ABSTRACTS

A SURVEY OF AVAILABLE SYSTEMS

Y. Sundblad, Royal Institute of Technology,
S-10044 Stockholm, Sweden.

No abstract received.

ALGEBRAIC COMPUTAIONS AND STRUCTURES*)

J. Davenport, Emmanual College, University of
Cambridge, England.

This lecture aims to cover the theoretical basis of computer algebra: to discuss the objects which computer algebra manipulates and the sort of manipulations it can perform. We will not go into great detail on the algorithms, and we will largely be concerned with questions like "Can X be computed" rather than "How do we compute X efficiently".

What might we want to compute with? Integers, Rational numbers give no real problem - use "bignum" arithmetic with no intrinsic limit on the size of integers. Numbers mod p (p normally, but not necessarily, prime) are easy, and very efficient if p is small. Elements of groups (and other abstract algebraic structures) are a somewhat specialised area. Polynomials (univeriate or multivariate, since a multivariate polynomial is just a univariate polynomial whose coefficients are multivariate polynomials in fewer variables. This may not be the most efficient way, however): addition and multiplication are easy – g.c.d.s are possible, but factorisation is very difficult. (Note that this contrasts with non-constructive algebra, in which the ability to take g.c.d.s implies unique factorisation, and vice versa.) Rational functions (which require g.c.d.s of polynomials for practically everything) can be very time-consuming, and there is great scope for "clever" algorithms to minimise the number of g.c.d.s.

Algebraic Extensions can cause great problems. There are many "schoolboy" fallacies to do with algebraic numbers and algebraic functions, and it is remarkably easy to write computer programs to reproduce them. One of the major problems with algebraic expressions is ensuring uniqueness - not only must we replace $\sqrt{3}^2$ by 3, but we must also regard $\frac{1}{\sqrt{5}-2}$ as equal to $\sqrt{5} + 2$, and treat $\sqrt{2}\sqrt{3} - \sqrt{6}$ as being zero. Further problems arise because algebraic extensions of unique factorisation domains are in general not unique factorisation domains.

More general functions (exponentials, logarithms etc.) can cause great problems, even in apparently trivial matters, since it is not obvious how to test two such functions for equality (why is $1- \sin^2 x = \cos^2 x$?). This whole area depends on so-called Structure Theorems, which describe the possible dependencies between such functions. Matrices should not be too difficult, but it turns out that classical (Gaussian elimination) techniques for determinants or inverses rapidly become very expensive.

We can approximate expressions by various forms of algebraic series (as opposed to numerical expressions), such as Taylor, Laurent, Puiseux series. The obvious way to do this is to expend everything to, say, the 10th. power in x, but there are techniques for computing with recurrence relations for the series, so that another term can always be obtained relatively cheaply. If time permits we will also describe some applications of the theory of summation towards accelerating the convergence of such series.

Suggested Reading List

A. C. Norman, Computing with Formal Power Series, ACM Transactions on Mathematical Software 1 (1975) pp. 346-356.

R.E. Zippel, Univariate Power Series Expansions in Algebraic Manipulation. Proc. 1976 Symposium on Symbolic & Algebraic Computation.

THE ROLE OF THE DAP IN SYMBOL MANIPULATION

R. Beardsworth, Dept. of Computer Studies,
University of Leeds, Leeds LS2 9ST, England.

An overview of the ICL Distributed Array Processor is given together with descriptions of a data structure and algorithms for a simple symbol manipulation system. These are followed by a description of a future, more general system.

THE SCRATCHPAD PROJECT: ITS PRESENT STATUS

J. Davenport, Emmanual College, University
of Cambridge, England.

No abstract received.

ATTEMPTS AT IMPLEMENTING MACSYMA/370

A.C. Norman, University of Cambridge, Computer
Laboratory, Corn Exchange Street, Cambridge
CB2 3QG, England

In mid-October 1980 the MACSYMA group at LCS sent a tape containing the source code for MACSYMA to Cambridge. This talk will discuss the problems that surfaced in attempting to run the code, and will indicate how much progress has been made towards an IBM implementation of the system.

A SIMPLIFIED PROOF OF THE CHARACTERIZATION THEOREM FOR GRÖBNER-BASES

L. Bachmair and B. Buchberger, Johannes
Kepler Universität, A-4045 Linz, Austria.

In [2] a certain type of bases ("Gröbner-bases") for polynomial ideals has been introduced whose usefulness stems from the fact that a number of important computability problems in the theory of polynomial ideals are reducible to the construction of bases of this type. The key to an algorithmic construction of Gröbner-bases is a characterization theorem for Gröbner-bases whose proof in [2] is rather complex. We present a simplified proof. The simplification is based on two new lemmas that are of some interest in themselves. The first lemma characterizes the congruence relation modulo a polynomial ideal as the reflexive-transitive closure of a particular reduction relation ("M-reduction") used in the definition of Gröbner-bases and its inverse. The second lemma is a lemma on general reduction relations, which allows to guarantee the Church-Rosser property under very weak assupmtions.

References

[1] L. Bachmair and B. Buchberger, A Simplified Proof of the Characterization Theorem for Gröbner-Bases, ACM SIGSAM Bulletin vol. 14 No. 4 (November 1980).

[2] B. Buchberger, A Theoretical Basis for the Reduction of Polynomials to canomical Forms, ACM SIGSAM Bulletin vol. 10 No 3 (August 1976), pp. 19-29

COMPILING ALGEBRAIC ABSTRACT DATA TYPES WITH HORN CLAUSES

M. Bergman, Faculté des Sciences de Luminy,

Case 901, F - 13009 Marseille, France.

Our aim is to present the denotational and procedural semantics of an Algebraic Abstract Data Type (AAT) in first order logic. An AAT is considered as an interpreter, the semantical actions of which are rewriting rules. The power of the methodology is used to construct hierarchical types, including genericity.

We start describing the formal semantics of

an AAT in terms of a rewriting rules system. After recalling classical results we introduce the notion of T-reducibility i.e. local reduction via a type T, permitting term-normalization for complex types.
Then we show how the reduction property, using substitutions "equals by equals", may be programmed with Horn clauses from a specification "à la Guttag". Or more precisely, how it may be automatically interpreted in Prolog language.
Finally we describe how this methodology respects the independence of the types and how it authorizes the implementation of hierarchical types in a way which takes its inspiration partly from the Martin-Lőf's theory of types (1972-80) and partly from Burstall and Goguen (1977).
Our work is an attempt to implement SAM as constructive mathematics, as in tact suggested by R. Loos (EUROSAM 74). This general viewpoint may be considered, for both user and designer, as an unique programming language. These ideas are under implementation, the main features are workable.

STATE-SPACE SETS, STATE-SPACE GRAPHS AND N-PREFIX EXPRESSIONS

V. Kőfalusi and E. Halmay, CSO International
Computer Education and Information Centre,
1502 Budapest 112, P.O. Box 146, Hungary.

