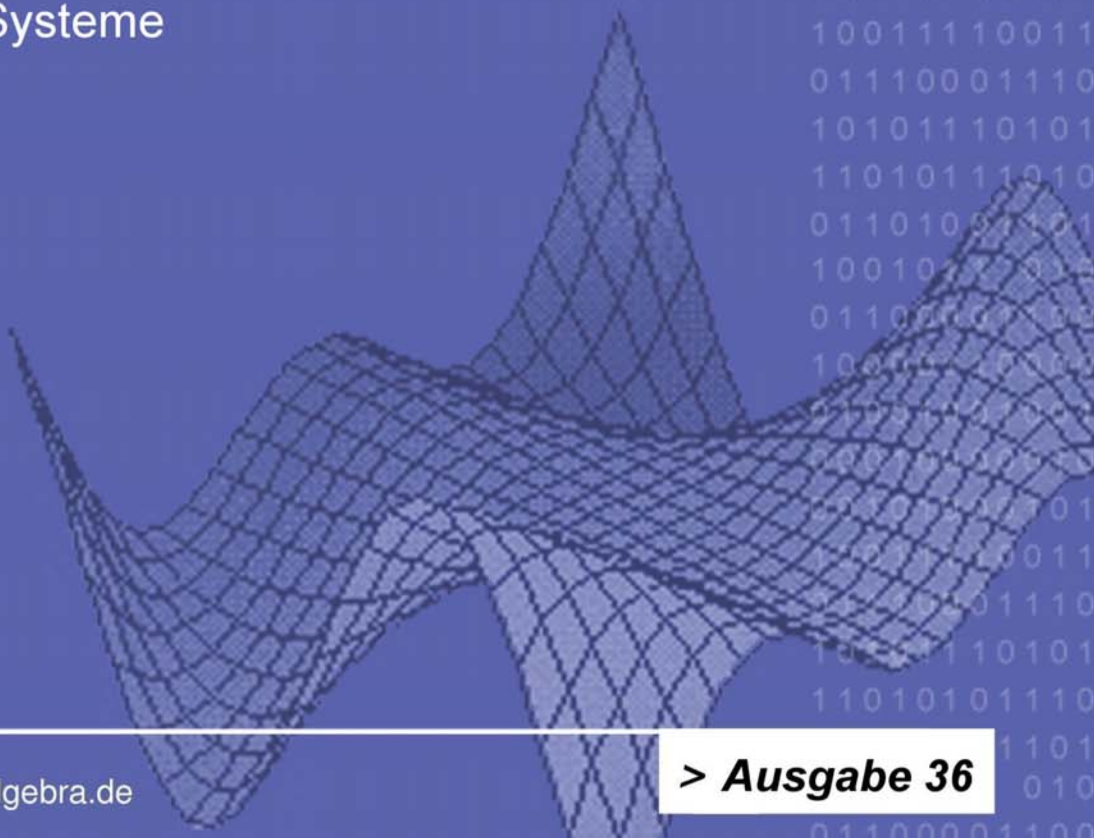
März 2005

# Computeralgebra Rundbrief

GI_DMV_GAMM

- Neue Fachgruppenleitung
- Tagungen der Fachgruppe in Kassel und Schönenberg
- Computeralgebra in der Theoretischen Physik
- Mechatronische Systeme



www.fachgruppe-computeralgebra.de

> Ausgabe 36

Computeralgebra-Rundbrief

Nr. 36 März 2005

Fachgruppe Computeralgebra

## Inhalt

## Impressum

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

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

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

Die Geschäftsstellen der drei Trägergesellschaften:

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

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

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

## Mitteilungen der Sprecher

*Liebe Mitglieder der Fachgruppe Computeralgebra,*

*am 18. Februar fand die letzte Sitzung der alten Fachgruppenleitung und im Anschluss daran die konstituierende Sitzung der neuen Fachgruppenleitung in Göttingen statt. Im Auftrag des Wahlleiters Prof. Dr. Hantzschmann gab der Sprecher das Ergebnis der Wahl bekannt. Es beteiligten sich 74 Mitglieder an der Wahl und gaben insgesamt 488 Stimmen ab (maximal 9 pro Wähler), die sich folgendermaßen verteilten:*

| | | |
|---|---|---|
| *Prof. Dr. Wolfram Koepf* | *Kassel* | *50* |
| *Prof. Dr. Bettina Eick* | *Braunschweig* | *39* |
| *Heiko Knechtel* | *Bückeburg* | *37* |
| *Prof. Dr. Gunter Malle* | *Kaiserslautern* | *34* |
| *Dr. Andreas Sorgatz* | *Paderborn* | *34* |
| *Prof. Dr. Gerhard Hiß* | *Aachen* | *33* |
| *Prof. Dr. Ulrich Kortenkamp* | *Berlin* | *31* |
| *Prof. Dr. Hans-Wolfgang Henn* | *Dortmund* | *30* |
| *Prof. Dr. Elkedagmar Heinrich* | *Konstanz* | *29* |

---

| | | |
|---|---|---|
| *Prof. Dr. Hans-Gert Gräbe* | *Leipzig* | *22* |
| *Dr. Ulrich Schwardmann* | *Göttingen* | *22* |
| *Prof. Dr. Gerd Baumann* | *Kairo* | *21* |
| *Dr. Thomas Hahn* | *München* | *19* |
| *Prof. Dr. Reinhard Laue* | *Bayreuth* | *19* |
| *Dr. Reinhard Oldenburg* | *Göttingen* | *19* |
| *Dr. Hans Schönemann* | *Kaiserslautern* | *19* |
| *Prof. Dr. Florian Heß* | *Berlin* | *17* |
| *Prof. Dr. Martin Kreuzer* | *Dortmund* | *13* |

*Damit sind die ersten neun Kandidaten gewählt. Die weiteren neun Kandidaten stehen in der Reihenfolge der Stimmenverteilung auf der Nachrückliste. Wir begrüßen sehr herzlich die Neugewählten Frau Prof. Dr. Eick und Frau Prof. Dr. Heinrich in unserer Mitte!*

*Mit den neu benannten bzw. bestätigten Vertretern der Trägergesellschaften, den Herren*

*Johannes Grabmeier (GI), B. Heinrich Matzat (DMV), Klaus Hackl (GAMM),*

*und den gemäß unserer Ordnung berufenen Fachexperten, den Herren*

*Thomas Hahn (Physik), Reinhard Laue (Chemie), Markus Wessler (Rundbrief)*

*ist die neue Fachgruppenleitung für die Amtszeit 2005–2008 komplett. Wir danken den nicht gewählten Kandidaten für ihre Bereitschaft, sich zur Wahl zu stellen und sich auch weiterhin für die Belange der Fachgruppe Computeralgebra einzusetzen. Gleicher Dank gilt dem Wahlleiter, K. Hantzschmann, und seinem Stellvertreter, B. H. Matzat, für die Durchführung der Wahl. Unser Dank gilt schließlich auch den ausscheidenden Mitgliedern K. Hantzschmann, H. M. Möller, U. Schwardmann und W. Werner für die geleistete Arbeit sowie für Ihre Bereitschaft, der Fachgruppe auch weiterhin mit Rat und Tat zur Verfügung zu stehen. Mit ihnen sind Kollegen aus der Fachgruppenleitung ausgeschieden, die lange die Entwicklung der Fachgruppe maßgeblich geprägt haben.*

*Prof. Dr. Karl Hantzschmann tritt demnächst in den Ruhestand und scheidet daher aus eigenem Wunsch aus der Position des GI-Vertreters in der Fachgruppenleitung aus. Er war seit 1992 Mitglied der Fachgruppenleitung, in den letzten Jahren als offizieller Vertreter der GI. Besonders hervorzuheben ist die exzellente Vorbereitung und Durchführung der ISSAC-Tagung in Rostock 1998 durch ihn als Local Arrangement Chair. Wir wünschen ihm für seinen Ruhestand alles Gute.*

*Prof. Dr. H. Michael Möller scheidet ebenfalls auf eigenen Wunsch aus der Fachgruppenleitung aus. Er war seit 1994 Mitglied der Fachgruppenleitung und in den Jahren 1999–2002 Sprecher der Fachgruppe. In seiner Amtszeit hatte der Wechsel der Druckerei für den Rundbrief eine spürbare Ersparnis gebracht. Schwerpunkte seiner Amtszeit waren die Benchmark-Aktivitäten der Fachgruppe sowie die Aktivitäten zum Thema „Computeralgebra in Lehre, Ausbildung und Weiterbildung“, welche sich vor allem in den von der Fachgruppe organisierten Tagungen in Thurnau und in Schöntal manifestierten. Diese Aktivitäten stehen weiterhin auf der Agenda der Fachgruppe, hierzu mehr auf Seite 10. Unser herzlicher Dank geht auch an Herrn Möller.*

*Dr. Ulrich Schwardmann war unser langjähriger Rundbrief-Redakteur (1994–2001) sowie Webmaster (seit 1997). Er betreute zunächst das CAIS (Computeralgebra Informationssystem) und setzte später die vom Sprecher vor drei Jahren initiierte Neugestaltung unserer Webseite* `http://www.fachgruppe-computeralgebra.de` *um. Er war seit 1995 Mitglied der Fachgruppenleitung. Aufgrund von Umstrukturierungsmaßnahmen bei der GWDG steht die Computeralgebra inzwischen weniger im Mittelpunkt seiner beruflichen Tätigkeit, und Herr Schwardmann steht daher für die genannten Aktivitäten nun nicht mehr zur Verfügung. Herzlichen Dank für seine tatkräftige Hilfe!*

*Auch Prof. Dr. Wilhelm Werner scheidet auf eigenen Wunsch aus der Fachgruppenleitung aus. Er hat sich bereits vor seinem Eintritt in die Fachgruppenleitung aktiv für die Belange der Fachgruppe eingesetzt. Er war der Initiator einer Tagungsreihe zum Thema „Computeralgebra an Fachhochschulen“, einige dieser Tagungen hat er an der FH Heilbronn selbst organisiert. Seit 1997 war er Mitglied der Fachgruppenleitung. Bei den beiden Tagungen der Fachgruppe in Schöntal und Schönenberg übernahm er die lokale Organisation. Auch ihm gilt unser Dank.*

Die neue Fachgruppenleitung auf der Sitzung in Göttingen

*Nach der Entlastung der Sprecher der alten Fachgruppenleitung, Wolfram Koepf und H. Michael Möller, konstituierte sich die neue Fachgruppenleitung. Die Sitzungsleitung wurde von Herrn Matzat übernommen, der Herrn Koepf als Sprecher vorschlug. Herr Koepf wurde einstimmig wieder zum Sprecher der*

*Fachgruppe gewählt. Ebenfalls einstimmig wurde auf Vorschlag des Sprechers Herr Hiß zum Stellvertreter gewählt.*

*Für die kommenden Jahre sollen die Arbeitsschwerpunkte Computeralgebra in den Ingenieurwissenschaften (K. Hackl), an Fachhochschulen (E. Heinrich), in der Lehre und Didaktik (H.-W. Henn), an der Schule (H. Knechtel), Computeralgebra-Neuerscheinungen (J. Grabmeier), Internet und Mathematische Software (U. Kortenkamp) sowie Computeralgebra in der Industrie (Sorgatz) durch Referenten der Fachgruppenleitung besonders gefördert werden. Gemäß* § *7.1 unserer Ordnung kann die Fachgruppenleitung ferner bis zu 3 Fachexperten hinzuziehen. Als weitere Schwerpunkte der Arbeit der Fachgruppenleitung wurden die Themen Physik, Chemie sowie Rundbrief festgelegt. Th. Hahn wurde als Fachexperte Physik bestätigt, und R. Laue (Chemie) sowie M. Wessler (Rundbrief) wurden ebenfalls zu Fachexperten berufen.*

*Wie in der letzten Wahlperiode wurden die Zuständigkeiten für bestimmte Rubriken des Computeralgebra-Rundbriefs wieder verbindlich einzelnen Mitgliedern der Fachgruppenleitung zugeordnet. Für die Rubrik „Tagungen der Fachgruppe“ ist der Sprecher W. Koepf zuständig. Es wurde ferner festgelegt, dass sich U. Kortenkamp um „Neues über Systeme“ sowie B. H. Matzat um „Themen und Anwendungen der Computeralgebra“ kümmern. Die Rubrik „Computeralgebra in der Schule“ wird von H. Knechtel und die Rubrik „Computeralgebra in der Lehre“ von G. Hiß übernommen. Herr Malle ist zuständig für „Berichte über Arbeitsgruppen“. Für „Publikationen über Computeralgebra“ und Buchbesprechungen ist J. Grabmeier federführend, und die Verantwortung für die Rubriken „Hinweise auf Konferenzen“, „Berichte von Konferenzen“ sowie „Lehrveranstaltungen zu Computeralgebra“ liegt bei M. Wessler.*

*Die nächste wissenschaftliche Tagung der Fachgruppe findet vom* ***2.–4. Juni 2005*** *statt. Wie beim letzten Mal ist der Donnerstag als Anreisetag gedacht, die Tagung beginnt nach dem Mittagessen und endet am frühen Samstagnachmittag. Die Fachgruppenleitung beschloss, auf dieser Tagung zum ersten Mal einen* **Nachwuchspreis** *für den besten Vortrag eines Nachwuchswissenschaftlers zu vergeben, der mit 500 € dotiert sein wird. Es wird ebenfalls wieder einige Übersichtsvorträge über relevante Gebiete der Computeralgebra geben. Die Abstracts dieser Vorträge sowie weitere Details zu dieser Tagung finden Sie auf S. 9.*

*In der Woche nach Ostern 2006 soll dann wieder eine Tagung „Computeralgebra in Lehre, Ausbildung und Weiterbildung“ zum Thema „Entdecken, Üben, Prüfen mit Computeralgebra – Neue Entwicklungen an Schule und Hochschule“ stattfinden. Wegen des späten im Sommersemester liegenden Termins wird auch diese Tagung von Donnerstag, 20. April 2006, bis Samstag, 22. April 2006, stattfinden. Tagungsort wird wieder das Haus Schönenberg bei Ellwangen sein. Organisiert wird die Tagung von den Herren Henn und Matzat. Näheres siehe S. 10.*

*Die internationale ISSAC-Tagung wird voraussichtlich nach Genua 2006 im Jahr 2008 wieder in Europa stattfinden. Die Fachgruppenleitung strebt an, sich 2008 mit einem geeigneten deutschen Austragungsort zu bewerben. Vorgespräche haben ergeben, dass sowohl Berlin als auch München als Tagungsorte in Frage kommen. Die Fachgruppenleitung plant, auf ihrer Herbstsitzung zu einer Entscheidung über unsere Bewerbung zu kommen.*

*Karin Gatermann, die bis vor kurzem Mitglied der Fachgruppenleitung war, ist tot. Diese erschütternde Nachricht haben wir Anfang des Jahres erhalten. Nach kurzer schwerer Erkrankung ist sie am 1. Januar 2005 verstorben. Einen Nachruf finden Sie auf der nächsten Seite.*

*Sollten Sie elektronische Reklame-Post erhalten haben, deren typischer Empfängerkreis die Mitglieder der Gesellschaft für Informatik (GI) sind, so könnte dies daran liegen, dass die GI Mitgliederadressen an kommerzielle Anbieter verkauft. Da die Differenzierung der E-Mail-Adressen in GI-interne und andere erst 2002 realisiert wurde, wurden die ganzen alten Adressen mangels Willenserklärung für Werbung freigegeben. Bitte melden Sie sich, wenn Ihre E-Mail-Adresse GI-intern geschaltet werden soll.*

*Wenn wir schon bei Werbung sind: Kennen Sie nicht den einen oder anderen interessierten Kollegen, der noch nicht Mitglied der Fachgruppe Computeralgebra ist? Vielleicht möchte sie oder er bei uns Mitglied werden? Wir senden gerne ein Probeexemplar des Rundbriefs zu.*

*Wir hoffen, Sie mit dem vorliegenden Heft wieder gut zu informieren.*

*Wolfram Koepf* *Gerhard Hiß*

# Nachruf

Karin Gatermann hat uns für immer verlassen.

Frau Prof. Dr. Karin Gatermann war nur kurz aktiv in der Fachgruppenleitung. Sie kandidierte 2002 und wurde mit einem für eine Erstkandidatur überragenden Ergebnis (drittbestes Ergebnis von allen Bewerbern) gewählt. Sie konnte nur an wenigen Sitzungen der Fachgruppenleitung teilnehmen, weil sie 2003 einen Ruf auf eine Forschungsprofessur für Computeralgebra an der University of Western Ontario in London (Kanada) annahm. Dort erkrankte sie unheilbar und verstarb, nach Deutschland zurückgekehrt, in ihrem Elternhaus am 1. Januar dieses Jahres im Alter von nur 43 Jahren.

Viele von uns haben sie in ihrem wissenschaftlichen Leben begleitet und sie in ihrer aufrichtigen, hartnäckigen und dabei doch liebenswerten Art schätzen gelernt. Die Ideen in ihren wissenschaftlichen Arbeiten, aber auch sie selbst als Person, werden in unserer Erinnerung lebendig bleiben. Wie sehr sie geschätzt wurde, zeigen die unten aufgeführten Nachrufe im Internet.

