Oktober 2013

Computeralgebra Rundbrief

> Ausgabe 53

- Symbolic summation and its applications in particle physics
- Neue Algorithmen für das DLP in kleiner Charakteristik
- FindStat
- Kubisch, quartisch und so weiter
- Mathematik hören und Musik sehen mit Hilfe eines Computeralgebrasystems.

www.fachgruppe-computeralgebra.de

Werbeseite

Computeralgebra-Rundbrief

Nr. 53 Oktober 2013

Inhaltsverzeichnis

Impressum

Der Computeralgebra-Rundbrief wird herausgegeben von der Fachgruppe Computeralgebra der GI, DMV und GAMM
(verantwortliche Redakteure: Prof. Dr. Michael Cuntz, Dr. Gohar Kyureghyan, car@mathematik.de)

Der Computeralgebra-Rundbrief erscheint halbjährlich, Redaktionsschluss 15.02. und 15.09. 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 die verantwortlichen Redakteure.

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.dmv.mathematik.de

GAMM (Gesellschaft für Angewandte
Mathematik und Mechanik e.V.)
Technische Universität Dresden
Institut für Statik und Dynamik der
Tragwerke
01062 Dresden
Telefon 0351-463-33448
Telefax 0351-463-37086
GAMM@mailbox.tu-dresden.de
http://www.gamm-ev.de

Mitteilungen der Sprecher

Liebe Mitglieder der Fachgruppe Computeralgebra,

die Amtszeit der derzeitigen Fachgruppenleitung läuft im Frühjahr 2014 aus. Zur Wahl der neuen Fachgruppenleitung erhalten Sie mit diesem Rundbrief die Wahlunterlagen. Ausführliche Informationen zum Wahlverfahren und zu den Kandidaten finden Sie weiter unten in dieser Rubrik.

Die Fachgruppenleitung traf sich am Freitag, den 27. September 2013 an der Universität Kassel zu ihrer Herbstsitzung. Die wichtigsten Themen der Sitzung waren die Tagungsaktivitäten der Fachgruppe, der Rundbrief sowie die Organisationsstruktur der Fachgruppe.

Nächstes Jahr wird wieder die Computeralgebra-Tagung der Fachgruppe in Kassel stattfinden, und zwar von Donnerstag 15. Mai, bis Samstag, 17. Mai. Die Tagung bietet gerade für Nachwuchswissenschaftler sehr gute Möglichkeiten, ihre Ergebnisse einem interessierten Publikum vorzustellen und in Sachen Computeralgebra neue Kontakte untereinander, aber auch zu erfahrenen Wissenschaftlerinnen und Wissenschaftlern sowie zu Vertretern großer Computeralgebrastandorte in Deutschland zu knüpfen. Bitte unterstützen Sie unsere Tagung duch Bekanntmachung in Ihrem Bereich sowie wenn möglich durch Ihre Teilnahme!

Die zweite Ausgabe unserer Industrie-Tagung „Industrial Applications and Prospects of Computer Algebra“, die erstmals 2011 in Kaiserslautern abgehalten wurde, konnte dieses Jahr nicht wie geplant im September stattfinden. Durch den Verlust von Herrn Hofmeister und durch personelle Einschränkungen in der Fachgruppenleitung konnten Planung und Vorbereitung nicht im gewünschten Maß vorangetrieben werden. Im Hinblick auf die Computeralgebra-Tagung und weitere Tagungen im nächsten Jahr ist die Ausrichtung der zweiten Industrie-Tagung erst wieder für 2015 angedacht.

In den nächsten Wochen steht eine Revision der Organisationsstruktur der Fachgruppe Computeralgebra an. Hierzu ist die Rolle unserer drei Trägerorganisationen DMV, GI und GAMM vereins- und steuerrechtlichen Gesichtspunkten anzupassen, wobei eine Neufassung der Kooperationsvereinbarung zwischen DMV, GI und GAMM sowie Modifikationen bei der Mitgliederverwaltung erforderlich werden. Betroffen sind diejenigen Mitglieder der Fachgruppe, welche Mitglied in der DMV oder GAMM, aber nicht Mitglied der GI sind. Wir werden die betroffenen Mitglieder in getrennter Post kontaktieren und über die Änderungen voraussichtlich im nächsten Rundbrief berichten.

Nun zur Neuwahl der Fachgruppenleitung: Die Fachgruppenleitung hat zwölf Mitglieder, von denen drei von den beteiligten Trägergesellschaften als deren Vertreter bestimmt werden. Die restlichen neun Leitungsmitglieder werden von allen Mitgliedern der Fachgruppe gewählt. Die Amtszeit der Fachgruppenleitung beträgt nach unserer Ordnung drei Jahre.

Von den von Ihnen zu dieser Wahl vorgeschlagenen Kollegen haben sich 13 bereit erklärt zu kandidieren. Sie werden Ihnen im Folgenden kurz vorgestellt:

- **Dr. Claus Diem**, *40, Privatdozent an der Universität Leipzig und Heisenbergstipendiat. Arbeitsgebiete: Algebraische Geometrie und algorithmische Mathematik mit besonderem Interesse am diskreten Logarithmusproblem für Kurven über endlichen Körpern. Unter anderem Autor auf dem Blog "ellipticnews - the elliptic curve cryptography blog" (`ellipticnews.wordpress.com`)*
`http://www.math.uni-leipzig.de/~diem`

- **Prof. Dr. Bettina Eick**, *45, Professorin am Institut Computational Mathematics der Technischen Universität Braunschweig. Arbeitsgebiete: Algebra, speziell die Entwicklung von Algorithmen in der Gruppentheorie und in angrenzenden Gebieten der Algebra sowie ihre Anwendung. Mitarbeit am Computeralgebrasystem GAP.*
`http://www.icm.tu-bs.de/~beick`

- **Prof. Dr. Claus Fieker**, *44, Professor für konstruktive Zahlentheorie und Computeralgebra an der TU Kaiserslautern. Arbeitsgebiete: Computeralgebra, konstruktive Zahlentheorie, Klassenkörpertheorie, Darstellungstheorie und Galois Theorie. Mitentwickler von Magma (elf Jahre in Sydney), Kant und Singular.*
`http://www.mathematik.uni-kl.de/~fieker`

- **Prof. Dr. Anne Frühbis-Krüger**, *43, wissenschaftliche Mitarbeiterin und apl. Professorin am Institut für Algebraische Geometrie der Leibniz Universität Hannover, Arbeitsgebiete: Algorithmische Singularitätentheorie, Algorithmische Algebraische Geometrie, insbesondere Desingularisierung und deren Anwendungen, seit 1996 Mitarbeit an der Entwicklung des Computeralgebrasystems SINGULAR.*
`http://gandalf.krueger-berg.de/~anne`

- **Prof. Dr. Meinolf Geck**, *50, Lehrstuhl für Algebra an der Universität Stuttgart. Arbeitsgebiete: Darstellungstheorie, algebraische Lie-Theorie, Computeralgebra. Mitautor des Chevie-Pakets, Entwicklung des PyCox-Pakets.*
`http://www.mathematik.uni-stuttgart.de/~geckmf`

- **Dr. Thomas Hahn**, *42, wissenschaftlicher Mitarbeiter am Max-Planck-Institut für Physik, München. In der Fachgruppenleitung seit 2002 als Fachexperte Physik, Autor der Computeralgebra-Softwarepakete FeynArts und FormCalc für Rechnungen im Bereich der Teilchenphysik.*
`http://wwwth.mppmu.mpg.de/members/hahn`

- **Prof. Dr. Florian Heß**, *43, Professor am Institut für Mathematik der Carl von Ossietzky Universität Oldenburg. Arbeitsgebiete: Algorithmische algebraische Zahlentheorie und Geometrie, speziell algebraische Funktionenkörper, Kurven und Anwendungen auf Kryptographie und Codierungstheorie. Umfangreiche Mitarbeit an den Computeralgebrasystemen Kash und Magma sowie Tätigkeiten im Bereich der Kryptographie.*
`http://www.staff.uni-oldenburg.de/florian.hess`

- **Prof. Dr. Gregor Kemper**, *50, Professor für algorithmische Algebra an der TU München. Arbeitsgebiete: Invariantentheorie, algorithmische kommutative Algebra, Computeralgebra.*
`http://www-m11.ma.tum.de/~kemper`

- **Prof. Dr. Jürgen Klüners**, *43, Professor für Computeralgebra und Zahlentheorie an der Universität Paderborn. Arbeitsgebiete: Computeralgebra, Galois- und Zahlentheorie. Mitentwickler der Computeralgebrasysteme Kant und Magma sowie einer Datenbank für Zahlkörper. Mitglied der Koordinatorengruppe des DFG-Schwerpunktprogramms 1489.*
  `http://www2.math.uni-paderborn.de/people/juergen-klueners`

- **Prof. Dr. Michael Stoll**, *49, seit 2008 Lehrstuhl für Computeralgebra an der Universität Bayreuth. Arbeitsgebiet: Theoretische und algorithmische Aspekte der arithmetischen Geometrie, insbesondere rationale Punkte.*
  `http://www.mathe2.uni-bayreuth.de/stoll`

- **Prof. Dr. Martin Kreuzer**, *51, Universitätsprofessor, Lehrstuhl für Symbolic Computation, Fakultät für Informatik und Mathematik, Universität Passau. Arbeitsgebiete: Computeralgebra, insbesondere Gröbnerbasen und Randbasen, industrielle Anwendungen der Computeralgebra, algebraische Kryptographie, algebraische Geometrie. Leiter des Entwicklerteams des Computeralgebrapakets ApCoCoA.*
  `http://staff.fim.uni-passau.de/~kreuzer`

- **OStR Jan Hendrik Müller**, *45, Schuldienst seit 1996, seit 2003 Lehrauftrag für Didaktik der Mathematik an der TU Dortmund (IEEM bei Prof. Dr. Hans-Wolfgang Henn). Schwerpunkte: Seit 6 Jahren im Bereich internationaler Comenius Bildungsprojekte tätig, Computereinsatz und freie Arbeitsformen im Mathematikunterricht. Fachdidaktische Veröffentlichungen seit 2003, seit 1998 in MINT-Initiativen aktiv, Unterrichtserfahrung mit CAS (TI-92, TI-Nspire, WXMaxima).*
  `http://www.mathebeimueller.de`

- **Prof. Dr. Eva Zerz**, *46, Professorin für Algebra am Lehrstuhl D für Mathematik der RWTH Aachen. Arbeitsgebiete: mathematische Kontrolltheorie, algebraische Systemtheorie, Netzwerktheorie, Anwendungen von computeralgebraischen Methoden, insbesondere Gröbnerbasen, in diesen Gebieten, z. B. Singular Control Library.*
  `http://www.math.rwth-aachen.de/~Eva.Zerz`

Die Wahlleitung für diese Wahl haben die Herren Mayr sowie Koepf übernommen, die als offizielle Vertreter der GI bzw. der DMV Mitglieder der Fachgruppenleitung sind.

Bitte kreuzen Sie auf Ihrem Stimmzettel bis zu neun Namen *an und senden ihn im verschlossenen Wahlumschlag zusammen mit der unterschriebenen „Versicherung zur Briefwahl“ im beigefügten Rücksendeumschlag bis zum*

Donnerstag, 19. Dezember 2013 (Eingang beim Wahlleiter)

an den Wahlleiter der Fachgruppe Computeralgebra, Prof. Dr. Ernst W. Mayr, Institut für Informatik, Technische Universität München, Boltzmannstr. 3, 85748 Garching, zurück. Bitte machen Sie von Ihrer Wahlmöglichkeit Gebrauch!

Die konstituierende Sitzung der neuen Fachgruppenleitung wird im Februar 2014 stattfinden. Wir hoffen, Sie mit dem vorliegenden Heft wieder gut zu informieren.

Florian Heß *Eva Zerz*

Symbolic Summation in Difference Fields and Its Application in Particle Physics

C. Schneider[1]
(Research Institute for Symbolic Computation (RISC))

Carsten.Schneider@risc.jku.at

The success story of symbolic summation started with Gosper's telescoping algorithm [10]: given a hypergeometric expression[2] $f(k)$, it finds – in case of existence – a hypergeometric expression $g(k)$ such that the telescoping equation

$$f(k) = g(k+1) - g(k) \tag{1}$$

holds. Then given $g(k)$, one can sum (1) over k and obtains, e.g., the identity

$$\sum_{k=1}^{a} f(k) = g(a+1) - g(1). \tag{2}$$

Moreover, the breakthrough concerning applications was lead by Zeilberger's extension of Gosper's algorithm to creative telescoping [24]: it enables one to derive recurrence relations for definite hypergeometric sums. In particular, solving such recurrences in terms of hypergeometric expressions [14] gave rise to the following toolbox: given a definite proper hypergeometric sum, one can decide algorithmically if it can be simplified in terms of a linear combination of hypergeometric expressions. For details on this machinery we refer to the pioneering book [15]; for a most recent point of view see [12]. In the last decades many further improvements and generalizations have been accomplished, like, e.g., summation over holonomic sequences [9] or over expressions represented in terms of difference fields.

In this introductory note we focus on the latter approach whose foundation was led by Karr's telescoping algorithm in $\Pi\Sigma^*$-fields [11]. There one can treat indefinite nested product-sums in a very elegant way; for details see, e.g.,[22].

Definition. Let $f(k)$ be an expression that evaluates at non-negative integers (from a certain point on) to elements of a field $\mathbb{K}$ containing as subfield the rational numbers $\mathbb{Q}$. Then $f(k)$ is called indefinite nested product-sum expression w.r.t. k (over $\mathbb{K}$) if it is composed by elements from the rational function field $\mathbb{K}(k)$, the four operations $(+, -, \cdot, /)$, and indefinite sums and products of the type $\sum_{i=l}^{k} h(i)$ or $\prod_{i=l}^{k} h(i)$ where $l \in \mathbb{N}$ and where $h(i)$ is an indefinite nested product-sum expression w.r.t. i over $\mathbb{K}$ which is free of k.

This class covers besides (q–)hypergeometric expressions, e.g., generalized harmonic sums [13, 3]

$$S_{a_1,\dots,a_r}(x_1,\dots,x_r;k) = \sum_{i_1=1}^{k} \frac{x_1^{i_1}}{i_1^{a_1}} \sum_{i_2=1}^{i_1} \frac{x_2^{i_2}}{i_2^{a_2}} \cdots \sum_{i_r=1}^{i_{r-1}} \frac{x_r^{i_r}}{i_r^{a_r}} \tag{3}$$

with $a_i \in \mathbb{N} \setminus \{0\}$ and $x_i \in \mathbb{K} \setminus \{0\}$; for $x_i \in \{1, -1\}$ these sums specialize to harmonic sums [8, 23] also denoted by $S_{x_1 a_1,\dots,x_r a_r}(n) (= S_{a_1,\dots,a_r}(x_1,\dots,x_r;n))$. More generally, e.g., (q–)hypergeometric expressions might occur inside of such sums.

In the following we will get a glimpse of how the difference field machinery works and elaborate on the crucial summation techniques based on this approach. Using this toolkit within the Mathematica summation package `Sigma` [17], we will illustrate how one can discover and prove multi-sum identities. In particular, we will report on challenging computations from particle physics that are currently performed.

The basic mechanisms in difference fields

We demonstrate the difference field approach by the following indefinite summation problem: simplify

$$\sum_{k=1}^{n} \underbrace{k\, S_1(k)}_{=:f(k)}$$

[1]Supported by the Austrian Science Fund (FWF) grants P20347-N18 and SFB F50 (F5009-N15) and by the EU Network LHCPhenoNet PITN-GA-2010-264564.

[2]A sequence $f(k)$ (resp. an expression that evaluates to a sequence) is called hypergeometric if there is a rational function $r(x)$ and a $\lambda \in \mathbb{N}$ such that $r(k) = \frac{f(k+1)}{f(k)}$ for all $k \in \mathbb{N}$ with $k \geq \lambda$.

where $S_1(k) = \sum_{i=1}^{k} \frac{1}{i}$ denotes the k-th harmonic numbers. To accomplish this task, we hunt for a solution $g(k)$ in terms of $S_1(k)$ such that (1) holds. In terms of the shift operator $\mathscr{S}_k$ w.r.t. k equation (1) reads as follows:

$$\mathscr{S}_k g(k) - g(k) = f(k). \tag{4}$$

First, the summation objects will be represented step by step in a field $\mathbb{F}$, and along with that the shift operator $\mathscr{S}_k$ is rephrased by a field automorphism $\sigma : \mathbb{F} \to \mathbb{F}$.

i) We start with the rational numbers $\mathbb{Q}$ and define the (only possible) field automorphism $\sigma : \mathbb{Q} \to \mathbb{Q}$ with $\sigma(q) = q$ for all $q \in \mathbb{Q}$.

ii) Next, we need to model k with the shift behavior $\mathscr{S}_k k = k + 1$: Since all elements in $\mathbb{Q}$ are constant (i.e., $\sigma(q) = q$ for all $q \in \mathbb{Q}$), we adjoin a variable t_1 to $\mathbb{Q}$ and extend the automorphism to $\sigma : \mathbb{Q}(t_1) \to \mathbb{Q}(t_1)$ with $\sigma(t_1) = t_1 + 1$.

iii) Finally, we represent $S_1(k)$ with the shift behavior $\mathscr{S}_k S_1(k) = S_1(k) + \frac{1}{k+1}$: First, we try to find such an element in $\mathbb{Q}(t_1)$, i.e., we look for a $\gamma \in \mathbb{Q}(t_1)$ such that $\sigma(\gamma) = \gamma + \frac{1}{t_1+1}$ or equivalently $\sigma(\gamma) - \gamma = \frac{1}{t_1+1}$ holds. Using our telescoping algorithms from [21] (or, e.g., Gosper's algorithm) proves that such an element γ does not exist. Therefore we adjoin the variable t_2 to $\mathbb{Q}(t_1)$ and extend the automorphism $\sigma : \mathbb{Q}(t_1)(t_2) \to \mathbb{Q}(t_1)(t_2)$ subject to $\sigma(t_2) = t_2 + \frac{1}{t_1+1}$.

In short, we have constructed a difference field $(\mathbb{F}, \sigma)$ with the rational function field $\mathbb{F} = \mathbb{Q}(t_1)(t_2)$ together with a field automorphism σ. In this setting, $f(k)$ is given by $\phi = t_1^2 t_2$ and one seeks $\gamma \in \mathbb{Q}(t_1)(t_2)$ such that

$$\sigma(\gamma) - \gamma = \phi.$$

We calculate $\gamma = \frac{1}{4}(t_1 - 1)t_1\big(2t_2 - 1\big)$ with our algorithm and obtain as a consequence the solution

$$g(k) = \frac{1}{4}(k-1)k\big(2S_1(k) - 1\big)$$

for (4) and thus for (1). Therefore we get (2) which produces the simplification

$$\sum_{k=1}^{n} kS_1(k) = \frac{1}{4}\big(2n(n+1)S_1(n) - (n-1)n\big).$$

Summarizing, we applied the following strategy; for more details we refer to [22].

(1) Represent the involved indefinite nested product-sum expressions, whose evaluation leads to elements from a field $\mathbb{K}$, in a difference field $(\mathbb{F}, \sigma)$. Here $\mathbb{F} = \mathbb{K}(t_1)\dots(t_e)$ is a rational function field where the generators t_i represent the sums and products. Moreover, the shift behavior of the objects is described by a field automorphism $\sigma : \mathbb{F} \to \mathbb{F}$ where for $1 \le i \le e$ either the sum relation $\sigma(t_i) = t_i + a_i$ or the product relation $\sigma(t_i) = a_i\, t_i$ with $0 \neq a_i \in \mathbb{F}_{i-1} := \mathbb{K}(t_1)\dots(t_{i-1})$ hold. As indicated in the example above, it is crucial that a new variable t_i with $\sigma(t_i) = t_i + a_i$ (similar for products) is only adjoined to $\mathbb{F}_{i-1}$ if there is no $\gamma \in \mathbb{F}_{i-1}$ with $\sigma(\gamma) = \gamma + a_i$. Exactly this problem can be decided constructively by Karr's algorithms [11]; for improved algorithms see [18, 21] and references therein.

(2) Solve the underlying summation problem (e.g., again telescoping, but also parameterized telescoping and recurrence solving given below) in this setting.

(3) Reformulate the solution to an expression in terms of indefinite nested product-sum expressions that yields a solution of the given summation problem.

Exactly this construction produces difference fields $(\mathbb{K}(t_1)\dots(t_e), \sigma)$ whose constants remain unchanged, i.e., $\{c \in \mathbb{K}(t_1)\dots(t_e) | \sigma(c) = c\} = \mathbb{K}$; see [11]. Difference fields with this property are also called $\Pi\Sigma^*$-fields.

In conclusion, all the summation paradigms presented in the next section rely on this mechanism: Reformulate the given problem in a $\Pi\Sigma^*$-field, solve it there and formulate the result back such that it is a solution of the input problem.

The summation paradigms of the package Sigma

In the following we will give an overview of `Sigma`'s symbolic summation toolbox that is based on the difference field approach introduced above.

A) Simplification of indefinite nested product-sum expressions

Whenever `Sigma` deals with indefinite nested product-sum expressions, in particular when it outputs such expressions, the following problem is solved implicitly.

