Math-for-industry
Education & Research Hub

ISSN 1881-4042

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

Development of Computer Algebra Research and Collaboration with Industry

編集：照井　章，小原功任，濱田龍義，横山俊一
穴井宏和，横田博史

九州大学グローバルCOEプログラム

COE Lecture Note Vol.49 : Kyushu University

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

Development of Computer Algebra Research and
Collaboration with Industry

編集

照井　章，小原功任，濱田龍義

横山俊一，穴井宏和，横田博史

About MI Lecture Note Series

The Math-for-Industry (MI) Lecture Note Series is a successor to the COE Lecture Notes, published for the 21st COE Program "Development of Dynamic Mathematics with High Functionality", sponsored by Ministry of Education, Culture, Sports, Science and technology-Japan (MEXT) (From 2003 to 2007).

The MI series reports lectures given by scholars invited under the following two programs: "Training Program of Ph.D. and new Master's in Mathematics as Required by Industry", adopted as a Support Program for Improving Gradute School Education by MEXT (from 2007 to 2009); and Education-and-Research Hub for Mathematics-for-Industry", newly adopted as a Global COE Program by MEXT (from 2008 to 2012).

July 2008
Masato Wakayama
Global COE Program "Education-and-Research Hub for Mathematics-for-Industry"
Program Leader

はじめに

本レクチャーノートは，2013 年 8 月 21 日から 23 日にかけて，九州大学伊都キャンパスで開催される IMI 共同利用研究集会 “数式処理研究と産学連携の新たな発展” の予稿集である．

本研究集会は，数式処理（計算代数 (Computer Algebra) や，より広い意味で，計算機で数式・数学情報を扱う技術や方法論）の産学連携および産業界への応用を中心とした今後の展望と方向性を検討することを目的として企画された．

数式処理に関する研究は，これまでに，数学，計算機科学に関連した分野で成果を挙げているが，産学連携や産業界への応用は，国内においてはまだ限定的であると思われる．一方で，欧米など世界の諸地域では，自動車産業や計算機援用デザイン (CAD) を中心とする諸分野との産学連携や産業界への応用を目指した研究・開発が活発に行われている．そこで，本研究集会では，計算代数・数式処理の産学連携や産業界への応用に携わっている国内外の著名な研究者を招き，実例を学ぶとともに，計算代数・数式処理の研究者と，応用分野の研究者や技術者が共に集い，計算代数・数式処理の産学連携を見据えた研究・開発や，産学連携の進むべき方向等について議論することを目指している．

本研究集会では，2 件のチュートリアルを企画した．Wen-Shin Lee 氏には，計算代数の理論研究者として，関数補間の信号処理への応用について紹介いただく．伊藤久弘氏には，自動車産業の技術者として，エンジン制御系設計における数式処理の活用について紹介いただく．多忙な中講演をお引き受け下さった両氏に感謝する．

一般講演は，16 件の講演申し込みがあった．内容は，数学への応用，Gröbner 基底の計算と応用，制御系設計，ソフトウェア，教育，数式・数値融合計算，線形代数，代数方程式と多岐にわたる．講演者各位に感謝するとともに，本研究集会が，国内の計算代数・数式処理の産業界への浸透に貢献できれば我々の喜びである．

九州大学マス・フォア・インダストリ研究所には，本研究集会の開催にあたり，旅費の助成をはじめとする研究集会開催への助力をいただいた．ここに感謝申し上げる．

2013 年 8 月

照井　章（筑波大学）
小原 功任（金沢大学）
濱田 龍義（福岡大学）
横山 俊一（九州大学）
穴井 宏和（富士通研究所／九州大学）
横田 博史（東芝インフォメーションシステムズ）

Preface

This lecture note contains invited and contributed papers that will be presented at "IMI Workshop on Developments in Computer Algebra Research and Collaboration with Industry", to be held at Ito Campus, Kyushu University, on August 21–23, 2013.

We have organized this workshop for communicating between researchers in computer algebra and industry to examine the prospects and future research directions for promotion of application and/or collaboration of computer algebra to industry.

Researches on computer algebra have contributed new results in various fields including mathematics and computer science so far, yet the application and/or collaboration of computer algebra research to industry seems still limited in Japan. On the other hand, in other side of the world such as Europe and the United States, researchers have been actively conducting collaboration with people in industry such as automotive industry and computer-aided design (CAD) for application of computer algebra. Thus, we have invited researchers actively working for application and/or collaboration of computer algebra to industry to share their practical experience and ideas. Furthermore, we bring researchers and engineers in computer algebra and application area together for sharing ideas for possible direction of research in computer algebra for application and/or collaboration of computer algebra with industry.

In the workshop, we have organized tutorial sessions with two invited talks. Wen-Shin Lee, as a researcher in computer algebra, will introduce application of sparse interpolation in signal processing. Hisahiro Ito, as an engineer in automotive industry, will introduce practical use of computer algebra system (CAS) in engine control system development. We are grateful to both speakers for accepting our invitations and presenting valuable talks at the workshop.

We also have 16 contributed talks that contain various topics: using computer algebra in mathematics, Gröbner bases, design of control systems, mathematical software, education, symbolic-numeric computation, linear algebra, and algebraic equations. Thanks to all the contributors for their talks and it is our pleasure if this workshop will contribute for promotion of application and/or collaboration of computer algebra to industry.

We also appreciate the Institute of Mathematics for Industry (IMI), Kyushu University for financial assistance and other help for holding the workshop.

August 2013

Akira Terui (University of Tsukuba)
Katsuyoshi Ohara (Kanazawa University)
Tatsuyoshi Hamada (Fukuoka University)
Shun'ichi Yokoyama (Kyushu University)
Hirokazu Anai (Fujitsu Laboratories Ltd. / Kyushu University)
Hiroshi Yokota (Toshiba I.S. Corporation)

九州大学 マス・フォア・インダストリ研究所 (IMI) 共同利用研究集会

数式処理研究と産学連携の新たな発展

日程: 2013年8月21日(水) 14:00～8月23日(金) 12:20
場所: 九州大学伊都キャンパス センター2号館2310号室 (福岡県福岡市西区元岡744)
Webサイト: https://sites.google.com/site/imidcar2013/

プログラム

8月21日(水)		
オープニング	14:00 ～ 14:10	
セッション1 (数学への応用)	14:10 ～ 14:40	Solving problems of Goldberg for rational maps on the projective space 藤村 雅代(防衛大学校)
	14:40 ～ 15:10	Lauricella 超幾何微分方程式系のグレブナー基底 中山 洋将(神戸大学 / JST CREST)
	15:10 ～ 15:40	Markov chain Monte Carlo methods for the regular two-level fractional factorial designs and cut ideals 青木 敏(鹿児島大学 / JST, CREST), 大杉 英史(立教大学 / JST, CREST), 日比 孝之(大阪大学 / JST, CREST)
セッション2 (Groebner基底の計算と応用)	15:50 ～ 16:20	Transformation of lexicographic Groebner bases to smaller systems Xavier Dahan(九州大学)
	16:20 ～ 16:50	代数的閉体における限量子消去アルゴリズムについて 深作 亮也, 井上 秀太郎, 佐藤 洋祐(東京理科大学)

8月22日(木)		
セッション3 (制御系設計)	9:00 ～ 9:30	An effective implementation of a special quantifier elimination for a sign definite condition by logical formula simplification 岩根 秀直, 樋口 博之(富士通研究所), 穴井 宏和(富士通研究所 / 九州大学)
	9:30 ～ 10:00	Optimal Controller Design for a Power Supply Unit Using Quantifier Elimination 松井 由信, 岩根 秀直(富士通研究所), 穴井 宏和(富士通研究所 / 九州大学)
セッション4 (ソフトウェア)	10:10 ～ 10:40	MathML Content Markupで書かれた数式に対する検索手法の提案 片岡 晃久, 甲斐 博(愛媛大学)
	10:40 ～ 11:10	MathLibre: distributable and customizable desktop environment for mathematics 濱田 龍義(福岡大学 / JST CREST / OCAMI)
セッション5 (教育)	11:20 ～ 11:50	数式処理を用いたルービックキューブの素数位数操作の探求 藤本 光史(福岡教育大学), 泊 昌孝(日本大学)
	11:50 ～ 12:20	数独パズルの計算機による研究について 北本 卓也(山口大学)
チュートリアル1 (招待講演)	14:00 ～ 15:20	Sparse interpolation and signal processing Wen-Shin Lee (University of Antwerp)
チュートリアル2 (招待講演)	15:30 ～ 16:50	Engine Control System Development and Symbolic Manipulation – Application and Challenges in Modelling – 伊藤 久弘 (トヨタ自動車)
情報交換	16:50 ～ 17:20	研究集会等の情報交換、アナウンス等

8月23日(金)		
セッション6 (数式・数値融合計算)	9:30 ～ 10:00	有理関数を基にした多変数近似GCD計算 讃岐 勝(筑波大学)
	10:00 ～ 10:30	厳密に与えられた系のGroebner基底を数値的に求める場合に必要な桁精度の考察 長坂 耕作(神戸大学)
セッション7 (線形代数, 代数方程式)	10:40 ～ 11:10	最小消去多項式候補を用いた行列の一般固有空間の構造の計算法について 小原 功任(金沢大学), 田島 慎一(筑波大学)
	11:10 ～ 11:40	行列の最小消去多項式候補を用いた固有ベクトル計算 (II) 田島 慎一, 照井 章(筑波大学)
	11:40 ～ 12:10	Computing the longest polynomial in the world – general discriminant formula of degree 17– 木村 欣司(京都大学)
クロージング	12:10 ～ 12:20	

組織委員:
照井　章(筑波大学)
小原 功任(金沢大学)
濱田 龍義(福岡大学)
横山 俊一(九州大学)
穴井 宏和(富士通研究所／九州大学)
横田 博史(東芝インフォメーションシステムズ)

本研究集会について

本研究集会は、数式処理(計算代数 (Computer Algebra) や、より広い意味で、計算機で数式・数学情報を扱う技術や方法論)の算法・システム開発・応用の各分野で活躍する研究者が、未解決問題の提起、萌芽的アイデアの紹介、最新の実装状況の報告、他分野との連携等に関する討論を行い、数式処理研究の産学連携および産業界への応用を中心とした今後の展望と方向性を検討することを目的として開催するものです。
数式処理、計算代数の研究や応用を行っている研究者・技術者の方々をはじめ、数式処理の理論や応用に興味や関心をもつ多数の皆様のご参加をお待ちしています。

Institute of Mathematics for Industry (IMI), Kyushu University, Workshop on

Developments in Computer Algebra Research and Collaboration with Industry

Date: from August 21, 2013, 14:00 to August 23, 2013, 12:20
Place: Room 2310, Center Zone 2, Kyushu University Ito Campus
744 Motooka, Nishi-ku, Fukuoka 819-0395, Japan
Web site: https://sites.google.com/site/imidcar2013/

Program

Wednesday, August 21		
Opening remarks	14:00 ～ 14:10	
Session 1 (Application to mathematics)	14:10 ～ 14:40	**Solving problems of Goldberg for rational maps on the projective space** Masayo Fujimura (National Defense Academy)
	14:40 ～ 15:10	**Groebner bases of Lauricella's hypergeometric equations and its applications** Hiromasa Nakayama (Kobe University / JST CREST)
	15:10 ～ 15:40	**Markov chain Monte Carlo methods for the regular two-level fractional factorial designs and cut ideals** Satoshi Aoki (Kagoshima University / JST, CREST), Hidefumi Ohsugi (Rikkyo University / JST, CREST) and Takayuki Hibi (Osaka University / JST, CREST)
Session 2 (Computation and application of Groebner bases)	15:50 ～ 16:20	**Transformation of lexicographic Groebner bases to smaller systems** Xavier Dahan (Kyushu university)
	16:20 ～ 16:50	**On QE Algorithms over algebraically closed field** Ryoya Fukasaku, Shutaro Inoue and Yosuke Sato (Tokyo University of Science)

Thursday, August 22		
Session 3 (Design of control systems)	9:00 ～ 9:30	**An effective implementation of a special quantifier elimination for a sign definite condition by logical formula simplification** Hidenao Iwane, Hiroyuki Higuchi (Fujitsu Laboratories Ltd) and Hirokazu Anai (Fujitsu Laboratories Ltd / Kyushu University)
	9:30 ～ 10:00	**Optimal Controller Design for a Power Supply Unit Using Quantifier Elimination** Yoshinobu Matsui, Hidenao Iwane (Fujitsu Laboratories Ltd) and Hirokazu Anai (Fujitsu Laboratories Ltd / Kyushu University)
Session 4 (Software)	10:10 ～ 10:40	**Proposal of a search method for MathML Content Markup** Akihisa Kataoka and Hiroshi Kai (Ehime University)
	10:40 ～ 11:10	**MathLibre: distributable and customizable desktop environment for mathematics** Tatsuyoshi Hamada (Fukuoka University / JST CREST / OCAMI)
Session 5 (Education)	11:20 ～ 11:50	**A hunting of operations with prime order on Rubik's Cube using computer algebra** Mitsushi Fujimoto (Fukuoka University of Education) and Masataka Tomari (Nihon University)
	11:50 ～ 12:20	**On the Analysis of Sudoku Puzzles by Computers** Takuya Kitamoto (Yamaguchi University)

Tutorial 1 (Invited talk)	14:00 ～ 15:20	**Sparse interpolation and signal processing** Wen-Shin Lee (University of Antwerp)
Tutorial 2 (Invited talk)	15:30 ～ 16:50	**Engine Control System Development and Symbolic Manipulation - Application and Challenges in Modelling -** Hisahiro Ito (TOYOTA MOTOR CORPORATION)
Communications	16:50 ～ 17:20	Communication on information of conferences and/or other announcements on computer algebra

Friday, August 23		
Session 6 (Symbolic-numeric computation)	9:30 ～ 10:00	**Computing the Approximate Multivariate Greatest Common Divisor via Rational Function** Masaru Sanuki (University of Tsukuba)
	10:00 ～ 10:30	**A note on required precision for computing numerical Groebner basis of exact input** Kosaku Nagasaka (Kobe University)
Session 7 (Linear algebra and algebraic equations)	10:40 ～ 11:10	**On determining the structure of the invariant space of matrices via pseudo-annihilating polynomials** Katsuyoshi Ohara (Kanazawa University) and Shinichi Tajima (University of Tsukuba)
	11:10 ～ 11:40	**Calculating eigenvectors of matrices using candidates for minimal annihilating polynomials II** Shinichi Tajima and Akira Terui (University of Tsukuba)
	11:40 ～ 12:10	**Computing the longest polynomial in the world - general discriminant formula of degree 17-** Kinji Kimura (Kyoto University)
Closing remarks	12:10 ～ 12:20	

Organizers:
Akira Terui (University of Tskuba)
Katsuyoshi Ohara (Kanazawa University)
Tatsuyoshi Hamada (Fukuoka University)
Shun'ichi Yokoyama (Kyushu University)
Hirokazu Anai (Fujitsu Laboratories Ltd. / Kyushu University)
Hiroshi Yokota (Toshiba I.S. Corporation)

About the Workshop

We organize this workshop for communicating between researchers in computer algebra and industry to examine the prospects and future research directions for promotion of application and/or collaboration of computer algebra to industry through discussions on various topics including algorithms, systems and applications of computer algebra.
All of those who are interested in computer algebra are welcome.

目　次

Contents

セッション 1

Session 1

数学への応用

Application to mathematics

Solving problems of Goldberg for rational maps on the projective space

Masayo FUJIMURA*
Department of Mathematics, National Defense Academy †

Abstract

In [3], we introduce the generalized Bell representation, and solve a problem of Goldberg that determine the number of equivalence classes of rational maps corresponding to each critical set. In this talk, we solve this problem by using rational maps on the projective space $\mathbb{P}^1(\mathbb{C})$. Symbolic and algebraic computation system is indispensable to determine defining equations of some singular loci.

1 Introduction

In [4], Goldberg suggested a problem that determine the number of equivalence classes of rational maps corresponding to each critical set. This problem is based on her theorem (Theorem 1.3 in [4]), and it is known that this theorem deeply concern with B. and M. Shapiro conjecture (see [1]).

As a joint work with M. Karima (Kabur Univ.) and M. Taniguchi (Nara Women's Univ.), we solved a problem of Goldberg for the generic case when the degree is small (see [2]). Moreover, in [3], we determine several kinds of the non-generic loci for the map from the generalized Bell locus to the space of the sets of critical points explicitly when the degree is small.

In this talk, we solve this problem by using a family of rational maps on the projective space $\mathbb{P}^1(\mathbb{C})$. By this technique, we can obtain the same result as in [3] more simply.

A rational map of degree d is a map with the following form,

$$R(z) = \frac{P(z)}{Q(z)},$$

where P and Q are coprime polynomials with $\max\{\deg P, \deg Q\} = d$.

Definition 1

Two rational maps R_1 and R_2 are said to be *Möbius equivalent* if there is a Möbius transformation $M : \hat{\mathbb{C}} \to \hat{\mathbb{C}}$ such that $R_2 = M \circ R_1$.

Let X_d be the set of all equivalence classes of rational maps of degree d, and $X_d^{(k)}$ be the set of classes of rational maps having critical point at ∞ with multiplicity k, where $k = 0$ means that ∞ is non-critical.

Remark 1

A rational map R of degree d has $2d-2$ critical points counted including multiplicity. The set of critical points of R is invariant under taking a Möbius conjugate.

For each rational map R of degree d, the multiplicity of critical point at ∞ is at most $d-1$. Therefore, the space X_d is the disjoint union of $X_d^{(0)}, X_d^{(1)}, \cdots, X_d^{(d-1)}$.

Goldberg showed the following theorem.

*The author is partially supported by Grant-in-Aid for Scientific Research (C) 22540240.

†masayo@nda.ac.jp

Theorem 2 (Goldberg [4])
A $(2d-2)$-tuple B is the critical set of at most $C(d)$ classes in X_d, where $C(d)$ means the d-th Catalan number $\frac{1}{d}\begin{pmatrix} 2d-2 \\ d-1 \end{pmatrix}$. The maximal is attained by a Zariski open subset of the space $\hat{\mathbb{C}}^{2d-2}$ of all B.

The map $\Phi_d : X_d \to \widehat{\mathbb{C}}^{2d-2}$ is defined by sending a equivalence class to the set of critical points, and the restriction of Φ_d to $X_d^{(k)}$ is denoted by $\Phi_d^{(k)}$.

Then Goldberg's problem (see [4]) is written as follows:

Problem 1
- *Describe in detail the ramification sets of the maps Φ_d.*
- *Given a critical set $\boldsymbol{\alpha}$, determine the number of points in the preimage $\Phi_d^{-1}(\boldsymbol{\alpha})$.*

The critical set is called *admissible* if every point has multiplicity at most $d-1$. She also asked in [4] whether every admissible set in $\mathbb{C}^{2d-2}$ is attained by some rational map of degree d.

2 Generalized Bell family

In this section, we summarize the results in [3].

Let $CB_d^{(k)}$ $(k = 0, 1, \cdots, d-1)$ be the *generalized Bell locus* consisting of all $H + \hat{P}/Q$, for

$$\begin{aligned} H(z) &= z^{k+1} + c_k z^k + \cdots + c_1 z, \\ \hat{P}(z) &= a_{d-k-2} z^{d-k-2} + \cdots + a_0, \\ Q(z) &= z^{d-k-1} + b_{d-k-2} z^{d-k-2} + \cdots + b_0, \end{aligned}$$

with $\mathrm{Resul}_z(\hat{P}, Q) \neq 0$.

Remark 2
If $k = d-1$, the generalized Bell locus is the family of polynomial maps $CB_d^{(d-1)} = \{z^d + c_{d-1} z^{d-1} + \cdots + c_1 z\}$. If $k = 0$, the generalized Bell locus coincides with the Bell locus; $CB_d^{(0)} = CB_d$ (see [2]).

The following proposition is an extended version of Proposition 5 in [2].

Proposition 3
For every $R \in CB_d^{(k)}$, $[R]$ belongs to $X_d^{(k)}$ for every k, and for each element $[S]$ in $X_d^{(k)}$, there is a unique R in $CB_d^{(k)}$ with $[R] = [S]$.

Hence, each locus $X_d^{(k)}$ has a system of coordinates consisting of coefficients of representatives R in the generalized Bell locus $CB_d^{(k)}$.

Now, consider the map $\Phi_d^{(k)}$ of $CB_d^{(k)}$ to $\mathbb{C}^{2d-2-k}$ defined from the equation

$$\begin{aligned} &\frac{1}{k+1}\Big\{H'(z)Q^2(z) + \hat{P}'(z)Q(z) - \hat{P}(z)Q'(z)\Big\} \\ &\qquad = z^{2d-k-2} + \alpha_{2d-k-3} z^{2d-k-3} + \cdots + \alpha_0 = 0 \end{aligned}$$

by sending

$$(\boldsymbol{c}, \boldsymbol{a}, \boldsymbol{b}) = (c_k, \cdots, c_1,\ a_{d-k-2}, \cdots, a_0,\ b_{d-k-2}, \cdots, b_0)$$

to

$$\boldsymbol{\alpha} = (\alpha_{2d-k-3}, \cdots, \alpha_0).$$

Set

$$R_d^{(k)} = \big\{(\boldsymbol{c}, \boldsymbol{a}, \boldsymbol{b}) \in \mathbb{C}^{2d-2-k} \ :\ \mathrm{Resul}_z(\hat{P}, Q) = 0\big\},$$

which is the locus where $\Phi_d^{(k)}$ is not defined. (In other words, $CB_d^{(k)}$ can be identified with $\mathbb{C}^{2d-2-k} - R_d^{(k)}$.)

Here, we recall the following results in [2].

Proposition 4
The map $\Phi_2^{(0)} : CB_2^{(0)} \to \mathbb{C}^2 - E^{(0)}(2)$ *is bijective, and the exceptional locus* $E^{(0)}(2)$ *is the algebraic curve defined by* $\alpha_1^2 - 4\alpha_0 = 0$. *And the map* $\Phi_2^{(1)} : CB_2^{(1)} \to \mathbb{C}$ *is bijective.*

Now, we recall the following results in [2] and [3].

Proposition 5
The ramification locus of $\Phi_3^{(0)}$ *is* $a_1 = b_1^2 - 4b_0$, $\Phi_3^{(0)}(CB_3^{(0)}) = \mathbb{C}^4 - E^{(0)}(3)$, *and* $\Phi_3^{(0)}$ *is 2-valent on the set of points in* $\mathbb{C}^4 - E^{(0)}(3)$ *satisfying that*

$$\alpha_2^2 - 3\alpha_1\alpha_3 + 12\alpha_0 \neq 0, \quad E_0 \neq 0.$$

Here, the exceptional locus $E^{(0)}(3)$ *is the algebraic variety defined by* $E_0 = E_1 = 0$. *Here*

$$\begin{aligned} E_1 =& 27\alpha_1^2 - 9\alpha_2\alpha_3\alpha_1 + (27\alpha_3^2 - 72\alpha_2)\alpha_0 + 2\alpha_2^3, \qquad (1)\\ E_0 =& -27\alpha_1^4 + (-4\alpha_3^3 + 18\alpha_2\alpha_3)\alpha_1^3 + ((-6\alpha_3^2 + 144\alpha_2)\alpha_0 + \alpha_2^2\alpha_3^2 - 4\alpha_2^3)\alpha_1^2 \\ &+ (-192\alpha_3\alpha_0^2 + (18\alpha_2\alpha_3^3 - 80\alpha_2^2\alpha_3)\alpha_0)\alpha_1 + 256\alpha_0^3 \\ &+ (-27\alpha_3^4 + 144\alpha_2\alpha_3^2 - 128\alpha_2^2)\alpha_0^2 + (-4\alpha_2^3\alpha_3^2 + 16\alpha_2^4)\alpha_0. \qquad (2) \end{aligned}$$

Remark 3
The exceptional locus $E^{(0)}(3)$ *is written as,*

$$\left\{108\alpha_1^2 + (-108\alpha_3\alpha_2 + 27\alpha_3^3)\alpha_1 + 32\alpha_2^3 - 9\alpha_3^2\alpha_2^2 = 0, \quad \textit{and} \quad 3\alpha_3\alpha_1 - \alpha_2^2 - 12\alpha_0 = 0\right\}.$$

In case $d = 3$, there remain the cases that ∞ is a critical point.

Proposition 6
The ramification locus of $\Phi_3^{(1)}$ *is given by* $c_1 - 2b_0 = 0$, $\Phi_3^{(1)}(CB_3^{(1)}) = \mathbb{C}^3 - E^{(1)}(3)$ *and* $\Phi_3^{(1)}$ *is 2-valent on the the set of the points in* $\mathbb{C}^3 - E^{(1)}(3)$ *satisfying that*

$$3\alpha_1 - \alpha_2^2 \neq 0, \quad 4\alpha_1^3 - \alpha_2^2\alpha_1^2 - 18\alpha_0\alpha_2\alpha_1 + 4\alpha_0\alpha_2^3 + 27\alpha_0^2 \neq 0.$$

Here, the exceptional locus $E^{(1)}(3)$ *is the algebraic variety defined by*

$$\left\{3\alpha_1 - \alpha_2^2 = 0, \quad 9\alpha_2\alpha_1 - 2\alpha_2^3 - 27\alpha_0 = 0\right\}.$$

Since the map $\Phi_3^{(2)} : CB_3^{(2)} \to \mathbb{C}^2$ is clearly bijective, we have obtained complete description for the case that $d = 3$.

3 Generalized Bell family on $\mathbb{P}^1(\mathbb{C})$

A rational map R on $\mathbb{P}^1(\mathbb{C})$ is defined by

$$R(z_0, z_1) = \frac{P(z_0, z_1)}{Q(z_0, z_1)}$$

where P and Q are homogeneous polynomial maps of degree d with $Q \not\equiv 0$.

Now, we give the following extended version of Proposition 3. Let PB_d be the family consisting of all $F_{(\boldsymbol{b},\boldsymbol{a})} = \dfrac{P}{Q}$, for

$$\begin{aligned} P(z_0, z_1) &= z_1^d + (1 - b_{d-1})a_{d-1}z_0z_1^{d-1} + (1 - (1 - b_{d-1})b_{d-2})a_{d-2}z_0^2z_1^{d-2} + \cdots \\ &\quad + (1 - (1 - b_{d-1}) \cdots (1 - b_1)b_0)a_0z_0^d, \\ Q(z_0, z_1) &= b_{d-1}z_0z_1^{d-1} + b_{d-2}z_0^2z_1^{d-2} + \cdots + b_0z_0^d, \\ \mathrm{GCD}(P, Q) &\in \mathbb{C}^*, \end{aligned}$$

where

$$\boldsymbol{b} = (b_{d-1} : \cdots : b_0) \in \mathbb{P}^{d-1}(\mathbb{C}),$$
$$\boldsymbol{a} = (1 : a_{d-1} : \cdots : a_0) \in \mathbb{P}^{d}(\mathbb{C}).$$

Remark 4
If the coefficients of Q satisfy $\boldsymbol{b} = (\underbrace{0 : \cdots : 0}_{k} : 1 : b_{d-k-2} : \cdots : b_0)$, then the coefficient of the term $z_0^{k+1} z_1^{d-k-1}$ of numerator P does not depend on a_{d-k-1} and is always zero.

The family PB_d represents the space X_d faithfully.

Theorem 7
For every $F(z_0, z_1) = \dfrac{P(z_0, z_1)}{Q(z_0, z_1)}$ in PB_d, the equivalence class $\Big[\dfrac{P(1, z_1)}{Q(1, z_1)}\Big]$ belongs to X_d. Conversely, For every $\big[R\big]$ in X_d, there is unique rational map $F(z_0, z_1) = \dfrac{P(z_0, z_1)}{Q(z_0, z_1)}$ in PB_d such that $\big[\tilde{R}\big] = \big[R\big]$, where

$$\tilde{R}\Big(\frac{z_1}{z_0}\Big) = \frac{P(z_0, z_1)}{Q(z_0, z_1)}.$$

Remark 5
For every rational map $F_{(\boldsymbol{b},\boldsymbol{a})}$ in PB_d, $F_{(\boldsymbol{b},\boldsymbol{a})}(0, 1) = (0, 1)$. The inverse image $F^{-1}_{(\boldsymbol{b},\boldsymbol{a})}(0, 1)$ is the set given by

$$\{(z_0, z_1)\,;\, z_0 = 0 \ \ or \ \ Q(z_0, z_1) = 0\}.$$

The map $\widehat{\Psi}_d : PB_d \to \mathbb{P}^{2d-2}(\mathbb{C})$ is defined by sending

$$(\boldsymbol{b}, \boldsymbol{a}) = \Big((\underbrace{0 : \cdots : 0}_{k} : 1 : b_{d-k-2} : \cdots : b_0), (1 : a_{d-1} : \cdots : a_{d-k-2} : 0 : a_{d-k} : \cdots : a_0)\Big)$$
$$(k = 0, 1, \cdots, d-1)$$

to

$$\boldsymbol{\alpha} = (\alpha_{2d-2} : \cdots : \alpha_0) \in \mathbb{P}^{2d-2}(\mathbb{C}),$$

where $F_{(\boldsymbol{b},\boldsymbol{a})} = \dfrac{P}{Q} \in PB_d$ and

$$\frac{\partial(Q, P)}{\partial(z_0, z_1)} = \alpha_{2d-2} z_1^{2d-2} + \alpha_{2d-3} z_0 z_1^{2d-3} + \cdots + \alpha_0 z_0^{2d-2}.$$

In the case of $d = 3$, rational map $F_{(\boldsymbol{b},\boldsymbol{a})}$ on $\mathbb{P}^1(\mathbb{C})$ is written by

$$P(z_0, z_1) = z_1^3 + a_2(1 - b_2) z_0 z_1^2 + a_1(1 - (1 - b_2)b_1) z_0^2 z_1 + a_0(1 - (1 - b_2)(1 - b_1)b_0) z_0^3,$$
$$Q(z_0, z_1) = b_2 z_0 z_1^2 + b_1 z_0^2 z_1 + b_0 z_0^3.$$

Now, set $\widehat{R}_3 = \{(\boldsymbol{b}, \boldsymbol{a})\,;\, IP_3 = 0\}$, where

$$\begin{aligned}
IP_3 =&(a_0b_0b_2 - a_0b_0)b_1^4 + (a_0b_0a_1b_2^4 + (a_0b_0a_2 - 2a_0b_0a_1)b_2^3 + (-2a_0b_0a_2 + a_0b_0a_1)b_2^2\\
&+ (a_0b_0a_2 + b_0a_1 - a_0b_0)b_2 - b_0a_1 + a_0b_0 - a_0)b_1^3 + (a_0^2b_0^2b_2^5 + (b_0a_1^2 - a_0b_0a_1\\
&- 2a_0^2b_0^2)b_2^4 + ((b_0a_1 - a_0b_0)a_2 - 2b_0a_1^2 + (3a_0b_0 - a_0)a_1 + a_0^2b_0^2)b_2^3\\
&+ ((-2b_0a_1 + 2a_0b_0 - a_0)a_2 + b_0a_1^2 + (-2a_0b_0 + a_0)a_1 - 3a_0b_0^2)b_2^2\\
&+ ((b_0a_1 - a_0b_0 + a_0)a_2 + 3a_0b_0^2)b_2 + b_0a_1)b_1^2 + (-2a_0^2b_0^2b_2^5 + (-2a_0b_0^2a_2 + 4a_0^2b_0^2\\
&- 2a_0^2b_0)b_2^4 + (4a_0b_0^2a_2 + 2b_0a_1^2 - a_0b_0a_1 - 2a_0^2b_0^2 + 2a_0^2b_0)b_2^3 + ((b_0a_1 - 2a_0b_0^2)a_2\\
&- 2b_0a_1^2 + (-2b_0^2 + a_0b_0 - a_0)a_1 + 3a_0b_0^2)b_2^2 + ((-b_0a_1 + b_0^2)a_2 + 2b_0^2a_1\\
&- 3a_0b_0^2 + 3a_0b_0)b_2 - b_0^2a_2)b_1 + a_0^2b_0^2b_2^5 + (2a_0b_0^2a_2 - 2a_0^2b_0^2 + 2a_0^2b_0)b_2^4\\
&+ (b_0^2a_2^2 + (-4a_0b_0^2 + 2a_0b_0)a_2 + a_0^2b_0^2 - 2a_0^2b_0 + a_0^2)b_2^3 + (-2b_0^2a_2^2\\
&+ (2a_0b_0^2 - 2a_0b_0)a_2 + b_0a_1^2)b_2^2 + (b_0^2a_2^2 - 2b_0^2a_1)b_2 + b_0^3.
\end{aligned}$$

Then, we have

Lemma 8
$\widehat{R}_3$ is the locus where $\widehat{\Psi}_3$ is not defined.

Now, Jacobian is given by

$$\begin{aligned}
J =&\frac{\partial(Q,P)}{\partial(z_0,z_1)}\\
=&3(b_2z_1^4 + 2b_1z_0z_1^3 + ((-a_1b_2^2 + (-a_2 + a_1)b_2 + a_2)b_1 - a_1b_2 + 3b_0)z_0^2z_1^2\\
&+ ((2a_0b_0b_2^2 - 2a_0b_0b_2)b_1 - 2a_0b_0b_2^2 + (-2b_0a_2 + 2a_0b_0 - 2a_0)b_2 + 2b_0a_2)z_0^3z_1\\
&+ ((a_0b_0b_2 - a_0b_0)b_1^2 + ((b_0a_1 - a_0b_0)b_2 - b_0a_1 + a_0b_0 - a_0)b_1 + b_0a_1)z_0^4).
\end{aligned}$$

Therefore, the map $\widehat{\Psi}_3$ is defined by $(\boldsymbol{b}, \boldsymbol{a}) \mapsto \boldsymbol{\alpha}$, where

$$\begin{aligned}
\alpha_4 &= b_2,\\
\alpha_3 &= 2b_1,\\
\alpha_2 &= (-a_1b_2^2 + (-a_2 + a_1)b_2 + a_2)b_1 - a_1b_2 + 3b_0,\\
\alpha_1 &= (2a_0b_0b_2^2 - 2a_0b_0b_2)b_1 - 2a_0b_0b_2^2 + (-2b_0a_2 + 2a_0b_0 - 2a_0)b_2 + 2b_0a_2,\\
\alpha_0 &= (a_0b_0b_2 - a_0b_0)b_1^2 + ((b_0a_1 - a_0b_0)b_2 - b_0a_1 + a_0b_0 - a_0)b_1 + b_0a_1. \qquad (3)
\end{aligned}$$

Eliminating the parameters $\boldsymbol{a}, \boldsymbol{b}$ from $IP_3 = t$ by using (3), we have a quadratic equation $T = 0$, where

$$\begin{aligned}
T =&432t^2 + (-216\alpha_4\alpha_1^2 + 72\alpha_3\alpha_2\alpha_1 - 16\alpha_2^3 + 576\alpha_0\alpha_4\alpha_2 - 216\alpha_0\alpha_3^2)t\\
&+ 27\alpha_4^2\alpha_1^4 + (-18\alpha_4\alpha_3\alpha_2 + 4\alpha_3^3)\alpha_1^3 + (4\alpha_4\alpha_2^3 - \alpha_3^2\alpha_2^2 - 144\alpha_0\alpha_4^2\alpha_2 + 6\alpha_0\alpha_4\alpha_3^2)\alpha_1^2\\
&+ (80\alpha_0\alpha_4\alpha_3\alpha_2^2 - 18\alpha_0\alpha_3^3\alpha_2 + 192\alpha_0^2\alpha_4^2\alpha_3)\alpha_1 - 16\alpha_0\alpha_4\alpha_2^4\\
&+ 4\alpha_0\alpha_3^2\alpha_2^3 + 128\alpha_0^2\alpha_4^2\alpha_2^2 - 144\alpha_0^2\alpha_4\alpha_3^2\alpha_2 + 27\alpha_0^2\alpha_3^4 - 256\alpha_0^3\alpha_4^3. \qquad (4)
\end{aligned}$$

There are no rational functions of degree 3 corresponding to $\boldsymbol{\alpha}$ if and only if the equation (4) has 0 as a unique solution for t.

Lemma 9
The exceptional locus $PE(3)$ is given by

$$\begin{aligned}
PE(3) =&\big\{108\alpha_4^2\alpha_1^2 + (-108\alpha_4\alpha_3\alpha_2 + 27\alpha_3^3)\alpha_1 + 32\alpha_4\alpha_2^3 - 9\alpha_3^2\alpha_2^2 = 0,\\
&\quad - 3\alpha_3\alpha_1 + \alpha_2^2 + 12\alpha_0\alpha_4 = 0,\\
&\quad 27\alpha_4\alpha_1^2 - 27\alpha_3\alpha_2\alpha_1 + 8\alpha_2^3 + 27\alpha_0\alpha_3^2 = 0\big\}. \qquad (5)
\end{aligned}$$

Lemma 10
If critical set $\boldsymbol{\alpha}$ *satisfies*

$$\boldsymbol{\alpha} \notin \{PE_0 = 0\}, \quad \textit{and} \quad \boldsymbol{\alpha} \notin \{\mathit{Discr}(T) = 0\},$$

$\#\widehat{\Psi}_3^{-1}(\boldsymbol{\alpha}) = 2$, *where* PE_0 *is the constant term of* T *for* t *and* $\mathit{Discr}(T) = 3\alpha_3\alpha_1 - \alpha_2^2 - 12\alpha_0\alpha_4$.

Then, we have the following.

Theorem 11
$\widehat{\Psi}_3(PB_3) = \mathbb{P}^4(\mathbb{C}) - PE(3)$ *and* $\widehat{\Psi}_3(PB_3)$ *is* 2*-valent on the the set of the points in* $\mathbb{P}^4(\mathbb{C}) - PE(3)$ *satisfying that*

$$\mathrm{Discr}(T) \neq 0 \quad \textit{and} \quad PE_0 \neq 0.$$

This theorem corresponds to Proposition 5 and Proposition 6 which are given by using generalized Bell locus. Theorem 11 is obtained without considering the multiplicity of critical point at the point at infinity.

References

[1] A. Eremenko and A. Gabrielov, Rational functions with real critical points and the B. and M. Shapiro conjecture in real enumerative geometry, *Ann. of Math.*, **155** (2002), 105–129.

[2] M. Fujimura, M. Karima, and M. Taniguchi, The Bell locus of rational functions and problems of Goldberg, *Comm. Japan Soc. Symb. Alg. Compt.* **1** (2012), 67–74.

[3] M. Fujimura, M. Karima, and M. Taniguchi, The generalized Bell locus of rational functions and problems of Goldberg, *Proc. of the 19th ICFIDCAA*, (2013), 103–110.

[4] L. Goldberg, Catalan numbers and branched covering by the Riemann sphere, *Adv. Math.*, **85** (1991), 129–144.

[5] I. Scherbak, Rational functions with prescribed critical points, *Geom. Funct. Anal.*, **12** (2002), 1365–1380.

Gröbner bases of Lauricella's hypergeometric equations and its applications

中山 洋将

Nakayama Hiromasa

神戸大学大学院理学研究科/ JST CREST

Department of Mathematics, Graduate school of Science, Kobe University * †

Abstract

We derive Gröbner bases of Lauricella's hypergeometric differential equations. By using these Gröbner bases, we determine the characteristic variety and the singular locus of Lauricella's F_B and a Pfaffian system of Lauricella's F_D.

1 Introduction

Lauricella 超幾何級数 F_A, F_B, F_C, F_D は

$$F_A(a,b_1,\dots,b_m,c_1,\dots,c_m;x_1,\dots,x_m)=\sum_{n_1,\dots,n_m\in\mathbb{Z}_{\geq 0}}\frac{(a)_{n_1+\cdots+n_m}(b_1)_{n_1}\cdots(b_m)_{n_m}}{(c_1)_{n_1}\cdots(c_m)_{n_m}(1)_{n_1}\cdots(1)_{n_m}}x_1^{n_1}\cdots x_m^{n_m},$$

$$F_B(a_1,\dots,a_m,b_1,\dots,b_m,c;x_1,\dots,x_m)=\sum_{n_1,\dots,n_m\in\mathbb{Z}_{\geq 0}}\frac{(a_1)_{n_1}\cdots(a_m)_{n_m}(b_1)_{n_1}\cdots(b_m)_{n_m}}{(c)_{n_1+\cdots+n_m}(1)_{n_1}\cdots(1)_{n_m}}x_1^{n_1}\cdots x_m^{n_m},$$

$$F_C(a,b,c_1,\dots,c_m;x_1,\dots,x_m)=\sum_{n_1,\dots,n_m\in\mathbb{Z}_{\geq 0}}\frac{(a)_{n_1+\cdots+n_m}(b)_{n_1+\cdots+n_m}}{(c_1)_{n_1}\cdots(c_m)_{n_m}(1)_{n_1}\cdots(1)_{n_m}}x_1^{n_1}\cdots x_m^{n_m},$$

$$F_D(a,b_1,\dots,b_m,c;x_1,\dots,x_m)=\sum_{n_1,\dots,n_m\in\mathbb{Z}_{\geq 0}}\frac{(a)_{n_1+\cdots+n_m}(b_1)_{n_1}\cdots(b_m)_{n_m}}{(c)_{n_1+\cdots+n_m}(1)_{n_1}\cdots(1)_{n_m}}x_1^{n_1}\cdots x_m^{n_m},$$

と定義される. ここで, a,b,c,a_i,b_i,c_i $(i=1,\dots,m)$ は複素パラメータであり, $c,c_i\notin\mathbb{Z}_{\leq 0}$ を満たすとする. これら F_A,F_B,F_C,F_D が満たす微分方程式系として, 次のものがそれぞれ知られている.

$$\ell_i^A\cdot F_A=0,\quad \ell_i^A=\theta_i(\theta_i+c_i-1)-x_i(\theta_1+\cdots+\theta_m+a)(\theta_i+b_i)\qquad (i=1,\dots,m).$$
$$\ell_i^B\cdot F_B=0,\quad \ell_i^B=\theta_i(\theta_1+\cdots+\theta_m+c-1)-x_i(\theta_i+a_i)(\theta_i+b_i)\qquad (i=1,\dots,m).$$
$$\ell_i^C\cdot F_C=0,\quad \ell_i^C=\theta_i(\theta_i+c_i-1)-x_i(\theta_1+\cdots+\theta_m+a)(\theta_1+\cdots+\theta_m+b)\qquad (i=1,\dots,m).$$

*nakayama@math.kobe-u.ac.jp

†本研究は科研費 (課題番号:24740064) および JST CREST "数学と諸分野の協働によるブレークスルーの探索" の助成を受けたものである。

$$\ell_i^D \cdot F_D = 0 \quad (i = 1, \ldots, m),$$
$$\ell_i^D = x_i(1-x_i)\partial_i^2 + (1-x_i) \sum_{k \neq i, 1 \leq k \leq m} x_k \partial_i \partial_k + (c - (a + b_i + 1)x_i)\partial_i - b_i \sum_{k \neq i, 1 \leq k \leq m} x_j \partial_j - ab_i.$$
$$\ell_{ij}^D \cdot F_D = 0, \quad \ell_{ij}^D = (x_i - x_j)\partial_i \partial_j - b_j \partial_i + b_i \partial_j \quad (1 \leq i < j \leq m).$$

ここで, $\cdot$ は微分作用素を関数に作用させることを表す記号, $\partial_i = \frac{\partial}{\partial x_i}$ は x_i についての微分作用素, $\theta_i = x_i \partial_i$ は x_i についての Euler 作用素である.

上で定義した微分作用素の生成する D イデアル

$$I_A(m) = D\{\ell_i^A \mid i = 1, \ldots, m\},\ I_B(m) = D\{\ell_i^B \mid i = 1, \ldots, m\},\ I_C(m) = D\{\ell_i^C \mid i = 1, \ldots, m\}$$

と $\widehat{\mathcal{D}}$ イデアル

$$\widehat{I_A}(m) = \widehat{\mathcal{D}}\{\ell_i^A \mid i = 1, \ldots, m\}, \quad \widehat{I_C}(m) = \widehat{\mathcal{D}}\{\ell_i^C \mid i = 1, \ldots, m\}.$$

と R イデアル

$$I_D(m) = R\{\ell_i^D, \ell_{jk}^D \mid i = 1, \ldots, m,\ 1 \leq j < k \leq m\}.$$

を考える. ここで $D = \mathbb{C}[x_1, \ldots, x_m]\langle \partial_1, \ldots, \partial_m \rangle$ は多項式係数微分作用素環, $\widehat{\mathcal{D}} = \mathbb{C}[[x_1, \ldots, x_m]]\langle \partial_1, \ldots, \partial_m \rangle$ は形式べき級数を係数に持つ微分作用素環, $R = \mathbb{C}(x_1, \ldots, x_m)\langle \partial_1, \ldots, \partial_m \rangle$ は有理関数係数微分作用素環である. この各イデアルについて, ある単項式順序, 項順序について, グレブナー基底がわかる. 得られたグレブナー基底を使うと, F_A, F_B, F_C の各微分方程式系の特異点集合や, F_D の Pfaff 系を計算することができる. ここでは F_B の特異点集合と F_D の Pfaff 系の計算について述べる.

2 Lauricella 超幾何微分方程式系についてのグレブナー基底

まず F_B の微分方程式系に対応する D イデアル $I_B(m)$ について, グレブナー基底を導く.

定理 1 ($I_B(m)$ のグレブナー基底)
D 上の項順序 $<_{(\mathbf{0},\mathbf{1})}$ を次のように定義する. ここで, ξ_i は ∂_i に対応する可換な変数とする.

$$x_1^{\alpha_1} \cdots x_m^{\alpha_m} \xi_1^{\beta_1} \cdots \xi_m^{\beta_m} <_{(\mathbf{0},\mathbf{1})} x_1^{\alpha'_1} \cdots x_m^{\alpha'_m} \xi_1^{\beta'_1} \cdots \xi_m^{\beta'_m}$$

を下のいづれかが成り立つ時と定義する.

1. $\beta_1 + \cdots + \beta_m < \beta'_1 + \cdots + \beta'_m$
2. $\beta_1 + \cdots + \beta_m = \beta'_1 + \cdots + \beta'_m$ かつ $\alpha_1 + \cdots + \alpha_m < \alpha'_1 + \cdots + \alpha'_m$
3. $\beta_1 + \cdots + \beta_m = \beta'_1 + \cdots + \beta'_m$ かつ $\alpha_1 + \cdots + \alpha_m = \alpha'_1 + \cdots + \alpha'_m$ かつ
   $x_1^{\alpha_1} \cdots x_m^{\alpha_m} \xi_1^{\beta_1} \cdots \xi_m^{\beta_m} <' x_1^{\alpha'_1} \cdots x_m^{\alpha'_m} \xi_1^{\beta'_1} \cdots \xi_m^{\beta'_m}$. ここで $<'$ はあらかじめ設定した適当な項順序, 例えば辞書式順序などである.

この時, D イデアル $I_B(m)$ の $<_{(\mathbf{0},\mathbf{1})}$ についてのグレブナー基底は, $\{\ell_1^B, \ldots, \ell_m^B\}$ である.

これを示すには, Buchberger の判定法を用いて, 各ペアの S 式について 0 に簡約できることを示す必要がある. これを簡単に言うため, 次の補題を用いる. この補題は多項式環ではよく知られた補題の微分作用素環版である.

補題 **2 (Buchberger の省ける S 式の判定法 (微分作用素環版))**
微分作用素 $P, Q \in D$, D 上の項順序を $<$ とする. 先頭項 $\mathrm{in}_<(P), \mathrm{in}_<(Q)$ が互いに素の時, P と Q の S 式 $S_<(P,Q)$ は交換子 $-[P,Q]$ まで簡約できる.

証明 簡単のため P, Q の先頭項の係数を 1 と仮定しておく. 先頭項 $\mathrm{in}_<(P), \mathrm{in}_<(Q)$ が互いに素より,

$$\begin{aligned} S_<(P,Q) &= (Q - \mathrm{rest}_<(Q))P - (P - \mathrm{rest}_<(P))Q \\ &= -\mathrm{rest}_<(Q)P + \mathrm{rest}_<(P)Q + QP - PQ \\ &= -\mathrm{rest}_<(Q)P + \mathrm{rest}_<(P)Q - [P,Q] \end{aligned}$$

ここで, $\mathrm{rest}_<(P)$ は P の先頭項以外の部分を表す. よって, S 式 $S_<(P,Q)$ は P, Q を使い簡約すれば, $-[P,Q]$ まで簡約できる. ∎

証明 (定理 1 の証明) 元 ℓ_i^B, ℓ_j^B $(1 \le i < j \le m)$ について, S 式が 0 に簡約できることを示せばよい. 先頭項は $\mathrm{in}_{<_{(\mathbf{0},\mathbf{1})}}(\ell_i^B) = x_i^3\xi_i^2, \mathrm{in}_{<_{(\mathbf{0},\mathbf{1})}}(\ell_j^B) = x_j^3\xi_j^2$ で, 互いに素であるから補題 2 を使うことができる. 交換子を計算すると,

$$\begin{aligned} [\ell_i^B, \ell_j^B] &= \ell_i^B\ell_j^B - \ell_j^B\ell_i^B \\ &= x_i(\theta_i + a_i)(\theta_i + b_i)\theta_j - x_j(\theta_j + a_j)(\theta_j + b_j)\theta_i \\ &\xrightarrow[\ell_i^B, \ell_j^B]{*} \theta_i(\theta_1 + \cdots + \theta_m + c - 1)\theta_j - \theta_j(\theta_1 + \cdots + \theta_m + c - 1)\theta_i = 0 \end{aligned}$$

ここで, $\xrightarrow[\ell_i^B, \ell_j^B]{*}$ は, ℓ_i^B, ℓ_j^B を使って簡約することを表す記号である. 交換子は 0 に簡約できるので, 補題 2 より, S 式 $S_{<_{(\mathbf{0},\mathbf{1})}}(\ell_i^B, \ell_j^B)$ は 0 に簡約できる. Buchberger の判定法より, $\{\ell_1^B, \ldots, \ell_m^B\}$ は $<_{(\mathbf{0},\mathbf{1})}$ についてグレブナー基底である. ∎

次に $\widehat{\mathcal{D}}$ イデアル $\widehat{I_A}(m)$ について, グレブナー基底を導く.

定理 **3 ($\widehat{I_A}(m)$ のグレブナー基底)**
$\widehat{\mathcal{D}}$ 上の単項式順序 $<_{(\mathbf{0},\mathbf{1})}{}'$ を次のように定義する.

$$x_1^{\alpha_1}\cdots x_m^{\alpha_m}\xi_1^{\beta_1}\cdots\xi_m^{\beta_m} <_{(\mathbf{0},\mathbf{1})}{}' x_1^{\alpha'_1}\cdots x_m^{\alpha'_m}\xi_1^{\beta'_1}\cdots\xi_m^{\beta'_m}$$

を下のいづれかが成り立つ時と定義する.

1. $\beta_1 + \cdots + \beta_m < \beta'_1 + \cdots + \beta'_m$
2. $\beta_1 + \cdots + \beta_m = \beta'_1 + \cdots + \beta'_m$ かつ $\alpha_1 + \cdots + \alpha_m > \alpha'_1 + \cdots + \alpha'_m$
3. $\beta_1 + \cdots + \beta_m = \beta'_1 + \cdots + \beta'_m$ かつ $\alpha_1 + \cdots + \alpha_m = \alpha'_1 + \cdots + \alpha'_m$ かつ
   $x_1^{\alpha_1}\cdots x_m^{\alpha_m}\xi_1^{\beta_1}\cdots\xi_m^{\beta_m} <' x_1^{\alpha'_1}\cdots x_m^{\alpha'_m}\xi_1^{\beta'_1}\cdots\xi_m^{\beta'_m}$. ここで $<'$ はあらかじめ設定した適当な項順序, 例えば辞書式順序などである.

$\widehat{\mathcal{D}}$ イデアル $\widehat{I_A}(m)$ の $<_{(\mathbf{0},\mathbf{1})}{}'$ についてのグレブナー基底は, $\{\ell_1^A, \ldots, \ell_m^A\}$ である.

証明 このような単項式順序 $<_{(\mathbf{0},\mathbf{1})}{}'$ であっても, 補題 2 と同様のことが成り立つ. このことを使い, 定理を証明する. 元 ℓ_i^A, ℓ_j^A $(1 \le i < j \le m)$ について, S 式が 0 に簡約できることを示せばよい. 先頭項は $\mathrm{in}_{<_{(\mathbf{0},\mathbf{1})}{}'}(\ell_i^A) = x_i^2\xi_i^2, \mathrm{in}_{<_{(\mathbf{0},\mathbf{1})}{}'}(\ell_j^A) = x_j^2\xi_j^2$ で, 互いに素であるから補題 2 の類似を使うことができる. 交換子を計算すれば $[\ell_i^A, \ell_j^A] = 0$ となる. よって, S 式 $S_{<_{(\mathbf{0},\mathbf{1})}{}'}(\ell_i^A, \ell_j^A)$ は 0 に簡約できる. $\widehat{\mathcal{D}}$ の $<_{(\mathbf{0},\mathbf{1})}{}'$ についての Buchberger の判定法 ([2], [7]) より, $\{\ell_1^A, \ldots, \ell_m^A\}$ はグレブナー基底である. ∎

$\widehat{\mathcal{D}}$ イデアル $\widehat{I_C}(m)$ の場合も, $\widehat{I_A}(m)$ と同様の手順でグレブナー基底がわかる.

定理 4 ($\widehat{I_C(m)}$ のグレブナー基底)
$\widehat{\mathcal{D}}$ イデアル $\widehat{I_C(m)}$ の $<_{(\mathbf{0},\mathbf{1})}{}'$ についてのグレブナー基底は, $\{\ell_1^C, \ldots, \ell_m^C\}$ である.

注意 1
D イデアル $I_A(m), I_C(m)$, また F_D の満たす微分方程式系に対応する D イデアルについて, 項順序 $<_{(\mathbf{0},\mathbf{1})}$ に関するグレブナー基底は複雑であり, どうなるかはわかっていない.

R イデアル $I_D(m)$ についてのグレブナー基底は次のようになる.

定理 5 ($I_D(m)$ のグレブナー基底)
R イデアル $I_D(m)$ の全次数辞書式順序 $< (\partial_1 > \partial_2 > \cdots > \partial_m)$ についてのグレブナー基底は,

$$\{p_i^D, \ell_{jk}^D \mid i = 1, \ldots, m,\ 1 \le j < k \le m\}$$

である. ここで,

$$p_i^D = x_i(1-x_i)\partial_i^2 + \left(c - (a+b_i+1)x_i + (1-x_i)\sum_{k\neq i, 1\le k\le m} \frac{b_k x_k}{x_i - x_k}\right)\partial_i - b_i \sum_{k\neq i, 1\le k\le m} \frac{x_k(1-x_k)}{x_i - x_k}\partial_k - ab_i$$

であり, この元は ℓ_i^D から $\partial_i\partial_k$ $(i \neq k)$ を ℓ_{ik}^D $(i \neq k)$ で簡約して得られる元である.

証明　Buchberger の判定法を用いて示す. すなわち, 各 S 式 $S_<(p_i^D, p_j^D), S_<(p_i^D, \ell_{jk}^D), S_<(\ell_{ij}^D, \ell_{kl}^D)$ が 0 に簡約されることを具体的に計算して示す. 単純計算であるが, 計算は非常に煩雑なものになる. 計算の詳細は省略する. ∎

3　Lauricella 超幾何関数 F_B の特異点集合の計算

[4] では, Lauricella 超幾何関数 F_C について特異点集合を計算している. その計算に倣い, 今得られたグレブナー基底を使って F_B の特異点集合を計算する. 定理 1 より, $I_B(m)$ の $<_{(\mathbf{0},\mathbf{1})}$ についてのグレブナー基底は $\{\ell_1^B, \ldots, \ell_m^B\}$ であった. ここで, 重みベクトル $(\mathbf{0},\mathbf{1}) = (0, \ldots, 0, 1, \ldots, 1) \in \mathbb{Z}^{2m}$ とおく. すなわち, x_i の重みを 0, ξ_i の重みを 1 とおいたものである. 微分作用素 $P = \sum_{\alpha,\beta\in(\mathbb{Z}_{\ge 0})^m} c_{\alpha,\beta} x^\alpha \partial^\beta \in D$ について, $(\mathbf{0},\mathbf{1})$ イニシャルフォームとは, 全表象 $P(x,\xi) = \sum_{\alpha,\beta\in(\mathbb{Z}_{\ge 0})^m} c_{\alpha,\beta} x^\alpha \xi^\beta \in \mathbb{C}[x,\xi]$ の項で $(\mathbf{0},\mathbf{1})$ 重みについて最大のものたちの和

$$\mathrm{in}_{(\mathbf{0},\mathbf{1})}(P) = \sum_{(\mathbf{0},\mathbf{1})\cdot(\alpha,\beta) \text{ が } P(x,\xi) \text{ 中で最大}} c_{\alpha,\beta} x^\alpha \xi^\beta$$

である. D イデアル I について, $(\mathbf{0},\mathbf{1})$ イニシャルフォームイデアルとは,

$$\mathrm{in}_{(\mathbf{0},\mathbf{1})}(I) = \langle \mathrm{in}_{(\mathbf{0},\mathbf{1})}(P) \mid P \in I \rangle$$

なる $\mathbb{C}[x,\xi]$ のイデアルである. グレブナー基底の一般論から, $(\mathbf{0},\mathbf{1})$ イニシャルフォームイデアル $\mathrm{in}_{(\mathbf{0},\mathbf{1})}(I_B(m))$ の生成元として, $\mathrm{in}_{(\mathbf{0},\mathbf{1})}(\ell_1^B), \ldots, \mathrm{in}_{(\mathbf{0},\mathbf{1})}(\ell_m^B)$ が得られる. ここで, ℓ_i^B の $(\mathbf{0},\mathbf{1})$ イニシャルフォームは,

$$\mathrm{in}_{(\mathbf{0},\mathbf{1})}(\ell_i^B) = x_i\xi_i\left(x_i(1-x_i)\xi_i + \sum_{1\le j\le m, j\neq i} x_j\xi_j\right)$$

である. これを L_i^B とおいておく. これより次のことがわかる.

命題 6 ($I_B(m)$ の特性多様体)
D イデアル $I_B(m)$ の特性多様体は $\mathrm{Ch}(I_B(m)) = \mathbf{V}(L_1^B, \ldots, L_m^B)$ である.