Karin Gatermann hat sich sehr um die Computeralgebra verdient gemacht. Sie war beispielsweise in den letzten Jahren mehrfach in das Program Committee der ISSAC-Konferenz gewählt worden, war im Editorial Board des Journal of Symbolic Computation vertreten und hat einen Sonderband des Journal of Symbolic Computation herausgegeben. Ihre Spezialität war die Anwendung der Computeralgebra bei algebraischen Gleichungssystemen, dynamischen Systemen und in der Chemie unter Benutzung von Symmetrieeigenschaften. Bemerkenswert sind auch ihre letzten Arbeiten zur Lösungstheorie von chemischen Reaktionsgleichungen.

Es war ihr leider nicht vergönnt, ihre Professur auszuüben.

Es ist geplant, am 6./7. Januar 2006 ein Gedenkkolloquium in Hamburg abzuhalten. An diesem Kolloquium wird sich auch die Fachgruppe beteiligen.

Nachruf der Arbeitsgruppe Computeralgebra der Western-Ontario-Universität
`http://www.csd.uwo.ca/karin_gatermann.html`

Nachruf der Fachbereiche Mathematik der FU Berlin, der Universität Hamburg, des Fritz-Haber-Instituts Berlin, des Weierstraß-Instituts Berlin
`http://www.math.uni-hamburg.de/home/werner/karin.jpg`

Nachruf der AG Angewandte Mathematik des Fachbereichs Mathematik der Universität Hamburg
`http://www.math.uni-hamburg.de/home/werner/GatermannNachruf.html`

**Computeralgebra: 02. – 04.06.2005, Kassel**

Nach dem großen Erfolg der Computeralgebra-Tagung, welche im Mai 2003 in Kassel stattfand, wird die Fachgruppe im Juni 2005 wieder eine derartige Tagung in Kassel durchführen.

*Tagung in Kassel 2003*

Die wissenschaftliche Tagung findet vom ***2. – 4. Juni 2005*** statt. Wie beim letzten Mal ist der Donnerstag als Anreisetag gedacht, die Tagung beginnt nach dem Mittagessen und endet am frühen Samstagnachmittag.

Die Hauptvorträge sind:

- Priv. Doz. Dr. David J. Green (Bergische Universität Wuppertal): *Computerberechnungen in der Kohomologie endlicher Gruppen*
  In der rechnergestützten Gruppenkohomologie berechnet man niedrigdimensionale Kohomologiegruppen (mit komplizierten Koeffizientenmoduln) aufgrund ihrer Bedeutung für eine Vielzahl von Klassifikationsproblemen. Außerdem berechnet man Kohomologieringe (mit Koeffizienten in einem endlichen Körper), um die kommutative Algebra von Kohomologieringen zu studieren. Nach der Erklärung der Grundbegriffe und der stichpunktartigen Darstellung der bisherigen Ansätze in beiden Bereichen werden die Methoden und Ergebnisse für Kohomologieringe näher erläutert.

- Prof. Dr. Florian Heß (Technische Universität Berlin): *Kryptographie mit einem Blick auf Anwendungen der Computeralgebra*
  Der Vortrag setzt sich zum Ziel einen Überblick über einige Fragestellungen, Techniken und offene Probleme in der Kryptographie zu geben, wobei dafür relevante Anwendungen der Computeralgebra besonders hervorgehoben werden sollen. Zwei der Hauptziele der Kryptographie sind Verschlüsselung und digitale Unterschriften. Dazu kommen eine große Zahl weiterer Aufgaben wie die Realisierung kryptographisch sicherer Hash-Funktionen und von Pseudozufallszahlengeneratoren oder die Realisierung elektronischen Gelds, Protokolle für elektronische Wahlen usw. Zur Lösung dieser Aufgaben werden im Allgemeinen geeignete Einwegfunktionen als grundlegene Bausteine verwendet. Einwegfunktionen sind Funktionen, für welche Bilder sehr leicht, aber Urbilder nur mit sehr großem Aufwand berechnet werden können. Die Existenz von Einwegfunktionen ist nicht bekannt. Es gibt allerdings ein paar Kandidaten, und an dieser Stelle kommt die Mathematik in Form von Algebra, Zahlentheorie und algebraischer Geometrie ins Spiel. Die Untersuchung dieser Kandidaten besitzt eine konstruktive und eine destruktive Seite. Auf der konstruktiven Seite beschäftigt man sich mit der effizienten Implementation von Einwegfunktionen, wohingegen auf der destruktiven Seite untersucht wird, wie groß der Aufwand der Urbildberechnung wirklich ist. Beide Seiten verwenden Methoden aus der Computeralgebra. Im Vortrag sollen diese und verwandte Themen diskutiert werden. Besonderes Augenmerk wird dabei auf Einwegfunktionen gerichtet, welche durch algebraische Kurven über endlichen Körpern erhalten werden können.

- Prof. Dr. Mark van Hoeij (Florida State University): *Maple's Algebraic Curves Package*
  Maple has a number of algorithms for algebraic curves: algebraic algorithms such as genus, holomorphic differentials, integral basis, parametrization, normal forms for (hyper)-elliptic curves, as well as numeric algorithms such as monodromy and period matrix. How these algorithms work is the topic of this talk. Possible future improvements will be discussed as well.

- Dr. Jürgen Klüners (Universität Kassel): *Faktorisierung von Polynomen*
  Bekannterweise gibt es Algorithmen, die Polynome über den ganzen Zahlen in Polynomlaufzeit faktorisieren. Leider sind diese Algorithmen, welche auf der Gitterreduktion nach Lenstra, Lenstra und Lovasz basieren, in der Praxis zu langsam. Kürzlich hat Mark van Hoeij einen neuen Faktorisierungsalgorithmus gefunden, welcher sehr gut in praktischen Anwendungen ist. Wir erweitern

diesen Algorithmus auf das Faktorisieren von Polynomen in $\mathbb{F}(t)[x]$, wobei $\mathbb{F}$ ein endlicher Körper ist. Überraschenderweise sind hier die Algorithmen viel einfacher und effizienter. Wir können beweisen, dass der neue Faktorisierungsalgorithmus (in $\mathbb{Q}[x]$ und $\mathbb{F}(t)[x]$) in Polynomlaufzeit terminiert. Die Algorithmen sind für beide Fälle im Computeralgebrasystem Magma implementiert.

- Prof. Dr. Gerhard Pfister (Universität Kaiserslautern): *Das Computeralgebrasystem SINGULAR - Entwicklung und Anwendungen*
  Das Computeralgebrasystem SINGULAR wird vorgestellt. Die Entwickler dieses Systems haben dieses Jahr den Richard D. Jenks Memorial Prize for Excellence in Software Engineering for Computer Algebra auf der ISSAC-Tagung erhalten. Es werden mehrere Anwendungen gezeigt. Darunter ist eine Anwendung aus der Gruppentheorie und mehrere Anwendungen in der Elektronik und Robotik.

Die Fachgruppenleitung beschloss, auf dieser Tagung zum ersten Mal einen ***Nachwuchspreis*** für den besten Vortrag eines Nachwuchswissenschaftlers zu vergeben, der mit 500 € dotiert sein wird. Ferner wird es auf der Tagung in Kassel eine Informationsveranstaltung der Fachgruppe geben. Die Tagung bietet wieder die einmalige Gelegenheit, in gut zwei Tagen eine Vielzahl von Forschungsgebieten der Computeralgebra zu beleuchten, und junge Forscher haben die Möglichkeit, ihre Forschung vor einem großen fachkompetenten Publikum zu präsentieren. Wir rechnen daher wieder mit einer zahlreichen Beteiligung.

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

In der letzten Zeit wurden und werden in allen Bundesländern Standards und kompetenzorientierte Kernlehrpläne für den Mathematikunterricht entwickelt. Der Einsatz von Computeralgebra (in Form von CAS-Taschenrechnern oder von Computeralgebrasystemen auf PCs) ist ebenfalls in allen Bundesländern zumindest auf freiwilliger Basis möglich. Welchen Einfluss können und sollen diese CA-Werkzeuge für den Mathematikunterricht an allgemeinbildenden Schulen und für die Anfängerausbildung an den Universitäten und Hochschulen haben? Die Tagung soll eine Bestandsaufnahme vornehmen und Perspektiven für eine weitere Entwicklung aufzeigen. Auf der Homepage der Fachgruppe (`http://www.fachgruppe-computeralgebra.de/CLAW`) finden Sie ausführliche Informationen über die vergangenen Tagungen und im Herbst 2005 nähere Informationen und Anmeldeformulare für die 5. Tagung.

# Themen und Anwendungen der Computeralgebra

## Einsatz von Computeralgebrasystemen zum Entwurf mechatronischer Systeme am Beispiel von CAMeL-View

**Martin Hahn (Paderborn)**

`Martin.Hahn@iXtronics.de`

Im Zentrum der Innovation in den Bereichen Fahrzeugtechnik und Maschinenbau steht das Querschnittsfach Mechatronik. Mechatronik stellt den integrierten Entwurf aus Komponenten der Mechanik, Aktorik, Sensorik und Informationsverarbeitung in den Mittelpunkt und erlaubt den Entwurf von Systemen mit neuen, innovativen Produktfunktionen. Der Entwurf derartiger Systeme stellt den Ingenieur vor eine Reihe von Herausforderungen:

- Die Komplexität der Systeme steigt durch die Multidisziplinarität des Mechatronikansatzes erheblich an.

- Der Spielraum, Funktionen von der Mechanik durch Einsatz moderner, regelungstechnischer Methoden in die Informationsverarbeitung zu verlagern (z. B. „elektronische Kurvenscheibe“ im Werkzeugmaschinenbau) oder neue Funktionen zu entwickeln, die rein mechanisch nicht denkbar wären (z. B. ABS-, ESP-Systeme in der Fahrzeugtechnik) ist erheblich.

- Bei mechatronischen Systemen handelt es sich fast immer um geregelte (also rückgekoppelte) Systeme. Diesen Systemen „sieht" man ihr Bewegungsverhalten (Positionen, Orientierungen, Geschwindigkeiten sowie Beschleunigungen) nicht an. Das Bewegungsverhalten ist aber häufig primäres Entwurfsziel der Produktentwicklung.

Um diese Herausforderungen meistern zu können, ist es notwendig den Prozess der Produktentwicklung geeignet zu strukturieren sowie Entwurfswerkzeuge einzusetzen, die bereits in den frühen Phasen des Entwurfs einen Einblick in die Systemdynamik erlauben. Dazu werden Entwicklungswerkzeuge benötigt, die den Entwurfsprozess optimal unterstützen. Dabei hat sich der Einsatz von Computeralgebrasystemen in Forschungsprojekten bewährt. Eine Möglichkeit, dem Entwickler die Vorteile von Computeralgebrasystemen in der Industrie zugänglich zu machen, besteht darin, Computeralgebrasysteme in Engineeringsoftwaresysteme zu integrieren. In diesem Beitrag wird gezeigt, wie das Computeralgebrasystem MUPAD [SF01] im Entwurfsprozess mechatronischer Systeme in der für den Ingenieur zugeschnittenen Softwarelösung CAMeL-View [IX01] eingesetzt wird.

**Entwurfsprozess mechatronischer Systeme**

Abbildung 1 zeigt eine mögliche Strukturierung des Entwurfsprozesses, die sich in Maschinenbau und Fahrzeugtechnik bewährt hat.

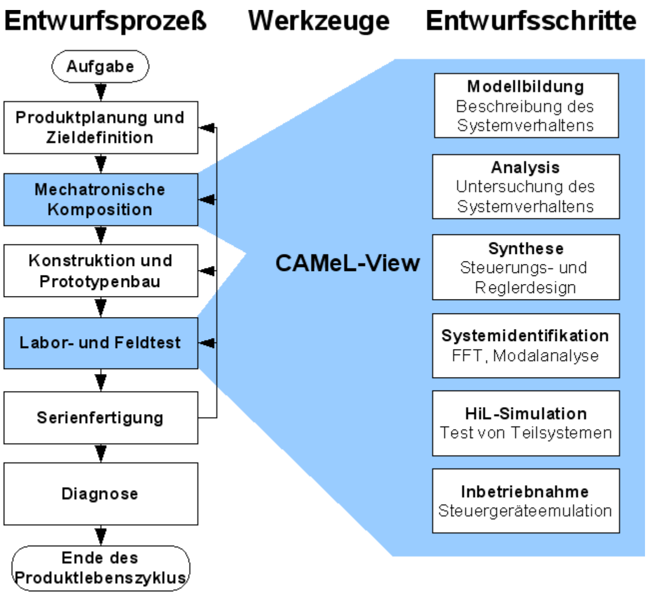



*Abbildung 1: CAMeL-View im Entwurfsprozess mechatronischer Systeme*

Kernpunkt des Entwurfsprozesses ist der Schritt der mechatronischen Komposition, bei dem die Architektur des Produkts, d. h. die Struktur des Systems aus Bausteinen der Mechanik, Aktorik, Sensorik und Informationsverarbeitung entworfen wird.

Um möglichst schnell eine Vielzahl von Lösungsvarianten untersuchen zu können, ist ein ausschließlich CAD-gestützter Ansatz nicht geeignet. Vielmehr benötigt man einen Ansatz, mit dem vereinfachte Baustrukturen auf der Basis physikalisch-topologischer Modelle untersucht werden können.

Dabei werden unterschiedliche Systemvarianten hinsichtlich der Erfüllung der (Hauptgebrauchs-) Funktionen untersucht. Zeigt die modellgestützte Untersuchung, dass die Systemvariante für die Realisierung geeignet erscheint, findet der Übergang vom simulierten zum physikalisch vorhandenen System schrittweise unter Verwendung der HIL-Simulation statt.

**Mathematische Beschreibung mechatronischer Systeme**

Eine derartige Vorgehensweise wird von kommerziell verfügbaren Softwarewerkzeugen nicht in ausreichendem Maß unterstützt, da diese in der Regel auf ihr Einsatzgebiet, z. B. die MKS-Simulation, beschränkt sind, mechatronische Systeme aber auch Bauelemente der Hydraulik, der Regelungstechnik und der Softwaretechnik enthalten. Zusätzlich werden für einen ganzheitlichen Entwurf – über die reine Simulation hinaus – weitere Analyse- und Synthesemethoden wie z. B. die Frequenzgangsberechnung oder die Parametervektoroptimierung benötigt [HA01] und [ME01]. Dies erfordert eine einheitliche mathematische Beschreibung des Gesamtsystems, dazu wird in der Regelungstheorie die nichtlineare Zustandsraumdarstellung verwendet, eine nichtlineare Vektordifferentialgleichung erster Ordnung:

$$\dot{x} = f(x, u, p, t), \quad \dot{y} = g(x, u, p, t)$$

In dieser mathematischen Form beschriebene Teilsysteme lassen sich dann in Form von Blockdiagrammen auf der Basis von gerichteten Eingangs-Ausgangs-Verbindungen (Signalfluss) miteinander verkoppeln.

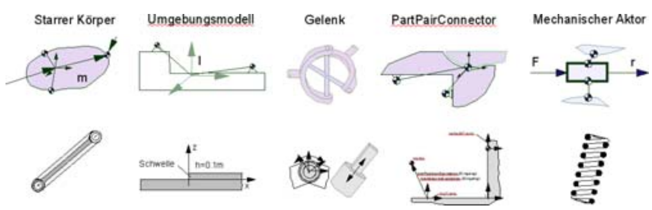



*Abbildung 2: Grundstrukturen von Mehrkörpersystemen*

Besonders gut geeignet ist die Modellbeschreibung mit Blockdiagrammen für regelungstechnische Teilsysteme, da hierbei der Signalfluß zwischen den Teilsystemen beschreibendes Element ist. Für physikalische Systeme wie z. B. Systeme der Mechanik ist diese Art der Modellbildung weniger geeignet, da die Ableitung der nichtlinearen, dreidimensionalen Differentialgleichungssysteme von Hand schwierig, zeitaufwändig und fehlerträchtig ist. Für die mechanischen Anteile mechatronischer Systeme eignet sich die physikalisch orientierte Beschreibung als Mehrkörpersystem.

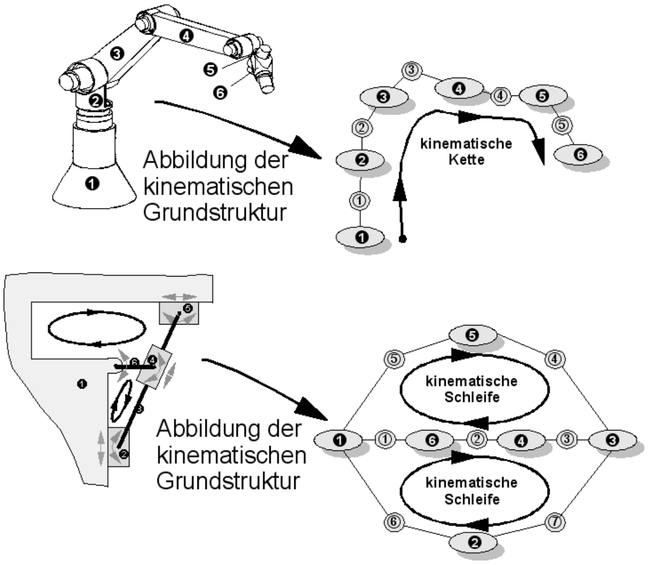



*Abbildung 3: Bauelemente der Mehrkörpersystemdynamik*

Bestandteile der Mehrkörpersysteme sind starre Körper und Gelenke sowie externe Kräfte, die auf das System einwirken. Je nach Gelenktyp werden die Freiheitsgrade des Systems eingeschränkt, um die Kinematik eines Systems nachzubilden. Beispiele für Gelenke sind Dreh- und Schubgelenke sowie Kardan- und Kugelgelenke.

