Mathematische Software im Unterricht?

Bruno Buchberger

Research Institute for Symbolic Computation (RISC)
Johannes Kepler Universität, Linz / Hagenberg, Österreich

Universität Passau, 21. Dezember 2016

Die Frage

Das White - Box / Black - Box Prinzip

Beispiel : Newton - Verfahren

Übung : Gröbner - Basen

Vorbemerkung

Dieser Vortrag ist nicht das Modell einer Unterrichtsstunde,
sondern eine Analyse für LehrerInnen*) des "Flusses der mathematischen Arbeit" an Hand von zwei Beispielen.

Daraus kann man Klarheit für die Verwendung mathematischer Software ("Technologie", "Algorithmik", ...) in allen anderen Beispielen erhalten (bis zum Kindergarten ...)

*) Ab jetzt gelten alle Formulierungen geschlechtsneutral.

Trivialisierte Mathematik

Nachdem ein mathematisches Problem (aufbauend auf einem Theorem) durch einen Algorithmus gelöst wurde, ist dieses Problem "**trivialisiert**".

Für ein trivialisiertes Problem ist es unvernünftig, eine gegebene Instanz des Problems "mit der Hand" (d.h. "mit dem Kopf") zu lösen.

Beispiele trivialisierter Probleme : Sortieren, lineare Gleichungen, kürzeste Wege in Graphen, Integrieren elementarer Funktionen, ...

Durch mehr und mehr Mathematik (Theoreme), werden mehr und mehr mathematische Probleme trivialisiert.

Es ist das Ziel der Mathematik, sich selbst zu trivialisieren.

Mathematische Software - Systeme : Die Summe trivialisierter Mathematik bis zum gegebenen Zeitpunkt.

Glücklicherweise gibt es keine obere Grenze für die Triviaisierung. (Gödels Unvollständigkeitssatz ...)

Sollen Studenten trivialisierte Mathematik lernen?

Sollten wir (Lehrer und Studenten*)) unsere Zeit nicht lieber für "kreative" Mathematik verwenden?

Und für die Anwendung der Mathematik für Probleme der "realen Welt"?

Anders ausgedrückt : Sollten wir den Studenten erlauben, mathematische Software ("Technologie", "CAS", ...) zu verwenden?

Die Antwort der "**Puristen**" : Nein! Technologie verdirbt die Studenten!

Die Antwort der "**Populisten**" : Ja! Technologie befreit die Studenten!

Erstaunlich, dass diese Diskussion immer noch im Gange ist ... (nach 5000 Jahren "Technologie", und insbesondere 50 Jahren "Software")

*)   Studenten und Schüler.

Sollen Studenten trivialisierte Mathematik lernen?

Ich denke, **beide Antworten sind falsch.**

(Für die Vermittlung von Mathematik für Anwendungsfächer, z.B. Architektur, Biologie, etc., kann man allerdings "populistisch" sein ...)

Ich betrachte hier die Vermittlung von Mathematik in den Schulen und für Mathematik - (und Informatik -) Studenten.

Meine Antwort : Das White - Box / Black - Box Prinzip : BB, Should Students Learn Integration Rules? SIGSAM Bulletin 24/1, pp. 10 - 17, Jan. 1990.

Die Frage

Das White - Box / Black - Box Prinzip

Beispiel : Newton - Verfahren

Übung : Gröbner - Basen

Die Antwort soll / kann nicht einfach "Ja" oder "Nein" sein, sondern hängt von der Vermittlungsphase ab :

Gegeben ein bestimmtes mathematisches Thema, hängt die Antwort von der "Vermittlungsphase" (Lehrphase, Lernphase, Studierphase) ab :

Die "**White-Box" Phase** : Das Thema ist für den Studenten neu.

Die "**Black-Box" Phase** : Das Thema wurde vollständig verdaut (unterrichtet, gelernt, studiert).

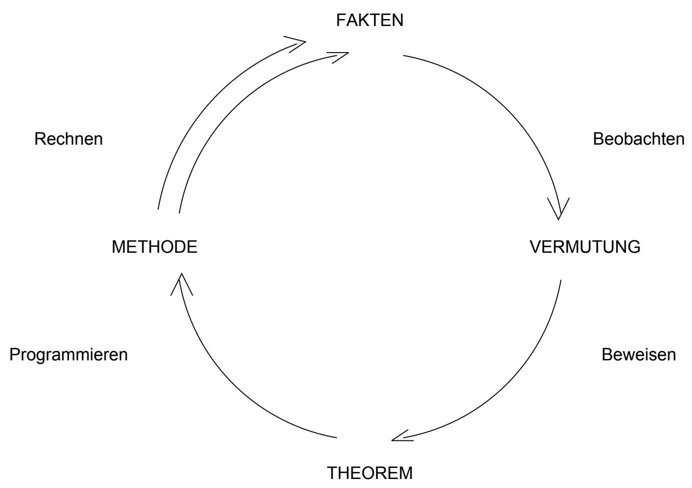
In der White - Box Phase : Die Anwendung von "Technologie" ist unsinnig.

In der Black - Box Phase : Die Nicht - Anwendung von "Technologie" ist unsinnig.

(Achtung : Man kann auch bewusst Black - Box / White - Box spielen! ... wenn man einmal eingesehen hat, dass White und Black verschieden sind.)

Typische Schritte in der White - Box Phase : die mathematische "Kreativitätsspirale"

In der White - Box Phase sind **folgende Schritte typisch :**



Die mathematische "Kreativitätsspirale"
(BB, in Vorträgen seit 1992)

Jeder Durchlauf in der Spirale "trivialisiert" einen Teil Mathematik.

Jeder Durchlauf in der Spirale bringt uns ein Stück "höher" in der Mathematik.

Das White - Box / Black - Box Prinzip ist rekursiv

In der White - Box Phase zu einem Thema können Begriffe, Sätze, Algorithmen aus der White - Box Phase von früheren ("niedereren") Themen im Black - Box Stil verwendet werden.

Die Begriffe, Sätze, Algorithmen, die in der jetzigen White - Box Phase zu einem Thema erarbeitet werden, können und sollen in späteren White - Box Phasen zu neuen ("höheren") Themen im Black - Box Stil verwendet werden.

Bemerkung : "niederer" und "höher" sind relative Begriffe!

Die Frage

Das White - Box / Black - Box Prinzip

Beispiel : Newton - Verfahren

Übung : Gröbner - Basen

Das Problem

Problem :

Gegeben ein system F von n nicht - linearen Gleichungen in n Unbekannten.

Gesucht ein n - Tuple von (reellen) Zahlen, die F lösen.

Wie sind die Gleichungen "gegeben" : durch "Ausdrücke", die n - Tuple in n - Tuple von Zahlen abbilden.

(Bemerkung : Wie verwenden im Folgenden Mathematica. Wir könnten jedes andere System verwenden, z.B. Maple, ...)

Black Boxes aus früheren Runden

Wir setzen die folgenden **Black Boxes** voraus :

. Lösen von linearen Gleichungen
. symbolisches Differenzieren elementarer Ausdrücken
. die Präsentation von Tangentialebenen
. und andere (z.B. "algebraische Manipulationen", "Evaluieren", ...)

