SYMSAC '81の報告

金田康正(東大・大型センター)，村尾裕一(東大・理)
佐々木建昭(理研)，渡辺隼郎(津田塾大・数学)

§1. 概観

本年8月5～7日，米国Utah州Snow Birdで表記の会議が開かれた。因みにSYMSACとは，SYMposium on Symbolic and Algebraic Computationの略で，ACM・SIGSAMの主催で，5年毎に開催される。その間に，ヨーロッパ諸国が中心になってEUROSAMを開催するので，今回の会議で発表されたのは，1979年6月のEUROSAM以降2年間の研究成果である。

会議は3つの招待講演と40の一般講演，およびポスターセッションより成る。今回の会議の特徴は，i)数式処理システムの現代化が始まったこと，ii)アルゴリズム研究の中心が，因数分解と不定積分から微分方程式に重心を移し始めたこと，iii)計算量とその下限の観点から，アルゴリズムの再検討の気運が高まってきたこと，の3点に要約されよう。

第1の特徴は，MACSYMAやREDUCEに代表される現在の数式処理システムを古典的システムと位置付け，"現代的"システムのあるべき姿を論じたCannonの論文に代表される。また，IBMのJenksとTragerは，SCRATCHPADを現代化する目的で，domainとcategoryの概念に基づく代数的システムの設計を述べた。会議では発表されなかったが，MACSYMAとREDUCEも程度の差はあれ，現代化をもくろんでいる様である。なお，現代的システムとは言えぬが，CALTECの物理屋たちが，応用を目的に高速な数式処理システムSMPを発表して注目された。

今回発表された微分方程式関係の論文数は5，因数分解と不定積分に関する論文がそれぞれ3と2である。このことから見ても，微分方程式の研究が重視されていることが知れる。今回の会議における微分方程式の処理の特徴は，最近20～30年間の数学の理論に根ざしたアルゴリズム的アプローチであると言える。Watanabeは，超幾何型の常微分方程式に対する1950～70年代の日本の数学者の理論をもとに，従来のどのシステムも処理できなかった"高度"な常微分方程式の処理法を示した。似たような，しかしより適用範囲の狭いアルゴリズムがKovacicにより提案されている。これは，初等関数に対するRischの決定論的アルゴリズムによく似ており，数学的にエレガントであるが，解が初等関数で表わせるものに限られる。

上記第3の特徴は，高速アルゴリズムと計算量に関する3人の巨人たち，チューリッヒ大学のStrassen，IBMのWinograd，およびカーネギー・メロン大学のKungが招待講演者として招かれたことに端的に表われている。現在，代数的計算量の理論の分野では，高速アルゴリズムの開発の時期をすぎて，計算量の下限やより精度のよい計算量を求める時期に入っている。しかし，扱う対象は多項式の乗除算や行列演算が主である。一方，数式処理の分野では，因数分解や多変数多項式のGCDなど，実際問題に興味が集中しており，現時点では，計算量屋さん達との間に少しギャップがあるようである。

講演の国別内訳は，一般講演40のうち約半数の21を米国が占め，次いで7.5で英国，日本3，仏とオランダ各2，以下各1である。

§2 数式処理システム

Conference の最初をかざったのが system design のセッションで，4件の発表があった。そのうちの2件は、この分野（Computer Algebra：代数的計算）における，これからの方向を指唆するものであり，既存のシステムがそうであったように，単に式の計算を行うのでなく，群・環 etc. の代数的な概念を取り入れつつ，その構造等あるいはそれらを用いてより一般的な"代数"的計算を行うシステムを模索あるいは設計したものである。残る2件は、システムの移植・作成の報告であった。移植とはかねてから噂に上っていた，MACSYMA の VAX-11/780 への移植である。そして，残る1件が，物理屋が持ち前の怪力を出して一年余りで作ってしまったという SMP というシステムで，実演も含めて注目の的であった。

