Computeralgebra-Rundbrief

Nr. 37 Oktober 2005

Inhalt

Impressum

Der Computeralgebra-Rundbrief wird herausgegeben von der Fachgruppe Computeralgebra der GI, DMV und GAMM (verantwortlicher Redakteur: Dr. Markus Wessler, Kopernikusstr. 6, 81679 München, Telefon: 089-69777336, Telefax: 089-69777335, wessler@mathematik.uni-kassel.de).

Der Computeralgebra-Rundbrief erscheint halbjährlich, Redaktionsschluss 28.02 und 30.09. ISSN 0933-5994. Mitglieder der Fachgruppe Computeralgebra erhalten je ein Exemplar dieses Rundbriefs im Rahmen ihrer Mitgliedschaft. Fachgruppe Computeralgebra im Internet:
`http://www.fachgruppe-computeralgebra.de`.

Konferenzankündigungen, Mitteilungen, einzurichtende Links, Manuskripte und Anzeigenwünsche bitte an den verantwortlichen Redakteur.

Die Geschäftsstellen der drei Trägergesellschaften:

GI (Gesellschaft für
Informatik e.V.)
Wissenschaftszentrum
Ahrstr. 45
53175 Bonn
Telefon 0228-302-145
Telefax 0228-302-167
gs@gi-ev.de
`http://www.gi-ev.de`

DMV (Deutsche Mathematiker-
Vereinigung e.V.)
Mohrenstraße 39
10117 Berlin
Telefon 030-20377-306
Telefax 030-20377-307
dmv@wias-berlin.de
`http://www.mathematik.`
`uni-bielefeld.de/DMV/`

GAMM (Gesellschaft für Angewandte
Mathematik und Mechanik e.V.)
Technische Universität Dresden
Institut für Festkörpermechanik
01062 Dresden
Telefon 0351-463-33448
Telefax 0351-463-37061
GAMM@mailbox.tu-dresden.de
`http://www.gamm-ev.de`

Mitteilungen der Sprecher

Liebe Mitglieder der Fachgruppe Computeralgebra,

am 16. September fand die 2. Sitzung der Fachgruppenleitung 2005–2008 in Deggendorf statt. Zum ersten Mal konnten wir auf dieser Sitzung den neuberufenen Fachxperten Chemie Prof. Reinhard Laue in unserer Runde begrüßen. Auf eine gute Zusammenarbeit!

Ein wichtiges Thema auf unserer Sitzung waren naturgemäß die Tagungen der Fachgruppe. Unsere letzte wissenschaftliche Tagung, welche vom 2.–4. Juni in Kassel stattfand, hatte diesmal 42 Teilnehmer. Einen Bericht finden Sie auf Seite 6. Als Novum hatte die Fachgruppe diesmal einen Preis für den besten Vortrag eines Nachwuchswissenschaftlers ausgelobt. Der Nachwuchspreis ging an Felix Noeske aus Aachen. Herzlichen Glückwunsch!

Unsere nächste Tagung Computeralgebra in Lehre, Ausbildung und Weiterbildung V wird vom 20.–22. April 2006 (Woche nach Ostern) wieder im Haus Schönenberg bei Ellwangen stattfinden. Sie steht diesmal unter dem Thema Entdecken, Üben, Prüfen mit Computeralgebra – Neue Entwicklungen an Schule und Hochschule. Wir gehen davon aus, dass auch diese Tagung wieder gut besucht sein wird. Bitte notieren Sie bereits jetzt unsere Deadlines: Vortragsanmeldungen bis zum 15. Februar 2006, Einreichen der ausgearbeiteten Artikel sowie Überweisung der Tagungsgebühren bis zum 15. März 2006, ebenso Anmeldungen ohne Vortrag.

Am 6. und 7. Januar 2006 findet zum Andenken an die verstorbene Karin Gatermann, die Mitglied der Fachgruppenleitung war, in Hamburg ein Gedenkkolloquium statt. Die Fachgruppe beteiligt sich hierbei finanziell. Näheres erfahren Sie auf Seite 34. Im Frühjahr 2007 ist wieder eine Tagung zum Thema Computeralgebra an Fachhochschulen geplant, die von Frau Prof. Heinrich organisiert wird und die in Konstanz stattfinden wird. Ebenso planen wir mittelfristig eine Tagung zum Thema Computeralgebraanwendungen in der Industrie, für die Herr Dr. Sorgatz zuständig ist.

Schließlich haben wir uns auf unserer Sitzung nun darauf festgelegt, dass unsere Fachgruppe sich für die internationale Konferenz ISSAC 2008 mit dem Tagungsort München bewerben wird. Im Falle des Zuschlags wird die lokale Organisation von Prof. Ernst Mayr übernommen. Hierfür herzlichen Dank! Auf der diesjährigen ISSAC-Tagung in Peking (siehe Bericht auf Seite 32) war beschlossen worden, dass nach Genua (2006) die ISSAC-Tagung im Jahr 2007 in Waterloo (Kanada) stattfinden wird. Damit ist der Weg frei für eine europäische Bewerbung im Jahr 2008. Wir sind sicher, dass wir mit München ein sehr attraktives Angebot haben werden.

Auf der Tagung in Peking wurde turnusgemäß ein neues Mitglied des ISSAC Steering Committees gewählt. Das ISSAC Steering Committee ist für die Austragung der jährlichen ISSAC-Tagung verantwortlich. Neues Mitglied ist Jeremy Johnson (Drexel University). Im Augenblick hat das Steering Committee folgende Mitglieder: Gilles Villard, Mark Giesbrecht, Jeremy Johnson, Emil Volcheck (SIGSAM), Wolfram Koepf (Fachgruppe Computeralgebra), Kazuhiro Yokoyama (Japan Society of Symbolic and Algebraic Computation). Neuer Chair ist Gilles Villard. Die Mitgliedschaft von Wolfram Koepf im Steering Committee wird turnusgemäß im nächsten Jahr enden.

Die Fachgruppe ist als Mitglied der GI im dortigen Fachbereich GInf (Grundlagen der Informatik) verankert. In der Führung dieses Fachbereichs hat es Neuwahlen gegeben. Neuer Sprecher und damit Nachfolger von Ernst Mayr ist (seit letztem Jahr) Volker Diekert (Stuttgart), sein neugewählter Stellvertreter ist Thomas Wilke (Kiel).

Unser Fachreferent für Neuerscheinungen Johannes Grabmeier hat sich in der letzten Zeit verstärkt dafür eingesetzt, Neuerscheinungen aus dem Bereich Computeralgebra zu finden und Besprechungen einzuwerben. Als Resultat dieser Initiative finden Sie in diesem Heft eine ganze Reihe interessanter Buchbesprechungen.

Die Fachgruppenleitung hat zu Werbezwecken einen Flyer gedruckt. Einen Abdruck des Flyers finden Sie auf Seite 41. Wenn Sie als Mitglied der Fachgruppe beispielsweise auf einer Tagung Flyer der Fachgruppe verteilen wollen, wenden Sie sich bitte an den Sprecher Wolfram Koepf.

Wolfram Koepf *Gerhard Hiß*

Tagungsfoto

Computeralgebra: 02. – 04.06.2005, Kassel

Auch die diesjährige Computeralgebratagung der Fachgruppe war wieder gut besucht (42 Teilnehmer). Außerhalb der fünf Hauptvorträge haben in 16 weiteren Vorträgen vor allem jüngere Forscher über die neuesten Entwicklungen auf dem Gebiet der Computeralgebra berichtet. Wie auch vor zwei Jahren mussten diese Vorträge auf zwei parallele Sektionen verteilt werden.

In diesem Jahr wurde erstmalig ein mit 500 € dotierter Nachwuchspreis vergeben. Er ging an Felix Noeske für seinen Vortrag „Methoden der rechnergestützten Darstellungstheorie sporadischer Gruppen“. In seinem sehr gut verständlichen Vortrag erläuterte er am Beispiel der Fi_{23} die Probleme, die bei der Berechnung der Darstellungen von einfachen Gruppen entstehen. Ergänzend zu seinem Vortrag zeigte Max Neunhöffer in seinem Vortrag als wichtigen Teilschritt, wie man die Doppelnebenklassenvertreter einer maximalen Untergruppe in Fi_{23} bestimmt.

Mark van Hoeji (Florida State University) berichtete in seinem Hauptvortrag über die Entwicklung des algebraischen Kurvenpakets von Maple. Florian Heß (Technische Universität Berlin) berichtete in seinem Überblicksvortrag über neue Entwicklungen in der Kryptographie basierend auf elliptischen Kurven. Gerhard Pfister (Kaiserlautern) stellte das Computeralgebrasystem SINGULAR vor, dessen neue Version (`http://www.singular.uni-kl.de/`) passend zur Konferenz fertiggestellt wurde. Im Anschluss an diesen Hauptvortrag wurden von Hans Schönemann neue Entwicklungen in SINGULAR 3 in einem vertiefenden Vortrag vorgestellt. Jürgen Klüners (Kassel) erklärte in seinem Vortrag ein neues und sehr schnelles Verfahren zur Faktorisierung von Polynomen. Der letzte Hauptvortragende war David J. Green (Bergische Universität Wuppertal) über die Berechnung der Kohomologie endlicher Gruppen. Auch in den weiteren Vorträgen wurde die ganze Bandbreite der Computeralgebra deutlich.

Fachgruppe Computeralgebra

URKUNDE

Die Fachgruppe Computeralgebra verleiht
den Preis für den
besten Vortrag
eines Nachwuchswissenschaftlers auf der
Computeralgebra-Tagung Kassel 2005 an

Felix Noeske

Der Preis ist mit 500 € dotiert.

Kassel, 4. Juni 2005

Wolfram Koepf
(Sprecher der Fachgruppe)

W. Koepf, F. Noeske

Die Anwendung in der Physik war durch Katja Rykhlinskaja („Computer-Algebra-Tools for Studying the Symmetry Properties of Molecules and Clusters“) und Thomas Radtke („Simulation von Quantenregistern mit Maple“) vertreten. Aus der Kryptographie kamen Andreas Klein („Angriffe auf RC4“) und Claus Diem („Index Calculus in Klassengruppen allgemeiner Kurven“). Die computergestützte Gruppentheorie war mit Frank Himstedt („Über niedrigdimensionale Darstellungen der Steinbergschen Trialitätsgruppen“) sowie den bereits erwähnten Vorträgen von Felix Noeske und Max Neunhöffer vertreten. Die restlichen Vorträge waren „Konstruktion von nicht-hyperelliptischen Funktionenkörpern vom Geschlecht 3 und 4“ (José Mendez), „Explizite Berechnung von Drinfeld-Moduln“ (Claus Fieker), „Neue Methoden zur algorithmischen Lösung nicht-linearer ODEs über Symmetrien und integrierte Faktoren“ (Kai Gehrs), „Ein Algebra-Programm, das nicht rechnet? Überblick über das Programm HR“ (Daniel Wagner), „Counting upper interactions in Dyck paths. Solution methods for q-algebraic equations“ (Yvan le Borgne), „Neue Entwicklungen in SINGULAR 3“ (Hans Schönemann), „Konstruktion von Klassenkörpern über lokalen Körpern“ (Sebastian Pauli), „Berechnung atypischer Werte von Polynomen – Diskriminanten und Newtonpolygone“ (Christian Gorzel) und „Der Kozykelkalkül und die Realisierung von Galoisgruppen“ (Michael Dettweiler).

Auf der Homepage der Tagung (`http://www.mathematik.uni-kassel.de/compmath/ca2005.html`) findet man eine Teilnehmerliste mit den Präsentationen der einzelnen Vorträge. Es gibt ferner ein Tagungsfoto sowie eine Fotogalerie.

Andreas Klein (Kassel)

Computeralgebra in Lehre, Ausbildung und Weiterbildung V : Entdecken, Üben, Prüfen mit Computeralgebra – Neue Entwicklungen an Schule und Hochschule, 20. – 22.04.2006, Haus Schönenberg bei Ellwangen

In der letzten Zeit wurden und werden in allen Bundesländern Standards und kompetenzorientierte Kernlehrpläne für den Mathematikunterricht entwickelt. Der Einsatz von Computeralgebra (in Form von CAS-Taschenrechnern oder von Computeralgebrasystemen auf PCs) ist ebenfalls in allen Bundesländern zumindest auf freiwilliger Basis möglich. Welchen Einfluss können und sollen diese CA-Werkzeuge für den Mathematikunterricht an allgemeinbildenden Schulen und für die Anfängerausbildung an den Universitäten und Hochschulen haben? Die Tagung soll eine Bestandsaufnahme vornehmen und Perspektiven für eine weitere Entwicklung aufzeigen. Auf der Homepage der Fachgruppe (`http://www.fachgruppe-computeralgebra.de/CLAW`) finden Sie das Anmeldeformular für die Tagung und ausführliche Informationen über die vergangenen Tagungen.

Haus Schönenberg

Computation of Gröbner Bases for Systems of Linear Difference Equations

Vladimir P. Gerdt (Dubna)

gerdt@jinr.ru

Daniel Robertz (Aachen)

daniel@momo.math.rwth-aachen.de

1 Introduction

Being invented 40 years ago by Buchberger [1] for algorithmic solving of the membership problem in the theory of polynomial ideals, the Gröbner bases method has become a powerful universal algorithmic tool for solving various mathematical problems arising in science and engineering.

Though overwhelming majority of the Gröbner bases applications is still found in commutative polynomial algebra, over the last decade a substantial progress has also been achieved in application of Gröbner bases to noncommutative polynomial algebra, to algebra of differential operators and to linear partial differential equations (see, for example, book [2]). As to the difference algebra, i.e. algebra of difference polynomials, in spite of its conceptual algorithmic similarity to differential algebra, only a few efforts have been done to extend the theory of Gröbner bases to difference algebra and to exploit their algorithmic power [3, 4].

Recently, two promising applications of difference Gröbner bases were revealed: generation of difference schemes for numerical solving of PDEs [5, 6] and reduction of multiloop Feynman integrals to the minimal set of basis integrals [7].

In this note we describe an algorithm (Section 4) for constructing Gröbner bases for linear difference systems that is an adaptation of our polynomial algorithm [8] to linear difference ideals. We construct a Gröbner basis in its Janet-like form (Section 3), since this approach has shown its computational efficiency in the polynomial case [8, 9]. We briefly outline these efficiency issues in Section 5. The difference form of the algorithm exploits some basic notions and concepts of difference algebra (Section 2) as well as the definition of Janet-like Gröbner bases and Janet-like reductions together with the algorithmic characterization of Janet-like bases (Section 3). In Section 6 we present the new Maple package LDA for computing Gröbner bases of linear difference ideals. The package is a modified version of our earlier package [13] oriented towards commutative and linear differential algebra and based on the involutive basis algorithm [9]. The modified version is specialized to linear difference ideals and uses both Janet and Janet-like divisions [8] adapted to linear difference polynomials [10]. In Sections 7 and 8 we show how LDA can be applied for the generation of difference schemes for PDEs and the reduction of Feynman integrals.

2 Elements of difference algebra

Let $\{y^1, \ldots, y^m\}$ be the set of *indeterminates*, for example, functions of n variables $\{x_1, \ldots, x_n\}$, and $\theta_1, \ldots, \theta_n$ be the set of mutually commuting *difference operators (differences)*, e.g.,

$$\theta_i \circ y^j = y^j(x_1, \ldots, x_i + 1, \ldots, x_n).$$

A *difference ring* R *with differences* $\theta_1, \ldots, \theta_n$ is a com-

mutative ring R such that for all $f, g \in R,\ 1 \le i, j \le n$

$$\theta_i\theta_j = \theta_j\theta_i,\ \theta_i \circ (f+g) = \theta_i \circ f + \theta_i \circ g,$$
$$\theta_i \circ (f\,g) = (\theta_i \circ f)(\theta_i \circ g)\,.$$

Similarly, one defines a *difference field*.

Let $\mathbb{K}$ be a difference field, and let $\mathbb{R} := \mathbb{K}\{y^1, \ldots, y^m\}$ be the difference ring of polynomials over $\mathbb{K}$ in variables

$$\{\ \theta^\mu \circ y^k\ \mid \mu \in \mathbb{Z}^n_{\ge 0},\ k = 1, \ldots, m\ \}\,.$$

Hereafter, we denote by $\mathbb{R}_L$ the set of linear polynomials in $\mathbb{R}$ and use the notations:

$$\Theta = \{\ \theta^\mu\ \mid\ \mu \in \mathbb{Z}^n_{\ge 0}\ \},\ \deg_i(\theta^\mu \circ y^k) = \mu_i,$$
$$\deg(\theta^\mu \circ y^k) = |\mu| = \textstyle\sum_{i=1}^n \mu_i\,.$$

A *difference ideal* is an ideal $I \subseteq \mathbb{R}$ closed under the action of any operator from Θ. If $F := \{f_1, \ldots, f_k\} \subset \mathbb{R}$ is a finite set, then the smallest difference ideal containing F will be denoted by $\mathrm{Id}(F)$. If for an ideal I there is $F \subset \mathbb{R}_L$ such that $I = \mathrm{Id}(F)$, then I is a *linear difference ideal*.

A total ordering $\succ$ on the set of $\theta^\mu \circ y^j$ is a *ranking* if for all i, j, k, μ, ν the following holds:

$$\theta_i\theta^\mu \circ y^j \succ \theta^\mu \circ y^j\,,$$
$$\theta^\mu \circ y^j \succ \theta^\nu \circ y^k \iff \theta_i\theta^\mu \circ y^j \succ \theta_i\theta^\nu \circ y^k\,.$$

If $\mu \succ \nu$ implies $\theta^\mu \circ y^j \succ \theta^\nu \circ y^k$, the ranking is *orderly*. If $j \succ k$ implies $\theta^\mu \circ y^j \succ \theta^\nu \circ y^k$, the ranking is *elimination*.

Given a ranking $\succ$, a linear polynomial $f \in \mathbb{R}_L \setminus \{0\}$ has the *leading term* $a\,\vartheta \circ y^j$, $\vartheta \in \Theta$, where $\vartheta \circ y^j$ is maximal w.r.t. $\succ$ among all $\theta^\mu \circ y^k$ which appear with nonzero coefficient in f. $\mathrm{lc}(f) := a \in \mathbb{K} \setminus \{0\}$ is the *leading coefficient* and $\mathrm{lm}(f) := \vartheta \circ y^j$ is the *leading monomial*.

A ranking acts in $\mathbb{R}_L$ as a *monomial order*. If $F \subseteq \mathbb{R}_L \setminus \{0\}$, $\mathrm{lm}(F)$ will denote the set of the leading monomials and $\mathrm{lm}_j(F)$ will denote its subset for indeterminate y^j. Thus,

$$\mathrm{lm}(F) = \cup_{j=1}^m \mathrm{lm}_j(F)\,.$$

3 Janet-like Gröbner bases

Given a nonzero linear difference ideal $I = \mathrm{Id}(G)$ and a ranking $\succ$, the ideal generating set $G = \{g_1, \ldots, g_s\} \subset \mathbb{R}_L$ is a *Gröbner basis* [2, 4] of I if for all $f \in I \cap \mathbb{R}_L \setminus \{0\}$:

$$\exists\, g \in G,\ \theta \in \Theta\ :\ \mathrm{lm}(f) = \theta \circ \mathrm{lm}(g)\,. \qquad (1)$$

It follows that $f \in I \setminus \{0\}$ *is reducible modulo* G:

$$f \underset{g}{\to} f' := f - \mathrm{lc}(f)\,\theta \circ (g/\,\mathrm{lc}(g)),\quad f' \in I\,.$$

If $f' \ne 0$, then it is again reducible modulo G, and, by repeating the reduction, in finitely many steps we obtain

$$f \underset{G}{\to} 0\,.$$

Similarly, a nonzero polynomial $h \in \mathbb{R}_L$, whose terms are reducible (if any) modulo a set $F \subset \mathbb{R}_L$, can be reduced to an irreducible polynomial $\bar{h}$, which is said to be in the *normal form modulo* F (denotation: $\bar{h} = NF(h, F)$).

In our algorithmic construction of Gröbner bases we shall use a restricted set of reductions called *Janet-like* (cf. [8]) and defined as follows.

For a finite set $F \subseteq \mathbb{R}_L$ and a ranking $\succ$, we partition every $\mathrm{lm}_k(F)$ into groups labeled by $d_0, \ldots, d_i \in \mathbb{Z}_{\ge 0}$, $(0 \le i \le n)$. Here $[0]_k := \mathrm{lm}_k(F)$ and for $i > 0$ the group $[d_0, \ldots, d_i]_k$ is defined as

$$\{u \in \mathrm{lm}_k(F) \mid d_0 = 0, d_j = \deg_j(u), 1 \le j \le i\}.$$

Denote by $h_i(u, \mathrm{lm}_k(F))$ the nonnegative integer

$$\max\{\deg_i(v) \mid u, v \in [d_0, ..., d_{i-1}]_k\} - \deg_i(u).$$

If $h_i(u, \mathrm{lm}_k(F)) > 0$, then $\theta_i^{s_i}$ such that

$$s_i := \min\{\deg_i(v) - \deg_i(u) \mid$$
$$u, v \in [d_0, ..., d_{i-1}]_k,\ \deg_i(v) > \deg_i(u)\}$$

is called a *difference power* for $f \in F$ with $\mathrm{lm}(f) = u$.

Let $DP(f, F)$ be the set of difference powers for $f \in F$, and $\mathcal{J}(f, F) := \Theta \setminus \bar{\Theta}$ be the subset of Θ with

$$\bar{\Theta} := \{\theta^\mu \mid \exists\, \theta^\nu \in DP(f, F)\ :\ \mu - \nu \in \mathbb{Z}^n_{\ge 0}\}.$$

A Gröbner basis G of $I = \mathrm{Id}(G)$ is called *Janet-like* [8] if for all $f \in I \cap \mathbb{R}_L \setminus \{0\}$:

$$\exists\, g \in G, \vartheta \in \mathcal{J}(g, G)\ :\ \mathrm{lm}(f) = \vartheta \circ \mathrm{lm}(g)\,. \qquad (2)$$

This implies $\mathcal{J}-$reductions and the $\mathcal{J}-$normal form $NF_{\mathcal{J}}(f, F)$. It is clear that condition (2) implies (1). Note, however, that the converse is generally not true. Therefore, not every Gröbner basis is Janet-like.