Um auf der Basis der topologischen Verknüpfung von Bauteilen automatisch die systembeschreibenden Gleichungen in Zustandsraumdarstellung zu generieren, werden insbesondere in der Mehrkörpersystemdynamik geeignete Formalismen verwendet. Diese Formalismen erzeugen, ausgehend von der Systemstruktur (s. a. Abbildung 3) die beschreibenden Differentialgleichungen. Eine Implementierung in einem Computeralgebrasystem hat sich an dieser Stelle besonders bewährt. Auf die Vorteile wird im Folgenden näher eingegangen.

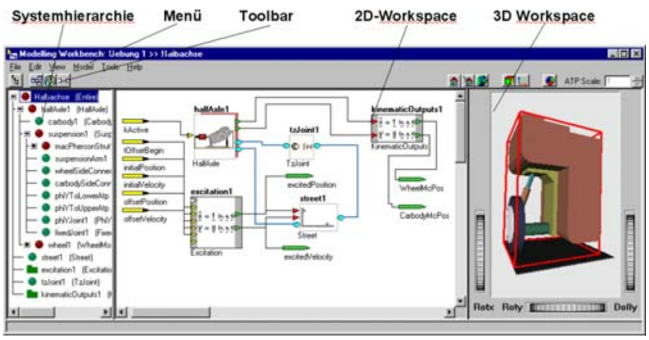



*Abbildung 4: CAMeL-View Modeller*

**CAMeL-View – eine Lösung für den Entwurf mechatronischer Systeme**

Um neuartige, innovative Produkte entwickeln zu können, sind rechnergestützte Entwicklungssysteme („virtuelles Prototyping“, CAE-Systeme) notwendig, die die fachübergreifende Modellbildung mechatronischer Systeme unterstützt. CAMeL-View ermöglicht die engineeringgerechte Beschreibung mechatronischer Systeme und bietet dazu einen bequemen interaktiven Weg an. Damit können Modelle komplexer mechatronischer Systeme aufgebaut werden, die unterschiedliche Fachdisziplinen (z. B. Mehrkörpersysteme, Hydraulik, Steuerungstechnik und diskrete Systeme) abdecken. Ferner wird der Entwickler durch eine umfangreiche Bibliothek unterstützt, die vordefinierte Bauelemente enthält. Des Weiteren lässt CAMeL-View die Integration von Modellen anderer Modellierungsumgebungen wie Matlab/Simulink [MW01, MW02] zu. Das fachspezifische Modell von Mehrkörpersystemen schließt auch die 3D-Beschreibung mit ein, die aus CAD-Systemen (z. B. OpenInventor, VRML, DXF und IGES) importiert und automatisch für Animationszwecke vereinfacht werden kann.

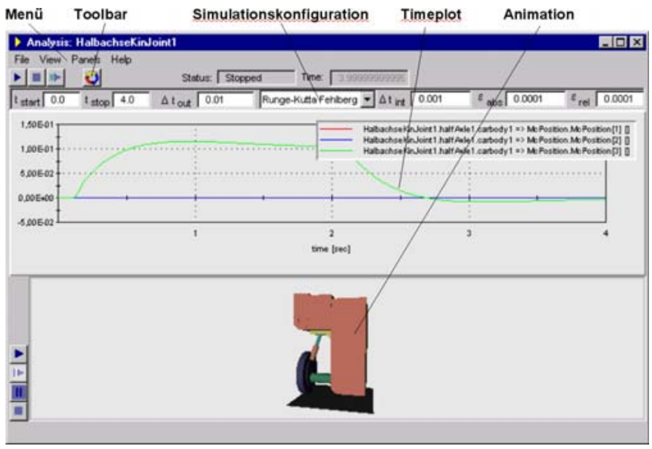



*Abbildung 5: CAMeL-View Analysefenster*

Zur Analyse wird das physikalische Modell automatisch auf eine mathematische Darstellung und optimierten C-Code übertragen und steht für Simulationsrechnungen sowie für lineare Analysen (Frequenzgangsberechnungen, Eigenwertberechnungen) zur Verfügung.

Um die Modelle in Prüfstandsexperimenten (HiL-Anwendungen) verwenden zu können, gibt es in CAMeL-View zwei Wege:

- Export von speziell zugeschnittenen C-Codes zur Nutzung in Hardware-in-the-Loop Umgebungen (z. B. als s-function in Matlab/Simulink unter Verwendung von dSpace-Hardware)
- Direkte Codegenerierung und Anbindung an Motorola PowerPC-basierte Hardware (CAMeLTest-Rig). Die Modelle können dann direkt auf eine Echtzeitplattform übertragen und simuliert werden. In beiden Fällen bleiben die Parameter des Modells ohne erneute Codegenerierung änderbar.

**Mehrkörpersystemformalismen**

Ist das mechanische Teilsystem in Form von Körpern, Gelenken, Aktoren und Sensoren beschrieben, dann kann daraus ablauffähiger Code für das mathematische Modell generiert werden. In CAMeL-View werden drei Mehrkörpersystemformalismen für die Generierung der Gleichungen unterstützt [HA01]:

- Formalismus der dynamischen Bindungen
- Rekursive Formulierung auf der Basis der Newton- Euler-Gleichungen

- Minimalkoordinatenformulierung auf der Basis der Langrangeschen Gleichungen

Eine Verwendung verschiedener Formalismen (für jeweils zusammenhängende Teilsysteme) in einem Gesamtsystem ist möglich. Dadurch eröffnen sich eine Vielzahl von Möglichkeiten zur Systemstrukturierung und zur Parallelisierung. Ist der Einsatz komplexer Modelle in Prüfstandsanwendungen (Hardware-in-the-Loop Anwendungen) das Ziel, dann werden erhebliche Anforderungen an den erzeugten Code gestellt. Wesentliche Forderung ist, dass der Code unter harten Echtzeitbedingungen läuft. Steht also für die Auswertung der Gleichungen eine Zeitdauer von z. B. 1msec zur Verfügung, dann muss die Auswertung der Gleichungen nach diesem Zeitraum abgeschlossen sein (inkl. der für die Prüfstandsauswertung notwendigen Signalwandlung und -aufbereitung).

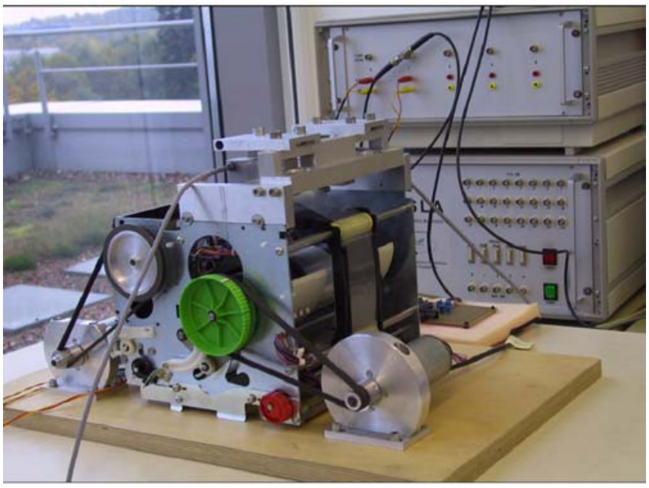


Abbildung 6: Prüfstandsanwendung von Modellcode

Wenn die Auswertung länger dauert, kann das System numerisch instabil werden, einem Prüfstand falsche Signale liefern und damit zum Abbruch des Versuchs führen. Um die Evaluierungsdauer pro Zeitschritt möglichst klein zu halten und komplexe Modelle in HiL-Applikationen überhaupt verwenden zu können, ist die Optimierung der Gleichungen mit Hilfe von Computeralgebrasystemen neben der Wahl des MKS-Formalismus von wesentlicher Bedeutung.

**Implementierung von MKS-Formalismen mit MuPAD**

Besonders bewährt bei der Generierung der Gleichungen für echtzeitfähige HiL-Anwendungen hat sich der MKS-Formalismus der Minimalkoordinaten. Aufbauend auf einem modifizierten Lagrangeformalismus werden die Systemgleichungen in nichtlinearer Zustandsraumdarstellung, einer Standarddarstellung der Regelungstechnik, mit Hilfe des Computeralgebrasystems MuPAD generiert.

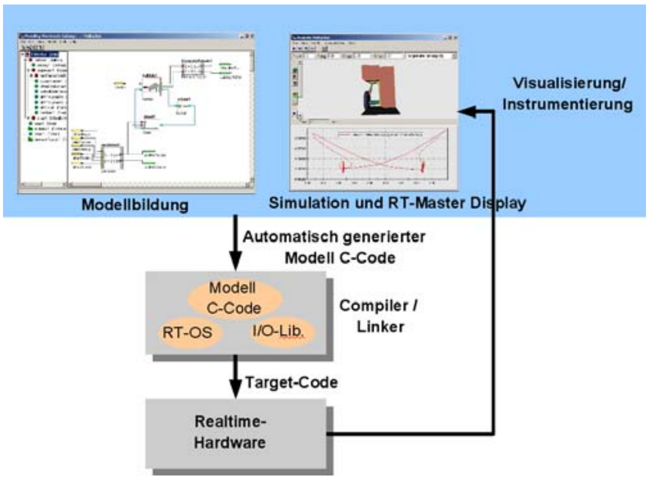



*Abbildung 7: Architektur der Codegenerierung*

Bei der Generierung der mathematischen Gleichungen kommen dabei die Vorteile eines Computeralgebrasystems zum Einsatz, beliebige Ausdrücke erzeugen und vereinfachen zu können. Im Folgenden sind einige Beispiele aufgeführt, die durch den Einsatz von MuPAD erheblich vereinfacht oder erst möglich werden:

- Die Systemgleichungen können nach der Codegenerierung in Matrizenform (Massen-, Dämpfungs- und Steifigkeitsmatrix) im MuPad-Format abgespeichert werden. Dies erlaubt eine nachgeschaltete symbolische Analyse der Gleichungen.

- In CAMeL-View kann das zu modellierende System parametrisch formuliert werden. Ist dies der Fall, dann werden die Parameter und deren Abhängigkeiten bei der Codegenerierung erhalten. Dies erlaubt es in anschließenden Untersuchungen symbolisch auf die Parameter zuzugreifen und so z. B. die Empfindlichkeit des Systems bzgl. bestimmter Parameter exakt zu berechnen.

- Durch den Einsatz von MuPAD konnte die Ableitung der Systemgleichungen konfigurierbar gestaltet werden. So lässt sich z. B. das Verfahren zur symbolischen Inversion der Massenmatrix konfigurieren. In CAMeL-View stehen dazu z. B. die Inversion auf der Basis der Choleskyzerlegung oder andere von Computeralgebrasystemen angebotene Verfahren zur Verfügung. Des Weiteren können Terme, die für ein Experiment nicht benötigt werden, abgeschaltet werden. Dies ist z. B. bei der Berechnung der Coriolis- und Zentrifugalkräfte hilfreich, da diese große symbolische Ausdrücke erzeugen, die sich negativ auf die Simulationsdauer auswirken.

- Darüber hinaus werden die Hilfsmittel zur Codevereinfachung genutzt. Insbesondere in der MKS-Dynamik besteht die Möglichkeit, bei Systemen mit ausgeprägten geometrischen Symmetrien Terme zusammenzufassen und durch Anwendung von z. B. Additionstheoremen die Codes stark zu vereinfachen. Damit wird die Evaluierungsdauer erheblich verbessert.

**Zusammenfassung**

Bei der Entwicklung von mechatronischen Systemen hat sich gezeigt, dass integrierte Programmpakete wie CAMeL-View (Download einer Testversion unter `www.iXtronics.de`) die Basis einer modellgestützten Entwicklung darstellen. Durch die an der Physik orientierte Modellbildung wird das Aufbauen von Modellen für Entwickler erheblich vereinfacht. Durch den Einsatz von Computeralgebrasystemen wird erreicht, dass sich der Entwickler voll auf die Lösung seiner Entwicklungsaufgabe konzentrieren kann und sich nicht mit der fehlerträchtigen und zeitaufwändigen Ableitung der mathematischen Gleichungen auseinandersetzen muss. Dies gilt insbesondere für das Feld der Mehrkörpersystemdynamik, um passgenaue C-Codes für die Modelle zu generieren und optimale Performance durch symbolische Vereinfachung der Gleichungen für Prüfstandsanwendungen zu erreichen. Damit steht ein auf den Mechatronikentwickler zugeschnittenes Werkzeug zur Verfügung, das die Vorteile eines Computeralgebrasystems nutzt und die Flexibilität derartiger Systeme in Engineeringapplikationen zur Verfügung stellt.

**Literatur**

[HA01] M. Hahn: OMD – Ein Objektmodell für den Mechatronikentwurf. Anwendung in der objektorientierten Modellbildung mechatronischer Systeme unter Verwendung von Mehrkörpersystemformalismen. Fortschr.-Ber. VDI, Reihe 20, Nr. 299, VDI-Verlag, Düsseldorf, 1999.

[IX01] iXtronics GmbH: CAMeL-View R5.0, Paderborn, 2004.

[ME01] U. Meier-Noe: Modellierung mechatronischer Systeme – Basis der Systemauslegung und der Wissenskonservierung. Fortschr.-Ber. VDI, Reihe 20, Nr. 379, VDI-Verlag, Düsseldorf, 2004.

[MW01] The MathWorks, Inc. (Hrsg..): MATLAB 7. Natick, MA, 2004.

[MW02] The MathWorks, Inc. (Hrsg.): SIMULINK 6. Natick, MA, 2004.

[SF01] SciFace GmbH (Hrsg.): MUPAD R3.0. Paderborn, 2004.

## Symbolic Summation in Theoretical Physics

**Sven-Olaf Moch (DESY, Zeuthen)**

`sven-olaf.moch@desy.de`

*We report on algorithms and applications of symbolic summation as used in theoretical physics, in particular in perturbative quantum field theory.*

Ever since, the subject of symbolic summation has been an important part of mathematics. Symbolic summation amounts to the problem of finding a closed form expression for a given sum or series. Systematic studies have been pioneered by Euler [1], and for specific sums, exact formulae are known since long, series representations of transcendental functions being a prominent example here. Today, general classes of sums, for example so-called harmonic sums which generalize Euler's construction, have been investigated along with their mathematical properties, see e.g. Ref. [2]. In addition, a large number of results is also available through the *Online Encyclopedia of Integer Sequences* [3].

Symbolic summation has further advanced through the development and improvement of algorithms suitable for computer algebra systems. Here, the possibility to obtain exact solutions for recurrences or, more generally, by means of recursive methods has led to significant progress, for instance in the summation of rational or hypergeometric series, see e.g. [4]. It is the aim of this article to give a brief overview of research on symbolic summation in theoretical physics, in particular in quantum field theory.

Quantum corrections to observables, say cross-sections for the scattering of particles in quantum field theory are commonly described by so-called Feynman diagrams. In physics, these diagrams can be interpreted as the trajectories of real and virtual particles, participating in the scattering process, see e.g. [6]. In mathematical terms, Feynman diagrams are given as integrals over the loop momenta of the associated particle propagators. These integrals are usually divergent, thus requiring some regularization. The state of the art is dimensional regularization which implies an analytical continuation of the dimension of space-time from four to $D$, thus keeping manifest gauge invariance with respect to any underlying gauge symmetry. The main computational task is then to obtain the relevant quantities as a Laurent series in the small parameter $\epsilon = D - 4$.

In general, Feynman integrals in $D$-dimensional regularization may depend on multiple scales. Analytical expressions may lead to transcendental or generalized hypergeometric functions, which have a series representation through nested sums with symbolic arguments. Thus, from the point of view of symbolic summation, their properties deserve a closer look, as we are eventually interested in an expansion in the small parameter $\epsilon$ around integer or rational numbers. These expansions generate multiple nested sums and for large classes of functions the symbolic summation can be performed by algebraic means based on recursive algorithms.

The basic recursive definition of nested sums is given by [7]

$$S(n; m_1, ..., m_k; x_1, ..., x_k) = \sum_{j=1}^{n} \frac{x_1^j}{j^{m_1}} S(j; m_2, ..., m_k; x_2, ..., x_k), \qquad (1)$$

where generally all $x_i \leq 1$. The sum of all $m_i$ is called the weight of the sum, while the index $k$ denotes the depth. This definition actually contains the series representations of classical polylogarithms, Nielsen functions and multiple polylogarithms. For all $x_i = 1$, the definition above reduces to harmonic sums [1, 8] and for the upper summation boundary $n \to \infty$, one recovers the (multiple) zeta values associated to Riemann's zeta function [2].