以下は、これらについての，筆者の独断と偏見とひがみによる報告である。

1) *The Basis of a Computer Algebra for Modern Algebra*

J. J. Cannon (U. of Sydney, Australia)

既存の，MACSYMA, SAC, SCRATCHPAD, REDUCE 等の汎用システムでは，その対象はほぼ多項式や有理関数に限られており，これらはいわば"古典的代数"に属するものである。そして，その機能はというと因数分解・不定積分等と豊富ではあるが，それら各要素（多項式ならばそれは環の一要素にすぎない）に対する具体的な演算であり，その集合全体の性質等を扱うものではない。講演者は、このようなシステムを CA(Classical Algebra) システムと呼んでいる。

一方代数においては，群論における Cayley のように，群等の対象物をより抽象的な概念として扱うようになり，そのような方向が確立されてきた。このような観点から，講演者は、CAシステムと対比させて，MA(Modern Algebra) システムが以下のようなものであるべきだと述べている。第一に、利用者が扱う式の属する集合を利用者自身が構成し決められること。つまり，多項式といっても，係数の領域を任意にとれ，又その中で演算も行われなければならない。さらに第二に，扱う代数的領域の構造自体のもつ性質も計算できることが望ましい。

以上のような MA システムの設計においては，どれだけの代数的対象物を受け付け，どう表現して扱うかが問題となる。このような点について，講演者は，彼自身が作成した有限群用システム Cayley（詳細な記述は SYMSAC '76）を用いて具体的な方法を示している。それによれば，各構造を表に記憶し，それらの間の関係を管理し，又重複計算を避けるなどして実現している。

2) *A Language for Computational Algebra*

R. D. Jenks & B. M. Trager

(IBM Thomas J. Watson Research Lab.)

Lisp レヴェルでの評価に mode を設けることにより，代数的計算における評価の領域を明確に規定しようという MODLISP の上に，いよいよ 1) でいう MA システム（NEWSPAD）を築こうということで，その核となる部分の設計について発表された。MODLISP に関する論文で度々述べられているように，いくつかの領域に対して共通なアルゴリズムを領域とは無関係に記述するために，abstract data type を実現するのが主眼であり，そのための構文が用意され，又，次のような概念が導入されている。

① **domain**：領域。整数環，多項式環 etc.,

② **functor**：領域を作るもの。例：領域「X の整係数多項式」は、Polynomial という functor に変数"X"と係数領域 Z（整数環）を引数として与えた時の値，

③ **category** ：共通の演算や性質をもったdomainの集合。

発表では、このcategoryという概念の説明が行われ，その重要性が述べられた。つまり，MAシステムとしてのNEWSPADは多くのdomainを扱う必要があるが，アルゴリズムの記述に際しては，対象となるいくつかのdomainの各々の構造や演算を知っている必要はなく，categoryとしての性質や演算さえ覚えておけばよいのである。現実には、引数に対するdomainの宣言（及びdomainの定義）さえあらかじめしておけば、システムは引数のdomainに対応する適切なMODLISPで書かれた関数を動的に（可能な限り翻訳時に）選び出し評価するわけである。

以上のように、NEWSPADは非常に一般的な代数的計算システムを目指しているわけだが，質問としても出た事だが，あまりに一般的すぎて効率の面に問題がでるのではないかとも考えられる。1)でも述べられていることだが，より多くの対象物を扱うことと効率とはtrade-offであり，どれほど効率のよいコンパイラが作られるかが部外者としては期待するところであろう（IBMとしてはSCRATCHPADも含めて純粋に研究用であり、公開の予定はないとのこと)。

3) *Charactorization of VAX Macsyma*
J.K. Foderaro & R.J. Fateman
(U.C., Berkeley)

一言で言ってしまえば、あのMACSYMAをVAX-11/780上に移植しましたということで，そのためにUNIX(Berkeley版，4BSD paging UNIX）上にLisp（Franz Lisp（因みにcompilerはLiszt，現在のversionはopus 34）；Berkeley UNIXと共に入手可)を作ったという話であった。米国内のいくつかの大学で稼動中ということで，いよいよ日本にもMACSYMAが正式に上陸かと期待されたが，版権がJ.Moses（MIT）にあり、Fatemanの意志とは裏腹に結局今のところは不可能。