We intend to discuss a method for solving a crucial problem in formule manipulation. This problem arises when a tree is elected - as the best-fitting data structure - for the representation of a formula.
The usage of trees always involves the permanent danger of a combinatorial explosion in storage requirement. We outline some solutions trying to grasp this problem from different points of view. However, these solutions are connected to each other by their common theoretical background.
We start introducing some new set theory notions we need.
Then we discuss implementation problems of a programming language, PROLOG (see in [1,2]), i.e. those theoretical considerations which are expected to result considerable improvements for PROLOG's implementations. Finally we briefly mention a new concept mathematical data structure and its usage.

References:

[1] R. Kowalski, Logic for problem solving, North-Holland, New York (1979).

[2] P. Szeredi, I. Futó, PROLOG reference manual (Hungarian), Journal SZÁMOLOGÉP VII No. 3-4 (1977), pp. 5-130.

[3] V. Kőfalusi, E. Halmay, State-space sets, state-space graphs and N-prefix expressions, Report CSO-ICEIC (1980).

THE FAST-LOADING MODEL FOR SLISP/360

J.P. Fitch, University of Bath, University
Computer Unit, Claverton Down, Bath BA2 7AY,
England.

In the construction of large packages on top of LISP it is necessary to consider how to load only those parts of the system that are required The presentation considers the way in which this is done on the SLISP/360 system, including the version written entirely in LISP for system bootstrapping. The introduction of this loader has led to a simplification in the system generation of REDUCE and other packages, and will save machine time in future.

The mechanism has a number of peculiar features that are dependent on the IBM 360 architecture and the original design of Stanford LISP/360.

UTILIZATION OF SECONDARY MEMORY IN RUNNING OF THE REDUCE-2

S.G. Kadantsev and V.A. Rostovtsev, Joint
Institute for Nuclear Research, Dubna, USSR.

The algebraic computation system REDUCE-2 is running on the CDC-6500 and ES-1040 at JINR [1]. The host language on CDC-6500 is UT LISP 4.1. This language and its interpreter was designed in the University of Texas to run on the CDC-6000/7000 computer series under CRONOS and NOS operating systems. An adaptation of the interpreter was needed to get to run the REDUCE-2 on CDC-6500 under NOS BE 1.0 and this adaptation was done at JINR [2].

The REDUCE-2 system occupies a large amount of core memory: at least 64K words of CDC-6500. As this operating system at JINR allocates only 48K words for a user's job it makes the REDUCE-2 difficult for utilization. In this connection we dicided to use UT LISP 4.1 means to store interpreted functions on the disk and dynamically return of them into core memory by call [3]. For this purpose it was necessary to make further modifications of the interpreter. Our works is similar to the work reported by P.W. Milne for CYBER-76 under the SCOPE 2.1.4 operating system [4].

The employment of mentioned means of LISP enables us to work with the REDUCE-2 system in interpreting mode only. Together with an intensive exchange with the external memory it causes an expence of the machine time. To decrease that expence we are planning to implement analogous facilities for compiled functions in future and to bring them into the LISP programming system on the ES computer series.

We also started to study of the problem of the BESM-6 [5] paged memory using. For this purpose we brought simple means of paged external exchange into a LISP interpreter on this computer. The preliminary results show the little efficiency of those means. We intend to test more perfect exchange algorithms and above - mentioned means of virtual functions. In the case of success we shall be able to use the REDUCE-2 system on the BESM-6 computer as well.

References

[1] R.N. Fedorova, in: International Conference on Systems and Techniques of Analitical Computing and Their Applications in Theoretical Physics, JINR D 11-80-13, Dubna,(1980), pp. 46-57.

[2] V.A. Rostovtsev, in: Meeting on Programming and Mathematical Methods for Solving the Physical Problems, JINR D 10, 11-11264, Dubna,(1978), pp. 175-179.

[3] LISP Reference Manual CDC-6000. The University of Texas at Austin, Computer Center, CCUM-2, (1975).

[4] P.W. Milne, REDUCE on the CDC CYBER 76, REDUCE Newsletter, No. 2, (April 1978), Univ.of Utah, Symbolic Computation Group, pp. 2-3.

[5] L.N. Korolev, Computers architecture and software, Fizmatgiz, Moscow, (1978).

THE OPTIMISATION OF USER PROGRAMS FOR AN ALGEBRAIC MANIPULATION SYSTEMS

R.J. Hicks, P.D. Pearce, School of Electronic Engeneering & Computer Science, Kingston Polytechnic
Surrey, England KT1 2EE.

Users of Algebraic Manipulation Systems frequently find that they are unable to obtain answers from programs (written in some user interface language,U) that are both syntactically and algorithmically correct, through either lack of space or of computer time. There are many ways in which programs in U may be optimised by transforming them to more efficient programs in U. At present no documentation exists to guide the inexperienced user in writing efficient algebraic programs. In fact, a detailed knowledge of the working of the Algebra System is necessary to achieve efficiency. It seems unreasonable for a physicist, say, with an algebraic problem to solve to have to grapple with more than the task of writing a correct program. Even with understanding of a particular Algebra System, many optimisations would be very tedious to incorporate and would obscure the algorithm. Futhermore the differences between Algebra Systems means that some optimisations are peculiar to a particular system. To investigate optimising transformations of user programs for Algebraic Manipulations Systems a widely available, general purpose system REDUCE has been chosen for study. This paper attempts to list some optimising transformations for user programs These transformations may then be applied manually. However the authors hope to automate the process. Optimisation for programs written in a numerical language, e.g. FORTAN are well documented and concentrate on time optimisation, this being the most significant problem in this area. These optimising transformations provide a starting point for investigating optimisation of algebraic programs. When considering the optimisation of algebraic programs space saving is as significant as time saving. Currently there is little literature available on space optimisation.

The very nature of algebraic programs also opens them up to optimisations that would not be possible in a numerical system. With algebraic programs it is possible to rearrange the calculations.

e.g. In $\sum_{i=1}^{10} \int x^i dx$

We may evaluate $\int x^i dx$ as $\frac{x^{i+1}}{i + 1}$ and then execute $\sum_{i=1}^{10} \frac{x^{i+1}}{i + 1}$ saving nine calculations of the integral.

Other optimisations, such as avoiding gcd calculations, are peculiar to algebraic programs and these are discussed.

The dramatic effect of these optimisations is illustrated with timings of REDUCE programs.

Reference:

R.J. Hicks, P.D. Pearce, The Optimization of User Programs for an Algebraic Manupulation System, Internal Report RJH/PDP/1, Kingston Polyttechnic (1980).

FORMAL MANIPULATIONS : GRAMMARS AND PROGRAMS

J. Beney, G. Caplat and L. Frécon, INSA,
20 Avenue Albert Einstein,
F - 69621 Villeurbanne Cedex, France.

It is well known that optimizations during compilation (in fact a kind of formal manipulations) can be described as rewriting contextual rules. Affix grammars (in particular 2 level grammars) allow to write down these contextual transformations. Koster [9] has established that programming languages can be built with reference to these grammars, the rules of which describe the algorithmic structure of the translation process. Such a language (LET: Langage d'Ecriture de Transducteurs) has been implemented [1,2,3] and is actually compiled using PL/1.

We intend to show how to pass from rewriting contextual rules to a program LET via an affix grammar. Two examples illustrate this process. The first deals with compiling optimization technices like canomical form setting and simplification of arithmetic expressions; the second, which is more "symbol manipulation" oriented, concerns the transformation of a rational function into continued fractions [4,8].

