

Wie die Mathematik zum Computer kam

Abschiedsvorlesung, 10. Dez. 1996 Audi Max ETH

Wie also kam die Mathematik zum Computer? Nun, wie die Jungfrau zum Kind, also nicht ganz unschuldig, aber schon ein bisschen über-raschend und die Lebensumstände verändernd.

Nicht ganz unschuldig: denn die Triebkräfte, welche die Mathematik bewegen, haben auch zum Computer geführt.

Die Mathematik will Fragestellungen und Methoden zu deren Lösung erkennen, formulieren und schliesslich auf der ganzen Breite der Wissenschaften anwenden. Sie will Zusammenhänge knüpfen zwischen ursprünglich disparaten Fragekreisen der Mathematik. Und schliesslich will jede Generation aufs neue einen Überblick über die Gesamtheit der Mathematik gewinnen und neue Gewichte setzen.

Ohne Zweifel hat der Computer auch, wie wir sagten, die Lebensumstände der Mathematik verändert. Auch davon soll hier die Rede sein. Es gibt also eine Geschichte zu erzählen; und hier sind die Themen:

Es wird eine Geschichte über die geistigen Quellen der Idee des Computers und seiner Zweckbestimmungen, über utopische Maschinen, künstliche Intelligenz und universale Theorien. Und schliesslich sprechen wir auch zum fortschreitenden Einfluss des Computers auf die Mathematik, ihre Methoden, Inhalte, ja selbst ihre Gestalt. Es soll auch etwas eine persönliche Geschichte sein, denn sie fällt ja gerade in die Zeit meines Aktivdienstes an der Logik, der Mathematik und der Informatik.

Die Idee des Computers schöpft aus zwei Quellen, der Logik und der Rechenkunst.

Aristoteles (384 - 322) gilt als der Vater der formalen Logik. In ihr brachte er die Gesetze des Denkens in die Form des regelgerechten Handhabens symbolischer Ausdrücke.

Die Aristotelische Logik befriedigte die Schulmänner des Triviums über zweitausend Jahre lang; sie gewannen ihr das Systematische, Abschliessende, ja Rechnerische ab. Die formale Logik des Stagiriten beflügelte aber auch alle Sorten von Schwärmern. Diese sahen in ihr das Rituelle, eigentliche Beschwörungs-Formeln der Wahrheit. Beschwörungen, die sozusagen maschinell, automatisch, erzeugt werden können. So ein logisch-mechanischer Schwärmer war Raymondus **Lullus**, der im 14. Jahrhundert mit einer formallogischen Vorrichtung die Ungläubigen in Agadir von der Wahrheit des Christentums überzeugen wollte. Er wurde gesteinigt. Einer aristotelischen Maschine begegnet man noch im letzten, und ich meine besten, Roman von Umberto **Eco**, *L'Isola del Giorno Prima*.

Soweit die Logik und ihre ersten Maschinen. Die Rechenkunst ist naturgemäss viel älter, hat auch schon früh zu Maschinen, Automaten Ansporn gegeben. Am bekanntesten sind wohl die Automaten des **Heron** von Alexandrien (3.Jh.v.Chr.) und die bronzenen Analogie-Rechengeräte der griechischen Astronomen.

Die moderne Geschichte des Computing beginnt damit, dass die beiden Stränge, Logik und Rechenkunst, zusammengeführt wurden. So etwas braucht ein Genie, und ich meine, das war **Leibniz** (1646 - 1716).

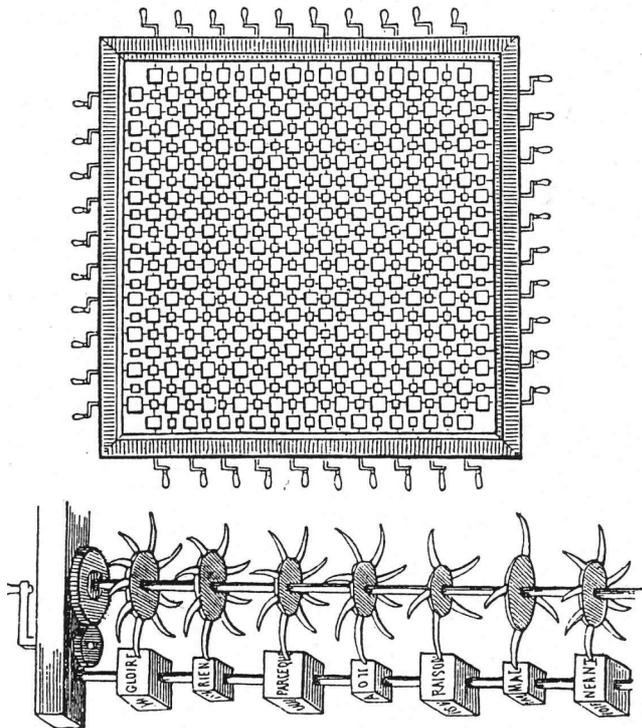
Ihm schwebte eine universale Rechenkunst, ein mechanisierbarer Universal-Kalkül vor. Alles inhaltliche Überlegen würde in eine formale Sprache gegossen und alle Probleme vom *calculus ratiocinator* unwidersprechbar