White - Box für Newton : Ein Bespielproblem

Gegeben : Ein System von zwei (polynomialen) nichtlinearen Gleichungen :

$-9 + x1^2 + x2^2 = 0$

$-9 + (-1 + x1)^2 + 2\,x2^2 = 0$

Wir möchten reelle Zahlen a1, a2 finden, sodass

$-9 + a1^2 + a2^2 = 0$

$-9 + (-1 + a1)^2 + 2\,a2 = 0$

White - Box für Newton : Problemverständnis durch bekannte (als Black - Box verwendete) Verfahren

In diesem Beispiel könnte man, mit ein paar "**algebraischen Umformungen**", die Variablen sukzessive "eliminieren". Dieses Vorgehen ist in Mathematica als "Block Box" mit Namen "Solve" vorhanden. (In seiner allgemeinsten Form basiert es auf der Methode der Gröbner - Bases, siehe Übung.)

```
Solve[{-9 + x1^2 + x2^2 == 0, -9 + (-1 + x1)^2 + 2 x2^2 == 0}, {x1, x2}]
```

$$\left\{\left\{x1 \to -1-\sqrt{11},\ x2 \to -i\sqrt{1-2\left(-1-\sqrt{11}\right)}\right\},\ \left\{x1 \to -1-\sqrt{11},\ x2 \to i\sqrt{1-2\left(-1-\sqrt{11}\right)}\right\},\ \left\{x1 \to -1+\sqrt{11},\ x2 \to -\sqrt{-1+2\left(-1+\sqrt{11}\right)}\right\},\ \left\{x1 \to -1+\sqrt{11},\ x2 \to \sqrt{-1+2\left(-1+\sqrt{11}\right)}\right\}\right\}$$

Man kann auch Näherungswerte für die Lösungen anschauen, um einen "besseren Überblick" zu haben :

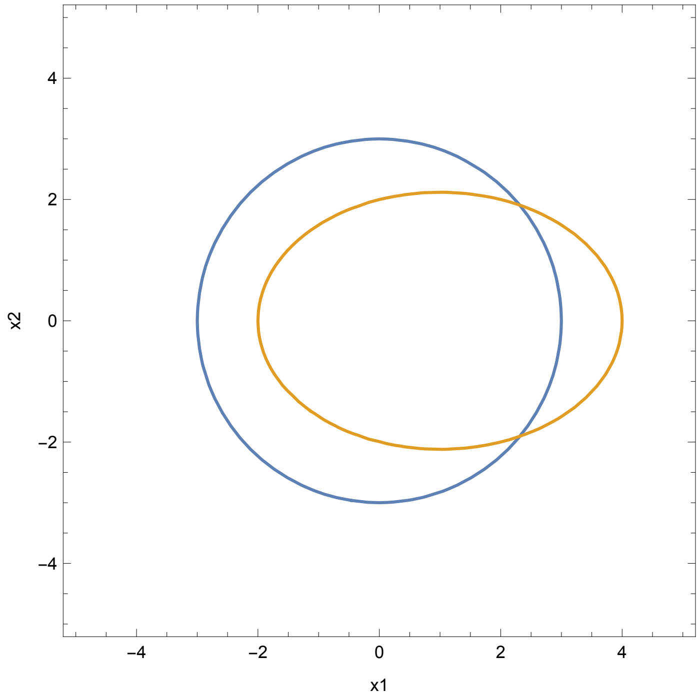
```
Solve[{-9 + x1^2 + x2^2 == 0, -9 + (-1 + x1)^2 + 2 x2^2 == 0}, {x1, x2}] // N
```

$$\{\{x1 \to -4.31662,\ x2 \to 0. - 3.10375\, i\},\ \{x1 \to -4.31662,\ x2 \to 0. + 3.10375\, i\},\ \{x1 \to 2.31662,\ x2 \to -1.90611\},\ \{x1 \to 2.31662,\ x2 \to 1.90611\}\}$$

White - Box für Newton : Problemverständnis durch bekannte (als Black - Box verwendete) Verfahren

Man könnte sich auch die Schnittlinien der zwei Polynome mit der (x1, x2) - Ebene "anschauen" :

```
ContourPlot[{ -9 + x1^2 + x2^2 , -9 + (-1 + x1)^2 + 2 x2^2},
 {x1, -5, 5}, {x2, -5, 5}, FrameLabel → {x1, x2}]
```



Welches bekannte (als Black Box verwendete) Verfahren steht hinter dem Plotten, d.h. dem "**Anschauen**"?

White - Box für Newton : Algebraische Umformungen stoßen an Grenzen

Bei diesem Beispiel kommt man mit "algebraischen Umformungen" nicht durch :

```
Sin[x1 x2] - x2 Sin[x1]  = 0
Sin[(1 - x1) (x2 - 1)] - x1 Sin[x2]  = 0
```

```
Solve[{Sin[x1 x2] - x2 Sin[x1] == 0, Sin[(1 - x1) (x2 - 1)] - x1 Sin[x2] == 0, {x1, x2}} ]
```

Solve: −x2 Sin[x1] + Sin[x1 x2] == 0 && Sin[(1 − x1) (−1 + x2)] − x1 Sin[x2] == 0 && {x1, x2} is not a quantified system of equations and inequalities.

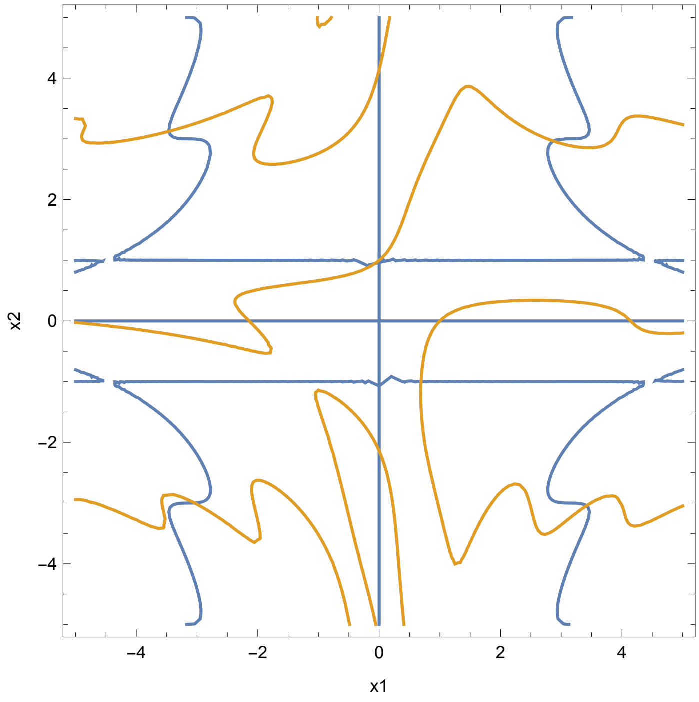
Sogar "Numerical Solve" ist hier nicht stark genug :

NSolve[{Sin[x1 x2] - x2 Sin[x1] == 0, Sin[(1 - x1) (x2 - 1)] - x1 Sin[x2] == 0, {x1, x2}}]

NSolve: −x2 Sin[x1] + Sin[x1 x2] == 0 && Sin[(1 − x1) (−1 + x2)] − x1 Sin[x2] == 0 && {x1, x2} is not a quantified system of equations and inequalities.

