平成27年度 共同利用研究報告書

平成 28 年 2 月 10 日

九州大学 マス・フォア・インダストリ研究所長 殿

所属・職名 (株)本田技術研究所・主任研究員 提案者 氏 名 佐藤 正浩

下記の通り共同研究の報告をいたします。

記

		※整理番号 201500		2015000	50004		
1.研究計画題目	数学・数理科学の視点からのエンジン適合の研究						
2.種別(○で囲む)	a. 研究	集会 I	b. 研究集会Ⅱ	②短期#	© 短期共同研究		d.短期研究員
	民 第	佐藤 正浩					
	所 属	(株)本田技術研究所				職	主任研究員
3.研究代表者	部局名	(株)本田技術研究別				名	工任切九貝
	連絡先	〒					
	e-mail				TEL		
4.研究実施期間	第一回]: 平成 27 年 6 月 17 日 (水曜日) ~平成 27 年 6 月 18 日 (木曜日)					
	第二回	: 平成 27	平成 27 年 11 月 27 日 (金曜日)				

5.参加者数・参加者リスト(*別紙「共同利用研究報告書作成上の注意」参照)

(aおよびbの場合、参加者数のみ記入し、集会参加者リストを添付。Cおよびd.の場合は下記欄に記入。)

参加者数: (第一回) 27人, (第二回) 28人 (参加者リストを別紙に添付)

氏名	所属	職名	(ふりがな) 氏 名	所属	職名

6.本研究で得られた成果の概要

この IMI 短期共同研究は、H26 年度に実施した IMI 短期共同研究「境界モデル手法の研究」の続きである。 6/17、18 に開催した第一回 IMI 短期共同研究では、それまで行っていた研究の発表や参加企業が抱えている、関連する問題について議論された。例えば、前回議論にあがったロジステック回帰の利用だけでなく、サポートベクターマシンの適用について議論された。また、エンジンの動的特性を考慮した境界モデリングなども提案された。これらの議論・提案内容をもとに、SGW や 11 月 28 日に開催された第二回 IMI 短期共同研究で検討した。また、第 58 回自動制御連合講演会(神戸大学)やワークショップ「自動車業界における IT・数理科学技術の活用 — 豊かな社会を作り出すイノベーションを目指して—」において、この IMI 短期共同研究の成果の宣伝・議論を行った。以上の議論から、以下の成果を得た:

- (1) エンジンシミュレータで生成したデータをもとに、ロジステック回帰とサポートベクターマシンの比較を行った. どちらの手法が優位か、というところまで検討は出来ていない. また正則化パラメータの選択によって、境界が大きく変化する傾向もあり、より綿密な分析が必要である.
- (2) エンジンシミュレータで生成したデータに、ある程度の許容範囲を与えて回帰分析を行った。これにより、シミュレータで生成された失火データに対する境界は、4変数で次数3の多項式で記述できることがわかった。また、正則化項つきロジステック回帰では基底関数を単項式に選ぶことで、4変数で次数4の多項式で概ね記述できることが後の数値実験で明らかになった。
- (3) 凸包を利用した境界モデリングにおいて、高速に計算する手法を提案した.これにより、9 入力の境界モデリングでは、この計算が従来のソフトウェアに実装されている計算よりも高速に計算し、さらに質の良い解を返すことがわかった.

平成27年度短期共同研究 「数学・数理科学の視点からのエンジン適合の研究」の報告書

研究代表者:佐藤 正浩 (本田技術研究所)

1 短期共同研究の実施概要

申請時にも記載した通り、議論する時間を確保するために 2 回に分けさらに IMI のイベントと組み合わせることで下記の通り実施した.

