then under Assumptions (3) and (4.1) there exists $n_0 \in \mathbb{N}$ (which depends only on ε_0 and σ) and a neighborhood $B^* \subseteq \bar{B}$ such that if $n \geq n_0$, for any $x \in B^*$,

$$P\nabla^2 f(x)P^\top \succeq \frac{(1-\varepsilon_0)^2 n}{2s}\sigma^2\bar{\lambda}I_s$$

holds with probability at least $1 - 6\exp(-s)$.

Proposition 2

Under Assumptions (1),(2') and (4). there exists $0 < \kappa < 1$, $k_0 \in \mathbb{N}$, $n_0 \in \mathbb{N}$, and $\bar{C} > 0$ such that if $n \geq n_0$, $k \geq k_0$, we have with probability $1 - 6(\exp(-s) + \exp(-\frac{\mathcal{C}_0}{4}s))$:

$$f(x_k) - \min_u f(x_k + P_k^\top u) \geq \bar{C}(f(x_k) - f(\bar{x})).$$

Local convergence: Theorem 1

Theorem

Under Assumptions (1),(2'),(3) and (4), there exists $0 < \kappa < 1$, $k_0 \in \mathbb{N}$, and $n_0 \in \mathbb{N}$ such that if $n \geq n_0$, $k \geq k_0$, then

$$f(x_{k+1}) - f(\bar{x}) \leq \kappa(f(x_k) - f(\bar{x})).$$

holds with probability at least $1 - 6(\exp(-s) + \exp(-\frac{\mathcal{C}_0}{4}s))$.

Super-linear convergence?

Assumption (5)

We assume that

$$(\mathcal{C} + 2)^2 s < n.$$

Super-linear convergence?

Assumption (5)

We assume that

$$(\mathcal{C}+2)^2 s < n.$$

Theorem

Under Assumptions (2') and (5), if f is locally strongly convex around $\bar{x}$. There exists a constant $c > 0$ such that for k large enough,

$$\|x_{k+1} - \bar{x}\| \geq c\|x_k - \bar{x}\|$$

holds with probability at least $1 - 2\exp(-\frac{\mathcal{C}_0}{4}) - 2\exp(-s)$.

Super-linear convergence?

Assumption (5)

We assume that

$$(\mathcal{C}+2)^2 s < n.$$

Theorem

Under Assumptions (2') and (5), if f is locally strongly convex around $\bar{x}$. There exists a constant $c > 0$ such that for k large enough,

$$\|x_{k+1} - \bar{x}\| \geq c\|x_k - \bar{x}\|$$

holds with probability at least $1 - 2\exp(-\frac{\mathcal{C}_0}{4}) - 2\exp(-s)$.

We deduce from the theorem and the assumptions that there exists a constant c' such that

$$f(x_{k+1}) - f(\bar{x}) \geq c'(f(x_k) - f(\bar{x})),$$

with probability at least $1 - 2\exp(-\frac{\mathcal{C}_0}{4}) - 2\exp(-s)$.

Numerical experiments: Support vector regression

Data: $\forall i \leq m,\ (x_i, y_i) \in \mathbb{R}^n \times \{0, 1\}$, we aim minimizing sum of a loss function and a regularizer

$$f(w) = \frac{1}{m} \sum_{i=1}^{m} \ell(y_i - x_i^\top w) + \lambda \|w\|^2.$$

- Internet advertisements dataset from UCI repository[Dua and Graff, 2017] processed so that the number of instances is $m = 600$ and and $n = 1500$.
- Comparison with Gradient Descent (GD) and Regularized Newton Method (RNM)
- Step sizes are all determined by Armijo backtracking line search
- The parameters are fixed as follows:

$$c_1 = 2, c_2 = 1, \gamma = 0.5, \alpha = 0.3, \beta = 0.5, s \in \{100, 200, 400\}.$$

Loss function

$$\ell(t) = \frac{2t^2}{t^2 + 4}$$

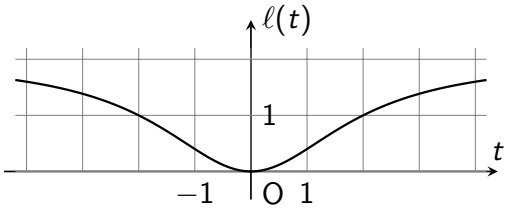


Figure: The robust loss functions.

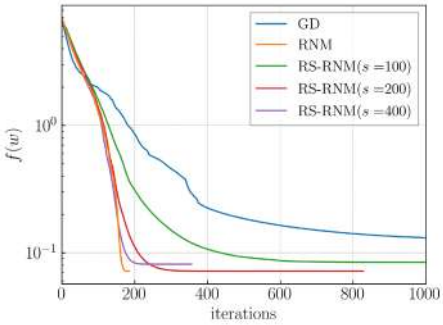


Figure: iterations versus $f(w)$ ($\log_{10}$-scale)

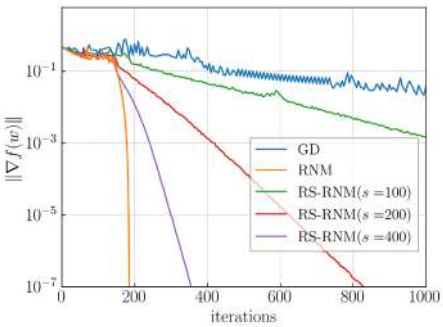


Figure: iterations versus $\|\nabla f(w)\|$ ($\log_{10}$-scale)).

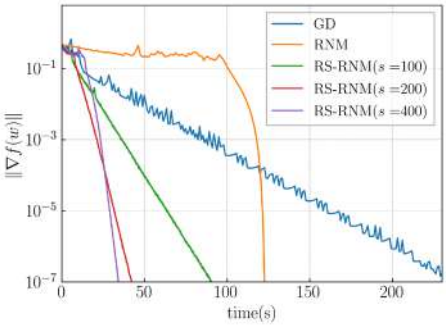


Figure: computation time versus $\|\nabla f(w)\|$ ($\log_{10}$-scale).

Future work

Can we find a second order subspace algorithm with local superlinear convergence ? Full paper: "T. Fuji, P.L. Poirion, A. Takeda, **Randomized**

subspace regularized Newton method for unconstrained non-convex optimization. arXiv:2209.04170, (2022)"

Reference I

Dua, D. and Graff, C. (2017).
UCI machine learning repository.

Gower, R., Kovalev, D., Lieder, F., and Richtárik, P. (2019).
RSN: Randomized Subspace Newton.
Adv. Neural Inf. Process. Syst., 32:616–625.

Hanzely, F., Doikov, N., Richtárik, P., and Nesterov, Y. (2020).
Stochastic subspace cubic Newton method.
In III, H. D. and Singh, A., editors, *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 4027–4038. PMLR.

Kovalev, D., Gower, R. M., Richtárik, P., and Rogozin, A. (2020).
Fast linear convergence of randomized bfgs.
arXiv preprint arXiv:2002.11337.

Roberts, L. and Royer, C. W. (2022).
Direct search based on probabilistic descent in reduced spaces.
arXiv preprint arXiv:2204.01275.

Reference II

Ueda, K. and Yamashita, N. (2010).
Convergence properties of the regularized newton method for the unconstrained nonconvex optimization.
Appl. Math. Optim., 62(1):27–46.

Minimax Analysis for Inverse Risk in Nonparametric Invertible Regression

Akifumi OKUNO

Institute of Statistical Mathematics / RIKEN AIP, Japan
okuno@ism.ac.jp

Learning invertibility from data and exploiting an invertible estimator are used in many domains, such as statistics, econometrics, and machine learning. Although the consistency and universality of invertible estimators have been well investigated, analysis on the efficiency of these methods is still under development. In this study, we study a minimax risk for estimating invertible functions. We first introduce an inverse L2-risk to evaluate an estimator which preserves invertibility. Then, we derive lower and upper rates for a minimax inverse risk by exploiting a representation of invertible functions using level-sets. To obtain an upper bound, we develop an estimator asymptotically almost everywhere invertible, whose risk attains the derived minimax lower rate up to logarithmic factors. This work is a joint work with M. Imaizumi (U. Tokyo), and is based on a preprint of ours [1].

References

[1] Akifumi Okuno, Masaaki Imaizumi, “Minimax Analysis for Inverse Risk in Nonparametric Planer Invertible Regression”, CoRR, arXiv preprint https://arxiv.org/abs/2112.00213

Minimax Analysis for Inverse Risk in Nonparametric Invertible Regression

(joint work with M. Imaizumi, arXiv:2112.00213)

Akifumi Okuno

Institute of Statistical Mathematics and RIKEN AIP

Sep. 2022

Summary of This Talk

This study focuses on **invertibility** of the function.
We estimate *invertible* regression function $\hat{\mathbf{f}}_n : [-1,1]^d \to [-1,1]^d$ and evaluate *invertible risk*

$$\mathrm{R}_{\mathrm{INV}}(\hat{\mathbf{f}}_n, \mathbf{f}_*) := \|\hat{\mathbf{f}}_n - \mathbf{f}_*\|^2_{L^2(P_X)} + \psi(\|\hat{\mathbf{f}}_n^{\dagger} - \mathbf{f}_*^{-1}\|_{L^2(P_X)}).$$

Our contribution ($d = 2$; planer invertible regression; OI2021)

With $\psi(z) = z^4$,

$$\inf_{\hat{\mathbf{f}}_n} \sup_{\mathbf{f}_* \in \mathcal{F}^{\mathrm{Lip}}_{\mathrm{Inv}}} \mathrm{R}_{\mathrm{INV}}(\hat{\mathbf{f}}_n, \mathbf{f}_*) \asymp n^{-2/(2+d)}$$

up to logarithmic factors, **same as the (standard) Lipschitz function estimation!**

- We can employ this minimax rate as **a baseline of efficiency!**
- Generalized to $d \in \mathbb{N}, \psi(z) = z^2$ by assuming C^2 in OI (in prep.)

Background

Invertible Functions

Let $I = [-1, 1]$. A function $\mathbf{f} : I^d \to I^d$ is *invertible* iff

$$\mathbf{f}^{-1}(\mathbf{y}) := \{\mathbf{x} \in I^d \mid \mathbf{f}(\mathbf{x}) = \mathbf{y}\}$$

is a unique point, for any $\mathbf{y} \in I^d$. We consider Lipschitz invertible functions $\mathbf{f} \in \mathcal{F}_{\text{Inv}}^{\text{Lip}}$.

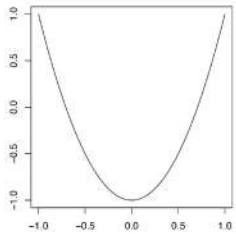


Figure: Non-Invertible $f(x) = 2x^2 - 1$

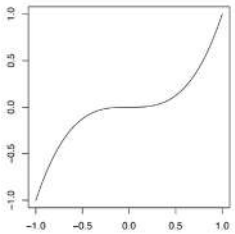
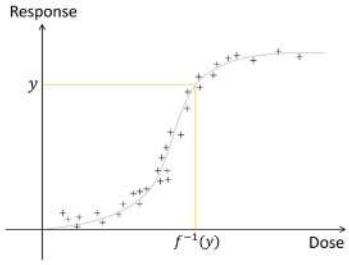


Figure: *Invertible* $f(x) = x^3$

Invertible Function Estimation ($d = 1$)

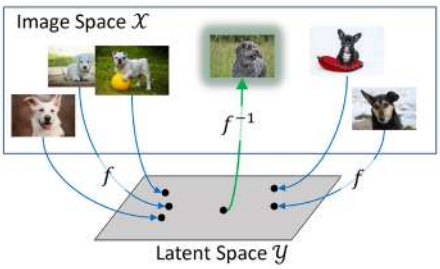
Invertiblility = (Strict) **Monotonicity**



- Many papers on application/theory of monotone func. estimation in econ/stats.
  - Nonparametric statistical calibration (e.g., Tang et al., 2011, 2015)
  - Nonparametric instrumental variable regression (e.g., Krief, 2017)

Invertible Function Estimation ($d \in \mathbb{N}$)

Invertiblility = **One-to-one correspondence**



Usually, it is quite difficult to define *invertibile* and *expressive* estimator for $d \geq 2$.
Recent way: Invertible Neural Network = *Normalizing Flow* (Dinh et al., 2014).

Types of Normalizing Flows and Universality

There are various types of normalizing flows (NF), where they are basically in the form of

$$\mathbf{f}(\mathbf{x}) = (\boldsymbol{\phi}_1 \circ \boldsymbol{\psi}_1 \circ \boldsymbol{\phi}_2 \circ \boldsymbol{\psi}_2 \cdots \circ \boldsymbol{\psi}_{L-1} \circ \boldsymbol{\phi}_L)(\mathbf{x})$$

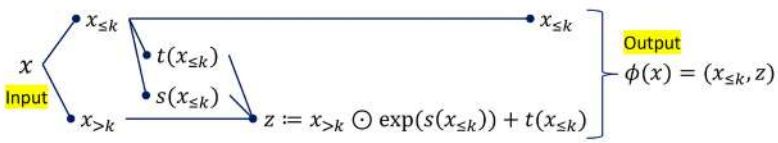
with invertible $\boldsymbol{\phi}_1, \boldsymbol{\phi}_2, \cdots, \boldsymbol{\phi}_L : \mathbb{R}^d \to \mathbb{R}^d$ and Affine mappings $\boldsymbol{\psi}_1, \boldsymbol{\psi}_2, \cdots, \boldsymbol{\psi}_{L-1}$.

(i) **Simple ones:** Non-universal
- Planar flow $\boldsymbol{\phi}_j(\mathbf{x}) = \mathbf{x} + \mathbf{a}_j \mathbf{h}(\mathbf{B}_j^\top \mathbf{x} + \mathbf{c}_j)$,
- Householder flow $\boldsymbol{\phi}_j(\mathbf{x}) = \mathbf{x} - 2\mathbf{v}_j \mathbf{v}_j^\top \mathbf{x}$, etc...

(ii) **Triangular map-based:** Universal (in the sense of distribution matching)
- Sum-of-Squares (SoS; Huang et al., 2018),
- Neural Autoregressive (NAF; Huang et al., 2018), etc...

(iii) **Real NVP:** Universal (in the usual sense)
- Affine-coupling flow (ACF; Dinh et al., 2014) $\boldsymbol{\phi}_j(\mathbf{x}) = (\mathbf{x}_{\leq k}, \mathbf{x}_{>k} \odot \exp(\mathbf{s}_j(\mathbf{x}_{\leq k})) + \mathbf{t}_j(\mathbf{x}_{\leq k}))$ equipped with NNs $\mathbf{s}_j, \mathbf{t}_j : \mathbb{R}^k \to \mathbb{R}^{d-k}$ and $k \in [d]$.

Affine-Coupling Flow (ACF)



- **ACF is invertible**: $\mathbf{f}^{-1}(\mathbf{y}) = (\boldsymbol{\phi}_L^{-1} \circ \boldsymbol{\psi}_{L-1}^{-1} \cdots \circ \boldsymbol{\psi}_2^{-1} \circ \boldsymbol{\phi}_2^{-1} \circ \boldsymbol{\psi}_1^{-1} \circ \boldsymbol{\phi}_1^{-1})(\mathbf{y})$ with

$$\boldsymbol{\phi}_j^{-1}(\mathbf{y}) = \left(\mathbf{y}_{\leq k}, \frac{\mathbf{y}_{>k} - \mathbf{t}_j(\mathbf{y}_{\leq k})}{\exp(\mathbf{s}_j(\mathbf{y}_{\leq k}))} \right).$$

- With increasing number of layers $L \to \infty$,
  ACF universally approximates C^2 invertible functions (Teshima et al., 2020).

Still difficult to evaluate the *efficiency, for* $d \geq 2$.

- Teshima et al. (2020) assumes $L \to \infty$.
- Even the (simple) minimax optimal convergence rate is not obtained.

- $d = 1$ is OK: monotonicity is easy enough to handle. $\exists$Many studies.
- $d \geq 2$ is very difficult: monotonicity is no longer available.
  Even the characterization of the invertible function is not known:
  nonparametric estimator (for theory) is not known.

There is a *HUGE* gap from $d = 1$ to $d \geq 2$:
we evaluate the efficiency for $d = 2$.

Conventional Theory and Our Problem Setup: Inverse Risk

Regression Problem

$$\mathcal{F}_{\mathrm{Inv}} := \{\mathbf{f} : I^2 \to I^2 \mid \forall \mathbf{y} \in I^2,\ !\exists \mathbf{x} \in I^2 \text{ such that } \mathbf{f}(\mathbf{x}) = \mathbf{y}\} \quad (I := [-1, 1]),$$
$$\mathcal{F}_{\mathrm{Inv}}^{\mathrm{Lip}} := \{\mathbf{f} \in \mathcal{F}_{\mathrm{Inv}} \mid \mathbf{f}, \mathbf{f}^{-1} \text{ are Lipschitz}\}.$$

Assume we have observations $\mathfrak{D}_n := \{(\mathbf{X}_i, \mathbf{Y}_i)\}_{i=1}^n \subset I^2 \times \mathbb{R}^2$ that independently follow

$$\mathbf{Y}_i = \mathbf{f}_*(\mathbf{X}_i) + \boldsymbol{\varepsilon}_i, \quad \boldsymbol{\varepsilon}_i \overset{\text{i.i.d.}}{\sim} N_2(\mathbf{0}, \sigma^2 \mathbf{I}_2), \quad i = 1, 2, \ldots, n,$$

for a true function $\mathbf{f}_* \in \mathcal{F}_{\mathrm{Inv}}^{\mathrm{Lip}}$ and $\sigma^2 > 0$.

- $\hat{\mathbf{f}}_n$ estimates $\mathbf{f}_*$, using the observations $\mathfrak{D}_n$.
- Note: $d = 2$ is assumed throughout this talk.

Consistency

Definition (Risk)

For any estimator $\bar{\mathbf{f}}_n$, we define a L^2-risk:

$$\mathrm{R}(\bar{\mathbf{f}}_n, \mathbf{f}_*) := |||\bar{\mathbf{f}}_n - \mathbf{f}_*|||^2_{L^2(P_X)},$$

where $|||\mathbf{f}|||_{L^2(P_X)} := (\sum_{j=1}^2 \int |f_j|^2 \mathrm{d}P_X)^{1/2}$ is an L^2-norm.

Definition (Consistency)

A estimator $\bar{\mathbf{f}}_n$ is *consistent* if

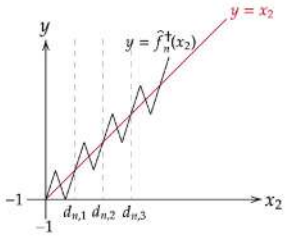
$$\mathbb{P}(\mathrm{R}(\bar{\mathbf{f}}_n, \mathbf{f}_*) \le C r_n) \ge 1 - \delta_n$$

holds for some $C \in (0, \infty)$ and decreasing sequences $r_n, \delta_n \searrow 0$. r_n is also called *convergence rate*.

Kernel smoother is consistent with $r_n = n^{-2/(2+d)}$, for Lipschitz $\mathbf{f}_*$.

Consistency $\neq$ Invertibility: An Example

$$\mathbf{f}_*(\mathbf{x}) = \mathbf{x}, \quad \hat{\mathbf{f}}_n(\mathbf{x}) = (x_1, \hat{f}_n^\dagger(x_2)),$$



With $d_{n,j} = -1 + j\gamma_n$, $\hat{\mathbf{f}}_n^\dagger$ is consistent with the (arbitrarily fast) rate γ_n, whereas it is *NOT* invertible over entire $I^2 = [-1, 1]^2$.

Inverse Risk Measures both Consistency and Invertibility

Definition (Empirical inverse function)

Let $\mathbf{c} \in \mathbb{R}^2 \setminus I^2$ be a constant vector. An inverse function for the estimator $\bar{\mathbf{f}}_n : I^2 \to I^2$ is:

$$\bar{\mathbf{f}}_n^\dagger(\mathbf{y}) := \begin{cases} \mathbf{x} & (\text{if } !\exists \mathbf{x} \in I^2 \text{ such that } \bar{\mathbf{f}}_n(\mathbf{x}) = \mathbf{y}) \\ \mathbf{c} & (\text{otherwise}) \end{cases}, \quad \forall \mathbf{y} \in I^2.$$

Definition (Inverse risk)

$$\mathrm{R}_{\mathrm{INV}}(\bar{\mathbf{f}}_n, \mathbf{f}_*) := \mathrm{R}(\bar{\mathbf{f}}_n, \mathbf{f}_*) + \psi(\mathrm{R}(\bar{\mathbf{f}}_n^\dagger, \mathbf{f}_*^{-1})), \quad \text{for } \bar{\mathbf{f}}_n : I^2 \to I^2.$$

- Inverse risk measures both invertibility (a.e.) and consistency (for both $\bar{\mathbf{f}}_n, \bar{\mathbf{f}}_n^\dagger$).
- The previous approximation example: $\mathrm{R}(\bar{\mathbf{f}}_n, \mathbf{f}_*) \to^p 0$, $\mathrm{R}_{\mathrm{INV}}(\bar{\mathbf{f}}_n, \mathbf{f}_*) > \exists c > 0$.

Level-Set Representation

Level-Set Representation

Definition (Level-Set Representation)

For $\mathbf{f}(\mathbf{x}) = (f_1(\mathbf{x}), f_2(\mathbf{x})) \in \mathcal{F}_{\text{Inv}}$, we define a level-set $L_{f_j}(y_j) := \{\mathbf{x} \in I^2 \mid f_j(\mathbf{x}) = y_j\}$ and the level-set representation

$$\mathbf{f}^{-1}(\mathbf{y}) = \{\mathbf{x} \in I^2 \mid \mathbf{f}(\mathbf{x}) = \mathbf{y}\} = L_{f_1}(y_1) \cap L_{f_2}(y_2), \quad \forall \mathbf{y} = (y_1, y_2) \in I^2.$$

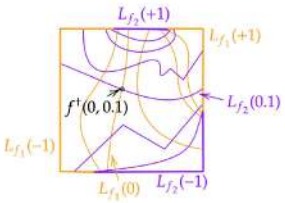


Figure: $\mathbf{f}^{-1}(0, 0.1) = L_{f_1}(0) \cap L_{f_2}(0.1)$

- Example: for $\mathbf{f}(\mathbf{x}) = \mathbf{x}$, $L_{f_1}(y_1) = (y_1, I)$, $L_{f_2}(y_2) = (I, y_2)$.

An Real Example

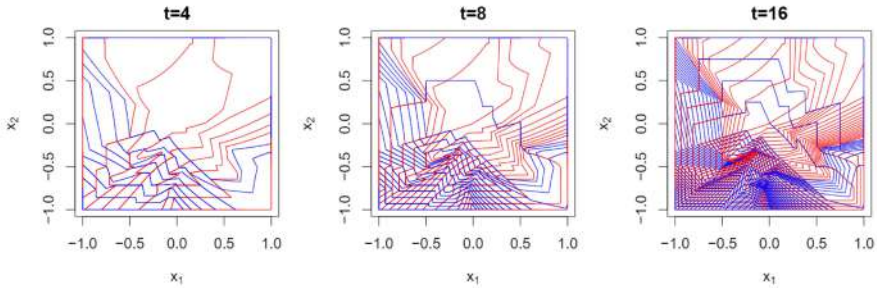


Figure: $\{L_{f_j}(\pm k/t)\}_{k=0,1,2,\ldots,t}$ (red for $j=1$, blue for $j=2$).

Level-Set Properties (in Theory)

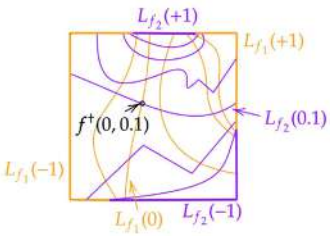


Figure: $\mathbf{f}^{-1}(0, 0.1) = L_{f_1}(0) \cap L_{f_2}(0.1)$

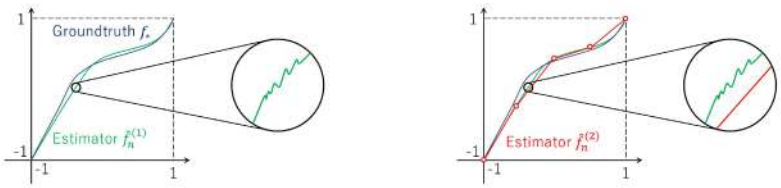
For any $\mathbf{f} = (f_1, f_2) \in \mathcal{F}_{\text{Inv}}^{\text{Lip}}$,

- $L_{f_1}(y_1) = \cup_{\alpha \in I} \mathbf{f}^{-1}(y_1, \alpha)$ and $L_{f_1}(y_2) = \cup_{\alpha \in I} \mathbf{f}^{-1}(\alpha, y_2)$.
- $d_{\text{Hausdorff}}(L_{f_j}(y), L_{f_j}(y')) \leq \exists C |y - y'|$, $\forall y, y' \in I$, $j = 1, 2$.
- $L_{f_j}(\pm 1) \subset \partial I^2$, $j = 1, 2$. (more specifically, $\mathbf{f}(\partial I^2) = \partial I^2 = \mathbf{f}^{-1}(\partial I^2)$)

Proposed (Asymptotically A.E.) Invertible Estimator

Basic Idea: Two-Step Estimation

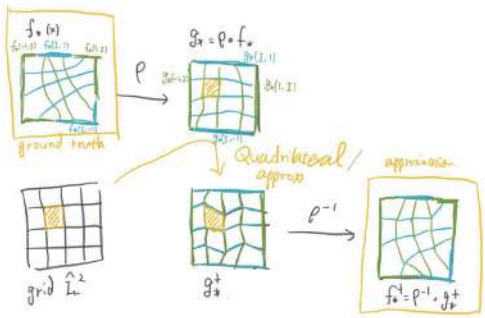
Example: in the case $d = 1$.



1. Compute $\hat{\mathbf{f}}_n^{(1)}$ over the grid
2. Interpolate them using the *line* (as the second-step estimator $\hat{\mathbf{f}}_n^{(2)}$).

Planer Invertible Regression ($d = 2$)

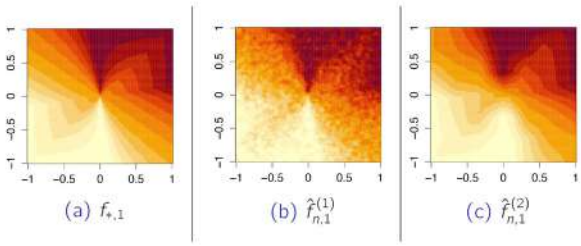
Level set representation of $\mathbf{f}_*^{-1}$ yields $\mathbf{f}_*(\mathbf{x}) = (\mathbf{f}_*^{-1})^{-1}(\mathbf{x}) = \mathbf{f}_*(x_1, l) \cap \mathbf{f}_*(l, x_2)$.



1. Compute $\hat{\mathbf{f}}_n^{(1)}$ over the grid
2. Interpolate them using the *quadrilateral* (as the second-step estimator $\hat{\mathbf{f}}_n^{(2)}$).

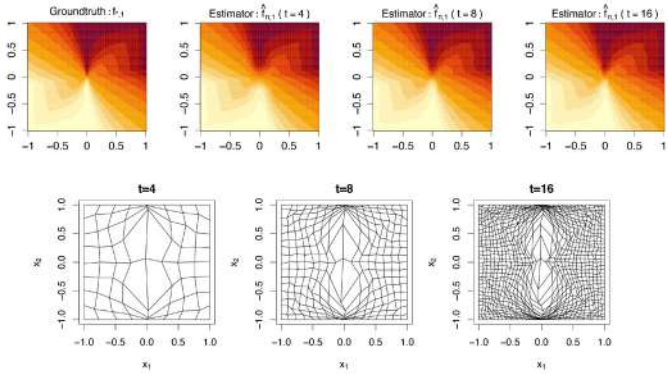
Numerical Experiments: Approximation

- $n = 10^4, \sigma^2 = 10^{-1}$(larger noise), $\mathbf{X}_i \sim U(l^2)$.
- $t = 3$.

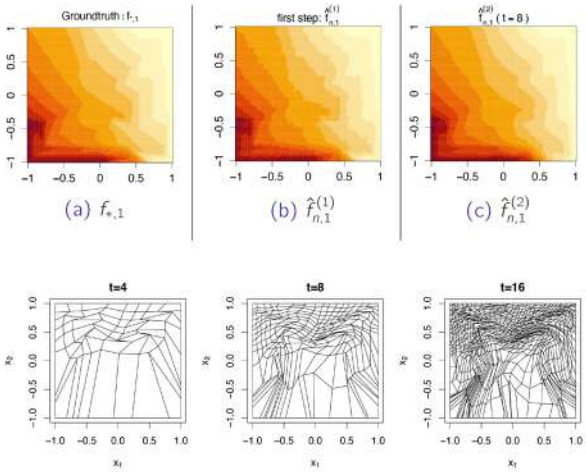


(a) $f_{*,1}$ (b) $\hat{f}_{n,1}^{(1)}$ (c) $\hat{f}_{n,1}^{(2)}$

Numerical Experiments: Number of Splits $t = t_n$



Numerical Experiments: Other Functions



(a) $f_{*,1}$ (b) $\hat{f}^{(1)}_{n,1}$ (c) $\hat{f}^{(2)}_{n,1}$

Lower/Upper Bound Analysis

Lower Bound Analysis

Let $d = 2, \psi(z) = z^4$.

Theorem (Lower Bound)

$$C_* n^{-2/(2+d)} \leq \inf_{\hat{\mathbf{f}}_n} \sup_{\mathbf{f}_* \in \mathcal{F}_{Inv}^{Lip}} \mathrm{R}_{INV}(\hat{\mathbf{f}}_n, \mathbf{f}_*)$$

with probability larger than $1/2$.

Theorem (Upper Bound)

$$\inf_{\hat{\mathbf{f}}_n} \sup_{\mathbf{f}_* \in \mathcal{F}_{Inv}^{Lip}} \mathrm{R}_{INV}(\hat{\mathbf{f}}_n, \mathbf{f}_*) \leq \bar{C} n^{-2/(2+d)} (\log n)^{2\alpha'}$$

w.p. $1 - \delta_n$ ($\nearrow 1$), *for any* $\alpha' > 0$.

See OI (2021) for details.

Ongoing Work and Conclusion

Ongoing Work

- Generalization to $d \in \mathbb{N}$ (OI, in prep.) by assuming C^q-smoothness ($q \geq 2$).

Theorem

Let $d \in \mathbb{N}$. There exists $\bar{C} \in (0, \infty)$ such that,

$$\inf_{\hat{\mathbf{f}}_n} \sup_{\mathbf{f}_* \in \mathcal{F}^q_{Inv}} \tilde{\mathrm{R}}_{INV}(\hat{\mathbf{f}}_n, \mathbf{f}_*) \leq \bar{C} n^{-2q/(2q+d)} (\log n)^{2\alpha'} \quad w.p.a.l.\ 1 - \delta_n$$

Table: Studies on minimax optimality of the estimation of invertible functions $\mathbf{f} \in C^q([-1,1]^d)$.

	$d = 1$	$d = 2$	$d = 3, 4, 5, 6, \ldots$
$q < 1$	Existing	$\times$	$\times$
Lipschitz (nearly $q = 1$)		OI (2021)	$\times$
$1 < q < 2$		$\times$	$\times$
$2 \leq q$		**OI (in prep.)**	

Conclusion

- We proved for $d = 2$ that

$$\inf_{\hat{\mathbf{f}}_n} \sup_{\mathbf{f}_* \in \mathcal{F}_{\mathrm{Inv}}^{\mathrm{Lip}}} \mathrm{R}_{\mathrm{INV}}(\hat{\mathbf{f}}_n, \mathbf{f}_*) \asymp n^{-2/(2+d)}$$

  in probability, up to logarithmic factors.
- We proposed a minimax optimal (whereby asymptotically a.e. invertible) estimator $\hat{\mathbf{f}}_n$.