Problem EAR: Elimination of algebraic relations. *Given* an indefinite nested product-sum expression $f(k)$. *Find* an indefinite nested product-sum expression $F(k)$ and $\lambda \in \mathbb{N}$ such that $f(k) = F(k)$ for all $k \ge \lambda$ and such that the occurring sums are algebraically independent.

As worked out above, the following mechanism is applied: $f(k)$ is rephrased in a suitable $\Pi\Sigma^*$-field $(\mathbb{K}(t_1)\dots(t_e), \sigma)$; this is accomplished by solving iteratively the telescoping problem. Here the sums (similarly the products) are represented by the variables t_i. Finally, reinterpreting the t_i as sums produces an alternative expression $F(k)$ of $f(k)$ where the occurring sums are algebraically independent; for details see [19, 22].

In particular, using refined telescoping algorithms, the sums can be given with certain optimality criteria. E.g., the found expression $F(k)$ can be given with minimal nesting depth and for any occurring sum in $F(k)$ there is no other indefinite nested sum representation with lower nesting depth; for details and further simplifications see [18, 20] and references therein.

E.g., after loading the `Sigma` package into Mathematica, such simplifications can be accomplished as follows.

```
In[1]:= <<Sigma.m
    Sigma - A summation package by Carsten Schneider© RISC
In[2]:= SigmaReduce[2^n Sum[1/i, {i,1,n}] Sum[1/2^j, {j,1,i}] Sum[2^k/k, {k,1,j}] Sum[1/2^l, {l,1,k}], n]
Out[2]= 2^n((Sum[1/i, {i,1,n}])^2 + (Sum[1/2^i/i, {i,1,n}]) Sum[1/i, {i,1,n}] +
      Sum[1/i^2, {i,1,n}] + Sum[1/(2^i i^2), {i,1,n}] - 3 Sum[(Sum[1/2^j/j, {j,1,i}])/i, {i,1,n}] - Sum[(Sum[2^j/j, {j,1,i}]/2^i)/i, {i,1,n}])
```

Here the product $2^n = \prod_{i=1}^{n} 2$ and the arising sums are algebraically independent and the sums have minimal nesting depth. In particular, the expression cannot be written in terms of indefinite nested sums with lower nesting depth.

B) Definite summation

Definite sums over indefinite nested product sums, like

$$A(n) = \sum_{k=0}^{n} \binom{n}{k} S_1(k)^2 = \sum_{k=0}^{n} \Big(\prod_{i=1}^{k} \frac{n+1-i}{i}\Big)\Big(\sum_{i=1}^{k} \frac{1}{i}\Big)^2, \quad (5)$$

can be handled by the following summation paradigms (see Fig. 1).

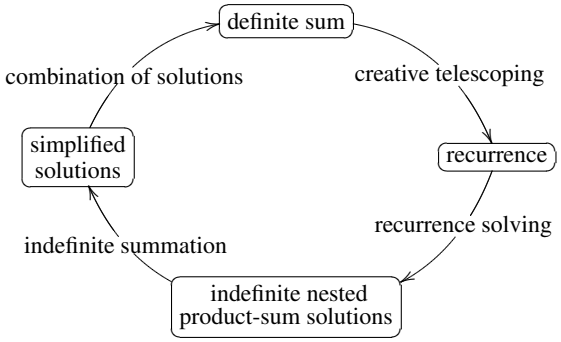


Figure 1: *Sigma's summation spiral*

First, there is the following tool in the setting of $\Pi\Sigma^*$-fields to obtain recurrences [21].

Problem PT: Parameterized Telescoping.
Given indefinite nested product-sum expressions $f_0(k), \dots, f_\delta(k)$. *Find* constants $a_0, \dots, a_\delta$, not all 0 and all free of k, and *find* an indefinite nested product-sum expression $g(k)$ being not more complicated than the $f_i(k)$ such that

$$a_0\, f_0(k) + a_1\, f_1(k) + \dots + a_\delta\, f_\delta(k) = g(k+1) - g(k). \quad (6)$$

For simplicity suppose that the found relation (6) holds for $0 \le k \le a$. Then by telescoping one gets, e.g., the sum relation

$$a_0 \sum_{k=0}^{a} f_0(k) + a_1 \sum_{k=0}^{a} f_1(k) + \dots + a_\delta \sum_{k=0}^{a} f_\delta(k) = g(a+1) - g(0) \quad (7)$$

where the right hand side is simpler than the sums of the left hand side.

Specializing to $f_i(k) := f(n+i,k)$ for a bivariate expression yields the creative telescoping paradigm. E.g., take $f_i(k) = \binom{n+i}{k} S_1(k)^2 = \prod_{j=1}^{i} \frac{n+j}{n-k+j} \binom{n}{k} S_1(k)^2$. Then we loop over $\delta = 0, 1, 2, \dots$ and try to solve the corresponding parameterized telescoping problem. Finally, at $\delta = 4$ we obtain a solution for (6) and we can deduce (7) which is a recurrence for $A'(n) = \sum_{k=0}^{a} \binom{n}{k} S_1(k)^2$. To this end, specializing $a = n$ and taking care of extra terms yield a recurrence of the form

$$a_0(n)A(n) + a_1(n)A(n+1) + \dots + a_\delta(n)A(n+\delta) = h(n) \quad (8)$$

for our sum (5). All these steps can be carried out for $A(n) =$ `SUM[n]` with the following function call:

```
In[3]:= rec = GenerateRecurrence[Sum[Binomial[n,k] S1[k]^2, {k,1,n}], n][[1]]
Out[3]= 8(n+1)(n+3)SUM[n] - 4(5n^2+25n+29)SUM[n+1] +
      2(3n+8)(3n+10)SUM[n+2] -
      (7n^2+49n+86)SUM[n+3] + (n+4)^2 SUM[n+4] == 1
```

Next, we can apply the following recurrence solver [16, 4].

Problem RS: Recurrence Solving.
Given a recurrence of the form (8) where the coefficients $a_i(n)$ and $h(n)$ are given in terms of indefinite nested product-sum expressions. *Find* all solutions that are expressible in terms of indefinite nested product sum expressions.

The corresponding sequences obtained by the evaluation of the found expressions are also called d'Alembertian solutions [5]. E.g., for the recurrence Out[3] we obtain with $c_1, c_2, c_3, c_4 \in \mathbb{Q}$ the general solution

$$2^n \Bigg[c_1 + c_2 \sum_{i=1}^{n} \frac{1}{i} + c_3 \sum_{i=1}^{n} \frac{1}{i} \sum_{j=1}^{i} \frac{1}{2^j} + c_4 \sum_{i=1}^{n} \frac{1}{i} \sum_{j=1}^{i} \frac{1}{2^j} \sum_{k=1}^{j} \frac{2^k}{k} + \sum_{i=1}^{n} \frac{1}{i} \sum_{j=1}^{i} \frac{1}{2^j} \sum_{k=1}^{j} \frac{2^k}{k} \sum_{l=1}^{k} \frac{1}{2^l}\Bigg].$$

By construction the solutions are highly nested: e.g., the depth of the particular solution equals usually the recurrence order plus the nesting depth of the inhomogeneous part of the recurrence. It is therefore a crucial (and often the most challenging) task to simplify these solutions further; for the simplification of the particular solution see Out[2] from above. With `Sigma` the recurrence `rec` given in Out[3] is solved (see Problem RS) and the found solutions are simplified by the following function call:

```
In[4]:= SolveRecurrence[rec, SUM[n]]
```

Out[4]= $\{\{0, 2^n\}, \{0, 2^n \sum_{i=1}^{n} \frac{1}{i}\}, \{0, 2^n (2 \sum_{i=1}^{n} \frac{1}{i} - 2 \sum_{i=1}^{n} \frac{1}{2^i i})\},$
$\{0, 2^n ((\sum_{i=1}^{n} \frac{1}{i})^2 + \sum_{i=1}^{n} \frac{1}{i^2} - \sum_{i=1}^{n} \frac{1}{2^i i} \sum_{j=1}^{i} \frac{2^j}{j})\},$
$\{1, 2^n ((\sum_{i=1}^{n} \frac{1}{i})^2 + \sum_{i=1}^{n} \frac{1}{i^2} + (\sum_{i=1}^{n} \frac{1}{i}) \sum_{i=1}^{n} \frac{1}{2^i i}$
$+ \sum_{i=1}^{n} \frac{1}{2^i i^2} - \sum_{i=1}^{n} \frac{1}{2^i i} \sum_{j=1}^{i} \frac{2^j}{j} - 3 \sum_{i=1}^{n} \frac{1}{i} \sum_{j=1}^{i} \frac{1}{2^j j})\}\}$

Finally, we take the linear combination of the homogeneous solutions (the entries with a 0) plus the particular solution (the entry with a 1) such that it agrees with $A(n)$ for $n = 1, 2, 3, 4$. In this instance, exactly the particular solution is doing the job. In terms of harmonic sums and their generalized versions this reads as follows:

$$\begin{aligned} A(n) &= 2^n \big[S_1(n)^2 + S_2(n) + S_1(n) S_1\big(\tfrac{1}{2}; n\big) \\ &\quad + S_2\big(\tfrac{1}{2}; n\big) - S_{1,1}\big(\tfrac{1}{2}, 2; n\big) \\ &\quad - 3 S_{1,1}\big(1, \tfrac{1}{2}; n\big)\big)\big]. \end{aligned} \tag{9}$$

Since both sides of (9) are a solution of the recurrence Out[3] and they agree for the first four initial values, the identity (9) holds for all $n \in \mathbb{N}$.

In summary, using the summation paradigms given in Fig. 1, we computed for the definite sum (5) a closed form (9) in terms of generalized harmonic sums.

Examples from Particle Physics

To complete this presentation, I give an account of the most challenging multi-sum expressions that I have faced so far. In a long term cooperation with the German Electron–Synchrotron (DESY, Zeuthen) 3-loop massive Feynman integrals from QCD (quantum chromodynamics) are reformulated in terms of definite multi-sums [7]. Then the crucial task is the simplification for further processing. On the one hand very huge expressions consisting of several 100000 definite multi-sums up to nesting depth 4 arose which needed to be simplified [2, 6]. On the other hand definite multi-sums up to nesting depth 7 from [1] occurred, like, e.g., in Fig. 2. Exactly here (as for all the other multi-sums in this context) the presented toolbox has been exploited. The inner sum (i) in Fig. 2 is a definite sum over an indefinite nested product-sum expression. Hence the definite summation technologies (Fig. 1) are activated. We calculate a recurrence in s (the summation index of the next sum (ii)), we succeed in obtaining the general solution in terms of indefinite nested product-expressions w.r.t. s, and thus we are able to combine the solutions to derive an alternative representation of sum (i) in terms of indefinite nested product-sums. Note that within these sums hypergeometric expressions occur. Given this form, the sum (ii) in Fig. 2 fits again to our summation paradigm. Continuing this process, we transform the multi-sum in Fig. 2 from inside (sum (i)) to outside (sum (vi)) to an indefinite nested product-sum expression. In the final output (filling several pages) all the sums with hypergeometric expressions collapse and the result can be written purely in terms of 48 generalized harmonic sums being algebraically independent (see Problem EAR). Here the most complicated sums are

$$S_{1,1,1,1}(1,1,2,1;n), S_{1,1,1,1}\big(2, \tfrac{1}{2}, 1, 1; n\big),$$
$$S_{1,1,1,1}\big(2, 1, \tfrac{1}{2}, 1; n\big), S_{1,1,1,1}\big(2, 1, 1, \tfrac{1}{2}; n\big).$$

We highlight that all these calculations are done automatically. E.g., after loading besides `Sigma` the newly developed package

In[5]:= `<< EvaluateMultiSums.m`

`EvaluateMultiSums by Carsten Schneider – © RISC`

our sum (5) can be simplified to indefinite nested sums by executing

In[6]:= $\mathbf{EvaluateMultiSum}[\sum_{k=1}^{n} \mathbf{S_1(k)^2} \binom{n}{k}, \{\}, \{n\}]$

Out[6]= $2^n ((\sum_{i=1}^{n} \frac{1}{i})^2 + \sum_{i=1}^{n} \frac{1}{i^2} + (\sum_{i=1}^{n} \frac{1}{i}) \sum_{i=1}^{n} \frac{1}{2^i i} + \sum_{i=1}^{n} \frac{1}{2^i i^2} -$
$\sum_{i=1}^{n} \frac{1}{2^i i} \sum_{j=1}^{i} \frac{2^j}{j} - 3 \sum_{i=1}^{n} \frac{1}{i} \sum_{j=1}^{i} \frac{1}{2^j j})$

Using in addition J. Ablinger's HarmonicSums package [3] enables one to rewrite the derived sums in terms of generalized harmonic sums as given in (9). Similarly our 6-fold sum can be transformed to generalized harmonic sums in about 1 day by just pressing the button.

References

[1] J. Ablinger, J. Blümlein, A. Hasselhuhn, S. Klein, C. Schneider, and F. Wissbrock. Massive 3-loop ladder diagrams for quarkonic local operator matrix elements. *Nucl. Phys. B*, 864:52–84, 2012. arXiv:1206.2252v1 [hep-ph].

[2] J. Ablinger, J. Blümlein, S. Klein, C. Schneider, and F. Wissbrock. The $O(\alpha_s^3)$ massive operator matrix elements of $O(n_f)$ for the structure function $F_2(x, Q^2)$ and transversity. *Nucl. Phys. B*, 844:26–54, 2011. arXiv:1008.3347 [hep-ph].

[3] J. Ablinger, J. Blümlein, and C. Schneider. Analytic and algorithmic aspects of generalized harmonic sums and polylogarithms. *J. Math. Phys. 54, 082301 (2013);*, 54(082301):1–74, 2013. arXiv:1302.0378 [math-ph].

[4] S. A. Abramov, M. Bronstein, M. Petkovšek, and C. Schneider. *In preparation*, 2013.

[5] S. A. Abramov and M. Petkovšek. D'Alembertian solutions of linear differential and difference equations. In J. von zur Gathen, editor, *Proc. ISSAC'94*, pages 169–174. ACM Press, 1994.

[6] J. Blümlein, A. Hasselhuhn, S. Klein, and C. Schneider. The $O(\alpha_s^3 n_f T_F^2 C_{A,F})$ contributions to the gluonic massive operator matrix elements. *Nuclear Physics B*, 866:196–211, 2013.

[7] J. Blümlein, S. Klein, C. Schneider, and F. Stan. A symbolic summation approach to Feynman integral calculus. *J. Symbolic Comput.*, 47:1267–1289, 2012.

[8] J. Blümlein and S. Kurth. Harmonic sums and Mellin transforms up to two-loop order. *Phys. Rev.*, D60, 1999.

[9] F. Chyzak. An extension of Zeilberger's fast algorithm to general holonomic functions. *Discrete Math.*, 217:115–134, 2000.

[10] R. W. Gosper. Decision procedures for indefinite hypergeometric summation. *Proc. Nat. Acad. Sci. U.S.A.*, 75:40–42, 1978.

[11] M. Karr. Summation in finite terms. *J. ACM*, 28:305–350, 1981.

[12] M. Kauers and P. Paule. *The concrete tetrahedron*. Texts and Monographs in Symbolic Computation. Springer, 2011.

[13] S. O. Moch, P. Uwer, and S. Weinzierl. Nested sums, expansion of transcendental functions, and multiscale multiloop integrals. *J. Math. Phys.*, 6:3363–3386, 2002.

[14] M. Petkovšek. Hypergeometric solutions of linear recurrences with polynomial coefficients. *J. Symbolic Comput.*, 14(2-3):243–264, 1992.

[15] M. Petkovšek, H. S. Wilf, and D. Zeilberger. $A = B$. A. K. Peters, Wellesley, MA, 1996.

[16] C. Schneider. Solving parameterized linear difference equations in terms of indefinite nested sums and products. *J. Differ. Equations Appl.*, 11(9):799–821, 2005.

[17] C. Schneider. Symbolic summation assists combinatorics. *Sém. Lothar. Combin.*, 56:1–36, 2007. Article B56b.

[18] C. Schneider. A refined difference field theory for symbolic summation. *J. Symbolic Comput.*, 43(9):611–644, 2008. [arXiv:0808.2543v1].

[19] C. Schneider. Parameterized telescoping proves algebraic independence of sums. *Ann. Comb.*, 14(4):533–552, 2010. [arXiv:0808.2596].

[20] C. Schneider. A symbolic summation approach to find optimal nested sum representations. In A. Carey, D. Ellwood, S. Paycha, and S. Rosenberg, editors, *Motives, Quantum Field Theory, and Pseudodifferential Operators*, volume 12 of *Clay Mathematics Proceedings*, pages 285–308. Amer. Math. Soc, 2010. arXiv:0808.2543.

[21] C. Schneider. Fast algorithms for refined parameterized telescoping in difference fields. In *arXiv:1307.7887 [cs.SC]*. 2013.

[22] C. Schneider. Simplifying multiple sums in difference fields. In J. Blümlein and C. Schneider, editors, *Computer Algebra in Quantum Field Theory: Integration, Summation and Special Functions*, Texts and Monographs in Symbolic Computation, pages 325–360. Springer, 2013.

[23] J. A. M. Vermaseren. Harmonic sums, Mellin transforms and integrals. *Int. J. Mod. Phys.*, A14:2037–2976, 1999.

[24] D. Zeilberger. The method of creative telescoping. *J. Symbolic Comput.*, 11:195–204, 1991.

$$\overbrace{\sum_{j=0}^{n-3}}^{(vi)}\overbrace{\sum_{k=0}^{j}}^{(v)}\overbrace{\sum_{l=0}^{k}}^{(iv)}\overbrace{\sum_{q=0}^{-j+n-3}}^{(iii)}\overbrace{\sum_{s=1}^{-l+n-q-3}}^{(ii)}\overbrace{\sum_{r=0}^{-l+n-q-s-3}}^{(i)}(-1)^{k-j-l+n-q-3}\binom{j+1}{k+1}\binom{k}{l}\times$$
$$\times\frac{\binom{n-1}{j+2}\binom{-j+n-3}{q}\binom{-l+n-q-3}{s}\binom{-l+n-q-s-3}{r}r!(-l+n-q-r-s-3)!(s-1)!}{(-l+n-q-2)!(-j+n-1)(n-q-r-s-2)(q+s+1)}$$
$$\times\Big[4S_1(-j+n-1)-4S_1(-j+n-2)-2S_1(k)-(S_1(-l+n-q-2)$$
$$+S_1(-l+n-q-r-s-3)-2S_1(r+s))+2S_1(s-1)-2S_1(r+s)\Big].$$

Figure 2: *Definite multi-sums.*

Neue Algorithmen für das Diskreter-Logarithmus-Problem in kleiner Charakteristik

Jens Zumbrägel
(University College Dublin, Irland)

jens.zumbragel@ucd.ie

Zahlreiche Kryptosysteme beruhen auf der Schwierigkeit, das Diskreter-Logarithmus-Problem (DLP) in einem endlichen Körper zu lösen, und so ist die Untersuchung von DLP-Algorithmen ein seit Jahrzehnten bedeutender Forschungsgegenstand. In jüngster Zeit sind sehr effiziente Angriffe auf das DLP, insbesondere in Erweiterungskörpern kleiner Charakteristik, entwickelt worden. Diese basieren im Wesentlichen auf speziellen Eigenschaften von Polynomen der Form $X^{q+1}+aX^q+bX+c \in \mathbb{F}_{q^k}[X]$. Die neuen Algorithmen resultieren sowohl auf theoretischer Seite in kürzeren asymptotischen Laufzeiten, als auch bei praktischen Implementierungen in beträchtlichen neuen DLP-Rekorden.

Diskrete Logarithmen

Das Diskreter-Logarithmus-Problem (DLP) in einem endlichen Körper $\mathbb{F}_Q$ mit Q Elementen besteht darin, zu gegebenen Erzeuger g von $\mathbb{F}_Q^*$ und einem Element $h \in \mathbb{F}_Q^*$ diejenige Zahl $x \in \{0, 1, \ldots, Q-2\}$ zu finden mit $h = g^x$. Diese Zahl x wird *diskreter Logarithmus* $\log_g h$ von h zur Basis g genannt.

In diesem Artikel betrachten wir den Fall, dass die Charakteristik p von $\mathbb{F}_Q$ klein ist, so dass $p = O(\log Q)$ gilt. Für kryptographische Paarungen sind insbesondere die Fälle $p = 2$ und $p = 3$ interessant.

Der effizienteste Algorithmus für den allgemeinen Fall ist Coppersmiths Methode von 1984 [3], mit einer subexponentiellen Laufzeit von $L_Q(1/3, (32/9)^{1/3} \approx 1.526)$, wobei wie üblich mit $L_Q(\alpha, c)$ die Laufzeit $\exp\big((c+o(1))(\log Q)^\alpha(\log\log Q)^{1-\alpha}\big)$ bezeichnet werde.

Wenn $\mathbb{F}_Q = \mathbb{F}_{q^n}$ ein Erweiterungskörper von $\mathbb{F}_q$ vom Grad n ist, so dass n und q geeignet balanciert sind (nämlich $q = L_Q(1/3, (1/9)^{1/3})$), dann wird eine verbesserte Laufzeit von $L_Q(1/3, 3^{1/3} \approx 1.442)$ durch den Joux-Lercier-Algorithmus von 2006 [7], eine Variante des Funktionenkörpersiebes, erreicht.