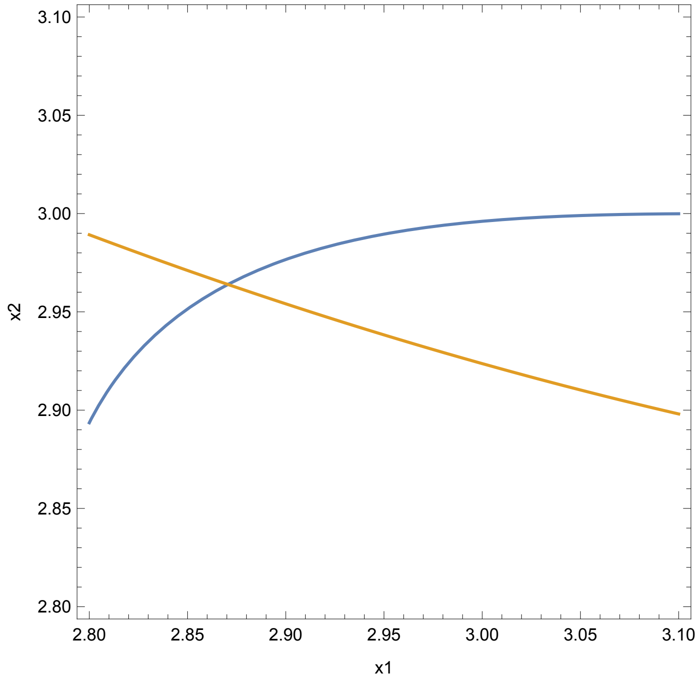
"Kein Wunder", wenn man sich den Plot "anschaut" (mit welchen Black - Box Verfahren?) :

ContourPlot[{   Sin[x1 x2] - x2 Sin[x1] , Sin[(1 - x1) (x2 - 1)] - x1 Sin[x2]}, {x1, -5, 5}, {x2, -5, 5}, FrameLabel -> {x1, x2}]



Man kann natürlich "experimentell" genauer hineingehen :

```
ContourPlot[{  Sin[x1 x2] - x2 Sin[x1] , Sin[(1 - x1) (x2 - 1)] - x1 Sin[x2]},
 {x1, 2.8, 3.1}, {x2, 2.8, 3.1}, FrameLabel -> {x1, x2}]
```



Mit gutem Raten für einen Anfangspunkt kann jetzt ein numerisches Verfahren, das die Information verwerten kann, zu einem sehr guten Ergebnis kommen :

```
FindRoot[{Sin[x1 x2] - x2 Sin[x1] == 0, Sin[(1 - x1) (x2 - 1)] - x1 Sin[x2] == 0},
 {{x1, 2.9}, {x2, 2.96}}]
```

```
{x1 → 2.87054, x2 → 2.96394}
```

In der Tat :

```
{Sin[x1 x2] - x2 Sin[x1], Sin[(1 - x1) (x2 - 1)] - x1 Sin[x2]} /.
 {x1 → 2.870539620280482`, x2 → 2.9639383645503554` }
```

$\{-6.66134\times10^{-16}, 3.33067\times10^{-16}\}$

Approximative Variante des Problems

Wie so oft, muss man sich mit der approximativen Lösung einer approximativen Version des Problems mit "heuristischer Hilfe" begnügen :

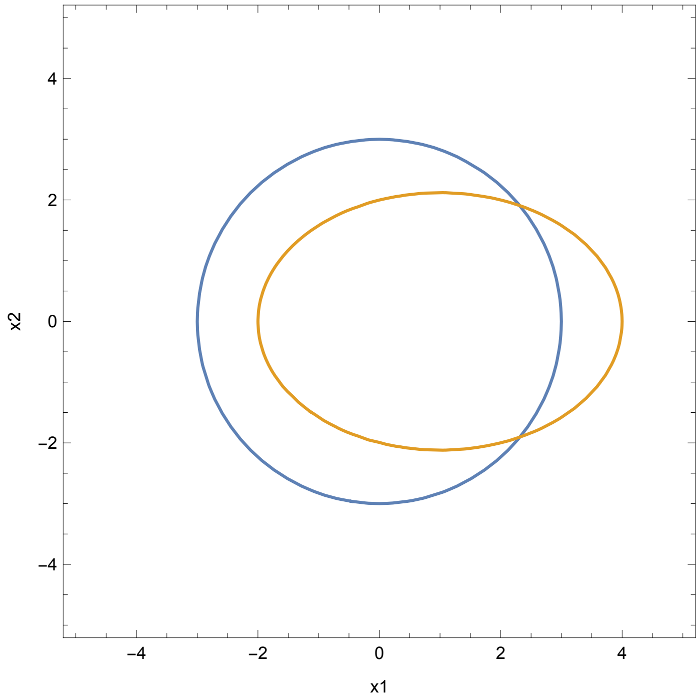
Problem :

Gegeben ein system F von n nicht - linearen Gleichungen in n Unbekannten und eine Fehlertoleranz ϵ und einen Anfangspunkt a.

Gesucht ein n - Tuple a von (reellen) Zahlen "in der Nähe von a0", die F mit "Fehler kleiner ϵ" lösen.

White - Box für Newton : Erraten eines Punkts in der Nähe einer Lösung

```
ContourPlot[{ -9 + x1^2 + x2^2 , -9 + (-1 + x1)^2 + 2 x2^2},
 {x1, -5, 5}, {x2, -5, 5}, FrameLabel → {x1, x2}]
```



Aus dem Plot raten wir, dass eine Lösung des Systems nahe beim Punkt b = (2.4, 2) ist.

Allerdings, wenn wir diese Zahlen in die Gleichungen einsetzen, erhalten wir "Residuen" von 0.75 and 0.96 statt 0 :

```
{ -9 + x1^2 + x2^2 , -9 + (-1 + x1)^2 + 2 x2^2} /. {x1 → 2.4, x2 → 2}
```

```
{0.76, 0.96}
```

Wenn wir mit der "approximativen Lösung" b nicht zufrieden sind, entsteht das Problem, wie wir ausgehend von b eine bessere und bessere und bessere ... approximative Lösung finden können.

Die Idee des Newton - Verfahrens zur Lösung dieses Problems : Linearisierung.

White - Box für Newton : Linearisierung

Idee der Linearisierung wird beim Newton - Verfahren wie folgt angewandt :

