mathPAD

Band 10, Ausgabe 1, August 2001

In diesem Heft

Redaktion:
Benno Fuchssteiner,
Frank Postel, Andreas Sorgatz
Universität-GH Paderborn
D-33095 Paderborn
email: `benno@mupad.de`

ISSN 0941-9187
V.i.S.d.P.: Benno Fuchssteiner

Liebe Leserin, lieber Leser,

die vorliegende Sonderausgabe der mathPAD widmet sich ganz dem Thema *MuPAD* in der Lehre. Wir berichten über aktuelle Aktivitäten, Neuerscheinungen und Neuentwicklungen, Tipps und Tricks sowie über Anwendungen mit *MuPAD*.

Beim Durchblättern der mathPAD werden sie auch diesmal auf kommerzielle Anzeigen stoßen. Diese sind zur Finanzierung des Magazins notwendig, da die Universität-GH die Druckkosten für eine derart auflagenstarke mathPAD nicht mehr wie in der bisherigen Weise tragen kann und will. Lassen Sie sich dadurch aber nicht in Ihrem Lesevergnügen stören.

Wir glauben, auch mit diesem Heft wieder interessante Neuheiten über MuPAD und Berichte über Computeralgebra in der Lehre zusammengetragen zu haben. Für das Gelingen dieses Heftes möchten wir insbesondere den auswärtigen Autoren herzlich danken!

B.F. & F.P. & A.S.

Kurz notiert

◇ *MuPAD* 2.0 WIRD AUSGELIEFERT: `http://www.sciface.com/sales.shtml`. Deutsche und englische Demo-Versionen finden Sie unter: `http://www.sciface.com/download.shtml`. Die technischen Details beschreibt das Paper `http://www.mupad.de/DOC/changes.shtml`. Die Freigabe der französischen und japanische Versionen von *MuPAD* Pro für Windows wird für das Frühjahr 2002 erwartet.

◇ *MuPAD* IN LANDES- UND KREISBILDSTELLEN: *MuPAD* wird in Kürze auch in den Landes- und Kreisbildstellen verfügbar sein. Wir starten derzeit in Nordrhein-Westfalen durch. Sollte *MuPAD* bei Ihrer Bildstelle noch nicht verfügbar sein, so wenden Sie sich bitte an `schule@mupad.de`. Wir können Sie über den aktuellen Stand informieren und das Verfahren gegebenenfalls beschleunigen.

◇ *MuPAD* AN FRANZÖSISCHEN GYMNASIEN: Das französische Bildungsministerium plant *MuPAD* als einziges Computeralgebra-System an alle Gymnasien des Landes zu verteilen. Hintergrund zu diesem Plan ist die Tatsache, dass *MuPAD* in Frankreich bereits seit 1999 als eines von zwei Computeralgebra-Systemen für das *Examen de Agrégation* (landesweite Prüfung für die Befähigung zum höheren Lehramt) vorgeschrieben ist. Siehe hierzu auch `http://www.math.u-psud.fr/~agreg/Agreg/installation.html`.

◇ *MuPAD* ALS EINZIGES CA-SYSTEM FÜR BLINDE GEEIGNET: Das Educational Endeavor "Computer Science for the Blind" des "Department for Computer Science" der Universität Linz in Österreich bescheinigt in einer Studie, dass *MuPAD* das einzige für Blinde geeignete Computeralgebra-System ist. Ermöglicht wird dies insbesondere durch sein Microsoft-Word^{TM}-ähnliches und Braille-Schrift-taugliches Notebook-Konzept sowie durch die Bereitstellung nahezu der gesamten *MuPAD*-Dokumentation in Form von HTML-Dokumenten: `http://www.mupad.de/DOC/`. Die weitere Zusammenarbeit beider Teams in diesem Bereich ist geplant. Weitere Informationen erhalten Sie auf Anfrage an `info@sciface.com`.

◇ SCILAB IN *MuPAD* INTEGRIERT: Die Integration des Mathlab^{TM}-Clones Scilab (hocheffiziente numerische Software) in *MuPAD* unter Solaris, Linux und Windows wurde auf der SMAI 2001 in Argnac-Pompadour, Frankreich erstmals einem größerem Publikum aus den Bereichen Numerik und Angewandte Mathematik in Industrie und Forschung vorgestellt. Die Freigabe des *MuPAD*-Scilab Power-Paketes für kombiniertes symbolisch-algebraisches und numerisches Rechnen wird für das kommende Frühjahr erwartet. Weitere Informationen erhalten Sie auf Anfrage an `info@sciface.com`.

◇ MUPAD ALS MATH-ENGINE IN SCIENTIFIC WORKPLACE: Die Produkte Scientific Notebook, Scientific Workplace und Scientific Word von MacKichan Software Inc., Seattle, USA sind ab Version 3.5 nun mit *MuPAD* als Rechenkomponente verfügbar. Weitere Informationen unter `http://www.mackichan.com`.

◇ *MuPAD* STARTET WELTWEITEN VERTRIEB: Über den Hauptdistributor MacKichan Software Inc., Seattle, USA und dessen Vertriebsnetz läuft in diesen Wochen der weltweite Vertrieb von *MuPAD* Pro durch landesspezifische Distributoren mit entsprechendem Support an. Neben der aktuellen deutschen und englischen Version stehen für den französischen und japanischen Markt die landessprachlichen Benutzungsoberflächen voraussichtlich im Frühjahr 2002 zur Verfügung. Weitere Informationen unter `http://www.mackichan.com`.

NaT–Working Mathematik NRW: Schüler, Lehrer und Wissenschaftler vernetzen sich

Wolfgang Lindner
Gerhard-Mercator-Universität Duisburg, lindner@math.uni-duisburg.de

Schüler, Lehrer und Wissenschaftler vernetzen sich in Naturwissenschaft und Technik. Das Zusammenführen der Interessengruppen am "runden Tisch" hat die Erarbeitung von Konzepten und Materialien für den computerunterstützten Unterricht in der Sekundarstufe II zum Ziel. NAT–WORKING MATHEMATIK NRW *ist ein gefördertes Projekt der* ROBERT-BOSCH-STIFTUNG.

Das Projekt NAT–WORKING MATHEMATIK NRW ist im Frühjahr 2001 gestartet und ist zunächst für einen Zeitraum von 3 Jahren geplant. Aktuell umfasst es die folgenden Einzelprojekte:

- Herausforderung Anwendungen - mathematische Praktika in Industrie und Wirtschaft
- Treffpunkt Mathematikunterricht - Veränderung der Kultur des Mathematikunterrichts
- Mathematiklabor - CAS-Seminar zur experimentellen Mathematik
- Begegnungen mit Mathematik - Mathematiktage
- Computergestützte Visualisierungen von Mathematik - eine vernachlässigte Dimension von Mathematik
- Lernen von Mathematik - Mathematik im Sokratischen Dialog

Weitere Informationen bekommen Sie unter `http://www.uni-duisburg.de/FB11/PROJECTS/NATW/`. Zur aktiven Teilnahme sind alle interessierten Personen herzlich eingeladen.

Mathematiklabor – CAS-Seminar zur experimentellen Mathematik

Diese Veranstaltungsreihe für Lehrerinnen und Lehrer möchte den Einsatz neuer Technologien zur Gestaltung offener und produktiver Lernumgebungen aufzeigen, die ein eigentätiges und weitgehend selbstständiges Arbeiten der Schüler ermöglichen sollen. Nach drei Auftaktveranstaltungen werden 2-tägige Workshops zur eigenen Gestaltung bzw. Erprobung solcher Lernumgebungen in Zusammenarbeit mit Schülern und Dozenten angeboten.

Grundgedanke und Ziel dieser CAS-Workshops ist einerseits die Einführung in Computeralgebra-Systeme und andererseits die Gestaltung zugehöriger offener Lernumgebungen und Problemstellungen in einem rechnergestützten Unterricht unter Berücksichtigung von forschungsrelevanten fachdidaktischen Fragestellungen. Zielgruppe sind Schülerinnen und Schüler der Sekundarstufe II aller weiterführenden Schulformen sowie interessierte Lehrerinnen und Lehrer.

Im Rahmen dieser Veranstaltungsreihe wird u.a. das in Paderborn entwickelte Computeralgebra-System *MuPAD* eingesetzt. Über die Inhalte einer Einführungsveranstaltung informiert der folgende Artikel *Digraphen als Einstieg in die Lineare Algebra* in dieser Ausgabe der *mathPAD*.

Digraphen als Einstieg in die Lineare Algebra

Wolfgang Lindner
Gerhard-Mercator-Universität Duisburg, lindner@math.uni-duisburg.de

Im Rahmen des Projektes NAT–WORKING MATHEMATIK NRW *wurde in der Veranstaltungsreihe Mathematiklabor – CAS-Seminar zur experimentellen Mathematik"das in Paderborn entwickelte Computeralgebra-System* MuPAD *für einen Einstieg in die Lineare Algebra / Geometrie in Grundkursen der gymnasialen Oberstufe eingesetzt.*

In einer Längsschnittstudie an der Universität Duisburg wird der Einfluß von Computeralgebra-Systemen wie z.B. *MuPAD* auf die mathematischen Weltbilder (sog. *belief systems*) von Grundkursschülern[1] und die zugehörige Genese fundamentaler Konzepte der elementaren Linearen Algebra / Geometrie untersucht. Dazu wird ein kompletter CAS-orientierter Lehrgang zur Linearen Algebra für *MuPAD* entwickelt und erprobt. Insbesondere wird dabei beobachtet, welchen Einfluß eine CAS-gestützte Lernumgebung als didaktisches Tool auf die unterrichtlichen Argumentationsmuster, das Explorations- bzw. Problemlöseverhalten und die nachhaltige Entwicklung von Fertigkeiten der Grundkursschüler hat.

Über Rahmenbedingungen, currikulare Designentscheidungen etc. des Versuchs wird an anderer Stelle berichtet.

Bemerkungen zum methodisch-didaktischen Design

Die *MuPAD*-Lernumgebungen sind als gemäßigt konstruktivistische Unterrichtsentwürfe angelegt: beziehungsreiche Sinnkontexte oder zielgerichtete Konfrontationen mit Lernbarrieren als Lernanlässe sind wichtige Designkomponenten. Ein nachhaltiger Wissenserwerb soll zudem durch multiple Repräsentationsformen (z.B. numerisch, algebraisch-formal, graphisch-visuell, deskriptiv sowie CAS-algorithmisch) der einzuführenden Konzepte unterstützt werden, die das individuelle Wissensnetz des Lernenden aus fundamentalen Ideen, zentralen Mathematisierungsmustern und lokalen Lösungsstrategien zu konstruieren helfen. Einsicht in Zusammenhänge soll der Schüler auch aus informalen Argumentationen und Präsentationen gewinnen, die auf CAS-gestützten quasi-experimentellen Untersuchungen und Explorationen von Problemstellungen basieren und von metakognitiven Reflexionen des Lernprozesses begleitet werden.

Überdies versucht dieser Entwurf der Linearen Algebra ED DUBINSKY's [2] Vorschlag zur Implementierung eines didaktischen `A.C.E.`-Zyklus umzusetzen: kooperative *A*ktivitiäten der Schüler in einer experimentellen CAS-Lernumgebung werden gefolgt von (*C*)Klassendiskussion/Präsentation und traditionellen Übungsaufgaben (*E*xercises); diese sind wiederum oft auch als Selbsttätigkeitsphasen[2] angelegt. Gelegentlich modifizieren wir diesen Ansatz zu einem C.A.E.-Zyklus oder zu BUCHBERGER's erkenntnistheoretischer Helix ("Kreativitätsspirale"), vgl. [3].

Die mathematische Konzeption dieser Linearen Algebra / Geometrie orientiert sich an den Vorgaben des NRW-Richtlinien sowie an der schon von Artmann/Törner [1] empfohlenen Anwendungsorientierung unter Benutzung

[1] im Folgenden als geschlechtsneutrale Kurzform für Schüler und Schülerinnen
[2] vgl. auch das vom LSW in Soest initiierte BLK-Projekt SelMa, `http://www.learn-line.nrw.de/angebote/selma/`

von Matrizen als universalem mathematischen Konzept: als Themen werden Matrizenalgebra, lineare Gleichungssysteme (LGS), der GAUSS-Algorithmus[3] und die Konstruktion der inversen Matrix, unterbestimmte LGS sowie (die Richtlinen ergänzend) überbestimmte LGS mit Pseudoinversen behandelt.

Jedem Thema sind dabei typische Referenzkontexte mit mindestens einer prototypischen Anwendung zugeordnet:

mathematisches Konzept	*Referenzkontext; prototypische Anwendung*
Matrizen	DiGraphen; Fluglinienoptimierung
Gleichungssysteme	Mischungsproblem; IO-Analyse
GAUSS-Algorithmus; A^{-1}	LEONTIEF-Modell
unterbestimmte LGS	Orthogonalität; Projektionen; Computergrafik
überbestimmte LGS; Pseudoinverse	Ausgleichrechnung; Regressionsgeraden

Im Folgenden skizziere ich als Beispiel einen Ausschnitt aus dem ersten Unterrichtsabschnitt.

Die Digraph-CAS-Lernumgebung

Die Digraph-Lernumgebung ist in diesem Entwurf eine von vier zentralen zeitlich getrennten CAS-Lernumgebungen der elementaren Linearen Algebra von jeweils einigen Stunden Dauer, innerhalb derer die Schüler diejenigen Routinen ausbilden sollen, die für einen hinreichend kompetenten Umgang mit dem CAS erforderlich sind. In der Digraph-Lernumgebung fokussieren wir zunächst die statische Grundvorstellung einer Matrix als Objekt (Tabelle): als Referenzkontext (Modellvorstellung) dient eine Fernstraßentabelle von Sizilien, die als gerichteter Graph (kurz: Digraph[4]) repräsentiert wird. Digraphen, zugehörige quadratische Adjazenzmatrizen und Digraphoperationen abstrahieren diesen Referenzkontext zu einem symbolischen Netzplan" untereinander verbundener Standorte $A, B, \ldots$ mit verschiedenen Interpretationsmöglichkeiten innerhalb dieses Modells z.B. als Bus- oder Bahnnetz:

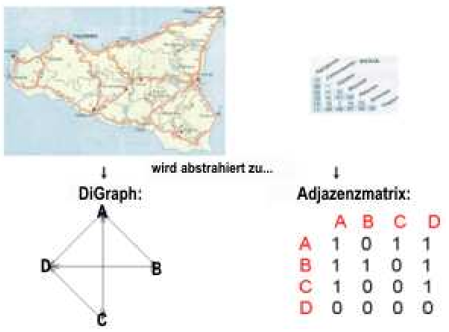


Abbildung 1: Digraph und assoziierte Adjazenzmatrix M mit 1/0 Einträgen zur Repräsention der Relation "..verbunden/nicht verbunden mit..". Zum Beispiel wird der Eintrag $M_{13} = 1$ interpretiert als 'A ist verbunden mit C' und der Eintrag $M_{32} = 0$ wird interpretiert als 'C ist nicht verbundnen mit B'.

Innerhalb dieses Modells studieren die Schüler die algebraischen Grundoperationen von Matrizen wie Addition, Subtraktion und Produkt von Matrizen. Zugleich werden diese Adjazenzmatrizen bzw. die Ergebnisse von Rechenoperationen mit und ohne *MuPAD* visualisiert: dazu wurde ein Tool *diGraph*[5] entwickelt, das von den Schülern als Blackbox zur Anfertigung von Digraph-Zeichnungen mit Hilfe von *MuPAD* verwendet werden kann. Zusätzlich werden die Resultate von Rechenoperationen in diesem Kontext interpretierbar.

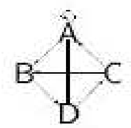
[3] Dieser Algorithmus wird in einer speziellen Weise mit *MuPAD* visualisiert und auch in Form eines CAS-Spiels präsentiert.

[4] von engl. **di**rected **graph**

[5] Eine erste Version dieses Entwurfs der Linearen Algebra/ Geometrie wurde vom Autor bei seiner Teilnahme am Schulversuch *Erprobung eines CAS im Mathematikunterricht der gymnasialen Oberstufe* des LSW/Soest 1997-1999 unter Nutzung von DERIVE entwickelt. M. IGEL vom *MuPAD* -Team danke ich für die Unterstützung bei der Portierung nach *MuPAD*.

Die Wegematrix einer Adjazenzmatrix

Besonders interessant ist dabei die Deutung der Potenz $M^n := M * \cdots * M$ der Adjazenzmatrix M eines Digraphen: der Exponent n beschreibt die Länge (Anzahl der Kanten) eines Weges im Digraphen und die nicht-auf-1-reduzierten "gewichteten" Matrixeinträge von M^n zählen die Anzahl der verschiedenen möglichen Wege, die zwei Positionen verbinden:



$$\begin{bmatrix} 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} = M$$

Abbildung 2: Wie viele Wege beliebiger Länge führen von B nach A? Der Wegematrix-Eintrag $wege(M)_{21} = 2$ gibt die Antwort. Diese zwei prognostizierten Wege sind im Digraph sichtbar: $B > D > C > A$ and $B > D > C > A > A$ (man beachte die Schlinge!)

Das hieraus destillierte Konzept einer "Wegematrix"(n = Zeilenzahl von M) ist eine sinnhaltige Operation, die aber nur umständlich und zeitaufwändig per Hand berechenbar ist, da zahlreiche Additionen und Multiplikationen von Matrizen auftreten. Hier erleben die Schüler unmittelbar den Nutzwert eines CAS wie *MuPAD*, denn die Wegematrix ist als eine einfache *MuPAD*-Funktion realisierbar und kann anschließend am Modell durch die Schüler getestet werden:

```
wege := (A) -> _plus( A^i  $  i=1..linalg::nrows(A)):

wege( matrix([ [1,1,0,1],[0,0,0,1],[1,1,0,0],[0,0,1,0] ]) );
```

Im Gegensatz zu einer traditionellen CAS-freien mathematik-theoretischen" Betrachtung gewinnen wir hier ein konstruktives lauffähiges" Konzept einer Wegematrix.

Abschließend wird diese Wegematrix-Funktion in einem realitätsnahen Projekt bei einem Optmierungsproblem für ein kleines Flugverbindungsnetz von 7 Stationen genutzt: eine CAS-freie Behandlung einer solchen Fragestellung ist kaum mehr möglich bzw. sinnvoll.

Die Implementierung von Operationen auf Adjazenzmatrizen

In einem *freiwilligen* fachübergreifenden Projekt können von den Schülern in einer Phase der Selbsttätigkeit zusätzlich *MuPAD*-Routinen zur Addition, Subtraktion und Multiplikation von Adjazenzmatrizen erarbeitet werden. Diese Operationen auf speziellen Adjazenzmatrizen sind im CAS als interne Funktionen nicht vordefiniert und bieten daher einen willkommenen Anlass zur Erstellung einer kleinen Toolbox in *MuPAD*.

Das Problem besteht darin, das Ergebnis einer normalen Matrixoperation auf 0 / 1 zu reduzieren, um wieder eine Adjazenzmatrix zu erhalten. Beispielsweise werden zwei Adjazenzmatrizen X und Y addiert, indem sie wie üblich addiert werden und anschließend jedes von Null verschiedene Element gleich 1 gesetzt wird. Dadurch wird zum Beispiel das Anwachsen der Verbindungen beim Zusammenlegen zweier Verkehrsbetriebe modelliert. Die Subtraktion zweier Adjazenzmatrizen wiederum erzeugt einen Digraph, in dem alle gemeinsamen Verbindungen von X und Y eliminiert werden.

Bei der Implementierung solcher Operationen auf Adjazenzmatrizen zeigt sich, dass die *MuPAD*-eigene Programmiersprache überaus flexibel und reichhaltig ist, um auch unterschiedliche Entwurfsvorstellungen zu verwirklichen:

- hier ist beispielsweise eine prozedural programmierte Version der Addition zweier Adjazenzmatrizen:

```
eins:=(x)-> (if x=0 then 0 else 1 end_if); // mod 2 wird vermieden

// etwas 'syntaktischer Zucker' ..
alias( zeilenZahl  = linalg::nrows ):
alias( spaltenZahl = linalg::ncols ):

//.. erleichtert die Lesbarkeit:
meins := A -> (((A[i,j] := eins(A[i,j]))
                 $ i = 1..zeilenZahl(A))
                 $ j = 1..spaltenZahl(A);
                return(A));

DADD1 := (A,B) -> meins(A+B);
```

- und hier eine funktional programmierte:

```
DADD2 := (A,B) -> zip(A,B,eins@_plus);

A:=matrix([[1,1],[1,0]]);
B:=matrix([[0,1],[1,1]]);

DADD1( A,B );
DADD2( A,B );
```

Wir erwähnen, dass der Implementationsprozess der Digraph(matrix)-Operationen in *MuPAD* in gewissem Sinne vergleichbar ist mit PLOYA's 4-phasigen Problemlösungszyklus, ehe er zu semi- bzw. vollautomatisierten Berechnungen führt. Wir nutzen im Unterricht solche vereinfachten semiautomatischen *MuPAD*-Funktionen, damit wir uns auf den entscheidenden Schritt im Algorithmus fokussieren konnen, ohne uns in administrativen programmierspezifischen technischen Detais zu verlieren. Damit versuchen wir, den formalen Aspekt von Mathematik bzw. Informatik auf das notwendige Maß zu reduzieren und Einfachheit und Klarheit der Entwurfsidee für die Schüler zu erhalten. Im Sinne von H. MÖLLER [4] werden dabei rein mathematische Existenzüberlegungen durch algorithmische Konstruktionen ersetzt und dadurch prozedurale Denkprozesse gefördert.

Bemerkung: *MuPAD* wird von mir zum Beispiel auch im Differenzierungskurs *Mathematik & Informatik* in Klassenstufe 10 zur Behandlung elementarer Programmierprobleme eingesetzt. Die so vorgebildeten Schüler können nun ihre Vorkenntnissse nutzbringend einsetzen.

Öffnung für die vertikale Vernetzung Schüler – Lehrer – Forscher

Das gesamte Lernmaterial wird in Form von kapitelweise aufgeteilten DVI-Files präsentiert; deren Inhalte sind mit dem *MuPAD*-eigenen HyperTeX-Viewer von den Schülern sowohl im Unterricht als auch zu Hause studierbar[6] Die angebotenen Problemstellungen werden von den Schülerteams kooperativ bearbeitet bzw. individuell ergänzt und in *MuPAD*-Notebooks oder RTF-Dateien präsentiert.

[6] Übrigens liegen die Aufgaben aus der am Staatsinstitut für Schulpädagogik und Bildungsforschung (ISB), München entwickelten Aufgabenbank SMART ebenfalls im DVI-Format vor und können daher mit dem *MuPAD*-Viewer betrachtet und innerhalb der *MuPAD*-Umgebung direkt benutzt werden. Momentan sind die Jahrgangsstufen 5, 6, 8, 9, 10 und 11 mit insgesamt ca. 1900 Aufgaben vertreten: ein überaus reichhaltiger Pool für *MuPAD*-Nutzer. `http://btmdx1.mat.uni-bayreuth.de/smart/intro.htm`.

Dieses Online-Konzept ermöglicht effektive Optimierungszyklen der Entwurfsvorlage und die Öffnung dieser Grundkonzeption unter anderem für den kollegialen Austausch prototypischer Referenzkontexte, d.h. die Wahl oder Erprobung anderer Leitprobleme oder abgeänderte optimierte Aufgabensequenzen für den nächsten Kurs-Durchlauf.

Das kooperativ-kollegiale Design weiterer *MuPAD*-Lernumgebungen soll auf der Grundlage der vorhandenen Materialien in NAT-WORKING Wochenend-Workshops erfolgen. Die Teilnahme ist kostenlos und wird (einschließlich der Übernachtungs- und Verpflegungskosten) von der fördernden ROBERT-BOSCH-STIFTUNG getragen. Die Reflexion von Erfahrungen in der Unterrichtserprobung wird angestrebt; ein Gedankenaustausch via eMail unter den Schülern findet bereits statt.

Interessierte Kolleg(inn)en sind zur Teilnahme herzlich eingeladen und können sich über die unten angegebene Kontaktadresse näher informieren.

Literatur

[1] ARTMANN, B. & TÖRNER, G.(1980): *Lineare Algebra. Grund- und Leistungskurs.* Göttingen: Vandenhoeck & Ruprecht.