Dieser Algorithmus konnte Ende 2012 von Joux verbessert werden, mit einer effizienteren $L_Q(1/3)$-Laufzeit. Seitdem sind weitere bedeutende Fortschritte bei den Algorithmen für das DLP in kleiner Charakteristik zu verzeichnen, welche wir nun schildern.

Index-Calculus-Methoden

Die bisher genannten und die neuen Algorithmen gehören zur Familie der Index-Calculus-Methoden, die durch folgende drei Phasen beschrieben werden können.

1. Wähle eine Faktorbasis; finde multiplikative Relationen zwischen Faktorbasiselementen, d.h. additive (lineare) Relationen zwischen deren Logarithmen.

2. Löse das entstehende lineare Gleichungssystem mit den diskreten Logarithmen als Variablen.

3. Finde eine Darstellung des Zielelements h als Produkt von Faktorbasiselementen.

Da nach Schritt 2 die Logarithmen aller Faktorbasiselemente bekannt sind, ergibt aus Schritt 3 sofort der Logarithmus von h.

Im konkreten Fall des endlichen Körpers $\mathbb{F}_{q^n}$ betrachten wir die übliche Darstellung

$$\mathbb{F}_{q^n} = \mathbb{F}_q[X]/\langle p\rangle = \mathbb{F}_q(x)\,,$$

wobei $p \in \mathbb{F}_q[X]$ ein irreduzibles Polynom vom Grad n und x eine Nullstelle von p in $\mathbb{F}_{q^n}$ ist; damit werden die Elemente von $\mathbb{F}_{q^n}$ durch Polynome über $\mathbb{F}_q$ vom Grad kleiner als n repräsentiert.

Sei $\mathbb{F}_q[X]^{\leq B}$ die Menge aller Polynome über $\mathbb{F}_q$ vom Grad höchstens B. Ein Polynom $f \in \mathbb{F}_q[X]$ heißt *B-glatt*, falls f Produkt von Polynomen in $\mathbb{F}_q[X]^{\leq B}$ ist.

Der Joux-Lercier-Algorithmus
Bei dem Algorithmus von Joux und Lercier [7] wird das die Körpererweiterung $\mathbb{F}_{q^n}$ definierende Polynom $p \in \mathbb{F}_q[X]$ so gewählt, dass

$$p(X) \mid g_1(g_2(X)) - X$$

für geeignete Polynome $g_1, g_2 \in \mathbb{F}_q[X]$ vom Grad d_1, d_2 gilt. Mit $y := g_2(x) \in \mathbb{F}_{q^n}$ folgt dann $g_1(y) = x$. Damit haben wir $\mathbb{F}_{q^n} = \mathbb{F}_q(x) = \mathbb{F}_q(y)$, und jedes Element in $\mathbb{F}_{q^n}$ kann als Polynom, evaluiert in x oder in y, dargestellt werden. Man wählt als Faktorbasis

$$\mathcal{F} := \{f(x) \mid f \in \mathbb{F}_q[X]^{\leq B}\} \cup \{g(y) \mid g \in \mathbb{F}_q[X]^{\leq B}\}\,.$$

Multiplikative Relationen werden nun generiert, indem B-glatte Polynome $\ell, r \in \mathbb{F}_q[X]$ gefunden werden, so dass $\ell(x) = r(y)$ gilt; hierbei wird der Ansatz

$$\ell(x) = m(x, g_2(x)) = m(g_1(y), y) = r(y)$$

für geeignete bivariate Polynome $m \in \mathbb{F}_q[X, Y]$ verwendet. Speziell für Polynome der Form $m = XY + aY + bX + c$ ergibt sich die Gleichung

$$xg_2(x) + ag_2(x) + bx + c = g_1(y)y + ay + bg_1(y) + c\,,$$

wobei die entsprechenden Polynome der beiden Seiten den Grad $d_2 + 1$ bzw. $d_1 + 1$ haben.

Höhere Zerfallswahrscheinlichkeiten

Für ein zufälliges Polynom vom Grad N gilt für die Wahrscheinlichkeit P, dass dieses B-glatt ist, in etwa

$$-\log P \approx N/B\, \log(N/B)\,.$$

Insbesondere ist die Wahrscheinlichkeit, dass das Polynom in Linearfaktoren zerfällt, etwa $1/N!$. Mit dieser Heuristik ergibt sich bei optimaler Balance von Schritt 1 und Schritt 2 der Index-Calculus-Methode eine Laufzeit von $L_{q^n}(1/3, 3^{1/3})$, während der Schritt 3 eine kürzere Laufzeit benötigt.

Durch eine spezielle Wahl der Parameter kann jedoch erreicht werden, dass die auftretenden Polynome mit einer weitaus höheren Wahrscheinlichkeit zerfallen. Und zwar sei nun der Grundkörper selbst ein Erweiterungskörper $\mathbb{F}_{q^k}$, $k \geq 2$, und betrachte den Fall $g_2 = X^q$, also $y = x^q$. Dann lautet die obige Gleichung

$$x^{q+1} + ax^q + bx + c = g_1(y)y + ay + bg_1(y) + c\,.$$

Polynome von der Form $X^{q+1} + aX^q + bX + c$ wurden von Bluher [2] untersucht (siehe auch Helleseth und Kholosha [6]). Man kann zeigen, dass diese Polynome für zufällige $a, b, c \in \mathbb{F}_{q^k}$ mit einer Wahrscheinlichkeit von ca. $1/q^3$ zerfallen, was viel höher ist als $1/(q+1)!$. Hieraus resultiert sogar ein *polynomieller* Algorithmus für die Schritte 1 und 2 der Index-Calculus-Methode, wenn q und n geeignet balanciert werden [4].

Individuelle Logarithmen

Wegen dieser Fortschritte beim Berechnen der diskreten Logarithmen für die Faktorbasiselemente liegt der Fokus nun auf den Schritt 3, der Berechnung eines individuellen Logarithmus $\log_g h$. Hierbei wird üblicherweise das Prinzip des *Abstiegs* verwendet: Für jedes Körperelement $Q(y) \in \mathbb{F}_{q^n}$, repräsentiert durch ein Polynom $Q \in \mathbb{F}_{q^n}[X]$, kann eine Darstellung als Produkt von Elementen $f(x)$ und $g(y)$ mit kleinerem Grad $\deg f, \deg g \leq m < \deg Q$ gefunden werden. Mit diesen Faktoren wird der Algorithmus rekursiv fortgeführt, bis nur noch Faktorbasiselemente als Faktoren auftreten. Somit ist schließlich eine Darstellung des Zielelements als Produkt von Faktorbasiselementen gefunden.

Klassischer Weise betrachtet man dabei einen Vektorraum von Polynompaaren $(\ell, r) \in \mathbb{F}_{q^k}[X]$ von kleinem Grad auf beiden Seiten, so dass die Gleichung $\ell(x) = Q(y)r(y)$ gilt. Sind beide Polynome $\ell(X)$ und $r(X)$ m-glatt, so ist der Abstieg erfolgreich.

Allerdings können wir dabei (außer für $\deg Q = 2$) keine erhöhten Zerfallswahrscheinlichkeit ausnutzen. Somit ergibt sich trotz der verbesserten Phasen 1 und 2 bei optimaler Balance mit Phase 3 eine $L(1/3)$-Gesamtlaufzeit; jedoch kann durch einen effizienteren Abstieg für $\deg Q = 2$ die Laufzeit-Konstante auf bis zu $c = (4/9)^{1/3} \approx 0.763$ reduziert werden [4].

Abstieg mit Gröbnerbasen

Die klassische Abstiegsmethode wird mit kleinerem Grad zunehmend schwieriger. Eine effizientere Strategie, welche von entscheidender Bedeutung für Jouxs neuem Index-Calculus-Algorithmus [8] mit Laufzeit $L(1/4 + o(1))$ ist, basiert auf Gröbnerbasen. Wir beschreiben eine Variante dieser Strategie, welche auf in Linearfaktoren zerfallende Polynome der Form $F(X) := X^{q+1} + BX + B$ basiert [5].

Substituiert man X in F durch den Quotienten $f(X)/g(X)$, wobei $f, g \in \mathbb{F}_{q^k}[X]^{\leq m}$, so folgt, dass das Polynom

$$\tilde{F}_{f,g}(X) := f(X)^{q+1} + Bf(X)g(X)^q + Bg(x)^{q+1}$$

m-glatt ist. Wegen $x^q = y$ und $x = g_1(y)$ kann $\tilde{F}_{f,g}(x)$ außerdem als Polynom, evaluiert in y, mit kleinem Grad $(d_1 + 1)m$ dargestellt werden. Wir betrachten dann den Ansatz

$$\tilde{F}_{f,g}(x) = Q(y)r(y)\,,$$

wodurch ein Gleichungssystem in den $\mathbb{F}_{q^k}$-Koeffizienten der Polynome f, g, r gegeben ist. Betrachtet man diese Koeffizienten nun über eine $\mathbb{F}_q$-Basis, so ergibt sich daraus ein multivariates quadratisches Gleichungssystem über $\mathbb{F}_q$. Dieses Gleichungssystem kann effizient mit einem Gröbnerbasis-Algorithmus gelöst werden. Ist der Kofaktor $r(X)$ ebenfalls m-glatt, so ist der Abstieg für $Q(y)$ erfolgreich.

Abstieg mit linearer Algebra

Im Juni 2013 veröffentlichten Barbulescu, Gaudry, Joux und Thomé ein Manuskript, welches einen quasipolynomiellen Algorithmus für diskrete Logarithmen in kleiner Charakteristik beschreibt [1]. Dieses Resultat ist auf theoretischer Seite bahnbrechend, jedoch ist die praktische Bedeutung der neuen Methode bisher noch unklar.

Grundlegend für den neuen Algorithmus ist eine neue Abstiegsstrategie, welche auf lineare Algebra beruht, und hier nur knapp skizziert werden soll.

Wir betrachten den gleichen Ansatz wie beim Abstieg mit Gröbnerbasen und verwenden nun den Quotienten $f(X)/g(X) = (aQ(X) + b)/(cQ(X) + d)$. Es folgt, dass das Polynom $\tilde{F}(X) := (aQ(X) + b)^{q+1} + B(aQ(X) + b)(cQ(X) + d)^q + B(cQ(X) + d)^{q+1}$ in

Faktoren der Form $Q(X)+\lambda$ für $\lambda \in \mathbb{F}_{q^k}$, also in Translationen von $Q(X)$, zerfällt.

Die Darstellung von $\tilde{F}(x)$ als Polynom, evaluiert in y, hat Grad $(d_1+1)\deg Q$. Ist dieses Polynom m-glatt, so ergibt sich eine lineare Relation zwischen den Logarithmen von Elementen $g(y)$ vom Grad höchstens m und von Translationen von $Q(x)$. Werden genügend solche Relationen gefunden, so können die Logarithmen der Translationen von $Q(x)$ in Abhängigkeit der Logarithmen der Elemente $g(y)$ ausgedrückt werden.

Obwohl diese Methode praktisch recht aufwändig ist und eine riesige Zahl von Faktoren erzeugt, die rekursiv zu bearbeiten sind, so bleibt der Aufwand pro Abstieg (mit $m = \deg Q/2$) polynomiell in q, woraus die quasipolynomielle Gesamtlaufzeit resultiert.

Angriff auf das DLP in $\mathbb{F}_{2^{6120}}$

Schließlich zeigen wir an einem Beispiel, dass die neuen Algorithmen [4, 8] für das DLP auch auf praktischer Ebene in beachtlichen DLP-Rekorden resultieren. Wir konnten diskrete Logarithmen im endlichen Körper $\mathbb{F}_{2^{6120}}$ in nur ca. 750 Rechnerstunden berechnen [5].

Wir betrachten dabei $\mathbb{F}_{(q^k)^n}$ mit $q = 2^8$, $k = 3$, $n = q-1$, gegeben durch eine Kummer-Erweiterung

$$\mathbb{F}_{(q^k)^n} \cong \mathbb{F}_{q^k}[X]/\langle X^n + \gamma\rangle\,,$$

und haben $x = g_1(y) = y/\gamma$ und $y = g_2(x) = x^q$.

Als Faktorbasis (mit $B = 1$) benötigen wir nur die Elemente $\mathcal{F} = \{x+u \mid u \in \mathbb{F}_{q^3}\}$, denn es gilt $y+v = (x+v^{1/q})^q$; also gibt es $|\mathcal{F}| = q^3 = 2^{24}$ Elemente. Ein bedeutender Vorteil der Kummer-Erweiterung ist, dass der Automorphismus $\sigma : \alpha \mapsto \alpha^q$ die Faktorbasis erhält. Dadurch operiert die Gruppe $\langle\sigma\rangle$ der Ordnung $3n$ auf $\mathcal{F}$, wodurch die Anzahl der Variablen für das Lineare-Algebra-Problem auf lediglich 21 932 reduziert wird.

Für $a, b, c \in \mathbb{F}_{q^k}$ mit $(a^q+b)^{q+1} = (ab+c)^q$ zerfällt das Polynom $X^{q+1}+aX^q+bX+c$ über $\mathbb{F}_{q^3}$, daher wird mit Wahrscheinlichkeit $1/2$ eine Relation erzeugt durch

$$x^{q+1} + ax^q + bx + c = y^2/\gamma + (a+b/\gamma)y + c\,.$$

Das Erzeugen von Relationen ist äußerst effizient und benötigt 15 Sekunden. Das entstehende lineare Gleichungssystem mit 21 932 Variablen modulo einer 5121-bit-Zahl (dem Produkt der 35 größten Primfaktoren von $2^{6120}-1$) konnte dann in 60.5 Stunden gelöst werden.

Zur Berechnung eines individuellen Logarithmus konstruierten wir das Zielelement $h \in \mathbb{F}^*_{2^{6120}}$ aus den Nachkommastellen der Konstante π. Wir verwendeten für den Abstieg neben der klassischen Methode auch die Gröbnerbasis-Methode: Seien $f, g \in \mathbb{F}_{q^3}[X]$ Polynome vom Grad höchstens m. Da das Polynom $F(X) = X^{q+1} + X + 1$ über $\mathbb{F}_{q^3}$ zerfällt, ist das Element

$$\tilde{F}_{f,g}(x) := f(x)^{q+1} + f(x)g(x)^q + g(x)^{q+1}$$

als Polynom m-glatt. Weiterhin ist es quadratisch in den $\mathbb{F}_q$-Koeffizienten von f, g, und es kann unter Berücksichtigung der Identität $x^q = \gamma x$ durch ein Polynom vom Grad $2m$ repräsentiert werden. Ein Element $Q(x) \in \mathbb{F}_{(q^k)^n}$ vom Grad höchstens $2m$ wird daher durch den Ansatz

$$\tilde{F}_{f,g}(x) = Q(x)r(x)$$

in Elemente vom Grad höchstens m aufgelöst, und diese Gleichung kann mit einem Gröbnerbasis-Algorithmus über $\mathbb{F}_q$ effizient gelöst werden.

Für $\deg Q \in \{5,6\}$, $m = 3$, erhalten wir ein System mit 21 Variablen, und für $\deg Q \in \{3,4\}$, $m = 2$ ein System mit 15 Variablen; diese Systeme können mit einer empirischen Wahrscheinlichkeit von ca. $2/3$ direkt gelöst werden. Schließlich verwendeten wir für $\deg Q = 2$ eine ad-hoc-Methode, welche für den Grundkörper $\mathbb{F}_{2^{24}} = \mathbb{F}_{(2^8)^3} = \mathbb{F}_{(2^6)^4}$ besondes geeignet ist. Der gesamte Abstieg benötigte 689 Rechnerstunden.

Literatur

[1] R. Barbulescu, P. Gaudry, A. Joux und E. Thomé. A quasi-polynomial algorithm for discrete logarithm in finite fields of small characteristic. *Manuskript*, `eprint.iacr.org/2013/400`

[2] A. Bluher. On $x^{q+1} + ax + b$. *Finite Fields Appl.* 10:3, 2004, S. 285–305.

[3] D. Coppersmith. Fast evaluation of logarithms in fields of characteristic two. *IEEE Trans. Inform. Theory* 30:4, 1984, S. 587–594.

[4] F. Göloğlu, R. Granger, G. McGuire und J. Zumbrägel. On the Function Field Sieve and the Impact of Higher Splitting Probabilities. In: *CRYPTO 2013*, Springer LNCS 8043, 2013, S. 109–128.

[5] F. Göloğlu, R. Granger, G. McGuire und J. Zumbrägel. Solving a 6120-bit DLP on a Desktop Computer. Erscheint in: *Selected Areas in Cryptography–SAC 2013*, Springer LNCS, 2013.

[6] T. Helleseth, A. Kholosha. $x^{2^l+1}+x+a$ and related affine polynomials over $\mathrm{GF}(2^k)$. *Cryptogr. Commun.* 2:1, 2010, S. 85–109.

[7] A. Joux und R. Lercier. The Function Field Sieve in the Medium Prime Case. In: *EUROCRYPT 2006*, Springer LNCS 4004, 2006, S. 254–270.

[8] A. Joux. A new index calculus algorithm with complexity L(1/4+o(1)) in small characteristic. Erscheint in: *Selected Areas in Cryptography–SAC 2013*, Springer LNCS, 2013.

FindStat – the combinatorial statistics database

Chris Berg (Google)
Christian Stump (Freie Universität Berlin)

chrisjamesberg@gmail.com
christian.stump@fu-berlin.de

The FindStat project was initiated by the two authors in 2011 at the Laboratoire de combinatoire et d'informatique mathématique, Université du Québec à Montréal, Canada. It provides an online platform for mathematicians, and in particular for combinatorialists, to gather information about combinatorial statistics and their relations. In 2013, we gained three new developers, V. Pons, T. Scrimshaw, and J. Striker.

An example

Combinatorial statistics arise naturally all over mathematics. Before we give the definition of a combinatorial statistic, let us first describe an example. Suppose you were studying a collection of polynomials $\{f_\pi\}$ indexed by permutations, and the degree of the first few polynomials were as follows:

$$\deg(f_{12}) = 0 \quad \deg(f_{21}) = 1$$

$$\deg(f_{123}) = 0 \quad \deg(f_{132}) = 2 \quad \deg(f_{213}) = 1$$
$$\deg(f_{231}) = 2 \quad \deg(f_{312}) = 1 \quad \deg(f_{321}) = 3$$

$$\deg(f_{1234}) = 0 \quad \deg(f_{1243}) = 3 \quad \deg(f_{1324}) = 2$$

You would like to know what underlying rule determines the degree of the polynomial. The savvy combinatorialist will tell you that the degree of f_π appears to be the *major index* of π,

$$\mathrm{maj}(\pi) = \sum_{\substack{1 \le i < n \\ \pi_i > \pi_{i+1}}} i.$$

What's that? You say you aren't a savvy combinatorialist? Never heard of a Mahonian statistic and are now riddled with the shame and dismay of your peers? Relax! This is what FindStat was built for! Just point your browser to

`www.FindStat.org/StatisticFinder`

and simply input the degree of each of your polynomials. Wait a few seconds, and FindStat will tell you that you're looking at the major index of a permutation.

Online Databases

The OEIS

We know what you're thinking! "Wait a second, isn't this exactly what the OEIS does?" Indeed, the success of the *On-Line Encyclopedia of Integer Sequences* (OEIS) [2] was a primary motivation in the creation of FindStat. The OEIS database contains more than 200.000 integer sequences and has had an immense impact on the work of mathematicians all over the world. There were two main reasons for starting the FindStat project, the first one being more of an annoyance, the second being more profound:

- Combinatorial collections do not necessarily admit a canonical total ordering which would be needed to index a given combinatorial statistic in the OEIS. In the example above, we study permutations. While there are many ways to index permuations (e.g. lexicographically), there is no default. Even more annoying, some combinatorial objects, like graphs, have much less obvious total orderings.

- There are many natural relationships between various combinatorial collections. A statistic on one combinatorial collection can often be transformed to a meaningful statistic on an entirely different collection of objects. In fact, those following along online may have noticed that FindStat didn't exclusively return the major index of a permutation as a solution to

the above problem. As of August 23, 2013, searching the FindStat database for this data yields almost 20.000 database searches after following more than 1.000 combinatorial maps. The database results provide the major index[1], as expected, as well as connections to the number of inversions[2] and 8 other statistics in the database.

Fingerprint databases for theorems
As discussed by S.C. Billey and B.E. Tenner in [1], FindStat is an example of a

fingerprint database for theorems.

Roughly speaking, this means that FindStat is actually providing a canonical way of storing and recognizing theorems; when I search for one statistic and find another, FindStat is implying that these statistics are in fact equal or related through well-described connections. Moreover, similar to a single sequence from the OEIS, a statistic page often contains information regarding the contexts in which the statistic arises, and references containing further information.

Project outlines

A *combinatorial collection* $\mathcal{S}$ is considered to be a set with interesting combinatorial properties, a *combinatorial map* is a map between two combinatorial collections that has an explicit combinatorial description, and a *combinatorial statistic* on $\mathcal{S}$ is a map st that associates an integer to each element in $\mathcal{S}$, $st : \mathcal{S} \longrightarrow \mathbb{Z}$.

The main aims
The two main aims of the FindStat project are to

- provide an online platform to gather information about combinatorial collections and statistics and their relations. This includes
  - adding new combinatorial statistics to the FindStat database, and
  - filling the corresponding wiki with information about combinatorial collections and statistics.
- provide a web interface to
  - test if your data is a known combinatorial statistic in the database, or
  - test if your data can be obtained from known combinatorial statistics in the database by applying combinatorial maps.