In both cases the resulting program is a compiled program which applied desired rewriting rules after a syntactical recognition defined in a wider frame, the Backus/Naur form.

References:

J. Beney, Langage d'Ecriture de transducteurs, Thesis Université de Lyon I.

J. Beney, L. Frécon, Manuel de reference LET, Internal Report INSA (1979).

J. Beney, L. Frécon, Langage et Système d'Ecriture de Transducteurs, to appear.

G. Caplat, Arithmetique d'intervalles, Internal Report INSA (1979).

A. Colmerauer, Metamorphosis Grammars, in Natural Language Communication with Computers Springer Verlag (1978).

R.M. Cowan, M.L. Griss, Hashing - the key to rapid pattern matching, Symbolic and Algebraic manipulation, Springer Verlag (1979).

R. Floyd, An algorithm for coding efficient arithmetic operations, CACM (Jan. 1961).

P. Henrici, Einige Anwendungen der Kreisschreiben arithmetik in der Kettenbruchtheorie, in Interval Mathematics, Springer Verlag (1975).

C.H.A. Koster, Affix Grammars, Algol 68 implementation, North Holland Pub. Company (1971).

ALGORITHMS FOR SOLVING DIFFERENTIAL EQUATIONS IN FINITE TERMS*)

B.F. Caviness, General Electric Company, Corporate Research and Development, Schenectady, N.Y. 12345, U.S.A. and Rensselaer Polytechnic Institute, Troy, N.Y. 12181, U.S.A.

This talk will survey the current status of algorithmic methods for performing indefinite integration and solving differential equations in closed form. The capabilities built into the MACSYMA, REDUCE, and SCRATCHPAD computer algebra systems will be described and a few computational examples will be presented. Some of the mathematical underpinnings of the integration and ODE (Ordinary Differential Equation) algorithms implemented in computer algebra systems will be presented including aspects of the Liouvillian theory of elementary functions, the Risch integration algorithm, recent work on th integration of algebraic functions, structure theorems for simplification of transcendental functions, Kovacic's algorithm for the algebraic solution of second order linear homogeneous ODE's with rational function coefficients, and Singer's method for elementary function solutions of general linear homogeneous ODE's.

If time permits, a small problem, typical of this research area, will be treated in some depth. An example is the problem of finding an efficient algorithm to compute the minimal algebraic extension of Q(x) that contains the arguments of the logarithmic terms in the integral of a given rational function.

Reading List

Sections 10 and 5 of Lecture Notes in Computer Science, No. 72: Symbolic and Algebraic Computation, E.W. Ng (editor), Springer-Verlag, (1979).

H.I. Epstein and B.F. Caviness, "A Structure Theorem for the Elementary Functions and Its Application to the Identity Problem", Int. J. of Comp. and Info. Sciences 8 (Feb. 1979) pp. 9-37. Especially sections 1-3 for an introduction to some basic material.

J. Moses, "Symbolic Integration: The Stormy Decade", Comm. ACM 14, 8, (Aug, 1971) pp. 548-560. This is a survey paper on the status of integration in finite terms up to 1971.

R.H. Risch, "The Problem of Integration in Finite Terms", Trans. AMS, 139, (May 1969) pp. 167-189.

M. Rosenlicht, "On Liouville's Theory of Elementary Functions", Pacific J. Math. 65, 2 (1976) pp. 485-492.

M. Rothsein, "Aspects of Symbolic Integration and Simplification of Exponential and Primitive Functions", Ph. D. Thesis, Univ. of Wisconsin, (1976).Especially section 8. This thesis is available fromUniversity Microfilms International in Ann Arbor, Michigan and London, England.

FORMAL SOLUTIONS OF DIFFERENTIAL EQUATIONS IN THE NEIGHBOURHOOD OF SINGULAR POINTS (REGULAR AND IRREGULAR)

J. Della Dora and E. Tournier, IMAG, B.P. 53,
F - 38041 Grenoble Cedex, France.

We consider the differential operator

$$L = \sum_{i=0}^{N} a_i \frac{d^i}{dx^i}$$

where the $a_1 \in \mathbb{C}[[x]]$ are formal series.
Let 0 be a singularity of L, i.e. $a_N(0)=0$.
Suppose that at least for one index j $(0 \leq j \leq N-1)$ holds $a_j(0) \neq 0$.
If 0 is a singularity of the solutions of L the form of the formal solutions of L in the neighbourhood of 0 depends on the nature of the singularity of the operator L at 0.
If 0 is a regular singularity (or of a Fuchs type), we propose a Frobenius-like algorithm to generate the solutions, which are of the form:

$$y = x^{\lambda} (\phi_0(x) + \phi_1(x) \log (x) + \ldots + \phi_k(x)(\log(x)^k)$$

If 0 is an irregular singularity we first extract the solutions of L which may be regular. We require 2 steps:

- Application of a Newton-Ramis-Malgrange algorithm giving the number of these solution.
- Application of the previous Frobenius algorithm.

Then , we have to find the irregular solutions of the form:

$$y = e^{P(\frac{1}{x})} x^{\lambda}(\phi_0(x) + \phi_1(x)\log (x) + \ldots + \phi_k(x)(\log(x)^k),$$

where $P \in \mathbb{C}[x]$ and $\phi_i \in \mathbb{C}[[x]]$
To achieve this we use a "blow-up" of the N-R-M polygone associated with the previously mentioned algorithm, allowing to determine λ and the polynomial P.
Then the solutions are completed by using the Frobenius-algorithm.
We can also determine the formal solutions of L by the Ramis-Thoman algorithms. In generic cases these algorithms allow a resummation of the solutions.

AN ALGEBRAIC APPROACH TO THE FUNCTIONAL EXPANSIONS OF THE SOLUTIONS OF FORCED DIFFERENTIAL SYSTEMS

M. Fliess, F. Lamnabhi and M. Lamnabhi, Lab. des
Signaux et Systèmes E.S.E., Plateau du Moulon,
F-91190 Gif-sur-Yvette, France.

In engineering and in physics the difficulties related to the computation of the functional expansions of the solutions of forced non-linear differential systems have been often studied. Here a new approach is proposed which uses non-commutative variables which were introduced in computer science by M.P. Schützenberger more than twenty years ago. This approach gives a non-linear generalization of Heaviside operational calculus which is well known among engineers and can be used to get Volterra kernels by constructive methods.

OBTAINING PROLONGATION STRUCTURES FOR NON-LINEAR EVOLUTION EQUATIONS

I. Cohen and I. Frick, University of Stockholm,
Institute of Theoretical Physics, Vanadisvägen 9
S-11346 Stockholm, Sweden

The method for investigating soliton type equations invented by Wahlquist and Estabrook is briefly discussed.
How computer algebra could be of assistance is pointed out. Of especial interest is the automatization of the process of extending Lie algebras which is central to the prolongation method.