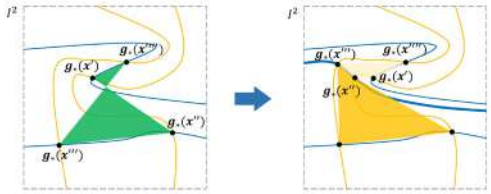
(a) Groundtruth (b) Estimator

https://arxiv.org/abs/2112.00213

Some Remarks

Problem: Quadrilateral Approximation and Twists

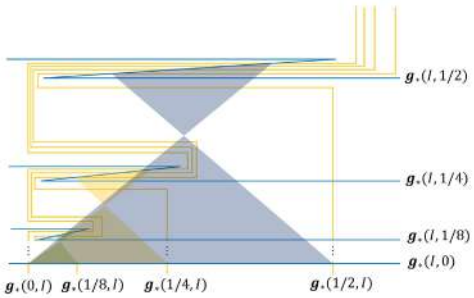
- If $L_{\mathbf{g}_*} \in [1, 2^{1/4})$, no twist appears when approximating quadrilaterals.
- Otherwise, there can be twists.



Each twist vanishes by increasing the number of division (for most suitable cases). Daneri and Pratelli (2014) Proposition 4.1 proves

$$\mathcal{L}(\text{twisted region}) \to^{p} 0.$$

Pathological Example



Whereas each twist is decomposed into smaller quadrilaterals (by increasing $t = t_n$), twists can appear indefinitely in some pathological examples. (They are ignored in the sense of Lebesgue measure, in our theory)

Which is better to assume: Lipschitz or C^2?

- **Nonparametric statistics** usually assumes that $\mathbf{f}_*$ is Lipschitz:
  - 👍 Less restrictive
  - 👎 Includes **pathological** examples

  This study assumes Lipschitz (with $d = 2$): as we are researchers of statistics...
  Almost impossible to extend to general $d \geq 3$.

- **Geometry** usually assumes that $\mathbf{f}_*$ is C^2:
  - 👍 Theoretically tractable (tangent space can be defined)
  - 👎 More restrictive

  Our ongoing work assumes C^2 (and generalize to $d \in \mathbb{N}$).

References I

Daneri, S. and Pratelli, A. (2014). Smooth approximation of bi-lipschitz orientation-preserving homeomorphisms. *Annales de l'IHP Analyse non linéaire*, 31(3):567–589.

Dinh, L., Krueger, D., and Bengio, Y. (2014). NICE: Non-Linear Independent Components Estimation. *arXiv preprint arXiv:1410.8516*.

Donaldson, S. K. and Sullivan, D. P. (1989). Quasiconformal 4-manifolds. *Acta Mathematica*, 163(none):181 – 252.

Huang, C.-W., Krueger, D., Lacoste, A., and Courville, A. (2018). Neural autoregressive flows. In *International Conference on Machine Learning*, pages 2078–2087. PMLR.

Kobyzev, I., Prince, S., and Brubaker, M. (2020). Normalizing flows: An introduction and review of current methods. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(11):3964–3979.

References II

Kong, Z. and Chaudhuri, K. (2020). The expressive power of a class of normalizing flow models. In *Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics*, volume 108 of *Proceedings of Machine Learning Research*, pages 3599–3609. PMLR.

Krief, J. M. (2017). Direct instrumental nonparametric estimation of inverse regression functions. *Journal of Econometrics*, 201(1):95–107.

Tang, R., Banerjee, M., and Michailidis, G. (2011). A two-stage hybrid procedure for estimating an inverse regression function. *The Annals of Statistics*, 39(2):956–989.

Tang, R., Banerjee, M., Michailidis, G., and Mankad, S. (2015). Two-stage plans for estimating the inverse of a monotone function. *Technometrics*, 57(3):395–407.

Teshima, T., Ishikawa, I., Tojo, K., Oono, K., Ikeda, M., and Sugiyama, M. (2020). Coupling-based Invertible Neural Networks Are Universal Diffeomorphism Approximators. In *Advances in Neural Information Processing Systems*, volume 33, pages 3362–3373. Curran Associates, Inc.

References III

Tsybakov, A. B. (2008). *Introduction to nonparametric estimation*. Springer Science & Business Media.

On the geometry of periodic timetables in public transport

Niels LINDNER

Zuse Institute Berlin, Germany
lindner@zib.de

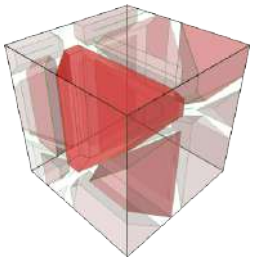
What rhythm is to music, is the timetable to a public transportation system. Many public transportation networks are operated periodically, and therefore the computation and optimization of periodic timetables is a frequent and important task. The mathematical foundation of periodic timetabling is the Periodic Event Scheduling Problem [1], which is easy to formulate, has a rich theory, but is notoriously hard to solve. In order to obtain a better understanding of how to solve periodic timetabling problems, we analyze the geometry of periodic timetables, and discover surprising connections to tropical and discrete geometry that are beyond the scope of the standard toolbox of combinatorial optimization [2]. We outline how tropical neighborhood search, a new heuristic developed from these geometric insights, helped to compute new incumbent solutions for instances of the timetabling benchmarking library PESPlib [3].

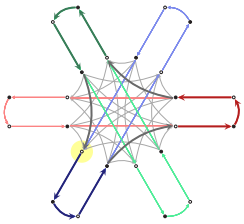
References

[1] P. Serafini and W. Ukovich. A mathematical model for periodic scheduling problems. SIAM Journal on Discrete Mathematics, 2(4):550–581, 1989. https://doi.org/10.1137/0402049

[2] E. Bortoletto, N. Lindner, and B. Masing. The tropical and zonotopal geometry of periodic timetables, 2022. https://doi.org/10.48550/ARXIV.2204.13501

[3] E. Bortoletto, N. Lindner, and B. Masing. Tropical Neighbourhood Search: A New Heuristic for Periodic Timetabling. 22nd Symposium on Algorithmic Approaches for Transportation Modelling, Optimization, and Systems (ATMOS 2022), 3:1-3:19, 2022. https://doi.org/10.4230/OASIcs.ATMOS.2022.3





On the geometry of periodic timetables in public transport

Niels Lindner, Enrico Bortoletto, Berenike Masing

MobilityLab
Department Network Optimization
Zuse Institute Berlin

6th RIKEN-IMI-ISM-NUS-ZIB-MODAL-NHR Workshop

September 21, 2022

Public Transport...

... is often operated **periodically**

Images: MaedaAkihiko and Trouper3000, CC-BY-SA 4.0; Rolf Heinrich, Köln, CC-BY 3.0; all via commons.wikimedia.org

Public Transport...

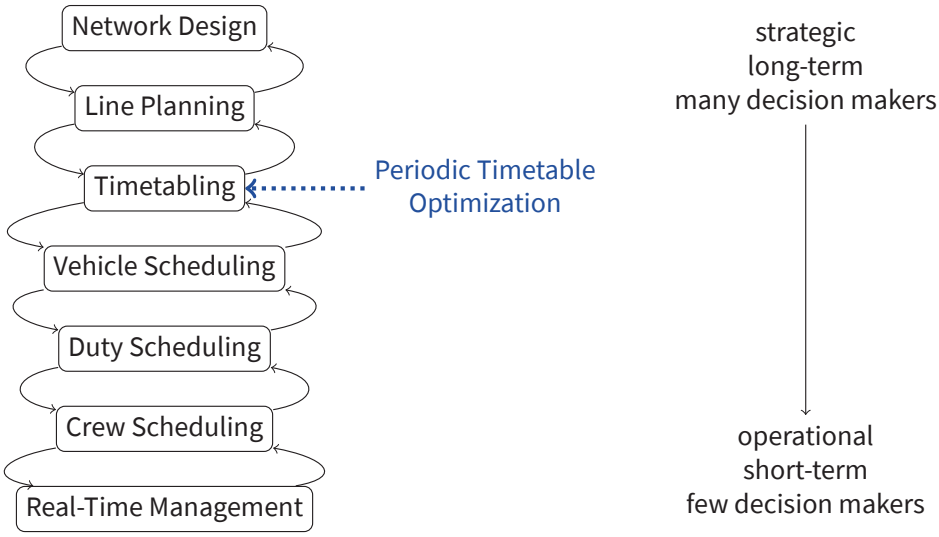


... is often operated **periodically**

→ Periodic Timetable Optimization

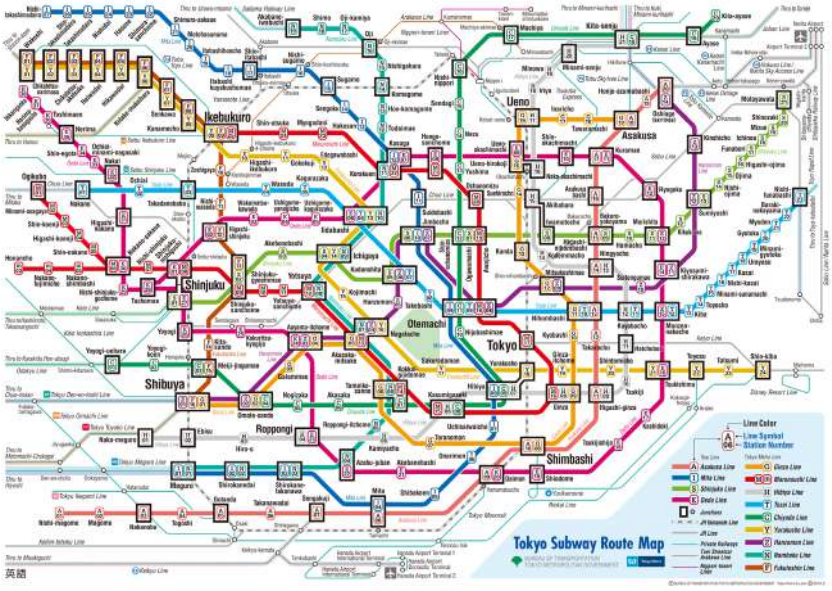
Images: MaedaAkihiko and Trouper3000, CC-BY-SA 4.0; Rolf Heinrich, Köln, CC-BY 3.0; all via commons.wikimedia.org

Public Transport Planning Cycle



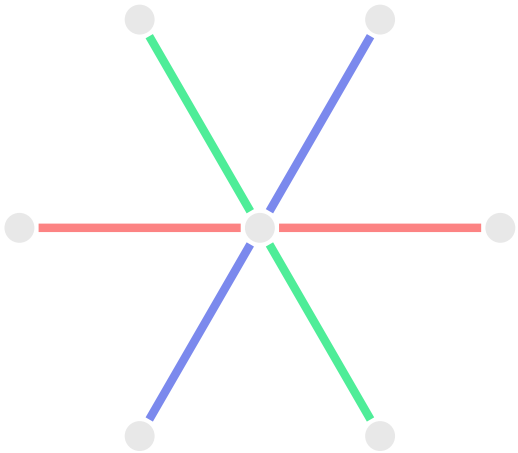
Bussieck et al.: Discrete optimization in public rail transport, 1997
Liebchen: Periodic timetable optimization in public transport, 2006

A Line Network: Tokyo Subway



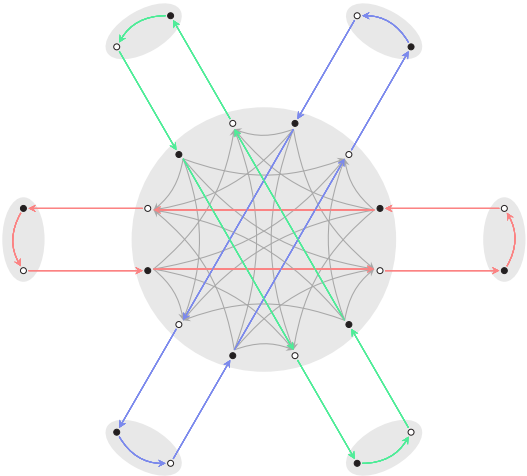
Tokyo Metro Co., Ltd.

From Line Networks to Event-Activity Networks



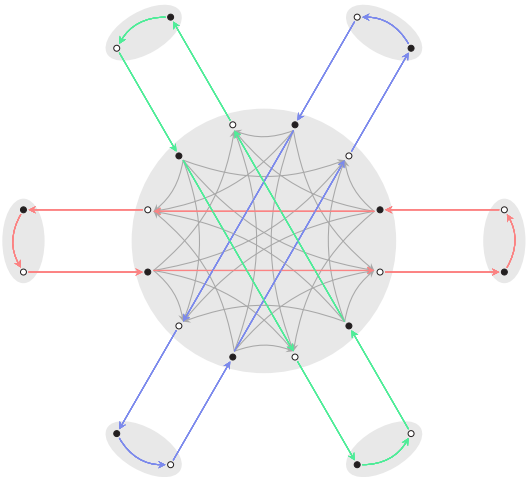
Line Plan: 3 bidirectional lines, same frequency

From Line Networks to Event-Activity Networks



Event-Activity Network: directed graph G

Periodic Timetable Optimization

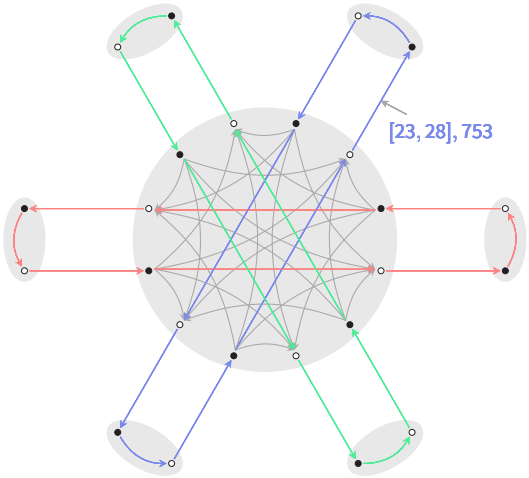


Event-Activity Network: directed graph G

Events:
- ● arrival
- ○ departure

Activities:
- → drive, dwell, turn
- → transfer
- . . .

Periodic Timetable Optimization



(G, T, ℓ, u, w)

Bounds $[\ell, u]$

- driving times
- minimum transfer times
- maximum dwell times
- minimum headway times
- ...

Weights w:

- passenger load
- turnaround penalties
- ...

Period time T:

- e.g., $T = 60$ for 1 hour, resolution of 1 minute

Periodic Event Scheduling Problem (PESP)

Given

$G = (V, A)$	event-activity network,
$T \in \mathbb{N}$	period time,
$\ell \in \mathbb{R}^A$	lower bounds,
$u \in \mathbb{R}^A$	upper bounds,
$w \in \mathbb{R}^A_{\geq 0}$	weights,

find

$\pi \in \mathbb{R}^V$	periodic timetable,
$x \in \mathbb{R}^A$	periodic tension

such that

(1) $\pi_j - \pi_i \equiv x_{ij} \mod T$ for all $ij \in A$,

(2) $\ell \leq x \leq u$,

(3) $w^\top x$ is minimum,

or decide that no such (π, x) exists.

(Serafini and Ukovich, 1989)

Periodic Event Scheduling Problem (PESP)

Given

$G = (V, A)$	event-activity network,
$T \in \mathbb{N}$	period time,
$\ell \in \mathbb{R}^A$	lower bounds,
$u \in \mathbb{R}^A$	upper bounds,
$w \in \mathbb{R}^A_{\geq 0}$	weights,

find

$\pi \in \mathbb{R}^V$	periodic timetable,
$x \in \mathbb{R}^A$	periodic tension

such that

(1) $\pi_j - \pi_i \equiv x_{ij} \mod T$ for all $ij \in A$,

(2) $\ell \leq x \leq u$,

(3) $w^\top x$ is minimum,

or decide that no such (π, x) exists.

(Serafini and Ukovich, 1989)

Incidence-based MIP formulation:

$$
\begin{array}{lll}
\text{Minimize} & w^\top x & \\
\text{s.t.} & \pi_j - \pi_i = x_{ij} - Tp_{ij}, & ij \in A, \\
& \ell_{ij} \leq x_{ij} \leq u_{ij}, & ij \in A, \\
& \pi_i \in \mathbb{R}, & i \in V, \\
& p_{ij} \in \mathbb{Z}, & ij \in A.
\end{array}
$$

$p \in \mathbb{Z}^A$ periodic offsets

Periodic Event Scheduling Problem (PESP)

Given

$G = (V, A)$	event-activity network,
$T \in \mathbb{N}$	period time,
$\ell \in \mathbb{R}^A$	lower bounds,
$u \in \mathbb{R}^A$	upper bounds,
$w \in \mathbb{R}^A_{\geq 0}$	weights,

find

$\pi \in \mathbb{R}^V$	periodic timetable,
$x \in \mathbb{R}^A$	periodic tension

such that

(1) $\pi_j - \pi_i \equiv x_{ij} \mod T$ for all $ij \in A$,

(2) $\ell \leq x \leq u$,

(3) $w^\top x$ is minimum,

or decide that no such (π, x) exists.

(Serafini and Ukovich, 1989)

Incidence-based MIP formulation:

$$
\begin{array}{lll}
\text{Minimize} & w^\top x & \\
\text{s.t.} & \pi_j - \pi_i = x_{ij} - Tp_{ij}, & ij \in A, \\
& \ell_{ij} \leq x_{ij} \leq u_{ij}, & ij \in A, \\
& \pi_i \in \mathbb{R}, & i \in V, \\
& p_{ij} \in \mathbb{Z}, & ij \in A.
\end{array}
$$

$p \in \mathbb{Z}^A$ periodic offsets

Assumptions after preprocessing:

- G is weakly (2-)connected
- G has no arc $a \in A$ with $\ell_a = u_a$
- $0 \leq \ell < T$ and $0 \leq u - \ell < T$

Periodic Event Scheduling Problem (PESP)

Given

$G = (V, A)$	event-activity network,
$T \in \mathbb{N}$	period time,
$\ell \in \mathbb{R}^A$	lower bounds,
$u \in \mathbb{R}^A$	upper bounds,
$w \in \mathbb{R}^A_{\geq 0}$	weights,

find

$\pi \in \mathbb{R}^V$	periodic timetable,
$x \in \mathbb{R}^A$	periodic tension

such that

(1) $\pi_j - \pi_i \equiv x_{ij} \mod T$ for all $ij \in A$,

(2) $\ell \leq x \leq u$,

(3) $w^\top x$ is minimum,

or decide that no such (π, x) exists.

(Serafini and Ukovich, 1989)

Incidence-based MIP formulation:

$$\begin{array}{lll} \text{Minimize} & w^\top x & \\ \text{s.t.} & \pi_j - \pi_i = x_{ij} - Tp_{ij}, & ij \in A, \\ & \ell_{ij} \leq x_{ij} \leq u_{ij}, & ij \in A, \\ & \pi_i \in \mathbb{R}, & i \in V, \\ & p_{ij} \in \mathbb{Z}, & ij \in A. \end{array}$$

$p \in \mathbb{Z}^A$ periodic offsets

Symmetry breaking:

- could impose $0 \leq \pi_i < T$ and $p_{ij} \in \{0, 1, 2\}$

Redundancy among periodic offsets p:

- could set $p_{ij} = 0$ along spanning tree

Hardness of PESP

Theory

- NP-hard for fixed $T \geq 3$ (Odijk, 1994, Nachtigall, 1996)
- NP-hard for G with fixed treewidth ≥ 2 (in particular for planar G) (L. and Reisch, 2022)
- NP-hard cutting plane separation (cycle, change-cycle, flip) (Borndörfer et al., 2020, L. and Liebchen, 2020)
- LP relaxation has trivial solution $\pi^* = 0,\ x^* = \ell,\ p^* = \ell/T$

Hardness of PESP

Theory

- NP-hard for fixed $T \geq 3$ (Odijk, 1994, Nachtigall, 1996)
- NP-hard for G with fixed treewidth ≥ 2 (in particular for planar G) (L. and Reisch, 2022)
- NP-hard cutting plane separation (cycle, change-cycle, flip) (Borndörfer et al., 2020, L. and Liebchen, 2020)
- LP relaxation has trivial solution $\pi^* = 0,\ x^* = \ell,\ p^* = \ell/T$

Practice

- rich literature on algorithms: MIP techniques, CP, SAT (also MaxSAT and SAT+ML), modulo network simplex, matching, merging, maximum cuts, graph partitioning, . . .
- several success stories (Berlin, Copenhagen, Netherlands, Switzerland, . . .)

Hardness of PESP

Theory

- NP-hard for fixed $T \geq 3$ (Odijk, 1994, Nachtigall, 1996)
- NP-hard for G with fixed treewidth ≥ 2 (in particular for planar G) (L. and Reisch, 2022)
- NP-hard cutting plane separation (cycle, change-cycle, flip) (Borndörfer et al., 2020, L. and Liebchen, 2020)
- LP relaxation has trivial solution $\pi^* = 0,\ x^* = \ell,\ p^* = \ell/T$

Practice

- rich literature on algorithms: MIP techniques, CP, SAT (also MaxSAT and SAT+ML), modulo network simplex, matching, merging, maximum cuts, graph partitioning, . . .
- several success stories (Berlin, Copenhagen, Netherlands, Switzerland, . . .)

Summary: primal: 🙂 dual: 😭

PESPlib

Railway timetabling instances

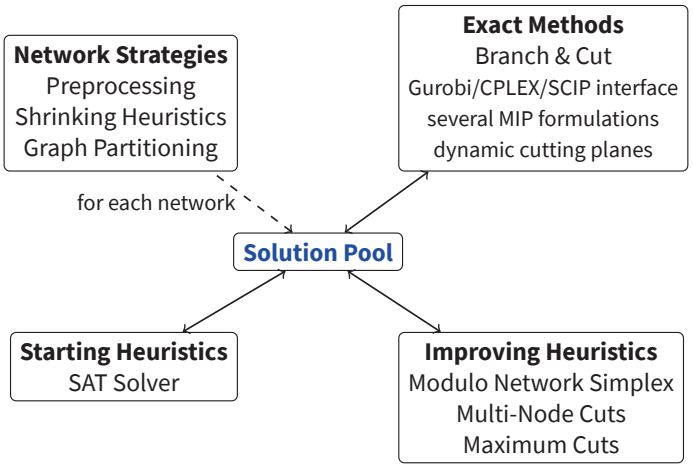
Name	Events	Activities	Best objective (weighted slack)	Author, Date	Best lower bound
R1L1	3664	6386	39,656,259	Goerigk 12/05/2017	20,230,655
			39,539,519	Goerigk 14/06/2012	
			39,216,699	Grossmann 14/09/2012	
			38,384,557	Pätzold 04/07/2017	
			37,338,904	Herrigel 04/06/2013	
			33,711,523	Liebchen 20/04/2017	
			31,838,103	Goerigk & Liebchen 08/05/2017	
			31,194,961	Goerigk & Liebchen 25/06/2017	
			31,099,786	Goerigk & Liebchen 19/05/2017	
			30,780,097	Pätzold 11/10/2018	
			30,463,638	Lindner & Roth 30/01/2019	
			30,415,672	Lindner 23/10/2018	
			29,894,745	Lindner & Liebchen 24/02/2021	

http://num.math.uni-goettingen.de/ m.goerigk/pesplib/

- benchmarking library, est. 2012 by Goerigk
- 22 hard to extremely hard PESP instances
- smallest instance $\rightarrow$ MIP with 2722 (general) integer variables
- *no* instance has been solved to optimality so far

ConcurrentPESP

Concurrent Framework for Periodic Timetable Optimization



...trades off by far more than just concurrency
...holds primal and dual records for *all* 22 PESPlib instances

(Borndörfer, L., Roth, 2020)

Periodic Timetabling Spaces

Question

Can we get more insight by studying the *geometry* of periodic timetables?

Periodic Timetabling Spaces

Question

Can we get more insight by studying the *geometry* of periodic timetables?

Timetabling Spaces

mixed-integer set of feasible solutions

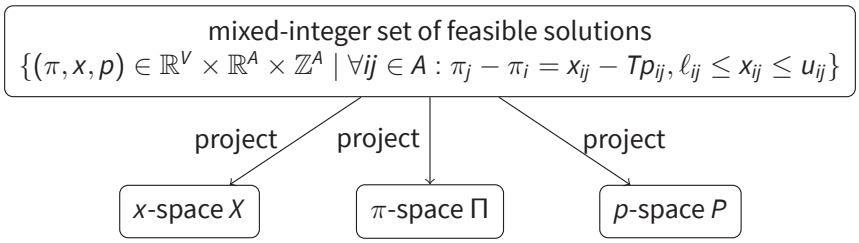
$$\{(\pi, x, p) \in \mathbb{R}^V \times \mathbb{R}^A \times \mathbb{Z}^A \mid \forall ij \in A : \pi_j - \pi_i = x_{ij} - Tp_{ij}, \ell_{ij} \leq x_{ij} \leq u_{ij}\}$$

Periodic Timetabling Spaces

Question

Can we get more insight by studying the *geometry* of periodic timetables?

Timetabling Spaces

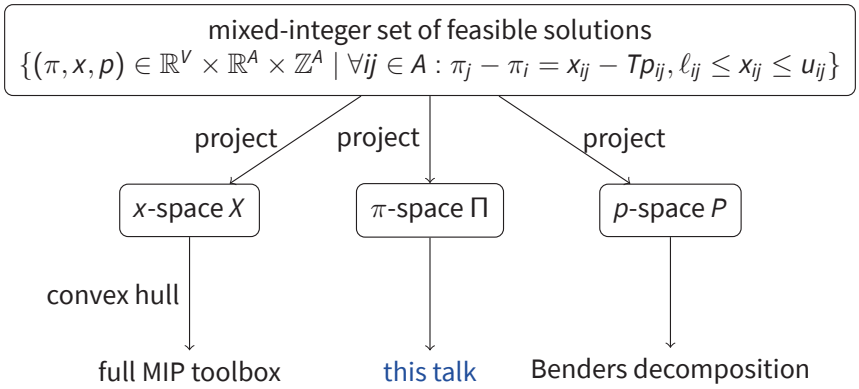


Periodic Timetabling Spaces

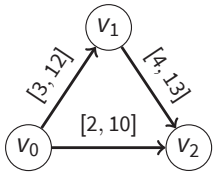
Question

Can we get more insight by studying the *geometry* of periodic timetables?

Timetabling Spaces

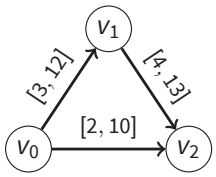


Gallery of Timetabling Spaces

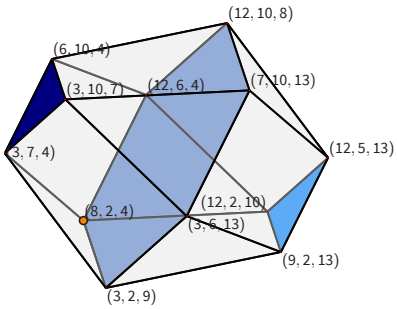


PESP instance with $n = 3, m = 3, T = 10$

Gallery of Timetabling Spaces

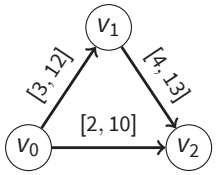


PESP instance with $n = 3, m = 3, T = 10$

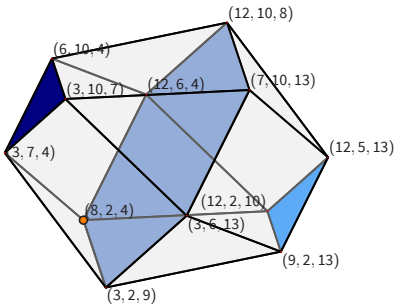


$\mathrm{conv}(X)$

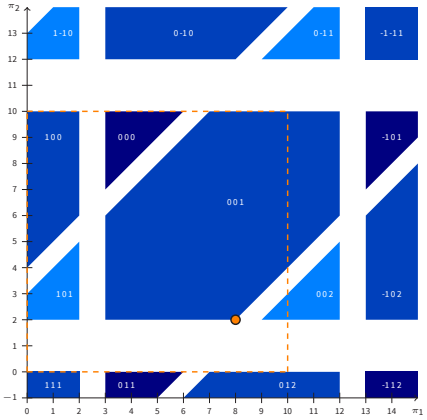
Gallery of Timetabling Spaces



PESP instance with $n = 3, m = 3, T = 10$



conv(X)



$\Pi/\mathbb{R}\mathbf{1}$ is periodically tiled by polyt(r)opes

Decomposing the Space of Periodic Timetables

Decomposition

The space of feasible periodic timetables is

$$\Pi := \{\pi \in \mathbb{R}^V \mid \exists p \in \mathbb{Z}^A : \forall ij \in A : \ell_{ij} \leq \pi_j - \pi_i + Tp_{ij} \leq u_{ij}\}.$$

The space Π decomposes into polyhedral regions:

$$\Pi = \bigcup_{p \in \mathbb{Z}^A} R(p), \text{ where } R(p) := \{\pi \in \mathbb{R}^V \mid \forall ij \in A : \ell_{ij} - Tp_{ij} \leq \pi_j - \pi_i \leq u_{ij} - Tp_{ij}\}.$$

Due to the preprocessing assumption $0 \leq u - \ell < T$, the union is disjoint.

Decomposing the Space of Periodic Timetables

Decomposition

The space of feasible periodic timetables is

$$\Pi := \{\pi \in \mathbb{R}^V \mid \exists p \in \mathbb{Z}^A : \forall ij \in A : \ell_{ij} \leq \pi_j - \pi_i + Tp_{ij} \leq u_{ij}\}.$$

The space Π decomposes into polyhedral regions:

$$\Pi = \bigcup_{p \in \mathbb{Z}^A} R(p), \text{ where } R(p) := \{\pi \in \mathbb{R}^V \mid \forall ij \in A : \ell_{ij} - Tp_{ij} \leq \pi_j - \pi_i \leq u_{ij} - Tp_{ij}\}.$$

Due to the preprocessing assumption $0 \leq u - \ell < T$, the union is disjoint.

Weighted Digraph Polyhedra

Add a reverse copy $\overline{a}$ of each arc a. This produces a new graph $\overline{G} = (\overline{V}, \overline{A})$ with $\overline{V} = V$. If we set $\kappa(p)_a := u_a - Tp_a$ and $\kappa(p)_{\overline{a}} := -\ell_a + Tp_a$, then

$$R(p) = \{\pi \in \mathbb{R}^{\overline{V}} \mid \pi_j - \pi_i \leq \kappa(p)_{ij} \text{ for all } ij \in \overline{A}\}.$$

This means that $R(p)$ is the *weighted digraph polyhedron* (Joswig, Loho, 2016) associated to $(\overline{G}, \kappa(p))$. In combinatorial optimization terms, $R(p)$ is the polyhedron of feasible potentials in $\overline{G}$ w.r.t. the arc costs $\kappa(p)$.

Decomposing the Space of Periodic Timetables

A First Symmetry

If G is weakly connected, then $\overline{G}$ is strongly connected and by (Joswig, Loho, 2016):

- The recession cone of $R(p)$ is $\mathbb{R}\mathbf{1}$.
- The quotient $R(p)/\mathbb{R}\mathbf{1}$ is a polytope.

Choosing coordinates on $R(p)/\mathbb{R}\mathbf{1}$ amounts to the periodic timetabler's wisdom that a timetable π can be fixed at one event $v_0 \in V$ to $\pi_{v_0} := 0$ without affecting feasiblity or optimality.

Decomposing the Space of Periodic Timetables

A First Symmetry

If G is weakly connected, then $\overline{G}$ is strongly connected and by (Joswig, Loho, 2016):

- The recession cone of $R(p)$ is $\mathbb{R}\mathbf{1}$.
- The quotient $R(p)/\mathbb{R}\mathbf{1}$ is a polytope.

Choosing coordinates on $R(p)/\mathbb{R}\mathbf{1}$ amounts to the periodic timetabler's wisdom that a timetable π can be fixed at one event $v_0 \in V$ to $\pi_{v_0} := 0$ without affecting feasiblity or optimality.

Polytropes

A *polytrope* is the convex hull of finitely many points, both in the ordinary and the tropical sense:

$$\operatorname{tconv}(x_1, \ldots, x_n) := \left\{ \bigoplus_{i=1}^{n} \lambda_i \odot x_i \;\middle|\; \lambda_1, \ldots, \lambda_n \in \mathbb{R} \right\} = \left\{ \min_{i=1}^{n} (\lambda_i + x_i) \;\middle|\; \lambda_1, \ldots, \lambda_n \in \mathbb{R} \right\}.$$

Polytropes are exactly the quotients of weighted digraph polyhedra of strongly connected digraphs by $\mathbb{R}\mathbf{1}$ (Joswig, Kulas, 2010).