An important property of the sums in (1) is the fact, that they obey an algebra of multiplication. Specifically, any product

$$S(n; m_1, ..., m_k; x_1, ..., x_k) \times S(n; m'_1, ..., m'_l; x'_1, ..., x'_l) \qquad (2)$$

can be expressed again as a sum of single nested sums. The underlying algebraic structure is a Hopf algebra and for products of nested sums as in (2), it is realized as a quasi-shuffle algebra. Thus, it can be implemented very efficiently on a computer. We give here sample code for the computer algebra system Form [9]. [1]

```
repeat;
 id,once S(n?,R(?m1),X(?x1))
  * S(n?,R(?m2),X(?x2)) =
  S2base(n,R(?m1),X(?x1)
   ,R,X,R(?m2),X(?x2));
 repeat id S2base(n?,R(n1?,?m1)
   ,X(y1?,?x1),R(?m3),X(?x3)
   ,R(n2?,?m2),X(y2?,?x2)) =
  S2base(n,R(?m1),X(?x1),R(?m3,n1)
   ,X(?x3,y1),R(n2,?m2),X(y2,?x2))
  + S2base(n,R(n1,?m1),X(y1,?x1)
   ,R(?m3,n2),X(?x3,y2),R(?m2),X(?x2))
  - S2base(n,R(?m1),X(?x1),R(?m3,n1+n2)
   ,X(?x3,y1*y2),R(?m2),X(?x2));
 id S2base(n?,R(?m1),X(?x1),R(?m3)
   ,X(?x3),R(?m2),X(?x2)) =
  S(n,R(?m3,?m1,?m2),X(?x3,?x1,?x2));
endrepeat;
```

[1] see the article by J. A. M. Vermaseren on page 17

The algorithm works recursively in the depth of the individual sums. It allows to express any product of nested sums in a canonical form, which is an important feature for practical applications.

Let us illustrate the preceding discussion and the structure of (1) with a simple example,

$$\begin{aligned} &\sum_{j=1}^{\infty} \frac{x^j}{j!} \Gamma(j - \epsilon) \\ &\quad = \sum_{j=1}^{\infty} \frac{x^j}{j} - \epsilon \sum_{j=1}^{\infty} \frac{x^j}{j} S(j-1; 1; 1) + \mathcal{O}(\epsilon^2) \\ &\quad = -\ln(1-x) - \epsilon \frac{1}{2} \ln^2(1-x) + \mathcal{O}(\epsilon^2), \end{aligned} \qquad (3)$$

provided $x < 1$. Here we see harmonic sums emerging from the expansion Gamma functions in a small parameter and we have expressed the $S$-sums in infinity through standard logarithms.

More generally, for a Feynman integral under consideration, the structure of nested sums arises from the solution of recursion relations. To that end, imagine a mapping of a given integral $I(x)$ depending on some (physical) scale $x$, $0 \leq x \leq 1$ to the space of discrete variables $N$, $N \in \mathbb{N}$. This is accomplished by means of an integral transformation, for example a Mellin transformation. Then one can obtain difference equations for the Feynman integral $I(N)$, which may be written as

$$\begin{aligned} &a_0(N)\, I(N) + a_1(N)\, I(N-1) + \ldots \\ &\quad + a_m(N)\, I(N-m) \;=\; G(N), \end{aligned} \qquad (4)$$

where $G(N)$ is some inhomogeneous term and $a_i$ are some coefficients depending on $N$. The solution of (4) needs $m$ boundary conditions $I(0), \ldots, I(m-1)$.

Single-step difference equations can be summed up analytically in closed form. Suppose we have the equation

$$a_0(N)\, I(N) - a_1(N)\, I(N-1) \;=\; G(N), \qquad (5)$$

then its solution will be

$$\begin{aligned} I(N) = &\frac{\prod_{j=1}^{N} a_1(j)}{\prod_{j=1}^{N} a_0(j)} I(0) \\ &+ \sum_{i=1}^{N} \frac{\prod_{j=i+1}^{N} a_1(j)}{\prod_{j=i}^{N} a_0(j)} G(i). \end{aligned} \qquad (6)$$

In the case that the functions $a_i$ can be factorized in linear polynomials of the type $N + m + n\,\epsilon$ with $m, n$ being integers and $N$ being symbolic, the products can be written as combinations of Gamma functions. In the presence of parametric dependence on $\epsilon$, the Gamma functions should be expanded around $\epsilon = 0$, which will lead to factorials and harmonic sums similar to our example in (3). If the function $G(N)$ is expressed as a Laurent series in $\epsilon$ with the coefficients being combinations of harmonic sums in $N + m$ and powers of $N + m$, $m$ being a fixed integer, the sum in (6) can be done and $I(N)$ will be a combination of harmonic sums in $N + k$ and powers of $N + k$ with $k$ being a fixed integer.

Equation (4) is an example of a recursion for Feynman integrals with dependence on symbolic parameters. Solutions such as (6) allow an efficient implementation in computer algebra systems like Form resulting in a largely automatic build-up of nested sums.

The underlying mathematics uses algorithms that allow recursive solution of a given sum that falls within one of the following classes: Sums involving $j$ and $n-j$

$$\sum_{j=1}^{n-1} \frac{x_1^j}{j^{m_1}} S(j; m_2, ..., m_k; x_2, ..., x_k) \times \frac{(x_1')^{n-j}}{(n-j)^{m_1'}} S(n-j; m_2', ..., m_l'; x_2', ..., x_l') \,, \tag{7}$$

conjugations

$$-\sum_{j=1}^{n} \begin{pmatrix} n \\ j \end{pmatrix} (-1)^j \times \frac{x_1^j}{j^{m_1}} S(j; m_2, ..., m_k; x_2, ..., x_k) \,, \tag{8}$$

and sums involving binomials, $j$ and $n - j$,

$$-\sum_{j=1}^{n-1} \begin{pmatrix} n \\ j \end{pmatrix} (-1)^j \times \frac{x_1^j}{j^{m_1}} S(j; m_2, ..., m_k; x_2, ..., x_k) \times \frac{(x_1')^{n-j}}{(n-j)^{m_1'}} S(n-j; m_2', ..., m_l'; x_2', ..., x_l') \,. \tag{9}$$

All sums in these classes can be solved recursively, i.e. they can be expressed in canonical form. The underlying algorithm either reduces successively the depth or the weight of the inner sum, so that eventually the inner nestings vanish and the results can be written in the basis (1). This procedure of solving nested sums realizes a creative telescoping and, of course, relies on the quasi-shuffle algebra [2, 7] of multiplication in (2).

Let us finish this brief overview by mentioning a few applications. The methods presented here have already been used in full-fledged calculations in particle physics. In Quantum Chromodynamics, which is the $SU(3)$ gauge theory of the strong interaction, the so-called splitting functions have recently been calculated to three loops in perturbation theory [10, 11]. The calculation required a systematic evaluation of nested sums for all integrals occuring in approximately 10.000 Feynman diagrams. Due to the excessive size of the expressions at intermediate stages, this task was only possible with cutting edge technology like the computer algebra system Form.[2]

Another interesting application, which exceeds current built-in capabilities of computer algebra systems like Maple or Mathematica is the expansion of (generalized) hypergeometric functions. An easy exercise is, for instance,

$$\begin{aligned} & {}_2F_1(a\epsilon, b\epsilon, 1 - c\epsilon, x) \\ &= \sum_{j=0}^{\infty} \frac{(a\epsilon)_j (b\epsilon)_j x^j}{(1 - c\epsilon)_j j!} \\ &= 1 + ab\mathrm{Li}_2(x)\epsilon^2 + ab\big\{c\mathrm{Li}_3(x) \\ &\quad +(a + b + c)\mathrm{S}_{1,2}(x)\big\}\epsilon^3 + \mathcal{O}(\epsilon^4) \,. \end{aligned} \tag{10}$$

Again, we have expressed the $S$-sums in infinity by standard polylogarithms $\mathrm{Li}_n$ and Nielsen functions $\mathrm{S}_{n,p}$. The necessary algorithms for the expansion in (10) are also available in GiNaC framework [5, 12, 13].

## Literatur


[1] L. Euler, Novi Comm. Acad. Sci. Petropol. **20**, 140 (1775).

[2] M. E. Hoffman, `http://www.usna.edu/Users/math/meh/biblio.html`.

[3] N. J. A. Sloane, `http://www.research.att.com/njas/sequences/index.html`.

[4] M. Petkovsek, H. Wilf and D. Zeilberger, (1997), `http://www.cis.upenn.edu/~wilf/AeqB.html`.

[5] C. Bauer, Computeralgebra-Rundbrief 34, p.14 (2004).

[6] T. Hahn, Computeralgebra-Rundbrief 33, p.10 (2003).

[7] S. Moch, P. Uwer and S. Weinzierl, J. Math. Phys. **43**, 3363 (2002), hep-ph/0110083.

[8] J. A. M. Vermaseren, Int. J. Mod. Phys. **A14**, 2037 (1999), hep-ph/9806280.

[9] J. A. M. Vermaseren, (2000), math-ph/0010025.

[10] S. Moch, J. A. M. Vermaseren and A. Vogt, Nucl. Phys. **B688**, 101 (2004), hep-ph/0403192.

[11] A. Vogt, S. Moch and J. A. M. Vermaseren, Nucl. Phys. **B691**, 129 (2004), hep-ph/0404111.

[12] C. Bauer, A. Frink and R. Kreckel, (2000), cs/0004015.

[13] S. Weinzierl, Comput. Phys. Commun. **145**, 357 (2002), math-ph/0201011.


[2] see the article by J. A. M. Vermaseren on page 17

## The Symbolic Manipulation Program FORM

**J. A. M. Vermaseren (Amsterdam)**

t68@nikhef.nl

*FORM is a Symbolic Manipulation system which is specially optimized for large expressions. This is achieved partially by the restriction to local operations, i.e. operations that work on one term at a time. A FORM program is divided into modules. Only at the end of each module the generated terms are sorted, which is the only completely non-local operation in FORM. Combined with an efficient memory and disk management, FORM can easily handle GByte-size expressions. FORM was originally developed in High Energy Physics, where such large expressions appear in intermediate stages of the calculation of Feynman diagrams. It is possibly also the only seriously parallelized computer algebra system.*

**Introduction**

With the large number of computer algebra systems available one might wonder why one should be interested in yet another one like FORM. It should be mentioned however that each existing program has a field of specialization. For some problems one program is better while for another problem another program may be optimal. Hence it becomes important to understand what the strong points of FORM are as compared to other systems. There are quite a few that can be mentioned, but the two most evident ones are the speed with which it does its tasks and the size of the expressions it can handle. Actually most FORM instructions are optimized with respect to these two criteria and its architecture is dominated by it.

The result of this is illustrated by the following example which was put together by D. Fliegner. He tuned a so-called random polynomial to an Alpha computer they had at the time at Karlsruhe, which contained 4 Gbytes of memory. Any bigger powers would make Mathematica start swapping. Updated for current memory sizes, the test produces the following results with a recent Mathematica version on a Pentium IV/3 GHz PC:

```
Mathematica 5.0 for Linux
Copyright 1988-2003
Wolfram Research, Inc.

In[1]:= poly = (
 (50*a1*a2^2 + 23*a2 + 17/2151*a4
            + a12)*
 (3*a2 - 12345*a2*a7 + 4*a11 - a4)^6 -
 (24*a7 + 13*a5 - 3121/313*a4 + 2793*a3
            - 13/325*a2)^3*
 (12*a5 + 1/7*a7 - 5*a11)^7 )^3;

In[2]:= Expand[poly]//Length//Timing

Out[2]= {2359.71 Second, 1556851}

In[3]:= MaxMemoryUsed[]

Out[3]= 1593433752
```

FORM processes the same polynomial on the same machine with the results

```
autodeclare symbol a;

l poly = (
 (50*a1*a2^2 + 23*a2 + 17/2151*a4
            + a12)*
 (3*a2 - 12345*a2*a7 + 4*a11 - a4)^6 -
 (24*a7 + 13*a5 - 3121/313*a4 + 2793*a3
            - 13/325*a2)^3*
 (12*a5 + 1/7*a7 - 5*a11)^7 )^3;

.end

    .................

Time = 98.05 sec
Generated terms = 14407695
poly  Terms in output = 1556851
          Bytes used = 84012692
```

We see a difference of almost a factor 25 in speed and 20 in use of memory (the ‘Bytes used’ figure is disk plus RAM, where the actual RAM usage was a mere 13 Mbytes in this run).

In addition FORM can use the hard disk for its intermediate expressions with good efficiency. When expressions become very large the disk is used and only during short periods of time the use of CPU power drops from 99% to around 50% when a disk to disk sort is being executed. Some users have already exceeded

100 Gbytes for the size of their expressions.

**The working of FORM**

Let us have a look at a little FORM program:

```
Symbols a,b,c;
Local F = (a+b+c)^2;
Print;
.end

Time = 0.00 sec
Generated terms = 6
F   Terms in output = 6
     Bytes used = 104

F = c^2 + 2*b*c + b^2
            + 2*a*c + 2*a*b + a^2;
```

FORM works out this formula into component terms. All FORM operations are on individual terms. The terms are sorted when an end of module is encountered. Although the program looks like action is horizontally, it is actually vertically like a big tree:

```
Symbols a,b,c;
Local F = (a+b+c)^2;
Print "<1> %t";
id a = b+2*c;
Print "<2> %t";
Print;
.end
<1>  + a^2
<2>  + b^2
<2>  + 4*b*c
<2>  + 4*c^2
<1>  + 2*a*b
<2>  + 2*b^2
<2>  + 4*b*c
<1>  + 2*a*c
<2>  + 2*b*c
<2>  + 4*c^2
<1>  + b^2
<2>  + b^2
<1>  + 2*b*c
<2>  + 2*b*c
<1>  + c^2
<2>  + c^2

Time = 0.00 sec
Generated terms = 10
F  Terms in output = 3
    Bytes used = 54

F = 9*c^2 + 12*b*c + 4*b^2;
```

The id-statement causes a substitution. The print statement forces a printout of the term each time a term passes the control of the statement. We see that first the first term is worked till the end and then the next term.

The sorting at the end is done by storing the terms, as they arrive into a buffer called the small buffer. Its size is installation dependent. When the small buffer is full and the next term doesn't fit, the small buffer is sorted and the sorted result is put in the large buffer. This is one sorted patch. If the large buffer is too full, the patches in the large buffer are merged and one large sorted patch is written to the intermediate sortfile. Once term generation has finished the patches on disk are merged using the combined small and large buffers for a cache area. The result is written either to memory or to disk, depending on its size. At the start of a new module the old input is discarded and the output file of the previous module becomes the input file for the next module. The terms in this input file are taken one by one and fed to the combined statements of the module.

The above method can work very fast, provided we can indeed step through the input expression linearly, i. e. that no statement needs more than one input term at the same time. We call such statements local statements. The only thoroughly nonlocal operation in FORM is the sort operation in which continuously two terms have to be compared. Once one is restricted to local operations the use of the disk is not a big penalty. One goes through the file linearly and any cache system will work very well. This is contrary to typical nonlocal operations that go through expressions in a quasi random fashion. If the expression is twice the size of the available memory, the chance of a page fault is 50% and the computer will be forced to spend most of its time swapping.

It is amazing how much can be done with the combination of local operations and the sort statement. If one introduces the concept of semi-local operations in which a number of consecutive terms are put together into one new term and constructs such an operation even more can be done. Among the examples is a library for performing large categories of sums involving harmonic sums including programs to solve massive numbers of linear equations to express sums in infinity in terms of a minimal set of Euler-Zagier sums. There is a whole library for determining so called color traces for semi simple Lie Algebras. In addition there exist libraries for doing categories of loop integrals in Perturbative Field Theory. Some of the calculations that involve these last integrals have a tendency to give an enormous intermediate expression swell. One user has to fight regularly to keep his expressions restricted to 400 Gbytes of disk space.