ALGEBRAIC OPERATOR, A POWERFUL FEATURE OF REDUCE AND ITS APPLICATION IN NON COMMUTATIVE ALGEBRAS

P. Gragert, Twente University of Technology,
Department of Applied Mathematics,
P.O. Box 217, 7500 AE Enschede, The Netherlands

The aim is to use ALGEBRAIC OPERATORS in non commutative algebras. This can partly be done with the help of the LET-statement or by flagging the relevant operator as NONCOM and others.
The new idea is, to use a PROCEDURE analogue to the already existing COEFF-PROCEDURE now with respect to an operator, which may occur in an expression with several different parameters. This is done with the help of a special purpose PROCEDURE. To show the correctness of this PROCEDURE a grammar is given for the relevant input parameter. Thereafter it is easy to show that the PROCEDURE will work correctly.
This new OPCOEFF-PROCEDURE is used to implement abstrct LIE-algebra and the algebra of differential forms together with exterior differentiation.

THE ROLE OF COMPUTER ALGEBRA IN ELECTRON AND LIGHT OPTICS

P.W. Hawkes, Laboratoire d'Optique Electronique
du C.N.R.S. B.P. 4347, F-31055 TOULOUSE CEDEX.

Evaluation of the aberrations of optical systems and the search for lens combinations optimized with respect to one or several parameters are laborious tasks. In particular, very heavy algebra is required to establish analytic formulae for the numerous aberration coefficients and so tedious are such calculations that, in electron optics, they very rarely go beyond the primary aberrations. The individual operations involved in such calculations are, however, extremely simple: series expansions, substitutions, rearrangements, differentiation(optional). The problem is that the expressionsto be manipulated are so very bulky.
The use of a computer algebra language is thus very tempting and at least two attemps to use such languages have been made: CAMAL by Ohiwa and by the present author and REDUCE by Goto and colleagues

in Japan. CAMAL is also being used by Lannes in connection with an optical interpolation problem. There has been some interest over the years in developing systematic procedures for obtaining higher-order aberration coefficients in both optical and electron optics. The work of Buchdahl in Australia and Rose and colleagues in Germany shows convincingly how this can be done and the vast amount of algebra involved when explicit formulae are required. It is clear that computer algebra is virtually essential here, although it has not yet been invoked, so far I am aware.

Finally, we mention the use of model fields. The distributions of refractive index encountered in electron optics can often be usefully represented by simple mathematical models, which allow the aberration coefficients to be evaluated explicity, usually by integrating expressions involving circular functions. This too is a task that can be confided to an algebra language, though the author has found that the programming effort required is somewhat greater than might have been expected given the elementary appearance of the integrals:

$$\int\left(\begin{matrix}\sin\\ \cos\end{matrix}\; mx\right)^p \left(\begin{matrix}\sin\\ \cos\end{matrix}\; nx\right)^q \ldots dx$$

USE OF SYMBOLIC CALCULUS IN TESTING A PRIORI IDENTIFIABILITY OF COMPARTMENTAL SYSTEMS

A. Bossi, Centro di Calcolo-Sez.Scientifica,
Via Belzoni, I-35100, Padova, Italy,
L. Colussi, Istituto di Algebra e Geometrica,
Univ. di Padova,
C. Cobelli and G. Romanin Jacur, Lab. per
Richerche di Dinamica dei Sistemi e di
Bioingegneria del CNR, Padova.

The problem of a priori or structural identiafibility of compartmental systems is of remarkable practical importance in several disciplines of science and industry. We approched and solved it by employing methods of symbolic calculus [11].

Many classes of biological systems, e.g. in the field of endocrinology and metabolism [1,2,3], pharmacokinetics [4,5], ecology [6,7] etc. are usefully represented by means of compartmental methods; in fact the use of these models permits to determine numerically some parameters of direct biological interest, otherwise not measureable, by means of properly designed input-output experiments. The structural or a priori identifiability problem is a neccessary preliminary step of the overall modelling and identification process in evaluating, before the performing of the chosen experiment, whether it is possible, at least from a theoretical point of view, to estimate all the unknown model parameters of interest [8,9].

The class of compartmental models can be dynamically described in the usual system theory notation as:

$$\underline{x} = A\underline{x} + B\underline{u}$$
$$\underline{y} = C\underline{x}$$

where $\underline{x}$, $\underline{u}$, $\underline{y}$ are the state vector (amount of material in the compartments), the input vector (injection of material) and the output vector (measurements of one or more compartments) respectively, where matrices $A=\left[a_{ij}\right]$, $B=\left[b_{il}\right]$, $C=\left[c_{mi}\right]$ are constrained as follows:

$a_{ij} = k_{ij} \qquad i \neq j$

where k_{ij} is the transport rate parameter from compartment j to i;

$a_{ii} = -\sum_{j \neq i} k_{ji}$

where k_{oi} is the transport rate parameter from i to the environment;

$b_{il} \geq 0$ and $c_{mi} \geq 0$

The model input-output relation may be expressed in Laplace transform:

$G(s) = Y(s)/U(s) = C(sI-A)^{-1}B$.

G(s) is a matrix of rational functions in s which contains all the informations supplied by the experiment. Every coefficient of the generic $G_{ml}(s)$ numerator or denomenator is a polynomial in the k_{ij}'s, b_{il}'s, c_{mi}'s: if it is equated to the respective numerical value obtained from the experiment, then a system of nonlinear equations in the parameters can be written. If this system of equations admits a finite number of solutions then the original compartment model is said to be system identifiable.

System identifiability is guaranteed if the Jacobian matrix $J=\frac{\partial E}{\partial P}$ is of full rank (where E is the symbolic coefficient vector and P is the parameter vector).

Parameter identifiability is achieved only if the considered system of equations admits one and only one solution.

Therefore an identifiability test is constructed in three steps:

1) The symbolic expression for G(s) is generated.
2) The symbolic expression for J is generated.
3) J is checked for full rank.

The coefficients of the symbolic expression for G(s) are computed by resorting, according to a rule derived from Mason's formule [10]. This rule binds the coefficients to cycles and paths of the compartmental graph. Their specific structure, multilinear monomials, allows an efficient representation via binary strings, implying that the differentiation operations, required during step 2, are particulary simple.

Step 3 is dominated by determinant calculations. Consequently, the string representation must allow efficient multiplication (resulting in nonlinear monomials) and addition (thus sorting). We derived a representation which, for instance, gives an $O(n'm')$-complexity for a multiplication, where n' and m' are the string-lengths of the operands.

The program is written in PASCAL. Running it on a CDC 6600, about 25 sec. CPU were required for computing a determinant of a 10 x 10 matrix, which proved to consist of about 3000 monomials.

References.

[1] J.A. Jacques, Compartmental Analysis in Biology and Medicine, Amsterdam, Elsevier (1972).

[2] E. Gurpide, Tracer Methods in Hormons Research, New York, Springer-Verlag (1975).

[3] E.R. Carson, C. Cobelli, L. Finkelstein: The identification of metabolic systems, A review in "Identification and Systems Parameter Estimation",5º IFAC Symposium (Editor R. Iserman), vol. 1, Oxford, Pergamon Press (1979), pp. 151-171

[4] J.G. Wagner, Clinical Pharmacokinetics, Hamilton, Drug Intelligence Corporation (1975).