さて内容はというと，VAX Macsyma（VAXIMA）でいくつかの問題を解かせた時のGC（廃品回収）の回数や時間，実行されたインストラクションの統計を持ち出して性能評価を行っている。発表は主にGCに関するものであった。アルゴリズム自体は単純なマーキングによるものだが，OSが自家製ということもあって，GC時にはペィヂングの機構を効率を考えてLRUからFIFOに切りかえている。又、各ペィヂにどれだけ回収されるべきセルがあるかを数え上げ，その率の低いペィヂに関してはGCを行わないという工夫もしている（彼らの計算では、最大8MBまで与えられるメモリを使い果たすには数十時間のCPU-timeがかかるというのである)。もうひとつ，bit map tableをGCの度に作り直すのでなく，何回かに一度だけクリアすることにより，マーキングの手間を省くという方法も提案してしいるが，実験はしていない。

4) SMP — A Symbolic Manipulation Program
C.A.Cole & S.Wolfram (Caltech)

実際に稼動するシステムとして最も注目を集めたのがこのSMPで，多機能そして効率（メモリ，スピードの両方)の両立を計り、ほぼ実現されているといえよう。彼らは本来,高エネルギー物理が専門であり，その計算を行うためのシステムを目指したとのこと（Caltechの超有名物理学者が本当に使っているかは不明）。設計方針として、高エネルギー物理計算用のSCHOONSHIP並の大規模な式が扱え，機能としては、MACSYMA並に強力ということになっている。SMP自体は言語Cで書かれており，勿論GCも行い，現在Cにして80K行，codeで900KBの量である。

記述言語としてLispでなくCを選んだ理由として、データ・タイプが柔軟性に富むことと移植性をあげている(現在はVAX/UNIX上のみ。これ自身は本年末にリリースの予定で，370系用のC-compilerも伝えられているのでIBM系で動くのも時間の問題であろう)。

機能は実に豊富で，分厚い電話帳のようなマニュアル(入手可。筆者らの手元に数冊あり)に書かれているので詳細は省略するが，グラフ出力(キャラクタ，グラフィック共)，又物理計算では頻出だが数式処理ではほとんど手のつけられていなかった特殊関数のライブラリまで備えている。因数分解，不定積分についても，未だ不完全(一変数等)とはいえ動いており，Wolframの手元には数十cmはあろうかというMACSYMAのソース・リストがあったので，これらもまた時間の問題であろう。このことからもわかるように，SMPはMACSYMAを踏襲するもので，デモを見た限りでは、使い勝手の面では，かなりMACSYMA(残念ながら筆者には使用経験はないのだが)の影響をうけているようである。

スピードの面では，Cによる記述にも依るのだが，ペイヂングを考慮して連続領域にデータを作ることを意識していること(そのためには複製をつくることも辞さない)もあって，かなり高速である。Wolfram自身のinformalな話では、KL-10 MACSYMAの5倍程度のスピードとのことである。

データの圧縮は撤底して煮つめてはいない(というより，今後の柔軟性のために1word程残している。筆者らがおしかけた時の話では、カラー・ディスプレイでの色の指定に数bit使おうかと考えているとのこと。MAの方向へは未だ向いていない)が，かなり考慮しているようで，Lispでtagを用いるようにbit操作(Cを用いたことの利点)も行っている。UNIXの提供する8MBの空間を用いれば、かなりの大きさの式を扱うことが可能であろう。筆者がここで気になるのは，何ペイヂ(勿論メモリの，=512B)にもわたる式を扱う場合，連続領域という点でどうするのか，そしてスピードにどの程度影響してくるのかということである。(余談だが，前件(この誌面で)のように，REDUCEでは11MBの空間を用いた例もあるが，REDUCEのデータ構造自体問題があるので比較は難しい。)

細かい点では，一般の関数(REDUCEのoperator)はprojectionと言って，変数に対するリストをもち，パターン・マッチングにはhashingを用いて高速化をはかっている(詳細は不明)。また、エディタももっているし，新たな構文の導入はYACCを使っているので容易とのこと。

驚くべきは、未だbugがいるとはいえ、設計を始めて('79-11月)から一年半程でマニュアルも含めて(英語国民でNROFF/TROFFをもつことは大きい。日本語処理が先か?)システムとしてでき上がってしまっていることである。(Cで一日数千行書いたこともあったとか!)