- 1. 数学・数理科学の視点からのエンジン適合の研究 (第1回)
 - 日時: 平成27年6月17日(水)13:30 18日(木)17:00
 - 場所: 九州大学 伊都キャンパス 数理学研究教育棟3階中セミナー室2
 - プログラム:表1を参照

表 1: 数学・数理科学の視点からのエンジン適合の研究 (第1回) のプログラム

時間	講演者	講演タイトル
17日 13:30 - 14:30	脇 隼人 (九大 IMI)	前回までの議論の復習
17 日 13:30 - 15:00	岩瀬 将美, 永長 克章, 芹澤 卓哉 (東	ガソリンエンジンの動的モデリング
	京電機大学)	の概要と境界探索の必要性
17 日 15:30 - 17:00	大畠 明 (トヨタ自動車株式会社)	動的境界モデルの考察
18 日 10:30 - 11:00	渡邊 智 (トヨタ自動車株式会社)	効率の良い凸包の計算について
18 日 13:00 - 14:00	脇 隼人, 落合 啓之 (九大 IMI)	SGW2014 のデータに対する考察
18 日 14:00 - 14:15	渡邊 智 (トヨタ自動車株式会社)	効率の良い凸包の計算について
18 日 13:00 - 14:00	落合 啓之 (九大 IMI)	依存しない変数に関するコメント
18 日 14:30 - 15:30	佐藤 正浩, 下城 孝名子 (ホンダ技術	SGW2014で用いたデータの境界探索
	研究所)	法
18 日 15:30 - 16:30	参加者全員	講演内容や今後の方針について

- 参加者数:大学(教員12名,学生8名),企業(7名),計27名
- 2. Study Group Workshop 2015

「Boundary model identification for automotive engine control systems II 」というタイトルで、大畠氏 (トヨタ自動車) と下城氏 (本田技術研究所) に問題提供をしていただき、境界モデルの数学的定式化と境界モデル作成の数学・数理科学技術について議論した.

3. 第 58 回自動制御連合講演会

セッション「OS 自動車エンジン制御の現状」を提案し、自動車エンジン制御に携わっている企業・大学関係者に講演を依頼した。このセッションでは脇も IMI 短期共同研究について議論・情報交換を行った。

4. 数学・数理科学の視点からのエンジン適合の研究 (第2回)

- 日時: 平成27年11月27日(金)13:30 19:00
- 場所: 九州大学 伊都キャンパス ウエスト 1 号館 D 棟 4 階 IMI コンファレンスルーム (W1-D-414)
- プログラム:表2を参照

表 2: 数学・数理科学の視点からのエンジン適合の研究 (第2回) のプログラム

時間	講演者	講演タイトル
27 日 10:30 - 11:30	大畠 明 (株式会社テクノバ)	境界モデルを使った制御設計
27 日 13:00 - 14:00	脇 隼人, 落合 啓之 (九州大学マス・	境界モデルに関する現状報告
	フォア・インダストリ 研究所), Florin	
	Nae (MathWorks Japan)	
27 日 14:30 - 15:30	三浦 達彦 (東京大学)	偏微分方程式を用いたデータ分離に
		ついて
27 日 16:00 - 17:00	芹澤 卓哉, 永長 克章, 岩瀬 将美 (東	境界モデリングのための実験データ
	京電機大学)	削減についての取り組み
27 日 17:00 - 19:00	参加者全員	IMI 短期共同研究のまとめと今後に
		ついて

- 参加者数:大学(教員9名,学生11名),企業(8名),計28名
- 5. ワークショップ「自動車業界における IT・数理科学技術の活用 豊かな社会を作り出すイノベーションを目指して –」

ワークショップを企画し, 自動車業界と関係しているセンターオブイノベーションや企業・大学関係者に講演を依頼した. このセッションでは脇も IMI 短期共同研究について議論・情報交換を行った.

2 成果

2.1 サポートベクターマシンの適用

サポートベクターマシンの適用を実施した.彼らのスライドの一部を図 1, 2, 3 に掲載する.去年度の IMI 短期共同研究で提案されたロジステック回帰との比較を実施したが、どちらが優位な手法なのかは判然としなかった.理由として、(i) 同一のデータであってもあ利用するソフトウェアの実装内容によって誤判別率が変化する、(ii) エンジンシミュレータのデータでは優位性を主張するほどの誤判別率が得られなかった、の 2 点が挙げられる.もう少し性質のことなるデータあるいは、実際の計測値を利用することで検討する必要があると考えている.