Contributing to the database
It might have happened in the introductory example that the degrees of the polynomials were not found in the database. You should then think of adding your data through the web interface

`www.FindStat.org/NewStatistic`

so that the next person searching for this statistic will be aware of the context in which you found it.

We tried to make it as simple as possible to contribute. All you need for your contribution is to provide your data, the context in which you encountered the statistic, and, if possible, references. The most important part is to provide the correct data; we recommend to use your favorite computer algebra system (or to use the Sage cell[1] provided by the web interface) to produce the data for you. If you do not have an algorithm computing the data for you, you can alternatively provide a few values by hand. Maybe the next person to find your statistic in the database will be able to provide further values.

The FindStat wiki
In addition to the pure database of combinatorial statistics, the FindStat project serves as a platform to provide information about combinatorial statistics and maps. For example, information about permutation statistics and combinatorial maps for permutations can be found at

`www.FindStat.org/Permutations`

and its subsequent pages. Anyone can contribute to the wiki to improve the content and the presentation of the information.

The FindStat blog
The FindStat database can be used to exhibit previously unknown connections between statistics. We recently created the blog FindStatFacts

`www.FindStat.org/FindStatFacts`

to discuss statistics that FindStat shows to be connected through combinatorial maps and to ask the community to explain these connections.

Licence and technical information
Contributions to the FindStat project, i.e., to the database and to the wiki, are licensed under a *Creative Commons License*[2].

The main technologies used in this project are a *MoinMoin wiki engine* for serving the website [3], and a server running *Sage* in which the combinatorial maps are implemented and the actual computations are performed [4].

References

[1] Sara C. Billey and Bridget E. Tenner Fingerprint databases for theorems. *Notices of the AMS* **60**(8) (2013).

[2] OEIS Foundation Inc. The On-Line Encyclopedia of Integer Sequences. `oeis.org` (2011).

[3] The MoinMoin Wiki Engine. `moinmo.in` (2013).

[4] Sage Mathematics Software (Version 5.11). `www.sagemath.org` (2013).

[1] `www.FindStat.org/St000004` [2] `www.FindStat.org/St000018`
[1] `https://github.com/sagemath/sagecell` [2] Creative Commons Attribution-ShareAlike 3.0 Unported License.

Kubisch, quartisch und so weiter

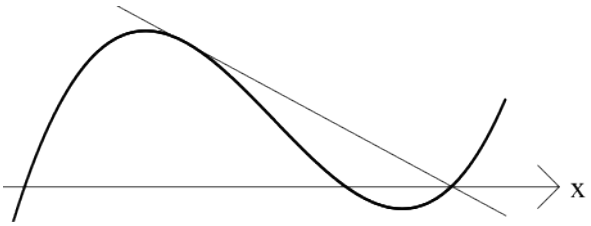
J. Meyer
(Hameln)

J.M.Meyer@t-online.de

Abstract: Eine bekannte Aufgabe zu kubischen Funktionen wird in verschiedenen Hinsichten auf Funktionen höheren Grades verallgemeinert.

Die folgende Aufgabe (nach Henn 2000; S. 31 f.) ist weithin bekannt:

Gegeben sei eine kubische Funktion mit drei reellen Nullstellen a, b und c. Die Tangente an der Stelle $(a+b)/2$ hat c als Nullstelle:



Wie lässt sich dieser – mit einem CAS einfach und schnell zu begründende – Sachverhalt auf Funktionen höheren Grades verallgemeinern? Beim Entdecken hilft GeoGebra und beim Begründen ein CAS.

1. Eine *erste Verallgemeinerung* geht aus vom umgekehrten Sachverhalt: Die durch den Punkt $(c, 0)$ verlaufende Tangente an den Graphen hat $(a+b)/2$ als Berührstelle.

Dies lässt sich leicht auf Funktionen vom Grad 4 verallgemeinern: Hat eine solche die reellen (möglicherweise zusammenfallenden) Nullstellen a, b, c und d, also die Gestalt

$$f(x) = k(x-a)(x-b)(x-c)(x-d)\,,$$

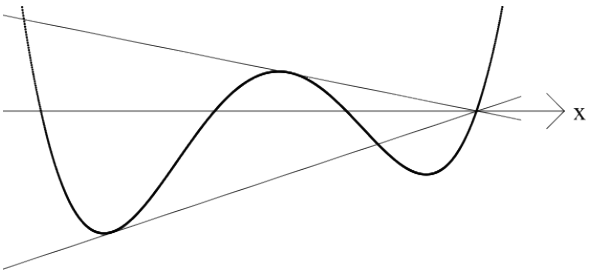
so hat die Tangente an der Stelle s die Gleichung

$$t(x) = f'(s)(x-s) + f(s)$$

und damit die Nullstelle $s - f(s)/f'(s)$. Diese stimmt genau dann mit d überein, wenn s die Gleichung

$$(s-d)^2(3s^2 + 2s\sigma_1 + \sigma_2) = 0$$

erfüllt; dabei seien $\sigma_1 = -(a+b+c)$ und $\sigma_2 = ab+bc+ca$ die elementar-symmetrischen Funktionen zu a, b, c. Es gibt somit zwei Stellen, an denen die zugehörige Kurventangente die Nullstelle d hat:



Wie sieht es aus mit Kurven vom Grad 5 und den reellen Nullstellen $a, \ldots, e$? Analoge Betrachtungen führen auf

$$s = (x-e)^2(4s^3 + 3s^2\sigma_1 + 2s\sigma_2 + \sigma_3)$$

mit den entsprechenden elementar-symmetrischen Funktionen.

Man erkennt das Bildungsgesetz: Hat eine Funktion die reellen Nullstellen $a_1, a_2, \ldots, a_n$ und bezeichnet σ_i die zu i Faktoren gehörige elementar-symmetrische Funktion zu $a_1, a_2, \ldots, a_n$ mit dem Vorzeichen $(-1)^i$, so hat die durch den Punkt $(a_n, 0)$ verlaufende Tangente an den Graphen die Nullstellen von

$$\sum_{i=1}^{n-1}(n-i)s^{n-i-1}\sigma_{i-1} = 0$$

als Berührstellen; dabei sei $\sigma_0 = 1$.

Das ist auch leicht einzusehen: Zu lösen ist $s - a_n = f(s)/f'(s)$. Wegen

$$f(s) = \left(\sum_{i=0}^{n-1}\sigma_i s^{n-1-i}\right)(s-a_n)$$

und daher

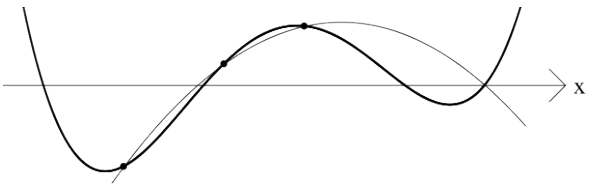
$$f'(s) = \left(\sum_{i=0}^{n-2}(n-1-i)\sigma_i s^{n-2-i}\right)(s-a_n) + \frac{f(s)}{s-a_n}$$

lässt sich die zu lösende Gleichung umformen zu

$$\left(\sum_{i=0}^{n-2}(n-1-i)\sigma_i s^{n-2-i}\right)(s-a_n)^2 = 0\,,$$

was zu zeigen war.

2. Eine zweite Verallgemeinerung geht von den arithmetischen Mitteln der Nullstellen aus. Eine kubische Funktion hat zu den Nullstellen a und b nur ein einziges arithmetisches Mittel; als ausgezeichnete Kurve kommt hier nur die Tangente in Betracht. Eine Funktion vom Grad 4 hat zu den drei Nullstellen a, b und c drei arithmetische Mittel; als ausgezeichnete Kurve durch die drei zugehörigen Punkte kommt die Parabel in Betracht, die durch eben diese drei Punkte verläuft. Man stellt fest und verifiziert mithilfe eines CAS, dass die Parabel tatsächlich d als Nullstelle hat:

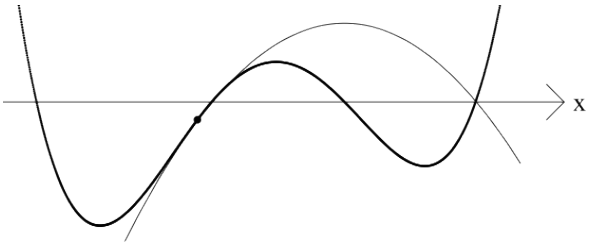


Hier ist eine Verallgemeinerung auf höhere Grade nicht mehr sinnvoll: Eine Funktion vom Grad 5 hat zu vier Nullstellen sechs verschiedene arithmetische Mittel, das zugehörige Interpolationspolynom daher den Grad 5. Folglich stimmt es mit dem Ausgangspolynom überein.

Eine Funktion vom Grad 6 hat zu fünf Nullstellen zehn verschiedene arithmetische Mittel, das zugehörige Interpolationspolynom hätte also den Grad 9. Da aber alle zu den arithmetischen Mitteln gehörigen Punkte auf der Ausgangskurve liegen, hat das Interpolationspolynom nur den Grad 6 und stimmt mit dem Ausgangspolynom überein.

Offensichtlich kann man für höhere Grade auch so argumentieren.

3. Eine dritte Verallgemeinerung geht von den arithmetischen Mitteln der Nullstellen aus. Eine Funktion mit 4 reellen Nullstellen hat zu drei Nullstellen ein einziges arithmetisches Mittel. Der zugehörige Punkt auf der Kurve hat als ausgezeichnete Kurve die Schmiegeparabel, die mit der Originalkurve die 0., 1. und 2. Ableitung gemeinsam hat. Die Schmiegeparabel verläuft durch die 4. Nullstelle der Ausgangsfunktion:



Allgemeine Begründung für höhere Grade: Man kann das Koordinatensystem so verschieben und strecken, dass die $n-1$ Nullstellen $a_1, a_2, \ldots, a_n$ als arithmetisches Mittel 0 haben und dass $a_n = 1$ ist. Dann ist die Schmiegekurve das Taylorpolynom p an der Stelle 0 vom Grad $n-2$.

Wegen

$$\begin{aligned} f(x) = {} & \big(x^{n-1} + \sigma_2 x^{n-3} + \cdots + \sigma_{n-1}\big)(x-1) \\ = {} & x^n + \sigma_2 x^{n-2} + \cdots + \sigma_{n-1}x \\ & -\big(x^{n-1} + \sigma_2 x^{n-3} + \cdots + \sigma_{n-1}\big) \end{aligned}$$

ist die Schmiegekurve gegeben durch

$$p(x) = \sigma_2 x^{n-2} + \cdots + \sigma_{n-1}x - \big(\sigma_2 x^{n-3} + \cdots + \sigma_{n-1}\big)\,;$$

sie hat die Eigenschaft

$$p(1) = \sigma_2 + \cdots + \sigma_{n-1} - \big(\sigma_2 + \cdots + \sigma_{n-1}\big) = 0\,,$$

was zu zeigen war.

Literatur

[1] Henn, H.-W. (2000): Analysisunterricht im Aufbruch. In: Der Mathematikunterricht, Jahrgang 46, Heft 4–5, S. 26–45.

Mathematik hören und Musik sehen mit Hilfe eines Computeralgebrasystems

U. Schürmann
(Westfälische Wilhelms-Universität Münster)

schuermann.uwe@uni-muenster.de

Abstract

Musik ist auch mathematisch fassbar. Die Erzeugung von Tönen und Klängen kann beispielsweise mit Hilfe von Funktionen beschrieben werden. Moderne Computeralgebrasysteme (CAS) bieten umgekehrt auch die Möglichkeit, funktionale Zusammenhänge nicht nur durch einen Funktionenplotter graphisch zu verdeutlichen, sondern Funktionen auch „hörbar“ zu machen. Im Folgenden werden daher praxisorientierte Aufgabenbeispiele für den Unterricht am Ende der Sekundarstufe I und in der Sekundarstufe II vorgestellt, die dieses Potenzial verdeutlichen sollen. Es wird beispielsweise gezeigt wie Töne, Klänge und Akkorde erzeugt werden können, um dann ein Unterrichtsvorhaben zu skizzieren, bei dem es darum geht, verschiedene Hörtests von Schülerinnen und Schülern eigenständig entwickeln zu lassen. Alle Beispiele wurden mit dem quelloffenen CAS Maxima umgesetzt. Sie sind mit kurzen Erläuterungen versehen worden, um den Kontext auch für musikalische Laien – vor und hinter dem Lehrerpult – zu öffnen.

Vorbemerkungen und technische Voraussetzungen

Unterrichtsvorschläge, die das Potenzial des Kontextes Musik bei der Behandlung von Funktionen im Mathematikunterricht kenntlich machen, werden im Folgenden unterbreitet. Dabei ist zu beachten, dass der Autor selbst vor dem Verfassen des Artikels weitgehend musikalischer Laie gewesen ist. Den sich aus diesem Umstand ergebenden Nachteilen stehen jedoch auch gewisse Vorteile gegenüber. So wurde besonders darauf geachtet, Aufgabenbeispiele derart zu gestalten, dass sie auch von „unmusikalischen“ Lehrpersonen sowie Schülerinnen und Schülern bewältigt werden können. Gleichwohl muss an dieser Stelle eingeräumt werden, dass der Kontext – wie jeder andere Kontext auch – gewisses kontextspezifisches Wissen verlangt. Hierzu werden den Aufgabenbeispielen Erläuterungen vorangestellt, die möglichst anschaulich und in mathematiknaher Sprache den jeweiligen musikalischen Hintergrund beleuchten.

Als Grundlage für die Arbeit im Unterricht wurde das CAS Maxima[1] gewählt, welches kostenlos unter verschiedenen Betriebssystemen genutzt werden kann.[2] Es bietet gegenüber kommerziellen Anwendungen den Vorteil, dass Schülerinnen und Schüler Unterrichtsmaterialien auch auf dem heimischen Computer nutzen können. Damit Maxima Funktionen in Musik umwandeln kann, ist ein Paket namens sound.lisp nötig.[3] Außerdem muss noch ein Programm zum Abspielen der so entstehenden Wave-Dateien vorhanden sein. Im Beispiel in Abbildung 1 wird der quelloffene VLC-Player mit `player = "vlc"` festgelegt.[4] Zur Ansicht der Funktionsplots wird das ebenfalls quelloffene Gnuplot ab Version 4.2 benötigt.[5] Das Paket sound.lisp wird dann, wie in Abbildung 1 gezeigt, geladen. Alle im Folgenden gezeigten Unterrichtsbeispiele sind in einer Maxima-Datei enthalten, die zum Download auf der In-

[1] Download: http://maxima.sourceforge.net/ (Zugriff am 12.09.13). Eine Einführung in das Programm Maxima findet sich z. B. hier: http://maxima.sourceforge.net/docs/tutorial/de/maxima-einfuehrung.pdf (Zugriff am 12.09.13).

[2] Alternativ ist auch das Java-Applet von Reinhard Oldenburg zu empfehlen, welches unter http://www.math.uni-frankfurt.de/~oldenbur/webAU/index.html (Zugriff am 12.09.13) abgerufen werden kann. Es bietet den Vorteil, dass keine Software installiert werden muss und Ton und Funktionsplot unmittelbar zu hören bzw. zu sehen sind. Einige Anwendungen aus diesem Artikel, wie z. B. Melodien interpretiert als abschnittsweise definierte Funktionen, können hingegen mit dem Applet nicht realisiert werden (vgl. auch Oldenburg 2005).

[3] Download: Auf der Internetseite des Autors unter http://tinyurl.com/pt6xp8r oder unter http://riotorto.users.sourceforge.net/sound/ (Zugriff am 12.09.13). Eine Anleitung zum Paket sound.lisp findet sich ebenfalls auf der Seite.

[4] Download: https://videolan.org/vlc/ (Zugriff am 12.09.13).

[5] Download: http://www.gnuplot.info/ (Zugriff am 12.09.13). Ist unter Windows im Download von Maxima enthalten.

ternetseite des Autors bereitsteht.[6]

Hingewiesen sei noch darauf, dass das Programm Maxima im Folgenden nicht als CAS im eigentlich Sinne fungiert, vielmehr wird es ausschließlich als Funktionenplotter verwendet. Denn neben der graphischen Darstellung von funktionalen Zusammenhängen können auch durch Funktionen generierte Audio-Dateien als Funktionsplots angesehen werden. Schließlich bewegen sich Lautsprecher beim Abspielen einer Audio-Datei in Abhängigigkeit der Zeit in vorher festgelegter, d. h. eindeutiger Weise. Computeralgebra, bei der symbolische Rechnungen oder Umformungen durchgeführt werden, braucht es dazu nicht.

```
--> load("PFAD ZUR DATEI/sound.lisp");
    set_sound_defaults(
       player    = "vlc",
       draw_wave = true,
       draw_wave_options = [terminal   = png,
                            dimensions = [1600,800]
                           ] );
```

Abbildung 1: *Laden des Pakets sound.lisp.*

Unterrichtsbeispiele

Es werden nun praxisorientierte Aufgabenbeispiele für den Unterricht vorgestellt, wobei jeweils der musiktheoretische Hintergrund soweit wie nötig erläutert und die Benutzung des CAS anhand von Ausschnitten aus der zugehörigen Maxima-Datei dargestellt wird. Dem Kontext entsprechend wird zunächst ein einzelner Ton mit Hilfe einer Sinusfunktion erzeugt, anschließend die Lautstärke des Tons mittels einer ganzrationalen Funktion reguliert, alsdann werden Melodien durch abschnittsweise definierte Funktionen erzeugt. Weiterhin wird aufgezeigt, wie Schülerinnen und Schüler unterschiedliche Hörtests selbst entwickeln können, wie mit dem CAS Maxima eine Zuordnungsaufgabe erstellt wird, bei der Hörerlebnisse den entsprechenden Funktionen zuzuordnen sind, wie sich Transformationen an Funktionsgraphen auditiv auswirken und was Akkorde und Klänge sind und wie diese mit mathematischen Mitteln erzeugt werden. Abschließend werden Perspektiven und Grenzen des Ansatzes diskutiert.

Die ersten Töne erzeugen

Zunächst soll der Kammerton A erzeugt werden, welcher nach einem DIN-Standard (DIN 1317-1) auf 440 Hz, d. h. 440 Schwingungen pro Sekunde festgelegt ist. Schwingungen können am besten mit Sinusfunktionen beschrieben werden. Daher wird die Funktion $f(t) = \sin(2\pi \cdot 440 \cdot t)$ verwendet. Wie in Abbildung 2 zu sehen ist, sind in der Eingabezelle von Maxima der Funktion noch die Parameter 0, 1 und 0,5 zugeordnet. 0 und 1 geben an, für welchen Zeitabschnitt t das Tonsignal zum Funktionsterm ausgegeben werden soll, hier also in der Zeit von 0 bis zur 1. Sekunde. 0.5 gibt an, zu welchem Zeitpunkt der Ton abgespielt werden soll. Standardmäßig, wenn der Parameter nicht angegeben wird, ist dies der Zeitpunkt 0.

[6]Download: http://tinyurl.com/pt6xp8r .

```
--> play (wave(sin(2*%pi*440*t), t, 0, 1, 0.5) );
```

Abbildung 2: *Der erste Ton.*

Wenn man nun mit [Strg]+[Enter] am Ende der Eingabezelle die Berechnungen von Maxima startet, wird der erste Ton generiert. Maxima erstellt eine Audio-Datei im Wave-Format und eine Bild-Datei im PNG-Format, welche den zur Audio-Datei passenden Funktionsplot beinhaltet. Beide Dateien liegen anschließend im Benutzerordner. Wichtig ist darauf hinzuweisen, dass insbesondere Funktionsplots periodischer Signale – wie z. B. hochfrequente Schallwellen von Tönen – aufgrund von Aliasing, dem sogenannten Stroboskopeffekt, fehlerbehaftet sein können (vgl. Hischer 2006). In den Funktionsplots zu den in diesem Artikel gemachten Beispielen lässt sich der Effekt leicht hervorrufen, indem Funktionsplots mit dem Mausrad zügig vergrößert oder verkleinert werden.