D イデアル $I_B(m)$ について特異点集合とは,

$$\mathrm{Sing}(I_B(m)) = \pi(\mathrm{Ch}(I_B(m)) \setminus \{\xi_1 = \cdots = \xi_m = 0\})$$

であった. ここで, $\pi : \mathbb{C}^{2m} \ni (x_1, \ldots, x_m, \xi_1, \ldots, \xi_m) \mapsto (x_1, \ldots, x_m) \in \mathbb{C}^m$ なる射影である. 今, 特性多様体 $\mathrm{Ch}(I_B(m)) = \mathbf{V}(L_1^B, \ldots, L_m^B)$ と具体的に分かっている.

$$L_1^B = 0, \ldots, L_m^B = 0$$

すなわち,

$$x_i\xi_i = 0 \ \text{ or } \ x_i(1-x_i)\xi_i + \sum_{1\le k\le m, k\neq i} x_k\xi_k = 0 \quad (i = 1, \ldots, m) \tag{1}$$

の解 $(x_1, \ldots, x_m, \xi_1, \ldots, \xi_m)$ で $(\xi_1, \ldots, \xi_m) \neq (0, \ldots, 0)$ なるものを計算し, それを x 座標だけに射影したものが特異点集合になる. 上の式 (1) を $\varepsilon_i \in \{0, 1\}$ を使ってまとめて書けば,

$$x_i(1-\varepsilon_i x_i)\xi_i + \sum_{1\le k\le m, k\neq i} \varepsilon_i x_k\xi_k = 0 \quad (i = 1, \ldots, m, \varepsilon_i \in \{0, 1\}) \tag{2}$$

$\varepsilon = (\varepsilon_1, \ldots, \varepsilon_m) \in \{0, 1\}^m$ を一つ固定する. 上の式 (2) を行列表示すれば,

$$\begin{pmatrix} x_1(1-\varepsilon_1 x_1) & \varepsilon_1 x_2 & \cdots & \varepsilon_1 x_m \\ \varepsilon_2 x_1 & x_2(1-\varepsilon_2 x_2) & \cdots & \varepsilon_2 x_m \\ \vdots & & & \vdots \\ \varepsilon_m x_1 & \varepsilon_m x_2 & \cdots & x_m(1-\varepsilon_m x_m) \end{pmatrix} \begin{pmatrix} \xi_1 \\ \xi_2 \\ \vdots \\ \xi_m \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix}$$

上の行列を A_ε とおく. 式 (2) が $(\xi_1, \ldots, \xi_m) \neq (0, \ldots, 0)$ なる解を持つための必要十分条件は, $\det(A_\varepsilon) = 0$ である. これは ε を固定した時なので, ε を $\{0, 1\}^m$ 全体を動かせば, 特異点集合の定義方程式が出てくる. $\prod_{\varepsilon\in\{0,1\}^m} \det(A_\varepsilon)$ を計算すれば,

$$\begin{aligned} \prod_{\varepsilon\in\{0,1\}^m} \det(A_\varepsilon) =& x_1^{2^m} \cdots x_m^{2^m} \prod_{1\le i_1\le m} (1-x_{i_1}) \prod_{1\le i_1<i_2\le m} (x_{i_1}x_{i_2} - x_{i_1} - x_{i_2}) \cdots \\ & (-1)^{m-1}(-x_1x_2\cdots x_m + x_2\cdots x_m + \cdots + x_1\cdots x_{m-1}) \end{aligned}$$

となる.

定理 7 (F_B の特異点集合)
F_B の特異点集合は,

$$\begin{aligned} \mathrm{Sing}(I_B(m)) =& \mathbf{V}(x_1\cdots x_m \prod_{1\le i_1\le m} (1-x_{i_1}) \prod_{1\le i_1<i_2\le m} (x_{i_1}x_{i_2} - x_{i_1} - x_{i_2}) \cdots \\ & (x_1x_2\cdots x_m - x_2\cdots x_m - \cdots - x_1\cdots x_{m-1})) \end{aligned}$$

で与えられる.

4 Lauricella 超幾何関数 F_D の Pfaff 系の計算

R イデアルが 0 次元イデアルであり, そのグレブナー基底がわかっている時, グレブナー基底を使って, その微分方程式系の Pfaff 系を計算することが可能である ([1]). 定理 5 より, R イデアル $I_D(m)$ は 0 次元イデアル (standard monomial は $1, \partial_1, \ldots, \partial_m$) であり, グレブナー基底がわかっているので, このグレブナー基底から Pfaff 系を計算することができる. 一般に m 変数の場合にわかるが, 簡単のため $m = 2$ 変数の例を示す.

例 **1** ($I_D(2)$ の **Pfaff** 系)
定理 5 より, R イデアル $I_D(2)$ のグレブナー基底 G は,

$$
\begin{aligned}
p_1^D &= x_1(1-x_1)\partial_1^2 + \left(c - (a+b_1+1)x_1 + (1-x_1)\frac{b_2 x_2}{x_1 - x_2}\right)\partial_1 - b_1\frac{x_2(1-x_2)}{x_1-x_2}\partial_2 - ab_1 \\
p_2^D &= x_2(1-x_2)\partial_2^2 + \left(c - (a+b_2+1)x_2 - (1-x_2)\frac{b_1 x_1}{x_1 - x_2}\right)\partial_2 + b_2\frac{x_1(1-x_1)}{x_1-x_2}\partial_1 - ab_2 \\
\ell_{12}^D &= (x_1-x_2)\partial_1\partial_2 - b_2\partial_1 + b_1\partial_2
\end{aligned}
$$

である. 各先頭項は,

$$
\mathrm{in}_<(p_1^D) = \xi_1^2, \quad \mathrm{in}_<(p_2^D) = \xi_2^2, \quad \mathrm{in}_<(\ell_{12}^D) = \xi_1\xi_2
$$

であり, $R/I_D(2)$ の *standard monomial* は $1, \partial_1, \partial_2$ となる. このことから, *Pfaff* 系は

$$
\partial_k \begin{pmatrix} F_D \\ \partial_1 \cdot F_D \\ \partial_2 \cdot F_D \end{pmatrix} = \begin{pmatrix} a_{11}^k & a_{12}^k & a_{13}^k \\ a_{21}^k & a_{22}^k & a_{23}^k \\ a_{31}^k & a_{32}^k & a_{33}^k \end{pmatrix} \begin{pmatrix} F_D \\ \partial_1 \cdot F_D \\ \partial_2 \cdot F_D \end{pmatrix} \quad (k = 1, 2)
$$

のような形になることがわかる. ここで $a_{ij}^k \in \mathbb{C}(x_1, x_2)$ は有理関数である. 1 行目は自明な関係式であり, 2,3 行目はそれぞれ $\partial_k\partial_1, \partial_k\partial_2$ をグレブナー基底 G で割った余りから導かれる. こうして得られる *Pfaff* 系は次のようになる.

$$
\partial_1 \begin{pmatrix} F_D \\ \partial_1 \cdot F_D \\ \partial_2 \cdot F_D \end{pmatrix} = \begin{pmatrix} 0 & 1 & 0 \\ \frac{ab_1}{x_1(1-x_1)} & -\frac{b_2(1-x_1)x_2 + (x_1-x_2)(c-(a+b_1+1)x_1)}{x_1(1-x_1)(x_1-x_2)} & \frac{b_1 x_2(1-x_2)}{x_1(1-x_1)(x_1-x_2)} \\ 0 & \frac{b_2}{x_1-x_2} & -\frac{b_1}{x_1-x_2} \end{pmatrix} \begin{pmatrix} F_D \\ \partial_1 \cdot F_D \\ \partial_2 \cdot F_D \end{pmatrix}
$$

$$
\partial_2 \begin{pmatrix} F_D \\ \partial_1 \cdot F_D \\ \partial_2 \cdot F_D \end{pmatrix} = \begin{pmatrix} 0 & 0 & 1 \\ 0 & \frac{b_2}{x_1-x_2} & -\frac{b_1}{x_1-x_2} \\ \frac{ab_2}{x_2(1-x_2)} & -\frac{b_2 x_1(1-x_1)}{x_2(1-x_2)(x_1-x_2)} & -\frac{-b_1(1-x_2)x_1 + (x_1-x_2)(c-(a+b_2+1)x_2)}{x_2(1-x_2)(x_1-x_2)} \end{pmatrix} \begin{pmatrix} F_D \\ \partial_1 \cdot F_D \\ \partial_2 \cdot F_D \end{pmatrix}
$$

参 考 文 献

[1] JST CREST 日比チーム, グレブナー道場, 共立出版, (2011)

[2] F. Castro, Calculs effectifs pour les idéaux d'opérateurs différentiels, Travaux en Cours 24, (1987), 1 - 19

[3] T. Koyama, H. Nakayama, K. Nishiyama, N. Takayama, The Holonomic Rank of the Fisher-Bingham System of Differential Equations, `arXiv:1205.6144`

[4] R. Hattori, N. Takayama, The singular locus of Lauricella's F_C, Journal of Mathematical Society of Japan (to appear), `arXiv:1110.6675`

[5] K. Matsumoto, Appell and Lauricella Hypergeometric Functions, preprint.

[6] H. Nakayama, Gröbner basis and singular locus of Lauricella's hypergeometric equations, Kyushu Journal of Mathematics (to appear), arXiv:1303.1674

[7] 大阿久俊則, グレブナ基底と線型偏微分方程式系 (計算代数解析入門), 上智大学数学講究録, No. 38, (1994)

[8] T. Oaku, T. Shimoyama, A Gröbner Basis Method for Modules over Rings of Differential Operators, Journal of Symbolic Computation 18 (1994), 223–248.

[9] T. Oaku, Computation of the characteristic variety and the singular locus of a system of differential equations with polynomial coefficients, Japan Journal of Industrial and Applied Mathematics 11 (1994), no. 3, 485–497

[10] M. Saito, B. Sturmfels, N. Takayama, *Gröbner Deformations of Hypergeometric Differential Equations*, Springer, 2000

Markov chain Monte Carlo methods for the regular two-level fractional factorial designs and cut ideals

Satoshi Aoki*† ‡, Hidefumi Ohsugi§†and Takayuki Hibi¶†

Abstract

It is known that a Markov basis of the binary graph model of a graph G corresponds to a set of binomial generators of cut ideals $I_{\widehat{G}}$ of the suspension $\widehat{G}$ of G. In this note, we give another application of cut ideals to statistics. We show that a set of binomial generators of cut ideals is a Markov basis of some regular two-level fractional factorial design. As application, we give a Markov basis of degree 2 for designs defined by at most two relations. This note is a summary of the paper [2].

1 Introduction

In the paper [2], following the Markov chain Monte Carlo approach in the designed experiments by [3], we give a new results on the correspondence between the regular two-level design and the algebraic concept, namely *cut ideals* defined in [10]. This note is a summary of the paper [2]. Because the Markov bases are characterized as the generators of well-specified toric ideals and are studied not only by statisticians but also by algebraists, it is valuable to connect statistical models to known class of toric ideals. In this note, we give a fundamental fact that the generator of cut ideals can be characterized as the Markov bases for the testing problems of log-linear models for the two-level regular fractional factorial designs.

2 Markov chain Monte Carlo method for regular two-level fractional factorial designs

In this section we introduce Markov chain Monte Carlo methods for testing the fitting of the log-linear models for regular two-level fractional factorial designs with count observations. Suppose we have nonnegative integer observations for each run of a regular

*Graduate School of Science and Engineering (Science Course), Kagoshima University.

†JST, CREST.

‡Email: aoki@sci.kagosgima-u.ac.jp

§Department of Mathematics, College of Science, Rikkyo University.

¶Department of Pure and Applied Mathematics, Graduate School of Information Science and Technology, Osaka University.

Table 1: Design and number of defects y for the wave-solder experiment

	Factor							y		
Run	**A**	**B**	**C**	**D**	**E**	**F**	**G**	1	2	3
1	1	1	1	1	1	1	1	13	30	26
2	1	1	1	2	2	2	2	4	16	11
3	1	1	2	1	1	2	2	20	15	20
4	1	1	2	2	2	1	1	42	43	64
5	1	2	1	1	2	1	2	14	15	17
6	1	2	1	2	1	2	1	10	17	16
7	1	2	2	1	2	2	1	36	29	53
8	1	2	2	2	1	1	2	5	9	16
9	2	1	1	1	2	2	1	29	0	14
10	2	1	1	2	1	1	2	10	26	9
11	2	1	2	1	2	1	2	28	173	19
12	2	1	2	2	1	2	1	100	129	151
13	2	2	1	1	1	2	2	11	15	11
14	2	2	1	2	2	1	1	17	2	17
15	2	2	2	1	1	1	1	53	70	89
16	2	2	2	2	2	2	2	23	22	7

fractional design. For simplicity, we also suppose that the observations are counts of some events and only one observation is obtained for each run. This is natural for the settings of Poisson sampling scheme, since the set of the totals for each run is the sufficient statistics for the parameters. We begin with an example.

Example 1 (Wave-soldering experiment)**.** Table 1 is a 1/8 fraction of a full factorial design (i.e., a 2^{7-3} fractional factorial design) defined from the defining relation

$$\mathbf{ABDE} = \mathbf{ACDF} = \mathbf{BCDG} = \mathbf{I}, \tag{1}$$

and response data analyzed in [4] and reanalyzed in [7]. In Table 1, the observation y is the number of defects arising in a wave-soldering process in attaching components to an electronic circuit card. In Chapter 7 of [4], he considered seven factors of a wave-soldering process: (A) prebake condition, (B) flux density, (C) conveyer speed, (D) preheat condition, (E) cooling time, (F) ultrasonic solder agitator and (G) solder temperature, each at two levels with three boards from each run being assessed for defects. The aim of this experiment is to decide which levels for each factors are desirable to reduce solder defects.

Because we only consider designs with a single observation for each run in [2], we focus on the totals for each run in Table 1. We also ignore the second observation in run 11, which is an obvious outlier as pointed out in [7]. Therefore the weighted total of run 11 is $(28 + 19) \times 3/2 = 70.5 \simeq 71$. By replacing 2 by -1 in Table 1, we rewrite $k \times p$ design

matrix as D, where each element is $+1$ or -1. Consequently, we have

$$D = \begin{pmatrix} +1 & +1 & +1 & +1 & +1 & +1 & +1 \\ +1 & +1 & +1 & -1 & -1 & -1 & -1 \\ +1 & +1 & -1 & +1 & +1 & -1 & -1 \\ & & & \vdots & & & \\ -1 & -1 & -1 & +1 & +1 & +1 & +1 \\ -1 & -1 & -1 & -1 & -1 & -1 & -1 \end{pmatrix}, \quad \mathbf{y} = \begin{pmatrix} 69 \\ 31 \\ 55 \\ \vdots \\ 212 \\ 52 \end{pmatrix}.$$

In [2], we consider designs of p factors with two-level. We write the observations as $\mathbf{y} = (y_1, \ldots, y_k)'$, where k is the run size and $'$ denotes the transpose. Write the design matrix $D = (d_{ij})$, where $d_{ij} \in \{-1, 1\}$ is the level of the j-th factor in the i-th run for $i = 1, \ldots, k, j = 1, \ldots, p$.

In this case it is natural to consider the Poisson distribution as the sampling model, in the framework of generalized linear models ([9]). The observations $\mathbf{y}$ are realizations from k Poisson random variables $Y_1, \ldots, Y_k$, which are mutually independently distributed with the mean parameter $\mu_i = E(Y_i), i = 1, \ldots, k$. We call the log-linear model written by

$$\log \mu_i = \beta_0 + \beta_i d_{i1} + \cdots + \beta_p d_{ip}, \ i = 1, \ldots, k \tag{2}$$

as the main effect model in [2]. The equivalent model in the matrix form is $(\log \mu_1 \ \cdots \ \log \mu_k)' = M\beta$, where $\beta = (\beta_0, \beta_1, \ldots, \beta_p)'$, $\mathbf{1} = (1, \ldots, 1)'$ and

$$M = \begin{pmatrix} \mathbf{1} & D \end{pmatrix}. \tag{3}$$

We call the $k \times (p+1)$ matrix M a *model matrix* of the main effect model. The interpretation of the parameter β_j in (2) is the parameter contrast for the main effect of the j-th factor. To consider the models including various interaction effects, see [3].

To judge the fitting of the main effect model (2), we can perform various goodness-of-fit tests. In the goodness-of-fit tests, the main effect model (2) is treated as the null model, whereas the saturated model is treated as the alternative model. Under the null model (2), β is the nuisance parameter and the sufficient statistic for β is given by $M'\mathbf{y} = (\sum_{i=1}^k y_i, \sum_{i=1}^k d_{i1} y_i, \ldots, \sum_{i=1}^k d_{ip} y_i)'$. Then the conditional distribution of $\mathbf{y}$ given the sufficient statistics is written as

$$f(\mathbf{y} \mid M'\mathbf{y} = M'\mathbf{y}^o) = \frac{1}{C(M'\mathbf{y}^o)} \prod_{i=1}^k \frac{1}{y_i!}, \tag{4}$$

where $\mathbf{y}^o$ is the observation count vector and $C(M'\mathbf{y}^o)$ is the normalizing constant determined from $M'\mathbf{y}^o$ written as

$$C(M'\mathbf{y}^o) = \sum_{\mathbf{y} \in \mathcal{F}(M'\mathbf{y}^o)} \left(\prod_{i=1}^k \frac{1}{y_i!} \right), \tag{5}$$

and

$$\mathcal{F}(M'\mathbf{y}^o) = \{\mathbf{y} \mid M'\mathbf{y} = M'\mathbf{y}^o, \ y_i \text{ is a nonnegative integer for } i = 1, \ldots, k\}. \tag{6}$$

Note that by sufficiency the conditional distribution does not depend on the values of the nuisance parameters.

In [2], we consider various goodness-of-fit tests based on the conditional distribution (4). There are several ways to choose the test statistics. For example, the likelihood ratio statistic

$$T(\mathbf{y}) = G^2(\mathbf{y}) = 2\sum_{i=1}^{k} y_i \log \frac{y_i}{\hat{\mu}_i} \tag{7}$$

is frequently used, where $\hat{\mu}_i$ is the maximum likelihood estimate for μ_i under the null model (i.e., fitted value). Note that the traditional asymptotic test evaluates the upper probability for the observed value $T(\mathbf{y}^o)$ based on the asymptotic distribution χ^2_{k-p-1}. However, since the fitting of the asymptotic approximation may be sometimes poor, we consider Markov chain Monte Carlo methods to evaluate the p values. Using the conditional distribution (4), the exact p value is written as

$$p = \sum_{\mathbf{y} \in \mathcal{F}(M'\mathbf{y}^o)} f(\mathbf{y} \mid M'\mathbf{y} = M'\mathbf{y}^o)\mathbf{1}(T(\mathbf{y}) \geq T(\mathbf{y}^o)), \tag{8}$$

where

$$\mathbf{1}(T(\mathbf{y}) \geq T(\mathbf{y}^o)) = \begin{cases} 1, & \text{if } T(\mathbf{y}) \geq T(\mathbf{y}^o), \\ 0, & \text{otherwise.} \end{cases} \tag{9}$$

Of course, if we can calculate the exact p value of (8) and (9), it is best. Unfortunately, however, an enumeration of all the elements in $\mathcal{F}(M'\mathbf{y}^o)$ and hence the calculation of the normalizing constant $C(M'\mathbf{y}^o)$ is usually computationally infeasible for large sample space. Instead, we consider a Markov chain Monte Carlo method. Note that, as one of the important advantages of Markov chain Monte Carlo method, we need not calculate the normalizing constant (5) to evaluate p values.

To perform the Markov chain Monte Carlo procedure, we have to construct a connected, aperiodic and reversible Markov chain over the conditional sample space (6) with the stationary distribution (4). If such a chain is constructed, we can sample from the chain as $\mathbf{y}^{(1)}, \ldots, \mathbf{y}^{(T)}$ after discarding some initial burn-in steps, and evaluate p values as

$$\hat{p} = \frac{1}{T}\sum_{t=1}^{T} \mathbf{1}(T(\mathbf{y}^{(t)}) \geq T(\mathbf{y}^o)).$$

Such a chain can be constructed easily by *Markov basis*. Once a Markov basis is calculated, we can construct a connected, aperiodic and reversible Markov chain over the space (6), which can be modified so that the stationary distribution is the conditional distribution (4) by the Metropolis-Hastings procedure. See [5] and [8] for details.

Markov basis is characterized algebraically as follows. Write indeterminates $x_1, \ldots, x_k$ and consider polynomial ring $K[x_1, \ldots, x_k]$ for some field K. Consider the integer kernel of the transpose of the model matrix M, $\mathrm{Ker}_{\mathbb{Z}} M'$. For each $\mathbf{b} = (b_1, \ldots, b_k)' \in \mathrm{Ker}_{\mathbb{Z}} M'$, define binomial in $K[x_1, \ldots, x_k]$ as

$$f_{\mathbf{b}} = \prod_{b_j > 0} x_j^{b_j} - \prod_{b_j < 0} x_j^{-b_j}.$$

Then the binomial ideal in $K[x_1, \ldots, x_k]$,

$$I(M') = \langle \{ f_{\mathbf{b}} \mid \mathbf{b} \in \mathrm{Ker}_{\mathbb{Z}} M' \} \rangle ,$$

is called a toric ideal with the configuration M'. Let $\{f_{\mathbf{b}^{(1)}}, \ldots, f_{\mathbf{b}^{(s)}}\}$ be any generating set of $I(M')$. Then the set of integer vectors $\{\mathbf{b}^{(1)}, \ldots, \mathbf{b}^{(s)}\}$ constitutes a Markov basis. See [5] for detail. To compute a Markov basis for given configuration M', we can rely on various algebraic softwares such as 4ti2 ([1]). See the following example.

Example 2 (Wave-soldering experiment, continued)**.** We analyze the data in Table 1. The fitted value under the main effect model is calculated as

$$\begin{aligned} \hat{\mu} = (&68.87,\ 19.70,\ 78.85,\ 147.59,\ 12.14,\ 54.77,\ 104.53,\ 54.54,\\ &75.31,\ 39.29,\ 75.00,\ 338.37,\ 27.83,\ 52.09,\ 208.47,\ 59.64)'. \end{aligned}$$

Then the likelihood ratio for the observed data is calculated as $T(\mathbf{y}^o) = G^2(\mathbf{y}^o) = 117.81$ and the corresponding asymptotic p value is less than 0.0001 from the asymptotic distribution χ^2_8. This result tells us that the null hypothesis is highly significant and is rejected, i.e., the existence of some interaction effects is suggested. To evaluate the p value by Markov chain Monte Carlo method, we have to calculate a Markov basis first. If we use 4ti2, we prepare the data file (configuration M') as

```
8 16
 1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1
 1  1  1  1  1  1  1  1 -1 -1 -1 -1 -1 -1 -1 -1
 1  1  1  1 -1 -1 -1 -1  1  1  1  1 -1 -1 -1 -1
 1  1 -1 -1  1  1 -1 -1  1  1 -1 -1  1  1 -1 -1
 1 -1  1 -1  1 -1  1 -1  1 -1  1 -1  1 -1  1 -1
 1 -1  1 -1 -1  1 -1  1 -1  1 -1  1  1 -1  1 -1
 1 -1 -1  1  1 -1 -1  1 -1  1  1 -1 -1  1  1 -1
 1 -1 -1  1 -1  1  1 -1  1 -1 -1  1 -1  1  1 -1
```

and run the command `markov`. Then we have a minimal Markov basis with 77 elements as follows.

```
77 16
 0  0  0  0  0  0  0  0  1  1 -1 -1 -1 -1  1  1
 0  0  0  0  0  1 -1  0  1  0  0 -1 -1 -1  1  1
 0  0  0  0  0  1  0 -1  0  1  0 -1 -1 -1  1  1
.....
```

Using this Markov basis, we can evaluate p value by Markov chain Monte Carlo method. After $50,000$ burn-in-steps from $\mathbf{y}^o$ itself as the initial state, we sample $100,000$ Monte Carlo sample by Metropolis-Hasting algorithm, which yields $\hat{p} = 0.0000$ again. Figure 1 is a histogram of the Monte Carlo sampling of the likelihood ratio statistic under the main effect model, along with the corresponding asymptotic distribution χ^2_8.

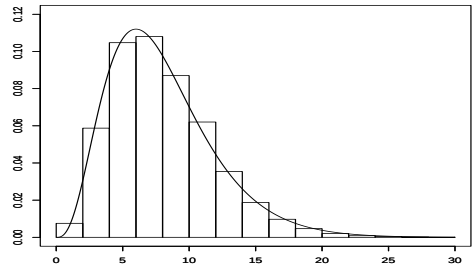


Figure 1: Asymptotic and Monte Carlo estimated distribution of likelihood ratio

3 Two-level regular fractional factorial designs and cut ideals

In this section, we show that a cut ideal for a finite connected graph can be characterized as the toric ideal $I(M')$ for a model matrix of the main effect model for some regular two-level fractional factorial designs.

3.1 Cut ideals

We start with the definition of the cut ideal. Consider a connected finite graph $G = (V, E)$. We also consider unordered partitions $A|B$ of the vertex set V. Let $\mathcal{P}(V)$ be the set of the unordered partitions of V, i.e., $\mathcal{P}(V) = \{A|B \mid A \cup B = V,\ A \cap B = \emptyset\}$. We introduce the sets of indeterminates $\{s_{ij} \mid \{i,j\} \in E\}$, $\{t_{ij} \mid \{i,j\} \in E\}$ and $\{q_{A|B} \mid A|B \in \mathcal{P}(V)\}$. Let $K[\mathbf{q}] = K[q_{A|B} \mid A|B \in \mathcal{P}(V)]$, $K[\mathbf{s}, \mathbf{t}] = K[s_{ij}, t_{ij} \mid \{i,j\} \in E]$ be polynomial rings over a field K. For each partition $A|B \in \mathcal{P}(V)$, we define a subset $\mathrm{Cut}(A|B)$ of the edge set E as $\mathrm{Cut}(A|B) = \{\{i,j\} \in E \mid i \in A, j \in B \text{ or } i \in B, j \in A\}$. Define homomorphism of polynomial rings as

$$\phi_G :\ K[\mathbf{q}] \to K[\mathbf{s}, \mathbf{t}],\quad q_{A|B} \mapsto \prod_{\{i,j\} \in \mathrm{Cut}(A|B)} s_{ij} \cdot \prod_{\{i,j\} \in E \setminus \mathrm{Cut}(A|B)} t_{ij}. \tag{10}$$

We may think of $\mathbf{s}$ and $\mathbf{t}$ as abbreviations for "separated" and "together", respectively. Then the cut ideal of the graph G is defined as $I_G = \mathrm{Ker}(\phi_G)$. We also use the following two examples given in [10].

Example 3 (Complete graph on four vertices)**.** Let $G = K_4$ be the complete graph on four vertices $V = \{1, 2, 3, 4\}$. Then the edge set is $E = \{12, 13, 14, 23, 24, 34\}$. The map ϕ_{K_4} is specified by

$$\begin{aligned}
q_{\emptyset|1234} &\mapsto t_{12}t_{13}t_{14}t_{23}t_{24}t_{34} & \qquad q_{4|123} &\mapsto t_{12}t_{13}s_{14}t_{23}s_{24}s_{34}\\
q_{1|234} &\mapsto s_{12}s_{13}s_{14}t_{23}t_{24}t_{34} & \qquad q_{12|34} &\mapsto t_{12}s_{13}s_{14}s_{23}s_{24}t_{34}\\
q_{2|134} &\mapsto s_{12}t_{13}t_{14}s_{23}s_{24}t_{34} & \qquad q_{13|24} &\mapsto s_{12}t_{13}s_{14}s_{23}t_{24}s_{34}\\
q_{3|124} &\mapsto t_{12}s_{13}t_{14}s_{23}t_{24}s_{34} & \qquad q_{14|23} &\mapsto s_{12}s_{13}t_{14}t_{23}s_{24}s_{34}.
\end{aligned}$$

In this case, the cut ideal is a principal ideal given by

$$I_{K_4} = \left\langle q_{\emptyset|1234}q_{12|34}q_{13|24}q_{14|23} - q_{1|234}q_{2|134}q_{3|124}q_{4|123} \right\rangle.$$

Example 4 (4-cycle)**.** Let $G = C_4$ be the 4-cycle with $V = \{1, 2, 3, 4\},\ E = \{12, 23, 34, 14\}$. The map ϕ_{C_4} is derived from ϕ_{K_4} in Example 3 by setting $s_{13} = t_{13} = s_{24} = t_{24} = 1$ as

$$\begin{array}{rclcrcl}
q_{\emptyset|1234} & \mapsto & t_{12}t_{14}t_{23}t_{34} & \qquad & q_{4|123} & \mapsto & t_{12}s_{14}t_{23}s_{34} \\
q_{1|234} & \mapsto & s_{12}s_{14}t_{23}t_{34} & & q_{12|34} & \mapsto & t_{12}s_{14}s_{23}t_{34} \\
q_{2|134} & \mapsto & s_{12}t_{14}s_{23}t_{34} & & q_{13|24} & \mapsto & s_{12}s_{14}s_{23}s_{34} \\
q_{3|124} & \mapsto & t_{12}t_{14}s_{23}s_{34} & & q_{14|23} & \mapsto & s_{12}t_{14}t_{23}s_{34}.
\end{array}$$

In this case, the cut ideal is given by

$$I_{C_4} = \left\langle q_{\emptyset|1234}q_{13|24} - q_{1|234}q_{3|124},\ q_{\emptyset|1234}q_{13|24} - q_{2|134}q_{4|123},\ q_{\emptyset|1234}q_{13|24} - q_{12|34}q_{14|23} \right\rangle.$$

Now we relates the cut ideals to the regular two-level fractional factorial designs. We express the map ϕ_G by $2^{|V|-1} \times 2|E|$ matrix $H = \{h_{A|B,e}\}$ where each row of H represents $A|B \in \mathcal{P}(V)$ and each two columns of H represents E as

$$h_{A|B,e} = \begin{cases} (1,0) & \text{if } e \in E \setminus \mathrm{Cut}(A|B) \\ (0,1) & \text{if } e \in \mathrm{Cut}(A|B). \end{cases}$$

Note that there are $|\mathcal{P}(V)| = 2^{|V|-1}$ unordered partitions of V. We also see that each two columns of H correspond to $\mathbf{t}$ and $\mathbf{s}$. Then the cut ideal, the kernel of ϕ_G of (10), is written as the toric ideal of the configuration matrix H'.

Example 5 (4-cycle, continued)**.** For the case of $G = C_4$ of Example 4, the matrix H can be written as follows.

$$\begin{array}{r|cccccccc}
 & t_{12} & s_{12} & t_{14} & s_{14} & t_{23} & s_{23} & t_{34} & s_{34} \\ \hline
q_{\emptyset|1234} & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \\
q_{3|124} & 1 & 0 & 1 & 0 & 0 & 1 & 0 & 1 \\
q_{4|123} & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 1 \\
q_{12|34} & 1 & 0 & 0 & 1 & 0 & 1 & 1 & 0 \\
q_{14|23} & 0 & 1 & 1 & 0 & 1 & 0 & 0 & 1 \\
q_{2|134} & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 \\
q_{1|234} & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 0 \\
q_{13|24} & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1
\end{array} \tag{11}$$

The kernel of H' coincides to the kernel of M' of (3) for the two-level design D of $|E|$ factors with $2^{|V|-1}$ runs, where the level of the factor X_e for the run $A|B \in \mathcal{P}(V)$ is given by the following map:

$$\begin{array}{rccc}
X_e: & \mathcal{P}(V) & \to & \{+1, -1\} \\
 & \cup & & \cup \\
 & A|B & \mapsto & \begin{cases} +1 & \text{if } e \in E \setminus \mathrm{Cut}(A|B) \\ -1 & \text{if } e \in \mathrm{Cut}(A|B) \end{cases}
\end{array} \tag{12}$$

Example 6 (4-cycle, continued)**.** For the case of $G = C_4$, the map X_e of (12) gives the design matrix D as follows.

	X_{12}	X_{14}	X_{23}	X_{34}
$q_{\emptyset\|1234}$	1	1	1	1
$q_{3\|124}$	1	1	−1	−1
$q_{4\|123}$	1	−1	1	−1
$q_{12\|34}$	1	−1	−1	1
$q_{14\|23}$	−1	1	1	−1
$q_{2\|134}$	−1	1	−1	1
$q_{1\|234}$	−1	−1	1	1
$q_{13\|24}$	−1	−1	−1	−1

For this D, it is easily seen that $\mathrm{Ker}(M')$ coincides to $\mathrm{Ker}(H')$ if H is given by (11).

3.2 Regular designs and cut ideals

In Example 6, we obtain the toric ideal for the main effect model of the regular two-level fractional factorial designs defined by $X_{12}X_{14}X_{23}X_{34} = 1$ from the the cut ideal of $G = C_4$. In fact, there is a clear relation between finite connected graphs G and regular two-level designs D. As we have seen in Example 6, the cut ideal for G can be related to the design of $p = |E|$ factors with $k = 2^{|V|-1}$ runs. Since each factor of this design corresponds to the edge E of G, we write each factor X_{ij} for $\{i, j\} \in E$. Since there are 2^p runs in the full factorial design of p factors, the design obtained from G by the relation (12) is a $2^{|V|-1-p}$ fraction of the full factorial design of p factors. We show this fraction is specified as the regular fractional factorial designs.

Let $G = (V, E)$ be a finite connected graph with the edge set $E = \{e_1, \ldots, e_p\}$. Then, the *cycle space* $\mathcal{C}(G)$ of G is a subspace of $\mathbb{F}_2^{|E|}$ spanned by

$$\left\{ \mathbf{e}_{i_1} + \cdots + \mathbf{e}_{i_r} \in \mathbb{F}_2^{|E|} \;\middle|\; (e_{i_1}, \ldots, e_{i_r}) \text{ is a cycle of } G \right\},$$

where $\mathbf{e}_j$ is the jth coordinate vector of $\mathbb{F}_2^{|E|}$. On the other hand, the *cut space* $\mathcal{C}^*(G)$ of G is a subspace of $\mathbb{F}_2^{|E|}$ defined by

$$\mathcal{C}^*(G) = \left\{ \sum_{e_j \in \mathrm{Cut}(A|B)} \mathbf{e}_j \in \mathbb{F}_2^{|E|} \;\middle|\; A|B \in \mathcal{P}(V) \right\}.$$

Fix a spanning tree T of G. For each $e \in E \setminus T$, the set $T \cup \{e\}$ has exactly one cycle C_e of G. Such a cycle C_e is called a *fundamental cycle* of G. Since T has $|V| - 1$ edges, there are $|E| - |V| + 1$ edges in $E \setminus T$. It then follows that there exists $|E| - |V| + 1$ fundamental cycles in G.

Theorem 7. *Let $G = (V, E)$ be a finite connected graph and let D be the design matrix of $|E|$ factors with $2^{|V|-1}$ runs defined by (12). Then D is a regular fractional factorial design with all relations*

$$X_{e_{i_1}}(A|B)X_{e_{i_2}}(A|B)\cdots X_{e_{i_m}}(A|B) = 1, \tag{13}$$

where $(e_{i_1}, \ldots, e_{i_m})$ is a fundamental cycle of G.

Theorem 7 shows the relation of the cut ideals and regular two-level fractional factorial designs. For a given connected finite graph, we can consider corresponding regular two-level fractional factorial designs from Theorem 7. Unfortunately, however, the converse does not always hold. For given regular two-level fractional factorial designs (strictly, we should say that "for given designs and *models*"), it does not always exist corresponding connected finite graphs.

Proposition 8. *If a 2^{p-q} design corresponds to a finite graph by the relation (12), then we have $p \leq \binom{p-q+1}{2}$.*

Thus, obvious counterexamples for the converse are given since some regular 2^{p-q} designs satisfy $\binom{p-q+1}{2} < p$ (for example, $(p,q) = (5,3), (5,4), (6,4), (6,5)$ and so on). On the other hand, a necessary condition related with the resolution is as follows.

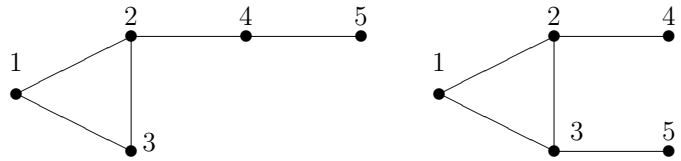
Proposition 9. *If a 2^{p-q} design of resolution* IV *or more corresponds to a finite graph by the relation (12), then we have $p \leq \lfloor (p-q+1)^2/4 \rfloor$.*

If the resolution of a design is V or more, then similar results are obtained by the results in [6]. From these considerations, an important question arises.

Question 10. *Characterize regular two-level fractional factorial designs that can correspond to a finite graph by the relation (12).*

A complete answer to this question is not yet obtained at present. We present several fundamental characterizations in the rest of this section. Note that the above correspondence is not one-to-one even if it exists. In fact, for any finite connected graph G, we can specify a design D uniquely by (13). However, for a given design G, we can consider several graphs satisfying the relation (13) if it exists.

Example 11 (2^{5-1} design with $X_{12}X_{13}X_{23} = 1$ of 5 factors)**.** Consider 2^{5-1} fractional factorial design $X_{12}X_{13}X_{23} = 1$ of 5 factors, or, $\mathbf{ABC} = \mathbf{I}$ in the convention of designed experiment literature. There are several corresponding graphs that give this design such as follows.



Now we show two important special cases, designs corresponding to complete graphs and trees.

Corollary 12. *Let $G = K_n$ be the complete graph on $|V| = n$ vertices. Then, G is specified as the regular $2^{c_1-c_2}$ fractional factorial design of c_1 two-level factors by (13), where*

$$c_1 = \binom{n}{2}, \quad c_2 = \binom{n-1}{2}.$$

The defining relation of this design is written as $X_{1i}X_{1j}X_{ij} = 1$ for any pair (i,j) with $2 \leq i < j \leq n$.

Another important case is as follows.

Corollary 13. *Any spanning tree $G = (V, E)$ is specified as the full factorial design of $|V| - 1$ two-level factors by (13).*

4 Discussion

We apply known results on cut ideals to the regular two-level fractional factorial designs. See [2] for details.

References

[1] 4ti2 team. 4ti2 – A software package for algebraic, geometric and combinatorial problems on linear spaces. Available at `www.4ti2.de`.

[2] S. Aoki, T. Hibi and H. Ohsugi (2013). Markov chain Monte Carlo methods for the regular two-level fractional factorial designs and cut ideals. *Journal of Statistical Planning and Inference*, to appear.

[3] S. Aoki and A. Takemura (2010). Markov chain Monte Carlo tests for designed experiments. *Journal of Statistical Planning and Inference*, **140**, 817–830.

[4] L. W. Condra (1993). *Reliability Improvement with Design of Experiments*. Marcel Dekker, New York, NY., 1993.

[5] P. Diaconis and B. Sturmfels (1998). Algebraic algorithms for sampling from conditional distributions. *Annals of Statistics*, **26**, 363–397.

[6] D. K. Garnick, Y. H. H. Kwong and F, Lazebnik (1993). Extremal graphs without three-cycles or four-cycles, *Journal of Graph Theory*, **17**, 633–645.

[7] M. Hamada and J. A. Nelder (1997). Generalized linear models for quality-improvement experiments. *Journal of Quality Technology*, 29:292–304.

[8] W. K. Hastings (1970). Monte Carlo sampling methods using Markov chains and their applications. *Biometrika*, **57**, 97–109.

[9] P. McCullagh and J. A. Nelder (1989). *Generalized linear models*. 2nd ed. Chapman & Hall, London.

[10] B. Sturmfels and S. Sullivant (2008). Toric geometry of cuts and splits. *Michigan Math. J.*, **57**, 689–709.

セッション２

Session 2

Groebner基底の計算と応用

Computation and application of Groebner bases

Transformation of lexicographic Gröbner bases to smaller systems

Xavier Dahan
Kyushu university, Department of Informatics *

Abstract

We consider a polynomial system $F \subset \mathbf{k}[X_1, \ldots, X_n]$ over a field $\mathbf{k}$ of characteristic zero. We assume that this polynomial system F has a finite number of solutions, and that none of them is a multiple solution. Given a lexicographic Gröbner basis $\mathcal{G}(F)$ of the system F, the purpose of this work is to extend the formula of [2] that were constrained to special families of lexicographic Gröbner bases called "triangular sets" to any lexicographic Gröbner bases that satisfy the above assumptions. This extension is made possible thanks to the recent new interpolation formula made available in [1].

Introduction A mainstream method to solve polynomial systems is to use "elimination" which allows to find coordinate of the solutions one by one. Gröbner bases that are fully endowed with this "elimination property" are the "lexicographic ones". Unfortunately, these lexicographic Gröbner bases (abbreviated lex. G.b hereafter) are usually the most difficult to compute compared to other monomial orders, and are usually very large in term of memory space. When the input polynomial equations have only a finite number of solutions (i.e. when the ideal they generate in $\mathbf{k}[X_1, \ldots, X_n]$ has dimension zero) a common method to compute such a lex. G.b is to first compute a Gröbner basis for a monomial order which is known to be more efficient, e.g, degree reverse lexicographic order (drevlex) and then to change of monomial order with e.g. FGLM algorithm [4].

Previous work: triangular set If the field $\mathbf{k}$ is infinite, the bit-size of coefficients of lex. G.b becomes quickly huge – this is actually one reason of the difficulty to compute lex. G.b. In [2] was introduced transformation formula that take as input a special kind of lex. G.b called "triangular sets". They have the following shape for the monomial order $X_1 < \cdots < X_n$:

$$\begin{cases} T_n(X_1, X_2, \ldots, X_{n-1}, X_n) &= X_n^{d_n} + \cdots \\ T_{n-1}(X_1, \ldots, X_{n-1}) &= X_{n-1}^{d_{n-1}} + \cdots \\ \ddots & \vdots \\ T_1(X_1) &= X_1^{d_1} + \cdots \end{cases}$$

It is immediate to see that triangular sets are regular sequences. The transformation consists in computing the following system denoted $(N_1, \ldots, N_n)$:

$$\begin{cases} N_n(X_1, X_2, \ldots, X_{n-1}, X_n) &= (T_1' \cdot \frac{\partial T_2}{\partial X_2} \cdots \frac{\partial T_{n-1}}{\partial X_{n-1}}) T_n(X_1, \ldots, X_n) \bmod \langle T_1, \ldots, T_{n-1} \rangle \\ T_{n-1}(X_1, \ldots, X_{n-1}) &= (T_1' \cdot \frac{\partial T_2}{\partial X_2} \cdots \frac{\partial T_{n-2}}{\partial X_{n-2}}) T_{n-1}(X_1, \ldots, X_{n-1}) \bmod \langle T_1, \ldots, T_{n-2} \rangle \\ \ddots & \vdots \\ N_2(X_1, X_2) &= T_1'(X_1) T_2(X_1, X_2) \bmod T_1 \\ N_1(X_1) &= T_1 \end{cases}$$

Here the notation "$f \bmod \langle T_1, \ldots, T_s \rangle$" has no ambiguity since $T_1, \ldots, T_s$ is a Gröbner basis and thus simply means to take the normal form of f. Since we assume that the system was also radical, then

*dahan@inf.kyushu-u.ac.jp

$\gcd(T_1, T_1') = 1$ and thus T_1' is invertible in the quotient ring $\mathbf{k}[X_1]/\langle T_1\rangle$. In the same way, it was shown that $\frac{\partial T_s}{\partial X_s}$ is invertible in the quotient ring $\mathbf{k}[X_1, \ldots, X_s]/\langle T_1, \ldots, T_s\rangle$. Therefore, the transformation is reversible, gcd computations are easily feasible considering that we were able to compute a lex. G.b which require much more computational power. This transformation is implemented in the software Maple inside `RegularChains` library [3] (the function was named `DahanShostTransform`). The bit-size of th coefficients of the system N are in many cases spectacularly much smaller than the corresponding Gröbner basis (this has been quantified by providing estimates).

Generalization to Gröbner bases The purpose of this note is to introduce the same transformation as above for general lex. G.bs, yet of course still radical and zero-dimensional. For the non-radical case, the situation is more complicated and direct generalizations are not possible. As for the case of triangular sets [2], the transformation is based on interpolation formulas. Those have been extended recently in [1], supplying the possibility to extend the transformation as well.

Let us start with a lex. G.b in two variables $X_1 < X_2$. It is well-known (see *e.g.* [5]) in this case that a Gröbner basis $(g_1, g_2, \cdots, g_s)$ has the following structural:

$$\left\{\begin{array}{rcl} g_s(X_1,X_2) & = & X_2^{e_s} + \cdots \\ g_{s-1}(X_1,X_2) & = & p_{S-1}(X_1) q_{s-1}(X_1,X_2) \qquad \text{where} \qquad q_{s-1}(X_1,X_2) = X_2^{e_{s-1}} + \cdots \\ & \vdots & \\ g_3(X_1,X_2) & = & p_3(X_1)\cdots p_{s-1}(X_1) q_3(X_1,X_2) \qquad \text{where} \qquad q_3(X_1,X_2) = X_2^{e_3} + \cdots \\ g_2(X_1,X_2) & = & p_2(X_1)\cdots p_{s-1}(X_1) q_3(X_1,X_2) \qquad \text{where} \qquad q_2(X_1,X_2) = X_2^{e_2} + \cdots \\ g_1(X_1) & = & X_1^{d_1} + \cdots \end{array}\right. ,$$

where $e_2 < e_3 < \cdots < e_s$, p_i are univariate polynomial in X_1 for $i = 2 \ldots s-1$, and $\langle g_1\rangle$ is the first elimination ideal of the system. From [5], we know that g_1 is a *multiple* of the product of polynomials $p_2 \cdots p_{s-1}$.
The transformation is defined as follows: let $p_{\geq i}(X_1) := p_i(X_1)\cdots p_{s-1}(X_1)$ and $p_{<i} := g_1/p_{\geq i} \in \mathbf{k}[X_1]$.

$$\left\{\begin{array}{rcl} n_1(X_1) & = & g_1(X_1) \\ n_2(X_1,X_2) & = & p'_{<2}(X_1) g_2(X_1,X_2) \bmod g_1 \\ & \vdots & \\ n_{s-1}(X_1,X_2) & = & p'_{s-1}(X_1) g_{s-1}(X_1,X_2) \bmod g_1 \\ n_s(X_1,X_2) & = & g_s(X_1,X_2) \end{array}\right.$$

Already in two variables, experimental results are very explicit. The generalization to three variables is more complicated to sate in this note. Experimental results from a preliminary implementation realized in Maple show a gain that grows with the number of variables. It is likely that the growth is *at least* of quadratic order with respect natural data of the system. Considering the formula of [1] it is not surprising that the gain strongly depends on the number of polynomials in the lex. G.b: the most there are the most the gain is important.

References

[1] X. Dahan. On lexicographic Gröbner bases of radical ideals in dimension zero: structure and interpolation. `arXiv:1207.3887`.

[2] X. Dahan and É. Schost. Sharp estimates for triangular sets. In *ISSAC '04: Proceedings of the 2004 International Symposium on Symbolic and Algebraic Computation*, pages 103–110. ACM Press, 2004.

[3] M. Moreno Maza F. Lemaire and Y. Xie. The `RegularChains` library.

[4] J. C. Faugère, P. Gianni, D. Lazard, and T. Mora. Efficient computation of zero-dimensional gröbner bases by change of ordering. *J. Symbolic Comput.*, 16(4):329–344, 1993.

[5] D. Lazard. Ideal bases and primary decomposition: case of two variables. *J. Symbolic Comput.*, 1(3):261–270, 1985.

代数的閉体における限量子消去アルゴリズムについて
(On QE Algorithms over algebraically closed field)

深作亮也
Ryoya Fukasaku *
井上秀太郎
Inoue Shutaro †
佐藤洋祐
Yosuke Sato ‡

東京理科大学
Tokyo University of Science

Abstract

Quantifier Elimination(QE) in the domain of an algebraically closed field is much simpler than that of a real closed field. We can construct a QE algorithm using only GCD computations of (parametric) unary polynomials. Though a more sophisticated QE algorithm using Gröbner bases computations is implemented in the computer algebra system Mathematica, it is basically based on GCD computations of (parametric) unary polynomials. We propose two algorithms, one is an improvement of the algorithm of Mathematica based on the result of [1], the another one is an algorithm based on computations of comprehensive Gröbner systems.

1 はじめに

複素数領域における限量子消去 (以下 QE と略記する) は実数領域における QE と比較すると、理論的には実装が容易である。1 変数多項式の GCD の計算を再帰的に繰り返すことでアルゴリズムを構成することができる。ただし、扱う 1 変数多項式は一般にパラメターを含んでいるので、擬似剰余演算が必要になり、計算を効率的に行うには様々な工夫が必要になる。数式処理システム Mathematica の組み込み関数 Reduce と Resolve で実装されている複素数領域における QE では、グレブナー基底の計算等を利用してパラメターを含んだ 1 変数多項式の GCD の計算をおこなうよう工夫されている [6]。

Comprehensive グレブナー基底系 (以下 CGS と略記する) を用いると、GCD 計算による再帰的アルゴリズムとは全く違った方法で複素数領域における QE が容易に実装できるが、この方法では新たな変数を導入する必要があり、これまで CGS の効率的アルゴリズムの実装がなかったこともあり、これまでこの方法による実装はなされていない。

*fukasaku@mi.kagu.tus.ac.jp
†sinoue@rs.kagu.tus.ac.jp
‡ysato@rs.kagu.tus.ac.jp

最近の一連の研究成果 [9, 7, 4, 5, 8] により、CGS 計算が実装され利用できるようになったことを踏まえ、われわれは CGS を用いる方法と、GCD 計算による再帰的アルゴリズムによる方法の改良版を数式処理システム Risa/Asir を用いて実装し、２つの方法について比較検証をおこなった。

複素数領域における限量子消去 (以下 QE と略記する) をおこなうには、以下の形の論理式から、限量子 $\exists X_1 \exists X_2 \ldots \exists X_n$ を消去した $Y_1, \ldots, Y_m$ のみの式が得られればよいので、以下ではこの形の論理式にたいする限量子消去アルゴリズムのみをあつかう。

$$\exists X_1 \exists X_2 \ldots \exists X_n (f_1(Y_1, \ldots, Y_m, X_1, \ldots, X_n) = 0 \wedge \cdots \wedge f_s(Y_1, \ldots, Y_m, X_1, \ldots, X_n) = 0 \wedge \\ g_1(Y_1, \ldots, Y_m, X_1, \ldots, X_n) \neq 0 \wedge \cdots \wedge g_t(Y_1, \ldots, Y_m, X_1, \ldots, X_n) \neq 0)$$

以下、２章で本論文で用いるバックグラウンドについて必要最低限の解説を与える。３章では GCD 計算による再帰的アルゴリズムによる方法について、われわれの改良版も含め述べる。４章では CGS に基づく方法について述べる。最後に、われわれの計算実験により得られた双方の問題点と今後の課題について報告する。

2　グレブナー基底の安定性と CGS

以下において K は任意の体、$\overline{K}$ をその代数閉包とする。$\bar{X}$ は n 個の変数 $X_1, X_2, \ldots, X_n$、$\bar{Y}$ はそれとは異なる m 個の変数 $Y_1, Y_2, \ldots, Y_m$ を表す。$m = 1, n = 1$ のときは単に X, Y と記すことにする。また、イデアル I の多様体を $\mathbb{V}(I)$ で表す。

まず、グレブナー基底の安定性に関する次の結果から述べる。

定理 1 ([1])
$K[\bar{Y}, X]$ の有限集合 F に対して G を X を辞書式に $\bar{Y}$ よりも大きくなるような項順序に関するイデアル $\langle F \rangle$ のグレブナー基底とするとき、任意の要素 $\bar{c} = c_1, c_2, \ldots, c_m \in \overline{K}^m$ に対して、$G(\bar{c}) = \{g(\bar{c}, X) : g \in G\}$ は $\overline{K}[X]$ において $F(\bar{c}) = \{f(\bar{c}, X)\}$ で生成されるイデアルのグレブナー基底になる。

次に CGS の定義を述べる。

定義 1
$\bar{X}$ の項順序 $>$ を一つ固定する。$K[\bar{Y}, \bar{X}]$ の有限部分集合 F に対して、以下をみたす順序対の有限集合 $\mathcal{G} = \{(G_1, P_1, Q_1), \ldots, (G_s, P_s, Q_s)\}$ をパラメーター $\bar{Y}$、主変数 $\bar{X}$ の $>$ に関する F の CGS とよぶ。ここで、各 G_i は $K[\bar{Y}, \bar{X}]$ の有限部分集合、各 P_i, Q_i は $K[\bar{Y}]$ の有限部分集合である。
(i)　$\cup_{i=1}^{s} \mathbb{V}(\langle P_i \rangle) - \mathbb{V}(\langle Q_i \rangle) = \overline{K}^m$、$(\mathbb{V}(\langle P_i \rangle) - \mathbb{V}(\langle Q_i \rangle)) \cap (\mathbb{V}(\langle P_j \rangle) - \mathbb{V}(\langle Q_j \rangle)) = \emptyset$ for $i \neq j$。
(ii) 任意の $\bar{c} \in \mathbb{V}(\langle P_i \rangle) - \mathbb{V}(\langle Q_i \rangle)$ にたいして、$G_i(\bar{c}, \bar{X}) = \{g(\bar{c}, \bar{X}) : g \in G_i\}$ は $\overline{K}[\bar{X}]$ において、
$\langle f(\bar{c}, \bar{X}) \rangle$ のグレブナー基底である。
さらに、各 $G_i(\bar{c}, \bar{X})$ が reduced(minimal) グレブナー基底であるであるとき (monic であることは仮定しない)、reduced(minimal)CGS とよぶ。

3　QE の再帰的アルゴリズム

論理式 $\exists X_1 \exists X_2 \ldots \exists X_n (f_1(\bar{Y}, \bar{X}) = 0 \wedge \cdots \wedge f_s(\bar{Y}, \bar{X}) = 0 \wedge g_1(\bar{Y}, \bar{X}) \neq 0 \wedge \cdots \wedge g_t(\bar{Y}, \bar{X}) \neq 0)$ は $g_1(\bar{Y}, \bar{X}) \cdots g_t(\bar{Y}, \bar{X}) = g(\bar{Y}, \bar{X})$ として以下の同値な式に変形できる。

$\exists X_1 \exists X_2 \ldots \exists X_n (f_1(\bar{Y}, \bar{X}) = 0 \wedge \cdots \wedge f_s(\bar{Y}, \bar{X}) = 0 \wedge g(\bar{Y}, \bar{X}) \neq 0)$
$\exists X_n (f_1(\bar{Y}, \bar{X}) = 0 \wedge \cdots \wedge f_s(\bar{Y}, \bar{X}) = 0 \wedge g(\bar{Y}, \bar{X}) \neq 0)$ の式から $\exists X_n$ を消去し、その式の $\vee\wedge$ 標準型からさらに $\exists X_{n-1}$ を消去し、これを繰り返すことで、すべての限量子が消去できる。
したがって、$n = 1$ の場合について、アルゴリズムを構成すれば、これを再帰的に繰り返すことですべての限量子が消去できる。数式処理システム Mathematica の組み込み関数 Reduce と Resolve で実装されている K が有理数体のときの QE アルゴリズムでは、基本的にこの方法が用いられている。
以下では、Mathematica におけるアルゴリズムの概要を述べ、次にわれわれによる改良を与える。

Mathematica の QE の概要
Input: $\exists X(f_1(\bar{Y}, X) = 0 \wedge \cdots \wedge f_s(\bar{Y}, X) = 0 \wedge g(\bar{Y}, X) \neq 0)$
Output: $\bar{Y}$ のみの多項式による論理式
Step1. 項順序 $X \gg \bar{Y}$ による $\langle f_1, \ldots, f_s \rangle$ のグレブナー基底
$G = \{g_1(\bar{Y}, X), \ldots, g_l(\bar{Y}, X), h_1(\bar{Y}), \ldots, h_t(\bar{Y})\}$ を計算する。
Step2. $h_1(\bar{Y}) = 0 \wedge \cdots \wedge h_t(\bar{Y}) = 0$ でなければ与式は偽になることに注意する。以下 $h_1(\bar{c}) = 0 \wedge \cdots \wedge h_t(\bar{c}) = 0$ をみたす $\bar{c} \in \mathbb{C}^m$ について考える。与式が真になるのは、$g(\bar{c}, X)$ が $\langle f_1(\bar{c}, X), \ldots, f_s(\bar{c}, X) \rangle$ の根基イデアルに属さないことと同値であることに注意する。各 g_i を X の多項式とみなして、その中の最小次数のもの g_i を一つ選び、その次数を d、その最大項の係数を $p(\bar{Y})$ とおく。$p(\bar{c}) \neq 0$ ならば、$\{g_i(\bar{c}, X)\}$ が $\langle f_1(\bar{c}, X), \ldots, f_s(\bar{c}, X) \rangle$ のグレブナー基底、すなわち $g_i(\bar{c}, X)$ が $f_1(\bar{c}, X), \ldots, f_s(\bar{c}, X)$ の GCD になることに注意すると、$g(\bar{c}, X)$ が $\langle f_1(\bar{c}, X), \ldots, f_s(\bar{c}, X) \rangle$ の根基イデアルに属さないことと $g(\bar{c}, X)^d$ を $g_i(\bar{c}, X)$ で割った余りが 0 でないことが同値になる。$g(\bar{Y}, X)^d$ の $g_i(\bar{Y}, X)$ による疑剰余の係数を $p_1(\bar{Y}), \ldots, p_r(\bar{Y})$ とすると、$p(\bar{c}) \neq 0$ ならば、与式は $p_1(\bar{c}) \neq 0 \vee \cdots \vee p_r(\bar{c}) \neq 0$ と同値になる。
$p(\bar{c}) = 0$ の場合は、$\exists X(f_1(\bar{Y}, X) = 0 \wedge \cdots \wedge f_s(\bar{Y}, X) = 0 \wedge p(\bar{Y}) = 0 \wedge g(\bar{Y}, X) \neq 0)$ を新たな入力として再帰的に計算をおこうなう。その出力を $\phi(\bar{Y})$ とすると、結局全体の出力は
$\phi(\bar{Y}) \vee (h_1(\bar{Y}) = 0 \wedge \cdots \wedge h_t(\bar{Y}) = 0 \wedge p(\bar{Y}) \neq 0) \wedge (p_1(\bar{c}) \neq 0 \vee \cdots \vee p_r(\bar{c}) \neq 0))$ となる。

われわれの改良
$p(\bar{c}) \neq 0$ ならば、$g_i(\bar{c}, X)$ が $f_1(\bar{c}, X), \ldots, f_s(\bar{c}, X)$ の GCD になることはグレブナー基底の安定性に関する [3, 2] 等の結果を使うと容易に示すことができるが、前章で述べた定理 1 を使うと、先頭項が 0 であるなしにかかわらず $\{g_1(\bar{c}, X), \ldots, g_l(\bar{c}, X)\}$ がグレブナー基底であるので、これらの中に 1 つでも 0 でないものがある場合は GCD が定まり、再帰計算をおこなう必要がない。$g_1(\bar{Y}, X), \ldots, g_l(\bar{Y}, X)$ の係数として現れる $K[\bar{Y}]$ の要素をすべてを並べて $r_1(\bar{Y}), \ldots, r_k(\bar{Y})$ とおくと、再帰計算が必要になるのは $\langle h_1(\bar{Y}), \ldots, h_t(\bar{Y}), r_1(\bar{Y}), \ldots, r_k(\bar{Y}) \rangle \neq \langle 1 \rangle$ の場合のみであるため、ほとんどの場合は再帰計算をおこなわずにすむ。

4 QE の CGS によるアルゴリズム

前章で述べたアルゴリズムでは、パラメターを含む 1 変数多項式の GCD の効率的な計算のためにグレブナー基底計算をおこなってはいるが、グレブナー基底計算の計算が本質的に必要なわけではない。これにたいし、CGS の計算を用いると限量子 $\exists X_1 \exists X_2 \ldots \exists X_n$ を一挙に消去できる。

CGS による QE アルゴリズム
Input: $\exists \bar{X}(f_1(\bar{Y}, \bar{X}) = 0 \wedge \cdots \wedge f_s(\bar{Y}, \bar{X}) = 0 \wedge g_1(\bar{Y}, \bar{X}) \neq 0 \wedge \cdots \wedge g_t(\bar{Y}, \bar{X}) \neq 0)$
Output: $\bar{Y}$ のみの多項式による論理式

新たな変数 $\bar{Z} = Z_1, \ldots, Z_t$ を用いて、
$\{f_1(\bar{Y}, \bar{X}), \ldots, f_s(\bar{Y}, \bar{X}), g_1(\bar{Y}, \bar{X})Z_1 - 1, \ldots, g_t(\bar{Y}, \bar{X})Z_t - 1\}$ のパラメーター $\bar{Y}$ 主変数 $\bar{X}, \bar{Z}$ の CGS $\mathcal{G} = \{(G_1, P_1, Q_1), \ldots, (G_r, P_r, Q_s)\}$ を計算する。
ここで、G_i が定数でない多項式、すなわち $\bar{X}$ のどれかの変数を含む多項式を少なくとも 1 つ持つものすべてを $G_1, \ldots, G_k$ とすると、出力する論理式は $\vee_{i=1}^{k} \phi_i$ となる。ここで各 ϕ_i は $P_i = \{p_1(\bar{Y}), \ldots, p_a(\bar{Y})\}, Q_i = \{q_1(\bar{Y}), \ldots, q_b(\bar{Y})\}$ にたいして、$\phi_i \equiv p_1(\bar{Y}) = 0 \wedge \cdots \wedge p_a(\bar{Y}) = 0 \wedge (q_1(\bar{Y}) \neq 0 \vee \cdots \vee q_b(\bar{Y}) \neq 0)$ で与えられる。

CGS の計算が利用できる場合は、この方法は前章の再起的方法と比べると実装は容易であるが、あらたな変数を導入するため、一般的に、この CGS の計算は再起的方法で用いるグレブナー基底の計算よりもかなり重たいグレブナー基底の計算を必要とする。

5 まとめ

本文では述べなかったが QE アルゴリズムの最重要なポイントの 1 つとして、得られた論理式の簡単化があげられる。例えば、$\exists x \exists y \exists z (x*y+a*x*z+y*z-1 = 0 \wedge x*y*z+x*z+x*y+a = 0 \wedge x*z+y*z-a*z-x-y-1 = 0)$ から限量子を除去するために、Mathematica の

```
Resolve[Exists[{x,y,z},x*y+a*x*z+y*z-1==0&&x*y*z+x*z+x*y+a==0&&x*z+y*z-a*z-x-y-1==0]]
```

を実行すると、以下のような出力がされる。

```
(2159 - 4829 a - 592 a^2 + 6293 a^3 - 3932 a^4 - 844 a^5 + 3494 a^6 -
     1783 a^7 - 308 a^8 + 632 a^9 - 260 a^10 + 35 a^11 !=
    0 && -1 + a + a^2 ==
    0) || (-55 + 608 a - 2250 a^2 + 1429 a^3 - 1233 a^4 + 1935 a^5 -
     178 a^6 - 575 a^7 + 216 a^8 + 78 a^9 - 80 a^10 + 15 a^11 !=
    0 && -1 + a + a^2 ==
    0) || (-1339 + 3981 a - 5010 a^2 - 150 a^3 + 4607 a^4 -
     3776 a^5 + 794 a^6 + 2220 a^7 - 2500 a^8 + 1268 a^9 - 350 a^10 +
     40 a^11 != 0 && -1 + a + a^2 ==
    0) || (1017 - 1640 a - 164 a^2 + 323 a^3 + 665 a^4 - 1557 a^5 +
     1060 a^6 + 779 a^7 - 1268 a^8 + 720 a^9 - 210 a^10 + 25 a^11 !=
    0 && -1 + a + a^2 ==
    0) || (-736 + 1666 a - 2634 a^2 + 1425 a^3 + 2505 a^4 -
     3144 a^5 + 444 a^6 + 1919 a^7 - 2060 a^8 + 990 a^9 - 250 a^10 +
     25 a^11 != 0 && -1 + a + a^2 == 0) || (2 + a - 4 a^2 + 2 a^3 !=
    0 && 1 - a + 5 a^2 - 6 a^3 + 2 a^4 == 0) || (a !=
    0 && -1 + a + a^2 != 0 &&
   1 - 5 a + 9 a^2 - 34 a^3 + 37 a^4 + 593 a^5 - 1814 a^6 - 558 a^7 +
     8218 a^8 - 8848 a^9 - 2449 a^10 + 7850 a^11 - 11542 a^12 +
     23516 a^13 - 5320 a^14 - 34315 a^15 + 23525 a^16 + 14094 a^17 -
     13659 a^18 - 1889 a^19 + 2819 a^20 - 62 a^21 - 192 a^22 +
     21 a^23 + a^24 != 0) || (a != 0 && -1 + a + a^2 != 0 &&
```

```
  2 - a - 23 a^2 - 309 a^3 + 2459 a^4 - 6333 a^5 + 5855 a^6 +
    2023 a^7 - 5443 a^8 - 3255 a^9 + 16135 a^10 - 27046 a^11 +
    11336 a^12 + 43457 a^13 - 59351 a^14 - 1657 a^15 + 37823 a^16 -
    10467 a^17 - 8711 a^18 + 3105 a^19 + 645 a^20 - 244 a^21 +
    4 a^22 != 0) || (a != 0 && -1 + a + a^2 != 0 &&
  3 - 24 a + 73 a^2 - 215 a^3 + 1220 a^4 - 4879 a^5 + 11776 a^6 -
    25023 a^7 + 64683 a^8 - 146092 a^9 + 225027 a^10 - 253592 a^11 +
    262404 a^12 - 218871 a^13 + 23516 a^14 + 175369 a^15 -
    142711 a^16 - 1717 a^17 + 40984 a^18 - 10560 a^19 - 2375 a^20 +
    1108 a^21 - 122 a^22 + 6 a^23 != 0) || (a != 0 && -1 + a + a^2 !=
    0 && -2 + 18 a - 68 a^2 + 238 a^3 - 1300 a^4 + 5866 a^5 -
    17520 a^6 + 38578 a^7 - 75524 a^8 + 143803 a^9 - 245133 a^10 +
    346252 a^11 - 396084 a^12 + 326258 a^13 - 94318 a^14 -
    161094 a^15 + 204788 a^16 - 61953 a^17 - 35949 a^18 +
    27510 a^19 - 3626 a^20 - 1245 a^21 + 657 a^22 - 200 a^23 +
    26 a^24 != 0) || (a != 0 && -1 + a + a^2 !=
   0 && -1 + 10 a - 25 a^2 - 23 a^3 + 98 a^4 + 303 a^5 - 1071 a^6 +
    258 a^7 + 1563 a^8 + 2283 a^9 - 15219 a^10 + 28567 a^11 -
    33638 a^12 + 46364 a^13 - 86553 a^14 + 103975 a^15 -
    33012 a^16 - 54793 a^17 + 52896 a^18 - 4243 a^19 - 11829 a^20 +
    4135 a^21 + 280 a^22 - 367 a^23 + 57 a^24 != 0) || (a !=
   0 && -1 + a + a^2 != 0 &&
  9 - 91 a + 435 a^2 - 1521 a^3 + 4807 a^4 - 13313 a^5 + 30577 a^6 -
    60336 a^7 + 105398 a^8 - 157341 a^9 + 191334 a^10 -
    175100 a^11 + 79427 a^12 + 65253 a^13 - 133039 a^14 +
    63412 a^15 + 23932 a^16 - 30149 a^17 + 3701 a^18 + 3572 a^19 -
    955 a^20 - 44 a^21 + 26 a^22 != 0) || (-1 + a + a^2 != 0 &&
  17 - 44 a + 36 a^2 - 12 a^3 + 4 a^4 != 0 &&
  a == 0) || (-1 + a + a^2 != 0 && 10 - 14 a + 3 a^2 + 2 a^3 != 0 &&
  a == 0) || (-1 + a + a^2 != 0 &&
  7 - 40 a + 48 a^2 - 20 a^3 + 2 a^4 != 0 &&
  a == 0) || (-1 + a + a^2 !=
   0 && -3 - 8 a + 18 a^2 - 12 a^3 + 8 a^4 != 0 && a == 0) ||
1 + a == 0
```

この出力は間違いではないが、実はこれを簡易化すると true になる。しかし Mathematica の Simplify や FullSimplify を使っても true は得られない。一方、$\{x*y+a*x*z+y*z-1, x*y*z+x*z+x*y+a, x*z+y*z-a*z-x-y-1\}$ の CGS を a をパラメーターとして計算すると、定数だけからなる G_i は存在しないので、これからただちに与式が true であることが得られる。われわれが使用した CGS の計算アルゴリズムは [8] のものを使っているため、他のアルゴリズムと比較すると CGS の個数が少なく抑えられている。このため、出力された論理式は一般的にある程度の簡易化がすでになされている。ただし、自由変数の個数が多い時は CGS の計算時間が増大するという欠点がある。計算時間をおさえつつ簡易化もある程度実現させるためには、再帰計算か CGS の計算のどちらか一方をおこなうのではなく、再帰計算の方法と CGS の計算の方法を融合させたアルゴリズムが有効であることが予想される。

参 考 文 献

[1] Fortuna,E., Gianni,P. and Trager,B. (2001). Degree reduction under specialization. J. Pure Appl. Algebra, 164, pp. 153-164, 2001.