2.2 線形回帰による判別

エンジンシミュレータで生成したデータに対して、外れ値の除去などの前処理を行い、そのデータの特性を利用した線形回帰を行った.

入出力データの取得

提供されたMATLAB&Simlinkエンジンシミュレータを使用

入出力データ

入力: $u = [u_1, u_2, u_3, u_4, u_5], u \in \mathbb{R}^5$

 u_1 : Engine speed (rpm)

 u_2 : Intake manifold pressure (kPa)

 $u = \langle u_3 : \text{Spark angle (deg)} \rangle$

 u_4 : Valve timing (deg)

 u_5 : EGR rate (%)

出力: y = 0or1(判別指標值) №

目標の計画点まで進めノッキング 失火が起こるか否か判定

失火した点 u_3 スタートポイント Var1



Page 11 Nov. 27, 2015

境界モデル同定条件~カーネル関数の選択~

□ 入出力データ

入力: $u = [u_1, u_2, u_3, u_4, u_5], u \in \mathbb{R}^5$

出力: y = 0or1(判別指標値)

 u_1 : Engine speed (rpm)

[600.0, 1200.0, 1800.0, 2400.0, 3000.0]

 u_2 : Intake manifold pressure (kPa) [20.0, 37.5, 55.0, 72.5, 90.0]

 $u = \{u_3 : \text{Spark angle (deg)}\}\$

[-15.0, 0.0, 15.0, 30.0, 45.0, 60.0]

 u_4 : Valve timing (deg)

[0.0, 10.0, 20.0, 30.0, 40.0, 50.0, 60.0]

 u_5 : EGR rate (%)

[0.0, 10.0, 20.0, 30.0, 40.0, 50.0];

- □ 教師データに122点のデータを使用し、失火を検出
- □ 4種のカーネル関数により境界を同定
- □ 教師データに用いたデータの半分を用いて判別率を評価

TDU

Nov. 27, 2015 Page 16

境界同定結果~カーネル関数の選択~

□ u₁, u₂ をそれぞれ1800rpm,55.0kPaで固定した際の境界面

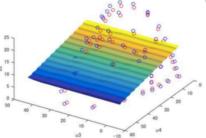


Fig.12 線形カーネル

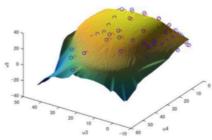


Fig.13 ガウシアンカーネル

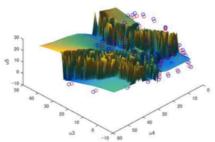


Fig.14 シグモイドカーネル

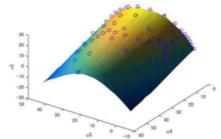


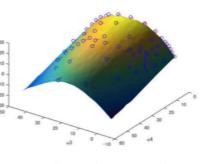
Fig.15 多項式カーネル

Page 20 Nov. 27, 2015

境界同定結果~カーネル関数の選択~

Table 教師データ100%使用時の正答数

教師データの割合	全25パターン の平均正答率 [%]	TN[個]
線形カーネル	57.8	19.8
ガウシアンカーネル	65.8	18.8
シグモイドカーネル	60.3	11.0
多項式カーネル(4次)	95.0	1.12