Decomposing the Space of Periodic Timetables

A First Symmetry

If G is weakly connected, then $\overline{G}$ is strongly connected and by (Joswig, Loho, 2016):

- The recession cone of $R(p)$ is $\mathbb{R}\mathbf{1}$.
- The quotient $R(p)/\mathbb{R}\mathbf{1}$ is a polytope.

Choosing coordinates on $R(p)/\mathbb{R}\mathbf{1}$ amounts to the periodic timetabler's wisdom that a timetable π can be fixed at one event $v_0 \in V$ to $\pi_{v_0} := 0$ without affecting feasiblity or optimality.

Polytropes

A *polytrope* is the convex hull of finitely many points, both in the ordinary and the tropical sense:

$$\operatorname{tconv}(x_1, \ldots, x_n) := \left\{ \bigoplus_{i=1}^{n} \lambda_i \odot x_i \;\middle|\; \lambda_1, \ldots, \lambda_n \in \mathbb{R} \right\} = \left\{ \min_{i=1}^{n} (\lambda_i + x_i) \;\middle|\; \lambda_1, \ldots, \lambda_n \in \mathbb{R} \right\}.$$

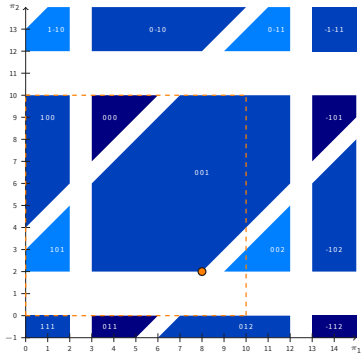
Polytropes are exactly the quotients of weighted digraph polyhedra of strongly connected digraphs by $\mathbb{R}\mathbf{1}$ (Joswig, Kulas, 2010).

Corollary: $\Pi/\mathbb{R}\mathbf{1}$ decomposes into the disjoint union of the polytropes $R(p)/\mathbb{R}\mathbf{1}$.

The Periodic Timetabling Torus

Periodicity: If $\pi \in \Pi$, then $\pi + Tq \in \Pi$ for all $q \in \mathbb{Z}^V$. $\rightsquigarrow$ Consider the space of timetables inside the $(|V| - 1)$-dimensional torus

$$\mathcal{T} := (\mathbb{R}^V/(T\mathbb{Z})^V)/\mathbb{R}\mathbf{1}.$$

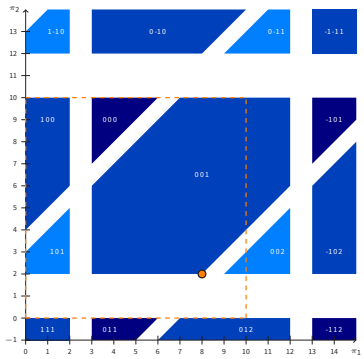


The Periodic Timetabling Torus

Periodicity: If $\pi \in \Pi$, then $\pi + Tq \in \Pi$ for all $q \in \mathbb{Z}^V$. $\rightsquigarrow$ Consider the space of timetables inside the $(|V| - 1)$-dimensional torus

$$\mathcal{T} := (\mathbb{R}^V/(T\mathbb{Z})^V)/\mathbb{R}\mathbf{1}.$$

Redundancy of periodic offsets: Let Γ be the cycle matrix of an integral cycle basis $\mathcal{B}$ of G. Then $R(p) \equiv R(p')$ on $\mathcal{T}$ iff $\Gamma p = \Gamma p'$. We can hence denote $R(p)$ modulo $\mathcal{T}$ by $\mathbf{R}(z)$, where $z := \Gamma p \in \mathbb{Z}^{\mathcal{B}}$.

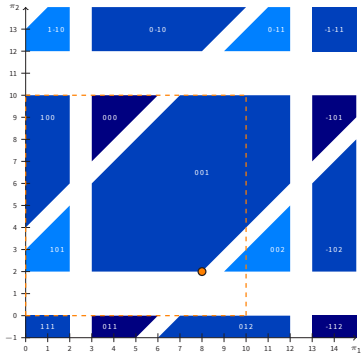


The Periodic Timetabling Torus

Periodicity: If $\pi \in \Pi$, then $\pi + Tq \in \Pi$ for all $q \in \mathbb{Z}^V$. $\rightsquigarrow$ Consider the space of timetables inside the $(|V|-1)$-dimensional torus

$$\mathcal{T} := (\mathbb{R}^V/(T\mathbb{Z})^V)/\mathbb{R}\mathbf{1}.$$

Redundancy of periodic offsets: Let Γ be the cycle matrix of an integral cycle basis $\mathcal{B}$ of G. Then $R(p) \equiv R(p')$ on $\mathcal{T}$ iff $\Gamma p = \Gamma p'$. We can hence denote $R(p)$ modulo $\mathcal{T}$ by $\mathbf{R}(z)$, where $z := \Gamma p \in \mathbb{Z}^{\mathcal{B}}$.



Running example:

$$z = \frac{\Gamma x}{T} \leq \left\lfloor \frac{12-2+13}{10} \right\rfloor = 2,$$

$$z = \frac{\Gamma x}{T} \geq \left\lceil \frac{3-10+4}{10} \right\rceil = 0,$$

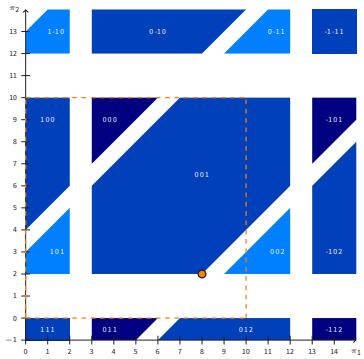
$\rightsquigarrow$ at most $\mathbf{R}(0), \mathbf{R}(1), \mathbf{R}(2)$ are in $\mathcal{T}$.

The Periodic Timetabling Torus

Periodicity: If $\pi \in \Pi$, then $\pi + Tq \in \Pi$ for all $q \in \mathbb{Z}^V$. $\rightsquigarrow$ Consider the space of timetables inside the $(|V|-1)$-dimensional torus

$$\mathcal{T} := (\mathbb{R}^V/(T\mathbb{Z})^V)/\mathbb{R}\mathbf{1}.$$

Redundancy of periodic offsets: Let Γ be the cycle matrix of an integral cycle basis $\mathcal{B}$ of G. Then $R(p) \equiv R(p')$ on $\mathcal{T}$ iff $\Gamma p = \Gamma p'$. We can hence denote $R(p)$ modulo $\mathcal{T}$ by $\mathbf{R}(z)$, where $z := \Gamma p \in \mathbb{Z}^{\mathcal{B}}$.

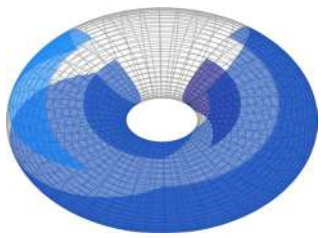


Running example:

$$z = \frac{\Gamma x}{T} \leq \left\lfloor \frac{12-2+13}{10} \right\rfloor = 2,$$

$$z = \frac{\Gamma x}{T} \geq \left\lceil \frac{3-10+4}{10} \right\rceil = 0,$$

$\rightsquigarrow$ at most $\mathbf{R}(0), \mathbf{R}(1), \mathbf{R}(2)$ are in $\mathcal{T}$.



More on Timetabling Polytropes

Dimension

- $R(p) = \emptyset$ if and only if $\overline{G}$ contains a negative weight directed cycle w.r.t. $\kappa(p)$.
- The dimension of $R(p)/\mathbb{R}\mathbf{1}$ is the number of connected components of the equality graph of $(\overline{G}, \kappa(p))$ minus 1 (Joswig, Loho, 2016).

More on Timetabling Polytropes

Dimension

- $R(p) = \emptyset$ if and only if $\overline{G}$ contains a negative weight directed cycle w.r.t. $\kappa(p)$.
- The dimension of $R(p)/\mathbb{R}\mathbf{1}$ is the number of connected components of the equality graph of $(\overline{G}, \kappa(p))$ minus 1 (Joswig, Loho, 2016).

Vertices

- Every vertex of $R(p)/\mathbb{R}\mathbf{1}$ corresponds to a unique spanning subgraph of $\overline{G}$.
- For each $i \in V$, the i-th tropical vertex of $R(p)/\mathbb{R}\mathbf{1}$ corresponds to a shortest path tree of $(\overline{G}, \kappa(p))$ rooted at i (Joswig, Kulas, 2010).

More on Timetabling Polytropes

Dimension

- $R(p) = \emptyset$ if and only if $\overline{G}$ contains a negative weight directed cycle w.r.t. $\kappa(p)$.
- The dimension of $R(p)/\mathbb{R}\mathbf{1}$ is the number of connected components of the equality graph of $(\overline{G}, \kappa(p))$ minus 1 (Joswig, Loho, 2016).

Vertices

- Every vertex of $R(p)/\mathbb{R}\mathbf{1}$ corresponds to a unique spanning subgraph of $\overline{G}$.
- For each $i \in V$, the i-th tropical vertex of $R(p)/\mathbb{R}\mathbf{1}$ corresponds to a shortest path tree of $(\overline{G}, \kappa(p))$ rooted at i (Joswig, Kulas, 2010).

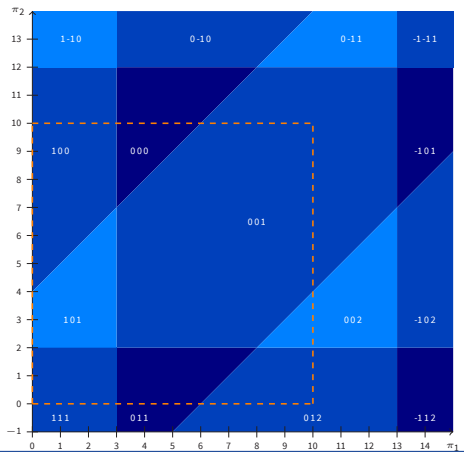
Relation to the Periodic Tension Polytope (aka $\mathrm{conv}(X)$)

- The map $m_p : \pi \mapsto (\pi_j - \pi_i + Tp_{ij})_{ij \in A}$ embeds $R(p)/\mathbb{R}\mathbf{1}$ into $\mathrm{conv}(X)$.
- $\mathrm{conv}(X) = \mathrm{conv}\{\mathrm{im}(m_p) \mid p \in \mathbb{Z}^A\}$.
- $\mathrm{im}(m_p)$ is the intersection of the affine space $\mathrm{im}(B^\top) + Tp$ with the LP relaxation polytope $X_{\mathrm{LP}} = \prod_{a \in A}[\ell_a, u_a]$, where B denotes the incidence matrix of G.

Tropical Neighborhood Search

Polytropes in the Limit Instance

Let $R(p)/\mathbb{R}\mathbf{1}$ be a polytrope. The offset p also defines a polytrope $R'(p)/\mathbb{R}\mathbf{1}$ a of the "limit" instance where $u := \ell + T$. The union of the polytropes is then no longer disjoint and covers all of $\mathbb{R}^V/\mathbb{R}\mathbf{1}$.



Tropical Neighborhood Search

Polytropes in the Limit Instance

Let $R(p)/\mathbb{R}\mathbf{1}$ be a polytrope. The offset p also defines a polytrope $R'(p)/\mathbb{R}\mathbf{1}$ a of the "limit" instance where $u := \ell + T$. The union of the polytropes is then no longer disjoint and covers all of $\mathbb{R}^V/\mathbb{R}\mathbf{1}$.

Observation

The $R'(p)$ induce a polyt(r)opal subdivision of $\mathbb{R}^V/\mathbb{R}\mathbf{1}$.

Tropical Neighborhood Search

Polytropes in the Limit Instance

Let $R(p)/\mathbb{R}\mathbf{1}$ be a polytrope. The offset p also defines a polytrope $R'(p)/\mathbb{R}\mathbf{1}$ a of the "limit" instance where $u := \ell + T$. The union of the polytropes is then no longer disjoint and covers all of $\mathbb{R}^V/\mathbb{R}\mathbf{1}$.

Observation

The $R'(p)$ induce a polyt(r)opal subdivision of $\mathbb{R}^V/\mathbb{R}\mathbf{1}$.

Neighbors

We call $R(p)/\mathbb{R}\mathbf{1}$ and $R(p')/\mathbb{R}\mathbf{1}$ *neighbors* if $R'(p)/\mathbb{R}\mathbf{1}$ and $R'(p')/\mathbb{R}\mathbf{1}$ intersect in a common facet.
If $R(p)/\mathbb{R}\mathbf{1}$ and $R(p')/\mathbb{R}\mathbf{1}$ are both neighbors, then $p = p' \pm e_a$ for some arc $a \in A$.

Tropical Neighborhood Search

Polytropes in the Limit Instance

Let $R(p)/\mathbb{R}\mathbf{1}$ be a polytrope. The offset p also defines a polytrope $R'(p)/\mathbb{R}\mathbf{1}$ a of the "limit" instance where $u := \ell + T$. The union of the polytropes is then no longer disjoint and covers all of $\mathbb{R}^V/\mathbb{R}\mathbf{1}$.

Observation

The $R'(p)$ induce a polyt(r)opal subdivision of $\mathbb{R}^V/\mathbb{R}\mathbf{1}$.

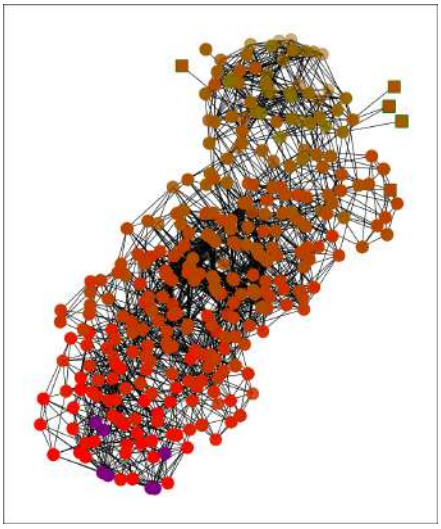
Neighbors

We call $R(p)/\mathbb{R}\mathbf{1}$ and $R(p')/\mathbb{R}\mathbf{1}$ *neighbors* if $R'(p)/\mathbb{R}\mathbf{1}$ and $R'(p')/\mathbb{R}\mathbf{1}$ intersect in a common facet.
If $R(p)/\mathbb{R}\mathbf{1}$ and $R(p')/\mathbb{R}\mathbf{1}$ are both neighbors, then $p = p' \pm e_a$ for some arc $a \in A$.

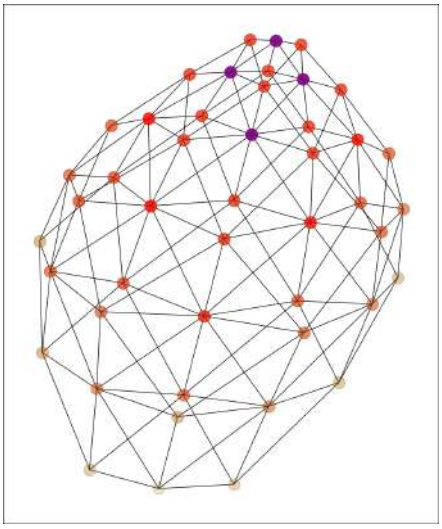
Tropical Neighborhood Search (Baseline)

Given a non-empty polytrope $R(p)/\mathbb{R}\mathbf{1}$, solve PESP on $R(p)/\mathbb{R}\mathbf{1}$ (this is a linear program, and dual to uncapacitated min cost flow). While there is an improving neighbor of $R(p)/\mathbb{R}\mathbf{1}$: Go to the best neighboring polytrope, and repeat.

Tropical Neighborhood Search



modulo network simplex search space
colored by objective value
squares are local non-global optima



tropical neighborhood search space
colored by objective value

Algorithm Tuning

- *work on the torus $\mathcal{T}$:*
  use $\mathbf{R}(z)$ instead of $R(p)/\mathbb{R}\mathbf{1}$

Algorithm Tuning

- *work on the torus $\mathcal{T}$:*
  use $\mathbf{R}(z)$ instead of $R(p)/\mathbb{R}\mathbf{1}$
- *LP formulation:*
  empirically: impact of up to a factor 14, the behavior seems to depend only on the instance and not on z

Algorithm Tuning

- *work on the torus $\mathcal{T}$:*
  use $\mathbf{R}(z)$ instead of $R(p)/\mathbb{R}\mathbf{1}$
- *LP formulation:*
  empirically: impact of up to a factor 14, the behavior seems to depend only on the instance and not on z
- *selection of neighbors:*
  do not explore all possible neighbors, only those the computed optimal vertex of $\mathbf{R}(z)$ is neighboring
  (best case: $2|A|$ vs. $|V| - 1$ neighbors, but we trade speed for quality)

Algorithm Tuning

- *work on the torus $\mathcal{T}$:*
  use $\mathbf{R}(z)$ instead of $R(p)/\mathbb{R}\mathbf{1}$
- *LP formulation:*
  empirically: impact of up to a factor 14, the behavior seems to depend only on the instance and not on z
- *selection of neighbors:*
  do not explore all possible neighbors, only those the computed optimal vertex of $\mathbf{R}(z)$ is neighboring
  (best case: $2|A|$ vs. $|V| - 1$ neighbors, but we trade speed for quality)
- *sorting of neighbors:*
  changing the order does affect the outcome, but unpredictably
  (we tried several strategies including “pseudocost branching”)

Algorithm Tuning

- *work on the torus $\mathcal{T}$:*
  use $\mathbf{R}(z)$ instead of $R(p)/\mathbb{R}\mathbf{1}$
- *LP formulation:*
  empirically: impact of up to a factor 14, the behavior seems to depend only on the instance and not on z
- *selection of neighbors:*
  do not explore all possible neighbors, only those the computed optimal vertex of $\mathbf{R}(z)$ is neighboring
  (best case: $2|A|$ vs. $|V| - 1$ neighbors, but we trade speed for quality)
- *sorting of neighbors:*
  changing the order does affect the outcome, but unpredictably
  (we tried several strategies including "pseudocost branching")
- *stopping criterion:*
  quality-first rule measured by relative improvement of the objective value
  (again trading speed for quality)

Algorithm Tuning

- *work on the torus $\mathcal{T}$:*
  use $\mathbf{R}(z)$ instead of $R(p)/\mathbb{R}\mathbf{1}$
- *LP formulation:*
  empirically: impact of up to a factor 14, the behavior seems to depend only on the instance and not on z
- *selection of neighbors:*
  do not explore all possible neighbors, only those the computed optimal vertex of $\mathbf{R}(z)$ is neighboring
  (best case: $2|A|$ vs. $|V| - 1$ neighbors, but we trade speed for quality)
- *sorting of neighbors:*
  changing the order does affect the outcome, but unpredictably
  (we tried several strategies including "pseudocost branching")
- *stopping criterion:*
  quality-first rule measured by relative improvement of the objective value
  (again trading speed for quality)
- *prevent cycling:*
  hashing visited $\mathbf{R}(z)$ showed only negligible effects

Performance of Tropical Neighborhood Search

Set-up

- 8 PESPlib instances
- 32 parameter configurations per instance
- 3 concurrency configurations for `ConcurrentPESP`
- 1 hour wall time, Intel i7-9700K CPU, 64 GB RAM

Performance of Tropical Neighborhood Search

Set-up

- 8 PESPlib instances
- 32 parameter configurations per instance
- 3 concurrency configurations for `ConcurrentPESP`
- 1 hour wall time, Intel i7-9700K CPU, 64 GB RAM

General Results

- Tropical Neighborhood Search can escape local optima
- slow in the beginning, but becomes important in the late game

Performance of Tropical Neighborhood Search

Set-up

- 8 PESPlib instances
- 32 parameter configurations per instance
- 3 concurrency configurations for `ConcurrentPESP`
- 1 hour wall time, Intel i7-9700K CPU, 64 GB RAM

General Results

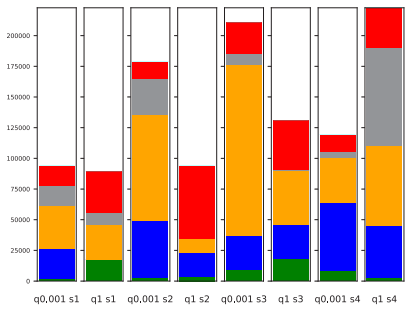
- Tropical Neighborhood Search can escape local optima
- slow in the beginning, but becomes important in the late game

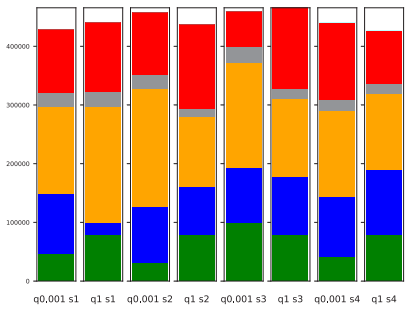
New PESPlib Incumbents

Instance	New Value	Old Value	Time (s)
BL3	6 675 098	6 999 313	25 732
R1L1v	42 591 141	42 667 746	9 110
R3L3	40 483 617	40 849 585	3 547
R4L4	36 703 391	36 728 402	11 122
R4L4v	61 968 380	64 327 217	3 625

found new incumbents for 5 out of 8 instances!

Tropical Neighborhood Search vs. Other Heuristics

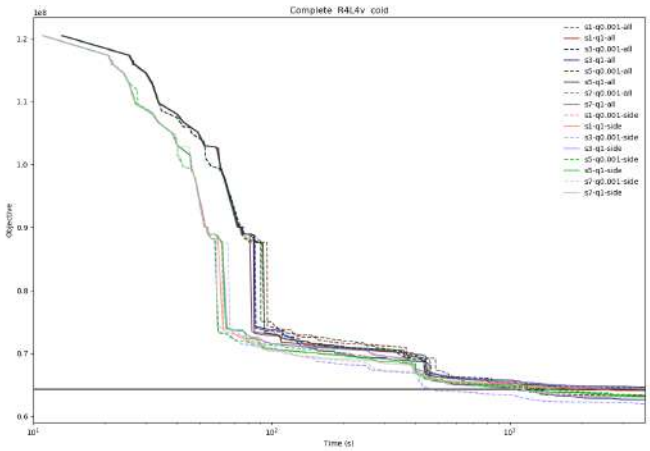




contribution of algorithms in `ConcurrentPESP` to overall improvement:
tropical neighborhood search, modulo network simplex, maximum cut, reflow, MIP

(8 parameter choices on R1L1)

Selection of Neighbors: All vs. Vertex-Tight



selecting only polytropes neighboring at an optimal vertex is an advantage in the beginning

Final Slide

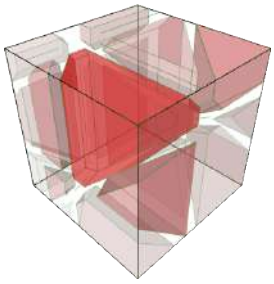
Conclusion

Tropical Neighborhood Search is a simple yet powerful geometry-inspired method for that adds new value to the zoo of periodic timetabling heuristics.

Final Slide

Conclusion

Tropical Neighborhood Search is a simple yet powerful geometry-inspired method for that adds new value to the zoo of periodic timetabling heuristics.



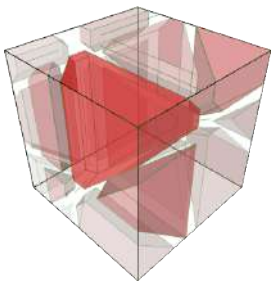
Further Geometric Questions

- Can we devise more heuristics from the polytropal decomposition of the timetable space?
- Can we extract dual information from the periodic timetabling torus?
- Can we exploit the duality relations between the Π- and *P*-spaces? ($\rightsquigarrow$ cycle offset zonotopes)

Final Slide

Conclusion

Tropical Neighborhood Search is a simple yet powerful geometry-inspired method for that adds new value to the zoo of periodic timetabling heuristics.

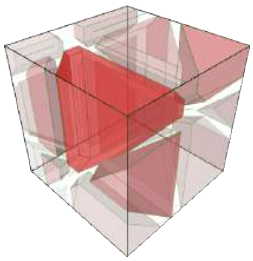


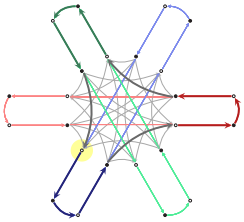
Further Geometric Questions

- Can we devise more heuristics from the polytropal decomposition of the timetable space?
- Can we extract dual information from the periodic timetabling torus?
- Can we exploit the duality relations between the Π- and *P*-spaces? ($\rightsquigarrow$ cycle offset zonotopes)

References

- E. Bortoletto, N. Lindner, B. Masing. *The Tropical and Zonotopal Geometry of Periodic Timetables.* `arXiv:2204.13501`
- E. Bortoletto, N. Lindner, B. Masing. *Tropical Neighbourhood Search: A New Heuristic for Periodic Timetabling.* ATMOS 2022 Best Paper Award.





On the geometry of periodic timetables in public transport

Niels Lindner, Enrico Bortoletto, Berenike Masing

MobilityLab
Department Network Optimization
Zuse Institute Berlin

6th RIKEN-IMI-ISM-NUS-ZIB-MODAL-NHR Workshop

September 21, 2022

Improving Data Quality in the Presence of Superhuman Complexity in Data Errors

Inci YÜKSEL-ERGÜN

Applied Algorithmic Intelligence Methods, Zuse Institute
Berlin, Germany
yueksel-erguen@zib.de

In our studies to analyze gas network systems, we study building public research data sets from incomplete data scattered around various data sources. These data sources may not be consistent with each other or accurate. Thus, during these studies, we use our domain-specific mathematical modeling know-how to eliminate the data errors by filling missing data, or fixing inconsistencies. However, when working with the resulting highly-connected data, we encountered several cases where our analysis detected data errors that were too complex for humans to understand. Examples are irreducible infeasible subsystems (IIS) of large mixed-integer programs (MIP) or bottlenecks in the pressure-coupled pipeline network that is non-linear. While detecting these errors is a significant achievement, removing such errors is extremely difficult. Hence, quantifying the data quality is also a key enabler in this study to tell whether the data is of sufficient quality for the aimed analysis. We present our studies on data quality improvement in the presence of superhuman complexity in data errors, and explain the challenges. We report our results on the German high-pressure gas transport network data set.

References

[1] ENTSOG. Transmission Capacity Map. Retrieved from https://www.entsog.eu/maps#transmission-capacity-map-2021. Accessed on 31.10.2022

[2] Inci Yueksel-Erguen, Dieter Most, Lothar Wyrwoll, Carlo Schmitt, Janina Zittel. Modeling the transition of the multimodal pan-European energy system including an integrated analysis of electricity and gas transport. Technical Report 22-17. Zuse Institute Berlin, Takustr. 7, 14195 Berlin, 2022. Available online < https://opus4.kobv.de/opus4-zib/frontdoor/index/index/docId/8777>

[3] Friedrich Kunz, Jens Weibezahn, Philip Hauser, Sina Heidari, Wolf-Peter Schill, Björn Felten, Mario Kendziorski, Matthias Zech, Jan Zepter, Christian von Hirschhausen, Dominik Möst, Christoph Weber. Reference Data Set: Electricity, Heat, and Gas Sector Data for Modeling the German System (Version 1.0.0), 2017. https://doi.org/10.5281/zenodo.1044463.

[4] Nicola Askham, Denise Cook, Martin Doyle, Helen Fereday, Mike Gibson, Ulrich Landbeck, Rob Lee, Chris Maynard, Gary Palmer, Julian Schwarzenbach. The six primary dimensions for data quality assessment. Technical Report, 2013, DAMA United Kingdom.

[5] Inci Yueksel-Erguen, Janina Zittel, Ying Wang, Felix Hennings, Thorsten Koch. Lessons learned from gas network data preprocessing. Technical Report 20-13. Zuse Institute Berlin, Takustr. 7, 14195 Berlin, 2020. Available online https://opus4.kobv.de/opus4-zib/frontdoor/index/index/docId/7826

[6] Thorsten Koch, Tobias Achterberg, Erling Andersen, Oliver Bastert, Timo Berthold, Robert E. Bixby, Emilie Danna, Gerald Gamrath, Ambros M. Gleixner, Stefan Heinz, Andrea Lodi, Hans

Mittelmann, Ted Ralphs, Domenico Salvagnin, Daniel E. Steffy, Kati Wolter MIPLIB 2010. Math. Prog. Comp. 3, 103 (2011). https://doi.org/10.1007/s12532-011-0025-9

SPONSORED BY THE Federal Ministry of Education and Research

MODAL FORSCHUNGSCAMPUS

FORSCHUNGS CAMPUS öffentlich-private Partnerschaft für Innovationen

Improving data quality in the presence of superhuman complexity in data errors

Inci Yüksel-Ergün, Thorsten Koch, Felix Hennings, Janina Zittel

21.09.2022

The 6th RIKEN-IMI-ISM-NUS-ZIB-MODAL-NHR Workshop
on Advances in Classical and Quantum Algorithms for
Optimization and Machine Learning

Chair of
Software and Algorithms for
Discrete Optimization

Department:
Applied Algorithmic
Intelligence Methods

ZIB ZUSE INSTITUTE BERLIN

Introduction

- Highly-connected data in industrial applications
  - provided by industrial partners
  - consolidated from public data sources
  - generated using mathematical models
- Data errors too complex for humans to understand detected by analysis tools, i.e.,
  - irreducible infeasible subsystems (IIS) of large mixed-integer programs
  - bottlenecks in the pressure-coupled pipeline network that is non-linear
- Detection and correction of such errors are extremely difficult

Supply Infrastructures

- Transport power, gas, water, data, goods from suppliers to customers
- Consist of complex networks
- Digitization of planning and operation of such networks is essential for vital problems, i.e.,
  - Security of supply of energy, supply chain management, energy transition, decarbonization, etc.

High quality data → Reliable analysis results

High quality data is incredibly costly to obtain both in commercial and public applications, since supply infrastructures

- have complex network structures
- were mostly built before digitization age
- consist of layered and connected structures that may be operated by different parties
- include complex facilities with intricate structures that can be handled by detailed mathematical models

Example: The European Gas Transport Network

[1] ENTSOG. Transmission Capacity Map. Retrieved from https://www.entsog.eu/maps#transmission-capacity-map-2021. Accessed on 31.10.2022

European gas transport network in numbers:

- 42 member, 10 associated partner, 2 observer TSOs
- > 200 interconnection points, > 170 storages
- ≈ 200,000 km transmission pipelines (EU+UK)

Example: The European and Germany Gas Transport Network

[1] ENTSOG. Transmission Capacity Map. Retrieved from https://www.entsog.eu/maps#transmission-capacity-map-2021. Accessed on 31.10.2022

German gas transport network in numbers:

- 16TSOs
- 44interconnection points, 60storages
- ≈ 31K km transmission pipelines (in total >500K km)

Example: Gas Transport Network Data

European Gas Transport Network: Interconnections

[1] ENTSOG. Transmission Capacity Map. Retrieved from https://www.entsog.eu/maps#transmission-capacity-map-2021. Accessed on 31.10.2022

[1] ENTSOG. Transmission Capacity Map. Retrieved from https://www.entsog.eu/maps#transmission-capacity-map-2021. Accessed on 31.10.2022

Germany Gas Transport Network: Topology & Complex facilities

- ≈70 entry & ≈900 exit points
- ≈1650 inner nodes
- ≈1770 pipes
- ≈95 control valves
- 58 compressor stations
  - 129 compressors & drivers
  - 200 valves

[2] Yueksel-Erguen et al. Modelling the transition of the multimodal pan-European energy system including an integrated analysis of electricity and gas transport. Technical Report 22-17. Zuse Institute Berlin, Tokustr. 7, 14195 Berlin, 2022.

[3] Kunz et al. Reference Data Set: Electricity, Heat, and Gas Sector Data for Modeling the German System (Version 1.0.0), 2017. https://doi.org/10.5281/zenodo.1044463.