[2] DUBINSKY, E. (1991): Reflective Abstraction in Advanced Mathematical Thinking. In D. TALL (Ed.), *Advanced mathematical thinking.*, 95–123. Dordrecht: Kluwer Academic Publishers.

[3] HEUGL,H. et al. (1996): *Mathematikunterricht mit Computeralgebra-Systemen.* Bonn: Addison-Wesley.

[4] MÖLLER, H. (1997): Algorithmische Lineare Algebra. Braunschweig: Vieweg.

Weitere Informationen

Weitere Informationen erhalten Sie unter `http://www.uni-duisburg.de/FB11/PROJECTS/NATW/`.

OStR Wolfgang Lindner
Gerhard-Mercator-Universität Duisburg
Fachbereich 11 - Mathematik
Lotharstr. 65
D-47057 Duisburg
Email: lindner@math.uni-duisburg.de
Fax: ++49 (0)203 - 379-2528
Tel: ++49 (0)203 - 379-1326

Linda – Laptops in netzbasierten Daten-Anbindungen

Wolfgang Kramer, Andreas Sorgatz
Felix-Fechenbach-Berufskolleg, Wolfgang.Kramer@ffb.lippe.de
SciFace Software, sorgatz@sciface.com

"Die Benutzung des Computers in der Schule wird in kurzer Zeit so selbstverständlich sein wie die Benutzung der Tafel oder des Taschenrechners."Bundespräsident Herzog, 9.6.1999, "Erziehung im Informationszeitalter", Rede zur Eröffnung des Paderborner Forums. Am Felix-Fechenbach-Berufskolleg wird mit Hochdruck daran gearbeitet.

Die Ausgangssituation

Am Felix-Fechenbach-Berufskolleg wurden in den vergangenen Jahren viele Mittel in die Ausstattung mit modernen Computerräumen und die Vernetzung derselben untereinander investiert. Es ist ein modernes Lernnetz mit Internet-connection gewachsen, in dem die Schüler von vielen verschiedenen Computerarbeitsplätzen aus auf den eigenen sowie auf einen zentralen Datenbestand zugreifen können. Nachteilig bei der derzeitigen Konstellation ist, dass nicht für jeden Schüler einer Lerngruppe ein eigener Rechner bereit steht und daher zwei oder ggf. drei Schüler während der Arbeit in einem PC-Raum an einem Gerät arbeiten müssen. Insbesondere bei den neuen IT-Berufen und den Informationstechnischen Assistenten stehen die Nutzung der DV-Systeme und die Arbeit in vernetzten Systemen von Intranet bis Internet im Mittelpunkt der täglichen Arbeit. Für diese Berufe soll ein modernes Konzept für die Ausstattung mit Hardware und Software aufgezeigt werden.

Rechner in Schülerhand

"In Deutschland sollte jedem Schüler ein Notebook zur Verfügung gestellt werden." Prof. Dr. Klaus Haefner, Bremen, Anhörung des Schulausschusses NW zum Thema Medienbildung in den Schulen des Landes Nordrhein-Westfalen" 01.06.1999 (Landtag intern 08.06.1999)

Rechner in Schülerhand, d.h. im Eigentum der Schüler, werden pfleglicher behandelt als geliehenes Material. Eine Finanzierung komplett durch den Staat würde nach Schätzungen ca. 8 MRD DM kosten. Dies ist zum einen nicht möglich, zum anderen entfallen aber auch die Anreize, die sich aus dem Eigentum am Rechner ergeben; ganz abgesehen von den Fragen der Nachinvestitionen für die Folgejahrgänge. Eine kontinuierliche Aktualisierung des Rechnerbestandes kann nur gekoppelt sein mit der Forderung nach Eigentum an diesen Rechnern.

Viele Schüler haben bereits heute Zugriff auf einen Rechner. Sie arbeiten dabei in der Schule und im privaten Bereich an unterschiedlich Systemen und transportieren ihre Daten mit Hilfe von Disketten von einem zum anderen Rechner, was eine effektive Arbeit durch Dateninkonsistenzen behindert. Abhilfe schaffen hier eigene Rechner, eingebunden in vernetzte Systeme.

Das Projekt

"Information muss jedermann zu jederzeit an jedem Ort verfügbar sein." Alexander Graham Bell, 1847 -1922) Erfinder des Telefons.

Eine Vision wird Wirklichkeit, Lernen erfährt durch die Nutzung von Netzwerk und Laptop eine neue Dimension, wird unabhängig von Raum und Zeit. Der individuelle Computer wird zum zentralen universalen Arbeitsmittel. Mit ihm wird nicht nur im Unterricht gearbeitet, es werden Hausaufgaben und Gruppenarbeiten erledigt, Klassenarbeiten geschrieben und Prüfungen abgelegt.

Am Felix-Fechenbach-Berufskolleg wird das Konzept der Laptops in Schülerhand im Bildungsgang "Berufsfachschule für Informationstechnische Assistenten" BFI erprobt. Wesentliche Aspekte der Erprobung sind:

- die Nutzung der Laptops in allen Unterrichtsfächern
- die Dokumentation des Unterrichts
- die Dokumentation der Hausaufgaben
- das Recherchieren im Internet
- das Präsentieren von Ergebnissen, Referaten,...
- das Kommunizieren

Das Modellprojekt `linda@ffb.lippe.de` wird in der Internet-Präsentation des Felix-Fechenbach-Berufskolleg fortlaufend dokumentiert und soll auch auf dem Nordrhein-Westfälischen Bildungsserver *learn-line* veröffentlicht werden.

Derzeitig sind folgende Projektpartner und Sponsoren von Hardware und Software-Ausstattung an dem Modellprojekt beteiligt:

- e-initiative.nrw
- co. Tec, Rosenheim
- SciFace Software GmbH & Co. KG, Paderborn
- Unirez GmbH, Detmold
- Volksbank, Detmold

Zur zukünftigen Fortführung und Ausweitung des Projektes werden noch weitere Sponsoren gesucht. Bitte wenden Sie sich dazu an die Projektleitung.

Mathematisch-technische Anwendungen

Am 15. Mai 2001 stieg *MuPAD* in dieses Projekt ein. Eine Lehrerfortbildung sowie freie *MuPAD*-Lizenzen für Schule, Lehrer und alle beteiligten Schüler brachten diesen neuen Projektabschnitt auf den Weg. Einsatz findet *MuPAD* in den mathematisch-technischen Unterrrichtsfächern zum symbolisch-numerischen Rechnen sowie zur Visualisierung für einen explorativen Zugang zur Mathematik.

LIPPE aktuell — Detmold

SciFace sponsort Algebra-Programm für dem Mathe-Unterricht am Felix-Fechenbach-Berufskolleg

Lust statt Frust: Rechenarbeit erledigt der Computer

Lippe-Detmold (ts). Die meisten Menschen verbinden Mathematik vor allem mit frustrierenden Erlenbnissen aus ihrer Schulzeit. Dabei kann die »Mutter aller Naturwissenschaften« sogar Laien in ihren Bann ziehen, wenn intelligente Systeme die mühselige Rechenarbeit erledigen.
Eines der besten Computerprogramme für algebraische Berechnungen wurde im benachbarten Paderborn entwickelt. MuPAD heisst die Software, die als einzige europäische Entwicklung zu den weltweit führenden Spitzensystemen gehört. Programmiert wurde das Algebra-System von Mitarbeiter der noch jungen Firma SciFace, die aus einem Forschungsprojekt der Paderborner Universität hervorgegangen ist und noch heute enge Kontakte zur Hochschule pflegt.
Die vielfach ausgezeichnete Software, die sich nicht nur für Lern- und Demonstrationszwecke von Lehrern und Schülern eingesetzt werden kann, sondern auch den Anforderungen anspruchsvoller Nutzer wie Ingenieuren und Wissenschaftlern gewachsen ist, wird jetzt auch am Felix-Fechenbach-Berufskolleg eingesetzt. Möglich wurde das durch Sponsoring der Herstellerfirma, die eine komplette Schullizenz der Windows-Vollversion von MuPAD für alle Lehrer und Schüler zur Verfügung stellte, sogar für die Unterrichtsvorbereitung zu Hause dürfen die Schüler das Programm benutzen. Vorher müssen aber noch deren Lehrer die Schulbank drücken, um sich mit der Bedienung der Software vertraut zu machen.
Aber nicht nur Pädagogen und Schüler, auch die Software-Schmiede profitiert von dem Projekt. »Wir werten die Unterrichtsmaterialien und Projektarbeiten aus, um mit diesen Erkenntnissen MuPAD noch weiter für den Einsatz an Schulen zu optimieren«, sagte Dr. Andreas Sorgatz, zuständig für externe Projekte und Entwicklungen bei SciFace. Er ist sich ganz sicher, dass Algebra-Programme in Zukunft verstärkt im Mathe-Unterricht eingesetzt werden. »Die Software nimmt dem Nutzer das rein mechanische Rechnen ab und visualisiert mathematische Sachverhalte. So bleibt mehr Zeit für das Wesentliche: Modellbildung und Problemlösungsstrategien.«
Auch Lehrer Wolfgang Kramer, auf dessen Initiative die Zusammenarbeit mit SciFace zurückgeht, ist sich ganz sicher, dass MuPAD den Mathematikunterricht nicht revolutionieren sondern vereinfachen wird. »Der Einsatz von Computern im Unterricht wird bald eine Selbstverständlichkeit sein«, sagte der Pädagoge, der auch das »linda«-Projekt angestoßen hat. Dahinter verbirgt sich der Versuch, möglichst alle Schüler mit eigenen Laptops auszustatten, die im Unterricht und für Hausarbeiten genutzt werden können.

Andreas Dieckmann und David Oesterhaus (sitzend v. links) haben schon einen eigenen Laptop, an dem sie mit Hilfe von MuPAD ihre Matheaufgaben bearbeiten. Lehrer Wolfgang Kramer, Schulleiter Heinrich Lindau und Landrat Friedel Heuwinkel (v. links) sind von den Fähigkeiten der Algebra-Software beeindruckt, die Dr. Andreas Sorgartz demonstriert. Foto: Schwarzmann

Lippe aktuell 2001-05-19

LIPPISCHE LAND

Rechnen per Mausklick: *Sehr angetan von MuPAD sind David Oesterhaus (vorne) und Andreas Diekmann, die sich von Dr. Andreas Sorgatz Feinheiten der Software erklären lassen. Nicht minder fasziniert zeigen sich Mathe-Lehrer Wolfgang Kramer, Schulleiter Heinrich Lindau und Landrat Friedel Heuwinkel (hinten, von links).* FOTO: FLAMME-BRÜNE

So macht Mathe Spaß

Fechenbach-Berufskolleg rechnet mit „MuPAD“

■ Detmold (fla). David Oesterhaus und Andreas Diekmann mögen Mathe. Seit kurzem haben die beiden Schüler des Felix-Fechenbach-Berufskollegs sogar noch mehr Spaß an der Lösung kniffliger mathematischer Aufgaben – seitdem sie nämlich an vernetzten Laptops und mit modernster Software arbeiten können.

Die Software kommt aus Paderborn von der Firma SciFace, ist in Zusammenarbeit mit der Universität Paderborn entwickelt worden, hat bereits mehrere Auszeichnungen erhalten, heißt „MuPAD“ – und ist laut Wolfgang Kramer, Mathematiklehrer am Fechenbach-Berufskolleg, sensationell. Der Pädagoge ist überzeugt davon, dass der Einsatz des Computeralgebra-Systems Lehrern und Schülern Vorteile bringt: „Mit Hilfe von MuPAD können moderne Ansätze der Didaktik im Mathematikunterricht hervorragend umgesetzt werden. Teile einer Aufgabe werden dem Programm übergeben. Die Schüler können sich auf die wesentlichen mathematischen und technologischen Fragestellungen konzentrieren“, lobt Kramer.

Aber nicht nur die Schule, sondern auch die Paderborner Software-Entwickler profitieren davon, wenn die Fechenbach-Schüler mit dem Programm arbeiten. „Wir versprechen uns von der Zusammenarbeit Erkenntnisse, um MuPAD für den Schulalltag noch optimieren zu können. Wir werten die Unterrichtsmaterialien und Projektarbeiten aus. Die Ergebnisse stellen wir über das Internet anderen Lehrern und Schülern zur Verfügung“, sagt Dr. Andreas Sorgatz von SciFace.

Bevor David Oesterhaus, Andreas Diekmann und ihre Mitschüler mit der Software lernen können, haben Andreas Sorgatz und ein Kollege die Mathe-Lehrer „fit“ gemacht im Umgang mit MuPAD.

Das Programm nimmt dem Nutzer das rein mechanische rechnen ab und visualisiert mathematische Sachverhalte. „MuPAD kann mit beliebig großen Zahlen und mit Variablen rechnen, differenzieren, integrieren und Funktionen, Kurven und 3-D-Körper darstellen. Es kann auch zum Programmieren oder einfach wie ein sehr leistungsfähiger Taschenrechner eingesetzt werden. Die Schüler können sich dabei auf das Wesentliche konzentrieren: Modellbildung und Problemlösungsstrategien“, zählt Sorgatz auf.

Alles dem Computer überlassen können die Schüler natürlich nicht: „Wir wollen den Unterricht nicht revolutionieren, sondern unterstützen und vereinfachen“, schmunzelt Sorgatz.

Von einer Demonstration des Programms im Fechenbach-Berufskolleg war auch Landrat Friedel Heuwinkel höchst beeindruckt: „Das zeigt mir, wie wichtig es ist, mit diesen Medien lernen zu können, um die Ausbildung an unseren Schulen auf den neuesten Stand zu bringen.“ Dass der Landrat in „seine“ Schule gekommen war, freute den Leiter des Fechenbach-Berufskollegs, Heinrich Lindau, außerordentlich. Verband er doch mit diesem Besuch die Hoffnung, dass „wir es mit Unterstützung des Landrats vielleicht möglich machen können, eine eigene Notebook-Klasse einzurichten“.

Auszüge aus dem Presseecho

Die Materialien und Schülerarbeiten, die im Rahmen dieses Projektes entwickelt werden, stehen in einer späteren Projektphase auch im Word-Wide-Web zur Verfügung. Weitere Informationen und Verweise zu diesem Modellprojekt finden Sie unter `http://www.sciface.com/edu/projects/linda/` und erhalten Sie auf Anfrage von der Projektleitung des Felix-Fechenbach-Berufskollegs:

Stichwort "linda"
Wolfgang Kramer
Felix-Fechenbach-Berufskolleg
Saganer Straße
D-32756 Detmold
Email: linda@ffb.lippe.de
Fax: ++49 (0)5231 608288

SmartTools: Entwicklung von Lernsoftware

Frank Postel
MuPAD–Forschungsgruppe, `frankp@mupad.de`

Dieser Artikel beschreibt zwei visuelle Software–Komponenten, die die Entwicklung mathematischer Lernsoftware unterstützen. Diese Komponenten greifen auf den Kern des Computeralgebra–Systems MuPAD *zu und gestatten es somit, die Lerneinheiten mit mathematischer „Intelligenz" auszustatten. Zur Entwicklung der Lernsoftware lassen sich Entwicklungsumgebungen wie Visual Basic und Delphi verwenden, Autorensysteme wie Authorware und Director oder auch Anwendungen wie PowerPoint, Microsoft Word und der Internet Explorer. Diese Systeme gestatten die Einbindung und das Konfigurieren dieser Komponenten durch den Entwickler der Lernsoftware.*

Die Zielsetzungen, die mit diesen Software–Komponenten verbunden werden, bestehen zum einen in einer signifikanten Reduzierung des Entwicklungsaufwands leistungsstarker mathematischer Lernsoftware, die sich insbesondere durch die Verwendung des Computeralgebra–Systems MuPAD ergeben können, zum anderen in der Verbindung mathematischer Komponenten mit Entwicklungsumgebungen wie beispielsweise Visual Basic oder Delphi als auch von Präsentationswerkzeugen wie z.B. Authorware, Director oder dem Internet Explorer. Sie ermöglichen daher auch die Erweiterung klassischer Lernsoftware aus dem Bereich „Computer–based Learning" um mathematisches Expertenwissen und können somit zu einer wesentlichen Verbesserung der Qualität klassischer Lernsoftware beitragen.

Die Verwendung der Komponenten setzt zwar an den Autor ein gewisses Maß an technischem Wissen und Programmiererfahrung voraus, jedoch wird über das „plug & play" ähnliche Vorgehen, also dem Einbetten und Skripten der Komponenten in moderne Softwaresysteme, die Komplexität der Entwicklung von Lernsoftware erheblich reduziert, so daß die Verwendung solcher Komponenten den Kreis potentieller Autoren stark vergrößern helfen kann.

Die beiden folgenden visuellen Komponenten werden in diesem Artikel kurz vorgestellt und ihre technischen Eigenschaften und Möglichkeiten demonstriert:

Simple Calculator Control – zur Darstellung und Bearbeitung mathematischer Formeln und Berechnungen.

Graph 3D Control – zur Darstellung und interaktiven Bearbeitung dreidimensionaler Grafiken.

Beide Komponenten liegen als sogenannte ActiveX-Controls vor, eine auf dem Windows–Betriebssystem von Microsoft basierende Technologie. Sie stellen eine Vielzahl von Methoden, Eigenschaften und Ereignissen zur Verfügung, über die der Entwickler von Lerneinheiten, im folgenden Autor genannt, sie hinsichtlich anwendungsbezogener Aspekte zusammenstellen und konfigurieren kann.

In der Entwicklungsumgebung Delphi von Inprise greift der Autor auf die Methoden und Eigenschaften der Komponenten mit der Programmiersprache Delphi[1] zu. Innerhalb von Authorware von Macromedia verwendet der Autor die systemeigene Skriptsprache Lingua. In HTML–Dokumenten für den Internet Explorer lassen sich die Komponenten über JavaScript konfigurieren.

Die Rechner–Komponente

Das `Simple Calculator Control` (im folgenden kurz: *Rechner–Komponente* genannt) dient der Durchführung mathematischer Berechnungen wie beispielsweise das Lösen von Gleichungen, das Differenzieren und Integrieren von Funktionen oder aus dem Bereich der Linearen Algebra die Durchführung des Gaußschen Eliminationsverfahrens, die Berechnung von Determinanten oder die Bestimmung von Eigenwerten einer Matrix.

Der Autor kann u.a. bestimmen, welche Operationen dem Lernenden zur Verfügung stehen, wie feinschrittig die Berechnungen ablaufen sollen und wie die Rechner–Komponente mit den übrigen Teilen der Lernsoftware interagiert. Darüberhinaus ist es dem Autor möglich, zur Laufzeit auf den aktuellen Rechenweg innerhalb der Rechner–Komponente zuzugreifen bzw. Inhalte der Rechnung zu editieren. Darüber können dem Lernenden verschiedene Stufen von Hilfestellungen geliefert werden, vom Vorschlag eines konkreten Rechenschrittes bis hin zur vollständigen (Muster–)Lösung einer Aufgabe.

Die Rechner–Komponente verfügt über keinerlei mathematisches Wissen, sie ist allein für die Darstellung und Editierung einer mathematischen Rechnung zuständig. Erst in Verbindung mit dem MuPAD–Kern wird aus dem Zusammenspiel dieser beiden Komponenten ein Werkzeug, mit dem mathematische Berechnungen durchgeführt werden können.[2]

Eingabe und Ausgabe mathematischer Ausdrücke

Die Eingaben und Ausgaben mathematischer Berechnungen müssen in einer dem Lernenden vertrauten Form erfolgen. Dazu integriert die Rechner–Komponente eine grafische Eingabe mathematischer Ausdrücke.

Die Eingabe von Operatoren geschieht über sogenannte *Templates*, d.h. zweidimensionale grafische Layoutmasken, die die Darstellung eines Operators definieren. Die Eingabe mathematischer Ausdrücke über Templates findet sich bereits in Softwareprodukten aus den früheren 80'er Jahren und wird noch heute in modernen Computeralgebra–Systemen und Textverarbeitungssystemen (wie beispielsweise dem Formeleditor von Microsoft) verwendet.

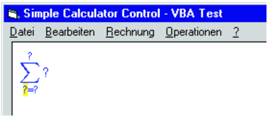


Bild 1:

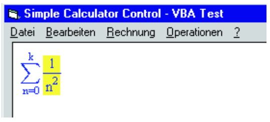


Bild 2:

Abbildung 1: Rechner–Komponente — Das Summen-Template

Das linke Bild in Abbildung 1 zeigt das Template für die Eingabe von Summen mit den drei Platzhaltern — durch Fragezeichen markiert — für die untere und obere Schranke sowie den Summanden. Im rechten Bild der Abbildung sind die Platzhalter vom Lernenden durch konkrete Werte belegt worden.

[1]Ein objekt–orientiertes Pascal.

[2]Der *MuPAD*-Kern steht auch als eigenständige Komponente zur Verfügung und kann von beliebigen Anwendungen aus über eine entsprechende Schnittstelle für Berechnungen genutzt werden. In diesem Artikel wird die Verwendung des *MuPAD*-Kerns allerdings nur in Verbindung mit den beiden oben genannten Komponenten beschrieben.

Jedem Template ist ein eindeutiger Namen zugeordnet und kann vom Autor über die Methode `InsertTemplate(name)` der Rechner–Komponente in die aktuelle Eingabeposition eingefügt werden.

Das Template für Summen beispielsweise trägt den Namen `"sum"`, d.h. in VBA[3] bewirkt die folgende Routine das Einsetzen des Templates, wobei die Instanz der Rechner–Komponente den Namen `Calc` trägt:

```
Private Sub TemplateSum()
  Calc.InsertTemplate "sum"
  Calc.GetFocus
End Sub
```

Die Rechner–Komponente stellt eine Vielzahl von vordefinierten Templates zur Verfügung, darunter Templates zur Eingabe von Integralen, Grenzwerten, Matrizen und vielen mehr. Es besteht für den Autor sogar die Möglichkeit, das Erscheinungsbild der Templates auf seine Bedürfnisse hin anzupassen, worauf in diesem Artikel jedoch nicht eingegangen werden kann.

Der Autor muß eine Auswahl von Templates zusammenstellen, die der Lernende zur Eingabe von Ausdrücken verwenden soll. Da die Rechner–Komponente über keine vordefinierten Menüs, Dialoge oder andere Eingabeformen verfügt[4], muß der Autor selbst entsprechende textuelle oder grafische Auswahlmöglichkeiten dem Lernenden zur Verfügung stellen.[5]

Abschließend sei noch erwähnt, daß Teilausdrücke innerhalb der Komponente kopiert bzw. ausgeschnitten und an anderer Stelle eingefügt werden können. Darüberhinaus ist ein mehrstufiges Undo/Redo implementiert, um dem Lernenden zu ermöglichen, Editionen an Eingaben in der Rechner–Komponente rückgängig machen bzw. wiederherstellen zu können.

Ausführen von Berechnungen

Jede mathematische Operation ist über einen (eindeutig zu wählenden) Namen mit einer MuPAD–Prozedur verknüpft, die in einer MuPAD–Quelldatei definiert ist. Zur Kommunikation mit dem MuPAD–Kern, u.a. zum Ausführen der Operationen, stellt die Rechner–Komponente zwei Verfahren zur Verfügung, die vom Autor je nach der Komplexität seiner Anwendung in Kombination verwendet werden:

1. Zum einen können über die Methoden `SetOperation` und `Execute` Operationen ausgewählt bzw. ausgeführt werden. Hierbei kümmert sich die Rechner–Komponente um den Eintrag der Operation in den aktuellen Rechenweg, um den Aufruf der entsprechenden MuPAD–Prozedur mit der Übergabe der Parameter (der Ausdruck, auf den die Operation angewendet werden soll, und — falls vorhanden — die zusätzlichen Argumente) sowie ggf. um den Eintrag des Ergebnisses der Operation in den aktuellen Rechenweg.

2. Darüberhinaus gibt es die Möglichkeit, beliebige MuPAD–Anweisungen über die Methode `ExecuteString` als Zeichenkette an den MuPAD–Kern zu senden und auszuführen. Das Ergebnis muß vom Autor explizit ausgewertet werden und steht in der Eigenschaft `ExecuteStringResult` zur Verfügung.