The properties of a Janet-like basis are very similar to those of a Janet basis [9], but the former is generally more compact than the latter. More preciously, let GB be a reduced Gröbner basis [2], JB be a minimal Janet basis, and JLB be a minimal Janet-like basis of the same ideal for the same ranking. Then their cardinalities satisfy

$$\mathrm{Card}(GB) \le \mathrm{Card}(JLB) \le \mathrm{Card}(JB), \qquad (3)$$

where Card abbreviates *cardinality*, that is, the number of elements.

Whereas the algorithmic characterization of a Gröbner basis is zero redundancy of all its $S-$polynomials [1, 2], the algorithmic characterization of a Janet-like basis G is the following condition (cf. [8]):

$$\forall g \in G,\ \vartheta \in DP(g, G) :\ NF_{\mathcal{J}}(\vartheta \circ g, G) = 0\,. \quad (4)$$

This condition is at the root of the algorithmic construction of Janet-like bases as described in the next section.

4 Algorithm

Algorithm: Janet-like Gröbner Basis($F, \succ$)

Input: $F \subseteq \mathbb{R}_L \setminus \{0\}$, a finite set; $\succ$, a ranking
Output: G, a Janet-like basis of $\mathrm{Id}(F)$

1: **choose** $f \in F$ with the lowest $\mathrm{lm}(f)$ w.r.t. $\succ$
2: $G := \{f\}$
3: $Q := F \setminus G$
4: **do**
5: $\quad h := 0$
6: $\quad$ **while** $Q \neq \emptyset$ and $h = 0$ **do**
7: $\quad\quad$ **choose** $p \in Q$ with the lowest $\mathrm{lm}(p)$ w.r.t. $\succ$
8: $\quad\quad Q := Q \setminus \{p\}$
9: $\quad\quad h := \mathbf{Normal\ Form}(p, G, \succ)$
10: $\quad$ **od**
11: $\quad$ **if** $h \neq 0$ **then**
12: $\quad\quad$ **for all** $g \in G$ such that $\mathrm{lm}(g) = \theta^{\mu} \circ \mathrm{lm}(h),\ |\mu| > 0$ **do**
13: $\quad\quad\quad Q := Q \cup \{g\};\ G := G \setminus \{g\}$
14: $\quad\quad$ **od**
15: $\quad\quad G := G \cup \{h\}$
16: $\quad\quad Q := Q \cup \{\, \theta^{\beta} \circ g \mid g \in G,\ \theta^{\beta} \in DP(g, G) \,\}$
17: $\quad$ **fi**
18: **od while** $Q \neq \emptyset$
19: **return** G

This algorithm is an adaptation of the polynomial version [8] to linear difference ideals. It outputs a minimal Janet-like Gröbner basis which (if monic, that is, normalized by division of each polynomial by its leading coefficient) is uniquely defined by the input set F and ranking $\succ$. Correctness and termination of the algorithm follow from the proof given in [8]; in so doing the displacement of some elements of the intermediate sets G into Q at step 13 provides minimality of the output basis. The algorithm terminates when the set Q becomes empty in accordance with (4).

[1] See Web page `http://invo.jinr.ru`.

The subalgorithm **Normal Form**$(p, G, \succ)$ performs the Janet-like reductions (Section 3) of the input difference polynomial p modulo the set G and outputs the Janet-like normal form of p. As long as the intermediate difference polynomial h has a term Janet-like reducible modulo G, the elementary reduction of this term is done at step 4. As usually in the Gröbner bases techniques [2], the reduction is terminated in finitely many steps due to the properties of the ranking (Section 2).

Algorithm: Normal Form$(p, G, \succ)$

Input: $p \in \mathbb{R}_L \setminus \{0\}$, a polynomial; $G \subset \mathbb{R}_L \setminus \{0\}$, a finite set; $\succ$, a ranking
Output: $h = NF_{\mathcal{J}}(p, G)$, the $\mathcal{J}-$normal form of p modulo G

1: $h := p$
2: **while** $h \neq 0$ **and** h has a monomial u with coefficient $b \in \mathbb{K}$ $\mathcal{J}-$reducible modulo G **do**
3: $\quad$ **take** $g \in G$ such that $u = \theta^{\gamma} \circ \mathrm{lm}(g)$ with $\theta^{\gamma} \in \mathcal{J}(\mathrm{lm}(g), \mathrm{lm}(G))$
4: $\quad h := h/b - \theta^{\gamma} \circ (g/\,\mathrm{lt}(g))$
5: **od**
6: **return** h

An improved version of the above algorithm can easily be derived from the one for the involutive algorithm [9] if one replaces the input involutive division by a Janet-like monomial division [8] and then translates the algorithm into linear difference algebra. In particular, the improved version includes Buchberger's criteria adjusted to Janet-like division and avoids the repeated prolongations $\theta^{\beta} \circ g$ at step 16 of the algorithm.

5 Computational aspects

The polynomial version of algorithm **Janet-like Gröbner Basis** implemented in its improved form in C++ [8] has disclosed its high computational efficiency for the standard set of benchmarks[1]. If one compares this algorithm with the involutive one [9] specialized in Janet division, then all the computational merits of the latter algorithm are retained, namely:

- Automatic avoidance of some useless reductions.
- Weakened role of the criteria: even without applying any criteria, the algorithm is reasonably fast. By contrast, Buchberger's algorithm without applying the criteria becomes unpractical even for rather small problems.
- Smooth growth of intermediate coefficients.

- Fast search of a polynomial reductor which provides an elementary Janet-like reduction of the given term. It should be noted that as well as in the involutive algorithm such a reductor, if it exists, is unique. The fast search is based on the special data structures called Janet trees [9].

- Natural and effective parallelism.

Though one needs intensive benchmarking for linear difference systems, we have solid grounds to believe that the above listed computational merits hold also for the difference case.

As this takes place, computation of a Janet-like basis is more efficient than computation of a Janet basis by the involutive algorithm [9]. The inequality (3) for monic bases is a consequence of the inclusion [8]:

$$GB \subseteq JLB \subseteq JB\,. \tag{5}$$

There are many systems for which the cardinality of a Janet-like basis is much closer to that of the reduced Gröbner basis than the cardinality of a Janet basis. Certain binomial ideals called toric form an important class of such problems. Toric ideals arise in a number of problems of algebraic geometry and closely related to integer programming. For this class of ideals the cardinality of Janet bases is typically much larger than that of reduced Gröbner bases [8]. For illustrative purposes consider a difference analogue of the simple toric ideal [8, 12] generated in the ring of difference operators by the following set:

$$\{\ \theta_x^7 - \theta_y^2\theta_z, \theta_x^4\theta_w - \theta_y^3, \theta_x^3\theta_y - \theta_z\theta_w\ \}\,.$$

The reduced Gröbner basis for the degree-reverse-lexicographic ranking with $\theta_x \succ \theta_y \succ \theta_z \succ \theta_w$ is given by

$$\{\ \theta_x^7 - \theta_y^2\theta_z, \theta_x^4\theta_w - \theta_y^3, \theta_x^3\theta_y - \theta_z\theta_w, \theta_y^4 - \theta_x\theta_z\theta_w^2\ \}\,.$$

The Janet-like basis computed by the above algorithm contains one more element $\theta_x^4\theta_w - \theta_y^3$, whereas the Janet basis adds another six extra elements to the Janet-like basis [8].

The presence of extra elements in a Janet basis in comparison with a Janet-like basis is obtained because of certain additional algebraic operations. That is why the computation of a Janet-like basis is more efficient than the computation of a Janet basis. Both bases, however, contain the reduced Gröbner basis as the internally fixed [9] subset of the output basis[2]. Hence, having any of the bases computed, the reduced Gröbner basis is easily extracted without any extra computational costs.

[2] In the improved versions of the algorithms.

6 The Maple Package LDA

The package LDA (abbreviation for **L**inear **D**ifference **A**lgebra) implements the involutive basis algorithm [9] for linear systems of difference equations using Janet division. In addition, the package implements a modification of the algorithm oriented towards Janet-like division [8] and, thus, computes Janet-like Gröbner bases of linear difference ideals.

Table 1 collects the most important commands of LDA. Its main procedure `JanetBasis` converts a given set of difference polynomials into its Janet basis or Janet-like Gröbner basis form. More precisely, let $\mathbb{R}$ be the difference ring (see Section 2) of polynomials in the variables $\theta^\mu \circ y^k$, $\mu \in \mathbb{Z}^n_{\geq 0}$, $k = 1, \ldots, m$, with coefficients in a difference field $\mathbb{K}$ containing $\mathbb{Q}$ for which the field operations can be carried out constructively in Maple. We denote again by $\mathbb{R}_L$ the set of linear polynomials in $\mathbb{R}$. Given a generating set $F \subset \mathbb{R}_L$ for a linear difference ideal I in $\mathbb{R}$, `JanetBasis` computes the minimal Janet(-like Gröbner) basis J of I w.r.t. a certain monomial order (ranking). The input for `JanetBasis` consists of the left hand sides of a linear system of difference equations in the dependent variables $y^1, \ldots, y^m$, e.g. functions of $x_1, \ldots, x_n$. The difference ring $\mathbb{R}$ is specified by the lists of independent variables $x_1, \ldots, x_n$ and dependent variables given to `JanetBasis`. The output is a list containing the Janet(-like Gröbner) basis J and the lists of independent and dependent variables.

After J is computed, the involutive/$\mathcal{J}-$normal form of any element of $\mathbb{R}_L$ modulo J can be computed using `InvReduce`. Given $p \in \mathbb{R}_L$ representing a residue class $\overline{p}$ of the difference residue class ring $\mathbb{R}/I$, `InvReduce` returns the unique representative $q \in \mathbb{R}_L$ of $\overline{p}$ which is not involutively/$\mathcal{J}-$reducible modulo J. A $\mathbb{K}$-basis of the vector space $\mathbb{R}_L/(I \cap \mathbb{R}_L)$ is returned by `ResidueClassBasis` as a list if it is finite or is enumerated by a formal power series [16] in case it is infinite. For examples of how to apply these two commands, cf. Section 8.

Given an affine (i.e. inhomogeneous) linear system of difference equations, a call of `CompCond` after the application of `JanetBasis` returns a generating set of compatibility conditions for the affine part of the system, i.e. necessary conditions for the right hand sides of the inhomogeneous system for solvability.

Moreover, combinatorial devices to compute the Hilbert series and polynomial and function etc. [13] are included in LDA.

For the application of LDA to the reduction of Feynman integrals, a couple of special commands were implemented to impose further relations on the master integrals: By means of `AddRelation` an infinite sequence of master integrals parametrized by indetermina-

tes which are not contained in the list of independent variables is set to zero. Subsequent calls of `InvReduce` and `ResidueClassBasis` take these additional relations into account (cf. Section 8).

LDA provides several tools for dealing with difference operators. Difference operators represented by polynomials can be applied to (lists of) expressions containing $y^1, \ldots, y^m$ as functions of the independent variables. Conversely, the difference operators can be extracted from systems of difference equations. Leading terms of difference equations can be selected.

We consider difference rings containing shift operators which act in one direction only. If a linear system of difference equations is given containing functions shifted in both directions, then the system needs to be shifted by the maximal negative shift in order to obtain a difference system with shifts in one direction only. However, LDA allows to change the shift direction globally.

Unnecessary computations of involutive reductions to zero are avoided using the four involutive criteria described in [14], [15]. Fine-tuning is possible by selecting the criteria individually.

The implemented monomial orders/rankings are the (block) degree-reverse-lexicographical and the lexicographical one. In the case of more than one dependent variable, priority of comparison can be either given to the difference operators ("term over position") or to the dependent variables ("position over term"/elimination ranking).

The ranking is controlled via options given to each command separately. The other options described above can be set for the entire LDA session using the command `LDAOptions` which also allows to select Janet or Janet-like division.

7 Generation of finite difference schemes for PDEs

We consider the Laplace equation $u_{xx} + u_{yy} = 0$ and rewrite it as the conservation law

$$\oint_\Gamma -u_y dx + u_x dy = 0.$$

Adding the integral relations

$$\int_{x_j}^{x_{j+2}} u_x dx = u(x_{j+2}, y) - u(x_j, y)$$

and

$$\int_{y_k}^{y_{k+2}} u_y dx = u(x, y_{k+2}) - u(x, y_k)$$

and using the midpoint integration method we obtain the following discrete system:

$$\begin{cases} -(\theta_x - \theta_x\theta_y^2)\circ u_y + (\theta_x^2\theta_y - \theta_y)\circ u_x = 0, \\ 2\triangle h\,\theta_x \circ u_x - (\theta_x^2 - 1)\circ u = 0, \\ 2\triangle h\,\theta_y \circ u_y - (\theta_y^2 - 1)\circ u = 0, \end{cases} \quad (6)$$

where θ_x and θ_y are the right-shift operators w.r.t. x and y, e.g., $\theta_x \circ u_y(x, y) = u_y(x + 1, y)$.

We show how to use LDA to find a finite difference scheme for the Laplace equation:

```
> with(LDA):
```

We enter the independent and the dependent variables for the problem ($ux > uy > u$):

```
> ivar:=[x,y]: dvar:=[ux,uy,u]:
```

Next, we translate (6) into the input format of `JanetBasis`. Note that one can in general use `AppShiftOp` to apply a difference operator given as a polynomial similar to the ones in (6) to a difference polynomial.

```
> L:=[2*h*ux(x+1,y)-u(x+2,y)+u(x,y),
> 2*h*uy(x,y+1)-u(x,y+2)+u(x,y),
> 2*h*(ux(x+2,y+1)-ux(x,y+1))+
> 2*h*(uy(x+1,y+2)-uy(x+1,y))]:
```

Then we compute the minimal Janet basis of the linear difference ideal generated by L w.r.t. a ranking which compares the dependent variables prior to the corresponding difference monomials ("position over term" order; this ranking is chosen when using the option 2 as below). The least element of this Janet basis is by construction a difference polynomial which does not contain any monomial in ux and uy because $ux > uy > u$.

```
> JanetBasis(L,ivar,dvar,2)[1][1];
```

$$\mathrm{u}(x+4,\, y+2) - 4\,\mathrm{u}(x+2,\, y+2) + \mathrm{u}(x,\, y+2) + \mathrm{u}(x+2,\, y+4) + \mathrm{u}(x+2,\, y)$$

The computation takes less than one second of time on a Pentium III (1 GHz).

Dividing this difference polynomial by $4h^2$ we obtain the following finite difference scheme:

$$D_j^2(u_{jk}) + D_k^2(u_{jk}) = 0,$$

where

$$D_j^2(u_{jk}) = \frac{u_{j+2\,k} - 2u_{j\,k} + u_{j-2\,k}}{4h^2}$$

and

$$D_k^2(u_{jk}) = \frac{u_{j\,k+2} - 2u_{j\,k} + u_{j\,k-2}}{4h^2}$$

are discrete approximations of the second order partial derivatives occurring in Laplace's equation.

Tabelle 1: Main commands of LDA

`JanetBasis`	Compute Janet(-like Gröbner) basis
`InvReduce`	Involutive / $\mathcal{J}-$reduction modulo Janet(-like Gröbner) basis
`CompCond`	Return compatibility conditions for inhomogeneous system
`HilbertSeries` etc.	Combinatorial devices
`Pol2Shift`, `Shift2Pol`	Conversion between shift operators and equations
Some interpretations of commands for the reduction of Feynman integrals:	
`ResidueClassBasis`	Enumeration of the master integrals
`AddRelation`	Definition of additional relations for master integrals
`ResidueClassRelations`	Return the relations defined for master integrals

8 Reduction of Feynman integrals

In order to demonstrate how to use LDA for the reduction of Feynman integrals, we consider a simple one-loop propagator type scalar integral with one massive and another massless particle:

$$f(k,n) := I_{k,n} = \frac{1}{i\pi^{d/2}} \int \frac{d^d s}{P^k_{s-q,m} P^n_{s,0}}.$$

(Here k, n are the exponents of the propagators.)

The basis integrals for this example and the corresponding reduction formulae were found and studied by several authors (see, e.g., [17, 18]). Here we apply the Gröbner basis method, as implemented in LDA, directly to the recurrence relations which have the form:

$$\begin{cases} [d-2k-n-2m^2 k\mathbf{1}^+ - \\ n\mathbf{2}^+(\mathbf{1}^- - q^2 + m^2)]\, f(k+1,n+1) = 0, \\ [n-k-k\mathbf{1}^+(q^2+m^2-\mathbf{2}^-) - \\ n\mathbf{2}^+(\mathbf{1}^- - q^2 + m^2)]\, f(k+1,n+1) = 0, \end{cases} \tag{7}$$

where

$$\mathbf{1}^\pm f(k,n) = f(k\pm 1,n),\ \mathbf{2}^\pm f(k,n) = f(k,n\pm 1).$$

In addition, it is known that

$$f(k+i,n+j) = 0 \quad \forall\, i \le 0 \quad \forall\, j \tag{8}$$

which we will take into account later.

```
> ivar:=[k,n]: dvar:=[f]:
```

We enter the recurrence relations (7):

```
> L:=[(d-2*k-n)*f(k+1,n+1)-2*m^2*k*
> f(k+2,n+1)-n*f(k,n+2)-n*(m^2-q^2)*
> f(k+1,n+2),  (n-k)*f(k+1,n+1)-
> k*(q^2+m^2)*f(k+2,n+1)+k*f(k+2,n)-
> n*f(k,n+2)-n*(m^2-q^2)*f(k+1,n+2)]:
> JanetBasis(L,ivar,dvar):
```

Again, the computation time is less than one second. Now, the master integrals are given by:

```
> ResidueClassBasis(ivar,dvar);
```

$$[\mathrm{f}(k,\, n+2),\, \mathrm{f}(k+1,\, n+1),\, \mathrm{f}(k+2,\, n)]$$

(8) implies additional relations on the master integrals. (Here, j is recognized as not being contained in `ivar` and thus serves as a parameter to define the additional relations.)

```
> AddRelation(f(k,n+j)=0,ivar,dvar):
```

The list of master integrals now becomes:

```
> ResidueClassBasis(ivar,dvar);
```

$$[\mathrm{f}(k+1,\, n+1),\, \mathrm{f}(k+2,\, n)]$$

Next, we recompute the Janet basis for $m = 0$:

```
> m:=0: J:=JanetBasis(L,ivar,dvar):
```

For the special case where $m = 0$, we impose the relation $f(k+i,n) = 0$ for all i:

```
> AddRelation(f(k+i,n)=0,ivar,dvar):
```

Now, we are left with one master integral:

```
> ResidueClassBasis(ivar,dvar);
```

$$[\mathrm{f}(k+1,\, n+1)]$$

We reduce $f(k+2,n+3)$ modulo J taking also the additionally imposed relations on the master integrals into account. (Here, the option "F" lets `InvReduce` return the result in factorized form.)

```
> InvReduce(f(k+2,n+3),J,"F");
```

$$\begin{aligned} &-((2\,n+4-d+2\,k)\,(2\,n+2-d+2\,k) \\ &(2\,k+n-d)\,(n+3-d+k) \\ &(n+2-d+k)\,\mathrm{f}(k+1,\, n+1))/((n+1) \\ &(2\,n-d+4)\,n\,q^6\,k\,(d-2\,k-2)) \end{aligned}$$

Using `ResidueClassRelations` one can display the relations imposed on the master integrals:

```
> ResidueClassRelations(ivar,dvar,
> [i,j]);
```

$$[\mathrm{f}(k,\, n+j),\, \mathrm{f}(k+i,\, n)]$$

The difference operators occurring in the last result can be extracted as polynomials in δ_k, δ_n:

```
> Shift2Pol(%,ivar,dvar,
> [delta[k],delta[n]]);
```

$$[{\delta_n}^j,\, {\delta_k}^i]$$

9 Conclusion

The above presented algorithm **Janet-like Gröbner Basis** is implemented, in its improved form, in the Maple package LDA, and already applied to generation of difference schemes for PDEs and to reduction of some loop Feynman integrals.

These two kinds of applications were illustrated by rather simple examples. The first difference system (discrete Laplace equation and integral relations) contains two independent variables (x, y) and three dependent variables (u, u_x, u_y). The second system (recurrence relations for one-loop Feynman integral) also contains two independent variables/indices (k, n), but the only dependent variable f. The second system, however, is computationally slightly harder than the first one because of explicit dependence of the recurrence relations on the indices and three parameters (d, m^2, q^2) involved in the dependence on indices.

Dependence on index variables and parameters is an attribute of recurrence relations for Feynman integrals. Similar dependence may occur in the generation of difference schemes for PDEs with variable coefficients containing parameters. Theoretically established exponential and superexponential (depends on the ideal and ordering) complexity of constructing polynomial Gröbner bases implies that construction of difference Gröbner bases is at least exponentially hard in the number of independent variables (indices). Besides, in the presence of parameters the volume of computation grows very rapidly as the number of parameters increases.

The reduction of loop Feynman integrals for more than 3 internal lines with masses is computationally hard for the current version of the package. One reason for this is that the Maple implementation does not support Janet trees since Maple does not provide efficient data structures for trees.

Another reason is that in the improved version of the algorithm there is still some freedom in the selection strategy for elements in Q to be reduced modulo G. Though our algorithms are much less sensitive to the selection strategy than Buchberger's algorithm, the running time still depends substantially on the selection strategy: mainly because of dependence of the intermediate coefficients growth on the selection strategy. To find a heuristically good selection strategy one needs to do intensive benchmarking with difference systems. In turn, this requires an extensive data base of various benchmarks that, unlike polynomial benchmarks, up to now is missing for difference systems.

The comparison of implementations of polynomial involutive algorithms for Janet bases in Maple and in C++ [13] shows that the C++ code is of two or three order faster than its Maple counterpart. Together with efficient parallelization of the algorithm this gives a real hope for its practical applicability to problems of current interest in reduction of loop integrals.

Thus, for successful application of the Gröbner basis technique to multiloop Feynman integrals with masses and to multidimensional PDEs with multiparametric variable coefficients we are going not only to improve our Maple code but also to implement the algorithms for computing Janet and/or Janet-like difference bases in C++ as a special module of the open source software available on the Web page `http://invo.jinr.ru`.