T:同定した境界上で発生

F: 同定した境界上で発生せず

P:実測データ上で発生

N:実測データ上で発生せず

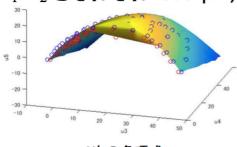
多項式カーネルを使用した際の正答率:高

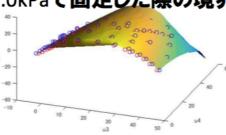
TDU

Page 21 Nov. 27, 2015

境界同定結果〜次数の選択〜

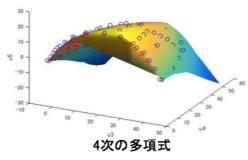
□ u₁, u₂ をそれぞれ1800rpm,55.0kPaで固定した際の境界面

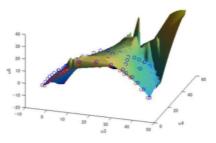




2次の多項式

3次の多項式





Page 25 Nov. 27, 2015

5次の多項式 TDU

境界同定結果~次数の選択~

Table 教師データ100%使用時の正答数

次数	全25パターンの 平均正答率[%]	TN[個]
2次	80.4	7.8
3次	89.1	3.8
4次	95.0	1.1
5 次	92.2	1.5

T:同定した境界上で発生

F:同定した境界上で発生せず

P:実測データ上で発生

N:実測データ上で発生せず

4次の多項式カーネルを使用した際の正答率:高

TDU

Page 26 Nov. 27, 2015

図 3: サポートベクターマシンの適用

エンジンシミュレータを利用したデータは以下の通りである: データは (u_1,\ldots,u_5,y) となっており、 (u_1,\ldots,u_5) が入力データ、 $y\in\{0,1\}$ が入力 (u_1,\ldots,u_5,y) に対してエンジンが正常に動作するかしないかの表す出力データである. データは、25 通りの (u_1,u_2) を生成し、各 (u_1,u_2) に対して 61 通りの方向 (d_3,d_4,d_5) を定め、 $\alpha\geq 0$ を少しずつ増加させて (u_3,u_4,u_5) を以下のように計算し、シミュレータ内のエンジンの挙動を判定する.

$$(u_3(\alpha), u_4(\alpha), u_5(\alpha)) = (15, 0, 0) + \alpha(d_3, d_4, d_5)$$

各方向に対して、エンジンが正常に動作する最大の α とエンジンが正常に動作しない最小の α を求めて、エンジンが正常に動作する入力と正常に動作しない入力のペアを求める。これで、 $25\times61\times2=3050$ のデータが生成される。

このように生成されたデータに対して、線形回帰を利用した以下のアルゴリズムを適用した:

Step 1 $y^i = 1$ となる要因を失火に限定 & 出力が 0, 1 のペアにならないものは削除

Step 2 u_1, u_2 を固定して、25 種類に分割して 3 次元プロット

- $\Rightarrow u_5$ が u_3 , u_4 の二次関数でかけている
- \Rightarrow 入力データを利用した二次関数の fitting. $u_5 \in \langle 1, u_3, u_4, u_3^2, u_3 u_4, u_4^2 \rangle$
- $\Rightarrow u_3u_4$ に関する係数が他よりも小さい ≈ 0

Step 3 Step 2 の fitting が良いことを確認する

Step 4 *u*₃, *u*₄ に関して平方完成 (*local* な推定式)

$$u_5 = c + a_3(u_3 + b_3)^2 + a_4(u_4 + b_4)^2$$

 \Rightarrow 各係数を線形関数で fitting. 係数 $\in \langle 1, u_1, u_2 \rangle$

Step 5 入力データ u_5 と推定値 \hat{u}_5 を比較して、外れ値や再確認が必要な入力を判定する.以下を global な推定式と呼ぶ

$$\hat{u}_5 = c(u_1, u_2) + a_3(u_1, u_2)(u_3 + b_3(u_1, u_2))^2 a_4(u_1, u_2)(u_4 + b_4(u_1, u_2))^2$$

このアルゴリズムを適用することで global な推定式が得られた. これは u_5 が (u_1, \ldots, u_4) の 3 次 の多項式となっている:

$$u_5 = c(u_1, u_2) + a_3(u_1, u_2)(u_3 + b_3(u_1, u_2))^2 + a_4(u_1, u_2)(u_4 + b_4(u_1, u_2))^2$$

$$c(u_1, u_2) = +15.2251 - 8.1648u_1/3000 + 31.2230u_2/100$$

$$a_3(u_1, u_2) = -0.0513 + 0.0049u_1/3000 + 0.0141u_2/100$$

$$b_3(u_1, u_2) = -23.0182 - 0.2231u_1/3000 - 0.0143u_2/100$$

$$a_4(u_1, u_2) = -0.0067 + 0.0011u_1/3000 + 0.0018u_2/100$$

$$b_4(u_1, u_2) = +10.8745 + 0.6250u_1/3000 + 19.8200u_2/100$$

このようにして得られた global な推定式の妥当性として, 的中率を以下のように提案し計算した: 探索方向に対して決まるペア (u^i,u^j) で考える $((y^i,y^j)=(0,1)$ を仮定). 各データペアに対して以下の通りにラベルづけする.