   Da es möglich ist, zur Laufzeit von dem MuPAD–Kern aus auf die Elemente einer Rechnung zuzugreifen, erhält der Autor über dieses Verfahren der Kommunikation mit dem MuPAD–Kern und der Implementierung geeigneter MuPAD–Funktionen eine vollständige Kontrolle über den Berechnungsablauf des Lernenden.

[3] VBA = Visual Basic für Applikationen. Diese Programmiersprache wird u.a. in Microsoft Word, PowerPoint und in der Entwicklungsumgebung Visual Basic zur Steuerung der Komponenten verwendet.

[4] Es gibt eine Ausnahme: Wird das Template `"matrix"` zur Eingabe von Matrizen ausgewählt, so öffnet die Komponente einen Dialog, in dem der Lernende die Zeilen- und Spaltenzahl der Matrix festlegen muß.

[5] Visual Basic, Delphi von Inprise oder Authorware stellen eine Vielzahl von Werkzeugen zur Erstellung geeigneter grafischer Eingabeelemente zur Verfügung. Das ermöglicht die Anpassung der Rechner–Komponente an die spezifischen Anforderungen der zu entwickelnden Lernsoftware.

Verwendet der Autor allein die beiden Methoden der ersten Variante zur Kommunikation mit dem MuPAD–Kern und zur Ausführung von Operationen, so entsteht ein von der Rechner–Komponente vorgegebener Berechnungsablauf. Eine neue Berechnung beginnt hierbei mit einem mathematischen Ausdruck, den der Lernende eingeben kann. Dieser Ausdruck definiert die *Wurzel* der nun entstehenden *Rechnung*.

Der Lernende beginnt die Berechnung über die Auswahl einer vom Autor zur Verfügung gestellten Operation, die auf die Wurzel der Rechnung angewendet wird. Das Ergebnis wird vom MuPAD–Kern berechnet und von der Rechner–Komponente als Ergebnis unterhalb der ausgewählten Operation eingetragen. Der Lernende ist nun erneut aufgefordert, eine Operation auszuwählen, die auf das zuvor berechnete Ergebnis angewendet werden soll.

Dadurch entsteht eine Abfolge von *Rechenschritten*, wobei sich ein Rechenschritt immer aus dem Ergebnis der vorhergehenden Operation und ggf. einer weiteren Operation zusammensetzt, die auf das Ergebnis angewendet werden soll (siehe Abbildung 2).

Die Ausdrücke der Rechenschritte sind vom Lernenden nicht veränderbar, mit Ausnahme der Wurzel der Rechnung. Auch die Operation eines Rechenschrittes kann vom Lernenden nachträglich durch eine neue Operation ersetzt werden. Wird die Wurzel der Rechnung oder die Operation eines Rechenschrittes verändert, so wird der darunterliegende Teil der Rechnung von der Rechner–Komponente visuell durch diagonale Streifen als ungültig markiert. Die Rechnung kann von dieser Stelle aus neu ausgeführt werden, wobei die darunterliegenden Ergebnisse zu den Rechenschritten schrittweise durch die neuen Ergebnisse ersetzt werden.[6] Dabei kann der Lernende bei jedem Rechenschritt die bei der ersten Berechnung gewählte Operation ausführen lassen oder auch eine neue Operation auswählen, wie es die Abbildung 3 zeigt.

Diese sehr einfache Strategie eines Berechnungsablaufes entlastet den Autor bei der „Organisation“ des Rechenweges. Jedoch wird die Aktivität des Lernenden hierbei auf die Auswahl von Operationen beschränkt, da die Ergebnisse nicht vom Lernenden eingeben werden, sondern vom Computeralgebra–System berechnet werden. Ebenso fehlt die Möglichkeit, die Aktivitäten des Lernenden zu analysieren, um rechtzeitig entsprechende Rückmeldungen liefern zu können.

Die zweite Variante zur Kommunikation mit dem MuPAD–Kern besteht in der Ausführung einer (beliebigen) MuPAD–Anweisung über die Methode `ExecuteString`. Das Ergebnis der MuPAD–Anweisung wird in der Eigenschaft `ExecuteStringResult` festgehalten, sobald das Ereignis `ExecuteStringReady()` ausgelöst worden ist.

Diese Form der Kommunikation kommt beispielsweise dann zum Tragen, wenn die vom Lernenden ausgewählte Operation analysiert werden soll, um ihm ggf. frühzeitig entsprechende Rückmeldungen liefern zu können. Weiterhin erhält der Autor hierüber die Möglichkeit, das Ergebnis zu einer Operation nicht vom MuPAD–Kern ausrechnen, sondern vom Lernenden eingeben zu lassen. Bei Bedarf kann auch das Ergebnis wieder von einer speziellen MuPAD–Prozedur auf Korrektheit überprüft und eine entsprechende Rückmeldung geliefert werden.

Es ist dem Autor darüberhinaus möglich, Daten in den aktuellen Rechenweg einzutragen, um beispielsweise dem Lernenden eine Operation vorzuschlagen oder das Ergebnis einer Operation vorzurechnen. Der Autor muß dafür MuPAD–Prozeduren zur Verfügung stellen, die auf die Rechnung innerhalb der Rechner–Komponente zugreifen und die Inhalte einer Rechnung analysieren. Solche MuPAD–Prozeduren müssen einen Wert zurückliefern (eine ganze Zahl, eine Fließkommazahl, eine Zeichenkette oder ein boolscher Wert), der dem Autor als Information darüber, wie beispielsweise auf eine bestimmte Aktivität des Lernenden reagiert werden soll (z.B. durch Anzeige eines entsprechenden Hinweises, falls der Lernende ein falsches Ergebnis angegeben hat), dient.

Es soll in diesem Artikel diesbezüglich nicht weiter auf technische Details eingegangen werden und stattdessen kurz auf ein umfangreicheres Beispiel eingegangen werden, das diese Funktionalitäten der Rechner–Komponente ausnutzt.

[6]Die Rechner–Komponente überprüft hierbei nicht die mathematische Konsistenz der Berechnungen, es handelt sich um einen rein visuellen Hinweis, daß die Rechnung möglicherweise nicht mehr in sich konsistent ist.

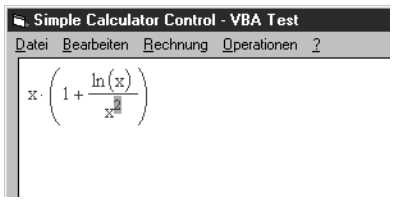


Bild 1: Eingabe durch den Lernenden

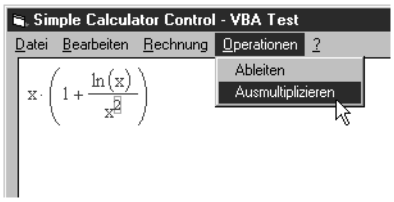


Bild 2: Auswahl durch den Lernenden

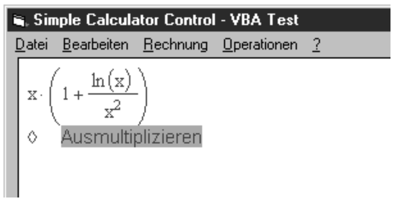


Bild 3: Nach Aufruf von `SetOperation`

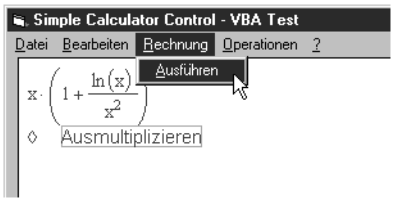


Bild 4: Auswahl durch den Lernenden

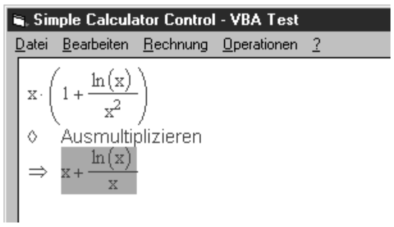


Bild 5: Nach Aufruf von `Execute`

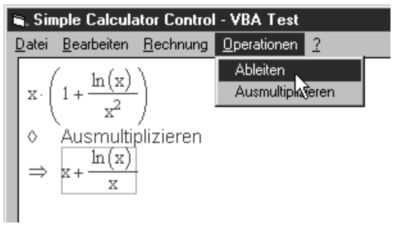


Bild 6: Auswahl durch den Lernenden

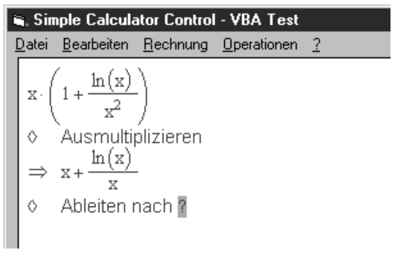


Bild 7: Nach Aufruf von `SetOperation`

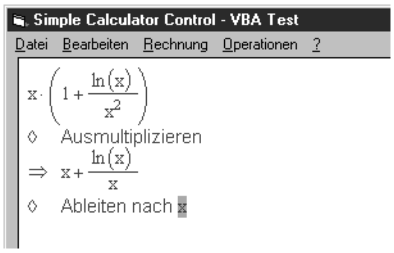


Bild 8: Eingabe durch den Lernenden

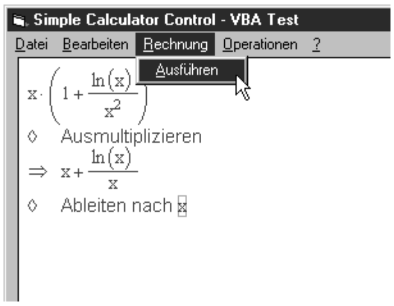


Bild 9: Auswahl durch den Lernenden

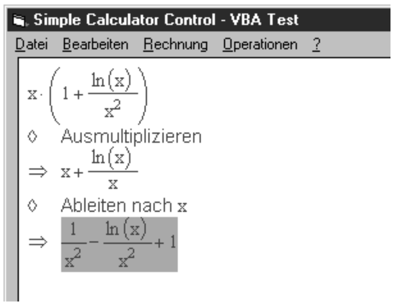


Bild 10: Nach Aufruf von `Execute`

Abbildung 2: Rechner–Komponente — Der Standard–Berechnungsablauf über die Methoden `SetOperation` und `Execute`

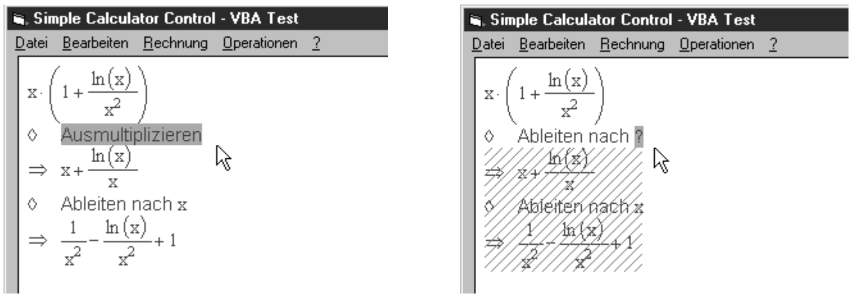


Abbildung 3: Rechner–Komponente — Inkonsistente Rechenschritte in einer Rechnung

Es handelt sich um einen sogenannten „Schrittrechner“ zur Algebra der Sekundarstufe I. Der hier vorgestellte Prototyp implementiert das Umformen von Termen nach bestimmten Umformungsregeln, die auf den Körpergesetzen in $\mathbb{Q}$ beruhen, in wertgleiche Terme, meist mit dem Ziel, sie in eine möglichst einfache Gestalt zu bringen. Zudem können lineare Gleichungen und Ungleichungen in äquivalente umgeformt werden mit dem Ziel, alle Lösungen der Ausgangsgleichung anzugeben.

Der Lernende soll in kleinen Schritten rechnen. Ihm steht es aber frei, Verkürzungen des Lösungsweges vorzunehmen.

Neben den rein mathematischen Anforderungen bietet der Schrittrechner tutorielle Aspekte, die den Lernenden beim Lösen einer (beliebigen) Aufgabe aus einer der unterstützten Aufgabenklassen begleitet. Der Schrittrechner überwacht den Lösungsprozeß des Lernenden und greift ein, sobald Gefahr droht, daß der Lernende von einem korrekten Lösungsweg abweicht. Zudem steht der Schrittrechner dem Lernenden in Form eines Experten jederzeit als Ratgeber zur Verfügung und schlägt auf Wunsch eine Umformungsregel vor oder berechnet das Ergebnis zur Anwendung einer Umformungsregel.

Die Bedienung des Schrittrechners gestaltet sich wie folgt: Der Lernende gibt einen beliebigen mathematischen Ausdruck ein und bestätigt die Eingabe durch Drücken der Eingabetaste. Daraufhin wird vom Schrittrechner die Aufgabenklasse festgestellt, zu der der eingegebene Ausdruck gehört. Nun können folgende Fälle auftreten:

- Ist die Eingabe (Auswahl einer Umformungsregel oder Eingabe eines Ergebnisses zur Anwendung einer Umformungsregel) vom Lernenden erfolgt, so wird die Eingabe auf Korrektheit überprüft. Wird die Eingabe akzeptiert, so wird ein Schritt innerhalb der Rechnung weitergegangen. Anderenfalls wird eine entsprechende Meldung auf dem Bildschirm ausgegeben und der Lernende erneut aufgefordert, eine Eingabe zu tätigen.
- Ist keine Eingabe vom Lernenden erfolgt[7], so wird das als Anfrage an den Tutor des Schrittrechners interpretiert. Über die entsprechenden (tutoriellen) Funktionen des Schrittrechner wird daraufhin dem Lernenden ein Schritt vorgegeben.

Dieser Ablauf setzt sich fort, bis der Schrittrechner das Ende der Rechnung ermitteln konnte. Die Abbildung 4 zeigt einen typischen Berechnungsablauf.[8]

Die Bedienungsoberfläche und die tutorielle Strategie, d.h. die Strategie eines Berechnungsablaufs, dieses Schrittrechners wurde mit Visual Basic und auf Basis der Rechner–Komponente entwickelt.

Die mathematische und tutorielle Funktionalität des Schrittrechners wurde in Form einer MUPAD–Bibliothek implementiert, die bereits Einsatz in einer Lernsoftware namens Mathlantis von Cornelsen Software gefunden hat.

[7]Keine Eingabe liegt vor, wenn anstelle eines mathematischen Ausdrucks der Platzhalter (grafisch durch ein „?“ gekennzeichnet) bzw. anstelle einer Umformungsregel das Fragezeichen gesetzt ist.

[8]Wie bereits erwähnt, kann der Lernende auch jederzeit direkt die Hilfe des Algebra–Experten über den entsprechenden Eintrag des Menüs „Tutor“ aufrufen.

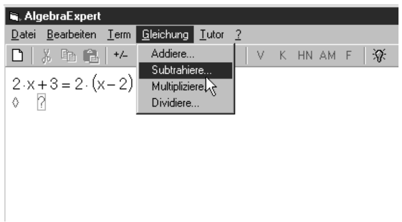


Bild 1: Auswahl einer Umformungsregel durch den Lernenden

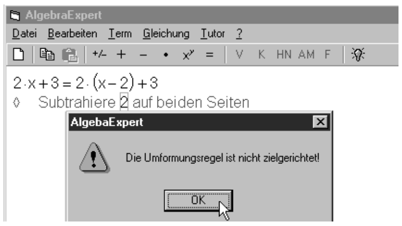


Bild 2: Umformungsregel wurde nicht akzeptiert

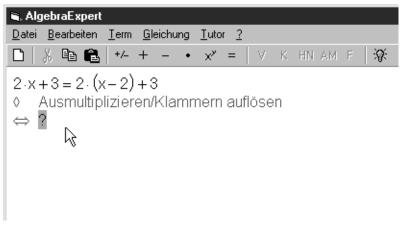


Bild 3: Eingabe des Ergebnisses durch den Lernenden

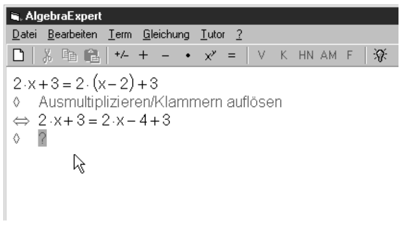


Bild 4: Ergebnis ist korrekt, wähle die nächste Umformungsregel

Abbildung 4: Ein „Schrittrechner“ zur Algebra der Sekundarstufe I — Beispiel eines Berechnungsablaufes

Mathlantis gliedert sich in insgesamt fünf sogenannte Bände und richtet sich an die Klassen 5 – 10 verschiedener Schulformen. Der Bibliothek des hier vorgestellten Schrittrechners ist (in erweiterter und etwas abgeänderter Funktionalität) Teil des Bandes „Algebra 1“ von Matlantis.[9]

Die 3D-Grafik-Komponente

Das `Graph 3D Control` (im folgenden kurz: *Grafik–Komponente* genannt) dient der Visualisierung dreidimensionaler mathematischer Daten. Neben den reinen Darstellungsmöglichkeiten bietet es zahlreiche Interaktionsformen mit der Grafikszene und deren Objekte, beispielsweise das Anwenden beliebiger affiner Transformationen (Drehung, Spiegelung, Verschiebung etc.) auf ausgewählte Objekte oder das Verknüpfen bestimmter Aktionen an Mausklicks und Mausbewegungen, darunter das Verschieben eines Objektes entlang einer Geraden oder auf einer Ebene.

Die Grafik–Komponente dient primär der Entwicklung von Lernsoftware aus dem Bereich der Analytischen Geometrie. Daher bietet die Komponente eine Vielzahl von grafischen Objekten wie beispielsweise Punkte, Vektoren, Ebenen, Kugeln, Kegel oder Zylinder.

Der Autor kann auf Attribute der Grafikszene und deren Objekte zur Laufzeit zugreifen, verschiedene Interaktionsmöglichkeiten dem Lernenden zur Verfügung stellen sowie Mausereignisse des Lernenden mit bestimmten Aktionen verknüpfen.

Wie die Rechner–Komponente stellt auch die Grafik–Komponente dem Lernenden keine Eingabeformen zum Zugriff auf die Grafikszene und deren Objekte, zum Einfügen neuer Grafikobjekte oder zur Ausführung von Operationen und Interaktionen zur Verfügung. Diese müssen vom Autor bereitgestellt werden, die er meist mit den Werkzeugen der von ihm gewählten Entwicklungsumgebung erstellt. Dadurch wird es möglich, Instanzen der Grafik–Komponente hinsichtlich der Funktionalität und der Bedienung gezielt auf die Anforderungen der Lernsoftware hin anzupassen.

[9] Cornelsen Software: *Mathlantis — Algebra 1*. Cornelsen Verlag, Berlin, 1999. PC-CD-ROM.

Es soll an dieser Stelle nicht auf die Vielzahl der von der Komponente angebotenen Methoden eingegangen werden, sondern deren Funktionalität an zwei kleineren Beispielen verdeutlicht werden.

Zwei Beispiele zum Einsatz der Grafik–Komponente

Es werden die drei Eulerschen Drehmatrizen definiert, die jeweils eine Drehung um eine der Raumachsen beschreiben. Der Drehwinkel wird dabei in Grad angegeben.

Die Prozedur `Main` (siehe unten) fügt einen Kegel und eine den Kegel schneidende Ebene in die Grafikszene ein. Die Ebene wird in einer roten Farbe, der Kegel in einer grünen Farbe gezeichnet. Anschließend wird die Ebene um $10°$ um die x–Achse und um $20°$ um die y–Achse (jeweils entgegen dem Uhrzeigersinn) gedreht.

Die Instanz der Grafik–Komponente ist im Programmcode (in VBA–Syntax) mit `Graph3D` bezeichnet:

```
Private Sub Main()
  Dim ebene As Integer, kegel As Integer

  ' Füge eine Ebene in die Grafikszene ein:
  ebene = Graph3D.AddPlane(0, 0, 1, 0, 0, 0.5, 0.6)
  ' und einen Kegel:
  kegel = Graph3D.AddCone(0, 0, 0, 0, 0, 1, 0.7, 0, 40)

  ' und färbe das eine Objekt Rot, das andere Blau-Grün:
  Graph3D.SetObjectColor ebene, 0.8, 0.1, 0.1
  Graph3D.SetObjectColor kegel, 0#, 1#, 0#

  ' Drehe die Ebene um die x-Achse um 10° entgegen dem
  ' Uhrzeigersinn:
  DrehungXAchse ebene, -10
  ' und anschließend um die y-Achse um 20° entgegen dem
  ' Uhrzeigersinn:
  DrehungYAchse ebene, -20
  ' Führe die beiden Drehungen aus:
  Graph3D.TransformObjectStart ebene
End Sub

Private Sub DrehungXAchse(n As Integer, alpha As Double)
' 1. Eulersche Drehmatrix (Drehung um die y-Achse)
  alpha = alpha / 180 * 3.141592654
  Graph3D.TransformObjectSetTransform n, _
    1, 0, 0, _
    0, Cos(alpha), -Sin(alpha), _
    0, Sin(alpha), Cos(alpha), _
    0, 0, 0, 20
End Sub
```

Entsprechend werden die Routinen `DrehungYAchse` und `DrehungZAchse` implementiert.

Interessant wird dieses Beispiel, wenn die Drehwinkel nicht fest vorgegeben werden, sondern vom Lernenden eingegeben werden können, wie es die Abbildung 5 zeigt.

Ein etwas umfangreicheres Beispiel zur Interaktion mit dem Lernenden zeigt die Abbildung 6. Es sollen Ortsvektoren in ein Koordinatensystem eingezeichnet werden, indem die Koordinaten eines Ortsvektors vom Lernenden durch Abfahren eines Punktes (dem sogenannte „Curser“) auf der x–, y– bzw. z–Achse festgelegt wird.

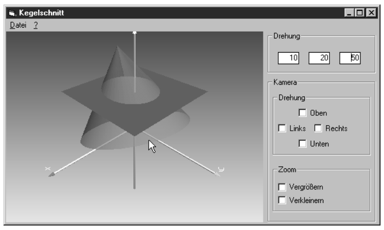


Bild 1: Drehung des Kegels

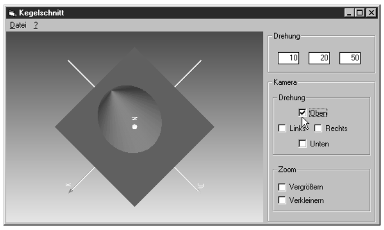


Bild 2: Drehung der Szene (Schnittebene parallel zur Bildschirmebene)

Abbildung 5: Grafik–Komponente — Beispielanwendung in Visual Basic: Kegelschnitte

Nach Betätigung der linken Maustaste wird die entsprechende Koordinate fixiert und die Bewegung des Punktes durch die Maus auf die entsprechende Achse der nächsten Koordinate eingeschränkt. Sind alle drei Koordinaten festgelegt worden, wird der entsprechende Ortsvektor eingezeichnet. Auf die Angabe des vollständigen VBA–Quellcodes wird verzichtet. Die für die entsprechenden Interaktionen zentralen Methoden der Grafik–Komponente sind im folgenden Programmausschnitt zu sehen:

```
  ...
  ' Füge den Cursor zum Einzeichnen des Vektors ein:
  Cursor = Graph3D.AddSphere(0#, 0#, 0#, Kugelradius, 20)
  Graph3D.SelectObject Cursor

  ...
  ' Fixiere die Bewegung auf die x-Achse und aktiviere die Selektion
  ' des Cursors:
  Graph3D.ShiftSelectedSetAxis 1.0, 0.0, 0.0
  Graph3D.SelectOn
```

Das Ziehen des Cursors soll nur aktiviert werden, wenn der Cursor zuvor selektiert worden ist. Eine Selektion eines Grafikobjektes durch den Lernenden bewirkt das Auslösen des Ereignisses `ObjectSelected` durch die Grafik–Komponente. Der Autor kann auf solche Ereignisse reagieren, indem er eine entsprechende sogenannte „Ereignisprozedur“ implementiert. Der Name einer solchen Ereignisprozedur setzt sich aus dem Namen der Instanz der Komponente und dem Namen des Ereignisses zusammen.