10 Acknowledgements

The first author was supported in part by grants 04-01-00784 and 05-02-17645 from the Russian Foundation for Basic Research and by grant 2339.2003.2 from the Russian Ministry of Science and Education.

Literatur

[1] B. Buchberger. *An Algorithm for Finding a Basis for the Residue Class Ring of a Zero-Dimensional Polynomial Ideal*. PhD Thesis, University of Innsbruck, 1965 (in German).

[2] B. Buchberger, F. Winkler (Eds.) *Gröbner Bases and Applications*, Cambridge University Press, 1998.

[3] F. Chyzak. *Gröbner Bases, Symbolic Summation and Symbolic Integration*. In book [2], pp. 32–60.

[4] A. V. Mikhalev, A. B. Levin, E. V. Pankratiev, M. V. Kondratieva. *Differential and Difference Dimension Polynomials. Mathematics and Its Applications*, Kluwer, Dordrecht, 1999.

[5] V. V. Mozzhilkin, Y. A. Blinkov. *Methods of Constructing Difference Schemes in Gas Dynamics*. Transactions of Saratov University, 1(2), 2001, pp. 145–156 (in Russian).

[6] V. P. Gerdt, Y. A. Blinkov, V. V. Mozzhilkin. *Linear Difference Ideals and Generation of Difference Schemes for PDEs*. Symmetry, Integrability and Geometry: Methods and Applications (SIGMA), Institute of Mathematics, Kiev, 2005, to appear.

[7] V. P. Gerdt. *Gröbner Bases in Perturbative Calculations*. Nuclear Physics B (Proc. Suppl.) 135, 2004, pp. 232–237. arXiv:/hep-ph/0501053.

[8] V. P. Gerdt, Yu. A. Blinkov. *Janet-like Monomial Division. Janet-like Gröbner Bases*. Computer Algebra in Scientific Computing / CASC 2005, V. G. Ganzha, E. W. Mayr, E. V. Vorozhtsov (Eds.), Springer-Verlag, Berlin, 2005, to appear.

[9] V. P. Gerdt. *Involutive Algorithms for Computing Gröbner Bases*. Computational commutative and non-commutative algebraic geometry, IOS Press, Amsterdam, 2005, pp. 199–225. arXiv:math.AC/0501111, 2005.

[10] V. P. Gerdt. *On Computation of Gröbner Bases for Linear Difference Systems*, to appear in the Proceedings of the X International Workshop on Advanced Computing and Analysis Techniques in Physics Research (ACAT 2005), Nuclear Instruments and Methods in Physics Research (NIMA).

[11] V. P. Gerdt, D. Robertz. *A Maple Package for Computing Gröbner Bases for Linear Recurrence Relations*, to appear in the Proceedings of the X International Workshop on Advanced Computing and Analysis Techniques in Physics Research (ACAT 2005), Nuclear Instruments and Methods in Physics Research (NIMA).

[12] A. M. Bigatti, R. La Scala, L. Robbiano. *Computing Toric Ideals*. Journal of Symbolic Computation 27, 1999, pp. 351–365.

[13] Yu. A. Blinkov, V. P. Gerdt, C. F. Cid, W. Plesken, D. Robertz. *The Maple Package "Janet": I. Polynomial Systems*. Computer Algebra in Scientific Computing / CASC 2003, V. G. Ganzha, E. W. Mayr, and E. V. Vorozhtsov (Eds.), Institute of Informatics, Technical University of Munich, Garching, 2003, pp. 31–40.

[14] V. P. Gerdt, D. A. Yanovich. *Experimental Analysis of Involutive Criteria*. Algorithmic Algebra and Logic 2005, A. Dolzmann, A. Seidl, T. Sturm (Eds.), BOD Norderstedt, Germany, 2005, pp. 105–109.

[15] J. Apel, R. Hemmecke. *Detecting unnecessary reductions in an involutive basis computation*, RISC Linz Report Series 02-22, 2002.

[16] W. Plesken, D. Robertz. *Janet's approach to presentations and resolutions for polynomials and linear pdes*. Archiv der Mathematik, 84 (1), 2005, pp. 22–37.

[17] O. V. Tarasov. *Reduction of Feynman graph amplitudes to a minimal set of basic integrals*. Acta Physica Polonica B29, 1998, pp. 2655–2666. arXiv:/hep-ph/9812250

[18] V. A. Smirnov. *Evaluating Feynman Integrals*, STMP 211, Springer, Berlin, 2004.

Neuere Entwicklungen bei der Faktorisierung von Polynomen

Jürgen Klüners (Kassel)

klueners@mathematik.uni-kassel.de

Sei $f \in \mathbb{Z}[x]$ ein Polynom vom Grad n mit ganzen Koeffizienten. Eine der klassischen Fragestellungen in der Computeralgebra ist das Finden einer Faktorisierung von f. Hier wurde Ende der 60er Jahre von Hans Zassenhaus [Zas69] ein Algorithmus entwickelt, der bis vor wenigen Jahren in den meisten Computeralgebrasystemen implementiert war. Dieser Algorithmus ist in vielen Beispielen sehr effizient, leider ist aber seine worst–case–Laufzeit exponentiell. 1982 konnte dann in der berühmten LLL–Arbeit [LLL82] bewiesen werden, dass man Polynome in polynomieller Laufzeit im Grad und der (logarithmischen) Größe der Koeffizienten faktori-

sieren kann. Obwohl die in dieser Arbeit entwickelte Gitterreduktion in vielen Bereichen einen großen praktischen Fortschritt darstellte, ist dieser Faktorisierungsalgorithmus in der Praxis nicht effizient genug.

2002 entwickelte Mark van Hoeij [Hoe02] einen neuen Algorithmus, der in praktischen Beispielen deutlich schneller als der Zassenhaus–Algorithmus ist. Leider konnte in der Originalarbeit keine theoretische Laufzeitabschätzung für diesen Algorithmus gezeigt werden. In einer gemeinsamen Arbeit des Autors mit Karim Belabas, Mark van Hoeij und Allan Steel [BHKS05] wird dieser Algorithmus in der Darstellung vereinfacht und auch für den bivariaten Fall über endlichen Körpern eingeführt. Weiterhin können wir zeigen, dass er in beiden Fällen in polynomieller Laufzeit arbeitet, wobei unsere (immer noch pessimistische) worst–case–Abschätzung besser ist als die worst–case–Laufzeit des LLL–Faktorisierungsalgorithmus.

Wir werden in diesem Artikel parallel die Fälle $f \in \mathbb{Z}[x]$ und $f \in \mathbb{F}_p[t][x]$ studieren, wobei $\mathbb{F}_p$ der endliche Körper mit p Elementen ist. Den zweiten Fall studieren wir, weil er in der Darstellung deutlich einfacher ist und somit eine Vorstufe zum ersten Fall darstellt.

Bevor wir den neuen van Hoeij–Algorithmus erklären können, müssen wir erst den Zassenhaus–Algorithmus verstehen. Wir können annehmen, dass unser Polynom f quadratfrei ist, da mehrfache Faktoren den größten gemeinsamen Teiler von f und f' teilen, welchen wir effizient berechnen können. Ein wenig vorsichtig müssen wir im bivariaten Fall sein, da hier die Ableitung eines Polynoms 0 sein kann, etwa bei $f(t,x) = x^p - t$. Entweder sind nun alle Monome p–te Potenzen und wir können einfach die p–te Wurzel ziehen oder die Ableitung nach t ist nicht Null und wir vertauschen die Rolle von t und x.

Sei nun $f \in \mathbb{Z}[x]$ quadratfrei und normiert. Im Zassenhaus–Algorithmus wählen wir eine Primzahl p so, dass f modulo p keine doppelten Faktoren besitzt. Hierzu können wir jede Primzahl wählen, die nicht die Diskriminante von f teilt. Wir bezeichnen mit $\bar{f} \in \mathbb{F}_p[x]$ dasjenige Polynom, welches aus f dadurch entsteht, dass wir jeden Koeffizienten modulo p reduzieren. Mit bekannten Algorithmen [GG99, Kapitel 14] erhalten wir folgende Faktorisierung:

$$\bar{f}(x) \equiv \bar{f}_1(x) \cdots \bar{f}_r(x) \in \mathbb{F}_p[x].$$

Mittels des sogenannten Hensel–Liftings können wir sehr effizient (Lösen von linearen Gleichungssystemen) für jedes $k \in \mathbb{N}$ eine Faktorisierung der folgenden Form bestimmen (siehe z.B. [GG99, Kapitel 15]):

$$f(x) \equiv \tilde{f}_1(x) \cdots \tilde{f}_r(x) \bmod p^k,$$

wobei (bei geeigneter Einbettung) $\tilde{f}_i \equiv \bar{f}_i \bmod p$ gilt. Wir erklären nun an dem Beispiel $f(x) = x^4 - 11$ den Zassenhaus-Algorithmus. Für $p = 13$ erhalten wir, dass $\bar{f} \in \mathbb{F}_{13}[x]$ irreduzibel ist. Hieraus folgt dann sofort die Irreduziblität von $f \in \mathbb{Z}[x]$. Wenn wir (etwas unglücklich) $p = 5$ wählen, so erhalten wir modulo 5 nur Linearfaktoren, die wir mittels Hensel–Lifting wie folgt liften:

$$f(x) \equiv (x+41)(x-38)(x+38)(x-41) \bmod 125.$$

Wir können nun mit dem folgenden Satz abschätzen, dass die Koeffizienten eines Faktors betraglich kleiner als 33 sind (siehe [GG99, S.155ff]).

Satz 9 (Landau-Mignotte)**.** *Sei g ein Faktor von einem normierten $f \in \mathbb{Z}[x]$ mit*

$$f(x) = \sum_{i=0}^{n} a_i x^i \text{ und } g(x) = \sum_{i=0}^{m} b_i x^i.$$

Dann gilt: $|b_i| \leq \binom{m}{i} ||f||_2$, wobei $||f||_2 := \sqrt{\sum_{i=0}^{n} a_i^2}$ die 2–Norm ist.

Somit kann f keinen Linearfaktor besitzen, da alle Linearfaktoren modulo 125 im symmetrischen Restsystem $\{-62, \ldots, 62\}$ einen Koeffizienten größer als 33 besitzen. Als nächstes müssen wir probieren, ob das Produkt von zwei mod 125–Faktoren von f zu einem echten Faktor korrespondiert.

$$\begin{aligned}
(x+41)(x-38) &\equiv \quad x^2 + 3x - 58 \quad \bmod 125,\\
(x+41)(x+38) &\equiv \quad x^2 - 46x + 58 \quad \bmod 125,\\
(x+41)(x-41) &\equiv \qquad x^2 - 56 \qquad \bmod 125.
\end{aligned}$$

Wieder enthält jedes dieser quadratischen Polynome wenigstens einen Koeffizienten, der betraglich größer ist als unsere obere Schranke 33. Damit kann der „modulare Faktor“ $(x+41)$ kein Teiler von einem linearen oder quadratischen Polynom $g \in \mathbb{Z}[x]$ sein mit $g \mid f \in \mathbb{Z}[x]$. Wir haben also wieder gezeigt, dass das Polynom f irreduzibel ist. Wenn unser gegebenes Polynom reduzibel ist, so finden wir bei unserer Suche modulare Faktoren, deren Koeffizienten kleiner als die berechnete Schranke sind. Für diese Polynome testen wir mit einer Probedivision, ob es sich tatsächlich um einen Faktor handelt. Da die Probedivisionen teuer sind, sollte p^k deutlich größer als die berechnete Schranke gewählt werden, damit falsche Kandidaten eine hohe Wahrscheinlichkeit auf zu große Koeffizienten haben.

Die Wahl von $p = 5$ im obigen Beispiel war recht künstlich, aber es gibt irreduzible Polynome, bei denen immer sehr viele „modulare Faktoren“ auftreten. Wenn wir z.B. ein Polynom vom Grad 2^n mit elementarabelscher Galoisgruppe $(\mathbb{Z}/2\mathbb{Z})^n$ wählen, so ist dieses irreduzibel, besitzt aber für jede Primzahl mindestens 2^{n-1} modulare Faktoren.

Bei der Analyse des Zassenhaus–Algorithmus stellen wir fest, dass die meisten Teile des Algorithmus sehr effizient sind. Da wir eine kleine Primzahl p wählen

können, stellt die Faktorisierung von $\bar{f} \in \mathbb{F}_p[x]$ kein Problem dar. Auch das Hensel–Lifting kann sehr effizient implementiert werden. Der Flaschenhals entsteht dann, wenn die Anzahl r der modularen Faktoren groß ist, da wir dann im wesentlichen 2^r Tests durchführen müssen.

An dieser Stelle steigt der neue van Hoeij–Algorithmus ein. Er reduziert das kombinatorische Suchproblem auf ein sogenanntes Rucksack–Problem, welches mit Hilfe von Gittertheorie und dem sogenannten LLL–Gitterreduktions–Algorithmus sehr effizient (theoretisch in polynomieller Laufzeit) gelöst werden kann. Der neue Algorithmus setzt auch Gitterreduktion ein, verwendet aber andere Gitter.

Wir fixieren die folgende Notation:

$$f = g_1 \cdots g_s \in \mathbb{Z}[x] \text{ und } f = \tilde{f}_1 \cdots \tilde{f}_r \in \mathbb{Z}_p[x].$$

Die Faktorisierung über den p–adischen Zahlen $\mathbb{Z}_p$ können wir algorithmisch immer nur modulo p^k bestimmen, wobei wir $k \in \mathbb{N}$ geeignet wählen. Für das Verständnis des Folgenden können wir uns die p–adischen Zahlen $\mathbb{Z}_p$ als modulo p^k–Approximationen vorstellen. Nun schreiben wir:

$$g_v := \prod_{i=1}^{r} f_i^{v_i} \text{ für } v = (v_1, \ldots, v_r) \in \{0,1\}^r$$

und erhalten als neues
Problem: Für welche $v \in \{0,1\}^r$ gilt: $g_v \in \mathbb{Z}[x]$?
Um unser Problem zu linearisieren, betrachten wir im wesentlichen die logarithmische Ableitung, wobei $\mathbb{Q}_p(x) := \{\frac{a(x)}{b(x)} \mid a, b \in \mathbb{Q}_p[x]\}$:

$$\Phi : \mathbb{Q}_p(x)^*/\mathbb{Q}_p^* \to \mathbb{Q}_p(x), g \mapsto \frac{fg'}{g}.$$

Es ist sofort klar, dass Φ additiv ist, d.h. $\Phi(g_{v_1}) + \Phi(g_{v_2}) = \Phi(g_{v_1+v_2})$. Weiterhin gilt für beliebige $v \in \mathbb{Z}^r$, dass $\Phi(g_v)$ ein Polynom ist, also in $\mathbb{Z}_p[x]$ liegt.

Jetzt fehlt noch ein kleiner Schritt zur Übersetzung auf ein gittertheoretisches Problem. Wir definieren Vektoren $w_1, \ldots, w_s \in \{0,1\}^r$ derart, dass für die echten Faktoren $g_1, \ldots, g_s \in \mathbb{Z}[x]$

$$g_i = \prod_{1 \le j \le r} \tilde{f}_j^{w_{i_j}}$$

gilt. Diese erzeugen ein Gitter $W = \langle w_1, \ldots, w_s \rangle \subseteq \mathbb{Z}^r$, wobei letzteres Gitter von den Einheitsvektoren erzeugt wird, die zu den Faktoren $\tilde{f}_i$ korrespondieren. Wesentlich für das nun folgende Verfahren ist die Eigenschaft, dass ein $v \in \mathbb{Z}^r$ genau dann in W liegt, wenn $\Phi(g_v) \in \mathbb{Z}[x]$ gilt. Wenn wir nun ein beliebiges Erzeugendensystem von W kennen, so können wir die „kanonischen“ Basisvektoren $w_1, \ldots, w_s$ sehr einfach mit Methoden der linearen Algebra oder einfach kombinatorisch bestimmen. Kennen wir die w_i und ist p^k größer als das Doppelte der Landau–Mignotte–Schranke, so können wir die Faktoren g_i wie im Zassenhaus–Algorithmus rekonstruieren.

Die Idee des Algorithmus ist wie folgt. Wir starten mit dem Gitter $L = \mathbb{Z}^r$ und wissen $W \subseteq L$. Nun konstruieren wir ein Untergitter $L' \subset L$, welches aber immer noch W enthalten soll. Die Hoffnung ist, dass wir nach endlich vielen Iterationen bei $L' = W$ landen. Wir merken an, dass wir diese Situation sehr einfach testen können. Wenn unser Normalformalgorithmus eine Basis aus 0-1–Vektoren liefert und diese zu Faktoren aus $\mathbb{Z}[x]$ korrespondieren (Test durch Probedivision), dann gilt $L' = W$.

An dieser Stelle wechseln wir erst einmal zu der Situation $f \in \mathbb{F}_p[t][x]$, da hier der nun folgende Teil des Algorithmus deutlich einfacher ist. Die Landau–Mignotte–Schranke vereinfacht sich zu

$$g \mid f \in \mathbb{F}_p[t][x] \Rightarrow \deg_t(g) \le \deg_t(f),$$

wobei $\deg_t(f)$ der t–Grad des Polynoms f ist. Der Einfachheit halber gehen wir davon aus, dass $\bar{f}(x) := f(0,x) \in \mathbb{F}_p[x]$ quadratfrei ist. Dies ist eine echte Einschränkung, da es sein kann, dass $f(a,x)$ doppelte Faktoren für alle $a \in \mathbb{F}_p$ besitzt. Zur Lösung dieses Problems verweisen wir auf [BHKS05]. Aus der Faktorisierung $\bar{f} = \bar{f}_1 \cdots \bar{f}_r \in \mathbb{F}_p[x]$ erhalten wir mittels Hensel–Lifting eine Faktorisierung $f(t,x) = \tilde{f}_1 \cdots \tilde{f}_r$ im Potenzreihenring $\mathbb{F}_p[[t]][x]$, welche wir in der Praxis modulo t^k approximieren können. Die Funktion Φ ist nun wie folgt definiert:

$$\Phi : \mathbb{F}_p[[t]](x)^*/\mathbb{F}_p[[t]](x^p)^* \to \mathbb{F}_p[[t]](x), g \mapsto \frac{fg'}{g}.$$

Die Gitter L und W werden analog zur Situation in $\mathbb{Z}[x]$ definiert. Sei nun $v \in L \setminus W$. Dann gilt:

$$\mathrm{Pol}(v) := \Phi(g_v)(x) = \sum_{i=1}^{r} v_i \Phi(f_i) =$$

$$\sum_{i=0}^{n-1} b_i x^i \in \mathbb{F}_p[[t]][x] \setminus \mathbb{F}_p[t][x].$$

Weiterhin gilt für $g_v \in \mathbb{F}_p[t][x]$ die Abschätzung $\deg_t(b_i) \le \deg_t(f)$. Wir wählen nun ein $k > \deg_t(f)$ und bestimmen für $v \in L$ das zugehörige Polynom

$$g_v \equiv \sum_{i=0}^{n-1} b_i(t) x^i \bmod t^k,$$

wobei wir kanonisch die Polynome $b_i(t)$ modulo t^k reduzieren. Falls nun eines dieser Polynome b_i einen t–Grad hat, der größer als $\deg_t(f)$ ist, so wissen wir, dass

das zugehörige v nicht in W liegen kann. Im Folgenden vermeiden wir den kombinatorischen Ansatz. Hierzu bezeichnen wir mit $e_1, \ldots, e_r \in \mathbb{F}_p^r$ die Standardbasis von $\mathbb{F}_p^r$ und identifizieren die Elemente von $\mathbb{F}_p$ mit $\{0, \ldots, p-1\}$. Wir definieren $m := \deg_t(f)$ und

$$A_i := \begin{pmatrix} b_{i,m,1} & \cdots & b_{i,m,r} \\ b_{i,m+1,1} & \cdots & b_{i,m+1,r} \\ \vdots & \ddots & \vdots \\ b_{i,k-1,1} & \cdots & b_{i,k-1,r} \end{pmatrix} \in \mathbb{F}_p^{(k-m)\times r},$$

wobei die $b_{i,j,\ell}$ durch

$$\mathrm{Pol}(e_\ell) \equiv \sum_{i=0}^{n-1} \sum_{j=0}^{k-1} b_{i,j,\ell} t^j x^i \bmod t^k \quad (1 \le \ell \le r)$$

gegeben sind. Nun erfüllen alle $v \in W$ die Gleichung $A_i v^{\mathrm{tr}} = 0$. Wir können also durch sukzessive Kernberechnung immer kleinere Gitter L' bestimmen, die immer noch W enthalten. In einer leicht verbesserten Version dieses Algorithmus zeigen wir in [BHKS05], dass schließlich $L' = W$ gilt:

Satz 10. *Sei $k > (2n-1)\deg_t(f)$. Dann ist W der Kern von $A_1, \ldots, A_{n-1}$.*

Im Folgenden geben wir eine kurze Beweisskizze für diesen Satz an. Sei $v \in L \setminus W$ so gewählt, dass g_v keine p-te Potenz ist. Dann können wir v mit Hilfe der $w_1, \ldots, w_s$ so abändern, dass folgendes gilt:

1. $f_i \mid \mathrm{Pol}(v)$ für ein $1 \le i \le r$.
2. $g_j \nmid \mathrm{Pol}(v)$ für alle $1 \le j \le s$.

Damit gilt aber für dieses neue v und $H := \mathrm{Pol}(v) \bmod t^k$ (aufgefasst als Polynom in $\mathbb{F}_p[t][x]$):

$$\mathrm{Res}(f, \mathrm{Pol}(v)) = 0 \text{ und } \mathrm{Res}(f, H) \neq 0.$$

Dies bedeutet, dass $t^k \mid \mathrm{Res}(f, H)$ gilt, was bei genügend großem k einen Widerspruch zur Definition der Resultante durch die Sylvester–Matrix bedeutet.

