

■ Mathematica 6.0 – empfehlenswertes Update

Nach mehr als drei Jahren ist jetzt endlich Mathematica 6.0 erschienen, das Ergebnis ist durchaus gelungen und viele der neu implementierten Funktionen berücksichtigen die Wünsche der Nutzer.

Mathematica besteht aus zwei Programmen, dem Kernel, der die Berechnungen ausführt und dem Frontend, welches die Eingaben des Benutzers an den Kernel überträgt und die Ergebnisse darstellt. Das Ergebnis einer Eingabe änderte sich bisher während der weiteren Arbeit nicht mehr. Dieses Konzept ist natürlich nützlich, wenn es gilt, die einzelnen Schritte einer Berechnung zu dokumentieren, für die interaktive Arbeit ist es gänzlich ungeeignet.

Die wichtigste Neuerung ist die Möglichkeit, interaktive Elemente in Mathematica-Dokumente integrieren zu können. Anstelle der bisher in Paletten benutzten Tastflächen und des auf Java basierenden GUIKits sind alle Elemente einer modernen grafischen Benutzeroberfläche in Mathematica vorhanden, das betrifft nicht nur das Frontend, sondern die gesamte Programmiersprache: Man kann also Listen von Schieberegler, Schaltflächen und Menu-Einträgen symbolisch bearbeiten und Nutzeroberflächen im Mathematica-Frontend gestalten. Das dient nicht nur der Gestaltung von Eingabemasken für die Parameter komplexer Berechnungen mit Mathematica, sondern auch der Programmierung interaktiver Demonstrationen für die Lehre und dem interaktiven Erkunden der Abhängigkeit der symbolischen oder numerischen Ergebnisse von den Parametern des Modells in der Forschung. Während das Gestalten solcher Nutzeroberflächen in Sprachen wie C++ oder Java eine mühselige Aufgabe ist, kommt man mit Mathematica schnell und mit wenigen Zeilen Programm-Code an sein Ziel. Am einfachsten demonstriert das ein Beispiel:

Dazu soll die Bahnkurve eines Foucaultschen Pendels berechnet und zusammen mit dem Pendel dargestellt werden. Die Differentialgleichungen für das Pendel

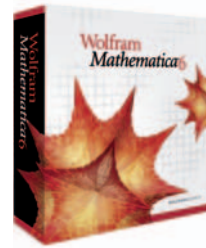
haben drei Parameter, nämlich die Frequenz des Pendels, die Winkelgeschwindigkeit der rotierenden Erde und den Winkel zur Erdachse des Punktes, an dem das Pendel schwingt. Außerdem soll noch ein kurzes Stück der dreidimensionalen Bahnkurve als eine Spur des Pendels gezeichnet werden. Mit 30 Zeilen Programm ist das geschafft (Infokasten).

Die wesentliche Neuerung, die die interaktiven Fähigkeiten ermöglicht, ist die Funktion *Dynamic[]*. Dynamische Ausdrücke werden vom Frontend gespeichert und aktualisiert, sobald ihr Wert vom Mathematica Kernel, während einer Berechnung, geändert wird. Mit der neuen Funktion *Manipulate[]* bietet Mathematica eine in den meisten Fällen ausreichende Schnittstelle an, um die Parameter eines Ausdruckes zu variieren. Dabei funktioniert *Manipulate[]* nicht nur mit Grafiken, sondern mit jedem Mathematica-Ausdruck. Entsprechend des Datentyps gibt es dabei sowohl Schieberegler als auch die Auswahl aus Listen, die Eingabe über die Tastatur und Punkte sowie Punkt-Listen. Auf den Demonstrationswebseiten für Mathematica 6.0 kann man sich einen Überblick

über all diese Möglichkeiten verschaffen [1].

Die Berechnung der Ergebnisse in der interaktiven Ausgabe sollte allerdings schnell vonstatten gehen. Bei Darstellungen, die mehr als ein bis zwei Sekunden für die Berechnung benötigen, sollte man über eine Optimierung der Berechnung nachdenken oder auf die interaktive Ausgabe verzichten.