erledigt. Leibniz hatte durchaus konkrete Vorstellungen für die Mechanisierung, hat er doch 1671 eine arithmetische Rechenmaschine gebaut. Das grosse Projekt aber blieb Utopie, musste Utopie bleiben, und wurde entsprechend belächelt.

So zum Beispiel von Jonathan **Swift** (1667 - 1745), der sie in Gullivers Travels (1727) satirisierte.



Sie sehen hier (in einem Holzschnitt von Grandville) das Porträt des Erfinders einer Maschine, mit der man alle Wissenschaften, insbesondere auch die ganze Mathematik in mechanischer Weise erzeugen kann.



Auf den Klötzchen stehen Formel- und Sprachfragmente. Hilfskräfte drehen die Kurbeln in zufälliger Weise. Rund herum stehen Mathematiker, die feststellen, ob der entstandene Text ein Beweis eines mathematischen Satzes sei oder nicht. Falls er es ist, so wird das Resultat in ein Buch eingetragen, dem Hauptbuch der Mathematik.

Jonathan Swift satirisierte in Gullivers Reise nach Laputa die zeitgenössische Royal Society. Wie schon Aristophanes' Karikatur der Pythagoräer zweitausend Jahre vorher, ist auch diese denkbar ungerecht. Kaum jemals in der Geschichte der Wissenschaft gab es einen Kreis von Gelehrten so verschiedener Herkunft und Temperaments wie in der Royal Society des späten 17. Jahrhunderts, Gelehrte die einander so ernsthaft begegneten und

aufmerksam zuhören. Wie gerne wäre ich damals dabei gewesen, auch ohne die laputische Universalmaschine.

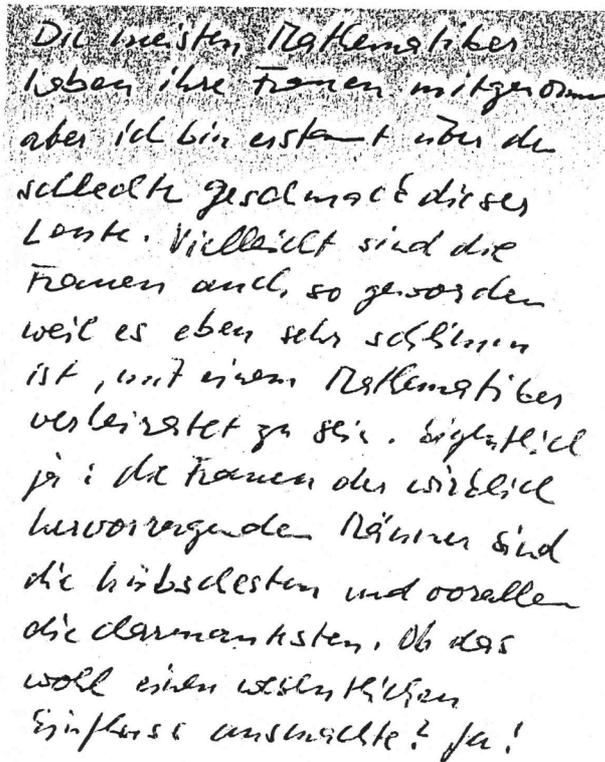


Erst im 20.ten Jahrhundert wurde die Zeit reif für den Computer. Einer seiner geistigen Väter war John **von Neumann** (1903 - 1957), eine Mathematiker-Figur, die mit Leibniz in vielem Ähnlichkeit aufweist.

Wie nur wenige in seiner Zeit, den 30er und 40er Jahren, vereinte er in sich, wie früher Leibniz, die beiden Stränge: Die neue, die mathematische Logik, und den Aufbruch ins programmgesteuerte numerische Rechnen.