Kommen wir nun zurück zu unserem Faktorisierungsproblem über $\mathbb{Z}$. Hier können wir nicht exakt denselben Algorithmus anwenden, da es beim Addieren Überträge gibt, z.B. $(3 + 1 \cdot 5^1) + (3 + 1 \cdot 5^1) = (1 + 3 \cdot 5^1) \neq 1 + 2 \cdot 5^1$ in $\mathbb{Z}_5$. Allerdings können wir zeigen, dass die Fehler durch Überträge klein sind. Anstatt des linearen Gleichungssystems verwenden wir nun ein geeignetes Gitter und suchen nach Vektoren kurzer Länge. Da das Finden von kürzesten Vektoren in Gittern i. A. NP–vollständig ist, ist es wesentlich, die Gitter geeignet zu wählen. Für diese Gitter verwenden wir dann die LLL–Gitterreduktion und können garantieren, dass die ersten Basisvektoren unseres Gitters gerade eine Darstellung von W liefern. Wir benutzen die analoge Notationen wie im bivariaten Fall und erhalten:

$$\mathrm{Pol}(e_\ell) \equiv \sum_{i=0}^{n-1} b_{i,\ell} x^i \bmod p^k \quad (1 \le \ell \le r).$$

Wir definieren nun ein Gitter Λ, welches durch die Spalten der folgenden Matrix definiert wird:

$$A := \begin{pmatrix} I_r & 0 \\ \tilde{A} & p^k I_n \end{pmatrix} \text{ mit } \tilde{A} := \begin{pmatrix} b_{0,1} & \cdots & b_{0,r} \\ \vdots & \ddots & \vdots \\ b_{n-1,1} & \cdots & b_{n-1,r} \end{pmatrix}.$$

Wenn wir einen Vektor aus Λ auf die ersten r Zeilen projizieren, erhalten wir einen Vektor aus L. Wenn wir die Präzision p^k groß genug wählen, dann können wir beweisen, dass alle Vektoren aus Λ, deren letzte n Einträge kleiner als die Landau–Mignotte–Schranke sind, zu einem Vektor aus W korrespondieren. Wir werden also obiges Gitter LLL–reduzieren und können dann beweisen, dass die ersten s Vektoren W erzeugen. Wir können sehr einfach eine obere Schranke B für die Norm der zu $w_1, \ldots, w_s$ gehörigen Gittervektoren in Λ berechnen. Für den LLL-Ansatz ist dann folgendes Lemma sehr nützlich, da es bei zu niedrig gewählter Präzision p^k immerhin erlaubt einen Fortschritt zu erzielen, d. h. ein Untergitter $L' \subset L$ zu berechnen, welches immer noch W enthält.

Lemma 11. *Sei Λ ein Gitter mit Basis $b_1, \ldots, b_m$ und Gram–Schmidt–Basis $b_1^*, \ldots, b_m^*$. Definiere $t := \min\{i \mid \forall i < j \le m : ||b_j^*||_2 > B\}$. Dann sind alle Vektoren b mit $||b||_2 \le B$ bereits in $\mathbb{Z}b_1 + \ldots + \mathbb{Z}b_t$ enthalten.*

Analog zum bivariaten Fall brauchen wir nun noch eine Präzisionsabschätzung, wann wir garantiert fertig sind. Wenn wir diese Präzision p^k verwenden, terminiert unser Algorithmus in einem Schritt und wir erhalten dann die polynomielle Laufzeit. Wir merken an, dass diese Abschätzung in der Praxis nicht benötigt wird, da wir dort solange die Präzision erhöhen, bis wir fertig sind.

Satz 12. *Sei $f \in \mathbb{Z}[X]$ vom Grad n. Dann terminiert der oben beschriebene Algorithmus, wenn*

$$p^k > c^n \cdot 4^{n^2} ||f||_2^{2n-1} \tag{13}$$

gilt, wobei c eine explizit berechenbare Konstante ist.

Wenn wir eine Laufzeitabschätzung für diesen Algorithmus bestimmen, so bekommen wir dieselbe Laufzeit, wie in der Original–LLL–Arbeit. Der wesentliche Unterschied ist, dass wir Beispiele angeben können (z.B. irreduzible Polynome), in denen diese Laufzeit im Original–LLL–Ansatz tatsächlich auftritt. In der Praxis zeigt sich für unseren Algorithmus, dass die abgeschätzte Laufzeit äußerst pessimistisch ist und wir kennen kein Beispiel, in denen diese annähernd erreicht wird. In der

Arbeit [BHKS05] geben wir noch eine Variante dieses Algorithmus an, in der wir die Gitterreduktion sukzessive nur mit kleinen Einträgen durchführen. Dies ergibt eine theoretische Laufzeitverbesserung, aber auch in der Praxis zeigen sich deutliche Vorteile.

Der neue Algorithmus ist mittlerweile in allen großen Computeralgebrasystemen implementiert. In der Praxis können hiermit Polynome in wenigen Sekunden oder Minuten faktorisiert werden, die mit den alten Algorithmen in Wochen oder Monaten nicht faktorisierbar waren.

Literatur

[BHKS05] K. Belabas, M. van Hoeij, J. Klüners und A. Steel, *Factoring polynomials over global fields*, eingereicht. `http://www.mathematik.uni-kassel.de/~klueners/pub.html`

[GG99] J. von zur Gathen und J. Gerhard, *Modern Computer Algebra*. Cambridge University Press, 1999.

[Hoe02] M. van Hoeij, *Factoring polynomials and the knapsack problem*, J. Number Theory, **95** (2002), 167–189.

[LLL82] A. K. Lenstra, H. W. Lenstra, Jr. und L. Lovász, *Factoring polynomials with rational coefficients*, Math. Ann., **261** (1982), no. 4, 515–534.

[Zas69] H. Zassenhaus, *On Hensel factorization I*, Journal of Number Theory, **1** (1969), 291–311.

Neues über Systeme

MuPAD Pro 3.2 für Linux

Andreas Sorgatz (Paderborn)

Neben Windows und MacOSX unterstützt MuPAD nun erstmals auch die Plattform Linux mit einer professionellen Benutzungsoberfläche. Eine Trial-Version von MuPAD Pro 3.2 für Linux ist im Web verfügbar (s. u.). Das Release ist ab Ende September 2005 im Handel.

MuPAD Pro 3.2 für Linux verfügt über ein neues XML-basiertes Arbeitsblattkonzept: deutlich komfortabler, flexibler und mit mehr Funktionalität als das bisherige Konzept unter Windows und MacOSX. In Zukunft wird diese Oberfläche einheitlich für alle drei Plattformen zur Verfügung stehen, wobei die alten Windows-Arbeitsblätter weiterhin importiert werden können.

MuPAD Pro 3.2 für Linux beinhaltet auch die bereits von der Windows- und MacOSX-Version her bekannte MuPAD-2D/3D-Grafik inklusive Animationen, transparenten 3D-Flächen und einem Grafik-Inspektor, in dem alle Eigenschaften einer Grafik interaktiv per Mausklick manipuliert werden können.

MuPAD Pro 3.2 für Linux enthält ebenso einige mathematische Verbesserungen gegenüber der Version 3.1.1. So sind hier unter Anderem eine erste Ausbaustufe des neuen Integrators und des erweiterten Solvers und Simplifiers verfügbar. Der Schwerpunkt des Release MuPAD Pro 3.2 für Linux liegt allerdings klar auf der Einführung der neuen Benutzungsoberfläche unter Linux, die nun die alte Version MuPAD Light ablöst.

Das Release MuPAD Pro 3.2 wird es nicht für Windows und MacOSX geben. Die drei Plattformen werden 2006 in einem neuen Release wieder zusammen geführt. Weitere Informationen sowie die Möglichkeit zum Download von MuPAD Pro 3.2 für Linux und Anforderung einer Evaluierungslizenz erhalten Sie im Web unter `http://studium.mupad.de/support/mupad-4-linux`.

Eine Abituraufgabe aus der Analysis

Reinhard Schmidt (Fachberater Mathematik, Sachsen)

Herr Reinhard Schmidt ist Fachlehrer für Mathematik am Christian-Weise-Gymnasium in Zittau und Fachberater für Mathematik im Regierungsschulamt Bautzen, Sachsen. An dieser Schule wurde der Unterricht in einzelnen Kursen vollständig mit Unterstützung des Computeralgebrasystems Mathcad durchgeführt. Dies hat naturgemäß neben der Planung und Durchführung von Unterricht auch Auswirkungen auf die Klausuraufgaben bis hin zum Abitur. Eine Abituraufgabe des letzten Prüfungsjahres wird hier abgedruckt. Hierbei ist erkennbar, dass neben klassischen Inhalten insbesondere komplexere Terme und umfangreichere Datenmengen bearbeitet werden müssen. Die Aufgabenstellungen sind aber noch deutlich an klassische Aufgaben angelehnt.

Heiko Knechtel

Am Christian-Weise-Gymnasium begann 1997 mit Genehmigung des Sächsischen Ministeriums für Kultus der wissenschaftlich begleitete Schulversuch „Computerunterstützter MathematikUnterricht (CuMaU)“. Ab der Klassenstufe 7 wurde in einigen Klassen neben dem GTR der Computer (Tabellenkalkulation EXCEL und DGS Geonext) kontinuierlich als Werkzeug und Medium genutzt. Vor Beginn der Klassenstufe 10 entscheiden sich die Schüler dann für ein Rechenhilfsmittel: GTR oder PC. Ab diesem Zeitpunkt wird auch verstärkt das Computeralgebrasystem Mathcad (Version 2001 Prof. als offene Schullizenz für alle Schüler und Lehrer) eingesetzt. Jedem CuMaU-Schüler steht nun permanent ein PC als Hilfsmittel im Unterricht, für Hausaufgaben, in Tests, Klassenarbeiten und Klausuren zur Verfügung. Die Bewertung erfolgt auf der Grundlage der ausgedruckten und/oder handschriftlichen Lösungen.

Natürlich werden grundlegende Lerninhalte nach wie vor ohne Hilfsmittel behandelt und in jeder Leistungsüberprüfung gibt es einen hilfsmittelfreien Teil, nur mit Stift und Papier.

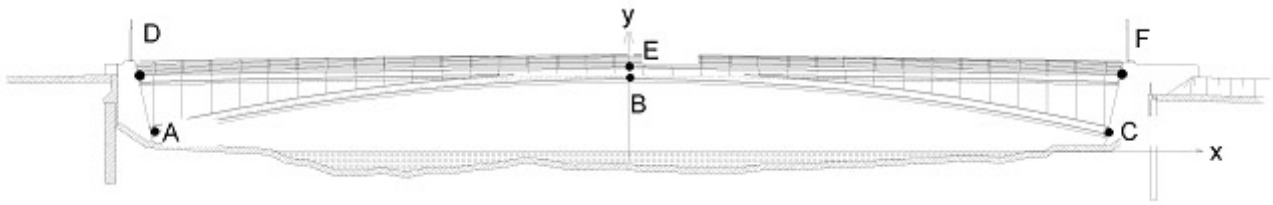
Insbesondere innerhalb der Lernbereiche „Gleichungen“ und „Funktionen“ werden verschiedenste Lösungsverfahren (numerische, graphische und algebraische) vorgestellt, diskutiert und angewendet. Neben Aufgaben mit klassischen innermathematischen Inhalten werden zunehmend offene und Anwendungsaufgaben gelöst.

Die ersten CuMaU-Schüler (je 2 Leistungs- und Grundkurse) unterzogen sich 2003 erstmalig einer eigenständigen Abiturprüfung (schriftlich und auch mündlich), bei der der PC als Hilfsmittel zugelassen war. Jedes Jahr folgen weitere Kurse.

Innerhalb der schriftlichen Abiturprüfung 2005 mussten unsere CuMaU-Schüler die unten abgedruckte Aufgabe im Leistungskurs lösen.

Abituraufgabe Analysis
(Gesamtpunktzahl der Aufgabe: 35; Gesamtpunktzahl der Prüfung: 90; Gesamtdauer der Prüfung: 300 Minuten) Hinweis: Der Lösungsweg mit Begründungen, Kommentaren und Nebenrechnungen muss deutlich erkennbar dargestellt werden.

Die Skizze (nicht maßstäblich) zeigt den axialsymmetrischen Querschnitt der im Oktober 2004 eröffneten „Görlitzer Altstadtbrücke“. Dargestellt sind u. A. das Brückenbogenteil, das Fahrbahnteil (22 dazwischen vertikal verlaufende Stützen) und das auf dem Fahrbahnteil befestigte Geländer. Das Brückenbogenteil ist Teil eines Kreisringes. Außerdem sind die Wasseroberfläche der Neiße und das Flussbett schematisch dargestellt.



Zur mathematischen Beschreibung sind in einem kartesischen Koordinatensystem die Koordinaten folgender Punkte gegeben:

$$B(0/6, 40), C(39, 90/1, 60), E(0/6, 90), F(41, 20/6, 30).$$

Eine Einheit entspricht einem Meter.

a) Die Mittellinie des Brückenbogenteils verläuft durch die Punkte A, B und C und kann durch einen Kreisbogen mathematisch beschrieben werden. Berechnen Sie die Gleichung des Kreises k in Koordinatenform, auf dem diese Mittellinie liegt, und stellen Sie den Kreisbogen in einem geeigneten Koordinatensystem grafisch dar.

Erreichbare Punktezahl: 6

b) Zur statischen Berechnung und Dimensionierung der Stützlager der Brücke sind u. A. die Gleichungen der Tangente und Normale von k im Punkt C erforderlich. Berechnen Sie die Gleichungen dieser Tangente und Normale.

Erreichbare Punktezahl: 4

c) Für jedes y_M und r mit $y_M, r \in \mathbb{R}, r > 0$ ist eine Funktion $k_{y_M,r}$ mit $k_{y_M,r}(x) = y_M + \sqrt{-x^2 + r^2}$ gegeben. Beschreiben Sie jeweils den Einfluss der Parameter y_M und r auf den Verlauf der Graphen der Funktion.

Ermitteln Sie die Gleichung der Funktion $k_{y_M,r}$, deren Graph durch die Punkte A und C verläuft und bei dem die Normale an den Graphen von $k_{y_M,r}$ im Punkt C einen Anstiegswinkel von $80°$ hat.

Erreichbare Punktezahl: 6

d) Die Form der Fahrbahndecke (im Querschnitt) kann durch Parabeln n-ten Grades beschrieben werden. Begründen Sie, welchen Grades eine solche Parabel mindestens sein muss, wenn die jeweils am linken und rechten Ende anschließenden Straßen horizontal verlaufen.

Ermitteln Sie rechnerisch die Gleichung dieser Parabel, so dass die Punkte D, E und F (siehe Skizze) auf dem Graphen dieser Parabel liegen.

Berechnen Sie die Koordinaten des Punktes zwischen den Punkten D und E, wo die Fahrbahn den größten Anstieg hat. Begründen und veranschaulichen Sie Ihre Aussagen.

Erreichbare Punktezahl: 7

e) Die Form der oberen Begrenzungslinie des Brückenbogenteils kann durch eine Funktion

$$b(x) = \frac{-16133}{100} + \frac{1}{100} \cdot \sqrt{283013329 - 10000x^2}$$

mit $x \in \mathbb{R}, -39,90 < x < 39,90$ beschrieben werden. Die Form der unteren Begrenzungslinie des Fahrbahnteils kann durch eine Funktion

$$f(x) = 2,0824 \cdot 10^{-7}x^4 - 7,0695 \cdot 10^{-4}x^2 + 6,40$$

mit $x \in \mathbb{R}$ beschrieben werden.

Berechnen Sie, wie viel Prozent des Graphen der Funktion $b(x)$ oberhalb des Graphen von $f(x)$ liegen.

Hinweis: Formel zur Bogenlängenberechnung:

$$L = \int_a^b \sqrt{1 + (f'(x))^2}dx$$

Aus statischen Gründen sind, wie im Querschnitt der Brücke dargestellt, zwischen dem Brückenbogenteil und dem Fahrbahnteil 22 vertikal verlaufende Stützen eingeschweißt. Die Strecke $\overline{AC}$ ist in 34 gleich breite Intervalle eingeteilt (siehe Skizze). Die Länge der Stützen kann als Zahlenfolge betrachtet werden. Beschreiben Sie eine Möglichkeit, die Glieder dieser Zahlenfolge zu berechnen. Begründen Sie rechnerisch, dass es sich hierbei nicht um eine arithmetische Zahlenfolge handeln kann. Berechnen Sie die Summe aller Stützenlängen (Gesamtstützenlänge).

Erreichbare Punktezahl: 7

f) Im Querschnitt des Neißeflussbettes wurden folgende Wassertiefen (in Metern, gerundet) gemessen:

Abstand zum linken Ufer	Wassertiefe
0	0
5	0,5
10	0,9
15	1,1
20	0,9
25	0,7
30	0,5
35	0,4
40	0,3
45	0,2
50	0,1
55	0

Ermitteln Sie die Durchflussmenge des Wassers pro Sekunde, wenn die durchschnittliche Fließgeschwindigkeit $0,9\,m/s$ beträgt.

Erreichbare Punktezahl: 5

A. Beutelspacher, H. B. Neumann, T. Schwarzpaul
Kryptographie in Theorie und Praxis

Vieweg Verlag, Wiesbaden, 2005, ISBN 3-528-03168-9, 319 Seiten, €24,90.

Das Buch stellt die wesentlichen Begriffe und Verfahren der Kryptographie leicht verständlich und mathematisch fundiert dar. Es erscheint in einer Zeit, in der die uns umgebenden Sicherungssysteme gut funktionieren und entsprechend dem technischen Fortschritt weiterentwickelt werden, in der aber an den Verfahren selbst nicht viel verändert wird. Die Autoren bezeichnen diesen Moment in der kryptographischen Forschung als Konsolidierungsphase, die eine gute Gelegenheit biete, einen umfassenden Überblick über alle relevanten Verfahren, ihre Anwendungen und derzeit aktuelle Sicherheitsstandards samt Schlüssellängen zu geben. Dies gelingt in diesem Buch in ausgezeichneter Weise.

In drei Teilen werden symmetrische Verschlüsselungen, asymmetrische Kryptographie und Anwendungen behandelt. Der erste Teil leitet anhand der historischen Entwicklung von der Cäsar- über die Vigenère-Chiffre zur Definition der perfekten Sicherheit. Danach folgen Strom-, Block- und Kaskadenchiffren, wie sie auch heute benutzt werden.

Die asymmetrische Kryptographie ist Inhalt des zweiten Teils. Neben dem Public-Key-Verfahren RSA und der ElGamal-Verschlüsselung werden weitere Verfahren und mathematische Konzepte vorgestellt, unter Anderem elliptische Kurven.

Wer sich über die Absicherung unserer modernen Kommunikationswege informieren will, wird im dritten Teil fündig. Hier werden für alle sicherheitsrelevanten Aufgaben Protokolle behandelt, die auf Verfahren der ersten beiden Teile beruhen. Dazu gehören unter Anderem Authentifikation, die Analyse der Anforderungen an Public-Key-Infrastrukturen und Multiparty-Computations zum Rechnen mit verteilten Geheimnissen. Das letzte Kapitel geht auf die Quantenkryptographie ein, die klassische Verschlüsselungsverfahren zur Absicherung eines Quantenkanals obsolet macht, da schon das schlichte Abhören vom Empfänger bemerkt wird.

Die für die Verfahren benötigte Mathematik wird bei Bedarf in Kürze bereitgestellt. Bei den Analysen der Angriffsmöglichkeiten betonen die Autoren, dass für Anwendungen nur die effiziente Sicherheit wichtig ist, die hauptsächlich von der Leistungsfähigkeit der heutigen Computer abhängt. Die Kapitel über Anwendungen bieten außer ausführlichen Sicherheitsanalysen auch Bemerkungen zu praxisrelevanten Details wie zum Beispiel dem Unterschied zwischen online- und offline-Münzsystemen beim elektronischen Bargeld. Der Stil des Buchs ist klar und übersichtlich.

Das Buch eignet sich gut für eine Vorlesung oder zur Lektüre, für die man sich auch einzelne Kapitel herausgreifen kann. Es bietet eine gute Referenz für die gängigen Verfahren der Kryptographie, und es kann Interessierten gut verständlich die Absicherung ihrer elektronischen Kommunikation erklären. Bis (Quanten-) Computer realisiert sind, mit denen man die heutige Public-Key-Kryptographie effizient angreifen kann, wird das Buch noch viele Freunde finden.

Harm Pralle (Braunschweig)

M. Bronstein
Symbolic Integration I, 2nd edition

Springer-Verlag, Berlin-Heidelberg-New York, 2005, ISBN 3-540-21493-3, €53,45.

Kurz vor seinem Tod (siehe Nachruf auf Seite 37) ist die zweite Auflage dieser Monographie über symbolische Integration erschienen. Die erste Auflage erschien 1997 und wurde damals vom Unterzeichner im Computeralgebra-Rundbrief mit der Nummer 20 vom Frühjahr 1997 ausführlich besprochen (`http://www.fachgruppe-computeralgebra.de/CAR/CAR20/node14.html`).

Das Buch ist das Standardwerk zur Frage, wie algorithmisch entschieden werden kann, ob die Stammfunktion einer elementaren Funktion wieder elementar ist oder nicht. Im ersten Fall wird diese dann auch berechnet. Die römische Eins bezieht sich auf die in diesem Buch ausschließlich behandelte Klasse der transzendenten Funktionen.

Die neun Kapitel der ersten Auflage konnten nahezu unverändert bleiben, lediglich kleinere Fehlerkorrekturen – man hat Mühe solche überhaupt zu finden – wurden eingearbeitet.