[2] Gianni, P.(1987). Properties of Grb̈ner bases under specializations. Lecture Notes in Comput. Sci., 378. pp. 293297. 1987.

[3] Kalkbrener, M.(1987). Solving systems of algebraic equations by using Gröbner bases. Lecture Notes in Comput. Sci., 378. pp. 282-292. 1987.

[4] Kapur, D., Sun, Y., and Wang, D. (2010). A New Algorithm for Computing Comprehensive Gröbner Systems. In International Symposium on Symbolic and Algebraic Computation, pp. 29-36. ACM-Press, 2010.

[5] Kurata, Y. (2011). Improving Suzuki-Sato's CGS Algorithm by Using Stability of Gröbner Bases and Basic Manipulations for Efficient Implementation. Communications of JSSAC Vol 1. pp 39-66. 2011.

[6] Mathematica Tutorial 複素多項式系　QE(量限定子除去)

[7] Nabeshima, K. (2007). A Speed-Up of the Algorithm for Computing Comprehensive Gröbner Systems. International Symposium on Symbolic and Algebraic Computation, pp. 299-306. ACM-Press, 2007.

[8] Nabeshima, K. (2012). Stability Conditions of Monomial Bases and Comprehensive Gröbner systems. Lecture Notes in Computer Science, Vol.7442, pp.248–259, 2012.

[9] Suzuki,A. and Sato,Y. (2006). A Simple Algorithm to Compute Comprehensive Grbner Bases Using Gröbner Bases. International Symposium on Symbolic and Algebraic Computation, pp. 326-331. ACM-Press, 2006.

セッション３

Session 3

制御系設計

Design of control systems

An Effective Implementation of a Special Quantifier Elimination for a Sign Definite Condition by Logical Formula Simplification

岩根秀直
(株) 富士通研究所*
Hidenao Iwane
Fujitsu Laboratories Ltd

樋口博之
(株) 富士通研究所†
Hiroyuki Higuchi
Fujitsu Laboratories Ltd

穴井宏和
(株) 富士通研究所/九州大学‡
Hirokazu Anai
Fujitsu Laboratories Ltd/Kyushu University

Abstract

This paper presents an efficient quantifier elimination algorithm tailored for a sign definite condition (SDC). The SDC for a polynomial $f \in \mathbb{R}[x]$ with parametric coefficients is written as $\forall x(x \geq 0 \rightarrow f(x) > 0)$. To improve the algorithm, simplification of an output formula is needed. We show a necessary condition for the SDC and an approach to simplify formulae by using a logic minimization method. Experimental results show that our approach significantly simplify formulae.

1 はじめに

限量記号消去アルゴリズム (quantifier elimination (QE) algorithm) とは，与えられた形式的理論 (formal theory) について「限量記号付きの一階述語論理式」を入力とし「等価で限量記号無しの論理式」を出力するアルゴリズムのことである．QE は工学や産業上の問題などの多くの応用があり重要なアルゴリズムである．しかし，QE は計算量の下限が限量記号がついた変数の数に対して二重指数であることが示されており，本質的に規模の大きな問題を解くことができない．そのため，限量記号がついた変数が線形の場合 [5] など制限された入力の一階述語論理式に対する専用アルゴリズムの研究がすすめられている．

多項式 $f(x) \in \mathbb{R}[x]$ に対して，$\forall x\ (x \geq 0 \rightarrow f(x) > 0)$ を sign definite condition (SDC) と呼ぶ．制御系設計の様々な条件が SDC で記述することができる [2] ため，SDC に対する QE は実用上重要な問題のクラスである．SDC が「定数項が正の多項式が $x \geq 0$ において実根を持たないこと」と等価なことを利用して，Sturm-Habicht 列 [4] を用いた実根の数え上げにより高速に計算する手法が提案されている [7].

本稿では，SDC 専用の QE の出力となる論理式の簡単化の手法について述べる．出力の論理式の簡単化は，QE の計算と実行可能領域の描画などの後処理の高速化につながる．論理式の簡単化のために，最初

*iwane@jp.fujitsu.com
†h-higuchi@jp.fujitsu.com
‡anai@jp.fujitsu.com

に，SDC を満たす多項式の Sturm-Habicht 列が満たす条件を示し，Sturm-Habicht 列による実根の数え上げによって得られる符号列の数を削減することで，論理式を簡単化する．次に，既存の論理関数処理を適用して $g > 0 \wedge g = 0 \leftrightarrow g \geq 0$ の規則による論理式の簡単化を行う．計算機実験結果により，本稿で提案する手法の効果を示す．

本稿の構成は以下の通りである．まず，2 章において Sturm-Habicht 列と SDC 専用 QE アルゴリズムについて紹介する．3 章では，SDC を満たす多項式の Sturm-Habicht 列においてあらわれない符号列を示し，4 章では，論理関数処理の紹介とそれを用いた論理式の簡単化手法を述べる．5 章で，実験結果により本稿で述べる手法による論理式簡単化の効果を示す．最後に 6 章で本稿のまとめと今後の課題を述べる．

2 Sign Definite Condition 専用 Quantifier Elimination

本章では，sign definite condition の定義と，[7] で提案されている専用の QE について述べる．

2.1 Sign Definite Condition と Sturm-Habicht 列による実根の数え上げ

最初に sign definite condition (SDC) を定義する．制御系設計の様々な条件が SDC で記述できるため，SDC は重要な問題のクラスであり，専用の QE アルゴリズムが提案されている [7, pp. 208–211].

定義 1
多項式 $f(x) \in \mathbb{R}[x]$ に対する以下の条件を **sign definite condition** *(SDC)* という．

$$\forall x \ (x \geq 0 \rightarrow f(x) > 0)$$

定義 2
$f(x)$ を n 次の実係数多項式とする．このとき，$\mathsf{SH}_n(f) = f, \mathsf{SH}_{n-1}(f) = \frac{df}{dx}$ とし，整数 j $(0 \leq j \leq n-2)$ について $\mathsf{SH}_j(f) = \delta_{n-j}\ \mathrm{Sres}_j(f, \frac{df}{dx})$ として構成する多項式の列 $\mathsf{SH}(f) := \{\mathsf{SH}_n(f), \ldots, \mathsf{SH}_0(f)\}$ を **Sturm-Habicht** 列という．ここで，$\mathrm{Sres}_j(f, g)$ は f と g の j 次部分終結式 *[7, p. 129]* で，$\delta_j = (-1)^{j(j-1)/2}$ である．また，$\deg(\mathsf{SH}_k(f)) = k$ のとき，$\mathsf{SH}_k(f)$ は正則であるという．

定義 3
符号とは，正，負または 0 のことで，それぞれ，$+1$, -1 または 0 で表す．実数の有限列 $A = \{a_m, \ldots, a_0\}$ における符号変化の数 $V(A)$ は，以下の規則に従い数える．

- 次の符号列を 1 と数える：$\{-1, +1\}, \{+1, -1\}, \{-1, 0, +1\}, \{+1, 0, -1\}, \{-1, 0, 0, +1\}, \{+1, 0, 0, -1\}$
- 次の符号列を 2 と数える： $\{+1, 0, 0, +1\}, \{-1, 0, 0, -1\}$
- 上記以外の符号列は 0 と数える

さらに，実数係数の有限個の多項式列 $S(x) = \{S_n(x), S_{n-1}(x), \ldots, S_0(x)\}$ とするとき，$\{h_m(x), \ldots, h_0(x)\}$ を $S(x)$ から恒等的に 0 になる多項式を取り除いたものとし，実数 α に対して，$V_\alpha(S)$ を $V(\{h_m(\alpha), \ldots, h_0(\alpha)\})$ と定義する．

次の定理により Sturm-Habicht 列を用いて任意の区間における多項式の実根を数え上げることができる．

定理 4
$f(x) \in \mathbb{R}[x]$, $a, b \in \mathbb{R} \cup \{-\infty, +\infty\}$ $(a < b)$ とし，$f(a)f(b) \neq 0$ を満たすとする．このとき，$V_a(\mathsf{SH}(f)) - V_b(\mathsf{SH}(f))$ は区間 $[a, b]$ における $f(x)$ の実根の数に一致する．

本稿の残りでは次の記法を用いる．

記法 **1**
$\mathsf{SH}_k(f)$ の $x=\infty$ における符号を s_k，$\mathsf{SH}_k(f)$ の $x=0$ における符号を c_k と表記する．

注意 **1**
$\mathsf{SH}_k(f)=a_{k,k}x^k+a_{k,k-1}x^{k-1}+\cdots+a_{k,0}$ とするとき，以下が成立する．

$$\begin{aligned} s_k=0 &\quad\leftrightarrow\quad a_{k,i}=a_{k,k-1}=\cdots=a_{k,0}=0\\ s_k>0 &\quad\leftrightarrow\quad (a_{k,k}>0)\vee(a_{k,k}=0\wedge a_{k,k-1}>0)\vee\cdots\vee(a_{k,k}=a_{k,k-1}=\cdots=a_{k,1}=0\wedge a_{k,0}>0)\end{aligned}$$

したがって，$s_k=0$ のとき，$\mathsf{SH}_k(f)$ は恒等的に 0 であり，c_k は $a_{k,0}$ の符号に一致する．また，n 次の多項式に対して，$s_n=s_{n-1}$, $s_0=c_0$ となることに注意する．

Sturm-Habicht 列に対して以下の Sturm-Habicht Structure Theorem [4] が成立する．

定理 **5**
f を次数 n (≥ 2) の多項式とする．$\mathsf{SH}_{k+1}(f)$ が正則となるすべての k に対して，$\deg(\mathsf{SH}_k(f))=r\leq k$ とするとき，以下が成立する．

(A) $r<k-1$ のとき，$\mathsf{SH}_{k-1}(f)=\cdots=\mathsf{SH}_{r+1}(f)=0$,

(B) $r<k$ のとき，$\mathrm{lc}(\mathsf{SH}_{k+1}(f))^{k-r}\mathsf{SH}_r(f)=\delta_{k-r}\mathrm{lc}(\mathsf{SH}_k(f))^{k-r}\mathsf{SH}_k(f)$,

(C) $r<k$ のとき，$\mathrm{lc}(\mathsf{SH}_{k+1}(f))^{k-r+2}\mathsf{SH}_{r-1}(f)=\delta_{k-r+2}\mathrm{Prem}(\mathsf{SH}_{k+1}(f),\mathsf{SH}_k(f))$.

ここで，$\mathrm{lc}(g)$ は g の主係数で，$\mathrm{Prem}(g,h)$ は，g と h の擬剰余であることを表す．

2.2 SDC 専用 QE の実装

本節では，穴井らにより提案された SDC 専用の QE アルゴリズムの実装方法 [7] について述べる．提案手法は，次数一定の多項式 $f(x)$ に対する SDC が，$f(x)$ が $x\geq 0$ で実根を持たないことと等価であることを利用する．

最初に，オフラインで次数毎に係数をパラメータとする多項式に対して，$x\geq 0$ で実根を持たない符号条件 φ_n をあらかじめ求め，その情報をデータベースなどに蓄積しておく．f が入力された後（オンライン）の計算は Sturm-Habicht 列を求めて代入する部分だけとなるので，高速な QE 計算が実現される．実際，5 次の問題は汎用の QE アルゴリズムである Cylindrical Algebraic Decomposition では 1 時間たっても計算が停止しないが，提案手法では 1 秒に満たない時間で計算できる．

オンラインの計算を高速化するためには，オフラインでの φ_n の表現を簡単化することが必要である．また，φ_n の表現の簡単化は出力される論理式の簡単化につながり，実行可能領域の描画や真偽値を判定する場合など後処理の高速化にもつながる．次章以降では，オフラインで計算する φ_n の簡単化について考える．

例 **1**
2 次の多項式 $f(x)=x^2+bx+c$ の場合を考える．$V_0(\mathsf{SH}(f))-V_\infty(\mathsf{SH}(f))=0$ となる符号条件は表 *1* のようになる．ここで，注意 *1* より，$s_2=s_1>0$，$s_0=c_0$ であり，$f(0)>0$ より，$c>0$ なので $c_2>0$ となることに注意する．

表 1: φ_2

s_2	s_1	s_0	c_2	c_1	c_0	s_2	s_1	s_0	c_2	c_1	c_0
$+1$	$+1$	-1	$+1$	-1	-1	$+1$	$+1$	-1	$+1$	0	-1
$+1$	$+1$	-1	$+1$	$+1$	-1	$+1$	$+1$	0	$+1$	0	0
$+1$	$+1$	0	$+1$	$+1$	0	$+1$	$+1$	$+1$	$+1$	$+1$	$+1$

$f(x)$ の *Sturm-Habicht* 列は $\mathsf{SH}(f) = \{x^2 + bx + c, 2x + b, b^2 - 4c\}$ なので，以下が φ_2 から得られる．

$$\begin{array}{lll} (b^2-4c<0 \wedge c>0 \wedge b<0) & \vee & (b^2-4c<0 \wedge c>0 \wedge b=0) \quad \vee \\ (b^2-4c<0 \wedge c>0 \wedge b>0) & \vee & (b^2-4c=0 \wedge c>0 \wedge b=0) \quad \vee \\ (b^2-4c=0 \wedge c>0 \wedge b>0) & \vee & (b^2-4c>0 \wedge c>0 \wedge b>0) \end{array} \tag{1}$$

式 (1) は $g > 0 \wedge g = 0 \leftrightarrow g \geq 0$ の規則を利用して，以下のように簡単化できる．

$$\begin{array}{lll} (b^2-4c<0 \wedge c>0) & \vee & (b^2-4c=0 \wedge c>0 \wedge b\geq 0) \quad \vee \\ (b^2-4c>0 \wedge c>0 \wedge b>0) & & \end{array}$$

$b^2 - 4c = 0 \wedge c > 0 \wedge b = 0$ を満たす実数 b, c が存在しないことを利用するとさらに簡単化できる．

$$(b^2-4c<0 \wedge c>0) \quad \vee \quad (b^2-4c\geq 0 \wedge c>0 \wedge b>0)$$

上記の条件をオフラインで構築しておき，具体的な問題に対して b, c に値を代入するだけで *QE* が実現される．このように事前の簡単化によりオンラインでの代入回数を削減し，専用 *QE* の高速化が実現できる．

3 SDC の必要条件

例 1 で見たように，Sturm-Habicht 列から得られる条件には，それを満たす実数が存在しないことがあり，それらを削減により論理式を簡単化できる．本章では SDC を満たす多項式 の Sturm-Habicht 列が満たす必要条件を示す．ここでは，定理 4 および記法 1 の記法を使用する．

定理 6
f を n 次の実係数多項式，u を $s_k \neq 0$ となる最小の非負整数 k とする．f が *SDC* を満足するとき，以下に示す条件を満たす．

$$\begin{array}{l} V_0(\mathsf{SH}(f)) - V_\infty(\mathsf{SH}(f)) = 0, \quad s_n > 0,\ s_{n-1} > 0,\ c_n > 0,\ s_0 = c_0, \\ c_u \neq 0, \quad c_{n-1} = 0 \rightarrow c_{n-2} < 0, \quad s_{n-2} = 0 \rightarrow s_{n-3} = \cdots = s_0 = 0, \\ s_k = 0 \rightarrow c_k = 0,\ (\forall k \in \{0, \ldots, n-2\}), \\ c_{k+2} \neq 0 \wedge c_{k+1} = 0 \rightarrow c_k \neq c_{k+2},\ (\forall k \in \mathcal{N} = \{u, \ldots, n-2\}), \\ c_k = c_{k+1} = 0 \wedge c_{k-1}c_{k+2}s_k s_{k+1} \neq 0 \rightarrow s_k s_{k+2} < 0,\ (\forall k \in \mathcal{N}), \\ c_k = \cdots = c_{k+m} = 0 \rightarrow s_{k+1} = \cdots = s_{k+m-1} = 0\ (\forall k \in \mathcal{N}, m > 1), \\ s_{k+2} = 0 \wedge s_{k+1} \neq 0 \rightarrow s_k \neq 0,\ (\forall k \in \mathcal{N}), \\ s_{k-1} \neq 0 \wedge s_k = \cdots = s_{k+m} = 0 \wedge s_{k+m+1} \neq 0 \rightarrow s^m_{k+m+2}s_{k-1} = \delta_{m+2}s^{m+1}_{k+m+1} \\ \qquad \wedge s^m_{k+m+2}c_{k-1} = \delta_{m+2}s^m_{k+m+1}c_{k+m+1},\ (\forall k \in \mathcal{N}, m \geq 0). \end{array}$$

定理の証明は以下の補題で与える．補題 8 と補題 12 の一部は [4] で示されている．

補題 7
次数 n の多項式 $f(x) \in \mathbb{R}[x]$ に対して，$s_n = s_{n-1} = c_n > 0$ は *SDC* の必要条件である．

証明 $f(x) = \sum_{i=0}^{n} p_i x^i$ $(p_n \neq 0)$ とする．今，$f(0) = p_0 > 0$ なので，$c_n > 0$．また，十分大きな x に対して $f(x) > 0$ を満たすには，$p_n > 0$ が必要なので，$s_n > 0$ となる．注意 1 より $s_{n-1} = s_n$ が成立する．∎
以下では，補題 7 より，入力の多項式 f の主係数および定数項は正，つまり，$s_n = c_n > 0$ と仮定する．また，以降では簡単のため，$\mathsf{SH}_k(f)$ を SH_k と表記する．

補題 8
u を $s_k \neq 0$ を満たす最小の非負整数とするとき，$c_u \neq 0$ が成立する．

証明 定義から SH_u は $\gcd(f, df/dx)$ の定数倍になる．$p_0 > 0$ から，$f(0) \neq 0$ なので，$\mathsf{SH}_u(0) \neq 0$. ∎

補題 9
$c_{n-1} = 0$ ならば $c_{n-2} < 0$ が成立する．

証明 $\mathsf{SH}_{n-1}(0) = p_1$, $\mathsf{SH}_{n-2}(0) = p_1 p_{n-1} - n^2 p_0 p_n^2$ で，今 $p_n > 0, p_0 > 0$ であり，$c_{n-1} = 0$ のとき，$p_1 =$ なので，$\mathsf{SH}_{n-2}(0) = -n^2 p_0 p_n^2 < 0$ ∎

補題 10
$s_{n-2} = 0$ ならば $s_{n-3} = \cdots = s_1 = s_0 = 0$ が成立する．

証明 $s_k \neq 0$ となる非負の整数 $k \leq n-3$ が存在すると仮定し，その最大値を k_m とする．このとき，定理 5 から SH_{n-1} の次数は k_m でなければならないが，$\mathsf{SH}_{n-1} = df/dx$ なので矛盾する．∎

補題 11
任意の $k = 0, \ldots, n$ に対して，$s_k = 0$ ならば $c_k = 0$ が成立する．

証明 注意 1 より，$s_k = 0$ のとき， $\mathsf{SH}_k = 0$ なので，$c_k = 0$. ∎

補題 12
任意の $k = 1, \ldots, n-1$ に対して，$c_{k+1} \neq 0 \wedge c_k = 0$ ならば $c_{k-1} \neq c_{k+1}$ が成立する．

証明 以下の 5 つの場合を考えればいい．

(1) $\deg(\mathsf{SH}_k) = r, \deg(\mathsf{SH}_{k+1}) = k+1, \mathsf{SH}_k(0) = c_k = 0$ のとき，

  (a) $r = k$ の場合，定理 5 (C) から，$\mathrm{lc}(\mathsf{SH}_{k+1})^2 \mathsf{SH}_{k-1} = -\mathrm{Prem}(\mathsf{SH}_{k+1}, \mathsf{SH}_k)$. よって，$c_{k-1} = -\mathrm{lc}(\mathsf{SH}_{k+1})^{-2} c_{k+1}$. $f(0) \neq 0$ なので，SH_k と SH_{k+1} が同時に x を共通因子に持つことはないので，$c_{k+1} \neq 0$ となる．したがって，$c_{k+1} \neq c_{k-1}$.

  (b) $r < k$ の場合，定理 5 (A)，(C) より，$c_{k-1} = 0$.

(2) $\deg(\mathsf{SH}_{k+1}) = r < k+1, \deg(\mathsf{SH}_{k+2}) = k+2, \mathsf{SH}_k(0) = c_k = 0$ のとき，

  (a) $r < k-1$ 場合，定理 5 (A) より $c_{k-1} = 0$.

  (b) $r = k-1$ 場合，定理 5 (B) より $\mathrm{lc}(\mathsf{SH}_{k+2})^2 \mathsf{SH}_{k-1} = -\mathrm{lc}(\mathsf{SH}_{k+1})^2 \mathsf{SH}_{k+1}$ なので，$\mathsf{SH}_{k-1} \mathsf{SH}_{k+1} \leq 00$ となる．

  (c) $r = k$ 場合，定理 5 (B) より SH_k は SH_{k+1} の定数倍となり，$c_{k+1} \neq 0 \wedge c_k = 0$ を満たさない．

∎

補題 13
任意の $k = u+1, \ldots, n=2$ に対して，$c_{k-1} \neq 0 \wedge c_k = c_{k+1} = 0 \wedge c_{k+2} \neq 0 \wedge s_k \neq 0 \wedge s_{k+1} \neq 0$ ならば $s_{k+2}s_k < 0$ が成立する．

証明　補題 11 より $c_{k+2} \neq 0$ なので $\mathsf{SH}_{k+2} \neq 0$．SH_{k+2} が正則でないと仮定すると，定理 5 (A) より SH_{k+1} は正則となる．さらに，定理 5 (B) より SH_{k+1} は SH_{k+2} の定数倍となり，$c_{k+2} \neq 0$ かつ $c_{k+1} = 0$ に矛盾する．したがって，SH_{k+2} は正則．
SH_{k+1} が正則とすると，定理 5 (C) より，$\mathrm{lc}(\mathsf{SH}_{k+2})^2\mathsf{SH}_k = \delta_2\mathrm{Prem}(\mathsf{SH}_{k+2}, \mathsf{SH}_{k+1})$ が成立．SH_k と SH_{k+1} が x を共通因子を持ち，$f(0) \neq 0$ に矛盾する．したがって，SH_{k+1} は正則ではない．また定理 5 (A) よりその次数は k でなければならない．定理 5 (B) より，$s_k = -s_{k+2}$ が得られる．∎

補題 14
任意の $k = u+1, \ldots, n-2$, $m = 2, \ldots, n-k-1$ に対して，$c_{k+m+1} \neq 0 \wedge c_{k+m} = \cdots = c_k = 0$ ならば $s_{k+m-1} = \cdots = s_{k+1} = 0$ が成立する．

証明　(1) $s_{k+m} \neq 0$ のとき，補題 13 の証明より，SH_{k+m} は正則ではなく，SH_{k+m+1} は正則となる．SH_{k+m} の次数 d_1 が k より大きいとすると，定理 5 (C) より，SH_{d_1-1} は $\mathrm{Prem}(\mathsf{SH}_{k+m+1}, \mathsf{SH}_{k+m})$ の定数倍になる．$c_{k+m} = c_{d_1-1} = 0$ より，x が共通因子となり，$f(0) \neq 0$ に矛盾する．したがって，$d_1 \leq k$．このとき，定理 5 (A) より，$s_{d_1+1} = \cdots = s_{k+m-1} = 0$.
(2) $s_{k+m} = 0$ のとき，SH_{k+m} の次数 d_2 が k 以上とすると，定理 5 (B) より，SH_{d_2} は SH_{k+m+1} の定数倍なので $c_{k+m+1} \neq 0$, $c_{d_2} = 0$ に矛盾する．したがって，$d_2 < k$．このとき，定理 5 (A) より，$s_{d_2+1} = \cdots = s_{k+m-1} = 0$.
∎

補題 15
任意の $k = u+1, \ldots, n-3$ に対して，$s_{k+2} = 0 \wedge s_{k+1} \neq 0$ ならば $s_k \neq 0$ が成立する．

証明　定理 5 (B) より，SH_{k+1} は正則．定理 5 (C) から，SH_k はある SH_r $(r > k+1)$ と SH_{k+1} の擬剰余により得られる．今，$k > u$ なので，$\mathsf{SH}_k \neq 0$ となる．∎

補題 16
$k = u+1, \ldots, n-3$ に対して，ある整数 $m \geq 0$ が存在し，$s_{k-1} \neq 0 \wedge s_k = \cdots = s_{k+m} = 0 \wedge s_{k+m+1} \neq 0$ ならば $s^m_{k+m+2}s_{k-1} = \delta_{m+2}s^{m+1}_{k+m+1} \wedge s^m_{k+m+2}c_{k-1} = \delta_{m+2}s^m_{k+m+1}c_{k+m+1}$ が成立する．

証明　定理 5 (B) より得られる．

∎

4　論理式の簡単化

本章では，論理関数処理を用いた論理式の簡単化について述べる．

4.1　論理代数と論理関数の簡単化

本節では論理代数と論理関数を定義する．

定義 17
論理代数は，論理値の集合 $B = \{0, 1\}$ に関する論理積 *(·)*，論理和 *(+)*，論理否定 *(′)* の *3* つの演算からなる代数系として定義される．ここで論理積，論理和，論理否定は図 *1* のように定義される．

x	y	$x \cdot y$
0	0	0
0	1	0
1	0	0
1	1	1

x	y	$x + y$
0	0	0
0	1	1
1	0	1
1	1	1

x	x'
0	1
1	0

図 1: 論理演算子（論理積・論理和・論理否定）

定義 18
論理変数およびその否定のことをリテラル *(literal)*，*1* 個のリテラル，または複数個の互いに異なる変数のリテラルの論理積を積項 *(product term)*，*1* 個の積項，または複数の異なる積項の論理和を積和形 *(sum-of-products form* または *disjunctive form)* と呼ぶ．

定義 19
定義 *17* における論理演算子と括弧，及び任意の個数の論理変数と論理定数 *(0* または *1)* を組み合わせて計算手順を表した式を論理式と呼ぶ．また，関数 $f : B^n \to B$ を論理関数 *(logic function)* と呼ぶ．

定義 20
n 変数論理関数の入力値の *0*，*1* の組み合わせは 2^n 通りある．通常の論理関数は，すべての入力の組み合わせに対する出力が定義されており，そのような論理関数を完全指定論理関数 *(completely specified logic function)* と呼ぶ．それに対して，一部の入力値に対しては，出力が未定義な論理関数を不完全指定論理関数 *(incompletely specified logic function)* と呼び，出力が未定義な入力値をドントケア *(don't care)* と呼ぶ．

例えば，$(x + y)'$ と $x' + y'$ は等価な論理式であるように 1 つの論理関数は複数の等価な論理式により表現できる．本稿では，与えられた論理式に対して，等価でより積項数の少ない論理式を得ることを論理式の簡単化と呼ぶ．不完全指定論理関数では，ドントケアを都合の良い値に解釈して，論理式をより簡単化できる場合がある．

論理関数を表す論理式を簡単化することは回路素子の数や配線の本数が少なくなる設計につながるため実用上重要である．そのため，二分決定グラフ（Binary Decision Diagram：BDD）を用いた厳密解法やヒューリスティクスを用いた近似解法 ESPRESSO [3] など多くの研究がなされている．

4.2 論理関数処理を用いた φ_n の簡単化

論理式簡単化手法を利用して，SDC を満足する論理式 φ_n を簡単化することを考える．

まず，論理変数は 2 つの値をとり，本稿で扱う多項式の符号は 3 つの値をとるので 2 つの論理変数を用いて s_k, c_k の符号を表現する．ここでは，x, y を用いて，$x'y'$ を 0，xy' を正，$x'y$ を負と表現し，φ_n を論理変数を用いた論理式として記述する．その結果に対して論理関数処理による簡単化手法の適用で，より簡単な φ_n の表現を得る．

より簡単な論理式を得るために，SDC 専用の QE においては次のようにドントケアを設定した．(1) 多項式の符号を x, y の 2 変数で表現することを上で述べた．符号は 3 値使用し，論理変数 2 変数で表すことができるのは 4 値であり，xy は使用していないため，ドントケアとして設定する．(2) 補題 8-16 を満たさない符号列は，補題 7 の仮定の元でそれを満たす実数が存在しない．したがって，真と偽，どちらの値

表 2: 計算実験結果

次数	変数	SyN	DC	ESP 近似	ESP 厳密	積項数	時間 近似	時間 厳密	SDC 近似 変数	厳密	時間
2	4	5	2	2	2	5	0.01	0.01	2	1	0.00
3	8	17	7	4	4	21	0.01	0.01	4	3	0.00
4	12	64	24	10	10	99	0.01	0.04	6	4	0.00
5	16	302	85	18	18	480	0.05	1.12	8	15	0.00
6	20	1229	299	57	57	2352	0.72	61.92	10	21	0.01
7	24	5238	1096	121	-	11656	21.40	>350h	12	84	0.15
8	28	20468	4037	353	-	58284	757.59	>350h	14	120	1.49

として扱っても良いためドントケアとして設定する．(3) $V_0(\mathsf{SH}(f)) < V_\infty(\mathsf{SH}(f))$ とはならないので，この場合もドントケアとして設定する．

5 実験結果

論理関数処理による論理式簡単化に対する近似手法 ESPRESSO [1] を用いて φ_n の簡単化を行った結果を表 2 に示す．計算は Intel(R) Core(TM) 2 Duo CPU 1.6 GHz, 2.0 Gbyte メモリ上で行った.「次数」は入力となる多項式の次数,「変数」は ESPRESSO に与える論理変数の数,「SyN」 は SyNRAC の以前の実装での積項の数,「DC」 は 4.2 節の (1) と補題 11 のみをドントケアで設定して ESPRESSO の近似解法を用いて簡単化した積項の数,「ESP 近似」 は ESPRESSO の近似解法を用いて簡単化した積項の数,「ESP 厳密」 は ESPRESSO の厳密解法を用いて簡単化した積項の数,「積項数」は定理 5 を満たす符号列の数,「時間近似」は 「ESP 近似」での実行時間（秒）を表す.「時間厳密」は 「ESP 厳密」での実行時間（秒）を表す.

次数 n の場合には Sturm-Habicht 列は $n+1$ 個からなるが，補題 7 より，$s_n = s_{n-1} = c_n > 0$ であり，$s_0 = c_0$ となるので，符号条件を求めるのに考慮すべき対象は $2(n+1)-4 = 2n-2$ 個である．したがって論理変数の数としては $4n-4$ となる．実験結果から ESPRESSO を用いた簡単化により以前の実装「SyN」に比べて論理和の数を大幅に削減できていることが確認できる.「DC」列と「ESP 近似」列の比較によりドントケアの設定で，より簡単化ができていることが確認できる．ESPRESSO の厳密解法では 7 次以上の結果が得られていないが，6 次以下の結果により近似解法で厳密解に近い結果が得られることが期待できる．

図 2 は，3 次の場合の ESPRESSO コマンドに対する入出力ファイルである．入力ファイルの 5 行目以降で入力とする論理式表現を定義し，出力ファイルの 6 行目以降で簡単化された論理式を表す．ハイフンは対応する論理変数が 0 でも 1 でも良いことを表し，行末の 1, 2 はそれぞれ，入力値が真とドントケアであることを表す．各変数は x_0y_0, x_1y_1, x_2y_2, x_3y_3 がそれぞれ，s_1, s_0, c_2, c_1 を表す．例えば 19 行目は，$s_1 < 0 \wedge s_0 = 0 \wedge c_2 > 0 \wedge c_1 > 0$ を表し，このとき，$V_0(\mathsf{SH}(f)) = 0 < 1 = V_\infty(\mathsf{SH}(f))$ なのでドントケアとして設定している．

19 行目，および 27 行目は $V_0(\mathsf{SH}(f)) < V_\infty(\mathsf{SH}(f))$ であること，30 行目から 33 行目は xy が未使用であることからドントケアを設定し，28 行目，および 36 行目から 37 行目は補題 12，34 行目から 35 行目は補題 11，38 行目から 39 行目は補題 9，40 行目から 41 行目は補題 8 を満たさないためドントケアとして設定している．

22 個の積項で表される論理式が，以下の等価で 4 個の積項からなる論理式に簡単化できた．

$$(s_1 < 0 \wedge s_0 > 0) \vee (s_1 < 0 \wedge c_1 < 0) \vee (s_0 < 0 \wedge c_1 < 0) \vee (c_2 \geq 0 \wedge c_1 \geq 0)$$

$f(x) = x^3 + ax^2 + bx + c$ とすると，Sturm-Habicht 列は $\mathsf{SH}_3 = f(x)$, $\mathsf{SH}_2 = f'(x) = 3x^2 + 2ax + b$, $\mathsf{SH}_1 = (2a^2 - 6b)x + ab - 9c$, $\mathsf{SH}_0 = -4b^3 + a^2b^2 - 4a^3c + 18bac - 27c^2$ なので，

$$\begin{aligned}\forall x(x \geq 0 \to f(x) > 0) \leftrightarrow^{QE} \quad & c > 0 \wedge \quad ((2a^2 - 6b < 0 \vee 2a^2 - 6b = 0 \wedge ab - 9c < 0) \wedge \mathsf{SH}_0 > 0 \vee \\ & (2a^2 - 6b < 0 \vee 2a^2 - 6b = 0 \wedge ab - 9c < 0) \wedge ab - 9c < 0 \vee \\ & \mathsf{SH}_0 < 0 \wedge ab - 9c < 0 \vee \\ & b \geq 0 \wedge ab - 9c \geq 0)\end{aligned}$$

となる．

SyNRAC の以前の実装では積項数が 17 個だったのに対し，提案手法では積項数が 4 個と簡単化が実現できているが，さらに簡単化する余地があることがわかっている．例えば，3 次の結果における最初の積項から得られる条件 $c > 0 \wedge ((2a^2 - 6b < 0 \vee 2a^2 - 6b = 0 \wedge ab - 9c < 0) \wedge \mathsf{SH}_0 > 0)$ を満たす実数は存在しないので，さらに簡単化できる．3 節で示した SDC を満たす多項式に対する Sturm-Habicht 列に対する条件は，s_k または c_k が 0 になる部分に着目しており，0 を含まない場合に不要な符号列を削減することは今後の課題である．

表 2 の最後の 3 列（SDC 近似）は Sturm-Habicht 列に現れる多項式 SH_k の k 次の係数と定数項が 0 の場合をドントケアとして扱った場合の結果である．つまり，すべての SH_k が正則で $c_k \neq 0$ と仮定した結果になる.「変数」は ESPRESSO に与える論理変数の数,「厳密」は ESPRESSO の厳密解法を用いて簡単化した積項の数,「時間」は ESPRESSO の厳密解法での実行時間（秒）を表す．描画結果のみを利用するなど正確な論理式を得る必要がない場合，かつ，すべての k に対して SH_k の k 次の係数と定数項が恒等的に 0 でないことがわかっている場合には，上記の制限を加えても問題がないことがある．この場合には，s_k および c_k が 0 となる場合を考慮する必要が無いので，論理変数としては次数 n に対して，$2n - 2$ となり，計算に必要な厳密解法でも時間は短く，より簡単な結果が得られた．別稿 [6] での実験結果では，この方法で生成される論理式を下に実行可能領域を描画した．

6 まとめ

実用上重要なクラスである SDC に対する専用の QE の高速化のために出力となる論理式の簡単化を行った．そのために，Sturm-Habicht 列が満たす必要条件を示し，真となる符号条件の数を削減し，簡単化には論理簡単化手法 ESPRESSO を利用した．このときドントケアの設定でより簡単な結果が得られることが確認できた．

今後の課題としては，SDC を満足する多項式が満たす必要十分条件を求めることと，7 次以上で得られているのは近似解であるため，BDD を用いた厳密解法による簡単化との比較することが考えられる．

謝　辞

SDC の出力を正則な場合に限定することで，出力の論理式をより簡単化する手法は，立教大学の横山和弘教授のコメントを元に行いました．貴重なコメントを頂いた横山教授に深く感謝いたします．

```
.lib x0 y0 x1 y1 x2 y2 x3 y3
.i 8
.o 1
.ob f0
01010101 1
01010001 1
01011001 1
01011000 1
01011010 1
00011000 1
10010101 1
10010001 1
10011001 1
10011000 1
10011010 1
01000101 1
01000001 1
01001001 1
01001010 2
00001000 1
10001010 1
01100101 1
01100001 1
01101001 1
01100100 1
01100110 1
01101010 2
00100100 2
10101010 1
11------ 2
--11---- 2
----11-- 2
------11 2
00----1- 2
00-----1 2
--101000 2
--010100 2
----0010 2
----0000 2
1000--00 2
0100--00 2
.e
```

```
.lib x0 y0 x1 y1 x2 y2 x3 y3
.i 8
.o 1
.ob f0
.p 4
-11----- 1
-1-----1 1
---1---1 1
-----0-0 1
.e
```

図 2: 3 次の問題に対する ESPRESSO コマンドの入力ファイル（左）と出力ファイル（右）

参 考 文 献

[1] Espresso. `http://embedded.eecs.berkeley.edu/pubs/downloads/espresso/`.

[2] H. Anai and S. Hara. A parameter space approach to fixed-order robust controller synthesis by quantifier elimination. *International Journal of Control*, 79(11):1321–1330, 2006.

[3] R. K. Brayton, A. L. Sangiovanni-Vincentelli, C. T. McMullen, and G. D. Hachtel. *Logic Minimization Algorithms for VLSI Synthesis*. Kluwer Academic Publishers, Norwell, MA, USA, 1984.

[4] L. González-Vega, T. Recio, H. Lombardi, and M.-F. Roy. *Sturm-Habicht sequences determinants and real roots of univariate polynomials*, pp. 300–316. Texts and Monographs in Symbolic Computation. Springer, 1998.

[5] R. Loos and V. Weispfenning. Applying linear quantifier elimination. *The Computer Journal*, 36(5):450–462, 1993.

[6] Y. Matsui, H. Iwane, and H. Anai. Two controller design procedures using SDP and QE for a power supply unit. 数式処理研究と産学連携の新たな発展, 2013.

[7] 穴井, 横山. QE の計算アルゴリズムとその応用 – 数式処理による最適化. 東京大学出版会, 8 2011.

Two controller design procedures using SDP and QE for a Power Supply Unit

Yoshinobu Matsui[*1,a)] Hidenao Iwane[*1,b)] Hirokazu Anai[*1,*2,c)]

Abstract: In this paper, we propose two controller design procedures using semi-definite programing (SDP) and quantifier elimination (QE), respectively. We consider to design controllers for a principal circuit in a power supply unit as an example. In general, a controller design problem is given as a problem finding a controller that satisfies given specifications in the open-loop transfer function's frequency characteristic. This is so-called an open-loop shaping problem in linear control theory. There exist some numerical methods for solving the problem using SDP. We propose an SDP-based controller design method via generalized Kalman-Yakubovich-Popov (GKYP) lemma. These SDP-based methods are effective for finding a feasible controller efficiently, but we cannot describe exact mathematical constraints for the required specifications by these methods.
In order to obtain exact controller's feasible regions for the required specifications, we describe the specifications as exact constraints formulated by sign definite conditions (SDCs) and solve them symbolically using QE.

Keywords: Open-loop shaping design problem, Linear matrix inequality, Semi-definite programing, Sign definite condition, Quantifier elimination

1. Introduction

The open-loop shaping design problem is a problem finding a controller, in a feedback control system, that satisfies given specifications in the open-loop transfer function's frequency characteristic. The open-loop shaping design problem for a single input and single output linear time-invariant system (SISO-LTI system) is a popular controller design problem in actual control system designs. Many control performances' characteristics are described by the open-loop transfer function's frequency characteristic. These are given as specifications.

Many methods for solving the problem have been proposed. The following methods are typical methods.

- The classical open-loop shaping design procedures in the classical linear control theory.
- The H^∞ mixed sensitivity design procedure [4] and the H^∞ loop shaping design procedure [13] in the H^∞ control theory.
- The design procedure using the generalized Kalman-Yakubovich-Popov (GKYP) lemma [11].
- The mixed sensitivity and Hurwitz stability design procedure using quantifier elimination (QE) [1], [3].

In modern linear control theory, the controller design procedures using semi-definite programing (SDP) have become the mainstream. The typical example is the procedure using the GKYP lemma. However, in the open-loop shaping design problem, we cannot describe exact mathematical constraints for the given specifications by the procedures using SDP, because SDP belongs to the convex programing problem. On the other hand, we may describe exact mathematical constraints by supposing to use QE and get the controller's exact feasible region.

In this paper, we propose two controller design procedures using SDP and QE for the open-loop shaping design problem. We apply these procedures to a controller design problem in a power supply unit, respectively and compare them. The controller design problem is the open-loop shaping design problem and many specifications are given in the open-loop transfer function's frequency characteristic.

We use the design procedure using the GKYP lemma as the SDP procedure. On the other hand, we formulate exact constrains for the required specifications by sign definite conditions (SDCs) and solve them exactly using QE. We note that we use a special QE algorithm for SDCs [1], [8].

We use the following notations. $\mathbb{R}$ denotes the field of real numbers. $\mathbb{C}$ denotes the field of complex numbers. $\mathbb{N}$ denotes the set of natural numbers. $\mathbb{R}^{n\times m}$ denotes the ring of $n\times m$ matrices, where $n, m \in \mathbb{N}$. j denotes an imaginary unit. $\mathcal{L}[\cdot]$ denotes a Laplace transform. For a square-integrable function $f(t)$,

$$\mathcal{L}[f(t)] := \int_0^\infty f(t)\exp(-st)dt,$$

where $s \in \mathbb{C}$, $t \in \mathbb{R}$. For a matrix M, its positive definiteness and transpose and complex conjugate transpose are denoted by $M > 0$, M^T and M^*, respectively. For a vector $v \in \mathbb{R}^n$,

*1 FUJITSU LABORATORIES LTD.
*2 Kyushu University.
a) m.yoshinobu@jp.fujitsu.com
b) iwane@jp.fujitsu.com
c) anai@jp.fujitsu.com

the real and imaginary parts are denoted by $\Re v$ and $\Im v$, the transpose is denoted by v^T. For matrices T and S, $S \otimes T$ denotes the Kronecker product.

We note that we use the following control theory's terms. For a real coefficient rational polynomial $h(s)$, the gain, the phase and the angular frequency are denoted as $|h(s)|$, $\angle h(s)$ and $\frac{d}{dt}\angle h(s)$, respectively. We call $x = 20\log_{10}|h(s)|$ the gain is x dB, $x = \frac{180}{\pi}\angle h(s)$ the phase is x degree and $x = \frac{1}{2\pi}\frac{d}{dt}\angle h(s)$ the angular frequency is x Hz, respectively.

2. Power supply unit

A principal circuit in a power supply unit is an AC/DC converter that converts alternating current (AC) input voltage to direct current (DC) output voltage. This mainly consists of the former power factor correction (PFC) circuit and the latter DC/DC converter circuit [5]. In this paper, we focus on the DC/DC back converter.

2.1 DC/DC back converter

The purpose of a DC/DC back converter is a conversion of the voltage level from the high DC input voltage V_{in} which is the output voltage of the former PFC circuit to a desired low DC output voltage V_{out}. This conversion must be done electrical efficiently in a power supply unit.

Fig. 1 shows a simplified equivalent circuit. This shows operating principles of a DC/DC back converter. S signifies a switch. The switching is done as follows:

$$S(t) = \begin{cases} 1, & kh \le t < (k + d[k])h, \\ 0, & (k + d[k])h \le t < (k+1)h, \end{cases}$$

where $t \in \mathbb{R}$ is continuous time (in seconds), $h \in \mathbb{R}$ is a constant period (in seconds), $k \in \mathbb{N}$ is discrete time, $d[k] \in \mathbb{R}$ is called a duty ratio. When S connects with 1 (we call

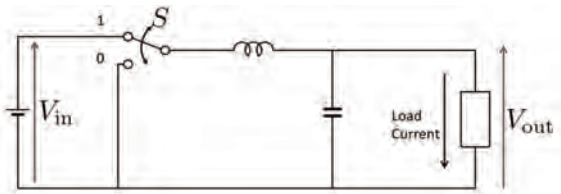


Fig. 1 Simplified equivalent circuit

this state ON), the output voltage level rises, because the load is connected with V_{in}. On the other hand, when S connects with 0 (OFF), the output voltage falls. The ON time changes at every period. This ON time ratio at each constant period is duty ratio. In order to make the level of V_{out} follow the desired level, we control the duty ratio. The level of V_{out} must follow the desired level robustly in some unpredictable situations. For example, PFC influences a DC/DC back converter or the load electrical changes and so on. Therefore, feedback control is used.

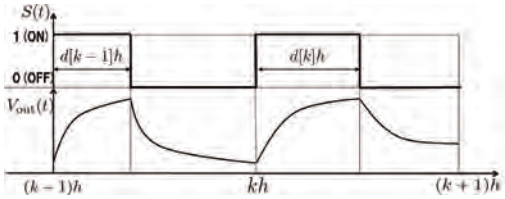


Fig. 2 Operating principle

2.2 Piecewise state space model

In this paper, we employ the normal equivalent circuit in Fig. 3 as an original model of a DC/DC back converter, where C is a condenser (unit is F), L is a coil (unit is H), I_L is an electric current in L, V_C is a voltage in C, and R, r_C, r_q, r_d, r_L are resistances (units are ohm). Here, we define the load electric current as $\frac{V_{\text{out}}}{R}$.

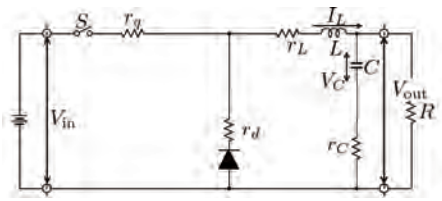


Fig. 3 Normal equivalent circuit

We get the following piecewise state space model from the normal equivalent circuit by Kirchhoff's law [12].

$$\frac{d}{dt}\xi(t) = \begin{cases} A_1\xi(t) + B_1 V_{\text{in}}, & S(t) = 1, \\ A_2\xi(t), & S(t) = 0, \end{cases}$$

$$V_{\text{out}}(t) = \begin{cases} C_V\xi(t), & S(t) = 1, \\ C_V\xi(t), & S(t) = 0, \end{cases}$$

where $\xi(t) := [I_L(t) \quad V_C(t)]^T$. $A_1 \in \mathbb{R}^{2\times 2}$, $A_2 \in \mathbb{R}^{2\times 2}$, $B_1 \in \mathbb{R}^{2\times 1}$, $C_V \in \mathbb{R}^{1\times 2}$ are defined as follows:

$$A_1 := \begin{bmatrix} -\frac{(r_q + r_L + \alpha r_C)}{L} & -\frac{\alpha}{L} \\ \frac{\alpha}{C} & -\frac{\alpha}{CR} \end{bmatrix},$$

$$A_2 := \begin{bmatrix} -\frac{(r_d + r_L + \alpha r_C)}{L} & -\frac{\alpha}{L} \\ \frac{\alpha}{C} & -\frac{\alpha}{CR} \end{bmatrix},$$

$$B_1 := \begin{bmatrix} \frac{1}{L} \\ 0 \end{bmatrix}, \quad C_V := \begin{bmatrix} \alpha r_C & \alpha \end{bmatrix},$$

where

$$\alpha := \frac{R}{R + r_C}.$$

Remark 1 Note that the electrical efficiency of a DC/DC back converter is deeply related to r_q and r_d. The study about the relationship between the electrical efficiency and the control performance would be one of our future works.

2.3 Averaged state space model

In order to design a controller by linear control theory, we need to employ a linear time-invariant system model. Here, we employ the averaged state space model as a linear time-invariant system model. When we design a controller of a DC/DC back converter, the averaged state space model is a popular model.

We get the following averaged state space model from the piecewise state space model by defining $\eta(t)$ as the average of $\xi(t)$ in a period [12].

$$\begin{cases} \dfrac{d}{dt}\Delta\eta(t) = A\Delta\eta(t) + B\Delta d(t), \\ \Delta V_{\rm out}(t) = C_V\Delta\eta(t), \end{cases} \tag{1}$$

where $\Delta\eta(t)$, $\Delta V_{\rm out}(t)$, $\Delta d(t)$ are small perturbation's signals of $\eta(t)$, $V_{\rm out}(t)$, $d[k]$, respectively. $A \in \mathbb{R}^{2\times 2}$, $B \in \mathbb{R}^{2\times 1}$ are defined as follows:

$$A := d_0A_1 + (1-d_0)A_2,$$

$$B := (A_1 - A_2)\eta_0 + B_1V_{\rm in},$$

where

$$d_0 := \frac{(r_d + r_L + R)V_0}{RV_0 - (r_q - r_d)V_0}, \tag{2}$$

$$\eta_0 := -A^{-1}B_1V_{\rm in}d_0, \tag{3}$$

where V_0 is the desired level of the output voltage. Let us suppose that the differential function of $\eta(t)$ is always 0 in a steady state, then we get (2) and (3).

We define Laplace transforms of $\Delta V_{\rm out}(t)$ and $\Delta d(t)$ as follows:

$$\Delta\hat{V}_{\rm out}(s) := \mathcal{L}[\Delta V_{\rm out}(t)],$$

$$\Delta\hat{d}(s) := \mathcal{L}[\Delta d(t)].$$

We get the following equation from (1).

$$\Delta\hat{V}_{\rm out}(s) = P(s)\Delta\hat{d}(s),$$

where

$$P(s) := C_V(sI - A)^{-1}B.$$

This $P(s)$ is a transfer function model for the averaged state space model.

Remark 2 Note that the averaged state space model is an accurate model in the only low frequency band for the piecewise state space model [14]. A DC/DC back converter its conversion is done electrical efficiently needs also high frequency band model to design the controller. An accurate model in the whole frequency band for the piecewise state space model by sampled-data control theory [15] would be one of our future works.

3. Controller design problem

In this section, we show a controller design problem for the normal DC/DC back converter. The original controller design problem is a problem finding a controller, in a feedback control system (shown in §3.1), satisfying the requirement that $V_{\rm out}$ follows V_0 under the following situations.

- **Situation 1** The PFC influences the DC/DC back converter.
- **Situation 2** The load electric current is time-independent.

The control performances for this requirement are described by the specifications in the open-loop transfer function's frequency characteristic as shown in §3.2.

3.1 Feedback control system

Fig. 4 shows a feedback control system for $P(s)$, where r is a reference signal, $K(s)$ is a controller to be designed. In this paper, we consider the following one order controller:

$$K(s) = \frac{b_{K_0}s + b_{K_1}}{s + a_{K_1}},$$

where $b_{K_0} \in \mathbb{R}$, $b_{K_1} \in \mathbb{R}$, $a_{K_1} \in \mathbb{R}$ are design parameters for the requirement. We define the open-loop transfer function

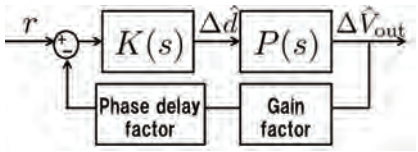


Fig. 4 Feedback control system

as follows:

$$L(s) := P(s)\times(\text{gain factor})\times(\text{phase delay factor})\times K(s).$$

Here, the open-loop transfer function's frequency characteristic is $L(j\omega)$, where ω is the angular frequency (unit is Hz). In this paper, we assume that the phase delay factor and the gain factor is given as $\exp(-1.4\times10^{-5}s)$ and 9.294×10^{-2}, respectively. See Fig. 5. We define $G(\cdot)$ as follows:

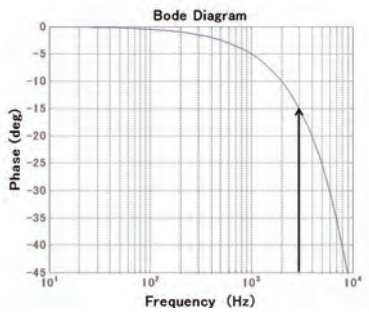


Fig. 5 Phase delay factor

$$G(\cdot) = P(\cdot)\times(\text{gain factor}).$$

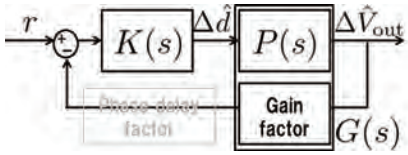


Fig. 6 Feedback control system for $G(s)$

3.2 Open-loop shaping design problem

For $L(j\omega)$, when ω increases from 0 to ∞, we call the trajectory in a complex plain a Nyquist diagram. The stability margin is defined by the separation condition between the trajectory and $-1+0j$ for the gain and the phase, respectively. These stability margins are called the gain margin and the phase margin, respectively (Fig. 7). The closed-loop system is internal stable when the following lemma holds.