[私見] これからのシステムを考えた場合、現況を確立されたと見るか,アルゴリズム面での行詰りと見るかにもよるが，今回示された2つの方向が有力であろう。一方はMAの方向で、その上に多くのpackageが作られ有機的な共存をした時数学の強力な武器となろう。もう一方は、SMPのように、確立した部分を総括し，小型化/高速化することであろう。さて、日本では?

☆REDUCE user's meetingから:本年末には，因数分解，不定積分(整理関数も含む),,NETFORM(巨大疎行列)，SOLVE(求根)をいれた新版をリリースの予定。S-Lispの改善，VAXへの移植も進行中。(以上，村尾)

§3. 基本的演算のアルゴリズム

本章では多項式・有理式や行列など，基本的演算に関するアルゴリズムを概観する。微分方程式や不定積分など，高度な演算の処理は次章で述べる。

因数分解も含め，基本的演算のアルゴリズムに関して今回の会議で発表された論文は，招待論文と群論関係を除いても15編にもなるが，画期的なアイデアの提出は認められなかった。強いて挙げれば，多変数多項式に対するヘンゼル構成の lifting に関して，新しい方法を提案した Zippel の論文と，数式処理の世界に並列計算をもちこんだ Sasaki と Kanada の論文であろうか？以下，おもしろそうな論文を紹介する。

V. Strassen, "The computational complexity of continued fractions"

ここで言う連分数とは，次のように多項式剰余列に関するものである。A_1 と A_2 を x の多項式とし，$A_2 \neq 0$ として除算により次々と剰余を計算する：

$$A_1 = Q_1 A_2 + A_3, \quad \deg(A_3) < \deg(A_2)$$
$$A_2 = Q_2 A_3 + A_4, \quad \deg(A_4) < \deg(A_3)$$
$$\cdots$$
$$A_{t-1} = Q_{t-1} A_t.$$

このとき次式が成立する：

$$A_1/A_2 = Q_1 + 1/(Q_2 + 1/\cdots + 1/(Q_{t-2} + 1/Q_{t-1})\cdots))$$

$(Q_1, \ldots, Q_{t-1})$ を A_1/A_2 の連分数と名づける。連分数の計算は多項式剰余列の計算と本質的に等価であり，多項式剰余列に基ずくGCD計算法と密接に関係している。剰余列計算の高速アルゴリズムとして，整数に対しては Schönhage と Knuth の方法があり，それを多項式に拡張したものとして Moenck の方法がある（章末の文献1参照）。Strassen は，これらの方法が乗除算の個数に関して最良のものであることを示した。しかし，その証明にはかなり高度の数学的理論が使われており，計算量の理論に不慣れな我々には分りにくい。

R. Zippel, "Newton's iteration and the sparse Hensel algorithm"

ヘンゼルの補題はニュートンの逐次近似法の観点からフォーミュレイトすることができ，その方が線型方程式や微分方程式などに容易に一般化できる。ここで，ニュートンの逐次近似法とは，方程式 $f(x)=0$ の近似根 $x^{(n)}$ が与えられたとき，よりよい近似根 $x^{(n+1)}$ を

$$x^{(n+1)} = x^{(n)} - f(x^{(n)})/f'(x^{(n)})$$

で計算し，これを反復する方法である。解くべき方程式が複数個で

$$F_i(x_1, \cdots, x_k) = 0, \quad i = 1, \cdots, k$$

であるときは，反復公式は

$$(x_1^{(n+1)}, \cdots, x_k^{(n+1)}) = (x_1^{(n)}, \cdots, x_k^{(n)}) - (F_1, \cdots, F_k)\begin{pmatrix} \frac{\partial F_1}{\partial x_1} & \cdots & \frac{\partial F_k}{\partial x_1} \\ \vdots & & \vdots \\ \frac{\partial F_1}{\partial x_k} & \cdots & \frac{\partial F_k}{\partial x_k} \end{pmatrix}^{-1}$$