Im Jahre 1954 fand der internationale Mathematikerkongress in Amsterdam statt. Ich reiste als junger Student hin, um mir Rat zu holen für aussichtsreiche Forschungsthemen. In jugendlicher Unverfrorenheit suchte ich den Logiker Alfred **Tarski**, den Gruppentheoretiker Richard **Brauer** und andere auf; und eben auch von Neumann. Ihn sprach ich zwar auf Logik und Computing an, erwähnte meine Lehrer Bernays und Stiefel. Er aber meinte, ich würde sicherer und aussichtsreicher fahren mit Operatoren-Algebren. Das habe ich aber denn doch nicht gemacht.-- Dieser Kongress war wohl mein erster

breiterer Kontakt mit der mathematical community, und ich muss schon sagen, das war ein besonders Völkchen, ist es noch heute. Ein Brief an meine Freundin Margaret drückte das so aus:



Die meisten Mathematiker
leben ihre Frauen mitgenommen
aber ich bin erstaunt über die
schlechte geschmacke dieses
Lente. Vielleicht sind die
Frauen auch, so geworden
weil es eben sehr schwierig
ist, mit einem Mathematiker
verheiratet zu sein. Bspw. weil
ja: die Frauen der wirklich
hervorragenden Männer sind
die hübschesten und overall
die charmantesten. Ob das
wohl einen evolutionären
Einfluss ausübt? ja!

Nun hat sich ja seit Gullivers Reisen das Verständnis dessen, was eine Maschine ist und wie man mit ihr Mathematik erfindet, sehr geändert. Allerdings ist man noch nicht soweit, als dass man damit etwa Newtons Gesetze finden könnte:

Sehr geändert – vorallem aber vertieft – hat sich auch das logische Instrumentarium. Und so ist der Traum von einer mathematischen Universalmaschine wieder zum Leben erwacht, ist aber nun konfrontiert mit Methoden für Unmöglichkeitsbeweise. Hier konkret:
Man kann beweisen, es gibt kein Verfahren, welches die Mathematik vollständig mechanisiert.

Zum Glück, meine ich, und nicht überraschend zudem. Denn Mathematik ist ein zentrales Kulturgut, und wer würde schon glauben, dass Kultur programmierbar sei. Es sei denn ein Ideologe. Im Fall der Mathematik ist es allerdings nicht die Geschichte, welche die Unhaltbarkeit dirigistischer Kultur - Programme ans Licht bringt, sondern ein Beweis. Seit dem ursprünglichen Beweis (durch Gödel 1931) sind darüber ganze Bücher und unzählige, auch populäre Darstellungen erschienen. Ich gestatte mir, hier eine anzufügen, welche kaum noch mehr vereinfacht werden kann, und in welcher Logik und Informatik schön ineinandergreifen.

Buch der Probleme : p_1, p_2, p_3, \dots

Buch der Verfahren : $v_1(\cdot), v_2(\cdot), \dots$

$v_i(j) \downarrow$: Aussage, dass Verfahren v_i die Lösung des Problems p_j findet

$v_i(j) \uparrow$: Aussage, dass Verfahren v_i die Lösung des Problems p_j nicht findet

Protokollbuch der Mathematik: $\alpha_1, \alpha_2, \alpha_3, \dots$

enthält alle von der Mathematik je bewiesenen Resultate.

$b(\cdot)$: hypothetisches Verfahren, welches alle Resultate der Mathematik ("das Protokollbuch") erzeugt.

In der formalen Sprache der Logik können wir alle möglichen Fragestellungen der Mathematik der Reihe nach aufzählen, der pythagoräische Lehrsatz, z.B. könnte etwa die Nummer j haben. Unter Benutzung der Programmiersprachen der Informatik können wir alle denkbaren Methoden

der Problemlösung aufzählen. Das i -te Verfahren bringe etwa einen der Beweise des Satzes von Pythagoras. Diese Situation bezeichnen wir mit dem Pfeil nach unten. Falls das i -te Verfahren die Lösung des j -ten Problems aber nicht findet, so bezeichnen wir diesen Sachverhalt mit dem Pfeil nach oben. Im (hypothetischen) Protokollbuch der Mathematik seien alle Fragen aufgezählt, welche die Mathematik je positiv entscheiden wird; wir setzen natürlich voraus, dass das Buch keine widersprüchlichen Aussagen enthält.

$b(\beta) \downarrow$: Aussage, dass die mathematische Aussage β von b erzeugt wird.

Annahme : b erzeugt entweder β oder dessen Verneinung $\neg\beta$ aber nicht beide.