**Some Examples**

FORM is relatively good with noncommuting objects as can be seen here (the default for a function is noncommuting):

```
Functions A,B;
Local F = (A+B)^8;
.sort

Time = 0.00 sec
Generated terms = 256
F  Terms in output = 256
     Bytes used = 5618
repeat id B*A = A*B-1;
Print +s;
.end

Time = 0.01 sec
```

```
Generated terms = 4465
F  Terms in output = 25
         Bytes used = 510

F =  + 105
      - 420*A*A
      + 210*A*A*A*A
      - 28*A*A*A*A*A*A
      + A*A*A*A*A*A*A*A
      + 8*A*A*A*A*A*A*A*B
      + 28*A*A*A*A*A*A*B*B
      - 168*A*A*A*A*A*B
      + 56*A*A*A*A*A*B*B*B
      - 420*A*A*A*A*B*B
      + 70*A*A*A*A*B*B*B*B
      + 840*A*A*A*B
      - 560*A*A*A*B*B*B
      + 56*A*A*A*B*B*B*B*B
      + 1260*A*A*B*B
      - 420*A*A*B*B*B*B
      + 28*A*A*B*B*B*B*B*B
      - 840*A*B
      + 840*A*B*B*B
      - 168*A*B*B*B*B*B
      + 8*A*B*B*B*B*B*B*B
      - 420*B*B
      + 210*B*B*B*B
      - 28*B*B*B*B*B*B
      + B*B*B*B*B*B*B*B
     ;
```

The .sort instruction marks the end of the first module. It forces a sort which in this case is rather useless, but it gives us in the statistics that there are indeed $2^8$ terms as should be in the noncommuting case. This is just a simple example because if the power in the second statement were much higher one could make the program significantly faster by adding a few statements. For instance a brute force application with the power 14 gives for the final statistics:

```
Time = 40.77 sec
Terms in output = 64
Bytes used = 1482
```

Changing the program to

```
Functions A,B;
Local F = (A+B)^14;
.sort
   #do i = 1,1
     id,ifmatch->1,B*A = A*B-1;
       goto 2;
     label 1;
       redefine i "0";
     label 2;
     .sort
   #enddo
Print +s;
.end
```

and forcing intermediate sorts avoids such a pileup of terms. Now the final statistics are

```
Time = 0.88 sec
Generated terms = 64
Terms in output = 64
Bytes used = 1482
```

We see that the possibility to optimize seemingly innocent algorithms is an important property in algebraic programs. Especially the control over when an expression is brought to normal form again (sorted) is important. Too many sorts slow down the execution also.

Actually it is possible to do the above program even much faster, but at the cost of generality in the reduction of the terms. The program

```
Functions A,B;
Local F = 1;
   #do i = 1,14
     Multiply,right,(A+B);
     repeat id,B*A = A*B-1;
     .sort
   #enddo
Print +s;
.end
```

gives for its final statistics

```
Time = 0.01 sec
Generated terms = 64
Terms in output = 64
Bytes used = 1482
```

This time however we tailored the program around a specific problem.

Finally an example from the color library. FORM has a mechanism to call procedures that contain prepared program segments. In addition it knows include files. Here we have put it all together that all is in one file color.h which we call at the beginning of the program. It makes the necessary declarations and loads the relevant procedures.

```
#include- color.h
.global
Local girth6 =
 f(i1,i2,i3)*f(i1,i4,i5)
 *f(i2,i6,i7)*f(i3,i8,i9)
 *f(i4,i10,i11)*f(i5,i12,i13)
 *f(i6,i14,i15)*f(i7,i16,i17)
 *f(i8,i18,i19)*f(i9,i20,i21)
 *f(i10,i21,i15)*f(i13,i19,i14)
 *f(i17,i11,i18)*f(i12,i16,i20);
sum i1,...,i21;
.sort

Time = 0.00 sec
Generated terms = 1
girth6   Terms in output = 1
              Bytes used = 178
#call color
#call SORT(tloop-1)

Time = 0.04 sec
Generated terms = 19
girth6  Terms in output = 19
          tloop-1 Bytes used = 682
#call adjoint
```

```
#call SORT(tloop-2)

Time = 0.04 sec
Generated terms = 13
girth6  Terms in output = 13
          tloop-2 Bytes used = 440
#call simpli
id  acc(x?) = x;
Print +f +s;
.end

Time = 0.17 sec
Generated terms = 3
girth6  Terms in output = 3
               Bytes used = 82

girth6 =
      + 1/648*NA*cA^7
      - 8/15*d444(pA1,pA2,pA3)*cA
      + 16/9*d644(pA1,pA2,pA3)
     ;
```

In the above we consider a Feynman diagram with 14 three point vertices and girth 6. At each vertex is an element of the adjoint representation. The notation is as in reference [1]. It should be noted that by the time we computed this quantity it was not yet known to mathematicians. The file color.h contains all declarations and procedures for this package and color, adjoint and simpli are the main routines, the first usually doing ordinary representations, the second the adjoint representation and the third one reducing the answer to topological invariants. The value of these invariants for the various groups and representations are given in the paper. All programs ran on a PIV-2800 notebook, unless mentioned otherwise.

To conclude: FORM is a fast and versatile program for working in many fields of science. It doesn't have much knowledge built in but it allows the construction of libraries that can deal with large varieties of tasks. FORM is available from the site `http://www.nikhef.nl/~form`. There are executables for several types of computers, the manual and a tutorial, several packages and the article [2].

## Literatur


[1] T. van Ritbergen, A.N. Schellekens and J. A. M. Vermaseren: Group Theory Factors for Feynman Diagrams, Int. J. Mod. Phys. A14, 41–96, 1999.

[2] J. A. M. Vermaseren: New Features of FORM, MATH-PH/0010025.

# MuPAD Pro 3.1.1 für Apple MacOS X

**Martin Knelleken (SciFace Software, Paderborn)**

Während MuPAD Pro 3 für Microsoft Windows bereits Anfang letzten Jahres mit vielen mathematischen Verbesserungen gegenüber der Version 2.5 und einer vollständig neu implementierten Grafik mit realistischem Beleuchtungsmodell, transparenten 3D-Flächen, flexiblen Animationsmöglichkeiten sowie massiver Interaktivität aufwartete (`http://research.mupad.de/doc/31/eng/changes.html`), mussten sich die Besitzer von Apple Macintosh Computern noch etwas gedulden.

Nun ist es soweit: MuPAD Pro 3.1.1 für MacOS X wird auf der DIDACTA vom 28. Februar bis 4. März 2005 in Stuttgart erstmals vorgestellt und wird ab Mitte März im Web sowie im Handel verfügbar sein. Neben den mathematischen Verbesserungen und den erweiterten grafischen Fähigkeiten bietet die neue Version für MacOS X insbesondere folgende Neuerungen:

- Ein neues MacOS X konformes Oberflächendesign mit modernen Bedienelementen wie beispielsweise benutzerkonfigurierbare Symbolleisten.
- MuPAD Notebooks im Windows-Format können gelesen und geschrieben werden.
- Zuschaltbares Anti-aliasing bei 2D-Grafiken.
- Beim RTF-Export von MuPAD-Notebooks kann die Applikation angegeben werden, mit dem das exportierte Dokument automatisch geöffnet werden soll (z. B. TextEdit, Microsoft Word, ...).
- MuPAD kann die von anderen Programmen angebotenen Dienste nutzen. Ein MuPAD-eigener Dienst ist bereits in Vorbereitung.
- Beliebige Finder-Objekte können nun einfach per Drag & Drop in MuPAD-Notebooks eingebettet und dann jederzeit per Doppelklick mit der Maus geöffnet werden. Sie erhalten so multimediale Dokumente.
- Zur Beschleunigung der Numerik wird in Kürze das in MuPAD verwendete Package Scilab nun auch für MacOS X zur Verfügung stehen.
- Viele weitere Detailverbesserungen erhöhen den Bedienkomfort und die Arbeitsgeschwindigkeit deutlich. MuPAD Pro 3 Vollversionen mit einer 30-Tage-Evaluierungslizenz erhalten Sie im Web unter `http://www.sciface.com/download.html`. Hier wird ab Mitte März auch MuPAD Pro 3.1.1 für MacOS X zur Verfügung stehen.

# Geogebra: Dynamische Geometrie mit etwas Algebra

**Reinhard Oldenburg (Göttingen)**

Markus Hohenwarter hat mit seinem Programm Geogebra (Synthese aus „Geometrie“ und „Algebra“, `www.geogebra.at`) eine neue Programmgattung geschaffen. In seiner Vorstellung von Geogebra im letzten Computeralgebra-Rundbrief (Nr. 35) begründet er aus überzeugenden didaktischen Argumenten den Wunsch nach einer Verbindung von Computeralgebrasystemen (CAS) und Dynamischer Geometriesoftware (DGS). Als Ziel skizziert er eine Bidirektionalität beider Programmarten, bei denen ein leichter Wechsel der Repräsentationsform statt finden kann. Allerdings wird kurz darauf eingeschränkt, dass „sich nicht alle Funktionen eines DGS bzw. CAS gleich gut“ für eine solche Verbindung eignen. In welchem Umfang löst Geogebra nun das Versprechen der Bidirektionalität ein? Um diese Frage zu beantworten, wird hier ein Blick auf die aktuelle Version Geogebra 2.4 geworfen.

Das Programm kann umgehen mit Punkten, Geraden, Strecken, Strahlen, Kreisen, Kegelschnitten und Funktionsgraphen. Bis auf die Funktionsgraphen können alle Objekte sowohl geometrisch als auch über ihre Gleichungen bzw. Koordinaten erzeugt werden. Es stehen eine Reihe von geometrischen Konstruktionswerkzeugen (Parallele, Lot, Mittelsenkrechte, Tangente, Winkelhalbierende) zur Verfügung.

**Geogebra als DGS**

Die üblichen Konstruktionen der Schulgeometrie können mit den vorhandenen Werkzeugen durchgeführt werden. Verglichen mit anderen DGS-Programmen wie Euklid, Cabri, Cinderella und ZuL fehlen vor allem folgende Fähigkeiten: die Erzeugung von Ortslinien (ist geplant) sowie Makros. Diese Dinge lassen sich aber in das vorhandene Konzept einbauen, so dass hier künftige Versionen sicher Fortschritte bringen werden.

Bei DGS zeigt sich oft in bestimmten Situationen ein unstetiges Verhalten beim Verziehen der Konstruktion mit der Maus. J. Richter-Gebert und U. Kortenkamp haben mit ihrer DGS Cinderella (`www.cinderella.de`) eine überzeugende Lösung auf der Basis der komplexen projektiven Geometrie gefunden. Auch Geogebra versucht Unstetigkeiten zu vermeiden [Hohenwarter: GeoGebra – dynamische Geometrie und Algebra. In: Der Mathematikunterricht 4/2003]. Die dazu eingesetzten heuristischen Verfahren schaffen das allerdings nicht annähernd so gut wie bei Cinderella: Schon die übliche Zirkel-und-Lineal-Konstruktion der Winkelhalbierenden zweier Geraden zeigt unstetiges Verhalten, wenn die Kreise kleiner werden und schließlich durch den Punkt mit Radius 0 durchgeschoben werden. Die didaktische Diskussion hat aber gezeigt, dass die Stetigkeit nicht nur Vorteile bringt, und so kann diese pragmatische Realisierung akzeptiert werden.

Wie andere DGS auch unterscheidet Geogebra zwischen Basisobjekten, die mit der Maus bewegt werden, und abhängigen (berechneten) Objekten, die nicht direkt, sondern nur in Folge der Änderung eines Basisobjektes bewegt werden. Intern wird diese Abhängigkeit durch einen gerichteten Abhängigkeitsgraphen repräsentiert. Zusammenfassend lässt sich sagen, dass Geogebra eine brauchbare DGS darstellt, die noch kleinere Lücken hat, die sich aber sicherlich bald stopfen lassen werden.

**Geogebra als CAS und als CAS-DGS**

Wenn man aus der Ankündigung einer bidirektionalen Verknüpfung von CAS und DGS folgert, Geogebra wie ein CAS handhaben zu können, wird man enttäuscht: Geogebra kann auf der Benutzeroberfläche nicht mit symbolischen Ausdrücken umgehen. Variablen haben immer einen Wert. Den kann man aber komfortabel und recht dynamisch durch Klicken ändern. Die Grundfunktionalität von CAS wie allgemeine Terme umformen und Gleichungen lösen wird nicht geboten. Lediglich Polynome in zwei Variablen $x, y$ bis zu zweitem Grad können eingegeben werden und werden dann zu einer Standardform vereinfacht. Geogebra erkennt dann, dass es sich um einen Kreis, einen Kegelschnitt oder eine Gerade handelt und erzeugt dieses Objekt auch geometrisch. Bei allen Objekten lässt sich auch eine algebraische Repräsentation sichtbar machen und auf Ebene der Koordinaten der Objekte ist die Bidirektionalität auch voll umgesetzt. Beim Ziehen am Mittelpunkt eines Kreises sieht man also seine Kreisgleichung etwa als $(x - 2.17)^2 + (y - 3,77)^2 = 3,1^2$, wobei sich die numerischen Koeffizienten beim Ziehen laufend aktualisieren.

Umgekehrt kann man die Zahlen ändern und hat sofort die Änderung im Geometriefenster. Man muss sich aber klar machen, dass man hier eigentlich nicht mit algebraischen Objekten arbeitet, sondern, dass man eher „Arithmetik in Schablonen“ als Algebra treibt. Es ist beispielsweise nicht möglich, den Exponenten oder die Namen der Variablen $x, y$ zu ändern. Allerdings kann die Definition des Objektes geändert werden, d. h. das ganze Objekt wird durch ein neues ersetzt.

CAS-Funktionalität existiert aber im Programm und wird punktuell verwendet: Zu einem eingegebenen Funktionsterm kann die Ableitung symbolisch gefunden werden. Zusammenfassend muss man urteilen, dass Geogebra kein CAS ersetzen kann. Dies will es allerdings auch gar nicht. Aber auch in dem speziellen Rahmen der geometrischen Anwendungen der Algebra sind viele naheliegende Operationen wegen der mangelnden CAS-Funktionalität unmöglich:

- Ortskurven lassen sich nicht algebraisch berechnen,
- in den Gleichungen der Objekte stehen immer nur numerische Koeffizienten, die parametrische Abhängigkeit von den Variablen andere Objekte lässt sich nicht algebraisch darstellen,
- algebraische Beweise geometrischer Sätze sind unmöglich und
- Schnittpunkte, Asymptoten etc. können nicht symbolisch berechnet werden.

Wie oben beschrieben verwendet Geogebra die DGS-typische Repräsentation der Konstruktion mit einem Abhängigkeitsgraphen, der folglich auch auf in der algebraischen Repräsentationsform grundlegend ist. Diese funktionale Modellierung (Koordinaten der Objekte sind Funktionen der Objekte, aus denen sie konstruiert wurden) kollidiert mit der üblichen relationalen Sichtweise der Algebra. Man kann in Geogebra keine Gleichungen eingeben, die Relationen im Geometriefenster beschreiben. Dass dies prinzipiell möglich ist, zeigt mein prototypisches System FeliX (siehe z. B. Rundbrief Nr. 34), es stellt eine qualitativ andere Stufe der Bidirektionalität von CAS und DGS dar, weil diese nicht nur auf der Ebene der Koordinaten von Objekten, sondern auch auf der Ebene der Gleichungen gegeben ist: Der Benutzer kann beliebige Gleichungen und Ungleichungen eingeben, und damit die „Konstruktion“ ändern. Allerdings ist für den Einsatz in der Schule nicht die konsequente Umsetzung eines Prinzips entscheidend, sondern die praktische Einsetzbarkeit durch die Schüler.

**Benutzeroberfläche**

Geogebra präsentiert ein übersichtliches Fenster mit den beiden Hauptbereichen für die geometrische und algebraische Repräsentation der Objekte sowie eine einzeilige Kommandoeingabe. Durch die Beschränkung der Funktionalität kann sich Geogebra eine weitgehend untechnische Eingabe leisten. So erzeugt $y = 2x-1$ sofort eine Gerade, ohne dass dazu eine Erzeugungsfunktion benutzt werden müsste. Die Eingabe von $2x - 1$ dagegen erzeugt einen Funktionsgraphen, der der Geraden täuschend ähnlich sieht, auf den man aber z. B. keine Lote fällen kann. Es ist also unumgänglich, dass sich die Schüler über die Bedeutung dieser unterschiedlichen Datentypen klar werden.

Aus dem gleichen Grund führt $y = \frac{1}{x}$ zu einer Fehlermeldung, während $\frac{1}{x}$ den Graphen der Hyperbel zeichnet. An vielen Details merkt man, dass das Programm „mitdenkt“: So erzeugt etwa die Eingabe von $(x - 7)^2 + (y - 2)^2 = 0$ ein Objekt, das Geogebra als Punkt ausweist. Durch nachträgliche Änderung der Gleichung ändert sich auch der angezeigte Typ (nicht aber der interne – es bleibt intern immer ein Kegelschnitt).

Besonders schön ist die Behandlung von numerischen Parametern in der Konstruktion: Neue Parametervariablen führt man z. B. mit $a = 3$ ein. Dieser Wert kann dynamisch verändert werden und zieht die Änderung aller abhängigen Objekte, z. B. Funktionsgraphen, die von $a$ abhängen, nach sich. Bei der Behandlung von Kurvenscharen erweist sich das als äußerst angenehm. Ab Version 2.5 wird es dafür auch Schieberegler geben, mit denen die Veränderung noch intuitiver wird.

Sehr praktisch ist auch das einfache Rechnen mit Vektoren sowie deren graphische Darstellung. Damit lassen sich weite Teile der zweidimensionalen linearen Algebra anschaulich und dynamisch erleben.

Fazit: Schüler können mit der aufgeräumten Oberfläche und der recht fehlertoleranten Eingabezeile gut umgehen. Für die Behandlung von Kegelschnitten und Kurvenscharen bieten sich interessante Möglichkeiten. Geogebra stellt insgesamt eine wertvolle Bereicherung des Spektrums der mathematischen Unterrichtssoftware dar. Es ist schon fast eine vollwertige DGS und leistet vieles, wozu in der Schule üblicherweise ein CAS wie Derive eingesetzt wird (Plotten, numerisches Lösen von Gleichungen). Der versprochene Brückenschlag von DGS und CAS (im Sinne des symbolischen Rechnens) wird allerdings nur ansatzweise geleistet.