とすればよい。Zippel はこの方法をヘンゼルの補題風にやき直して，多変数多項式に対するヘンゼル lifting 法を作成した。さらに，多変数多項式の因数分解に対して，因子を未定係数項（一般に多項式）を用いて表現しておき，これらの未定係数に対する連立方程式を上述の方法で解くことを提案した。

J. van Hulzen, "Breuer's grow factor algorithm in computer algebra"

計算機で多項式や行列処理をする際の最大の問題点は，数式の巨大化である。これを避ける最も簡単な方法は，共通部分表式を次々と新しい変数でおきかえることである。この方法の一種である Breuer のアルゴリズムを発展させ，インプリメントしてその有効性をテストした。

M. Kaminski, "Note on probabilistic algorithms in integer and polynomial ..."

多項式 $P(x_1, \cdots, x_n)$ が零に等しいかどうかをチェックする方法として，1979年

ムが効率がよいが，問題の一つは因子多項式の係数の上限である。簡単な評価によると，実際上の上限が10であるような多項式でも，理論的上限は10^{10}になることもある。著者は一貫して多項式に関するこの種の問題を研究しており，理論的上限値を著しく改善した。本論文では，上述の上限に対して新しい理論式を提出するとともに，整数係数方程式の根の最小間隔の下限に対して新しい理論式を与えた。後者は根を数値・代数的に分離するとき有用。

J. Davenport & B. Trager, "Factorization over finitely generated fields"

数式処理のアルゴリズム研究には，大別して，数学的により高度なものを追求する，効率化を追求する，の二通りの流派がある。著者らは前者の代表的存在で，論文の内容はかなり数学的である。まず，代数的無限次拡大体上での因数分解は常に可能とは限らないことを，計算可能で決定問題不可解な関数の存在を主張するKleeneの定理を利用して指摘した。一方，有限次代数的拡大体については，標数0の（例えば有理数体の）拡大体上では因数分解は可能であり，TragerやWangによりアルゴリズムが論じられている（章末の文献2参照）。しかし，標数pの拡大体の場合は単純ではない。著者らは，1950年代の数学の成果をもとに，この場合も因数分解可能であると指摘している。さらに，陽に生成される代数的閉包上での因数分解も論じている。

M. Pohst & D. Yun, "On solving systems of algebraic equations via ideal basis and elimination theory"

応用で頻繁に現れる代数的計算は多項式係数の連立線型方程式と連立非線型方程式で，本論文では後者を論じる。問題は，終結式で一つずつ変数を消去して一変数方程式に帰着させることにより，原理的には解ける。しかし，この方法では方程式の次数が著しく増大し，変数の数が増えると計算は極端に困難になる。一方，いくつかの多項式の組 $F=\{f_1,\cdots,f_k\}$ に対して，$f_i=0$, $i=1,\cdots,k$，の条件下で任意の多項式を簡単化したとき，答が常に一意的ならば，FはGröbner basisと呼ばれる。与えられた多項式の組$\{\tilde{f}_1,\cdots,\tilde{f}_k\}$からGröbner basisを構成する簡単な方法は，1969年にBuchbergerが与えた。著者らは，このアルゴリズムと終結式を組合せて，効率的な解法を得た。

P. Wang, "A p-adic algorithm for univariate partial fractions"

通常の部分分数分解の計算では有理数係数の多項式が扱われる。Wangのアイデアは，pを素数として mod p で有理数を整数に"落とし"て部分分数分解する。ついでpを$\to p^2 \to p^3 \cdots$とあげて各因子の係数をliftし，最後に有理数に逆変換する。pは語長一杯にとるので，実際上はliftingをほとんどしないでよいとのことである。

H. Kung, "USE of VLSI in algebraic computation: Some suggestions"

著者は論文多産家で，最近はVLSIアルゴリズムの提唱者で知られている。近年のVLSIブームに目をつけ，VLSIチップ上に簡単なアルゴリズムをパイプライン的ハードウェアとして組込もうというのである。論文には，多項式の加減乗除のVLSIアルゴリズムが詳述してある。

章末文献

1) 佐々木建昭，情報処理叢書・数式処理，情報処理学会発行，1981年5月.
2) 数学セミナー増刊，入門・現代の数学13，計算の効率化とその限界，§5.2.