Komplett neu hingegen ist ein Kapitel 10 zur „Parallel Integration“. Hier hat der Autor Ideen von Risch, Norman, Davenport, Trager und anderen, die teilweise bis ins Jahr 1976 zurückgehen, im Stil des Buches aufgearbeitet und ergänzt. Ausgangspunkt für diese Theorie ist der zentrale Satz von Liouville, der aussagt, dass sich der Integrand im Falle der elementaren Integrierbarkeit in der Form $f = Dv + \sum_{i_1}^{m} c_i \frac{Du_i}{u_i}$ darstellen lässt. Dabei sind die u_i multivariate Polynome in den transzendenten Körpererweiterungen des Konstantenkörpers, v entsprechend eine rationale Funktion und D die Ableitung (Derivation) (siehe (10.1) auf Seite 297). Nun werden für die u_i, für den Nenner von v und den Grad des Zählers von v sinnvolle Ansätze gemacht. Das geschieht parallel! Die Berechnung der algebraischen Größen reduziert sich dann auf das Lösen von linearen Gleichungssystemen. Ist man erfolgreich, hat man sehr schnell und effizient eine Stammfunktion gefunden, falls nicht, dann weiß man natürlich nichts, da ja die Ansätze falsch sein können. Das macht das Verfahren zu einer Heuristik, die aber in einigen Computeralgebrasystemen sehr erfolgreich zum Einsatz kommt.

Bronstein selbst hat mit Datum vom 10.05.2005 eine Maple-Implementierung mit weniger als 100 Zeilen Code im Netz bereit gestellt. `http://www-sop.inria.fr/cafe/Manuel.Bronstein/pmint`. Dabei steht `pmint` für „Pure Man's Integrator“.

Bronstein schreibt dazu

> *„It is based on recent improvements to a powerful heuristic called parallel integration. While it is not meant to be as complete as the large commercial integrators based on the Risch algorithm, its very small size makes it easy to port to any computer algebra system or library capable of factoring multivariate polynomials. Because it is applicable to special functions (such as Airy, Bessel, Whittaker, Lambert), it is able to compute integrals not handled by the existing integrators.* `pmint` *is not meant as a replacement for existing integrators, but either as an extension, or as a cheap and powerful alternative for any computer algebra project.“*

Meine Besprechung der ersten Auflage 1997 endete mit dem folgenden Wunsch: *„Dem rundum gelungenen Buch ist eine weite Verbreitung zu wünschen, dem Autor die Zeit um bald einen zweiten Teil zur Theorie der symbolischen Integration algebraischer Funktionen folgen zu lassen!“* Der erste Teil des Wunsches ist mit dieser Neuauflage in Erfüllung gegangen und gilt weiter, der zweite Teil kann nicht mehr in Erfüllung gehen.

Johannes Grabmeier (Deggendorf)

F. Colombo, I. Sabadini, F. Sommen, D.C. Struppa
Analysis of Dirac Systems and Computational Algebra

Birkhäuser Verlag, Basel, Boston, 2004, ISBN 0-8176-4255-2, 336 Seiten, €98,44.

Das Gebiet der Algebraischen Analysis ist in Deutschland leider immer noch kaum bekannt, obwohl es viele schöne Resultate hervor gebracht hat. Man kombiniert hier algebraische Techniken, vor allem für den Polynomring oder die Weyl-Algebra, mit der Analysis linearer partieller Differentialgleichungen. Ein herausragendes Ergebnis dieser Kombination ist sicherlich das Fundamentalprinzip von Ehrenpreis-Palamodov, das notwendige und hinreichende Bedingungen für die Existenz von Lösungen inhomogener überbestimmter linearer Systeme mit konstanten Koeffizienten liefert. Die Reduktion analytischer Fragen auf rein algebraische Probleme erlaubt außerdem häufig eine effektive algorithmische Behandlung, z. B. mit Hilfe von Gröbnerbasen.

Dirac-Operatoren sind lineare partielle Differentialoperatoren erster Ordnung mit konstanten Koeffizienten, die Werte in einer Clifford-Algebra annehmen. Sie spielen eine große Rolle in der mathematischen Physik; ein erstes Beispiel wurde von Dirac abgeleitet auf der

Suche nach einer relativistischen Form der Schrödinger-Gleichung. Formal kann man Dirac-Operatoren als das Ergebnis der Faktorisierung einer Wellengleichung ansehen.

Das Ziel des vorliegenden Buchs ist es, effektive Verfahren zur Behandlung von Dirac-Operatoren zu entwickeln. Es gliedert sich in sechs Kapitel. Zunächst werden die benötigten Konzepte aus der Algebra (Kommutative Algebra mit Gröbnerbasen sowie Garbentheorie) bzw. der Analysis (topologische Vektorräume und Fourier-Analysis) sowie aus der Theorie der Hyperfunktionen eingeführt. Das zweite Kapitel beschäftigt sich mit rechnerischer Algebraischer Analysis. Neben einer allgemeinen Einführung in die Algebraische Analysis liegt dabei das Schwergewicht auf dem bereits erwähnten Fundamentalprinzip. Ein Abschnitt behandelt dabei auch das praktische Arbeiten mit Computeralgebrasystemen (insbesondere CoCoA).

Die nächsten drei Kapitel behandeln konkrete Klassen linearer Differentialgleichungssysteme. Dies beginnt mit Variationen des Cauchy-Fueter-Systems, einer Verallgemeinerung der Cauchy-Riemannschen Gleichungen aus der Funktionentheorie für Funktionen einer quaternionischen Variablen. Die Autoren stellen hier vor allem ihre eigenen Ergebnisse für den Fall mehrerer quaternionischer Variablen vor. Das vierte Kapitel studiert dann Dirac-Operatoren über allgemeinen Clifford-Algebren. Themen sind hier unter Anderem die Fischer-Zerlegung eines Clifford-Polynoms und der Kompatibilitätskomplex eines Dirac-Operators. Im fünften Kapitel werden schließlich eine Reihe von bekannten Feldgleichungen aus der Physik wie die Maxwell- oder Proca-Gleichungen als Regularitätsbedingungen im Sinne der Algebraischen Analysis interpretiert. Dies erlaubt einige Aussagen über die Existenz von Lösungen oder die Hebbarkeit von Singularitäten. Das letzte Kapitel stellt eine Reihe offener Probleme und möglicher Forschungsrichtungen vor.

Das Buch stellt eine willkommene Ergänzung der (nicht sonderlich üppigen) Literatur zur Algebraischen Analysis dar. Insbesondere eine ausführlichere Behandlung von rechnerischen Fragen findet man ansonsten nur in dem Werk von Saito, Sturmfels und Takayama über hypergeometrische Gleichungen. Allerdings liegt in dem vorliegenden Buch der Schwerpunkt insgesamt deutlich stärker auf den analytischen Aspekten. Es werden auch keine neuen algebraischen Methoden entwickelt, sondern bekannte Algorithmen aus der Kommutativen Algebra angewendet, um Erkenntnisse über eine in der Physik wichtige Klasse von Differentialgleichungen zu gewinnen.

Werner M. Seiler (Heidelberg)

M. Drmota, P. Flajolet, D. Gardy, B. Gittenberger (Eds.) Mathematics and Computer Science III: Algorithms, Trees, Combinatorics and Probabilities

Birkhäuser Verlag, Basel, Boston, 2004, ISBN 3-7643-7128-5, 554 Seiten, $ 109,00.

Es gibt neben der Computeralgebra noch viele Bereiche mit fruchtbaren Wechselwirkungen zwischen Mathematik und Informatik. Der Untertitel des vorliegenden Bandes spezifiziert klar, worum es hier in der Sache geht: um kombinatorische, analytische und stochastische Konzepte und Methoden, wie sie im Umfeld der quantitativen Algorithmenanalyse entstanden sind. Der Verweis auf das grundlegende Werk von Donald E. Knuth und die Arbeiten von Robert Sedegwick und Philippe Flajolet, deren lang erwartetes Buch *Analytic Combinatorics* demnächst erscheinen wird, mag aus der Sicht der Algorithmik andeuten, worum es hier geht. Hinzu kommt hier ein ganz starkes Gewicht auf stochastischen Methoden, was sich etwa mit Stichworten wie: Grenzwertsätze, Verzweigungsprozesse, schnellmischende Markov-Ketten, Random Walks etc. umschreiben lässt.

Es handelt sich um den Tagungsband der dritten Tagung *Mathematics and Computer Science*, die im September 2004 in Wien stattgefunden hat. Anschließend an die beiden Vorgängertagungen in Versailles (2000 und 2002) hat sich damit ein Forum auf höchstem Niveau für diesen Forschungssektor etabliert. Der Band enthält 54 Beiträge unterschiedlicher Länge, in der Regel *full papers*, aber auch einige, die eher als *extended abstracts* anzusprechen sind. Sie sind in den sieben Untergruppen *Combinatorics and Random Structures*, *Graph Theory*, *Analysis of Algorithms*, *Trees*, *Probability*, *Combinatorial Stochastic Processes* und *Applications* zusammengefasst, wobei die Zuordnung kei-

neswegs immer eindeutig ist. Keinen der Beiträge, auch nicht die zu den eingeladenen Vorträgen, von denen ich den profunden Artikel *Perfect Matchings in Random Graphs with Prescribed Minimal Degree* von Alan Frieze und Boris Pittel besonders hervorheben möchte, kann man als Überblicksartikel bezeichnen.

Obwohl algorithmische Problemstellungen in vielen Fällen Motivation und Ausgangspunkt für die vorgestellten analytischen Untersuchungen waren, kommen in den Artikeln konkrete Algorithmen eher selten vor, von Implementierungen und deren Beschreibungen ganz zu schweigen. Auch die Nähe der analytischen Kombinatorik zur Computeralgebra, wie sie sich beispielsweise in den Arbeiten von Philippe Flajolet und seiner Gruppe bei der INRIA zur Generierung von Strukturen und ihrer algorithmischen Asymptotik zeigt, wird hier kaum deutlich. Insofern mag der erste Blick auf das Inhaltsverzeichnis für einen Interessenten aus der Computeralgebra eher enttäuschend sein. Wer aber darüber hinaus ein generelles Interesse an der quantitativen und stochastischen Analyse von diskreten Strukturen, Algorithmen und Prozessen hat, wird hier höchst interessante und technisch (bisweilen sogar sehr) anspruchsvolle Beispiele aktueller Forschung finden. Die Qualität der Beiträge und der editorischen Sorgfalt sind einwandfrei. Ein einziger kleiner Einwand: der Index ist etwas bescheiden ausgefallen angesichts der vielen Querbeziehungen zwischen Artikeln in unterschiedlichen Gruppen.

Volker Strehl (Erlangen)

K. Feng, H. Niederreiter, C. Xing (Eds.)
Coding, Cryptography and Combinatorics

Birkhäuser Verlag, Basel, Boston, Berlin, ISBN 3-7643-2429-5, 2004, 405 Seiten, € 115,56.

Bei diesem Buch handelt es sich um den Tagungsband zur gleichnamigen Tagung von 2003. Anders als der Titel vermuten lässt, sind die drei Themen nicht gleichberechtigt. Etwa die Hälfte der Beiträge kommen jeweils aus der Codierungstheorie bzw. der Kryptographie. Die Kombinatorik ist praktisch nur als Hilfsmittel dieser beiden Disziplinen vertreten.

Die fünf „Invited Papers“ haben einen Umfang von jeweils etwa 25 Seiten und sind tendenziell mehr überblicksartig gestaltet. Die 28 „Contributed Papers“ haben einen Umfang von jeweils etwa 15 Seiten und beschäftigten sich mit aktuellen Ergebnissen aus ihren Gebieten. Die durchweg hohe Qualität der Beiträge wurde durch ein Gutachterverfahren sicher gestellt, bei dem nur 28 von 39 eingereichten Arbeiten angenommen wurden.

Wegen der großen Spannbreite der vorgestellten Themen ist ein vollständiger Überblick im Rahmen dieser Besprechung nicht möglich. Im Folgenden möchte ich am Beispiel von drei Artikeln die Spannbreite des vorliegenden Werks dokumentieren.

Von P. Lu und L. Huang stammt ein „Invited Paper“ über „A New Correlation Attack on LFSR Sequences with High Error Tolerance“. In der Kryptographie der Stromchiffren spielen die linearen Schieberegister eine zentrale Rolle. Wenn man eine Pseudozufallsfolge $z = (z_1, \ldots, z_n)$ untersucht, dann interessiert man sich unter Anderem für lineare Schieberegisterfolgen $u = (u_1, \ldots, u_n)$ mit $p(u_i = z_i) = \frac{1}{2} + \delta$ für ein $\delta > 0$. Es sind also z und u zueinander positiv korreliert. Sind außerdem das Rückkopplungspolynom und der Grad l von $u = (u_1, \ldots, u_n)$ bekannt, so kann man u bestimmen, indem man die Startwerte $u_1, \ldots, u_l$ ermittelt (Korrelationsattacke). Ein naiver Ansatz besteht darin, alle 2^l möglichen Werte von $u_1, \ldots, u_l$ durch zu probieren. Dies ist jedoch nur für sehr kleine l möglich. Es gibt aber mehrere Möglichkeiten, den Aufwand für die Suche nach den Startwerten zu verringern (schnelle Korrelationsattacken). Die hier vorliegende Arbeit stellt zwei neue Algorithmen vor, mit denen man Probleme bis $l = 100$ effektiv bearbeiten kann.

Eine Arbeit über „Good Self-Dual Quasi-Cyclic Codes over $\mathbb{F}_q$, q Odd“ stammt von S. Ling und P. Solé. Mit nur vier Seiten ist sie der kürzeste Beitrag der Sammlung. Ein Code C ist schlicht ein Unterraum von $\mathbb{F}_q^n$. Gilt bezüglich des Standardskalarprodukts $C^\perp = C$, so heißt C selbstdual. Selbstduale Codes sind bereits intensiv untersucht worden, da sie sich durch ihre Struktur besonders gut handhaben lassen. Ein Code der Länge $2n$ ist quasizyklisch, falls mit (u, v) auch (v, u) in C liegt ($u, v \in \mathbb{F}_q^n$). Die Autoren benutzen eine relative $(u + v|u - v)-$Konstruktion von G. Huges (2000), um nachzuweisen, dass es asymptotisch gute selbstduale quasizyklische Codes gibt. Der Stil ist recht knapp gehalten und für viele zentrale Resultate wird auf voran-

gegangene Arbeiten verwiesen. Hier hätte eine größere Ausführlichkeit das Lesen deutlich erleichtern können.

Einer der wenigen „reinen“ Kombinatorik-Artikel in dieser Sammlung ist „Combinatorial Tableaux in Isoperimetry“ von C. C. Pinter. Die Arbeit beschäftigt sich mit der Geometrie des binären Hammingraums $\{0,1\}^n$ und dort insbesondere mit den „level subsets“. Ein „level subset“ ist eine Verallgemeinerung des Begriffs der Hamming-Kugeln. Für jedes $m \in \{1,\ldots,2^n\}$ besteht das „level subset“ mit m Elementen aus den m kleinsten Elementen des Hammingraums. Dabei werden die Elemente zunächst nach Gewicht und Elemente gleichen Gewichts lexikographisch geordnet. Der Autor untersucht Fragen wie die Anzahl der inneren Punkte etc. Dazu entwickelt er ein kombinatorisches Diagramm, das sich leicht rekursiv berechnen lässt und an dem sich die Antworten auf diese und andere Fragen leicht ablesen lassen.

Fazit: Es handelt sich um eine Sammlung von interessanten Artikeln. Jeder der sich für Codierungstheorie oder Kryptographie interessiert, wird in diesem Buch etwas für sich finden.

Andreas Klein (Kassel)

D. F. Holt, B. Eick und E. O'Brien
Handbook of Computational Group Theory

Chapman & Hall / CRC Press, 2005, ISBN 1-58488-372-3, — Seiten, € 82,90.

Dies ist das erste Textbuch, das sämtliche relevanten und aktuellen Themen der algorithmischen Gruppentheorie (im Folgenden kurz CGT für „Computational Group Theory“ genannt) behandelt. Vorgänger beschränkten sich jeweils auf Teilbereiche, wie etwa endliche Permutationsgruppen oder endlich präsentierte Gruppen.

Das Buch beginnt mit einem kurzen, lesenswerten Abriss der Geschichte der CGT. Danach werden, zum Teil ohne Beweis, die mathematischen Hintergründe aus der Algebra, insbesondere der Gruppentheorie vorgestellt. Es folgt ein allgemeines Kapitel über die verschiedenen Datenstrukturen und Modelle zur Repräsentation einer Gruppe auf einem Computer.

Ein sehr langes und ausführliches Kapitel ist dem Rechnen in Permutationsgruppen gewidmet. Insgesamt vier Kapitel befassen sich mit verschiedenen Aspekten endlich präsentierter Gruppen. Das erste davon behandelt das grundlegende Todd-Coxeter-Verfahren zur Abzählung von Nebenklassen. Das zweite thematisiert das Problem, eine endliche Präsentation einer Gruppe, die etwa als Permutations- oder Matrixgruppe gegeben ist, zu konstruieren. In einem weiteren Kapitel wird die algorithmische Theorie polyzyklischer Gruppen umfassend dargestellt. Im letzten Kapitel aus diesem Umkreis geht es um die Berechnung von Quotienten spezieller Struktur endlich präsentierter Gruppen. Die fundamentalen Algorithmen der Darstellungstheorie, wie die Meataxe oder der Dixon-Schneider-Algorithmus, sowie Methoden zum Berechnen von Kohomologiegruppen werden ebenfalls in einem eigenen Kapitel sehr gründlich vorgestellt. Bisher hatte die algorithmische Darstellungstheorie noch nicht Eingang in ein Textbuch gefunden.

Ein weiteres Kapitel behandelt speziellere Methoden, zum Beispiel zum Aufzählen von Untergruppen einer gegebenen Gruppe. Bemerkenswert ist der Platz, der den vorhandenen Datenbanken endlicher Gruppen, Charaktertafeln und Darstellungen eingeräumt wird. Je ein Kapitel über das Knuth-Bendix Verfahren und automatische Gruppen runden das Bild ab. Gerade diese beiden letztgenannten Kapitel bieten, neben den Methoden für polyzyklische Gruppen, eine Perspektive auf die substanzielle Ausweitung der CGT auf gewisse Klassen unendlicher Gruppen.

Die vorgestellten Algorithmen und Methoden werden durch instruktive, explizit vorgeführte Beispiele verdeutlicht. Die Algorithmen werden mithilfe von Pseudocode beschrieben. Insgesamt finden sich in dem Buch 89 hervorgehobene Prozeduren in Pseudocode. Fast jeder Abschnitt schließt mit einer Reihe von Übungsaufgaben. Sehr schön sind auch die Hinweise auf Anwendungen außerhalb der Gruppentheorie, z. B. in der Galoistheorie oder der Topologie, oder sogar außerhalb der Mathematik, z. B. in der Chemie.

Den weitaus größten Teil dieses Buches hat Derek Holt verfasst, die beiden anderen Autoren haben einzelne Sektionen und Kapitel beigesteuert. Die Autoren sind weltweit anerkannte Experten auf allen in diesem Band angesprochenen Gebieten. Das Buch ist sehr klar geschrieben und besticht durch einen besonders systematischen Aufbau. Es schließt mit einem zwölfseitigen Register.

Es ist ein großes Verdienst dieses Buches, die enge inhaltliche Verzahnung der drei Teilbereiche der CGT,

Permutationsgruppen, endlich präsentierte Gruppen und Matrixgruppen, offen gelegt zu haben. Randomisierte Methoden, die in der CGT eine zunehmend große Rolle spielen, der hier natürlich auch entsprechend Rechnung getragen wird, bilden die methodische Klammer für die genannten Teilbereiche.

Der vorliegende Band hat den Namen Handbuch wahrlich verdient. Er ist als Grundlage für Vorlesungen wie auch als Referenzwerk gleichermaßen geeignet. Besonders viel Nutzen werden aber auch Studierende, sowohl für die Begleitung von Vorlesungen als auch für eigene Forschungsarbeiten, aus diesem Buch ziehen können.

Gerhard Hiss (Aachen)

J. R. Shackell
Symbolic Asymptotics

Springer Verlag, Berlin, Heidelberg, New York, ISBN 3-540-21097-0, 2004, 243 Seiten, € 59,95.

This book is the first text book about symbolic asymptotics, written by one of its major contributors. It deals with the question to calculate limits and asymptotic expansions of real functions symbolically.

The book contains the following chapters:

Chapter 1: Introduction. In the introduction the author explains the concepts of towers of fields, exp-log-functions etc., and shows by examples which algorithmic problems occur.

Chapter 2: Zero Equivalence. In this chapter zero equivalence of constants and of functions is considered. Schanuel's Conjecture and Richardson's Uniformity Conjecture are touched. Algorithms and examples are given. Modular methods and Hensel lifting are discussed. Systems of partial differential equations and differential ideals are introduced.

Chapter 3: Hardy Fields. The concept of Hardy fields is introduced, their closure properties and their relation to algebraic differential equations is discussed. Further properties of Hardy fields are investigated.

Chapter 4: Output Data Structures. Here asymptotic power series as well as multiseries which enable the use of different scales are introduced. Exponentials, logarithms, powers etc. of multiseries are treated. Finally the algebra of star products is investigated. Star products are generalizations of nested expansions and multiseries.

Chapter 5: Algorithms for Function Towers. To compute limits and asymptotic expressions, one regards functions as elements in a tower of Hardy fields

$$\mathcal{F}_0 \subset \mathcal{F}_1 \subset \cdots \subset \mathcal{F}_N \ .$$

Problems that occur in this treatment are discussed, and an algorithm for the exp-log case is given. The exp-log case is generalized by adding exponentials, integrals, algebraic entensions etc. Finally Cartesian representations are studied.

Chapter 6: Algebraic Differential Equations. Nested forms of Hardy field solutions of algebraic differential equations are discussed, and an algorithm to detect such solutions is given. Theorems about the number of cases and ideas to reduce the complexity are given. Finally sparse differential equations are considered.

Chapter 7: Inverse Functions. Until recently it was unknown whether the inverse of an exp-log function is necessarily asymptotic to an exp-log function. That this is not the case was discovered by Shackell in 1993. However, inverses of Hardy field members that tend to infinity belong to a Hardy field, again. In this chapter the inversion of nested expansions is covered, and multiseries of inverse functions are discussed.

Chapter 8: Implicit Functions. In this chapter asymptotic series of implicit functions are considered.

Chapter 9: Star-Product Expansions. The rewriting of exp-log expressions into standard star expansion form is discussed. Growth classes in Hardy fields are introduced.

Chapter 10: Oscillating Functions. Oscillating functions are never members of Hardy fields. Hence this chapter deals with different methods for this purpose. The book finishes with an example.