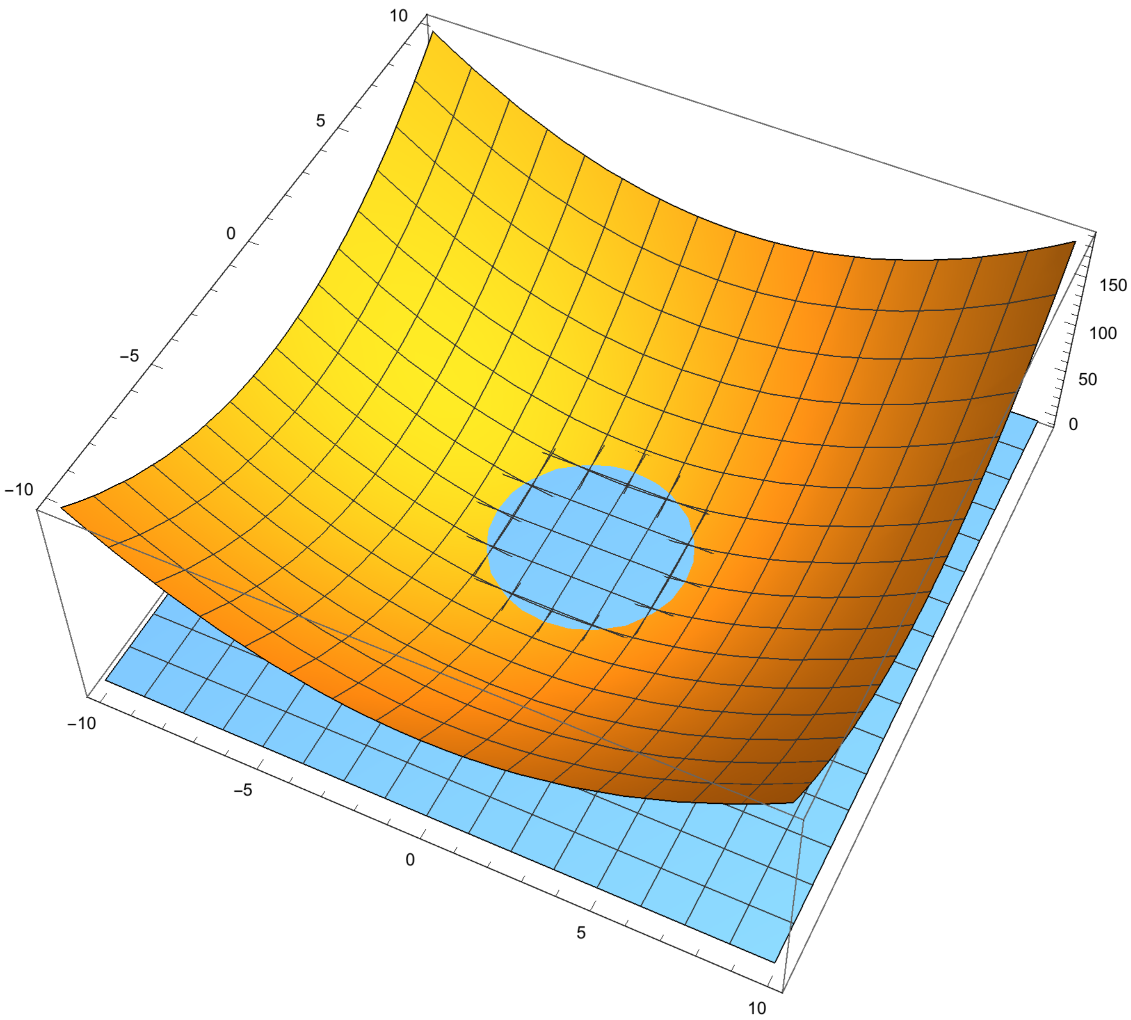
Die zwei linken Seiten (Polynome) p1, p2 des Gleichungssystems beschreiben (gekrümmte) Flächen S1, S2 im Raum, die sich mti der (x1, x2) - Ebene schneiden :

$p1 = -9 + x1^2 + x2^2$
$p2 = -9 + (-1 + x1)^2 + 2\, x2^2$

$-9 + x1^2 + x2^2$

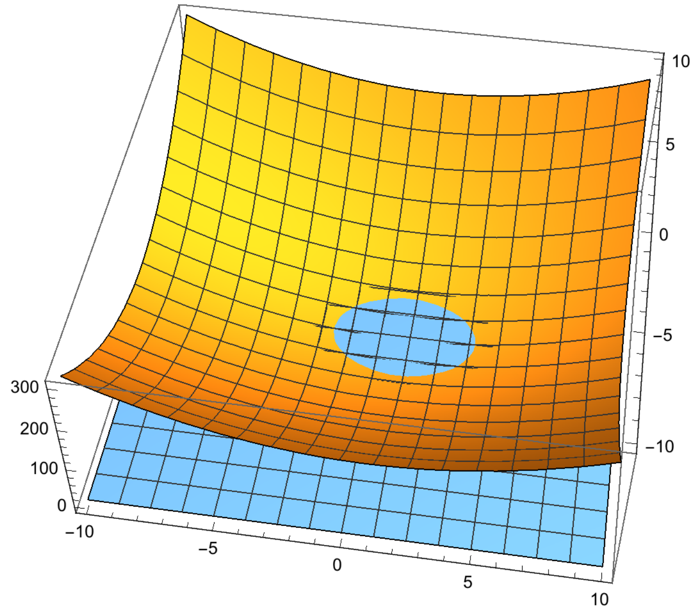
$-9 + (-1 + x1)^2 + 2\, x2^2$

```
Plot3D[{p1, 0}, {x1, -10, 10}, {x2, -10, 10}]
```



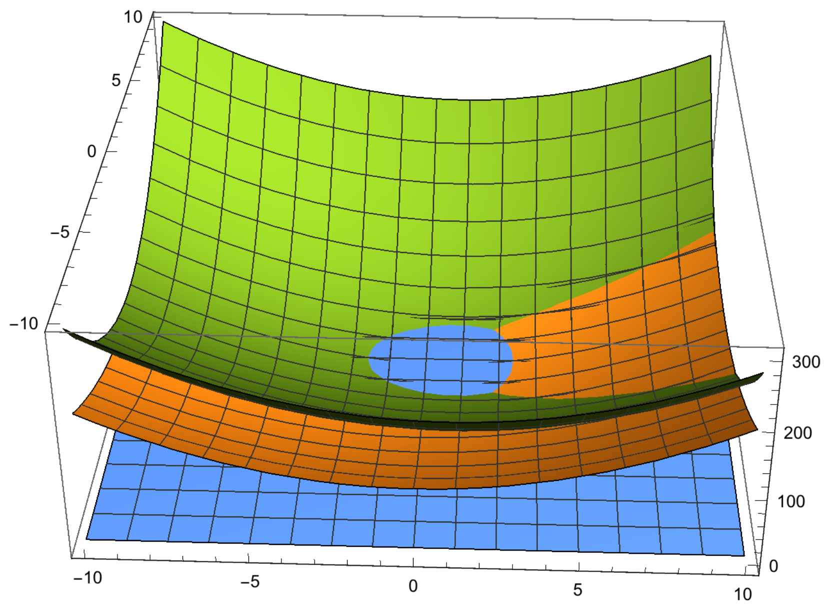
Die Fläche S1

```
Plot3D[{p2, 0}, {x1, -10, 10}, {x2, -10, 10}]
```

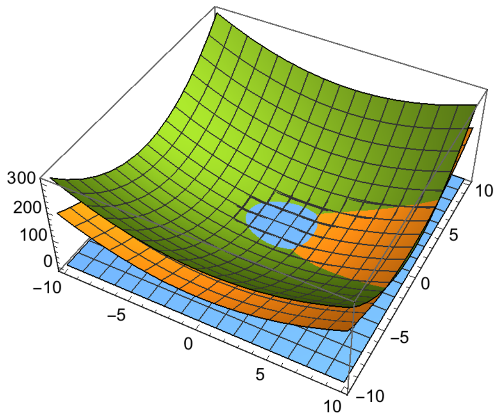


Die Fläche S2

```
Plot3D[{p1, 0, p2}, {x1, -10, 10}, {x2, -10, 10}]
```



White - Box für Newton : Linearisierung



Wir **starten** also vom Punkt b = (2.4, 2) auf der (x1, x2) - Ebene.

Seien **C1, C2, die ebenen Kurven**, die **durch Schneiden der Flächen** S1 und S2 mit der (x1, x2) - Ebene entstehen.

Am Punkt b = (2.4, 2) sind die Funktionswerte auf S1 und S2 != 0, nämlich 0.76 und 0.96.

Wenn wir der Fläche S1 vom Punkt (2.4, 2, 0.76) "abwärts folgten, würden wir auf der Kurve C1 ankommen.

Wenn wir der Fläche S2 vom Punkt (2.4, 2, 0.96) "abwärts folgten, würden wir auf der Kurve C2 ankommen.

Die Intersektion a* der Kurven C1, C2 (die wir aber nicht kennen) wäre eine Lösung des Systems.