のEUROSAMでSchwartzが確率的アルゴリズムを発表して注目をひいた。著者はこの確率的アルゴリズムの計算量を細かく調べ，アルゴリズムを詳細化するとともに，法を多項式の場合にまで拡張した。

S. Winograd, "Algebraic construction of algorithms"
著者は高速アルゴリズムの"手品師"の一人で，種々のアルゴリズムを開発している。その中で，多項式乗算や行列積，フーリエ変換などに対するいくつかの基本的考え方をsurveyした。

J. Smit, "A cancellation free algorithm, with factoring capabilities, for efficient solution of large sparse set of equations"
著者は，電気回路網解析への応用を目的に，スパースな記号行列の数式処理アルゴリズムを一貫して研究している。彼の作成したシステムは，共通部分表式おきかえ手法により，20～50次の巨大な応用例も比較的短時間で計算できるまでになっている。本論文では，小行列式展開法で同じ小行列の再計算を避けることにより得られる共通因子に加えて，さらに共通因子をとり出す方法を述べた。さらに，それらの共通部分表式を新しい変数でおきかえる際，もとの変数に関するべき展開をどう扱うかを述べた。

T. Sasaki & H. Murao, "Efficient Gaussian elimination for symbolic determinants and linear systems"
記号行列式をガウスの消去法で計算すると中間表式膨張がひき起こされ，それは要素が多変数多項式のとき著しい。そのため，ガウスの消去法はそのような行列には向かないとされていた。著者らは，対角要素を新変数で置きかえ巧妙なテクニックにより，中間表式膨張なく計算できることを示した。

T. Sasaki & Y. Kanada, "Parallelism in algebraic computation and parallel algorithms for symbolic linear systems"
代数的計算の並列処理を，1)基礎となる言語（LISPを想定）の並列処理，2)アルゴリズムで明示される並列性，の二つの観点から論じ，前者は有望ではないと予想している。後者については，分割征服法とモジュラー型のアルゴリズムをとりあげ，分割征服法は並列度は高いがプロセッサの稼動率は低く，モジュラー型のアルゴリズムを推奨している。並列アルゴリズムが将来最も必要視されるであろう代数的計算として，線型代数をとりあげ，遅延評価小行列式展開法（章末の文献1参照）がデータフロー型の計算機で効率よく計算できることを指摘した。さらに，行列式に対するラプラスの一般展開公式をもとに，モジュラーアルゴリズムと同様の高効率が期待できるアルゴリズムを，行列式計算と線形方程式の解法に対して与えた。

E. Kaltofen, D. Musser & B. Sounders, "A generalized class of polynomials that are hard to factor"
ベールカンプ・ヘンゼル型因数分解法では，mod p で因数分解し，mod p^2, …, mod p^k と"lift"して整数上での因子を求める。そのアルゴリズムが最も困る多項式は，整数上では既約であるのに任意の素数 p に対して多数の因子をもつ多項式である。著者らはそのような多項式のクラスの存在を証明し，具体的に多項式を構成してみせた。

M. Mignotte, "Some inequalities about polynomials"
整係数多項式を因数分解する場合，ベールカンプ・ヘンゼル型のアルゴリズ

§4 微分方程式，その他

An extension of Liouville's theorem on integration in finite terms

M.F. Singer, B.D. Saunders, B.F. Caviness
N.C. State U., RPI and GE Research,

Rischの決定手続きを特殊函数の適当な族を含むものまで拡張する可能性を最初に述べたのは1969年のMosesの論文であり，この方向への1歩は1979年のMosesとZippelによる論文であった。この論文はRischの決定手続きの元の形であるLiouvilleの定理を次の特殊函数を含む函数族に対して拡張した：誤差函数，Fresnel積分，初等函数の積分に現われる対数積分。しかし重対数函数（Spence函数）は含まない。ただしこの定理の証明はもっと後に出す論文に載せる予定である。

Formal solutions of differential equations in the neighborhood of singular points