$d(x)$: davon abgeleitetes Verfahren

$$u := 1; \text{ while } u \neq 0 \text{ do } u := \begin{cases} 1 & \text{if } b(v_x(x) \downarrow) \downarrow \\ 0 & \text{if } b(v_x(x) \uparrow) \downarrow \end{cases}$$

$d(\cdot)$ ist ein $v_m(\cdot)$ für ein gewisses m

$d(m) \downarrow$ falls $b(v_m(m) \uparrow) \downarrow$, falls $v_m(m) \uparrow$, also $d(m) \uparrow \#$

$d(m) \uparrow$ falls $b(v_m(m) \downarrow) \downarrow$, falls $v_m(m) \downarrow$, also $d(m) \downarrow \#$

Was wäre nun eine vollständige Mechanisierung der Mathematik? Doch wohl eben auch ein Verfahren b , das nun aber das Problem behandelt, ob eine Aussage im Protokoll der Mathematik je erscheinen wird. Bemerken Sie, dass eine solche fragliche Aussage ebenfalls die Form einer Erfolgsbehauptung hat, für eben dieses Verfahren b : Ein mathematisches Resultat erscheint im Protokoll, falls b auf ihm zum Erfolg führt. Wäre nun b ein vollständiges Verfahren, so würde b auf jeder Aussage selbst oder dann auf ihrem Gegenteil zum Erfolg führen.

Diese Bemerkung kann man dazu benützen, um aus dem hypothetischen Verfahren b das hier angegebene Verfahren $d(x)$ zu konstruieren. Dieses versucht, bei gegebenem x einen Zahlenwert u zu berechnen und beginnt, indem es u gleich 1 setzt. Dann geht das Verfahren in einen Repetitionsschleife. Solange u verschieden ist von Null so wird u entsprechend der angegebenen Bedingung verändert. Weil b nach Annahme vollständig ist, können wir im Programm d die Fallunterscheidung ganz rechts in der Tat durchführen.

Das Verfahren d ist der Kernpunkt des Beweises. Existiert es, so folgt ein Widerspruch, wie wir gleich zeigen werden. Also existiert es nicht, und also auch nicht das vollständige Verfahren b auf dem es fusst.

Nun sind ja Beweise, dass etwas nicht gemacht werden kann, dem mathematischen Laien - und nicht nur dem - eher fremd und überzeugen ihn selten ganz voll. Dazu eine kleine Geschichte: Einer der berühmtesten Geiger der Zwischenkriegszeit in Deutschland, Freund unserer Familie, hat im hohen Alter versucht, den Winkel zu dreiteilen. Nun weiss, in Anführungszeichen, die Mathematik, dass die Winkeldreiteilung mit Zirkel und Lineal nicht geht. Doch weder ich noch Heisenberg, mit dem er damals auch korrespondierte, konnten dies dem alten Herrn glaubhaft machen.

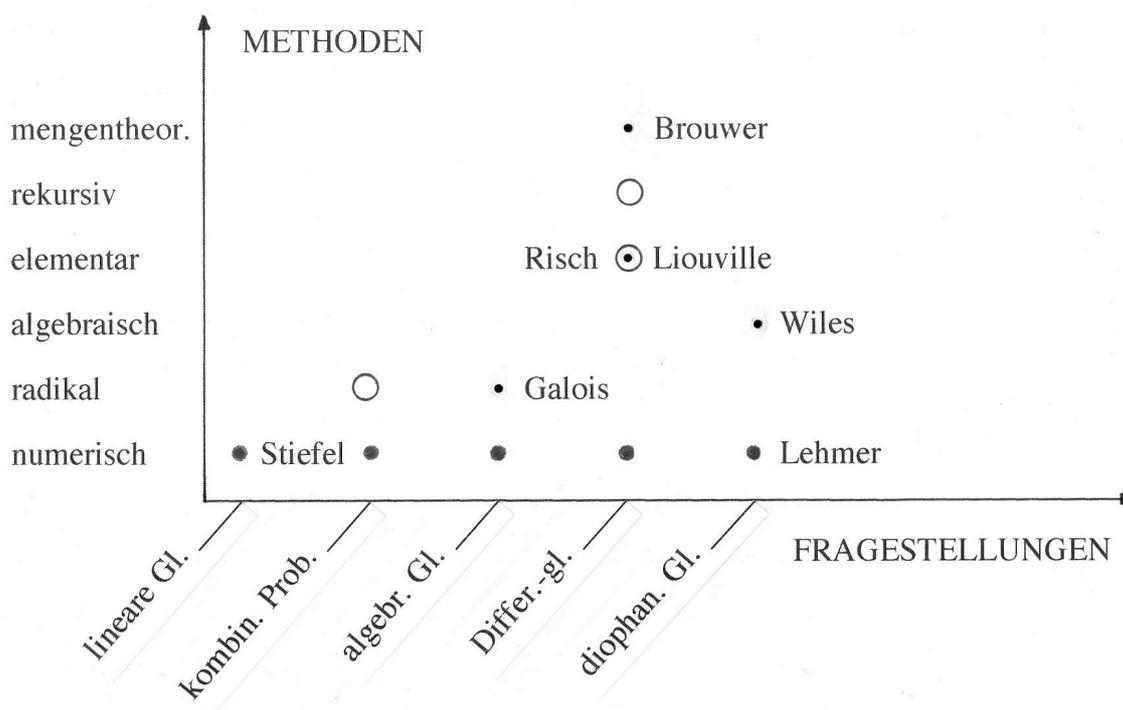
Nun also zurück zu meinem Versuch, Sie von der Unmöglichkeit der vollständigen Computerisierung der Mathematik zu überzeugen: Angenommen, die Mathematik wäre mechanisierbar, so gäbe es eben das Verfahren d und dieses wäre eines unter der aufgezählten Totalität aller Verfahren, sagen wir das m -te. Daraus aber folgt sofort ein Widerspruch: Das Verfahren d führt auf m zum Erfolg falls u einmal Null wird, falls also die Aussage gilt, dass das m -te Verfahren das m -te Problem nicht löst. Das