Idee der Linearisierung : Ersetze die Flächen S1 und S2 durch ihre "lineare Approximationen" (die Tangentenflächen) T1 und T2 in den Punkten (2.4, 2, 0.76) bzw. (2.4, 2, 0.96).

Auch die Tangentenflächen T1 und T2 schneiden die (x1, x2) - Ebene. Nennen wir die Schnittgeraden (linear!) L1 und L2.

Der Schnittpunkt von L1 und L2 ist ein Punkt a = (a1, a2), der durch Lösen eines linearen Gleichungssystems gefunden werden kann.

Unter gewissen Bedingungen an p1, p2 (nämlich dass die Flächen in der Nähe von b "schön gekrümmt sind wie Schüsseln", d.h. "konkav", sind und in der Nähe von b die (x1, x2) - Ebene schneiden), liegt der Punkt a (bedeutend) näher beim unbekannten a* als b.

Wir rechnen nun die Residuen des Punktes a auf den Flächen S1 und S2.
Wenn wir noch nicht zufrieden sind (d.h. die Residuen größer als das gewünschte ϵ sind), nehmen wir den Punkt a als den neuen Punkt b und iterieren!

White - Box für Newton : Benutze frühere Black Boxes

Die Berechnung der Tangentenflächen (die die partiellen Ableitungen von p1 und p2 benötigen) werden in der "Jacobi - Matrix" zusammengefasst.

Die (symbolische) Berechnung von Ableitungen und das Lösen linearer Gleichungen werden wir nun hier als bekannte Black Boxes betrachten.

White - Box für Newton : Das Newton - Verfahren

Unter Verwendung dieser Black Boxes kann man dann den obigen Gedanken wie folgt beschreiben **(Newton - Verfahren)** (für n = 2) :

```
Clear[JacobianMatrix]
JacobianMatrix[{p1_, p2_}] := {{D[p1, x1], D[p1, x2]}, {D[p2, x1], D[p2, x2]}}
```

```
Clear[Newton] (* Global indeterminates x1, x2 *)
Newton[{p1_, p2_}, {a01_, a02_}, ∈_] :=
 Module[
  {J = JacobianMatrix[{p1, p2}], a1 = a01, a2 = a02, p1a, p2a, d1, d2},
  {p1a, p2a} = {p1 /. {x1 → a1, x2 → a2}, p2 /. {x1 → a1, x2 → a2}};
  While[Abs[p1a] > ∈ ∨ Abs[p2a] > ∈,
   Print[{a1, a2, p1a, p2a}];
   Ja = J /. {x1 → a1, x2 → a2};
   {d1, d2} = LinearSolve[Ja, {-p1a, -p2a}];
   {a1, a2} = {a1, a2} + {d1, d2};
   {p1a, p2a} = {p1 /. {x1 → a1, x2 → a2}, p2 /. {x1 → a1, x2 → a2}}];
  {a1, a2}
 ]
```

White - Box für Newton : "Kopfrechnen" des Newton - Verfahrens

Man könnte nun z.B. für die obigen Polynome p1, p2, Anfangspunkt (2.4, 2) und $\epsilon = 0.1$ das Verfahren "mit dem Kopf" (Hand?) rechnen, wobei man die Berechnung der Jacobi - Matrix und das Lösen des bei jeder Iteration auftretenden neuen linearen Gleichungssystems als Black Box auslagern darf. (Das "mit Hand Rechnen" dieser Black Boxes bringt in dieser Entwicklungs - oder Vermittlungsphase nichts!)

Der Übergang zu Black - Box für Newton : Langeweile und Programmieren

Ab einem gewissen Zeitpunkt wird das **langwelig** werden. Das ist der Zeitpunkt, in welchem man die Methode "programmiert" (die obige Beschreibung ist in der Tat bereits ein Programm in Mathematica) und in Zukunft nie mehr "händisch" durchführt, sondern immer dem "Oberidioten" (dem Computer) zur Ausführung überlässt.

Ab diesem Zeitpunkt kann und soll man das Verfahren also als Black Box verwenden.

Der Übergang von der White - Box Phase in die Black - Box Phase ist die aufregendste Phase in der Mathematik.

Dies sowohl in der Forschung (Erfindung) als auch in der Lehre.

Gute Lehre ist nachgestellte Forschung.

Mathematik lernen heißt die Fähigkeit zu erlernen, die White - Box Phase mit Intelligenz so lange zu betreiben, bis das entsprechende Gebiet durch ein Verfahren "trivialisiert" ist und in Zukunft durch Black Box Aufrufen eines Verfahrens erledigt werden kann.

Sowohl die Puristen als auch die Populisten verfehlen diesen entscheidenden Knackpunkt von "Mathematik".

Black - Box Newton : Enjoy

```
p1
p2
Newton[{p1, p2}, {a1, a2}, 0.01]
```

$-9 + x1^2 + x2^2$

$-9 + (-1 + x1)^2 + 2\, x2^2$

{2.4, 2, 0.76, 0.96}

{2.31765, 1.90882, 0.0150952, 0.0234083}

{2.31662, 1.90611}

$\text{Newton}\left[\{p1, p2\}, \{a1, a2\}, 10^{-12}\right]$

{2.4, 2, 0.76, 0.96}

{2.31765, 1.90882, 0.0150952, 0.0234083}

$\{2.31662, 1.90611, 8.40521 \times 10^{-6}, 0.0000157657\}$

$\{2.31662, 1.90611, 4.0794 \times 10^{-12}, 8.13216 \times 10^{-12}\}$

{2.31662, 1.90611}

$\text{Newton}\left[\{p1, p2\}, \{a1, a2\}, 10^{-12}\right]$ // Timing

{2.4, 2, 0.76, 0.96}

{2.31765, 1.90882, 0.0150952, 0.0234083}

$\{2.31662, 1.90611, 8.40521 \times 10^{-6}, 0.0000157657\}$

$\{2.31662, 1.90611, 4.0794 \times 10^{-12}, 8.13216 \times 10^{-12}\}$

{0.001127, {2.31662, 1.90611}}

Wow! Sehr schnelle Konvergenz!

Vergleiche :

```
Solve[{p1 == 0, p2 == 0}] // N // Timing
```

$\{0.040967, \{\{x1 \to -4.31662, x2 \to 0. - 3.10375\, i\}, \{x1 \to -4.31662, x2 \to 0. + 3.10375\, i\},$
$\{x1 \to 2.31662, x2 \to -1.90611\}, \{x1 \to 2.31662, x2 \to 1.90611\}\}\}$

Die nächste White - Box Phase :

Jeder Abschluss einer White - Box Phase durch eine Black - Box ergibt sofort dutzende Fragen für die nächste White - Box Phase (den nächsten Durchgan in der Kreativitätsspirale) :

In unserem Fall :

- Bedingungen für die Konvergenz mit Theorem und Beweis
- Geschwindikgeit der Konvergenz mit Theorem und Beweis
- Verallallgmeinerung auf allgemeines n
- Wie findet man Anfangspunkte?

...

Konklusion

Sei kein Purist!

Sei kein Populist!

Steig immer weiter in der Kreativitätsspirale durch White - Box über Black - Box zu White - Box über Black - Box zu ... ...

Das ist der Sinn der Mathematik : Denke "einmal" tief nach in der White - Box Phase, um dann in der Black - Box Phase unendlich oftmals nicht mehr denken zu müssen.

Es gibt beweisbar keine Grenze nach oben für den mathematischen Intellekt. ("Gödel")

Aber nicht jedes Ziel ist durch Intellekt zu erreichen, nicht jedes Problem durch Intellekt zu lösen. Die Erfahrung der Stille (der wache Nicht - Intellekt) ist genauso wichtig wie die Erfahrung des Intellekts.