Dementsprechend wird folgende Ereignisprozedur in VBA–Syntax implementiert, um auf die Selektion des Cursors in geeigneter Form zu reagieren:

```
Private Sub Graph3D_ObjectSelected()
  If Graph3D.GetSelectedObject() = Cursor Then
    Graph3D.ShiftSelectedOn
  End If
End Sub
```

Die eigentliche Steuerung zum Einzeichnen eines Vektors, die abhängig von der aktuellen Koordinate die Bewegung der Maus auf die nächste Achse fixiert bzw. die Eingabe beendet, wird wie folgt erreicht:

```
Private Sub Graph3D_ShiftSelectedFinished()
  Dim vektor As Integer

  ' Deaktiviere das Verschieben des Cursors:
  Graph3D.ShiftSelectedOff
```

```
  Select Case Koordinate
  Case 0
    ' Zeichne einen Vektor in Rot vom Ursprung zum aktuellen Punkt
    ' auf der x-Achse:
    xVektor = Graph3D.AddVector( _
      0.0, 0.0, 0.0, xKoord, 0.0, 0.0, 2*Pfeilradius _
    )
    Graph3D.DisableSelect xVektor
    Graph3D.SetObjectColor xVektor, 1.0, 0.0, 0.0

    ' Die Bewegung des Cursors wird in Richtung der y-Achse fixiert:
    Graph3D.ShiftSelectedSetAxis 0.0, 1.0, 0.0
    Koordinate = Koordinate + 1
  Case 1
    ...
  Case 2
    ...
  End Select
End Sub
```

Die Fälle 1 und 2 für die (globale) Variable `Koordinate` im angegebenen VBA–Code sind ähnlich. Im Fall 2 wird zusätzlich der Vektor eingezeichnet und der Wert für `Koordinate` wieder auf 0 gesetzt.

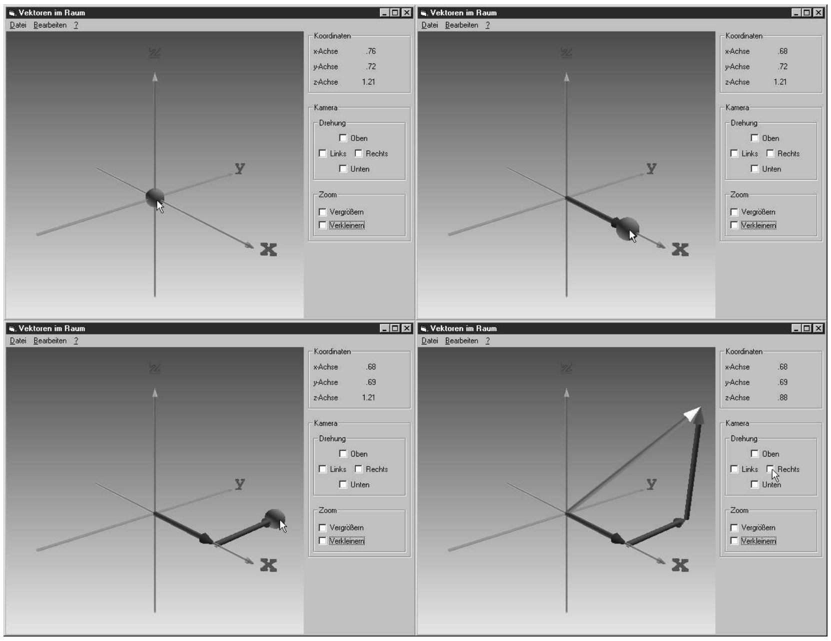


Abbildung 6: Grafik–Komponente — Beispielanwendung in Visual Basic: Einzeichnen von Vektoren im Raum

Die Kommunikation mit MUPAD

Für den Autor kann es sehr nützlich sein, wenn beispielsweise für Schnittberechnungen, umfangreichere Matrizenrechnungen oder für Korrektheitsüberprüfungen von Konstruktionen auf die mathematischen Fähigkeiten eines Computeralgebra–Systems zurückgegriffen werden kann.

Zur Lösung dieses Problems kann der Autor eine Instanz der Rechner–Komponente in seine Anwendung integrieren, um deren Möglichkeit zur Kommunikation mit dem MUPAD–Kern, aber auch, falls gewünscht, zur Darstellung mathematischer Ausdrücke zu nutzen.

Die Abbildung 7 zeigt ein Beispiel, mit dem die gegenseitige Lage dreier Ebenen im Raum untersucht werden soll. Bei jeder Veränderung einer der drei Ebenengleichungen werden die drei Ebenen neu gezeichnet und der Schnitt der drei Ebenen über einen Aufruf des MUPAD–Kerns neu berechnet. Das Ergebnis der Schnittberechnung wird zudem mit Hilfe der eingebetteten Rechner–Komponente auf dem Bildschirm ausgegeben.

Auf die Angabe des VBA–Quellcodes wird an dieser Stelle verzichtet.

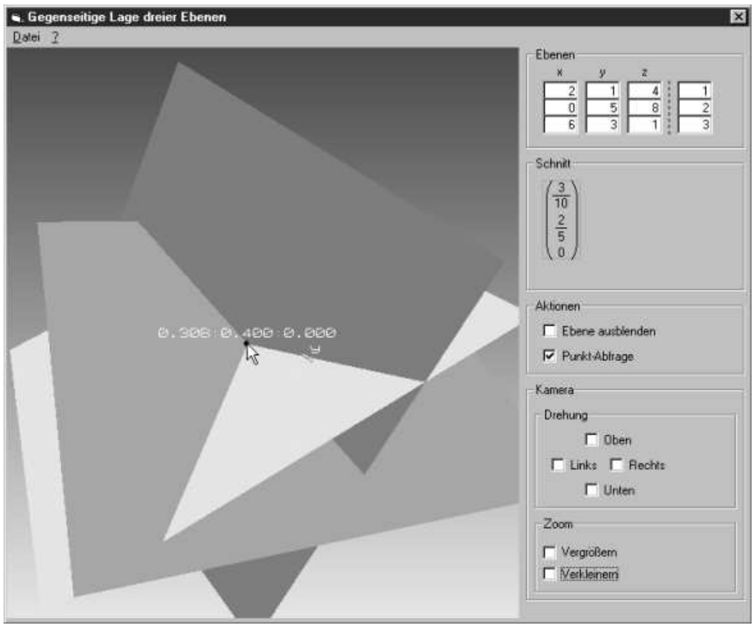


Abbildung 7: Grafik–Komponente — Beispielanwendung in Visual Basic: Lage dreier Ebenen im Raum

Die hier beschriebene Form der Kommunikation mit dem MUPAD–Kern wird aber nur in seltenen Fällen ausreichen. Benötigt wird eine direkte Verbindung zwischen der Grafik–Komponente und dem MUPAD–Kern, damit MUPAD–Funktionen ihre Ergebnisse unmittelbar an die Grafik–Komponente zur Visualisierung weiterreichen können bzw. die Grafik–Komponente ihrerseits auf Objekte und Funktionen des MUPAD–Kerns zugreifen kann. Im aktuellen Entwicklungsstand der Grafik–Komponente fehlt eine direkte Verbindung zum MUPAD–Kern.

Die Grafik–Komponente ist speziell zur Darstellung und Manipulation dreidimensionaler grafischer Szenen ausgelegt und nur bedingt für zweidimensionale Grafiken geeignet. Es gibt zwar die Eigenschaft `Dimension` der Schnittstelle `Graph3D`, mit der eine zweidimensionale Sicht ermöglicht wird (die Kamera des Betrachters wird so fixiert, daß sie von oben auf die dreidimensionale Grafikszene zielt), jedoch ist dieses Verfahren nur beschränkt sinnvoll. So sind die Textobjekte der Komponente im zweidimensionalen Fall nicht sichtbar, und Interaktionen beziehen sich weiterhin auf dreidimensionale Grafikobjekte.[10]

Aufgrund der unterschiedlichen Anforderungen an zwei- und dreidimensionale Grafiken, insbesondere hinsichtlich wünschenswerter Interaktionsformen, bietet es sich an, eine eigenständige Grafik–Komponente zur Darstellung und Manipulation zweidimensionaler grafischer Objekte zu entwickeln.

[10] So erlaubt die Komponente das Abfragen von Koordinaten nur von Flächenobjekten.

Desweiteren wurde nicht das Ziel verfolgt, eine Art „dynamisches Geometriesystem“, wie man es von Softwaresystemen wie CabriGeòmètre, Cinderella oder GEONET[11] für die Geometrie in der Ebene kennt, zu entwickeln. Es handelt sich vielmehr um eine relativ *statische*[12], jedoch weitgehend konfigurierbare Grafik–Komponente, die auf den MUPAD–Kern zugreifen soll, um symbolische Komponenten zu integrieren (z.B. Überprüfung der Korrektheit von Konstruktionen, Schnittberechnungen, algebraische Formulierung grafischer Operationen etc.). Zwar ließen sich auch während der direkten Manipulation der Objekte MUPAD–Anweisungen ausführen, jedoch muß berücksichtigt werden, daß MUPAD–Berechnungen für einen flüssigen Ablauf direkter Manipulation im allgemeinen zu laufzeitintensiv sind.

Zusammenfassung

Die Entwicklung dieser Komponenten zum derzeitigen Zeitpunkt ist noch nicht abgeschlossen. Es ist mit weiteren Anforderungen zu rechnen, sobald sie verbreiteten Einsatz in Entwicklungen von Lernsystemen finden.

Derzeit wird an einer zweidimensionalen Grafik–Komponente zur Darstellung und interaktiven Bearbeitung von Diagrammen und Funktionsgraphen gearbeitet.

In einer aktuellen Entwicklung eines Lernsystems zur Linearen Algebra auf Basis des Autorensystems Authorware werden diese Komponenten in Verbindung mit dem MUPAD–Kern verwendet. Diese Entwicklung wird in Kooperation mit dem Institut für Mediendidaktik der Universität in Koblenz unter Leitung von Herrn Prof. Dr. Fraunholz und dem Cornelsen Verlag durchgeführt.[13]

In einem aktuellen Entwicklungsprojekt von SciFace Software und der MUPAD–Gruppe wird derzeit an einer Softwarelösung gearbeitet, die die Nutzung des MUPAD–Kerns und anderer Komponenten in heterogen verteilten Rechner–Umgebungen erlaubt. Neben den dafür notwendigen standardisierten Austauschformaten für mathematische Daten und mit dem MUPAD–Kern berechnete Grafiken treten eine Vielzahl technischer Aspekte und Probleme auf, die sich u.a. aus dem (möglicherweise gleichzeitigen) Zugriff (mehrerer) entfernter Clients ergeben. Darunter fallen Fragen nach der Leistungsfähigkeit, Installation und Konfiguration des MUPAD–Kerns sowie der Stabilität und Sicherheit der Kommunikationsprozesse.

Das Forschungsprojekt hat zum Ziel, den MUPAD–Kern innerhalb von verteilten Lernumgebungen zur Verfügung zu stellen.

[11] Siehe den Artikel „Dynamische Mathematik mit GEONET“ in diesem Heft.

[12] Damit soll hier zum Ausdruck gebracht werden, daß im allgemeinen *während* der direkten Manipulation der Objekte durch den Lernenden Inhalte nicht neu berechnet und ausgegeben oder mathematische Bedingungen überprüft werden sollten (wie es bei dynamischen Geometriesystemen der Fall ist).

[13] Wolfgang Fraunholz and Frank Postel: *Eine Computer–Lernumgebung zur Linearen Algebra unter Verwendung des Computeralgebra–Systems MuPAD*. Eingereicht zur *The Fifth International Conference on Technology in Mathematics Teaching* (ICTMT 5), Klagenfurt, August 2001.

Mathematik im Web – Der MuPAD Computing Server

Andreas Sorgatz
SciFace Software, sorgatz@sciface.com

Interaktives Lehrmaterial mit mathematisch-technischem Inhalt im Internet zu präsentieren und zur Vor- und Nachbereitung von Lehrveranstaltungen oder auch als reine Web-Kurse (E-Learning) einzusetzen, ist ein weit verbreiteter Wunsch. Dieser Artikel gibt einen Überblick über das Konzept des MuPAD Computing Servers für algebraisch-symbolisches und numerisches Rechnen im World-Wide-Web und informiert über die aktuelle Entwicklung.

Am Anfang war die Idee

Auf Initiative und mit Förderung des Vereins *Schulen ans Netz e.V.* hat die *MuPAD*-Forschungsgruppe der Universität Paderborn in Zusammenarbeit mit SciFace Software ein Konzept entwickelt, das auf sehr einfache Weise ein algebraisch-symbolisches und numerisches Rechnen im World-Wide-Web ermöglicht. Kern dieses Konzeptes ist ein Client-Server Modell, bei dem das Computeralgebra-System *MuPAD* über einen Server Internet-fähig gemacht wird – vergleichbar zu einem Http- oder Ftp-Server. Als Klient dient eine Java-Kommunikationskomponente in Form eines Applets, das in die Web-Seite eingebunden wird. Als Klebstoff zwischen den interaktiven, ggf. multimedialen Eingabe- und Ausgabekomponenten und dem Applet innerhalb einer HTML-Seite dient JavaScript.

Nach Erstellung eines Prototypen und Abschluß des Projektes an der Universität Paderborn im Februar 2001 wird die Entwicklung nun von SciFace Software fortgeführt. Ziel ist die Realisierung einer einsatzreifen Version dieser Technik für die Lehre in Schulen und Hochschulen sowie für den Bereich des verteilten Rechnens in der Forschung und Entwicklung.

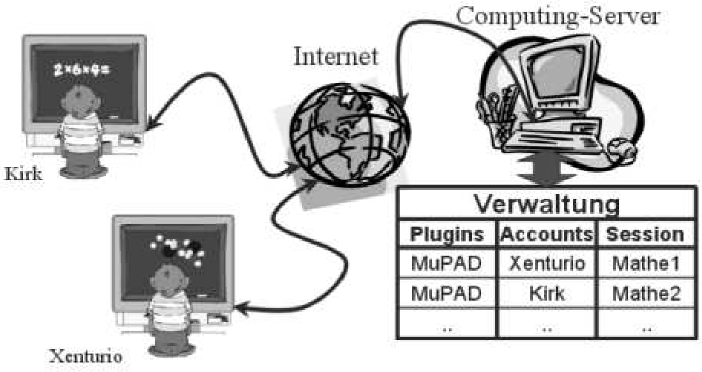


Abb. 1: Mathematik interaktiv im World-Wide-Web

Die Ergebnisse der laufenden Entwicklung bei SciFace werden unter anderem in zwei, vom Bundesministerium für Bildung und Forschung geförderte Projekte eingebracht: das Projekt *in2mat – Interaktive Mathematik- und Informatikausbildung* (`http://www.uni-koblenz.de/ag-ki/PROJECTS/in2math/`) der Hochschule für Technik und Wirtschaft des Saarlands, der Universität des Saarlandes, der Humboldt-Universität und der Universitäten Berlin, Koblenz-Landau und Paderborn sowie das Projekt *math-kit – Ein multimedialer Baukasten für die Mathematik-Ausbildung im Grundstudium* (`http://www.math-kit.de/`) der Fernuniversität Hagen und der Universitäten Bayreuth, Hamburg und Paderborn. Ein neues Projekt *Mathematik multimedial – Lernmodule Optimale Lösungen* läuft in diesen Tagen im Rahmen der Verbundstudien der Fachhochschulen NRW an.

Weitere interaktive Lehrmaterialien für das Grundstudium werden an der Colorado Technical University, USA in Zusammenarbeit mit der Firma EduScape, USA entwickelt und in Lerngruppen von je ca. 20 Studenten eingesetzt und erprobt. Bemerkenswert bei dieser Kooperation ist, dass die Lehrinhalte, d.h. die HTML-Seiten, auf einem Web-Server in den Vereinigten Staaten liegen, während alle Berechnungen mit *MuPAD* on-line über das Internet auf dem, an der Universität Paderborn installierten Computing Server ausgeführt werden. Dieser Versuch zeigt insbesondere auch die Perspektiven dieser Technologie für ein orts- und zeitungebundenes Lehren und Lernen auf.

Der MuPAD Computing Server

Die folgende Abbildung zeigt ein vereinfachtes Modell des MuPAD Computing Servers (MCS). Im Vordergrund stehen die interaktiven Lerneinheiten (links oben) in Form von HTML-Dokumenten, die im allgemeinen über einen üblichen Web-Server bereitgestellt und mit handelsüblichen Browsern gelesen bzw. bearbeitet werden. Eingebettet darin sind multimediale Eingabe- und Ausgabekomponenten, beispielsweise in Form von Java-Applets oder ActiveX-Controls, die der direkten Interaktion mit dem Lernenden dienen.

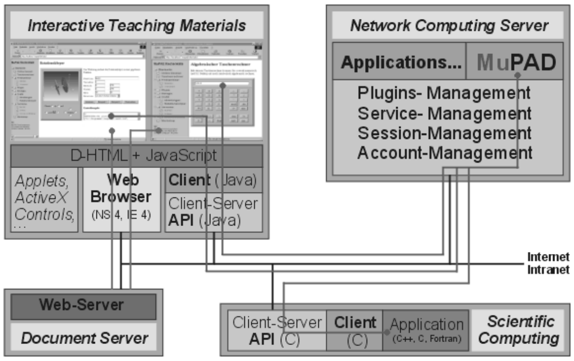


Abb. 2: Client-Server Modell des Computing Servers

Für algebraisch-symbolische und numerische Berechnungen wird ein `mupad`-Applet in die Web-Seite eingebunden (siehe Beispiel unten). Mittels JavaScript-Aufrufen der Art `mupad.eval('diff(1/ln(1/x),x))')` werden dabei alle Berechnungsanforderungen vom Browser an den Computing Server geleitet und das berechnete Ergebnis in Form einer Textausgabe zurückgeliefert. Alternativ stehen zur nicht-blockierenden Kommunikation Methoden wie `send`, `status`, `receive` sowie weitere Methoden zur Verfügung. Das folgende Beispiel zeigt eine minimalistische Anwendung dieser Technik. In ihm werden einfache Formularfelder für die Eingabe sowie für die Ausgabe verwendet. Über eine Schaltfläche (Button) führt der Benutzer die eingegebene Berechnung aus.

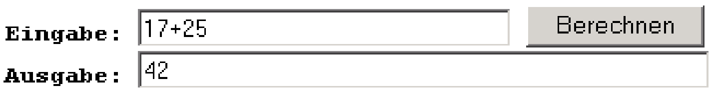


Abb. 3: Ein minimalistisches Beispiel zur Nutzung des MuPAD Computing Servers

Der zu Abbildung 3 gehörige HTML- und JavaScript-Quellcode sieht dabei wie folgt aus (auf weitere Stilelemente sowie Fehlerkontrollen wurde in diesem Beispiel aus Gründen der Übersichtlichkeit verzichtet):

```
<html>
...
  <!-- Einbinden des MCS-Kommunikationsapplets -->

  <applet name="mupad" code ="MupadApplet.class"><param  ...></applet>

  <!-- Ein-/Ausgabe und Berechnung mit JavaSccript steuern -->

  Eingabe: <input name="Ein">
  Ausgabe: <input name="Aus">
  <input value="Rechnen" type="Button"> onClick="Aus.value=mupad.eval(Ein.value)">
...
</html>
```

Mit dieser Technik lassen sich nun sehr flexibel eigene Lehreinheiten mit problemspezifischen Benutzungsschnittstellen gestalten, wobei zur Interaktion mit dem Benutzer beliebige Web-Browser- und JavaScript-fähige multimediale Komponenten eingesetzt und mittels JavaScript an den Computing Server gebunden werden können.

Die folgende Abbildung zeigt dazu als Beispiel u.a. ein vollständig in HTML realisiertes Taschenrechner-Interface zu *MuPAD* sowie ein Modul zum explorieren von Rotationskörpern. Als 3D-Visualisierungskomponente wurde dabei ein von SciFace Software entwickeltes ActiveX-Control eingesetzt.

Abb. 4: Interaktives Lehrmaterial mit dem *MuPAD* Computing Server

Abbildung 5 zeigt ein kleines Modul zum Differenzieren von Funktionen. Die Funktion und ihre Ableitung werden in diesem Fall über ein kleines 2D-Funktionsplotter-Applet dargestellt:

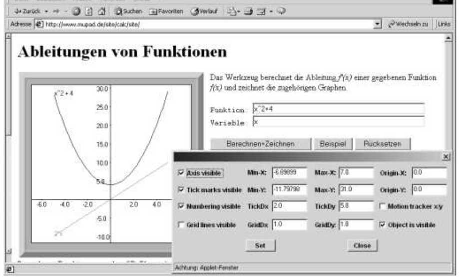


Abb. 5: Client-Server Modell des Computing Servers

Alternativ zu dem oben beschriebenen Kommunikationsapplet ohne eigenes Layout steht in der aktuellen Version des Computing Servers ein Terminal-Applet zur Verfügung (Abbildung 4, vorne unten). Hier ist ein direktes interaktives Arbeiten mit dem Computeralgebra-System *MuPAD* möglich. Im Gegensatz zu den zuvor beschriebenen Beispielen, muss der Anwender in diesem Fall mit der *MuPAD*-Eingabesyntax vertraut sein.

Technisches

Der Server ist derzeit für Windows NT4.0/2000, Linux und Solaris in einer ALPHA-Version realisiert. Als Klienten stehen eine C-Schnittstelle, beispielsweise für das symbolisch-algebraische Rechnen aus Fortran-Programmen heraus, sowie Java-Klassen und Applets für die Nutzung im World-Wide-Web zur Verfügung. Letztere sind Browser-unabhängig. Die Basisklassen können zudem zur Entwicklung eigener Applets sowie auch innerhalb von Servelets oder Java-Applikationen verwendet werden.

Der Computing Server ermöglicht dem Administrator die Netzwerkzugriffe zu konfigurieren, d.h. Rechner und Netzwerke freizuschalten bzw. gezielt zu sperren. Zudem unterstützt er eine einfache Benutzerverwaltung. So können auch in offenen und öffentlichen Netzen Kennwort-geschütze Zugriffe auf diesen Server realisiert werden. Ein Aspekt, der gerade im Bereich des E-Learning und Web-Based-Training an öffentlichen Lehreinrichtungen von Bedeutung ist.

Neben *MuPAD* können vom Administrator auch weitere Dienste, d.h. Anwendungsprogramme, über eine C++ Plugin Schnittstelle eingebunden werden. Dazu wird für den Computing Server eine abstrakte C++ Plugin Klasse zur Verfügung gestellt, von der eigene Services abgeleitet werden können. Dazu sind nur grundlegende C++ Programmierkenntnisse notwendig.

Ausblicke

Derzeit werden die Protokoll- und Konfigurationsformate auf XML umgestellt und das Kommunikationsapplet und das Terminal-Applet zu einem allgemeinen, vom Autor frei konfigurierbaren Applet zusammengefasst.

Schwerpunkt der Entwicklung in Paderborn bleibt aber der Server selbst. Dieser wird derzeit für den Großeinsatz ausgebaut. Erste Experimente lassen den Schluss zu, dass eine Anzahl von 100 gleichzeitig mit *MuPAD* arbeitenden Nutzern auf einem herkömmlichen PC in ereichbarer Nähe liegt.

Der *MuPAD* Computing Server kann als mathematische Komponente in interaktiven Lehreinheiten im World-Wide-Web eingesetzt werden. Genauso sind HTML-basierte Anwendungen auf CDROM denkbar. Eine Weiterentwicklung des JNI-Interfaces zum *MuPAD*-Kern unter Windows eröffnet die Möglichkeit der interaktiven Wahl der Berechnungsinstanz. Das heisst, der Anwender kann sich dann entscheiden, ob er auf dem *MuPAD* Computing Server im Internet oder (sofern vorhanden) mit einem lokalen, auf seinem Rechner installierten *MuPAD*-System rechnen möchte.