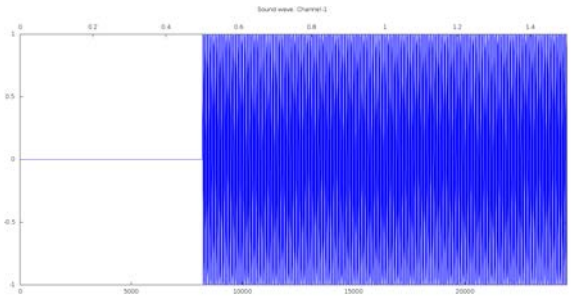


Abbildung 3: *Funktionsplot zum ersten Ton.*

Die Lautstärke festlegen

Zunächst wurde ein Ton mit gleichbleibender, voller Lautstärke erzeugt. Nun soll mittels einer Verknüpfung der Sinusfunktion mit einer Lautstärkefunktion, der sogenannten Amplitude, die Lautstärke variiert werden. Im folgenden Beispiel (siehe Abbildung 4) wird hierzu eine lineare Funktion mit der Steigung 0,9 verwendet. Die Funktionswerte der Amplitude sollten zwischen 0 und 1 liegen, da sich negative Werte der Amplitude wie positive auf die Lautstärke auswirken und Werte, die größer sind als 1, die Lautstärke nicht weiter erhöhen.

```
--> play (wave(0.9*t*sin(2*%pi*440*t), t, 0, 1) );
```

Abbildung 4: *Lautstärke festlegen.*

Melodien als abschnittsweise definierte Funktionen

Die im Unterricht mitunter behandelten, abschnittsweise definierten Funktionen lassen sich im Rückgriff auf Töne veranschaulichen, zum Beispiel dann, wenn eine Melodie gespielt werden soll. Im Beispiel (siehe Abbildung 5) ist dies eine einfache Melodie aus den Tönen C (261,63 Hz), D (293,67 Hz) und E (329,63 Hz). Unsere Melodie würde dann ungefähr der abschnittsweise definierten Funktion f entsprechen mit:

$$f(t) = \begin{cases} 0 \;\; für \;\; t < 0 \;\vee\; t \geqslant 3 \\ \sin(2\pi \cdot 261{,}63t) \;\; für \;\; 0 \leqslant t < 1 \\ \sin(2\pi \cdot 293{,}67t) \;\; für \;\; 1 \leqslant t < 2 \\ \sin(2\pi \cdot 329{,}63t) \;\; für \;\; 2 \leqslant t < 3 \end{cases}$$

```
--> play(
    wave(sin(2*%pi*261.63*t), t, 0, 1, 0),
    wave(sin(2*%pi*293.67*t), t, 0, 1, 1),
    wave(sin(2*%pi*329.63*t), t, 0, 1, 2)
    );
```

Abbildung 5: *Melodie als abschnittsweise definierte Funktion.*

Hörtest generieren

Mit dem Wissen über die Frequenz, die Amplitude und abschnittsweise definierte Funktionen können Schülerinnen und Schüler bereits erste realistische Anwendungsaufgaben bearbeiten. Denkbar wäre, die Schülerinnen und Schüler einen *Hörtest* mit Hilfe des CAS erstellen zu lassen, bei dem unterschiedliche Lautstärken auf unterschiedlichen Frequenzen getestet werden (siehe Abbildung 6). Bedenkt man, dass die tatsächliche Lautstärke je nach verwendeter Hardware (Verstärker, Lautsprecher) variiert, so wird deutlich, dass der Test nicht unbedingt reliabel ist. Allerdings kann in der Schule an einem Rechner das Hörvermögen verschiedener Schülerinnen und Schüler miteinander verglichen werden. Dies erhöht sehr wohl den Aussagewert des Tests.

Das Spektrum hörbarer Frequenzen bewegt sich für einen jungen erwachsenen Menschen in der Regel zwischen 20 Hz und 16 kHz. Üblicherweise decken Kopfhörer diesen Frequenzbereich ab. Mit einem Test könnte dann z. B. deutlich gemacht werden, dass das menschliche Gehör einen bestimmten Frequenzbereich, den Hauptsprachbereich (ca. 500 bis 3000 Hz), besonders gut erfassen kann. Dies äußert sich darin, dass Frequenzen in diesem Bereich schon früh, d. h. bei geringem Lautstärkepegel gehört werden. Genügend gute (und leider auch teure Kopfhörer) geben auch sehr niedrige Frequenzen (Infraschall) und sehr hohe Frequenzen (Ultraschall) wieder. Mit solchen Kopfhörern könnte auch getestet werden, welche Frequenzen überhaupt noch hörbar sind. Von Tests an der Unbehaglichkeits- oder gar Schmerzschwelle sollte im Unterricht aus nachvollziehbaren Gründen abgesehen werden.

Im Beispiel in Abbildung 6 wurde ein einfacher Hörtest erstellt, bei dem die Frequenzen 50 Hz, 100 Hz und 150 Hz mit linear ansteigender Lautstärke für je 10 Sekunden getestet werden.

```
--> play(
    wave(0.1*t*sin(2*%pi*50*t),  t, 0, 10,  0),
    wave(0.1*t*sin(2*%pi*100*t), t, 0, 10, 10),
    wave(0.1*t*sin(2*%pi*150*t), t, 0, 10, 20)
    );
```

Abbildung 6: *Hörtest.*

Auch das Phänomen der *Schwebung* kann von Schülerinnen und Schülern eigenständig untersucht werden. Von einer Schwebung spricht man, wenn zwei Töne nur leichte Unterschiede in der Wellenlänge aufweisen und gleichzeitig gespielt werden. Zwei gleichzeitig erklingende Töne können vom menschlichen Gehör in der Regel nur dann unterschieden werden, wenn sie mindestens eine Terz auseinander liegen, d. h. dass der eine Ton eine um den Faktor 6/5 höhere Frequenz haben muss als der andere. Erfüllen zwei Töne dieses Kriterium nicht, hört man lediglich die Schwebung: Gehört wird nur ein einziger Ton aus dem Mittelwert der beiden Tonfrequenzen, der allerdings in seiner Lautstärke mit der sogenannten Schwebungsfrequenz variiert, die der Differenz der Frequenzen der beiden Töne entspricht (vgl. Hischer 2006, S. 20).[7] In Abbildung 7 sind zwei Sinusfrequenzen dargestellt, wobei die mittlere im selben Intervall zwei Schwingungen mehr aufweist als die obere. Für die Lautstärke der Differenzfrequenz (untere Darstellung) bedeutet das: Der Abstand der benachbarten Lautstärkemaxima vergrößert sich derart, dass Schwankungen der Lautstärke deutlich hörbar werden.

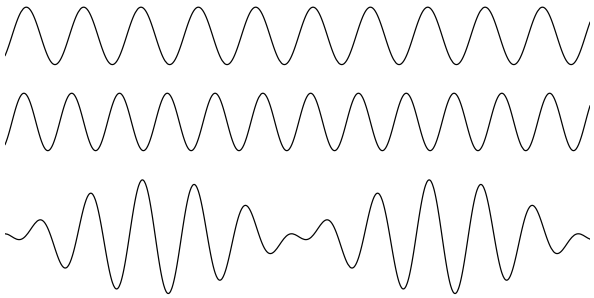


Abbildung 7: *Schwebung.*

Maskierung nennt man den Umstand, wenn neben Tönen oder Klängen aus einer Frequenzgruppe noch weitere Töne oder Klänge zu hören sind, wodurch die Hörbarkeit der eigentlichen Frequenzen deutlich verschlechtert wird. Man stelle sich ein Gespräch vor, bei dem derweil noch andere Töne (z. B. von Elektronikgeräten) zu hören sind. Die Situation dürfte den meisten Schülerinnen und Schülern geläufig sein. Auch dieser Umstand kann mit Hilfe eines selbst generierten Hörtests von Schülerinnen und Schülern näher untersucht werden (zu den Ausführungen über hörbare Frequenzen, Hauptsprachbereich, Frequenzgruppen und Maskierung vgl. Klinke und Silbernagl 1996, S. 571f).

Funktionen hören

Zum tieferen Verständnis des Zusammenhangs zwischen Funktionsvorschrift und Funktionsgraphen werden im Mathematikunterricht häufig Zuordnungsaufgaben verwendet, bei denen aus diversen Graphen der zu einem Funktionsterm passende ausgewählt werden muss. Hierdurch soll beim Lernenden eine Verknüpfung zwischen ikonischer und symbolischer Ebene hergestellt werden (vgl. Bruner 1974). Durch den Rückgriff

[7] Die Untersuchung funktioniert am besten mit reinen Tönen. Zur Unterscheidung von Tönen, Klängen und Geräuschen siehe Abschnitt „Akkorde und Klänge erzeugen“.

auf akustische Reize, wie sie durch das CAS Maxima erzeugt werden können, wird eine weitere sinnliche Verknüpfung der Lerninhalte möglich. Nach der Theorie des Conceptual Change führt die mannigfaltige Verknüpfung von Inhalten zu mehr Nachhaltigkeit im Lernen (vgl. Vosniadou 1994).

Der Vergleich zwischen Funktionsterm und akustischem Signal kann hierbei z. B. über eine ganzrationale Amplitudenfunktion erfolgen, in zweiter Linie kann auch die Tonfrequenz als Unterscheidungsmerkmal herangezogen werden.

Im folgenden Beispiel (siehe Abbildung 8) wird lediglich die Amplitude betrachtet. Diese wird hier mit Hilfe von vier verschiedenen linearen Funktionen dargestellt (zur Verbindung von Funktionsgraphen und Tonfrequenz in Abhängigkeit von der Zeit siehe Reiter 2011).

```
--> play (wave((-0.5*t+1)*sin(2*%pi*440*t), t, 0, 2) );
--> play (wave(0.4*sin(2*%pi*440*t), t, 0, 2) );
--> play (wave(0.8*sin(2*%pi*440*t), t, 0, 2) );
--> play (wave((0.2*t)*sin(2*%pi*440*t), t, 0, 2) );
```

Abbildung 8: *Funktionen hören.*

Töne und Transformationen

Transformationen sind ein weiterer Aspekt im Umgang mit Funktionen, der ebenso im Kontext von Musik anschaulich gemacht werden kann. Was bewirkt eine Verschiebung des Graphen der Amplitude nach links, rechts, oben oder unten? Was bewirkt eine Spiegelung? Dies sind Fragen, die von Schülerinnen und Schülern zunächst experimentell und anschließend analytisch geklärt werden können. In Abbildung 9 wird eine Amplitude, eine nach unten geöffnete Parabel, auf der x-Achse um eine Einheit nach rechts verschoben.

```
Beispiel 1: Amplitude als nach unten geöffnete Parabel
 mit den Nullstellen 0 und 2.
--> play (wave((-t^2+2*t)*sin(2*%pi*440*t), t, 0, 2) );
Beispiel 2: Amplitude um 1 Einheit nach rechts verschoben.
--> play (wave((-(t-1)^2+2*(t-1))*sin(2*%pi*440*t), t, 0, 2) );
```

Abbildung 9: *Verschieben der Amplitude.*

Hört man sich nun das Ergebnis nach der Verschiebung an, so wird deutlich, dass negative Werte der Amplitude die gleichen Lautstärken hervorrufen wie positive. Dies liegt selbstredend darin begründet, dass es sich bei dem eigentlichen Ton um eine Schwingung handelt, was wiederum am Funktionsplot gut zu erkennen ist, wie Abbildung 10 und 11 zeigt.

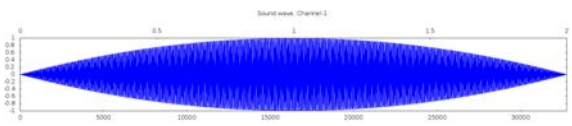


Abbildung 10: *Vor der Verschiebung.*

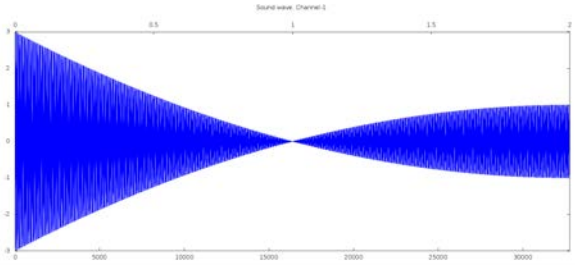


Abbildung 11: *Nach der Verschiebung.*

Akkorde und Klänge erzeugen

Bisher wurden nur einzelne Töne betrachtet. Ein wohlklingendes Zusammenspiel von Tönen, ein Akkord, kann ebenso mit Maxima realisiert werden. Hierzu wird einfach aus verschiedenen Sinusfunktionen eine Audio-Datei generiert, wie in Abbildung 12 zu sehen ist.

```
--> play(   wave((-t^2+2*t)*sin(2*%pi*261.63*t), t, 0, 2),
            wave((-t^2+2*t)*sin(2*%pi*329.63*t), t, 0, 2),
            wave((-t^2+2*t)*sin(2*%pi*392.00*t), t, 0, 2)
        );
```

Abbildung 12: *Akkord erzeugen.*

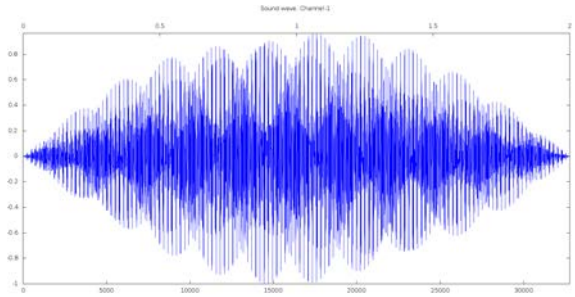


Abbildung 13: *Funktionsplot zum Akkord.*

Auffällig bleibt, dass sich Akkorde, so wie einzelne Töne auch, immer noch steril und unnatürlich anhören. Töne von Musikinstrumenten hören sich anders an. Was aber macht den Unterschied zu einem echten Instrument aus? Woher stammt seine eigentümliche Klangfarbe?

Der Unterschied liegt darin begründet, dass Musikinstrumente niemals reine Sinustöne wiedergeben. Zunächst ist der Ton als reine Sinusschwingung nur ein Modell. Das Modell berücksichtigt nicht die Masse der Saiten eines Saiteninstrumentes, welche den Schwingungsverlauf beeinflusst. Des Weiteren variieren Musiker bespielsweise das Tempo und den Anschlag oder sie spielen auf bundlosen Saiteninstrumenten den Ton nicht exakt. Dies muss kein Mangel sein, schließlich setzen viele Musiker derlei Techniken bewusst für ihr Spiel ein. Eine tatsächlich exakt gespieltes Stück aus dem Computer kann dagegen sehr eintönig klingen.

Der jedoch wichtigste Unterschied zwischen reinen Tönen und echten Klängen eines Instruments besteht darin, dass Klänge von Instrumenten gar nicht bloß aus einem einzigen Ton bestehen. Vielmehr gesellen sich zu einem Grundton noch diverse Obertöne hinzu. Bei diesen handelt es sich um Vielfache der Frequenz des Grundtons, also Oktaven. Des Weiteren spricht man von Geräuschen, wenn mehrere Frequenzen ohne die eben genannte Gesetzmäßigkeit zusammen zu hören

sind. Die Mischung aus Grund- und Obertönen und einem kleinen Anteil an Geräuschen gibt einem Instrument dann seine je charakteristische Klangfarbe (vgl. Michels 1977, S. 17). Die Modellierung von Instrumenten am Computer war aufgrund der Vielzahl von Faktoren, die den Klang eines Instruments beeinflussen, lange Zeit nur eingeschränkt möglich. Moderne Heimcomputer verfügen aber mittlerweile über genügend Rechenleistung, um auch dieser Aufgabe in ausreichendem Maße gewachsen zu sein.[8]

Mit einem CAS ist es allerdings sehr aufwendig, den Klang eines Instruments nachzubilden. Im Beispiel in Abbildung 14 wurde mit einfachen Mitteln versucht den Kammerton A (440 Hz) eines Klaviers nachzubilden, wobei das Ergebnis immer noch deutlich vom Hörerlebnis beim Anspielen des Tons auf einem Klavier abweicht.[9]

```
--> play (  wave((-t^2*(-t+2)^2)*0.7*sin(2*%pi*440*t), t, 0, 2),
            wave((-t^2*(-t+2)^2)*0.08*sin(2*%pi*880*t), t, 0, 2),
            wave((-t^2*(-t+2)^2)*0.01*sin(2*%pi*1320*t), t, 0, 2),
            wave((-t^2*(-t+2)^2)*0.02*sin(2*%pi*1760*t), t, 0, 2),
            wave((-t^2*(-t+2)^2)*0.02*sin(2*%pi*2200*t), t, 0, 2),
            wave((-t^2*(-t+2)^2)*0.1*sin(2*%pi*2640*t), t, 0, 2),
            wave((-t^2*(-t+2)^2)*0.02*sin(2*%pi*3080*t), t, 0, 2),
            wave((-t^2*(-t+2)^2)*0.05*sin(2*%pi*3520*t), t, 0, 2)
         );
```

Abbildung 14: *Klang erzeugen.*

Ausblick und Grenzen des Ansatzes

Hischer (2005, S. 120) postuliert, „dass auch der *Umgang* mit den Neuen Medien und ihre *Anwendung* nicht nur mediendidaktischen Zielen dienen, sondern dass entsprechende individuelle Erfahrungen eine geradezu unverzichtbare Voraussetzung dafür sind, dass sie zum Unterrichtsinhalt werden können, indem ihre *Grundlagen und Grundstrukturen* und ihre *Bedeutung für Individuum und Gesellschaft* erörtert werden“ (Hervorhebungen im Original).

Bei den unterbreiteten Unterrichtsvorschlägen fungierte das CAS im Sinne Hischers (vgl. ebenda, S. 118f) zum einen als mediendidaktisches Werkzeug, welches die Schülerinnen und Schüler im Lernprozess unterstützen soll. Zum anderen wird durch die Arbeit mit dem CAS Medienkunde betrieben: Die Lernenden erproben den Umgang mit einem CAS und erhalten anhand von konkreten Beispielen Einblick in das Potenzial von Computeralgebra. Nicht zuletzt kann der Gegenstand aber auch zum Thema in medienerzieherischer Absicht gemacht werden, indem z. B. die Bedeutung computergenerierter Musik in Zeiten der Kulturindustrie kritisch beleuchtet wird. Derlei Betrachtungen weisen über den Mathematikunterricht hinaus und erfordern eine integrierte Medienpädagogik an Schulen.

Wie sich zeigt, kann das Thema Funktionen und Musik bei entsprechender mediengestützter Umsetzung im Unterricht den von Hischer formulierten Ansprüchen im Umgang mit Neuen Medien gerecht werden. Aber auch abseits von Funktionen lassen sich weitere Themen der Schulmathematik benennen, in denen mathematische Betrachtungen zum tieferen Verständnis von Musik beitragen und Computeralgebra sinnvoll eingesetzt werden kann.

So können beispielsweise Aspekte von Musik auch aus Sicht der Linearen Algebra bzw. Analytischen Geometrie erhellend betrachtet werden. Denkbar wäre an dieser Stelle ein mit dem Fach Musik verbundener Unterricht.

Für die Sekundarstufe II bieten sich hier verschiedene Möglichkeiten an: Werden im Musikunterricht Tonsysteme behandelt, so können deren zugrunde liegenden Tonstrukturen mathematisch betrachtet werden. Z. B. kann nach einer Idee von Leonhard Euler in Tonsystemen in reiner Stimmung jedes Verhältnis zwischen den einzelnen Frequenzen mit einer Zahl des Typs $2^x \cdot 3^y \cdot 5^z$ mit den ganzen Zahlen x, y und z gedeutet werden, also als Vektor (x, y, z) bestehend aus den ganzzahligen Potenzen von Oktave, Quinte und Terz. Man erhält so einen anschaulich fassbaren Ton- und Intervallraum, wobei die Punkte dieses Raums den Tönen entsprechen und die Vektoren den musikalischen (!) Intervallen (vgl. Mazzola 1990, S. 25ff). Mit Hilfe von Maxima könnten dabei verschiedene Tonsysteme relativ schnell, ohne dass man ein Instrument bauen oder anders stimmen müsste, hörbar gemacht werden. Denkbar wäre die Behandlung von historischen Tonsystemen, sowie Tonsystemen aus anderen Kulturkreisen.

Werden im Musikunterricht grundlegende Kompositionstechniken behandelt, so bietet sich eine Anknüpfung an affine Abbildungen aus der Analytischen Geometrie an. Hierbei wäre ein Vergleich zwischen den affinen Abbildungen, wie sie in der geometrischen Ebene möglich sind, und den isometrischen Transformationen, die in der musikalischen Ebene – bestehend aus der Zeit als x-Achse und Tonhöhe als y-Achse – möglich sind, sinnvoll (vgl. Hart 2009).

Aber auch im Mathematikunterricht der frühen Sekundarstufe I kann der Kontext bereits zur Anwendung kommen. Leuders und Leuders (2011) diskutieren die Einführung der Bruchrechnung anhand von geklopften Rhythmen. Aus Sicht des Autors wäre es jedoch auch möglich ein Monochord (Saiteninstrument mit nur einer Seite) zu bauen und daran den Zusammenhang zwischen Teilungsverhältnissen und Tonfrequenzen zu erkunden.

Grenzen sind dem hier gezeigten Ansatz in technischer Hinsicht gesetzt. Es zeigt sich, dass die Berechnung von Klängen oder gar Melodien sehr rechenaufwendig für den Computer ist. Es bedarf demnach moderner Hardware, um im Unterricht nicht übermäßig große Wartezeiten entstehen zu lassen. Eine inhaltliche Hürde – für die Schülerinnen und Schüler als auch für die Lehrperson – tut sich immer dann auf, wenn Fachwissen aus der Musiklehre für die Anwendungen benötigt wird. Es muss jeweils bezogen auf die ein-

[8] In einem Artikel der Zeit (Ausgabe 2002/03) heißt es dazu: „Normale Heimcomputer ersetzen schwere Synthesizer und ganze Tonstudios. Eine neue Ära der Laienmusik hat begonnen.“

[9] Um den Klang eines Instruments zu rekonstruieren benötigt man die Fourieranalyse, welche in der Regel im Schulunterricht nicht zum Thema gemacht wird.

zelne Lerngruppe entschieden werden, bis zu welchem Grad musiktheoretische Grundlagen im Mathematikunterricht berücksichtigt werden können, oder aber, ob eine Kooperation mit dem Fach Musik möglich und sinnvoll ist.