## Computeralgebra in der Schule

# Der CAS-Schulversuch an den Technischen Gymnasien Baden-Württembergs

**Siegfried Schwehr (Emmendingen)**

*Dem kürzlich verstorbenen Klaus-Dieter Roth gewidmet, einem Wegbereiter der CAS-Schulversuche an den beruflichen Schulen.*

Erfahrungsgemäß ist das berufliche Schulwesen weniger gut bekannt als das allgemeinbildende. Daher will ich einige Informationen dazu geben. Etwa 30% aller Abiturienten in Baden-Württemberg erwerben ihren Abschluss an einem beruflichen Gymnasium oder einer technischen bzw. wirtschaftlichen Oberschule, mehr als der bundesweite Durchschnitt von 27,5 %, was – mit anderen Aspekten zusammen – dem Land ein großes Lob in der TOSCA-Studie einbrachte (Köller et al. 2004). Um die Aufnahme an einem beruflichen Gymnasium können sich Schülerinnen und Schüler mit mittlerer Reife bewerben, bei Realschulabsolventen ist ein geeigneter Notenschnitt verlangt. Zur Oberstufe des beruflichen Schulwesens gehören noch die zahlreichen Berufskollegs, die zur Fachhochschulreife führen, sowie einige Berufsoberschulen, an denen man nach einer abgeschlossenen Berufsausbildung die Hochschulreife erwerben kann.

In Baden-Württemberg gab und gibt es mehrere Schulversuche zum CAS-Einsatz in der Sekundarstufe II. In allgemeinbildenden Gymnasien wurde von 1996 bis 2000 das Pilot-Projekt „Mobiles Klassenzimmer“ durchgeführt (PiMoKl, vgl. Henn 2004). Im Rahmen dieses Projekts führen derzeit drei Arbeitsgruppen Schulversuche zum Thema „Computeralgebrasysteme im Mathematikunterricht“ durch: Das PiMoKl-Projekt wird unter dem Namen Maple-Gruppe durchgeführt, eine Gruppe „MathCom“ arbeitet mit Maple, und eine dritte Gruppe arbeitet mit Kleinrechnern (TI-92, Voyage und anderen). Der Modellversuch „Einsatz computerunterstützter Lernprogramme (CBT) in beruflichen Schulen“ dauerte von 1994 bis 1998; er wurde vom Landesinstitut für Erziehung und Unterricht (LEU) in Stuttgart betreut. Hier wurden unter Anderem Einsatzmöglichkeiten von Derive in der beruflichen Oberstufe untersucht (LEU-Berichte B-98/01b, B-98/01c sowie die LEU-Handreichungen H-98/39a und H-98/39b). An einigen Berufskollegs kann die Fachhochschulreifeprüfung mit CAS abgelegt werden. CAS wird inzwischen auch im Abitur an den technischen und wirtschaftlichen Oberschulen eingesetzt.

Der CAS-Schulversuch an den technischen Gymnasien (TG) läuft seit 1998. Die Entscheidung über die Teilnahme (oder den Ausstieg) liegt bei der Fachkonferenz der jeweiligen Schule. Dieser Beschluss ist der vorgesetzten Behörde mitzuteilen. Im Rahmen dieses Schulversuchs wurden Beispielaufgaben entwickelt (LEU H-99/41). Bei dem BLK-Projekt „Weiterentwicklung der Unterrichtskultur im Mathematikunterricht" (WUM) wurde auf Fortbildungen ein Modul CAS angeboten (vgl. LEU H-02/21).

An den Schulen werden im Allgemeinen TI-Rechner (89, 92, 92+, Voyage), seltener PCs mit Maple oder Derive eingesetzt. Die bekannten Argumente dafür sind die ständige Verfügbarkeit, die leichte Transportabilität und der Preis. Je nach Schule bezahlen die Schüler diese Rechner ganz oder zum Teil selbst, oder sie leihen sie aus. Die gute Akzeptanz des Schulversuchs lässt sich daran ablesen, dass 2004 über 50% der Schülerinnen und Schüler am TG ein CAS-Abitur abgelegt haben.

Im beruflichen Schulwesen werden von den Schulen Aufgabenvorschläge angefordert. Eine sogenannte Aufgabenauswahlkommission wählt aus diesen Vorschlägen Aufgaben aus; im Allgemeinen müssen die ausgewählten Vorschläge noch überarbeitet werden, da sie für das ganze Land verbindlich sein sollen. Die Sammlung der Abituraufgaben wird daher gerne als „heimlicher Lehrplan" bezeichnet. Durch dieses Verfahren sind die Lehrkräfte in den Weiterentwicklungsprozess eingebunden. Im (ehemaligen) Grundkurs hatten die Schüler zwei, im (ehemaligen) Leistungskurs drei Aufgaben zu bearbeiten. Typischerweise besteht eine Aufgabe aus 4 bis 5 Teilaufgaben. Aufgabentypen wie etwa ein mathematischer Aufsatz sind bei zentralen Prüfungen (bis jetzt) kaum denkbar.

Die nachfolgend behandelten Aufgabenbeispiele zeigen, wie durch behutsame „Akzentverschiebungen" (vgl. Henn 2004) die Aufgaben langsam in Richtung der im Lehrplan von 1998 angegebenen Ziele weiterentwickelt wurden: mehr inhaltliches Verständnis und weniger Kalküle (s. auch Borneleit et al 2001), Modellieren, Verwendung grafischer Methoden usw. Die „klassische Kurvendiskussion" verliert immer mehr an Wert, die Aufgaben werden offener.

Es folgen einige Beispiele aus dem Prüfungszeitraum von 2000 bis 2004, zuerst aus dem Bereich Lineare Algebra und Vektorgeometrie.

In einer Leistungskursaufgabe (Nachprüfung 2000) findet sich folgende Teilaufgabe:

Gegeben sind für $0 < k < 1$ die Ebenen $E_k$ : $kx + 2y - z = 2$, $F_k : x + (k+1)y + z = -1$ und $G_k : 2x + y + (k-1)z = -2$. Die Schnittpunkte von $E_k$ mit der $x$-Achse, von $F_k$ mit der $y$-Achse sowie von $G_k$ mit der $z$-Achse bilden ein Dreieck. Zu untersuchen ist, für welchen Wert des Parameters $k$ dieses Dreieck minimalen Flächeninhalt hat.

Die für diese Aufgaben charakteristischen Inhalte sind die Berechnung des Flächeninhaltes $A(k)$ sowie die analytische Untersuchung dieser Funktion. Den Flächeninhalt bekommt man als Betrag des Kreuzprodukts geeigneter Vektoren; es ergibt sich hierfür

$$A(k) = \sqrt{\frac{6k^2 + 6k + 5}{k^2(k^2-1)^2}}$$

mit $0 < k < 1$. Alle erforderlichen Rechnungen können mit dem CAS durchgeführt werden. Dass der Flächeninhalt in etwas komplizierter Weise vom Parameter $k$ abhängt, spielt hier für die Bearbeitung keine Rolle. Der Schwierigkeitsgrad der Aufgabe wird durch die Inhalte und nicht (wie früher oft) durch die Komplexität der erforderlichen Termumformungen festgelegt. Eine Aufgabe wie diese wäre daher für eine Prüfung ohne CAS wohl nicht gestellt worden. Hier zeigt sich ein für die Lehrkräfte wichtiger Nebeneffekt: Beim Entwurf von (Prüfungs-)aufgaben – eine aufwändige Arbeit – braucht man nicht (zu sehr) auf die Einfachheit der entstehenden Ausdrücke zu achten. Der CAS-Einsatz sollte aber umgekehrt nicht dadurch begründet sein, dass „endlich" komplizierte Ausdrücke bearbeitet werden können. Andere Beispiele für die Verwendung analytischer Methoden sind etwa die Berechnung des maximalen Abstands eines parameterabhängigen Punktes von einer fest gewählten Ebene oder der maximale Winkel zwischen einer Ebene und einer parameterabhängigen Geraden.

Eine deutliche Akzentverschiebung findet man in einer Leistungskursaufgabe von 2000:

Es bezeichne $d_B(t)$ den Abstand des Punktes $A_t(1-t|1+2t|t^2)$ vom Punkt $B(1|2|-2)$ und $d_C(t)$ den Abstand des Punktes $A_t$ vom Punkt $C(5|2|2)$. Die Schaubilder der Funktionen $d_B(t)$ und $d_C(t)$ sind zu skizzieren. Anhand dieser Schaubilder soll entschieden werden, wie viele Punkte $A_t$ es gibt, die von $B$ und $C$ denselben Abstand haben, welcher der beiden Punkte $B$ und $C$ näher an der durch die Punkte $A_t$ gebildeten Kurve liegt, sowie welche Punkte $A_t$ näher an $B$ als an $C$ liegen. Mit dem Rechner lassen sich zur Erstellung der Skizzen schnell Wertetabellen erstellen. Die Aufgabe wird durch Interpretation dieser Schaubilder gelöst.

Im Bereich der Analysis wurden Akzentverschiebungen über die Jahre verteilt sehr behutsam vorgenommen. In den ersten Prüfungsjahren wurden „klassische" Kurvendiskussionen (mit und ohne Parameter) verlangt. Mit der Zeit wurden diese Inhalte in anderen Aufgaben abgeprüft, wie das folgende Beispiel einer Leistungskursaufgabe von 2004 zeigt:

Gegeben ist eine Schar von Funktionen durch $f_t(x) = 2x^3 - 6tx^2 + 7x (x \in \mathbb{R})$. Das Schaubild von $f_t$ ist $K_t$.
a) Zeichnen Sie $K_{1,2}$. Die Normale im Wendepunkt schließt mit $K_{1,2}$ zwei Flächenstücke ein. Berechnen Sie den Inhalt der Gesamtfläche.
b) Ermitteln Sie, für welchen Wert des Parameters $t$ das Schaubild $K_t$ zwei Punkte, einen oder keinen Punkt mit

waagerechter Tangente hat. Skizzieren Sie für jeden dieser Fälle ein Schaubild in ein gemeinsames Koordinatensystem.
c) Bestimmen Sie drei verschiedene Funktionen, deren Schaubilder aus $K_{1,2}$ durch Verschiebung hervorgehen und mit der $x$-Achse genau zwei Punkte gemeinsam haben.
d) Hier wird die Trassierung einer Straße untersucht. Die Daten sind einer Zeichnung zu entnehmen. Die Qualität der berechneten Lösung ist zu beurteilen.

In allen diesen Aufgabenteilen sind klassische Inhalte gefragt, es gibt aber keine Kurvendiskussion. Das zu Aufgabenbeginn zu zeichnende Schaubild ist Ausgangspunkt für weitere Fragen. In b) kann man auch experimentell vorgehen. Aufgabe c) ist eine offene Aufgabe; verschiedene Antworten sind möglich. Trassierungen zu berechnen gehört seit längerer Zeit (vgl. Scheuermann 1998) fast zum Standard. Neu ist die Aufforderung, die Qualität der Lösung verbal zu beurteilen, eine auch für die korrigierenden Lehrkräfte neue Herausforderung.

Die Palette der in Aufgaben verwendbaren Funktionen wurde erweitert, da mit Computeralgebrasystemen wie auch mit grafikfähigen Taschenrechnern Integrale numerisch berechnet werden können. So gibt es inzwischen mehrere Aufgaben, in denen nicht elementar integrierbare Funktionen wie z.B. $f(x) = e^{x^2}$ vorkommen.

Wie oben schon angedeutet versucht man immer wieder behutsam, in diesen zentralen Prüfungen etwas offenere Aufgaben in bekanntem Kontext zu stellen. So wurde im Leistungskurs 2004 die klassische Steckbriefaufgabe modifiziert: Bestimmen Sie zwei verschiedene Polynomfunktionen dritten Grades, deren Schaubilder folgende Eigenschaften besitzen: Sie berühren die Gerade mit der Gleichung $y = 4$ im Ursprung und sie schneiden die $x$-Achse an der Stelle $x = 2$ .

Im selben Prüfungsjahrgang findet sich ein Beispiel dafür, dass man für die Rechenarbeit das CAS verwendet und sich auf die inhaltlichen Fragen konzentriert: Berechnen Sie den Mittelwert von $f(x) = \sin(7x) + \sin(9x)$ im Bereich $0 \leq x \leq \pi$ sowie im Bereich $-\frac{\pi}{2} \leq x \leq \frac{\pi}{2}$ . Interpretieren Sie die Ergebnisse. Für das erste Integral ergibt sich $\frac{32}{63\pi}$, also näherungsweise 0,162, für das zweite Integral kommt Null heraus. Man kann die Ergebnisse als Hinweise auf die jeweilige Bilanz der Flächeninhalte interpretieren; das 2. Ergebnis kann man auch mit der Punktsymmetrie begründen.

Aufgrund der Schulreform in Baden-Württemberg ist seit 2002/2003 ein neuer Lehrplan in Kraft. Es gibt keine Grund- und Leistungskurse mehr. An den Schulen werden entweder grafikfähige Taschenrechner oder CAS eingesetzt. An dem CAS-Schulversuch können sich auch nicht-technische Gymnasien beteiligen. Die Prüfungsmodalitäten wurden geändert. Über diese Änderungen, insbesondere im Hinblick auf die Aufgaben, soll bei späterer Gelegenheit berichtet werden.

Für wertvolle Hinweise danke ich Frau Annemarie Ahring-Novak (Stuttgart) sowie Herrn Dr. Gerhard Keller (Karlsruhe; ehem. LEU Stuttgart ).

**Literatur:**

Borneleit, P; R. Danckwerts; H.-W. Henn; H.-G- Weigand: Expertise zum Mathematikunterricht in der Oberstufe. Journal für Didaktik der Mathematik 22 (2001), S. 73 -90.

Henn, H.-W.: Computer-Algebra-Systeme- junger Wein oder neue Schläuche? Journal für Didaktik der Mathematik 25 (2004), S. 198 - 220.

Köller, O.; R. Watermann; U. Trautwein; O. Lüdtke: Wege zur Hochschulreife in Baden-Württemberg. TOSCA - eine Untersuchung an allgemeinbildenden und beruflichen Gymnasien. MPI für Bildungsforschung Berlin (2004).

LEU-Berichte B-98/01b: Modellversuch Einsatz computerunterstützter Lernprogramme (CBT) in Beruflichen Schulen. Fachband Mathematik Derive. Landesinstitut für Erziehung und Unterricht Stuttgart 1998.

LEU-Berichte B-98/01c : Modellversuch Einsatz computerunterstützter Lernprogramme (CBT) in Beruflichen Schulen. Fachbereich Mathematik/Physik Abschlußbericht (Juli 1998). Landesinstitut für Erziehung und Unterricht Stuttgart 1998.

LEU-Handreichungen H-98/39a: CAS-Grundlagen. Landesinstitut für Erziehung und Unterricht Stuttgart 1998.

LEU-Handreichungen H-98/39b: Derive. Landesinstitut für Erziehung und Unterricht Stuttgart 1998.

LEU-Handreichungen H-99/41: CAS-Pilotaufgaben. Landesinstitut für Erziehung und Unterricht Stuttgart 1999.

LEU-Handreichungen H-00/29: CAS-Pilotaufgaben für BOS. Landesinstitut für Erziehung und Unterricht Stuttgart 2000.

LEU-Handreichungen H-02/21: WUM (BS)-Materialien für die Lehrerfortbildung. Landesinstitut für Erziehung und Unterricht Stuttgart 2002.

Scheuermann, H.: Computereinsatz im anwendungsorientierten Analysisunterricht. Verlag Franzbecker, Hildesheim 1998.

# Neuer internationaler Master-Studiengang Computational Mathematics beginnt an der Universität Kassel

**Wolfram Koepf (Kassel)**

Ein neuer Master-Studiengang Computational Mathematics beginnt zum Wintersemester 2005/2006 am Fachbereich Mathematik/Informatik der Universität Kassel. Der zugehörige Bachelor-Studiengang hatte zum Wintersemester 2002/2003 begonnen. Hierüber war im Computeralgebra-Rundbrief 30 berichtet worden. Die beiden Studiengänge waren als erste ihrer Art und ihres inhaltlichen Zuschnitts in Deutschland akkreditiert und von einem Expertenteam als hervorragend bewertet worden. Die Studiengänge bieten eine zukunftsträchtige Verzahnung von Mathematik und Informatik. In ihrem Aufbau und in der Art der Prüfungen entsprechen die Studiengänge internationalen Strukturen. Der nun beginnende Master-Studiengang baut auf dem Bachelor in Computational Mathematics oder einem gleichwertigen Bachelor-Studiengang auf und schließt nach drei Semestern mit dem Master of Science ab. Die Berufschancen der Absolventen des Studiengangs werden als ausgezeichnet angesehen.