Many algorithms and examples are discussed in the book. I would have been interested to read in detail which of these algorithms are implemented in an existing computer algebra system like Maple. It is not clearly stated which parts of the book are covered by Maple's `asympt` command. However, on pp. 91–92 the author informs about Dominik Gruntz's `limit` implementation in Maple, and announces a current update process of `asympt` by Bruno Salvy.

The book contains a wealth of information on sym-

bolic asymptotics. It fills a gap in the literature and should be found in every library. The user of a computer algebra system who is interested to understand the output of a command like Maple's `asympt` finds much interesting information. Actually many users might not even be aware about the complexity of such a command.

Wolfram Koepf (Kassel)

I. Shparlinski
Cryptographic Applications of Analytic Number Theory

Birkhäuser Verlag, Basel, Boston, Berlin, ISBN 3-7643-6654-0, 2004, 411 Seiten, € 104,86.

Anwendungen der Zahlentheorie in der Kryptographie wie RSA, Diffie-Hellman etc. sind allgemein bekannt. Das vorliegende Werk geht jedoch weit über diese Grundlagen hinaus und richtet sich an Leser, die bereits Vorbildung sowohl in Kryptographie als auch in der Zahlentheorie haben. Unterteilt ist das Buch in sieben Teile mit insgesamt 31 Kapiteln.

Der erste Teil (7 Kapitel) ist eine Einführung, in der alle später verwendeten Resultate und Bezeichnungen zusammen getragen werden.

Das erste Kapitel führt die im restlichen Buch verwendeten Notationen ein. Darunter sind auch eher exotische wie $\lfloor s \rfloor_m$ für den Rest von s bei Division durch m. Ich hätte es als hilfreich empfunden, wenn diese Notationen noch einmal in einem gesonderten Symbolverzeichnis zusammen gefasst worden wären. Die anderen Kapitel dieses Abschnitts wurden eher als eine „Erinnerung" an bereits bekannte Ergebnisse gestaltet. Ergebnisse wie der LLL-Algorithmus oder $\frac{m}{\phi(m)} = O(\log\log(m+1))$ werden als bekannt vorausgesetzt.

Der zweite Teil (4 Kapitel) ist der Untersuchung des Diskreten-Logarithmus-Problems gewidmet. Dabei geht es um die Frage, wie gut der diskrete Logarithmus mit einfachen Funktionen angenähert werden kann. Die vier Kapitel beschäftigen sich jeweils mit der Approximation $\bmod p$, $\bmod p-1$, durch boolesche Funktionen und reellwertige Polynome. Der Autor konzentriert sich hier ganz auf die zahlentheoretischen Probleme, die Bedeutung der Resultate für die kryptographische Praxis wird nicht diskutiert.

Der dritte Teil (3 Kapitel) untersucht das Diffie-Hellman-Problem. Im Aufbau und in den Methoden ähneln die Untersuchungen denen aus dem zweiten Teil.

Im vierten Teil (8 Kapitel) werden verschiedene Public-Key-Kryptosysteme genauer untersucht. Die Resultate dieses Abschnitts sind aus Sicht der Kryptographie viel anwendungsbezogener als die der vorangegangenen Abschnitte.

Pseudozufallsgeneratoren sind das Thema des fünften Teils (5 Kapitel). Die einzelnen Kapitel sind unterschiedlichen Generatoren gewidmet (unter Anderem Blum-Blum-Shub, Naor-Reingold, $1/M$). Hier kommen sowohl die kryptographischen Anwendungen als auch die zahlentheoretischen Methoden besonders gut zur Geltung.

Der sechste Teil enthält noch weitere 4 Kapitel, die vom Thema nicht in die anderen Teile gepasst haben. Im Einzelnen sind dies: zahlentheoretische Funktionen mit kryptographischen Anwendungen wie z. B. Testen der Quadratfreiheit (Kapitel 28), der Zusammenhang zwischen der arithmetischen Komplexität und der Komplexität eines zugehörigen booleschen Schaltkreises (Kapitel 29), Polynom-Approximation in endlichen Körpern (Kapitel 30), Untersuchung spezieller Funktionen wie z. B. Permutationspolynome ($f : \mathbb{F}_q \to \mathbb{F}_q$ ist bijektiv) (Kapitel 31).

Im siebten Teil stellt der Autor noch 55 offene Probleme vor. Die Auswahl erscheint mir sehr gelungen und ich denke, dass diese Liste eine gute Anregung für künftige Forschungen ist.

Das Literaturverzeichnis ist mit seinen 571 Einträgen sehr umfangreich und eignet sich gut für einen noch tieferen Einstieg in die Materie.

Fazit: Das Buch wird seinem Anspruch, sowohl für Kryptologen als auch Zahlentheoretiker interessant zu sein, voll und ganz gerecht. Für Einsteiger ist das Buch allerdings wegen der hohen Voraussetzungen an die Kenntnisse des Lesers weniger geeignet.

Andreas Klein (Kassel)

1. GAMM 2005 – 76. Jahrestagung der Gesellschaft für Angewandte Mathematik und Mechanik e.V.

Luxemburg, 28.03. – 01.04.2005

`http://www.univ.lu/GAMM2005/GAMM_1_Deutsch.html`

Sektion Computeralgebra und Computeranalysis

Die Jahrestagung der GAMM, einer unserer Muttergesellschaften, fand dieses Jahr in Luxemburg statt, wie üblich in der Woche nach Ostern (28.03. – 01.04.). Schon Tradition auf dieser Tagung ist, dass Computeralgebra und Computeranalysis mit einer gemeinsamen Sektion vertreten sind. Die Organisation lag in diesem Jahr in den Händen von Günter Mayer (Uni Rostock) für die Computeranalysis und H. Michael Möller (Uni Dortmund) für die Computeralgebra. In unserer Sektion trugen vor: V. Lewandowskyy (Kaiserslautern): Anwendungen nichtkommutativer Computeralgebra, E. Zerz (Kaiserslautern): Computeralgebraische Methoden zur Strukturanalyse von Kontrollsystemen, B. Tibken (Wuppertal): Observability of nonlinear systems, R. Steinwand (Karlsruhe): Polynomial systems of equations as building block of asymmetric cryptographic schemes, H. M. Möller (Dortmund): Ein neues Verfahren zum Nullstellenzählen im CAGD.

Während diese Vorträge im üblichen Rahmen stattfanden und mit dem für diese Tagung gewohnten Problem (geringes Interesse bei Fachfremden) zu kämpfen hatten, fanden die beiden folgenden Vorträge eine große Zuhörerschaft und ein reges Interesse, wie die anschließenden Diskussionen zeigten. Im ersten Vortrag von C. Gnörlich (Oberwolfach): „Das Oberwolfach-Referenzzentrum für mathematische Software“ wurde hauptsächlich Bezug genommen auf die bereits installierte Computeralgebra-Software. Im zweiten Vortrag von A. Frühbis-Krüger (Kaiserslautern): „Das Computeralgebra-System SINGULAR in praktischen Anwendungen“ fand das mit dem Richard-Jenks-Preis ausgezeichnete Computeralgebrasystem die verdiente Aufmerksamkeit.

H. M. Möller (Dortmund)

2. Algorithmic Algebra and Logic 2005 – Conference in Honor of the 60th Birthday of Volker Weispfenning

Passau, 03. – 08.04.2005

`http://www.a3l.org`

Diese Konferenz zu den Themen Algebraische Modelltheorie, Algorithmen zur Quantorenelimination und Gröbnerbasen, die allesamt zu den Arbeiten von Volker Weispfenning in Beziehung stehen, wurde von Thomas Sturm (General Chair), Andreas Dolzmann (Program Chair) und Andreas Seidl (Publicity Chair) organisiert.

In den eingeladenen Hauptvorträgen sprachen Eberhard Becker über *Geometric Radicals of Polynomial Ideals*, Thomas Becker über *Volker Weispfenning: Scientist, Teacher, Mentor* und Anthony Hearn über *REDUCE: The First Forty Years*.

Daneben wurde das breite Spektrum der Themen in 46 Vorträgen behandelt, deren Ausarbeitungen in einem Konferenzband vorliegen, der bereits zu Beginn der Tagung verfügbar war. Der Konferenzband enthält auch einen Abdruck der Habilitationsschrift von V. Weispfenning mit dem Thema *Model Theory of Lattice Products*.

Auch von der schönen Drei-Flüsse-Stadt Passau konnten sich die Teilnehmer bei einer Schiffsrundfahrt ein Bild machen, zu der der Jubilar eingeladen hatte. Weitere Informationen finden sich auf der Homepage der Konferenz.

Johannes Grabmeier (Deggendorf)

3. CAPP 2005 – DESY School on Computer Algebra and Particle Physics

Zeuthen, 03. – 08.04.2005

`http://www-zeuthen.desy.de`

Die erste DESY School on Computer Algebra and Particle Physics (CAPP 2005) fand Anfang April 2005 am DESY Zeuthen statt.

Ziel der Schule war es, etwa 25 Diplomanden und Doktoranden in der theoretischen Elementarteilchenphysik einen Einblick in aktuelle Programme und Anwendungen der Computeralgebra auf diesem Gebiet zu geben. Der Schwerpunkt lag dabei auf Programmen zur Generierung und der anschließenden automatischen Berechnung von Feynman-Diagrammen sowie Algorithmen und Techniken zur numerischen Auswertung von Schleifen-Integralen. Aber auch ein Einblick in Computeralgebrasysteme sowie paralleles Rechnen und 64-Bit-Prozessoren wurde gegeben. Tipps zu effizientem Programmieren sowie ein alternativer Ansatz zu Feynman-Diagrammen mit Hilfe von Helizitätsamplituden rundeten das Programm ab.

Dabei wurde der Stoff nicht nur theoretisch vorgetragen. Viele der Vorlesungen beinhalteten auch praktische Übungen, in denen das gerade erworbene Wissen in einfachen Beispielen direkt angewandt und vertieft werden konnte. Dazu stand für jeden Teilnehmer im Vortragsraum ein Computer zur Verfügung, der auch intensiv, oft über die Vorlesungszeit hinaus, zum Ausprobieren und Nachvollziehen genutzt wurde.

Am Ende waren sich alle Teilnehmer einig, dass dies eine sehr gelungene Schule war, bei der jeder viel neues Wissen mit nach Hause nehmen konnte.

Michael Rauch (München)

4. ALCOMA'05 – Algebraic Combinatorics and Applications, Designs and Codes

Thurnau, 03. – 10.04.2005

`http://www.mathe2.uni-bayreuth.de/ALCOMA05`

Vom 03.04. (Anreisetag) bis zum 10.04.2005 (Abreisetag) fand in Thurnau die Tagung ALCOMA'05 (Algebraic Combinatorics and Applications, Designs and Codes) statt, organisiert von Prof. Dr. A. Kerber und Dr. A. Kohnert am Lehrstuhl II für Mathematik. Sie war Nachfolgetagung von ALCOMA'99 in Gößweinstein, Anlass war auch der 60. Geburtstag von Prof. Dr. R. Laue. Im Mittelpunkt der Tagung standen gruppentheoretische Methoden bei kombinatorischen Designs und fehlerkorrigierenden Codes. Für beide Forschungsgebiete (die am Lehrstuhl II für Mathematik durch Prof. Laue und Prof. Kerber samt Mitarbeitern stark vertreten werden) waren weltweit führende Fachvertreter anwesend, u. A. die Herausgeber Colbourn und Dinitz des CRC Handbook of Combinatorial Designs, Frau Vera Pless, Mitherausgeberin des zweibändigen CRC Handbook of Coding Theory, sowie Eamonn O'Brien als Mitherausgeber des CRC Handbook of Computational Group Theory. Insgesamt waren es 73 Teilnehmerinnen und Teilnehmer aus Australien, Belgien, Bulgarien, Canada, Dänemark, Deutschland, England, Finnland, Holland, Iran, Israel, Italien, Kroatien, Neuseeland, Österreich, Russland, Slowenien, aus der Tschechischen Republik und aus den USA.

Tagungsfoto

Die 54 Vorträge (davon 17 eingeladene Hauptvorträge) wurden in dem kürzlich renovierten Kutschenhaus gehalten, das sich als ausgezeichnet geeignet erwiesen hat. Die Vorträge waren vor allem Berichte über neueste Entwicklungen in der Design-, Codierungs-, Gruppen- und Graphentheorie. Sie reichen von tiefgehenden theoretischen Überlegungen über ausgefeilte Algorithmen und deren Resultate bis hin zu konkreten Anwendungen in der Mathematik, der Informatik, der Telekommunikation, der Genomik und Systemtipps im Lotto. Ein Tagungsband wird demnächst in der Reihe der Bayreuther Mathematischen Schriften erscheinen. Er wird neben Vortragsausführungen u. A. auch eine Liste von während der Tagung vorgetragenen offenen Problemen und Herausforderungen enthalten. Die Tagung wurde finanziell unterstützt vom Bayerischen Staatsministerium, von der DFG, von der Firma Siemens und vom Universitätsverein Bayreuth.

Adalbert Kerber und Axel Kohnert (Bayreuth)

5. ACAT 2005 – 10th International Workshop on Advanced Computing And Analysis

Zeuthen, 23.05. – 27.05.2005

`http://www.desy.de/acat05`

Die ACAT 2005 war der zehnte Workshop in einer Konferenzreihe, die sich besonders mit den Computing-Aspekten in der Hochenergiephysik befasst. Wie immer gab es drei Sessions, die dieses Jahr unter den Überschriften Grid Computing und Data Acquisition, Analysetechniken und Simulation sowie Computeralgebra und Algorithmen zusammengefasst werden können.

Im Rahmen dieser Publikation ist natürlich vor allem die dritte Gruppe interessant. Eine (subjektive) Auswahl der wichtigsten Trends: Auf der algorithmischen Seite ist der Einsatz von Gröbnerbasen zur Lösung von Rekursionsrelationen, wie sie in Multiloop-Rechnungen auftreten, und die Applikation der seit relativ kurzer Zeit bekannten Twistor-Methoden für Loop-Rechnungen erwähnenswert. Bei den Programmen geht die Tendenz eindeutig zu mehr Schleifen und mehr externen Linien, was den Rechenaufwand enorm nach oben treibt. Hier sind noch technische Hürden zu überwinden, einige Schlagworte mögen dazu genügen: Parallelisierung von FORM, Probleme mit der numerischen Genauigkeit, Verbesserung der numerischen Integrationsalgorithmen, neue Methoden (z. B. für symbolische Summation, Dyson-Schwinger-Formalismus).

Die allesamt sehr informativen Hauptvorträge wurden dieses Jahr von Jos Vermaseren über „Perspectives of FORM", Bertrand Georgeot über „Quantum computing for physics research", Ulrich Ramacher über „Information Processing with Pulsed Neural Nets" und Bruno Buchberger über „Symbolic Computation: Current Trends" gehalten. Der Abendvortrag von Siegmund Brandt, der anlässlich des Einsteinjahrs unter dem Thema „Physics – Made in Berlin" stand, zeichnete ein lebhaftes Bild von der ungemein fruchtbaren wissenschaftlichen Atmosphäre im Berlin der ersten Dekaden des zwanzigsten Jahrhunderts.

Die Konferenz war hervorragend organisiert und insbesondere die Bootsfahrt auf der Dahme und das anschließende Konferenzdinner werden noch lange in Erinnerung bleiben.

Thomas Hahn (München)

6. MEGA 2005 – The 8th International Symposium on Effective Methods in Algebraic Geometry

Porto Conte, Alghero, Sardinien, 26.05. – 02.06.2005

`http://www.dm.unipi.it/MEGA05`

Metodi Effettivi in Geometria Algebraica 2005 (das Akronym kann in mindestens sechs Sprachen gelesen werden) stand diesmal unter dem Zeichen des sechzigsten Geburtstags von C. Traverso, der MEGA 1990 zum ersten Mal veranstaltete und bei allen bisher 7 MEGAs aktiv mitwirkte. Mit den Attraktionen der näheren Umgebung im Norden Sardiniens (Badestrände am Meer und an der Lagune, mittelalterlicher Stadtteil von Alghero) konnte das wissenschaftliche Programm locker mithalten. Als Hauptvortragende waren H. Hauser (Singularitäten), P. Fitzpatrick (algebraische

Codierungstheorie), E. Allmann (mathematische Biologie), B. Poonen (arithmetische Geometrie) und T. Theobald (tropische Geometrie) eingeladen.

Die Teilnehmer und Teilnehmerinnen wurden eingeladen, Arbeiten zu einem Sonder-Doppelband des Journal of Symbolic Computation einzureichen. Die Einladung gilt auch für Forscher, die an den Themen der Konferenz interessiert sind. Die Einreichfrist endet am 31. Oktober 2005.

Josef Schicho (Linz)

7. Tagung der Fachgruppe Computeralgebra

Kassel, 02. – 04.06.2005

`http://www.mathematik.uni-kassel.de/compmath/ca2005.htm`

Einen ausführlichen Bericht zu dieser Tagung der Fachgruppe finden Sie auf Seite 6

8. Zahlentheorie-Konferenz anlässlich des 60. Geburtstags von Prof. Dr. Michael E. Pohst

Berlin, 09. – 11.06.2005

`http://www.math.tu-berlin.de/~kant/MP60`

Die Konferenz fand vom Donnerstagmittag bis Samstagmittag an der Technischen Universität Berlin statt. Unter den 42 Teilnehmern waren langjährige Kollegen, der Doktorvater Prof. Curt Meyer sowie ehemalige Doktoranden und Diplomanden des Jubilars.

Das wissenschafliche Programm reichte von zahlentheoretischen Themen wie Klassenkörpertheorie oder komplexe Multiplikation über digitale Unterschriften in der Gegenwart von Quantencomputern bis zu graphischen Benutzeroberflächen für Computeralgebrasysteme.

Die Vortragstitel in chronologischer Reihenfolge waren:

Helmut Koch: Remarks on the foundations of class field theory, Jürgen Klüners: The number of S4-fields with given discriminant, Claus Fieker: Computations of and with units, Johannes Buchmann: Post-quantum signatures, Henri Cohen: Applications of the p-adic gamma function, Attila Pethö: Units and diophantine equations, Mark Watkins: Indefinite LLL, Ken Nakamula: New polynomials producing absolute pseudoprimes with many factors, Marcus Wagner: On the Computation of Humbert-Hermite constants of real quadratic number fields, Aneesh Karve: GiANT - Graphical Interfaces for Computer Algebra Systems, Reinhard Schertz: Applications of complex multiplication, Curt Meyer: Über die Diskriminante der n-ten Teilwerte der Weierstraßschen p-Funktion, Florin Nicolae: On Artin's L-functions, Tobias Finis: On anticyclotomic mu-invariants for CM fields.

Weitere Informationen über die Konferenz sowie Folien zu einigen der Vorträge sind über die oben angegebene Webseite abrufbar.

Das Journal de Théorie des Nombres de Bordeaux wird eine Prof. Pohst anlässlich seines 60. Geburtstags gewidmete Ausgabe herausbringen, für die Arbeiten noch bis Ende Dezember 2005 eingereicht werden können.

Florian Heß (Berlin)

9. ISSAC 2005 – International Symposium on Symbolic and Algebraic Computation

Peking, China, 24. – 27.07.2005

`http://www.mmrc.iss.ac.cn/~issac2005`

In Peking (Beijing), der Hauptstadt Chinas, fand in diesem Jahr die jährliche Tagung ISSAC 2005 (International Symposum on Symbolic and Algebraic Computation) statt. Diese wurde von etwa 130 Teilnehmern besucht, unter diesen auch viel „Nachwuchs“ aus China.

Tagungsfoto

Überschattet war das Treffen vom Tode zweier Kollegen, Prof. Karin Gatermann und Prof. Manuel Bronstein, die beide in diesem Jahr jung verstorben sind. Beide haben viel zu vorherigen ISSAC-Treffen beigetragen, sowohl durch wissenschaftliche Beiträge als auch durch Mitarbeit bei der Organisation der ISSAC-Symposien. Die Konferenz wurde dem Andenken beider gewidmet.

Wie immer wurde ein breites Spektrum aus dem Symbolischen Rechnen durch das wissenschaftliche Programm abgedeckt. In diesem Jahr wurden 48 Paper akzeptiert und präsentiert. Einzelheiten kann man der Webseite der Konferenz entnehmen.

Als beste Paper wurden vom Programmkomittee ausgewählt:

E. Kaltofen, P. Koiran: „On the Complexity of Factoring Bivariate Supersparse (lacunary) Polynomials“ und B. Mourrain, P Trebuchet: „Generalized Normal Forms and Polynomial System Solving“.

Eingeladene Vorträge wurden gehalten von Bruno Buchberger: „A View on the Future of Symbolic Calculation“, Bruno Salvy: „D-finiteness: Algorithms and Applications“ und Wu Wen-Tsun: „On a Finite Kernel Theorem for Polynomial-Type Optimization Problems and Some of its Applications“.

Vor der Tagung fanden zwei weitere Workshops statt: AMC (Workshop on Algebraic Methods in Cryptography) und IAMC (Internet Accessible Mathematical Computation).

Die Tagung fand in dem Hotelkomplex des Beijing Friendship Hotels statt, das neben der Tagung ISSAC noch eine weitere Tagung zur Geschichte der Wissenschaft mit über 1000 Teilnehmern mühelos unterbringen konnte.

Die Organisation der ISSAC war im Großen und Ganzen gut, das Tagungsgebäude war gut ausgestattet. Wie bereits im letzten Jahr erhielten die Teilnehmer eine Daten-CD mit sämtlichen Beiträgen in pdf-Form und weiterem Material, z. B. Computeralgebra-Software.

Hörsaal

Die nächsten ISSAC-Tagungen finden im Juli 2006 in Genua (Italien) und 2007 in Waterloo (Kanada) statt.

Winfried Neun (Berlin)

10. International Conference on Difference Equations, Special Functions and Applications

München, 25. – 30.07.2005

`http://www-m6.ma.tum.de/~ruffing`

Diese Tagung wurde gemeinsam von den drei Gesellschaften *International Society of Difference Equations* (ISDE, `http://web.umr.edu/~isde`), *Orthogonal Polynomials, Special Functions and Applications* (OPSFA, `http://math.nist.gov/opsf`) und *Symmetries and Integrability of Difference Equations* (SIDE, `http://www.physics.utu.fi/theory/SIDE`) veranstaltet. Sie setzte Tagungsreihen dieser drei großen Organisationen fort. Erwartungsgemäß war die Tagung daher sehr gut besucht und hatte 313 Teilnehmer.