m -te Verfahren ist aber unser d , also haben wir damit gerade die Aussage, dass d auf m nicht zum Erfolg führt. Widerspruch! Entsprechend auch für die zweite Alternative.

So haben wir also den berühmten Unvollständigkeits - Satz von Kurt **Gödel** (1906 - 1978) bewiesen:

Es gibt kein Verfahren, welches die Mathematik vollständig mechanisiert.

Was leistet ein Unmöglichkeitbeweis wie dieser wirklich? Das Resultat selbst scheint ja ein abschliessend negatives zu sein. Ich meine, ein Unmöglichkeitbeweis bringt eine Gliederung in die Geographie der Mathematik. Lassen Sie mich dies illustrieren:



Auf diesem Schema stellen wir wieder die mathematischen Methoden, den Fragestellungen gegenüber; diesmal etwas konkreter. Die Fragestellungen haben hier die Form von zu lösenden Gleichungen: lineare Gleichungen, kombinatorische Gleichungen, algebraische Gleichungen, Differential-

gleichungen, diophantische Gleichungen. An Lösungsmethoden bieten sich an: numerische Verfahren, algebraisch-symbolische Verfahren, Verwendung elementarer Funktionen wie Sinus und Exponentialfunktion, Verwendung des allgemeinen, des mengentheoretischen Funktionsbegriffs. Ein paar Beispiele: Lösung algebraischer Gleichungen durch Wurzelziehen (**Galois** und **Abel**); geschlossene Lösung von Differentialgleichungen durch elementare Funktionen (**Liouville**); Lösung des Fermat'schen Problems (**Wiles**); Existenzsätz (**Brouwer**, **Cauchy-Weierstrass**)

Unmöglichkeitbeweise bringen, wie wir sagten, eine Gliederung in die Landschaft der Probleme und Methoden. Sie entdecken ganze Bergzüge von fundamentalen Schwierigkeiten und eröffnen Aussichten auf zentrale, herausstechende Probleme. Ich will dies anhand einer Panorama-Karte veranschaulichen.

Der Gödel'sche Unmöglichkeitbeweis eröffnet den Ausblick auf mathematische Hindernisse, figurativ eine Bergkette, der ich hier den Namen Gödel-Gebirge gegeben habe. Gödel und viele andere zeigten dann, dass es Probleme gibt, welche sich mit dem Computer im Prinzip nicht lösen lassen: das Entscheidungsproblem für die Logik (**Church** und **Turing**), das 10.te Hilbertsche Problem über die ganzzahlige Lösbarkeit algebraischer, sog. diophantischer Gleichungen (**Matijasevich**), etc.

Ein näher gelegener Gebirgseinzug wurde von William **Cook** identifiziert: es sind diejenigen Probleme, die sich nicht mit zumutbarem Aufwand, feasibly wie man sagt (genauer: in polynomialer Zeit), auf dem Computer lösen lassen. Dazu gehören die von Richard **Karp** um 1970 herausgestellten Probleme wie etwa die Entscheidbarkeit der Booleschen Algebra und das Problem des optimalen Stundenplans. Weiter hinten, aber immer noch diesseits des Gödel-Gebirges liegt ein Gipfel, Piz **Fischer-Rabin** die bewiesen, dass die elementare Algebra zwar entscheidbar, aber nicht feasibly entscheidbar ist.