Für weitere Informationen wenden Sie bitte sich an:

Stichwort ”MCS”
Dr. Andreas Sorgatz
SciFace Software GmbH & Co. KG
Technologiepark 11
D-33100 Paderborn
Deutschland
Email: sorgatz@sciface.com
Fax: ++49 (0)5251 640799

Dynamische Mathematik mit GEONExT

M. Ehmann, C. Miller
Universität Bayreuth, GEONExT-Group
http://geonext.de
info@geonext.de

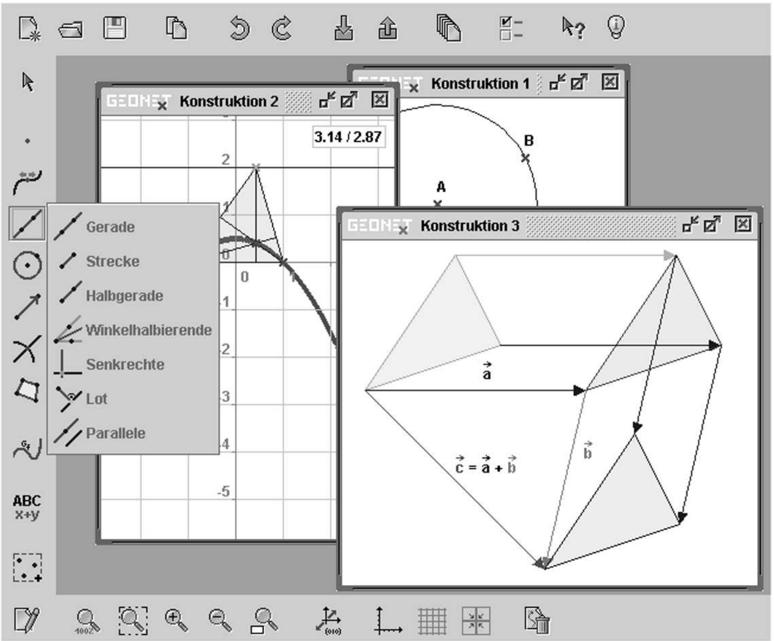
GEONExT eröffnet durch die Verknüpfung von dynamischer Geometrie und Analysis neue Möglichkeiten der Darstellung mathematischer Zusammenhänge

Bereits seit 1995 befasst sich der Lehrstuhl für Mathematik und ihre Didaktik der Universität Bayreuth mit der Entwicklung dynamischer Geometriesoftware und Konzepten für deren Einsatz im mathematisch-naturwissenschaftlichen Unterricht. Erste Erfahrungen sammelte das Team um Prof. Dr. Peter Baptist mit Geonet, das die Grundlage für die derzeitige Arbeit am Nachfolgeprojekt GEONExT bildet.

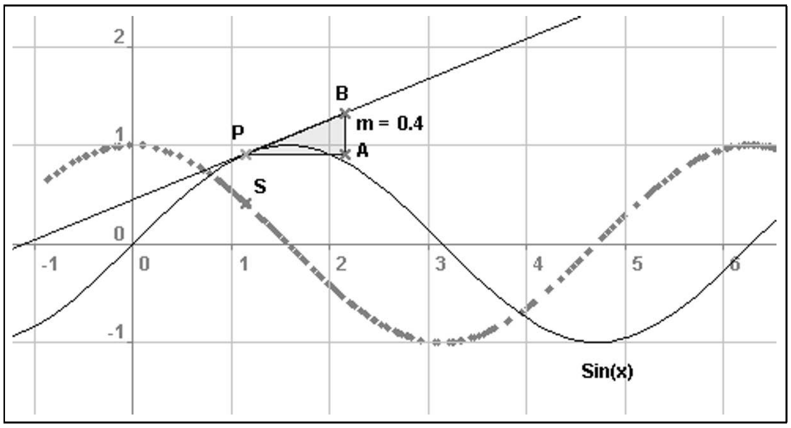
Die Implementierung in JAVA ermöglicht die Kombination von Texten, Grafiken und dynamischer Geometrie in HTML-Seiten. Durch das Multi Document Interface können mehrere Konstruktionen in einer GEONExT-Umgebung unabhängig voneinander dargestellt werden. Die vorgegebene Menüstruktur lässt sich den jeweiligen Erfordernissen anpassen. Dabei erlauben Untermenüs die Strukturierung der GEONExT-Funktionen.

Neben Punkten, Geraden und Kreisen sowie daraus zusammengesetzten Elementen, z.B. Streckenmittelpunkt, Winkelhalbierende und Umkreis, sind Vektoren, Polygone und mathematische Texte darstellbar. Über einen umfangreichen Dialog ist es möglich die Eigenschaften der Objekte zu verändern. So sind etwa Farbgebung, Transparenz, Spurmodus und Darstellungsform (Gerade – Halbgerade – Strecke) variierbar.

Zu Gruppen zusammengefasste Punkte lassen sich inklusive ihrer abhängigen Objekte verschieben und drehen. Eine einfache Anwendung stellt die abgebildete Parallelverschiebung dar.
GEONExT erlaubt neben dem eigenen XML-basierten Dateiformat das Speichern von HTML-Seiten, so dass die Konstruktionen direkt im Internet Browser geöffnet werden können. Damit bietet GEONExT die Möglichkeit komfortabel dynamische Arbeitsblätter zu erstellen.
Exzellente Druckausgaben gewährleistet der SVG-Export. Dabei werden Konstruktionen als Vektorgrafiken ausgegeben, die sich zur individuellen Weiterverarbeitung mit aktueller Grafiksoftware eignen.



Die Integration eines Computeralgebra-Systems schafft neue Visualierungsmöglichkeiten für den Geometrie- und Analysisunterricht. Die Basis für das GEONE$_X$T-Algebra-System (G.A.S.) bildet das freie Java-Applet Hartmath, das um eine Schnittstelle zum Umgang mit dynamischen geometrischen Objekten erweitert und dafür optimiert wurde. Dem Nutzer stehen dadurch drei neue Werkzeuge zur Verfügung. So lassen sich einerseits Funktionsgraphen in Abhängigkeit von geometrischen Elementen verändern, andererseits können Punkten für x- und y-Komponente beliebige Terme zugewiesen werden (sog. $(x; y)$-Punkte). Des Weiteren werden statische Texte durch die Einbindung von Berechnungen und Messungen dynamisch.



Ein konkretes Beispiel für den Einsatz der drei beschriebenen Optionen stellt nebenstehende GEONE$_X$T-Konstruktion zur Einführung der Ableitung der Sinus-Funktion dar.
Auf den Sinus-Graphen wird ein Gleiter P gesetzt, der entlang der Kurve beweglich ist. Mit Hilfe der $(x; y)$-Punkte A und B werden das Steigungsdreieck und die Tangente im Punkt P erzeugt. Durch Ziehen an P passt sich das Dreieck PAB seiner jeweiligen Position an.

Ausgehend von einer solchen vorgegebenen Konfiguration kann der Schüler die Steigung durch eine Berechnung als dynamischen Text, wie in der Abbildung ersichtlich, ausgeben lassen. Dabei muss er deren Abhängigkeit von P, A und B berücksichtigen. So werden erste Vermutungen bezüglich Wertebereich und Verlauf der Ableitungsfunktion ermöglicht. Diese lassen sich durch Zeichnen des $(x; y)$-Punktes S, der beim Bewegen eine Spur hinterlässt (Spurmodus) und dadurch den Graphen der Cosinus-Funktion skizziert, konkretisieren.

Derzeit wird das G.A.S. um einen Ein-/Ausgabe-Dialog für symbolisches und numerisches Rechnen erweitert. In der voraussichtlich Ende des Jahres erscheinenden Version 1.0 werden neben abschnittsweise definierten Funktionen auch Punkte, deren Lage sich aus logischen Operationen ergibt, integriert sein. Der geometrische Bereich von GEONE$_X$T wird unter anderem um Makrofunktionen zum Aufzeichnen von Konstruktionsschritten, dynamische Winkelelemente sowie umfangreiche Präsentationsmöglichkeiten (automatisch erzeugte Konstruktionsbeschreibung, Animation, u.a.) erweitert.

Der im Vergleich zu konventioneller Geometriesoftware vergrößerte Funktionsumfang von GEONE$_X$T fördert den mathematischen Modellbildungsprozess nicht nur in der Geometrie, sondern hilft auch im Bereich der Analysis Zusammenhänge und Gesetzmäßigkeiten dynamisch zu entdecken. Der Lehrstuhl für Mathematik und ihre Didaktik bietet zu ausgewählten Themen aus der Geometrie Sammlungen multimedialer Unterrichtseinheiten, sog. Lernumgebungen, an. In Zukunft werden auch Inhalte aus den Gebieten Funktionen und Infinitesimalrechnung der Mittel- und Oberstufenmathematik sowie der Mechanik und Optik in der Physik eine zentrale Rolle spielen.

Da es sich bei GEONE$_X$T um frei verfügbare Software handelt, kann es auf beliebig vielen Rechnern installiert und den Schülern mit nach Hause gegeben werden. Unter http://geonext.de steht die jeweils aktuelle Version kostenlos zum Download bereit.

Links

GEONE$_X$T-Website	http://geonext.de
Homepage des Lehrstuhls für Mathematik und ihre Didaktik	http://did.mat.uni-bayreuth.de
Multimediale Lernumgebungen	http://did.mat.uni-bayreuth.de/mmlu

Mathematik 1 × anders – Computerunterstütztes Lernen

Kai Gehrs, Andreas Sorgatz
MuPAD-Team, `acrowley@mupad.de`, `sorgatz@sciface.com`

Die neue Buchreihe "Mathematik 1 × anders – Materialien und Werkzeuge für computerunterstütztes Lernen" läuft mit Band 1 und 2 im August 2001 an.

Die Verwendung von Werkzeugen für computerunterstütztes Lernen ist kaum noch aus den Schulen wegzudenken. In Brandenburg schreiben die Richtlinien für den Mathematikunterricht bereits den Einsatz von Computeralgebra-Systemen vor. Im Spannungsfeld dieser Entwicklungen und um dem Wandel der Gestaltung des Mathematikunterrichts gerecht zu werden, ist im *MuPAD*-Team der Plan gereift, eine Buchreihe mit dem Titel "Mathematik 1 × anders – Materialien und Werkzeuge für computerunterstütztes Lernen" aufzulegen.

Die Reihe dient als Plattform zur Verbreitung von Materialien für den Einsatz von *MuPAD* im Unterrricht an weiterführenden Schulen aber auch für die Lehre an Universitäten sowie der Vorstellung von neuen Entwicklungen und der Veröffentlichung von Erfahrungsberichten und Evaluationen. Dabei steht der direkte Praxisbezug stets im Vordergund unseres Interesses.

Die Bände erscheinen im Eigenverlag bei SciFace Software, um geringere Kosten und ein Plus an Flexibilität zu ermöglichen: So können wir in dieser Reihe 30-seitige Broschüren bis 200-seitige Bücher sehr schnell und flexibel publizieren. Jeder Band ist jeweils mit einer ISBN ausgestattet, so dass er über den Buchhandel beziehbar ist. Zudem werden wir mit dieser Reihe auf Fachtagungen, Messen, Lehrer-Fortbildungen sowie im World-Wide-Web vertreten sein.

Band 1 der im August 2001 anlaufenden Reihe bietet einen einfachen und schnellen Einstieg in *MuPAD* anhand klassischer Themen aus dem Mathematikunterricht. Band 2, ebenfalls im August 2001, widmet sich Anwendungsbeispielen aus der Physik und ihren Lösungen mit *MuPAD*. Näheres hierzu finden Sie in den Kurzvorstellungen auf den folgenden Seiten.

Neben den eigenen Autoren, möchten wir aus dem *MuPAD*-Nutzerkreis weitere Autoren gewinnen, so beispielsweise interessierte Lehrerinnen und Lehrer, die bereit sind, ihr mit *MuPAD* entwickeltes Unterrichtsmaterial einem größerem Publikum zur Verfügung zu stellen.

Wenn Sie sich mit *MuPAD* in der Lehre beschäften und nun interessiert sind Ihr Material zu veröffentlichen, dann wenden Sie sich bitte an:

Stichwort: "Schule"
MuPAD, FB17-Mathematik
Universität Paderborn
Warburger Str. 100
D-33095 Paderborn
Email: schule@mupad.de
Fax: ++49 (0)5251 605534

MuPAD – Eine praktische Einführung

Kai Gehrs, Frank Postel

August 2001, SciFace Software GmbH & Co.KG, Paderborn, ISBN-3-933764-02-5, ca. 80 Seiten, `http://www.sciface.com/edu/`

MuPAD – Eine praktische Einführung ist der erste Band der neuen Reihe *Mathematik 1 × anders – Materialien und Werkzeuge für computerunterstütztes Lernen*. Diese Reihe wird von SciFace Software verlegt und unterstützt den Einsatz von *MuPAD* in der Sekundarstufe II an Gymnasien, Gesamtschulen und berufsbildenden Schulen sowie im Grundstudium an Universitäten und Fachhochschulen.

Das Buch bietet Neulingen einen schnellen, einfachen und vor allem praxisorientierten Zugang zu dem Computeralgebra-System *MuPAD*. Der Einsatz von *MuPAD* wird anhand einfacher Beispiele aus der Schulmathematik der Sekundarstufe II demonstriert und motiviert. Die Anwendungsbeispiele aus der Analysis, der Linearen Algebra sowie der Stochastik wurden bewußt elementar gehalten, so dass auch Schülern ein einfacher Einstieg in die Computeralgebra geboten wird. Dies eröffnet Lehrern die Möglichkeit, *MuPAD* in den Unterricht zu integrieren.

Der Aufbau des Buches wurde aufgrund von Erfahrungen aus Lehrerfortbildungen konzipiert und ermöglicht nach einer kurzen grundlegenden Einführung in das *MuPAD*-eigene Notebook-Konzept (Microsoft-Word^{TM} ähnliche interaktive Rechenblätter) sofort ein direktes Arbeiten mit *MuPAD*, ohne dass sich der Leser mit allzu technischen Details auseinandersetzen muss.

In allen Anwendungsbereichen werden zunächst die relevanten *MuPAD*-Funktionen eingeführt und Aufgaben vorgestellt, die dann mit MuPAD beispielhaft und nachvollziehbar gelöst werden. Jeder Abschnitt endet mit weiterführenden Tipps und Übungsaufgaben, durch deren Bearbeitung der Leser sein Verständnis vertiefen und den Umgang mit *MuPAD* selbstständig erlernen und üben kann.

Das letzte Kapitel des Buches geht auf grafische Visualisierungsmöglichkeiten in *MuPAD* ein. Dies ist ein wesentlicher Bestandteil des Computeralgebra-Systems, der die Darstellung mathematischer Sachverhalte vereinfacht und insbesondere auch Schüler darin unterstützt, die notwendige Anschauung und das überaus wichtige intuitive Verständnis von Mathematik weiterzuentwickeln.

Die im Buch gerechneten Aufgaben stehen im Web als *MuPAD*-Notebooks zum Download bereit. Da auch Test-Versionen von *MuPAD* kostenlos aus dem Internet heruntergeladen werden können, steht Ihrem persönlichen Einstieg in *MuPAD* nun nichts mehr entgegen.

Ein Streifzug durch die Physik mit MuPAD

Alessandro Dell'Aere

August 2001, SciFace Software GmbH & Co.KG,
Paderborn, ISBN-3-933764-03-3, ca. 34 Seiten,
`http://www.sciface.com/edu/`

Ein Streifzug durch die Physik mit MuPAD ist der zweite Band der neuen Reihe *Mathematik 1 × anders – Materialien und Werkzeuge für computerunterstütztes Lernen*. Diese Reihe wird von SciFace Software verlegt und unterstützt den Einsatz von *MuPAD* in der Sekundarstufe II an Gymnasien, Gesamtschulen und berufsbildenden Schulen sowie im Grundstudium an Universitäten und Fachhochschulen. Hier wird im wesentlichen auf den Wissensstand von Band I *MuPAD - Eine praktische Einführung* zurückgegriffen, so dass auch Personen mit wenig *MuPAD*-Erfahrung die Beispiele mühelos nachvollziehen können.

Die kleine Aufgabensammlung bietet Lehrern und Schülern eine gute Möglichkeit, mit Hilfe von *MuPAD* eine Brücke zwischen den Fächern Mathematik und Physik einerseits, und zwischen dem Schulunterricht und berufsorientierten Arbeitsmethoden andererseits, zu schlagen. Zu jedem, der im Buch angeschnittenen physikalischen Themenbereiche, wie Mechanik, Thermodynamik und Elektrizität werden zunächst Aufgaben vorgestellt und vollständig mit *MuPAD* gelöst. Am Ende eines jeweiligen Abschnitts werden weitere Aufgabenstellungen ohne Lösung aufgeführt. Sie dienen dem selbständigen Üben und der Vertiefung der erworbenen Kenntnisse sowie als Anregung für den weitergehenden Einsatz von *MuPAD*.

Dank MuPAD können neben den klassischen Unterrichtsinhalten auch problemlos Themen behandelt werden, die nicht in jedem Schulbuch zu finden sind. Hier ist zum Beispiel die Raumfahrt oder im Bereich Elektrizität der Schwingkreis und in dem Zusammenhang das numerische Lösen von Differentialgleichungen zu nennen. Fast spielerisch lassen sich die Aufgaben am Computer bearbeiten. Dabei ermöglicht es *MuPAD* sich auf Modellbildung und Lösungsstrategien zu konzentrieren.

Alles ist so gestaltet, dass auch ein Neuling einen schnellen Einstieg in das selbstständige Arbeiten mit MuPAD findet. Aber auch für Fortgeschrittene ist diese Aufgabensammlung eine interessante Bereicherung, denn sie bietet eine praxisorientierte Alternative zu den oft üblichen rein mathematischen Anwendungen von Computeralgebra-Systemen.

MuPAD Pro 2.0 für Windows im Griff

Andreas Sorgatz
MuPAD-Team, `andi@mupad.de`

Dieser Artikel stellt kurz einige Bedienungselemente der MuPAD Pro 2.0 Benutzungsschnittstelle vor, die den Umgang mit MuPAD erleichtern und insbesondere für Lehrer und Schüler eine sehr nützliche Unterstützung im Alltag bieten.

MuPAD-Notebooks

MuPAD-Notebooks sind interaktive Arbeitsblätter, die Eingaben, Ausgaben von mathematischen Formeln und Grafiken sowie auch formatierte Kommentartexte vereinen. Sie eignen sich nicht nur zum interaktiven Rechnen mit *MuPAD* sondern auch zur Dokumentation von Berechnungen.

Die Grafische Formalausgabe

Ab MuPAD Pro 2.0 steht unter Windows 95/98/ME,NT4.0/2000 eine grafische Formelausgabe zur Verfügung:[1]

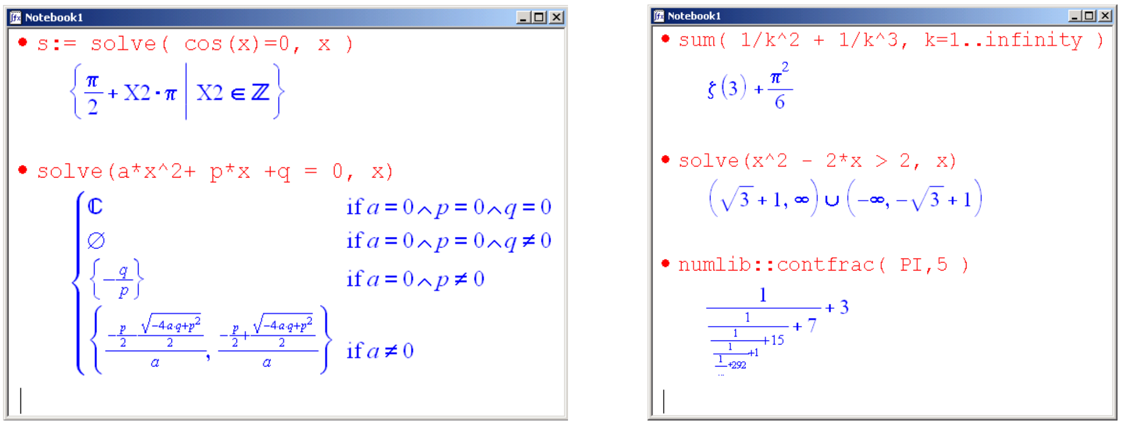


Abb. 1: Die grafische Formelausgabe in MuPAD Pro 2.0 für Windows

Sie kann über das Menü FORMAT:FORMELSATZ aktiviert oder deaktiviert werden. Unter dem Menüpunkt ANSICHT:OPTIONEN finden Sie die Registerkarte FORMELSATZ. Wenn Sie darauf klicken, gelangen Sie zu einem

[1] Anmerkung: Sie werden in einigen Fällen bemerken, dass *MuPAD* bei sehr großen Ausgaben auf die einfache Text-Darstellung, die sogenannte PRETTY PRINT-Darstellung der Formeln zurückschaltet. Dies ist oft die technisch-bedingt kompaktere Ausgabeform.

Konfigurationsmenü, in dem Sie die Form, Farbe und Skalierung der grafischen Formelausgabe in *MuPAD* auf Ihre persönlichen Vorlieben hin einstellen können.[2]

Die im *MuPAD*-Notebook ausgegebenen Formeln können Sie auch in Office-Anwendungen wie beispielsweise MS-Word[3] übernehmen und dort auch weiterhin frei skalieren. Klicken Sie dazu einfach in *MuPAD* mit der Maus auf die Formel und wählen Sie den Menüpunkt BEARBEITEN:KOPIEREN. Wechseln Sie in Ihr MS-Word Dokument und fügen Sie die Formel über den Menüpunkt BEARBEITEN:EINFÜGEN ein. Die Skalierung innerhalb von MS-Word erfolgt dann wie üblich.

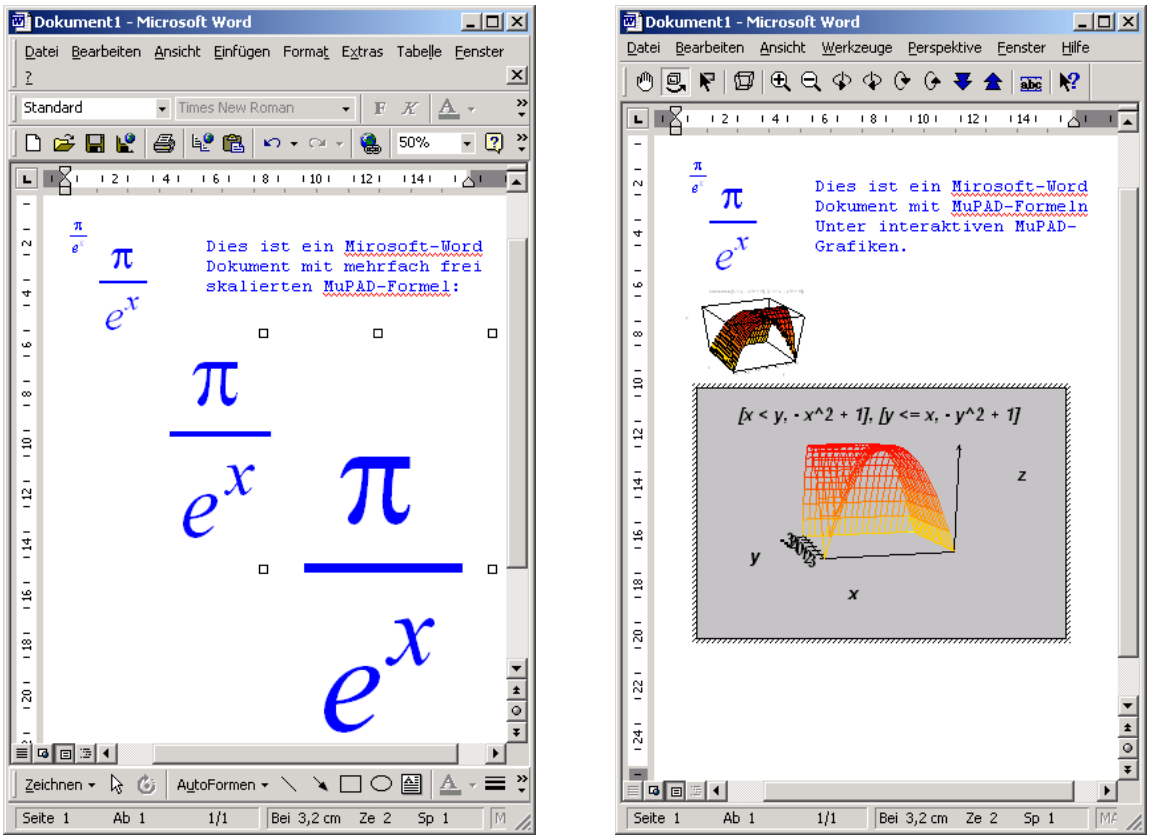


Abb. 2: MuPAD Formeln (links) und Grafiken (rechts) können in Office-Anwendungen übernommen und dort weiterhin frei skaliert und manipuliert werden.