[5] M. Gibaldi, D. Pernier: Pharmacokinetics, New York, Marcel Dekker, (1975).

[6] E. Halfon, Theoretical System Ecology, New York, Academic Press (1979).

[7] J.H. Matis, B.C. Patten, G.C. White, Compartmental Analysis of Ecosystem Models, Fairland, International Co-operative Publishing House (1979).

[8] A. Leschy, C. Cobelli, G. Romanin Jacur, Structural identifiability of linear compartmental models. in "Theoretical System Ecology" (e.d. E. Halfon), New York, Academic Press (1979), pp. 238-258.

[9] C. Cobelli, A. Lepschy, G. Romanin Jacur, Identifiability of compartmental systems and related structural properties, Mathematical Biosciences, vol 44, (1979), pp. 1-18.

[10] A. Bossi, C. Cobelli, L. Colussi, G. Romanin Jacur, A method of writing simbolically the transfer matrix of a compartmental model, Mathematical Biosciences, vol. 43, n. 3/4, (1979), pp.187-198.

[11] A. Bossi, C. Cobelli, L. Colussi, G. Romanin Jacur, Use of symbolic calculus in an identifiability problem, Report LADSEB-CNR 80-04 (September 1980).

THE ACTIVITY ON ALGEBRAIC COMPUTATION AT JINR

V.A. Rostovstev, Joint Institute for Nuclear Research, Dubna, USSR.

Individual works on using computers for obtaining analitical results of tasks took place at JINR since 1962 [1,2,3]. Regular activity in application of algebraic computation to physics and applied mathematics have been starting at JINR since 1976. It was caused mainly by the needs of theoreticians of the Institute.

By that time we had received the programming system SCHOONSCHIP from CERN. The system was installed on the CDC-6500 and become used by physicists [6,7]. Then the system was somewhat improved and afterwards replaced by the new version developed by Strubbe [8].

During his first visit to Dubna in 1976 professor A. Hearn had placed at our disposal the LISP interpreter for CDC 6000 computers series, the LISP compiler and the programming system REDUCE. In 1977 the LISP interpreter was adapted to the operating system NOS BE 1.0 [9]. In september 1977 professor A. Hearn had visited Dubna for the second time. He took part in the Meeting on programming and mathematical methods for solving the physical problems [10] and helped us in the initial state of the installation of new REDUCE version on the CDC-6500. Besides this version we also got the LISP interpreter for IBM system 360. In 1978 the REDUCE system was running on the CDC-6500 with the operating system NOS BE 1.0 and on the ES-1040 with the operating system with fixed tasks.

Due to kindness of the authors and holders of the systems we also received CLAM, SYMBAL, CAMAL systems and not long ago -FORMAC/PL 1 system. The first two are running on CDC-6500 and the last two -on ES-1040. We have also two algebraic computation systems developed in USSR on the BESM-6 computer [11,12].

Now the system SCHOONSCHIP is used most intensively. It was applied tot the solution of problems in physics, mathematics and engineering [5]. Essential results were obtained by O.V. Tarasov, A.A. Vladimirov and A.Ju. Zharkov [13,14], D.Ju. Bardin with his colleagues [15] and V.P. Gerdt [16]. V.P. Gerdt used also the SYMBAL and REDUCE systems in his investigations. We think its reasonable to quote final words of the article [15]: "...analitical computations became for us as necessary and daily as numeric calculations by FORTRAN. At present we do not carry out any of our investigations without using both of these methods in combination".

We also try to stimulate an interest in using of algebraic computations among scientists, especially among physicists. The most complete of today survey of the works in this field was published in the journal "Uspekhi Fiz. Nauk" by the scientists of JINR [17]. The program of the traditional meeting on programming and mathematical methods for solving the physical problems in 1977 [4] includes five reports on this theme. In 1979 the international conference on systems and techniques of analitical computing and their applications in theoretical physics was held at JINR. 20 reports were presented to the conference. The representatives from seven member-countries of JINR and the representatives from twenty four soviet institutes amongst them took part in this conference. The conference has shown the great interest and active work on this scope carried by the scientists of the member-countries of JINR. The conference showed also that not only existing systems are used and are developed but new researches carried out.

We intend to carry out the research on this scope at JINR both using algebraic computation and in further development of programming systems for algebraic computation. From the point of view their development systems based on the high level languages especially REDUCE seem to be the most perspective. Separate article presented to this meeting describes our efforts in solving one of the urgent tasks on the improvement of operational factors of the REDUCE system.

References

[1] H.J. Kaiser, Nucl. Phys., 43, (1963), p. 620.

[2] B.I. Sharonov, JINR 1668, Dubna, (1964).

[3] A.A. Hoshenko, JINR 11-4655, Dubna, (1969), pp. 157-162.

[4] Meeting on programming and mathematical methods for solving the physical problems, JINR D10, 11-11264, Dubna, (1978).

[5] International conterence on systems and techniques of analytical computing and their applications in theoretical physics, JINR D11-80-13, Dubna,(1980).

[6] L.V. Bobyleva et al., in [4], pp.161-165.
[7] V.P. Gerdt, in [4], pp. 166-174.
[8] R.N. Fedorova, in [5], pp. 46-57.
[9] V.A. Rostovstev, in [4], pp.175-179.
[10] A.C. Hearn, in [4], pp. 96-116.
[11] Ju. K. Demianovich, in [5], pp. 92-103.
[12] I.O. Babaev et al., in [5], pp.80-91.
[13] O.V. Tarasov et al., Physics Letters, 93B, 4, (1980), pp. 429-432.
[14] O.V. Tarasov, A.A. Vladimir, JINR E2-80-483, Dubna, (1980).
[15] A.A. Ahundov et al., in [5], pp. 170-172.
[16] V.P. Gerdt, JINR P2-80-436, Dubna, (1980)
[17] V.P. Gerdt et al., Uspekhi Fiz. Nauk, 130, 1, (1980), pp. 113-147.

A NEW APPROACH TO "PROVABLE" SOLVING A SYSTEM OF NON-LINEAR EQUATIONS

S.M. Rump, Institut für Angewandte Mathematik,
Universität Karlsruhe, Kaiserstrasse 12,
D-7500 Karlsruhe, W-Germany

Let f: $\mathbb{R}^n \rightarrow \mathbb{R}^n$ be a continuously differentiable function. We consider the problem of finding regions of $\mathbb{R}^n$ containing exactly one solution of the equations f(x)=0 and show that it can be "provable" solved using single-precision arithmetic only, assume one assembler-routine for a rounded scalar-product is available.

Reference:

S.M. Rump, Kleine Fehlerschranken bei Matrix problemen, Dr-Dissertation, Institut für Angewandte Mathematik, Universität Karlsruhe (Febr. 1980).

SYMBOLIC-NUMERIC INTERFACE

P. Kemp, Computing Lab., University of
Newcastle upon Tyne, Claremont Road,
Newcastle upon Tyne, NE1 7RU England,

No abstract received.

FACTORIZATION OF UNIVARIATE POLYNOMIALS: A STATISTICAL STUDY

M. Mignotte, Centre de Calcul de l'Esplanade,
Université Louis Pasteur, 7 Rue René Descartes,
F-67084 Strasbourg, France.