Lemma 1 (Nyquist's satbility criterion [4]) The closed-loop is internal stable as long as the intersection point axis between the trajectory and the negative real axis > -1.

In order to satisfy the requirement mentioned above, $L(j\omega)$ must satisfy the following specifications.

- **Specification 0** The closed-loop system is internal stable.
- **Specification 1** The gain$>$ 45 dB when $0 \leq \omega \leq 1$.
- **Specification 2** The gain$>$ 25 dB when $1 \leq \omega \leq 100$.
- **Specification 3** The gain crossover frequency $>$3000.
- **Specification 4** The phase margin (PM) $>$45 degree.
- **Specification 5** The gain margin (GM) $>$7 dB.

Here, we call the problem finding a controller that satisfies these specifications "the open-loop shaping design problem". These specifications are described in a Nyquist diagram of $L(j\omega)$ (Fig. 7).

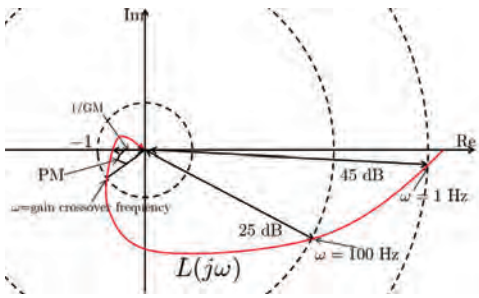


Fig. 7 Specifications described by a Nyquist diagram

These specifications are specified on the circuit designers experiences. We show what these specifications mean for the original controller design problem, but do not prove them mathematically in this paper.

First, specification 0 must be satisfied so as to make the control system stable. Second, specification 1 must be satisfied so as to make V_{out} follow r. Third, the influence by the PFC becomes smaller when specification 2 is satisfied. Finally, even if the load electric current changes, V_{out} follows r robustly, when specifications 3,4,5 are satisfied. There is a trade-off between specification 3 and specification 4. The larger the gain crossover frequency and the phase margin are, the better the control performance for the load electric current changing is.

4. SDP and QE

In modern linear control theory, the controller design procedures using SDP have become the mainstream [2].

SDP is a convex optimization. On the other hand, QE is a symbolic and algebraic algorithm to deal with first-order formulas over $\mathbb{R}$ and can solve non-convex optimization exactly [3], [9], [10].

For the exact optimal controller design, QE is better, but QE requires enormous computation time.

5. Mathematical formulation

In this section, we formulate mathematical constraints for the open-loop shaping deign problem's specifications in two formulations: a linear matrix inequality (LMI) and a sign definite condition (SDC). We propose two procedures to solve the LMI formulation problems by using SDP and the SDC formulation problems by using QE.

An LMI is a matrix inequality that can be come down to the following inequality.

$$F(z) > 0,$$

where $F(z) := F_0 + z_1F_1 + \cdots + z_nF_n$. Each $F_i \in \mathbb{R}^{n\times n}$ is a symmetric matrix, and $z := [z_1, \ldots, z_n]^T$ is a variable vector. A mathematical optimization problem whose constraints are formulated by LMIs and objective functions are linear in z, belongs to the class of SDP.

An SDC is defined for a real coefficient rational polynomial $f(x)$ as follows:

$$\forall x \geq 0\,(f(x) > 0).$$

This can be described by a first-order formula.

$$\forall x(x \geq 0 \rightarrow f(x) > 0).$$

We can solve an SDC efficiently by using QE which uses the Strum-Habicht sequence [1], [8].

5.1 LMI formulation

The open-loop shaping design problem's specifications can be formulated by LMI constraints using the generalized Kalman-Yakubovich-Popov (GKYP) lemma. Here, we introduce a special case of the GKYP lemma.

Lemma 2 The following inequality is called a frequency domain inequality (FDI) for $G(s)$.

$$\begin{bmatrix} G(s) & I \end{bmatrix} \Pi \begin{bmatrix} G(s) & I \end{bmatrix}^* < 0,\ \forall s \in \Lambda(\Phi_c, \Psi). \qquad (4)$$

Π and $\Lambda(\Phi_c, \Psi)$ are defined as follows:

$$\Pi(a_g, b_g, \gamma) := \begin{bmatrix} 0 & a_g - jb_g \\ a_g + jb_g & -2\gamma \end{bmatrix}.$$

$$\Lambda(\Phi_c, \Psi) := \{\lambda \in \mathbb{C} | \sigma(\lambda, \Phi_c) = 0,\ \sigma(\lambda, \Psi) \geq 0\},$$

where

$$\sigma(\lambda, \Phi_c) := \begin{bmatrix} \lambda^* & 1 \end{bmatrix} \Phi_c \begin{bmatrix} \lambda \\ 1 \end{bmatrix}, \Phi_c := \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix},$$

$$\Psi(\omega_L, \omega_H) := \begin{bmatrix} -1 & j(\omega_L + \omega_H)/2 \\ -j(\omega_L + \omega_H)/2 & -\omega_L \omega_H \end{bmatrix},$$

A necessary and sufficient condition for (4) is given as follows:

There exist a symmetric matrix $P \in \mathbb{R}^{3\times 3}$ and a positive definite matrix $Q \in \mathbb{R}^{3\times 3}$ such that $\mathcal{G}(\Psi, \Pi) < 0$, where

$$\mathcal{G}(\Psi, \Pi) := W(P, Q) + V,$$

$$W(P, Q) := \begin{bmatrix} A_G & I \\ C_G & 0 \end{bmatrix} (\Phi_c^T \otimes P + \Psi^T \otimes Q) \begin{bmatrix} A_G & I \\ C_G & 0 \end{bmatrix}^T,$$

$$V := \begin{bmatrix} 0 & B_G(a_g - jb_g) \\ B_G^T(a_g + jb_g) & 2a_g D_G - 2\gamma \end{bmatrix},$$

where the set $\{A_G, B_G, C_G, D_G\}$ is the state-space representation of $G(s)$.

Proof See [11].

We explain what Lemma 2 means. Equation (4) means the following convex region in a complex plain in which $G(j\omega)$ occurs, that is a feasible region for $G(j\omega)$.

$$a_g \Re G(j\omega) + b_g \Im G(j\omega) < \gamma,\ \omega_L \leq \omega \leq \omega_H.$$

Φ decides $s = j\omega$, Ψ decides the interval $\omega_L \leq \omega \leq \omega_H$, Π decides the convex region.

We define the following matrix inequalities.

$$\mathcal{G}_1 := \{\mathcal{G}(\Psi, \Pi) < 0 | \Psi(L_1, H_1),\ \Pi(-1, 0, -g_1)\},$$

$$\mathcal{G}_2 := \{\mathcal{G}(\Psi, \Pi) < 0 | \Psi(L_2, H_2),\ \Pi(0, 1, -g_2)\},$$

$$\mathcal{G}_3 := \{\mathcal{G}(\Psi, \Pi) < 0 | \Psi(L_3, H_3),\ \Pi(0, 1, -g_3)\} \text{ and}$$

$$\mathcal{G}_4 := \{\mathcal{G}(\Psi, \Pi) < 0 | \Psi(L_4, \infty),\ \Pi(-10, 1, \gamma)\},$$

where, $L_1 := 0$, $H_1 := 1\times 2\times \pi$, $L_2 := H_1$, $H_2 := 100\times 2\times \pi$, $L_3 := H_2$, $H_3 := 3000 \times 2 \times \pi$, $L_4 := H_3$, $g_1 := 10^{45/20}$, $g_2 := 10^{25/20}$, $g_3 := 1$.

Then the following Lemma 3 holds.

Lemma 3 When $\gamma < 5 - \sqrt{3}/2 \fallingdotseq 4.134$,

$$\text{Specification } 1 \leftarrow \mathcal{G}_1, \tag{5}$$

$$\text{Specification } 2 \leftarrow \mathcal{G}_2, \tag{6}$$

$$\text{Specification } 3 \leftarrow \mathcal{G}_3 \text{ and} \tag{7}$$

$$\text{Specifications } 4, 5 \leftarrow \mathcal{G}_4 \tag{8}$$

hold.

Proof (5),...,(7) are obvious by Fig. 7 and Fig. 8. For $\gamma < 5 - \sqrt{3}/2$ and (8), see Fig. 9. Note that we must assure the phase margin > 60 degree, because the phase delay factor is given as Fig. 5 (§3.1). Fig. 5 shows the phase delay is 15 degree in 3 kHz.

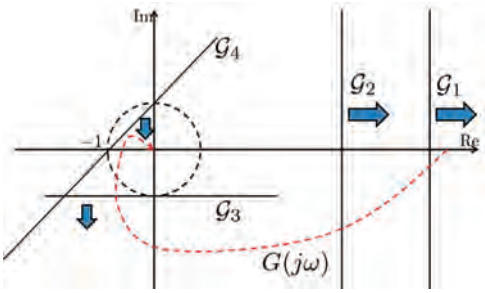


Fig. 8 Specifications formulated by $\mathcal{G}_1, \ldots, \mathcal{G}_4$

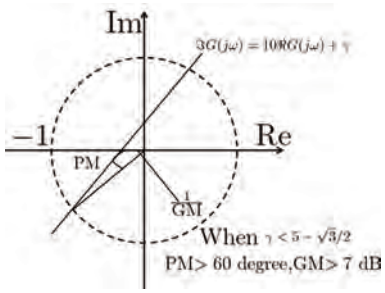


Fig. 9 Stability margin formulated by $\mathcal{G}_4$

Remark 3 Note that we can formulate other formulations to the specifications by LMIs using the GKYP lemma. For example we also define $\mathcal{G}_2$ as

$$\mathcal{G}_2 := \{\mathcal{G}(\Psi, \Pi) < 0 | \Psi(L_2, H_2),\ \Pi(1, 1, -\sqrt{2}g_2)\}.$$

However, we cannot express the outside region of a circle by LMIs, because LMIs are based on convex regions.

When we consider parameterizing the controller, a_{K_1}, b_{K_0}, b_{K_1}, P_i and Q_i, $i = 1, \ldots, 4$ are parameters. In this case $\mathcal{G}_i < 0$ are not LMIs but are bilinear matrix inequalities (BMIs), because there exists a cross-term between a_{K_1} and P_i. BMIs are not SDP. For converting $\mathcal{G}_i < 0$ to LMIs, first we must fix a_{K_1}. In this paper, we fix a_{K_1} as 35.34. This a_{K_1} is given by circuit designers experience. Second, we define the following B_G

$$B_G := \begin{bmatrix} w \\ Bb_{K_0} \end{bmatrix},$$

where w is a design parameter and $w := b_{K_1} - a_{K_1} b_{K_0}$.

Remark 4 Note that the controller's pole (= −35.34) is not always the best pole for the open-loop shaping design problem. We should consider the case that a_{K_1} is a free parameter and a full-order controller case, but we do not consider the cases, in this paper, it would be one of our future works.

Finally, we can convert $\mathcal{G}_i < 0$ to LMIs $\hat{\mathcal{G}}_i(P_i, Q_i, w, b_{K_0}) < 0$ in case parameterizing the controller and formulate the original open-loop shaping design problem as the following SDP problem, and we can solve it by an interior point method.

Problem (SDP) minimize γ
subject to

$$\begin{bmatrix} \hat{\mathcal{G}}_1(P_1, Q_1, w, b_{K_0}) & & \\ & \ddots & \\ & & \hat{\mathcal{G}}_4(P_4, Q_4, w, b_{K_0}) \end{bmatrix} < 0.$$

We can get w, b_{K_0}, P_i and Q_i by solving this SDP, and from this w, we can parametrize b_{K_1} as $w + a_{K_1} b_{K_0}$. These are optimal controller parameters for the open-loop shaping design problem.

5.2 SDC formulation

The open-loop shaping design problem's specifications can be formulated by SDC constraints. We formulate the specifications 2, . . . ,6 exactly by SDCs. We define the following SDCs.

$$\mathcal{S}_1 : \forall\omega(L_1 \le \omega \le H_1 \to |G(j\omega)|^2 - g_1^2 > 0), \quad (9)$$

$$\mathcal{S}_2 : \forall\omega(L_2 \le \omega \le H_2 \to |G(j\omega)|^2 - g_2^2 > 0), \quad (10)$$

$$\mathcal{S}_3 : \forall\omega(L_3 \le \omega \le H_3 \to |G(j\omega)|^2 - g_3^2 > 0) \text{ and} \quad (11)$$

$$\mathcal{S}_4 : \forall\omega(L_4 \le \omega \to 3.8\Re G(j\omega) + 1 - \Im G(j\omega) > 0). \quad (12)$$

These show the feasible regions for $G(j\omega)$ as Fig. 10. Note

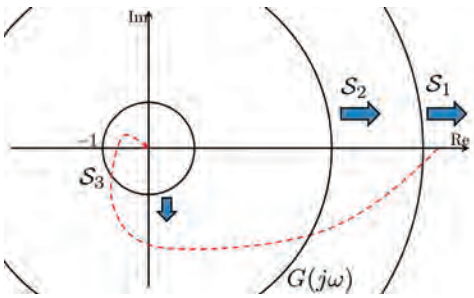


Fig. 10 Specifications formulated by $\mathcal{S}_1, \ldots, \mathcal{S}_3$

that 3.8 in $\mathcal{S}_4$ is given by Lemma 5. Obviously, the following lemma holds.

Lemma 4 The specifications 2, . . . ,6 are formulated by SDCs as follows:

$$\text{Specification } 1 \leftrightarrow \mathcal{S}_1, \quad (13)$$

$$\text{Specification } 2 \leftrightarrow \mathcal{S}_2, \quad (14)$$

$$\text{Specification } 3 \leftrightarrow \mathcal{S}_3 \text{ and} \quad (15)$$

$$\text{Specifications } 4,\ 5 \leftarrow \mathcal{S}_4. \quad (16)$$

Proof For specification 1, . . . ,3, (13), . . . ,(15) obviously hold by Fig. 7 and Fig. 10, respectively.

In order to show why (16) holds, we indicate the following lemma.

Lemma 5 When the closed-loop system of a feedback system in Fig. 6 (§3.1) is internal stable and $a\Re G(j\omega) + 1 - \Im G(j\omega) > 0$ holds, the phase margin and the gain margin satisfy the followings:

$$\text{PM} \ge 360 \arctan(a)/\pi - 90 \,\text{degree},$$

$$\text{GM} \ge 20 \log_{10}(a) \,\text{dB}. \quad (17)$$

Proof When the closed-loop system is internal stable, (17) is obviously holds. See Fig. 11.

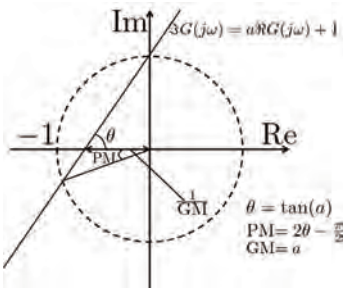


Fig. 11 Stability margin formulated by $\mathcal{S}_4$

From (17), when $a > 3.8$, the phase margin is over 60 degree and the gain margin is over 7 dB. See Fig. 12. Therefore, when $a > 3.8$, (16) holds.

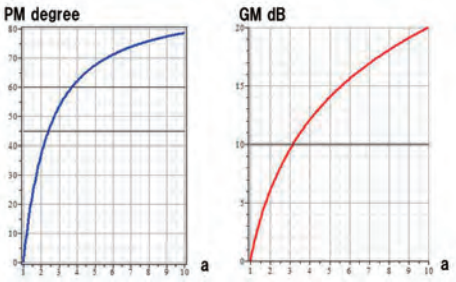


Fig. 12 a vs PM, GM

We solve the specification 0 algebraically by the Hurwitz stability condition. The following Lemma 6 holds [7].

Lemma 6 Let a characteristic polynomial for $G(s)$ be $g(s)$. The following two propositions are equivalent.

- The closed-loop system is internal stable.
- All coefficients of $g(s)$ are positive or negative and the all leading principal minor of a Hurwitz matrix for $g(s)$ are positive.

We can get feasible regions of the controller parameters for

each specification by using Lemma 6 algebraically and solve $\mathcal{S}_1, \ldots, \mathcal{S}_4$ using QE, and by superposing obtained feasible regions, we can get the feasible regions of the controller for the open-loop shaping design problem's specifications.

6. Numerical example

We show a numerical example with the following $P(s)$.

$$P(s) = \frac{4.622 \times 10^7 s + 2.140 \times 10^{12}}{1.128 \times 10^4 s^2 + 1.906 \times 10^8 s + 1.453 \times 10^{12}}.$$

The computational experiments for the following SDP solution (§6.1) was executed on a computer with an Intel (R) Core (TM) i5-2520M CPU 2.5 GHz and 4.0 GByte memory. We solved SDP by LMI control toolbox [6]. The computational experiments for the following QE solution (§6.2) was executed on a computer with an Intel (R) Core (TM) i7-3540M CPU 3.0 GHz and 2.0 GByte memory. We solved QE by our own solver SyNRAC [8].

6.1 SDP solution

We get the following optimal controller by solving the SDP problem.

$$K_{\mathrm{gkyp}} = \frac{1.944s + 7587}{s + 35.34}, \tag{18}$$

and the optimal γ,

$$\gamma_{\mathrm{opt}} = 3.303. \tag{19}$$

The computing time to obtain the K_{gkyp} is 1.443 seconds.

6.2 QE solution

We consider decreasing the degree of the polynomial in $\mathcal{S}_i$ for reducing the computing time.

$$\mathcal{S}_i : \forall\omega(L_i \le \omega \le H_i \to |G(j\omega)|^2 - g_i^2 > 0),$$

where $i = 1, 2, 3$. The degrees of the numerator polynomial and the denominator polynomial of $|G(j\omega)|^2 - g_i^2$ are 12 and 12, respectively. We can decrease the degrees to 6 and 6 by substituting ω^2 for Ω, because the numerator and the denominator of $|G(j\omega)|^2 - g_i^2$ are even polynomials. We can decrease the degree of $\mathcal{S}_i$ in Ω to 6, because the denominator polynomial is always positive.

Feasible regions for $\mathcal{S}_1$ and $\mathcal{S}_2$ are obtained by QE and shown as shaded regions in Fig. 13. We note the feasible region for $\mathcal{S}_2$ is non-convex. The computing time to obtain the feasible regions for $\mathcal{S}_1$ and $\mathcal{S}_2$ are 3.386 seconds and 4.852 seconds, respectively.

The feasible region for $\mathcal{S}_3$ is given as Fig. 14. We note the feasible region for $\mathcal{S}_3$ is non-convex. The computing time to obtain the feasible region for $\mathcal{S}_3$ is 1.794 seconds. In general, a gain-cross over frequency is deeply related to a dead-beat step response. Fig. 14 shows the feasible region for a dead-beat step response is non-convex.

Feasible regions for the Hurwitz stability condition and $\mathcal{S}_4$ are shown in Fig. 15. The superposition of these feasible regions shows a robust stability region that assures PM> 60 and GM> 7 for $G(j\omega)$. The computing time to obtain the feasible region for $\mathcal{S}_4$ is 2.948 seconds.

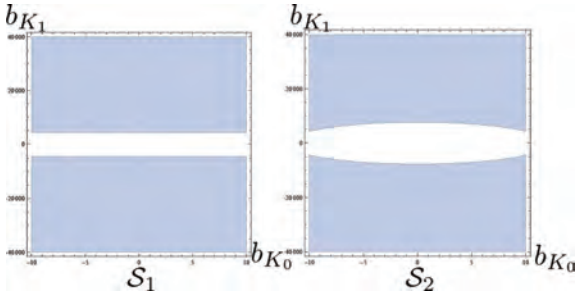


Fig. 13 Feasible regions for $\mathcal{S}_1$ and $\mathcal{S}_2$

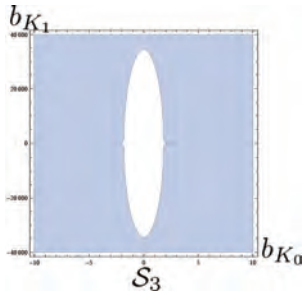


Fig. 14 Feasible region for $\mathcal{S}_3$

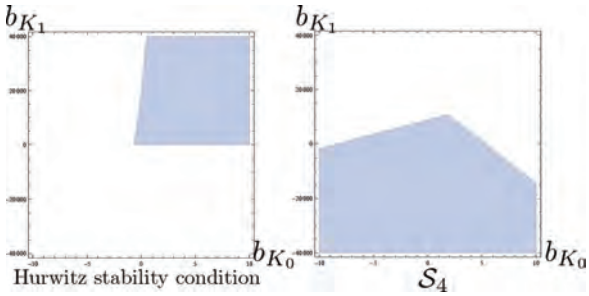


Fig. 15 Feasible regions for Hurwitz stability and $\mathcal{S}_4$

The superposition of all the feasible regions is given by Fig. 16. This is a feasible region for the open-loop shaping design problem. K_{gkyp} is in the superposition of all the feasible regions.

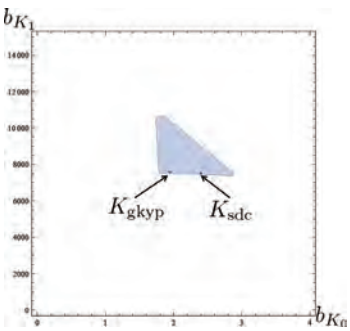


Fig. 16 Feasible region for the open-loop shaping design problem

6.3 Comparison

We select the desired controller from Fig. 16 as follows:

$$K_{\mathrm{sdc}} = \frac{2.4s + 7500}{s + 35.34}. \tag{20}$$

The Bode diagram of the open-loop shaped by K_{sdc} and K_{gkyp} are Fig. 17. This shows both controllers designed by

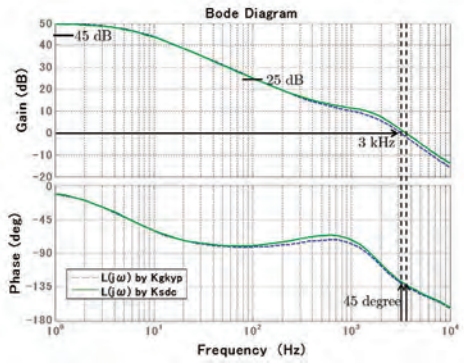


Fig. 17 Bode diagram

the two procedures satisfy the required specifications.

The time response controlled by K_{sdc} and K_{gkyp} are Fig. 18. When the load electric current changing occurs,

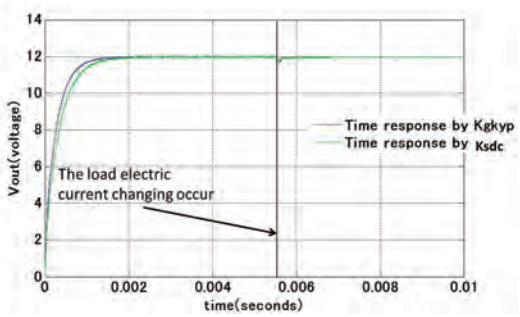


Fig. 18 Time response

V_{out} follows V_0 robustly. Here, the load electric current changes from 108 to 208, $V_0 = 12$, K_{gkyp} and K_{sdc} are discretized.

7. Conclusion

In this paper, we proposed two controller design procedures using SDP and QE, and compared them by applying to the controller design problem of a normal DC/DC back converter.

We designed controllers that satisfy the desired specifications by the both procedures. We designed an optimal controller by the procedure using SDP numerically. The mathematical constraints and the objective function that we formulated by LMIs were not exact for the desired specifications. We cannot formulate exact (i.e. relaxed expression) mathematical constraints and an objective function by LMIs in principle, because, in the open-loop shaping design problem, many of the desired specifications are non-convex. On the other hand, we can formulate exact mathematical constraints for the many desired specifications by SDCs straight forwardly, and we could get controller's exact feasible regions for the open-loop shaping design problem's specifications by the procedure using a specialized QE. We confirmed the controller's feasible region is non-convex. Therefore, we can design an exact optimal controller for the open-loop shaping problem by the procedure using QE. Hereby, circuit designers can select the best controller for the specifications which they set by their experience even if the specifications are non-convex.

In other words, this means we showed an open-loop shaping design problem for an LTI-system its order is 1/2 and the controller its order is 1/1 can be actually resolved by the procedure using QE as long as the controller's pole is fixed. However, we should consider the case that a_{K_1} is a free parameter and a full-order controller case, respectively as we remarked before. Especially, from the viewpoint of modern linear control theory we should consider the full-order controller case. In modern control theory, the existence of the controller that stabilize the closed-loop in a feedback control system is assured by a full-order controller. Therefore, we should consider at least a full-order controller case for the case that the dynamics of the DC/DC back converter changes significantly.

References

[1] H. Anai and S. Hara, "A parameter space approach to fixed-order robust controller synthesis by quantifier elimination," *International Journal of Control.*, vol. 79, pp. 1321-1330, 2006.

[2] S. Boyd, L. El Ghaoui, E. Feron, and V.Balakrishnan, "Linear Matrix Inequalities in System and Control Theory," *Studies in Applied Mathematics.*, SIAM, Philadelphia, PA, vol. 15, 1994.

[3] R.F. Caviness and J.R. Johnson (Fds.), "Quantifier Elimination and Cylindrical Algebraic Decomposition, Texts and Monographs in Symbolic Computation," Springer, 1996.

[4] J.C. Doyle, B.A. Francis and A.R. Tannenbaum, "Feedback Control Theory," Prince Hall, 1992.

[5] R.W. Erickson and D. Maksimovic, "Fundamentals of Power Electronics Second Edition," Kluwer Academic Publishers, 2000.

[6] P. Gahinet, A. Nemiroviski, A.J. Laub and M. Chilali, "LMI control toolbox for use with MATLAB,"The Mathworks, 1996.

[7] A. Hurwitz, "Ueber die Bedingungen, unter welchen eine Gleichung nur Wurzeln mit negativen reellen Theilen besitzt," *Mathematische Annalen.*, vol. 46, pp 273-284, 1895.

[8] H. Iwane, H. Higuchi, and H. Anai, "An effective implementation of a special quantifier elimination for a sign definite condition by logical formula simplification," *CASC.*, to appear, 2013.

[9] H. Iwane, H. Yanami, H. Anai, and K. Yokoyama, "An effective implementation of symbolic-numeric cylindrical algebraic decomposition for quantifier elimination," *Theor. Comput. Sci*, vol. 479, pp. 43-69, 2013.

[10] H. Iwane, H. Yanami, and H. Anai, "A Symbolic-Numeric Approach to Multi-Objective Optimization in Manufacturing Design," *Mathematics in Computer Science*, vol. 50, pp. 315-334, 2011.

[11] T. Iwasaki and S. Hara, "Generalized KYP lemma: Unified frequency domain inequalities with design applications," *IEEE Trans. Automat. Control.*, vol. 50, pp. 41-59, 2005.

[12] J.G. Kassakian, M.F. Schelecht and G.C. Verghese, "Princi-

ples of Power Electronics," Prentice Hall, 1991.

[13] D.C. McFarlane and K. Glover, "Robust Controller Design Using Normalized Compromise Factor Plant Description," *Springer-verlag, Berlin, Heidelderg, Lecture Note in Control and Sciences*, no. 138, 1990.

[14] R. Ortega, A. Loria, P.J. Nicklasson and H. SriaRamirez, "Passivity-based Control of Eular-Lagrange Systems," Springer, 1998.

[15] Y. Yamamoto, "A function space approach to sampled-data control systems and tracking problems," *IEEE Trans. Automat. Control.*, vol. 39, pp. 703-713, 1994.

セッション４

Session 4

ソフトウェア

Software

MathML Content Markupで書かれた数式に対する検索手法の提案

Proposal of a search method for MathML Content Markup

片岡晃久
Akihisa Kataoka
愛媛大学大学院　理工学研究科　電子情報工学専攻
Department of Electrical and
Electronic Engineering and Computer Science,
Graduate School of Science and Engineering,
Ehime University *

甲斐博
Hiroshi Kai
愛媛大学大学院　理工学研究科　電子情報工学専攻
Department of Electrical and
Electronic Engineering and Computer Science,
Graduate School of Science and Engineering,
Ehime University †

Abstract

In this paper, We propose a Formula search method for MathML using a concept of XML fragment.We implemented a formula search system for MathML Content Markup,and examined the effectiveness of the proposal method compaired with other method.

1　序論

数式は，理工学・社会・経済などのあらゆる分野において用いられており，現象や技術などの知識を表現することに最も適した方法の一つである．Web 上で数式検索を行う場合，一般的に Web 上で用いられる検索エンジンはテキストベースであるために，数式の構造を正確に捉えることができず，適切な数式を検索することができない．一方，数式を XML[1] で表現する MathML[2] の普及により，MathML で記述された数式データを再利用可能な形で Web ページに含めることが可能になってきた．そのため，MathML で表現された数式を対象とした検索システムの開発の必要性が高まっている．MathML には Presentation Markup と Content Markup という二種類の記述形式があり、Web 上で主に用いられている記述形式は，記述の自由度の高さから Presentation Markup が多い．

MathML を対象とした数式検索の研究事例として，橋本らによる MathML を対象とした数式検索のためのインデックスに関する調査 [3] がある．これは，Presentation Markup で書かれた MathML のルートから葉ノードまでのパスを XPath 形式で表し転置インデックスを作成することで，数式検索を行っている．しかし，この検索手法では各要素間の関係が考慮されておらず，条件に完全に一致したもののみを検索結果としているため，Presentation Markup の曖昧性に対応しておらず，またユーザの曖昧な情報要求に応えることができない．数式の部分構造を考慮した数式検索の研究事例として，独自の問い合わせ言語によって数式検索を行う小田切らの MathML を用いた数式検索 [4] や，演算子や変数などの数式の構成要素の木構造

*kataoka_a@hpc.cs.ehime-u.ac.jp
†kai@cs.ehime-u.ac.jp

における位置と演算子適用範囲から数式を検索する高田らの数式の構造を反映した検索法 [5] がある．しかしこれらの数式検索は Content Markup で書かれた MathML を対象としており，Presentation Markup と Content Markup の両者を対象とした数式検索で確立された手法はいまだに存在しない．

本研究では，MathML における意味を表現するのに必要な最小単位の MathML（フラグメント）を定義する．フラグメントを用いることで，MathML の記述形式によらない数式検索手法の検討を行った．その第一歩として，本研究では Content Markup に関する従来の検索手法をフラグメントを用いて実装し，フラグメントを用いて Content Markup に関する数式検索を行うことの有効性を考察する．

2 XML による数式表現と従来の数式検索手法

コンピュータによる処理や通信のために数式を符号化する必要性は高く，様々な表現形式が開発されている．一般的に数式を扱うアプリケーションは独自の内部表現により数式を表現している．よって，異なる内部表現を用いるアプリケーション間で数式を伝達し共有するためには，数式表現を標準化する必要がある．そのため，数式を XML により表現する標準的なデータ形式として，MathML が提案された．本研究で対象とする Content Markup は，数式の意味を記述するための MathML である．数式の意味上の構造を保持しており，数式処理システムを利用して計算等を行うことが可能である．しかし，数式の正確なレンダリングを行うには不向きである．

例として，$x^2 + y = 1$ という数式を Content Markup で記述したものを図 1 に示す．

```
<apply>
   <eq/>
   <apply>
      <plus/>
      <apply>
         <power/>
         <ci>x</ci>
         <cn>2</cn>
      </apply>
      <ci>y</ci>
   </apply>
   <cn>1</cn>
</apply>
```

図 1: $x^2 + y = 1$ を表す Content Markup

MathML における意味を表現するのに必要な最小単位の MathML を本研究ではフラグメントと呼ぶ．MathML には Presentation Markup と Content Markup の二種類の記述方法があるため，Presentation Markup と Content Markup に対するフラグメントをそれぞれ Presentation フラグメントと Content フラグメントと定義する．本研究では Content Markup に対する数式検索を考えるため，以下 Content フラグメントについて述べる．

Content フラグメントは XML である MathML と同じ構造を持ち，大きく二つに分けられる．一つは apply 要素のような親要素が持つ子要素の引数を，SUBTREE 要素で置き換えたものである．SUBTREE 要素とは部分木を意味し，階層数が 1 以上の木構造が存在することを表す要素である．本研究ではこれを引数を持つフラグメントと呼ぶ．もう一つは ci 要素や cn 要素のように，引数となり子要素を持たない要素である．本研究ではこれを引数を持たないフラグメントと呼ぶ．

例えば，累乗を表す Content Markup はルートに apply 要素を持ち，子要素に power 要素と引数を二つ持たなければならない．具体的な例として，x^3 という数式に存在する Content フラグメントを考える．この数式を Content Markup で記述したものを図 2 に示す．

```
<apply>
   <power/>
   <ci>x</ci>
   <cn>3</cn>
</apply>
```

図 2: x^3 を表す Content Markup

図 2 において，引数は`<ci>x</ci>`と`<cn>3</cn>`である．これを SUBTREE 要素に置き換えたものが累乗を表す Content フラグメントとなる．SUBTREE 要素は id 属性の値により一意に区別する．また，引数である`<ci>x</ci>`と`<cn>3</cn>`は子要素を持たない要素なので，それぞれ変数 x を表す Content フラグメントと値 3 を表す Content フラグメントとなる．この Content Markup に存在する Content フラグメントを表 1 に示す．

表 1: x^3 に存在する Content フラグメント

Content フラグメント	意味

`<apply>`
`  <power/>`
`  <SUBTREE id="1"/>`
`  <SUBTREE id="2"/>`
`</apply>`
	累乗

`<ci>x</ci>`	変数 x
`<cn>3</cn>`	値 3

あらゆる Content Markup は，Content フラグメントの SUBTREE 要素を Content フラグメントに置き換えることで構成することができる．x^3 を表す Content Markup では，累乗を表す Content フラグメントの SUBTREE 要素を変数 x を表す Content フラグメントと値 3 を表す Content フラグメントで置き換えて Content Markup を構成している．つまり Content フラグメントを組み合わせることであらゆる数式が表現可能である．

3　フラグメントによる検索手法

本研究では MathML における意味を表現するのに必要な最小単位の MathML（フラグメント）を数式の持つ特徴と定義し，これを適合条件とした MathML を対象とする数式の検索手法を提案する．また高田らの研究を参考に，フラグメントを用いた数式検索システムを実装し，本提案手法の有効性を検証する．

高田らの研究では，Content Markup で書かれた MathML を対象として，ユーザの曖昧な情報要求に応える数式検索手法を提案している．曖昧な情報とは，検索キーとして数式の名前も数式自体も曖昧な特徴しか分からない場合である．よって，入力された数式から構造的な特徴を抽出し検索を行っている．具体的な方法として，演算子や変数などの数式の構成要素をキーワードと見なし，木構造におけるキーワードの位置と演算子の適用範囲を数式の構造的な特徴と定義して，S 式を入力とした MathML に対する数式検索を行っている．また，曖昧性を持つ数式検索では曖昧性を持たない数式検索と比べて，検索結果の数が多くなるため，ある尺度でランク付けをする必要がある．高田らの研究では，ランク付けをキーワード間の階層の距離と，式の大きさの二つの尺度を用いて行っている．

本提案手法では，適合条件として以下のものを用いる．

1. MathML に含まれるフラグメント
2. MathML に含まれるフラグメントの引数となっているフラグメント
3. MathML に含まれるフラグメントの木構造内での位置，階層
4. MathML に含まれるフラグメントの種類

条件 4 について，フラグメントの種類とはフラグメントが引数を持つフラグメントか引数を持たないフラグメントかを表すものである．次に，上記の適合条件を用いて数式検索を行う手順を以下に示す．

1. 検索対象の MathML について，適合条件となる情報を解析しデータベースに登録しておく．
2. 検索キーの MathML について，適合条件となる情報を解析する．
3. 検索キーの MathML の解析情報と，データベースに登録されている検索対象の MathML の解析情報から，適合を確認する．
4. 適合した検索対象の MathML を検索結果として出力する．

高田らの研究では，検索結果として適合条件に完全に一致するものだけでなく，ある程度の曖昧性を持たせている．フラグメントを用いた数式検索では曖昧性の有無による二つの適合条件を以下のように表現する．

曖昧性を持たない検索の適合条件

検索キーとなる MathML と検索対象となる MathML の適合条件は以下の通りである．

1. 検索キーが持つフラグメントを検索対象が全て持つこと．
2. 検索キーの各フラグメントと対応する検索対象のフラグメントについて，各フラグメントが持つ引数が全て一致すること．

曖昧性を持つ検索の適合条件

曖昧性を持つ検索の適合条件では，曖昧性を持たない検索の適合条件から，引数を持たないフラグメントを除いたものとなる．よって，検索キーとなる MathML と検索対象となる MathML の適合条件は以下の通りである．

1. 検索キーにおける引数を持つフラグメントを検索対象が全て含んでいること．
2. 検索キーにおける各引数を持つフラグメントと対応する検索対象の引数を持つフラグメントについて，各フラグメントが持つ引数が引数を持たないフラグメントを除いて全て一致すること．

4 実装と実験

提案手法を実装したシステムは大きく分けて以下のシステムに分かれる．

1. 検索対象となる Content Markup の解析と索引化．
2. 検索キーとなる Content Markup の解析と検索．

それぞれのシステム要素について，4.1 節と 4.2 節で述べる．また MathML Test Suite[6] を使った実験を 4.3 節でまとめる．

4.1 検索対象となる Content Markup の解析と索引化

検索対象となる Content Markup の解析と索引化では，以下のような構成のテーブルをデータベース中に用意する．なお，本研究ではデータベースに postgresql を用いた．

- 数式テーブル (数式 ID，Content Markup，ファイル名)
- 解析テーブル (数式 ID，フラグメント ID，フラグメント番号，SUBTREE 要素のフラグメント ID，階層)
- フラグメントテーブル (フラグメント ID，Content フラグメント，種類)

数式テーブルには検索対象となる Content Markup を格納する．数式 ID は Content Markup を一意に識別するための整数値である．

解析テーブルには Content Markup から抽出した Content フラグメントの Content Markup 内での情報を格納する．数式 ID とフラグメント ID はそれぞれ数式テーブルの数式 ID とフラグメントテーブルのフラグメント ID と紐付いており，数式内の各 Content フラグメントを一意に表す．フラグメント番号は数式内の Content フラグメントに振り分けた番号である．SUBTREE 要素のフラグメント ID は引数を持つフラグメントの SUBTREE 要素と置き換わる Content フラグメントのフラグメント ID である．また，引数を持たない Content フラグメントは自分自身のフラグメント ID を格納している．階層は Content フラグメントの Content Markup の階層構造における深さである．

フラグメントテーブルには，Content Markup から抽出した Content フラグメントを格納する．フラグメント ID は Content フラグメントを一意に識別するための整数値である．種類は Content フラグメントが引数を持つか持たないかを識別するものであり，引数を持つ場合は整数値の 0 が格納され引数を持たない場合は整数値の 1 が格納される．

3 つのテーブルを用意した上で，本システムでは Content Markup の解析と索引化を以下の手順で行う．

1. 検索対象となる Content Markup を読み込み，数式テーブルに Content Markup の情報を登録する．
2. 読み込んだ Content Markup を解析して Content フラグメントを抽出し，フラグメントテーブルに Content フラグメントの情報を順次登録する．また，Content フラグメントに Content Markup 内での番号を振り分け，Content フラグメントの階層を調べる．
3. 解析が終われば，数式テーブルの数式 ID とフラグメントテーブルのフラグメント ID を紐付けて解析テーブルに情報を登録する
4. 次の検索対象に処理を移る．全ての Content Markup を登録し処理を終了する．

4.2 検索キーとなる Content Markup の解析と検索

4.1 節で述べたデータベースについて，Content Markup の検索を以下の手順で行う．

1. 検索キーとなる Content Markup を解析して Content フラグメントを抽出し，データベースからフラグメント ID を取得する．また Content フラグメントに番号を振り分け，Content フラグメントの階層を調べる．
2. 検索キーの Content Markup のルートに相当する Content フラグメントのフラグメント ID から解析テーブルに問い合わせを行いレコードの情報を取得し，適合条件を満たすかを確認する．適合すれば数式 ID をリストに登録する．
3. 検索キーの Content Markup のルートより下の階層にある Content フラグメントについて解析テーブルに順次問い合わせを行い，適合を確認する．手順 2 で適合した Content Markup が手順 3 において一度でも適合しなかった場合，リストに登録しておいた数式 ID を削除する．
4. 検索キーの Content Markup の全ての Content フラグメントについて適合を確認したら，リストに登録された数式 ID から数式テーブルに問い合わせを行い Content Markup のファイル名を取得する．数式 ID とそれに対応した Content Markup のファイル名を検索結果として出力する．

手順 3 において，同じ数式 ID に複数の同じ Content フラグメントが存在する場合，適合が複数回行われる場合がある．適合した場合，同じ数式 ID の同じ Content フラグメントについては以降適合を確認しない．適合しなかった場合，手順 3 に従いリストから数式 ID を削除し，以降で適合した場合は再度リストに数式 ID を登録する．

4.3 実験

本提案手法を実装した数式検索システムの有効性を検証するために実験を行った．高田らの研究と同様に，MathML TEST Suite[6] から取得した約 350 個の Content Markup で書かれた MathML を検索対象としてデータベースに登録し検索を行った．

4.3.1 曖昧性を持たない検索

曖昧性を持たない数式検索の実験について結果を表 2 に示す.

表 2: 曖昧性を持たない数式検索の結果

入力数式	適合件数	適合例
ab	9	$-(ab)$ $ab+c$ $-(x+ab)$
$\sin x$	13	$\sin x$ $\sin x+\cos x$ $\lim_{x\to a}\sin x$ $(\sin x/\cos x)^2$

表 2 について，入力数式 ab の場合に着目すると，数式中に ab の構造を持つ数式が適合例として示されていることが確認できる．また，入力数式が $\sin x$ の場合も同様のことが確認できる．よって，入力数式の全体構造を持つ数式を検出できていると考えられる．

4.3.2 曖昧性を持つ検索

曖昧性を持つ数式検索の実験について結果を表 3 に示す.

表 3: 曖昧性を持つ数式検索の結果

入力数式	適合件数	適合例
ab	32	$x\left(a/b+c\right)-1$ $\arccos\left(5\times\pi\right)$ $\overline{x+iy}$
$\sin x$	26	$\sin\left(x\left(y+z\right)z\right)$ $\lim_{x\to a}\sin\left(x+y\right)$ $A\times B=ab\sin\theta N$
xyz	22	$x(y+z)z$ $\sin\left(x\left(y+z\right)z\right)$ $A\cdot B=ab\cos\theta$

表 3 について，入力数式 ab の場合に着目する．数式 ab 中には，引数を持つフラグメントとして 2 項乗算の Content フラグメントが存在するので，数式中に 2 項乗算の Content フラグメントを持つ数式が適合例として示されていることが確認できる．また，入力数式 $\sin x$ 中には関数 sin の Content フラグメントが存在し，入力数式 xyz 中には 3 項乗算の Content フラグメントが存在する．入力数式 ab の場合と同様に，これらの Content フラグメントを持つ数式が適合例として示されていることが確認できる．よって，入力

数式の引数を持たない Content フラグメントを他の Content フラグメントに置き換えた構造を持つ数式を検出できていることが確認できる．

4.4 従来手法との比較

高田らの研究において，本研究と同様に MathML Test Suite から取得した MathML に対する検索結果を表 4 に示す．

表 4: 高田らの研究における数式検索の結果 [5]

入力数式	適合件数	適合例
ab	34	$-(ab)$
		$x(y+z)z$
		$x(a/b+c)-1$
$\sin x$	18	$\sin(a+b)$
		$1/\sin t$
		$\sin\left(\cos x+x^3\right)$

表 3 と表 4 より，入力数式が $\sin x$ の場合は，提案手法の方が適合件数が多いことが確認できる．しかし入力数式が ab の場合は，提案手法の方が適合件数が少ない．これは，Content フラグメントが SUBTREE 要素の数で全く別の Content フラグメントとして扱われるためである．つまり，提案手法では加算などの n 項演算が項の数で明確に区別される．表 4 より，高田らの研究では入力数式 ab に対して二項乗算である $-(ab)$ と，三項乗算である $x(y+z)z$ が適合しているため，n 項演算を区別していないと考えられる．提案手法において，SUBTREE 要素が二つである乗算の Content フラグメントを持つ ab の適合件数と，SUBTREE 要素が三つである乗算の Content フラグメントを持つ xyz の適合件数を合わせると，高田らの研究における入力数式が ab の場合の適合件数より多くの数式を検索できていることが確認できる．

高田らの研究における数式検索の適合件数と，本研究の適合件数が異なる理由として，高田らの手法では，Content Markup の階層の深い部分で適合する数式を検索していないことと，構造の適合を階層での親子関係まで許可している点が挙げられる．本提案手法では Content Markup の階層の深い部分で適合する数式も検索している．しかし，構造の適合は同一の階層までしか許可していない．よって高田らの研究における数式検索手法と本研究の数式検索手法の違いは以下のようになる．

- n 項演算が区別されているか．
- 検索キーが検索対象の部分的な構造と適合することを検索しているか．
- 構造の適合を他の階層まで許可しているか．

以上の結果より，フラグメントを用いても従来手法と同等な数式検索を実現できると確認できた．また，今回は Content Markup に対する実験を行ったが，本手法は Presentation Markup についても同様な検索が同じように実現できると考えられる．

5 結論

本研究では，MathML における意味を表現するのに必要な最小単位の MathML（フラグメント）を提案し，フラグメントを用いて Content Markup に関する数式検索を行うことの有効性を検討するために，高田らの研究を参考にして，フラグメントを用いた Content Markup に関する検索手法を実装した．結果として，フラグメントを用いて従来の検索手法と同程度の数式検索が実現できることを確認した．

最後に，フラグメントを用いた Presentation Markup に関する数式検索について述べる．橋本らの研究では，Presentation Markup における数式検索手法を提案している．階層構造の一番深いパスを数式のもっとも特徴的な部分を表していると考え，これを数式検索の適合条件としている．しかし，ユーザの曖昧な情報要求に応えることができない

本研究で実装した Content Markup に関する数式検索システムと同様の方法で Presentation Markup を検索する場合に問題となるのが，Presentation Markup の数式記述に対する表現の曖昧性である．Content Markup では一つの数式表現に対して一意に表現が決まるが，Presentation Markup には一つの数式表現に対して複数の表現が存在する．例えば，数式中に ab という乗算の表記がある場合，Presentation Markup では変数 a と変数 b を連結して ab として書くことができ，またタグを用いて乗算 $a \times b$ と意味を明示して書くこともできる．そのため，Presentation Markup を対象に検索を行う場合，ユーザが想定した数式が全て検索されないかもしれない．Presentation Markup へのフラグメントの適用と予想される問題についての検討が今後の課題である．

参 考 文 献

[1] XML，http://www.w3c.org/TR/xml/

[2] MathML，http://www.w3c.org/TR/MathML/

[3] 橋本英樹，土方嘉徳，西田正吾： MathML を対象とした数式検索のためのインデックスに関する調査，情報処理学会研究報告，No. 54，pp.55-59，2007

[4] 小田切健一，村田剛志： MathML を用いた数式検索，人工知能学会全国大会，pp.1-4，2008

[5] 高田真澄，村尾裕一： 数式の構造を反映した検索法，データ工学と情報マネジメントに関するフォーラム（DEIM），C7-4，2010

[6] MathML TEST Suite，http://www.w3.org/Math/testsuite/

MathLibre: distributable and customizable desktop environment for mathematics

濱田龍義
福岡大学理学部/JST CREST/OCAMI
Tatsuyoshi Hamada
Fukuoka University/JST CREST/OCAMI *

Abstract

MathLibre offers many documents and mathematical software packages. Once you run the live DVD system, you can enjoy a wonderful world of mathematical software without installing anything yourself. MathLibre is supporting various ways to boot it, with DVD, on Virtual Machine, from USB memory disk, or hard disk. From the recent version, MathLibre 2013 is a customizable live system, if you want to make an own version, you can rebuild it for yourself easily. We will show how to use and rebuild MathLibre system.

1 序

2003 年に KNOPPIX/Math Project を開始して以来，これまでに様々な数学ソフトウェアを紹介してきた．当初 CD-R で始まったプロジェクトも，年を重ねるに連れ，収録するソフトウェアの種類を増やしていき，2006 年からは DVD で提供を行なっている．起動形態も DVD だけでなく，USB メモリーディスクや，仮想マシン等，様々な方法が用意されており，ユーザの環境に応じて用意することが可能となっている．2012 年には，後述する理由からプロジェクト名を MathLibre と改めた．最新版となる 2013 年度版では，ベースとなるシステムを KNOPPIX から変更し，ユーザが容易に改変することも可能となった．本稿では，最新版の MathLibre について，起動方法，および改変方法を中心に紹介を行う．

2 KNOPPIX/Math から MathLibre へ

KNOPPIX はドイツの Kluas Knopper を中心に開発が進められている Linux の一種である．KNOPPIX は CD/DVD や USB メモリーディスクからの起動に注力している．このような Linux ディストリビューションは Live Linux と呼ばれる．ハードウェアの自動認識に優れていること，起動速度が速いことなどから，Live Linux の代表的なものとして知られている．KNOPPIX は Debian GNU/Linux を原型に開発されている．Debian GNU/Linux は世界中の有志によって開発が進められている Linux ディストリビューションである．フリーソフトウェアの理念に忠実なディストリビューションとして知られている．数ある Linux の中でも特に人気のあるディストリビューションである Ubuntu もまた Debian GNU/Linux を原型に開発が進められている．Debian GNU/Linux は，その開発段階に応じて，oldstabe, stable, testing, unstable, experimental 等の各リリースを用意しているが，KNOPPIX はこれらリリースが混在して構築されている

*hamada@holst.sm.fukuoka-u.ac.jp

ため，新たな数学ソフトウェアのパッケージ化を検討する際にライブラリの依存性について問題が生じやすい．また，リリースの混在環境はアップデート時にパッケージの整合性の問題を引き起こしやすく恒常的な利用が難しい．そこで，2011 年頃からベースとなるシステムについて再検討を始め，まず，2012 年にプロジェクトの名称を MathLibre に変更した．さらに，2013 年 3 月には，KNOPPIX から Debian Live への開発ベースの移行を行った．この移行に際しては，姫路独協大学の野方純氏，信州大学の松本成司氏からは，数多くの有益な助言や激励の言葉をいただき，感謝している．

3　取得方法

MathLibre 最新版の DVD イメージは FTP によって `ftp://ftp.mathlibre.org/pub/mathlibre/` から提供されている．このサーバは福岡大学に設置されているが，その他にも筑波大学[1)]の岡田昌史氏，九州大学[2)]の溝口佳寛氏によってミラーサーバが設置されている．ユーザはネットワーク的に近いサーバからダウンロードすることができる．

ネットワーク上から取得した DVD イメージは，DVD-R に焼き付けてもよいが，後で述べる起動方法でも紹介しているが，USB メモリーディスクに複製したり，仮想マシン上で利用することもできる．

4　利用方法

ここでは，DVD から起動した場合について述べる．無事に DVD から起動した場合には，起動メニューが表示される．起動メニューでは標準で “Live (amd64)” が選択されており，Enter キーを押すことで Linux の起動が行われる．後で述べるが，ハードディスク等へのインストールについては，この起動メニュー内から行うことができる．

なお，PC の BIOS の設定によっては DVD が優先起動しない場合がある．この場合には再起動を行い，BIOS の起動メニューが表示されてすぐに F2 もしくは F12 等のファンクションキーを押して，起動優先順位を変更する必要がある．ただ，どのファンクションキーが BIOS 設定変更ができるかについては，機種に依存する問題であり悩ましいものである．

未確認であるが，Microsoft 社の最新の Windows 8 を採用した PC については UEFI (Unified Extensible Firmware Interface) という次世代の基本システムを採用している場合がある．その際，Secure Boot が有効になっていると Windows 以外の OS を起動することができない．本流の Debian GNU/Linux でも対応が進みつつあるが，今後，さらなる調査と検討が必要である．

無事に起動すれば，スタートメニューを通して，様々な数学ソフトウェアを利用することができる．代数，解析，幾何，統計，応用数学等様々な分野にわたる．また，論文執筆環境として TeXLive を収録しており，すぐに TeX 文書を作成，組版することができる．beamer によるプレゼンテーション，a0poster によるポスターの作成にも対応しており，学生，大学院生のための環境としてもおすすめすることができる．

なお，現在のところ，正式には多言語環境に対応していないが，信州大学の松本成司氏によって実験的な実装が公開されている[3)]．数学史などの分野においては日本語だけでなく，中国語，韓国語の入力等の要望もあり，今後の検討課題であると考えている．

[1)] `http://axis.md.tsukuba.ac.jp/MathLibre/mathlibre/`

[2)] `http://mirror.math.kyushu-u.ac.jp/mathlibre/`

[3)] `https://github.com/seijimtmt/mathlibre/tree/m17n`

5　起動方法

MathLibre の起動方法は大きく分けて DVD，USB，仮想マシン，ハードディスクの 4 種類である．

5.1　DVD

通常，MathLibre の紹介には DVD が用いられており，年に 1，2 度，DVD をプレスしている．一般に，1000 枚を基本単位として DVD のプレスを行う．これは型を作成する関係上，1000 枚以上は何枚作成しようと価格に変化がないためである．学会の年会，国際会議等では数百枚単位で配布を行う．できる限り効率良く配布できるように，発注枚数には気をつけている基本的に年に 1 度，大きなアップデートを行なっているが，MathLibre 2013 より “Debian Live” を原型にするようになったため，以前よりは頻繁にマイナーチェンジを行うことが可能となった．プレスでは対応しきれない場合には DVD-R による配布も行うことがある．しかし，プレスに比べてメディアの安定性が劣るため，あくまで一時的な対応となる．DVD は，その安定性から，配布には非常に便利な反面，起動速度が遅いという欠点もある．そこで，2012 年度については以下で説明する USB メモリーディスクも検討を行った．

5.2　USB メモリーディスク

KNOPPIX/Math および KNOPPIX を原型とした MathLibre 2012，最新版の MathLibre 2013 は USB メモリーディスクへのインストールに対応している．DVD の大きさが約 4GB のため，通常はホームディレクトリとあわせて 8GB の USB メモリーを利用する．ここ数年，8GB の製品の値段が下がっていることもあり，2012 年度には試験的に配布実験を行った．結果はと言えば，製品の品質が想定していた以上に悪く，確認しているだけでも，ほぼ 1 割の製品で読み込み書き込みエラーが発生するという状態であった．USB メモリーディスクによる起動は，DVD に比べれば，読み込み速度も速く，継続的なホームディレクトリを作成することができる点からも重宝するが，DVD に比べると，複製方法がわかりにくいという欠点もある．現在の最新版である MathLibre 2013 では，mksusbmath という管理者向けのスクリプトを新たに実装し，ユーザが自分で作成する環境を整えている．継続的な利用ができるようになった反面，既存の Windows との共用利用という観点からは，利用の際に，その都度，再起動を行わなければいけないという欠点がある．また，USB による起動，および DVD による起動については Linux のドライバが対応していない際に，煩雑な手間を要する．以前に比べれば，それほど困ることはなくなったが，無線 LAN が利用できない等の問題が生じた場合には，その対応に困惑するユーザも多いかもしれない．そこで，次に紹介する仮想マシンを利用することをおすすめしている．

5.3　仮想マシン

最近の PC の CPU 速度の向上，およびメモリーの巨大化により，ソフトウェアを用いて仮想的に PC 環境を構築し，その上でオペレーティングシステムを稼働させることができるようになった．現在，仮想マシンソフトウェアとして入手できるものは幾つか存在するが，ここでは　VMware Player を紹介する．VMware Player は VMware 社によって開発，公開されている仮想マシンソフトウェアである．Windows や Linux 上で動作し，インターネット上に無料で公開されている．Mac OS については，VMware Fusion という商品が販売されている．

VMware Player および VMware Fusion 上で MathLibre を動かす一番簡単な方法は，DVD もしくは DVD の ISO イメージを利用して起動すればよい．この場合，マウスのドラッグアンドドロップ，時刻の

同期等の機能を追加するには VMware Tools と呼ばれるソフトウェアを追加インストールする必要がある．そこで，神戸大学では VMware Tools をインストール済みの仮想マシンパッケージを公開している[4]．パッケージを展開し，DVD の ISO イメージを追加すれば，すぐに MathLibre 環境を手に入れることが可能である．仮想マシンを通すことで，若干，速度は落ちるが，それでも，ほぼ全てのハードウェア依存の問題を避けることができることは，大変，魅力的である．

問題点としては，仮想マシンの概念自体がユーザ全般に広がっているとは言えない点である．現在，サーバ利用等において仮想マシンの概念は十分に周知されていると思われるが，一方でクライアント側での利用，特に初心者にとっては敷居が高いと思われる．MathLibre のような環境にとっては，非常に便利なものであるだけに，誘導が難しい点が残念である．

5.4 ハードディスクへのインストール

2010 年にインド，ハイデラバードで行われた国際数学者会議に参加し，MathLibre の前身である KNOPPIX/Math を紹介した．その際，最も多かった質問，要望が，このシステムをハードディスクにインストールして恒久的に使えるかどうかということである．当時の KNOPPIX/Math も “0wn” という命令を用いてインストールが可能であったが，KNOPPIX の特性上，アップデートを行うと，パッケージ間の整合性が崩れやすいという欠点があるため，あまり推奨することができなかった．

MathLibre 2013 では，Debian Live を原型に用いることで，ハードディスクへのインストールに対応している．起動メニューから “Install”，もしくは “Grahpical Install” を選択することで，インストールが可能である．また，容量の大きなものであれば，USB フラッシュディスクへのインストールも可能である．注意点としては，この場合のインストールには圧縮ファイルシステムを用いないため，全体で約 15GB ほどの容量を必要とすることである．したがって USB フラッシュディスクを用いる際は 32 GB 以上の環境が望ましい．

インストールされた状態は Debian GNU/Linux と呼ばれる Linux ディストリビューションの一種となるので，パッケージ管理システムによるアップデートも可能である．MathLibre では標準環境として LXDE という軽量のデスクトップ環境を用いているが，好みに応じて GNOME や KDE といったデスクトップ環境等への変更もできる．

6 再構築方法

MathLibre 2013 より Debian GNU/Linux を原型として開発を進めている．特に Debian の公式プロジェクトである “Debian Live” の成果に負うところが大きい．Debian Live を用いることで，比較的簡単に収録しているソフトウェアの構成をカスタマイズすることができるようになった．以下に簡単に紹介する．

まずは，再構築のための Debian 環境を用意する．現行 stable バージョンである．wheezy をインストールしたマシンを用意し，Debian Live のイメージを作成するパッケージである live-build，バージョン管理システム git をインストールする．また，パッケージ管理システム apt で用いたファイルをキャッシュする apt-cacher-ng をインストールしておくと良い．この際，配布されている MathLibre 自身を再構築環境として利用することもできる．

再構築のための環境が整ったら，git を用いて，作業領域に MathLibre の設定ファイルを取得する．MathLibre は DVD の大きさに収まるように圧縮ファイルシステムを用いている．従って，展開時の容量を考え，最低でも 40GB ほどは確保しておいたほうが良い．外部ハードディスクを用いるのも一つの方法である．

[4] http://www.math.kobe-u.ac.jp/vmkm/

```
git clone https://github.com/knxm/mathlibre/
```

作業ディレクトリに mathlibre というディレクトリが作成される．ディレクトリ内に移動し，

```
make
```

を実行することで，DVD や USB 用のディスクイメージが作成される．

```
make ja
```

で日本語版を構築することができる．

主な設定ファイルは config というディレクトリ内に記述されている．apt, archives, bootloarders, hooks, includes.chroot, package-lists の 6 個のディレクトリが git レポジトリに登録されている．

apt TEXLive をインストールすると，各種ドキュメントも同時にインストールされる．これらのドキュメント群は，かなりの容量を必要とするため，これらのファイルを削除する設定を preferences に記述している．また，X11 や GNOME に関するドキュメントも同様である．

archives apt で参照するリポジトリの URL，及び暗号キーを記述したファイルを収録している．ファイル名の末尾が chroot で終わっているものは構築時に利用するリポジトリであり，binary で終わっているものは，構築された ISO イメージ内で利用されるリポジトリである．

bootloarders 各種，起動形式に応じて，起動メニューおよび背景画面が収録されている．

hooks chroot での構築時や起動時，最終的なイメージ確定時に自動的に実行される命令を記述して置いておくためのディレクトリである．

includes.chroot chroot での構築時に，上書き展開するためのディレクトリであり，Sage をインストールするためにも用いている．includes.chroot が chroot でのルートディレクトリとなり，apt パッケージによるインストール以外で対応する際に重要なディレクトリである．

package-lists Debian のパッケージ管理システム apt を用いてインストールするパッケージ名を記述しておくためのディレクトリである．現在は，そのライセンスに応じて，何種類かのファイルに分割しているが，今後，アーキテクチャに依存したファイルについても，適宜，整理を行う必要がある．

この中でも，includes.chroot と package-lists は独自にソフトウェアやドキュメントを収録するときに特に重要なディレクトリであることに注意しておく．MathLibre は統計処理システム R を収録しているため，数学以外の他の分野からも注目を浴びているようである．例えば，心理学に関するツール等を集めた Live DVD を MathLibre をベースにして制作しようという試みも存在する．

MathLibre では，Debian Live の機能に加え，mathlibre ディレクトリ下の lang ディレクトリに独自に言語環境設定ファイルを用意している．このことにより，各国語版への対応も容易になった．2013 年 7 月に釜山で行われた AMC2013 の影響もあり，今後はベトナム，インドネシア等の言語環境への対応についても検討している．理想的には，各言語を母国語とする開発者の協力が必要であり，各国ごとに独自のドキュメントの収録等が行われるような体制に持っていきたい．

7　今後の課題

今のところ，Debian Live ベースの MathLibre は amd64 のみに対応している．i386 への対応の要望もあるが，作業の量を考えると悩ましい限りである．また，KNOPPIX ベースで実装していたドキュメント

検索システム MathDocSearch の再実装が遅れている．こちらについては，新たな検索システムの採用も含めて近日中に検討したいと考えている．

参　　献

[1] MathLibre Project, http://www.mathlibre.org/

[2] KNOPPIX/Math 作成方法, 濱田龍義, 数理解析研究所講究録 第 1572 巻 2007 年 p94–108.