Das Scientific Committee setzte sich wie folgt zusammen: Richard Askey, Bernd Aulbach, Christian Berg, Alexander Bobenko, Saber Elaydi, Basil Grammaticos, Jarmo Hietarinta, Mourad Ismail, Nalini Joshi, Gerry Ladas, Rupert Lasser, Lance Littlejohn, Vassilis Papageorgiou, Allan Peterson, George Sell, Alexandr Sharkovsky, Sergei Suslov, Pavel Winternitz. Da Bernd Aulbach, Vorsitzender und Gründungspräsident der ISDE, am 14. Januar 2005 völlig unerwartet verstorben war (`http://www.math.uni-augsburg.de/~aulbachb`), wurde die Tagung seinem Andenken gewidmet.

Es gab insgesamt 32 einstündige Hauptvorträge und unzählige halbstündige Sektionsvorträge, die in sechs Parallelsektionen stattfanden.

Aus der Sicht der Computeralgebra war der Freitag besonders interessant. Der Berichterstatter war aufgefordert worden, nach seinem Hauptvortrag zum Thema *Computer algebra algorithms for orthogonal polynomials* einen Workshop zum Computeralgebraeinsatz in den tagungsrelevanten Gebieten zu organisieren. Im Rahmen dieses Workshops fanden folgende Vorträge statt: Marko Petkovsek: *Solution spaces of hypergeometric systems and the structure of hypergeometric terms*, Anne Fredet: *Linear differential equations and exponential extensions*, Peter Paule: *Contiguous relations and creative telescoping*, Mark van Hoeij: *Solving linear ODEs in terms of solutions of linear ODEs of lower order.*

Insbesondere stellte Mark van Hoeij in seinem Vortrag die Lösung eines Problems von W. Norrie Everitt vor, welches dieser in seinem Hauptvortrag und in der Problem Session am Montag gestellt hatte, und zeigte, wie man mit seiner Maple-Software und der richtigen Strategie die gesuchten Lösungen einer gegebenen Differentialgleichung vierter Ordnung finden kann.

Koepf, van Hoeij, Askey und Fredet

Weitere Informationen kann man der oben genannten Webseite der Tagung entnehmen.

Wolfram Koepf (Kassel)

Tagungsfoto

1. **AC 2005 – 6th Symposium on Algebra and Computation**

   Tokyo, 15. – 18.11.2005

   `http://tnt.math.metro-u.ac.jp/ac/2005/ac05.en.html`

   The Symposium on Algebra and Computation is a workshop held every two years aiming to create an active interrelationship between several branches of algebra and computer science.

   Topics include algorithms in algebra and discrete mathematics, cryptography, coding theory, symbolic computation and many more.

2. **ASCM 2005 – Asian Symposium on Computer Mathematics**

   Seoul, 08. – 10.12.2005

   `http://newton.kias.re.kr/ASCM2005`

   The Asian Symposium on Computer Mathematics (ASCM) is a series of conferences which offers an opportunity for participants to present original research, to learn of research progress and new developments, and to exchange ideas and views on doing mathematics using computers.

   The previous ASCM 1995, 1996, 1998, 2000, 2001, 2003 in the series were held in Beijing (China), Kobe (Japan), Lanzhou (China), Chiang Mai (Thailand), Matsuyama (Japan), and Beijing (China) respectively.

   The 7-th ASCM will be held at the Korea Institute for Advanced Study, Seoul Korea. This year, the meeting will focus on computer algebra.

   The meeting is sponsored by Korea Institute for Advanced Study.

3. **Gedenkkolloquium für Karin Gatermann**

   Hamburg, 06. – 07.01.2006

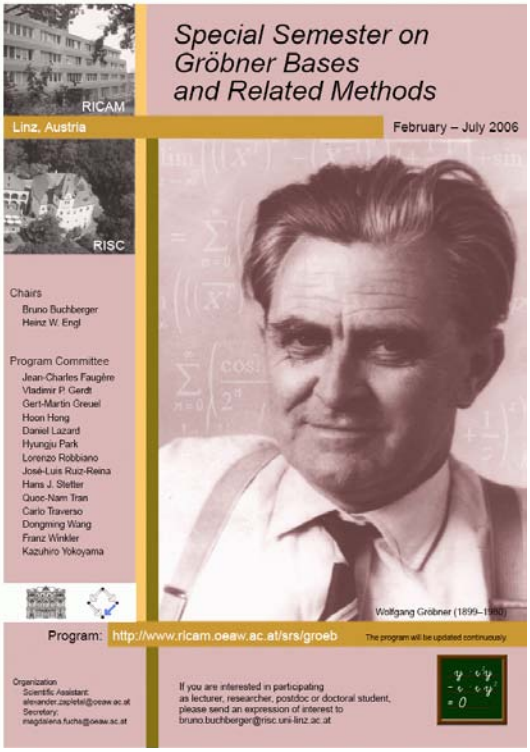
   `http://kolloquium.math.uni-hamburg.de/gatermann/`

   Am 6. und 7. Januar 2006 findet in Hamburg ein Kolloquium zu Ehren der verstorbenen Karin Gatermann statt. Als Sprecher haben zugesagt: Rob Corless (Western Ontario), Ronald Cools (Leuven), Serkan Hosten (San Francisco), Wolfram Koepf (Kassel), Markus Kirkilionis (Heidelberg), Anke Sensse (Leverkusen) und Matthias Wolfrum (Berlin).

4. **Special Semester on Gröbner Bases and Related methods**

   Linz, 02 – 07 2006

   `http://www.ricam.oeaw.ac.at/srs/groeb`



   The algorithmic theory of Gröbner bases has been introduced in 1965 by Bruno Buchberger with various forerunners since the end of the 19th century and various related theories. In the meantime, the method of Gröbner bases has been heavily studied and is now available in all major mathematical software systems.

   The special semester on Gröbner bases aims at bringing together researchers from all over the world for joint research on Gröbner bases and related theories and methods. Also, through the special semester, knowledge on these theories should be made availabe and disseminated in a new way that uses both recent advances in formalized mathematics as well as web technology. Both established researchers as well as junior researchers, postdocs, and PhD students are welcome to participate in the activities of the special semester.

5. **Computers in Scientific Discovery III**

   Gent, Belgien, 06. – 09.02.2006

   `http://caagt.ugent.be/csd3`

   Computers are used in increasingly diverse ways in Mathematics and the Physical and Life Sciences. This workshop aims to bring together researchers in Mathematics, Computer Science, Chemistry and Biology to explore the links between their disciplines and to encourage new collaborations.

   The main themes are mathematics and computer science in applications for chemistry, bioinformatics, conjecture making and mathematical education. With these aims in mind,

the workshop will combine formal lectures, short talks, parallel moderated discussion sessions and of course ample time for informal discussions.

6. **CHEP 2006 – Computing in High Energy and Nuclear Physics**

   Mumbai, Indien, 13. – 17.02.2006

   `http://www.tifr.res.in/~chep06`

   CHEP conferences provide an international forum to exchange information on computing experience and needs for the High Energy Physics and Nuclear Physics communities, and to review recent, ongoing and future activities. CHEP conferences are held every 18 months.

7. **Computeralgebra in Lehre, Ausbildung und Weiterbildung V: Entdecken, Üben, Prüfen mit Computeralgebra – Neue Entwicklungen an Schule und Hochschule**

   Haus Schönenberg bei Ellwangen, 20. – 22.04.2006

   `http://www.fachgruppe-computeralgebra.de/CLAW`

   Diese Tagung der Fachgruppe Computeralgebra wurde bereits auf Seite 7 angekündigt.

8. **GCR 2006 – Geometric Computing and Reasoning (Technical track of the 21st Annual ACM Symposium on Applied Computing SAC 2006)**

   Dijon, 23. – 27.04.2006

   `http://axis.u-strasbg.fr/gcr06`

   For the past twenty years, the ACM Symposium on Applied Computing has been a primary gathering forum for applied computer scientists, computer engineers and application developers to gather, interact, and present their work.

   Geometric Computing and Reasoning (GCR) is a new track of SAC and it is dedicated to the recent trends in the domain of geometric constraint solving (GCS) and automated, or computer aided, deduction in geometry (ADG). Geometric problems are within the heart of many theoretical studies and engineering applications.

   This track will be a great opportunity to gather researchers coming from communities concerned by subject as different as constraint programming, numeric analysis, CAD, theorem proving and computer graphics.

9. **GAMM 2006 – 77. Jahrestagung der Gesellschaft für Angewandte Mathematik und Mechanik e.V. (GAMM)**

   Berlin, 27. – 31.05.2006

   `http://www3.math.tu-berlin.de/gamm_2006`

   Die Gesellschaft für Angewandte Mathematik und Mechanik e.V. (GAMM) lädt herzlich zu ihrer Jahrestagung nach Berlin vom 27.05. bis zum 31.05.2005 ein. Es ist wieder eine Sektion zum Thema Computeralgebra und Computeranalysis geplant, die von Werner Seiler (Heidelberg) organisiert wird.

10. **TC 2006 – Transgressive Computing**

    Granada, 24. – 26.06.2006

    `http://www.orcca.on.ca/conferences/tc2006`

    A conference in honor of Jean Della Dora will be held next year at the University of Granada (Spain) from the 24th to the 26th of April 2006, a few weeks after his 60th birthday. In order to acknowledge the many innovative contributions of Jean, this event will be called „Transgressive Computing 2006“.

    The presentations will include invited and contributed talks which topics comprise, but are not limited to, the various research interests of Jean, among those: Computer Algebra, Differential Equations, Pade Approximants, Parallel Computing and many more.

    Authors are invited to submit an extended abstract of at least 4 pages or a full paper of at most 16 pages in LaTeX 11pt article size. This submitted article can be written in English, French or Spanish but must include in any case English versions of the title and abstract.

11. **ACA 2006 – 12th International Conference on Applications of Computer Algebra**

    Varna, Bulgarien, 26. – 29.06.2006

    `http://www.math.bas.bg/artint/mspirid/ACA2006`

    The 12th International Conference on Applications of Computer Algebra will take place in Varna, Bulgaria. More information will soon be available on the website.

12. **ICTM 2005 – Third International Conference on Teaching of Mathematics**

    Istanbul, 30.06. – 05.07.2006

    `http://caagt.ugent.be/csd3`

    Following two very successful international conferences (ICTM98, Samos Greece, ICTM02, Crete Greece), the third International Conference on the Teaching of Mathematics will address new ways of teaching undergraduate mathematics. The conference will be based at the hotel The Marmara in the center of Istanbul. It will provide a unique international and centralized forum and bring together faculty members from countries with different educational and pedagogical systems around the world who are committed to introducing and using innovative teaching methods. The conference will be of great interest to mathematics faculty as well as anyone involved in the teaching and learning process of undergraduate mathematics.

    Topics include effective integration of computer technology (calculators, computer algebra systems, www resources) into the undergraduate curriculum.

13. **Computational Group Theory**

    Oberwolfach, 02. – 08.07.2006

    http://www.mfo.de/programme/schedule/2006

    This workshop is organized by Gerhard Hiß (Aachen), Derek Holt (Coventry) and Mike Newman (Canberra).

14. **ISSAC 2006 – International Symposium on Symbolic and Algebraic Computation**

    Genua, 16. – 19.07.2006

    ISSAC is the yearly premier international symposium in Symbolic and Algebraic Computation that provides an opportunity to learn of new developments and to present original research results in all areas of symbolic mathematical computation.

15. **CCCG 2006 – 18th Canadian Conference on Computational Geometry**

    Kingston, Ontario, Kanada, 14. – 16.08.2006

    http://www.cs.queensu.ca/cccg

    The Canadian Conference on Computational Geometry (CCCG) focuses on the mathematics of discrete geometry from a computational point of view. Abstracting and studying the geometry problems that underlie important applications of computing (such as geographic information systems, computer-aided design, simulation, robotics, solid modeling, databases, and graphics) leads not only to new mathematical results, but also to improvements in these application areas. Despite its international following, CCCG maintains the informality of a smaller workshop and attracts a large number of students.

16. **DMV Jahrestagung 2006**

    Bonn, September 2006

    . . . . . .

    Die Deutsche Mathematiker-Vereinigung lädt herzlich ein zu ihrer Jahrestagung, diesmal in Bonn. Nähere Informationen demnächst auf der Webseite der DMV.

Kurze Mitteilungen

Prof. Dr. Werner Bley (Augsburg) hat den Ruf auf eine W2-Professur an die Universität Kassel angenommen. (http://www.mathematik.uni-kassel.de)

Leider wurde im Computeralgebrarundbrief Nr. 36 (März 2005) im Artikel „Einsatz von Computeralgebrasystemen zum Entwurf mechatronischer Systeme am Beispiel von CAMeL-View“ die Email-Adresse des Autors falsch abgedruckt. Die korrekte Adresse lautet: Martin.Hahn@iXtronics.de.

Zum frühen Tod von Manuel Bronstein

Im Juni dieses Jahres verstarb unerwartet Manuel Bronstein im Alter von 42 Jahren. Er hinterlässt eine Frau und sechs Kinder. Zur Unterstützung der Familie hat sein Bruder einen Fond eingerichtet, Informationen dazu `http://www.astav.net/manu/Manuel_Bronstein_Fund.htm`.

Manuel Bronstein stammte aus Paris und studierte an der University of California in Berkeley. Seine Doktorarbeit schrieb er zum Thema „Integration of Elementary Functions“ unter der Betreuung von Maxwell Rosenlicht, Zweitgutachter war Richard J. Fateman. Von 1987 bis 1990 war er in der Scratchpad/AXIOM-Entwicklungsgruppe um Richard Jenks am IBM Thomas Watson Research Center. Ab 1990 war er dann bei G. Gonnet am Institut für wissenschaftliches Rechnen an der ETH Zürich als Oberassistent und erhielt am 1. März 1993 eine Assistenzprofessur. Im Jahr 1997 wechselte er an das französische Institut Nationale de Recherche en Informatique et en Automatique (INRIA) in Sophia Antipolis bei Nizza und trat dort die Stelle des Forschungsdirektors an. Zuletzt leitete er dort das Projekt *CAFE — Computer algebra and functional equations*.

Die Weiterentwicklung und Implementierung des Algorithmus von Risch zur Bestimmung von elementaren Stammfunktionen und als nächster Schritt das symbolische Lösen von linearen Differentialgleichungen zweiter und dritter Ordnung waren die zentralen Themen seines kurzen Forschungslebens. Seine 1997 vorgelegte Monographie zum Thema „Symbolic Integration I – Transcendental Functions“ [22] ist zum Standardwerk auf diesem Gebiet geworden und in diesem Jahr in zweiter, erweiterter Auflage neu erschienen [26] – siehe die Besprechung auf Seite 23.

Bronstein war Program Chair der ISSAC’1993 [19] und treibende Kraft als Koorganisator einer Konferenz über „Symbolic Rewriting Techniques“ [28].

Auf seiner Homepage `http://www-sop.inria.fr/cafe/Manuel.Bronstein/` finden sich eine ganze Reihe von Programmen und Bibliotheken zum Finden von Stammfunktionen und zum Lösen von linearen Differentialgleichungen zweiter und dritter Ordnung in Aldor und in Maple. Bemerkenswert ist hier eine der letzten Arbeiten Bronsteins, eine Weiterentwicklung und Implementierung einer Heuristik mit paralleler Lösungstechik zum Finden von Stammfunktionen. Seine Maple-Implementierung (datiert 10.05.2005) mit bemerkenswerten Ergebnissen hat weniger als 100 Zeilen Code!

Hier erhebt sich die Frage, wie lange diese Seite weiter erreichbar und funktionsfähig bleiben wird. INRIA, aber auch die „Computer Algebra Community“ insgesamt sind gefordert, die Arbeiten Manuel Bronsteins über die Publikationen hinaus, die am Ende aufgelistet sind, zu erhalten! Eine Bibtex-Datei mit den Arbeiten Bronsteins kann auf der Internetseite der Fachgruppe `http://www.fachgruppe-computeralgebra.de/bronstein.bib` heruntergeladen werden.

In meiner Besprechung der ersten Auflage des Buches im Computeralgebra-Rundbrief über symbolische Integration habe ich am Schluss den Wunsch ausgesprochen, dass der Autor die Zeit finden möge bald einen zweiten Teil zur Theorie der symbolischen Integration algebraischer Funktionen folgen zu lassen. Dieser Wunsch kann nicht mehr in Erfüllung gehen. Wer wird das Werk Bronsteins hier fortsetzen können?

Literatur

[1] J. Abbott, M. Bronstein, and T. Mulders. Fast deterministic computation of determinants of dense matrices. In

Proceedings of ISSAC'99, pages 197–204. ACM Press, 1999.

[2] S. A. Abramov and M. Bronstein. Hypergeometric dispersion and the orbit problem. In *Proceedings of ISSAC'2000*, pages 8–13. ACM Press, 2000.

[3] S. A. Abramov and M. Bronstein. On solutions of linear functional systems. In *Proceedings of ISSAC'2001*, pages 1–6. ACM Press, 2001.

[4] M. Bronstein. Gsolve: A faster algorithm for solving systems of algebraic equations. In *Proceedings of SYMSAC'86*, pages 247–249. ACM Press, 1986.

[5] M. Bronstein. An algorithm for the integration of elementary functions. In *Proceedings of EUROCAL'87*, volume 378 of *Springer LNCS*, pages 491–497, 1987.

[6] M. Bronstein. Fast reduction of the Risch differential equation. In *Proceedings of ISSAC'88*, volume 358 of *Springer LNCS*, pages 64–72, 1987.

[7] M. Bronstein. Symbolic integration: towards practical algorithms. In E. Tournier, editor, *Computer Algebra and Differential Equations*, pages 59–85. 1988.

[8] M. Bronstein. Simplification of real elementary functions. In *Proceedings of ISSAC'89*, pages 207–211. ACM Press, 1989.

[9] M. Bronstein. The transcendental Risch differential equation. *Journal of Symbolic Computation*, 9:49–60, 1989.

[10] M. Bronstein. A unification of Liouvillian extensions. *Applicable Algebra in Engineering, Communication and Computing*, 1:5–24, 1989.

[11] M. Bronstein. Computer algebra and indefinite integrals, computer aided proofs in analysis. *IMA Volumes in Mathematics and its Applications*, 28:33–42, 1990.

[12] M. Bronstein. Integration of elementary functions. *Journal of Symbolic Computation*, 9:117–173, 1990.

[13] M. Bronstein. Formulas for series computations. *Applicable Algebra in Engineering, Communication and Computing*, 2:195–205, 1991.

[14] M. Bronstein. The Risch differential equation on an algebraic curve. In *Proceedings of ISSAC'91*, pages 241–246. ACM Press, 1991.

[15] M. Bronstein. Integration and differential equations in computer algebra. *Programming and Computer Software*, 18:201–217, 1992.

[16] M. Bronstein. Linear ordinary differential equations: breaking through the order 2 barrier. In *Proceedings of ISSAC'92*, pages 42–48. ACM Press, 1992.

[17] M. Bronstein. On solutions of linear ordinary differential equations in their coefficient field. *Journal of Symbolic Computation*, 13:413–439, 1992.

[18] M. Bronstein, editor. *Proceedings of ISSAC'93*, New York, 1993. ACM Press.

[19] M. Bronstein. Some effective methods in pseudo-linear algebra. In *Proceedings of MEGA'94*, volume 143 of *Progress in Mathematics*, pages 105–113. Birkhauser, 1993.

[20] M. Bronstein. On the factorisation of linear ordinary differential operators. *Mathematics and Computers in Simulation*, 42:387–389, 1995.

[21] M. Bronstein. Sum-it: A strongly-typed embeddable computer algebra library. In *Proceedings of DISCO'96*, volume 1128 of *Springer LNCS*, pages 22–33. Springer, 1996.

[22] M. Bronstein, editor. *Symbolic Integration 1 - Transcendental Functions*. Algorithms and Computation in Mathematics 1. Springer-Verlag, Heidelberg, 1997.

[23] M. Bronstein. Solutions of linear ordinary difference equations in their coefficient field. *Journal of Symbolic Computation*, 29:841–877, 1999.

[24] M. Bronstein. Computer algebra algorithms for linear ordinary differential and difference equations. In *Proceedings of the third European Congress of Mathematics, vol.II*, Progress in Mathematics 202, pages 105–119. Birkhauser, 2000.

[25] M. Bronstein. Symbolic integration. In J. Grabmeier, E. Kaltofen, and V. Weispfenning, editors, *Computer Algebra Handbook*, pages 94–96. Springer-Verlag, Heidelberg, 2003.

[26] M. Bronstein, editor. *Symbolic Integration 1 - Transcendental Functions*. Algorithms and Computation in Mathematics 1. Springer-Verlag, Heidelberg, second edition, 2005. extended.

[27] M. Bronstein and A. Fredet. Solving linear ordinary differential equations over $c(x, e^{\int f(x)dx})$. In *Proceedings of ISSAC'99*, pages 173–179. ACM Press, 1999.

[28] M. Bronstein, J. Grabmeier, and V. Weispfenning, editors. *Symbolic Rewriting Techniques*. Progress in Computer Science and Applied Logic 15. Birkhauser-Verlag, Basel, 1998.

[29] M. Bronstein and S. Lafaille. Solutions of linear ordinary differential equations in terms of special functions. In *Proceedings of ISSAC'2002*, pages 23–28. ACM Press, 2002.

[30] M. Bronstein, Z. Li, and M. Wu. Picard-vessiot extensions for linear functional systems. In *Proceedings of ISSAC'2005*. ACM Press, 2005.

[31] M. Bronstein, T. Mulders, and J. A. Weil. On symmetric powers of differential operators. In *Proceedings of ISSAC'97*, pages 156–163. ACM Press, 1997.

[32] M. Bronstein and M. Petkovsek. On ore rings, linear operators and factorisation. *Programming and Computer Software*, 20:14–26, 1994.