Die Werbeeinschaltung

Neues Buch : Bruno Buchberger. Mathematik - Management - Meditation. Molden Verlag, Wien, 19, 90 €, 2016.

Erhältlich bei Buchhandlungen und bei www.amazon.de (Lieferzeit zwei bis vier Tage).

Aus der Ankündigung :

Mathematik die Kunst des Denkens
Management die Kunst des Handelns
Meditation die Kunst des Nicht - Denkens und Nicht - Handelns

Der vielfach ausgezeichnete Mathematiker, Computerwissenschafts - Pionier und Softwarepark - Gründer Bruno Buchberger kondensiert in diesem Buch seine Gedanken zu diesen zentralen Aspekten seines Lebens.Dafür wählt er nicht die Form einer klassischen Autobiografie, sondern setzt sich mit Fragen auseinander, die ihm in dieser oder ähnlicher Form im Lauf der Jahre in Interviews gestellt wurden.

Für Eilige formuliert er die Antworten als kurze, mitunter provokante Schlaglichter.Für Leser, die in die Tiefe gehen wollen, erarbeitet der Autor die Zusammenhänge in umfassenderen Diskursen und Anekdoten. E

Ein Buch für alle, die denken, handeln und sich manchmal nach dem Nicht - Denken und Nicht - Handeln sehnen.

Die Frage

Das White - Box / Black - Box Prinzip

Beispiel : Newton - Verfahren

Übung : Gröbner - Basen

Lösen nicht - linearer (polynomialer) Systeme durch "algebraische Umformungen"

Ein nicht - lineares polynomiales System (dritten Grades)

```
p1 = 2 - x^2 + y + x y
p2 = - x y + x^2 y
```

$2 - x^2 + y + x\,y$

$-x\,y + x^2\,y$

Lösen durch algebraisches Umformen :

```
Solve[{p1 == 0, p2 == 0}] // Simplify
```

$\left\{\{x \to 0,\ y \to -2\},\ \left\{x \to 1,\ y \to -\frac{1}{2}\right\},\ \left\{x \to -\sqrt{2},\ y \to 0\right\},\ \left\{x \to \sqrt{2},\ y \to 0\right\}\right\}$

Eliminieren von Variablen durch "algebraische Umformungen"

Geht das immer? Was steckt dahinter? Gröbner - Basen.

```
G = GroebnerBasis[{p1, p2}]
```

$\{2\,y + 5\,y^2 + 2\,y^3,\ -4\,y + 3\,x\,y - 2\,y^2,\ -6 + 3\,x^2 - 7\,y - 2\,y^2\}$

Die Gröbner - Basis des Polynomsystems hat die Variablen "eliminiert". Man kann die Gleichungen jetzt sukzessive lösen (wenn man Gleichungen beliebig hohen Grades in einer Variablen lösen kann!)

```
sol = Solve[2 y + 5 y^2 + 2 y^3 == 0]
```

$\{\{y \to -2\},\ \{y \to -\frac{1}{2}\},\ \{y \to 0\}\}$

```
G /. sol
```

$\{\{0,\ -6\,x,\ 3\,x^2\},\ \{0,\ \frac{3}{2} - \frac{3\,x}{2},\ -3 + 3\,x^2\},\ \{0,\ 0,\ -6 + 3\,x^2\}\}$

Was ist die Idee hinter den Groebner Basen?

Wir nehmen wieder an, dass das Behandeln linearer Systeme schon als Black Box vorhanden sei. Das heißt unter anderem : Man hat Verfahren, wie man Matrizen triangularisiert (z.B. durch Gauß) : In Mathematica heißt diese Black Box "RowReduce" :

$$\text{RowReduce}\left[\left\{\left\{1, 1, \frac{3}{2}, \frac{1}{2}, 0, 0\right\}, \{4, 5, 6, 0, 1, 0\}, \{7, 8, 9, 0, 0, 1\}\right\}\right]$$

$$\left\{\{1, 0, 0, 1, -2, 1\}, \{0, 1, 0, -2, 1, 0\}, \left\{0, 0, 1, 1, \frac{2}{3}, -\frac{2}{3}\right\}\right\}$$

$$\left\{\left\{1, 1, \frac{3}{2}, \frac{1}{2}, 0, 0\right\}, \{4, 5, 6, 0, 1, 0\}, \{7, 8, 9, 0, 0, 1\}\right\} \text{ // MatrixForm}$$

$$\begin{pmatrix} 1 & 1 & \frac{3}{2} & \frac{1}{2} & 0 & 0 \\ 4 & 5 & 6 & 0 & 1 & 0 \\ 7 & 8 & 9 & 0 & 0 & 1 \end{pmatrix}$$

$$\text{RowReduce}\left[\left\{\left\{1, 1, \frac{3}{2}, \frac{1}{2}, 0, 0\right\}, \{4, 5, 6, 0, 1, 0\}, \{7, 8, 9, 0, 0, 1\}\right\}\right] \text{ // MatrixForm}$$

$$\begin{pmatrix} 1 & 0 & 0 & 1 & -2 & 1 \\ 0 & 1 & 0 & -2 & 1 & 0 \\ 0 & 0 & 1 & 1 & \frac{2}{3} & -\frac{2}{3} \end{pmatrix}$$

Was ist die Idee hinter den Groebner Basen? Linearisieren in einem ganz anderen Sinne

Wir ordnen die "Potenzprodukte" lexikographisch und nehmen die Potenzprodukte als neue Variable. Hier machen wir das einmal bis zum Grad 5 :

```
{MacaulayRow[5]} // MatrixForm
```

```
( x[5, 5]  x[5, 4]  x[5, 3]  x[5, 2]  x[5, 1]  x[5, 0]  x[4, 5]  x[4, 4]  x[4, 3]  x[4, 2]  x
```

Wir verwenden hier folgende Codierung : x[i, j] steht für $x^i y^j$.

p1 und p2 haben dann folgende Gestalt :

```
p1M = MacaulayComposed[5, {{0, 0, 2}, {2, 0, -1}, {0, 1, 1}, {1, 1, 1}}]
p2M = MacaulayComposed[5, {{1, 1, -1}, {2, 1, 1}}]
```

```
{0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
 0, 0, 0, 0, 0, 0, 0, -1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 1, 2}
```

```
{0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, -1, 0, 0, 0, 0, 0, 0, 0}
```

Zusammen in einer "Koeffizientenmatrix" :

```
{MacaulayRow[5], p1M, p2M} // MatrixForm
```

```
( x[5, 5]  x[5, 4]  x[5, 3]  x[5, 2]  x[5, 1]  x[5, 0]  x[4, 5]  x[4, 4]  x[4, 3]  x[4, 2]  x[4, 1]  x[4, 0]  x[3, 5]  x[3, 4]  x[3, 3]  x[3, 2]  x[3, 1]  x[3, 0]  x[2, 5]  x[2, 4]  x[2, 3]  x[
     0        0        0        0        0        0        0        0        0        0        0        0        0        0        0        0        0        0        0        0        0
     0        0        0        0        0        0        0        0        0        0        0        0        0        0        0        0        0        0        0        0        0
```

Was ist die Idee hinter den Groebner Basen? Was passiert, wenn wir diese Matrix triangulieren?

```
RowReduce[{p1M, p2M}]
```

```
{{0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0,
  0, 0, 0, 0, -1, 0, 0, 0, 0, 0, 0, 0}, {0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
  0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, -1, 0, 0, 0, 0, 0, -1, -2}}
```

Da passiert noch nichts, weil die Matrix schon trianguliert ist. Und in beiden Polynomen kommen noch beide Variablen vor.