Gas Data Files: Technical properties

Network topology data: .net file (>34K lines; >28K data attributes)

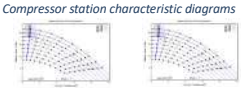
```
<source id="N1" x="-100" y="500">
  <height unit="m" value="200.0"/>
  <pressureMin unit="bar" value="30.0"/>
  <pressureMax unit="bar" value="70.0"/>
  <flowMin unit="1000m_cube_per_hour" value="55.0"/>
  <flowMax unit="1000m_cube_per_hour" value="750.0"/>
  <gasTemperature unit="Celsius" value="10.0"/>
  <calorificValue value="37" unit="MJ_per_m_cube"/>
  <normDensity value="0.785" unit="kg_per_m_cube"/>
  <coefficient-A-heatCapacity value="32.237"/>
  <coefficient-B-heatCapacity value="-6.01"/>
  <coefficient-C-heatCapacity value="0"/>
  <molarMass value="19.5" unit="kg_per_kmol"/>
  <pseudocriticalPressure value="44.5" unit="bar"/>
  <pseudocriticalTemperature value="190" unit="K"/>
</source>
```

```
<pipe from="N1" id="P12" to="N2">
  <flowMin unit="1000m_cube_per_hour" value="0.0"/>
  <flowMax unit="1000m_cube_per_hour" value="73500"/>
  <length unit="km" value="50"/>
  <diameter unit="m" value="1.1"/>
  <roughness unit="mm" value="0.01"/>
  <heatTransferCoefficient unit="W_per_m_square_per_K" value="2.0"/>
</pipe>
```

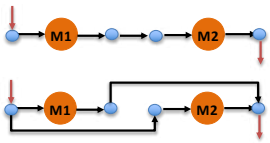
Compressor station data: .cs file (>54K lines)

```
<compressorStation id="CS3">
  <compressors>
    <turboCompressor driver="T_CS3_M1" id="T_CS3_M3">
    <turboCompressor driver="T_CS3_M1" id="T_CS3_M3">
  </compressors>
  <drivers>
    <gasTurbine id="P_CS3_M1">
    <gasTurbine id="P_CS3_M2">
  </drivers>
  <configurations>

    <configuration confID="1" noOfSerialStages="2">
      <stage stageNo="1" noOfParallelUnits="3">
      <stage stageNo="2" noOfParallelUnits="3">
    </configuration>

    <configuration confID="3" noOfSerialStages="1">
      <stage stageNo="1" noOfParallelUnits="1">
    </configuration>

    <configuration confID="2" noOfSerialStages="1">
      <stage stageNo="1" noOfParallelUnits="2">
    </configuration>

    <configuration confID="4" noOfSerialStages="1">
      <stage stageNo="1" noOfParallelUnits="3">
    </configuration>

  </configurations>
</compressorStation>
```

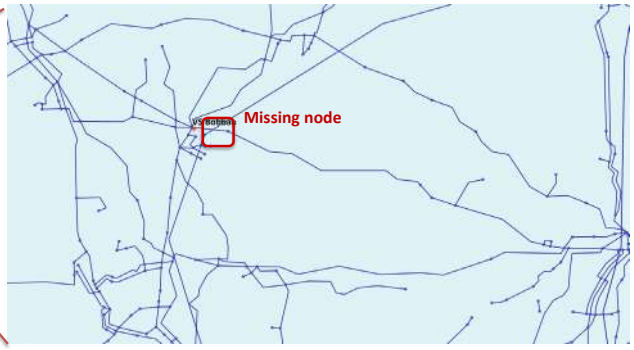
Compressor station characteristic diagrams



Alternative compressor machine configurations



Gas Transport Network – Data Error Example 1



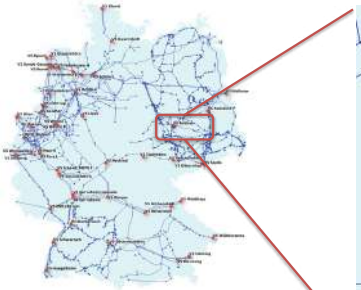
[2] Yueksel-Erguen et al. Modeling the transition of the multimodal pan-European energy system including an integrated analysis of electricity and gas transport. Technical Report 22-17. Zuse Institute Berlin, Takustr. 7, 14195 Berlin, 2022.

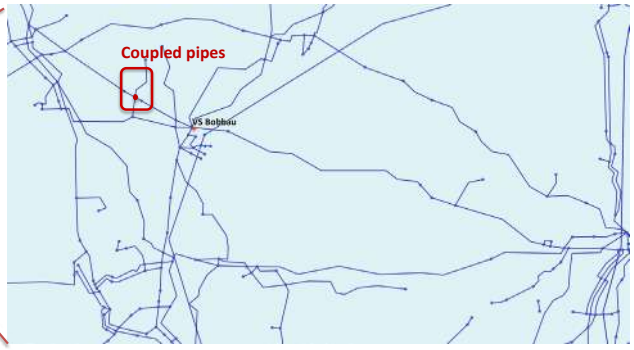
[3] Kunz et al. Reference Data Set: Electricity, Heat, and Gas Sector Data for Modeling the German System (Version 1.0.0), 2017. https://doi.org/10.5281/zenodo.1044463.

.net file should be corrected by:

- adding the missing node
- updating end node of existing pipes
- adding two pipes between the end nodes of the existing pipes and the added node

Gas Transport Network – Data Error Example 2





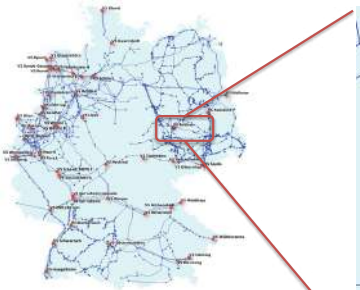
[2] Yueksel-Erguen et al. Modeling the transition of the multimodal pan-European energy system including an integrated analysis of electricity and gas transport. Technical Report 22-17. Zuse Institute Berlin, Takustr. 7, 14195 Berlin, 2022.

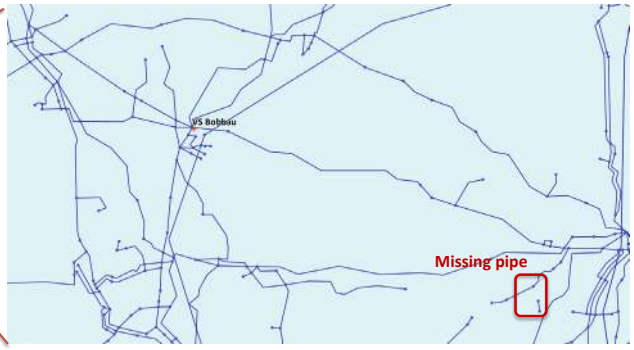
[3] Kunz et al. Reference Data Set: Electricity, Heat, and Gas Sector Data for Modeling the German System (Version 1.0.0), 2017. https://doi.org/10.5281/zenodo.1044463.

.net file should be corrected by:

- deleting the node
- deleting two of the existing pipes
- updating end nodes of the two pipes

Gas Transport Network – Data Error Example 3





[2] Yueksel-Erguen et al. Modeling the transition of the multimodal pan-European energy system including an integrated analysis of electricity and gas transport. Technical Report 22-17. Zuse Institute Berlin, Takustr. 7, 14195 Berlin, 2022.

[3] Kunz et al. Reference Data Set: Electricity, Heat, and Gas Sector Data for Modeling the German System (Version 1.0.0), 2017. https://doi.org/10.5281/zenodo.1044463.

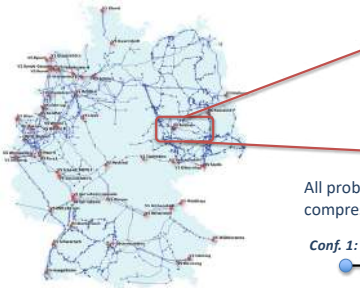
.net file should be corrected:

- adding missing pipe

or,

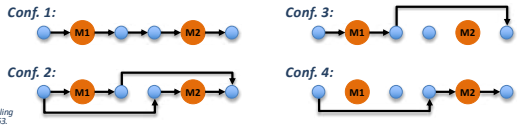
- deleting the disconnected part

Gas Transport Network – Data Error Example 4



Compressor Station Configuration Definition

All probable configurations of compressors in a compressor machine with a two compressor machines:



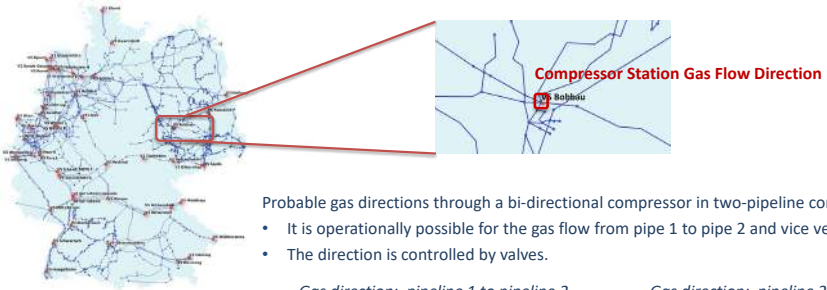
[2] Yueksel-Erguen et al. Modeling the transition of the multimodal pan-European energy system including an integrated analysis of electricity and gas transport. Technical Report 22-17. Zuse Institute Berlin, Takustr. 7, 14195 Berlin, 2022.

[3] Kunz et al. Reference Data Set: Electricity, Heat, and Gas Sector Data for Modeling the German System (Version 1.0.0), 2017. https://doi.org/10.5281/zenodo.1044463.

.cs file should be corrected by:

- updating the probable configuration(s)

Gas Transport Network – Data Error Example 5

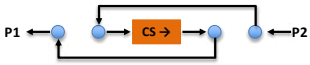


Probable gas directions through a bi-directional compressor in two-pipeline connection case:

- It is operationally possible for the gas flow from pipe 1 to pipe 2 and vice versa.
- The direction is controlled by valves.

Gas direction: pipeline 1 to pipeline 2

Gas direction: pipeline 2 to pipeline 2



.net file should be corrected by:

- updating valve(s)
- updating end nodes of the pipe(s)

[2] Yueksel-Erguen et al. Modeling the transition of the multimodal pan-European energy system including an integrated analysis of electricity and gas transport. Technical Report 22-17. Zuse Institute Berlin, Takustr. 7, 14195 Berlin, 2022.

[3] Kunz et al. Reference Data Set: Electricity, Heat, and Gas Sector Data for Modeling the German System (Version 1.0.0), 2017. https://doi.org/10.5281/zenodo.1044463.

Modeling Infrastructure Networks in Industrial Applications

Infrastructure network data consists of

- Tabular data: technical bounds, physical properties, etc.
- Spatial data: network topology
- Complex components: often induce non-linear relations of data elements

It is challenging to detect and correct data errors:

- Highly connected
- Requires subject matter expertise to explain errors
- Optimization is actively exploiting errors

Dimensions of Data Quality

Accuracy is the ability to reflect reality, i.e., to be able to correctly describe real world objects

Completeness is existence of all required data in the data set, comprehensiveness

Validity is conformance to predetermined type, range and format

Consistency means conflict-free data, i.e., attributes of data entities addressing the same physical attribute should be consistent

Uniqueness means representing each real world object in the data set exactly once

Timeliness shows whether data is up-to-date data depending on the analysis context

[4] Askham et al. The six primary dimensions for data quality assessment. Technical Report, 2013, DAMA UK.

Data Errors: Non-conformances to data quality

Accuracy is the ability to reflect reality, i.e., to be able to correctly describe real world objects

- Insufficient detail
  → cumulative capacities
- Wrong modeling assumptions
  → flow capacity computation
- Data preprocessing/integration issues
  → erroneous node coupling

Completeness is existence of all required data in the data set, comprehensiveness

- Missing attributes → missing pipes, valves, etc.
- Missing entities → missing pipe maximum pressure values, node height values, etc.

Validity is conformance to predetermined type, range and format

- Typos in names/non-conformance to bounds (mostly data entry related issues)

Consistency means conflict-free data, i.e., attributes of data entities addressing the same physical attribute should be consistent

- Conflicts in the data sets
  → node pressure vs. pipe pressure bounds

(may be induced by data set integration or data preprocessing assumptions)

Uniqueness means representing each real world object in the data set exactly once

- Repeated use of object names

Timeliness shows whether data is up-to-date data depending on the analysis context

- Untimely data → wrongly forecasted infrastructure related data for future extensions due to limited information

[4] Askham et al. The six primary dimensions for data quality assessment. Technical Report, 2013, DAMA UK.

Data Error Detection and Correction

Uniqueness
Validity

*Easier to detect and correct non-conformance using automated systems like **schema validation,** i.e., GasLib format (www.gaslib.zib.de)*

Accuracy
Completeness
Consistency
Timeliness

***Manual** detection or correction is often not possible*

*Correction of non-conformance often requires **subject matter expertise** and **human interaction***

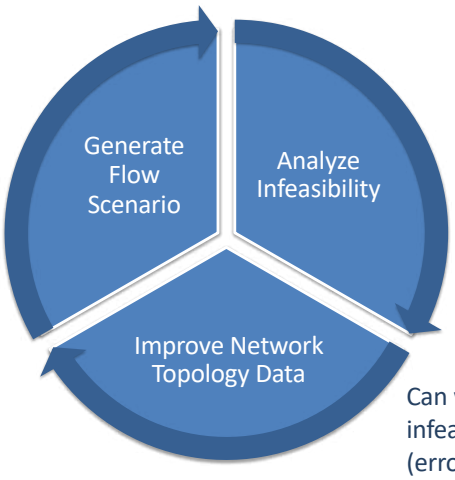
Eliminating Data Errors

Data Improvement Method		Related Data Quality Dimension	Error
Schema validation		Uniqueness Validity	Format mismatch Nonconformance to bounds
Data augmentation		Completeness Accuracy	Insufficient detail Missing entries/attributes
Data generation		Completeness Accuracy	Insufficient detail Missing entries/attributes
Error diagnosis and correction	Consistency check heuristics	Consistency Accuracy	Conflicts in the data set Wrong modeling assumptions Data preprocessing errors
	Extensive scenario analysis	Accuracy Consistency Timeliness	Conflicts in the data set Wrong modeling assumptions Data preprocessing errors Untimely data

Examples for consistency check heuristics: [5] Inci Yueksel-Erguen, J. Zittel, Y. Wang, F. Hennings, T. Koch. Lessons learned from gas network data preprocessing. Technical Report 20-13. Zuse Institute Berlin, Takustr. 7, 14195 Berlin: ZIB, 2020.

Extensive Scenario Analysis

Generate practically valid scenarios using historical flow data



Utilize mathematical modeling methods to detect infeasibility

Can we relate infeasibility to data (errors)?

An Example Analysis Set-up – Gas Network Data

Related Data Sets
- ENTSOG IP data set
- GIE Storage data set

M1: High-level Pan-European Gas Network Model

Cumulative gas in-/out- flow of Germany based on IPs

Historical flow data

Source: ENTSOG Security of Supply Report Scenarios

Related Data Sets
- DE network topology data set
- Node pressure data from TSOs

M2: German Gas Transport Network Model (Linear)

- Estimated pipeline capacities
- ENTSOG IP- DE network associations

Gas in-/out- flow of Germany based on physical entry/exit nodes

Resulting data
- DE gas network
- CS data

M3: German Gas Transport Network Model (Nonlinear)

- Parameters related to gas properties

Details for the models: [2] I. Yueksel-Erguen, D. Most, L. Wyrwoll, C. Schmitt, J. Zittel. Modeling the transition of the multimodal pan-European energy system including an integrated analysis of electricity and gas transport. Technical Report 22-17. Zuse Institute Berlin, Takustr. 7, 14195 Berlin, 2022.

Mathematical Modeling Methods for Infeasibility Analysis

Slack Formulations

- Different aspects of the formulations can be relaxed with slack variables
- The objective is to minimize the deviation from the original model → zero objective function
- The smallest distance from the feasibility

(Minimum) Irreducible Infeasible Subsystems (IIS)

- Isolates the infeasibility by variables and constraints
- Not an explicit reason why a IIS is infeasible
- Long computation times for large-scale MIPs
- A trivial IIS is not informative

ns1158817 from MIPLIB 2010*	Constraints	Variables	NZ
Problem	68,455	1,804,022	2,842,044
IIS	2,003	6,002	12,002

[6] Koch et al. MIPLIB 2010. Math. Prog. Comp. 3, 103 (2011). https://doi.org/10.1007/s12532-011-0025-9

Scenario Analysis – Infeasibility Analysis Method

S1. Initiate scenario analysis:
relax all non-linear constraints in the slack formulation
if there is at least one scenario in the scenario set S, select a scenario s from the scenario set and go to S2, else go to S5

S2. Solve the mathematical model
If feasible save the solution, delete the scenario s from the scenario set S and go to S1; else go to S3

S3. Solve the slack formulation
if infeasible go to S4
if feasible with non-zero slack: correct the scenario and turn to S3
if feasible with a zero objective function value, tighten one set of the nonlinear constraints in the slack formulation if available and turn to S3, else save the solution and scenario correction, delete the scenario s from the scenario set S and go to S1;

S4. Find the minIIS using the LP minIIS model
if feasible, save the minIIS solution for the scenario, rescale the scenario by 0.95, and go to S4
if infeasible go to S1.

S5. Analyze the saved minIISs
to detect the frequency of existence of network components and constraint types in the minIIS and correct data set, and go to S1.

Automated Infeasibility Analysis Method

S0. Initiate data set:

restore the scenario set: S:=S0; reset the data rating DR:=0

S1. Initiate scenario analysis:

relax all non-linear constraints in the slack formulation

if there is at least one scenario in the scenario set S, select a scenario s from the scenario set and go to S2, else go to S5

S2. Solve the mathematical model

If feasible save the solution and the scenario neighborhood scale association with the scenario, delete the scenario s from the scenario set S and go to S1; else go to S3

S3. Solve the slack formulation

if infeasible go to S4

if feasible with non-zero slack: correct the scenario and turn to S3

if feasible with a zero objective function value, tighten one set of the nonlinear constraints in the slack formulation if available and turn to S3, else update the scenario neighborhood scale and go to S2

S4. Find the minIIS using the LP minIIS model

if feasible, save the minIIS solution for the scenario, rescale the scenario by 0.95, and go to S4

if infeasible go to S1.

S5. Analyze the saved minIISs

to detect the frequency of existence of network components and constraint types in the minIIS, correct data set,

S6. Measure the data quality

If data quality is not sufficient, got to S1, else terminate.

Data Quality Rating

Quality of data can be measured by its **ability to represent the addressed entities**

We need a data rating measure to **facilitate the automated improvement** by enabling us to

- understand whether the improved data set is of **sufficient quality level for the aimed analysis**
- **compare** the performance of alternative improvements

.. and also lead us/the search to **the potential errors and error sources…**

Example: Number of scenarios generated from the historical data that the data set finds a feasible routing

- Not informative enough
- Especially in the very beginning of the improvement process – we may get 0 for all scenario sets

Data Quality Rating of Infrastructure Network Data - 1

An infrastructure network is an engineered system that is operational:

- Designed to meet **certain quality of service requirements given operational requirements**
- **Robust** against **uncontrollable** factors in the working environment

So, what does it mean from data perspective?

- The network is designed to find feasible routings to scenarios within **the operational concept**
- The network is designed to eliminate **systematic errors**
- The network has **random errors** that cannot be totally eliminated by the design process, so the network should provide **alternative routing solutions** for changes in the scenarios
- The amount of the change depends on (robustness of) the network

If a scenario gives a feasible result with the data set,
then a set of scenarios in its **neighborhood** is expected to be feasible.

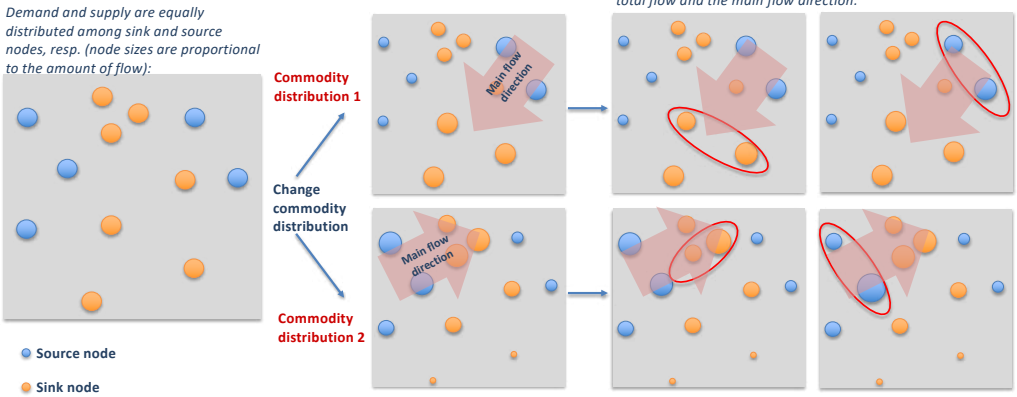
Data Quality Rating of Infrastructure Network Data - 2

Operational scenarios of an infrastructure network have two main measurable characteristics:

- The **amount** of commodity that can be routed
- The **distribution** of the commodity in the network

Demand and supply are equally distributed among sink and source nodes, resp. (node sizes are proportional to the amount of flow):

Small changes in demand and supply without changing the total flow and the main flow direction:

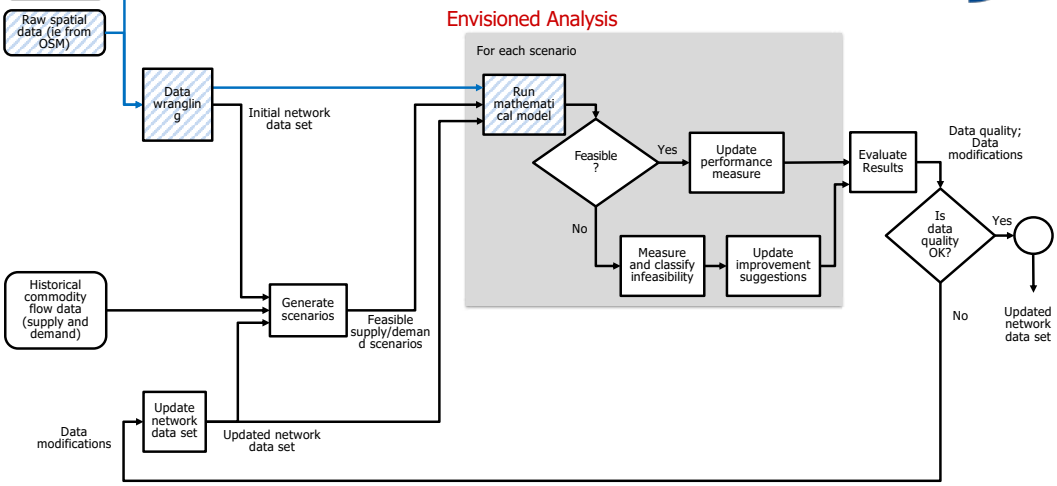


Future Outlook: Automated Data Improvement Using Extensive Scenario Analysis 1

Traditional analysis

Future Outlook: Automated Data Improvement Using Extensive Scenario Analysis 2

Envisioned Analysis



References

[1] ENTSOG. Transmission Capacity Map. Retrieved from https://www.entsog.eu/maps#transmission-capacity-map-2021. Accessed on 31.10.2022

[2] Inci Yueksel-Erguen, Dieter Most, Lothar Wyrwoll, Carlo Schmitt, Janina Zittel. Modeling the transition of the multimodal pan-European energy system including an integrated analysis of electricity and gas transport. Technical Report 22-17. Zuse Institute Berlin, Takustr. 7, 14195 Berlin, 2022. Available online < https://opus4.kobv.de/opus4-zib/frontdoor/index/index/docId/8777>

[3] Friedrich Kunz, Jens Weibezahn, Philip Hauser, Sina Heidari, Wolf-Peter Schill, Björn Felten, Mario Kendziorski, Matthias Zech, Jan Zepter, Christian von Hirschhausen, Dominik Möst, Christoph Weber. Reference Data Set: Electricity, Heat, and Gas Sector Data for Modeling the German System (Version 1.0.0), 2017. https://doi.org/10.5281/zenodo.1044463.

[4] Nicola Askham, Denise Cook, Martin Doyle, Helen Fereday, Mike Gibson, Ulrich Landbeck, Rob Lee, Chris Maynard, Gary Palmer, Julian Schwarzenbach. The six primary dimensions for data quality assessment. Technical Report, 2013, DAMA United Kingdom.

[5] Inci Yueksel-Erguen, Janina Zittel, Ying Wang, Felix Hennings, Thorsten Koch. Lessons learned from gas network data preprocessing. Technical Report 20-13. Zuse Institute Berlin, Takustr. 7, 14195 Berlin, 2020. Available online https://opus4.kobv.de/opus4-zib/frontdoor/index/index/docId/7826

[6] Thorsten Koch, Tobias Achterberg, Erling Andersen, Oliver Bastert, Timo Berthold, Robert E. Bixby, Emilie Danna, Gerald Gamrath, Ambros M. Gleixner, Stefan Heinz, Andrea Lodi, Hans Mittelmann, Ted Ralphs, Domenico Salvagnin, Daniel E. Steffy, Kati Wolter MIPLIB 2010. Math. Prog. Comp. 3, 103 (2011). https://doi.org/10.1007/s12532-011-0025-9

Thank you for listening!

yueksel-erguen@zib.de

Optimal discrete pipe sizing for tree-shaped CO_2 networks

Jaap Pedersen, Thi Thai Le, Thorsten Koch, Janina Zittel

Zuse Institute Berlin, Berlin, Germany

Technische Universität Berlin, Berlin, Germany

pedersen@zib.de

Many energy-intensive industries, like the steel industry, plan to switch to renewable energy sources. Other industries, such as the cement industry, have to rely on carbon capture utilization and storage (CCUS) technologies to reduce their production processes' inevitable carbon dioxide (CO_2) emissions. However, a new transport infrastructure needs to be established to connect the point of capture and the point of storage or utilization. Given a tree-shaped network transporting captured CO_2 from multiple sources to a single sink, we investigate how to select optimal pipeline diameters from a discrete set of diameters. The general problem of optimizing arc capacities in potential-based fluid networks is already a challenging mixed-integer nonlinear optimization problem. Adding the highly sensitive and nonlinear behavior of CO_2 regarding temperature and pressure changes the problem becomes even harder. We propose an iterative algorithm that splits the problem into two parts: a) the pipe-sizing problem under a fixed supply scenario and temperature distribution and b) the thermophysical modeling, including mixing effects, the Joule-Thomson effect, and the heat exchange with the surrounding environment. We show the effectiveness of our approach by applying our algorithm to a real-world network planning problem for a CO_2 network in Germany.

Optimal discrete pipe sizing for tree-shaped CO_2 networks

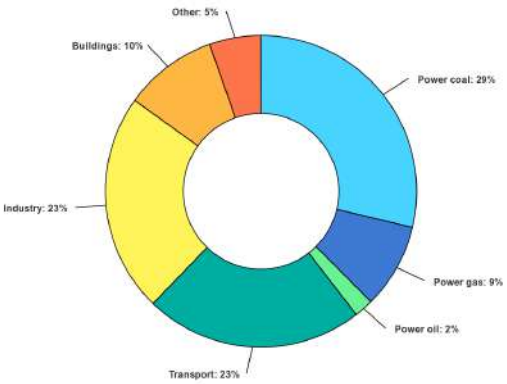
Jaap Pedersen,
T. Le, J. Zittel and T. Koch

21st of September 2022

6th RIKEN-IMI-ISM-NUS-ZIB-MODAL-NHR Workshop on
Advances in Classical and Quantum Algorithms for
Optimization and Machine Learning

Acknowledgement: This work has been conducted in the Research Campus MODAL funded by the German Federal Ministry of Education and Research (BMBF) (fund numbers 05M14ZAM, 05M20ZBM).

Motivation - Net-zero by 2050



Global greenhouse gas emissions (GHG) in 2018 [IEA, 2020]

- Decarbonization of industry, e.g, by shifting towards renewables
- 8 % of global GHG from cement industry
- CO_2 emissions from chemical process itself
- Carbon Capture and Storage (CCS) or Utilization (CCU)
- New infrastructure to transport CO_2 from point of capture to storage/utilization

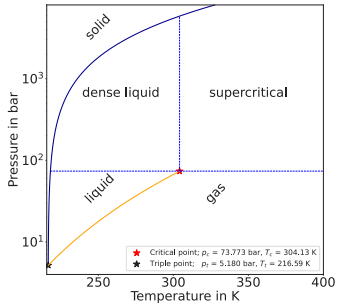
How to transport CO_2? - Goal

- CO_2 is transported in liquid or supercritical state
- Supercritical: State in which the liquid and gaseous phases cannot be distinguished
- Pipeline networks are most cost efficient
- Network planning involves finding the cost-optimal pipeline diameters

Goal

Determine the cost-optimal pipeline diameters from a discrete set of diameters

- in a tree-shaped CO_2 network
- with multiple sources and a single sink
- for a given supply scenario
- transport CO_2 in liquid or supercritical state
  $\Longrightarrow$ no transition into gaseous phase

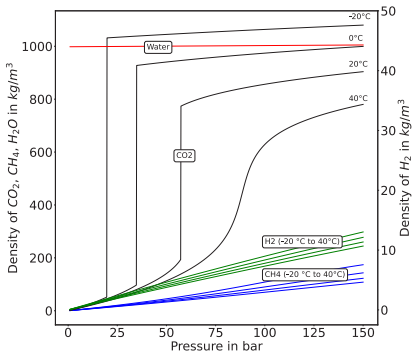


Own figure, data from [Bell et al., 2014]

Background

- Problem of finding optimal pipeline diameters is a common problem in real-world applications, e.g., in water, natural gas, and hydrogen networks
  [D'Ambrosio et al., 2015, Lenz and Becker, 2022, Robinius et al., 2019]
- Discrete decision variables w.r.t. diameter size of pipelines
- Non-convex physics describing flow in pipelines
- Physical properties of CO_2 are sensitive against changes in pressure and temperature

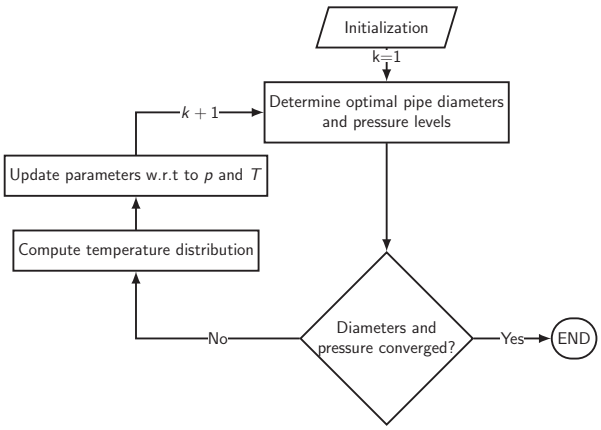
$\Longrightarrow$ Mixed Integer NonLinear Program; hard to solve



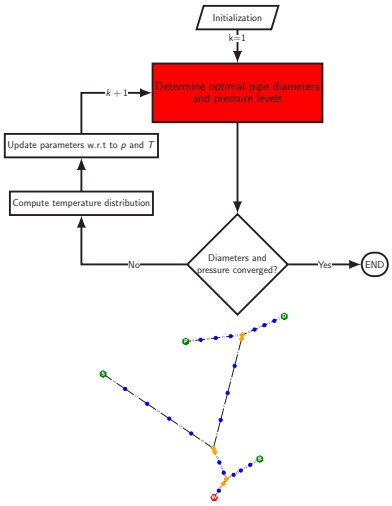
Own figure, data from [Bell et al., 2014]

Solution approach

- Iterative algorithm
- Decouple finding optimal pipe diameters and thermophysical modeling
- Exploit the benefits of tree structure to reduce complexity
- In each iteration update parameters w.r.t to pressure p and temperature T
- Stop algorithm if diameters and pressure levels converge



Determine optimal pipe diameters - Notation



$G = (V, A)$	Directed tree with set of vertices V and arcs A; arcs orientated from leafs to root
V^+, V^-, V^0	Set of entry, exit and inner nodes
A^{pi}, A^{pu}	Set of pipes and pumps
D_a	Set of pipe diameters for $a \in A^{\text{pi}}$
$b_v \in \mathbb{R}$	Inflow/Outflow of vertex $v \in V$
$p_v \in [\underline{p}_v, \overline{p}_v]$	Pressure of vertex $v \in V$
$q_a \in \mathbb{R}_{\geq 0}$	Flow over arc $a \in A$
$d_a \in D_a$	Diameter of pipe $a \in A^{\text{pi}}$
$x_{a,d} \in \{0, 1\}$	Choice of $d_a \in D_a$ for $a \in A^{\text{pi}}$