[3] KNOPPIX/Math/2011 について, 濱田龍義, 数理解析研究所講究録 第 1793 巻 2012 年 p46–49.

[4] Debian Live Project, http://live.debian.net/

[5] 関西 Debian 勉強会 Wiki, のがたじゅん, http://www.nofuture.tv/index.rb?DebianLive

[6] Ubuntu Weekly Recipe: 第 113 回　Debian Live の live-helper を使って Ubuntu Live を作成する, のがたじゅん, http://gihyo.jp/admin/serial/01/ubuntu-recipe/0113

[7] Ubuntu Weekly Recipe: 第 114 回　Debian Live の live-helper を使って Ubuntu Live を作成する (2), のがたじゅん, http://gihyo.jp/admin/serial/01/ubuntu-recipe/0114

[8] 信州大学ロボティックス入門ゼミ, 松本成司, http://yakushi.shinshu-u.ac.jp/robotics/?DebianLive

セッション５
Session 5

教　育
Education

数式処理を用いたルービックキューブの素数位数操作の探求

A hunting of operations with prime order on Rubik's Cube using computer algebra

藤本 光史
Mitsushi Fujimoto
福岡教育大学
Fukuoka University of Education *

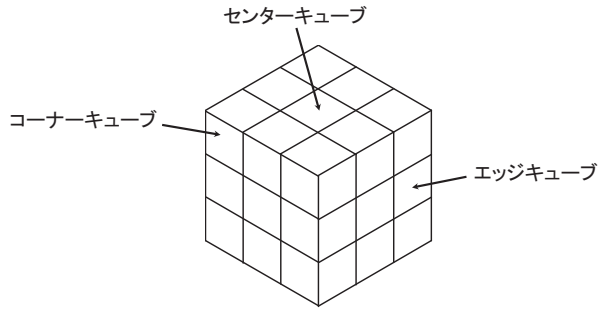
泊 昌孝
Masataka Tomari
日本大学文理学部
Nihon University †

Abstract

Rubik's cube is a suitable teaching material which connects puzzles to mathematics. In particular, every student will be surprised the phenomenon which any sequence of moves returns the Rubik's cube to the original position by repeating the operations. This phenomenon is able to be explained by group theory. In this article, we would like to report a result for a hunting of operations with prime order on Rubik's Cube using computer algebra systems.

1 はじめに

ルービックキューブは、ハンガリーの建築学者 Ernö Rubik によって 1974 年に考案された立方体パズルである。3 × 3 × 3 のタイプが「ルービックキューブ」として良く知られているが、この他に 2 × 2 × 2（ポケットキューブ）、4 × 4 × 4（ルービックリベンジ）、5 × 5 × 5（プロフェッサーズキューブ）などのタイプがある。3 × 3 × 3 のタイプは、8 個のコーナーキューブと、12 個のエッジキューブと、6 個のセンターキューブから成り、各面を回転させることで様々な模様を作ることができる。

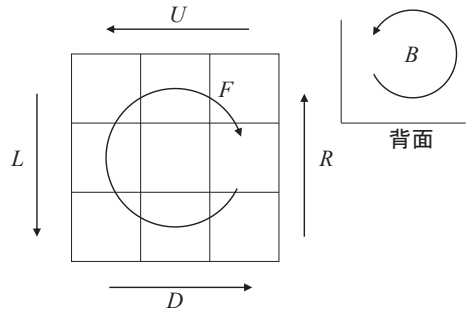


ルービックキューブは「バラバラになった状態から 6 面を揃える」ことが基本的な遊び方であり、6 面が簡単に揃えられるようになれば、「様々な模様を作る」という応用的な遊び方もできる。しかし、ここでは 3 × 3 × 3 のタイプのルービックキューブを使った「**同一の操作を繰り返すと必ず元に戻る**」という現象に着目する。

*fujimoto@fue.ac.jp
†tomari@math.chs.nihon-u.ac.jp

2　ルービックキューブの操作表現

ルービックキューブの左側面、右側面、背面、前面、下面、上面を時計回りに 90 度回転する操作をそれぞれ L, R, B, F, D, U で表す。



ルービックキューブのあらゆる操作は、この L, R, B, F, D, U とそれらの逆の積として表現できる。

例 1

$R'UL U'RUL'U'$ は「上面の右奥・右前・左前のコーナーキューブをこの順で置換する操作」である。ただし、R' は操作 R の逆（つまり、右側面を反時計回りに 90 度回転）を表す。

ルービックキューブのセンターキューブは、L, R, B, F, D, U の操作で回転するだけでその位置は変化しない。そこで、センターキューブを除いて以下の図のように 1 から 48 の番号を付ける。

```
                +--------------+
                |  1    2    3 |
                |  4   Up    5 |
                |  6    7    8 |
+---------------+---------------+---------------+---------------+
|  9   10   11  | 17   18   19  | 25   26   27  | 33   34   35  |
| 12  Left  13  | 20  Front 21  | 28  Right 29  | 36  Back  37  |
| 14   15   16  | 22   23   24  | 30   31   32  | 38   39   40  |
+---------------+---------------+---------------+---------------+
                | 41   42   43  |
                | 44  Down  45  |
                | 46   47   48  |
                +--------------+
```

この番号を用いて、L, R, B, F, D, U の各操作は次のような巡回置換の積で表現できる。

$$
\begin{aligned}
L &= (9,11,16,14)(10,13,15,12)(1,17,41,40)(4,20,44,37)(6,22,46,35)\\
R &= (25,27,32,30)(26,29,31,28)(3,38,43,19)(5,36,45,21)(8,33,48,24)\\
B &= (33,35,40,38)(34,37,39,36)(3,9,46,32)(2,12,47,29)(1,14,48,27)\\
F &= (17,19,24,22)(18,21,23,20)(6,25,43,16)(7,28,42,13)(8,30,41,11)\\
D &= (41,43,48,46)(42,45,47,44)(14,22,30,38)(15,23,31,39)(16,24,32,40)\\
U &= (1,3,8,6)(2,5,7,4)(9,33,25,17)(10,34,26,18)(11,35,27,19)
\end{aligned}
$$

3　ルービックキューブと群

前節の結果により、ルービックキューブの操作の全体は、L, R, B, F, D, U を生成元とする置換群（48次対称群 S_{48} の部分群）となる。数式処理システム GAP [1] では、次のように簡単にルービックキューブ群を定義できる。

```
gap> L := (9, 11, 16, 14)(10, 13, 15, 12)(1, 17, 41, 40)(4, 20, 44, 37)
(6, 22, 46, 35);;
gap> R := (25, 27, 32, 30)(26, 29, 31, 28)(3, 38, 43, 19)(5, 36, 45, 21)
(8, 33, 48, 24);;
gap> B := (33, 35, 40, 38)(34, 37, 39, 36)(3, 9, 46, 32)(2, 12, 47, 29)
(1, 14, 48, 27);;
gap> F := (17, 19, 24, 22)(18, 21, 23, 20)(6, 25, 43, 16)(7, 28, 42, 13)
(8, 30, 41, 11);;
gap> D := (41, 43, 48,46)(42, 45, 47, 44)(14, 22, 30, 38)(15, 23, 31, 39)
(16, 24, 32, 40);;
gap> U := (1, 3, 8, 6)(2, 5, 7, 4)(9, 33, 25, 17)(10, 34, 26, 18)
(11, 35, 27, 19);;
gap> Cube := Group(L,R,B,F,D,U);
```

次の定理は、群論を学んだことがある人には馴染み深いものであろう。

定理 1 (Lagrange)
G:有限群, $H \leq G \Rightarrow |H|$ は $|G|$ の約数

$Cube$ をルービックキューブ群、s を L, R, B, F, D, U とその逆の積で表した操作とする。上の Lagrange の定理により、$|\langle s \rangle|$ は $|Cube|$ の約数となる。すなわち、操作 s は有限の位数 d を持つ。故に、操作 s を d 回繰り返すと $s^d = e$ により元の状態に戻る。

例 2
操作 $R'\,L\,U\,D$ は何回繰り返すと元に戻るか、*GAP* で位数を求めてみよう。

```
gap> Order(R^-1*L*U*D);
12
```

この結果から、操作 $R'\,L\,U\,D$ のルービックキューブ群における位数は *12* であり、この操作を *12* 回繰り返すと元に戻ることがわかる。

4　素数位数問題

前節で「同一の操作を繰り返すと必ず元に戻る」理由が群論の初等的な結果から証明されたが、この現象に気が付いた者は、次の疑問を抱くことだろう。

問題 1 (最大位数問題)
ルービックキューブ群における元の最大位数を求めよ。また、最大位数を持つ元はどのようなものか？

これは「どんな出鱈目な操作を行っても、それを最悪何回繰り返せば元に戻るか？」という問題と見ることもできる。この問題に対する解答は、1981 年に発行された書籍 [2] で与えられた。ルービックキューブ群における元の最大位数は $1260 = 2^2 \cdot 3^2 \cdot 5 \cdot 7$ であり、その操作は $R\,F^2\,B'\,U\,B'$ である。GAP で位数を計算すると、確かに 1260 であることがわかる。このような短い操作 (6 操作) で最大の位数になるという事実には驚かされる。

また、ルービックキューブ群の位数は、GAP で簡単に確認できる。

```
gap> Size(Cube);
43252003274489856000
```

$43252003274489856000 = 2^{27} \cdot 3^{14} \cdot 5^3 \cdot 7^2 \cdot 11$ であり、このことから次の問題が考えられる。

問題 2 (素数位数問題)
ルービックキューブの素数位数を持つ操作を求めよ。

ルービックキューブ群の位数の素因数分解から、この問題は「位数 2, 3, 5, 7, 11 の操作を求めよ」ということになる。素数位数を持つ操作の存在性は、以下の有名な定理から保証される。

定理 2 (Cauchy)
G:有限群, $|G| = pk$ (p は素数) $\Rightarrow$ G は位数 p の元を持つ

5 位数 2 と 3 の操作

ルービックキューブの位数 2 の操作は、キュービストと呼ばれるルービックキューブ愛好者なら誰でも知っている。以下にそのいくつかを紹介する。

- 最短操作 R^2
- 格子模様 $L^2\,R^2\,B^2\,F^2\,D^2\,U^2$
- Rubik's Maneuver（上面の 2 個のエッジキューブを位置を変えずに 180 度入れ替える操作）
  $R'\,L\,F'\,R'\,L\,D'\,R'\,L\,B'^2\,R\,L'\,D'\,R\,L'\,F'\,R\,L'\,U'^2$
- Superflip（12 個の全エッジキューブを位置を変えずに 180 度入れ替える操作）
  $F\,R\,F'\,R\,F\,B\,L'\,U\,R\,L\,B\,U\,F'\,D^2\,R\,D'\,B\,U'\,L\,D'\,F\,D'\,B'$

次の位数 3 の操作も、バラバラの状態から 6 面を揃える際に用いられる技の中に現れる。

- 上面の右奥・右前・左前のコーナーキューブの位置をこの順で置換する操作
  $R'\,U\,L\,U'\,R\,U\,L'\,U'$
- 上面の右奥・右前のコーナーキューブを位置はそのままで向きだけを順に置換する操作
  $R'\,U^2\,R\,U\,R'\,U\,R\,L\,U'^2\,L'\,U'\,L\,U'\,L'$

6　位数 11 の操作の探求

前節で見たように、6 面を揃える際に用いられる技の中には、位数 2 や位数 3 の操作が数多く見られる。しかし、位数 5, 7, 11 の操作は、6 面を揃える際には通常使用されない。それは、位数 5, 7, 11 の操作もいくつかの小キューブの巡回置換（または巡回置換の積）であり、ルービックキューブのプレーヤーがこれらの操作を用いて目的の配置を得るには、位数 2 や 3 の操作と比較して繰り返す回数が多くなってしまうためである。

つまり、位数 5, 7, 11 の操作はキュービストが精通していない操作と言える。これらの操作を数式処理システムの GAP/Magma/Sage を用いて見つけたい。

6.1　GAP の利用

有限群論で最も有名な定理は、次の Sylow の定理であろう。

定理 3 (Sylow)
G:有限群, $|G| = p^m k$ (p は素数, p と k は互いに素) $\Rightarrow$ G は位数 p^m の部分群を持つ

GAP は Sylow 部分群を求めることができる。

```
gap> SylowSubgroup(Cube,11);
Group([ (4,7,12,39,44,21,26,31,42,20,29)(5,45,23,13,36,10,18,37,47,15,28) ])
```

このように、ルービックキューブ群の 11-Sylow 部分群を得た。この部分群の生成元

$$s = (4, 7, 12, 39, 44, 21, 26, 31, 42, 20, 29)(5, 45, 23, 13, 36, 10, 18, 37, 47, 15, 28)$$

は位数 11 の元であり、この置換 s を L, R, B, F, D, U を用いて表現することが次のステップである。

このためには、操作 L, R, B, F, D, U を生成元とする自由群からルービックキューブ群 $Cube$ への準同型写像

$$\phi : \langle L, R, B, F, D, U \rangle \ni word \rightarrow permutation \in Cube$$

による置換 s の逆像（の一つ）を求めればよい。これを求める関数を GAP のプログラミング言語で書くと次のようになる。

```
GetWordOfElements:=function(G,GenName,x)
  local gen,F,hom;
  F:=FreeGroup(GenName);
  gen:=GeneratorsOfGroup(G);
  hom:=GroupHomomorphismByImages(F,G,GeneratorsOfGroup(F),gen);
  return PreImagesRepresentative(hom,x);
end;
```

この関数を用いれば、どんなルービックキューブの状態からでも 6 面を揃える操作を求めることができる[1)]。実際、藤本はこの関数を利用したルービックキューブ解法表示ソフト [3] を開発した。

この関数を用いて、上で求めた位数 11 の元 s から L, R, B, F, D, U による操作を求める。

[1)] ただし、この関数から得られるルービックキューブの解法が最小手数であることは保証されない。

```
gap> GetWordOfElements(Cube,["L","R","B","F","D","U"],
(4,7,12,39,44,21,26,31,42,20,29)(5,45,23,13,36,10,18,37,47,15,28));
L^-1*U*B^-1*U^-1*B^-1*F^-1*U^-1*B*L^2*D^-1*R*F^-1*R^-1*D*L^2*R^-1*D^-1*R*D*F^-1
*D^-1*U*L^-1*U^-1*F^-1*L^-1*F*U*F*L*F^-1*U^-1*F*D*F^2*D^-1*F^-1*L*D^-1*L^-1*D*F
*L*F^-2*D*F*D^-1*F*L*F^-1*L^2*F^-1*L^-1*F*L*F*U^-1*F^-1*U*F*L*F^-1*L*D^-1*L*D
*L^-2*F*L^-1*U*L*U^-1*L^-1*F^-2*D^-1*L^-1*D*L*F*L^-1*D^-1*L^-1*B^-1*L*B*D*L*F*U
*L*U^-1*L^-1*F^-1
Length(last);
100
```

6.2 Magma の利用

GAP を利用して求めた位数 11 の操作は 100 手もあり、とても覚えられるものではない。この結果は、GAP の組み込み関数 PreImagesRepresentative が出力したものであり、GAP のソースを読んでそのアルゴリズムを解析するのは最後の手段にして、別の数式処理システム Magma [4] を試してみることにしたい。

Magma 上で、ルービックキューブ群は次のようにして定義できる。

```
magma> s48:=Sym(48);
magma> L := s48!(9, 11, 16, 14)(10, 13, 15, 12)(1, 17, 41, 40)
(4, 20, 44, 37)(6, 22, 46, 35);
magma> R := s48!(25, 27, 32, 30)(26, 29, 31, 28)(3, 38, 43, 19)
(5, 36, 45, 21)(8, 33, 48, 24);
magma> B := s48!(33, 35, 40, 38)(34, 37, 39, 36)(3, 9, 46, 32)
(2, 12, 47, 29)(1, 14, 48, 27);
magma> F := s48!(17, 19, 24, 22)(18, 21, 23, 20)(6, 25, 43, 16)
(7, 28, 42, 13)(8, 30, 41, 11);
magma> D := s48!(41, 43, 48,46)(42, 45, 47, 44)(14, 22, 30, 38)
(15, 23, 31, 39)(16, 24, 32, 40);
magma> U := s48!(1, 3, 8, 6)(2, 5, 7, 4)(9, 33, 25, 17)
(10, 34, 26, 18)(11, 35, 27, 19);
magma> Cube:=PermutationGroup<48|L,R,B,F,D,U>;
```

さらに、準同型 $\phi : \langle L, R, B, F, D, U\rangle \ni word \to permutation \in Cube$ を定義し、置換

$$s = (4,7,12,39,44,21,26,31,42,20,29)(5,45,23,13,36,10,18,37,47,15,28)$$

の ϕ による逆像として、位数 11 の操作を求める。

```
magma> w<L,R,B,F,D,U>:=FreeGroup(6);
magma> images := [w.i -> Cube.i : i in [1..6] ];
magma> phi:=hom< w -> Cube | images >;
magma> s:=s48!(4,7,12,39,44,21,26,31,42,20,29)(5,45,23,13,36,10,18,37,47,15,28);
magma> s @@ phi;
```

しかし、得られた結果は 1600 行を超える異常に長い操作になってしまった。この原因は、自由群 $F_6 = \langle L, R, B, F, D, U\rangle$ に対して関係式を与えていないためだと思われる。そこで、ルービックキューブ群の

L, R, B, F, D, U を用いた生成元と関係式による群表示について過去の研究成果を調査したが、5 個の生成元と 44 個の関係式による群表示 [5] しか見つけることができず断念した。

6.3 Sage の利用

ここでは、GAP で得られた位数 11 の 100 手から成る操作を「簡約」するアプローチについて述べる。次の問題は、ルービックキューブの最小手数問題として有名である。

問題 3

$3 \times 3 \times 3$ のルービックキューブを解くために必要な手数の上限を求めよ。

ここでの手数は face turn metric であり、連続する同じ操作（つまり、180 回転する操作）は 1 手とカウントするものである。この上限の手数は God's number と呼ばれており、2010 年に Tomas Rokicki, Herbert Kociemba, Morley Davidson, John Dethridge によって、20 手であることが示された。

数式処理システム Sage [6] は、Michael Reid による Optimal Solver アルゴリズム [7] が実装されており、与えられた操作の簡約が可能である。ただし、この実装は face turn metric ではなく、quarter turn metric（90 度回転を 1 手とカウントする）での最適解を求めるものとなっている。Sage を用いて、GAP が求めた位数 11 の 100 手の操作の簡約は以下のように行う。

```
sage: rubik = CubeGroup()
sage: s = rubik.faces("L^-1*U*B^-1*U^-1*B^-1*F^-1*U^-1*B*L^2*D^-1*R*F^-1
*R^-1*D*L^2*R^-1*D^-1*R*D*F^-1*D^-1*U*L^-1*U^-1*F^-1*L^-1*F*U*F*L*F^-1
*U^-1*F*D*F^2*D^-1*F^-1*L*D^-1*L^-1*D*F*L*F^-2*D*F*D^-1*F*L*F^-1*L^2*F^-1
*L^-1*F*L*F*U^-1*F^-1*U*F*L*F^-1*L*D^-1*L*D*L^-2*F*L^-1*U*L*U^-1*L^-1*F^-2
*D^-1*L^-1*D*L*F*L^-1*D^-1*L^-1*B^-1*L*B*D*L*F*U*L*U^-1*L^-1*F^-1")
sage: ans = rubik.solve(s, algorithm='optimal')
sage: print ans
F  R' L  F  U  D  L  B  U' F' U' F' D  F' B  L  U' L' U  R
```

以上によって、位数 11 の 20 手の操作を得ることができた。

7 位数 5 と 7 の操作

前節で得られた手法を以下の手順で位数 5,7 にも適用する。

1. 【GAP】ルービックキューブ群を定義し、SylowSubgroup 関数を用いて、5-Sylow 部分群と 7-Sylow 部分群を求め、その生成元から位数 5,7 の置換表現を得る。
2. 【GAP】GetWordOfElements 関数を用いて、位数 5,7 の置換表現から対応する L, R, B, F, D, U による操作を求める。
3. 【Sage】CubeGroup().solve 関数を用いて、操作手数の簡約を行う。

これによって、次の操作が得られた。

- （位数 5）12 個のエッジキューブのうち 7 個を固定し、残り 5 個の巡回置換を行う操作
  $D\,F'\,R\,U'\,F'\,L'\,R\,U\,L\,R'\,U\,R'\,F\,D'$

- （位数 5）8 個のコーナーキューブのうち 3 個を固定し、残りの 5 個の巡回置換を行う操作
  $RU\,F\,U'\,L\,U\,F'\,U'\,R'\,F\,L'\,F'$
- （位数 7）8 個のコーナーキューブのうち 1 個を固定し、残りの 7 個の巡回置換を行う操作
  $R'\,U\,R\,D\,F^2\,R'\,D\,U'\,R\,U'\,R\,D'\,R\,D'\,U\,F^2$

8 記憶可能な操作の探求

GAP と Sage を駆使して、ルービックキューブの素数位数 2,3,5,7,11 の操作を求めることができた。これで問題 2 は解決である。素数位数の操作を求めることができたので、これを実演したいと考えるのは自然なことであろう。そこで実際のルービックキューブを用いて練習してみたが、何度やっても操作を記憶することができない。ルービックキューブの F や B の操作は、手の移動距離が大きく、できるだけ避けたい操作である。それで、求めた素数位数の操作を F や B をなるべく使用しないように置換してみたが、それでも記憶することは不可能であった。そして、我々は次の問題を考えることにした。

問題 4
ルービックキューブの記憶可能な単純な素数位数操作を求めよ。

位数 11 の操作について考えることにする。前々節で得られた手数 20 の位数 11 の操作は、12 個のエッジキューブのうち 1 個を固定し、残りの 11 個の巡回置換を引き起こす。下面と背面にまたがるエッジキューブを固定することにすると、それを動かさない操作は、U, F, L, R の 4 操作である。よって、位数 11 の操作はこの 4 操作の組み合わせで作ることが自然と考えられる。後は、4 操作の記憶しやすい組み合わせ x を作り、その位数を GAP で計算する。位数が 11 の m 倍になったとき、x^m が求める位数 11 の操作になる。この手順をまとめると次のようになる。

1. 位数 p の操作がルービックキューブ上のどのような巡回置換を引き起こすか考える。
2. 使用する操作を限定する。
3. 限定した操作から記憶しやすい単純な組み合わせを考える。
4. その組み合わせの位数を Gap で計算する。
5. 位数が p の m 倍になるものを見つける。

この手順の $3 \to 4 \to 3 \to 4 \to \cdots$ の繰り返しにより、

```
gap> Order((U*R^-1)^3*(L*F^-1)^3);
33
gap> Order(L*R^-1*F*U^-1);
44
gap> Order(R^-1*U*F^-1*L);
33
```

という組み合わせを見つけることができた。この最後の結果から、位数 11 の記憶可能な単純な操作（手数 12）が得られた。また、同様の手法により、位数 7,5 の記憶可能な単純な操作も得られた。以下に、これらの結果をまとめておく。

- 位数 11 の操作：$(R'\,U\,F'\,L)^3$
- 位数 7 の操作：$(D^2\,R)^2\,(U^2\,L)^2$
- 位数 5 の操作：$R'\,U\,R\,U$

9 おわりに

本稿ではルービックキューブの素数位数操作を数式処理システムを利用して求める手法について解説した。ここで取り扱った問題は、大学での代数学の講義や高校での出前授業などの教材として用いることが可能である。実際、我々は所属する大学での講義や高校での出前講演で使用している。ルービックキューブは、数学とパズルを結びつける格好の教材の一つと言えよう。

最後に、二つの発展的な問題を残してこの稿を終えることとする。

問題 5

ルービックキューブ群の元が取り得るすべての位数を求め、その位数に対応する最短の操作を求めよ。

問題 6

人間が記憶可能なルービックキューブの操作を自動生成することは可能か。

参 考 文 献

[1] GAP – Groups, Algorithms, Programming – a System for Computational Discrete Algebra, http://www.gap-system.org

[2] David Singmaster, Notes on Rubik's Magic Cube, Enslow Pub Inc, 1981.

[3] 田崎拓馬, 藤本光史, GAP を用いた Rubik's Cube 解法表示ソフトについて, 京都大学数理解析研究所講究録 1652, 「Computer Algebra – Design of Algorithms, Implementations and Applications」(2009) 125–131.

[4] Wieb Bosma, John Cannon, and Catherine Playoust, The Magma algebra system. I. The user language, J. Symbolic Comput., 24 (1997) 235–265. http://magma.maths.usyd.edu.au/magma/

[5] Dan Hoey, Presenting Rubik's Cube, Cube-lovers mailing list, http://www.math.rwth-aachen.de/~Martin.Schoenert/Cube-Lovers/Dan_Hoey__Presenting_Rubik%27s_Cube.html

[6] William A. Stein, et al., Sage Mathematics Software (Version 5.9), The Sage Development Team, 2013, http://www.sagemath.org

[7] Michael Reid, Optimal Rubik's cube solver, http://math.cos.ucf.edu/~reid/Rubik/optimal_solver.html

数独パズルの計算機による解析について
On the Analysis of Sudoku Puzzles by Computers

北本　卓也
Takuya Kitamoto*
山口大学
Yamaguchi University

Abstract

This paper presents a method to analyze Sudoku puzzles by computers. Generally speaking, it is not so difficult to solve Sudoku puzzles by computers, and many programs to solve Sudoku puzzles are avaiable. However, most of the programs use recursion and backtracking, which is significantly different from methods used by a human. Hence, a human-like method to solve a Sudoku puzzle is unknown even if the solution of the puzzle is computed by compute programs.

We created a computer program which solves Sodoku program with human-like methods. The program employs three basic techniques and solve almost all Sudoku puzzle in published books and magazines.

1　はじめに

数独パズルと呼ばれるバズルが流行しており、書店に行くとそのパズルの雑誌や書籍がたくさんおいてある。このパズルは日本だけでなく、世界中で流行しており、世界大会も行われている。

計算機を用いて、この数独パズルを解く試みが行われている。数学的な手法を用いた方法としては、Boolean Groebner Bases を用いた方法（[1], [2]）や、マルコフ基底を用いた方法（ [3]）などがあるが、世の中に出ている数独パズルを解くフリーソフトのほとんどは「仮置き」と呼ばれる単純な方法を用いている。「仮置き」を用いればどんなパズルも解くことができるが、その方法は通常、人間が手でパズルを解く時の手法とは異なっている（人が手でパズルを解くときには、仮置きは論理的でなく避けるべき手法だと考えられている)。このため、コンピュータソフトを用いればどんなパズルでもその答えはわかるが、どうやればそのパズルが解けるかまではわからない。

そこで、なるべく人間が行うものに近い方法でパズルを解くプログラムを作成した。本稿では、そのプログラムの解法アルゴリズムと、それを用いて実際のパズルを解かせた結果を報告する。

2　数独パズルとは？

数独パズルは 9 × 9 マスからなるパズルであり、各マスには 1 〜 9 までの数字が入る。また、各列、各行、各サブブロック（9 × 9 マスを 9 つの 3 × 3 のマスの塊に分けたもの）には 1 〜 9 までの数字が 1 つずつ入る。あらかじめ、いくつかのマスが数字で埋められた 9 × 9 マスが与えられたとき、上のルールのもとでまだ埋まっていないマスの数字を決めていくパズルである。

*kitamoto@yamaguchi-u.ac.jp

3　計算機による数独パズルの表現

プログラム内では、数独パズルを 9×9 の行列（行列の各要素はパズルの各マスに当たる）で表し、行列の各要素を 1〜9 までの数字を要素とする集合（例えば $\{1,3,8,9\}$）とする。この集合の要素は、そのマスの数字として可能性がある数字の集合であり、先ほど挙げたルールを用いてこの数字を削除していくことがパズルを解くことである。最終的に集合の要素が 1 つになると、そのマスの数字が確定する。全てのマスの数字が確定すれば（すなわち、全てのマスの集合の要素数が 1 になれば）、パズルが解けたことになる。以下では、$\mathrm{A}_{i,j}$ で (i,j) マスの数字の候補を表すことにする。

4　解法アルゴリズム

3 つの基本テクニックを用いてパズルを解いていき、それでも解けない場合には、仮置きを用いる。

4.1　基本テクニック 1 — n 国同盟

これは、一般に 2 国同盟、3 国同盟と呼ばれているテクニックの一般化であり、[4] で解説されている方法である。まず、2 国同盟について説明する。例で説明した方が早いので例で説明する。今、$\mathrm{A}_{1,1} = \mathrm{A}_{1,4} = \{1,2\}$ であったとする。このとき、$\mathrm{A}_{1,2}, \mathrm{A}_{1,3}, \mathrm{A}_{1,5}, \mathrm{A}_{1,6}, \mathrm{A}_{1,7}, \mathrm{A}_{1,8}, \mathrm{A}_{1,9}$ から $\{1,2\}$ を取り除くことができる。何故ならば $\mathrm{A}_{1,1} = \mathrm{A}_{1,4} = \{1,2\}$ とすると、$\mathrm{A}_{1,1}, \mathrm{A}_{1,4}$ のどちらかが 1 でもう一方が 2 である。$\mathrm{A}_{1,1} \sim \mathrm{A}_{1,9}$ には 1,2 が 1 回ずつしか出ないため、$\mathrm{A}_{1,2}, \mathrm{A}_{1,3}, \mathrm{A}_{1,5}, \mathrm{A}_{1,6}, \mathrm{A}_{1,7}, \mathrm{A}_{1,8}, \mathrm{A}_{1,9}$ が 1, 2 を含む可能性がなくなるからである。3 国同盟も同様である。この場合は 3 つのマスが $\{1,2\}$, $\{2,3\}$, $\{1,3\}$ の場合にも使えるので 2 国同盟よりも気づきにくい。

n 国同盟は、これを n マスに拡張したものである。具体的には、次のように述べることができる。$j_1, \cdots, j_n$ を 1 〜 9 までの数字から取った n (< 9) 個の数字の組だとするとき、

$$\mathrm{A}_{i,j_1} \cup \cdots \cup \mathrm{A}_{i,j_n} = \{k_1, \cdots, k_n\} \text{ ならば、} \mathrm{A}_{i,r} \ (r \neq j_1, \cdots, j_n) \text{ から } k_1, \cdots, k_n \text{ を削除できる。} \quad (1)$$

上は行に対する n 国同盟を述べているが、もちろん、各列、各サブブロックにも同じことが言える。

4.2　基本テクニック 2 — AB 簡約

これは、一般に X-Wing や Swordfish と呼ばれているテクニックの一般化である。X-Wing について例を用いて説明する。今、$1 \notin \mathrm{A}_{1,j}$ $(j = 2,3,5,6,7,8,9)$ かつ $1 \notin \mathrm{A}_{2,j}$ $(j = 2,3,5,6,7,8,9)$ が成り立っているとすると、$\mathrm{A}_{i,1}$ $(i = 3, \cdots, 9)$ と $\mathrm{A}_{i,4}$ $(i = 3, \cdots, 9)$ から 1 を除くことができる。これは次のような理由からである。与えられた条件より $\mathrm{A}_{1,1}, \mathrm{A}_{1,4}$ のうち 1 つが 1 であり、$\mathrm{A}_{2,1}, \mathrm{A}_{2,4}$ のうち 1 つが 1 である。よって、$\mathrm{A}_{1,1}, \mathrm{A}_{1,4}, \mathrm{A}_{2,1}, \mathrm{A}_{2,4}$ のうち、2 つが 1 である。$\mathrm{A}_{i,1}$ $(i = 1, \cdots, 9)$ と $\mathrm{A}_{i,4}$ $(i = 1, \cdots, 9)$ のうち、1 であるものは 2 つであり、$\mathrm{A}_{1,1}, \mathrm{A}_{1,4}, \mathrm{A}_{2,1}, \mathrm{A}_{2,4}$ のいずれかなので、$\mathrm{A}_{i,1}$ $(i = 3, \cdots, 9)$, $\mathrm{A}_{i,4}$ $(i = 3, \cdots, 9)$ は 1 ではない。これを一般化すると、次のようになる。

$$\text{領域 } \mathrm{A}, \mathrm{B} \text{ が同じ個数の } k \text{ を含むならば } \ k \notin \mathrm{A} \cap \mathrm{B}^C \Rightarrow k \notin \mathrm{A}^C \cap \mathrm{B} \quad (2)$$

先の例は、$k = 1$, $\mathrm{A} = \mathrm{A}_{1,1} \cup \cdots \cup \mathrm{A}_{1,9} \cup \mathrm{A}_{2,1} \cup \cdots \cup \mathrm{A}_{2,9}$, $\mathrm{B} = \mathrm{A}_{1,1} \cup \cdots \cup \mathrm{A}_{9,1} \cup \mathrm{A}_{1,4} \cup \cdots \cup \mathrm{A}_{2,4}$ としたものである。ちなみに 領域 A, B として、それぞれ 3 つの行と 3 つ列を取ったものが、Swordfish と呼ばれているテクニックである。また、領域の取り方としては、行、列の他にもサブブロックを取ることができる。

4.3 基本テクニック 3 — チェーン

これは、一般に Simple Chain や XY Chain と呼ばれているテクニックの一般化である。これらのチェーン系のテクニックは上級のテクニックとみなされており、他のテクニックが使えない場合も有効なことが多いが、考え方は仮置きに近い。

例を挙げる。$A_{1,1} = \{1,2,3\}$, $A_{1,4} = \{1,2\}$, $A_{4,4} = \{1,2\}$, $A_{4,2} = \{1,2\}$, $A_{3,2} = \{1,2\}$ とすると、$A_{1,1}$ から 1, 2 を除くことができる（すなわち、$A_{1,1} = \{3\}$）。何故ならば、今、仮に $A_{1,1} = \{1\}$ と仮定すると、$A_{1,4} = \{2\}$ である（第 1 行に 1 は 1 つだけ）。以下、同様に $A_{4,4} = \{1\}$, $A_{4,2} = \{2\}$, $A_{3,2} = \{1\}$ となり、$A_{1,1}$ のサブブロックに 1 が 2 つある（$A_{1,1} = A_{3,2} = \{1\}$）ことになってしまう。$A_{1,1} = \{2\}$ と仮定しても同様である。この例での

$$A_{1,1} \to A_{1,4} \to A_{4,4} \to A_{4,2} \to A_{3,2} \to A_{1,1}$$

のつながりをチェーンと呼ぶが、実際のプログラムでは、再帰的にこのようなチェーンを探し、矛盾が生じないかをチェックしている。チェーンの検索は、同じ行、同じ列、同じサブブロックにある同じ数字、もしくは同じマスの別の数字の候補に対して再帰呼び出しを用いて行なっているが、無制限に検索すると時間がかかりすぎるため、再帰呼び出しの深さに制限を設けている（今の所は 10 に設定）。

5 実験

前節の解法アルゴリズムを実行するプログラムを *Mathematica* で実装した。数独の問題を入力しやすいように図 1 にあるような問題入力用のインターフェイスを備えている。

出版されている数独パズルに関する書籍のうち、難しい問題が掲載されていると思われる書籍 [5] - [9] を選び、このプログラムで解答させた所、基本テクニック 1 〜 3 の範囲で問題が解け、仮置きを必要とするものはなかった。もちろん、全てのパズルが仮置きなしで解けるわけではなく、図 2 の「世界で最も難しいと言われているパズル」をはじめ、仮置きを必要とするパズルもたくさんある。特に海外のサイトにあるパズルには難しいものが多いようである。例えば [10] には、本稿のプログラムでは仮置きを必要とするパズルがたくさんある。

6 結論

人間に近い方法で数独パズルを解くプログラムを作成し、実際の問題を解いた。そのプログラムでは、まず 3 つの基本テクニックでパズルを解き、それでは解けない場合には仮置きを用いる。出版されている書籍にある問題は仮置きなしで解くことができたが、インターネットのサイト（特に海外のもの）には、仮置きを必要とするものが多く存在する。今回のプログラムでは、問題を解く際の各テクニックの利用状況がわかるので、次のような応用が考えられる。

- 与えられた問題の難易度判定
- 与えられた問題の解法の解説作成
- 与えられた問題の面白さの判定

今後は、これらの応用とともに、問題作成などについても取り組んでいきたい。

図 1: 問題入力用のインターフェイス

		5	3					
8							2	
	7			1		5		
4					5	3		
	1			7				6
		3	2				8	
	6		5					9
		4					3	
					9	7		

図 2: 世界で最も難しいと言われているパズル

参 考 文 献

[1] 井上秀太郎, 佐藤洋祐, 鈴木晃, 鍋島克輔: ブーリアン・グレブナ基底を使った数独の解法, 数理解析研究所講究録, **1666**, pp. 1-5, 2009.

[2] 井上秀太郎, 佐藤洋祐: グレブナー基底を使った数独の難易度判定と問題作成, 数理解析研究所講究録, **1785**, pp. 51-56, 2012.

[3] 中山洋将: Asir でのマルコフ基底計算とマルコフ基底によるパズルの解の列挙, *Risa/Asir Conference 2012*, 神戸, 2012.

[4] J. F. Crook: A Pencil-and-Paper Algorithm for Solving Sudoku Puzzles, *Notices of the American Mathematical Society*, **56**(4), pp. 460-468, 2009.

[5] 西尾徹也: ナンプレ超上級編 29, 世界文化社, 2012.

[6] ニコリ: ポケット数独 11 上級篇, ソフトバンククリエイティブ, 2012.

[7] たかせあきひこ: 難問ナンプレ 252 題 vol.15, 白夜ムック, 2012.

[8] ナンプレ研究会: 究極難解ナンプレ 8, 晋遊舎, 2013.

[9] 川崎光徳: 超難問ナンプレ 130 選, 永岡書店, 2012.

[10] http://www.sudokuwiki.org/sudoku.htm

チュートリアル 1

Tutorial 1

Sparse interpolation and signal processing

Annie Cuyt and Wen-shin Lee*

Department of Mathematics and Computer Science
University of Antwerp, Belgium

Conventional interpolation algorithms do not take sparsity into consideration and depend on the total degree or the maximum possible size of the function. Traditionally, polynomial interpolation of n values f_j at points x_j is a technique that determines the coefficients $a_i, i = 1, \ldots, n$ in the model $a_1\phi_1(x) + \cdots + a_n\phi_n(x)$ from the conditions

$$\sum_{i=1}^{n} a_i \phi_i(x_j) = f_j, \quad j = 1, \ldots, n,$$

where the functions $\phi_1(x), \ldots, \phi_n(x)$ satisfy the Haar condition. Several numerical techniques to determine the values a_i, for use with different $\phi_i(x)$, are well-known. The numerical conditioning of the problem and the stability of the algorithms have been analyzed in great detail.

On the other hand, sparse interpolation algorithms are sensitive to the number of nonzero terms in the underlying representation and thus account for the sparsity of the function. In computer algebra, the problem of interpolating a sparse polynomial has always been a major research focus. The purpose is to improve computational performance: sparse interpolation and representation algorithms are developed to control the intermediate swell encountered in symbolic computation.

In 1979, Zippel gave the first sparse polynomial interpolation algorithm [22]. Then in 1988, Ben-Or and Tiwari presented a different algorithm [2] that is based on the Berlekamp/Massey algorithm [15] from coding theory. The Ben-Or/Tiwari sparse interpolation algorithm can determine both the correct indices k_i and the coefficients a_i, for $i = 1, \ldots, m$, in the model $a_1 x^{k_1} + \ldots + a_m x^{k_m}$, with $k_1 < \ldots < k_m$, from the $2m$ conditions

$$\sum_{i=1}^{m} a_i x_j^{k_i} = f_j, \quad j = 1, \ldots, 2m.$$

Besides the monomial basis x^{i-1}, the problem of interpolating

$$\sum_{i=1}^{m} a_i \phi_{k_i}(x_j) = f_j, \quad j = 1, \ldots, 2m,$$

*wen-shin.lee@ua.ac.be

from $2m$ evaluations is also solved for certain sequences of functions $\phi_i(x)$, including the Chebyshev polynomials $T_{i-1}(x)$, the Pochhammer symbols $(x)_{i-1}$ [13] and some multivariate generalizations of these [2]. In addition, a probabilistic strategy called "early termination" is developed to detect the number of nonzero terms (being m) when it is not supplied in the input [12]. Sparse techniques solve the interpolation problem from a number of samples f_j proportional to the number of terms in the representation (being m) rather than the number of available generating elements (being k_m). In floating point arithmetic, the connection between Prony's method [18] and error-correcting codes has led to the development of symbolic-numeric sparse polynomial interpolation [9], which exploits a generalized eigenvalue reformulation [11, 10] and a link to Rutishauser's qd-algorithm [5]. This connection further enables a generalization of variants of Prony to other basis functions [8].

Closely related to Padé approximation, the classical method of Prony has found applications in the shape from moments problem [16], spectral analysis [14], and lately sparse sampling of signals with finite rate of innovation [21], etc. The modern least squares approaches [20, 19] of exponential modeling have evolved quite significantly from Prony's original version. Still, it is well-known that in general such inverse problem can be both ill-posed and ill-conditioned.

Interestingly, techniques from symbolic-numeric sparse interpolation can be used to tackle these numerical issues. New sparse interpolation algorithms are thus developed by drawing from various disciplines such as numerical linear algebra, computer algebra and numerical approximation theory. The new method is efficient, and the technique is generalized for functions $\phi_k(x)$ where the parameter k can vary continuously [6].

In signal processing, sparsity has recently emerged as an important concept [3, 7, 4]. Sparse signals admit a representation by a linear combination of only a few elementary waveforms or atoms. Currently, the acquisition and reconstruction of such signals receives a great deal of attention. The ultimate goal is to determine the underlying sparse representation directly from as few data samples as possible. In many applications, such technique offers a promising alternative to the standardized Fourier transform. Moreover, the fact that signals can be reconstructed from undersampled data opens up a whole new range of possibilities. In this talk, we discuss the use of interpolation methods in this setting. We depart from sparse polynomial interpolation in the field of computer algebra and explain an interesting connection to Prony's method, as well as some of its variants. We present some new signal processing methods and discuss the corresponding issues in linear algebra and approximation theory. Selected applications will be presented (e.g. [17, 1]).

References

[1] `www.sparsit.com`.

[2] M. Ben-Or and P. Tiwari. A deterministic algorithm for sparse multivariate polynomial interpolation. In *Proc. Twentieth*, pages 301–309, New York, N.Y., 1988. ACM Press.

[3] E. Candès, J. Romberg, and T. Tao. Robust uncertainty principles: exact signal reconstruction from highly incomplete information. *IEEE Trans. on Information Theory*, 52(2):489–509, 2006.

[4] E. Candès and M. B. Wakin. An introduction to compressive sampling. *IEEE Signal Pro. Magazine*, pages 21–30, March 2008.

[5] A. Cuyt and W.-s. Lee. A new algorithm for sparse interpolation of multivariate polynomials. *Theoretical Computer Science*, 409(2):180–185, 2008.

[6] A. Cuyt, W.-s. Lee, and S. Peelman. Sparse trigonometric interpolation, 2013. In preparation.

[7] D. Donoho. Compressed sensing. *IEEE Trans. on Information Theory*, 52(4):1289–1306, 2006.

[8] M. Giesbrecht, G. Labahn, and W.-s. Lee. Symbolic-numeric sparse polynomial interpolation in Chebyshev basis and trigonometric interpolation. In V. G. Ganzha, E. W. Mayr, and E. V. Vorozhtsov, editors, *CASC 2004 Proc. 7th Internat. Workshop on Computer Algebra in Scientific Computing*, pages 195–205. TUM Press, 2004.

[9] M. Giesbrecht, G. Labahn, and W.-s. Lee. Symbolic-numeric sparse interpolation of multivariate polynomials. *J. Symbolic Comput.*, 44(8):943–959, 2009.

[10] G. H. Golub, P. Milanfar, and J. Varah. A stable numerical method for inverting shape from moments. *SIAM J. Sci. Comput.*, 21(4):1222–1243, 1999.

[11] Y. Hua and T. K. Sarkar. Matrix pencil method for estimating parameters of exponentially damped/undamped sinusoids in noise. *IEEE Trans. Acoustics, Speech, and Signal Processing*, 38(5):814–824, 1990.

[12] E. Kaltofen and W.-s. Lee. Early Termination in Sparse Interpolation Algorithms. *Journal of Symbolic Computation*, 36(3-4):365–400, 2003.

[13] Y. N. Lakshman and B. D. Saunders. Sparse polynomial interpolation in non-standard bases. *SIAM Journal on Computing*, 24(2):387–397, 1995.

[14] S. L. Marple, Jr. *Digital Spectral Analysis*. Prentice-Hall, Englewood Cliffs, N.J., 1987.

[15] J. L. Massey. Shift-register synthesis and BCH decoding. *IEEE Trans. Inf. Theory*, IT-15:122–127, 1969.

[16] P. Milanfar, G. Verghese, C. Karl, and A. Willsky. Reconstructing polygons from moments with connections to array processing. *IEEE Transactions on Signal Processing*, 43:432–443, 1995.

[17] S. Peelman, J. Ver der Herten, M. De Vos, W.-s. Lee, S. Van Huffel, and A. Cuyt. Sparse reconstruction of correlated multichannel activity. In *Proc. of the 35th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (IEEE EMBC 2013)*, 2013. To appear.

[18] Baron de Prony, Gaspard-Clair-François-Marie Riche. Essai expérimental et analytique sur les lois de la Dilatabilité des fluides élastique et sur celles de la Force expansive de la vapeur de l'eau et de la vapeur de l'alkool, à différentes températures. *J. de l'École Polytechnique*, 1:24–76, 1795.

[19] R. Roy and T. Kailath. ESPRIT-Estimation of signal parameters via rotational invariance techniques. *IEEE Trans. Acoustics, Speech, and Signal Processing*, 37(7):984–995, 1989.

[20] R. O. Schmidt. Multiple emitter location and signal parameter estimation. *IEEE Transactions on Antennas and Propagation*, AP-34(3):276–280, 1986.

[21] M. Vetterli, P. Marziliano, and T. Blu. Sampling signals with finite rate of innovation. *IEEE Trans. on Signal Processing*, 50(6):1417–1428, 2002.

[22] R. Zippel. Probabilistic algorithms for sparse polynomials. In *Proc. EUROSAM '79*, volume 72, pages 216–226, 1979.

チュートリアル２

Tutorial 2

Invited paper for the Developments in Computer Algebra Research and Collaboration with Industry / DCAR 2013, Institute of Mathematics for Industry, Kyushu University, August 21-23, 2013

Engine Control System Development and Symbolic Manipulation - Application and Challenges in Modelling -

Hisahiro Ito

TOYOTA MOTOR CORPORATION
1200, Mishuku, Susono, Shizuoka 410-1193 JAPAN
hisahiro_ito@mail.toyota.co.jp

Abstract

The integrated use of symbolic and numeric technologies is presented through application examples in automotive industry. The effectiveness of such an integrated approach is emphasized by showing concrete uses of symbolic manipulation and numerical computation in Maple [1]. In the first of the two examples, the chemical equilibrium of combustion phenomena which is a constrained optimization problem is analysed symbolically and turned into an unconstrained optimization problem. As opposed to numeric-only approach, target equations for the minimization problem are explicitly shown, which facilitates the understanding of the problem. In the second example, a simple phenomenological model for the heat release process of an internal combustion engine is introduced where a differential equation containing a conditional branch is symbolically analysed and then numerically integrated to obtain some key physical quantities of the combustion dynamics. Finally, the paper is concluded with some remarks on challenges that modelling technologies in automotive industry are facing.

Introduction

Today, there are about 800 million vehicles including passenger cars, trucks and buses worldwide and the number is still growing rapidly, particularly in emerging countries such as China and India [2]. While the new types of power sources for a vehicle, notably hybrid electric, are increasingly becoming popular in some countries, the vast majority of new vehicles produced and sold today come with an internal combustion engine only, and this trend may be attributed to various facts such as 1) the balance between the income level of individual households and the price range of vehicles, or 2) the car ownership rate in each country and people's appetite for buying expensive goods including a car. Because of the constant increase in the number of engines operating at every corner of the cities around the world, the amount of exhaust emissions from vehicles is also increasing, and thus regulations on emissions are under constant revisions to get more stringent.

On this backdrop, the control system of an engine of a vehicle has grown in its complexity with more actuators and sensors, and/or more precise control than ever before. This trend has necessitated the front-loading in a development process, and the use of models (generally called "plant models") of engines and other vehicle components such as transmission or battery at an early stage of a development process (commonly called "Model-Based Development" or acronymed MBD), has become the norm in automotive industry. Although the hitherto effort for the front-loaded development is enabling ever more efficient development, still more effort is required as the complexity of the control system keeps growing.

In order to meet increasingly demanding requirements on plant models, new technologies for modelling and simulation of a physical system are gaining a foothold (for example Modelica [3], VHDL-AMS [4] and Simscape [5]). These technologies provide important features like automatic physical equation generation for component connections and DAE solvers for a hybrid system, and are playing a key role in the system level simulation. However, capturing the physics behind a system in question is out of scope of these technologies, i.e., the authors of physical component models must take responsibility of the equations that they derive and define. Today, this equation-derivation process for a physical component is considered a task with paper and a pen or some word-processing software, but the productivity of this process needs to increase so that essential physical phenomena can be modelled in a timely manner and used efficiently as part of a system for fully fledged system level simulations. As an attempt to fill this gap, in this article the author would like to propose to use symbolic manipulation technology together with numerical computation technology and show how the equational analysis process can be streamlined.

This paper is organized as follows. First, the combustion chemistry is analysed symbolically so that the problem can be presented as an unconstrained optimization problem. Second, the phenomenological heat release process in an internal combustion engine is analysed and numerically integrated. Through these two cases, the potential benefit of the integrated use of symbolic and numeric approaches will be exposed. Finally, the paper is concluded with some brief remarks on other issues not dealt in the paper.

This paper itself is a Maple worksheet where the symbolic and numeric operations appearing below are computationally processed. The version of Maple being used is as follows.

Standard Worksheet Interface, Maple 17.00, Windows 7, April 10 2013 Build ID 827314 **(1)**

Modelling Example 1 - Combustion Chemistry & Chemical Equilibrium

Background

In an internal combustion engine of a vehicle, combustion is repeatedly occurring several thousand times at every minute. For example, suppose an engine is running at 1000 or 6000 revolutions per minute (RPM), then a single revolution of an engine crank takes just 60 or 10 milliseconds, respectively. A combustion of the mixture of air and fuel vapour (and, strictly speaking, some residual burned gas too) is controlled in terms of the air-to-fuel ratio and the combustion start timing. At every combustion phenomenon, fuel vapour of about 10 to 20 milligrams is burned to generate torque which drives the wheels. At the same time, the water and carbon dioxide, together with other various kinds of molecules depending on operating conditions, are produced.

In the case of combustion at stoichiometry where both the air and the fuel are completely consumed, the chemical reaction formula is described as follows [6].

$$C_8H_{18} + 12.5\left(O_2 + 3.773\,N_2\right) = 9\,H_2O + 8\,CO_2 + 47.16\,N_2$$

where the fuel is assumed octane. In general, however, the combustion reaction in an internal combustion engine can never occur at stoichiometry because of various kinds of disturbances such as variances in fuel injection amount, thermal exchange with cylinder wall or piston action and so on. Considering these effects, the reaction may be generally represented as follows [8].

$$\frac{\phi}{a + \frac{b}{4} - \frac{c}{2}}\,C_aH_bO_c + O_2 + 3.773\,N_2 = x_1H_2O + x_2CO_2 + x_3N_2 + \ldots$$

The last term on the R.H.S. of the above equation (...) indicates species such as *NOx, CO, OH, H* etc. whose production depends on operating conditions, and ϕ is the fuel-to-air equivalence ratio. Regardless of the product types, the element mass involved in the reaction must conserve.

$$\sum_{j=1}^{n}\left(A_{ij}\,n_j - \beta_i\right) = 0 \qquad \textbf{(2)}$$

where A_{ij} are the stoichiometric coefficients, n_j is the number of moles of product species *j*, β_i is the number of moles of element $i = C, H, O, N$ for 1 mole of oxygen in the air. Specifically, β_i for each element can be defined based on the above general reaction formula as follows.

$$\beta_C = \frac{a\,\phi}{a + \frac{1}{4}\,b - \frac{1}{2}\,c},\ \beta_H = \frac{b\,\phi}{a + \frac{1}{4}\,b - \frac{1}{2}\,c},\ \beta_O = 2 + \frac{c\,\phi}{a + \frac{1}{4}\,b - \frac{1}{2}\,c},\ \beta_N = 7.546 \qquad \textbf{(3)}$$

According to [6], "it is a good approximation for performance estimates in engines to regard the burned gases produced by the combustion of fuel and air as in chemical equilibrium" excluding the late expansion stroke and during the exhaust process. Chemical equilibrium is achieved when the Gibbs free energy of a mixture

$$g = \sum_{j=1}^{n} g_{mole_j}\,n_j \qquad \textbf{(4)}$$

is minimized where g_{mole_j} denotes the Gibbs free energy (or equivalently the chemical potential) of species *j* per mole. For gases, the chemical potential g_{mole_j} is

$$g_{mole_j}\left(x_j, p, T\right) = g_{p,mole_j}(T) + R_{univ}\,T\ln\left(x_j\right) + R_{univ}\,T\ln\left(\frac{p}{p0}\right) \qquad \textbf{(5)}$$

where $g_{p,mole_j}$ is the chemical potential in the standard state, x_j is the mole fraction of species *j*, *p* is the pressure of the gas, *T* is the temperature of the gas, and *p0* is the reference pressure. The mole fraction x_j is defined as follows.

$$x_j = \frac{n_j}{n_{tot}} \qquad \textbf{(6)}$$

where n_{tot} is the sum of the number of moles of all product species.

Several methods to minimize the Gibbs free energy *g* were proposed in literatures [6], [7], [8]. One way in [6] is to use the method of Lagrange multipliers λ_i and the equilibrium conditions are given as

$$g_{mole_j}\left(x_j, p, T\right) + \sum_{i=1}^{l} \lambda_i A_{i,j} = 0 \tag{7}$$

So far, the explanations were kept as general as possible. Now, concrete calculations are going to be performed step by step so that this problem can better be exposed for a specific case.

Example

Definitions of Basic Quantities

First, the fuel is assumed to be iso-octane C_8H_{18}

$$[\, a = 8,\, b = 18,\, c = 0\,] \tag{8}$$

and the following product species are assumed

$$[\, H2O,\, CO2,\, N2,\, O2,\, CO,\, NO,\, OH,\, H,\, H2,\, \mathrm{O}\,] \tag{9}$$

The elements included in the reaction are as follows.

$$[\, C,\, H,\, \mathrm{O},\, N\,] \tag{10}$$

Then A_{ij}, n_j and β_i are determined.

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 2 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 2 & 0 \\ 1 & 2 & 0 & 2 & 1 & 1 & 1 & 0 & 0 & 1 \\ 0 & 0 & 2 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{11}$$

$$n^T = \begin{bmatrix} n_{H2O} & n_{CO2} & n_{N2} & n_{O2} & n_{CO} & n_{NO} & n_{OH} & n_H & n_{H2} & n_{\mathrm{O}} \end{bmatrix} \tag{12}$$

$$\beta^T = \begin{bmatrix} \frac{16}{25}\,\phi & \frac{36}{25}\,\phi & 2 & 7.546 \end{bmatrix} \tag{13}$$

Note that the fuel/air equivalence ratio ϕ is kept as symbol for β_i as it varies in every combustion cycle.

Element Conservations

Now the element conservations, Eq.**(2)**, can be written in a concrete manner.

$$\begin{gathered} n_{CO2} + n_{CO} - \frac{16}{25}\,\phi = 0 \\ 2\,n_{H2O} + n_{OH} + n_H + 2\,n_{H2} - \frac{36}{25}\,\phi = 0 \\ n_{H2O} + 2\,n_{CO2} + 2\,n_{O2} + n_{CO} + n_{NO} + n_{OH} + n_{\mathrm{O}} - 2 = 0 \\ 2\,n_{N2} + n_{NO} - 7.546 = 0 \end{gathered} \tag{14}$$

where the fuel-to-air equivalence ratio ϕ is a parameter that is determined at every combustion cycle through fuel injection.

Note that understanding the physical meanings of the above equations is facilitated thanks to physically meaningful subscripts such as *CO2* or *OH* etc. instead of integers while the use of the symbolic manipulation software package maintains the possibility to apply manipulations on these equations efficiently.

Chemical Potential & Lagrange Multipliers for Minimization

Similar to the mass conservation equations, Eq.**(7)** can also be made specific for the current case.