J. Della Dora and E. Tournier
IMAG, Grenoble, France

有理函数係数のn階線形同次方程式の各特異点（確定でも不確定でもよい）における形式解を求めるために，まずJ.P. RamisとB. MalgrangeのNewtonの多角形を用いる方法により特異点の種類を判定し，確定特異点の場合に帰着させる。次にS. Watanabeの手続きにより特異点における特性方程式を整数係数の範囲で因数分解して特性方程式の2根の差が整数か否かを判定してFrobeniusの一般の方法を用いる。

Elementary first integrals of differential equations

M.J. Prelle and M.F. Singer
RPI and N.C. State U., USA

連立微分方程式の解を初等函数で表わすのは一般には不可能であるし，また表わすのが良いともいえない。しかし時には解曲線上で定数となる初等函数（第1積分）を見つけることが可能である。第1積分は解がはっきり表わさない隠れた性質を明らかにしてくれる。この論文は初等函数で表わされる第1積分を見つける決定手続きを見つける試みの第1歩を踏み出し，この問題と同値な未解決の問題を提出した。

A technique for solving ordinary differential equations using Riemann's P-functions

S. Watanabe, Tsuda College, Japan

有理函数係数の2階線形常微分方程式を適当な変換を用いて可能なかぎり簡単な形にし，次でそれが超幾何方程式または合流型の超幾何方程式であれば，それぞれRiemannのP函数，福原のP函数の解法を用いて初等函数または特殊函数でその解が表わされるか否かを判定し，表わされるならその表現を求める。以上が論文の内容であるが，発表ではKamkeの微分方程式を集めた本の中の関係する方程式の82%以上が以上の方法に数個の変数変換を採用することで解けることを示した。

Using Lie transformation groups to find closed form solutions to first order ordinary differential equations

B. Char, Argonne National Laboratory, USA

19世紀にS. Lie他は1階の常微分方程式の解曲線を解曲線に移す連続変換群が見つかれば元の方程式は求積法で解けることを示した。この論文は，多くの場合に解曲線を前もって求めておかなくても変換群を求める手続きが存在することを示した。変換群を求め求積法を実行する手続きはKamkeの微分方

程式を集めた本の中の1階の方程式の約58%の方程式を解くことができることを報告している。

The automatic derivation of periodic solutions to a class of weakly nonlinear differential equations

J.P. Fitch, A.C. Norman, and M.A. Moore,
U. of Bath and U. of Cambridge, England

天体力学の古典的な問題にDelaunayの月の運動の研究がある。これは数学的には微分方程式 $y''+y=\varepsilon g(y,t)+f(t)$ の解を $y=y_0+\varepsilon y_1+\varepsilon^2 y_2+\cdots$ の形で求める摂動問題である。解法はよく知られており、応用もよく知られている。またこのための特別な数式処理システム(CAMAL等)が作られた。この論文では、user levelで考えたときこの種の問題の量のぼく大さ、パラメータ数の多さにより必要になった特殊目的パッケージを作成したことの報告とそのあるべき姿の考察をしている。

An implementation of Kovacic's algorithm for solving second order linear homogeneous differential equations

B.D. Saunders, RPI, USA

Kovacicは複素係数の x の有理式 a, b, c を係数とする方程式 $ay''+by'+c\cdot y=0$ が積分・指数函数・代数函数で表わされる解を持つときはそれを求め、持たないときはないという手続きを与えた。まず $a\cdot u''+b\cdot u'+c\cdot u=0$ を変換 $u=\exp(\int -b/2a\,dx)\,y$ により方程式 $y''=ry$ に直す。微分体のGalois理論を用いてKovacicは $y''=ry$ が初等函数で表わされる解を持つことと、$y=\exp(\int \omega\,dx)$ ここに ω は複素数係数の x の有理式上の1, 2, 4, 6, 12次の代数函数と表わされることが同値である、すなわち同じ ω がRicattiの方程式をみたすことと同値であることを示した。Kovacicの手続きはこの事実に基いている。