Determine optimal pipe diameters - Notation

Incoming and outgoing arcs:

$$\delta^-(v) = \{a \in A \,|\, a = (u, v)\}$$
$$\delta^+(v) = \{a \in A \,|\, a = (v, u)\}$$

Intermediate nodes:

$$V^m := \{v \in V \,|\, |\delta^-(v)| = |\delta^+(v)| = 1 \;\wedge\; \delta^-(v) \cup \delta^+(v) \in A^{\text{pi}} \wedge$$
$$|D_{a=(u,v)}| > 1 \wedge |D_{a=(v,w)}| > 1\}.$$

Note: Node $v \notin V^m$ if junction node, tail or head node of pump, tail or head node of pipeline with a fixed diameter

Determine optimal pipe diameters - Network elements

- **Pipelines** are main type of element to transport CO_2
- In general, fluid flows from higher pressure to lower pressure
- Pressure change in a pipeline

$$p_v - p_u = \underbrace{\phi_a(d_a, q_a)\, q_a |q_a|}_{\text{friction term}} - \underbrace{(H_v^0 - H_u^0)\rho g}_{\text{elevation difference}} \qquad \forall a = (v, u) \in A^{\text{pi}}$$

- Friction loss coefficient $\phi_a(d_a, q_a)$ is highly nonlinear
- **Pumps** can be used to increase pressure of fluid

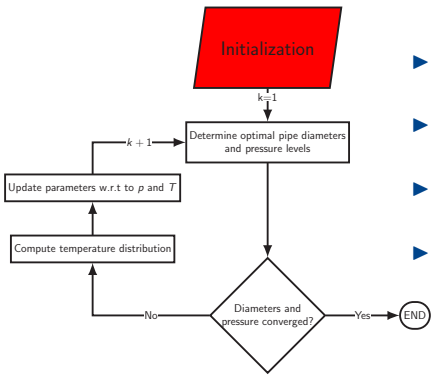
$$p_u \geq p_v \qquad \forall a = (v, u) \in A^{\text{pu}}$$

H_v^0: height of v; ρ: density of fluid, g: gravitational constant

Determine optimal pipe diameters - Model

$$
\begin{aligned}
\min_{x,q,p} \sum_{a\in A^{\mathrm{pi}}} \sum_{d\in D_a} c_{a,d} x_{a,d} & \\
\text{s.t.} \sum_{a\in\delta^+(v)} q_a - \sum_{a\in\delta^-(v)} q_a &= b_v && \forall v \in V, \\
p_v - p_u &= \sum_{d\in D_a} x_{a,d}\phi_a(d,q_a)\, q_a|q_a| - (H_v^0 - H_u^0)\rho g && \forall a=(v,u)\in A^{\mathrm{pi}}, \\
p_u &\geq p_v && \forall a=(v,u)\in A^{\mathrm{pu}}, \\
\underline{p}_v &\leq p_v \leq \overline{p}_v && \forall v \in V, \\
\sum_{d\in D_a} x_{a,d} &= 1 && \forall a \in A^{\mathrm{pi}}, \\
\sum_{d\in D_{(u,v)}} x_{(u,v),d}\, d &= \sum_{d\in D_{(v,w)}} x_{(v,w),d}\, d && \forall v \in V^m, \\
x_{a,d} &\in \{0,1\} && \forall a \in A^{\mathrm{pi}}, \forall d \in D_a.
\end{aligned}
$$

Initialization - Reduce complexity



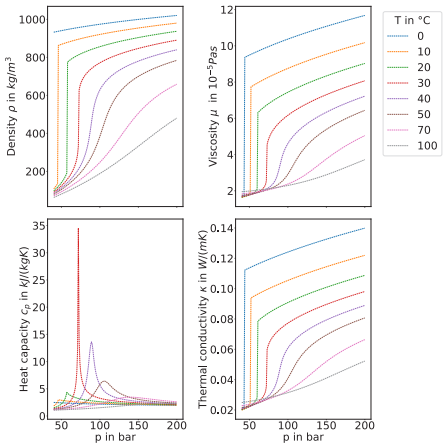
- Reduce complexity by precomputing unique flow values and friction loss coefficients
- We have an in-tree graph rooted at a single exit node and a given inflow scenario, i.e., all inflow are known
- Recursively determine arc flow values starting at leaf nodes towards the exit node
- For each pipeline $a \in A$ and each possible diameter $d \in D_a$ determine friction loss coefficient $\phi_a(d, q_a)$

Assume constant physical parameters, e.g., $\rho = \text{const}$, the model to determine optimal pipe diameters becomes a tractable mixed integer linear program (MILP)

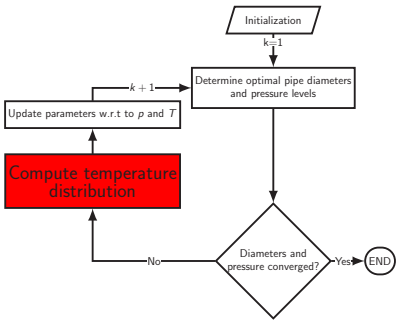
Thermophysical modeling

- Physical properties are nonlinear functions of pressure and temperature, e.g.:

Density	$\rho = \rho(p, T)$
Viscosity	$\nu = \nu(p, T)$
Heat capacity	$c_p = c_p(p, T)$
Thermal conductivity	$\kappa_f = \kappa_f(p, T)$



Thermophysical modeling



Effects on the temperature in the network

- Temperature T_v at a junction node v due to mixing

$$T_v = \frac{\sum_{a \in \delta^-(v)} c_{p,a} q_a T_a^{out}}{\sum_{a \in \delta^-(v)} c_{p,a} q_a}$$

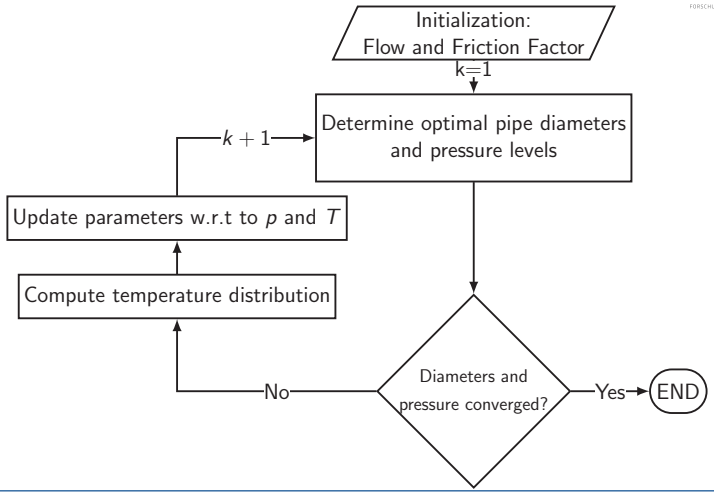
- Temperature change along a pipeline

$$T^{out} = T^{in} + \underbrace{\mu_{JT}(p^{out} - p^{in})}_{\textit{Joule-Thomson effect}} - \underbrace{\frac{1}{q_a c_p} k L \Delta T_{ln}}_{\text{Heat exchange with surrounding}}$$

- For given flow values and pressure distribution, calculate temperature distribution similar to computing flow values

T^{out}: Outlet temperature of pipe; T^{in}: Inlet temperature of pipe; q_a: Mass flow over pipe; c_p: Heat capacity of fluid; μ_{JT}: Joule-Thomson coefficient; k: Heat transmission rate; ΔT_{ln}: Logarithmic temperature difference with surrounding

Algorithm

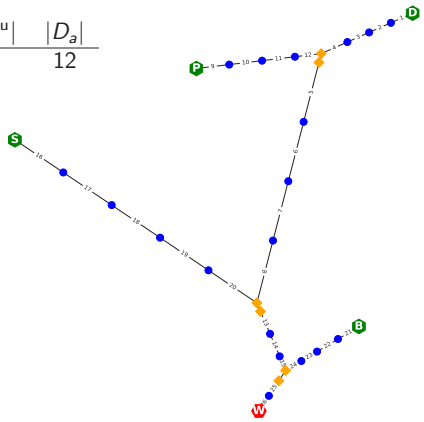


Case study - Network data

Connect four cement plants to harbour in North-Western Germany by tree-shaped network

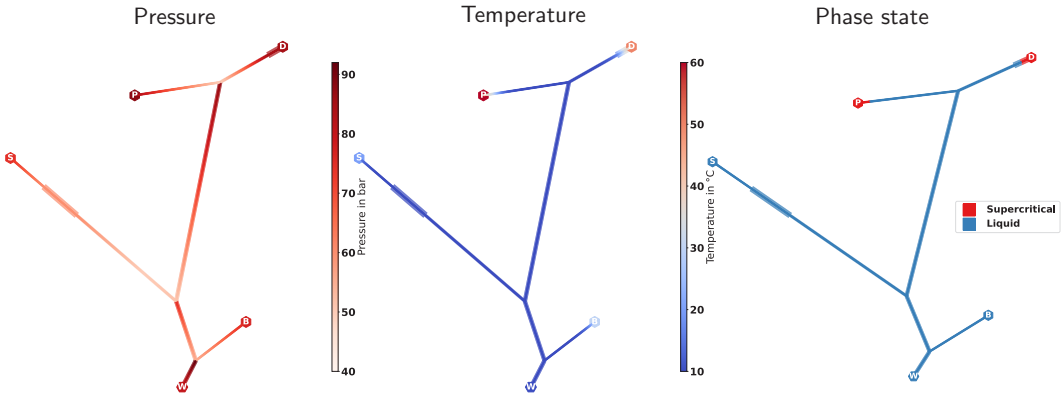
Set	$\lvert V^- \rvert$	$\lvert V^+ \rvert$	$\lvert V^0 \rvert$	$\lvert V^m \rvert$	$\lvert A^{pi} \rvert$	$\lvert A^{pu} \rvert$	$\lvert D_a \rvert$
N	1	4	1661	1652	1662	3	12

	D	P	S	B
Inflow in kg/s	16	8	8	8
Temperature in °C	50	60	20	30
Max Pressure in bar	95	105	88	88

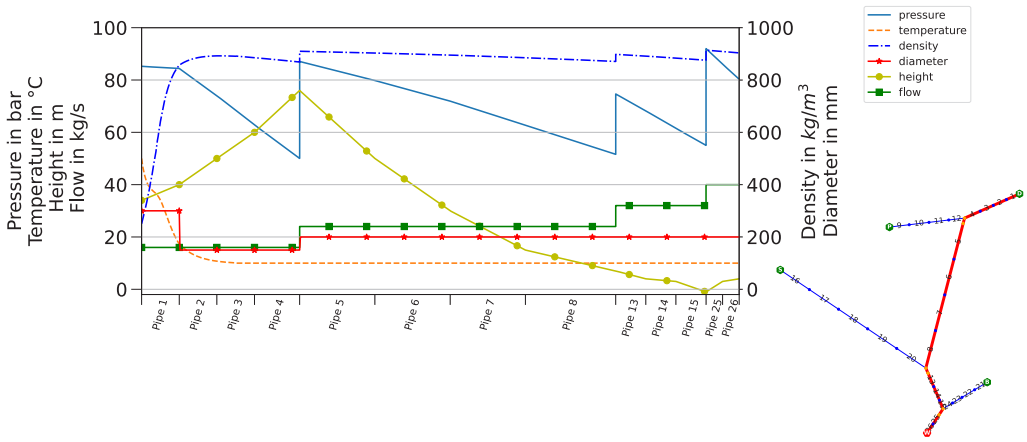


Minimal pressure at exit: 80 bar
Network consists of: ∼830 km pipelines
Length of each pipe segment: 500 m
Sections 1 and 17: Fixed to 300 mm
Surrounding temperature: 10 °C

Case study - Results



Case study - Results



Conclusion

- We presented an iterative algorithm for finding optimal pipeline diameters in a tree-shaped multi-source single-sink network for CO_2 transport
- To account for the complex thermophysical behavior of CO_2, we split the problem into
  a) Finding optimal pipe diameters by solving a tractable MILP
  b) modeling the thermophysical effects in the network, including mixing, heat exchange with the surrounding and the *Joule-Thomson effect*
- In each iteration we update physical parameters w.r.t. pressure and temperature levels
- We showed Proof-of-Concept by applying our method to a real-world planning instance

Outlook

- Verify coarse solution of optimization with results by a simulator
- Robustness of method by applying method to multiple instances and scenarios
- Extend method to handling CO_2-rich fluids $\implies$ change of phase envelope, i.e., more complex pressure bounds
- Extend method to *demand-based* components, e.g., hydrogen or ammonia as energy carrier $\implies$ single-source multiple-sink network

Thank You For Your Attention!
Any Questions?

References

Bell, I. H., Wronski, J., Quoilin, S., and Lemort, V. (2014).
Pure and Pseudo-pure Fluid Thermophysical Property Evaluation and the Open-Source Thermophysical Property Library CoolProp.
Industrial & Engineering Chemistry Research, 53(6):2498–2508.

D'Ambrosio, C., Lodi, A., Wiese, S., and Bragalli, C. (2015).
Mathematical programming techniques in water network optimization.
European Journal of Operational Research, 243(3):774–788.

IEA (2020).
The role of CCUS in low-carbon power systems.
Technical report, IEA, Paris.

Lenz, R. and Becker, K. H. (2022).
Optimization of capacity expansion in potential-driven networks including multiple looping: A comparison of modelling approaches.
OR Spectrum, 44(1):179–224.

Robinius, M., Schewe, L., Schmidt, M., Stolten, D., Thürauf, J., and Welder, L. (2019).
Robust optimal discrete arc sizing for tree-shaped potential networks.
Computational Optimization and Applications, 73(3):791–819.

Additional Slides

Determine optimal pipe diameters - Pipeline

Pressure change in a pipeline

$$p_v - p_u = \underbrace{\phi_a(d_a, q_a)\, q_a|q_a|}_{\text{friction term}} - \underbrace{(H_v^0 - H_u^0)\rho g}_{\text{elevation difference}} \qquad \forall a = (v, u) \in A^{\text{pi}}$$

Friction loss coefficient ϕ_a using Weymouth equation

$$\phi_a(d_a, q_a) = \frac{8L_a}{\pi^2 \rho d_a^5} \lambda_a(d_a, q_a)$$

Darcy friction factor λ_a using implicit Colebrook-White equation

$$\frac{1}{\sqrt{\lambda_a}} = -2\log_{10}\left(\frac{\varepsilon}{3.7 d_a} + \frac{2.51}{Re\sqrt{\lambda_a}}\right) \quad \text{with} \quad Re = \frac{d_a q_a}{A_a \nu}$$

H_v^0: height of v; ρ: density of fluid, g: gravitational constant; L_a: length of pipe; ε: roughness of pipe; Re: Reynold's number; ν: dynamic viscosity of fluid; A_a: cross section of pipe

Heat exchange with surrounding in buried pipe[1]

Determine heat transmission factor α

$$Pr = Pr(\nu, \kappa_f, c_p) = \frac{\nu}{\kappa_f / c_p} = \frac{c_p \nu}{\kappa_f}$$

$$Re = Re(\nu, d, q) = \frac{dq}{A\nu}$$

$$Nu = \frac{\alpha d}{\kappa_f} \quad \Longleftrightarrow \quad \alpha = \frac{Nu \kappa_f}{d}$$

$$Nu = \frac{(\zeta/8)\, Re\, Pr}{1 + 12.7\sqrt{\zeta/8}(Pr^{2/3} - 1)} \left[1 + \left(\frac{d}{L}\right)^{2/3}\right]$$

$$\zeta = (1.8 \log_{10} Re - 1.5)^{-2}$$

Heat exchange

$$\Delta Q = L Q_c = k L \Delta T_{ln}$$

$$\Delta T_{ln} = \frac{T^{in} - T^{out}}{\log \frac{T^{in} - T^{s}}{T^{out} - T^{s}}}$$

$$k = \frac{2\pi}{\frac{2}{\alpha d} + \frac{1}{\kappa_p} \log \frac{d_o}{d} + \frac{1}{\kappa_s} \log \frac{4s}{d}}$$

Start with initial guess $T^{out,0}$, using mean temperature $T_m = (T^{in} + T^{out,i-1})/2$ and pressure $p_m = (p^{in} + p^{out})/2$

$$c_p(p_m, T_m),\ \nu(p_m, T_m),\ \kappa_f(p_m, T_m),\ \mu_{JT}(p_m, T_m)$$

$$T^{out,i} = T^{in} + \mu_{JT}(p^{out} - p^{in}) - \frac{1}{q c_p} k L \Delta T_{ln}^{i-1}$$

[1] VDI Wärmeatlas, berlin, Heidelberg: Springer, 2013.

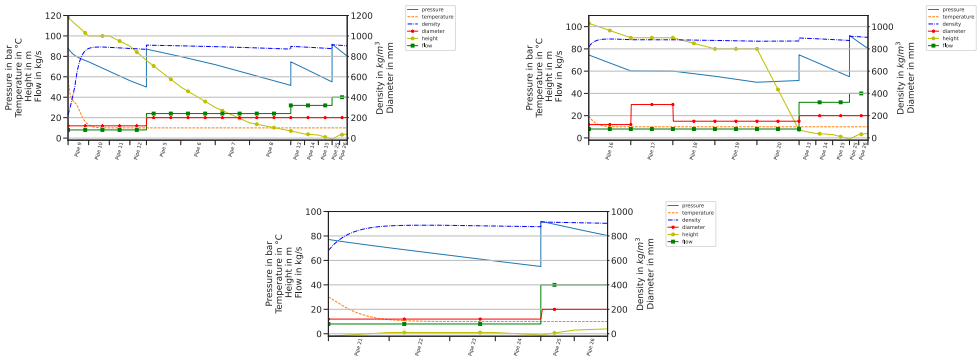
Compute flow values

- Let $G = (V, A)$ be directed in-tree graph rooted at a single exit node
- Let $V(u) := \{v \in V : (u, v) \in A \vee (v, u) \in A\}$ be the set of nodes adjacent to node u
- Let $L := \{v \in V : |V(v)| = 1 \wedge v \notin V^-\}$ be the set of leafs in the tree G
- For each $v \in L$ set the flow of the unique outgoing arc $a \in \delta^+(v)$ to $q_a = b_v$ and update $b_u := b_u + q_a$ for each neighbor $u \in V(v)$
- Let $V = V \backslash L$ and $A = A \backslash \bigcup_{v \in L} \delta^+(v)$, update L
- Iterate until A is empty

Compute temperature distribution

- Let $G = (V, A)$ be a directed in-tree graph with given flow value q_a for each $a \in A$ and pressure level p_v for each $v \in V$.
- Let $L := \{v \in V : |V(v)| = 1 \wedge v \notin V^-\}$ be the set of leafs in the tree G
- Let $V(u) := \{v \in V : (u, v) \in A \vee (v, u) \in A\}$ be the set of nodes adjacent to node u
- For each $v \in V^+$, an inflow temperature $T_v = T^{\text{in}}$ is given.
- For each $v \in L$, set the inlet temperature of the unique outgoing arc $a \in \delta^+(v)$ to $T_a^{\text{in}} = T_v$.
- If $a \in A^{\text{pi}}$, the outlet temperature T_a^{out} is determined by solving correlations for buried pipelines
- If $a \in A^{\text{pu}}$, set $T_a^{\text{out}} = T_a^{\text{in}}$.
- Then, for each node $u \in \bigcup_{v \in L} V(v)$, compute its mixing temperature T_u
- Let $V = V \backslash L$ and $A = A \backslash \bigcup_{v \in L} \delta^+(v)$
- Update L, and iterate until A is empty

Results



Spotlights on success stories of public-private partnership

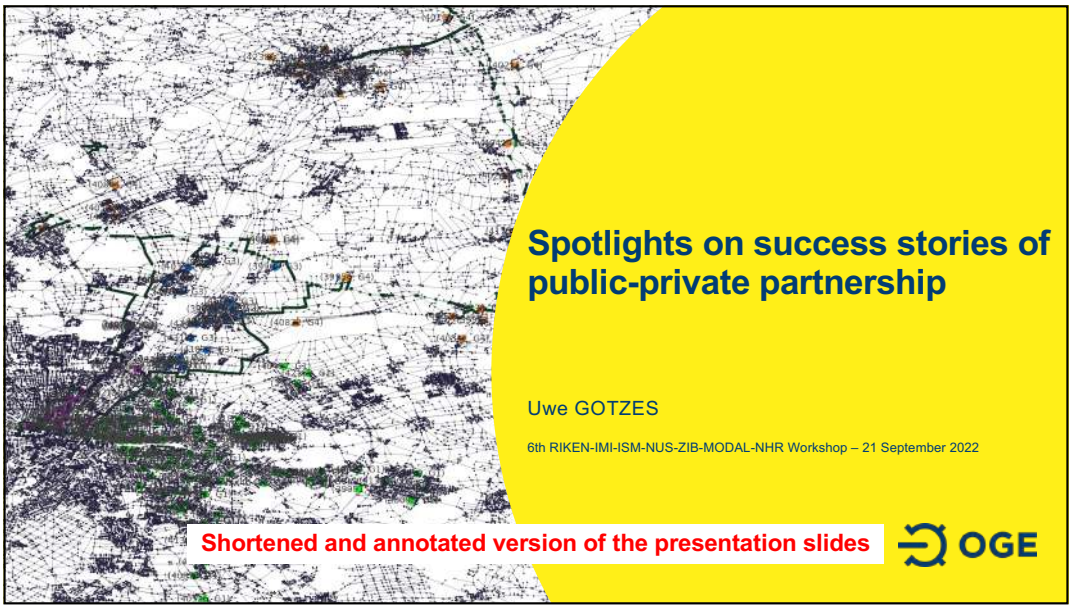
Uwe GOTZES

Open Grid Europe GmbH, Essen, Germany
uwe.gotzes@oge.net

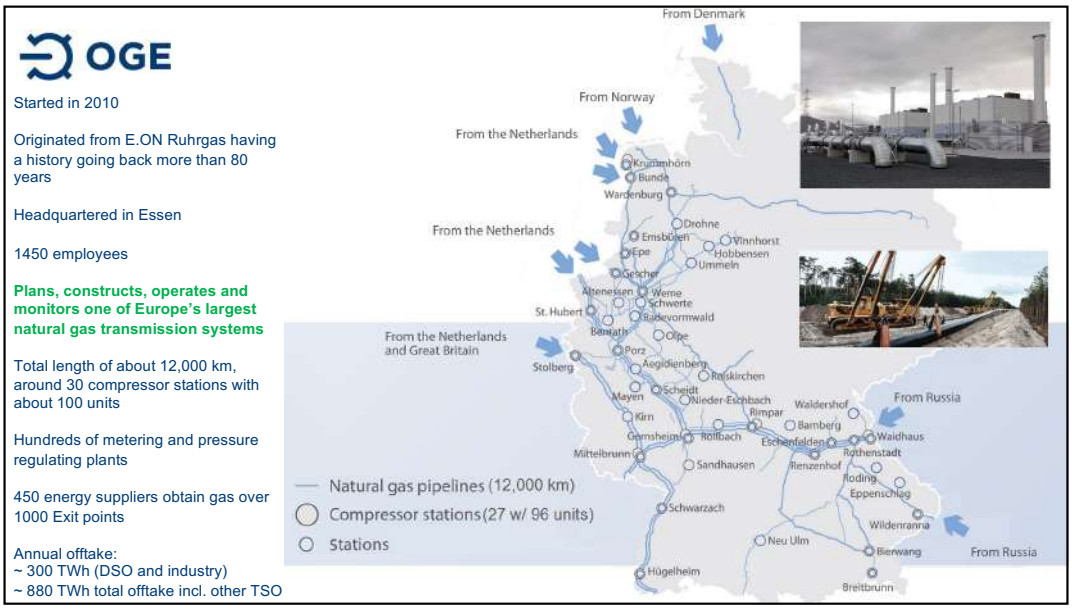
The Zuse Institute Berlin [1] and OGE, Germany's largest natural gas transmission system operator [2], have been cooperating for more than a decade in a variety of application-oriented research projects [3]. In my talk I will briefly go through the history of the projects and the collaboration. A more recent project addressing telecommunication network design by the application of ZIB's SCIP-Jack and SCIP [4][5] will be presented in detail.

References

[1] https://www.zib.de/
[2] https://oge.net/en
[3] https://www.zib.de/features/research-campus-modal
[4] https://scipjack.zib.de/
[5] https://www.scipopt.org/


Spotlights on success stories of public-private partnership
Uwe GOTZES
6th RIKEN-IMI-ISM-NUS-ZIB-MODAL-NHR Workshop – 21 September 2022
Shortened and annotated version of the presentation slides
OGE

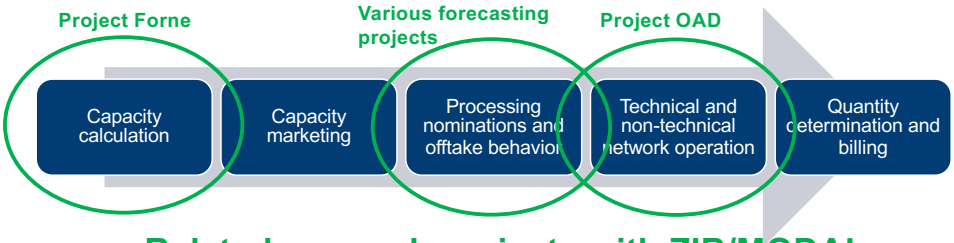
1


OGE
Started in 2010
Originated from E.ON Ruhrgas having a history going back more than 80 years
Headquartered in Essen
1450 employees
Plans, constructs, operates and monitors one of Europe's largest natural gas transmission systems
Total length of about 12,000 km, around 30 compressor stations with about 100 units
Hundreds of metering and pressure regulating plants
450 energy suppliers obtain gas over 1000 Exit points
Annual offtake:
~ 300 TWh (DSO and industry)
~ 880 TWh total offtake incl. other TSO
From Denmark
From Norway
From the Netherlands
From the Netherlands
From the Netherlands and Great Britain
From Russia
From Russia
Natural gas pipelines (12,000 km)
Compressor stations (27 w/ 96 units)
Stations

2

We offer…

- the transmission of natural gas through our network
- the technical and commercial services to go with it



Related research projects with ZIB/MODAL

and we provide…

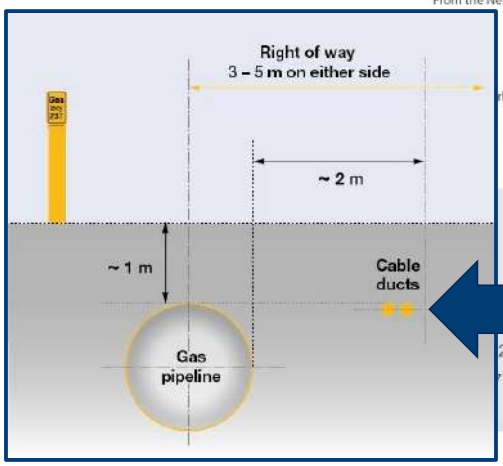
Project Carp (Detailed presentation on the following slides)

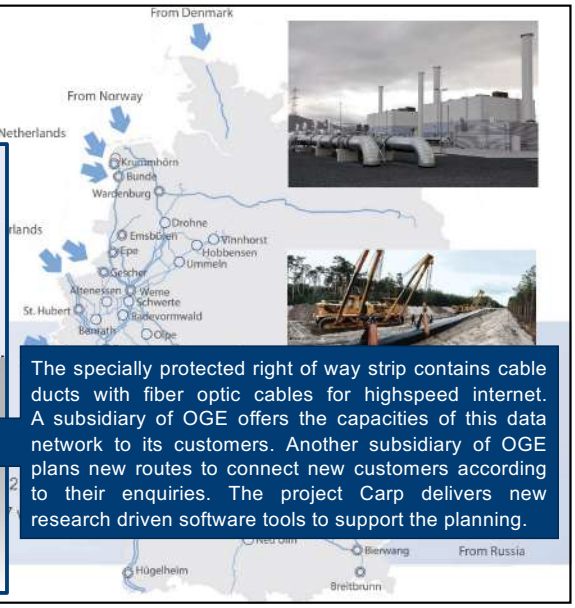
- commercial, technical and IT services for other companies

3

OGE

A cross section through a pipeline route





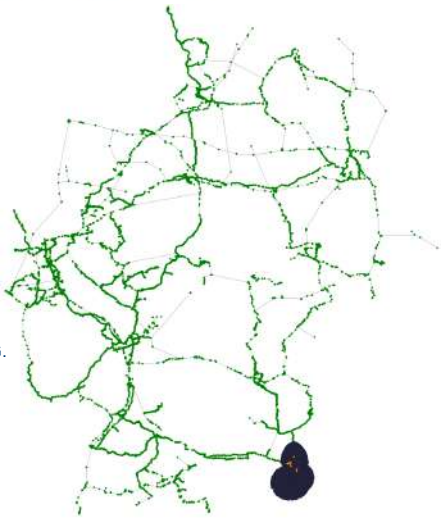
The specially protected right of way strip contains cable ducts with fiber optic cables for highspeed internet. A subsidiary of OGE offers the capacities of this data network to its customers. Another subsidiary of OGE plans new routes to connect new customers according to their enquiries. The project Carp delivers new research driven software tools to support the planning.

4

Our Project **Carp** – **C**omputer **a**ided **r**oute **p**lanning

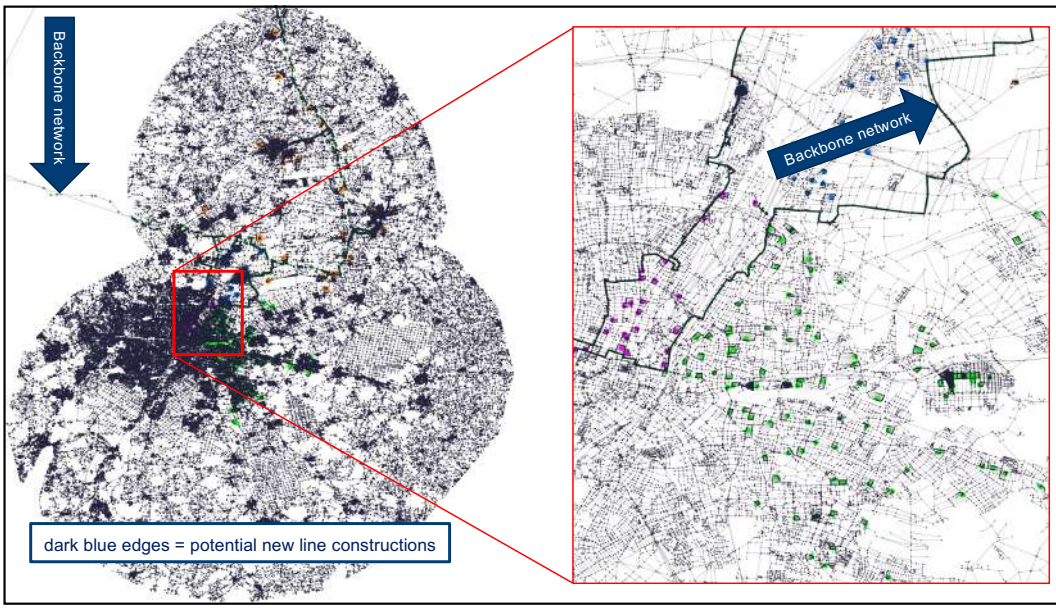
Given

- undirected weighted graph $G = (V, E)$
- a subset $I \subset V$
- k groups of n_k vertices



How does a customer enquiry look like?
Mathematically spoken, we are given an undirected weighted graph G.
The subset *I* consists of the green vertices that form our backbone network in the cable ducts right next to the gas pipelines.
The *k* disjoint groups of *n_k* nodes are all hidden in the dark blue, densely packed area in the lower left corner of the graph
(next slide is a zoom into this area).

5



6

Tasks

A) Find a minimal subset $S \subset E$ with two properties

1. the vertices in n_k are pairwisely connected by a path inside of S
2. inside of S there exists a path from one vertex in n_k to a vertex in I

B) A special case of A) with
$k = 1$, $n_1 = \{T_0, T_1\}$ and $n_1 \cap I \neq \emptyset$

C) $k = 1$, $n_1 = \{T_0, \ldots, T_{n-1}\}$
Find a shortest path that 'visits' $T_0, \ldots, T_{n-1}$ in order and contains a vertex $i \in I$

D) $k = 1$, $n_1 = \{T_0, T_1\}$
Find a shortest node disjoint circle that contains $\{T_0, T_1\}$ and a vertex $i \in I$

7

Approach to solve Task A

- Almost a Steiner Tree Problem…but not exactly
- It just so happens that Daniel Rehfeldt from ZIB has developed (one of) the best Steiner tree solvers in the world…so why not talk with him first?