First, all the subscripts are made specific. Here, only three equations out of ten are shown to save the space.

$$\begin{gathered} g_{mole_{H2O}}\left(x_{H2O}, p, T\right) + \lambda_1 A_{1,1} + \lambda_2 A_{2,1} + \lambda_3 A_{3,1} + \lambda_4 A_{4,1} = 0 \\ g_{mole_{CO2}}\left(x_{CO2}, p, T\right) + \lambda_1 A_{1,2} + \lambda_2 A_{2,2} + \lambda_3 A_{3,2} + \lambda_4 A_{4,2} = 0 \\ g_{mole_{N2}}\left(x_{N2}, p, T\right) + \lambda_1 A_{1,3} + \lambda_2 A_{2,3} + \lambda_3 A_{3,3} + \lambda_4 A_{4,3} = 0 \end{gathered} \tag{15}$$

Then the matrix A_{ij} is substituted, and the subscripts for λ_i are replaced from integers to element symbols.

$$
\begin{aligned}
g_{mole_{H2O}}(x_{H2O}, p, T) + 2\lambda_H + \lambda_O &= 0 \\
g_{mole_{CO2}}(x_{CO2}, p, T) + \lambda_C + 2\lambda_O &= 0 \\
g_{mole_{N2}}(x_{N2}, p, T) + 2\lambda_N &= 0 \\
g_{mole_{O2}}(x_{O2}, p, T) + 2\lambda_O &= 0 \\
g_{mole_{CO}}(x_{CO}, p, T) + \lambda_C + \lambda_O &= 0 \\
g_{mole_{NO}}(x_{NO}, p, T) + \lambda_O + \lambda_N &= 0 \\
g_{mole_{OH}}(x_{OH}, p, T) + \lambda_H + \lambda_O &= 0 \\
g_{mole_{H}}(x_{H}, p, T) + \lambda_H &= 0 \\
g_{mole_{H2}}(x_{H2}, p, T) + 2\lambda_H &= 0 \\
g_{mole_{O}}(x_{O}, p, T) + \lambda_O &= 0
\end{aligned}
\qquad \textbf{(16)}
$$

These equations are part of the target equations for minimization.

Now the first term on the L.H.S. of Eq.**(16)**, which is the Gibbs free energy of each species defined in Eq.**(5)**, needs to be computed. In Eq.**(5)**, the first term on the R.H.S. is the chemical potential in the standard state $g_{p,mole_j}(T)$ and can be computed as follows. First, the chemical potential is related to other thermodynamic quantities as

$$g_{p,mole_j}(T) = h_{mole_j}(T) - T s_{p,mole_j}(T) \qquad \textbf{(17)}$$

where h_{mole_j} and $s_{p,mole_j}$ are the enthalpy and the entropy of species *j* per mole in the standard state, respectively. These quantities can be computed from the following polynomials.

$$
\begin{aligned}
\frac{h_{mole_j}(T)}{R_{univ} T} &= a_{1_j} + \frac{1}{2} a_{2_j} T + \frac{1}{3} a_{3_j} T^2 + \frac{1}{4} a_{4_j} T^3 + \frac{1}{5} a_{5_j} T^4 \\
\frac{s_{p,mole_j}(T)}{R_{univ}} &= a_{1_j} \ln(T) + a_{2_j} T + \frac{1}{2} a_{3_j} T^2 + \frac{1}{3} a_{4_j} T^3 + \frac{1}{4} a_{5_j} T^4 + a_{7_j}
\end{aligned}
\qquad \textbf{(18)}
$$

where R_{univ} = 8.3143 (J/mol/K) is the universal gas constant, and the coefficients on the R.H.S. a_{k_j} (k = 1 ..7) are known values for various kinds of molecules and publicly available as a data table released as GRI-Mech [9]. The polynomials for enthalpy and entropy were derived from the one for specific heat $c_{p,mole_j}$, and these are called the NASA polynomials [8].

$$\frac{c_{p,mole_j}(T)}{R_{univ}} = T^4 a_{5_j} + T^3 a_{4_j} + T^2 a_{3_j} + T a_{2_j} + a_{1_j} \qquad \textbf{(19)}$$

The values of the coefficients for each species consist of two different groups below and above a threshold temperature. For example, the values for *NO* are

$$
\begin{aligned}
&T1 = 1000.0,\ a_{hi_1} = 3.2606056,\ a_{hi_2} = 0.0011911043,\ a_{hi_3} = -4.2917048\ 10^{-7},\ a_{hi_4} = 6.9457669\ 10^{-11},\ a_{hi_5} = \\
&\quad -4.0336099\ 10^{-15},\ a_{hi_6} = 9920.9746,\ a_{hi_7} = 6.3693027,\ a_{low_1} = 4.2184763,\ a_{low_2} = -0.0046389760,\ a_{low_3} \\
&\quad = 0.000011041022,\ a_{low_4} = -9.3361354\ 10^{-9},\ a_{low_5} = 2.8035770\ 10^{-12},\ a_{low_6} = 9844.6230,\ a_{low_7} = 2.2808464
\end{aligned}
\qquad \textbf{(20)}
$$

where $T1$ is the threshold temperature in Kelvin, a_{hi_k} and a_{low_k} (k = 1 ..7) are the values above and below that threshold, respectively.

Now all the necessary equations were presented. Eqs.**(17)** and **(5)** are going to be computed for each species.

As an example, the Gibbs free energy of *H2O* in the standard state looks as follows. Below the threshold temperature of 1000 K,

$$g_{p,mole_{H2O}}(T) = 8.3143\,T\,(5.047672768 + 0.001018217050\,T - 0.000001086733685\,T^2 + 4.57330885\,10^{-10}\,T^3 - 8.85989085\,10^{-14}\,T^4 - 4.19864056\ln(T)) \quad \textbf{(21)}$$

Above the threshold temperature,

$$g_{p,mole_{H2O}}(T) = 8.3143\,T\,(-1.932777610 - 0.001088459020\,T + 2.734541967\,10^{-8}\,T^2 + 8.08683225\,10^{-12}\,T^3 - 8.41004960\,10^{-16}\,T^4 - 3.03399249\ln(T)) \quad \textbf{(22)}$$

The complete Gibbs free energy of *H2O* as a function of the mole fraction, pressure and temperature is

$$g_{mole_{H2O}}(x_{H2O}, p, T) = g_{p,mole_{H2O}}(T) + 8.3143\,T\ln(x_{H2O}) + 8.3143\,T\ln(0.000009869232667\,p) \quad \textbf{(23)}$$

Similar calculations can be performed for the other species. By substituting Eq.**(23)** and similar equations for the other species into Eqs.**(16)**, ten equations with 14 unknowns (ten mole fractions x_j and four Lagrange multipliers λ_i) and 2 parameters (i.e., pressure p and temperature T) are obtained. Four more equations are mass conservation equations Eq.**(14)**.

The mole fraction x_j and the number of moles n_j are related through Eq.**(6)** where the total number of moles n_{tot} can be specifically described for the current case as

$$n_{tot} = n_{H2O} + n_{CO2} + n_{N2} + n_{O2} + n_{CO} + n_{NO} + n_{OH} + n_H + n_{H2} + n_O \quad \textbf{(24)}$$

Unconstrained Optimization Problem

The equations **(14)**, **(16)**, **(24)**, together with **(21)**, **(22)**, **(23)** and similar equations for the other species, constitute the set of equations to solve. Note that some operating conditions need to be specified. Specifically, in Eq.**(23)**, pressure p and temperature T must be specified. In Eq.**(14)**, equivalence ratio ϕ needs to be specified. The target variables for the minimization are n_j. The mole fractions x_j are basically the same as n_j except that they are normalized by n_{tot}.

In the spirit of seeing is believing, let us see the final set of equations for minimization by specifying all the remaining parameter values. This can be very easily performed here because this article is in fact a Maple worksheet where all the manipulations and calculations so far were carried out in Maple.

When the conditions are

$$p = 2.026500\,10^5,\ T = 2000,\ \phi = 1.0 \quad \textbf{(25)}$$

the equations are

$$-4.376150190\,10^5 + 16628.6000\ln(x_{H2O}) + 2.\lambda_H + \lambda_O = 0.$$
$$-5.036846888\,10^5 + 16628.6000\ln(x_{CO2}) + \lambda_C + 2.\lambda_O = 0.$$
$$-4.286312600\,10^5 + 16628.6000\ln(x_{N2}) + 2.\lambda_N = 0.$$
$$-4.577501428\,10^5 + 16628.6000\ln(x_{O2}) + 2.\lambda_O = 0.$$
$$-4.420203868\,10^5 + 16628.6000\ln(x_{CO}) + \lambda_C + \lambda_O = 0.$$
$$-4.680479299\,10^5 + 16628.6000\ln(x_{NO}) + \lambda_O + \lambda_N = 0.$$
$$-4.121053883\,10^5 + 16628.6000\ln(x_{OH}) + \lambda_H + \lambda_O = 0.$$
$$-2.554556388\,10^5 + 16628.6000\ln(x_H) + \lambda_H = 0.$$
$$-3.044823405\,10^5 + 16628.6000\ln(x_{H2}) + 2.\lambda_H = 0.$$
$$-3.490007755\,10^5 + 16628.6000\ln(x_O) + \lambda_O = 0.$$
$$n_{CO2} + n_{CO} - 0.6400000000 = 0.$$
$$2.n_{H2O} + n_{OH} + n_H + 2.n_{H2} - 1.440000000 = 0.$$
$$n_{H2O} + 2.n_{CO2} + 2.n_{O2} + n_{CO} + n_{NO} + n_{OH} + n_O - 2. = 0.$$
$$2.n_{N2} + n_{NO} - 7.546 = 0. \quad \textbf{(26)}$$

When the conditions are

$$p = 1.0132500\,10^6,\ T = 2500,\ \phi = 0.5 \tag{27}$$

the equations are

$$\begin{aligned}
-5.367924725\,10^5 + 20785.7500\ln\left(x_{H2O}\right) + 2.\lambda_H + \lambda_O &= 0.\\
-6.255020045\,10^5 + 20785.7500\ln\left(x_{CO2}\right) + \lambda_C + 2.\lambda_O &= 0.\\
-5.203828734\,10^5 + 20785.7500\ln\left(x_{N2}\right) + 2.\lambda_N &= 0.\\
-5.580052376\,10^5 + 20785.7500\ln\left(x_{O2}\right) + 2.\lambda_O &= 0.\\
-5.371483275\,10^5 + 20785.7500\ln\left(x_{CO}\right) + \lambda_C + \lambda_O &= 0.\\
-5.703948517\,10^5 + 20785.7500\ln\left(x_{NO}\right) + \lambda_O + \lambda_N &= 0.\\
-4.989796192\,10^5 + 20785.7500\ln\left(x_{OH}\right) + \lambda_H + \lambda_O &= 0.\\
-2.974616894\,10^5 + 20785.7500\ln\left(x_H\right) + \lambda_H &= 0.\\
-3.643732573\,10^5 + 20785.7500\ln\left(x_{H2}\right) + 2.\lambda_H &= 0.\\
-4.144924437\,10^5 + 20785.7500\ln\left(x_O\right) + \lambda_O &= 0.\\
n_{CO2} + n_{CO} - 0.3200000000 &= 0.\\
2.n_{H2O} + n_{OH} + n_H + 2.n_{H2} - 0.7200000000 &= 0.\\
n_{H2O} + 2.n_{CO2} + 2.n_{O2} + n_{CO} + n_{NO} + n_{OH} + n_O - 2. &= 0.\\
2.n_{N2} + n_{NO} - 7.546 &= 0.
\end{aligned} \tag{28}$$

There already exist many solution methods and implementations for this type of problem, but the real challenge of this problem from the standpoint of the development of an engine control system is not simply to choose an optimization method. Rather, the challenges are 1) the above optimization problem must be solved many times - actual phenomenon are repeated several thousand times per minute - with different initial conditions for the fuel-to-air ratio, pressure and temperature for every combustion cycle, 2) the gas pressure and temperature are affected by, not only the combustion, but also the piston action as well as thermal energy exchange between the gas and the cylinder wall, 3) the combustion phenomena is just one of many dynamics occurring in an engine, all of which must be computed as a system, 4) an engine plant model needs to be connected with an engine controller, which may be a model or a real control unit, to form a closed-loop system, 5) such a closed-loop simulation needs to run at a reasonable speed and 6) more and more precise calculations will be required in future as the regulations on the emissions become tighter and tighter.

Although there are some dedicated simulation packages for combustion chemistry, it is difficult to meet these challenges with them as they usually are not designed to be integrated as part of an entire engine model, which has to be further integrated with (a) controller(s).

With the symbolic manipulation technology, seemingly complicated problems like combustion chemistry may be understood and grappled with efficiently as presented above. Furthermore, it would be very productive if "code" to solve the optimization problem can be directly produced for a target simulation environment from the symbolic-numeric analysis. In such a case, the code may not necessarily be C or other procedural programming languages. Rather, it may be a plant modelling language such as Modelica or Simscape so that the code can be used as part of an entire engine model implemented in that target environment.

Modelling Example 2 - Combustion Dynamics in Internal Combustion Engines

In the previous section, a single combustion phenomenon was examined. However, as mentioned previously, real engine cycle involves several thousand combustion phenomena at every minute. During each cycle, a piston and valves work together to inhale and compress the gas which is then burned and presses down the piston for work and is exhaled from the cylinder. In this section, a phenomenological heat release model for this engine cycle is introduced. As opposed to the previous example where the system was algebraic, the system of this example consists of a differential equation containing a conditional branch, and it will be shown that the symbolic manipulation technology works well for this type of system when it is used in conjunction with numerical computation technology.

First, model parameters are defined, but the detailed explanations for each parameter is omitted and some are going to be explained later as needed. Here, a single-cylinder engine is assumed to be operating at a constant speed of $\omega_{eng} = 40\,\pi$ (rad/s) = 1200 (RPM).

33 *parameters*

$$\left[p_{amb} = 101325,\ l_{bore} = 0.081,\ l_{stroke} = 0.077,\ l_{rod} = 0.122,\ s_{tdc} = 0.008500000000,\ l_{crk} = 0.03850000000,\ A_{cyl} \right. \tag{29}$$

$$= 0.005152997351,\ V_{clear} = 0.00004380047748,\ V_{disp} = 0.0004405812735,\ f_{air} = 5,\ R_{air} = 287,\ Cv_{air} = \frac{1435}{2},$$

$$H_{low} = 4.25\,10^7,\ T0 = 293.15,\ P0 = 101325,\ V0 = 0.0004405812735,\ m_{air} = 0.0002122417620,\ e_{ini} = 44.64189754,$$

$$KL = 0.4,\ AFR = 14.6,\ SA_{deg} = 15,\ RPM = 1200,\ T_{wall} = 373.15,\ k_{therm} = \frac{2}{5},\ SA_{rad} = 2.879793266,\ m_{fuel}$$

$$= 0.00001453710699,\ Q_{\max} = 617.8270471,\ \omega_{eng} = 40\,\pi,\ c_1 = 1.2,\ c_2 = 500,\ c_3 = 0.000001,\ a_{wi} = 6.907755279,\ m_{wi}$$

$$= 3 \left. \right]$$

Piston position z_{piston} as a function of crank angle θ_{rad} can be calculated by

$$z_{piston}(\theta_{rad}) = l_{stroke} - l_{crk}\left(1 - \cos(\theta_{rad})\right) - l_{rod}\left(1 - \sqrt{1 - \frac{\sin(\theta_{rad})^2\, l_{crk}^2}{l_{rod}^2}}\right) \tag{30}$$

where l_{stroke}, l_{rod}, l_{crk} are the lengths of piston stroke, connecting-rod and crank, respectively.

Using Eq.**(30)**, the volume of a cylinder is readily computed as

$$V_{cyl}(\theta_{rad}) = V_{clear} + A_{cyl} z_{piston}(\theta_{rad}) \tag{31}$$

where V_{clear} is the clearance volume and A_{cyl} is the area of a piston head. This will be used when thermodynamic quantities of the gas are calculated.

Thermodynamics of ideal gas in a cylinder

According to thermodynamics, internal energy of ideal gas e_{cyl} is given by

$$e_{cyl}(\theta_{rad}) = \frac{1}{2} f_{air} p_{cyl}(\theta_{rad})\, V_{cyl}(\theta_{rad}) \tag{32}$$

where f_{air} is the degrees of freedom of air molecule. Also, the ideal gas law states

$$p_{cyl}(\theta_{rad})\, V_{cyl}(\theta_{rad}) = m_{air} R_{air} T_{cyl}(\theta_{rad}) \tag{33}$$

where m_{air} is the mass of air in the cylinder and R_{air} is the specific gas constant of air. It should be noted that the use of parameters for air is approximation since the actual gas in the cylinder is a mixture of air, fuel vapour and residual burned gas and their composition varies as combustion reaction progresses.

Eq.**(32)** can be rearranged so that the gas pressure can be calculated from the internal energy.

$$p_{cyl}(\theta_{rad}) = \frac{2\, e_{cyl}(\theta_{rad})}{f_{air}\, V_{cyl}(\theta_{rad})} \tag{34}$$

Note that the cylinder volume V_{cyl} can be calculated from Eq.**(31)**.

From Eqs.**(32)** and **(33)**, the gas temperature can also be calculated from the internal energy.

$$T_{cyl}(\theta_{rad}) = \frac{2\, e_{cyl}(\theta_{rad})}{f_{air}\, m_{air}\, R_{air}} \tag{35}$$

In the remainder of this section, the dynamics of the internal energy of the gas during compression and expansion strokes will be modelled as a differential equation. So, let us first clarify the initial condition, which can be determined from Eq.**(32)**.

$$e_{cyl}(0) = 44.64189754 \tag{36}$$

Adiabatic Motoring

For the sake of better comprehension of the problem, the simplest case of the gas dynamics in a closed volume with a moving piston is considered first.

When a piston is moving but there is no combustion (i.e. so-called "motoring") as well as no thermal energy exchange between the gas and the cylinder wall (i.e. adiabatic), the change in the internal energy of the gas is caused by the piston work only.

$$\frac{\mathrm{d}}{\mathrm{d}\theta_{rad}} e_{cyl}(\theta_{rad}) = -p_{cyl}(\theta_{rad}) \left(\frac{\mathrm{d}}{\mathrm{d}\theta_{rad}} V_{cyl}(\theta_{rad}) \right) \tag{37}$$

The R.H.S. of the above equation can be computed from Eq.**(34)** for the pressure and Eq.**(31)** for the cylinder volume.

Adiabatic Combustion

Now let us consider combustion. (Thermal exchange is not considered yet.) To deal with combustion phenomenon of an engine, there are many models proposed over the past decades. In this article, a phonomenological model often used for analyzing basic thermodynamics of combustion, called Wiebe heat-release model [6], is going to be employed. This model can estimate the heat release due to the combustion of fuel and air during an engine cycle with relatively small number of empirical model parameters, and is suitable for cycle-by-cycle torque estimation. Here, Wiebe model will be defined, analysed and simulated using both symbolic and numeric approaches.

Wiebe model needs another submodel for burn duration estimation, which is given in this article as a function of the engine speed.

$$\theta_{bd}(\omega_{eng}) = \frac{c_1\, \omega_{eng}}{c_2} - c_3\, \omega_{eng}^2 \tag{38}$$

where c_i (i = 1, 2, 3) need to be adjusted to experimental data.

Wiebe heat-release model itself is given as follows.

$$x_{burn}(\theta_{rad}) = \begin{cases} 0 & \theta_{rad} \le SA_{rad} \\ 1 - \mathrm{e}^{-a_{wi}\left(\frac{\theta_{rad} - SA_{rad}}{\theta_{bd}(\omega_{eng})}\right)^{m_{wi}+1}} & otherwise \end{cases} \tag{39}$$

where SA_{rad} is spark angle (or start of combustion) which is a control signal, a_{wi} and m_{wi} are adjustable parameters. For your information, the equation looks as follows when parameter values are substituted.

$$x_{burn}(\theta_{rad}) = \begin{cases} 0. & \theta_{rad} \le 2.879793266 \\ 1. - 1.\,\mathrm{e}^{-1035.329906\,(\theta_{rad} - 2.879793266)^4} & otherwise \end{cases} \tag{40}$$

This helps understand that this heat-release model returns 0 prior to the combustion, and that once the combustion starts at SA_{rad}, the value increases until it reaches 1 which corresponds to the end of combustion.

Heat release rate can be derived from Wiebe function.

$$\frac{\mathrm{d}}{\mathrm{d}\theta_{rad}} x_{burn}(\theta_{rad}) = \begin{cases} 0 & \theta_{rad} \le SA_{rad} \\ \dfrac{a_{wi}\left(\frac{\theta_{rad} - SA_{rad}}{\theta_{bd}(\omega_{eng})}\right)^{m_{wi}+1} \left(m_{wi}+1\right) \mathrm{e}^{-a_{wi}\left(\frac{\theta_{rad} - SA_{rad}}{\theta_{bd}(\omega_{eng})}\right)^{m_{wi}+1}}}{\theta_{rad} - SA_{rad}} & SA_{rad} < \theta_{rad} \end{cases} \tag{41}$$

When combustion is considered in addition to the piston work, the dynamics of the internal energy of the gas is described as follows instead of Eq.**(37)**.

$$\frac{\mathrm{d}}{\mathrm{d}\theta_{rad}} e_{cyl}(\theta_{rad}) = -p_{cyl}(\theta_{rad}) \left(\frac{\mathrm{d}}{\mathrm{d}\theta_{rad}} V_{cyl}(\theta_{rad}) \right) + Q_{\max} \left(\frac{\mathrm{d}}{\mathrm{d}\theta_{rad}} x_{burn}(\theta_{rad}) \right) \quad \textbf{(42)}$$

Note that this ordinary differential equation now contains a conditional branch in the second term on the R.H.S.

Combustion with Thermal Exchange between Gas and Wall

Because the temperature of the cylinder wall is kept much lower (at about 80 degrees C) than burning gases (over 2000 degrees C at peak), there is significant amount of thermal energy exchange ΔQ_w between the gas and the cylinder wall, which needs to be added to the combustion dynamics as follows.

$$\frac{\mathrm{d}}{\mathrm{d}\theta_{rad}} e_{cyl}(\theta_{rad}) = -p_{cyl}(\theta_{rad}) \left(\frac{\mathrm{d}}{\mathrm{d}\theta_{rad}} V_{cyl}(\theta_{rad}) \right) + Q_{\max} \left(\frac{\mathrm{d}}{\mathrm{d}\theta_{rad}} x_{burn}(\theta_{rad}) \right) - \Delta Q_w(\theta_{rad}) \quad \textbf{(43)}$$

In this paper, a simplistic model is used for thermal exchange just as an example.

$$\Delta Q_w(\theta_{rad}) = k_{therm} \left(T_{cyl}(\theta_{rad}) - T_{wall} \right) \quad \textbf{(44)}$$

where k_{wall} is the thermal conductivity and T_{wall} is the cylinder wall temperature being assumed constant.

Numerical Integration

Now the simulation results of Eqs.**(37)**, **(42)** and **(43)** are shown.

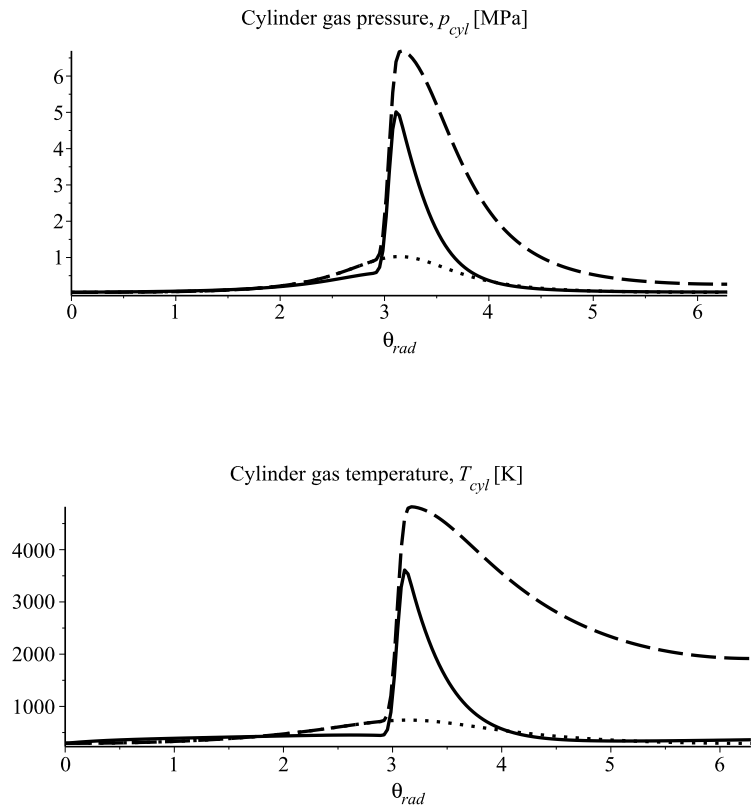


Figure 1. Plots of cylinder gas pressure (top) and temperature (bottom) for an engine cycle, obtained by numerical integration of an ODE for three different cases.

In the above plots for gas pressure and temperature, dotted lines represent adiabatic motoring Eq.**(37)**, dashed lines represent adiabatic combustion Eq.**(42)**, and solid lines represent combustion with thermal exchange Eq.**(43)**.

In the previous section where the combustion chemistry was examined, the pressure and the temperature of the gas were given as operating conditions. Calculations performed in this section may be used to set such operating conditions for chemical equilibrium calculation, thereby enabling the estimation of exhaust emissions from a combustion in engine cycle. However, the challenges listed in the previous section are only partially solved with such an approach. Thus, what has been presented here should be considered as just one baby step forward. Yet, the author would like to emphasize the usefulness of the combined use of symbolic and numeric approaches which greatly speed up comprehension of the problem.

Conclusions

In this paper, the importance and the effectiveness of the integrated use of symbolic and numeric approaches are presented. From the standpoint of the modelling process for the control system development, however, there still remain many other technical aspects that were unfortunately not discussed in this paper.

For example, modelling of chemical reactions occurring in aftertreatment system is as important as that of the combustion chemistry since the combustion and the aftertreatment system are controlled in an integrated manner in the modern engine control system. As for the modelling technologies, modelling a single phenomenon such as a chemical reaction is not even a start of system modelling. As mentioned in the introduction, there are a couple of system modelling technologies that are widely used in automotive industry today [3], [4], [5]. These system modelling technologies have distinctive features such as 1) their own declarative languages to directly describe differential-algebraic equations (DAEs), 2) automatic equation generation mechanism for component connections, known as "acausal" connection, 3) DAE solvers with index-reduction mechanism and support for hybrid-and-stiff system and so on.

Working with these plant modelling technologies is essential when it comes to the development of plant models to meet today's requirements imposed on the control system development, but these technologies are still too much oriented to the numeric approach, and the author believes that the symbolic manipulation technology can be further integrated to the plant modelling as presented in this article so that requirements for plant models can be fulfilled in a more timely manner.

References

[1] Maplesoft, "Maple," http://www.maplesoft.com/products/maple
[2] International Energy Agency, "World Energy Outlook 2010," Organization for Economic Cooperation and Development, 2010. http://www.worldenergyoutlook.org/
[3] Modelica Association, "Modelica," http://www.modelica.org/
[4] IEEE, "IEEE Standard VHDL Analog and Mixed-Signal Extensions," IEEE Std 1076.1-2007, ISBN 0-7381-5627-2, 2007
[5] MathWorks Inc., "Simscape," http://www.mathworks.com/products/simscape
[6] J.B. Heywood, "Internal Combustion Engine Fundamentals," McGraw-Hill, 1988
[7] S. Gordon, B.J. McBride, "Computer Program for Calculation of Complex Chemical Equilibrium Compositions and Applications. I. Analysis," NASA Reference Publication 1311, October 1994
[8] L. Eriksson, "CHEPP - A Chemical Equilibrium Program Package for Matlab," SAE Technical Paper, 2004-01-1460, 2004
[9] G.P. Smith, D.M. Golden, M. Frenklach, N.W. Moriarty, B. Eiteneer, M. Goldberg, C.T. Bowman, R.K. Hanson, S. Song, W.C. Gardiner, Jr., V.V. Lissianski, Z. Qin, GRI-Mech 3.0 Thermochemical Tables. http://www.me.berkeley.edu/gri_mech/

セッション６

Session 6

数式・数値融合計算

Symbolic-numeric computation

有理関数を基にした多変数近似GCD計算

Computing the Approximate Multivariate Greatest Common Divisor via Rational Function

讃岐 勝
筑波大学医学医療系 & 筑波大学附属病院総合臨床教育センター
Masaru Sanuki
Faculty of Medicine, University of Tsukuba
Center for Medical Education and Training, University of Tsukuba Hospital
sanuki@md.tsukuba.ac.jp

Abstract

In this paper, we propse two method to compute the approximate multivarite GCD for polynomial with floating-point numbers. One is based on Páde approximation, the other is based on Barnett's theorem. Also, we propose one refinement technique solving the linear equation within polynomial entries.

1 はじめに

1 変数多項式の近似 GCD（最大公約子）計算に比べて，多変数多項式の近似 GCD の計算に関する研究は盛んではない．本稿では，1 変数多項式の近似 GCD は精度よく計算できると仮定して，多変数多項式の近似 GCD 計算法を新たに提案する．

まず，多項式 F と G の近似 GCD を次のように定義する．

定義 1 (近似 GCD)**.** F と G が $F = C\tilde{F} + \Delta_F$ と $G = C\tilde{G} + \Delta_G$ と多項式の要素でかけるとき，C を許容度 $\varepsilon = \varepsilon(\Delta_F, \Delta_G)$ の近似共通因子といい，次数が最大の近似共通因子を近似 GCD といい $\mathrm{appGCD}(F, G) = C$ でかく（近似 GCD は一意に決定しない）. □

注意 1. 許容度を $\varepsilon = \varepsilon(\Delta_F, \Delta_G)$ と Δ_F, Δ_G の関数で表記した．いろいろ流儀があるが，本稿では

$$\varepsilon(\Delta_F, \Delta_G) = \max\left\{\frac{||\Delta_\mathrm{F}||}{||\mathrm{F}||}, \frac{||\Delta_\mathrm{G}||}{||\mathrm{G}||}\right\} \tag{1}$$

と係数の最大値の大きさを比較することで摂動部を見積もることにする. □

定義から，近似 GCD の許容度を正確に見積もる場合には C の計算以外に $\tilde{F}$ と $\tilde{G}$ を計算する必要がある．これまでの研究では，1. C だけを計算（許容度は正確ではない），2. C を求めた後，除算によって $\tilde{F}$ と $\tilde{G}$ を計算，3. $\tilde{F}$ と $\tilde{G}$ を先に計算し，除算によって C を計算，が主流であ

り，除算の方法によって許容度は変化する（多くの方法は 2-ノルムの意味で最小になるように除算を行う）．実際，すべての要素の決定は refinement（精度の改善）によって行われる．本稿では，すべての要素 $C, \tilde{F}, \tilde{G}$ を求め許容度も正確に計算することを念頭に考える．

上記の目的を達成するため，本稿では有理関数による近似法である Páde 近似を用いた多変数近似 GCD 計算法を提案する．多項式を要素に持つ線形連立方程式を解く必要があるが効率が悪いとされている．[讃岐 2012] では数値計算の算法に帰着する方法が提案され効率は悪くない．また，1 変数 GCD が既知であれば線形連立方程式を解くまでもなく多変数近似 GCD ができることを示す．

近似 GCD の計算では，近似 GCD または余因子のみが計算される．そのため，除算および refinement を通して全ての情報を得る必要があるが，本稿では多項式の関係式から refinement する方法を提案する．多項式要素の線形連立方程式を解くことが可能なため，非効率ではないと推測される．

本稿では次の記号を用いる．主変数 x，従変数 $\boldsymbol{u} = (u_1, \ldots, u_\ell)$ からなる浮動小数係数多項式 $F(x, \boldsymbol{u}), G(x, \boldsymbol{u}) \in \mathbb{F}[x, \boldsymbol{u}]$ を次で表現する．

$$\begin{aligned} F(x, \boldsymbol{u}) &= f_m(\boldsymbol{u})x^m + f_{m-1}(\boldsymbol{u})x^{m-1} + \ldots + f_0(\boldsymbol{u}), \\ G(x, \boldsymbol{u}) &= g_n(\boldsymbol{u})x^n + g_{n-1}(\boldsymbol{u})x^{n-1} + \ldots + g_0(\boldsymbol{u}). \end{aligned}$$

$\deg(F)$ を主変数 x に関する次数とする．多項式 $F(x, \boldsymbol{u}) \in \mathbb{F}[x, \boldsymbol{u}]$ に対して，従変数 $\boldsymbol{u}$ に関する全次数 w の斉次式を $\delta F^{(w)} \in \mathbb{F}[x, \boldsymbol{u}]$ で表す：$F = \sum_{i=0} \delta F^{(i)}$．ただし，$j = 0$ の場合には $\delta F^{(0)} = F^{(0)}$ と表記する場合もある．また，$[F]_i^i = \delta F^{(i)}$ と表記する場合もある．多項式に限らず，行列・ベクトルに関しても同様の表記法を用いる．ベクトル $\boldsymbol{v} \in \mathbb{F}[\boldsymbol{u}]^m$ に対して，$\delta \boldsymbol{v}^{(w)} \in \mathbb{F}[\boldsymbol{u}]^m$ はベクトルの各要素が全次数 w の斉次式から構成されるベクトルである．

1.1 線形方程式の解法

[**?**, 讃岐 2012] では，多項式を要素に持つ線形連立方程式を反復法により求める方法を提案した．以降，何度も利用するので簡単に述べる．次の線形連立方程式を考える．

$$A\boldsymbol{x} = \boldsymbol{b}. \tag{2}$$

ここで，$A \in \mathbb{F}[\boldsymbol{u}]^{m \times m}$ および $\boldsymbol{b} \in \mathbb{F}[\boldsymbol{u}]^m$ である．

$A^{(0)} \in \mathbb{F}^{m \times m}$ が正則と仮定する．このとき，$A^{(0)}\boldsymbol{x} = \boldsymbol{b}^{(0)}$ は線形代数・数値計算で知られた方法で簡単に解くことができる．

今，$A\boldsymbol{x} \equiv \boldsymbol{b} \pmod{I^w}$ が解くことができたと仮定する：$\boldsymbol{x} = \boldsymbol{c}^{(w-1)}$．このとき，$A\boldsymbol{x} \equiv \boldsymbol{b} \pmod{I^{w+1}}$ は次のように解く．この式において，全次数 w の斉次項のみを集めると，

$$\begin{aligned} \delta A^{(w)}\delta\boldsymbol{x}^{(0)} + \ldots + \delta A^{(1)}\delta\boldsymbol{x}^{(w)} + A^{(0)}\delta\boldsymbol{x}^{(w)} &= \delta\boldsymbol{b}^{(w)} \\ A^{(0)}\delta\boldsymbol{x}^{(w)} &= \delta\boldsymbol{b}^{(w)} - \sum_{j=1}^{w} \delta A^{(j)}\delta\boldsymbol{x}^{(w-j)}. \end{aligned} \tag{3}$$

方程式 (3) の右辺について $\delta\boldsymbol{x}^{(j)}$ $(j = 0, \ldots, w-1)$ は仮定より計算済みである．方程式 (3) は行列 $A^{(0)}$ の要素がすべて数値なので，線形代数による方法で解くことが可能である．

逆行列を用いる方法

方程式 (3) において，$w = 0$ のとき，すなわち $A^{(0)}\boldsymbol{x} = \boldsymbol{b}^{(0)}$ の計算を逆行列の計算によって行うと，$w \geq 1$ のとき $(A^{(0)})^{-1}$ はすでに既知なので行列とベクトルの積の計算のみのよって解 $\delta\boldsymbol{x}^{(w)}$ を計算することができる．

反復法による方法

Gauss-Seidel 法，Jacobi 法また Krylov 部分空間法に基づく方法によって計算することができる．ただ，多項式同士の加減算を多く行う必要があり反復回数が多くなったり行列の次数が大きくなると効率的でなくなる．

2　有理関数を利用する方法

今，有理関数 G/F の主変数 x に関する級数展開が得られたとする．

$$\frac{G(x,\boldsymbol{u})}{F(x,\boldsymbol{u})} = h_0(\boldsymbol{u}) + h_1(\boldsymbol{u})x + h_2(\boldsymbol{u})x^2 + \ldots \in \mathbb{F}\{x,\boldsymbol{u}\}. \tag{4}$$

$F = C\tilde{F} + \Delta_F$，$G = C\tilde{G} + \Delta_G$ とかくとき，

$$\begin{aligned}\frac{G}{F} = \frac{C\tilde{G}+\Delta_G}{C\tilde{F}+\Delta_F} &= \frac{C\tilde{G}+\Delta_G}{C\tilde{F}\Big(1+\dfrac{\Delta_F}{C\tilde{F}}\Big)} \\ &= \frac{C\tilde{G}+\Delta_G}{C\tilde{F}}\Big(1-\frac{\Delta_F}{C\tilde{F}}+\tilde{\Delta}^2\Big) \\ &= \frac{\tilde{G}}{\tilde{F}} + \frac{\tilde{F}\Delta_G - \tilde{G}\Delta_F + \Delta^2}{C\tilde{F}^2}\end{aligned} \tag{5}$$

と近似できるので，有理関数の場合においても許容度 $O(\Delta)$ の摂動が入っているとみなすことができる．ここで，$\Delta^2 \in \mathbb{F}\{x\boldsymbol{u}\}$ であり，$||\Delta^2|| = O(\varepsilon^2)$ である．

べき級数展開の方法

G/F の級数展開は実際に次の前処理を行った後，Henrici による方法を用いて行う．べき級数 $A = \sum_{i=0} a_i(\boldsymbol{u})x^i$ と $B = \sum_{i=0} b_i(\boldsymbol{u})x^i$ の積は Cauchy の積法則により，$P = AB$ の x^q の係数 $p_q(\boldsymbol{u})$ は $p_q = \sum_{i=0}^{q} a_i b_{q-i}$ と多項式の積と同様の表現で書くことができる．これによって P/B の係数 a_q は次でかける（$b_0 \neq 0$）．

$$a_p = \frac{p_q - \sum_{i=0}^{q-1} a_i b_{q-i}}{b_0}. \tag{6}$$

ゆえに b_0 に定数項があれば，$1/b_0$ が級数展開することができるため，$a_p \in \mathbb{F}\{\boldsymbol{u}\}$ になるように展開できる．この式は次数の低い項から順に構成される展開式になっていることに注意する．

2.1 Páde 近似による方法

主変数に関する近似 GCD の次数 k が既知とする．このとき，(4) で得た級数を分母の多項式の次数 $m-k$，分子の多項式の次数 $n-k$ の有理関数近似したものは，与えられた多項式から近似 GCD を取り除いた余因子によって構成されたものになる．このような分子・分母を求める方法として Páde 近似による方法がある．1 変数の場合には既知の方法であるが [Pan01]，多変数の場合は多項式を要素に持つ線形連立方程式を解く必要があるため避けられる傾向がある．

実際に次の関係式でかける．

$$L(F)_q \boldsymbol{h}_q = \boldsymbol{g}_q. \tag{7}$$

ここで，各行列，ベクトルは次で表される．

$$L(F)_q = \begin{pmatrix} f_0 & & & \\ f_1 & f_0 & & \\ \vdots & \ddots & \ddots & \\ f_q & f_{q-1} & \cdots & f_0 \end{pmatrix} \in \mathbb{F}[\boldsymbol{u}]^{(q+1)\times(q+1)}, \boldsymbol{g}_q = \begin{pmatrix} g_0 \\ g_1 \\ \vdots \\ g_q \end{pmatrix}, \boldsymbol{h}_q = \begin{pmatrix} h_0 \\ h_1 \\ \vdots \\ h_q \end{pmatrix} \in \mathbb{F}[\boldsymbol{u}]^{q+1}.$$

G/F の級数展開から $\tilde{G}/\tilde{F}$ の分子・分母を求めるためには．近似 GCD の次数 k とするとき，分母の次数 $m-k$，分子の次数 $n-k$ となる Páde 近似により有理関数近似すればよく，各係数は次の関係式で表現される．

$$L(H)_{m+n-2k} \tilde{\boldsymbol{f}}_{m+n-2k} = \tilde{\boldsymbol{g}}_{m+n-2k}. \tag{8}$$

これをみたす $\tilde{F}$ および $\tilde{G}$ のすべての係数を求めるためには，係数を 1 つ定める必要がある．1 変数近似 GCD 計算の場合，$\tilde{F}^{(0)}$ または $\tilde{G}^{(0)}$ の定数係数を 1 にし，その上で補間法によって関係式をみたす係数を計算する．多変数多項式の場合も，同様の方法がとることができるが非常に効率が悪い．

2.2 1 変数 GCD が既知の場合

1 変数 GCD の主変数 x に関する次数 k および近似 GCD $\mathrm{appGCD}(F,G) = C^{(0)} \pmod I$ がわかっていると仮定する（$\tilde{F}^{(0)}$ および $\tilde{G}^{(0)}$ も既知）．あらかじめ，$\mathrm{appGCD}(f_0(\boldsymbol{u}), g_0(\boldsymbol{u})) = c_0(\boldsymbol{u})$ を計算し，$\tilde{f}_0$ を f_0 と c_0 による近似除算によって計算する．定数項を計算した上で，

$$F \to F/f_0 \tag{9}$$

とべき級数除算することによって，入力多項式 F の定数項を 1 にする（数にする）．実際には，計算に必要な従変数の最大全次数 t がわかっているものとし，$F \to F/f_0 \pmod{I^{t+1}}$ を計算する．

このとき，

- $\tilde{F}^{(1)}$ と $\tilde{G}^{(1)}$ の定数項：
  $\delta\tilde{f}_0^{(1)}$ は既知であり，$\delta\tilde{g}_0^{(1)}$ は $h_0\tilde{f}_0 = \tilde{g}_0$ より $\delta\tilde{g}_0^{(1)} = h_0^{(0)}\delta\tilde{f}_0^{(1)} + \delta h_0^{(1)}\delta\tilde{f}_0^{(0)}$ と和・積のみによって計算可能である．

- $\tilde{F}^{(1)}$ と $\tilde{G}^{(1)}$ の x^1 の係数：
  $\delta\tilde{g}_1^{(1)} = [\delta h_0\delta f_1 + \delta h_1\delta f_0]_1^1 = [\delta h_0\delta f_1 + \delta h_1]_1^1$ であり，$\delta g_1^{(1)} = h_0^{(0)}\delta f_1^{(1)} + \delta h_0^{(1)}\delta\tilde{f}_1^{(0)} + \delta h_1^{(1)}$ なる関係式が得られるが，$\delta\tilde{g}_1^{(1)}$ および $\delta\tilde{f}_1^{(1)}$ は定まらない．

- $\tilde{F}^{(1)}$ と $\tilde{G}^{(1)}$ の x^p の係数：
  $\delta\tilde{g}_p^{(1)} = [\sum_{i=0}^{p-1} \delta h_i \delta f_{p-j}^{(1)}]_1^1$ であり，$\tilde{F}^{(1)}$ および $\tilde{G}^{(1)}$ について x^{p-1} までの係数がわかっていたとしても，$\delta\tilde{g}_p^{(1)}$ および $\delta\tilde{f}_p^{(1)}$ は定まらない．

以上のように，1 変数 GCD がわかっていても状況は変わらず，求めるためには $p = m + n - 2k$ まで計算を行い，全次数ごとで Páde 近似そのものを行う必要がある．問題サイズが少し小さくなる．

2.3 Barnett の定理の改良

Diaz-Toca と G. Vega によって，べき級数の係数から構成する行列の線形結合から GCD を得る方法が提案されている [DG02]．これは [Sanuki09] により次のように拡張できる．

定理 1 (Barnett の定理の拡張 [DG02, Sanuki09])**.** 行列 Q_m を次で定義する．

$$Q_m = \begin{pmatrix} h_0 & h_1 & \dots & h_{m-1} \\ h_1 & h_0 & \ddots & \vdots \\ \vdots & \ddots & \ddots & \vdots \\ h_{m-1} & \dots & h_1 & h_0 \end{pmatrix} = (\boldsymbol{q}_1, \dots, \boldsymbol{q}_m) \in \mathbb{F}[\boldsymbol{u}]^{m\times m}.$$

近似 GCD の主変数 x に関するが k のとき，前から $m-k$ 列 $\boldsymbol{q}_1, \dots \boldsymbol{q}_{m-k}$ は $\mathbb{F}[\boldsymbol{u}]$-線形独立であり，後ろ k 列 $\boldsymbol{q}_{m-k+1}, \dots \boldsymbol{q}_m$ は前から $m-k$ 列のベクトルで張ることができる：

$$\boldsymbol{q}_{m-k+j} = \sum_{i=1}^{m-k-1} r_{j,i}\boldsymbol{q}_i + r_{j,m-k}\boldsymbol{q}_{m-k} \quad (j = 1, \dots, k). \tag{10}$$

このとき，$r_{j,m-k}$ は GCD の x^{k-j} の係数である． □

3 Refinement

多変数多項式 GCD の refinement は 1 変数の場合と同じ方法を選択すると多項式の和・積の計算する必要がある，また行列のサイズが病徴する傾向があるため計算時間がかかってしまう．

本稿では行列のサイズを減らすことに重みを置き，次のような行列 $T(\tilde{F}, \tilde{G}, C, F, G)$・ベクトル $U(\tilde{F}, \tilde{G}, C)$ を考える．

$$T(\tilde{F}, \tilde{G}, C, F, G) = \begin{pmatrix} \tilde{F} & 0 & 0 \\ 0 & 0 & C \\ 0 & G & F \end{pmatrix} \in \mathbb{F}[x, \boldsymbol{u}]^{3\times 3}, U = \begin{pmatrix} C \\ \tilde{F} \\ \tilde{G} \end{pmatrix} \in \mathbb{F}[x, \boldsymbol{u}]^3. \tag{11}$$

まず，次を保証する．

補題 1. $\det(T) = -G(\tilde{F}C) \neq 0.$ □

行列 T が正則であるので，線形連立方程式 $T\boldsymbol{x} = (F, G, 0)^T$ を考えると解 $\boldsymbol{x} = U$ が得られるので，refinement を行うための関係式を導くことができる．行列のサイズが小さいので，逆行列を用いて計算しても効率がよい．実際には次のように計算をする．

算法 1 (多変数多項式の refinement)**.**

- 入力：$C_0, \tilde{F}_0, \tilde{G}_0 \in \mathbb{F}[x, \boldsymbol{u}]$
  $T_0 = T(\tilde{F}_0, \tilde{G}_0, C_0, F, G)$ ，$U_0 = (\tilde{F}_0, \tilde{G}_0, C_0)^T$, $S = (F, G, 0)^T$
- 残差が小さくなるまで，次を繰り返す.
  $i = 0$
  $r_i = T_i U_i - S$;
  $r_i = T_i \boldsymbol{y}$ を解く（3 次正方行列の線形方程式）
  $U_{i+1} = U_i + r_i$
  r_i が十分小さくなったら計算終了

補題 2 (条件数)**.** 各 T_i の条件数は近似 *GCD* の定数項に依存する．定数項が $O(1)$ であれば計算は安定する． □

注意 2. 1 変数近似 GCD をあらかじめ refinement しないと計算は収束しない．そのため，上記の方法以外の方法であらかじめ refienment をする必要がある.

4 まとめ

本稿では，有理関数のべき級数を基にした多変数多項式の近似 GCD 算法について考察した．いずれにおいても 1 変数の場合を拡張しただけにとどまった．refinement に関しては，多項式要素で線形連立方程式が解けることによりこれまでできなかった方法ができるようになったことを確認した．ただ，1 変数多項式の近似 GCD をあらかじめ処理しなければいけないなど，改良の余地はまだある.

参考文献

[Barnett70] S. Barnett. *Greatest common divisor of two polynomials.* Linear Algebra Appl., **3**, 1970, 7–9.

[Barnett71] S. Barnett. *Greatest common divisor of several polynomials.* Proc. Camb. Phil. Soc., **70**, 1971, 263–268.

[BP94] D. Bini and V. Pan. *Polynomial and matrix computations: volume 1 fundamental algorithms.* Birkhäuser, 1994.

[DG02] G. M. Diaz-Toca and L. Gonzalez-Vega. *Barnett's theorems about the greatest common divisor of several univariate polynomials though Bézout-like matrices. J. Symb. Compu.*, **34**, (2002), 59–81.

[DG06] G. M. Diaz-Toca and L. Gonzalez-Vega. *Computing greatest common divisors and squarefree decompositions through matrix methods: The parametric and approximate cases. Linear Algebra Appl.*, **412(2-3)**, (2006), 222–246.

[Henrici56] P. Henrici. Automatic computations with power series. *Journal of the ACM*, 1956 **(3)**, 10–15.

[Knuth97] D. E. Knuth. Art of Computer Programming, Volume 2: Seminumerical Algorithms (Third Edition), Addison-Wesley Professional, 1997.

[ONS91] M. Ochi, M-T. Noda and T. Sasaki, *Approximate greatest common divisor of multivariate polynomials and its application to ill-conditioned systems of algebraic equations*, J. Inform. Proces., **14** (1991), 292–300.

[Pan01] V.Pan, *Univariate polynomials: nearly optimal algorithms for factorization and rootfinding*, Proc. of ISSAC'01, ACM Press, 2001, 253–267.

[Sanuki08] M. Sanuki. A Study on the Approximate GCD, Ph. D. Thesis, University of Tsukuba, 2008.

[Sanuki09] M. Sanuki. Computing multivariate approximate GCD based on Barnett's theorem, *Proc. of SNC'09*, ACM Press, 2009, 149–157.

[讃岐 2012] 讃岐勝, 多項式を要素にもつ線形連立方程式の解法：その 2, 数式処理研究の新たな発展 2012, 京都大学数理解析研究所, (2012 年 7 月 4-6 日)

[讃岐 2013] 讃岐勝, Jacobi 法を基にした多項式要素の線形方程式の解法, 第 42 回数値解析シンポジウム講演予稿集, 2013, 144-147.

[SN89] T. Sasaki and M-T. Noda, *Approximate square-free decomposition and root-finding of ill-conditioned algebraic equations*, J. Inform. Proces., **12** (1989), 159–168.

厳密に与えられた系の Groebner 基底を数値的に求める場合に必要な桁精度の考察
A note on required precision for computing numerical Groebner basis of exact input*

長坂耕作 (Kosaku Nagasaka)†
神戸大学 (Kobe University)

Abstract

Recently Y. Liang studied some numerical error analyses of computing Gröbner basis by floating-point numbers. One is for rounding errors in "Selecting lengths of floats for the computation of approximate Gröbner bases", J. Symb. Comp., **53**:40–52, 2013 and the other is for cancellation errors in "Structures of precision losses in computing approximate Gröbner bases", J. Symb. Comp., **53**:81–95, 2013. In this paper, we give a short survey on his results and some notes on his approaches.

1 はじめに

近年，Y. Liang により Gröbner 基底を浮動小数点数を用いて計算する場合の誤差解析が行われています。1 つが，丸め誤差（情報落ち）に対する論文 [5] で，もう 1 つが，桁落ち誤差に対する論文 [3] です。本報告では，これらの先行研究に関するサーベイと考察を，特にそのアプローチを中心に行います。なお，佐々木ら [6, 7, 8] による関連研究を除くと，近年に発表されている同種の研究成果は余りありません。

詳細については以下で議論しますが，はじめに端的な結論を示しておきます。Liang による結果 [5, 3] については，与えられている定義に基づく理論展開としては非常に興味深いのですが，数値計算における誤差解析としても，Gröbner 基底計算における誤差解析としても，実際の誤差発生過程（原因，要因）との関連付けが十分ではありません。情報落ちに関する論文 [5] では，実験結果が実際に発生した誤差と相関している可能性はありますが，因果関係については示されておらず，後者については具体的な見積りまで至っていないため，その効果については（因果を別にしても）分かりません。

1.1 記法

本報告で取り扱う多項式は，$x_1, \ldots, x_n$ を変数とする実係数多項式です。その全体を，$\mathbb{R}[\vec{x}] = \mathbb{R}[x_1, \ldots, x_n]$ で表します。Liang の原論文 [5, 3] では，$\mathbb{FR}_\ell$ を仮数部が ℓ ビットの浮動小数点数の集合[1)] としているものの，$0 \notin \mathbb{FR}_\ell$ が明らかに成り立つと述べています。これは仮数部が v で指数部が m の浮動小数点数は，$|v| \in [1, 2)$, $m \in \mathbb{Z}$ を満たす[2)] としているためのようです。本報告では，$\mathbb{FR}_\ell$ を仮数部が ℓ ビットの浮動小

*This work was supported in part by MEXT KAKENHI (22700011).

†nagasaka@main.h.kobe-u.ac.jp

1) Denote by $\mathbb{FR}_\ell$ the set of floating-point real numbers whose significands have ℓ binary bits.