。 推定式でペアの内側と外側を正しく推定している : $y^i=0, u^i_5 < \hat{u}^i_5, y^j=1, u^j_5 < \hat{u}^j_5$

 \triangle データでは失火、だが推定式では失火でないと推定: $y^j = 1, u_5^j > \hat{u}_5^j$

× データでは失火でない、だが推定式では失火と推定: $y^i=0, u^i_5>\hat{u}^i_5$

黄色 $|u_5^i - \hat{u}_5^i| \le 1$ の時は許容

赤青 $|u_5^i - \hat{u}_5^i| > 1$ の時は許容しない

これらの記号を組み合わせて利用する. その組み合わせが何を意味するのか以下に記載する:

•
$$y_{\pm, \nu}^{j} = 1, -1 < u_{5}^{j} - \hat{u}_{5}^{j} : \triangle$$
 (許容する)

•
$$y^j_{\pm,k} = 1, u^j_5 - \hat{u}^j_5 < -1: \triangle$$
 (許容しない)

•
$$y_{失火}^i = 0, \, 0 < u_5^i - \hat{u}_5^i < 1: \times$$
 (許容する)

•
$$y_{失 k}^i = 0, u_5^i - \hat{u}_5^i > 1: \times$$
 (許容しない)

• それ以外: 0

20.0

各データに対して \circ あるいは \triangle , \times とラベルづけされたデータのペアを的中したデータという. 的中したデータペアの比率を的中率 Gr とする:

$$G := \{\circ, \triangle, \times\}, nG := \{\triangle, \times\}, Gr := \frac{\#G}{\#G + \#nG}.$$

local な推定式と global な推定式の的中率を表 3,4 に記載する.

1.00(60/60)

600 1200 1800 2400 3000 $u_2 \backslash u_1$ 90.0 1.00(30/30) 0.98(46/47)0.98(48/49)0.98(50/51)0.98(51/52)72.5 0.98(40/41)1.00(109/109)0.98(115/117)0.99(121/122)1.00(58/58)55.0 1.00(59/59)1.00(126/126)1.00(132/132)0.98(126/128)1.00(60/60)37.5 1.00(60/60) 1.00(131/131)1.00(131/131)0.99(112/113)0.95(57/60)

表 3: Ratio of Hit(local)

表 4: Ratio of Hit(global)

0.97(59/61)

0.87(53/61)

0.63(38/60)

$u_2 \backslash u_1$	600	1200	1800	2400	3000
90.0	0.57(17/ 30)	0.98(46/47)	0.98(48/49)	0.98(50/ 51)	0.98(51/52)
72.5	0.63(26/ 41)	0.96(105/109)	0.99(116/117)	0.97(118/122)	0.98(57/ 58)
55.0	1.00(59/ 59)	1.00(126/126)	1.00(132/132)	0.91(116/128)	0.93(56/60)
37.5	0.62(37/ 60)	0.54(71/131)	0.65(85/131)	0.89(101/113)	0.95(57/ 60)
20.0	0.00(0/ 61)	0.05(3/ 60)	0.61(37/61)	0.84(51/ 61)	0.60(36/ 60)

表 3, 4 から, 次のことが観察される:

1.00(61/61)

- $(u_1, u_2) = (20.0, 3000), (20.0, 2400)$ を除いて local な推定式はほぼ的中している. これは local な推定式の推定精度の高さを表している.
- $u_2 = 55.0, 72.5$ などでは、global な推定式の精度も高い.このあたりのデータに対して global な推定式が有効に働くものと思われる.
- 一方, $u_2 = 37.5, 20.0$ では, global な推定式の精度が低い. これらの local な推定式だけ集めて再度 global な推定式を求めると精度が高い推定式が得られるかもしれない.