Was ist die Idee hinter den Groebner Basen? Wir nehmen jetzt noch "Shifts" der Polynome dazu :

Die Shifts entstehen durch Multiplizieren mit x und y (so oft wir wollen). Damit ändert sich das Glelchungssystem nicht. Warum nicht?

Wir machen das einmal bis zum fünften Grad :

```
p1Mx = ShiftX[5, p1M]
p1My = ShiftY[5, p1M]
p1Mxx = ShiftX[5, p1Mx]
p1Mxy = ShiftX[5, p1My]
p1Myy = ShiftY[5, p1My]
p1Mxxx = ShiftX[5, p1Mxx]
p1Mxxy = ShiftX[5, p1Mxy]
p1Mxyy = ShiftX[5, p1Myy]
p1Myyy = ShiftY[5, p1Myy]

p2Mx = ShiftX[5, p2M]
p2My = ShiftY[5, p2M]
p2Mxx = ShiftX[5, p2Mx]
p2Mxy = ShiftX[5, p2My]
p2Myy = ShiftY[5, p2My]
```

```
{0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
 0, -1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 1, 2, 0, 0, 0, 0, 0, 0}

{0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
 0, 0, 0, 0, 0, 0, -1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 1, 2, 0}

{0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, -1, 0, 0, 0, 0,
 1, 0, 0, 0, 0, 0, 1, 2, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0}

{0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
 -1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 1, 2, 0, 0, 0, 0, 0, 0, 0}

{0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
 0, 0, 0, 0, 0, -1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 1, 2, 0, 0}

{0, 0, 0, 0, 0, -1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0,
 1, 2, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0}

{0, 0, 0, 0, 0, 0, 0, 0, 0, 0, -1, 0, 0, 0, 0, 1,
 0, 0, 0, 0, 0, 1, 2, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0}

{0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, -1,
 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 1, 2, 0, 0, 0, 0, 0, 0, 0, 0}

{0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
 0, 0, 0, 0, -1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 1, 2, 0, 0, 0}
```

```
{0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
 1, 0, 0, 0, 0, 0, -1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0}

{0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, -1, 0, 0, 0, 0, 0, 0, 0, 0}

{0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0,
 -1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0}

{0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1,
 0, 0, 0, 0, 0, -1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0}

{0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, -1, 0, 0, 0, 0, 0, 0, 0, 0, 0}
```

Was ist die Idee hinter den Groebner Basen? Die Macaulay Matrix fünften Grades

```
MCMatrix5 = {p1M, p1Mx, p1My, p1Mxx, p1Mxy, p1Myy, p1Mxxx,
  p1Mxxy, p1Mxyy, p1Myyy, p2M, p2Mx, p2My, p2Mxx, p2Mxy, p2Myy}
```

```
{{0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, -1,
  0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 1, 2}, {0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
  0, 0, 0, 0, 0, -1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 1, 2, 0, 0, 0, 0, 0, 0},
 {0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, -1, 0, 0,
  0, 0, 1, 0, 0, 0, 0, 0, 1, 2, 0}, {0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, -1,
  0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 1, 2, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0},
 {0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, -1, 0, 0, 0, 0, 1, 0, 0,
  0, 0, 0, 1, 2, 0, 0, 0, 0, 0, 0, 0}, {0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
  0, 0, 0, 0, 0, 0, 0, 0, 0, -1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 1, 2, 0, 0},
 {0, 0, 0, 0, 0, -1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 1, 2, 0, 0, 0, 0, 0, 0, 0,
  0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0}, {0, 0, 0, 0, 0, 0, 0, 0, 0, 0, -1, 0,
  0, 0, 0, 1, 0, 0, 0, 0, 0, 1, 2, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0},
 {0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, -1, 0, 0, 0, 0, 1, 0, 0, 0,
  0, 0, 1, 2, 0, 0, 0, 0, 0, 0, 0, 0}, {0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
  0, 0, 0, 0, 0, 0, 0, 0, -1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 1, 2, 0, 0, 0},
 {0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0,
  0, 0, 0, -1, 0, 0, 0, 0, 0, 0, 0}, {0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
  0, 0, 0, 0, 1, 0, 0, 0, 0, 0, -1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0},
 {0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0,
  0, 0, -1, 0, 0, 0, 0, 0, 0, 0, 0}, {0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0,
  0, 0, 0, 0, -1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0},
 {0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, -1, 0, 0,
  0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0}, {0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
  0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, -1, 0, 0, 0, 0, 0, 0, 0, 0, 0}}
```

Was ist die Idee hinter den Groebner Basen? Jetzt triangulieren wir diese Matrix :

```
MCMatrix5R = RowReduce[MCMatrix5]
```

$$\{\{0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,$$
$$0, 0, 0, 0, -4, 0, 0, 0, -4, -8, 0\}, \{0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0,$$
$$0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, -\frac{2}{3}, -\frac{4}{3}, 0\},$$
$$\{0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,$$
$$0, 0, 0, 0, 0, 0, 0, -\frac{8}{3}, -\frac{22}{3}, -4\}, \{0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,$$
$$0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, \frac{1}{3}, \frac{2}{3}, 0\},$$
$$\{0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0,$$
$$0, 0, 0, 0, 0, 0, 0, -\frac{2}{3}, -\frac{4}{3}, 0\}, \{0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,$$
$$0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, -2, 0, 0, 0, -\frac{4}{3}, -\frac{8}{3}, 0\},$$
$$\{0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0,$$
$$0, 0, 0, 0, 0, 0, 0, 0, -\frac{1}{6}, -\frac{1}{3}, 0\}, \{0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,$$
$$0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, \frac{1}{3}, \frac{2}{3}, 0\},$$
$$\{0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0,$$
$$0, 0, 0, 0, 0, 0, 0, -\frac{2}{3}, -\frac{4}{3}, 0\}, \{0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,$$
$$0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, -\frac{2}{3}, -\frac{7}{3}, -2\},$$
$$\{0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1,$$
$$0, 0, 0, 0, 0, 1, 0, -\frac{31}{6}, -\frac{7}{3}, 0\}, \{0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,$$
$$0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, -\frac{1}{6}, -\frac{1}{3}, 0\},$$
$$\{0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,$$
$$0, 0, 1, 0, 0, 0, 0, 0, \frac{1}{3}, \frac{2}{3}, 0\}, \{0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,$$
$$0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, -\frac{2}{3}, -\frac{4}{3}, 0\},$$
$$\{0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,$$
$$0, 0, 0, 0, 0, 0, 0, 1, \frac{5}{2}, 1, 0\}, \{0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,$$
$$0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\}\}$$

```
Prepend[MCMatrix5R, MacaulayRow[5]] // MatrixForm
```

x[5, 5]	x[5, 4]	x[5, 3]	x[5, 2]	x[5, 1]	x[5, 0]	x[4, 5]	x[4, 4]	x[4, 3]	x[4, 2]	x[4, 1]	x[4, 0]	x[3, 5]	x[3, 4]	x[3, 3]	x[3, 2]	x[3, 1]	x[3, 0]	x[2, 5]	x[2, 4]	x[2, 3]	x[
0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	
0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	

G

$$\left\{2\,y + 5\,y^2 + 2\,y^3,\ -4\,y + 3\,x\,y - 2\,y^2,\ -6 + 3\,x^2 - 7\,y - 2\,y^2\right\}$$

Die Gröbner - Basis findet sich in der "Contour" der triangulierten Matrix!