Hinter dem Gödel-Gebirge erhebt sich nochmals ein Gebirgszug, der Cohen-Range. Paul **Cohen** hat nämlich 1963 gezeigt, dass auch die Mengenlehre nicht ausreicht, um alle mathematischen Fragen zu entscheiden. Beispiele für solche mengentheoretisch unentschiedene Fragen sind das Auswahlaxiom (**Zermelo**) und die Kontinuum-Hypothese (**Cantor**). Und so erhebt sich Gebirgskette um Gebirgskette und der Berg der Wahrheit, der *Monte Verità*, ist noch weit dahinter.

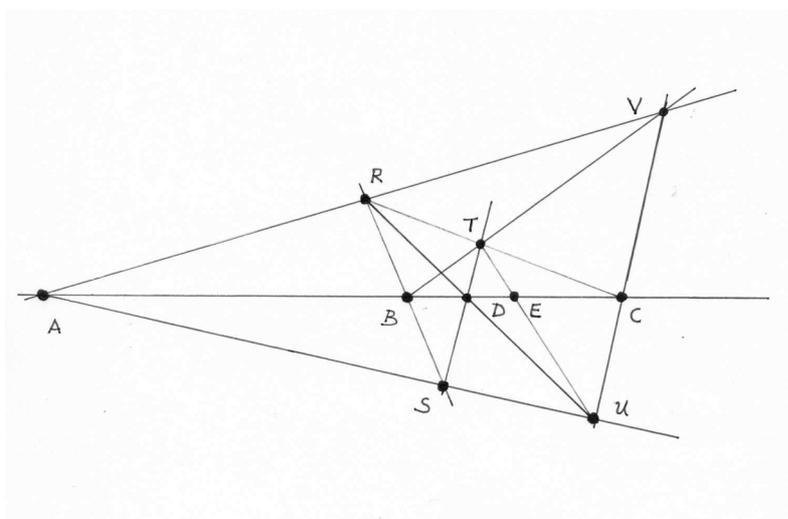
Für den Gödelschen Beweis ebenso wie für die Cook'sche These (feasible = polynomial) ist der Computer, das Programmieren und Ausführen eine Abstraktion, eine verständnisinnige mathematische Abstraktion. Und so betreffen auch die Resultate allgemeine, obwohl faszinierende, Grenzfragen der Mathematik eher als ihre alltägliche Praxis.

Wenn man sich nun bescheiden würde, gäbe es dann wenigstens für beschränkte Themen der Mathematik allgemeine, auf den heutigen Rechnern programmierbare Methoden?

Die Computer-Algebra hat in dieser Beziehung in den letzten Jahren erstaunliches geleistet, und wird in Zukunft noch mehr und über-raschendes beitragen. Denn jede Generation hat ihre bevorzugten Konzeptions-Modi. Alte Bücher, wie die über Galois Theorie und klassische algebraische Geometrie, verstauben. Und wenn etwas wieder hervorgeholt wird durch die Computer Algebra, so ist es dann doch etwas ganz anderes, nicht etwa einfach Aufgewärmtes. Es geht primär nicht um die Erledigung von Fällen, die bisher ohne Computer zu rechenaufwendig waren. Die computergestützte Mathematik lebt in einem neuen Beziehungsfeld, entwickelt neue Einsichten, schafft neue Zusammenhänge und bereichert die traditionelle Mathematik und ihre Anwendungen in vielfältigster Weise.

Etwas aber ist ganz neuartig und auch offensichtlich: Statt mit einem anderen Menschen kommuniziert man jetzt mit dem Computer. Die Schwierigkeit der computer - gestützten Mathematik ist aber genau die der Kommunikation, nämlich: Wie kann der Computer mit mir kommunizieren, um nicht zu sagen mitdenken, über das, was ich zu verstehen versuche? Denn dies ist der Gehalt der Mathematik: Nicht, gib mir ein Problem, so löse ich es. Sondern, wie kann ich das Auge, wie kann ich das abstrakte Erkennungsvermögen weiterbilden und womöglich instrumentell erweitern?