[33] M. Bronstein and M. Petkovsek. An introduction to pseudo-linear algebra. *Theoretical Computer Science*, 157:3–33, 1995.

[34] M. Bronstein and B. Salvy. Full partial fraction decomposition of rational functions. In *Proceedings of ISSAC'93*, pages 157–160. ACM Press, 1992.

[35] M. Bronstein and W. Sit, editors. *Differential Algebra and Differential Equations*. 1999. Special Issue of the Journal of Symbolic Computation 28.

[36] M. Bronstein and P. Solé. Linear recurrences with polynomial coefficients. *Journal of Complexity*, 20:171–181, 2003.

[37] M. Bronstein and B. M. Trager. A reduction for regular differential systems. *Proceedings of MEGA'2003*, 2002.

Johannes Grabmeier (Deggendorf)
− unter Benutzung der Nachrufe der ETH Zürich und des INRIA Sophia Antipolis.

Lehrveranstaltungen zu Computeralgebra im WS 2005/2006

- **Rheinisch–Westfälische Technische Hochschule Aachen**
  Einführung in das Computeralgebrasystem MAPLE, V. Dietrich (einmalig)
  Arbeitsgemeinschaft MAPLE, V. Dietrich, G. Hartjen, Triesch, AGT2
  Computermathematik I (Begleitpraktikum), W. Plesken, G. Hartjen, V. Dietrich, P3
  Computermathematik II (Begleitpraktikum), W. Plesken, G. Hartjen, V. Dietrich, P2
  Computermathematik III (Begleitpraktikum), W. Plesken, G. Hartjen, V. Dietrich, P1
  Algebraische Systemtheorie, E. Zerz, V4 + Ü2
  Computeralgebra II, W. Plesken, V2 + Ü1

- **Universität Bayreuth**
  Codierungstheorie II, A. Kerber, V2 + Ü2
  Seminar zur Codierungstheorie, A. Kerber, S2
  Konstruktionsalgorithmen, R. Laue, V2 + Ü2

- **Technische Universität Berlin**
  Kryptographie, M. Pohst, V4 + Ü2
  Konstruktive Zahlentheorie I, M. Pohst, V2
  Seminar Algorithmische Algebra und Zahlentheorie, M. Pohst, S2

- **Technische Universität Braunschweig**
  Codierungstheorie, H. Pralle, V2 + Ü1
  Computerorientierte Mathematik, H. Weiß, V2 + Ü1

- **Universität Dortmund**
  Algorithmische Invariantentheorie, M. Kreuzer, V4 + Ü2
  Seminar Kryptographie, M. Kreuzer, S2

- **Fachhochschule Flensburg**
  Maple in Differentialgleichungen, M. Kersken, V4 + Ü2
  Analysis mit Maple, N. Pavlik, Ü1
  Mathematik IV mit Maple für Technische InformatikerInnen, P. Thieler, Ü2
  Software Tools: Maple für KommunikationstechnologInnEn, P. Thieler, Ü2

- **Justus–Liebig–Universität Gießen**
  Computeralgebra, T. Sauer, V4 + Ü2

- **Technische Universität Hamburg-Harburg**
  Diskrete Mathematik Ia, K.-H. Zimmermann, V2 + Ü1
  Diskrete Mathematik II, K.-H. Zimmermann, V2 + Ü1
  Softwarepraktikum, K.-H. Zimmermann, P2
  Algebraische Methoden, Batra, V2

- **Martin-Luther-Universität Halle (Saale)**
  Seminar Wirtschaftsmathematik mit MATHEMATICA, MATHCAD und MATLAB, H. Benker, S4
  Seminar Mathematik mit MATHCAD und MATLAB, H.Benker, S2

- **Universität Heidelberg**
  Konstruktive Kommutative Algebra, W. M. Seiler, V4
  Praktikum Konstruktive Kommutative Algebra, M. Dettweiler, W. M. Seiler, P4

- **Universität Kaiserslautern**
  Einführung in die Computeralgebra, G. Malle, V2 + Ü1
  Special Topics in Computeralgebra, A. Frühbis-Krüger, V2
  Algorithmic Algebraic Number Theory, G. Malle, V4 + Ü2
  Kryptographie und Kodierungstheorie, G. Pfister, V4 + Ü2

- **Universität Kassel**
  Einführung in Computeralgebrasysteme I, R. Schaper, V2
  Computeralgebra I, P. Horn, W. Koepf, V4 + Ü2
  Computeralgebra und orthogonale Polynome, W. Koepf, V3 + Ü1
  Seminar Computational Mathematics, W. Koepf, S2
  Oberseminar Computational Mathematics, W. Koepf, S2

- **Universität Leipzig**
  Einführung in das Symbolische Rechnen, H.-G. Gräbe, V2 + Ü1
  Algorithmen für Zahlen und Primzahlen, H.-G. Gräbe, V2 + Ü1

- **Technische Universität München**
  Ausgewählte Kapitel der Computeralgebra, M. Kaplan, V2

- **Universität Oldenburg**
  Differentielle Algebra, W.Schmale, V2 + S2

- **Universität Paderborn**
  Kryptographie, J. Blömer, V2 + Ü1
  Kryptographische Protokolle, J. Blömer, V2 + Ü1
  Proseminar Kryptographie, J. Blömer, PS2
  Oberseminar Codes und Kryptographie, J. Blömer, OS2
  Mathematik am Computer, D. Guhe, V2 + Ü2
  Oberseminar Komplexität und Algorithmen, P. Bürgisser, OS2

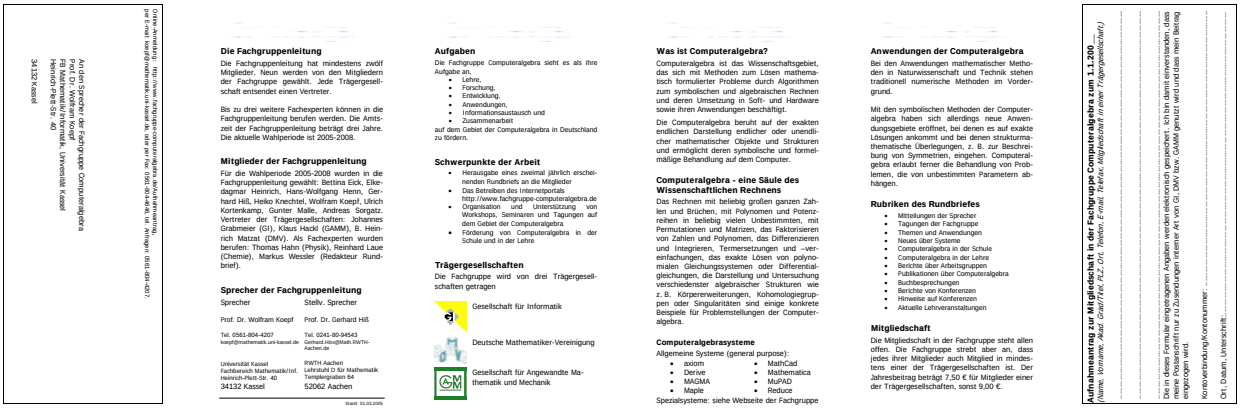
  Algorithmische Behandlung von Differentialgleichungen, B. Fuchssteiner, V4 + Ü2
  Projektstudium: Entwicklung und Implementierung eines Computeralgebrasystems, B. Fuchssteiner, P4
  MuPAD Seminar, B. Fuchssteiner und W. Oevel, OS2

- **Universität Tübingen**
  Seminar Computeralgebra, R. Loos, S2

- **Universität Würzburg**
  Angewandte Physik I (Computational Physics), H. Hinrichsen, V2 + Ü4

Flyer der Fachgruppe

Die Fachgruppenleitung hat zu Werbezwecken einen Flyer gedruckt. Wenn Sie als Mitglied der Fachgruppe beispielsweise auf einer Tagung Flyer der Fachgruppe verteilen wollen, wenden Sie sich bitte an den Sprecher Wolfram Koepf.



Online-Anmeldung: http://www.fachgruppe-computeralgebra.de/Anmeldung.pdf
e-mail: koepf@mathematik.uni-kassel.de, tel: (0561) 804-4207, fax: (0561) 804-4287

An den Sprecher der Fachgruppe Computeralgebra
Prof. Dr. Wolfram Koepf
Fachbereich Mathematik, Universität Kassel
Heinrich-Plett-Str. 40
34132 Kassel

Die Fachgruppenleitung

Die Fachgruppenleitung hat mindestens zwölf Mitglieder. Neun werden von den Mitgliedern der Fachgruppe gewählt. Jede Trägergesellschaft entsendet einen Vertreter.

Bis zu drei weitere Fachexperten können in die Fachgruppenleitung berufen werden. Die Amtszeit der Fachgruppenleitung beträgt drei Jahre. Die aktuelle Wahlperiode ist 2005-2008.

Mitglieder der Fachgruppenleitung

Für die Wahlperiode 2005-2008 wurden in die Fachgruppenleitung gewählt: Bettina Eick, Elke-dagmar Heinrich, Hans-Wolfgang Henn, Gerhard Hiß, Heiko Knechtel, Wolfram Koepf, Ulrich Kortenkamp, Gunter Malle, Andreas Sorgatz. Vertreter der Trägergesellschaften: Johannes Grabmeier (GI), Klaus Hackl (GAMM), B. Heinrich Matzat (DMV). Als Fachexperten wurden berufen: Thomas Hahn (Physik), Reinhard Laue (Chemie), Markus Wessler (Redakteur Rundbrief).

Sprecher der Fachgruppenleitung

Sprecher	Stellv. Sprecher
Prof. Dr. Wolfram Koepf	Prof. Dr. Gerhard Hiß
Tel. 0561-804-4207	Tel. 0241-80-94543
koepf@mathematik.uni-kassel.de	Gerhard.Hiss@Math.RWTH-Aachen.de
Universität Kassel	RWTH Aachen
Fachbereich Mathematik/Inf.	Lehrstuhl D für Mathematik
Heinrich-Plett-Str. 40	Templergraben 64
34132 Kassel	52062 Aachen

Stand: 01.03.2005

Aufgaben

Die Fachgruppe Computeralgebra sieht es als ihre Aufgabe an,

- Lehre,
- Forschung,
- Entwicklung,
- Anwendungen,
- Informationsaustausch und
- Zusammenarbeit

auf dem Gebiet der Computeralgebra in Deutschland zu fördern.

Schwerpunkte der Arbeit

- Herausgabe eines zweimal jährlich erscheinenden Rundbriefs an die Mitglieder
- Das Betreiben des Internetportals http://www.fachgruppe-computeralgebra.de
- Organisation und Unterstützung von Workshops, Seminaren und Tagungen auf dem Gebiet der Computeralgebra
- Förderung von Computeralgebra in der Schule und in der Lehre

Trägergesellschaften

Die Fachgruppe wird von drei Trägergesellschaften getragen

Gesellschaft für Informatik

Deutsche Mathematiker-Vereinigung

Gesellschaft für Angewandte Mathematik und Mechanik

Was ist Computeralgebra?

Computeralgebra ist das Wissenschaftsgebiet, das sich mit Methoden zum Lösen mathematisch formulierter Probleme durch Algorithmen zum symbolischen und algebraischen Rechnen und deren Umsetzung in Soft- und Hardware sowie ihren Anwendungen beschäftigt.

Die Computeralgebra beruht auf der exakten endlichen Darstellung endlicher oder unendlicher mathematischer Objekte und Strukturen und ermöglicht deren symbolische und formelmäßige Behandlung auf dem Computer.

Computeralgebra - eine Säule des Wissenschaftlichen Rechnens

Das Rechnen mit beliebig großen ganzen Zahlen und Brüchen, mit Polynomen und Potenzreihen in beliebig vielen Unbestimmten, mit Permutationen und Matrizen, das Faktorisieren von Zahlen und Polynomen, das Differenzieren und Integrieren, Termersetzungen und –vereinfachungen, das exakte Lösen von polynomialen Gleichungssystemen oder Differentialgleichungen, die Darstellung und Untersuchung verschiedenster algebraischer Strukturen wie z. B. Körpererweiterungen, Kohomologiegruppen oder Singularitäten sind einige konkrete Beispiele für Problemstellungen der Computeralgebra.

Computeralgebrasysteme

Allgemeine Systeme (general purpose):

- axiom
- Derive
- MAGMA
- Maple
- MathCad
- Mathematica
- MuPAD
- Reduce

Spezialsysteme: siehe Webseite der Fachgruppe

Anwendungen der Computeralgebra

Bei den Anwendungen mathematischer Methoden in Naturwissenschaft und Technik stehen traditionell numerische Methoden im Vordergrund.

Mit den symbolischen Methoden der Computeralgebra haben sich allerdings neue Anwendungsgebiete eröffnet, bei denen es auf exakte Lösungen ankommt und bei denen strukturmathematische Überlegungen, z. B. zur Beschreibung von Symmetrien, eingehen. Computeralgebra erlaubt ferner die Behandlung von Problemen, die von unbestimmten Parametern abhängen.

Rubriken des Rundbriefes

- Mitteilungen der Sprecher
- Tagungen der Fachgruppe
- Themen und Anwendungen
- Neues über Systeme
- Computeralgebra in der Schule
- Computeralgebra in der Lehre
- Berichte über Arbeitsgruppen
- Publikationen über Computeralgebra
- Buchbesprechungen
- Berichte von Konferenzen
- Hinweise auf Konferenzen
- Aktuelle Lehrveranstaltungen

Mitgliedschaft

Die Mitgliedschaft in der Fachgruppe steht allen offen. Die Fachgruppe strebt aber an, dass jedes ihrer Mitglieder auch Mitglied in mindestens einer der Trägergesellschaften ist. Der Jahresbeitrag beträgt 7,50 € für Mitglieder einer der Trägergesellschaften, sonst 9,00 €.

Aufnahmeantrag zur Mitgliedschaft in der Fachgruppe Computeralgebra zum 1.1.200_
(Name, Vorname, Akad. Grad/Titel, PLZ, Ort, Telefon, E-mail, Titel/Art, Mitgliedschaft in einer Trägergesellschaft)

.....

.....

.....

Die in dieses Formular eingetragenen Angaben werden elektronisch gespeichert. Ich bin damit einverstanden, dass meine Postanschrift nur zu Zusendungen interner Art von GI, DMV bzw. GAMM genutzt wird und dass mein Beitrag eingezogen wird.

Kontoverbindung/Kontonummer: .....

Ort, Datum, Unterschrift: .....

Aufnahmeantrag für Mitgliedschaft in der Fachgruppe Computeralgebra

(Im folgenden jeweils Zutreffendes bitte im entsprechenden Feld [ ] ankreuzen bzw. _____ ausfüllen.)

Titel/Name: _____ Vorname: _____

Privatadresse
Straße/Postfach: _____

PLZ/Ort: _____ Telefon: _____

e-mail: _____ Telefax: _____

Dienstanschrift
Firma/Institution: _____

Straße/Postfach: _____

PLZ/Ort: _____ Telefon: _____

e-mail: _____ Telefax: _____

Gewünschte Postanschrift: [ ] Privatadresse [ ] Dienstanschrift

1. Hiermit beantrage ich zum 1. Januar 200__ die Aufnahme als Mitglied in die Fachgruppe

   Computeralgebra (CA) (bei der GI: 0.2.1).

2. Der Jahresbeitrag beträgt €7,50 bzw. €9,00. Ich ordne mich folgender Beitragsklasse zu:
   - [ ] **€7,50** für Mitglieder einer der drei Trägergesellschaften
     - [ ] GI Mitgliedsnummer: _____
     - [ ] DMV Mitgliedsnummer: _____
     - [ ] GAMM Mitgliedsnummer: _____

     Der Beitrag zur Fachgruppe Computeralgebra wird mit der Beitragsrechnung der Trägergesellschaft in Rechnung gestellt. (Bei Mitgliedschaft bei mehreren Trägergesellschaften wird dies von derjenigen durchgeführt, zu der Sie diesen Antrag schicken.) [ ] Ich habe dafür bereits eine Einzugsvollmacht erteilt. Diese wird hiermit für den Beitrag für die Fachgruppe Computeralgebra erweitert.
   - [ ] **€7,50**. Ich bin aber noch nicht Mitglied einer der drei Trägergesellschaften. Deshalb beantrage ich gleichzeitig die Mitgliedschaft in der

     [ ] GI [ ] DMV [ ] GAMM.

     und bitte um Übersendung der entsprechenden Unterlagen.
   - [ ] **€9,00** für Nichtmitglieder der drei Trägergesellschaften. [ ] Gleichzeitig bitte ich um Zusendung von Informationen über die Mitgliedschaft in folgenden Gesellschaften:

     [ ] GI [ ] DMV [ ] GAMM.
3. Die in dieses Formular eingetragenen Angaben werden elektronisch gespeichert. Ich bin damit einverstanden, dass meine Postanschrift durch die Trägergesellschaften oder durch Dritte nach Weitergabe durch eine Trägergesellschaft wie folgt genutzt werden kann (ist nichts angekreuzt, so wird c. angenommen).
   - [ ] a. Zusendungen aller Art mit Bezug zur Informatik, Mathematik bzw. Mechanik.
   - [ ] b. Zusendungen durch wiss. Institutionen mit Bezug zur Informatik, Mathematik bzw. Mechanik.
   - [ ] c. Nur Zusendungen interner Art von GI, DMV bzw. GAMM.

Ort, Datum: _____ Unterschrift: _____

Bitte senden Sie dieses Formular an:

Sprecher der Fachgruppe Computeralgebra
Prof. Dr. Wolfram Koepf
Fachbereich Mathematik/Informatik
Universität Kassel
Heinrich-Plett-Str. 40
34132 Kassel
0561-804-4207,-4646 (Fax)
koepf@mathematik.uni-kassel.de

Fachgruppenleitung Computeralgebra 2005-2008

Sprecher:
Prof. Dr. Wolfram Koepf
Universität Kassel
Fachbereich Mathematik/Informatik
Heinrich-Plett-Str. 40
34132 Kassel
0561-804-4207,-4646 (Fax)
koepf@mathematik.uni-kassel.de
http://www.mathematik.uni-kassel.de/~koepf

Stellvertretender Sprecher:
Prof. Dr. Gerhard Hiß
Lehrstuhl D für Mathematik
RWTH Aachen
Templergraben 64
52062 Aachen
0241-80-94543, -92108 (Fax)
Gerhard.Hiss@Math.RWTH-Aachen.de
http://www.math.rwth-aachen.de/~Gerhard.Hiss

Prof. Dr. Bettina Eick
Arbeitsgruppe Algebra und diskrete Mathematik
Institut Computational Mathematics
Technische Universität Braunschweig
Pockelsstrasse 14
38106 Braunschweig
0531-391-7525, -8206 (Fax)
beick@tu-bs.de
http://www.tu-bs.de/~beick

Vertreter der GI,
Prof. Dr. Johannes Grabmeier
Fachhochschule Deggendorf
94469 Deggendorf
0991-3615-141
johannes.grabmeier@fh-deggendorf.d
http://www.fh-deggendorf.de/
home/allgemein/professoren/grabmeier

Vertreter der GAMM,
Fachreferent Computational Engineering:
Prof. Dr. Klaus Hackl
Ruhr-Universität Bochum
Lehrstuhl für Allgemeine Mechanik
Universitätsstr. 150
44780 Bochum
0234-32-26025, -14154 (Fax)
hackl@am.bi.rub.de

Fachexperte Physik:
Dr. Thomas Hahn
Max-Planck-Institut für Physik
Föhringer Ring 6
80805 München
089-32354-300, -304 (Fax)
hahn@feynarts.de
http://wwwth.mppmu.mpg.de/members/hahn

Fachreferentin Fachhochschulen:
Prof. Dr. Elkedagmar Heinrich
Fachhochschule Konstanz
Fachbereich Informatik
78462 Konstanz
07531-206-343, -559 (Fax)
heinrich@fh-konstanz.de
http://www.in.fh-konstanz.de/de/
Fachbereich/Kontakt/persseiten_nbc/heinrich.html

Fachreferent Lehre und Didaktik:
Prof. Dr. Hans-Wolfgang Henn
Universität Dortmund
Fachbereich Mathematik
44227 Dortmund
0231-755-2939, -2948 (Fax)
wolfgang.henn@mathematik.uni-dortmund.de
http://www.wolfgang-henn.de

Fachreferent Schule:
OStD. Heiko Knechtel
An der Tränke 2a
31675 Bückeburg
05722-23628
HKnechtel@aol.com

Fachreferent Internet/Math. Software:
Prof. Dr. Ulrich Kortenkamp
Technische Universität Berlin
Fachbereich Mathematik
Straße des 17. Juni 136
10623 Berlin
030-314-25748, -21269 (Fax)
kortenkamp@math.tu-berlin.de
http://www.kortenkamps.net/

Fachexperte Chemie:
Prof. Dr. Reinhard Laue
Universität Bayreuth
Mathematisches Institut
95440 Bayreuth
0921-55-3275, -3385 (Fax)
laue@uni-bayreuth.de
http://www.mathe2.uni-bayreuth.de/
people/laue.html

Prof. Dr. Gunter Malle
Universität Kaiserslautern
Fachbereich Mathematik
Gottlieb-Daimler-Straße
67663 Kaiserslautern
0631-205-2264, -3989 (Fax)
malle@mathematik.uni-kl.de
http://www.mathematik.uni-kl.de/~malle

Vertreter der DMV:
Prof. Dr. B. Heinrich Matzat
IWR, Universität Heidelberg,
Im Neuenheimer Feld 368
69120 Heidelberg
06221-54-8242,-8318(Sekr.),-8850 (Fax)
matzat@iwr.uni-heidelberg.de
http://www.iwr.uni-heidelberg.de/
groups/compalg/matzat

Fachexperte Industrie:
Dr. Andreas Sorgatz
SciFace Software
Technologiepark 11
33100 Paderborn
05251-6407-51, -99 (Fax)
sorgatz@sciface.com
http://math-www.uni-paderborn.de/~andi

Fachexperte Rundbrief:
Dr. Markus Wessler
Kopernikusstr. 6
81679 München
089-69777336
wessler@mathematik.uni-kassel.de
http://www.mathematik.uni-kassel.de/~wessler