Grafiken

Übrigens können Sie auch die in einem *MuPAD*-Notebook erstellten Grafiken in der gleichen Weise in Ihre Office-Anwendungen übernehmen und dort frei skalieren. Da *MuPAD* ein sogenannter OLE2-Container und -Server ist, können Sie Ihre *MuPAD*-Grafiken innerhalb Ihrer Office-Anwendung zudem auch noch weitergehender manipulieren: Mit einem Doppelklick auf die *MuPAD*-Grafik aktivieren Sie z.B. innerhalb von MS-Word-Dokumenten das *MuPAD*-Vcam Tool. Der Rahmen um die Grafik ändert sich und die MS-Word Menüzeile wird um die Vcam-Befehlszeile erweitert. Die Abbildung oben rechts zeigt diese Situation, wobei die *MuPAD*-Grafik bereits perspektisch im Raum gedreht, ihre Achsenform und Beschriftung, Hintergundfarbe, Text und Textformat geändert und statt einer flächigen Abbildung die Präsentation als Drahtmodell spezifiziert wurde.

[2]Die Einstellungen werden mit dem nästen Starten von *MuPAD* aktiv.
[3]Microsoft Word ist ein eingetragenes Warenzeichen von Microsoft.

Notebooks exportieren und importieren

Darüber hinaus bietet *MuPAD* die Möglichkeit *MuPAD*-Notebooks im RTF-Format zu exportieren. Wählen Sie dazu den Menüpunkt Datei:Exportieren…. Wählen Sie dort den Dateityp Rich Text Format (.rtf) aus und klicken Sie auf Speichern. Die so gespeicherte Datei können Sie direkt in MS-Word und anderen Office-Anwendungen einladen und dort weiterbearbeiten. Entsprechend können Sie RTF-Dokumente, die Sie beispielsweise in MS-Word erstellt und abgespeichert haben auch wieder in *MuPAD* als Notebook einlesen. Wählen Sie dazu den Menüpunkt Datei:Öffnen. Wählen Sie dort den Dateityp Rich Text Format (.rtf) aus und klicken Sie nach der Anwahl der zu ladenden Datei auf Öffnen.

Auf diese Weise können Sie interaktive Rechenblätter bequem als *MuPAD*-Notebooks erstellen, exportieren, importieren und auch publizieren. Mittels OLE2 und RTF bietet *MuPAD* Ihnen eine sehr komfortable und weitreichende Integration der Computeralgebra in Ihre Office-Anwendungen.

Alternativ zu den oben geschilderten Verfahren können Sie *MuPAD*-Grafiken in *MuPAD* auch direkt speichern und in den gängigen Grafikformaten exportieren. Aktivieren Sie zunächst das *MuPAD*-Vcam Tool mit einem Doppelklick auf die zu speichernde Grafik. Wählen Sie dann den Menüpunkt Bearbeiten:Speichere Grafik…. Wählen Sie dort den gewünschten Dateityp aus, beispielsweise WMF, JPEG, TIFF, PNG, EPS oder ein anderes Format, und klicken Sie nach Angabe eines Dateinamens auf Speichern. Dies ist übrigens genauso innerhalb Ihrer Office-Anwendungen möglich, wenn Sie die *MuPAD*-Grafik dort per *copy & paste* oder *drag & drop* eingefügt haben.

Eingabe-Assistenten

Die direkte Benutzung eine Computeralgebra-Systems ist nicht immer trivial und kein Mensch kann alle Funktionen eines derart komplexen Systems auswendig wissen. Um gerade Einsteigern den Umgang mit *MuPAD* zu erleichtern, stehen sogenannte Eingabe-Assistenten zur Verfügung, die im folgenden genauer beschrieben werden.

Die Kommando-Symbolleiste

Die Kommando-Symbolleiste gibt Ihnen per Mausklick den direkten Zugriff auf einige zentrale Funktionen in *MuPAD*, so beispielsweise zum Differenzieren und Integrieren, zur Grenzwertberechnung, zum Berechnen von exakten oder numerisch genäherten Lösungen, zum Zeichnen von zwei- und dreidimensionalen Funktiongraphen zur Eingabe von Matrizen verschiedener Dimensionen oder zur Auswahl griechischer Buchstaben.

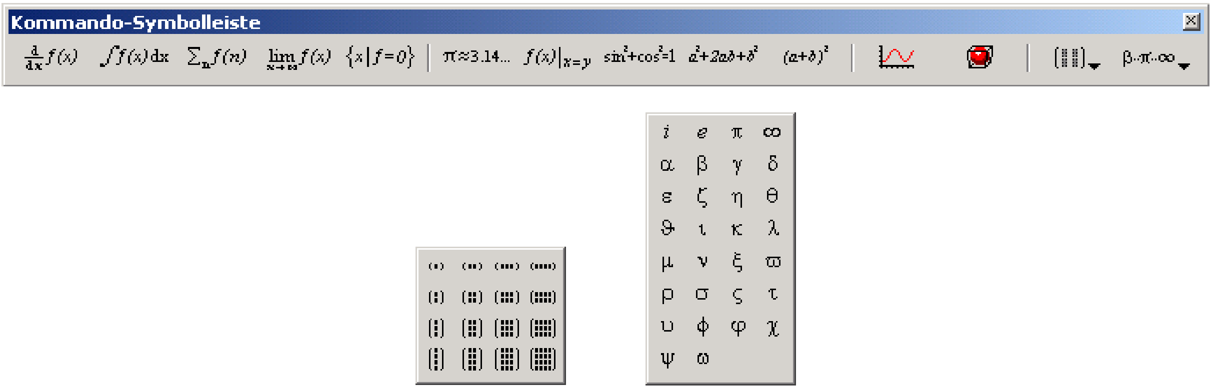


Abb. 3: Die Kommando-Symbolleiste stellt zentrale mathematische Funktionen sowie Paletten zur Matrizen- und Symboleingabe per Mausklick zur Verfügung.

Per Mausklick wird die Funktion in die aktuelle Eingaberegion innerhalb des *MuPAD*-Notebooks eingefügt. Fehlende Parameter werden dabei durch einen Platzhalter `'%?'` vertreten. Sie können dann einfach mittels TAB (rechts) und SHIFT-TAB (links) von Platzhalter zu Platzhalter springen und die gewünschten Werte eintragen. Die folgende Abbildung demonstriert dies am Beispiel der Darstellung des Funktionsgraphen $sin(x)$ im Bereich $-5\ldots5$ (automatische Vorauswahl) unter Verwendung des Symols 2D-GRAFIK:

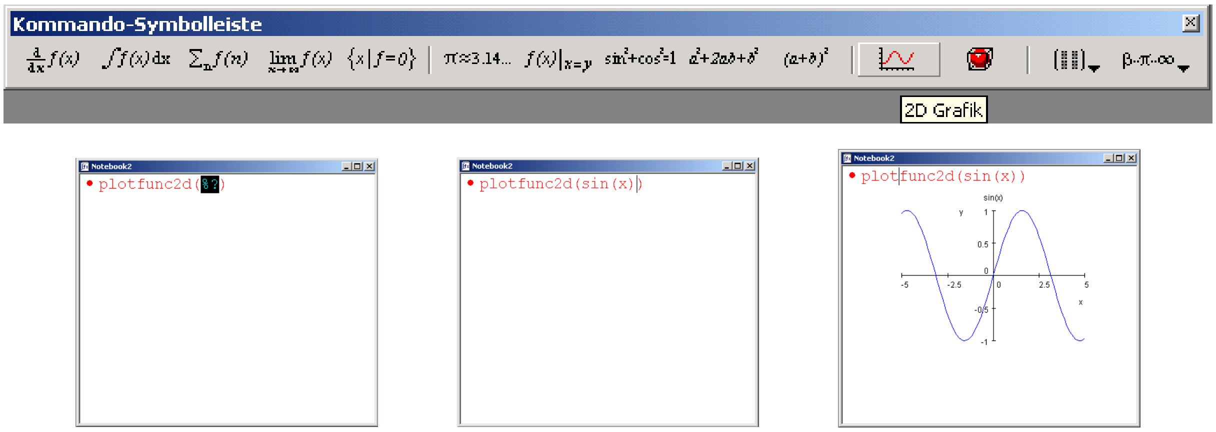


Abb. 4: So zeichnen Sie den Funktionsgraphen zu $sin(x)$ mit Hilfe der Kommando-Symbolleiste.

Sie können aber auch wie folgt vorgehen, um das gleiche Ziel zu erreichen: Geben Sie in einer Eingaberegion des *MuPAD*-Notebooks '$sin(x)$' ein und selektieren Sie diesen Funktionsaufruf mit der Maus. Klicken Sie nun auf das Symbol für 2D-Grafiken in der Kommando-Symbolleiste. *MuPAD* fügt das Kommando `plotfunc2d` in die Eingaberegion ein und verwendet für den ersten Parameter der Funktion automatisch Ihre Selektion.

Das Extras-Menü

Eine weitere geordnete Auswahl von wichtigen *MuPAD*-Funktionen des täglichen (mathematischen) Lebens werden über das EXTRAS-Menü bereitgestellt. Auch hier kann die Funktion mit einem Mausklick in das Notebook eingefügt werden und die weitere Bedienung erfolgt analog zur Verwendung der Kommando-Symbolleiste. Die folgende Abbildung zeigt das EXTRAS-Menü sowie das geöffnete Untermenü zum Lösen von Gleichungen etc.:

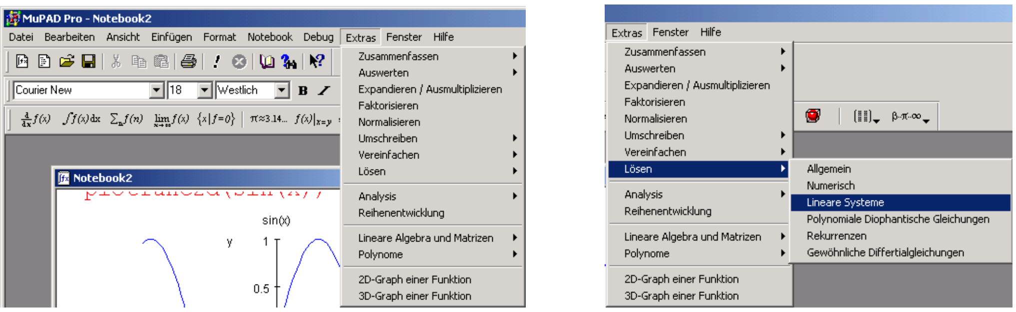


Abb. 5: Das konfigurierbare EXTRAS-Menü in MuPAD Pro zur vereinfachten Befehlseingabe per Mausklick.

Eine weitere Besonderheit des EXTRAS-Menüs ist die Tatsache, dass es vom Anwender ausgetauscht und/oder verändert werden kann. Dies ermöglicht es Lehrern ihren Schülern ein speziell auf die aktuelle Unterrichtseinheit abgestimmtes Menü bereitzustellen. Dazu finden Sie unter dem Menüpunkt ANSICHT:OPTIONEN die Registerkarte MUPAD. Wenn Sie darauf klicken, gelangen Sie zu einem Konfigurationsmenü, in dem Sie die vorkonfigurierte

Datei `extras.xml` gegen Ihre persönliche Datei austauschen können. Die folgende Abbildung zeigt den Anfang der Datei `extras.xml`. Eigene Menüs können hier sehr einfach in einem XML[4]-Format spezifiziert werden.

```
<?xml version="1.0" encoding="iso-8859-1"?>
<extras>
    <menu id="sub eval">
        <keywd>Auswerten</keywd>
        <cmd type="template">
             <keywd>Numerisch</keywd>
             <info>Schablone der Funktion float einfügen.</info>
             <exec>float(#)</exec>
        </cmd>
        <cmd type="template">
             <keywd>Logisch</keywd>
             <info>Schablone der Funktion bool einfügen.</info>
             <exec>bool(\#)</exec>
        </cmd>
    </menu>
    ...
```

Abb. 6: Ein Auszug aus der XML-Konfigurationdatei `extras.xml` des EXTRAS-Menüs.

Folgende Stukturelemente (sogenannte XML-Tags) stehen zum Aufbau des EXTRAS-Menüs zur Verfügung. Nach einem kurzen Blick in die mit *MuPAD* ausgelieferte Menüdatei können Sie auch schon gleich Ihre eigenen Vorstellungen verwirklichen.

Tag / Symbol	**Erläuterung**
`xmenu`	das Hauptmenü, dies darf nur einmal auftreten
`menu`	ein Untermenü, die Anzahl ist nicht beschränkt
`keywd`	der sichtbare Name eines Untermenüs
`ref`	ein Verweis auf ein Untermenü
`cmd`	leitet einen Befehlseintrag in einem Menü ein
`info`	für einen Informationstext zu einem Befehlseintrag
`exec`	der zu einem Befehlseintrag gehörige MuPAD-Code
`#`	ein Platzhalter für einen Funktionsparameter
`sep`	ein Separator, in Form einer horizontalen Trennlinie

Abb. 7: Wichtige `XML−Tags` zur Definition von benutzereigenen EXTRAS-Menüs.

Befehlsergänzung in der Eingabezeile

Sie sind das Tippen langer Funktionsnamen leid oder kennen nur den Anfang eines *MuPAD*-Funktionsnamens? Kein Problem. *MuPAD* unterstützt Sie auch dabei mit der neuen Befehlsergänzung in der Eingabezeile. Tippen Sie einfach die ersten Buchstaben eines Befehls ein und drücken Sie dann die Taste STRG in Kombination mit der Leertaste. Gibt es genau einen eindeutigen Funktionsnamen, der mit dieser eingegebenen Zeichenfolge beginnt, so wird der Name automatisch vervollständigt. Beispielsweise wird `ga` in diesem Fall automatisch zu `gamma` vervollständigt. Gibt es keine eindeutige Erweiterung, so erscheint über der Eingaberegion ein Fenster, in dem alle Bezeichner aufgelistet sind, auf die die eingegebene Zeichenkette ergänzt werden kann. Die folgende Abbildung (links) zeigt dieses für die Eingabe des Buchstabens `g`:

[4] XML ist eine sehr flexible Stukturbeschreibungssprache, die viel im Zusammenhang mit World Wide Web-Anwendungen eingesetzt wird und in Zukunft wahrscheinlich HTML in weiten Teilen ersetzen wird.

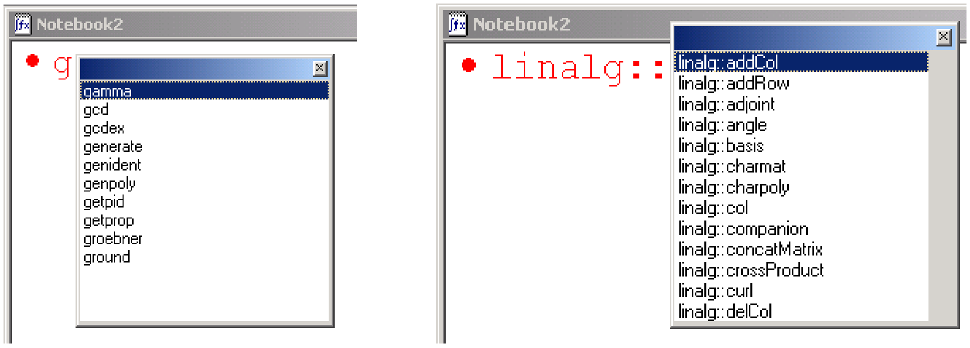


Abb. 8: Das Befehlsergänzungsmenü erscheint auf den Tastendruck STRG-Leertaste:
links für die Eingabe `g`, rechts für die Eingabe `linalg::`-

Die Befehlsergänzung berücksichtigt auch die benutzerdefinierten Bezeichner und Funktionen. Probieren Sie es aus, indem Sie beispielsweise `gandalf:=7` ausführen und dann eine Befehlsergänzung für die eingegebene Zeichenkette `ga` anfordern. Die Ergänzung ist nun nicht mehr eindeutig und Sie erhalten daher ein Auswahlmenü, in dem sowohl die Funktion `gamma` als auch die Variable `gandalf` aufgefürt wird.

Die Technik der Befehlsergänzung kann in gleicher Weise zur Auswahl von Funktionen aus einer *MuPAD*-Bibliothek verwendet werden. Geben Sie dazu beispielsweise `linalg::` ein (die Doppelpunkte '`::`' sind hierbei wichtig) und drücken Sie erneut die Taste STRG in Kombination mit der Leertaste. Sie erhalten die Liste aller *MuPAD*-Funktionen aus dem Paket zur Linearen Algebra (Abbildung 8, rechts).

Hilfe-Assistenten und Online-Dokumentation

Hilfe zur Benutzungsoberfläche

Weitere Unterstützung zur Benutzung der Oberflächenelemente von *MuPAD* erhalten Sie über die Windows Online-Hilfe. Wählen Sie den Menüpunkt HILFE:HILFETHEMEN, um einen Überblick über alle Aspekte der *MuPAD*-Benutzungsschnittstelle zu erhalten. Wenn Sie Auskunft zu einem speziellen Menüpunkt oder Symbol haben möchten, dann klicken Sie zunächst auf das Symbol[5] und danach auf das Bedienelement zu dem Sie weitere Erläuterungen möchten. Klicken Sie beispielsweise erneut auf das Symbol, so wird das folgende Fenster mit einer kurzen Beschreibung dieser Schaltfläche geöffnet:

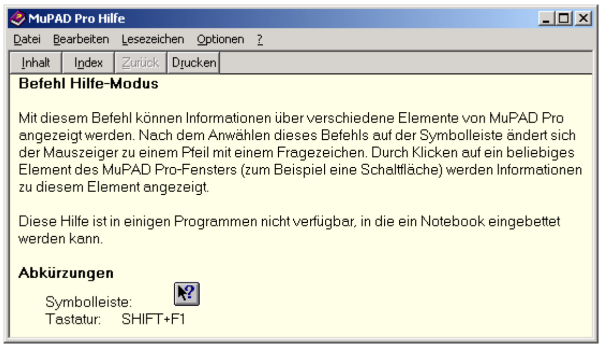


Abb. 9: Die Windows Online-Hilfe bietet Kurzbeschreibungen zur Benutzungsschnittstelle.

[5] Sie werden bemerken, dass sich die Form des Mauszeigers daraufhin ändert und der Pfeil nun von einem Fragezeichen begleitet wird.

Der Hilfe-Assistent

Der *MuPAD* Hilfe-Assistent hilft Ihnen bei der Navigation durch die Online-Dokumentation zur mathematischen Funktionalität von *MuPAD*. Mit ihm finden Sie schnell alle benötigten Informationen zu Funktionen, Bibliotheken und zur PASCAL-ähnlichen Programmiersprache von *MuPAD*. Er bietet dazu u.a. einen lexikalisch sortierten sowie einen themenbezogenen Zugriff auf die Hilfeseiten und unterstützt eine Kommando- und Schlüsselwortsuche. Sie starten den Hilfe-Assistenten mit der Funktionstaste F2. Alternativ können Sie in *MuPAD* die Anweisung `?`*zeichenkette* eingeben und ausführen, wobei *zeichenkette* ein Bezeichner oder Begriff ist, zu dem Sie Hilfe benötigen. Der Hilfe-Assistent präsentiert daraufhin eine Liste von Verweisen zu dem gesuchten Bezeichner:

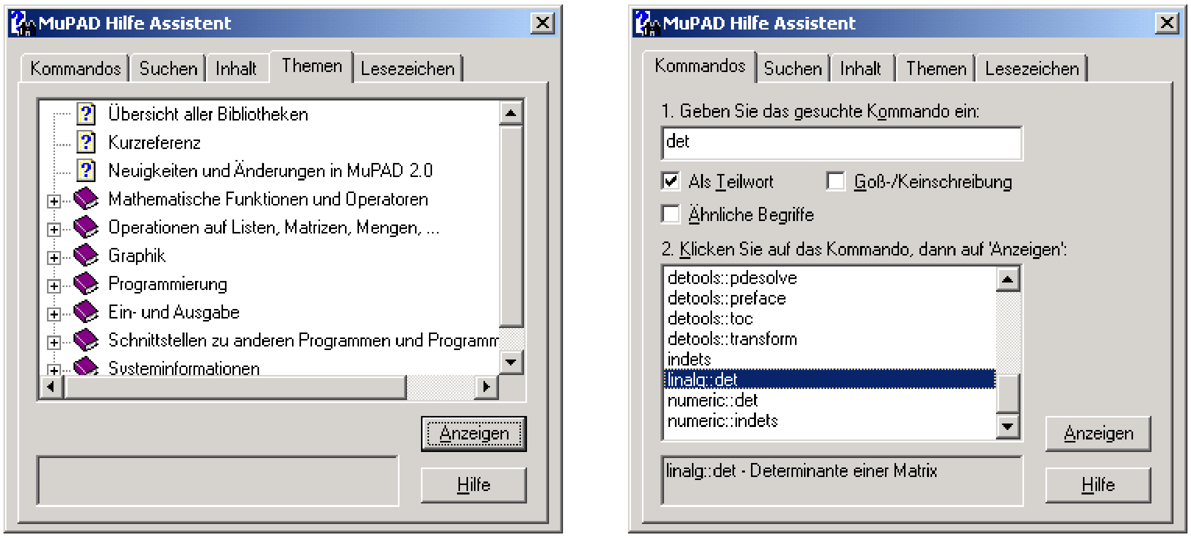


Abb. 10: Der Hilfeassistent – links: themenbezogener Zugriff, rechts: lexikalisch sortierte Funktionsliste.

Wählen Sie einen Verweis aus und klicken Sie auf ANZEIGEN, um die entsprechende Hilfe-Seite der *MuPAD* Online-Dokumentation zu lesen. Daraufhin wird der *MuPAD*-eigene Hilfe-Browser gestartet, der neben der Präsentation der Textinformation weitere Navigationshilfen bietet und benutzerdefinierte Querverweise und Notizen unterstützt.

Die Online-Dokumentation

Die *MuPAD* Online-Dokumentation ist als Hypertext-Dokumentation realisiert und umfasst weit über 2000 Hilfeseiten mit einführenden Informationen sowie technischen Details und vielen *click & go* Beispielen zu den einzelnen *MuPAD*-Funktionen. Sie können im Hilfe-Browser direkt durch den Text navigieren: über den Index, über Lesezeichen oder Querverweise, auch seitenweise und per Volltextsuche, oder Sie greifen auf den zuvor beschriebenen Hilfe-Assistenten zurück.

Die folgende Abbildung (links) zeigt den Hilfe-Browser mit der Hilfe-Seite zur Funktion `plotfunc2d`. Neben den farbig unterlegten vordefinierten Querverweisen sehen Sie einen Kasten um den Begiff `Graphen`. Hierbei handelt es sich um einen benutzerdefinierten Querverweis[6], der hier zusätzlich in die *MuPAD*-Dokumentation eingefügt wurde. Desweiteren enthält die Seite eine geöffnete und eine geschlossenen benutzerdefinierte Notiz, die ebenfalls von Benutzer eingefügt wurden. Ihre benutzerdefinierte Verweise und Notizen werden automatisch gespeichert und stehen Ihnen damit auch später jederzeit zur Verfügung.

[6] Der MuPAD Hilfe-Browser unterstützt zudem auch benutzerdefinierte Lesezeichen.