Construction of nilpotent Lie algebras over arbitrary fields

R.E. Beck and B. Kolman,
Villanova U. and Drexel U., USA

n 次元の巾零(べきれい)Lie代数を n より小さい次元の巾零Lie代数の中心拡大として得るための効率良い手続きの概要とその1応用として6次元の実巾零Lie代数の完全な表の作成報告。1958年以来このような表は4つMorozov, Shedler, VergneとSkjelbred and Sundによって作られて来た。ところがこのどの2つも正確には一致していなかったが、その問題に終止符を打ったと述べている。

Algorithms for central extensions of Lie algebras

R.E. Beck and B. Kolman,
Villanova U. and Drexel U., USA

この論文は次の2つの問題を取扱う手続きについて論じている。(1)与えられた巾零Lie代数をより低い次元の巾零Lie代数の中心拡大の有限列に分解すること。(2)すべての n 次元巾零Lie代数を、より低い次元の巾零Lie代数の中心拡大として構成すること。

Computing an invariant subring of $k[X,Y]$

R. Neuman, Washington U., USA

標数Pの体 k 上の2変数の多項式環 $k[X,Y]$ の k-線形環自己同形群を $GA_2(k)$ とする。$GA_2(k)$ は次の2つの部分群を持つ。1つはアファイン部分群 Af_2 で次の形の自己同型の群である:$(a_1X+b_1Y+c_1, a_2X+b_2Y+c_2)$ ここで $\begin{bmatrix} a_1 & a_2 \\ b_1 & b_2 \end{bmatrix}\in GL_2(k)$。も

う1つは$(vX+f(Y), wY+c)$の形をした自己同型のなす群E_2である。ここに，$v, w\in k^*$，$c\in k$，$f(Y)\in k[Y]$，k^*はkから0を除いた群．$k[X,Y]^G$はGA_2の任意の部分群Gの不変部分環とする。GがAf_2に含まれる有限群のとき$k[X,Y]$は非常に興味深い対象である。この論文はGがE_2に含まれる位数がpの中である部分群であるとき$k[X,Y]^G$は常に多項式環であることを主張し，それを求める手続きも与えている。

Double cosets and searching small groups
G. Butler, Concordia U., Canada

群の要素の数が10^4より小さい群を小さい群ということにする。1971年Bell研究所のDiminoは群の右剰余類を用いて群の要素を数え上げる手続きを発表した。この論文はDiminoの手続きを改造して右剰余類の代わりに両側剰余類を用いることを提案している。現在のところ群の要素を探す手続きにおいて，右剰余類を用いる方がより良い結果を生むけれども，両側剰余類を用いる方法を改良していけば，この方が良くなるであろうと主張している。

Views on transportability of LISP and LISP-based systems
R.J. Fateman, U. Calif. at Berkeley, USA

大きなaddress空間を有する中価格の機械が使えるようになったので，Lispのようなシステムを新しい計算機に移植する技術を試みる機会が生じたといえる。これに対して２つの扱い方がありうる。１つはLispシステムを仮想機械の上で設計製作するもので，今１つはシステムをCのような携帯可能なプログラミング言語で書き直すことである。この論文において著者は後者の方がより良いであろうと主張している。

User-based integration software
J.P. Fitch, U. of Bath, England

手続きとそれの実現物であるsoftwareは純粋数学とその応用位の差があると著者は主張している。RischとNormanが1976年に，NormanとDavenportが1979年に提出した積分手続きを実現する上でのsoftware工学・人間工学的問題について REDUCE 上の積分package を基に議論している。とくに三角函数の処理，次数の上限について述べている。

Implementing a polynomial factorization and GCD package
P.M.A. Moore and A.C. Norman
U. of Cambridge, England

REDUCE２の因数分解とGCD packageを作成するときHenselの構成法やBerlekampの手続き等を用いる。その際の種々の注意と経験をWangの提出した１５個の例題を用いて説明している。

A case study in interlanguage communication: fast LISP polynomial operations written in 'C'
R.J. Fateman, U. Calif. at Berkeley, USA

プログラミング言語Cで記述した簡単なプログラムをLispを基にした代数処理システムと協調して使用すると本質的な効率の向上が見られたことを実験結果として報告している。

以上の他に紹介しなかった論文が９編ある。内分けは応用が４編，効率が２編，半記号計算が３編。紙数の制約と了解下さい。（以上§４は渡辺）