2) 通常は，指数部にも表現可能な範囲があり，仮数部は必ずしもこの範囲に入らないことがあります。

数点数の集合と仮定するものの，Liang の主張に従い，$0 \notin \mathbb{FR}_\ell$ であることについては，基本的に否定せずに説明します。任意の実数 $a \in \mathbb{R}$ を上記の表現を持つ浮動小数点数（ただし，無限精度）に変形するため，指数部を計算するのに便利なように，$\mathrm{ilog}_2(a) = \lfloor \log_2(a) \rfloor$ という記法も導入しておきます。指数部は，$\mathrm{ilog}_2(a)$ となります。最後に，誤差により失われた桁数を表す集合として，$\hat{\mathbb{Z}} = \mathbb{Z} \cup \{-\infty\}$ を定義します。

その他，Gröbner 基底やイデアルに関する記法として一般的なものを使います。本報告では，変数のみからなる積 $x_1^{e_1} \cdots x_n^{e_n}$ を項 (term) と呼び，係数までを含むものを単項式 (monomial) と呼びます。多項式 $f(\vec{x}) \in \mathbb{R}[\vec{x}]$ の係数がゼロでない単項式の項のうち，項順序最大のものを頭項と呼び，$\mathrm{ht}(f)$ で表します。

2　情報落ち誤差に対する評価

本章では，Liang[5] による情報落ち誤差に対する評価方法について簡単にまとめておきます。桁落ち誤差（を場合によっては含む，特定の状況）は対象から除外されており，基本的に情報落ち誤差（大きさの異なる数の加減算により，下位の桁が無視されることによる誤差）のみに着目していることに注意してください。さらに，多項式系が連続な場合のみ扱うこととし，不連続な場合は扱わないとされています（Faugère と Liang[1] によれば，入力の多項式系の係数を摂動させた場合に，その Gröbner 基底が連続的に変化する場合とそうでない場合です）。原論文では，解くべき問題を次のように表現していますが，曖昧な表現（過剰決定系でない，ほとんど，信頼できる）については以下で数学的な定義を与えることにします。

問題 1
与えられたサポート（構造）を持つ過剰決定系でない多項式系の族に対して，個々の多項式系の *Gröbner* 基底を数値的に計算する際，ほとんどの演算が信頼できるのに必要な桁精度を求めよ。 ◁

以下，多項式 $f(\vec{x}) \in \mathbb{R}[\vec{x}]$ に対して，そのサポート（係数が 0 でない単項式の項）の集合を $\mathcal{T}(f)$ で表し，変数 $\vec{x}$ による項全体の集合を $\mathcal{T}^{\vec{x}}$ とします。即ち，$\mathcal{T}(f) \subset \mathcal{T}^{\vec{x}}$ が成り立ちます。一方，項が $t \in \mathcal{T}^{\vec{x}}$ の単項式の係数部分は，$\mathrm{coeff}_t(f)$ で表します。相対的に大きな係数を持つ項集合として，閾値 $\mu \in \mathbb{N}$ に対して，$\mathcal{T}_\mu(f) = \left\{ t \in \mathcal{T}(f) \,\middle|\, |\mathrm{coeff}_t(f)| \geq 2^{-\mu} \|f\|_\infty \right\}$ を定義します。多項式の有限集合を多項式系 $\Psi = \{\Psi_i \in \mathbb{R}[\vec{x}]\}$ $(i \in \mathbb{N})$ とし，サポートを $\mathfrak{M} = \left\{ \mathfrak{M}_i \subset \mathcal{T}^{\vec{x}} \,\middle|\, \mathfrak{M}_i = \mathcal{T}(\Psi_i),\ i = 1, \ldots, \#\Psi \right\}$ で表します。

定義 1
与えられた $\mu, \delta \in \mathbb{N}$ に対して，信頼できる計算とは，次の $\mathcal{H}(\delta, \mu)$ が真と定義する。

$$
\begin{aligned}
\mathcal{H}(\delta,\mu) \quad = \quad & \forall f_1 \forall f_2 \forall t_1 \forall t_1' \forall t_2 \forall t_2' (\\
& (f_1, f_2 \in \mathbb{R}[\vec{x}] \wedge t_1, t_1' \in \mathcal{T}_\mu(f_1) \wedge t_2, t_2' \in \mathcal{T}_\mu(f_2) \wedge t_1' t_2 = t_1 t_2') \\
& \quad \to |\log_2(|\mathrm{coeff}_{t_1'}(f_1)\mathrm{coeff}_{t_2}(f_2)|/|\mathrm{coeff}_{t_1}(f_1)\mathrm{coeff}_{t_2'}(f_2)|)| \leq \delta)
\end{aligned}
$$

ここで，δ は情報落ちの（およその）許容度，μ は相対的に考慮する範囲の閾値である。 ◁

上記の定義において，具体的な $f_1, f_2 \in \mathbb{R}[\vec{x}]$ についての議論ではなく，任意の多項式を対象としていることに留意する必要があります。原論文には，この定義により繰り返される演算も対象となっているような記述があります。しかしながら，情報落ちした部分の誤差の伝播についての考慮は定義において行われておりませんので，少し注意が必要と考えられます。なお，$\mathcal{H}(\delta, \mu)$ が真となる必要十分条件は，$\delta \geq 2\mu$ であることが示されています。

信頼できる計算の定義により，「ほとんど」も定義することが可能になります。サポートが $\mathfrak{M}$ の多項式系の族を $\Omega_{\mathfrak{M}}$ で表し，多項式系 Ψ の項順序 $\prec$ に関する簡約 Gröbner 基底を $G_{\Psi,\prec}$ で表します。このとき，$\Omega_{\mathfrak{M}}$ の部分集合 $\Lambda_{\mu,\prec}$ を，$\Lambda_{\mu,\prec} = \{ \Psi \in \Omega_{\mathfrak{M}} \mid \exists g \in G_{\Psi,\prec} \text{ s.t. } \mathrm{ht}(g) \notin \mathcal{T}_\mu(g) \}$ と定義します。この集合は，

基底多項式の頭係数が相対的に小さい多項式系を集めたものです。原論文では，この集合 $\Lambda_{\mu,\prec}$ に入らない多項式系に対して十分な桁精度を見積ることを目的としています。その議論のため，適当に選んだ具体的な多項式系が，この集合に入ってしまう確率を $p_{\mathfrak{M}}(\Lambda_{\mu,\prec})$ で表します（確率が十分小さいことを「ほとんど」と定義します）。ところで，$p_{\mathfrak{M}}(\Lambda_{0,\prec}) = 0$ の場合，μ の値に関わらず，常に確率は 0 となります。なぜなら，$p_{\mathfrak{M}}(\Lambda_{0,\prec}) = 0$ は，頭係数が基底多項式において常に絶対値最大を意味しているため，μ を変化させても，確率は変化しないからです。原論文では，これは少し問題だということで，このような状況を発生させないことを，(ざっくりと）過剰決定系でない，と表現し条件に加えています。例として，$G_{\Psi,\prec} \not\subset \mathcal{T}^{\vec{x}}$ が挙げられています。実際，単項イデアルは問題とされる条件を満たすのでが，いわゆる過剰決定系ではありません。これらを踏まえ，原論文の問題は次のように定式化されました。

問題 2
与えられた $\tilde{\ell} \in \mathbb{N}$, $\alpha \in (0,1)$ と，$p_{\mathfrak{M}}(\Lambda_{0,\prec}) \neq 0$ を満たす $(\mathfrak{M}, \prec)$ に対し，$p_{\mathfrak{M}}(\Lambda_{\mu,\prec}) \leq \alpha$ かつ $\mathcal{H}(\delta, \mu)$ が $\delta = \ell - \tilde{\ell}$ に対して真になるような，$(\mu, \ell) \in \mathbb{N}^2$ を求めよ。 ◁

原論文では，問題 2 を解くためのアルゴリズムが提案されています。確率の許容度 $\alpha \in (0,1)$ と得たい有効桁 $\tilde{\ell} \in \mathbb{N}$ に対して，$p_{\mathfrak{M}}(\Lambda_{\mu,\prec}) \leq \alpha$ となる μ を決定することで，信頼できる計算に必要な δ が自動的に求まるので，必要な桁精度を $\ell = \tilde{\ell} + \delta$ として見積ることになります。引用はしませんが，基本的にモンテカルロ法で μ を見積もり，そこから ℓ を計算するアルゴリズムとなっています。正しいサンプリングが行われること，モンテカルロ法による正しい見積もりとなっていることの証明も与えられています。アルゴリズムの証明の他，モンテカルロ法の計算時間を短くするための工夫として，μ と $-\log_2(p_{\mathfrak{M}}(\Lambda_{\mu,\prec}))$ に漸近的な性質があることを示しています。これにより，切片と傾きが分かれば見積もりを簡略化できます。

数値実験は，Maple13 の FGb MAple package を用いた実装（後述の漸近的な性質も活用）を使って，$(\tilde{\ell}, \alpha) = (10, 2^{-5})$ と全次数逆辞書式順序の組み合わせで行われています。実際に発生する誤差の大きさは，Traverso と Zanoni[2] のものを使っているようです。見積もりの評価を行う指標として，原論文では，$2 \times \text{loss} < \text{length} < 5 \times \text{loss}$ を満たすものを reasonably large と定義しています。この定義に基づくと，多くの結果が reasonably large の見積もりとなるようです。見積もり以上に誤差が発生している多項式系もあるようですが，これらは不連続な系であり，原論文の仮定を満たしていないと述べられています。

2.1 考察

原論文のタイトルは，Gröbner 基底を数値的に計算する場合に必要とされる桁精度の選択となっていますが，実際は，μ と $p_{\mathfrak{M}}(\Lambda_{\mu,\prec})$ の相関についての論文と考えられます。実験結果は，一部を除き reasonably large と評価されていますが，因果関係については証明されていません。原論文の方法は，(1) 確率の許容度 $\alpha \in (0,1)$ と得たい有効桁 $\tilde{\ell} \in \mathbb{N}$ に対して，(2) $p_{\mathfrak{M}}(\Lambda_{\mu,\prec}) \leq \alpha$ となる μ を決定し，(3) $\mathcal{H}(\delta, \mu)$ が真となるように δ を $\delta \geq 2\mu$ から求め，(4) 必要な桁精度を $\ell = \tilde{\ell} + \delta$ として見積っているに過ぎず，Gröbner 基底を数値的に計算する場合に必要とされる桁精度の選択として適切であるかについては議論されていません。

ステップ (2) の $p_{\mathfrak{M}}(\Lambda_{\mu,\prec}) \leq \alpha$ は，簡約 Gröbner 基底の基底多項式のノルムに対する，頭係数の相対的な大きさが μ 未満となる確率が α 以下となる μ を求めることに対応しています。あくまでも同一サポートを持つ多項式系の族における簡約 Gröbner 基底の性質をモンテカルロ法で調べていることになります。

ステップ (3) の $\mathcal{H}(\delta, \mu)$ が真となる計算について考えてみます。丸めにより失われる下位の桁が δ 以下になれば良いのですが，演算結果における誤差の大きさに下位の桁の長さは依存しません。仮にこのモデルが情報落ち（丸め誤差）によって失われる桁精度を正しく見積もっていたとしても，μ が結果の簡約 Gröbner 基底から求められており，計算過程にどのように反映可能なのかは未知です。

ここで情報落ちが問題となる状況を考察してみます。単に，情報落ちが発生しただけでは，相対的な誤差の大きさは丸め誤差と変わりません。丸め誤差が積み重なった結果が情報落ちとなりますので，見積もりの上では丸め誤差の見積もりで十分とも考えられます（過大な評価となり得ますが）。一方，佐々木ら [6] が明らかにしているように，自己簡約などの発生により上位桁が打ち消されることにより下位桁の不足が露見することによって問題となることもありますが，これは，桁落ち誤差と考えられます。そのため，原論文のようなモンテカルロ法を用いるのであれば，演算回数に着目して必要な桁精度を見積る方が適切ではないかと考えられます。講演においては，この方針で数値実験を行った結果を報告する予定です。

3 桁落ち誤差に対する評価

本章では，Liang[3] による桁落ち誤差に対する評価方法について簡単にまとめておきます。前述の論文 [5] で扱われている丸め誤差（または情報落ち）とは異なるので注意してください。また，原論文では桁落ち誤差の評価が目的でなく，Gröbner 基底を数値的に計算する際に精度が失われる仕組みを解明することが目的となっているように思えます。中心的な役割を果たすのは，以下で定義する τ-representation という，有効桁を追跡しようとする仕組みです。これは，Mathematica の多倍長浮動小数点数や佐々木らの `efloat`[4] と同じく，浮動小数点数演算による誤差を追跡可能とするものです。これらは，数学的厳密性を考慮した区間数や白柳らによる安定化理論 [9] とは異なり，ある程度簡略化された方法ですが，効果的に誤差の大きさを測ることが可能です。

定義 2 (τ-Representations of reals)
$a \in \mathbb{R} \setminus \{0\}$ と $\tilde{a} \in \mathbb{FR}_\ell$ が以下を満たすならば，$\tilde{a}\tau^A$ を a の **τ-representation** という。

$$a\tilde{a} > 0,\ A \in \hat{\mathbb{Z}} \cap \left[\ell - \mathrm{ilog}_2 \left(\frac{|a|}{|a - \tilde{a}|} \right),\ +\infty \right)$$

また，$\tilde{a}\tau^A$ において，τ^A の部分を **τ-part**，A を **τ-exponent** という。 ◁

τ-exponent は，$\tilde{a}$ の a に対して失われた桁数の上限を表しています。以後，$\mathbb{FR}_\ell$ を拡張した集合として，$\mathbb{FT}_\ell = \left\{ \tilde{a}\tau^A \,\middle|\, \tilde{a} \in \mathbb{FR}_\ell,\ A \in \hat{\mathbb{Z}} \right\}$ を定義します。次に，τ-representation を多項式に拡張し，τ-representation 間の関係を導入します。

定義 3 (τ-Representations of polynomials)
多項式の係数それぞれに *τ-representation* を導入したものを，多項式の **τ-representation** とする。つまり，$\tilde{f} = \tilde{a}_1 t_1 + \cdots + \tilde{a}_s t_s \in \mathbb{FR}_\ell[\vec{x}]$ の $f = a_1 t_1 + \cdots + a_s t_s \in \mathbb{R}[\vec{x}]$ $(a_i \neq 0)$ に対する *τ-representation* は，$f_\tau = \tilde{a}_1 \tau^{A_1} t_1 + \cdots + \tilde{a}_s \tau^{A_s} t_s$ となる。 ◁

定義 4 (多項式間の失われた桁の比較)
$f_{\tau_1} = \sum_{i=1}^{s} \tilde{a}_i \tau^{A_{1,i}} t_i$ と $f_{\tau_2} = \sum_{i=1}^{s} \tilde{a}_i \tau^{A_{2,i}} t_i$ は，同じ $f \in \mathbb{R}[\vec{x}] \setminus \{0\}$ の *τ-representation* とし，サポート $t_i \in \mathcal{T}^{\vec{x}}$ は異なる（$t_i \neq t_j$ $(i \neq j)$）とする。このとき，$A_{2,i} \geq A_{1,i}$ $(i = 1, \ldots, s)$ ならば，f_{τ_2} は f_{τ_1} よりも精度が失われていると言い，$f_{\tau_2} \rtimes f_{\tau_1}$ と書く。 ◁

誤差を追跡する場合，それを踏まえたゼロ判定がアルゴリズム上大きな役割を果たし，これにより，アルゴリズムの流れが大きく変化します。原論文では，以下の方法でゼロ判定を行うと提案しています。

仮定 5 (ゼロ判定基準)
$L \in \mathbb{Z}_{\geq 0} (< \ell)$ をゼロ判定の相対的な尺度とし，$A_1, A_2 < \ell - 3$ を満たす $\tilde{a}_1 \tau^{A_1},\ \tilde{a}_2 \tau^{A_2} \in \mathbb{FT}_\ell$ を，$a_1,\ a_2 \in \mathbb{R} \setminus \{0\}$ の *τ-representation* とする。このとき，$\tilde{a}_1 \tilde{a}_2 < 0$ の場合の加法と $\tilde{a}_1 \tilde{a}_2 > 0$ の場合の減算では，ゼ

ロ判定が必要となり，その基準を以下のように設定する。

$A^* < \ell - 3$	$\Longrightarrow$	$\tilde{a}_1\tau^{A_1} \pm \tilde{a}_2\tau^{A_2} \neq 0$ と見做す
$A^* \geq \ell - 3,\ \min(K_1, K_2) > L$	$\Longrightarrow$	$\tilde{a}_1\tau^{A_1} \pm \tilde{a}_2\tau^{A_2} = 0$ と見做す
$A^* \geq \ell - 3,\ \min(K_1, K_2) \leq L$	$\Longrightarrow$	*undecidable* と見做す

$A^* = \max(A_1 + \mathrm{ilog}_2(|\tilde{a}_1|/|\tilde{a}_1 \pm \tilde{a}_2|),\ A_2 + \mathrm{ilog}_2(|\tilde{a}_2|/|\tilde{a}_1 \pm \tilde{a}_2|))$
$K_1 = \ell - \max(A_1,\ A_2),\ K_2 = \mathrm{ilog}_2(\max(|\tilde{a}_1|,\ |\tilde{a}_2|)/|\tilde{a}_1 \pm \tilde{a}_2|)$

◁

言い換えると，次のようなゼロ判定基準となります。

- 計算後の有効桁数が少なくとも 4 桁残っていれば，ノンゼロ（$\neq 0$）と見做す。
- 計算前の有効桁数と計算による桁落ちが，閾値（L）を共に越えると，ゼロ（$= 0$）と見做す。
- それ以外は，決定不可能と見做す。

原論文では，丸め誤差の考慮はしないと明言されています。そのため，本節で導入する $\mathbb{FT}_\ell$ 上の演算では，丸め誤差は発生しないという仮定が組み入れられていることに注意してください。

命題 6
$a_1, a_2 \in \mathbb{R} \setminus \{0\}$ を表す $\tilde{a}_1\tau^{A_1}, \tilde{a}_2\tau^{A_2} \in \mathbb{FT}_\ell$ に対し，$A_1, A_2 \leq \ell - 3$ ならば以下が成り立つ。

1. a_1a_2（a_1/a_2）に対する $(\tilde{a}_1\tilde{a}_2)\tau^A$（$(\tilde{a}_1/\tilde{a}_2)\tau^A$）で最小の $A \in \hat{\mathbb{Z}}$ は，$A \leq \max(A_1, A_2) + 2$ を満たす。
2. $\tilde{a}_1\tau^{A_1} \pm \tilde{a}_2\tau^{A_2}$ は，仮定 5 の下で，ノンゼロとする。このとき，$a_1 \pm a_2$ に対する $(\tilde{a}_1 \pm \tilde{a}_2)\tau^A$ で最小の $A \in \hat{\mathbb{Z}}$ は，$A \leq 3 + \max(A_1 + \mathrm{ilog}_2(|\tilde{a}_1|/|\tilde{a}_1 \pm \tilde{a}_2|),\ A_2 + \mathrm{ilog}_2(|\tilde{a}_2|/|\tilde{a}_1 \pm \tilde{a}_2|))$ を満たす。 ◁

この命題に基づいて，Liang は次の演算を $\mathbb{FT}_\ell$ に定義しています。

定義 7 ($\mathbb{FT}_\ell$ 上の四則演算)
$a_1, a_2 \in \mathbb{R} \setminus \{0\}$ を表す $\tilde{a}_1\tau^{A_1}, \tilde{a}_2\tau^{A_2} \in \mathbb{FT}_\ell$ $(A_1, A_2 \leq \ell - 3)$ に対し，以下の演算を定義する。

1. $a_1a_2 = (\tilde{a}_1\tilde{a}_2)\tau^A$, $a_1/a_2 = (\tilde{a}_1/\tilde{a}_2)\tau^A$ $(A = \max(A_1, A_2))$
2. $\tilde{a}_1\tau^{A_1} \pm \tilde{a}_2\tau^{A_2}$ が，仮定 5 の下でノンゼロならば，
   $\tilde{a}_1\tau^{A_1} \pm \tilde{a}_2\tau^{A_2} = (\tilde{a}_1 \pm \tilde{a}_2)\tau^A$ $(A = \max(A_1 + \mathrm{ilog}_2(\sqrt{2}|\tilde{a}_1|/|\tilde{a}_1 \pm \tilde{a}_2|),\ A_2 + \mathrm{ilog}_2(\sqrt{2}|\tilde{a}_2|/|\tilde{a}_1 \pm \tilde{a}_2|)))$

◁

$\mathbb{FT}_\ell$ 上の四則演算を定義したことにより，Gröbner 基底計算を最後まで行うことで，個々の基底多項式で失われた桁を見積もることが可能になります。ところが，この方法はうまくいくこともありますが，往々にして過大な評価を与えてしまうこともあります（区間数などと同様の理由と考えられます）。原論文で考察されている内容については，少し議論の余地があるように思えますが，結論については，佐々木ら [7] と同じく，個々の係数で失われる桁数は独立ではなく相互に関係しており，それを考慮した見積りが必要と書かれています。そのため，より深く桁精度が失われることを調べるための空間として PL 空間を定義していくのが本節の目的となっています。

以下，$f = a_1t_1 + \cdots + a_st_s \in \mathbb{R}[\vec{x}]$ $(a_i \neq 0)$ に対する様々な τ-representation $f_\tau = \tilde{a}_1\tau^{A_1}t_1 + \cdots + \tilde{a}_s\tau^{A_s}t_s$ を，$(A_1, \ldots, A_s) \in \hat{\mathbb{Z}}^s$ で表すことにします。このとき，与えられた $\vec{u} \in \hat{\mathbb{Z}}^s$ の i 番目を基準に，逸失桁精度を上書きする操作 $\vartheta(\vec{u}, i) = (A'_1, \ldots, A'_s)$ を，$A'_i = -\infty$, $A'_j = \max(\vec{u}_i, \vec{u}_j)$ $(j \neq i)$ と定義します。即ち，i 番目の係数は $(\tilde{a}_i/a_i)f$ の i 番目の係数と厳密な意味で等しくなります。加えて，集合 $H \in \hat{\mathbb{Z}}^s$ が与えられたときに，各係数で逸失桁精度下限（つまり，桁精度上限）を取った要素を $\inf_\rtimes(H)$ で表します。つまり，$\inf_\rtimes(H) = (A^*_1, \ldots, A^*_s)$ ならば，$A^*_i = \inf_{\vec{h} \in H}\{\vec{h}_i\}$ $(i = 1, \ldots, s)$ です。

定義 8 (Comparable set)
$H \subset \hat{\mathbb{Z}}^s$ を空集合でないとする。このとき，$\vec{h} \in H$, $\vec{h}_k = -\infty$ ならば，H は **k-comparable** という。また，$U \subset \hat{\mathbb{Z}}^s$ に対して，$CP^k_s(U) \subset U$ を，$CP^k_s(U) = \{\, \vec{u} \in U \mid \vec{u} \textit{ is } k\textit{-comparable} \,\}$ と定義する。 ◁

命題 9
$\vec{u}, \vec{v}, \vec{w} \in \hat{\mathbb{Z}}^s$ に対して，次が成立する。

- $\vec{u} \rtimes \vec{v},\ \vec{v} \rtimes \vec{w} \Longrightarrow \vec{u} \rtimes \vec{w}$
- $\exists i \in \{1, \ldots, s\},\ \vec{u} \rtimes \vartheta(\vec{u}, i)$
- $\vec{u} \rtimes \vec{v} \Longrightarrow \forall i \in \{1, \ldots, s\},\ \vartheta(\vec{u}, i) \rtimes \vartheta(\vec{v}, i)$
- $\forall i, j \in \{1, \ldots, s\},\ \vartheta(\vartheta(\vec{u}, i), j) \rtimes \vartheta(\vec{u}, j)$
- $\vec{u} \rtimes \vec{v},\ \vec{v} \rtimes \vec{u} \Longrightarrow \vec{u} = \vec{v}$ ◁

この系として,「$\rtimes$ は，$\hat{\mathbb{Z}}^s$ における半順序」であることが分かります（全順序でないことも明らか)。

定義 10 (Precision loss space（PL 空間，逸失桁精度空間）)
空でない $\mathcal{P} \subset \hat{\mathbb{Z}}^s$ が，次を満たすとき，$\mathcal{P}$ を **Precision loss space（PL 空間，逸失桁精度空間）**という。また，$F \subset \hat{\mathbb{Z}}^s$ を含む最小の PL 空間を，F で生成される **PL 空間**といい，$\langle F \rangle$ で表す。

1. $\vec{u} \in \mathcal{P},\ \vec{v} \rtimes \vec{u} \Longrightarrow \vec{v} \in \mathcal{P}$ （逸失桁精度がより大きいものは全て含む）
2. $\vec{u} \in \mathcal{P} \Longrightarrow \forall i \in \{1, \ldots, s\},\ \vartheta(\vec{u}, i) \in \mathcal{P}$ （1 点を無限精度にしたものも含む）
3. $i \in \{1, \ldots, s\},\ H \in CP^i_s(\mathcal{P}) \Longrightarrow \inf_{\rtimes}(H) \in \mathcal{P}$
（同じ 1 点で無限精度の要素集合に対し，その逸失桁精度最小の組み合わせを含む）

◁

原論文においては，条件 (1) は弱い基底の存在性，条件 (2) を加えることにより強い基底の存在性，そして，条件 (3) を加えることにより PL 空間の複雑さの評価を可能にしていると解説されていますが，この時点において，Gröbner 基底計算における逸失桁精度との関係について議論は行われていません。

定義 11 (Weak basis（弱い基底）)
PL 空間 $\mathcal{P} \subset \hat{\mathbb{Z}}^s$ に対し，$B \subset \mathcal{P}$ が $\mathcal{P} = \langle B \rangle_w := \left\{ \vec{u} \in \hat{\mathbb{Z}}^s \,\middle|\, \vec{u} \rtimes \vec{v},\ \vec{v} \in B \right\}$ を満たすならば，B を $\mathcal{P}$ の **Weak basis（弱い基底）**，または，$\mathcal{P}$ は B により **Weakly generated（弱く生成される）**という。 ◁

定義 12 (Strong basis（強い基底）)
PL 空間 $\mathcal{P} \subset \hat{\mathbb{Z}}^k$ に対し，$B \subset \mathcal{P}$ が $\mathcal{P} = \langle B \rangle_s := \left\{ \vec{u} \in \hat{\mathbb{Z}}^k \,\middle|\, \vec{u} \rtimes \vartheta(\vec{v}, i),\ \vec{v} \in B,\ i \in \{1, \ldots, k\} \right\}$ を満たすならば，B を $\mathcal{P}$ の **Strong basis（強い基底）**，または，$\mathcal{P}$ は B により **Strongly generated（強く生成される）**という。 ◁

任意の部分集合 $B \subset \hat{\mathbb{Z}}^s$ は，必ずしも PL 空間を生成しませんが，次のような性質を持っています。

命題 13
空でない $A, B \subset \hat{\mathbb{Z}}^s$ に対し，次が成り立つ。

- $B \subset \langle B \rangle_w \subset \langle B \rangle_s \subset \langle B \rangle$
- $\langle \langle B \rangle_w \rangle_w = \langle B \rangle_w,\ \langle \langle B \rangle_s \rangle_s = \langle B \rangle_s,\ \langle \langle B \rangle \rangle = \langle B \rangle$
- $A \subset B \Longrightarrow \langle A \rangle_w \subset \langle B \rangle_w,\ \langle A \rangle_s \subset \langle B \rangle_s,\ \langle A \rangle \subset \langle B \rangle$ ◁

定理 14
PL 空間は,（有限の）弱い基底と（有限の）強い基底を持つ。 ◁

この定理は構成的に証明されています。実際の基底をどのように構成するかについて，十分な理解をするために，弱い基底の構成方法とそこから強い基底を構成する方法について述べておきます（原論文では，この定理とその系の証明において記述されている内容に相当します）。

1. $\langle F \rangle = \mathcal{P} \subset \hat{\mathbb{Z}}^s$ なる生成系 F を取る（最悪，$F = \mathcal{P}$ とすれば良い）。
2. i-comparable な $H_i = \{ \vartheta(\vec{u}, i) \mid \vec{u} \in F \}$ $(i = 1, \ldots, s)$ を構成する（PL 空間の性質から，$\vartheta(\vec{u}, i) \in \mathcal{P}$）。
3. 逸失桁精度最小の元 $\vec{u}_i^{(1)} = \inf_{\rtimes}(H_i)$ を構成する（PL 空間の性質から，$\vec{u}_i^{(1)} = \inf_{\rtimes}(H_i) \in \mathcal{P}$）。
4. $F^{(1)} = \left\{ \vec{u}_i^{(1)} \,\middle|\, i = 1, \ldots, s \right\}$ とし，行ベクトルが $\vec{u}_i^{(1)}$ である (s, s) 行列 $A^{(1)}$ を構成する。
5. $A^{(k^*)} = A^{(k^*+1)}$ となるまで以下の構成を繰り返す。
   (a) 逸失桁精度最小の元 $\vec{u}_i^{(k+1)} = \inf_{\rtimes} \left(\left\{ \vartheta(\vec{u}_e^{(k)}, i) \,\middle|\, \vec{u}_e^{(k)} \in F^{(k)},\ e = 1, \ldots, s \right\} \right)$ を構成する。
   (b) $F^{(k+1)} = \left\{ \vec{u}_i^{(k+1)} \,\middle|\, i = 1, \ldots, s \right\}$ とし，行ベクトルが $\vec{u}_i^{(k+1)}$ である行列 $A^{(k+1)}$ を構成する。
6. 結果，$F \subset \langle F^{(1)} \rangle_w \subset \langle F^{(2)} \rangle_w p \subset \cdots \subset \langle B \rangle_w = \mathcal{P}$ となり，$B = F^{(k^*)}$ が得られる。

最後に，PL 空間の複雑度を強い基底に基づいて定義するという議論が行われています。

定義 15 (Dependence numbers（依存数）)
$\mathcal{P}$ を *PL* 空間とするとき，以下でその **Dependence number**（依存数）を定義する。

$$Dpn(\mathcal{P}) := \begin{cases} \min \{ \#B \mid \langle B \rangle_s = \mathcal{P} \}, & when\ \mathcal{P} \neq \hat{\mathbb{Z}}^s; \\ 0, & when\ \mathcal{P} = \hat{\mathbb{Z}}^s. \end{cases}$$

◁

なお，有限集合の強い基底について存在性が示されていますので，$Dpn(\mathcal{P}) < \infty$ です。問題としては，一意でないことが分かっている強い基底を用いて定義されているため，どのように依存数を計算すれば良いかが残されます。原論文では，最小の強い基底という概念を導入することで計算を可能としています。

定義 16 (Minimal strong basis（最小の強い基底）)
PL 空間 $\mathcal{P}$ とその強い基底 B に対し，B のどの真部分集合も $\mathcal{P}$ の強い基底を構成しないとき，B を $\mathcal{P}$ の **Minimal strong basis**（最小の強い基底）という。 ◁

最小の強い基底も一意には定まりませんが，原論文では次の性質が示されています。

定理 17
PL 空間 $\mathcal{P} \subset \hat{\mathbb{Z}}^s$ に対し，$\mathcal{P}$ の強い基底 B が最小であるための必要十分条件は，$\mathcal{P} = \hat{\mathbb{Z}}^s$ のとき $\#B = 1$ であり，$\mathcal{P} \neq \hat{\mathbb{Z}}^s$ のとき $\#B = Dpn(\mathcal{P})$ である。従って，p 最小の強い基底は必ず同じ要素数からなる。 ◁

定理 18
PL 空間 $\mathcal{P} = \langle F \rangle \subset \hat{\mathbb{Z}}^s$ に対し，$Dpn(\mathcal{P}) \leq \lfloor s/2 \rfloor$ が成り立つ（$\lfloor \cdot \rfloor$ は，値を越えない最大の整数）。 ◁

3.1 考察

PL 空間の性質については，τ-representation にのみ基づいており，その計算方法に依存してないように思われます。そのため，既存の誤差追跡方法にも適用できる可能性があります。既存の方法との違いという点では，特に Mathematica や efloat と比べて本質的な違いがあるか分かりません。区間数におけるゼロ判

定方法と異なり，Mathematica や efloat の場合には，τ-representation の判定方法と同じく，有効桁が不十分なときに判定不能とすることも可能でしょう。

τ-representation は，同一の多項式の係数間における逸失桁精度を表現するものです。個々の演算による新たな逸失桁精度については，導入した四則演算方法に基づいて逐次計算をする必要があります。しかしながら，これを繰り返すだけでは，原論文で Liang も書いていますが，過剰な見積もりになりやすく効果的な値を得ることが出来ません。いまのところ，τ-representation に基づき仮定を導入することで，いくつかの性質を有する PL 空間というのを考えることが出来るようになることしか分かりません。これが，本来の目的である数値的な計算方法で必要とされる桁精度の見積もりにつながるのかも分かりません。

4 まとめ

Gröbner 基底を数値的に計算する際に必要となる桁精度の見積もりは非常に重要と考えられます。そのような意味で，最近の Liang による研究成果をサーベイし，講演に向けて取り組んでいる内容を簡単にまとめました。Liang による情報落ち（丸め誤差）の論文は，因果関係について研究の余地があり，桁落ち誤差の論文は，さらなる一般化という方向と実際の逸失桁精度の見積もりへの応用という研究の余地がありそうだということが分かりました。講演においては，これらの方向からの研究結果について続報を発表したいと考えています。

参 考 文 献

[1] J. C. Faugère and Y. Liang. Artificial discontinuities of single-parametric grobner bases. *Journal of Symbolic Computation*, 46(4):459 – 466, 2011.

[2] C. Traverso and A. Zanoni. Numerical stability and stabilization of groebner basis computation. In *ISSAC 2002: Proceedings of the 2002 international symposium on Symbolic and algebraic computation*, pages 262–269, New York, NY, USA, 2002. ACM.

[3] Y. Liang. Structures of precision losses in computing approximate gröbner bases. *Journal of Symbolic Computation*, 53:81 – 95, 2013.

[4] F. Kako and T. Sasaki. Proposal of “effective” floating-point number. Preprint of Univ. Tsukuba, May 1997 (unpublished).

[5] Y. Liang. Selecting lengths of floats for the computation of approximate gröbner bases. *Journal of Symbolic Computation*, 53:40 – 52, 2013.

[6] T. Sasaki and F. Kako. Term cancellations in computing floating-point gröbner bases. In *Proceedings of CASC 2010*, volume 6244 of *Lecture Notes in Comput. Sci.*, pages 220–231, Berlin, 2010. Springer.

[7] T. Sasaki and F. Kako. Floating-point gröbner basis computation with ill-conditionedness estimation. In *Proceedings of ASCM 2007*, volume 5081 of *Lecture Notes in Comput. Sci.*, pages 278–292. Springer, Berlin, 2008.

[8] T. Sasaki and F. Kako. Computing floating-point gröbner bases stably. In *Proceedings of SNC 2007*, pages 180–189. ACM, New York, 2007.

[9] K. Shirayanagi and M. Sweedler. A theory of stabilizing algebraic algorithms. *Technical Report 95-28*, pages 1–92, 1995. http://www.ss.u-tokai.ac.jp/s̃hirayan/msitr95-28.pdf.

セッション７

Session 7

線形代数，代数方程式

Linear algebra and algebraic equations

最小消去多項式候補を用いた行列の一般固有空間の構造の計算法について

On determining the structure of the invariant space of matrices via pseudo-annihilating polynomials.

小原功任　　田島慎一
Katsuyoshi OHARA　　Shinichi TAJIMA
金沢大学・理工*　　筑波大学・数理物質†

Abstract

The theory of the invariant space and Jordan canonical form of square matrices is well-known in linear algebra. However, the famous method for computing Jordan canonical forms is not efficient in computer algebra systems. In recent studies, we showed that fast algorithms in computer linear algebra can be given via minimal annihilating polynomials. In this paper, new method is developed for determining "the structure of the invariant space" via minimal and pseudo-annihilating polynomials.

1　はじめに

体 $K = \mathbf{Q}$ 上の n 次正方行列 A を考えよう．いま，行列 A のある固有値 λ に注目する．λ の定義多項式 $f(x)$ は行列 A の特性多項式 $\chi_A(x)$ の既約因子となっている．体 K を代数拡大して，A をジョルダン標準形で表したとすれば，固有値 λ に対応するジョルダン細胞が，

$$J_{k_1}(\lambda) \text{ が } n_1 \text{ 個}, J_{k_2}(\lambda) \text{ が } n_2 \text{ 個}, \cdots, J_{k_m}(\lambda) \text{ が } n_m \text{ 個}, \quad (k_1 > k_2 > \cdots > k_m \geq 1)$$

と現れるはずである ($J_k(\lambda)$ は固有値 λ に関する，大きさ k のジョルダン細胞を表す)．われわれが知りたい情報はジョルダン細胞の大きさと個数の列 $(k_1, n_1), \cdots, (k_m, n_m)$ である．これを本稿では "一般固有空間の構造" と呼ぶことにする．本研究の目的は，行列 A の一般固有空間の構造を具体的に決定するためのアルゴリズムを与えることである．

よく知られているように，線形代数学では，行列の相似変形によってジョルダン標準形 (またはジョルダン細胞) が得ることできる．しかしながら，相似変形によってジョルダン細胞を得るのは，計算量的には効率がよいとは言えない．本研究では，変形理論によらずに一般固有空間の構造を決定することを論じる．

2　最小消去多項式とその候補

正方行列 A の最小多項式 $\pi_A(x)$ は，どのような列ベクトル $\boldsymbol{u} \in K^n$ に対しても，$\pi_A(A)\boldsymbol{u} = \mathbf{0}$ を満たす．逆に，ある列ベクトル $\boldsymbol{u} \in K^n$ が与えられたとき，$h(A)\boldsymbol{u} = \mathbf{0}$ となる次数最小のモニック多項式

*ohara@air.s.kanazawa-u.ac.jp
†tajima@math.tsukuba.ac.jp

$h(x) \in K[x]$ をベクトル $\boldsymbol{u}$ に関する A の最小消去多項式と呼び，$\pi_{A,\boldsymbol{u}}(x)$ で表わす．$\pi_{A,\boldsymbol{u}}(x)$ は A の最小多項式 $\pi_A(x)$ を割り切る．また，すべての基本ベクトルに関する最小消去多項式の最小公倍多項式は最小多項式と一致する．基本ベクトル $\boldsymbol{e}_j$ に関する最小消去多項式は固有ベクトル計算やスペクトル分解計算に威力を発揮する ([3], [5], [6])．

いま，行ベクトル ${}^t\boldsymbol{v} \in K^n$ をランダムに選び，${}^t\boldsymbol{v}h(A)\boldsymbol{u} = 0$ かつ $h(x) \,|\, \chi_A(x)$ を満たす次数最小のモニック多項式 $h(x) \in K[x]$ 最小消去多項式候補と呼び，$\tilde{\pi}_{A,\boldsymbol{u}}(x)$ で表すことにする．最小消去多項式候補の探索とは，すなわち最小消去多項式の確率的計算アルゴリズムである．定義から分かるように，最小消去多項式を計算するには，$\pi_{A,\boldsymbol{u}}(A)\boldsymbol{u}$ が零ベクトルに等しいことを確認する必要がある．一方，最小消去多項式候補の探索では，${}^t\boldsymbol{v}\tilde{\pi}_{A,\boldsymbol{u}}(A)\boldsymbol{u}$ の値について調べるが，これはスカラー値であるので，特に行列の次数が大きい場合には計算量の観点からは有利になる．また，${}^t\boldsymbol{v}$ が乱数ベクトルであるので，ほとんどの場合，最小消去多項式候補は最小消去多項式に一致する．本稿では，最小消去多項式候補および最小消去多項式の計算アルゴリズムについて概説する．

まず，行列 A の特性多項式の既約分解が $\chi_A(x) = f_1(x)^{m_1} \cdots f_q(x)^{m_q}$, $(m_i \in \mathbf{N})$ で与えられているとしよう．最小消去多項式は特性多項式を割りきるので，$\pi_{A,\boldsymbol{u}}(x)$ は必ず，

$$\pi_{A,\boldsymbol{u}}(x) = f_1(x)^{\ell_1} \cdots f_q(x)^{\ell_q}, \quad (0 \le \ell_i \le m_i)$$

の形になる．したがって，最小消去多項式候補も同じく，

$$\tilde{\pi}_{A,\boldsymbol{u}}(x) = f_1(x)^{\ell'_1} \cdots f_q(x)^{\ell'_q}, \quad (0 \le \ell'_i \le m_i)$$

の形を仮定してよい．最小消去多項式候補の指数 $\ell'_1, \cdots, \ell'_q$ を次の手順で決定しよう．

アルゴリズム 1 (最小消去多項式候補)．
入力: 行列 A, 特性多項式 $\chi_A(x) = \prod_{i=1}^{m} f_i(x)^{m_i}$
出力: 最小消去多項式候補 $\tilde{\pi}_{A,\boldsymbol{u}}(x)$

1. 行ベクトル ${}^t\boldsymbol{v} \in K^n$ をランダムに選ぶ．
2. for $i = 1, \cdots, q$, do
   $\chi_i(x) \leftarrow \chi(x)/f_i(x)^{m_i}$, ${}^t\boldsymbol{v}_i \leftarrow {}^t\boldsymbol{v}\chi_i(A)$.
3. for $i = 1, \cdots, q$, do
$$\ell'_i \leftarrow \begin{cases} \min\{k \in \mathbf{N} \mid {}^t\boldsymbol{v}_i f_i(A)^k \boldsymbol{u} = 0\}, & ({}^t\boldsymbol{v}_i\boldsymbol{u} \neq 0) \\ 0. & ({}^t\boldsymbol{v}_i\boldsymbol{u} = 0) \end{cases}$$
4. $\tilde{\pi}_{A,\boldsymbol{u}}(x) \leftarrow \prod_i f_i(x)^{\ell'_i}$.

ステップ 2 では，すべての ${}^t\boldsymbol{v}_1, \cdots, {}^t\boldsymbol{v}_q$ を求めるには，行列多項式乗算 ${}^t\boldsymbol{w} \mapsto {}^t\boldsymbol{w}f_i(A)^{m_i}$ が q^2 回必要であるように思えるが，2 分探索の方法を用いれば，およそ $q \log_2 q$ 回で実行可能である (行列多項式乗算の高速算法については，[7] 参照)．さらに，${}^t\boldsymbol{v}_k$ は一度計算しておけば，どの基本ベクトル $\boldsymbol{e}_j$ に関する最小消去多項式候補 $\tilde{\pi}_{A,\boldsymbol{e}_j}(x)$ の計算でも共通に利用できることにも注意する．

さて，アルゴリズム 1 で定まる最小消去多項式候補 $\tilde{\pi}_{A,u}(x)$ は次の性質を持つ．

補題 1. 各既約因子の指数について，$\ell'_i \le \ell_i \le m_i$ が成り立つ．また，$\boldsymbol{u}' = \tilde{\pi}_{A,\boldsymbol{u}}(A)\boldsymbol{u}$ について，$\pi_{A,\boldsymbol{u}}(x) = \tilde{\pi}_{A,\boldsymbol{u}}(x)\pi_{A,\boldsymbol{u}'}(x)$ を満たす．

上の補題から，最小消去多項式候補の指数は，最小消去多項式の指数 ℓ_i の下からの評価を与えていることが分かる．しかも ${}^t\boldsymbol{v}$ が乱数ベクトルであることに注意すると，ほぼ確率 1 で $\ell'_i = \ell_i$ となる．したがって，この評価は非常に効率的である．

しかしながら，一般には最小消去多項式候補は最小消去多項式と異なる可能性がある．そのため候補ではなく真の最小消去多項式が必要な場合には比較・検証し，異なる場合には最小消去多項式を決定しなければならない．その方法を述べよう．

まず $\boldsymbol{u}' = \tilde{\pi}_{A,\boldsymbol{u}}(A)\boldsymbol{u}$ を求める．$\boldsymbol{u}' = \boldsymbol{0}$ ならば，$\tilde{\pi}_{A,\boldsymbol{u}}(x) = \pi_{A,\boldsymbol{u}}(x)$ であることが分かる．一方，$\boldsymbol{u}' \neq \boldsymbol{0}$ の場合は，上の補題から $\pi_{A,\boldsymbol{u}}(x) = \pi_{A,\boldsymbol{u}'}(x)\tilde{\pi}_{A,\boldsymbol{u}}(x)$ となるので，$\boldsymbol{u}'$ に関する最小消去多項式 $\pi_{A,\boldsymbol{u}'}(x)$ を求めればよい．ここで，$\pi_{A,\boldsymbol{u}'}(x)$ の各指数 $\ell_i - \ell'_i$ は $m_i - \ell'_i$ で上から抑えられている．このことに注意して，$\boldsymbol{u}'$ についてアルゴリズム 1 を適用することで，u' に関する最小消去多項式候補 $\tilde{\pi}_{A,\boldsymbol{u}'}$ を計算することができる．これを順次繰り返して最小消去多項式が得られる．

3　一般固有空間の構造の決定法 (最小消去多項式による場合)

いま，λ を行列 A のある固有値，$f(x)$ を λ の定義多項式とする．前述したように，固有値 λ に対応するジョルダン細胞のサイズと個数の列

$$J_{k_1}(\lambda) \text{ が } n_1 \text{ 個}, J_{k_2}(\lambda) \text{ が } n_2 \text{ 個}, \cdots, J_{k_m}(\lambda) \text{ が } n_m \text{ 個}, \quad (k_1 > k_2 > \cdots > k_m \geq 1)$$

が知りたいものである．k_1 については，最小多項式 $\pi_A(x)$ を既約分解したときに現れる $f(x)$ の指数と一致することが容易に分かる．では，n_1 はどのようにすれば知ることができるだろうか．そのためにジョルダン細胞と最小消去多項式の関係を考えよう．

いま，ある列ベクトル $\boldsymbol{u} \in K^n$ に関して最小消去多項式が $\pi_{A,\boldsymbol{u}}(x) = f(x)^\ell g(x)$ と表されたとする．このとき $\boldsymbol{p} = g(A)\boldsymbol{u} \in K^n$ は，

$$f(A)^\ell \boldsymbol{p} = f(A)^\ell g(A)\boldsymbol{u} = \pi_{A,\boldsymbol{u}}(A)\boldsymbol{u} = \boldsymbol{0}$$

を満たすことから，$\boldsymbol{p}$ の最小消去多項式は $\pi_{A,\boldsymbol{p}}(x) = f(x)^\ell$ である．また

$$\Psi(x,y) = \frac{f(x) - f(y)}{x - y} \in K[x,y]$$

と置くと，$f(A) = (A - \lambda E)\Psi(A, \lambda E)$ なので，$\boldsymbol{v} = \Psi(A,\lambda E)^\ell \boldsymbol{p} \in K(\lambda)^n$ について，$f(A)^\ell \boldsymbol{p} = \boldsymbol{0}$ と $(A - \lambda E)^\ell \boldsymbol{v} = \boldsymbol{0}$ が同値であることが分かる．しかも $\pi_{A,\boldsymbol{p}}(x)$ の最小性より，$(A - \lambda E)^{\ell-1}\boldsymbol{v} \neq \boldsymbol{0}$ である．すなわち，

$$(A - \lambda E)^\ell \boldsymbol{v} = 0 \quad \text{かつ} \quad (A - \lambda E)^k \boldsymbol{v} \neq 0 \ \ (0 \leq k < \ell)$$

となるので，$\boldsymbol{v}$ はサイズ ℓ のジョルダン細胞に属することが分かる．

すべての基本ベクトル $\boldsymbol{e}_j \ \ (1 \leq j \leq n)$ について，最小消去多項式が分かっているとし，$\pi_{A,\boldsymbol{e}_j}(x) = f(x)^{\ell_j} g_j(x)$ と表すこととする．また，$f(x)$ に関する指数の最大値を $\ell = \max(\ell_1, \cdots, \ell_n)$ とする．

われわれがまず知りたいのはサイズ $k_1 = \ell$ のジョルダン細胞の個数 n_1 である．そこで，最小消去多項式の $f(x)$ に関する指数がちょうど k となる基本ベクトルの添字の集合 $J_k = \{j \mid \ell_j = k\}$ を考える．$\boldsymbol{p}_j = g_j(A)\boldsymbol{e}_j$ および $\boldsymbol{v}_j = \Psi(A,\lambda E)^{\ell_j}\boldsymbol{p}_j$, と置こう．ベクトル $\boldsymbol{p}_j \ (j \in J_\ell)$ の最小消去多項式はすべて $f(x)^\ell$ である．また，$\Psi(x,y)$ の定義から

$$(A - \lambda E)^{\ell-1}\boldsymbol{v}_j = \Psi(A,\lambda E) f(A)^{\ell-1}\boldsymbol{p}_j$$

が成り立つことが分かる．したがって，$\Psi(A,\lambda E)$ を左からかけるという線形写像によって，

$$\Psi(A,\lambda E) : \operatorname{span}\{f(A)^{\ell-1}\boldsymbol{p}_j \mid j \in J_\ell\} \to \operatorname{span}\{(A-\lambda E)^{\ell-1}\boldsymbol{v}_j \mid j \in J_\ell\}$$

は全射である．ここで $\operatorname{span} S$ で，集合 S の生成する K-ベクトル空間を表す．$\operatorname{span}\{f(A)^{\ell-1}\boldsymbol{p}_j \mid j \in J_\ell\}$ の次元 r_ℓ は容易に計算できて，

$$r_\ell = \dim_K \operatorname{span}\{f(A)^{\ell-1}\boldsymbol{p}_j \mid j \in J_\ell\} = \operatorname{rank}([\, f(A)^{\ell-1}\boldsymbol{p}_j \,]_{j\in J_\ell}) = \operatorname{rank}(f(A)^{\ell-1}[\, \boldsymbol{p}_j \,]_{j\in J_\ell}).$$

ここで，$[\, \boldsymbol{p}_j \,]_{j\in J_\ell}$ は列ベクトル $\boldsymbol{p}_j$ たちを並べた行列を表す．$f(x)$ の共役な根を区別しないことに注意すれば，

$$r_\ell = (\text{サイズ } \ell \text{ のジョルダン細胞の個数}) \times \deg f(x)$$

であることが分かる．

次に，サイズ $\ell-1$ のジョルダン細胞の個数について調べたい．$|J_\ell| \times n$ 行列 $[\, f(A)^{\ell-1}\boldsymbol{p}_j \,]_{j\in J_\ell}$ のランク r_ℓ が，$r_\ell < |J_\ell|$ であるときには，一次結合 $\boldsymbol{p} = \sum_{j\in J_\ell} a_j \boldsymbol{p}_j$ でかつ最小消去多項式が $f(x)^{\ell'}$ $(\ell' < \ell)$ なるベクトルが存在する．したがって，サイズの小さいジョルダン細胞の個数を求めるときには，このようなベクトルも考慮しなくてはならない．このことに注意して調べていこう．

$r_\ell = \operatorname{rank}([\, f(A)^{\ell-1}\boldsymbol{p}_j \,]_{j\in J_\ell})$ であったので，ベクトルの集合 $\{f(A)^{\ell-1}\boldsymbol{p}_j \mid j \in J_\ell\}$ には $|J_\ell| - r_\ell$ 個の一次従属関係式がある．つまり，ある定数 $c_{kj} \in K$ によって

$$\sum_{j\in J_\ell} c_{kj} f(A)^{\ell-1}\boldsymbol{p}_j = \boldsymbol{0}, \qquad 0 \le k < |J_\ell| - r_\ell$$

と表せ，しかも c_{kj} は行列 $[\, f(A)^{\ell-1}\boldsymbol{p}_j \,]_{j\in J_\ell}$ の掃き出しによって，ランク計算と同時に得られる．このとき，ベクトル $\boldsymbol{w}_k = \sum_{j\in J} c_{kj}\boldsymbol{p}_j$ の最小消去多項式は $f(x)^{\ell-1}$ である．まとめると，集合 $U = \{\boldsymbol{p}_j \mid j \in J_{\ell-1}\} \cup \{\boldsymbol{w}_k \mid 0 \le k < |J_\ell| - r_\ell\}$ に属するベクトルはすべて最小消去多項式 $f(x)^{\ell-1}$ をもつ．

よって $\boldsymbol{u} \in U$ について，$\boldsymbol{v} = \Psi(A,\lambda E)^{\ell-1}\boldsymbol{u}$ と置くと，前述したように，$f(A)^{\ell-1}\boldsymbol{u} = \boldsymbol{0}$ と $(A-\lambda E)^{\ell-1}\boldsymbol{v} = \boldsymbol{0}$ と同値である．このとき

$$(A-\lambda E)^{\ell-2}\boldsymbol{v} = \Psi(A,\lambda E) f(A)^{\ell-2}\boldsymbol{u}$$

が成り立つ．以下，サイズ $\ell-1$ のときも同じ議論ができて，全射

$$\Psi(A,\lambda E) : \operatorname{span}\{f(A)^{\ell-2}\boldsymbol{u} \mid \boldsymbol{u} \in U\} \to \operatorname{span}\{(A-\lambda E)^{\ell-2}\boldsymbol{v} \mid (A-\lambda E)^{\ell-1}\boldsymbol{v} = \boldsymbol{0}\}$$

が得られる．$r_{\ell-1} = \operatorname{rank}(f(A)^{\ell-2} \cdot [\, \boldsymbol{u} \,]_{\boldsymbol{u}\in U})$ としよう．

$(A-\lambda E)^{\ell-1}\boldsymbol{v} = \boldsymbol{0}$ ならば $(A-\lambda E)^{\ell}\boldsymbol{v} = \boldsymbol{0}$ であることに注意すると，

$$r_{\ell-1} - r_\ell = (\text{サイズ } \ell-1 \text{ のジョルダン細胞の個数}) \times \deg f(x)$$

である．

よって次のアルゴリズムが得られる．

アルゴリズム **2** (ある固有値に関する一般固有空間の構造)．
入力: 行列 A, 特性多項式 $\chi_A(x)$, 固有値の定義多項式 $f(x)$
出力: ジョルダン細胞列 S_f

1. $\{\pi_{A,\boldsymbol{e}_1}(x), \cdots, \pi_{A,\boldsymbol{e}_n}(x)\}\leftarrow$ 最小消去多項式

2. for $j = 1, \cdots, n$, do
   $\ell_j \leftarrow (\pi_{A,\boldsymbol{e}_j}(x)$ の $f(x)$ に関する指数$)$, $g_j(x) \leftarrow \pi_{A,\boldsymbol{e}_j}(x)/f(x)^{\ell_j}$, $\boldsymbol{p}_j \leftarrow g_j(A)\boldsymbol{e}_j$

3. $\ell \leftarrow \max\{\ell_1, \cdots, \ell_n\}$,
   for $k = 0, 1, \cdots, \ell$, do
   $J_k \leftarrow \{j \mid \ell_j = k\}$

4. $W_{\ell+1} \leftarrow \emptyset$, $r_{\ell+1} \leftarrow 0$,
   for $k = \ell, \ell-1, \cdots, 1$, do
   $U \leftarrow \{\boldsymbol{p}_j \mid j \in J_k\} \cup W_{k+1}$,
   $P \leftarrow (\boldsymbol{u})_{\boldsymbol{u}\in U}$: 行列,
   $r_k \leftarrow \operatorname{rank}(f(A)^{k-1}P)$,
   $W_k \leftarrow \{\boldsymbol{w} \mid \boldsymbol{w} = \sum_{\boldsymbol{u}\in U} c_{\boldsymbol{u}}\boldsymbol{u} \neq \boldsymbol{0},\ f(A)^{k-1}\boldsymbol{w} = \boldsymbol{0}\}$,
   $n_k \leftarrow (r_k - r_{k+1})/\deg f(x)$.

5. $S_f \leftarrow \{(k, n_k) \mid n_k \neq 0\}$.

なお，ステップ 4 では，ベクトルに $f(A)^i$ をかける計算が何度も行われているように見えるが，W_k の元が $\boldsymbol{p}_j$ たちの一次結合であることに注意すると，各 $\boldsymbol{p}_j$ に対し，$f(A)\boldsymbol{p}_j, \cdots, f(A)^{\ell_j-1}\boldsymbol{p}_j$ を一度だけ計算して記憶しておくことで，計算量を抑えることができる．

例 **1.**

$$A = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ -5 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ -3 & 0 & 0 & 0 & 0 & 1 \\ 6 & 3 & -25 & -10 & -11 & -2 \end{pmatrix}$$

ここで $f = x^2 + x + 5$ とすると，特性多項式・最小多項式は

$$\chi_A(x) = f^3, \quad \pi_A(x) = f^2$$

また，基本ベクトルに対する最小消去多項式は

$$\pi_{A,\boldsymbol{e}_1}(x) = f^2, \quad \pi_{A,\boldsymbol{e}_2}(x) = f, \quad \pi_{A,\boldsymbol{e}_3}(x) = \pi_{A,\boldsymbol{e}_4}(x) = \pi_{A,\boldsymbol{e}_5}(x) = \pi_{A,\boldsymbol{e}_6}(x) = f^2$$

最小消去多項式はわかったので，アルゴリズム 2 に基づいて計算してみると，$r_2 = 2$, $r_1 = 4$, $\deg(x^2 + x + 5) = 2$ となる．よって，サイズ 2 のジョルダン細胞は $r_2/2 = 1$ 個，サイズ 1 のジョルダン細胞は $(r_1 - r_2)/2 = 1$ 個，あることがわかる．

4　一般固有空間の構造の決定法 (最小消去多項式候補による場合)

ここでは，最小消去多項式を用いるかわりに，最小消去多項式候補を利用すること考える．前述したように最小消去多項式候補から最小消去多項式を構成するためには，$\pi_{A,\boldsymbol{u}}(A)\boldsymbol{u} = \boldsymbol{0}$ であるかの検証が不可欠となる．そこで，その検証にあたる手続きを一般固有空間の構造の決定とあわせて行うことで計算量の削減を試みる．

前節と同じように着目した固有値 λ の定義多項式を $f(x)$ とする．まず，アルゴリズム 1 により，すべての基本ベクトル $\boldsymbol{e}_j$ に対して，最小消去多項式候補 $\tilde{\pi}_{A,\boldsymbol{e}_j}(x) = f(x)^{\ell'_j} g'_j(x)$ を求める．$f(x)$ に関する指数が k と一致する基本ベクトルの添字の集合を $J'_k = \{j \mid \ell'_j = k\}$ と置く．$\boldsymbol{p}'_j = g'_j(A)\boldsymbol{e}_j$ が $f(A)^{\ell'_j}\boldsymbol{p}'_j = \boldsymbol{0}$ を満たせば，$\tilde{\pi}_{A,\boldsymbol{e}_j}(x) = \pi_{A,\boldsymbol{e}_j}(x)$ であることになるが，この段階では分からない．最小消去多項式候補の指数は，最小消去多項式の指数の下限を与えているので，$J'_k \subset J_k \cup J_{k+1} \cup \cdots \cup J_m$ の関係にある．前節のアルゴリズム 2 を見ると，正しい J_k と $\boldsymbol{p}_j$ を与えないと計算できない．したがって $\{J'_k\}$, $\{\boldsymbol{p}'_j\}$ から $\{J_k\}$, $\{\boldsymbol{p}_j\}$ を構成することが問題になる．

まず，$k = 0$ のときについて考える．$j \in J'_0$ に対し，$\boldsymbol{p}'_j = \boldsymbol{0}$ ならば，最小消去多項式候補は最小消去多項式に一致するので，$j \in J_0$ であることが分かる．また，$\boldsymbol{p}_j = \boldsymbol{p}'_j$ とする． しかしながら，$\boldsymbol{p}'_j \neq \boldsymbol{0}$ であったとしても，$f(x)$ 以外の因子の推定が誤っていることもあり得るので，$j \notin J'_0$ とは言えない．そこで，$\boldsymbol{p}'_j \neq \boldsymbol{0}$ のときには次のように考える．特性多項式 $\chi_A(x) = f(x)^m G(x)$ に着目して，$c_j(x) = G(x)/g_j(x)$ としよう．$\boldsymbol{p}_j = c_j(A)\boldsymbol{p}'_j$ とする．$c_j(A)\boldsymbol{p}'_j = G(A)\boldsymbol{e}_j$ は $f(A)$ 以外の全ての因子について左からかけたものであるから，もし，$\boldsymbol{p}_j = \boldsymbol{0}$ ならば，$g_j(x)$ の推定が誤っていたことになり，$j \in J_0$ である．$c_j(A)\boldsymbol{p}'_j \neq \boldsymbol{0}$ のときは，$f(x)$ の指数の推定が誤っているので，$f(A)^s\boldsymbol{p}_j$ $(s = 1, 2, \cdots)$ を順に求め，$f(A)^s\boldsymbol{p}_j = \boldsymbol{0}$ となる最小の s を探索する．このとき $j \in J_s$ である．

同様に $k > 0$ の場合を考える．$j \in J'_k$ に対し，$\boldsymbol{u}'_j = f(A)^{k-1}\boldsymbol{p}'_j$, $f(A)\boldsymbol{u}'_j$ をそれぞれ求める．$f(A)\boldsymbol{u}'_j = \boldsymbol{0}$ のときは，$\tilde{\pi}_{A,\boldsymbol{e}_j}(x) = \pi_{A,\boldsymbol{e}_j}(x)$ であるので，$j \in J_k$, $\boldsymbol{p}_j = \boldsymbol{p}'_j$ としてよい．$f(A)\boldsymbol{u}'_j \neq \boldsymbol{0}$ のときは，$k = 0$ のときと同様に，特性多項式から $c_j(x)$ を定め，$\boldsymbol{p}_j = c_j(A)\boldsymbol{p}'_j$, $\boldsymbol{u}_j = c_j(A)\boldsymbol{u}'_j$ とする．$f(A)\boldsymbol{u}_j = \boldsymbol{0}$ ならば，$f(x)$ の指数の推定は正しいので，$j \in J_k$ である．$f(A)\boldsymbol{u}_j \neq \boldsymbol{0}$ ならば，$f(A)^s(f(A)\boldsymbol{u}_j) = \boldsymbol{0}$ となる最小の s を求めれば，$j \in J_{k+s}$ となることが分かる．

以上より真の J_k と $\boldsymbol{p}_j$ が探索できたので，アルゴリズム 2 のステップ 4,5 を適用することで，一般固有空間の構造を求めることができる．また，この節で述べた J_k の探索アルゴリズムは明らかに各 j について並列化可能である．

参 考 文 献

[1] K. Ohara and S. Tajima: Spectral Decomposition and Eigenvectors of Matrices by Residue Calculus, Proceedings of the Joint Conference of ASCM 2009 and MACIS 2009, COE Lecture Note **22**, Kyushu University, 137–140.

[2] 田島慎一, 奈良洸平, 小原功任: 行列の最小多項式計算について, 京都大学数理解析研究所講究録 **1814**(2012), 1–8.

[3] 照井章, 田島慎一: 行列の最小消去多項式候補を利用した固有ベクトル計算, 京都大学数理解析研究所講究録 **1815**(2012), 13–20.

[4] 小原功任・田島慎一: レゾルベントを用いた行列のスペクトル分解と固有ベクトル計算, 日本数学会 2009 年度秋季総合分科会, 函数論分科会講演アブストラクト.

[5] 小原功任, 田島慎一: 最小消去多項式を用いた行列スペクトル分解計算の並列化, 京都大学数理解析研究所講究録 **1815**(2012), 21–28.

[6] 小原功任, 田島慎一: 最小消去多項式を用いた行列スペクトル分解の並列算法, 日本数学会 2011 年度秋季総合分科会, 函数論分科会講演アブストラクト.

[7] 小原功任, 田島慎一: 行列の最小消去多項式とその候補の計算法, 日本数学会 2013 年度年会, 代数学分科会講演アブストラクト.

行列の最小消去多項式候補を用いた固有ベクトル計算 (II)
Calculating eigenvectors of matrices using candidates for minimal annihilating polynomials II

田島 慎一*
Shinichi Tajima
筑波大学 数理物質系
Faculty of Pure and Applied Sciences, University of Tsukuba

照井　章†
Akira Terui
筑波大学 数理物質系
Faculty of Pure and Applied Sciences, University of Tsukuba

Abstract

Based on analysis of the residues of the resolvent, we have proposed an efficient algorithm for calculating eigenvector(s) of matrices. Our algorithm uses candidates for minimal annihilating polynomials, and the elements in eigenvector are represented as a polynomial in eigenvalue represented as a variable, thus we do not need to find eigenvalues by solving the characteristic equation. Whereas the previous algorithm calculates an eigenvector of the eigenvalue whose multiplicity is equal to one, the present algorithm extends the restriction such that we are now able to calculate eigenvector of eigenvalue whose multiplicity in the characteristic equation is greater than one under certain conditions.