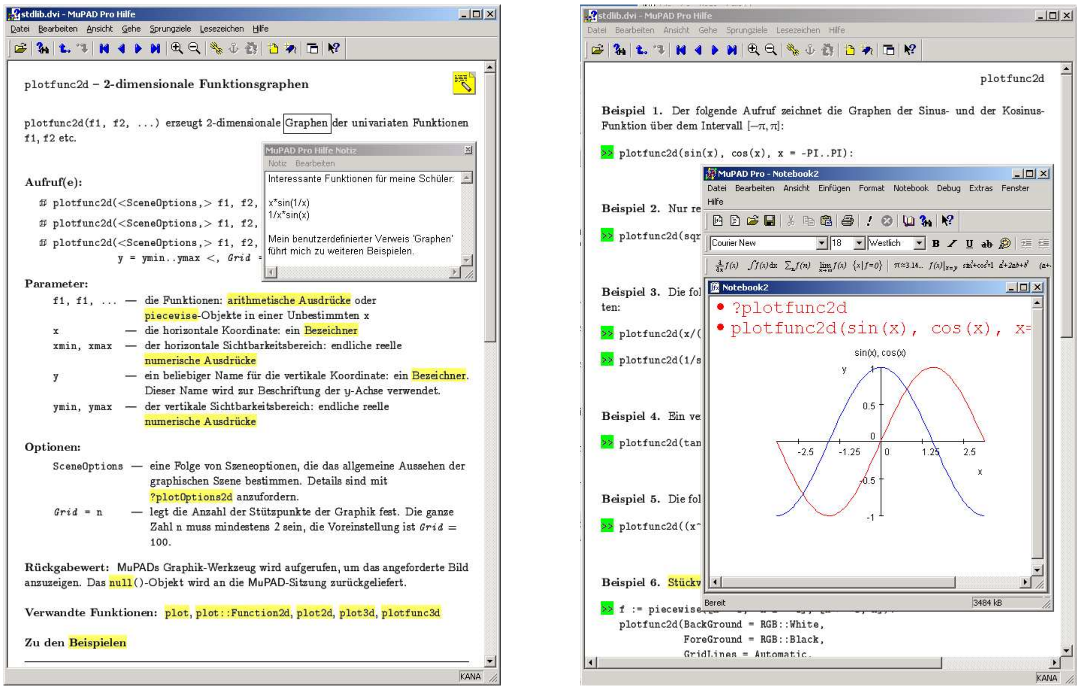


Abb. 11: Die MuPAD Online Hilfe – links: benutzerdefinierte Querverweise
und Notizen, rechts: Anwendungsbeispiele einfach per Mausklick.

Die folgende Abbildung zeigt die Schaltflächen des Hilfe-Browsers, mit denen Sie ganz einfach Ihre eigenen Querverweise, Lesezeichen und Anmerkungen in die MuPAD-Dokumentation einfügen können:

Abb. 12: Benutzerdefinierte Querverweise, Notizen und Lesezeichen in der MuPAD-Dokumentation.

Alle Beispiele der *MuPAD*-Dokumentation können einfach per Mausklick (Doppelklick) sowie auch per *drag & drop* in ein *MuPAD*-Notebook kopiert und dort verändert und ausgeführt werden. Die vielen Beispiele bieten dem Benutzer einen schnellen und explorativen Zugang zur mathematischen Funktionalität von *MuPAD*. Abbildung 11 (rechts) zeigt ein interaktives Beispiel, das in das *MuPAD*-Notebooks kopiert und dort ausgeführt wurde.

Weitere Informationen zu diesen und anderen Eigenschaften sowie zu der Konfigurierbarkeit des *MuPAD* Hilfe-Browsers finden Sie in dessen Online-Hilfe.

Startup-Konfiguration für Notebooks

Das Startup-Notebook

Standardmäßig öffnet *MuPAD* beim Start automatisch ein Notebook mit einführenden Informationen zur Nutzung von *MuPAD*. Das Startup-Notebook wird über den Menüpunkt ANSICHT:OPTIONEN in der Registerkarte MUPAD spezifiziert. Löschen Sie diesen Eintrag, wenn Sie kein vordefiniertes Startup-Notebook benötigen. Alternativ können Sie dort auch ein speziell auf Ihren Kurs oder Ihre Unterrichtseinheit abgestimmtes Notebook eintragen, beispielsweise wie in der folgenden Abbildung:

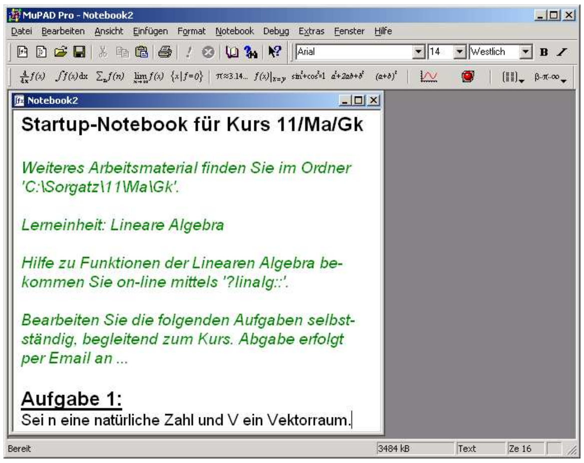


Abb. 13: Eigene Startup-Notebooks, die beim Start von MuPAD Pro automatisch geöffnet werden.

Das Startup-Befehle für Notebooks

Häufig werden Lerneinheiten als dokumentierte Rechenblätter in Form von *MuPAD*-Notebooks Lernenden zur Verfügung gestellt werden. Dabei sollen die Lernenden zunächst von allen technischen Details, wie dem Laden von Hilfsroutinen, die der Autor zur Verfügung stellt und dem Initialisieren von Startwerten, vollkommen entbunden werden. Mehr noch, entsprechende Definitionen sollen automatisch beim Laden des Notebooks erfolgen.

MuPAD unterstützt ein derartiges Vorgehen durch die Möglichkeit der Zuweisung von Eigenschaften an ein Notebook. Öffnen Sie dazu zunächst das entsprechende Notebook in *MuPAD* und wählen Sie dann den Menüpunkt DATEI:EIGENSCHAFTEN (Abbildung 14). Sie können dort im oberen Feld eine Datei angeben, die beim Öffnen des Notebooks automatisch geladen und ausgeführt wird. Im unteren Feld können Sie direkt *MuPAD*-Befehle eingeben, die in dem Fall ausgeführt werden soll. Die Methoden schließen sich nicht aus, sondern ergänzen sich. Sollen *MuPAD*-Notebooks ins Internet gestellt werden, so bietet sich die zweite Methode an, da Referenzen auf externe Dateien hier vermieden werden sollten.

Abb. 14: Startup-Befehle für MuPAD-Notebooks können frei konfiguriert werden.
Ihre Ausführung erfolgt automatisch beim Laden dieses Notebooks.

Weitere Informationen

Weitere Informationen zu *MuPAD* – beispielsweise zum Quellcode-Debugger (Abbildung 15), mit dem Sie *MuPAD*-Funktionen schrittweise ausführen und analysieren können oder zum Grafiktool, entnehmen Sie am besten der *MuPAD* Online-Hilfe.

MuPAD ist für verschiedene Betriebssysteme als 30-Tage[7] Evaluierungsversion auf dem Web-Server der Firma SciFace Software unter `http://www.sciface.com/download.shtml` verfügbar.

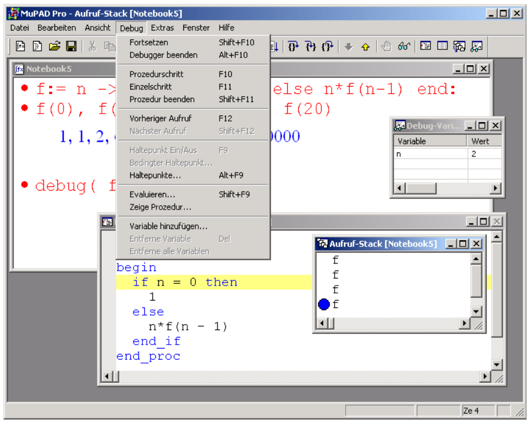


Abb. 15: Der MuPAD Quellcode-Debugger: schrittweises Ausführen und Analysieren von MuPAD-Funktionen.

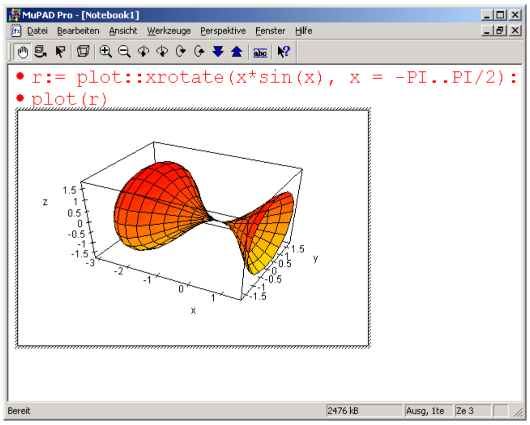


Abb. 16: Das MuPAD Grafik-Tool: 2D/3D-Szenen visualisieren und interaktiv bearbeiten.

Das "Schule"-Team der MuPAD-Gruppe steht Ihnen bei Fragen zum Einsatz von *MuPAD* in der Lehre gern mit Rat und Tat zur Seite. Bitte wenden Sie sich dazu per Email an `schule@mupad.de`:

Stichwort: "Schule"
SciFace Software GmbH & Co. KG
Technologiepark 11
D-33100 Paderborn
Email: schule@mupad.de
Fax: ++49 (0)5251 640799

[7] Längerfristige Evaluierungslizenzen sind bei Bedarf auf direkte Anfrage bei `schule@mupad.de` erhältlich.

Wieviele Rinder hat der Sonnengott?

Friedrich Schwarz
Universität Paderborn, `fritz@uni-paderborn.de`

An dem berühmten Rinderproblem von Archimedes kann man zeigen, wie gut MuPAD dank der Multipräzisionsarithmetik mit wirklich großen ganzen Zahlen rechnen kann.[1]

Im Jahr 1773 veröffentlichte G. E. Lessing, der damals Bibliothekar an der Herzoglich-Braunschweigischen Bibliothek in Wolfenbüttel war, einen griechischen Text aus einem Codex, den er in seiner Bibliothek gefunden hatte (vgl. [1]). Dieser Text enthält eine Aufgabe, die, wie in ihrem Vorspann vermerkt ist, einst Archimedes an Eratosthenes und die anderen Mathematiker in Alexandria geschickt hat. Eine philologisch zuverlässige Version des griechischen Texts und eine Übersetzung ins Lateinische findet man im zweiten Band der von I. L. Heiberg besorgten Ausgabe der Werke von Archimedes (vgl. [2]).

Die Aufgabe

Die Rinderherde des Gottes Helios besteht aus N_1 weißen, N_2 schwarzen, N_3 gescheckten und N_4 braunen Stieren und aus N_5 weißen, N_6 schwarzen, N_7 gescheckten und N_8 braunen Kühen.

(a) Es gilt

$$\left.\begin{aligned} N_1 &= \left(\frac{1}{2}+\frac{1}{3}\right) N_2 + N_4, \\ N_2 &= \left(\frac{1}{4}+\frac{1}{5}\right) N_3 + N_4, \\ N_3 &= \left(\frac{1}{6}+\frac{1}{7}\right) N_1 + N_4 \end{aligned}\right\} \quad \text{und} \quad \left\{\begin{aligned} N_5 &= \left(\frac{1}{3}+\frac{1}{4}\right)(N_2+N_6), \\ N_6 &= \left(\frac{1}{4}+\frac{1}{5}\right)(N_3+N_7), \\ N_7 &= \left(\frac{1}{5}+\frac{1}{6}\right)(N_4+N_8), \\ N_8 &= \left(\frac{1}{6}+\frac{1}{7}\right)(N_1+N_5). \end{aligned}\right.$$

(b) Die weißen und die schwarzen Stiere zusammen können sich in Form eines Quadrats aufstellen, d.h. $N_1 + N_2$ ist eine Quadratzahl.

(c) Die gescheckten und die braunen Stiere zusammen können sich in Form eines Dreiecks aufstellen, d.h. es gibt eine natürliche Zahl m mit

$$N_3 + N_4 \;=\; \frac{1}{2}\,m(m+1).$$

Zu ermitteln ist die Anzahl der Rinder in der Herde. Es dürfte reichen, die kleinste Lösung zu finden – falls es überhaupt Lösungen gibt. Gesucht ist also ein 8-tupel $(N_1, N_2, \ldots, N_8) \in \mathbb{N}^8$, das die Bedingungen (a), (b) und (c) erfüllt und das aus möglichst kleinen natürlichen Zahlen besteht.

[1] Erstabdruck Band 6, Ausgabe 1, November 1996.

Der Lösung erster Teil

(A) Man löst zuerst das lineare Gleichungssystem aus (a), was für *MuPAD* keine Schwierigkeit bedeutet. Man erhält: Mit den Zahlen

$$\begin{array}{llllllll} a_1 & := & 103\,66482, & a_2 & := & 74\,60514, & a_3 & := & 73\,58060, & a_4 & := & 41\,49387, \\ a_5 & := & 72\,06360, & a_6 & := & 48\,93246, & a_7 & := & 35\,15820, & a_8 & := & 54\,39213 \end{array}$$

ist die Menge der 8-tupel $(N_1, N_2, \ldots, N_8) \in \mathbb{N}^8$, die den sieben Bedingungen in (a) genügen,

$$\big\{(a_1 v,\ a_2 v,\ a_3 v,\ a_4 v,\ a_5 v,\ a_6 v,\ a_7 v,\ a_8 v) \mid v \in \mathbb{N}\big\}.$$

(B) Für $v \in \mathbb{N}$ erfüllt $(a_1 v, a_2 v, a_3 v, a_4 v, a_5 v, a_6 v, a_7 v, a_8 v)$ die Bedingung in (b), genau wenn $(a_1 + a_2) \cdot v$ eine Quadratzahl ist. Es ist $a_1 + a_2 = 178\,26996 = 2^2 \cdot 3 \cdot 11 \cdot 29 \cdot 4657$, und daher ist $(a_1 + a_2) \cdot v$ eine Quadratzahl, genau wenn mit $b := 3 \cdot 11 \cdot 29 \cdot 4657 = 44\,56749$ gilt: Es gibt ein $w \in \mathbb{N}$ mit $v = b \cdot w^2$.

(C) Es ist $a_3 + a_4 = 115\,07447 = 7 \cdot 353 \cdot 4657$. Für $w \in \mathbb{N}$ erfüllt das 8-tupel

$$(a_1 bw^2,\ a_2 bw^2,\ a_3 bw^2,\ a_4 bw^2,\ a_5 bw^2,\ a_6 bw^2,\ a_7 bw^2,\ a_8 bw^2)$$

genau dann die Bedingung aus (c), wenn es eine natürliche Zahl m mit

$$\frac{1}{2}\, m(m+1) \;=\; a_3 bw^2 + a_4 bw^2 \;=\; 3 \cdot 7 \cdot 11 \cdot 29 \cdot 353 \cdot 4657^2 \cdot w^2 \;=\; 5128\,58029\,09803 \cdot w^2$$

gibt. Es ist also die Lösung $(m, w) \in \mathbb{N} \times \mathbb{N}$ der Diophantischen Gleichung

$$\frac{1}{2}\, M(M+1) \;=\; 5128\,58029\,09803 \cdot W^2$$

zu finden, in der die Zahlen m und w möglichst klein sind – falls es überhaupt Lösungen aus natürlichen Zahlen gibt. Dazu ermittelt man, falls vorhanden, die Lösung $(x, w) \in \mathbb{N} \times \mathbb{N}$ der Gleichung

$$X^2 - 8 \cdot 5128\,58029\,09803 \cdot W^2 \;=\; 1, \tag{1}$$

in der x und w möglichst klein sind, sieht, daß darin x ungerade und größer als 1 ist, und setzt

$$m \;:=\; \frac{1}{2}\,(x - 1).$$

Die Pellsche Gleichung

Für Diophantische Gleichungen vom Typ der Gleichung (1) gibt es eine vollständige Theorie, die im wesentlichen auf Fermat, Euler und Lagrange zurückgeht, jedenfalls was die europäische Mathematikgeschichte betrifft. Euler nannte diese Gleichungen nach dem englischen Mathematiker John Pell, der sich wohl nicht weiter mit ihnen befaßt hat.

Es sei $d \in \mathbb{N}$ keine Quadratzahl. Die Pellsche Gleichung

$$X^2 - d \cdot Y^2 \;=\; 1 \tag{PELL}$$

besitzt unendlich viele Lösungen $(x, y) \in \mathbb{N} \times \mathbb{N}$. Es genügt, die Lösung $(x, y) \in \mathbb{N} \times \mathbb{N}$ von (PELL) mit minimalem y und daher auch minimalem x, die sogenannte Fundamentallösung von (PELL), zu finden. Definiert man nämlich für jedes $n \in \mathbb{N}$ die natürliche Zahlen x_n und y_n durch

$$x_n + y_n \sqrt{d} \;=\; \big(x + y\sqrt{d}\big)^n,$$

so ist $\{(x_n, y_n) \mid n \in \mathbb{N}\}$ die Menge aller Lösungen $(x, y) \in \mathbb{N} \times \mathbb{N}$ von (PELL).

Einen brauchbaren Algorithmus zur Berechnung der Fundamentallösung von (PELL) bekommt man mit Hilfe der Theorie der Kettenbrüche. Man berechnet den regelmäßigen Kettenbruch

$$\sqrt{d} \;=\; [\,a_0, a_1, a_2, \ldots, a_k, a_{k+1}, \ldots\,] \;=\; a_0 + \cfrac{1}{a_1 + \cfrac{1}{a_2 + \cfrac{1}{a_3 + \cdots}}}$$

für $\sqrt{d}$. Dieser Kettenbruch ist periodisch mit einer Vorperiode der Länge 1, es gilt also, wenn l die Länge einer minimalen Periode ist,

$$\sqrt{d} \;=\; [\,a_0, \overline{a_1, a_2, \ldots, a_{l-1}, a_l}\,],$$

und die Periode $(a_1, a_2, \ldots, a_{l-1}, a_l)$ besitzt die folgende Symmetrie-Eigenschaft: Es ist $a_l = 2a_0$, und für jedes $i \in \{1, 2, \ldots, l-1\}$ ist $a_i = a_{l-i}$. Ein vergleichsweise schnelles Verfahren zur Berechnung der Periode des Kettenbruchs für $\sqrt{d}$, das diese Symmetrie-Eigenschaft ausnützt, wird in Perrons klassischer Darstellung der Theorie der Kettenbrüche beschrieben (vgl. [3]); in [4] sind die Verfahren zur Berechnung periodischer Kettenbrüche so beschrieben, daß sie leicht in *MuPAD*-Programme übersetzt werden können. Die Fundamentallösung (x, y) von (PELL) erhält man dann so (vgl. dazu zum Beispiel [4]):

Man berechnet den $(l-1)$-ten Näherungsbruch r_{l-1}/s_{l-1} des Kettenbruchs für $\sqrt{d}$ und setzt

$$\begin{array}{lllllll} x & := & r_{l-1} & \text{und} & y & := & s_{l-1}, \quad \text{falls } l \text{ gerade ist,} \\ x & := & r_{l-1}^2 + d s_{l-1}^2 & \text{und} & y & := & 2 r_{l-1} s_{l-1}, \quad \text{falls } l \text{ ungerade ist.} \end{array}$$

Zur Berechnung von (x, y) benötigt man somit die Näherungsbrüche $r_0/s_0,\ r_1/s_1, \ldots,\ r_{l-1}/s_{l-1}$ des Kettenbruchs

$$\sqrt{d} \;=\; [\,a_0, a_1, a_2, \ldots, a_k, a_{k+1}, \ldots\,].$$

Deren Zähler und Nenner berechnet man rekursiv: Man setzt

$$(2) \qquad \left\{ \begin{array}{llllllll} r_{-2} := 0, & r_{-1} := 1 & \text{und} & r_k & := & a_k r_{k-1} + r_{k-2} & \text{für jedes } k \in \mathbb{N}_0\,, \\ s_{-2} := 1, & s_{-1} := 0 & \text{und} & s_k & := & a_k s_{k-1} + s_{k-2} & \text{für jedes } k \in \mathbb{N}_0\,. \end{array} \right.$$

Diese klassischen Ergebnisse hat – etwa 100 Jahre nach Lagrange – A. Amthor in [5] verfeinert. Er zeigte: Ist p eine ungerade Primzahl, die d nicht teilt, so ist die kleinste natürliche Zahl n, für die in der Lösung (x_n, y_n) von (PELL) die Zahl y_n durch p teilbar ist, ein Teiler von $p-1$, falls d ein quadratischer Rest modulo p ist, bzw. ein Teiler von $p+1$, falls d ein quadratischer Nichtrest modulo p ist.

Der Lösung zweiter Teil

Wie bereits gezeigt wurde, ist zur Berechnung der Lösung der Aufgabe von Archimedes die Fundamentallösung der Pellschen Gleichung

$$(1) \qquad X^2 - 8 \cdot 5128\,58029\,09803 \cdot W^2 \;=\; 1$$

zu ermitteln, und wie das zu tun ist, wurde eben beschrieben. Aber *MuPAD* scheint nicht in der Lage zu sein, die Fundamentallösung dieser Pellschen Gleichung direkt zu berechnen. Man kommt mit folgendem Trick weiter, den Amthor in [5] verwendet: Es ist

$$8 \cdot 5128\,58029\,09803 \;=\; 2^3 \cdot 3 \cdot 7 \cdot 11 \cdot 29 \cdot 353 \cdot (2 \cdot 4657)^2 \;=\; 47\,29494 \cdot (2 \cdot 4657)^2.$$

Statt die Fundamentallösung (x, w) von (1) direkt zu berechnen, ermittelt man zuerst die Fundamentallösung (x, y) der Pellschen Gleichung

$$X^2 - 47\,29494 \cdot Y^2 \;=\; 1. \tag{3}$$

Dazu berechnet man mittels *MuPAD* den periodischen Kettenbruch für $\sqrt{47\,29494}$. Seine Vorperiode besteht aus der einen Zahl 2174, und seine Periode ist

$$\begin{aligned}(&1,2,1,5,2,25,3,1,1,1,1,1,1,15,1,2,16,1,2,1,1,8,6,1,21,1,1,3,1,1,1,2,2,6,\\ &\quad 1,1,5,1,17,1,1,47,3,1,1,\mathbf{6},1,1,3,47,1,1,17,1,5,1,1,\\ &\qquad 6,2,2,1,1,1,3,1,1,21,1,6,8,1,1,2,1,16,2,1,15,1,1,1,1,1,1,3,25,2,5,1,2,1,4348).\end{aligned}$$

(Die fettgedruckte 6 markiert die Mitte der Periode). Die Periode dieses Kettenbruchs hat die Länge 92, also sind Zähler $x := r_{91}$ und der Nenner $y := s_{91}$ seines 91-ten Näherungsbruchs zu berechnen. Mit Hilfe von *MuPAD* ergibt sich aus (2)

$$\begin{aligned} x &= 10993\,19867\,32829\,73497\,98662\,32821\,43354\,39010\,88049 \quad \text{und}\\ y &= 5\,05494\,85234\,31503\,30744\,77819\,73554\,04089\,86340.\end{aligned}$$

Damit ist die Fundamentallösung (x, y) der Pellschen Gleichung (3) berechnet. Die Menge aller Lösungen aus natürlichen Zahlen von (3) ist, wie bereits vermerkt, $\{(x_n, y_n) \mid n \in \mathbb{N}\}$, wobei für jedes $n \in \mathbb{N}$

$$x_n + y_n\sqrt{47\,29494} \;=\; \left(x + y\sqrt{47\,29494}\right)^n \tag{4}$$

gilt. Ist n die kleinste natürliche Zahl, für die y_n durch $2 \cdot 4657$ teilbar ist und ist $w := y_n/(2 \cdot 4657)$, so ist (x_n, w) offensichtlich die Fundamentallösung der Pellschen Gleichung (1).