Das Problem der intelligenten Kommunikation mit dem Computer ist eines der vordringlichsten ungelösten Probleme. Ich meine, genau hier fände die künstliche Intelligenz statt naïve oder hochstaplerische Utopien ein reales Objekt. Zur Illustration zeige ich hier ein sehr einfaches Beispiel aus jüngster Zeit:



Wenn in dieser Figur alle Punkte bis auf C gegeben sind und so liegen wie angegeben, so liegt C auf dem Schnittpunkt der drei gezeigten Geraden. Computer-Algebra Systeme erlauben es heute, Sätze der Elementargeometrie automatisch in algebraische Gleichungen zu fassen und so zu beweisen. Bei dieser Übersetzung tritt etwas zutage, was man in der Geometrie oft vernachlässigt, nämlich stilschweigende Hypothesen über die sogenannte allgemeine Lage der Punkte und Geraden. Aber was bringt der Computer ans Licht ? Bei unserem einfachen Beispiel ergibt sich folgendes:

Nondegeneracy Conditions

$$\begin{aligned}
& x_a, y_c, x_r, y_u, x_a - x_r, y_c x_s - y_u x_s - x_r y_c + x_a y_u, y_c^2 x_a^2 - \\
& y_c^2 x_r x_a - y_c^2 x_r^2 + 3y_u y_c x_r^2 - y_u y_c x_r x_a - x_r^2 y_u^2, x_a^2 y_c x_s^2 - \\
& x_r^2 y_c x_s^2 - 2x_r x_a y_u x_s^2 + 2x_r^2 y_u x_s^2 - x_r x_a^2 y_c x_s + 2x_r^2 x_a y_c x_s + \\
& 2x_r x_a^2 y_u x_s - 3x_r^2 x_a y_u x_s - x_r^2 x_a^2 y_c + x_r^2 x_a^2 y_u, x_a^4 y_c^3 x_s^2 - \\
& x_r x_a^3 y_c^3 x_s^2 - 2x_r^2 x_a^2 y_c^3 x_s^2 - x_r^3 x_a y_c^3 x_s^2 + 2x_r^4 y_c^3 x_s^2 - 2x_r x_a^3 y_u y_c^2 x_s^2 + \\
& 6x_r^2 x_a^2 y_u y_c^2 x_s^2 + 6x_r^3 x_a y_u y_c^2 x_s^2 - 7x_r^4 y_u y_c^2 x_s^2 - x_r^2 x_a^2 y_u^2 y_c x_s^2 - \\
& 10x_r^3 x_a y_u^2 y_c x_s^2 + 8x_r^4 y_u^2 y_c x_s^2 + 4x_r^3 x_a y_u^3 x_s^2 - 3x_r^4 y_u^3 x_s^2 - x_r x_a^4 y_c^3 x_s + \\
& 3x_r^2 x_a^3 y_c^3 x_s + 3x_r^2 x_a^3 y_c^3 x_s + 2x_r^3 x_a^2 y_c^3 x_s - 3x_r^4 x_a y_c^3 x_s + x_r x_a^4 y_u y_c^2 x_s - \\
& 4x_r^3 x_a^3 y_u y_c^2 x_s - 11x_r^3 x_a^2 y_u y_c^2 x_s + 11x_r^4 x_a y_u y_c^2 x_s + x_r^2 x_a^3 y_u^2 y_c x_s + \\
& 15x_r^3 x_a^2 y_u^2 y_c x_s - 13x_r^4 x_a y_u^2 y_c x_s - 6x_r^3 x_a^2 y_u^3 x_s + 5x_r^4 x_a y_u^3 x_s - \\
& x_r^2 x_a^4 y_c^3 + x_r^4 x_a^2 y_c^3 + 2x_r^2 x_a^4 y_u y_c^2 + 2x_r^3 x_a^3 y_u y_c^2 - 4x_r^4 x_a^2 y_u y_c^2 - \\
& x_r^2 x_a^4 y_u^2 y_c - 4x_r^3 x_a^3 y_u^2 y_c + 5x_r^4 x_a^2 y_u^2 y_c + 2x_r^3 x_a^3 y_u^3 - 2x_r^4 x_a^2 y_u^3
\end{aligned}$$

Ein durchaus uneinsichtiger Formelsalat, in dem sich die gesuchte geometrische Hypothese versteckt !

In anderen Beziehungen hat der Computer selbst, die Hardware wie die Software, ungeheure Fortschritte gemacht. Als 1977 eine kleine Gruppe von Wissenschaftler mich unterstützte, um an der ETH ein Zentrum für Interaktives Rechnen zu schaffen, kauften wir für rund 3 Mio Franken eine DEC-10 Maschine. Die Vergleichszahlen zu heute sprechen für sich.