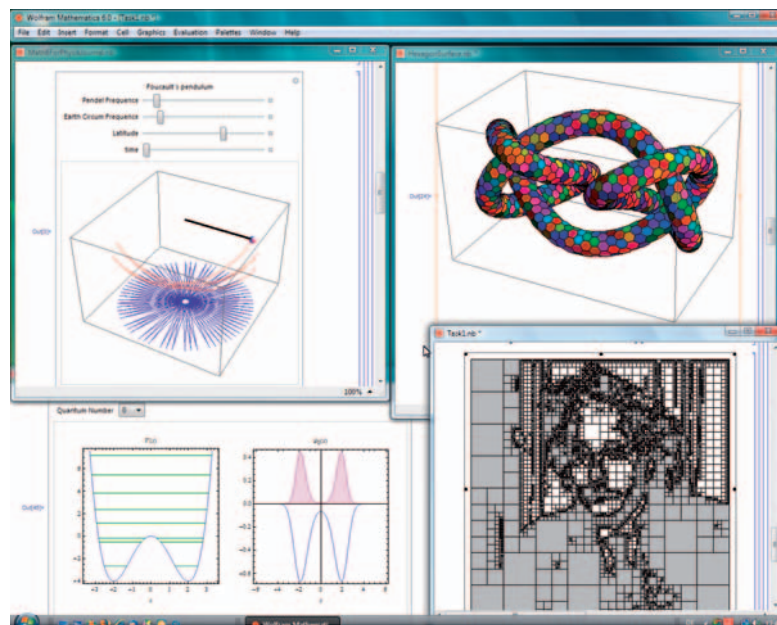
Die zweite wesentliche Neuerung ist die Grafik. Endlich hat sich



Hersteller: Wolfram Research, www.wolfram.com
Systemanforderungen: 32- und 64-Bit-Windows, Mac OS X (PPC, Intel), Linux (X86, X64, Itanium), Unix
Preis: 3790 / 405 / 152 € (Industrie/Lehrer/Studenten, inkl. 1 Jahr Premier Service) 758 € pro Versionsstufe (Update der Industrielizenz, wenn kein Premier Service läuft)

Mathematica von PostScript als Grafik-Grundlage verabschiedet. Bisher erzeugte der Mathematica-Kernel PostScript-Anweisungen, die dann vom Frontend oder einem externen PostScript-Interpreter in eine Bitmap umgewandelt wurden. Da PostScript aber eine Beschreibungssprache für Druckseiten ist, eignet sich das Format für die interaktiven Grafiken nicht.

Das Frontend stellt nun die symbolischen Grafikobjekte direkt dar und benutzt dabei die Grafik-Hardware des Computers. Die erste und offensichtliche Neuerung ist, dass nun Transparenz möglich ist



Die neue Version 6 von Mathematica beinhaltet auch zahlreiche Neuerungen im grafischen Bereich.

Dr. Jens-Peer Kuska,
Interdisziplinäres
Zentrum für Bio-
informatik (IZBI),
Universität Leipzig,
Härtelstr. 16-18,
04107 Leipzig

– alle Farbfunktionen haben einen Transparenzkanal erhalten. Auch die Algorithmen zum Berechnen der Grafiken sind vollständig überarbeitet worden. Während früher alle Punkt-Koordinaten in den symbolischen Ausdrücken für die Grafiken direkt und oft mehrfach enthalten waren, werden nun indizierte Punktmengen in einem `GraphicsComplex[]` verwendet. Das spart oft zwei Drittel des Speicherplatzes für die Grafiken. Man hat sich auch von den speziellen Grafik-Strukturen für Oberflächen-, Höhenlinien- und Dichteplots befreit. Alle Funktionen zum Zeichnen von Funktionen und parametrischen und impliziten Flächen sind nun adaptiv, verwenden also in Abhängigkeit von der Krümmung unterschiedlich große Liniensegmente oder Dreiecke. Besonders bei der dreidimensionalen Grafik sollte man aber kein beschleunigtes Zeichnen im Vergleich zur Vorgängerversion erwarten. Das liegt daran, dass wesentlich mehr Dreiecke erzeugt werden und nun Oberflächennormalen zur glatten

Interpolation der beleuchteten Flächen berechnet werden. Die Qualität der Grafiken ist jedoch deutlich gestiegen. Bei der Verwendung von Dichte und Kontur-Grafiken muss bedacht werden, dass die Funktionen für die Darstellung glatter, stetiger Flächen optimiert wurden, denn nur dann ist die adaptive Zerlegung in Dreiecke effizient.

Da Mathematicas 3D-Grafik jetzt mit Dreiecks- und Tetraedergittern umgehen kann, hat sich auch die Darstellung von Daten verbessert. Alle Funktionen zum Visualisieren von Daten erlauben nun auch Daten auf irregulären Gittern, eine dazu notwendige Delauny-Triangulation wird automatisch ausgeführt.

Die dreidimensionalen Objekte lassen sich im Frontend interaktiv drehen, skalieren und verschieben, was nun bedeutend besser geht als mit dem alten RealTime3D Paket. Für zweidimensionale Grafiken gibt es jetzt einen interaktiven Editor, mit dem sich Beschriftungen und Pfeile anbringen lassen, das erspart in vielen Fällen die Verwendung eines externen Vektorzeichensprogramms.

Neu und teilweise irritierend ist die Eigenschaft, dass Daten und Funktionen, die nicht darstellbar sind, ohne Warnung ausgelassen werden, so erzeugt `Plot[I*Sin[x], {x, 0, Pi}]` einfach eine leere Grafik, weil die komplexen Werte nicht dargestellt werden können.