Mit Hilfe der *MuPAD*-Funktion `isprime` zeigt man, daß 4657 eine Primzahl ist, und mit Hilfe der *MuPAD*-Funktion `numlib::jacobi`, daß 47 29494 ein quadratischer Nichtrest modulo 4657 ist. Daher ist nach dem Ergebnis von Amthor die kleinste natürliche Zahl n, für die y_n durch 4657 teilbar ist, einer der Teiler von $4657 + 1 = 4658 = 2 \cdot 17 \cdot 137$, also eine der Zahlen 1, 2, 17, 34, 137, 274, 548, 2329, 4658. (Man verwendet `numlib::divisors`). Mit Hilfe von `mod` zeigt man, daß $y_1 = y$ nicht durch 4657 teilbar ist. Aus (4) ergibt sich, daß für alle $m, n \in \mathbb{N}$

$$x_{m+n} \;=\; x_m x_n + 47\,29494\,y_m y_n \quad \text{und} \quad y_{m+n} \;=\; x_m y_n + x_n y_m$$

gilt, also für jedes $n \in \mathbb{N}$

$$x_{2n} \;=\; x_n^2 + 47\,29494\,y_n^2 \;=\; 2x_n^2 - 1 \quad \text{und} \quad y_{2n} \;=\; 2x_n y_n$$

und

$$x_{n+1} \;=\; x x_n + 47\,29494\,y y_n \quad \text{und} \quad y_{n+1} \;=\; x y_n + y x_n$$

gilt, und damit berechnet man der Reihe nach x_n und y_n für

$$n \;=\; 2,\ 4,\ 8,\ 16,\ 17,\ 34,\ 68,\ 136,\ 137,\ 274,\ 548,\ 1096,\ 2192,$$

stellt für $n = 2, 17, 34, 137, 274$ und 548 fest, daß y_n nicht durch 4657 teilbar ist, und berechnet schließlich

$$y_{2329} \;=\; y_{137+2192} \;=\; x_{137}\,y_{2192} + x_{2192}\,y_{137};$$

die Berechnung von x_{2329} ist nicht nötig. y_{2329} ist gerade und durch 4657 teilbar, und daher gilt für $w := y_{2329}/(2 \cdot 4657)$: (x_{2329}, w) ist die Fundamentallösung der Pellschen Gleichung (1). Man berechnet w^2 und mit den anfangs angegebenen Zahlen $a_1, a_2, \ldots, a_8$ und b für jedes $i \in \{1, 2, \ldots, 8\}$ die Zahl $N_i := a_i b w^2$. Dann ist $(N_1, N_2, \ldots, N_8)$ ein 8-tupel natürlicher Zahlen, das den Bedingungen (a), (b) und (c) der Aufgabe genügt, und es ist unter allen 8-tupeln, die dies tun, das mit minimalen Einträgen.

Die Herde des Sonnengottes besteht also aus

$$\begin{aligned} N \quad &:= \quad (a_1 + a_2 + \cdots + a_8) \cdot bw^2 \;=\; 7.76027\,14064 \ldots \cdot 10^{206544} \;= \\ &= \quad 77602\,71406\,48681\,82695\,30232\,83321 \;\ldots\; 88311\,89737\,23406\,62671\,94550\,81800 \end{aligned}$$

Rindern, ist also doch wohl ziemlich groß: Die Zahl N hat $2\,06545$ Stellen. Und doch kann *MuPAD* sie ohne Probleme berechnen.

Zur Geschichte

Der von Lessing 1773 veröffentlichte Text enthält eine aus der Antike stammende Lösung, die aber nicht korrekt ist; die darin angegebenen Anzahlen N_1, $N_2, \ldots, N_8$ erfüllen nur die Bedingungen in (a) (mit den hier verwendeten Bezeichnungen ist $N_i = 4a_i$ für jedes $i \in \{1, 2, \ldots, 8\}$), aber weder die Bedingung in (b), noch die in (c). Lessings Veröffentlichung enthält auch einige Bemerkungen eines Wolfenbütteler Mathematiklehrers Chr. Leiste, der zeigte, wie man alle Lösungen aus $\mathbb{N}^8$ des linearen Gleichungssystems in (a) berechnen und darunter die ermitteln kann, die auch der Bedingung in (b) genügen. Einen vollständigen Weg zur Lösung der Aufgabe veröffentlichte erstmals A. Amthor im zweiten Teil von [5] (diesem Weg folgt die vorliegende Darstellung); er gab die ersten vier Stellen und die Stellenzahl der Zahl N korrekt an. Im Jahr 1889 berechnete A. H. Bell, zusammen mit einem dazu gegründeten mathematischen Stammtisch, in vier Jahren die ersten 28 Stellen (davon die beiden letzten nicht korrekt) und die letzten 12 Stellen von N. Zum ersten Mal wurde N wohl im Jahr 1965 von H. C. Williams, R. A. German und C. R. Zarnke vollständig berechnet; sie benötigten dazu auf dem damaligen Computer der Universität von Waterloo 7 Stunden und 49 Minuten (vgl. [6]). Im Jahr 1981 wurde N erstmals veröffentlicht: H. L. Nelson berechnete N und fünf weitere, größere Lösungen der Aufgabe auf einem CRAY-1-Rechner, der dazu zehn Minuten brauchte, und druckte in [7] alle $2\,06545$ Stellen von N auf nicht ganz zwölf Druckseiten ab. Heute kann jeder, der *MuPAD* zu Verfügung hat, die Zahl N berechnen; wie lange sein Rechner dazu benötigt, mag jeder Leser selbst feststellen. So ist die alte Aufgabe von Archimedes auch eine Illustration zur Entwicklung des Scientific Computing geworden.

Die Aufgabe hat seit ihrer Veröffentlichung durch Lessing zu mancherlei Diskussionen geführt. Nach Meinung von B. Krumbiegel, der im ersten Teil von [5] neben einer Übersetzung ins Deutsche eine eingehende historische und philologische Untersuchung des Texts veröffentlicht hat, ist es "möglich, ja wahrscheinlich", daß die Aufgabe wirklich von Archimedes stammt. Auch Heiberg schreibt in [2], daß es keinen Grund gibt, die Aufgabe nicht Archimedes zuzuschreiben. Der Wortlaut der Aufgabenstellung wurde im 19. Jahrhundert immer wieder diskutiert, wovon Krumbiegel im ersten Teil von [5] berichtet. So dachte man etwa darüber nach, wie die Bedingung (b) wirklich zu interpretieren sei, wo doch ein Rind im allgemeinen länger als breit ist.

Ob aber Archimedes oder einer seiner Zeitgenossen die Aufgabe gelöst hat, ist eine andere Frage; niemand kann sie nach den heutigen Kenntnissen über Archimedes und die Rechentechniken, über die er verfügte, mit Sicherheit beantworten. Es sind eben nicht alle mathematischen Schriften aus der Antike erhalten geblieben – wer rettet denn zuerst Mathematikbücher aus einer brennenden Bibliothek? Daher weiß man nicht, inwieweit den antiken Mathematikern Lösungsverfahren für Pellsche Gleichungen bekannt waren. Jedenfalls konnte man Pellsche Gleichungen schon vor Fermat lösen, wenn auch außerhalb des Rahmens der europäischen Mathematikgeschichte: Bereits um das Jahr 1000 lösten indische Mathematiker (Acarya Jayadeva und später Bhaskara Acharya) Pellsche Gleichungen, und manches spricht dafür, daß die von ihnen verwendeten Methoden schon 500 Jahre früher bekannt waren. Wie Clas-Olof Selenius in [8] zeigt, läßt sich übrigens auch das Lösungsverfahren der alten indischen Mathematiker mit Hilfe von Kettenbruchentwicklungen beschreiben; die dabei auftretenden Kettenbrüche sind, anders als bei Lagrange, halbregelmäßige Kettenbrüche und zwar solche, die besonders gute Approximationen liefern (vgl. Selenius [9]). Man darf selbstverständlich nicht meinen, daß die Verfahren der indischen Mathematiker in

Wirklichkeit aus der griechischen Welt stammen, aber man darf sich vorstellen, daß vielleicht auch dort schon früh ähnliche Methoden bekannt waren.

Man braucht nicht daran zu zweifeln, daß die Aufgabe, die Zahl der Rinder in der Herde des Gottes Helios zu berechnen, von Archimedes stammt. Von welchem anderen Mathematiker der griechischen Antike sollte sie schon stammen? Nicht sicher ist, ob Archimedes sie auch gelöst hat. Aber jeder, der sie selbst gelöst hat, darf darüber nachdenken und sich vorstellen, welche Aufgaben Archimedes gefunden und gelöst hätte, wenn er *MuPAD* zu Verfügung gehabt hätte.

Literatur

[1] G. E. Lessing, Zur Griechischen Anthologie. In: Gotthold Ephraim Lessings sämtliche Schriften, herausgegeben von K. Lachmann, Bd. 12, 99–115, G. J. Göschen'sche Verlagshandlung, Leipzig, 3. Auflage 1897, Nachdruck: de Gruyter, Berlin 1968

[2] Archimedes, Opera Omnia, herausgegeben von I. L. Heiberg (Bibliotheca Scriptorum Graecorum et Romanorum Teubneriana). Teubner, Stuttgart 1972, Nachdruck der Ausgabe von 1913

[3] O. Perron, Die Lehre von den Kettenbrüchen I und II. Teubner, Stuttgart, 2. Auflage 1954 und 1957

[4] F. Schwarz, Elementare Zahlentheorie mit *MuPAD*. Erscheint demnächst

[5] B. Krumbiegel, A. Amthor, Das Problema bovinum des Archimedes. Z. Math. Phys., Hist.-lit. Abt. **25** (1880) 121–136 und 153–171

[6] H. C. Williams, R. A. German, C. R. Zarnke, Solution of the cattle problem of Archimedes. Math. Comp. **19** (1965) 671–674

[7] H. L. Nelson, A solution to Archimedes' cattle problem. J. Recreational Math. **13** (1980/81) 162–176

[8] C.-O. Selenius, Rationale of the chakravāla process of Jayadeva and Bhāskara II. Historia Math. **2** (1975) 167–184

[9] C.-O. Selenius, Konstruktion und Theorie halbregelmäßiger Kettenbrüche mit idealer relativer Approximation. Acta Acad. Aboensis, Math. Phys. **22.2** (1960) 1–77

Der MuPAD Workshop 2000

Andreas Sorgatz
SciFace Software, `sorgatz@sciface.com`

Im September 2000 fand im Technologiepark Paderborn der MuPAD-Workshop 2000 statt. Ein internationales Publikum von 80 Teilnehmern diskutierte über Anwendungen und Neuentwicklungen sowie Perspektiven und feierte zugleich das 10-jährige Jubiläum von MuPAD.

Traditionell war der Workshop auch diesmal wieder in einen Tag für Forschung und Technik und einem zweiten zum Thema Schule, Studium und multimediale Anwendungen, begleitet von einem *MuPAD*-Kurs, gegliedert. In Vorträgen, Demonstrationen und Postern wurden dazu von den Entwicklern und geladenen Referenten neue Entwicklungen, Computeralgebra-Anwendungen sowie insbesondere auch zukünftige Entwicklungspläne vorgestellt und mit den Teilnehmern diskutiert.

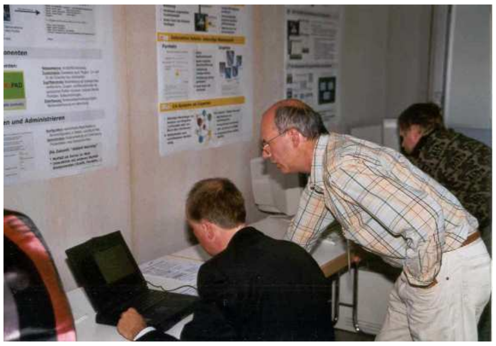
RegDir. Dr. Köster, NRW:
Der offizielle Vertreter des Landes Nordrhein-Westfalen informiert über die Unterstützung der MuPAD-Forschungsgruppe durch das Land und erklärt warum er die Einrichtung eines kommerziellen Zweiges zu *MuPAD* angeregt hat.

PhD. Michael Wester, USA:
Michael Wester berichtet über den Nutzen von Computeralgebra Systemen, d.h. ihre Vorteile und Nachteile, und zählt Aspekte auf, die bisher noch nicht geeignet berücksichtigt wurden und/oder Gegenstand zukünftiger Forschung sind.

Prof. Dr. Wolfram Koepf:
Der Referent für Lehre und Didaktik der Fachgruppe Computeralgebra der DMV/GI/GAMM stellt die Fachgruppe vor und berichtet über Aktivitäten im Bereich des Einsatzes von Computeralgebra in Lehre, Ausbildung und Weiterbildung.

PhD. Steve Swanson, USA:
Der Leiter der Entwicklungsabteilung der Firma MacKichan Inc. führt die neue Version des mathematischen Textverarbeitungssystems *Scientific Workplace integrated with MuPAD* vor.

Während der Poster Session:
Poster Session und Kaffeepausen wurden für den Erfahrungsaustausch und fruchtbare Diskussionen genutzt.



Schwerpunkt der Präsentationen der *MuPAD*-Entwickler war die Einführung der neuen *MuPAD*-Version *MuPAD* Pro 2.0 mit vielen nützlichen Neuerungen in der Benutzungsoberfläche (lesen Sie hierzu auch den Artikel *MuPAD Pro 2.0 für Windows im Griff* in dieser Ausgabe) sowie der algebraischen Funktionalität des Systems, die Vorstellung einer Internet-fähigen *MuPAD* Version für E-Learning und Web-Based-Training (einen kurzen Überblick gibt der Artikel *Mathematik im Web – Der MuPAD Computing Server* in dieser Ausgabe) sowie die Demonstration der neu entwickelten multimedialen Computeralgebra-Komponenten von SciFace Software (eine Anwendung dieser Komponenten wird im Artikel *SmartTools: Entwicklung von Lernsoftware* in dieser Ausgabe beschrieben).

Besonders erfreulich an diesem Workshop war auch der hohe Anteil von LehrerInnen und Vertretern von Schulbuchverlagen. Weitere Informationen zum *MuPAD*-Workshop 2000 erhalten Sie auf dem NRW-Bildungsserver:

`http://www.learn-line.nrw.de/info/news/reports/mupad/mupad.htm`

sowie auf dem Web-Server der *MuPAD*-Forschungsgruppe:

`http://www.mupad.de/mw2000/proceedings/index_d.shtml`

MuPAD – Schnupperkurse und Lehrerfortbildungen

Kai Gehrs, Andreas Sorgatz
MuPAD-Team, acrowley@mupad.de, sorgatz@sciface.com

Lehrer fit gemacht für MuPAD – Unter diesem Motto bietet das MuPAD-Team der Universität Paderborn in Zusammenarbeit mit SciFace Einführungskurse und Lehrerfortbildungen an.

MuPAD-Einführungskurs (”Schnupperkurs”)

Wir geben Ihnen einen kurzen Überblick zu *MuPAD* und den Konzepten des algebraisch-symbolischen Rechnens in der Computeralgebra. Dabei erklären wir auch die Unterschiede zwischen einem Taschenrechner und einem Computeralgebra-System.

Im Anschluss bieten wir Ihnen die Möglichkeit, *MuPAD* in einer ein- bis zweistündigen ”Hands-on-session” unter Anleitung direkt auszuprobieren. Wir stellen dazu die Software *MuPAD Pro 2.0 für Windows* sowie interaktive Arbeitsblätter (*MuPAD*-Notebooks) zur Verfügung, anhand derer der Umgang mit *MuPAD* erlernt und geübt wird.

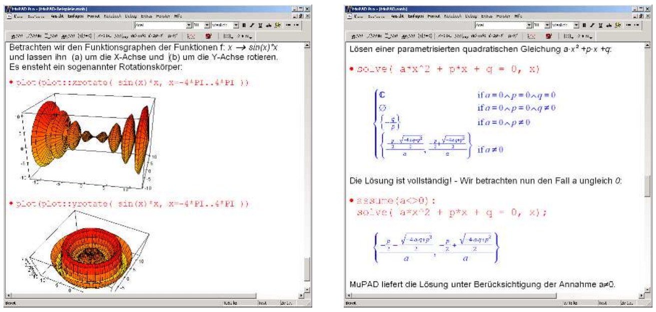
Gedrucktes Lehrmaterial und CDROMs mit *MuPAD* 100-Tage-Lizenzen können auf Wunsch gegen eine geringe Kostenpauschale bereitgestellt werden. Weitere Details auf Anfrage.

MuPAD-Schnupperkurs am Felix-Fechenbach-Berufskolleg in Detmold

MuPAD-Lehrerfortbildung

Ein- bis zweitägige Lehrerfortbildungsseminare bieten wir von Zeit zu Zeit über die Lehrerfortbildungsstätten der Länder und Bezirke an. Bitte wenden Sie sich zunächst an die entsprechenden Dezernate Ihres Regierungsbezirkes.

Sollte in Ihrem Bezirk derzeit kein *MuPAD*-Kurs angeboten werden, so schreiben Sie uns bitte bei entsprechendem Bedarf an die untenstehende Adresse.



Interaktives Arbeiten mit Microsoft-Word^{TM}-ähnlichen *MuPAD*-Notebooks

MuPAD-Kurse im Rahmen von Tagungen

Schnupperkurse und Lehrerfortbildungsseminare bieten wir auch auf Fachtagungen, Messen und Workshops an. Bitte schreiben Sie uns dazu frühzeitig an die untenstehende Adresse. Die Planung und Organisation benötigt erfahrungsgemäß einige Zeit.

Frank Postel auf dem *MuPAD*-Workshop in Paderborn im September 2000

Weitere Informationen bekommen Sie von der Sektion Schule des *MuPAD*-Teams der Universität Paderborn:

Stichwort: "Schule"
MuPAD, FB17-Mathematik
Universität Paderborn
Warburger Str. 100
D-33095 Paderborn
Email: schule@mupad.de
Fax: ++49 (0)5251 605534

Kalender

ICTMT5 – The Fifth International Conference on Technology in Mathematics Teaching

Universtät Klagenfurt, Österreich, August 6-9, 2001

Weitere Informationen: `http://www2.ifi.uni-klu.ac.at/ictmt5/`

European Conference on Educational Research

Universit´e Charles de Gaulle, Lille, Frankreich, September 5.-8., 2001

Weitere Informationen: `http://www.eera.ac.uk/events.html`

MNU-Tagung Nordrhein

Köln, Deutschland, September 11, 2001

Weitere Informationen: `Sabine Schmalstieg (ssgne@t-online.de)`

MNU-Tagung Westfalen

Dortmund, Deutschland, September 25, 2001

Weitere Informationen: `Egbert Bubel (e.bubel@t-online.de)`

Medien verbreiten Mathematik - Herbsttagung 2001 des AK Mathematik und Informatik”

Dillingen/Donau, Deutschland, September 28.-30., 2001

Weitere Informationen: `http://www.didmath.ewf.uni-erlangen.de/ak_wob/`

Europa in Schule und Lehrerbildung

Universität Paderborn, Deutschland, Oktober 1-2, 2001

Weitere Informationen: `http://www.uni-paderborn.de/plaz/europatagung/`

23. Interpaedagogica

Linz, Österreich, November 8.-10., 2001

Weitere Informationen: `http://www.interpaedagogica.at`

Die Bildungsmesse

Köln, Deutschland, Februar 19.-23., 2002

Weitere Informationen: `http://www.didacta.de`

Jahrestagung der GDM

Klagenfurt, Österreich, Februar 25.-März 1., 2002

Weitere Informationen: `http://www.uni-giessen.de/gdm`

MNU-Kongress

Hannover, Deutschland, März 24.-28, 2002

Weitere Informationen: `http://www.gbg-seelze.de/MNU-2002/`

Computeralgebra in Lehre, Ausbildung und Weiterbildung III

Kloster Schöntal, Deutschland, April 2.-5, 2002

Weitere Informationen: `http://www.mathematik.uni-kassel.de/~koepf/Schoental2002/`

IMACS-ACA - Applications of Computer Algebra

Volos, Griechenland, Juni 25.-28., 2002

Weitere Informationen: `http://www.orcca.on.ca/~ilias/aca2002.html`

Surfin' the Web

Und noch einige interessante, nützliche und kuriose Plätze im „global village“.

Die Tipps in dieser Ausgabe wurden zusammengestellt von Malte Beecken (`MalteBeecken@gmx.de`), Wolfgang Lindner (`lindner@math.uni-duisburg.de`) und Andreas Sorgatz (`sorgatz@sciface.com`).

Mathematik

`http://btmdx1.mat.uni-bayreuth.de/smart/intro.htm`
Hier liegen die Aufgaben aus der am Staatsinstitut für Schulpädagogik und Bildungsforschung (ISB), München entwickelten Aufgabenbank SMART ebenfalls im DVI-Format vor und können daher mit dem *MuPAD*-Viewer betrachtet und innerhalb der *MuPAD*-Umgebung direkt benutzt werden. Momentan sind die Jahrgangsstufen 5, 6, 8, 9, 10 und 11 mit insgesamt ca. 1900 Aufgaben vertreten: ein überaus reichhaltiger Pool für *MuPAD*-Nutzer.

`http://wmax04.mathematik.uni-wuerzburg.de/~gdm/`
Die Home-Page der *Gesellschaft für Didaktik der Mathematik* mit vielen Informationen zum Lehren und Lernen von Mathematik.

`http://www.bezreg-duesseldorf.nrw.de/schule/mathe/mathe.htm`
Der *Mathe-Treff der Bezirksregierung Düsseldorf* bietet einen Mathe-Chat, Matheknobeleien für alle Alterstufen, Hinweise zu amtlichen Lehrerfortbildungen, Unterrichsmaterialien mit weiteren Links und vieles mehr.

Wörterbücher

`http://www.math.uni-goettingen.de/baule/wbuch.html`
Kleines Mathematik-Wörterbuch im Web: Kennen Sie den englischen Begriff für 'Wendepunkt'? Nein? Englische Übersetzungen von diesem und vielen weiteren mathematischen Fachbegriffen können Sie hier nachlesen.

`http://news.lycos.de/ndr/ge/`
Die Hilfe in der Not: Das Rechtschreibwörterbuch zur Neuen Deutschen Rechtschreibung.

`http://dict.tu-chemnitz.de/`
Deutsch-Englisches Wörterbuch mit Einheitenumrechnung. Eine Hilfe bei allgemeinen Übersetzungsproblemen.

Weitere Services

`http://routenplaner.web.de`
Sie wollen die Geburtsstadt von *MuPAD* kennenlernen :-). Der WEB.DE-Routenplaner beschreibt Ihnen den Weg straßgenau bis nach Paderborn. Sie können sich ebenso Eruopaweite Routen bestimmen lassen.

`http://de.biz.yahoo.com/tx/`
Der jährliche Zauber mit der Lohn- und Einkommenssteuererklärung gibt immer wieder Anlass für Fragen. Hier erhalten Sie aktuelle Informationen zum Thema Steuern 2001 sowie Links auf weitere Web-Seiten zu diesem Thema.

`http://www.steuer.niedersachsen.de/default2.htm`
Die Besitz- und Verkehrsteuerabteilung der Oberfinanzdirektion Hannover bietet Online-Dienste zur Berechnung der Kfz-Steuer, zur Wahl der Steulerklasse für Ehegatten und mehr.

`http://www.wetteronline.de/`
Wetter Online bietet nahezu alles, was das Wetter zu bieten hat: Aktuelle Wetterkarten, Pegel, Ozonkonzentrationen, Pollenflug, Blitze und vieles mehr weltweit. In Deutschland sind die Informationen zum Teil Städte-genau.

`http://www.search4science.com`
Eine Suchmaschine im World-Wide-Web, spezialisiert auf wissenschaftliche Informationen.

`http://www.c4.com`
Eine weitere praxistaugliche Meta-Suchmaschine im World-Wide-Web.

Für die eigene Web-Seite

`http://www.w3schools.com`
Hier finden Sie kostenlose Tutorien zur Erstellung eigener Web-Seiten mit Hilfe von HTML, DHTML, JavaScript und vieles mehr.

`http://www.geocities.com/SiliconValley/7116/`
The JavaScript Planet: Eine Sammlung von freien JavaScript und Java-Applets.

Software

`ftp://ftp.franken.de/pub/win32/develop/gnuwin32/cygwin/`
Hier gibt es die GNU-Tools, die Sie ggf. von Linux und UNIX her kennen, auch für Ihren PC mit Microsoft-Windows. Enthalten sind allerlei nützliche Programme - nicht nur für Entwickler.

In Eigener Sache

`http://www.sciface.com/edu/`
Informationen über unsere Aktivitäten im Bereich Schule und Studium.

`schule@mupad.de`
Fragen oder Anregungen zu *MuPAD* in der Lehre in Schule und Studium? Unter dieser eMail-Adresse erreichen Sie das *MuPAD*-Schulteam. Wir ermöglichen Ihnen einen direkten Kontakt und die Diskussion mit den Entwicklern von *MuPAD*.

Allerlei

`http://www.max-der-erfinder.de/`
Anregungen für den Werkunterricht: Die Erfinderfamilie Heindl (Max, Ursula, Tobias und Marion) demonstriert Ihre (wohl nicht immer ganz ernst gemeinten Erfindungen im Web.

`http://www.waskochen.de`
Ein schönes Rezept-Archiv mit vielen zusätzlichen Informationen. Besondere Unterstützung für rechenschwache Köche: die Mengenangaben der Rezepte lassen sich on-line für 1-10 Personen berechnen ;-).