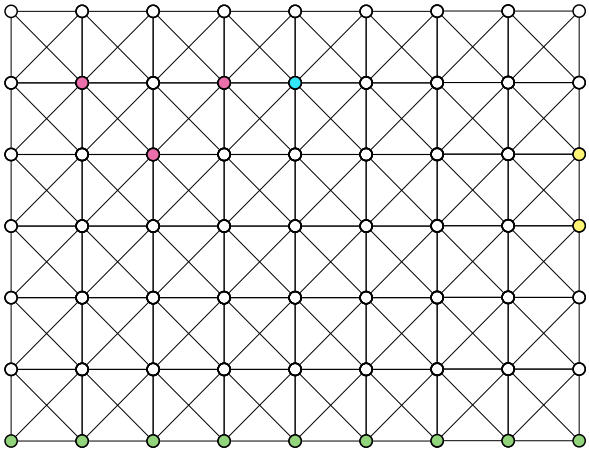
Algorithm:

1. Replace all edges by an antiparallel pair of directed edges
2. Connect all vertices from I unidirectionally witch a 'supernode' i^* at cost 0
3. Solve as many *Steiner Aborescence Problems* (SAP) as there are groups and consider vertices of other groups as already connected to the backbone network

After all SAP for all groups are solved, it might happen, that vertices of several groups are connected in a tree structure, but not yet to the backbone network. To overcome this flaw:

1. Set all edge costs of tree edges to 0
2. Solve ordinary *Steiner Problem* with all group nodes as terminals

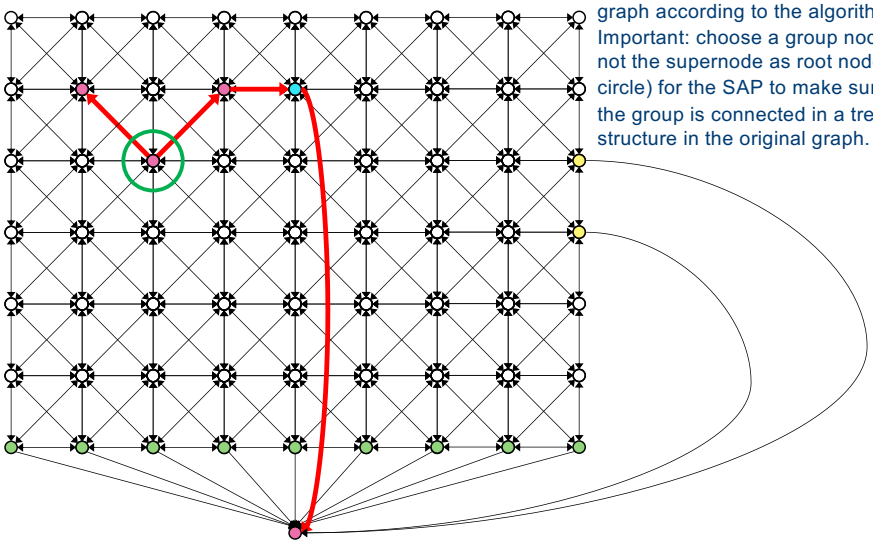
8



Example graph with euclidean distance as edge weights

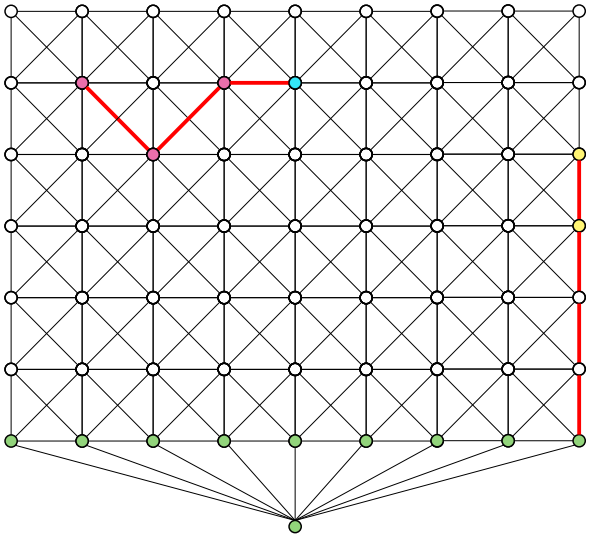
Backbone network (green) and three vertex groups (magenta, turquois and yellow)

9



Solution of the first SAP in the modified graph according to the algorithm. Important: choose a group node and not the supernode as root node (green circle) for the SAP to make sure that the group is connected in a tree structure in the original graph.

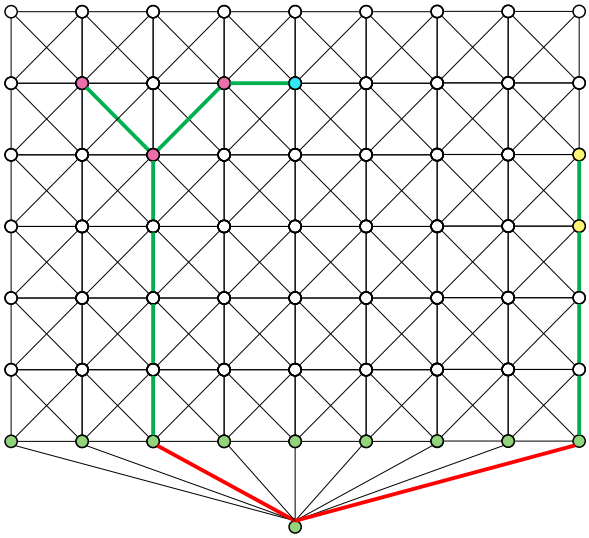
10



Solution edges of all three SAP in the original graph.

All groups are connected by a tree structure, but turqouis and magenta groups are not yet connected to the backbone network. To achieve this: Solve ordinary SPG with magenta, turqouis, yellow groups and supernode as terminals.

11

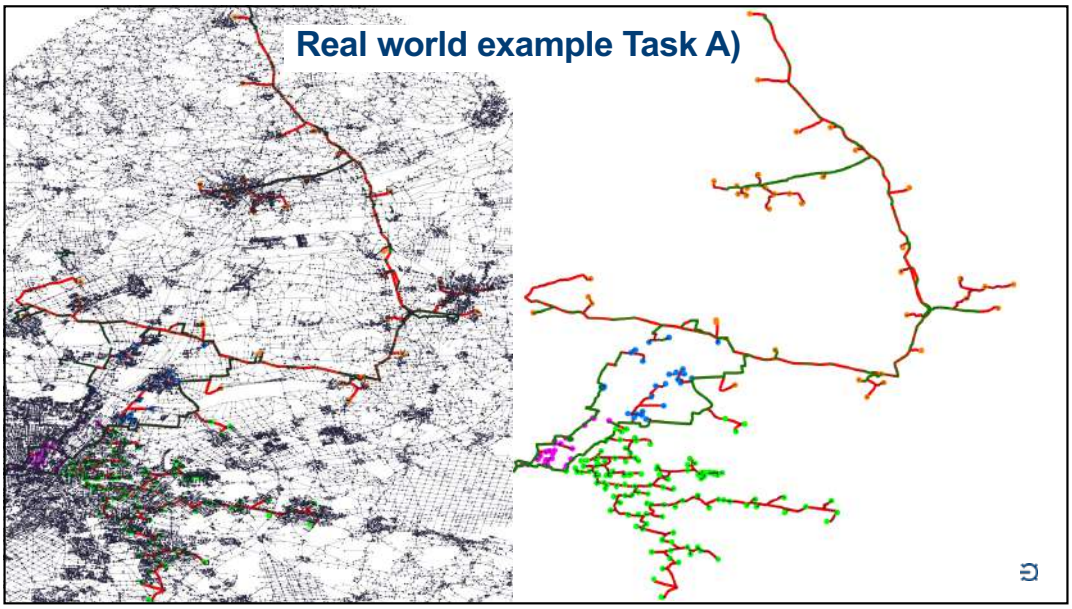


Hooray, nice feasible solution found!

However, the method is a heuristic

1. How good is the heuristic?
2. How can the method be improved?
3. Example where the heuristic does not yield the optimal solution?

12



13

Tasks

A) Find a minimal subset $S \subset E$ with two properties

1. the vertices in n_k are pairwisely connected by a path inside of S
2. inside of S there exists a path from one vertex in n_k to a vertex in I

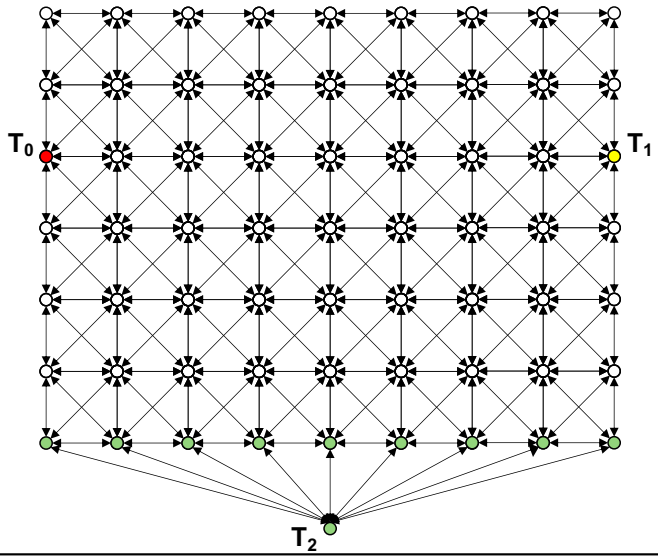
B) A s[illegible] case of A) with
$k = 1$, [illegible] $\{T_0, T_1\}$ and $n_1 \cap I \neq \emptyset$

Not discussed in detail here

C) $k = 1$, $n_1 = \{T_0, [illegible]_{-1}\}$
Find a shortest path t[illegible]its' $T_0, \ldots, T_{n-1}$ in order and contains a vertex $i \in I$

D) $k = 1$, $n_1 = \{T_0, T_1\}$
Find a shortest node disjoint circle that contains $\{T_0, T_1\}$ and a vertex $i \in I$

14



Nightly E-Mail ping-pong-session with Thorsten Koch to solve task D as follows:

1. Modify original graph as shown here (now directed and new vertex T_2)
2. Set up three shortest path problems
3. Bundle all three shortest path problems in one huge *Three ways round trip problem* containing three copies of the modified graph
4. Add additional constraints to forbid shortcuts via T_2

15

Shortest path from T_0 to T_1

$$\text{Min}: \sum_{(i,j)\in E} c_{ij} \cdot x_{ij}$$

s.t.

$$\sum_{(i,j)\in E} x_{ij} \le 1 \quad \wedge \quad \sum_{(i,j)\in E} x_{ij} = \sum_{(j,i)\in E} x_{ji} \qquad \forall i \in V \setminus \{T_0, T_1\} \qquad (1)$$

$$\sum_{(T_0,j)\in E} x_{T_0 j} = 1 \qquad (2)$$

$$\sum_{(i,T_1)\in E} x_{iT_1} = 1 \qquad (3)$$

Shortest path from T_2 to T_0

$$\text{Min}: \sum_{(i,j)\in E} c_{ij} \cdot x_{ij}$$

s.t.

$$\sum_{(i,j)\in E} x_{ij} \le 1 \quad \wedge \quad \sum_{(i,j)\in E} x_{ij} = \sum_{(j,i)\in E} x_{ji} \qquad \forall i \in V \setminus \{T_2, T_0\} \qquad (1)$$

$$\sum_{(T_2,j)\in E} x_{T_2 j} = 1 \qquad (2)$$

$$\sum_{(i,T_0)\in E} x_{iT_0} = 1 \qquad (3)$$

Shortest path from T_1 to T_2

$$\text{Min}: \sum_{(i,j)\in E} c_{ij} \cdot x_{ij}$$

s.t.

$$\sum_{(i,j)\in E} x_{ij} \le 1 \quad \wedge \quad \sum_{(i,j)\in E} x_{ij} = \sum_{(j,i)\in E} x_{ji} \qquad \forall i \in V \setminus \{T_1, T_2\} \qquad (1)$$

$$\sum_{(T_1,j)\in E} x_{T_1 j} = 1 \qquad (2)$$

$$\sum_{(i,T_2)\in E} x_{iT_2} = 1 \qquad (3)$$

Three Ways Round Trip Problem

$$\text{Min}: \sum_{(i,j,k)\in E\times\{0,1,2\}} c_{ij} \cdot x_{ij}^k$$

s.t.

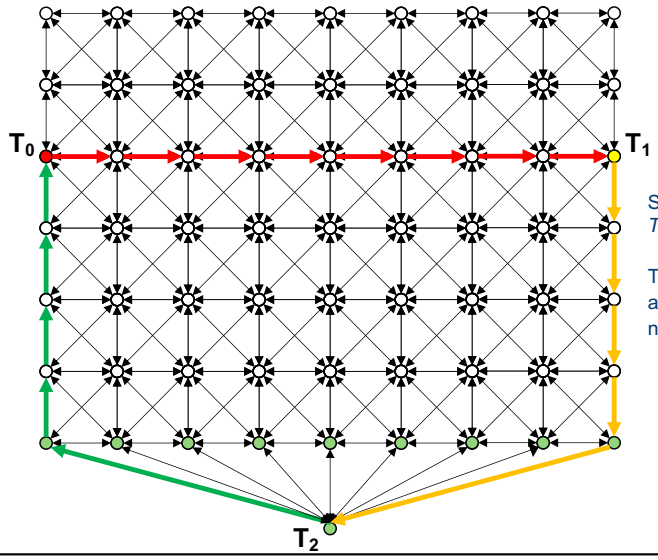
$$\sum_{(i,j,k)\in E\times\{0,1,2\}} x_{ij}^k \le 1 \qquad \forall i \in V \setminus \{T_0, T_1, T_2\} \qquad (1)$$

$$\sum_{(i,j)\in E} x_{ij}^k = \sum_{(j,i)\in E} x_{ji}^k \qquad \forall i \in V \setminus \{T_0, T_1, T_2\}, \forall k = 0,1,2 \qquad (2)$$

$$\sum_{(T_j,i)\in E} x_{T_j i}^k = \delta_{jk} \qquad \forall j, k = 0,1,2 \qquad (3)$$

$$\sum_{(i,T_j)\in E} x_{iT_j}^k = \delta_{(j+2) \bmod 3,\, k} \qquad \forall j, k = 0,1,2 \qquad (4)$$

16



Solution of the
Three ways round trip problem.

This solution has a flaw, because it took a shortcut via T_2 and hence is not a circle in the original graph.

17

Three Ways Round Trip Problem

Modifications of the model in red. Effect: Shortcuts via T_2 are now forbidden. ¥overline{E} denotes the edges of the original graph. ¥delta is Kronecker's delta.

$$\text{Min}: \sum_{(i,j,k)\in E\times\{0,1,2\}} c_{ij}\cdot x_{ij}^k$$

s.t.

$$\sum_{(i,j)\in E} x_{ij}^k = \sum_{(j,i)\in E} x_{ji}^k \qquad \forall i \in V\setminus\{T_0, T_1, T_2\},\ \forall k = 0,1,2 \qquad (1)$$

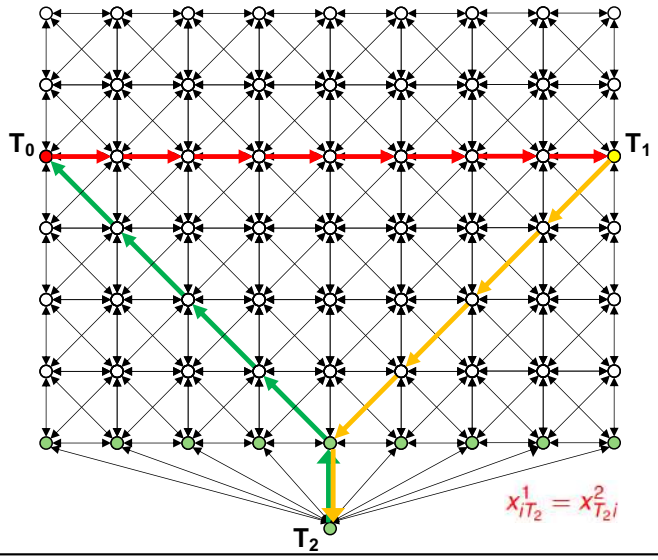
$$\sum_{(i,j,k)\in \bar{E}\times\{0,1,2\}} x_{ij}^k \le 1 \qquad \forall i \in V\setminus\{T_0, T_1, T_2\} \qquad (2)$$

$$\sum_{(T_j,i)\in E} x_{T_j i}^k = \delta_{jk} \qquad \forall j,k = 0,1,2 \qquad (3)$$

$$\sum_{(i,T_j)\in E} x_{iT_j}^k = \delta_{(j+2)\bmod 3,\,k} \qquad \forall j,k = 0,1,2 \qquad (4)$$

$$x_{iT_2}^1 = x_{T_2 i}^2 \qquad \forall (i,T_2)\in E \qquad (5)$$

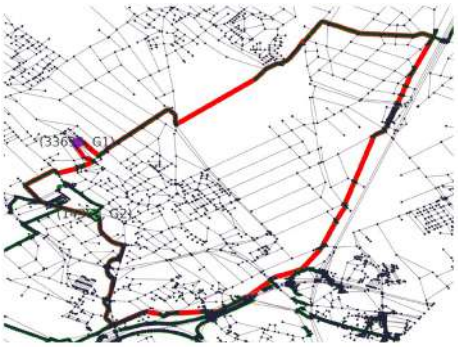
18



$$\sum_{(i,j,k)\in\bar{E}\times\{0,1,2\}} x^k_{ij} \leq 1$$

19

Shortest node disjoint circle
Real world example Task D)



Thank you!

20

The 6th RIKEN-IMI-ISM-NUS-ZIB-MODAL-NHR Workshop on
Advances in Classical and Quantum Algorithms for
Optimization and Machine Learning

September 16th - 19th, 2022, Tokyo (The university of Tokyo), Japan,

and September 21st - 22nd, 2022, Fukuoka (Kyushu University), Japan

Deep Switching State Space Model (DS3M) for Nonlinear Time Series Forecasting with Regime Switching

Xiuqin Xu*1 and Ying Chen*2

*1 Integrative Sciences and Engineering Programme, NUS
Graduate School Institute of Data Science, National University of Singapore
xiuqin.xu@u.nus.edu

*2 Department of Mathematics, National University of Singapore
Asian Institute of Digital Finance, National University of Singapore,
Risk Management Institute, National University of Singapore, Singapore
matcheny@nus.edu.sg

We propose a deep switching state space model (DS3M) for efficient inference and forecasting of nonlinear time series with irregularly switching among various regimes. The switching among regimes is captured by both discrete and continuous latent variables with recurrent neural networks. The model is estimated with variational inference using a reparameterization trick. We test the approach on a variety of simulated and real datasets. In all cases, DS3M achieves competitive performance compared to several state-of-the-art methods (e.g. GRU, SRNN, DSARF, SNLDS), with superior forecasting accuracy, convincing interpretability of the discrete latent variables, and powerful representation of the continuous latent variables for different kinds of time series. Specifically, the MAPE values increase by 0.09\% to 15.71\% against the second-best performing alternative models.

Deep Switching State Space Model (DS^{3}M) for Nonlinear Time Series Forecasting with Regime Switching

Xiuqin XU
Ying CHEN

https://arxiv.org/abs/2106.02329
Institute of Data Science & Integrative Sciences and Engineering Programme
Department of Mathematics & Risk Management Institute & Asian Institute of Digital Finance
National University of Singapore

NUS National University of Singapore | Faculty of Science | Faculty of Arts & Social Sciences | RMI | Institute of Operations Research and Analytics | Institute of Data Science | Asian Institute of Digital Finance

Nonlinear time series with regime switching

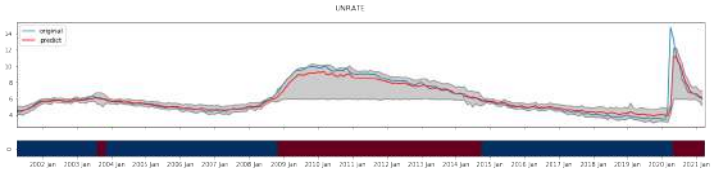


Figure: The U.S. monthly unemployment rate 2002 – 2021.

- The U.S. unemployment rate depends on (discrete) unobservable economic status of booming or recession. It is also influenced by some latent (continuous) variables, e.g. elasticity of regional wage level and others, that in turn vary with the discrete status.
- Deciphering these discrete and continuous latent variables can gain insights.

Technical challenges

- In many studies, researchers need to model and infer similar type of time series that disobey traditional assumptions of e.g. linearity, normality and stationarity in the statistical modeling, but exhibit nonlinearity with stochastic regime switching behaviours.
- Examples: health care (sleep apnea), economics (unemployment rate), traffic and transportation (metro passengers volume), meteorology (sea surface temperature), energy (electricity demand), to name just a few.
- Two challenges:
  - a severe modeling misspecification
  - lack of interpretation on the stochastic regimes

Switching state space models (SSSM)

- The evolution of time series is assumed to be driven by hidden factors switching among discrete regimes, see Bae et al., 2014; Fox et al., 2009; François et al., 2014; Ghahramani et al., 2000.
- The SSSM is a generalization of the Hidden Markov Models (HMMs) and State Space Models (SSMs).

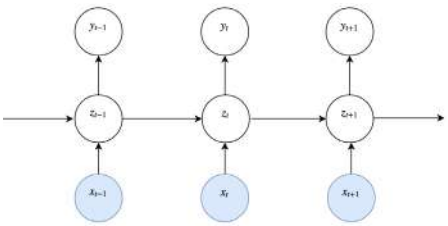


Figure: State Space Model

Linear Gaussian State Space Model (LGSSM)

The dynamics in each regime are usually represented by simple models that could be efficiently estimated even with a small sample size (Durbin et al., 2012), and the switching among regimes is controlled by hidden transition probabilities of a Markov chain.

- Transition function for the latent continuous variable z_t:

$$(z_t|z_{t-1}, x_t) \sim N(\mu_t, \Sigma_z) \tag{1}$$

  where $\mu_t = W_z z_{t-1} + W_x x_t + b_z$ and Σ_z is covariance.

- Emission function for the observation y_t:

$$y_t|z_t \sim N(m_t, \Sigma_y) \tag{2}$$

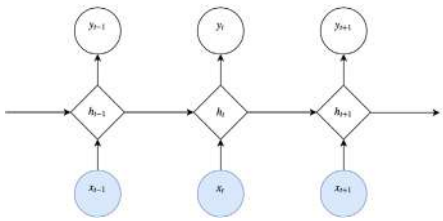
  where $m_t = W_y z_t + b_y$ and Σ_y is covariance.

SSSMs and misspecification

- Hidden Markov Model (HMM): Baum et al. (1966), Linear Gaussian State Space Model (LGSSM): Durbin et al. (2012),
- Non-linear and non-Gaussian SSM: Doucet et al. (2009), Julier et al. (1997), and Smith et al. (1962)
- Non-stationary SSM: Ackerson et al. (1970), Chang et al. (1978), Fox et al. (2009), Ghahramani et al. (2000), Hamilton (1990), and Murphy (1998)
  - *By extending the local linear models with different regimes, the resulting model approximates a globally nonlinear behaviour and is expected to retain interpretation.*
  - *The existing nonlinear models rely on pre-specified local parametric forms that usually have simple structures, either linear or nonlinear, which may not be comprehensive enough to describe the actual patterns in the modern nonlinear time series, and thus easily lead to model misspecification.*

Deep learning

Recurrent neural networks (RNN) have emerged as the new benchmark to model nonlinear time series with highly complex dependence.



- Gate structure alleviates gradient vanishing: Long-Short Term Memory (LSTM,Hochreiter et al., 1997), Gated recurrent unit (GRU,Chung et al., 2014), Transformers (Li et al., 2019) and temporal convolution networks (Sen et al., 2019)

Recurrent Neural Networks (RNN)

- Transition function for hidden states h_t to encode past input $x_{1:t}$ with a deterministic nonlinear function

$$h_t = f(h_{t-1}, x_t) \tag{3}$$

f is commonly chosen as LSTM or GRU.

- Emission function for y_t

$$y_t | h_t \sim \pi(y_t; \phi) \tag{4}$$

$$\phi = g(h_t) \tag{5}$$

g is often chosen to be a nonlinear function.

Overparametrization and interpretation

- The classic DL models are deterministic and ignore the presence of unobserved stochastic signals.
- The only randomness allowed appears in the conditional output probability models, with either a simple unimodal distribution, e.g. Gaussian (Salinas et al., 2020), or a mixture of simple unimodal distributions, e.g. Gaussian Mixture models (Graves, 2013).
- It has to require a large number of parameters to ensure a reasonable modeling accuracy (Zhang et al., 2005). This in turn requires a large sample size to ensure estimation efficiency and to avoid overfitting.

The relatively small sample size of real data, and more importantly, the stochastic behaviors of regime switching make standard deep learning approaches computationally infeasible and lack of interpretation on the fitted models.

SSM and RNN

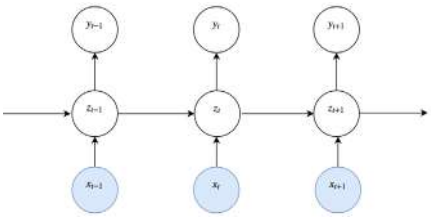


Figure: State Space Model

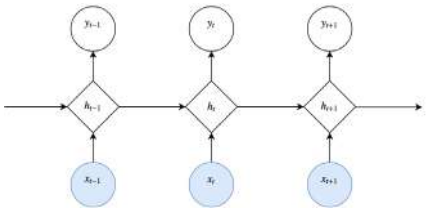


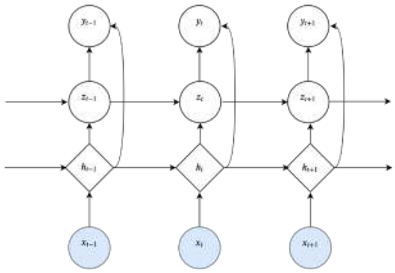
Figure: Recurrent Neural Network

- SSM only allows for simple state structure or linear transitions; RNN enables to represent complex dependence with richer internal states and nonlinear transitions
- The latent variables in SSM are random;
  The hidden states in RNNs are deterministic.

Deep state space models (DSSM)

DSSM introduce continuous Gaussian latent variables at each time-step and combine SSM and RNN/MLP.

- SRNNs (Fraccaro et al., 2016) incorporate the deterministic dynamics of RNNs by interlacing them with SSM. The hidden states do not depend on z_t and transition is linear.
- STORNs (Bayer et al., 2014), VRNN (Chung et al., 2015) make the transition equation non-linear via cutting the ties between the latent states and connect them through the deterministic state of RNNs.



z_t continuous. Uneasy to interpret

DSSM with discrete latent variable

While **continuous** latent variables has more expressive power, **discrete** latent variables representing regime switching are natural.

- Johnson et al. (2016): emission function is a neural network.
- Dong et al. (2020): both the emission and transition functions are nonlinear neural networks.
- Farnoosh et al. (2021): approximates high-dimensional time series with a multiplication of latent factors and latent weights, where the latent weights are modeled by a nonlinear autoregressive model, switched by a Markov chain of discrete latent variables.

The evolution of time series is purely driven by the discrete latent variable only.

Deep Switching State Space Model (DS^{3}M)

- Incorporate both continuous and discrete latent into RNN and prove consistency and stability:
  - RNN + nonlinear SSSM: emission and transition governed by a Markov chain of d_t and parameterized by MLPs.
  - discrete latent d_t represents unknown regimes and influences both Y_t and continuous latent z_t
  - z_t in the SSSM could use the long-term information embedded in the RNN; the RNN is skip-connected to the observations to further improve the forecasting.
- Develop efficient estimation based on an approximate variational inference algorithm that can scale to large data sets
- DS^{3}M can leverage the interpretability of discrete latent variables, the powerful representation ability of continuous latent variables, and the nonlinearity of deep learning models compared to SOTA.

Outline

1. Motivation ✓
2. SOTA
3. Model
4. Experiments
5. Conclusion

Settings

- A time series of T observations as $y_{1:T} = \{y_1, y_2, \cdots, y_T\}$, $y_t \in \mathbb{R}^D$.
- A sequence of inputs as $x_{1:T} = \{x_1, x_2, \cdots, x_T\}$, $x_t \in \mathbb{R}^U$.
  - In time series forecasting, x_t can be one or multiple lagged values of the time series, e.g. y_{t-1} and higher orders $y_{t-2}, y_{t-3}, \cdots$.
  - The inputs x_t could also contain exogenous variables.

We are interested in modeling $p(y_{1:T}|x_{1:T})$ and inferring the predictive distributions for the one-step-ahead to τ-step-ahead observations $\{y_{T+1}, \cdots, y_{T+\tau}\}$ and the discrete latent states $\{d_{T+1}, \cdots, d_{T+\tau}\}$.

Switching linear dynamical system (SLDS)

- The dynamics of each regime is explained by a linear state space model.

$$z_t = W_z^{(d_t)} z_{t-1} + W_x^{(d_t)} x_t + b_z^{(d_t)} + e_t, e_t \sim N(0, \Sigma_z^{(d_t)}) \quad (6)$$

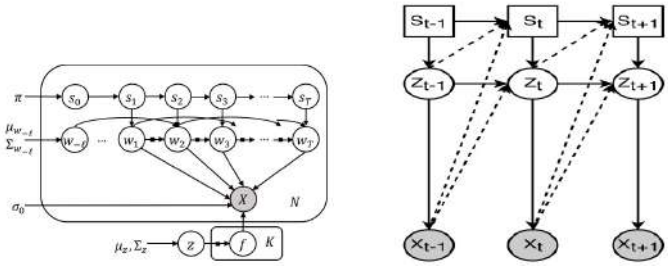
$$y_t = W_y^{(d_t)} z_t + b_y^{(d_t)} + \epsilon_t, \epsilon_t \sim N(0, \Sigma_y^{(d_t)}) \quad (7)$$

- The discrete latent variables $d_t \in \{1, 2, \cdots, K\}$ at each time step $t = 1, 2, \cdots, T$, follows a Markov chain, $d_t|d_{t-1}$ follows a transition matrix $\Gamma \in R^{K \times K}$, where $\Gamma_{i,j} = p(d_t = j|d_{t-1} = i)$.
- The discrete latent variables d_t have impact on both the continuous latent variables $z_t \in \mathbb{R}^Z$ and y_t

When $K = 1$, the model is also termed as the Linear Gaussian State Space Model (LGSSM).

SOTA

- GRU, SRNN (Fraccaro et al., 2016), DSARF (Farnoosh et al., 2021) and SNLDS (Dong et al., 2020)



(a) DSARF (b) SNLDS

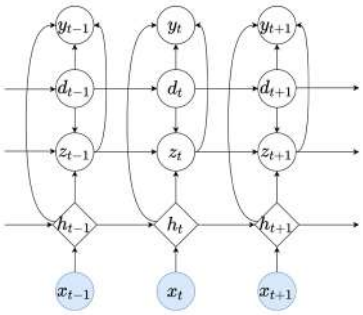
- Most of the extensions assumed that the discrete latent d_t only influences the transition of the continuous latent z_t.
- SRNN can be viewed as DS^{3}M model without discrete latent variables.

Other alternative models

- Recurrent SLDS (rSLDS) by Becker-Ehmck et al. (2019) and Linderman et al. (2017) which extends the open-loop Markov dynamics and makes d_t depending on the hidden state z_{t-1}. Dong et al. (2020) extended the open-loop Markov dynamics by making d_t depends on last observations.
- Tree structure prior on the switching variables of rSLDS (Nassar et al., 2018), deep Rao-Blackwellised Particle Filter (Kurle et al., 2020)
- Sometimes can improve accuracy, but can also lead to unnecessarily frequent state shifts in the estimated discrete latent variables, making interpretations difficult.

DSARF has been shown to outperform several models such as rSLDS, SLDS for time series forecasting (Farnoosh et al., 2021).