### Was ist Computational Mathematics?

Sowohl in Anwendungsbereichen als auch bei der Schaffung und Erforschung von eigenen Theorien hat sich die Mathematik dank der enormen Entwicklung der Computer in den letzten Jahrzehnten zunehmend komplizierterer Aufgabenstellungen annehmen können. Insbesondere ermöglichen Computer umfangreiche Berechnungen und dadurch im Sinne naturwissenschaftlicher Experimente Beobachtungen, die wiederum Anstöße für Problemlösungen sowie weitere Theorie- und Modellbildungen geben. Die Studiengänge Computational Mathematics an der Universität Kassel beschäftigen sich speziell mit der Bearbeitung von Fragestellungen und der Lösung von Problemen der Diskreten Mathematik, insbesondere der Gebiete Algebra, Gruppentheorie, Zahlentheorie, Arithmetische Geometrie, Algebraische Summations- und Integrationstheorie, Kombinatorik, Kryptographie und Kodierungstheorie, mit Hilfe des Einsatzes von Computern. Neben innermathematischen Fragestellungen gehören hierzu beispielsweise alle modernen Verschlüsselungsverfahren, die zur sicheren Datenübertragung im Internet benötigt werden. Dabei werden Computeralgebrasysteme sowie eigenständige Computerprogramme verwendet, die auf exakten und symbolischen Rechenmethoden beruhen.

### Ziele und Aufbau des Master-Studiengangs Computational Mathematics

Entsprechend sind die Ziele im Master-Studiengang Computational Mathematics die Vermittlung der wissenschaftlichen Zusammenhänge und der Erwerb der für die berufliche Tätigkeit erforderlichen fachlichen Kenntnisse und Fähigkeiten. Berufsfelder liegen in dem zukunftsträchtigen und sich schnell wandelnden Grenzgebiet zwischen Mathematik und Informatik.

Das Studium soll grundlegende theoretische und praktische Kenntnisse der Mathematik und Informatik vermitteln. Aus Sicht der Computational Mathematics gehört dazu insbesondere das Zusammenspiel von mathematischen Methoden und deren Umsetzung in Algorithmen. Die Beherrschung von Computeralgebrasystemen sowie Erfahrung im Umgang mit mathematischer Software sind dabei wichtige Studienziele. Der Master-Abschluss berechtigt zur Promotion. Aufgrund der studienbegleitenden Prüfungen ist eine Studienzeit von 3 Semestern möglich.

### Informationen und Kontakt

Informationen über die Akkreditierung sind auf der Internetseite `http://www.akkreditierungsrat.de` des Akkreditierungsrates nachzulesen.

Eine ausführliche Beschreibung des Studiengangs findet sich im Internet unter `http://www.mathematik.uni-kassel.de/cm`.

Zusendung der Studieninformation über den Studienservice der Universität Kassel, Tel. 0561 804 2209.

Weitergehende Auskünfte sind über `master-cm@mathematik.uni-kassel.de` zu erhalten.

# Arbeitsgruppe Algebra und diskrete Mathematik im Institut „Computational Mathematics“ an der TU Braunschweig

**Bettina Eick (Braunschweig)**

An der TU Braunschweig ist im WS 2003/2004 die Arbeitsgruppe Algebra und diskrete Mathematik im Institut „Computational Mathematics“ eingerichtet worden. Diese Arbeitsgruppe ist der Nachfolger einer entsprechenden Arbeitsgruppe im Institut für Geometrie.

Die Arbeitsgruppe beschäftigt sich unter anderem mit algorithmischen Methoden in der Algebra, Geometrie und diskreten Mathematik und hat eine enge Verbindung zur Computeralgebra. Einige Mitglieder der Arbeitsgruppe beteiligen sich an der Entwicklung von Software zum Computeralgebrasystem GAP und schreiben eigenständige Packages für das GAP-System.

**Mitglieder:** Dr. Jens-Peter Bode, Prof. Dr. Bettina Eick, apl. Prof. Dr. Arnfried Kemnitz, Dipl.-Math. M. Marangio, Dr. Harm Pralle, Dipl.-Inf. Christian Sievers sowie eine Reihe Diplomanden.

**Ehemalige Mitglieder:** Dipl. Math. Björn Assmann, Dr. Burkhard Höfling, Dr. Csaba Schneider.

**Arbeitsgebiete:**

- Gruppentheorie (Eick et al.): Algorithmische Gruppentheorie, Konstruktion von Gruppen kleiner Ordnung, Untersuchung von Gruppen, insbesondere polyzyklische und freie Gruppen sowie $p$-Gruppen
- Geometrie (Pralle): Algorithmische Untersuchung und Konstruktion von Geometrien, Hyperebenen, Faserungen und Ovoide (dualer) Polarräume
- Lie-Algebren (Eick, Pralle): Konstruktion und Untersuchung von Lie-Algebren über endlichen Körpern, Darstellungstheorie solcher Lie-Algebren
- Kryptographie (Eick et al.): Analyse von Kryptosystemen aus nicht-kommutativen Gruppen
- Kristallographische Gruppen (Eick): Konstruktion und Untersuchung von kristallographischen Gruppen
- Graphentheorie (Kemnitz, Marangio): Algorithmische Graphentheorie, Färbungen, Kreise, Darstellungen von Graphen
- Diskrete Mathematik (Bode, Kemnitz): Reguläre und halbreguläre Körper, Newtonsche Zahlen, Polyominos

**Software:** Die Arbeitsgruppe beteiligt sich an der Entwicklung von Algorithmen zum Computeralgebrasystem GAP. Dazu sind von Mitgliedern der Arbeitsgruppe diverse Pakete zu GAP entwickelt worden. Während der letzten zwei Jahre haben die Mitglieder der Arbeitsgruppe an der Entwicklung und Weiterentwicklung der folgenden Pakete mitgewirkt:

- Alnuth (Assmann, Distler, Eick): Ein Interface zwischen GAP und KANT
- Polycyclic (Eick, Nickel): Untersuchung polyzyklischer Gruppen
- SmallGroups (Eick, O’Brien, Besche): Eine Bibliothek von Gruppen kleiner Ordnung
- FGA (Sievers): Algorithmen für freie Gruppen
- IrredSol (Höfling): Eine Bibliothek von auflösbaren irreduziblen Matrixgruppen
- Polenta (Assmann): Algorithmen für polyzyklische rationale Matrixgruppen
- Radiroot (Distler): Ein Algorithmus zum Lösen eines Polynoms durch Radikale
- CFgroups (Dietrich): Berechnung der Gruppen von kubikfreier Ordnung
- Sophus (Schneider): Konstruktion nilpotenter Lie Algebren

**Literatur:** Handbook of Computational Group Theory. Derek Holt, Bettina Eick, Eamonn O’Brien. CRC Press.

**Homepage:** `http://www.icm.tu-bs.de/ag_algebra`

# Arbeitsgruppe Computeralgebra an der Technischen Universität München

**Ernst Mayr (München)**

Die Arbeitsgruppe Computeralgebra an der Technischen Universität München unter der Leitung von Prof. Dr. Ernst W. Mayr (Lehrgebiet Effiziente Algorithmen) besteht seit 1993. Mitarbeiter der Arbeitsgruppe sind bzw. waren Thomas Bayer, Anna Bernasconi, Ulla Koppenhagen, Klaus Kühnle, Michal Mnuk, Martin Raab, Sandeep Sadanandan und Peter Ullrich.

Die Arbeitsgruppe beschäftigt sich mit der Komplexität von Problemen und Algorithmen für Polynomideale, insbesondere von Gröbnerbasen, dem Algorithmenentwurf zur Berechnung von Gröbnerbasen sowie der Anwendung von Gröbnerbasen im Bereich des wissenschaftlichen Rechnens sowie der Kombinatorik. Neuerdings werden auch Effizienzuntersuchungen im Bereich Kryptographie (hyperelliptische Kurven) unternommen.

Einige spezifische Themengebiete sind

- explizite Beschreibung der Strata von Orbit-Räumen endlicher und kompakter Gruppenoperationen,
- Darstellung von Grapheneigenschaften durch Polynomideale mit möglicher Anwendung in der Graphentheorie,
- Komplexität und Algorithmen für Gröbnerbasen von binomialen, torischen und allgemeinen Polynomidealen,
- Anwendung von (klassischen) Gröbnerbasen zur Quantorenelimination in der Theorie der algebraisch/reell-abgeschlossenen Körper und
- Effizienz der Arithmetik auf Jakobi-Varietäten hyperelliptischer Kurven.

In der Lehre werden (Spezial-)Vorlesungen zu Kodierungsverfahren mittels hyperelliptischer Kurven und zur Algorithmischen Algebra (zweisemestrig) angeboten.

# Berichte von Konferenzen

1. **CHEP 2004**

   Interlaken, Schweiz, 27.09. – 30.09.2004

   `http://www.chep2004.org`

   Die CHEP 2004 Konferenz fand vom 27. bis 30. September 2004 im schweizerischen Interlaken statt. CHEP steht für „Computing in High Energy and Nuclear Physics“.

   Im Blickpunkt der Konferenz stand vor allem der Large Hadron Collider (LHC) und seine Anforderungen an die Datenverarbeitung. Hierbei ist insbesondere der Status des Grid-Computings und der Experiment Online- und Analysesoftware zu nennen. Aber auch neue Technologien im Bereich der Hardware waren ein Thema, da der LHC die effiziente Verteilung und Speicherung sehr großer Datenmengen erfordert. Eines der Ziele der Konferenz war es, aus den Erfahrungen von derzeit laufenden Experimenten für LHC zu lernen. Die Konferenz richtete sich entsprechend an Vertreter der Experimente zur Hochenergiephysik und den beteiligten Rechenzentren und bot ein gutes Forum zum Informationsaustausch zwischen IT und Experimenten. Auch auf den intensiven Kontakt zur Industrie wurde hoher Wert gelegt. Firmen hatten die Möglichkeit ihre neuesten Produkte und Entwicklungen an Ständen und in Vorträgen in die Diskussion einzubringen. Die Zahl der Teilnehmer aus Experimenten, IT und Industrie lag deutlich über 500. Neben täglichen Plenarvortraegen von allgemeinem Interesse gab es wegen der Vielzahl und der Vielfalt der Beiträge sechs parallele Sitzungen zu den Themen Online Computing, Event Processing, Core Software, Distributed Computing Services, Distributed Computing Systems and Experiences sowie Computer Fabrics.

   Die Kernergebnisse und Highlights wurden am letzten Tag in einer Plenarsitzung zusammengefasst. Die Kaffeepausen wurden durch Postersitzungen zu denselben Themenbereichen angereichert, was den Kontakt zwischen den einzelnen Gruppen weiter förderte.

   Alle Konferenzbeiträge und weitere Informationen sind über die Konferenzwebseite `http://chep2004.web.cern.ch/chep2004` abrufbar.

   Ulrich Schwickerath (Karlsruhe)

2. **ADG – Automated Deduction in Geometry**

   Gainesville, Florida, 16.09 – 18.09.2004

   `http://www.math.ufl.edu/~white/ADG2004.html`

   Der fünfte internationale Workshop *Automated Deduction in Geometry* konnte trotz der vielen Hurrikane im Sommer 2004 an der University of Florida stattfinden. Die Organisatoren waren Neil White, Hoon Hong und Manfred Minimair. Unser „Wetterfrosch“ Neil White hat in den Tagen vor dem Workshop die Satellitenbilder der Meteorologen intensiv verfolgt und alle Teilnehmer auf dem Laufenden gehalten. Trotz dieser „natürlichen“ Schwierigkeiten konnten fast alle Interessenten kommen, und es war ein sehr schöner Workshop.

Der ADG-Workshop findet jedes zweite Jahr statt und verbindet die Themen *Geometrie* und *Automatisches Beweisen*. Ileana Streinu und Doron Zeilberger wurden eingeladen und hielten sehr interessante Vorträge über *Folding Carpenter's Rules, Robot Arms, Proteins: a Combinatorial Approach* bzw. *Plane Geometry: An Elementary Textbook by Shalosh B. Ekhad 14 (circa 2050), Downloaded from the future by Doron Zeilberger*.

Die anderen Vorträge behandelten u. A. Themen aus den Gebieten Geometric Constraint Solving, Applications of Symbolic Computation und Dynamische Geometrie. Die Atmosphäre während des Workshops war sehr produktiv und es wurden viele interessante Diskussionen geführt. Zum Konferenzessen haben Neil White und seine Frau alle Teilnehmer zu sich nach Hause eingeladen.

Britta Broser (Berlin)

# Hinweise auf Konferenzen

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

Luxemburg, 28.03. – 01.04.2005

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

**Sektion Computeralgebra und Computeranalysis**

Die Gesellschaft für Angewandte Mathematik und Mechanik e.V. (GAMM) lädt Sie zur Teilnahme an der wissenschaftlichen Jahreskonferenz in Luxemburg vom 28. März bis zum 01. April 2005 ein.

**Sektionsleitung:**
Günter Mayer (Universität Rostock), H. Michael Möller (Universität Dortmund)

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

Passau, 03. – 08.04.2005

`http://www.a3l.org`

This conference takes place from Sunday April 3 to Friday April 8, 2005 at the University of Passau, Germany. There are two possible types of submissions: an extended abstract for inclusion into the conference proceedings or optionally, in addition to (but not instead of) the extended abstract, a full paper for inclusion into a special issue of the Journal of Symbolic Computation. For both sorts of submissions the authors are asked to make clear in the abstract and in the introduction what is the relation to Volker Weispfenning's work.

The extended abstracts are going to be reviewed for correctness and for fitting into the conference (relation Volker Weispfenning's work). For the possible additional journal submissions the rules of the Journal of Symbolic Computation fully apply. Submitters are expected to give 15 minute-presentations at the conference.

For question and comments, please write an email to `info@a3l.org`.

**Topics:**
Conference topics include any work related to former or present research areas of Volker Weispfenning, as for instance algebraic model theory and Gröbner bases.

**Important Dates:**
Nov 28, 2004: Deadline for both types of submissions
Jan 13, 2005: Notification of acceptance

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

Zeuthen, 03. – 08.04.2005

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

During the last years, computer algebra methods have been used widely throughout elementary particle physics. By now, applications of modern computer algebra are an essential and established calculational tool.

To initiate education and training of about 20 students and young researchers at graduate and Ph.D. level on these important aspects in the field of computer algebra, DESY organizes jointly with German universities a *School on Computer Algebra and Particle Physics*. It will be held at DESY Zeuthen and the focus will be on perturbation theory and the calculation of Feynman diagrams.

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

Zeuthen, 23.05. – 27.05.2005

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

The main purpose of the ACAT (formerly AIHENP) series of workshops is to gather physicists (experimentalists and theorists) and computer science oriented researchers to exchange ideas, to discuss standards and to promote new technologies related to “Computing intelligence” in physics research. The applications are targeted mainly to particle and nuclear physics, astrophysics and accelerator science.

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

   Porto Conte, Alghero, Sardinien, 26.05. – 02.06.2005

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

   MEGA is the acronym for Effective Methods in Algebraic Geometry (and its equivalent in Italian, French, Spanish, German, Russian, etc.), a series of roughly biennial conferences on computational and application aspects of Algebraic Geometry and related topics with very high standards.

   Proceedings containing a selection of the papers and invited talks presented at previous Mega conferences have been published by Birkhäuser in the series Progress in Mathematics (volumes no. 94, 109 and 143), by the Journal of Pure and Applied Algebra (volumes no. 117 and 118, 139 and 164) and by the Journal of Symbolic Computation (to appear).

   The conference will be held at Hotel Corte Rosada, a sea resort at Porto Conte (Alghero). Full board at the hotel will cost approximately € 370 for the seven nights from May 26 to June 2 for double occupancy, € 470 for single occupancy.

   Porto Conte is in a natural park 10 km from Alghero, a small walled town in the northwestern coast of Sardinia. The town has catalan origin, and catalan is spoken locally. Alghero has an international airport, served bylow-cost airlines, and connected to Milano, Roma, London, Hahn-Frankfurt and Girona-Barcelona. Another possible airport is Olbia.

   **Organization:**
   Carlo Traverso, Pisa (Conference Chair)


*(Foto: Seidl)*

   **Important Dates:**
   Deadline for submissions of talks: January 7, 2005
   Notification of acceptance of contributions: March 10, 2005
   Early registration deadline: March 29, 2005

6. **Tagung der Fachgruppe Computeralgebra**

   Kassel, 02. – 04.06.2005

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

   Diese Tagung der Fachgruppe Computeralgebra wurde bereits auf Seite 9 angekündigt.

7. **Joint meeting of AMS, DMV, ÖMG**

   Mainz, 16. – 19.06.2005

   `http://math-www.upb.de/~klausd/Mainz2005/`

   The second joint meeting of the three mathematical societies American Mathematical Society (AMS), Deutsche Mathematiker-Vereinigung (DMV) and Österreichische Mathematische Gesellschaft (MG) will take place from June 16 through June 19, 2005 at the Johannes Gutenberg University in Mainz, close to Frankfurt (Main) in Germany.

   We expect about 700 participants from the United States, Germany, Austria, and other countries, and this conference will be the biggest among all mathematics conferences in Germany in 2005. Six distinguished mathematicians are going to present their results in plenary lectures, and a broad spectrum of mathematical research will be covered in 30 Special Sessions.

   We hereby cordially invite all mathematicians and those who are interested in mathematics to participate in the conference. The conference starts on Thursday, June 16 at 1 p.m. and ends on Sunday, June 19 in the late afternoon.