Literatur

[1] Jerome S. Bruner: *Entwurf einer Unterrichtstheorie*. Berlin: Berlin Verlag, 2009.

[2] Vi Hart: Symmetry and Transformations in the Musical Plane. *Proceedings of the 12th Annual BRIDGES Conference: Mathematics, Music, Art, Architecture, Culture (BRIDGES 2009)*, Banff:169–176, 2009.

[3] Horst Hischer: Aliasing und Neue Medien – Ein Beitrag zur Integrativen Medienpädagogik. *Kaune, Christa & Schwank, Inge & Sjuts, Johann (Hrsg.): Mathematikdidaktik im Wissenschaftsgefüge – Zum Verstehen und Unterrichten mathematischen Denkens. Festschrift für Elmar Cohors-Fresenborg*. Osnabrück: Schriftenreihe des FMD, 40/1:115–129, 2005.

[4] Horst Hischer: Abtast-Moiré-Phänomene als Aliasing. *Der Mathematikunterricht*, 52/1:18–31, 2005.

[5] Rainer Klinke und Stefan Silbernagl (Hrsg.): *Lehrbuch der Physiologie*. 2. Auflage, Stuttgart, New York: Thieme, 1996.

[6] Juliane Leuders und Timo Leuders: Ich bin ganz Ohr – Mathematik hören und verstehen. *PM 42/53*:2–12, 2011.

[7] Guerino Mazzola: *Geometrie der Töne. Elemente der mathematischen Musiktheorie*. Basel: Bierkäuser, 1990.

[8] Ulrich Michels: *dtv-Atlas zur Musik, Tafeln und Texte*. Band I, München: Deutscher Taschenbuch Verlag, 1977.

[9] Reinhard Oldenburg: Funktionen, Sound und MuPAD. *MNU 59/1*:16–18, 2005.

[10] Stefanie Reiter: Funktionen hören – ein auditiver Zugang zum Bereich funktionale Veränderung. *PM 42/53*:19–23, 2011.

[11] Stella Vosniadou: Capturing and Modeling the Process of Conceptual Change. *Learning and instruction*, 4:45–69, 1994.

Berichte über Arbeitsgruppen

Neu in der Arbeitsgruppe Algebra, Geometrie und Computeralgebra, Fachbereich Mathematik, TU Kaiserslautern

Claus Fieker, Mathias Schulze

Claus Fieker
Mit der Berufung von Claus Fieker von Sydney (Magma) nach Kaiserslautern ist die AGAG, die Arbeitsgruppe Algebra, Geometrie und Computeralgebra, um explizite konstruktive Zahlentheorie bereichert worden. Seine Arbeitsgruppe besteht z. Zt. aus einem Doktoranden (Tommy Hofmann) und 3 Masterstudenten. Die Gruppe wird jedoch ab Oktober noch Unterstützung von einem PostDoc (Bill Hart) erhalten. Die aktuellen Projekte der Arbeitsgruppe decken ein weites Feld ab, Tommy Hofmann arbeitet an ganzen Darstellungen endlicher Gruppen (über Zahlkörpern), Henning Kopp an Problemen in linearer Algebra in Ganzheitsringen und die beiden anderen Mastersstudenten (Simone Deppert, Daniel Behr) untersuchen Fragestellungen aus der Kryptographie im Hinblick auf den Unterricht in der Oberstufe. Der eigentliche Schwerpunkt für die nächsten Jahre liegt jedoch in dem DFG Projekt zur die Berechnung von Klassengruppen. In den nächsten zwei bis drei Jahren werden hier zunächst Grundlagen für Experimente in Zahlkörpern entwickelt und in Singular bzw. Gap eingebunden. Darauf aufbauend werden dann multiplikative Probleme in Zahlkörpern untersucht mit dem Ziel Klassengruppen effizient und schnell berechnen zu können.

Mathias Schulze
Mathias Schulze trat im Herbst 2012 als Professor für Algebra und Geometrie die Nachfolge von Gerhard Pfister an der TU Kaiserslautern an. Seine aktuelle Forschung befaßt sich mit Derivationen und Differentialformen auf Singularitäten. In einem von der EU durch ein Marie Curie-Stipendium geförderten Projekt, geht es konkret um die algebraische Charakterisierung von normalen Überkreuzungen sowie das Studium freier Divisoren. Im Rahmen dieses Projekts konnte der Aluffi-Student Xia Liao als erster PostDoc der neu entstehenden Arbeitsgruppe in Kaiserslautern gewonnen werden. Zur experimentellen Untermauerung des Projekts sollen logarithmische und reguläre Differentialformen in SINGULAR berechenbar gemacht werden. Als Co-PI des DFG-Schwerpunktprojekts "Fundamentale Algorithmen in SINGULAR"(zusammen mit Wolfram Decker und Gerhard Pfister) ist Schulze in die SINGULAR-Entwicklung eingebunden. Ziele dieses Projekts umfassen u.A. die Modularisierung und Paralellisierung des SINGULAR-Kerns sowie Methoden zur absolute Faktorisierung und absolute Primärzerlegung. Hierzu wurden Reimer Behrends (ehem. St Andrews) und Bill Hart (ehem. Warwick) als neue Mitarbeiter eingestellt. Schulze versucht eine bereits in der Vergangenheit erfolgreiche Kooperation mit der Elektrotechnik an der TU Kaiserslautern wiederzubeleben. Hierzu sollen in SINGULAR Grundlagen geschaffen werden, die neuartige Modelle zur Verifikation von Schaltkreisen realisieren.

Besprechungen zu Büchern der Computeralgebra

Hans Wußing
Carl Friedrich Gauß. Biographie und Dokumente.

6. Auflage, Edition am Gutenbergplatz Leipzig, 2011,
279 pp., ISBN 978-3-937219-51-6, € 26,50

Die ersten ca. 100 Seiten dieses Buches enthalten die zuerst 1974 in einer populärwissenschaftlichen Buchreihe erschienene, leicht bearbeitete Biographie. Der Mathematikhistoriker Wußing wendet sich hier also an ein breites Publikum, ohne aber Fachwissenschaftliches zu kurz kommen zu lassen. In ausgewogener, meisterhafter Weise stellt er Leben und wissenschaftliches Werk von Gauß dar, wobei die Folge der Abschnitte Jugendzeit, Studienzeit, Familiäres, Gauß in seiner Zeit, Lebensende unterbrochen wird durch Abschnitte zur Zahlentheorie, Astronomie, Geometrie, Geodäsie und Physik. (In seiner ungefähr doppelt so viele Seiten umfassen-

den Gauß-Biographie von 1981 folgt Kaufmann-Bühler einem ähnlichen Schema; deutliche Unterschiede treten in der Wertung der Zeitumstände hervor.)

Neu in dieser 6.Auflage und anderthalbmal so lang wie die Biographie ist ihr Dokumentenanhang. Der Hauptteil besteht aus 50 Dokumenten, jedes unter ein eigenes Thema gestellt und von wenigen Ausnahmen abgesehen nicht länger als zwei Seiten. Man findet z.B. Passagen aus dem Gauß-Briefwechsel eingebettet in einen Kommentartext, überwiegend aber Reproduktionen von Auszügen aus der Sekundärliteratur. Die ersten 25 Dokumente zeigen Gauß in seinem wissenschaftlichen und privaten Umfeld, so die Generalüberschrift, die weiteren 25 speziell im Spiegel von Teubner-Publikationen (aus Anlass des 200.Jahrestages der Firmengründung). Am Ende begegnet man Gauß auf Briefmarken, Geldscheinen, Münzen ebenso wie als Namensgeber in Technik und Wissenschaft. So werden z.B. die Gauß-Vorlesungen der DMV mit Vortragenden und Themen bis zum April 2011 aufgeführt. Im gleichen Monat ist Hans Wußing 83jährig verstorben.

Heinz-Georg Quebbemann (Oldenburg)

Weitere Bücher können auf der Seite `http://www.fachgruppe-computeralgebra.de/Buecher` oder direkt bei Anne Frühbis-Krüger (`fruehbis-krueger@math.uni-hannover.de`) zur Besprechung angefordert werden.

Ehrenpromotion in der Computeralgebra

Verleihung des Ehrendoktortitels an Bruno Buchberger
Universität Genua, 25. September 2013

Prof. Juan Elias (Barcelona), Prof. Teo Mora (Genua), Prof. Lorenzo Robbiano (Genua), Prof. Bruno Buchberger (Linz), Prof. Martin Kreuzer (Passau)

Am Mittwoch, den 25. September 2013 verlieh die Universität Genua (Italien) Herrn Prof. Dr. Bruno Buchberger (Linz) eine Ehrendoktorwürde in Mathematik. Es war die erste derartige Ehrung, die vom *Dipartimento di Matematica* der Universität vorgeschlagen wurde. In ihrer Begründung für die Verleihung verweist die naturwissenschaftliche Fakultät auf die bemerkenswerte Kombination technischer und organisatorischer Fähigkeiten des Rezipienten: er leistete wesentliche Beiträge zur Entwicklung der modernen Mathematik, insbesondere durch den nach ihm benannten Buchberger-Algorithmus und erreichte Ausserordentliches, indem er die Gründung und den Ausbau des Research Institute of Symbolic Computation (RISC) betrieb, dessen Direktor er 12 Jahre lang war, und indem er den Technologietransfer mit Hilfe des von ihm initiierten, nahegelegenen Software Parks förderte.

Die Laudatio hielt Prof. Dr. Lorenzo Robbiano (Genua). Er verglich die Entdeckung des Buchberger-Algorithmus mit einem Schlüssel, der den Mathematikern eine Tür zu einer wunderbaren neuen Welt öffnete. Ein oft zu wenig beachteter Aspekt der Rolle eines Universitätsprofessors sei die Möglichkeit, die gesellschaftliche Entwicklung zu beeinflussen. Gerade dieser Aspekt sei bei Prof. Buchberger hervorragend ausgeprägt: er spielte zum Beispiel eine wichtige Rolle bei der Einführung der Verwendung von Computern in der Lehre an österreichischen Gymnasien und Universitäten. Prof. Buchberger nutze seinen wohlverdienten Ruhenstand jedoch nicht um sich auf seinen Lorbeeren (u.a. fünf Ehrendoktorwürden) auszuruhen, sondern um sein Theorema Projekt noch energischer voranzutreiben. Er sei somit ein Paradebeispiel eines ganzheitlichen Wissenschaftlers.

Der Geehrte ging in seiner *Lectio Magistralis*, die den Titel *Matematica: il turbo-motore della Scienza e della Tecnologia* trug und in fliessendem Italienisch vorgetragen wurde, auf die Mathematik als die Kultur des klaren Denkens und Sprechens ein. Das Ziel der Mathematik sei es, aus einfachen Anfängen durch einfache Schritte zu immer komplexeren Erkenntnissen zu gelangen. Diese stehen nicht isoliert, sondern in inniger Verbindung zu Wissenschaft und Technik. Prof. Buchberger wies auch darauf hin, dass man trotz der Betonung des wissenschaftlichen Denkens und der technologischen Fortschritte nicht vergessen dürfe, das Naturverständnis harmonisch und intuitiv mit einem Leben in Einklang mit der Natur zu verbinden. Diese Verbindung des Intellekts und der Intuition sei die ultimative Herausforderung unserer Zeit.

Die Verleihung fand in einem sehr feierlichen Rahmen in der altehrwürdigen *Aula Magna* der Universität statt. Die meisten Anwesenden waren in Talare gekleidet und trugen die Insignien ihrer Ämter und Fakultäten. Die Promotionskommission um die bekannten Genueser

Computeralgebraiker Teo Mora und Lorenzo Robbiano wurde um die externen Mitglieder Juan Elias (Barcelona) und den Berichterstatter ergänzt. Die italienische Presse und das Fernsehprogramm TG2 berichteten ausführlich von der Veranstaltung. Der Ehrentag klang mit einem Galadinner im romantischen Ortsteil *Boccadasse* aus.

M. Kreuzer (Passau)

Promotionen in der Computeralgebra

Moritz Minzlaff: Frobenius-stable lattices in rigid cohomology of curves
Betreuer: Florian Heß (Oldenburg)
Zweitgutachter: Remke Kloosterman (HU Berlin)
März 2013
http://opus.kobv.de/tuberlin/volltexte/2013/3934/pdf/minzlaff_moritz.pdf

Zusammenfassung: Die Arbeit beschäftigt sich mit theoretischen und praktischen Aspekten des Berechnens von Zetafunktionen von Kurven über endlichen Körpern. Dieses Problem hat in den gut zehn letzten Jahren bereits viel Aufmerksamkeit erlangt. So gibt es sehr effiziente Algorithmen für elliptische und hyperelliptische Kurven, was insbesondere für Anwendungen in der Kryptographie von Interesse ist. Doch obwohl auch innerhalb der Mathematik die Zetafunktionen zentrale Objekte sind, ist die algorithmische Frage ihrer effizienten Berechnung jenseits von hyperelliptischen Kurven noch weitgehend ungelöst. Diese Arbeit liefert drei Beiträge, um diese Situation mit Hilfe p-adischer Techniken zu verbessern. Zunächst wird ein aus der Deformationstheorie von komplexen Kurven bekanntes Verfahren auf den Basisring der Wittvektoren übertragen. Dies macht es möglich, Liftungen von beliebigen Kurven zu approximieren, auf deren log de Rham Kohomologie eine natürliche Frobenius-Operation existiert, die sich zur Berechnung der Zetafunktion heranziehen lässt. Eine Implementierung des vorgeschlagenen Algorithmus wurde in Sage zu Testzwecken umgesetzt und ist als Patch frei verfügbar.

Als nächstes beschäftigt sich die Arbeit mit einer für Berechnungen möglichst effektiven Beschreibung dieser log de Rham Kohomologien. Im Wesentlichen wird gezeigt, dass Differentiale mit hinreichend großer Polordnung und integrierbarem Hauptteil die Kohomologie erzeugen, wobei das Residuum keine Rolle spielt. Der vorgelegte Beweis ist auch für beliebig-dimensionale Objekte gültig und geht auf Edixhoven und van den Bogaart zurück, welche aber nur den Fall hyperelliptischer Kurven mit einem einzigen Punkt im Unendlichen behandeln. Die Arbeit zeigt ferner, dass die „gewöhnliche“ de Rham Kohomologie für Berechnungen ungeeignet ist: Obwohl eine Frobenius-Operation existiert, enthält der Wittvektor-Modul mindestens g Faktoren des Quotientienkörpers und ist insbesondere nicht endlich erzeugt. (Hier ist g das Geschlecht der Kurve.)

Als letzten Beitrag beschreibt die vorliegende Dissertationsschrift eine Verallgemeinerung von Harveys Algorithmus zur Berechnung der Zetafunktion von hyperelliptischen Kurven für gewisse superelliptische Kurven. Die besondere Eigenschaft des Algorithmus ist ein für p-adische Techniken um den Faktor $p^{1/2}$ verbessertes Laufzeitverhalten. Dies wird durch eine geeignete Approximation des Frobenius und einem Baby-Step/Giant-Step-Ansatz zur Berechnung von Normalformen von Differentialen erreicht. Der Algorithmus wurde im Rahmen der Arbeit in Magma implementiert und ist unter https://github.com/mminzlaff/superelliptic frei verfügbar.

Berichte von Konferenzen

1. **ACAT 2013 – 15th International Workshop on Advanced Computing and Analysis Techniques in Physics Research**

   Peking, China, 16. – 21.5.2013

   http://acat2013.cern.ch

   Der für die Computeralgebra relevante Teil der Konferenz war wie immer Track 3, hier gab es viele kleine Fortschritte bei diversen Packages. Interessant als Nebenaspekt ist, dass Python die Sprache der Wahl in vielen Projekten geworden ist, insbesondere solchen, die sich 'Open Source' auf die Fahnen schreiben.

   Ergänzend gab es eine Diskussion mit Lawrence Pinsky (Rechtsanwalt IP-Recht + ALICE-Kollaboration) zum Thema Lizenzen und Rechte an Software, speziell Open Source Software. Diese Thema wird durch die zunehmende zur-Verfügung-Stellung von Computercodes z.B. für die Berechnung theoretischer Vorhersagen immer relevanter.

   Thomas Hahn (München)

2. **Workshop "Questions, Algorithms, and Computations in Abstract Group Theory"**

   TU Braunschweig, 21.5. – 24.5.2013

   http://www.icm.tu-bs.de/ag_algebra/ws-qac

   Das Ziel dieses Workshops war es, Forscher aus der abstrakten Gruppentheorie, der Algorithmik und der algebraischen Geometrie zusammen zu bringen, um neue Fortschritte in der algorithmischen Gruppentheorie zu ermöglichen. Mit 11 Hauptvorträgen und 18 Kurzvorträgen in vier Tagen erwartete die Teilnehmer ein sportliches, aber auch sehr anregendes und vielfältiges Programm.

   Die Themen reichten von neuen Ansätzen in den klassischen Methoden der algorithmischen Gruppentheorie über Gröbner-Basistechniken, formale Sprachen und Automata bis hin zu Anwendungen in der gruppenbasierten Kryptographie. Mit 58 Teilnehmern war der Workshp sehr gut besucht und von Bettina Eick (Braunschweig) sowie ihren Mitarbeitern Andreas Distler und Matthias Neumann-Brosig hervorragend organisiert. Für die finanzielle Unterstützung der Veranstaltung sorgten der SPP 1489 der DFG, die NSF und die TU Braunschweig.

   Martin Kreuzer (Passau)

3. **Eighth Int. School on Computer Algebra: COCOA 2013**

   Universität Osnabrück, 10.6. – 14.6.2013

   http://cocoa.dima.unige.it/conference/cocoa2013

   Bereits zum achten Mal fand vom 10.6.-14.6.2013 die sog. CoCoA-Schule für internationale Masterstudenten und Doktoranden statt, die für ihre Arbeit Computeralgebra benötigen oder erlernen wollen. Die diesjährige Ausgabe wurde an der Universität Osnabrück durchgeführt und hatte 24 Teilnehmer aus neun Ländern.

   Entsprechend dem üblichen Format fanden zwei Kurse zu je acht Stunden statt, die diesmal die folgenden Themen hatten:

   Winfried Bruns (Osnabrück), Algorithms for toric geometry

   Lorenzo Robbiano (Genua), Sets of points and mathematical models

   Nach zwei Vorträgen am Vormittag gab es jeweils zugehörige Computerübungen am Nachmittag, die mit dem Computeralgebrasystem CoCoA und dem Paket Normaliz zu bearbeiten waren. Die Tutorien wurden von Christof Söger und Laura-Maria Torrente sachkundig geleitet. Ferner gab es Gastvorträge von John Abbott (über den BM-Algorithmus), dem Berichterstatter (über approximative Interpolation) und Dan Grayson (über Homotopietyptheorie als neue Grundlage der Mathematik). Es gab auch eine Postersession, in der die Teilnehmer ihre eigenen Arbeiten vorstellen konnten.

   Die CoCoA-Schule wurde von der Universität Osnabrück, dem SPP 1489 der DFG, dem ital. GNSAGA und der Universität Genua finanziell unterstützt. So konnte den Teilnehmern kostenlose Unterkunft gewährt werden. Die Veranstaltung wurde von allen beteiligten als sehr gelungen gelobt und man darf schon gespannt sein, wo in zwei Jahren die nächste Ausgabe stattfinden wird.

   Martin Kreuzer (Passau)

4. **Workshop on SymbolicData Design**

   Leipzig, 27. – 28.8.2013

   http://symbolicdata.org/wiki/Events.2013-08

   The workshop was designed as final milestone of the E-Science Benchmarking Project promoted for 12 months within the *E-Science Saxony Framework*. Unfortunately, the event was completely ignored by the Computer Algebra Communities, so that we had no opportunity to present the results of the project to a larger audience. Instead we had intense discussions with people from the *swmath* project (http://www.swmath.org, a project of the *Zentralblatt Mathematik* towards an information service for mathematical software) about trends in Semantic Web Technologies that are suitable to support future common efforts towards a semantic aware IT infrastructure for Computer Algebra.

   In a first talk *Hans-Gert Gräbe* presented the state of the SymbolicData project. Note that at the end of September 2013 version 3 of SymbolicData was released, thus finishing a major redesign of SymbolicData, that marks a milestone across the implementation of semantic techniques within Computer Algebra. We strongly use RDF and Linked Data principles in the organisation of the data. These principles are also reflected in the presentation of the data at symbolicdata.org. All resources are delivered via http/rdf+xml and a Sparql endpoint allows for navigation in the metadata. This can be installed also on a localhost and thus can be integrated into a local benchmarking or profiling infrastructure (best using python as scripting language and a web server at localhost). A more detailed description of the new release is available from the SymbolicData web pages and will be given also in the next issue of the *Computeralgebra Rundbrief*.

   Andreas Nareike presented in a second talk his prototypical integration of the Polynomial Systems subproject with sagemath and SymbolicData as a sage package *sdsage* that smoothly integrates both the global SD network infrastructure and a local installation into the sagemath process. One can load data and metadata transparently into sage objects and process them as mathematical objects in the usual way within sage.

   Ulf Schöneberg gave a talk about effort at the ZBMath to discover and understand mathematical formulas in Zentralblatt mathematical reviews, mixing classical colocation approaches with semantic enriched opportunities of latex mark-up.

This research is part of larger efforts within, e.g., the OpenMath activities.