Deep Switching State Space Model (DS^{3}M)



- stack an RNN below the switching state space model
- design a direct connection of the RNN hidden state h_t to the time series y_t inspired by the skip connection in ResNet, Transformers and SRNN.

Formulation

1. Recurrent step:
$$h_t = f_h(h_{t-1}, x_t)$$
f_h is chosen as an LSTM or GRU
2. Switching step: $p(d_t|d_{t-1})$ follows a Markovian transition matrix $\Gamma \in R^{K\times K}$
$$\Gamma_{i,j} = p(d_t = j|d_{t-1} = i)$$
3. Transition step:
$$(z_t|z_{t-1}, h_t, d_t = k) \sim N(z_t; \mu_t^{(k)}, \Sigma_t^{(k)})$$
$$\mu_t^{(k)} = f_1^{(k)}(z_{t-1}, h_t), \quad \log \Sigma_t^{(k)} = f_2^{(k)}(z_{t-1}, h_t)$$
$f_1^{(k)}$,$f_2^{(k)}$ are parameterized by neural network models (MLP)

Formulation (Con't)

4. Output step:

$$y_t|z_t, h_t, d_t = k \sim \pi(\Phi_t^{(k)}) \tag{8}$$

$$\Phi_t^{(k)} = f_o^{(k)}(z_t, h_t) \tag{9}$$

$f_o^{(k)}$ is parameterized by neural network models (MLP)
π can be chosen according to the stochastic nature of the time series, e.g. Gaussian for bell-shaped data, Log-Gaussian for data with asymmetry etc.

The DS^{3}M includes all the parameters that parameterize the following functions:

$$\theta = \{f_h, \Gamma, f_o, \{f_1^{(k)}\}_{k=1}^K, \{f_2^{(k)}\}_{k=1}^K, \{f_o^{(k)}\}_{k=1}^K\}$$

Estimation

The loglikelihood is

$$\mathcal{L}(\theta) = \log p_\theta\left(\mathbf{y}_{1:T}|\mathbf{x}_{1:T}\right)$$

$$p_\theta\left(\mathbf{y}_{1:T}, \mathbf{z}_{1:T}, \mathbf{d}_{1:T}|\mathbf{x}_{1:T}\right) = \prod_{t=1}^T p_\theta\left(\mathbf{y}_t|\mathbf{z}_t, \mathbf{h}_t, \mathbf{d}_t\right) p_\theta\left(\mathbf{z}_t|\mathbf{z}_{t-1}, \mathbf{h}_t, \mathbf{d}_t\right) p_\theta\left(\mathbf{d}_t|\mathbf{d}_{t-1}\right)$$

- $\mathcal{L}(\theta)$ can be obtained by averaging out $\mathbf{z}_{1:T}$ and $\mathbf{d}_{1:T}$ in above joint probability. Intractable!
- Maximum likelihood method is not applicable!
- Use variational inference instead. Specifically, we design an inference network with parameter ϕ, i.e. using an approximated posterior $q_\phi\left(\mathbf{z}_{1:T}, \mathbf{d}_{1:T}|\mathbf{y}_{1:T}, \mathbf{x}_{1:T}\right)$ for the true posterior $p_\theta(\mathbf{z}_{1:T}, \mathbf{d}_{1:T}|\mathbf{x}_{1:T}, \mathbf{y}_{1:T})$ and then optimize an evidence lower bound $ELBO(\theta, \phi)$.

Variational Inference

- For any approximated posterior $q_\phi\left(\mathbf{z}_{1:T}, \mathbf{d}_{1:T} | \mathbf{y}_{1:T}, \mathbf{x}_{1:T}\right)$

$$\begin{aligned}
&\mathcal{L}(\theta) \geq ELBO(\theta, \phi) \\
&= \iint q_\phi\left(\mathbf{z}_{1:T}, \mathbf{d}_{1:T} | \mathbf{y}_{1:T}, \mathbf{x}_{1:T}\right) \log \frac{p_\theta\left(\mathbf{y}_{1:T}, \mathbf{z}_{1:T}, \mathbf{d}_{1:T} | \mathbf{x}_{1:T}\right)}{q_\phi\left(\mathbf{z}_{1:T}, \mathbf{d}_{1:T} | \mathbf{y}_{1:T}, \mathbf{x}_{1:T}\right)} \mathrm{d}\mathbf{z}_{1:T} \mathrm{d}\mathbf{d}_{1:T} \\
&= \mathbb{E}_{q_\phi}\left[\log p_\theta\left(\mathbf{y}_{1:T} | \mathbf{z}_{1:T}, \mathbf{d}_{1:T}, \mathbf{x}_{1:T}\right)\right] \\
&\quad - \mathrm{KL}\left(q_\phi\left(\mathbf{z}_{1:T}, \mathbf{d}_{1:T} | \mathbf{y}_{1:T}, \mathbf{x}_{1:T}\right) \| p_\theta\left(\mathbf{z}_{1:T}, \mathbf{d}_{1:T} | \mathbf{x}_{1:T}\right)\right)
\end{aligned}$$

- When $q_\phi\left(\mathbf{z}_{1:T}, \mathbf{d}_{1:T} | \mathbf{y}_{1:T}, \mathbf{x}_{1:T}\right) = p_\theta\left(\mathbf{z}_{1:T}, \mathbf{d}_{1:T} | \mathbf{y}_{1:T}, \mathbf{x}_{1:T}\right)$ (the true posterior), we have $\mathcal{L}(\theta) = ELBO(\theta, \phi)$.

How to choose $q_\phi\left(\mathbf{z}_{1:T}, \mathbf{d}_{1:T} | \mathbf{y}_{1:T}, \mathbf{x}_{1:T}\right)$?

- To achieve a tight ELBO, consider true posterior factorization derived from the d-separation (Geiger et al., 1990):

$$p_\theta\left(\mathbf{z}_{1:T}, \mathbf{d}_{1:T} | \mathbf{y}_{1:T}, \mathbf{x}_{1:T}\right) = \prod_t p_\theta\left(\mathbf{z}_t | \mathbf{z}_{t-1}, \mathbf{d}_t, \mathbf{y}_{t:T}, \mathbf{h}_{t:T}\right) p_\theta\left(\mathbf{d}_t | \mathbf{d}_{t-1}, \mathbf{y}_{t:T}, \mathbf{h}_{t:T}\right)$$

- We design the approximated posterior with the same form of factorization:

$$q_\phi\left(\mathbf{z}_{1:T}, \mathbf{d}_{1:T} | \mathbf{y}_{1:T}, \mathbf{x}_{1:T}\right) = \prod_t q_{\phi_z}\left(\mathbf{z}_t | \mathbf{z}_{t-1}, \mathbf{d}_t, A_t\right) q_{\phi_d}\left(\mathbf{d}_t | \mathbf{d}_{t-1}, A_t\right)$$

where $A_t = g_{\phi_A}\left(A_{t+1}, [\mathbf{y}_t, \mathbf{h}_t]\right)$, $\phi = \{\phi_z, \phi_d, \phi_A\}$, and g_{ϕ_A} is parameterized as a backward RNN.

Parameterization

- $q_{\phi_z}(\mathbf{z}_t|\mathbf{z}_{t-1}, \mathbf{d}_t, A_t)$ is parameterized to be a Gaussian density:

$$z_t|z_{t-1}, A_t, d_t = k \sim N(z_t; \mu_t^{(k)}, \Sigma_t^{(k)})$$

$$\mu_t^{(k)} = g_1^{(k)}(z_{t-1}, A_t), \quad \log \Sigma_t^{(k)} = g_2^{(k)}(z_{t-1}, A_t)$$

  $g_1^{(k)}$, $g_2^{(k)}$ is parameterized by neural network model (MLP)

- $q_{\phi_d}(\mathbf{d}_t|A_t, \mathbf{d}_{t-1})$ is parameterized to be a Categorical distribution:

$$d_t|_{A_t, d_{t-1}=k} \sim \text{Cat}(\text{softmax}(W^{(k)} A_t))$$

Inference Network

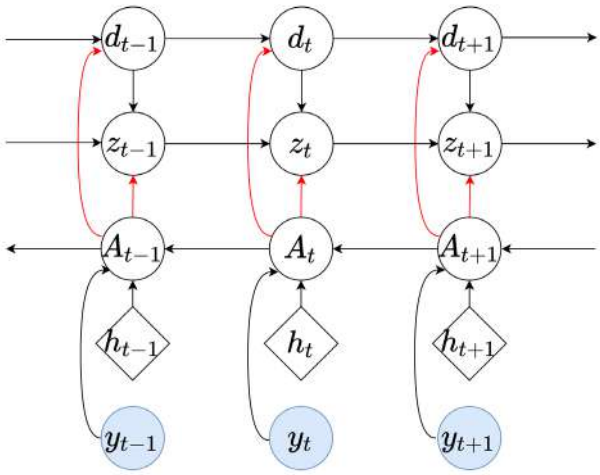


Figure: The graphical model which represents $q_\phi(\mathbf{z}_{1:T}, \mathbf{d}_{1:T}|\mathbf{y}_{1:T}, \mathbf{x}_{1:T})$.

How to estimate the parameters?

- Instead of maximizing $\mathcal{L}(\theta)$ w.r.t θ, we maximize a variational evidence lower bound $ELBO(\theta, \phi)$ w.r.t θ, ϕ
- $\frac{\partial ELBO(\theta,\phi)}{\partial \theta}$ and $\frac{\partial ELBO(\theta,\phi)}{\partial \phi}$
- Method: Stochastic gradient descent

Factorization of ELBO

- With the defined approximate posterior, the ELBO can be rewritten as

$$ELBO(\theta, \phi) = \mathbb{E}_{q_\phi}\left[\log p_\theta\left(\mathbf{y}_{1:T}|\mathbf{z}_{1:T}, \mathbf{d}_{1:T}, \mathbf{h}_{1:T}\right)\right] - \mathrm{KL}\left(q_\phi\left(\mathbf{z}_{1:T}, \mathbf{d}_{1:T}|\mathbf{y}_{1:T}, \mathbf{h}_{1:T}\right) \| p_\theta\left(\mathbf{z}_{1:T}, \mathbf{d}_{1:T}|\mathbf{h}_{1:T}\right)\right)$$

- The factorization of the approximated posterior:

$$q_\phi\left(\mathbf{z}_{1:T}, \mathbf{d}_{1:T}|\mathbf{y}_{1:T}, \mathbf{x}_{1:T}\right) = \prod_t q_{\phi_z}\left(\mathbf{z}_t|\mathbf{z}_{t-1}, \mathbf{d}_t, A_t\right) q_{\phi_d}\left(\mathbf{d}_t|\mathbf{d}_{t-1}, A_t\right)$$

- The factorization for the prior:

$$p_\theta\left(\mathbf{z}_{1:T}, \mathbf{d}_{1:T}|\mathbf{x}_{1:T}\right) = p_\theta\left(\mathbf{z}_{1:T}, \mathbf{d}_{1:T}|\mathbf{h}_{1:T}\right) = p(\mathbf{d}_{1:T}) p\left(\mathbf{z}_{1:T}|\mathbf{d}_{1:T}, \mathbf{h}_{1:T}\right) = \prod_t p(\mathbf{d}_t|\mathbf{d}_{t-1}) p\left(\mathbf{z}_t|\mathbf{z}_{t-1}, \mathbf{d}_t, \mathbf{h}_t\right)$$

Factorization of ELBO (Con't)

$$
\begin{aligned}
&\text{ELBO}(\theta, \phi) \\
&= \sum_t \Big\{ \mathbb{E}_{q_\phi^*(\mathbf{z}_{t-1}, \mathbf{d}_{t-1})} \sum_{\mathbf{d}_t} q_{\phi_d}(\mathbf{d}_t) \mathbb{E}_{q_{\phi_z}(\mathbf{z}_t)} \left[\log p_\theta \left(\mathbf{y}_t | \mathbf{z}_t, \mathbf{d}_t, \mathbf{h}_t \right) \right] - \\
&\quad \mathbb{E}_{q_\phi^*(\mathbf{z}_{t-1}, d_{t-1})} \sum_{\mathbf{d}_t} q_{\phi_d}(\mathbf{d}_t) KL \left[q_{\phi_z}(\mathbf{z}_t | \mathbf{z}_{t-1}, \mathbf{d}_t, \mathbf{A}_t) \| p_\theta(\mathbf{z}_t | \mathbf{z}_{t-1}, \mathbf{d}_t, \mathbf{h}_t) \right] \\
&\quad - \mathbb{E}_{q_\phi^*(d_{t-2})} \sum_{\mathbf{d}_{t-1}} q_{\phi_d}(\mathbf{d}_{t-1}) KL \left[q_{\phi_d}(\mathbf{d}_t | \mathbf{d}_{t-1}, \mathbf{A}_t) \| p_\theta(\mathbf{d}_t | \mathbf{d}_{t-1}) \right] \Big\},
\end{aligned}
$$

- $q_\phi^*(\mathbf{z}_t, \mathbf{d}_t) = \int q_\phi(\mathbf{z}_{1:t}, \mathbf{d}_{1:t} | \mathbf{y}_{1:T}, \mathbf{x}_{1:T}) \, \mathrm{d}\mathbf{z}_{1:t-1} \mathrm{d}\mathbf{d}_{1:t-1}$
- $q_\phi^*(\mathbf{d}_t) = \int q_{\phi_d}(\mathbf{d}_{1:t} | \mathbf{y}_{1:T}, \mathbf{x}_{1:T}) \, \mathrm{d}\mathbf{d}_{1:t-1}$.

Approximate ELBO

We approximate the ELBO using a Monte Carlo method. We sample $(z_t^{(s)}, d_t^{(s)})$ for $t = 1 \cdots, T$ from $q_\phi^*(\mathbf{z}_t, \mathbf{d}_t)$ using ancestral sampling.

$$
\begin{aligned}
&\text{ELBO}(\theta, \phi) \\
&\approx \sum_t \Big\{ \sum_{\mathbf{d}_t} q_{\phi_d}(\mathbf{d}_t) \log p_\theta \left(\mathbf{y}_t | \mathbf{z}_t^{(s)}, \mathbf{d}_t, \mathbf{h}_t \right) \\
&\quad - \sum_{\mathbf{d}_t} q_{\phi_d}(\mathbf{d}_t) KL \left[q_{\phi_z} \left(\mathbf{z}_t | \mathbf{z}_{t-1}^{(s)}, \mathbf{d}_t, \mathbf{A}_t \right) \| p_\theta \left(\mathbf{z}_t | \mathbf{z}_{t-1}^{(s)}, \mathbf{d}_t, \mathbf{h}_t \right) \right] \\
&\quad - \sum_{\mathbf{d}_{t-1}} q_{\phi_d}(\mathbf{d}_{t-1}) KL \left[q_{\phi_d}(\mathbf{d}_t | \mathbf{d}_{t-1}, \mathbf{A}_t) \| p_\theta(\mathbf{d}_t | \mathbf{d}_{t-1}) \right] \Big\}.
\end{aligned}
$$

Gradient: $\frac{\partial ELBO(\theta,\phi)}{\partial \theta}$

The derivative of the $ELBO(\theta, \phi)$ with respect to θ can be calculated as:

$$
\begin{aligned}
&\frac{\partial ELBO(\theta, \phi)}{\partial \theta} \\
&= E_{q_\phi(\mathbf{z}_{1:T}, \mathbf{d}_{1:T} | \mathbf{y}_{1:T}, \mathbf{x}_{1:T})} \frac{\partial \log p_\theta \left(\mathbf{y}_{1:T}, \mathbf{z}_{1:T}, \mathbf{d}_{1:T} | \mathbf{x}_{1:T}\right)}{\partial \theta} \\
&= \sum_{t=1}^{T} E_{q_\phi} \{ \frac{\partial \log p_\theta \left(\mathbf{y}_t | \mathbf{z}_t, \mathbf{d}_t, \mathbf{h}_t\right)}{\partial \theta} + \frac{\partial \log p_\theta \left(\mathbf{z}_t | \mathbf{z}_{t-1}, \mathbf{d}_t, \mathbf{h}_t\right)}{\partial \theta} + \frac{\partial \log p_\theta \left(\mathbf{d}_t | \mathbf{d}_{t-1}\right)}{\partial \theta} \} \\
&\approx \sum_{t=1}^{T} E_{q_\phi} \{ \frac{\partial \log p_\theta \left(\mathbf{y}_t | \mathbf{z}_t^{(s)}, \mathbf{d}_t^{(s)}, \mathbf{h}_t\right)}{\partial \theta} + \frac{\partial \log p_\theta \left(\mathbf{z}_t^{(s)} | \mathbf{z}_{t-1}^{(s)}, \mathbf{d}_t^{(s)}, \mathbf{h}_t\right)}{\partial \theta} + \frac{\partial \log p_\theta \left(\mathbf{d}_t^{(s)} | \mathbf{d}_{t-1}^{(s)}\right)}{\partial \theta} \}
\end{aligned}
\tag{10}
$$

Gradient: $\frac{\partial ELBO(\theta,\phi)}{\partial \phi}$

The derivative of the $ELBO(\theta, \phi)$ with respect to ϕ is more tricky as ϕ appears in the expectation in $ELBO(\theta, \phi)$.

- Score function gradient estimator (Williams, 1992) can be used but suffer from **high variance**.
- Reparameterization trick (Kingma et al., 2014; Rezende et al., 2014) is often used instead, **low variance** gradient estimator
  - $z = g(\epsilon; \phi)$

$$
\frac{\partial \mathbb{E}_{q_{(z|x;\phi)}} f(z)}{\partial \phi} = \frac{\partial \mathbb{E}_\epsilon f(g(\epsilon; \phi))}{\partial \phi} = \mathbb{E}_\epsilon [\frac{\partial f(g(\epsilon; \phi))}{\partial \phi}] \tag{11}
$$

  - e.g.
    $z_t \sim N(\mu_t, \Sigma_t)$, $\epsilon_t \sim N(0, 1)$, $z_t = g(\epsilon; \mu_t, \Sigma_t) = \mu_t + \epsilon_t \Sigma_t$

Predictive distributions using Monte Carlo

- First make inference on the posterior distributions of $\{z_t, d_t\}_{t=1}^{T}$ and then generate samples of $\{z_t^{(s)}, d_t^{(s)}, y_t^{(s)}\}_{t=T}^{T+\tau}$, $s = 1, \cdots, S$. S represents the number of Monte Carlo samples.
- The predictive distributions for the one-step-ahead to τ-step-ahead observations $\{y_{T+1}, \cdots, y_{T+\tau}\}$ and the discrete latent variables $\{d_{T+1}, \cdots, d_{T+\tau}\}$ are then approximated with empirical distribution functions of the generated samples.

Stability in mean square

Theorem

Under regular conditions, for the neural networks $f_1^{(k)}$ and $f_2^{(k)}$ that parameterize the mean $\mu_t^{(k)}$ and diagonal covariance matrix $\Sigma_t^{(k)}$ of the latent state dynamics $z_t \sim N(\mu_t^{(k)}, \Sigma_t^{(k)})$ with arbitrary activation function a, there exists an equivalent pointwise affine map which ensures that the latent variable z_t is globally stable in mean-square.

Experiments

- Simulations
  - Toy example
  - Lorenz attractor
- Real data analysis
  - Six datasets in Econ, Medicine, Traffic, Meteorology, Energy
  - Sleep apnea, Hangzhou metro flow, Seattle traffic flow, Pacific temperature, Unemployment rate, French electricity
  - First four datasets are analyzed in Farnoosh et al. (2021); The French electricity is analyzed in Xu et al. (2021), the unemployment rate is selected to represent Econ data.

Toy example

$$d_0 \sim \text{Bernouli}(0.5), z_0 = 0$$

$$d_t | d_{t-1} \sim \Gamma = \begin{bmatrix} 0.95 & 0.05 \\ 0.05 & 0.95 \end{bmatrix}, d_t \in \{0, 1\}$$

$$x_t = y_{t-1}$$

$$z_{t|d_t=0} = 0.6 z_{t-1} + 0.4 \times \tanh(x_t + z_{t-1}) + w_t^{(0)}, w_t^{(0)} \sim N(0, 10)$$

$$z_{t|d_t=1} = 0.1 z_{t-1} + 0.2 \times \sin(x_t + z_{t-1}) + w_t^{(1)}, w_t^{(1)} \sim N(0, 1)$$

$$y_{t|d_t=0} = 1.5 z_t + \tanh(z_t) + v_t^{(0)}, v_t^{(0)} \sim N(0, 5)$$

$$y_{t|d_t=1} = 0.5 z_t + \sin(z_t) + v_t^{(1)}, v_t^{(1)} \sim N(0, 0.5)$$

Toy example (1- step-ahead forecasting)

Figure: Prediction for the toy example. The red color represents $d_t = 0$ and the blue color for $d_t = 1$

Toy example

Table: Summary of the simulation results (mean ± standard deviation)

		Toy		
		DS³M	SNLDS	DSARF
Forecasting	RMSE	**14.572** ± 0.352	16.541 ± 0.024	15.244 ± 0.136
	Duration for dt=1	**7.509** ± 1.579	1.282 ± 0.001	3.946 ± 0.426
	Duration for dt=0	**7.634** ± 1.667	1.667 ± 0.012	3.274 ± 0.985
	Accuracy (%)	**0.788** ± 0.033	0.543 ± 0.001	0.765 ± 0.047
	F1 score	**0.778** ± 0.023	0.549 ± 0.001	0.757 ± 0.035
Inference	Accuracy (%)	**0.849** ± 0.004	0.692 ± 0.003	0.819 ± 0.044
	F1 score	**0.831** ± 0.005	0.544 ± 0.002	0.808 ± 0.039

Lorenz attractor

Lorenz attractor is a canoinal nolinear dynamical system with the dynamics:

$$y_t = Wz_t + v_t, \text{where } W \in R^{10\times 3}, v_t \sim N(0, 0.5I_{10}).$$

$$\frac{d\mathbf{z}}{dt} = \begin{bmatrix} \alpha\left(z_2 - z_1\right) \\ z_1\left(\beta - z_1\right) - z_2 \\ z_1 z_2 - \gamma z_3 \end{bmatrix}$$

- The latent variable $z_t = [z_{t,1}, z_{t,2}, z_{t,3}]^T$. $y_t \in R^{10}$ is observable.
- Simulated the time series with a length of 3 000 and transform the time series into subsequences with a length of 5.
- Training: Validation: Testing = 1:1:1

Lorenz attractor forecasted switching variable

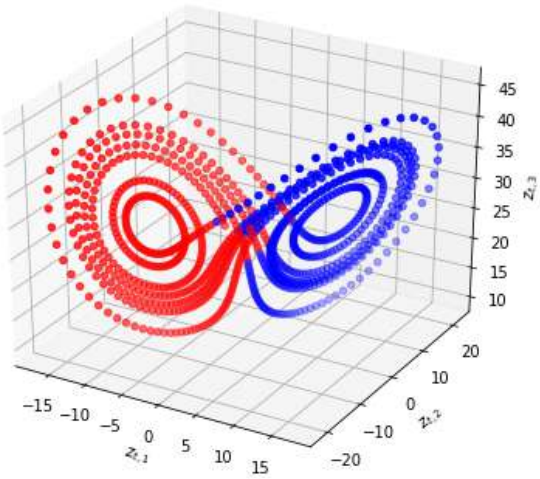


Figure: DS^{3}M: The forecasted switching variable against the true z_t.

Lorenz attractor

Table: Summary of the simulation results (mean ± standard deviation)

		Lorenz		
		DS^{3}M	SNLDS	DSARF
Forecasting	RMSE	0.168 ± 0.017	0.226 ± 0.065	**0.030** ± 0.000
	Accuracy (%)	**0.882** ± 0.079	0.616 ± 0.065	0.788 ± 0.143
	F1 score	**0.837** ± 0.127	0.600 ± 0.100	0.775 ± 0.124
Inference	Accuracy (%)	**0.911** ± 0.068	0.744 ± 0.174	0.789 ± 0.146
	F1 score	**0.883** ± 0.103	0.680 ± 0.244	0.761 ± 0.113

Real data

Table: Description of the datasets

Dataset	**Frequency**	**Dimension**	T**_train+**T**_valid**	T**_test** (time)	
Sleep	half a second	1	1000	1000	(500 seconds)
Unemployment	month	1	639	240	(20 years)
Hangzhou	10 mins	80	2160	540	(5 days)
Seattle	5 mins	323	6624	1440	(5 days)
Pacific	month	2520	336	60	(5 years)
Electricity	half a hour	48	2601	320	(1 year)

- **Short-term**: 1-step-ahead forecast
- **Long-term**: make multiple forecasts simultaneously for all T_test, standing at the end of the T_train+T_valid

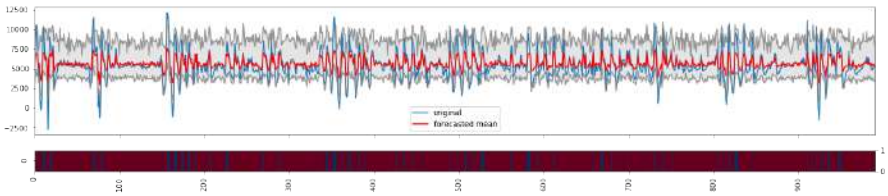
Real data analysis

Table: Summary of forecastes on testing data. The best models are in bold. "-" indicates the model forecasts diverge to unreasonable values and are omitted.

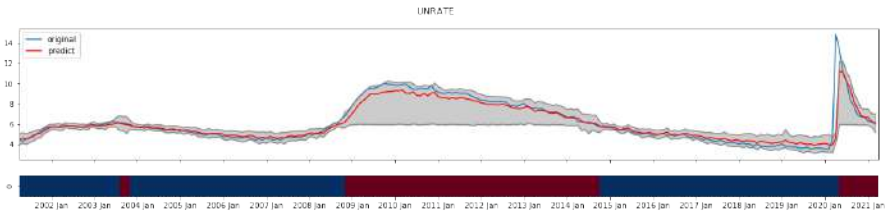
	Datasets	RMSE					MAPE (%)				
		DS^{3}M	SNLDS	DSARF	SRNN	GRU	DS^{3}M	SNLDS	DSARF	SRNN	GRU
Short-term	Sleep	**1201**	2789	1557	1806	1264	**15.46**	88.06	39.25	50.8	31.17
	Unemployment	**0.75**	1.59	1.06	2.01	1.05	**4.53**	16.13	8.11	23.15	5.13
	Hangzhou	**32.53**	36.67	34.81	33.80	38.42	24.04	**23.90**	29.73	25.40	30.48
	Seattle	**4.16**	4.18	4.44	4.17	4.18	**5.81**	5.85	7.27	6.00	6.89
	Pacific	0.57	15.78	**0.53**	0.58	0.56	1.69	58.01	**1.57**	1.74	1.68
	Electricity	**2971**	5133	8805	3642	4784	**4.58**	7.79	18.64	5.34	6.60
Long-term	Hangzhou	47.50	42.83	**42.28**	60.89	73.18	**38.20**	50.6	43.65	82.81	86.61
	Seattle	**4.17**	4.19	-	**4.17**	16.93	**5.81**	5.86	-	**5.81**	27.95
	Pacific	**0.72**	-	0.73	0.98	0.76	**2.15**	-	2.29	2.99	2.22

- Long-term forecasting is only achievable for time series with regular patterns (Farnoosh et al., 2021). Thus, we exclude some datasets for long-term forecasting.

Short-term forecasts

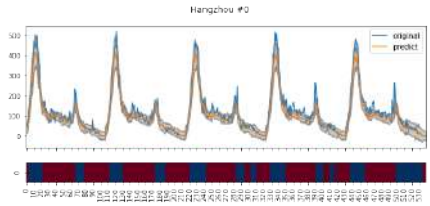


(a) Sleep apnea (measured at 2 Hz)

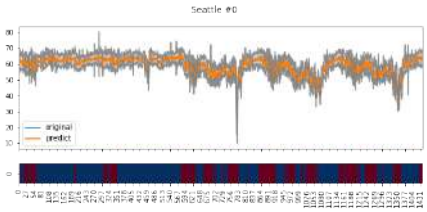


(b) US unemployment rate

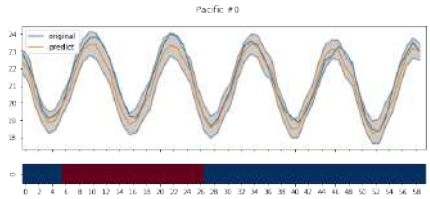
Short-term forecasts II



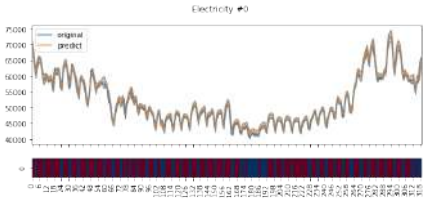
(c) Hangzhou metro station 0



(d) Seattle traffic loop 0



(e) Pacific location 0



(f) French Electricity 0:00

Conclusion

- Proposed a deep switching state space model (DS^{3}M) for forecasting nonlinear time series with regime switching.
- The switching among regimes is modeled by both discrete and continuous latent variables with recurrent neural networks.
- Developed an **efficient scalable inference and learning** method
- The DS^{3}M achieved competitive performance against several state-of-the-art methods for a variety of simulated and real datasets.
- Code and data are available at https://github.com/Sherry-Xu.

Deep Switching State Space Model (DS^{3}M) for Nonlinear Time Series Forecasting with Regime Switching

Xiuqin XU
Ying CHEN

https://arxiv.org/abs/2106.02329

Institute of Data Science & Integrative Sciences and Engineering Programme
Department of Mathematics & Risk Management Institute & Asian Institute of Digital Finance
National University of Singapore

NUS National University of Singapore | Faculty of Science | Faculty of Arts & Social Sciences | RMI | Institute of Operations Research and Analytics | Institute of Data Science | Asian Institute of Digital Finance

References

Ackerson, G and K Fu (1970)
On state estimation in switching environments
IEEE Transactions on Automatic Control 15 (1), pp. 10–17.

Bae, G. I., W. C. Kim, and J. M. Mulvey (2014)
Dynamic asset allocation for varied financial markets under regime switching framework
European Journal of Operational Research 234 (2), pp. 450–458.

References II

Baum, L. E. and T. Petrie (1966)
Statistical inference for probabilistic functions of finite state Markov chains
The annals of mathematical statistics 37 (6), pp. 1554–1563.

Bayer, J. and C. Osendorfer (2014)
'Learning Stochastic Recurrent Networks'
NIPS 2014 Workshop on Advances in Variational Inference, pp. 1–9.

References III

Becker-Ehmck, P., J. Peters, and P. Van Der Smagt (2019)
'Switching Linear Dynamics for Variational Bayes Filtering'
International Conference on Machine Learning, pp. 553–562.

Chang, C.-B. and M. Athans (1978)
State estimation for discrete systems with switching parameters
IEEE Transactions on Aerospace and Electronic Systems AES-14 (3), pp. 418–425.

References IV

- Chung, J., C. Gulcehre, K. Cho, and Y. Bengio (2014)
  'Empirical evaluation of gated recurrent neural networks on sequence modeling'
  NIPS 2014 Workshop on Deep Learning, pp. 1–9.
- Chung, J., K. Kastner, L. Dinh, K. Goel, A. C. Courville, and Y. Bengio (2015)
  'A recurrent latent variable model for sequential data'
  Advances in Neural Information Processing Systems, pp. 2980–2988.

References V

- Dong, Z., B. Seybold, K. Murphy, and H. Bui (2020)
  'Collapsed amortized variational inference for switching nonlinear dynamical systems'
  International Conference on Machine Learning, pp. 2638–2647.
- Doucet, A. and A. M. Johansen (2009)
  A tutorial on particle filtering and smoothing: Fifteen years later
  Handbook of nonlinear filtering 12 (656-704), p. 3.
- Durbin, J. and S. J. Koopman (2012)
  Time series analysis by state space methods
  Vol. 38
  Oxford University Press.

References VI

Farnoosh, A., B. Azari, and S. Ostadabbas (2021)
'Deep Switching Auto-Regressive Factorization: Application to Time Series Forecasting'
Proceedings of the AAAI Conference on Artificial Intelligence
8, pp. 7394–7403.

Fox, E., E. B. Sudderth, M. I. Jordan, and A. S. Willsky (2009)
'Nonparametric Bayesian learning of switching linear dynamical systems'
Advances in Neural Information Processing Systems, pp. 457–464.