8. **Maple Conference 2005**

   Waterloo, 17. – 21.07.2005

   `http://www.maplesoft.com/MC05/`

   The Maple Conference is an annual meeting of Maple enthusiasts and experts from around the globe and a wide range of disciplines. The high-impact program enables you to benefit from the experience and insights of your peers and leading figures in mathematics and computing.

   **Organization:**
   Ilias Kotsireas (Program Chair)

9. **International Workshop on Symbolic-Numeric Computation**

   Xi'an, China, 19. – 21.07.2005

   `http://www-calfor.lip6.fr/~wang/SNC2005`

   The growing demand of speed, accuracy, and reliability in scientific and engineering computing has been accelerating the merging of symbolic and numeric computations. To facilitate the symbolic-numeric interaction and integration, an international workshop on Symbolic-Numeric Computation (SNC 2005) will be held in Xi'an, China from July 19-21, 2005 as a satellite conference of ISSAC 2005.

   Potential participants of SNC 2005 are invited to submit an extended abstract of three or more pages or a full paper describing their work to be presented at the workshop. Electronic submissions are preferred, and should be sent to both of the PC co-chairs: Dongming Wang (`Dongming.Wang@lip6.fr`) or Lihong Zhi (`lzhi@mmrc.iss.ac.cn`)

   **Important Dates:**
   Deadline for extended abstract submission: April 10, 2005
   Notification of acceptance or rejection: May 15, 2005
   Deadline for full paper submission: October 10, 2005

   **Invited Speakers:**
   Robert M. Corless (University of Western Ontario, Canada), Matu-Tarow Noda (Ehime University, Japan), Victor Y. Pan

(City University of New York, USA), Hans J. Stetter (Technical University of Vienna, Austria).

10. **ISSAC 2005 – International Symposium on Symbolic and Algebraic Computation**

Peking, China, 24. – 27.07.2005

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

ISSAC is the yearly premier international symposium in Symbolic and Algebraic Computation that provides an opportunity to learn of new developments and to present original research results in all areas of symbolic mathematical computation. In 2005, ISSAC will take place at the Key Laboratory of Mathematics Mechanization, Chinese Academy of Science, Beijing, China, July 24-27.

**Organization:**
General Co-Chairs: Xiao-Shan Gao, George Labahn. SIGSAM Chair: Emil Volcheck

**Invited Speakers:**
Bruno Buchberger (RISC-Linz), Bruno Salvy (INRIA), Wen-Tsun Wu (Chinese Academy of Sciences)

**Important Dates:**
Deadline for Submissions: January 14, 2005
Notification of Acceptance: March 15, 2005
Camera-Ready Copy Received: April 15, 2005

11. **International Conference on Difference Equations, Special Functions and Applications**

München, 25. – 30.07.2005

`http://www-m6.ma.tum.de/~ruffing/\\`
`Conference2005`

Auf dieser Tagung ist eine Sektion zum Thema Computeralgebra geplant. Weitere Einzelheiten findet man auf der Homepage der Tagung.

12. **ACA 2005 – Conference on Applications of Computer Algebra**

Nara City, Japan, 01. – 03.08.2005

`http://www.jssac.org/Conference/ACA/`

The ACA 2005 conference will take place at Nara Women's University located in the central Nara city. This city is well-known for rich history as an ancient capital of Japan, and has a number of national important properties, shrines and temples with Japanese traditional style, including 8 properties of the World Heritage.

The ACA conference is devoted to promoting the applications and development of Computer Algebra and Symbolic Computation. Topics include Computer Algebra and Symbolic Computation in science, engineering, mathematics, computer science, education, and so on. A very peculiar feature of ACA conference is that all the technical sessions are organized by voluntary organizers.

**Organization:**
General Co-Chairs: Fujio Kako (Nara Women's Univ.), Matu-Tarow Noda (Ehime Univ.)

13. **CCCG 2005 – The 17th Canadian Conference on Computational Geometry**

University of Windsor, 10. – 12.08.2005

`http://cccg.cs.uwindsor.ca/`

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

Papers must be submitted in two steps: In the first step, the abstract of the paper must be electronically submitted. You will get a confirmation by email. The confirmation will include a login name and password, which you need in the second step. In this second step, the full paper must be electronically submitted before May 2, 2005. The login name and password received after the first step are required. Authors who cannot submit their papers electronically should inform the PC chair at `cccg@cs.uwindsor.ca`. There is also a third step, but only for those submitters whose paper is accepted. In that case, you will be asked to also submit the camera-ready copy electronically. Instructions will be sent after acceptation of your paper.

14. **CASC 2005 – The 8th International Workshop on Computer Algebra in Scientific Computing**

Kalamata, Greece, 12. – 16.09.2005

`http://www.cargo.wlu.ca/casc2005/`

The 8th International Workshop in Computer Algebra in Scientific Computing, CASC 2005, will be held in the city of Kalamata, Greece, in September 12-16, 2005. The methods of scientific computing play an important role in research and engineering applications in the natural and the engineering sciences. The significance and impact of computer algebra methods and computer algebra systems for scientific computing has increased considerably in recent times. Nowadays, such general-purpose computer algebra systems as Maple, Magma, Mathematica, MuPAD, Singular, CoCoA and others enable their users to solve the following three important tasks within a uniform framework: symbolic manipulation, numerical computation and visualization. The ongoing development of such systems, including their integration and adaptation to modern software environments, puts them to the forefront in scientific computing and enables the practical solution of many complex applied problems in the domains of natural sciences and engineering.

The topics addressed in the workshop cover all the basic areas of scientific computing as they benefit from the application of computer algebra methods and software. They include numerical simulation using computer algebra systems, symbolic-numeric methods for differential and differential-algebraic equations, algebraic methods in geometric modeling, algebraic methods for nonlinear polynomial equations and inequalities, symbolic and numerical computation in systems engineering and modeling and many more.

The workshop is intended to provide a forum for researchers and engineers in the fields of mathematics, computer science, numerical analysis, and industry, to interact and exchange ideas. An important goal of the workshop is to bring together all these specialists for the purpose of fostering progress on current questions and problems in advanced scientific computing.

**Organization:**
CASC General Chairs: V. P. Gerdt (Dubna), E. W. Mayr (Munich).
CASC 2005 General Chairs: I. Z. Emiris (Athens), I. S. Kotsireas (Waterloo), M. N. Vrahatis (Patras).

15. **DMV Jahrestagung 2005 und 16. Internationaler Mathematikerkongress der Österreichischen Mathematischen Gesellschaft**

Klagenfurt, 18. – 23.09.2005

`http://www.mathematik.uni-bielefeld.de/DMV/aktuell/aktuell_u.html`

Vom 18. bis zum 23. September 2005 findet an der Alpen-Adria-Universität Klagenfurt der 16. Internationale Mathematikerkongress der Österreichischen Mathematischen Gesellschaft und zugleich die Jahrestagung 2005 der Deutschen Mathematiker-Vereinigung statt.

16. **INFORMATIK2005 – 35. Jahrestagung der Gesellschaft für Informatik e.V. (GI)**

Bonn, 19. – 21.09.2005

`http://www.informatik2005.de`

Die alljährlich stattfindende Jahrestagung der Gesellschaft für Informatik präsentiert traditionell das breite Spektrum aktueller Entwicklungen in der Informatik. Angesprochen sind Fachleute aus Wissenschaft und Praxis, die sich einen fundierten Überblick über die wichtigsten aktuellen Trends in der Informatik verschaffen möchten. Die INFORMATIK 2005 steht unter dem Motto „Informatik LIVE!“. Sowohl in den Hauptvorträgen als auch in den Workshops soll dieses Motto unter verschiedenen Gesichtspunkten behandelt werden.

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

Haus Schönenberg bei Ellwangen, 20. – 22.04.2006

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

Diese Tagung der Fachgruppe Computeralgebra wurde bereits auf Seite 10 angekündigt.

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

Genua, 16. – 19.07.2006

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

## Kurze Mitteilungen

An dieser Stelle wollen wir künftig über Berufungen, Rufannahmen und universitäre Stellenausschreibungen hinweisen. Bitte reichen Sie entsprechende Informationen beim Redakteur des Rundbriefs ein.

**Prof. Dr. Joachim von zur Gathen** (Paderborn) hat einen Ruf auf eine C4-Professur (Chair of IT-Security) am Bonn-Aachen International Center for Information Technology angenommen.
(`http://www.bit.uni-bonn.de`)

**Prof. Dr. Gunter Malle** (Kassel) hat einen Ruf auf eine W3-Professur für Algebra an der Universität Kaiserslautern angenommen.
(`http://www.mathematik.uni-kl.de`)

**Dr. Preda Mihailescu** (Paderborn) hat einen Ruf auf eine W2-Professur für Computeralgebra / Kryptographie / Angewandte Zahlentheorie (Forschungsprofessur der Volkswagenstiftung) in Göttingen angenommen.
(`http://www.uni-math.gwdg.de`)

**Dr. Werner Bley** (Augsburg) hat einen Ruf auf eine W2-Professur für Computational Mathematics an der Universität Kassel erhalten.
(`http://www.mathematik.uni-kassel.de/compmath`)

Einige PostDoc-Stellen für validiertes Scientific Computing, Kryptologie und Intervallarithmetik sind an der Ecole Normal Supérieur in Lyon zu vergeben. Weitere Informationen auf der Webseite `http://www.ens-lyon.fr/LIP/Arenaire`.

# Lehrveranstaltungen zu Computeralgebra im SS 2005

- **Rheinisch–Westfälische Technische Hochschule Aachen**
  *Computeralgebra I,* W. Plesken, V2+Ü1
  *Begleitpraktikum Computermathematik I,* W. Plesken, V. Dietrich, G. Hartjen, P3
  *Begleitpraktikum Computermathematik II,* W. Plesken, V. Dietrich, G. Hartjen, P2
  *Algebraisches Praktikum*, G. Nebe, F. Lübeck, P2

- **Technische Universität Berlin**
  *Algebra I*, F. Heß, V4
  *Übungen zur Algebra I*, F. Heß, Ü2
  *Codierungstheorie*, M. Pohst, V4
  *Übungen zur Codierungstheorie*, M. Pohst, Ü2
  *Seminar Algorithmische Algebra und Zahlentheorie*, M. Pohst, S2

- **Fachhochschule Flensburg**
  *Analysis II mit Maple*, N. Pavlik, Ü2
  *Mathematical Modelling using Maple (V4)*, N. Pavlik, Ü2
  *Numerical Simulation using Maple (V4)*, N. Pavlik, Ü2
  *Numerical Software (V4) - Part I: Maple/MATLAB* , . . . , P. Thieler, Ü2
  *Applied Logics* “`with(logic);`“ *(V4)*, P. Thieler, Ü2
  *Mathematisches Seminar - Exploring SIAM Review Papers on Education with Maple*, P. Thieler, S2

- **Technische Universität Hamburg-Harburg**
  *Diskrete Mathematik Ib*, K.-H. Zimmermann, V2+Ü1
  *Algebraische Kombinatorik*, K.-H. Zimmermann, V2
  *Proseminar Mathematik*, P.Batra, S2

- **Universität Heidelberg**
  *Software-Praktikum Computeralgebra*, M. Dettweiler, P4

- **Universität Kaiserslautern**
  *Computeralgebra*, G. Pfister, V4
  *Praktikum Computeralgebra*, A. Frühbis-Krüger, P2
  *Praktikum Computeralgebra (Bachelor)*, A. Frühbis-Krüger, P2

- **Pädagogische Hochschule Karlsruhe**
  *Informatik II*, J. Ziegenbalg, V2
  *Modellbildung in Mathematik und Informatik*, J. Ziegenbalg, V2

- **Universität Kassel**
  *Einführung in Computeralgebrasysteme II (MuPAD)*, R. Schaper, V2
  *Kryptographie*, A. Klein, V4 + Ü2
  *Oberseminar Computational Mathematics*, W. Koepf, H.-G. Rück, S2

- **Universität Leipzig**
  *Geometrie mit dem Computer*, H.-G. Gräbe, V2

- **Universität Paderborn**
  *Introduction to Cryptography*, J. Blömer, V2 + Ü1
  *Computeralgebra*, P. Bürgisser, V4 + Ü2
  *Komplexitätstheorie*, P. Bürgisser, V2 + Ü1
  *Seminar: Computeralgebra, Kombinatorik und Komplexität*, P. Bürgisser, S2
  *Mathematik am Computer*, W. Oevel, V2 + Ü1
  *Projektstudium: Entwicklung und Implementierung eines Computeralgebrasystems*, W. Oevel, P4
  *MUPAD Seminar*, B. Fuchsteiner, W. Oevel, S2
  *PaSCo Oberseminar* , PaSCo (Paderborn Institute for Scientific Computation), S2
  *Kryptographie (Didaktik der Mathematik)* , C. Nelius, V3 + Ü1

## Aufnahmeantrag für Mitgliedschaft in der Fachgruppe Computeralgebra

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

Name: ______________________________ Vorname: ______________

Akademischer Grad/Titel: ______________________

**Privatadresse**

Straße/Postfach: ______________________

PLZ/Ort: ______________________________ Telefon: ______________

e-mail: ______________________________ Telefax: ______________

**Dienstanschrift**

Firma/Institution: ______________________

Straße/Postfach: ______________________

PLZ/Ort: ______________________________ Telefon: ______________

e-mail: ______________________________ Telefax: ______________

Gewünschte Postanschrift: [ ] Privatadresse [ ] Dienstanschrift

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

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

2. Der Jahresbeitrag beträgt €7,50 bzw. €9,00. Ich ordne mich folgender Beitragsklasse zu:

   [ ] **€7,50** für Mitglieder einer der drei Trägergesellschaften
   [ ] GI Mitgliedsnummer: ____________
   [ ] DMV Mitgliedsnummer: ____________
   [ ] GAMM Mitgliedsnummer: ____________
   Der Beitrag zur Fachgruppe Computeralgebra wird mit der Beitragsrechnung der Trägergesellschaft in Rechnung gestellt. (Bei Mitgliedschaft bei mehreren Trägergesellschaften wird dies von derjenigen durchgeführt, zu der Sie diesen Antrag schicken.) [ ] Ich habe dafür bereits eine Einzugsvollmacht erteilt. Diese wird hiermit für den Beitrag für die Fachgruppe Computeralgebra erweitert.

   [ ] **€7,50**. Ich bin aber noch nicht Mitglied einer der drei Trägergesellschaften. Deshalb beantrage ich gleichzeitig die Mitgliedschaft in der

   [ ] GI [ ] DMV [ ] GAMM.

   und bitte um Übersendung der entsprechenden Unterlagen.

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

   [ ] GI [ ] DMV [ ] GAMM.

3. Die in dieses Formular eingetragenen Angaben werden elektronisch gespeichert. Ich bin damit einverstanden, dass meine Postanschrift durch die Trägergesellschaften oder durch Dritte nach Weitergabe durch eine Trägergesellschaft wie folgt genutzt werden kann (ist nichts angekreuzt, so wird c. angenommen).

   [ ] a. Zusendungen aller Art mit Bezug zur Informatik, Mathematik bzw. Mechanik.
   [ ] b. Zusendungen durch wiss. Institutionen mit Bezug zur Informatik, Mathematik bzw. Mechanik.
   [ ] c. Nur Zusendungen interner Art von GI, DMV bzw. GAMM.

Ort, Datum: ____________________ Unterschrift: ______________________________

---

Bitte senden Sie dieses Formular an:

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

# Fachgruppenleitung Computeralgebra 2005-2008

**Sprecher:**

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


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

**Vertreter der GAMM,**
**Fachreferent Computational Engineering:**

Prof. Dr. Klaus Hackl
Ruhr-Universität Bochum
Universitätsstr. 150
44780 Bochum
0234-32-26025, -14154 (Fax)
`hackl@am.bi.rub.de`

**Fachreferentin Fachhochschulen:**

Prof. Dr. Elkedagmar Heinrich
Fachhochschule Konstanz
Fachbereich Informatik
Brauneggerstrasse 55
78462 Konstanz
07531-206-343, -559 (Fax)
`heinrich@fh-konstanz.de`

**Fachreferent Schule:**

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

**Fachexperte Chemie:**

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

**Vertreter der DMV:**

Prof. Dr. B. Heinrich Matzat
IWR, Univ. Heidelberg,
Im Neuenheimer Feld 368
69120 Heidelberg
06221-54-8242,-8318(Sekr.),-8850 (Fax)
`matzat@iwr.uni-heidelberg.de`

**Fachexperte Rundbrief:**

Dr. Markus Wessler
Kopernikusstr. 6
81679 München
089-69777336
`wessler@mathematik.uni-kassel.de`

**Stellvertretender Sprecher:**

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

**Vertreter der GI,**

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

**Fachexperte Physik:**

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

**Fachreferent Lehre und Didaktik:**

Prof. Dr. Hans-Wolfgang Henn
Universität Dortmund
Fachbereich Mathematik
Vogelpothsweg 87
44227 Dortmund
0231-755-2939, -2948 (Fax)
`wolfgang.henn@mathematik.uni-dortmund.de`
`http://www.wolfgang-henn.de`

**Fachreferent Internet/Math. Software:**

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


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

**Fachexperte Industrie:**

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