We discussed in great detail the potential interplay between

- the efforts at ZBMath to organize access to data in well established RDF based formats,
- the SymbolicData intercommunity efforts and experience with Linked Data standards, Sparql endpoints, Virtuoso and Ontowiki based local installations,
- ongoing efforts of the DNB and other libraries (SLUB Dresden, UB Leipzig) to reshape their catalogue data towards Linked Data standards and get them interoperating within the GND project,
- perspectives to join forces with these library projects to strengthen the IT infrastructure for Computer Algebra Communities.

Hans-Gert Gräbe (Leipzig)[10]

5. LMS/EPSRC Short Instructional Course Computational Group Theory

St.Andrews, 29.7. – 2.8.2013

www-circa.mcs.st-and.ac.uk/cgt2013

Im Vorfeld der "Groups St.Andrews 2013" fand vom 29. Juli bis zum 02. August der "LMS/EPSRC Short Instructional Course Computational Group Theory St Andrews 2013" statt. Der Kurs hatte zum Ziel, eine Einführung in das Gebiet der algorithmischen Gruppentheorie zu geben.

Es gab vier Hauptvortragende, die jeweils eine vierstündige Minivorlesung hielten. Darüber hinaus fanden praktische Übungen am Computer mit dem Computeralgebrasystem GAP statt, und es gab spezielle Vorträge zu verschiedenen Themen. Sowohl die Folien der Vortragenden als auch die Ãœbungszettel für die Lab Sessions finden sich online unter //www-circa.mcs.st-and.ac.uk/cgt2013/.

Alexander Hulpke (Colorado State University) sprach in seiner Minivorlesung über Permutationsgruppen. Seine Themen beinhalteten den Bahn- und Stabilisatoralgorithmus, den Schreier-Sims-Algorithmus und das O'Nan-Scott-Theorem.

Derek Holt (University of Warwick) behandelte in seiner Minivorlesung das Thema der Matrixgruppen. Unter anderem diskutierte er Straight Line Programs (SLPs), probabilistische Methoden, BSGS-Methoden sowie Aschbachers Satz über Matrixgruppen und den Composition Tree Algorithmus.

Max Neunhöffer (University of St.Andrews) hielt eine Minivorlesung über endlich präsentierte Gruppen, unter anderem über Untergruppen von endlichem Index, Todd-Coxeter, Reidemeister-Schreier, Knuth-Bendix, den Dehn-Algorithmus, Small Cancellation Theorie sowie hyperbolische und automatische Gruppen.

Bettina Eick (TU Braunschweig) trug in ihrer Minivorlesung über auflösbare Gruppen und p-Gruppen vor, in der es um die Themen polyzyklische Präsentationen, Algorithmen für endliche auflösbare Gruppen bzw. unendliche polyzyklische Gruppen und die Klassifikation endlicher p-Gruppen ging.

Die zwei Spezialvorträge wurden von Alexander Konovalov ("Distributed computations with GAP") und Richard Parker (über Small Cancellation Theorie) gehalten. Außerdem gab es zwei zusätzliche, technische Vorträge von Max Neunhöffer ("Objects, Types and Method Selection in GAP") und Alexander Konovalov ("How to write a GAP package").

Abgerundet wurde der Kurs durch ein "Conference Dinner" sowie die bereits erwähnten Lab Sessions, in denen einige der Themen aus den Vorträgen direkt in GAP nachvollzogen werden konnten. Die verschiedenen Aufgaben variierten deutlich in ihrem Schwierigkeisgrad, sodass sich für jeden Teilnehmer passende Aufgaben fanden.

M. Neumann-Brosig (TU Braunschweig)

[10] to be published in Computeralgebra-Rundbrief 54, März 2014

6. Groups St.Andrews 2013

St.Andrews, 29.7. – 11.8.2013

http://www.groupsstandrews.org/2013

Die Konferenzen der Serie "Groups St Andrews" finden alle vier Jahre statt, aber nicht immer in St Andrews. Die neunte Ausgabe kehrte jedoch in den Ursprungsort zurück. Vorgeschaltet fand am selben Ort ein LMS/EPSRC Short Instructional Course über Computational Group Theory statt, in dem die Kursleiter Alexander Hulpke (Colorado), Bettina Eick (Braunschweig), Derek Holt (Warwick) und Max Neunhöffer (St Andrews) den versammelten Doktoranden und Postdoktoranden die neuesten Entwicklungen in der computerunterstützten Gruppentheorie sowie ihre Umsetzung in GAP nahebrachten. Richard Parker ("the man behind the meat axe") und Alexander Konovalov rundeten den Workshop mit Spezialvorträgen ab.

Die eigentliche Konferenz (vom 3.8.-11.8.) besticht durch die Besonderheit, dass es mehrere Vortragsreihen eingeladener Hauptvortragender gibt. In diesem Jahr waren dies Emmanuel Breuillard (Paris), Martin Liebeck (London), Alan Reid (Texas) und Karin Vogtmann (Cornell). Obwohl das ganze Spektrum der Gruppentheorie vertreten war, nahmen viele Vorträge auf explizite Berechnungen und neue algorithmische Methoden bezug. Besonders zu erwähnen sind hier die Beiträge von Alexander Hulpke (über praktische Algorithmen für Matrixgruppen), Leo Margulis, Andreas Bächle und Alexander Konovalov (über die Zassenhaus-Vermutung) sowie Christopher Voll (über Zeta-Funktionen von Gruppen und Ringen).

Mehr als 170 Konferenzteilnehmer freuten sich über die perfekt Organisation und die angenehmen Bedingungen in St Andrews. Für diejenigen, die am Workshop "Computational Group Theory" interessiert waren, aber nicht teilnehmen konnten, sei auf die ausführlichen Materialien auf der Webseite http://www-circa.mcs.st-and.ac.uk/cgt2013/ verwiesen.

Martin Kreuzer (Passau)

7. Parallel Programming in GAP

St.Andrews, 19. – 23.8.2013

http://www.gap-system.org/hpcgap2013

Vom 19. bis 23. August 2013 fand in St Andrews, Schottland, die internationale Arbeitstagung "Parallel Programming in GAPßtatt, die von Steve Linton, Reimer Behrends, Vladimir Janjic, Alexander Konovalov, John McDermott, Angela Miguel und Max Neunhöffer (alle St Andrews) organisiert wurde.

Hauptthema der Tagung war das Programm HPC-GAP, eine in den vergangenen drei Jahren entwickelte multi-threaded Variante des Computeralgebrasystems GAP. In HPC-GAP können, möglicherweise voneinander abhängige, Berechnungen parallel auf mehreren Prozessoren eines Computers durchgeführt werden. Nach einem ersten Überblick von Steve Linton über die Projektentwicklung wurde den 25 Teilnehmern die Erweiterung der GAP Programmiersprache erklärt, sowohl die Auswirkungen auf das Nutzerinterface, wo sich mehrere Threads im gleichen Terminal nutzen lassen,

als auch bezüglich der Implementation eigener Programme. Zusätzliche Schwierigkeiten und typische Stolpersteine Zusammenhang mit der gemeinsamen Speichernutzung der parallelen Rechnungen wurden diskutiert und an Beispielen verdeutlicht. Zudem wurde besprochen, wie existierender GAP Code angepasst werden kann, um problemfrei unter HPC-GAP zu laufen.

In einer Reihe von Vorträgen der Entwickler Reimer Behrends, Alexander Konovalov und Markus Pfeiffer wurde die Umsetzung der Parallelisierung von GAP in verschiedener Detailtiefe beleuchtet. Zur parallelen Berechnung auf verteilten Systemen wurden zudem das auf ParGAP basierende MPIGAP (Vladimir Janjic) und das GAP-Paket SCSCP (Alexander Konovalov) vorgestellt. Anhand dreier Beispiele, des Partitions-Rücksetzverfahren (Markus Pfeiffer), der Berechnung von Bahnen (Max Neunhöffer, Vladimir Janjic) und von Untergruppen von kleinem Index (Michael Torpey), wurde die Theorie mit Leben gefüllt. Abgerundet wurde das Vortragsprogramm durch Beiträge von Teilnehmern, die mathematische Rechenprobleme vorstellten, bei denen sie sich eine effektive Parallelisierung erhofften beziehungsweise eine solche schon durchgeführt hatten.

Im praktischen Teil der Tagung hatten die Teilnehmer die Möglichkeit, sich selbst an der Programmierung in HPC-GAP und der Nutzung von MPIGAP und SCSCP zu versuchen. Über die im offiziellen Programm vorgesehen Zeit hinaus, wurde von einigen Teilnehmern auch zu später Stunde noch motiviert an der Anpassung und Weiterentwicklung von Programmen gearbeitet. Die Veranstaltung schloss mit einer Diskussion über die zukünftige Entwicklung von GAP im Allgemeinen und HPC-GAP im Besonderen.

Zusätzliche Informationen, Folien der Vorträge sowie die während der Tagung aktuelle Version von HPC-GAP finden sich auf der Webseite.

Andreas Distler (TU Braunschweig)

8. Summer School in Algorithmic Mathematics

Hamburg, 2. bis 6. September 2013

`http://www.computeralgebra.de/s2am-2013`

Die „Summer School in Algorithmic Mathematics“ ist eine alljährliche Sommerschule mit Fokus auf die algorithmische Aspekte der algebraischen Geometrie, der algebraischen Zahlentheorie und der Gruppentheorie. Es fanden drei Vortragsreihen statt, sowie viele Teilnehmervorträge.

Claus Fieker (TU Kaiserslautern) gab eine ausführliche Einführung in die algorithmische Zahlentheorie anhand den vier fundamentalen Problemen von, passend zur gastgebenden Universität, Hans Julius Zassenhaus: Gegeben ein Zahlkörper, die Bestimmung des Ganzheitsringes sowie die Berechnung deren Einheitengruppe, Klassengruppe und Galoisgruppe. Wenn auch die Zeit nicht ausreichte, um vollständige Algorithmen zur Lösung vorzustellen, so wurden zumindest zu jedem Problem Lösungsmethoden vorgestellt. Auf exakte Komplexitätsberechnungen wurde verzichtet, zugunsten von kurzen Aussagen über die Praktibilität plus Analyse der Schwächen und Möglichkeiten zur Verbesserung, meistens Gegenstand der aktuellen Forschung.

Anne Frühbis-Krüger (Universität Hannover) gab einen umfangreichen und vor allem anschaulichen Überblick über die algorithmische Singularitätentheorie. Dies beinhaltete eine maßgeschneiderte Einführung in die Gröbner- bzw. Standardbasentheorie, eine Einführung in die Theorie der Singularitäten und deren Invarianten, die Klassifizierung und die Auflösung von Singularitäten. Auf letzteres wurde nach ausdrücklicher Bitte des Publikums näher eingegangen und somit die anfängliche geplante Betrachtung von Varietäten mit vielen Singularitäten übersprungen. Insbesondere wurde eine Übersicht über den aktuellen Stand der Klassifizierung gegeben, sowohl in Charakteristik 0 als auch in positiver Charakteristik.

Alice Niemeyer (RWTH Aachen) stellte das Projekt zur Erkennung von Matrixgruppen vor, welches seit 20 Jahren aktiv ist und zur Zeit die schnellsten Algorithmen zur Behandlung von Gruppen in GAP und Magma liefert. Nach einer kurzen Einführung in Blackbox Gruppen und Straight-Line Programmen gab es eine Kurzübersicht über die Vorgehensweise des Projekts, so wie es zur Zeit in GAP vorzufinden ist, wobei auf ausgewählte Algorithmen näher eingegangen wurde. Interessant war auch die Vorstellung des ranked methods Konzepts in GAP von Max Neunhöfer, zum Umgang mit vielen potentiellen Algorithmen mit je unterschiedlichen Stärken und Schwächen. Außerdem gab es einen Einblick in die Berechnung von Proportionen von Gruppenelementen mit speziellen Eigenschaften.

Darüber hinaus haben von den Teilnehmern vorgetragen: Oliver Braun über die Berechnung der maximalen Untergruppen modulo Konjugation der Automorphismengruppe eines Gitters; David Dursthoff über die Berechnung von Hilbert Modulformen über reellen Zahlkörpern; Jens Eberhardt über das Finden einer Linearisierung eines gegebenen Matroids; Michal Farnik über ein Theorem von Kleinman zur Vermutung Chevalley's auf torischen Varietäten; Sebastian Gutsche über den Kern des Garbifizierungsfunktors auf torischen Varietäten; Tommy Hofmann über das Finden von ganzzahligen Darstellungen in einer gegebenen Konjugationsklasse; Marta Pieropan über das Zählen von rationalen Punkten auf torischen Varietäten und einen alternativen Beweis von Manin's Vermutung; Sebastian Schönnenbeck über die Homologiegruppen der Einheitengruppe von Ordnungen.

Die Sommerschule wurde gesponsort vom Schwerpunktprogramm 1489 der Deutschen Forschungsgemeinschaft und organisiert von Jan Steffen Müller (Universität Hamburg) und Yue Ren (TU Kaiserslautern). Mehr Informationen zur Sommerschule in Hamburg, insbesondere die Zusammenfassungen und die Folien der Teilnehmervorträge, finden Sie auf der offiziellen Homepage.

Yue Ren (TU Kaiserslautern)

9. The 15th International Workshop on Computer Algebra and Scientific Computing (CASC 2013)

Berlin, 9.–13. September 2013

`http://www14.in.tum.de/CASC2013/`

The development of powerful computer algebra systems has considerably extended the scope of problems of scientific computing which can now be solved successfully with the aid of computers. This stimulated the research in the field of computer algebra (CA) both in Western Europe and in the former Soviet Union (fSU). However, as the field of applications of computer algebra in scientific computing becomes

broader and more complex, there is a danger of separation between theory, systems, and applications. For this reason, there was an increasing need in bringing together the researchers who work in these areas. That's why it was decided to establish the International Workshop on Computer Algebra in Scientific Computing (CASC) and to hold it alternatingly in CIS countries and in Germany.

An important point confirming the success of the CASC Workshops is that the proceedings of most of these workshops have been published by Springer-Verlag, which is among the top ten world publishers of scientific literature. One reason for the cooperation between Springer-Verlag and CASC is that the CASC proceedings have been one of the top 50 % most downloaded ebooks in the relevant Springer eBook collection in the years from 2009 to 2012. Such a success is partially due to a thorough reviewing process so that each submitted paper receives from two to five reviews, and these reviews contribute to a further improvement of the quality of the final versions of accepted papers.

The *15th International Workshop on Computer Algebra in Scientific Computing* was held at the Konrad Zuse Institute Berlin (ZIB) from September 9 to 13, 2013, and it was organized very well by the local organizer Winfried Neun.

The delegation from the fSU was the most numerous at the 15th CASC Workshop. According to Russian tradition, anniversaries having numbers $= 5 \bmod 10$ are considered nearly as good round numbers as those divisible by 10. The 15th anniversary of the CASC Workshops marks fifteen years of successful work of this series of International Workshops devoted to CA and its applications in scientific computing all of which were organized by the General Chair Ernst Mayr.

Polynomial algebra, which is at the core of computer algebra, was represented by contributions devoted to the complexity of solving systems of polynomial equations with small degrees, highly scalable multiplication of distributed sparse multivariate polynomials on many-core systems, fast approximate polynomial evaluation and interpolation, application of Groebner bases for mechanical theorem proving in geometry, application of quantifier elimination for determining whether a univariate polynomial satisfies the sign definite condition, solution of polynomial systems with approximate complex-number coefficients with the aid of a polyhedral algorithm, the solution of a problem of interpolating a sparse, univariate polynomial with the aid of a recursive algorithm using probes of smaller degree than in previously known methods, computation of limit points of the quasi-component of a regular chain with the aid of Puiseux series expansion, solution of a system of polynomial equations as part of algebraic cryptoanalysis by reducing to a mixed integer linear programming problem, an improved QRGCD algorithm for computing the greatest common divisor of two univariate polynomials, construction of classes of irreducible bivariate polynomials.

A number of papers included in the proceedings of the CASC 2013 Workshop are devoted to using computer algebra for the investigation of various mathematical and applied topics related to ordinary differential equations (ODEs): computing divisors and common multiples of quasi-linear ordinary differential equations, investigation of local integrability of the ODE systems near a degenerate stationary point, the computation of the dimension of the solution space of a given full-rank system of linear ODEs, application of symbolic calculations and polynomial invariants to the classification of singularities of planar polynomial systems of ODEs, the use of Vessiot's vector field based approach for an analysis of geometric singularities of ODEs.

Several papers dealt with applications of symbolic computations for solving partial differential equations (PDEs) in mathematical physics. In one of them, a general symbolic framework is described for boundary problems for linear PDEs. The methods of computer algebra are used intensively in the other two papers for deriving new methods for the numerical solution of two- and three-dimensional viscous incompressible Navier-Stokes equations.

A number of papers were devoted to the applications of symbolic and symbolic-numeric algorithms in mechanics and physics: the investigation of gyrostat satellite dynamics, modelling of identical particles with pair oscillator interactions, tunneling of clusters through repulsive barriers, application of CAS Maple for investigating a quantum measurements model of hydrogen-like atoms, development of efficient methods to compute the Hopf bifurcations in chemical networks with the aid of the package REDLOG, which is an integral part of CAS Reduce, determination of stationary points for the family of Fermat-Torricelli-Coulomb-like potential functions, the determination of stationary sets of Euler's equations on the Lie algebra with the aid of CASs MAPLE and MATHEMATICA.

The other topics included the application of the CAS MATHEMATICA for the simulation of quantum error correction in quantum computing, the application of the CAS GAP for the enumeration of Schur rings over the group A_5, constructive computation of zero separation bounds for arithmetic expressions, the parallel implementation of fast Fourier transforms with the aid of the SPIRAL library generation system, the use of object-oriented languages such as Java or Scala for implementation of categories as type classes, a survey of industrial applications of approximate computer algebra, *i.e.*, algebraic computation of expressions with inaccurate coefficients represented by floating-point numbers.

(Excursion in front of the New Palais)

Two invited talks dealt with a number of problems arising in the theory of ordinary and partial differential equations. In particular, T. Wolf in his talk considered the application of computer algebra methods for finding infinitesimal symmetries, first integrals or conservation laws, Lax-pairs, etc. when investigating the integrability of PDEs or ODEs. The invited talk by A. Griewank was concerned with methods, algorithms, software for, and some history about the field of automatic differentiation, highlighting original developments and the use of adjoints. And, finally, D. Grigoriev in his invited talk surveyed the complexity results concerning the solution of tropical linear systems and tropical polynomial systems.

(Castle Sanssouci)

The program of the CASC 2013 may be found at the

web site http://wwwmayr.in.tum.de/CASC2013/index.php/schedule

The scientific program of the CASC 2013 Workshop was combined by the local organizers in an excellent way with the cultural program including an exciting excursion to Potsdam. We walked through the park Sanssouci — which was very nice although we couldn't avoid a short rain shower — visited the Russian Colony Alexandrowka and participated in an organized tour through Cecilienhof Palace. The excursion ended with the conference dinner which took place at the Meierei, a nicely situated nearby restaurant and brewery.

(Alexandrowka)

(Cecilienhof Palace)

In a second excursion most participants visited the Reichstag cupola.

The online version of the Proceedings (LNCS 8136) is available at http://link.springer.com/book/10.1007/978-3-319-02297-0/

(Photos: Wolfram Koepf)

Wolfram Koepf (Kassel)
Evgenii V. Vorozhtsov (Novosibirsk)

(Conference Photo CASC 2013 in front of the Konrad Zuse Institute Berlin)

Hinweise auf Konferenzen

1. **Darstellungstheorietage und Nikolaus-Konf.**

   Aachen, 5. – 7.12.2013

   www.math.rwth-aachen.de/Nikolaus2013/

   Die Darstellungstheorietage finden dieses Jahr gemeinsam mit der Nikolaus-Konferenz in Aachen statt. Das Vortragsprogramm beginnt am Donnerstag um ca. 14 Uhr und endet am Samstag um ca. 18 Uhr. Dabei sind der Donnerstagnachmittag und Freitagvormittag der Darstellungstheorie gewidmet. Am Freitagnachmittag und Samstag schließt sich die traditionelle Nikolaus-Konferenz an, die jährlich am Lehrstuhl D für Mathematik der RWTH Aachen stattfindet. Die Nikolaus-Konferenz hat das Ziel Leute zusammenzubringen, die kürzlich eine Abschlussarbeit im Bereich Gruppen- oder Darstellungstheorie angefertigt haben, mit solchen, die in diesem Bereich bereits länger Erfahrung gesammelt haben. Besonders interessiert sind die Teilnehmer an Berichten über Projekte, die mit Algorithmen und Rechnen zu tun haben.

2. **Der XV. Mathematica-Tag**

   Berlin, 10.12.2013

   www.ordinate.de/mathematicaTag.htm

   ma*themas* ordinate organisert regelmäßig seit 1999 den Berliner Mathematica-Tag. Es trafen und treffen sich dabei an Mathematica Interessierte. Ziel ist vor allem auch ein Erfahrungsaustausch der Teilnehmer.