References VII

Fraccaro, M., S. K. Sønderby, U. Paquet, and O. Winther (2016)
'Sequential neural models with stochastic layers'
Advances in Neural Information Processing Systems, pp. 2199–2207.

François, P., G. Gauthier, and F. Godin (2014)
Optimal hedging when the underlying asset follows a regime-switching Markov process
European Journal of Operational Research 237 (1), pp. 312–322.

Geiger, D., T. Verma, and J. Pearl (1990)
Identifying independence in Bayesian networks
Networks 20 (5), pp. 507–534.

References VIII

Ghahramani, Z. and G. E. Hinton (2000)
Variational learning for switching state-space models
Neural Computation 12 (4), pp. 831–864.

Graves, A. (2013)
Generating Sequences With Recurrent Neural Networks
CoRR abs/1308.0850
arXiv: 1308.0850.

Hamilton, J. D. (1990)
Analysis of time series subject to changes in regime
Journal of Econometrics 45 (1-2), pp. 39–70.

References IX

Hochreiter, S. and J. Schmidhuber (1997)
Long short-term memory
Neural Computation 9 (8), pp. 1735–1780.

Johnson, M., D. K. Duvenaud, A. Wiltschko, R. P. Adams, and S. R. Datta (2016)
'Composing graphical models with neural networks for structured representations and fast inference'
Advances in Neural Information Processing Systems, pp. 2946–2954.

References X

Julier, S. J. and J. K. Uhlmann (1997)
'New extension of the Kalman filter to nonlinear systems'
Signal processing, sensor fusion, and target recognition VI
Vol. 3068
International Society for Optics and Photonics, pp. 182–194.

Kingma, D. P. and M. Welling (2014)
'Auto-Encoding Variational Bayes'
International Conference on Learning Representations, pp. 1–9.

References XI

Kurle, R., S. S. Rangapuram, E. de Bézenac, S. Günnemann, and J. Gasthaus (2020)
'Deep Rao-Blackwellised Particle Filters for Time Series Forecasting'
Advances in Neural Information Processing Systems, pp. 15371–15382.

Li, S., X. Jin, Y. Xuan, X. Zhou, W. Chen, Y.-X. Wang, and X. Yan (2019)
'Enhancing the locality and breaking the memory bottleneck of transformer on time series forecasting'
Advances in Neural Information Processing Systems, pp. 5243–5253.

References XII

Linderman, S., M. Johnson, A. Miller, R. Adams, D. Blei, and L. Paninski (2017)
'Bayesian learning and inference in recurrent switching linear dynamical systems'
Artificial Intelligence and Statistics, pp. 914–922.

Murphy, K. P. (1998)
Switching kalman filters
.

References XIII

Nassar, J., S. Linderman, M. Bugallo, and I. M. Park (2018)
'Tree-Structured Recurrent Switching Linear Dynamical Systems for Multi-Scale Modeling'
International Conference on Learning Representations, pp. 1–9.

Rezende, D. J., S. Mohamed, and D. Wierstra (2014)
'Stochastic Backpropagation and Approximate Inference in Deep Generative Models'
International Conference on Machine Learning, pp. 1278–1286.

References XIV

Salinas, D., V. Flunkert, J. Gasthaus, and T. Januschowski (2020)
DeepAR: Probabilistic forecasting with autoregressive recurrent networks
International Journal of Forecasting 36 (3), pp. 1181–1191.

Sen, R., H.-F. Yu, and I. S. Dhillon (2019)
'Think globally, act locally: A deep neural network approach to high-dimensional time series forecasting'
Advances in Neural Information Processing Systems, pp. 4837–4846.

References XV

Smith, G. L., S. F. Schmidt, and L. A. McGee (1962)
Application of statistical filter theory to the optimal estimation of position and velocity on board a circumlunar vehicle
.

Williams, R. J. (1992)
Simple statistical gradient-following algorithms for connectionist reinforcement learning
Machine Learning 8 (3-4), pp. 229–256.

Xu, X., Y. Chen, Y. Goude, and Q. Yao (2021)
Probabilistic Forecasting for Daily Electricity Loads and Quantiles for Curve-to-Curve Regression
Applied Energy 301, pp. 1–14.

References XVI

Zhang, G. P. and M. Qi (2005)
Neural network forecasting for seasonal and trend time series
European Journal of Operational Research 160 (2), pp. 501–514.

Institute of Mathematics for Industry: its uniqueness, strength and prospects

Osamu SAEKI

Institute of Mathematics for Industry, Kyushu University, Japan
saeki@imi.kyushu-u.ac.jp

As director of the Institute of Mathematics for Industry (IMI), Kyushu University, which is a unique institute of industrial mathematics in Japan, the speaker will present various activities of IMI which characterize its strength; researches in fundamental mathematics, mathematics applied to other disciplines, joint projects with industry, together with educational activities, including the WISE program funded by the Japanese government. Some prospects of IMI for the future will also be presented.

References

[1] Institute of Mathematics for Industry, Kyushu University, An Unparalleled Research Institute for Industrial Mathematics Based on Diverse Fields of Mathematical Study, official homepage, https://www.imi.kyushu-u.ac.jp/en/

Institute of Mathematics for Industry: its uniqueness, strength and prospects

Osamu Saeki

Director, Institute of Mathematics for Industry (= IMI)

Kyushu University, Fukuoka, Japan

The 6th RIKEN-IMI-ISM-NUS-ZIB-MODAL-NHR Workshop on Advances in Classical and Quantum Algorithms for Optimization and Machine Learning

September 21, 2022

九州大学 KYUSHU UNIVERSITY

Institute of Mathematics for Industry Kyushu University

Institute of Mathematics for Industry Kyushu University Since 2011

MEXT Joint Use / Research Center
Joint Research Center for Advanced and Fundamental Mathematics-for-Industry

Institute of Mathematics for Industry (IMI)

Unparalleled industrial mathematics research institute with a foundation in diverse mathematical research

Math-for-Industry: A new area of research in mathematics that is valuable as mathematics itself as well, responding to the demands of society and industry, by reorganizing pure and applied mathematics.

Joint Use / Research Center

- Workshops (I),(II)
- Short-term joint research
- Short-term visiting researcher

×

- General Research
- Project Research
- International Project Research
- Young Researcher
- Female Researcher

Participation from industry is essential.

Joint researches with Industry and social implementations

Joint Research	Commissioned research
25	15

2021

Research Achievements

First in the world ranking in the Graph500 benchmark test using the supercomputer Fugaku (total of 15 terms, 5 consecutive victories).

International Activities

IMI Australia Branch

Asia-Pacific Consortium of Mathematics for Industry

APCMfI

Forum "Math-for-Industry"

Publications

International Journal of Mathematics for Industry

12 volumes published

Springer Series Mathematics for Industry

31 volumes published

Education

Graduate School of Mathematics

Graduate Program of Mathematics for Innovation

Long-term research internship (2006--)

Study Group Workshops (2010--)

International Activities of IMI

Forum "Math-for-Industry" **2008 –**

Asia-Pacific Consortium of Mathematics for Industry (APCMfI)

Since October 2014

FMI2010	FMI2011	FMI2012	FMI2013	FMI2014	FMI2015	FMI2016	FMI2017	FMI2018	FMI2019	FMI2021	FMI2022
Fukuoka	Honolulu, US	Fukuoka	Fukuoka	Fukuoka	Fukuoka	Brisbane, AU	Honolulu, US	Fudan Univ., China	Massey Univ., NZ	Vietnam Institute for Advanced Study in Mathematics	La Trobe Univ.
Oct.21-23	Oct.24-28	Oct.22-26	Nov. 4-8	Oct.27-31	Oct.26-30	Nov.21-23	Oct.23-26	Nov. 17-21	Nov. 18-21	Dec. 13-16	Nov. 21-24
Information Security, Visualization, and Inverse Problems, on the basis of Optimization Techniques	TSUNAMI - Mathematical Modelling Using Mathematics for Natural Disaster Prediction, Recovery and Provision for the future -	Information Recovery and Discovery	The Impact of Applications on Mathematics	Applications + Practical Conceptualization + Mathematics = fruitful Innovation	The Role and Importance of Mathematics in Innovation	Agriculture as a metaphor for creativity in all human endeavors	Responding to the Challenges of Climate Change: Exploiting, Harnessing and Enhancing the Opportunities of Clean Energy	Big Data Analysis, AI, Fintech, Math in Finances and Economics	Mathematics for the Primary Industries and the Environment	Mathematics for Digital Economy	Mathematics of Public Health and Sustainability

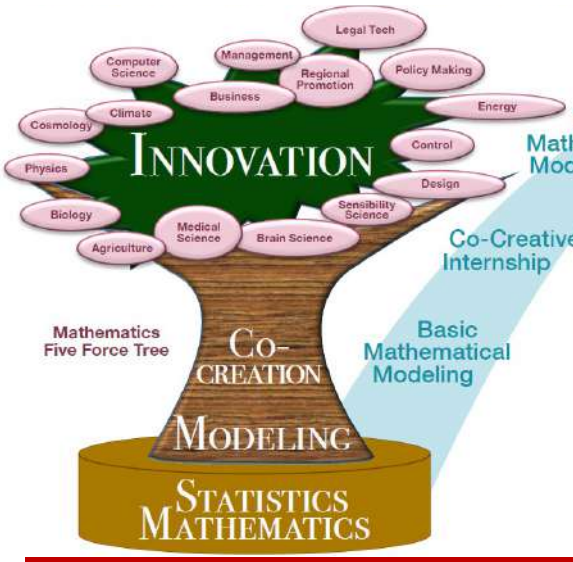
IMI Australia Branch at Melbourne

GPMI Graduate Program of Mathematics for Innovation

Nurturing Mathematics Talents Active in Society

Cross-Disciplinary Program Nurturing Advanced Mathematical Modeling Talents Endowed with Mathematics Five Forces

Graduate Program of Mathematics for Innovation



Advanced Mathematical Modeling

Co-Creative Internship

Basic Mathematical Modeling

RIKEN, ISM, NUS, ZIB also participate in this WISE Program!

Math-for-Innovation Professionals

Outstanding Mathematics Talents who lead innovations in both academia and industry by creating mathematical modeling under collaboration crossing the boundaries between organizations or disciplines based on internationally excellent abilities in Mathematics and Statistics

New cross-disciplinary education system that leads to Graduate School Reform

The most important graduate program of Kyushu University

WISE Program supported by MEXT, Japan

MI レクチャーノートシリーズ刊行にあたり

本レクチャーノートシリーズは、文部科学省 21 世紀 COE プログラム「機能数理学の構築と展開」（H15-19 年度）において作成した COE Lecture Notes の続刊であり、文部科学省大学院教育改革支援プログラム「産業界が求める数学博士と新修士養成」（H19-21 年度）および、同グローバル COE プログラム「マス・フォア・インダストリ教育研究拠点」（H20-24 年度）において行われた講義の講義録として出版されてきた。平成 23 年 4 月のマス・フォア・インダストリ研究所（IMI）設立と平成 25 年 4 月の IMI の文部科学省共同利用・共同研究拠点として「産業数学の先進的･基礎的共同研究拠点」の認定を受け、今後、レクチャーノートは、マス・フォア・インダストリに関わる国内外の研究者による講義の講義録、会議録等として出版し、マス・フォア・インダストリの本格的な展開に資するものとする。

2022 年 10 月
マス・フォア・インダストリ研究所
所長　梶原 健司

IMI Workshop of the Joint Usage Research Projects

Construction of Mathematical Basis for Realizing Data Rating Service

発　行　2022年12月27日
編　集　Katsuki Fujisawa, Shizuo Kaji, Toru Ishihara, Masaaki Kondo, Yuji Shinano, Takuji Tanigawa, Naoko Nakayama
発　行　九州大学マス・フォア・インダストリ研究所
九州大学大学院数理学府
〒819-0395 福岡市西区元岡744
九州大学数理・IMI 事務室
TEL 092-802-4402　FAX 092-802-4405
URL https://www.imi.kyushu-u.ac.jp/

印　刷　城島印刷株式会社
〒810-0012 福岡市中央区白金 2 丁目 9 番 6 号
TEL 092-531-7102　FAX 092-524-4411

シリーズ既刊

Issue	Author／Editor	Title	Published
COE Lecture Note	Mitsuhiro T. NAKAO Kazuhiro YOKOYAMA	Computer Assisted Proofs - Numeric and Symbolic Approaches - 199pages	August 22, 2006
COE Lecture Note	M.J.Shai HARAN	Arithmetical Investigations - Representation theory, Orthogonal polynomials and Quantum interpolations- 174pages	August 22, 2006
COE Lecture Note Vol.3	Michal BENES Masato KIMURA Tatsuyuki NAKAKI	Proceedings of Czech-Japanese Seminar in Applied Mathematics 2005 155pages	October 13, 2006
COE Lecture Note Vol.4	宮田　健治	辺要素有限要素法による磁界解析 - 機能数理学特別講義 21pages	May 15, 2007
COE Lecture Note Vol.5	Francois APERY	Univariate Elimination Subresultants - Bezout formula, Laurent series and vanishing conditions - 89pages	September 25, 2007
COE Lecture Note Vol.6	Michal BENES Masato KIMURA Tatsuyuki NAKAKI	Proceedings of Czech-Japanese Seminar in Applied Mathematics 2006 209pages	October 12, 2007
COE Lecture Note Vol.7	若山　正人 中尾　充宏	九州大学産業技術数理研究センター キックオフミーティング 138pages	October 15, 2007
COE Lecture Note Vol.8	Alberto PARMEGGIANI	Introduction to the Spectral Theory of Non-Commutative Harmonic Oscillators 233pages	January 31, 2008
COE Lecture Note Vol.9	Michael I.TRIBELSKY	Introduction to Mathematical modeling 23pages	February 15, 2008
COE Lecture Note Vol.10	Jacques FARAUT	Infinite Dimensional Spherical Analysis 74pages	March 14, 2008
COE Lecture Note Vol.11	Gerrit van DIJK	Gelfand Pairs And Beyond 60pages	August 25, 2008
COE Lecture Note Vol.12	Faculty of Mathematics, Kyushu University	Consortium “MATH for INDUSTRY” First Forum 87pages	September 16, 2008
COE Lecture Note Vol.13	九州大学大学院 数理学研究院	プロシーディング「損保数理に現れる確率モデル」 ― 日新火災・九州大学 共同研究2008年11月 研究会 ― 82pages	February 6, 2009

シリーズ既刊

Issue	Author／Editor	Title	Published
COE Lecture Note Vol.14	Michal Beneš, Tohru Tsujikawa Shigetoshi Yazaki	Proceedings of Czech-Japanese Seminar in Applied Mathematics 2008 77pages	February 12, 2009
COE Lecture Note Vol.15	Faculty of Mathematics, Kyushu University	International Workshop on Verified Computations and Related Topics 129pages	February 23, 2009
COE Lecture Note Vol.16	Alexander Samokhin	Volume Integral Equation Method in Problems of Mathematical Physics 50pages	February 24, 2009
COE Lecture Note Vol.17	矢嶋　　徹 及川　正行 梶原　健司 辻　　英一 福本　康秀	非線形波動の数理と物理 66pages	February 27, 2009
COE Lecture Note Vol.18	Tim Hoffmann	Discrete Differential Geometry of Curves and Surfaces 75pages	April 21, 2009
COE Lecture Note Vol.19	Ichiro Suzuki	The Pattern Formation Problem for Autonomous Mobile Robots —Special Lecture in Functional Mathematics— 23pages	April 30, 2009
COE Lecture Note Vol.20	Yasuhide Fukumoto Yasunori Maekawa	Math-for-Industry Tutorial: Spectral theories of non-Hermitian operators and their application 184pages	June 19, 2009
COE Lecture Note Vol.21	Faculty of Mathematics, Kyushu University	Forum “Math-for-Industry” Casimir Force, Casimir Operators and the Riemann Hypothesis 95pages	November 9, 2009
COE Lecture Note Vol.22	Masakazu Suzuki Hoon Hong Hirokazu Anai Chee Yap Yousuke Sato Hiroshi Yoshida	The Joint Conference of ASCM 2009 and MACIS 2009: Asian Symposium on Computer Mathematics Mathematical Aspects of Computer and Information Sciences 436pages	December 14, 2009
COE Lecture Note Vol.23	荒川　恒男 金子　昌信	多重ゼータ値入門 111pages	February 15, 2010
COE Lecture Note Vol.24	Fulton B.Gonzalez	Notes on Integral Geometry and Harmonic Analysis 125pages	March 12, 2010
COE Lecture Note Vol.25	Wayne Rossman	Discrete Constant Mean Curvature Surfaces via Conserved Quantities 130pages	May 31, 2010
COE Lecture Note Vol.26	Mihai Ciucu	Perfect Matchings and Applications 66pages	July 2, 2010

シリーズ既刊

Issue	Author／Editor	Title	Published
COE Lecture Note Vol.27	九州大学大学院 数理学研究院	Forum “Math-for-Industry” and Study Group Workshop Information security, visualization, and inverse problems, on the basis of optimization techniques　100pages	October 21, 2010
COE Lecture Note Vol.28	ANDREAS LANGER	MODULAR FORMS, ELLIPTIC AND MODULAR CURVES LECTURES AT KYUSHU UNIVERSITY 2010　62pages	November 26, 2010
COE Lecture Note Vol.29	木田　雅成 原田　昌晃 横山　俊一	Magma で広がる数学の世界　157pages	December 27, 2010
COE Lecture Note Vol.30	原　　隆 松井　　卓 廣島　文生	Mathematical Quantum Field Theory and Renormalization Theory 201pages	January 31, 2011
COE Lecture Note Vol.31	若山　正人 福本　康秀 高木　　剛 山本　昌宏	Study Group Workshop 2010 Lecture & Report　128pages	February 8, 2011
COE Lecture Note Vol.32	Institute of Mathematics for Industry, Kyushu University	Forum “Math-for-Industry” 2011 “TSUNAMI-Mathematical Modelling” Using Mathematics for Natural Disaster Prediction, Recovery and Provision for the Future　90pages	September 30, 2011
COE Lecture Note Vol.33	若山　正人 福本　康秀 高木　　剛 山本　昌宏	Study Group Workshop 2011 Lecture & Report　140pages	October 27, 2011
COE Lecture Note Vol.34	Adrian Muntean Vladimír Chalupecký	Homogenization Method and Multiscale Modeling　72pages	October 28, 2011
COE Lecture Note Vol.35	横山　俊一 夫　　紀恵 林　　卓也	計算機代数システムの進展　210pages	November 30, 2011
COE Lecture Note Vol.36	Michal Beneš Masato Kimura Shigetoshi Yazaki	Proceedings of Czech-Japanese Seminar in Applied Mathematics 2010 107pages	January 27, 2012
COE Lecture Note Vol.37	若山　正人 高木　　剛 Kirill Morozov 平岡　裕章 木村　正人 白井　朋之 西井　龍映 栄　伸一郎 穴井　宏和 福本　康秀	平成23年度 数学・数理科学と諸科学・産業との連携研究ワークショップ　拡がっていく数学　～期待される “見えない力” ～ 154pages	February 20, 2012

シリーズ既刊

Issue	Author／Editor	Title	Published
COE Lecture Note Vol.38	Fumio Hiroshima Itaru Sasaki Herbert Spohn Akito Suzuki	Enhanced Binding in Quantum Field Theory　204pages	March 12, 2012
COE Lecture Note Vol.39	Institute of Mathematics for Industry, Kyushu University	Multiscale Mathematics; Hierarchy of collective phenomena and interrelations between hierarchical structures　180pages	March 13, 2012
COE Lecture Note Vol.40	井ノ口順一 太田　泰広 筧　　三郎 梶原　健司 松浦　　望	離散可積分系・離散微分幾何チュートリアル2012　152pages	March 15, 2012
COE Lecture Note Vol.41	Institute of Mathematics for Industry, Kyushu University	Forum “Math-for-Industry” 2012 “Information Recovery and Discovery”　91pages	October 22, 2012
COE Lecture Note Vol.42	佐伯　　修 若山　正人 山本　昌宏	Study Group Workshop 2012 Abstract, Lecture & Report　178pages	November 19, 2012
COE Lecture Note Vol.43	Institute of Mathematics for Industry, Kyushu University	Combinatorics and Numerical Analysis Joint Workshop　103pages	December 27, 2012
COE Lecture Note Vol.44	萩原　　学	モダン符号理論からポストモダン符号理論への展望　107pages	January 30, 2013
COE Lecture Note Vol.45	金山　　寛	Joint Research Workshop of Institute of Mathematics for Industry (IMI), Kyushu University “Propagation of Ultra-large-scale Computation by the Domain-decomposition-method for Industrial Problems (PUCDIP 2012)” 121pages	February 19, 2013
COE Lecture Note Vol.46	西井　龍映 栄　伸一郎 岡田　勘三 落合　啓之 小磯　深幸 斎藤　新悟 白井　朋之	科学・技術の研究課題への数学アプローチ ―数学モデリングの基礎と展開―　325pages	February 28, 2013
COE Lecture Note Vol.47	SOO TECK LEE	BRANCHING RULES AND BRANCHING ALGEBRAS FOR THE COMPLEX CLASSICAL GROUPS　40pages	March 8, 2013
COE Lecture Note Vol.48	溝口　佳寛 脇　　隼人 平坂　　貢 谷口　哲至 島袋　　修	博多ワークショップ「組み合わせとその応用」 124pages	March 28, 2013

シリーズ既刊

Issue	Author／Editor	Title	Published
COE Lecture Note Vol.49	照井　　章 小原　功任 濱田　龍義 横山　俊一 穴井　宏和 横田　博史	マス・フォア・インダストリ研究所　共同利用研究集会 II 数式処理研究と産学連携の新たな発展　137pages	August 9, 2013
MI Lecture Note Vol.50	Ken Anjyo Hiroyuki Ochiai Yoshinori Dobashi Yoshihiro Mizoguchi Shizuo Kaji	Symposium MEIS2013: Mathematical Progress in Expressive Image Synthesis　154pages	October 21, 2013
MI Lecture Note Vol.51	Institute of Mathematics for Industry, Kyushu University	Forum “Math-for-Industry” 2013 “The Impact of Applications on Mathematics”　97pages	October 30, 2013
MI Lecture Note Vol.52	佐伯　　修 岡田　勘三 髙木　　剛 若山　正人 山本　昌宏	Study Group Workshop 2013 Abstract, Lecture & Report 142pages	November 15, 2013
MI Lecture Note Vol.53	四方　義啓 櫻井　幸一 安田　貴徳 Xavier Dahan	平成25年度　九州大学マス・フォア・インダストリ研究所 共同利用研究集会　安全・安心社会基盤構築のための代数構造 ～サイバー社会の信頼性確保のための数理学～　158pages	December 26, 2013
MI Lecture Note Vol.54	Takashi Takiguchi Hiroshi Fujiwara	Inverse problems for practice, the present and the future　93pages	January 30, 2014
MI Lecture Note Vol.55	栄　伸一郎 溝口　佳寛 脇　　隼人 渋田　敬史	Study Group Workshop 2013 数学協働プログラム Lecture & Report 98pages	February 10, 2014
MI Lecture Note Vol.56	Yoshihiro Mizoguchi Hayato Waki Takafumi Shibuta Tetsuji Taniguchi Osamu Shimabukuro Makoto Tagami Hirotake Kurihara Shuya Chiba	Hakata Workshop 2014 ~ Discrete Mathematics and its Applications ~　141pages	March 28, 2014
MI Lecture Note Vol.57	Institute of Mathematics for Industry, Kyushu University	Forum “Math-for-Industry” 2014: “Applications + Practical Conceptualization + Mathematics = fruitful Innovation”　93pages	October 23, 2014
MI Lecture Note Vol.58	安生健一 落合啓之	Symposium MEIS2014: Mathematical Progress in Expressive Image Synthesis　135pages	November 12, 2014

シリーズ既刊

Issue	Author／Editor	Title	Published
MI Lecture Note Vol.59	西井　龍映 岡田　勘三 梶原　健司 髙木　　剛 若山　正人 脇　　隼人 山本　昌宏	Study Group Workshop 2014 数学協働プログラム Abstract, Lecture & Report　196pages	November 14, 2014
MI Lecture Note Vol.60	西浦　　博	平成26年度九州大学 IMI 共同利用研究・研究集会（I） 感染症数理モデルの実用化と産業及び政策での活用のための新たな展開　120pages	November 28, 2014
MI Lecture Note Vol.61	溝口　佳寛 Jacques Garrigue 萩原　　学 Reynald Affeldt	研究集会 高信頼な理論と実装のための定理証明および定理証明器 Theorem proving and provers for reliable theory and implementations (TPP2014)　138pages	February 26, 2015
MI Lecture Note Vol.62	白井　朋之	Workshop on “β-transformation and related topics”　59pages	March 10, 2015
MI Lecture Note Vol.63	白井　朋之	Workshop on “Probabilistic models with determinantal structure” 107pages	August 20, 2015
MI Lecture Note Vol.64	落合　啓之 土橋　宜典	Symposium MEIS2015: Mathematical Progress in Expressive Image Synthesis　124pages	September 18, 2015
MI Lecture Note Vol.65	Institute of Mathematics for Industry, Kyushu University	Forum “Math-for-Industry” 2015 “The Role and Importance of Mathematics in Innovation”　74pages	October 23, 2015
MI Lecture Note Vol.66	岡田　勘三 藤澤　克己 白井　朋之 若山　正人 脇　　隼人 Philip Broadbridge 山本　昌宏	Study Group Workshop 2015 Abstract, Lecture & Report 156pages	November 5, 2015
MI Lecture Note Vol.67	Institute of Mathematics for Industry, Kyushu University	IMI-La Trobe Joint Conference “Mathematics for Materials Science and Processing” 66pages	February 5, 2016
MI Lecture Note Vol.68	古庄　英和 小谷　久寿 新甫　洋史	結び目と Grothendieck-Teichmüller 群 116pages	February 22, 2016
MI Lecture Note Vol.69	土橋　宜典 鍛治　静雄	Symposium MEIS2016: Mathematical Progress in Expressive Image Synthesis　82pages	October 24, 2016
MI Lecture Note Vol.70	Institute of Mathematics for Industry, Kyushu University	Forum “Math-for-Industry” 2016 “Agriculture as a metaphor for creativity in all human endeavors” 98pages	November 2, 2016
MI Lecture Note Vol.71	小磯　深幸 二宮　嘉行 山本　昌宏	Study Group Workshop 2016 Abstract, Lecture & Report　143pages	November 21, 2016

シリーズ既刊

Issue	Author／Editor	Title	Published
MI Lecture Note Vol.72	新井　朝雄 小嶋　　泉 廣島　文生	Mathematical quantum field theory and related topics　133pages	January 27, 2017
MI Lecture Note Vol.73	穴田　啓晃 Kirill Morozov 須賀　祐治 奥村　伸也 櫻井　幸一	Secret Sharing for Dependability, Usability and Security of Network Storage and Its Mathematical Modeling　211pages	March 15, 2017
MI Lecture Note Vol.74	QUISPEL, G. Reinout W. BADER, Philipp MCLAREN, David I. TAGAMI, Daisuke	IMI-La Trobe Joint Conference Geometric Numerical Integration and its Applications　71pages	March 31, 2017
MI Lecture Note Vol.75	手塚　　集 田上　大助 山本　昌宏	Study Group Workshop 2017 Abstract, Lecture & Report　118pages	October 20, 2017
MI Lecture Note Vol.76	宇田川誠一	Tzitzéica 方程式の有限間隙解に付随した極小曲面の構成理論 ―Tzitzéica 方程式の楕円関数解を出発点として―　68pages	August 4, 2017
MI Lecture Note Vol.77	松谷　茂樹 佐伯　　修 中川　淳一 田上　大助 上坂　正晃 Pierluigi Cesana 濵田　裕康	平成29年度　九州大学マス・フォア・インダストリ研究所 共同利用研究集会（I） 結晶の界面，転位，構造の数理　148pages	December 20, 2017
MI Lecture Note Vol.78	瀧澤　重志 小林　和博 佐藤憲一郎 斎藤　　努 清水　正明 間瀬　正啓 藤澤　克樹 神山　直之	平成29年度　九州大学マス・フォア・インダストリ研究所 プロジェクト研究　研究集会（I） 防災・避難計画の数理モデルの高度化と社会実装へ向けて 136pages	February 26, 2018
MI Lecture Note Vol.79	神山　直之 畔上　秀幸	平成29年度　AIMaP チュートリアル 最適化理論の基礎と応用　96pages	February 28, 2018
MI Lecture Note Vol.80	Kirill Morozov Hiroaki Anada Yuji Suga	IMI Workshop of the Joint Research Projects Cryptographic Technologies for Securing Network Storage and Their Mathematical Modeling　116pages	March 30, 2018
MI Lecture Note Vol.81	Tsuyoshi Takagi Masato Wakayama Keisuke Tanaka Noboru Kunihiro Kazufumi Kimoto Yasuhiko Ikematsu	IMI Workshop of the Joint Research Projects International Symposium on Mathematics, Quantum Theory, and Cryptography　246pages	September 25, 2019
MI Lecture Note Vol.82	池森　俊文	令和2年度　AIMaP チュートリアル 新型コロナウイルス感染症にかかわる諸問題の数理 145pages	March 22, 2021

シリーズ既刊

Issue	Author／Editor	Title	Published
MI Lecture Note Vol.83	早川健太郎 軸丸　芳揮 横須賀洋平 可香谷　隆 林　　和希 堺　　雄亮	シェル理論・膜理論への微分幾何学からのアプローチと その建築曲面設計への応用　49pages	July 28, 2021
MI Lecture Note Vol.84	Taketoshi Kawabe Yoshihiro Mizoguchi Junichi Kako Masakazu Mukai Yuji Yasui	SICE-JSAE-AIMaP Tutorial Advanced Automotive Control and Mathematics　110pages	December 27, 2021
MI Lecture Note Vol.85	Hiroaki Anada Yasuhiko Ikematsu Koji Nuida Satsuya Ohata Yuntao Wang	IMI Workshop of the Joint Usage Research Projects Exploring Mathematical and Practical Principles of Secure Computation and Secret Sharing　114pages	February 9, 2022
MI Lecture Note Vol.86	濱田　直希 穴井　宏和 梅田　裕平 千葉　一永 佐藤　寛之 能島　裕介 加葉田雄太朗 一木　俊助 早野　健太 佐伯　　修	2020年度採択分　九州大学マス・フォア・インダストリ研究所 共同利用研究集会 進化計算の数理　135pages	February 22, 2022
MI Lecture Note Vol.87	Osamu Saeki, Ho Tu Bao, Shizuo Kaji, Kenji Kajiwara, Nguyen Ha Nam, Ta Hai Tung, Melanie Roberts, Masato Wakayama, Le Minh Ha, Philip Broadbridge	Proceedings of Forum “Math-for-Industry” 2021 -Mathematics for Digital Economy-　122pages	March 28, 2022
MI Lecture Note Vol.88	Daniel PACKWOOD Pierluigi CESANA, Shigenori FUJIKAWA, Yasuhide FUKUMOTO, Petros SOFRONIS, Alex STAYKOV	Perspectives on Artificial Intelligence and Machine Learning in Materials Science, February 4-6, 2022　74pages	November 8, 2022

MI Lecture Note Vol.89	松谷　茂樹 落合　啓之 井上　和俊 小磯　深幸 佐伯　　修 白井　朋之 垂水　竜一 内藤　久資 中川　淳一 濵田　裕康 松江　　要 加葉田雄太朗	2020年度採択分 九州大学マス・フォア・インダストリ研究所 共同利用研究集会 材料科学における幾何と代数 III　356pages	December 7, 2022
MI Lecture Note Vol.90	中山　尚子 谷川　拓司 品野　勇治 近藤　正章 石原　　亨 鍛冶　静雄 藤澤　克樹	2022年度採択分 九州大学マス・フォア・インダストリ研究所 共同利用研究集会 データ格付けサービス実現のための数理基盤の構築　58pages	December 12, 2022

Institute of Mathematics for Industry
Kyushu University

九州大学マス・フォア・インダストリ研究所
九州大学大学院 数理学府
〒819-0395 福岡市西区元岡744　TEL 092-802-4402　FAX 092-802-4405
URL http://www.imi.kyushu-u.ac.jp/