In order to obtain information about the average cost of the current algorithms to factorize univariate integral polynomials, we study the statistical behaviour of several natural functions on univariate polynomials over finite fields.

Reference

M. Mignotte, Factorization of Univariate polynomials: a statistical study, ACM SIGSAM Bulletin Vol. 14. No.4 (November 1980).

CAN RABIN'S PROBABILISTIC FACTORIZATION FOR LARGE FINITE FIELDS REPLACE THE HENSEL CONSTRUCTUION?

J. Calmet and R. Loos, Universität Karlsruhe,
Institut für Informatik I, D-7500 Karlsruhe 1,
Postfach 6380, West Germany.

Rabin has given a probabilistic algorithm for polynomial factorization in large finite fields which makes precise the ideas of Berlekamp. One would like to see whether the expensive Hensel construction in factoring integral polynomials can be replaced by Rabin's algorithm.

AN ANALYSIS OF QUANTIFIER ELIMINATION ALGORITHMS FOR THE THEORIES OF DENSE ORDER AND DENSE ORDER WITH ADDITION

R. Loos and R. Ottmann, Universität Karlsruhe,
Institut für Informatik I, D-7500 Karlsruhe 1,
Postfach 6380, West Germany.

We have implemented and analyzed a decision procedure of Ferrante and Rackoff for the first order theory of dense order (without constants) and a quantifier elimination algorithm of Collins for the first order theory of dense order under addition (with constants). Both algorithms belong to an algebraic simplifier.

COMPUTATIONAL GROUP THEORY

J. Neubüser, RWTH Aachen, Lehrstuhl D für
Mathematik, Templergraben 64, D-5100 Aachen,
West Germany

In the talk I shall try to give:

a) a short overview of existing general-purpose implementations of methods for the investigation of finite and finitely presented groups,

b) some idea of the underlying thoughts for only some selected of these methods.

Here are some references on the topic, taken, together with keywords, from a bibliograpghy of the field kept current by and available from: Dr. V. Felsch, Lehrstuhl D für Mathematik, Templergraben 64, D-5100 Aachen.

M.D. Atkinson, An algorithm for finding the blocks of a permutation group, Math. Comput. 29 (1975), pp. 911-913 .

H. Brown, An algorithm for the determination of space groups, Math. Comput. 23 (1969), pp. 499-514.
[Zassenhaus Algorithm]

G. Butler, The Schreier algorithm for matrix groups, SYMSAC '76, proceedings of the 1976 ACM Symposium on symbolic and algebraic computation (Yorktown Heights, N.Y.,1976) edited by R.D. Jenks, ACM, New York (1976), pp. 167-170.

J. Cannon, Construction of defining relators for finite groups, Discrete Math. 5 (1973), pp. 105-129.

J. Cannon, A general purpose group theory program, Proceedings of the second international conference on the theory of groups (Austral. Nat. Univ. Canberra, 1973), edited by M.F. Newman, Lecture Notes in Math., Vol 372, Springer, Berlin (1974), pp. 204-217.

J. Cannon, A draft description of the group theory language CAYLEY, SYMSAC '76, proceedings of the 1976 ACM Symposium on Symbolic and algebraic computation (Yorktown Heights, N.Y., 1976), edited by R.D. Jenks, ACM, New York (1976), pp. 66-84.

J.J. Cannon; A.L. Dimind, G. Havas, J.M. Watson, Implementation and analysis of the Todd-Coxeter algorithm. Math. Comput. 27 (1973), pp. 463-490.

H.S.M. Coxeter; W.O.J. Moser, Generators and relations for discrete groups. Ergebnisse der Mathematik und ihrer Grenzgebiete, Band 14, Springer, Berlin (1957) Third Edition(1972)

A. Dietze; M. Schaps, Determining subgroups of a given finite index in a finitely presented group, Canad. J. Math. 26 (1974), pp.769-782.
[Low index subgroups].

J.D. Dixon, High speed computation of group characters. Numer. Math. 10 (1967), pp.446-450.

V. Felsch, A machine independent implementation of a collection algorithm for the multiplication of group elements. SYMSAC '76 proceedings of the 1976 ACM Symposium on symbolic and algebraic computation (Yorktown Heights, N.Y., 1976), edited by R.D. Jenks, ACM New York (1976), pp. 159-166.

V. Felsch, J. Neubüser, Über ein Program zur Berechnung der Automorphismengruppe einer eindlichen Gruppe. Numer. Math. 11 (1968), pp. 277-292.

V. Felsch, J. Neubüser, An algoritm for the computation of conjugacy classes and centralizers in p-groups. Symbolic and algebraic computations (proceedings of Eurosam '79, an international symposium on symbolic and algebraic manipulation, Marseille, 1979), edited by E.W. Ng, Lecture Notes in Computer Science, Vol. 72, Springer, Berlin (1979), pp. 452-465.

Th. Gabrysch, Ein Computerprogramm zur Berechnung von Charactertafeln. Beschreibung des Programms "Charac". Mimeographed Notes, Version 78-2, Fakultät für Mathematik, Univ. Bielefeld, (1978), 42 pages.

G. Havas, A Reidemeister-Schreier Program. Proceedings of the second international conference on the theory of groups (Austral. Nat. Univ., Canberra, 1973), edited by M.F. Newman, Lecture Notes in Math., Vol 372, Springer, Berlin (1974) pp. 347-356.

G. Havas, M.F. Newman, Application of computers to questions like those of Burnside. Burnside groups (Proceedings of a workshop, Univ. Bielefeld, Germany, 1977), eidted by J.L. Mennicke, pp. 211-230. Lecture Notes in Math., Vol. 806, Springer, Berlin, 1980.
[Nilpotent Quotient]

G. Havas, T. Nicholson, Collection, SYMSAC '76, Proceedings of the 1976 ACM Symposium on symbolic and algebraic computation (Yorktown

Heights, N.Y., 1976), edited by R.D. Jenks, ACM, New York (1976), pp. 9 - 14. [Nilpotent quotient].

G. Havas; L.S. Sterling, Integer matrices and abelian groups. Symbolic and algebraic computations (Proceedings of Eurosam '79, an international symposium on symbolic and algebraic manipulation, Marseille, 1979), edited by Edward W. NG, Lectures Notes in Computer Science, Vol. 72, Springer, Berlin (1979), pp. 431-451. [Elementary divisor algorithm (abelian decomposition)].

J. Leech, Coset enumeration, Computational problems in abstract algebra (Proc. Conf., Oxford, 1967), edited by J. Leech, Pergamon, Oxford (1970) , pp.21-35 [Todd-Coxeter].

J. Leech, Computer proof of relations in groups. Topics in group theory and computation (Proc. summer school, Univ. College, Galway, 1973), edited by Michael P.J. Curran, Academic Press, London (1977), pp. 38-61. [Application of Todd-Coxeter].

J.S. Leon, On an algorthim for finding a base and strong generating set for a group given by generating permutations. To appear in Math. of Comp.

J.S. Leon, Finding the order of a permutation group, to appear in Proc. Sympos. Pure Math., Vol. 37. [Schreier Todd-Coxeter Sims method, Held group].