3. **Tagung der Fachgruppe Computeralgebra**

   Kassel, 15. – 17.05.2014

   www.fachgruppe-computeralgebra.de/TagungKassel

   In Fortsetzung der erfolgreichen Tagungen 2003, 2005, 2009, 2012 in Kassel und 2007 in Kaiserslautern führt die Fachgruppe im Mai 2014 wieder eine derartige Tagung in Kassel durch. Ziel ist es, wie auf den Vorgängerkonferenzen ein Forum zu bieten, das es erstens Nachwuchswissenschaftlern ermöglicht, ihre Ergebnisse vorzustellen, andererseits aber auch einige Hauptvortragende zu gewinnen, die Übersichtsvorträge über wichtige Gebiete der Computeralgebra und über Computeralgebra-Software geben sollen.

4. **ACA 2014 – 20th Conference on Applications of Computer Algebra**

   New York, 9. – 12.07.2014

   faculty.fordham.edu/rlewis/aca2014

   This is the twentieth in a series of annual international meetings devoted to promoting the applications and development of Computer Algebra and Symbolic Computation. Topics include computer algebra and symbolic computation in engineering, the sciences, medicine, pure and applied mathematics, education, communication and computer science. ACA 2014 will be held July 9-12, 2014 at Fordham University, New York City. The first ACA was held at the University of New Mexico in Albuquerque, New Mexico, USA, in 1995. The 2013 conference was held in Malaga, Spain.

5. **ISSAC 2014 – International Symposium on Symbolic and Algebraic Computation**

   Kobe University, Japan, 23. – 25.07.2014

   www.issac-conference.org/2014/

   The International Symposium on Symbolic and Algebraic Computation is the premier conference for research in symbolic computation and computer algebra. ISSAC 2014 is the 39th meeting in the series. The conference traditionally presents a range of invited speakers, tutorials, poster sessions and software demonstrations with a centre-piece of contributed research papers.

   ISSAC 2014 is held July 23-25, 2014 at Kobe University, Japan. Workshops and Tutorial Sessions will be held prior to the conference, between July 21-22, 2014.

   ISSAC 2014 is affiliated with "Kobe Computing Week 2014", an event of Academic Exchange Weeks, Graduate School of Human Development and Environment, Kobe University. ISSAC 2014 is a satellite conference of ICM 2014 (International Congress of Mathematicians), Korea. Also, SNC 2014 (Symbolic-Numeric Computation), Shanghai, China, is a satellite conference of ISSAC 2014.

6. **SNC 2014 – International Conference on Symbolic Numeric Computation**

   Shanghai, China,
   28. – 31.07.2014

   symbolic-numeric-computation.org/snc-2014

   Algorithms that combine techniques from symbolic and numeric computation have been of increasing importance and interest over the past decade. The necessity for a mathematically sound framework for algebraic problems in an approximate setting, to work reliably with imprecise or noisy data, and for speed and accuracy in hybrid symbolic-numeric problems have fuelled a new synergy between the numeric and symbolic computing fields. The SNC meetings explore new algorithms in which symbolic and numeric methods are combined to do more than can be done with either alone.

   The goal of the present conference is to support the interaction and integration of symbolic and numeric computing. Earlier meetings in this series include the SNAP 96 Workshop, held in Sophia Antipolis, France, the SNC 2005 meeting, held in Xi'an, China, SNC 2007 which was held in London, Canada, SNC 2009, held in Kyoto, Japan and SNC 2011, held in San Jose, USA.

   This 5th international conference on Symbolic-Numeric Computation will be held on July 28-31 at East China Normal University in Shanghai, China. This conveniently follows the nearby International Symposium on Symbolic and Algebraic Computation (ISSAC 2014), held in Kobe, Japan.

7. **ANTS-XI – Eleventh Algorithmic Number Theory Symposium**

   Hotel Hyundai, GyeongJu, Korea,
   7. – 11.08.2014

   ants2014.kookmin.ac.kr

   The ANTS meetings, held biannually since 1994, are the premier international forum for new research in computational number theory. They are devoted to algorithmic aspects of number theory, including elementary number theory, algebraic number theory, analytic number theory, geometry

of numbers, arithmetic algebraic geometry, finite fields, and cryptography.

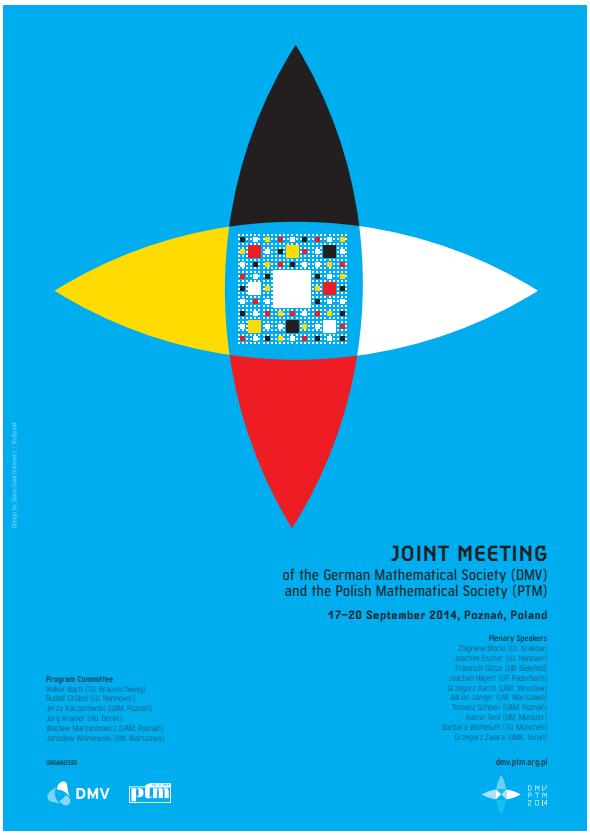
8. CASC 2014 — 16th International Workshop on Computer Algebra in Scientific Computing

Warsaw, Poland, 8. – 12.9.2014

`www14.in.tum.de/CASC2014/`

The methods of Scientific Computing play an important role in the natural sciences and engineering. Significance and impact of computer algebra methods and computer algebra systems for scientific computing have increased considerably over the last decade. The topics addressed in the CASC workshops cover all the basic areas of scientific computing as they benefit from the application of computer algebra methods and software.

The 16th International Workshop on Computer Algebra in Scientific Computing (CASC'2014) will be held in Warsaw, Poland, from September 8 to 12, 2014. The Local Arrangements Chair is Alexander Prokopenya.



9. PTM-DMV-Jahrestagung 2014

Poznan, Polen, 17. – 20.09.2014

`dmv.ptm.org.pl`

Die DMV-Jahrestagung findet 2014 gemeinsam mit den Kolleginnen und Kollegen der Polnischen Mathematischen Gesellschaft (PTM) vom 17. bis 20. September 2014 in Poznan statt. Auf der Website der Konferenz kann man sich ab jetzt für die Teilnahme registrieren. Auf der kommenden Jahrestagung wird es keine Sektionen geben, sondern ausschließlich Minisymposien (Thematic Sessions). Interessent(inn)en sind herzlich eingeladen die Organisation eines Minisymposiums zu übernehmen, sodass die Mathematik wieder in ihrer ganzen Breite dargestellt wird. Vorschläge für die Sessions können bis zum 31. Dezember 2013 auf der oben genannten Webseite eingereicht werden.

10. INFORMATIK 2014 — 44. Jahrestagung der Gesellschaft für Informatik

Stuttgart, 22. – 26.9.2014

`www.informatik2014.de`

Im BMBF-Wissenschaftsjahr 2014 „Die digitale Gesellschaft" steht die GI-Jahrestagung unter dem Motto „Big Data – Komplexität meistern".

Big Data charakterisiert die Datenflut, deren Bewältigung uns überall beschäftigt: im Internet, in der Kommunikationsbranche, bei der medizinischen Bildgebung, der Simulation von Crashtests, den Special Effects in der Filmindustrie, der Datenintegration für Fahrerassistenzsysteme, der Marktforschung, kurz, in fast allen Bereichen, in denen die Informatik Impulse für neue Funktionalität gegeben hat.

Jenseits der Herausforderungen, die mit Big Data einhergehen, kennzeichnet eine explodierende Komplexität der Anwendungen viele der heutigen Probleme der IT-Branche. Schlagworte wie Cloud Computing, Industrie 4.0, Smart Factory, Smart Enterprise oder auch Smart Mobility zeichnen eine Zukunft, in der durch das Zusammenwirken vieler Hard- und Softwarekomponenten eine Intelligenz der Systeme entsteht, die weit über das Bekannte hinausgehen wird. Voraussetzung ist allerdings, dass es gelingt, die dabei entstehende Komplexität dieser Systeme zu meistern.

Beide Aspekte, die Komplexität der Daten und die Komplexität der Systeme und Anwendungen, werden das Leitthema der Jahreskonferenz INFORMATIK 2014 sein.

Kurze Mitteilungen

Algorithmische und experimentelle Methoden in Algebra, Geometrie und Zahlentheorie – DFG-Schwerpunktprogramm SPP1489 geht in die zweite Runde

Wolfram Decker
Technische Universität Kaiserslautern

decker@mathematik.uni-kl.de

Im Rahmen des oben genannten Schwerpunktprogramms hat die Deutsche Forschungsgemeinschaft Mitte des Jahres die Bewilligungsschreiben für die Projekte der zweiten Periode verschickt. Kurzdarstellungen der geförderten Projekte und weitere Informationen zum Programm finden sich hier:

http://www.computeralgebra.de

Im Rahmen seiner Aktivitäten organisiert der Schwerpunkt ein umfangreiches Programm an Schulen, Workshops und Konferenzen. Dabei sind Gäste von außerhalb des Schwerpunkts herzlich willkommen. Das gilt insbesondere für die Schulen des Typs S^2AM, die von einem Kommittee junger Nachwuchsforscher für junge Nachwuchsforscher organisiert werden, und ein hervorragendes Forum zum gegenseitigen Kennenlernen bieten. Die **nächste Jahrestagung** des Schwerpunkts wird vom 3. bis 7. Februar 2014 in Bad Boll stattfinden. Für detaillierte Informationen zu den Tagungsaktivitäten sei auf die oben genannte Webpage verwiesen.

Über neue rechnerische Möglichkeiten, die sich aus der Interaktion von verschiedenen Computeralgebrasystemen ergeben sowie über die Distributionsplattform LMONADE für wissenschaftliche Software (siehe http://www.computeralgebra.de) werden wir ausführlich in der nächsten Ausgabe des Computeralgebrarundbriefs berichten.

Berufungen

Prof. Dr. Peter Bürgisser (Paderborn) hat 2013 eine Professur an der TU Berlin angetreten.
(http://www.math.tu-berlin.de/fachgebiete_ag_diskalg/fachgebiet_algebra_zahlentheorie/v-menue/research_group/prof_dr_peter_buergisser)

Dr. Michael Cuntz (Kaiserslautern) hat 2013 eine Professur (Diskrete Mathematik) an der Leibniz Universität Hannover angetreten.
(http://www.iazd.uni-hannover.de/~cuntz)

Prof. Dr. Michael Joswig (Darmstadt) hat 2013 eine Professur (Discrete Mathematics/Geometry) an der TU Berlin angetreten.
(http://page.math.tu-berlin.de/~joswig)

Leserbriefe

Stellungnahme zu

„Computeralgebra in der Schule - Stand der Dinge!?“

Ausgabe 50, März 2012

Es ist zwar über ein Jahr her, dass der Artikel, auf den ich mich hier beziehe, im CA-Rundbrief zu lesen war. Aber das Thema hat inzwischen nichts von seiner Aktualität verloren. Zunächst habe ich noch einmal die Passagen des Aufsatzes herausgeschrieben, die ich für besonders wichtig halte:

> Das CAS kann beim Erwerb algebraischer Kompetenzen eingesetzt werden. Dazu müssen besonders in der Sekundarstufe I neue Konzepte entwickelt werden.
> Es fehlt aber noch an konkreten und evaluierten Unterrichtsszenarien für den CAS-Einsatz beim Problemlösen. Ebenso fehlen Kriterien für die Erstellung von geeigneten Aufgaben für den CAS-Einsatz in Unterricht und Prüfungen.
> Wir benötigen also didaktisch reflektierte und methodisch durchdachte Lernumgebungen für die neuen Technischen Möglichkeiten, ... Dazu gehört schließlich auch die Sensibilisierung der Lernenden für die Entscheidung, was sie noch im Kopf berechnen und lösen können sollten und was sie dem CAS überlassen.

Hier werden Notwendigkeiten für die nahe Zukunft – eigentlich schon Versäumnisse der jüngsten Vergangenheit – genannt. Man braucht keine seherischen Fähigkeiten, um zu ahnen, dass die hier formulierten Postulate mittelfristig bei den Bildungsbehörden ungehört verhallen werden. Die Erfahrungen mit der verbindlichen Einführung des Taschenrechners lassen nichts Gutes erwarten. Bis heute fehlen wissenschaftlich begründete, praktisch untermauerte Konzepte, Kriterien und evaluierte Unterrichtsszenarien für den Einsatz von GTR beim Lernen von Mathematik. Jeder Lehrer hat sein eigenes Konzept. Manches mag erfolgreich sein – vieles ist es nicht, sondern eher kontraproduktiv. Es ist offenbar immer noch nicht allen damit Befassten klar geworden, dass Mathematik-Lernen einen anderen Werkzeuggebrauch – vielleicht sogar andere Werkzeuge – erfordert, als Mathematik-Treiben.

Und: Man kann nichts ernsthaft und erfolgreich betreiben, wenn nicht zuvor einiges erlernt wurde. Werkzeuggebrauch setzt Wissen voraus – Wissen über das Werkzeug und – nicht zu vergessen – Wissen über das Werkstück, die Mathematik.

Nun zum Konkreten: Die Aufgabe zu den seltsamen Zahlen ist hochgradig attraktiv, besonders auf Grund folgender Merkmale:

- Sie regt dazu an, gegenüber maschinell erzeugten Ergebnissen skeptisch zu bleiben.
- Sie zeigt, dass Computer zwar bei Problemlösungen helfen können aber Fertigkeiten von Hand und im Kopf unverzichtbar bleiben.

Das Arbeitsblatt zu dieser Aufgabe (die Seite 21) muss aber auch kritisch gesehen werden:

- Die Schülerinnen und Schüler werden über 9 Stationen geschoben und gezogen. Sie können und sollen zwar jeden Schritt nachvollziehen aber erst gegen Schluss wird klar, wozu die anfänglichen Schritte gut waren.
- Die Problemlösung kommt als fertiges Phänomen daher, es gibt kaum etwas zu entdecken.
- Auch der vorgeschlagene Umgang mit einem CAS ist überwiegend auf Nachvollziehen fertiger Angebote beschränkt.

Als Alternative schlage ich ein Arbeitsblatt vor, das zwar den Lösungsweg ebenfalls vorstrukturiert, aber mehr Freiheiten zum selbständigen Arbeiten und Entdecken lässt:

Arbeitsblatt
Aufgabe 1:
a) Berechnen Sie die Zahl mit dem Taschenrechner. Vertrauen Sie dem Ergebnis? Könnte ein Rundungs- oder Rechenfehler vorliegen?
b) Berechnen Sie die Zahlen x und y mit dem Taschenrechner

$$\begin{aligned} x &= (1 + 10^{-21} + 10^{-21} - 1) \cdot 10^{22}, \\ y &= (1 - 1 + 10^{-21} + 10^{-21}) \cdot 10^{22}. \end{aligned}$$

Vertrauen Sie den Ergebnissen? Machen Sie eine Probe im Kopf und von Hand.
Aufgabe 2:
Berechnen Sie $(a \pm \sqrt{b})^3$ mit CAS. Begründen Sie mit Hilfe des Ergebnisses die Formel (*). Berechnen Sie mit dieser Formel den Term. Erfinden Sie weitere Terme der Art, die sich zu einer natürlichen Zahl vereinfachen lassen und in denen m, n, und k natürliche Zahlen sind.
Aufgabe 3: Zeigen Sie, dass die Formel (*) in im Falle der Aufgabe 1.a) versagt, nicht aber die Formel (**). Begründen Sie zunächst kurz die Gültigkeit der Formel (**) und wenden Sie diese dann auf Aufgabe 1.a) an.

Erfinden Sie selbst Terme der Art, die sich zu einer natürlichen Zahl vereinfachen lassen, in denen m, n sowie k natürliche Zahlen sind und die den Einsatz der Formel (**) ermöglichen bzw. erfordern.

Roland Schröder (Celle)

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 201__ 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:

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

Fachgruppenleitung Computeralgebra 2011-2014

Sprecher:
Prof. Dr. Florian Heß
Carl-von Ossietzky Universität Oldenburg
Institut für Mathematik, 26111 Oldenburg
0441-798-2906, -3004 (Fax)
florian.hess@uni-oldenburg.de
http://www.staff.uni-oldenburg.de/florian.hess

Fachreferentin Publikationen und Promotionen:
Prof. Dr. Anne Frühbis-Krüger
Institut für Algebraische Geometrie
Welfengarten 1, 30167 Hannover
0511-762-3592
fruehbis-krueger@math.uni-hannover.de
http://www.iag.uni-hannover.de/~anne

Fachreferent 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

Fachexperte Industrie:
Prof. Dr. Michael Hofmeister†
Siemens AG
Corporate Technology
Modeling, Simulation, Optimization
Otto-Hahn-Ring 6, 81739 München
089-636-49476, -42284 (Fax)
michael.hofmeister@siemens.com
http://www.siemens.com

Fachreferentin Computational Engineering, Vertreterin der GAMM:
Dr.-Ing. Sandra Klinge
Lehrstuhl für Mechanik - Materialtheorie
Ruhr-Universität Bochum
Universitätsstr. 150, 44780 Bochum
0234-32-26552, -14154 (Fax)
sandra.klinge@rub.de
www.am.bi.ruhr-uni-bochum.de/Mitarbeiter/Ilic

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

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

Fachexperte Schule:
OStR Jan Hendrik Müller
Rivius-Gymnasium der Stadt Attendorn
Westwall 48, 57439 Attendorn
02722-5953 (Sekretariat)
jan.mueller@math.uni-dortmund.de
www.mathebeimueller.de

Redakteur Rundbrief:
Prof. Dr. Michael Cuntz
Institut für Algebra, Zahlentheorie und Diskrete Math.
Leibniz Universität Hannover
Welfengarten 1, 30167 Hannover
0511-762-4252
cuntz@math.uni-hannover.de
http://www.iazd.uni-hannover.de/~cuntz

Stellvertretende Sprecherin:
Prof. Dr. Eva Zerz
Lehrstuhl D für Mathematik
RWTH Aachen
Templergraben 64, 52062 Aachen
0241-80-94544, -92108 (Fax)
eva.zerz@math.rwth-aachen.de
http://www.math.rwth-aachen.de/~Eva.Zerz/

Fachexperte Lehre und Didaktik:
Prof. Dr. Gilbert Greefrath
Westfälische Wilhelms-Universität Münster
Institut für Didaktik der Mathematik und der Informatik
Fliednerstr. 21, 48149 Münster
0251-8339396
greefrath@uni-muenster.de
http://www.greefrath.de

Fachreferentin Fachhochschulen:
Prof. Dr. Elkedagmar Heinrich
Fachbereich Informatik, Hochschule für Technik, Wirtschaft und Gestaltung Konstanz
Brauneggerstr. 55, 78462 Konstanz
07531-206-343, -559 (Fax)
heinrich@htwg-konstanz.de
http://www.in.fh-konstanz.de/inhalte/de/KONTAKT/persseiten_nbc/heinrich.html

Fachreferent CA-Systeme und -Bibliotheken:
Prof. Dr. Gregor Kemper
Zentrum Mathematik – M11
Technische Universität München
Boltzmannstr. 3, 85748 Garching
089-289-17454, -17457 (Fax)
kemper@ma.tum.de
http://www-m11.ma.tum.de/~kemper

Fachreferent Schwerpunktprogramm 1489:
Prof. Dr. Jürgen Klüners
Mathematisches Institut der Universität Paderborn
Warburger Str. 100, 33098 Paderborn
05251-60-2646, -3516 (Fax)
klueners@math.uni-paderborn.de
http://www2.math.uni-paderborn.de/people/juergen-klueners.html

Fachreferent Themen und Anwendungen:
Prof. Dr. Martin Kreuzer
Fakultät für Informatik und Mathematik
Universität Passau
Innstr. 33, 94030 Passau
0851-509-3120, -3122 (Fax)
martin.kreuzer@uni-passau.de
http://www.fim.uni-passau.de/~kreuzer

Vertreter der GI:
Prof. Dr. Ernst W. Mayr
Lehrstuhl für Effiziente Algorithmen
Fakultät für Informatik
Technische Universität München
Boltzmannstraße 3, 85748 Garching
089-289-17706, -17707 (Fax)
mayr@in.tum.de
http://www.in.tum.de/~mayr/

Koordinator Internetauftritt:
Prof. Dr. Hans-Gert Gräbe
Institut für Informatik
Universität Leipzig
Postfach 10 09 20, 04009 Leipzig
0341-97-32248
graebe@informatik.uni-leipzig.de
http://www.informatik.uni-leipzig.de/~graebe

Redakteurin Rundbrief:
Dr. Gohar Kyureghyan
Otto-von-Guericke Universität Magdeburg
Institut für Algebra und Geometrie
Universitätsplatz 2, 39106 Magdeburg
0391-67-11650, -11213 (Fax)
gohar.kyureghyan@ovgu.de
http://fma2.math.uni-magdeburg.de/~gkyureg

Werbeseite