Da die Grafiken jetzt vom Frontend dargestellt werden, ist die Funktion `Show[]` überflüssig geworden, man benötigt sie nur noch, um die Optionen einer Grafik zu ersetzen. Wer innerhalb einer Funktion Grafiken erzeugt und diese mit `Show[]` hat anzeigen lassen, muss nun `Print[]` benutzen.

Die neuen sehr tief greifenden Änderungen in Mathematica 6.0 haben natürlich Folgen für die Anwendungspakete. Sofern die Pakete vor allem symbolische oder numerische Funktionen enthalten, sind die Probleme mit der neuen Version klein. Hat man jedoch Pakete, die viele Grafik-Funktionen benutzen, so ist eine Anpassung oft nicht einfach. Bei Notebooks der

Version 5.2 macht das Frontend Vorschläge, um den enthaltenen Programm-Code an die neue Version anzupassen. Bei Mathematica-Packages kann die aktuelle Version der Wolfram-Workbench unter der Refactor Option Vorschläge für die Anpassung des Quelltextes an die neue Version machen.

Die meisten Standardpakete sind in den Kernel integriert und verbessert worden, was leider zu einigen Namensänderungen geführt hat.

Neben den aufgeräumteren Menüs des Frontends fällt vor allem das Syntax-Highlight auf, bekannte Symbole, Symbole mit einem Wert, lokale Symbole und Muster werden unterschiedlich dargestellt, was viele Tippfehler vermeidet. Zudem hat das Frontend einen integrierten Debugger, der an die Debugger der gängigen Entwicklungsumgebungen erinnert. Für große Projekte empfiehlt sich aber der Debugger der Wolfram-Workbench.

Die Online-Hilfe ist völlig umgestellt worden und soll gerade Anfängern den Umgang mit Mathematica erleichtern. Auch wenn die Hilfe zahlreiche interaktive Beispiele bietet, die man am Computer sofort ausprobieren kann, fehlt ein gedrucktes Handbuch. Gerade um einen Überblick über die Änderung und die neuen Funktionen zu erhalten, wäre ein klar strukturiertes Handbuch notwendig. Eine Neuaufgabe des „The Mathematica Book“ für Mathematica 6.0 gibt es leider nicht und es ist auch keine geplant. Man kann nur hoffen, dass diese Entscheidung nicht endgültig ist, denn das exzellente Handbuch war immer eine der Stärken von Mathematica. Die gesamte Dokumentation ist auch im Web vorhanden [2].

An die zahlreichen neuen Eigenschaften und Verbesserungen hat man sich schnell gewöhnt, und ein Update auf die neue Version kann nur empfohlen werden.

Jens-Peer Kuska

[1] <http://demonstrations.wolfram.com>

[2] <http://reference.wolfram.com/mathematica/guide/Mathematica.html>

BEISPIEL FOUCAULTSCHES PENDEL

Programmcode

```
pendelPos[{x_, y_}] := {x, y, -Sqrt[10 - x^2 - y^2]}
pendel[tau_?NumericQ, sol_] := {AbsoluteThickness[3],
  Line[{{0, 0, 0},
    pendelPos[{x[t], y[t]}]}],
  Sphere[pendelPos[{x[t], y[t]}], 0.15] /.
    sol[[1]]] /. t -> tau
Manipulate[
  DynamicModule[{traj, sol, fed, fulltraj},
    fde =
      {x'[t] == -omega^2 * x[t] + 2 * Omega * Sin[Phi] * y'[t],
       y'[t] == -omega^2 * y[t] - 2 * Omega * Sin[Phi] * x'[t]};
    sol = NDSolve[
      Join[fde, {x[0] == 2, y[0] == 2, x'[0] == 0, y'[0] == 0},
        {x[t], y[t]}, {t, 0, 64 * Pi}, MaxSteps -> Infinity];
    fulltraj = ParametricPlot3D[
      {x[t], y[t], -4} /. sol[[1]], {t, 0, 64 * Pi},
      PlotPoints -> 1024
    ];
    traj = ParametricPlot3D[
      pendelPos[{x[t], y[t]}] /. sol[[1]],
      {t, t1, t1 + 4 * Pi},
      PlotStyle -> RGBColor[1, 0, 0], Evaluated -> True
    ];
    Graphics3D[
      {traj[[1]], fulltraj[[1]], pendel[t1, sol]},
      PlotRange -> {{-3, 3}, {-3, 3}, {-4, 0}}
    ], {{omega, 1, „Pendel Freqeunce“}, 0.1, 10},
    {{Omega, 1/16, „Earth Circum Freqeunce“}, 0, 1/2},
    {{Phi, Pi/3, „Latitude“}, 0, Pi/2},
    {{t1, 0, „time“}, 0, 60 * Pi},
    FrameLabel -> {None, None, „Foucault's pendulum“},
    SaveDefinitions -> True
  ]
]
```