1　はじめに

これまでに，我々は，レゾルベントの留数解析に基づき，行列の固有ベクトルを効率的に計算する算法を提案した [5]．我々の算法は，行列の最小消去多項式候補を用いるものであり，固有ベクトルの成分は固有値を変数とする多項式で表されるため，行列の特性多項式を解くことによる固有値の直接計算が不要であるという特徴をもつ．最小消去多項式候補の算法は，著者（田島）らによるレゾルベントの留数解析に基づく効率的な算法が提案されている [4]．また，我々は，この固有ベクトル算法を並列処理を用いて効率化する実装も提案している [3]．

本稿では，これまでに提案した固有ベクトル算法の拡張を提案する．これまでに提案した算法は，着目する固有値の重複度が 1 の場合に限られたが，本稿で提案する算法は，着目する固有値に属する一般固有ベクトル空間が，固有ベクトル空間に等しいという条件下で，着目する固有値の特性方程式における重複度が 1 よりも大きい場合にも固有ベクトルを計算可能にするものである．

*tajima@math.tsukuba.ac.jp
†terui@math.tsukuba.ac.jp

以下，本稿では次の内容を述べる．第 2 章では，問題設定および本稿における仮定と目的を説明する．第 3 章では，基本最小消去多項式候補がすべて真の基本最小消去多項式に等しいとあらかじめわかっている場合の固有ベクトルの算法を述べる．第 4 章では，基本最小消去多項式候補がすべて真の基本最小消去多項式に一致するかどうかが不明な場合の固有ベクトルの算法を述べる．この場合には “行列 Horner 法” を用いることにより，先に固有ベクトル候補を計算してから，基本最小消去多項式候補が真の最小消去多項式に一致するかを検査ことで，固有ベクトルをより効率的に計算可能にする．

2 問題設定

2.1 前置き (Preliminaries)

行列 A を有理数体 $\mathrm{K} = \mathbb{Q}$ 上の n 次正方行列とし，E_n を n 次単位行列とする．A の特性多項式 $\chi_A(\lambda)$ は次式の形で，整数上の既約因数分解があらかじめ求められているものとする．

$$\chi_A(\lambda) = f_1(\lambda)^{m_1} f_2(\lambda)^{m_2} \cdots f_p(\lambda)^{m_p} \cdots f_q(\lambda)^{m_q}. \tag{1}$$

本稿で提案するアルゴリズムの目的は，式 (1) のある既約因子 $f_p(\lambda)$ $(1 \le p \le q)$ に対し，$f_p(\alpha) = 0$ をみたす A の固有値 $\lambda = \alpha$ に属する固有ベクトルを求めることである．なお，本稿では $m_p \ge 1$ $(p = 1, \ldots, q)$ とする．

$\boldsymbol{e}_j = {}^t(0, \ldots, 0, 1, 0, \ldots, 0)$ を，第 j 成分が 1 に等しい n 次単位ベクトルとし，列のインデックスを $J = \{1, 2, \ldots, n\}$ とする．n 次列ベクトル $\boldsymbol{v}$ に対し，A における $\boldsymbol{v}$ の最小消去多項式 $p(\lambda)$ は，イデアル $\{p(\lambda) \mid p(A)\boldsymbol{v} = \boldsymbol{0}\}$ のモニックな生成元として定義される．A における $\boldsymbol{e}_j$ に対する最小消去多項式を $\pi_{A,j}(\lambda)$ とするとき，$\pi_{A,j}(\lambda)$ は

$$\pi_{A,j}(\lambda) = f_1(\lambda)^{l_{j,1}} f_2(\lambda)^{l_{j,2}} \cdots f_p(\lambda)^{l_{j,p}} \cdots f_q(\lambda)^{l_{j,q}}, \quad 0 \le l_{j,p} \le m_p, \quad j \in J \tag{2}$$

と表される．

本稿では，固有ベクトルの計算に $\boldsymbol{e}_j$ の “最小消去多項式候補” $\pi'_{A,j}(\lambda)$ を用いる．$\pi'_{A,j}(\lambda)$ は

$$\pi'_{A,j}(\lambda) = f_1(\lambda)^{l'_{j,1}} f_2(\lambda)^{l'_{j,2}} \cdots f_p(\lambda)^{l'_{j,p}} \cdots f_q(\lambda)^{l'_{j,q}} \tag{3}$$

と表される．ここに，我々の $\pi'_{A,j}(\lambda)$ の求め方より，$\pi'_{A,j}(\lambda)$ の各既約因子の多重度は $0 \le l'_{j,p} \le l_{j,p}$ を満たすことに注意する．

以下，$j \in J$ に対し，$\pi_{A,j}(\lambda)$ を A の “基本最小消去多項式”， $\pi'_{A,j}(\lambda)$ を A の “基本最小消去多項式候補” と呼ぶことにする．また，$f_p(\lambda)$ に対し，2 変数多項式 $\psi_p(x, y)$ を

$$\psi_p(x, y) = \frac{f_p(x) - f_p(y)}{x - y}. \tag{4}$$

で定める．このとき，$\psi_p(x, y)$ は変数 y に関して $\deg(f_p) - 1$ 次の多項式であることに注意する．

以下では，ベクトル空間 $V = \mathrm{K}^n$ の有限部分集合 $S = \{\boldsymbol{v}_1, \ldots, \boldsymbol{v}_r\}$ に対し，S を含む V の最小の部分空間を $\mathrm{Span}(S)$ で表す．

2.2 仮定と目的

本稿では，行列 A とその特性多項式 $\chi_A(\lambda)$, $\chi_A(\lambda)$ の因数分解 (1) ，A の基本最小消去多項式候補 (3) が与えられているもとで，$\chi_A(\lambda)$ の因子 $f_p(\lambda)$（および $f_p(\lambda)$ の零点である A の固有値 $\lambda = \alpha$）に着目する．

$f_p(\lambda)$ に対し，$l'_{j,p}$ $(j=1,\ldots,n)$ の最大値は 1 に等しい，すなわち

$$l'_p := \max_{j\in J}\{l'_{j,p}\} = 1$$

と仮定する．

注意 1
A の基本最小消去多項式 (2) に対し，$l_p = \max_{j\in J}\{l_{j,p}\}$ とおく．このとき，A の固有値 $\lambda=\alpha$ で $f_p(\alpha)=0$ をみたすものに対し，$l_p=1$ ならば，またそのときに限り，固有値 α に属する一般固有ベクトル空間は固有ベクトル空間に等しい．∎

我々が本稿で提案する固有ベクトル算法の目的は，$f_p(\alpha)=0$ をみたす A の固有値 α に着目し，($l_p=l'_p$ を確かめた上で）固有値 α に属する固有ベクトルの（各成分を α の多項式として表した）表現をすべて求めることである．

3　基本最小消去多項式候補がすべて真の基本最小消去多項式に等しい場合

3.1　固有ベクトルの表現

まず，式 (3) の基本最小消去多項式候補 $\pi'_{A,j}(\lambda)$ がすべて，式 (2) の真の基本最小消去多項式 $\pi_{A,j}(\lambda)$ に等しい場合を扱う．

各 $\pi_{A,j}(\lambda)$ における $f_p(\lambda)$ の多重度 $l_{j,p}$ に着目する．仮定より $l_p=\max_{j\in J}\{l_{j,p}\}=1$ $(J=\{1,\ldots,n\})$ であるので，$l_{j,p}$ は 1 または 0 に等しい．このとき，インデックスの集合 J を

$$J_0=\{i\in J \mid l_{j,p}=0\},\quad J_1=\{i\in J \mid l_{j,p}=1\},\quad J=J_0\cup J_1$$

と分割する．そして，$j\in J$ に対し，多項式 $g_j(\lambda)$ を，

$$g_j(\lambda)=\begin{cases}\pi_{A,j}(\lambda) & \text{for}\quad j\in J_0,\\ \pi_{A,j}(\lambda)/f_p(\lambda) & \text{for}\quad j\in J_1\end{cases}$$

で定義する．このとき，すべての $j\in J$ に対し，$g_j(\lambda)$ と $f_p(\lambda)$ は互いに素であることに注意する．

$j\in J_1$ に対し，ベクトル $\boldsymbol{v}_j$ を

$$\boldsymbol{v}_j = g_j(A)\boldsymbol{e}_j \tag{5}$$

で定義する．このとき，次の命題が成り立つ．

命題 1
$j\in J_1$ とし，α を $f_p(\alpha)=0$ をみたす A の固有値とする．式 (4) の $\psi_p(x,y)$ および式 (5) の v_j に対し，ベクトル $\psi_p(A,\lambda E)\boldsymbol{v}_j$ に $\lambda=\alpha$ を代入したベクトルは A の固有値 α に属する固有ベクトルである．

証明 式 (4) より $f_p(x)-f_p(y)=(x-y)\psi_p(x,y)$. よって $\psi_p(A,\lambda E)\boldsymbol{v}_j$ を考えると

$$(A-\lambda E)(\psi_p(A,\lambda E)\boldsymbol{v}_j)=(f_p(A)-f_p(\lambda E))\boldsymbol{v}_j.$$

が成り立つ．ここで $\lambda=\alpha$ を代入すると，$f_p(\alpha)=0$ より

$$(A-\alpha E)(\psi_p(A,\alpha E)\boldsymbol{v}_j)=f_p(A)\boldsymbol{v}_j=f_p(A)g_j(A)\boldsymbol{e}_j=\pi_{A,j}(A)\boldsymbol{e}_j=\boldsymbol{0}$$

を得る．∎

$\deg(f_p(\lambda)) = d_p = d$ とおき，$\alpha_1, \alpha_2, \ldots, \alpha_d$ を $f_p(\lambda)$ の相異なる零点とするとき，命題 1 より，$i = 1, \ldots, d$, $j \in J_1$ に対し，ベクトル $\psi_p(A, \alpha_i E)\boldsymbol{v}_j$ はすべて A の固有値 α_i に属する固有ベクトルを表す．すなわち，$j \in J_1$ に対し，$\psi_p(A, \lambda E)\boldsymbol{v}_j$ なるベクトルを 1 個求めれば，変数（記号）$\alpha_1, \alpha_2, \ldots, \alpha_d$ で表される $d = \deg(f_p)$ 個の固有値に属する固有ベクトルをすべて構成したことになる．

3.2 $f_p(\alpha) = 0$ をみたすすべての固有値に属する固有ベクトル空間（の基底）の構成

前節では，$j \in J_1$ に対し，$f_p(\alpha) = 0$ をみたす A の固有値 α に属する合計 $d = \deg(f_p)$ 個の固有ベクトルを構成する方法を示した．ところで，式 (1) より，$f_p(\alpha) = 0$ をみたす A の固有値 α の重複度は m_p である．f_p は K 上の既約多項式であり，$\deg(f_p) = d_p = d$ より，方程式 $f_p(\lambda) = 0$ は d 個の異なる根をもつ．それらの根を $\alpha_1, \ldots, \alpha_d$ とおくと，$i = 1, \ldots, d$ に対し，各 $\lambda = \alpha_i$ に属する固有ベクトル空間（これを F_{p,α_i} とおく）は m_p 次元なので，$f_p(\alpha) = 0$ をみたすすべての固有値に属する固有ベクトル空間（これを F_p とおく）の次元は $d_p m_p = d m_p$ となる．本節では，前節までの議論を踏まえ，固有ベクトル空間 F_p の基底をなす dm_p 個の固有ベクトルを計算する方法を導く．

$V = \mathrm{Span}(\boldsymbol{v}_j \mid j \in J_1)$ とする（$\boldsymbol{v}_j$ の定義は (5) を参照）．さらに，$\boldsymbol{0} \neq \boldsymbol{v}$ なる $\boldsymbol{v} \in V$ に対し，$\psi_p(A, \lambda E)\boldsymbol{v} \neq 0$ を考える．このとき，$f_p(\alpha) = 0$ をみたす α を λ に代入すると，命題 1 より

$$(A - \alpha E)\psi_p(A, \alpha E)\boldsymbol{v} = (f_p(A) - f_p(\alpha E))\boldsymbol{v} = f_p(A)\boldsymbol{v} = \boldsymbol{0}$$

が成り立つ．（ここで，$f_p(\lambda)$ は A における $\boldsymbol{v}$ の最小消去多項式であることに注意する．）すなわち，$\psi_p(A, \alpha E)\boldsymbol{v}$ は A の固有値 α に属する固有ベクトルである．よって，固有ベクトル空間 F_p の基底をなす dm_p 個の固有ベクトルは，K^n の部分空間 V から m_p 個のベクトル

$$\boldsymbol{w}_1, \boldsymbol{w}_2, \ldots, \boldsymbol{w}_{m_p} \tag{6}$$

を適当に選び，$k = 1, 2, \ldots, m_p$ に対し，ベクトル $\psi_p(A, \lambda E)\boldsymbol{w}_k$ が

$$F_p = \mathrm{Span}(\psi_p(A, \alpha_i)\boldsymbol{w}_k \mid i = 1, \ldots, d,\ k = 1, \ldots, m_p) \tag{7}$$

をみたす，すなわち $\psi_p(A, \alpha_i)\boldsymbol{w}_k$ $(k = 1, 2, \ldots, m_p)$ が一次独立になるように構成すればよいことがわかる．では，この条件を満たすような式 (6) のベクトル $\boldsymbol{w}_1, \boldsymbol{w}_2, \ldots, \boldsymbol{w}_{m_p}$ の選択をどのように行えばよいだろうか?

$f_p(\lambda)$ が A における $\boldsymbol{v}$ の最小消去多項式であることから，ベクトル $\boldsymbol{v}$, $A\boldsymbol{v}$, $A^2\boldsymbol{v}$, …, $A^{d-1}\boldsymbol{v}$ は一次独立である．そこで，$\boldsymbol{v} \in V$ に対し

$$L_A(\boldsymbol{v}) = \mathrm{Span}(\boldsymbol{v}, A\boldsymbol{v}, A^2\boldsymbol{v}, \ldots, A^{d-1}\boldsymbol{v})$$

とおく．このとき，次の命題が成り立つ．

命題 2
$\boldsymbol{u}, \boldsymbol{w} \in V$（ただし $\boldsymbol{u}, \boldsymbol{v} \neq \boldsymbol{0}$）とする．このとき，以下は同値:

1. $\mathrm{Span}(\psi_p(A, \alpha_i)\boldsymbol{u} \mid i = 1, \ldots, d) = \mathrm{Span}(\psi_p(A, \alpha_i)\boldsymbol{w} \mid i = 1, \ldots, d)$,
2. $L_A(\boldsymbol{u}) = L_A(\boldsymbol{w})$,
3. $\boldsymbol{w} \in L_A(\boldsymbol{u})$,
4. $\boldsymbol{u} \in L_A(\boldsymbol{w})$.

証明 $\boldsymbol{u} \in V$ に対し，上記より $\boldsymbol{u}$, $A\boldsymbol{u}$, $A^2\boldsymbol{u}$, …, $A^{d-1}\boldsymbol{u}$ は一次独立．また $k = 0, \ldots, d-1$ に対し

$$\psi_p(A, \lambda E)(A^k \boldsymbol{u}) = A^k(\psi_p(A, \lambda E)\boldsymbol{u})$$

が成り立つ．ところが，$f_p(\alpha) = 0$ をみたす A の固有値 α に対し，$\psi_p(A, \alpha E)\boldsymbol{u}$ は α に属する A の固有ベクトルであるので，$k = 0, \ldots, d-1$ に対し

$$\psi_p(A, \alpha E)(A^k \boldsymbol{u}) = A^k(\psi_p(A, \alpha E)\boldsymbol{u}) = \alpha^k(\psi_p(A, \alpha E)\boldsymbol{u}) \tag{8}$$

が成り立つ．すなわち，$\boldsymbol{0} \neq \boldsymbol{w} \in L_A(\boldsymbol{u})$ に対し，$\psi_p(A, \alpha E)\boldsymbol{w}$ は $\psi_p(A, \alpha E)\boldsymbol{u}$ のスカラー倍に過ぎないことがわかる．よって 3. と 1. は同値．

一方，$\boldsymbol{w}$, $A\boldsymbol{w}$, $A^2\boldsymbol{w}$, …, $A^{d-1}\boldsymbol{w}$ についても (8) と同様に

$$\psi_p(A, \alpha E)(A^k \boldsymbol{w}) = A^k(\psi_p(A, \alpha E)\boldsymbol{w}) = \alpha^k(\psi_p(A, \alpha E)\boldsymbol{w})$$

が成り立つ．$\psi_p(A, \alpha E)\boldsymbol{u}$ も $\psi_p(A, \alpha E)\boldsymbol{w}$ のスカラー倍に過ぎないことに注意すると 3. と 4. が同値，ゆえに（3. および 4. と）2. も同値であることがわかる．∎

命題 2 より，式 (7) をみたすような式 (6) のベクトル $\boldsymbol{w}_1, \boldsymbol{w}_2, \ldots, \boldsymbol{w}_{m_p} \in V$ として

$$V = L_A(\boldsymbol{w}_1) \oplus L_A(\boldsymbol{w}_2) \oplus \cdots \oplus L_A(\boldsymbol{w}_{m_p})$$

をみたす $\boldsymbol{w}_1, \boldsymbol{w}_2, \ldots, \boldsymbol{w}_{m_p}$ を選べばよいことがわかる．

3.3 固有ベクトル計算の手順

前節までの議論を踏まえ，基本最小消去多項式候補 $\pi'_{A,j}(\lambda)$ がすべて真の基本最小消去多項式 $\pi_{A,j}(\lambda)$ に一致していることがわかっている場合に、$f_p(\alpha) = 0$ をみたす A の固有値 α に属する固有ベクトル（空間の基底）を計算する方法を以下の通り示す．

アルゴリズム **1**
(基本最小消去多項式候補がすべて真の基本最小消去多項式に一致していることがわかっている場合の固有ベクトルの算法)

[Step 1] $J = \{1, 2, \ldots, n\}$ を

$$J_0 = \{j \in J \mid l_{j,p} = 0\}, \quad J_1 = \{j \in J \mid l_{j,p} = 1\}, \quad J_0 \cup J_1 = J$$

に分割する．

[Step 2] $j \in J_1$ に対し $\boldsymbol{v}_j = g_j(A)\boldsymbol{e}_j$ を計算する．

[Step 3] $V = \mathrm{Span}(\boldsymbol{v}_j \mid j \in J_1\}$ の基底（ベクトル）

$$\{\boldsymbol{b}_1, \boldsymbol{b}_2, \ldots, \boldsymbol{b}_{dm_p}\} = B$$

を，$\{\boldsymbol{v}_j \mid j \in J_1\}$ に対する掃き出し法によって求める．

[Step 4] 1. V の基底ベクトルの集合 B から最も “単純な” 形のベクトルを選び，これを $\boldsymbol{u}_1$ とする．これに対し

$$L_A(\boldsymbol{u}_1) = \mathrm{Span}(\boldsymbol{u}_1, A\boldsymbol{u}_1, \ldots, A^{d-1}\boldsymbol{u}_1)$$

を計算し，掃き出し法で $L_A(\boldsymbol{u}_1)$ の基底 B_1 を求める．

2. B の要素であり，かつ $L_A(\boldsymbol{u}_1)$ に属さないものから最も “単純な” 形のベクトルを選び，これを $\boldsymbol{u}_2$ とする．これに対し

$$L_A(\boldsymbol{u}_2) = \mathrm{Span}(\boldsymbol{u}_2, A\boldsymbol{u}_2, \ldots, A^{d-1}\boldsymbol{u}_2)$$

を計算し，さらに $L_A(\boldsymbol{u}_1) \oplus L_A(\boldsymbol{u}_2)$ に掃き出し法を適用し，$L_A(\boldsymbol{u}_1) \oplus L_A(\boldsymbol{u}_2)$ の基底 B_2 を求める．

3. B の要素であり，かつ $L_A(\boldsymbol{u}_1) \oplus L_A(\boldsymbol{u}_2)$ に属さないものから最も “単純な” 形のベクトルを選び，これを $\boldsymbol{u}_3$ とする．これに対し

$$L_A(\boldsymbol{u}_3) = \mathrm{Span}(\boldsymbol{u}_3, A\boldsymbol{u}_3, \ldots, A^{d-1}\boldsymbol{u}_3)$$

を計算し，さらに $L_A(\boldsymbol{u}_1) \oplus L_A(\boldsymbol{u}_2) \oplus L_A(\boldsymbol{u}_3)$ に掃き出し法を適用し，$L_A(\boldsymbol{u}_1) \oplus L_A(\boldsymbol{u}_2) \oplus L_A(\boldsymbol{u}_3)$ の基底 B_3 を求める．

4. 以下，同様にして，$k = 4, \ldots, m_p - 1$ に対し，$L_A(\boldsymbol{u}_1) \oplus \cdots \oplus L_A(\boldsymbol{u}_k)$ の基底 B_k を求める．

5. 最後に，B の要素であり，かつ $L_A(\boldsymbol{u}_1) \oplus \cdots \oplus L_A(\boldsymbol{u}_{m_p-1})$ に属さないものから最も “単純な” 形のベクトルを選び，これを $\boldsymbol{u}_{m_p}$ とおく．このステップにおいては，$L_A(\boldsymbol{u}_{m_p})$ 等の計算は不要である点に注意する．

[Step 5] 上のステップで得られた $\boldsymbol{u}_1, \boldsymbol{u}_2, \ldots, \boldsymbol{u}_{m_p}$ に対し

$$\psi_p(A, \lambda E)\boldsymbol{u}_k, \quad k = 1, \ldots, m_p$$

を計算する．これが求める固有ベクトルを与える．∎

4 基本最小消去多項式候補がすべて真の基本最小消去多項式に一致するかどうかが不明な場合

4.1 基本的なアルゴリズム

これ以降では，式 (3) の基本最小消去多項式候補 $\pi'_{A,j}(\lambda)$ が与えられているが，これらすべてが式 (2) の真の基本最小消去多項式 $\pi_{A,j}(\lambda)$ に一致するかどうかがまだ不明であるとする．ほとんどの場合 $\pi'_{A,j}(\lambda) = \pi_{A,j}(\lambda)$ $(i \in J)$ が成り立つことが期待できるが，何らかの方法で $\pi'_{A,j}(\lambda)$ $(i \in J)$ が真の最小消去多項式を与えていることを確かめる必要がある．

そこで，上述のアルゴリズム 1 に，すでに与えられている基本最小消去多項式候補 $\pi'_{A,j}(\lambda)$ が真の基本最小消去多項式に等しいかどうかを検査する計算を加えたアルゴリズムを以下の通り示す．

アルゴリズム **2**
(基本最小消去多項式候補が真の基本最小消去多項式に一致しているかどうかの検査を固有ベクトルの算法)

[Step 1] $J'_0 = \{j \in J \mid l'_{j,p} = 0\}$, $J'_1 = \{j \in J \mid l'_{j,p} = 1\}$ とおく．仮定より $\max\{l'_{j,p} = 1 \mid i \in J\} = 1$ であるので $J'_0 \cup J'_1 = J$ が成り立つ．

各 $j \in J'_0$ に対し $\pi'_{A,j}(\lambda) = g_j(\lambda)$ （$g_j(\lambda)$ は $f_p(\lambda)$ と互いに素）と表せる．そこで，$j \in J'_0$ に対し $g_j(A)\boldsymbol{e}_j$ を計算し，$g_j(A)\boldsymbol{e}_j = \boldsymbol{0}$ が成り立つことを確かめる．

[Step 2] $j \in J_1'$ に対し $\boldsymbol{v}_j = g_j(A)\boldsymbol{e}_j$ を計算する．(ここに，$\pi'_{A,j}(\lambda) = f_p(\lambda)g_j(\lambda)$, g_j は f_p と互いに素.)

ここで，基本最小消去多項式候補が真の基本最小消去多項式に一致するかどうかを以下の手順で確かめる．$j \in J_1'$ に対し，整数 c_j を無作為に選び，$\boldsymbol{v} = \sum_{j\in J_1'} c_j\boldsymbol{v}_j$ とおき，

$$f_p(A)\boldsymbol{v} = \boldsymbol{0} \tag{9}$$

が成り立つことを確かめる．

もし $f_p(A)\boldsymbol{v} \neq \boldsymbol{0}$ ならば，少なくとも 1 つのベクトル $\boldsymbol{v}_j$ $(j \in J_1')$ が $f_p(A)$ で $\boldsymbol{0}$ に写らないことがわかる．すなわち，基本最小消去多項式 $\pi'_{A,j}(\lambda)$ で真の基本最小消去多項式 $\pi_{A,j}(\lambda)$ に一致しないものが存在することがわかる．

以下のステップは，式 (9) が成り立つ場合に実行する．

[Step 3] $V = \mathrm{Span}(\boldsymbol{v}_j \mid j \in J_1')$ の基底（ベクトル）

$$\{\boldsymbol{b}_1, \boldsymbol{b}_2, \ldots, \boldsymbol{b}_{dm_p}\} = B$$

を，$\{\boldsymbol{v}_j \mid j \in J_1'\}$ に対する掃き出し法によって求める．

[Step 4] 1. V の基底ベクトルの集合 B から最も “単純な” 形のベクトルを選び，これを $\boldsymbol{u}_1$ とする．これに対し，$\boldsymbol{u}_1, A\boldsymbol{u}_1, \ldots, A^{d-1}\boldsymbol{u}_1$ を求めた上で，$f_p(A)\boldsymbol{u}_1 = \boldsymbol{0}$ を確かめる．もし $f_p(A)\boldsymbol{u}_1 = \boldsymbol{0}$ が成り立つ場合は

$$L_A(\boldsymbol{u}_1) = \mathrm{Span}(\boldsymbol{u}_1, A\boldsymbol{u}_1, \ldots, A^{d-1}\boldsymbol{u}_1)$$

を計算し，掃き出し法で $L_A(\boldsymbol{u}_1)$ の基底 B_1 を求める．

2. B の要素であり，かつ $L_A(\boldsymbol{u}_1)$ に属さないものから最も “単純な” 形のベクトルを選び，これを $\boldsymbol{u}_2$ とする．これに対し，$\boldsymbol{u}_2, A\boldsymbol{u}_2, \ldots, A^{d-1}\boldsymbol{u}_2$ を求めた上で，$f_p(A)\boldsymbol{u}_2 = \boldsymbol{0}$ を確かめる．もし $f_p(A)\boldsymbol{u}_2 = \boldsymbol{0}$ が成り立つ場合は

$$L_A(\boldsymbol{u}_2) = \mathrm{Span}(\boldsymbol{u}_2, A\boldsymbol{u}_2, \ldots, A^{d-1}\boldsymbol{u}_2)$$

を計算し，さらに $L_A(\boldsymbol{u}_1) \oplus L_A(\boldsymbol{u}_2)$ に掃き出し法を適用し，$L_A(\boldsymbol{u}_1) \oplus L_A(\boldsymbol{u}_2)$ の基底 B_2 を求める．

3. 以下，同様にして，$k = 3, \ldots, m_p - 1$ に対し，$L_A(\boldsymbol{u}_1) \oplus \cdots \oplus L_A(\boldsymbol{u}_k)$ の基底 B_k を求める．

4. 最後に，B の要素であり，かつ $L_A(\boldsymbol{u}_1) \oplus \cdots \oplus L_A(\boldsymbol{u}_{m_p-1})$ に属さないものから最も “単純な” 形のベクトルを選び，これを $\boldsymbol{u}_{m_p}$ とおく．そして $f_p(A)\boldsymbol{u}_{m_p}$ が成り立つことを確かめる．

[Step 5] 上のステップで得られた $\boldsymbol{u}_1, \boldsymbol{u}_2, \ldots, \boldsymbol{u}_{m_p}$ に対し

$$\psi_p(A, \lambda E)\boldsymbol{u}_k, \quad k = 1, \ldots, m_p$$

を計算する．これらが求める固有ベクトルを与える．∎

注意 2

基本最小消去多項式候補が真の基本最小消去多項式に一致しているかどうかを検査する方法として，すべての $j \in J_1'$ に対し $\pi'_{A,j}(A)\boldsymbol{e}_j = f_p(A)\boldsymbol{v}_j = \boldsymbol{0}$ を求める方法が考えられる．しかしながら，この検査は，アルゴリズム 2 で与えた通り，$V = \mathrm{Span}(\boldsymbol{v}_j \mid j \in J_1')$ の基底 $B = \{\boldsymbol{b}_1, \ldots, \boldsymbol{b}_{m_p d}\}$ $(d = d_p)$ から選んだ m_p 個のベクトル $\boldsymbol{u}_1, \ldots, \boldsymbol{u}_{m_p}$ のみに対して $\pi'_{A,j}(A)\boldsymbol{b}_k = f_p(A)\boldsymbol{b}_k = \boldsymbol{0}$ を確かめれば十分である（なぜなら，命題 1, 2 より，$\boldsymbol{0} \neq \boldsymbol{w} \in L_A(\boldsymbol{u}_k)$ をみたすすべてのベクトル $\boldsymbol{w}$ に対し，$\boldsymbol{u}_k$ の最小消去多項式が $f_p(\lambda)$ であれば，$\boldsymbol{w}$ の最小消去多項式も $f_p(\lambda)$ になるからである）．これにより，最小消去多項式候補が真の最小消去多項式になることを確かめる手間が $1/d$ に抑えられたことになる．∎

4.2 アルゴリズムの効率化

アルゴリズム 2 は，以下の点で効率化が可能である．

1. [Step 1] および [Step 2] において $\boldsymbol{v}_j$ を求める計算は j 毎に独立した計算なので，並列化が可能である．

2. [Step 4] および [Step 5] に着目する．[Step 4] では，ある $\boldsymbol{u}_k$ に対し，$\boldsymbol{u}_k, A\boldsymbol{u}_k, \ldots, A^{d-1}\boldsymbol{u}_k$ を求めた上で，$f_p(A)\boldsymbol{u}_k = \boldsymbol{0}$ が成り立つことを確かめ，$L_A(\boldsymbol{u}_k) = \mathrm{Span}(\boldsymbol{u}_k, A\boldsymbol{u}_k, \ldots, A^{d-1}\boldsymbol{u}_k)$ の基底を求めている．ここで計算される $\boldsymbol{u}_k, A\boldsymbol{u}_k, \ldots, A^{d-1}\boldsymbol{u}_k$ を記憶しておくことで，[Step 5] の $\psi_p(A, \lambda E)\boldsymbol{u}_k$ の計算に再利用することが可能であるが，加算の回数が多くなる．

   ところが，[Step 5] の $\psi_p(A, \lambda E)\boldsymbol{u}_k$ の計算を “行列 Horner 法” を用いて行った後，Horner 法の計算をさらにもう 1 度行うことで $f_p(A)\boldsymbol{u}_k$ を計算できる（詳細は照井・田島 [5] を参照）．

ここでは，特に上記 2. を考慮することにより，アルゴリズム 2 を以下の形で効率化する．

アルゴリズム **3**
(基本最小消去多項式候補が真の基本最小消去多項式に一致しているかどうかの検査を固有ベクトルの算法: 改良版)

[Step 1], [Step 2], [Step 3] はアルゴリズム 2 と同一であるので省略．

[Step 4] 1. V の基底ベクトルの集合 B から最も “単純な” 形のベクトルを選び，これを $\boldsymbol{u}_1$ とする．$\psi_p(A, \lambda E)\boldsymbol{u}_1$ を計算し，メモリに保存する．この結果を用いて $f(A)\boldsymbol{u}_1 = \boldsymbol{0}$ を確かめる．もし $f_p(A)\boldsymbol{u}_1 = \boldsymbol{0}$ が成り立つ場合は

$$L_A(\boldsymbol{u}_1) = \mathrm{Span}(\boldsymbol{u}_1, A\boldsymbol{u}_1, \ldots, A^{d-1}\boldsymbol{u}_1)$$

を計算し，掃き出し法で $L_A(\boldsymbol{u}_1)$ の基底 B_1 を求める．

2. B の要素であり，かつ $L_A(\boldsymbol{u}_1)$ に属さないものから最も “単純な” 形のベクトルを選び，これを $\boldsymbol{u}_2$ とする．これに対し，$\psi_p(A, \lambda E)\boldsymbol{u}_2$ を計算し，メモリに保存する．この結果を用いて $f(A)\boldsymbol{u}_2 = \boldsymbol{0}$ を確かめる．もし $f_p(A)\boldsymbol{u}_2 = \boldsymbol{0}$ が成り立つ場合は

$$L_A(\boldsymbol{u}_2) = \mathrm{Span}(\boldsymbol{u}_2, A\boldsymbol{u}_2, \ldots, A^{d-1}\boldsymbol{u}_2)$$

を計算し，さらに $L_A(\boldsymbol{u}_1) \oplus L_A(\boldsymbol{u}_2)$ に掃き出し法を適用し，$L_A(\boldsymbol{u}_1) \oplus L_A(\boldsymbol{u}_2)$ の基底 B_2 を求める．

3. 以下，同様にして，$k = 3, \ldots, m_p - 1$ に対し，$\psi_p(A, \lambda E)\boldsymbol{u}_k$ を計算し，メモリに保存した上で，この結果を用いて $f(A)\boldsymbol{u}_k = \boldsymbol{0}$ を確かめる．もし $f_p(A)\boldsymbol{u}_k = \boldsymbol{0}$ が成り立つ場合は

$$L_A(\boldsymbol{u}_k) = \mathrm{Span}(\boldsymbol{u}_k, A\boldsymbol{u}_k, \ldots, A^{d-1}\boldsymbol{u}_k)$$

を計算し，さらに $L_A(\boldsymbol{u}_1) \oplus \cdots \oplus L_A(\boldsymbol{u}_k)$ に掃き出し法を適用し，基底 B_k を求める．

4. 最後に，B の要素であり，かつ $L_A(\boldsymbol{u}_1) \oplus \cdots \oplus L_A(\boldsymbol{u}_{m_p-1})$ に属さないものから最も “単純な” 形のベクトルを選び，これを $\boldsymbol{u}_{m_p}$ とおく．そして $\psi_p(A, \lambda E)\boldsymbol{u}_{m_p}$ を計算し，メモリに保存した上で，この結果を用いて $f_p(A)\boldsymbol{u}_{m_p} = \boldsymbol{0}$ が成り立つことを確かめる．

[Step 5] 上のステップで得られた $\psi_p(A, \lambda E)\boldsymbol{u}_k$ $(k = 1, \ldots, m_p)$ を A の固有ベクトルとして出力する．∎

5 まとめ

本稿では，レゾルベントの留数解析に基づき，行列の固有ベクトルを効率的に計算する算法として，我々がこれまでに提案した算法の拡張を提案した．これまでに提案した算法では，着目する固有値の重複度が 1 の場合に限られたが，本稿で提案する算法では，着目する固有値に属する一般固有ベクトル空間が，固有ベクトル空間に等しいという条件下で，着目する固有値の特性方程式における重複度が 1 よりも大きい場合にも固有ベクトルを計算可能にした．

実際の固有ベクトル算法としては，まず，基本最小消去多項式候補がすべて真の基本最小消去多項式に等しいとあらかじめわかっている場合の算法を提案し，次に基本最小消去多項式候補がすべて真の基本最小消去多項式に一致するかどうかが不明な場合の算法を提案した．そして，特に後者においては，“行列 Horner 法” を用いることにより，先に固有ベクトル候補を計算してから，基本最小消去多項式候補が真の最小消去多項式に一致するかどうかを検査することで，固有ベクトルをより効率的に計算可能にすることを示した．

現在の課題は以下の通りである．提案したアルゴリズム内には “基底ベクトルの集合から最も ‘単純な’ ベクトルを選ぶ” 手順があるが，“単純” の基準をどのようにとるかは今後の検討課題である．また，各基底ベクトルに対し，格子算法 ([1], [2]) などによるベクトルの簡約を行うことがその後の計算の効率化に結びつくか等についても今後検討の余地がある．

今後は，以上の課題とともに，第 4.2 節で述べたように，並列処理なども含めた固有ベクトル計算の効率化を計り，算法の実装と検証を行いたいと考えている．

参 考 文 献

[1] Murray R. Bremner. *Lattice Basis Reduction.* CRC Press, 2012.

[2] Phone Q. Nguyen and Brigitte Vallée, editors. *The LLL Algorithm.* Information Security and Cryptography. Springer, 2010.

[3] 田島慎一, 小原功任, 照井章. 行列 Horner 法の並列化の実装について. 数式処理研究の新たな発展, 数理解析研究所講究録. 京都大学数理解析研究所, 印刷中.

[4] 田島慎一, 奈良洸平. 最小消去多項式候補とその応用. *Computer Algebra — Design of Algorithms, Implementations and Applications*, 数理解析研究所講究録, 第 1815 巻, pp. 1–12. 京都大学数理解析研究所, 2012 年 10 月.

[5] 照井章, 田島慎一. 行列の最小消去多項式候補を利用した固有ベクトル計算. *Computer Algebra — Design of Algorithms, Implementations and Applications*, 数理解析研究所講究録, 第 1815 巻, pp. 13–22. 京都大学数理解析研究所, 2012 年 10 月.

Computing the longest polynomial in the world -general discriminant formula of degree 17-

Kinji Kimura(Graduate School of Informatics, Kyoto University) *

Abstract

We introduce how to compute the general discriminant formula of degree 17 by using multivariate Newton interpolation. Our approach is not heuristic but deterministic. Therefore, our result is clearly world-record in computing the general discriminant formula.

1 問題の設定

1.1 判別式の定義

Sylvester 表現を利用して, 終結式 (resultant) を以下のように定義する,

$$f(x) = a_m x^m + a_{m-1}x^{m-1} + \cdots + a_0, g(x) = b_n x^n + b_{n-1}x^{n-1} + \cdots + b_0,$$

$$\mathrm{Sylvester}(f(x), g(x)) = \begin{pmatrix} a_m & a_{m-1} & \cdots & a_0 & & & \\ & a_m & a_{m-1} & \cdots & a_0 & & \\ & & \ddots & \ddots & \ddots & \ddots & \\ & & & a_m & a_{m-1} & \cdots & a_0 \\ b_n & b_{n-1} & \cdots & \cdots & b_0 & & \\ & b_n & b_{n-1} & \cdots & \cdots & b_0 & \\ & & \ddots & \ddots & \ddots & \ddots & \\ & & b_n & b_{n-1} & \cdots & \cdots & b_0 \end{pmatrix},$$

$$\mathrm{resultant}(f(x), g(x)) = \det(\mathrm{Sylvester}(f(x), g(x))).$$

すると, 判別式 (discriminant) は, 次のように定義できる,

$$\mathrm{discriminant}(f(x)) = (-1)^{\frac{1}{2}m(m-1)}\frac{1}{a_m}\mathrm{resultant}(f(x), f'(x)).$$

ただし, $f(x)$ は, m 次多項式である.

1.2 判別式の特徴

3 次式 ax^3+bx^2+cx+d の判別式は, $27d^2a^3-18dcba^2+4c^3a^2+4adb^3-ac^2b^2$ であり, 5 次の斉次多項式になっている (簡単に証明できる). よって, $a=1$ にしても一般性を失わない. よって, 3 次式 x^3+bx^2+cx+d の判別式を考えると, $27d^2-18dcb+4c^3+4db^3-c^2b^2$ である.

*Yoshida-Honmachi, Sakyo-ku, Kyoto 606-8501 JAPAN, kkimur@amp.i.kyoto-u.ac.jp

不変式論の立場では, 平行移動 $x \leftarrow x+\alpha$ することで, 3 次式 $x^3+c'x+d'$ の判別式を計算すれば十分であるということになっている. $\alpha=-b/3$ であるから, 計算結果に対して, 置換

$$c'=-1/3b^2+c, d'=2/27b^3-1/3cb+d$$

を行わなければならないため, 数式処理の立場からみると合理的な計算の方法ではない.

1.3 斉重多項式

3 次式 x^3+bx^2+cx+d の判別式 $27d^2-18dcb+4c^3+4db^3-c^2b^2$ の計算結果は, $\mathrm{weight}([b,c,d])=[1,2,3]$ において weighted homogenious になっている. すなわち, 斉重多項式になっている.

3 次方程式 x^3+bx^2+cx+d の解 x_1,x_2,x_3 とすると, 解と係数の関係より,

$$-b=x_1+x_2+x_3, c=x_1x_2+x_1x_3+x_2x_3, -d=x_1x_2x_3.$$

判別式の定義式は, $\{(x_3-x_1)(x_1-x_2)(x_2-x_3)\}^2$ であるから, x_1,x_2,x_3 について, 6 次の斉次式になる.

判別式が b,c,d の多項式で表現できるならば (容易に証明可能), $b^kc^ld^m$ の次数は, x_1,x_2,x_3 について 6 次の項を生成するため, $k+2l+3m=6$ とならなければならない. すなわち, 3 次式 x^3+x^2+cx+d の判別式が計算できることは, 3 次式 ax^3+bx^2+cx+d の判別式が計算できることに等しい.

以上より, 17 次の判別式は,

$$G'_{17}=x^{17}+x^{16}+a_{15}x^{15}+\cdots, H'_{17}=\mathrm{resultant}_x\left(G'_{17}, \frac{dG'_{17}}{dx}\right)$$

H'_{17} より容易に手に入れることができるため, H'_{17} を計算することを問題とする.

2 Cayley の方法

一般的な多項式の判別式の公式を設計するためにのみ用いることができる方法である. $f_n(x)=x^n+a_1x^{n-1}+\cdots+a_{n-1}x+a_n$ に対して, $\hat{f}_n(x)=x^n+a_1x^{n-1}+\cdots+a_{n-1}x+0$ とおくと, $\mathrm{disc.}(\hat{f}_n(x))=a_{n-1}^2\mathrm{disc.}(f_{n-1}(x))$ となる. ここから, a_n を変化させて, $\mathrm{disc.}(f_n(x))$ を求める.

2.1 判別式が満たす微分方程式

$f_n(x)=x^n+a_1x^{n-1}+\cdots+a_{n-1}x+a_n$ の判別式 D は,

$$n\frac{\partial D}{\partial a_1}+(n-1)a_1\frac{\partial D}{\partial a_2}+\cdots+a_{n-1}\frac{\partial D}{\partial a_n}=0$$

を満たすことを証明する.

根を $x_1,\cdots,x_n$ とすると, $D=\prod_{i>j}(x_i-x_j)^2$ となる. $x_i\rightarrow x_i-h$ と平行移動しても不変であるから,

$$\frac{dD}{dh}=\frac{\partial D}{\partial a_1}\frac{da_1}{dh}+\frac{\partial D}{\partial a_2}\frac{da_2}{dh}+\cdots+\frac{\partial D}{\partial a_n}\frac{da_n}{dh}=0.$$

$x_i\rightarrow x_i-h$ と平行移動は,

$$f_n(x+h)=(x+h)^n+a_1(x+h)^{n-1}+\cdots+a_{n-1}(x+h)+a_n.$$

x の $n-1$ 次の係数は, a_1+hn, x の $n-2$ 次の係数は, $a_2+h(n-1)a_1+h^2(\cdots)$, x の $n-3$ 次の係数は, $a_3+h(n-2)a_2+h^2(\cdots)$, $\cdots$ より, 証明終わり.

判別式 D を, a_n でべき級数展開してみる,

$$D_i=\left.\frac{\partial^i D}{\partial a_n^i}\right|_{a_n=0}, D=D_0+\frac{1}{1!}D_1a_n+\frac{1}{2!}D_2a_n^2+\cdots+\frac{1}{(n-1)!}D_{n-1}a_n^{n-1}.$$

べき級数が, $n-1$ 次までであることは Sylvester 表現から自明. さらに,

$$\frac{\partial D}{\partial a_n}=\frac{-1}{a_{n-1}}(n\frac{\partial D}{\partial a_1}+(n-1)a_1\frac{\partial D}{\partial a_2}+\cdots),$$

と変形し, 代入すると,

$$D_1=\frac{-1}{a_{n-1}}(n\frac{\partial D_0}{\partial a_1}+(n-1)a_1\frac{\partial D_0}{\partial a_2}+\cdots), D_2=\frac{-1}{a_{n-1}}(n\frac{\partial D_1}{\partial a_1}+(n-1)a_1\frac{\partial D_1}{\partial a_2}+\cdots),\cdots$$

となる.

2.1.1 アルゴリズム全体像

$\mathrm{disc.}(f_2)=a_1^2-4a_2$ とし, 以下のアルゴリズムを繰り返し適用して, 自分が必要な n まで計算する. $f_n(x)=x^n+a_1x^{n-1}+\cdots+a_{n-1}x+a_n$ に対して, $\hat{f}_n(x)=x^n+a_1x^{n-1}+\cdots+a_{n-1}x+0$ とおくと, $D_0=\mathrm{disc.}(\hat{f}_n(x))=a_{n-1}^2\mathrm{disc.}(f_{n-1}(x))$ となる. この D_0 を seed solution として, べき級数の係数を求め, $D=D_0+\frac{1}{1!}D_1a_n+\frac{1}{2!}D_2a_n^2+\cdots+\frac{1}{(n-1)!}D_{n-1}a_n^{n-1}$ を計算する.

3 小行列式展開による方法

3.1 終結式 (resultant) の関孝和, Bezout 表現

x について m 次の多項式 f と n 次の多項式 g の終結式を計算する. ただし, $m>n$ とする.

$$\begin{aligned}f &= a_mx^m+a_{m-1}x^{m-1}+\cdots+a_1x+a_0=0\\ x^{m-n}g &= b_nx^m+b_{n-1}x^{m-1}+\cdots+b_1x^{m-n+1}+b_0x^{m-n}=0\end{aligned}$$

この 2 本の式をもとに以下の操作を行い, x についての新しい多項式を n 本用意する.

$$\begin{aligned}&(a_mx^{n-1}+\cdots+a_{m-n+1})x^{m-n}g-(b_nx^{n-1}+\cdots+b_1)f\\ &=(a_mb_0-a_{m-n}b_n)x^{m-1}+(a_{m-1}b_0-a_{m-n}b_{n-1}-a_{m-n-1}b_n)x^{m-2}+\cdots=0\\ &\cdots\\ &a_mx^{m-n}g-b_nf=(a_mb_{n-1}-a_{m-1}b_n)x^{m-1}+(a_mb_{n-2}-a_{m-2}b_n)x^{m-2}+\cdots=0\end{aligned}$$

効率的に計算を行うために,

$$(a_mx+a_{m-1})x^{m-n}g-(b_nx+b_{n-1})f=x(a_mx^{m-n}g-b_nf)+(a_{m-1}x^{m-n}g-b_{n-1}f)$$

などが成立することに注意する. さらに, x についての多項式を $m-n$ 本を用意する,

$$\begin{aligned}x^{m-n-1}g &= b_nx^{m-1}+b_{n-1}x^{m-2}+\cdots+b_0x^{m-n-1}=0,\\ \cdots &\quad ,\\ g &= b_nx^n+b_{n-1}x^{n-1}+\cdots+b_0=0.\end{aligned}$$

これを行列の形式で書くと,

$$A = \begin{pmatrix} a_m b_0 - a_{m-n} b_n & \cdots & \cdots \\ \cdots & & \\ a_m b_{n-1} - a_{m-1} b_n & \cdots & \cdots \\ b_n & b_{n-1} & \cdots \\ & b_n & \cdots \\ & & \cdots \end{pmatrix}, v = \begin{pmatrix} x^{m-1} & \cdots & 1 \end{pmatrix}^\top, Av = 0$$

$\det(A)$ は, 関孝和, Bezout による終結式の行列式表現である.

3.2 関孝和, Bezout による終結式の行列式表現の小行列式展開法

4×4 の行列式を使って説明する.

$$\begin{vmatrix} a_1 & a_2 & a_3 & a_4 \\ b_1 & b_2 & b_3 & b_4 \\ c_1 & c_2 & c_3 & c_4 \\ d_1 & d_2 & d_3 & d_4 \end{vmatrix} = \begin{array}{l} +a_1(b_2(\underline{c_3d_4 - d_3c_4}) - c_2(\underline{b_3d_4 - d_3b_4}) + d_2(\underline{b_3c_4 - c_3b_4})) - b_1(a_2(\underline{c_3d_4 - d_3c_4}) \\ -c_2(\underline{a_3d_4 - d_3a_4}) + d_2(\underline{a_3c_4 - c_3a_4})) + c_1(a_2(\underline{b_3d_4 - d_3b_4}) - b_2(\underline{a_3d_4 - d_3a_4}) \\ +d_2(\underline{a_3b_4 - b_3a_4})) - d_1(a_2(\underline{b_3c_4 - c_3b_4}) - b_2(\underline{a_3c_4 - c_3a_4}) + c_2(\underline{a_3b_4 - b_3a_4})) \end{array}$$

下線部は, 使い回しができる. その事実を有効に使い, メモリに計算結果を蓄え計算を行う方法である.

4 提案手法

提案手法を説明するために, 準備をする.

4.1 linear assignment problem に基づく行列式の次数の上界公式

4.1.1 linear assignment problem:LAP とは

「あなたは, 働き手を持っています. Jim, Steve と Alan です, 一人に, 浴室の掃除をさせます. もう一人に, 床の掃き掃除をさせます. 三人目の人には, 窓の洗浄をさせます. 最大 (最小) のコストになるには, どのように仕事を割り当てればよいでしょうか? それぞれの仕事に対する 3 名のコストは, 以下のテーブルで与えられています.」このような問題を, linear assignment problem という. この問題を解くことと, 行列式の次数の上界公式を設計することは同値である.

	Clean bathroom	*Sweep floors*	*Wash windows*
Jim	\$1	\$2	\$3
Steve	\$3	\$3	\$3
Alan	\$3	\$3	\$2

我々の方法では, Jonker-Volgenant algorithm(LAPJV), 1987 を用いてこの問題を解いている.

4.1.2 多変数多項式を要素とする行列式の次数の上界

多変数多項式を要素とする行列式の次数の上界について考える. 具体的には, 次の A の次数の上界を考える,

$$A = \begin{vmatrix} x+y+z & xy \\ 2 & xyz \end{vmatrix}.$$

変数とパラメータの解釈により, 以下の表が得られる.

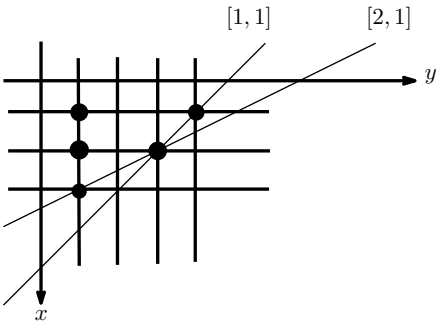
変数	パラメータ	(partial) total degree
x	y, z	2
y	x, z	2
z	x, y	2
x, y	z	3
y, z	x	3
z, x	y	3
x, y, z		4

ここでは, weight$([x, y, z])$=$[1, 1, 1]$ で考えている.

4.2 カット面付き多変数 Newton 補間

4.2.1 weight による計算量の削減

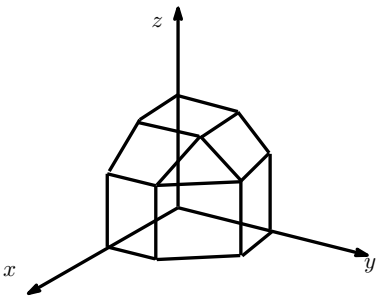
●で解の項の位置を表す. 2 種類の weight を用いて, cut 面を生成することを考える.
weight$([x, y])$=$[1, 1]$,weight$([x, y])$=$[2, 1]$ とする.



上記の図より, 2 つの weight を利用することで, より少ない評価点から解を補間できることがわかる. たくさんの weight を使えば, 原理的にはよりタイトな上界になる. しかし, 凸包の内点格子を生成する部分で, 多くの時間を費やすことになる.

4.2.2 カット面付き多変数 Newton 補間の具体例

下の図は, weight が 1 種類の場合である. 複数の weight でカットを入れる場合もあるが, ここでは, 1 種類の場合のみとする. 具体的には, weight$([x, y, z])$=$[1, 1, 1]$ を用いている. 下の凸包の内点格子の数分の評価点で, 終結式や行列式の値をサンプリングし, 補間する. 詳細は, [4] を参照されたい.

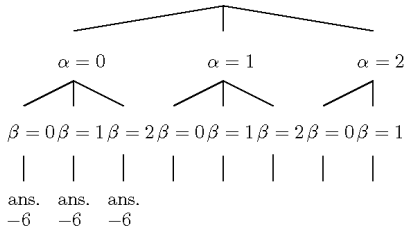


具体例として, 次の B を用いて説明する.

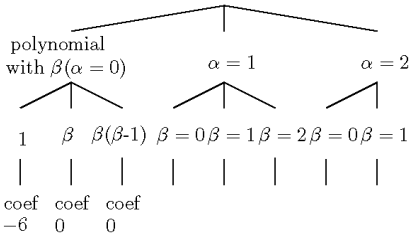
$$B = \begin{vmatrix} \alpha+\beta & 2 \\ 3 & \alpha\beta \end{vmatrix}$$

B の展開後の結果は, $(c_0+c_1\beta+c_2\beta(\beta-1))+\alpha(c_3+c_4\beta+c_5\beta(\beta-1))+\alpha(\alpha-1)(c_6+c_7\beta+c_8\beta(\beta-1))$, $c_8=0$, と仮定できる. $c_0,\cdots,c_7$ は, 未知数である.

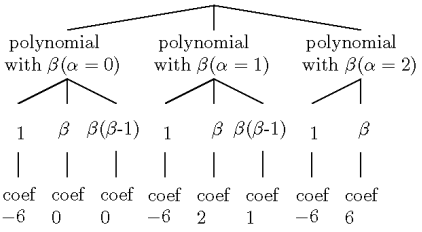
ここでは説明のため $\mathbb{Q}$ 上で表記するが, 実際のプログラムでは, $\mathbb{Z}/p\mathbb{Z}$ を利用して計算する. はじめに, $\alpha=0$, $\beta=0,1,2$ における値を計算したとする.



次に, $\alpha=0$ において Newton 補間をする.

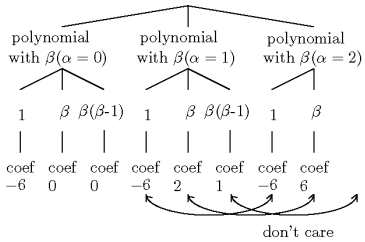


$\alpha=1,\alpha=2$ においても同様の計算をおこなう.

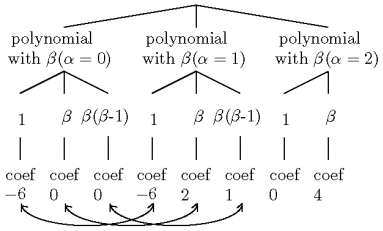


$\beta = 2$ の多項式は, あえて次数 1 で打ち切っている. Newton 補間は, 逐次補間であるため, このような途中で打ち切ったものを入力としても正しい計算ができる.

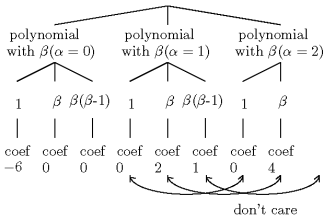
ここからは, 多項式を入力として Newton 補間をする. まず, $\alpha = 2$ の多項式から $\alpha = 1$ の多項式の減算をおこない, その結果を再度 $\alpha = 2$ に格納する.



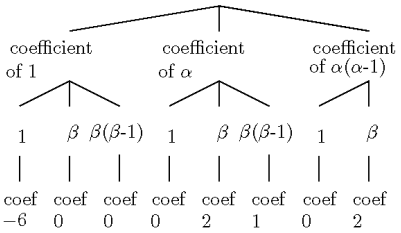
つぎに, $\alpha = 1$ の多項式から $\alpha = 0$ の多項式の減算をおこない, その結果を $\alpha = 1$ に格納する.



最後に $\alpha = 1$ の多項式から $\alpha = 0$ の多項式の減算をおこなうと, 多項式を入力とする Newton 補間が完了する.



以上より, つぎの結果を得る.



$B' = (-6 + 0\beta + 0\beta(\beta - 1)) + \alpha(0 + 2\beta + 1\beta(\beta - 1)) + \alpha(\alpha - 1)(0 + 2\beta + 0\beta(\beta - 1))$. 最後に, B' を展開し, B を得る.

5 アルゴリズムの比較

5.1 16 次までの次数の問題を使った Cayley の方法, 小行列式展開とカット面付き多変数 Newton 補間法の計算時間の比較

k	Cayley method Singular-3-1-6	TRIP minor Serial	TRIP minor Parallel	Newton Serial	Newton Parallel
12	19.337s	3m6.419s	3m5,686s	3m11.984s	13.596s
13	2m40.525s	27m38.563s	21m26.231s	22m29.047s	1m16.732s
14	12m40.276s	244m17.104s	146m44.051s	210m58.967s	10m17.354s
15	105m3.749s	-	-	1495m47.376s	67m23.748s
16	725m14.435s	-	-	???	462m46.799s

Cayley method Singular-3-1-6 は, 数式処理ソフト Singular[1] を用いて Cayley の方法を用いて計算した場合のタイミングデータを示している. TRIP minor は, 数式処理ソフト TRIP[2] を用いて小行列式展開を行った場合のタイミングデータを示している. Newton は, カット面付き多変数 Newton 補間法のタイミングデータを示している. -は, out of memory を意味する. TRIP は, 14 次の判別式を計算するために, メモリとして 483056 MB を必要とするため, それよりも次数の高い判別式を計算することは, 明らかにできない. ???は, あまりにも計算時間を必要とするため測定していない. なお, この表には, 凸包を生成する時間が含まれていない. なぜならば, 凸包を生成する部分は現在, 数式処理ソフト Risa/Asir[3] にて動作しているためである. 当日は, 全てのプログラムを C 言語で書き, その上で, すべてのタイミングデータを示す予定である. 実験環境は, CPU:E5-4650L, 4 CPU, 32 コア, mem:1440Gbyte, Intel C++ Compiler:13.1.2 である.

5.1.1 利用している weight について

16 方程式の判別式の計算を例に, 利用している weight を紹介する,

$$f(x) = x^{16} + x^{15} + a_{14}x^{14} + a_{13}x^{13} + a_{12}x^{12} + a_{11}x^{11} + a_{10}x^{10} + a_9x^9 + a_8x^8 + a_7x^7 + a_6x^6 + a_5x^5 + a_4x^4 + a_3x^3 + a_2x^2 + a_1x + a_0.$$

下記の 2 種類を使う.

	a_{14}	a_{13}	a_{12}	a_{11}	a_{10}	a_9	a_8	a_7	a_6	a_5	a_4	a_3	a_2	a_1	a_0
weight_1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
weight_2	14	13	12	11	10	9	8	7	6	5	4	3	2	1	16

6 17 次方程式の判別式の計算について

数式処理ソフト Singular-3-1-6 は, ディスクを使って計算を行うように改良することが容易ではないため, Cayley の方法を用いて 17 次方程式の判別式を計算することはしない. もちろん, 並列計算が容易に可能な算法でないことも, 選択肢から外した理由である. 小行列式展開は, 莫大なメモリ使用量を必要とするため,

選択肢から外した. よって, ここでは, カット面付き多変数 Newton 補間法のみを用いて, 17 次方程式の判別式を計算することを試みる.

$$f(x) = x^{17} + x^{16} + a_{15}x^{15} + a_{14}x^{14} + a_{13}x^{13} + a_{12}x^{12} + a_{11}x^{11} + a_{10}x^{10} + a_9x^9 + a_8x^8 + a_7x^7 + a_6x^6 + a_5x^5 + a_4x^4 + a_3x^3 + a_2x^2 + a_1x + a_0.$$

の判別式を計算することを考える.

6.1 計算結果

17 次方程式の判別式の項数は, 21976689397 であり, ディスク容量では, 427,470,114,659byte であった.

6.2 ディスクを使った使用メモリ量の削減方法について

計算結果について考察すると, a_{15} について, 17 次式であることから, a_{15} に, -9 から 8 まで数字を代入し, その結果を, 1 度ディスクに蓄え, さらに, ディスク上にて 1 変数の Newton 補間を行うことで使用メモリを削減できる. 次に示す結果は, a_{15} に, -9 から 8 まで数字を代入することによって得られた子問題を計算する際に要した時間を表している.

問題番号	計算時間
子問題 0	919m48.880s
子問題 1	928m1.009s
子問題 2	930m53.303s
子問題 3	933m17.794s
子問題 4	922m18.824s
子問題 5	929m57.208s
子問題 6	930m48.341s
子問題 7	922m16.684s
子問題 8	924m56.980s
子問題 9	864m15.264s
子問題 10	910m48.913s
子問題 11	924m9.816s
子問題 12	927m40.655s
子問題 13	922m10.048s
子問題 14	925m17.299s
子問題 15	934m1.590s
子問題 16	942m42.936s
子問題 17	937m48.058s

ディスク上にて 1 変数の Newton 補間を行った際の計算時間は, 612m15.268s であった. 項数のカウントに, 20m34.528s を必要とした. さらに, a_{15} から a_0 に具体的な数値を代入して, 判別式であることを確認するために, 109m47.591s を必要とした. 検算のため, そのような数値の代入を, 10 回行った.

7　まとめ

17 次方程式の判別式を計算する方法を紹介した.

参　考　文　献

[1] Singular, http://www.singular.uni-kl.de.

[2] TRIP, http://www.imcce.fr/Equipes/ASD/trip/trip.php.

[3] Risa/Asir, http://www.math.kobe-u.ac.jp/Asir/asir-ja.html.

[4] H. Werner, Mtinster, Remarks on Newton Type Multivariate Interpolation for Subsets of Grids, Computing 25(1980), 181-191.

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 21COE プログラム「機能数理学の構築と展開」(H.15-19 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」(H.20-24 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。

平成 25 年 7 月
マス・フォア・インダストリ研究所
所長　若山正人

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

Development of Computer Algebra Research and Collaboration with Industry

発　行　2013年8月9日
編　集　照井章，小原功任，濱田龍義，横山俊一，穴井宏和，横田博史
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研究院
九州大学大学院数理学府
〒819-0395 福岡市西区元岡744
九州大学伊都キャンパス数理学研究教育棟 共同利用・共同研究拠点事務室
TEL 092-802-4408　FAX 092-802-4405
URL http://gcoe-mi.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

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 数理学研究院
九州大学大学院 数理学府

〒819-0395 福岡市西区元岡744　TEL 092-802-4404　FAX 092-802-4405
URL http://gcoe-mi.jp/