Gröbner - Basen durch Shiften, Triangulieren und Contour feststellen

BB. Miscellaneous Results on Gröbner - Bases for Polynomial Ideals II. Technical Report 83 - 1, Department of Computer And Information Sciences, University of Delaware, 1983.

Es war aber die Frage offen, wieviele Shifts der Anfangspolynome man in die Macaulay - Matrix nehmen muss, damit in der Contour der triangulierten Matrix die Gröbner - Basis erscheint.

Die Antwort hast erst **M. Wiesinger in ihrer Dissertation 2015** gegeben (unter Verwendung von Schranken von G. Hermann und Dubé für ähnliche Probleme) : Wenn man die Shifts bis zu folgendem Grad dazu nimmt, kann man sicher sein :

$$2\left(\frac{d^2}{2}+d\right)^{2^{n-1}}+\sum_{j=0}^{n-1}(r\,d)^{2^j}.$$

d ... Grad der Anfangspolynome, n ... Anzahl der Variablen, r ... Anzahl der Anfangspolynome.

Diese Schranke ist aber viel zu hoch für die Praxis. Im Beispiel genügt Grad 3, die Formel liefert aber Grad 150 !!.

Gröbner - Basen durch "S - Polynome"

Eine ganz andere Antwort durch BB, Dissertation 1965 : Einführung des Begriffs der Gröbner - Basen. Shifts nur so weit notwendig dazu nehmen. **Dynamisches Terminationskriterien ("S - Polynome"),** das bei so kleinen Graden wie möglich und notwendig gecheckt wird. Es braucht einen Terminationsbeweis, dass dieses Verfahren immer abbricht.

Das ist die Methode, die heute (immer noch) in den meisten mathematischen Softwaresystemen angewandt wird. Aber immer mehr auch der Macaulay - Ansatz mit Heuristiken, wann man aufhören kann.

Gröbner - Basen als Black - Box einige "Spiralwindungen" weiter oben :

Symbolic Solution for Linear Boundary DE Problems (M. Rosenkranz, BB et al. since 2003)

Example : Given $f \in C^\infty[0, 1]$,
find $u \in C^\infty[0, 1]$ such that :

$$\boxed{\begin{array}{l} u'' = f \\ u(0) = u(1) = 0 \end{array}}$$

Solution via Green's Function

Traditional way of writing solution :

$$u(x) = \int_0^1 g(x, \xi) f(\xi) \, d\xi.$$

In the above example :

$$g(x, \xi) = \begin{cases} (x-1)\,\xi & \xi \le x \\ x\,(\xi - 1) & \xi > x \end{cases}$$

Existing Support in Computer Algebra Systems

Coverage in Mathematica and Maple unsystematic & unpredictable.

Solved neither by Mathematica nor by Maple :

$$\boxed{\begin{array}{l} u''' - (e^x + 2)\, u'' - u' + (e^x + 2)\, u = f \\ u(0) = u(1) = u'(1) = 0 \end{array}}$$

How are Gröbner Basis Used for DEs?

First approach : differential expressions of the DE are considered as polynomials (in a suitable domain) and the Gröbner basis of these expressions allow to draw some conclusions on the solutions.

Second approach : differential equations are transformed into algebraic equations, which are then solved with the Gröbner basis approach.

Our approach is very different : Gröbner bases are used on the "meta level".

The Groebner Basis of Relations Between Differentiation and Integration

Theorem (Rosenkranz 2003) : The axioms that describe the interaction of the differential and the integration operators form a Groebner basis in an (abstract) polynomial domain and the reduction of differential operators w.r.t. this Groebner basis gives the solution of the corresponding boundary value problem in the form of Green' s functions.

$\tilde{f}\, f \to \tilde{f} \cdot f$	$\partial f \to \partial \cdot f + f\, \partial$	$\int f \int \to \left(\int \cdot f\right) \int - \int\left(\int \cdot f\right)$
$\tilde{\varphi}\, \varphi \to \varphi$	$\partial\, \varphi \to 0$	$\int f\, \partial \to f - \int(\partial \cdot f) - (E \cdot f)\, E$
$\varphi f \to (\varphi \cdot f)\, \varphi$	$\partial \int \to 1$	$\int f\, \varphi \to \left(\int \cdot f\right) \varphi$

The proof is done automatically! (By using the BB 1965 criterion for Gröbnerianithy computing all S - poly reductions automatically.)

The method for solving LBV based on this Theorem is integrated in the THEOREMA System. (PhD thesis of Loredana Tec.)

The moethod is completely symbolic and the first general method for all linear boundary value problems.

Note : The solution is given relative to a system of fundamental solutions.

The Simple Example

```
GreensOperator[u'' = f, u(0) = 0, u(1) = 0]
```

$$g(x, \xi) = \begin{cases} (x-1)\,\xi & \xi \le x \\ x\,(\xi-1) & \xi > x \end{cases}$$

A Third Order Example

$$\text{GreensOperator}\left[u''' - \left(e^{x}+2\right) u'' - u' + \left(e^{x}+2\right) u = f,\ u(0) = 0,\ u(1) = 0,\ u'(1) = 0\right]$$

$$\begin{cases} \frac{-1}{2}\,(-1+e)^{-2}\, e^{-1\,e+(-1)\,x+(-2)\,\xi} \Big(-2\,e^{e} + 2\,e^{e^{x}} + 2\,e^{1+e} + (-1)\,e^{2+e} + \\ \quad (-2)\,e^{e^{x}+x} + e^{e+2\,x}\Big) \left(-1+e^{\xi}\right)^{2} & \Leftarrow\ \xi \le x \\ \frac{1}{2}\,(-1+e)^{-2}\, e^{-1\,e+(-1)\,e^{\xi}+(-1)\,x+(-2)\,\xi} \left(-1+e^{x}\right) \Big(-2\,e^{e+e^{x}} + \\ \quad 4\,e^{1+e+e^{x}} + \\ \quad (-2)\,e^{2+e+e^{x}} + (-2)\,e^{1+e+e^{\xi}} + e^{2+e+e^{\xi}} + 2\,e^{e^{x}+e^{\xi}} + (-2)\,e^{1+e+e^{\xi}+x} + \\ \quad e^{2+e+e^{\xi}+x} + 2\,e^{e^{\xi}+\xi}\left(e^{e} + (-2)\,e^{e^{x}} + e^{e+x}\right) + (-1)\,e^{e^{\xi}+2\,\xi} \\ \quad\quad \left(e^{e} + (-2)\,e^{e^{x}} + e^{e+x}\right)\Big) & \Leftarrow\ x \le \xi \end{cases}$$

Konklusion

Grundstrategien wie "Linearisieren" kann man in ganz verschiedener Weise anwenden.

Die automatische Kombination von Grundstrategien : Ein heißes Forschungsgebiet "sehr weit oben in der Spirale".

Das ist mein aktuelles Forschungsgebiet, siehe Theorema : Automated Reasoning, Automated Algorithm Synthesis.

Übung 1:

Wenn Sie ein mathematisches Softwaresystem dabei haben :

Mit einfachen Polynomen Gröbner - Basen Algorithmus aufrufen und mit Shifts und Triangulieren experimentieren. Vermutungen generieren über den notwendigen Grad.

Übung 2:

Andere Themen in der Mathematik (aus Schulmathematik) : Durchwanderen der White - Box Phase bis zur Black - Box Phase.

Hilfsalgorithmen (für diesen Vortrag)