2.3 凸包を利用した境界モデルの効率化

Mathworks が開発している Model Based Calibration (MBC) Toolbox では、凸包を利用して境界 モデルを構成するプログラムが実装されている。この中では、エンジンが正常に動作する入力の集合 S に対して凸包をとり、

$$S = \{ u \in \mathbb{R}^n : Au \le b \}$$

となる行列 $A \in \mathbb{R}^{m \times n}$ とベクトル $b \in \mathbb{R}^n$ を計算している. これは, n が大きくなると (i) m の数が指数オーダで増加する, (ii) 数値誤差の影響で精度の悪い A,b しか求められない, ということになり, 計算精度の面でも速度の面でも不利である. これを用いて, 応答局面法では, 以下の最適化問題を解く:

$$(\mathbb{P}): \begin{cases} \min_{x} & f(x) \\ \text{subject to} & g_{p}(x) \leq 0 \ (p = 1, \dots, q), \\ & Ax \leq b, \ell \leq x \leq u \end{cases}$$
 (1)

そこで, 行列 $A \in \mathbb{R}^{m \times n}$ とベクトル $b \in \mathbb{R}^n$ を計算しない方法を提案した. それは, (1) の代わりに以下の問題を解くことである:

$$(\mathbb{Q}): \begin{cases} \min_{\alpha} & f(D\alpha) \\ \text{subject to} & g_p(D\alpha) \leq 0 \ (p=1,\ldots,q), \\ & \ell \leq D\alpha \leq u, \sum_{j=1}^k \alpha_j = 1, \alpha_j \geq 0 \end{cases}$$
 (2)

ここで, $D=(u_1,\ldots,u_k)\in\mathbb{R}^{n\times k}$ は各データを列ベクトルとする行列である. この提案の利点・欠点は以下の通りである:

- (1) の場合, $n \ge 10$ だと凸包の計算には時間が掛かる & 誤差を多く含む可能性がある, がそれを skip できる
- 生成された行列 A の行の数が n に関して指数オーダーになるので、最適化計算 (1) に時間が掛かる & 計算精度が良くない
- n が小さいと、凸包を作って(1)を計算した方が速い

数値実験を行い、提案手法と既存手法の比較を MBC Toolbox を利用して行った. 利用した計算機は、OS が Windows 7、CPU が Intel(R) Core(TN) i7-04790、3.60GHz である. 用いた数値例は、Diesel エンジンデータ $(7\times125=825\ \vec{r}-9)$ で 7 つのオペレーティングポイント (回転数、BTQ) において BSFC(正味燃料消費率 (Brake Specific Fuel Consumption)、g/kWh) の最小化であり、入力は 4 つ、7 つ、9 つの 3 つのパターンで、

MAINSOI, FUELPRESS, VGTPOS, EGRPOS, MAINFUEL, EGRMF, AFR, VGTSPEED, PEAKPRESS

である. また、制約は入力データの凸包のみである. つまり (1), (2) において g_p は存在しない. 目的 関数が非線形関数なので適用する最適化手法は反復法で、MBC Toolbox で利用される fmincon を利用しているが、初期点は実行可能解を与えている.

数値結果を表5に掲載する:

表 5: 計算時間の比較 [sec]

	N	Create Boundary	Create Model	Opt.	# of feas. sol.
MBC	4	0.06	6.50	5.33	7
MBC	7	7.72	9.87	15.05	5
MBC	9	285.31	11.16	66.65	1
提案	4	_	6.83	45.52	7
提案	7	_	9.52	26.06	7
提案	9	_	11.13	37.94	7

表 5 において、「MBC」とある行が MBC Toolbox に実装されている凸包生成プログラムを実行した結果である。「Create Boundary」がその凸包生成にかかる時間である。「Create Model」が MBC Toolbox 内で目的関数を生成する時間であり、既存手法も提案手法もどちらも同じ命令を実行している。「Opt.」は最適化問題 (1), (2) を解く時間であり、「# of feas. sol.」は fmincon が 7 つの最適化問題を解いた際に実行可能解を生成した回数である。表 5 から以下のことが観察できる:

- N=4 くらいまでは、既存手法の方が高速に計算する
- N = 7.9 では提案手法の方が高速に計算する
- fmincon は、提案手法だと必ず実行可能解を生成しているが、既存手法だと生成できないこともある.