J.S. Leon; V. Pless, CAMAC 1979, Symbolic and algebraic computations (Proceedings of EUROSAM '79, an international symposium on symbolic and algebraic manipulation, Marseille, 1979), edited by Edward W. NG, Lectures Notes in Computer Science, Vol. 72, Springer, Berlin (1979), pp. 249-257.

I.D. MacDonald, A computer application to finite p-groups. J. Austral. Math. Soc. 17 (1974), pp. 102-112. [Nilpotent quotient]

J. McKay, The construction of the character table of a finite group from generators and relations, Computational problems in abstract algebra (Proc.Conf., Oxford, 1967), edited by J. Leech, Pergamon, Oxford (1970), pp. 89-100.

J. McKay, Subgroups and permutation characters, Computers in algebra and number theory (Proc. Sympos. Appl. Math., New York, 1970), edited by Garrett Birkhoff and Marshall Hall, Jr., SIAM-AMS Proc., Vol.4, Amer. Math. Soc., Providence, R.I. (1971), pp. 177-181.

N.S. Mendelsohn, An algorithmic solution for a word problem in group theory; Canad. J. Math. 16 (1964), pp. 509-516. Correction: Canad. J. Math. 17 (1965), pp. 505. [Todd-Coxeter].

J. Neubüser, Untersuchungen des Untergruppenverbandes endlicher Gruppen auf einer programmgesteuerten elektronischen Dualmaschine. Numer. Math. 2 (1960), pp. 280-292

J. Neubüser, Investigations of groups on computers. Computational problems in abstract algebra (Proc. Conf., Oxford, 1967), edited by J. Leech, Pergamon, Oxford,(1970)pp. 1-19. [Survey, Todd-Coxeter, subgroups, characters, reprentations].

J. Neubüser, Computing moderately large groups: some methods and applications, Computers in algebra and number theory (Proc. Sympos. appl. Math., New York, 1970), edited by Garrett Birkhoff and Marshall Hall, Jr., SIAM-AMS Proc., Vol. 4, Amer. Math. Soc., Providence, R.I. (1971), pp. 183-190.

J. Neubüser, Some computational methods in group theory, Third international colloquium on advanced computing methods in theoretical physics (Proc. Conf., Marseille, 1973), pp. B-II-1 - B-II-35. Centres de physique theorique C.N.R.S., Marseille, 1973. [Survey, Todd-Coxeter, low index subgroups, Reidemeister-Schreier, defining relations, nilpotent quotient, characters].

M.F. Newman, Calculating presentations for certain kinds of quotient groups, SYMSAC '76, Proceedings of the 1976 ACM Symposium on symbolic and algebraic computation (Yorktown Heights, N.Y., 1976), edited by R.D. Jenks, ACM, New York (1976), pp.2-8. [Nilpoint quotient].

M.F. Newman, Determination of groups of prime-power order. Group theory (Proc. Miniconf., Austral. Nat. Univ., Canberra, 1975), edited by R.A. Bryce, J. Cossey and M.F. Newman, Lecture Notes in Math., Vol. 573, Springer, Berlin, (1977), pp. 73-84.

[p-groups, application of nilpotent quotient].

Ch.C. Sims, Computational methods in the study of permutation groups, Computational problems in abstract algebra (Proc. Conf., Oxford, 1967), edited by J. Leech, Pergamon, Oxford (1970), pp. 169-183.

Ch.C. Sims, Determining the conjugacy classes of a permutation group, Computers in algebra and number theory (Proc. Sympos. appl. Math., New York, 1970), edited by Garrett Birkhoff and Marshall Hall, Jr., SIAM-AMS Proc., Vol.4, Amer. Math. Soc., Providence, R.I. (1971), pp. 191-195.

Ch.C. Sims, Computation with permutation groups. Proceedings of the second symposium on symbolic and algebraic manipulation (Los Angeles, Calif., 1971), edited by S.R. Petrick, ACM, New York (1971), pp. 23-28.

Ch.C. Sims, Some group-theoretic algorithms, Topics in algebra (Proc. 18th summer research Inst., Austral. Math. Soc., Austral. Nat. Univ., Canberra, 1978), edited by M.F. Newman, Lectures Notes in Math., Vol. 697, Springer, Berlin (1978), pp. 108-124.

Ch. C. Sims, Group-theoretic algorithms, a survey, Proceedings of the international congress of mathematicians (Helsinki, Finland, 1978), edited by Olli Lehto, pp. 979-985. [Elementary divisor algorithm (abelian decomposition), Todd-Coxeter, nilpotent quotient, low index subgroups, Reidemesiter-Schreier, subgroup lattice, permutation group algorithms, Schreier-Todd-Coxeter]. Coxeter].

J.A. Todd; H.S.M. Coxeter, A practical method for enumerating cosets of a finite abstract group, Proc. Edinburgh Math. Soc. (2) 5 (1936), pp. 26-34.[Todd-Coxeter].

J.W. Wamsley, Computation in nilpotent groups (theory), Proceedings of the second international conference on the theory of groups (Austral. Nat. Univ., Canberra, 1973), edited by M.F. Newman. Lecture notes in Math., Vol. 372, Springer, Berlin (1974), pp. 691-700.[Nilpotent quotient].

J.W. Wamsley, Computing soluble groups , Group theory (Proc. Miniconf., Austral. Nat. Univ., Canberra, 1975), edited by R.A. Bryce, J. Cossey and M.F. Newman. Lecture Notes in Math., Vol. 573, Springer, Berlin, (1977), pp. 118-125.

LINEARIZATION OF PRODUCTS OF STRUCTURE FACTORS IN CRISTALLOGRAPHY

Chr. de Polignac, Inst. Von Laue-Langevin,
Avenue des Martyrs, F-38000 Grenoble, France
and
J. Dulac, Lab. de Cristallographie, CNRS,
Grenoble, France.

A statistical method in crystallography for the determination of the structure factor $F(\vec{h}) = \sum \exp 2\pi i\, C_j \vec{x}$ requires the formal derivation of successive powers (up to 4) of this quantity.

This talk shows how these computations have been performed using full possibilities of REDUCE including its symbolic mode to take into account the symmetry rules relevant to the space group considered.

Because of core limitations, this talk gives some tricks to split critical examples. All these results are given for space group 201 (according to the International tables for X-ray crystallography).

SOME BASIC PROCEDURES IN COMPUTATIONAL NUMBER THEORY

H.G. Zimmer, Universität des Saarlandes,
Fachbereich Mathematik, D-6600 Saarbrücken,
West Germany.

A famous unsolved problem in algebraic number theory is Fermat's conjecture asserting that the diophantine equation

$$x^n + y^n = z^n$$

has no solutions in nonzero rational integers x, y, z when n is a positive integer greater than 2. This problem concerns the solutions of a diophantine equation in the ring Z of rational integers or, more generally, in the field Q of rational numbers. However, an attempt to prove Fermat's conjecture immediately leads to the necessity of looking at the diophantine equation over the subring R of integers of an algebraic number field K instead of merely the subring Z of Q. With this new setting of the problem some basic questions arise about the field K and its subring R.

(1) How to represent the elements of R; in other words, does R have an *integral basis*?

(2) What are the *units* and what the *irreducible elements* of R?

(3) Do the elements of R uniquely factor into products of a unit and irreducible elements; in other words, is R a *unique factorization domain*?

(4) If R is not a unique factorization domain, can the irreducible elements of R be replaced by the prime ideals of R in order to obtain unique factorization into prime ideals as a substitute for unique factorization into irreducible elements; in other words, is R a *Dedekind domain*?

(5) If R is a Dedekind domain containing Z, how to implement unique factorization into prime ideals for the ideals of R as well as those of Z; in other words, what is the *decomposition law* in R?

(6) How to decide for a given ideal of R whether or not it is *principal*?

(7) How many ideal classes do exist in the ring R; in other words, what is the *class number* of R?

(8) Do the ideal classes of R form an abelian group under multiplication and, if so, what is the structure of this group?

These questions constitute the main objective of algebraic number theory, and answers are provided by the theory. From a computational point of view, however, constructive answers are required. For carrying out actual calculations some very effective basis procedures are needed by which it is possible to determine e.g. an integral basis, the unit group, the class number and the decomposition law of R in K. The corresponding problems are only briefly discussed.

In a more detailed manner a special diophantine equation is considered over an algebraic number field K for which the basic procedures just mentioned are available. We start out with the special case n = 3 of Fermat's equation. This equation can be birationally transformed into an elleptic curve, that is, a plane cubic curve E having no singularities. Elliptic curves are of special interest because the set E(K) of points of E with coordinates in K forms an additive abelian group. The rational point group E(K) enjoys the highly important property of being finitely generated. Hence E(K) is the direct sum of a finite group $E_{tor}(K)$, the torsion group of points of finite order in E(K), and a free group $E_{fr}(K)$ with finitely many independent generators of infinite order whose number r is called the rank of E over K.

Our attention focuses on the problems of dertermining, for any given elliptic curve E over a certain algebraic number field K, the torsion group $E_{tor}(K)$ and the rank r. The tasks of finding elliptic curves E over K having nontrivial torsion groups $E_{tor}(K)$ and of constructing elliptic curves E over K with high ranks are also discussed. It is interesting to notice that the rank of E over K depends in some way on the class number of K. Solving these problems in a computational manner involves the basic procedures for carrying out calculations in the number field K mentioned in connection with the above questions (1)-(8).

ON THE COMPUTATION OF LATTICE VECTORS OF MINIMAL LENGTH, SUCCESSIVE MINIMA AND REDUCED BASES WITH APPLICATIONS

M. Pohst, Mathematisches Institut, Universität zu Köln, Weyertal 86-90, 5 Köln 41,
West Germany.

Abstract. The problem of determining shortest vectors and reduced bases or successive minima of lattices often occurs in algebra and number theory. Nevertheless, computational methods for the solution hardly exist in the literature. It is our aim to discuss how to develop efficient algorithms for this purpose.

We start with an algorithm for the determination of shortest vectors in a lattice, or - slightly more general - in a residue class of an $\mathbb{R}$-linear space modulo a sublattice of equal dimension. Then we use the theory of reduction and this algorithm to develop a method for the computation of a reduced basis of a lattice, and -closely related- the computation of successive minima. Finally we present some examples of succesful application of these new algorithms taken from entirely different fields, i.e. lattice theory, integral matrix groups and algebraic number theory.

Reference:

M. Pohst, On the computation of lattice vectors of minimal length, successive minima and reduced bases with applications, ACM SIGSAM Bulletin Vol. 14 No.4 (November 1980).

ALGORITHMS ON MONOMIAL CURVES

F. Mora, Istituto di Matematica, Università
di Genova, Via L.B. Alberti 4, I-16132
Genova, Italy.

Applying results in numerical group theory allowed Herzog [2] to give a characterization of the ideal of monomial curves $x_1=t^{n_1}$, $x_2=t^{n_2}$, $x_3=t^{n_3}$.
Using these results of Herzog in combination with some numerical observations about the equations of the curve, allows to characterize its tangent cone by applying an algorithm which is based on the cartesian equation of the curve. This leads to a classification of all curves

$$x_1=t^{n_1}, \; x_2=t^{n_1\lambda+n_2}, \; x_3=t^{n_1\mu+n_3}$$

where n_1,n_2,n_3 are natural numbers and λ,μ are natural parameters. First we survey some of our theoretical results conserning the classification of monomial curves [3], leading to a classification algorithm [4]. Then we discuss the algorithm for the computation of the tangent cone [4].

Both algorithms are implemented in BASIC on a PDP-Vo3. In this context it is worthwhile mentioning related work of Graetz [1]

References

[1] C. Graets, Ein Program zur Berechnung von Invarianten eines lokales Rings der Dimension 1, Diplomarbeit Regensburg (1979).

[2] J. Herzog, Generators and Relations of Abelian Semigroups and Semigroup Rings, Manuscripta math. 3, pp. 175-193.

[3] F. Mora, Classification of monomial curves, Internal Report Ist. Mat. Univ. Genova (1980).

[4] F. Mora, Algorithms on Monomial curves, Internal Report Ist. Mat. Univ. Genova (1980).

*) These lectures were also part of the short AMS course on Computer Algebra (see ACM SIGSAM Bulletin Vol 14 No.2 (May 1980)). The notes of this course will appear as AMS monograph.

Late abstract of presentation to be included in session 7:

BRIEF INFORMATION ON ALGEBRAIC MANIPULATION IN CSECHOSLOVAKIA

Z. Kalina, Czech. Acad. of Sciences,
Astronomial Institute, Budečská 6,
12023 PRAHA 2, Czechoslovakia.

The first experiments in algebraic manipulation by computer were performed in Czechoslovakia at the beginning of 1970s.

At first, they were not algebraic systems but only special solutions of particular problems. For instance the question of pillars elasticity in construction was solved at the University of Brno. A simple system for polynomial manipulation was designed at the Faculty of the electrical enigneering in Prague.

The FORMAC, version PL/1 was the first general algebraic system installed in Czeschoslovakia. It is used in the Institute of the Information Theory and Automation in Prague. Another algebraic system called ALITA is used for Poisson series manipulation. It is used by scientists from the Institute of Plasma Physics in the computer centre of the Academy of Sciences.

In 1977 the algebraic system REDUCE-2 was implemented in the Astronomical Institute of Czechoslovak Academy of Sciences in Ondřejov.

I should like to express our thanks to the REDUCE author - professor Hearn from the University of Utah.

The REDUCE system is implemented in batch processing version and works under operating system OS on computer machine EC 1040. In the first stage we learned how to use the system and we got acquainted with the algebraic manipulation by computer in general. Later we began to use REDUCE system for solving particular tasks, mainly the tasks from celestial mechanics.

In 1979 we obtained the latest version of REDUCE from professor Hearn. At present, the display terminals are established in the Astronomical Institute computer center and we are getting ready to carry on the Time Sharing Option system.

In connection with TSO, we want to rewrite batch processing version of REDUCE to interactive version under TSO. We expect better possibilities in programming and algorithm debugging, better testing and control of complicated algebraic expressions and manipulations.