1978 ZIR	1996 Personal Computer
Zentrum für Interaktives Rechnen	Büro
DEC-10	PowerMac 8500
2K Cache Memory	256K Cache Memory
256K 36-Bit Hauptspeicher	32 Megabyte Hauptspeicher
3 Speicherplatten à 170 Megabyte	2 Gigabyte Speicherplatte
27 Nanosekunden Zykluszeit	8.3 Nanosekunden

Die Liste der damals jungen Beteiligten liest sich heute wie ein *Who is Who in Swiss Science*. Es waren dabei unter anderen der heutige Vizepräsident Forschung der ETH (Kübler, Bildwissenschaften), der Entdecker der Struktur des mad-cow Prions (Wüthrich, Molekularbiologie), Kommunikationsforscher (Moschytz), Statistiker, Bauingenieure und Architekten.

In der Mathematik selbst kann man den fortschreitenden Einfluss des Computing gut verfolgen anhand unserer – wie soll ich es nennen? – Koordinatisierung der Mathematik(siehe oben).

Das populäre Paradigma des Computing (ja überhaupt der Mathematik) ist das Zahlenrechnen, die Numerik. Dieses Gebiet hat der Computer auf seiner ganzen Breite erobert und eröffnet immer neue Möglichkeiten. Das nicht-numerische, das symbolische Rechnen auf dem Computer hat auch grosse

Fortschritte gemacht. Hier finden Sie sogar ein paar von meinen eigenen Beiträgen:

- Galois-Theorie für kombinatorische Probleme,
- symbolische Lösung von Differentialgleichungen in geschlossener Form, wo “geschlossene Form” nun nicht mehr einfach “Formel” heisst, sondern ein Computer-Programm sein darf.

Kehren wir zurück zum Traum von Leibniz. Der Traum von einer Grand Unification, einer Weltformel, ist noch nicht ausgeträumt; die Physik verfolgt ihn immer noch.

Aber auch die Mathematik hat dieses, beinahe magisch zu nennende, Bedürfnis nach einer Grand Unification, einer Theory of Everything. Zu Anfang des Jahrhunderts versuchten es Bertrand **Russell** (1872 - 1970) und A.N.**Whitehead** (1861 - 1947) mit einer Reduktion der Mathematik auf die Logik – mit ihrem Monumentalwerk der *Principia Mathematica* (1913).

Was das *Reduktum* der Mathematik sein soll, hat sich in der Geschichte seither des öfteren geändert.

Erstens: “*Alles ist Logik*”: dies war Russells These.

Zweitens: “*Alles ist Arithmetik*”: diese These beherrscht die Arithmetisierung seit Descartes, von der Algebraisierung der Geometrie bis zur Reduktion der

Mathematik auf einen Formalismus durch Gödel. Dieser macht aus der Mathematik einen komplizierten Zweig der Arithmetik.

Drittens: “*Mengenlehre umfasst und begründet alles*”: diese These ist aus dem Kreis der Logiker ausgebrochen und hat in vielen schulmeisterlichen Köpfen böse Verwirrung gestiftet.

Logizismus, Formalismus, Platonismus: diese drei Ansätze für eine Mathematical Theory of Everything, beherrschten das nun zu Ende gehende 20. Jahrhundert. Was wird das 21. bringen? Das “Everything”, von dem die Mathematik handeln soll, ist ja ungeheuer in Bewegung geraten. Nicht zuletzt durch den Computer: er ist heute nicht nur ein Werkzeug der Mathematik, vielmehr ist der Computer auch ein Gegenstand der Mathematik. Dies einzubeziehen haben die erwähnten drei Ansätze zur Grundlegungen der Mathematik nicht geschafft. Nicht dass es an Versuchen fehlte.

Darunter gibt es auch meinen eigenen, mit einer Reduktion der Mathematik auf die kombinatorische Algebra.

Programme	p	}	gehören zum selben Datenbereich D
Daten	d		
$p \cdot d$	Resultat der Anwendung des Programms p auf die Eingabe d		

$\mathcal{D} = \langle D, \cdot \rangle$ ist eine kombinatorische Algebra, falls:

Für jeden Ausdruck $t(x_1, \dots, x_n)$ existiert ein Element \mathbf{T} in D mit

$$\mathbf{T} \cdot x_1 \cdot x_2 \cdots x_n = t(x_1, \dots, x_n).$$

(“Prinzip der kombinatorischen Abstraktion”)

Schon von Neumann hat deutlich darauf hingewiesen, dass im Computer die Daten und die Programme letztlich das gleiche seien, *stored bits*. Der Computer wendet ein Programm p auf die Eingabedaten d an. So ist ein Computer wiederum nichts anderes als eine Rechenmaschine, welche p mit d gewissermassen “multipliziert”. Ein Bereich D mit einer solchen Multiplikation ist das, was man eine Struktur im Sinne der Algebra nennt, in diesem Fall eine kombinatorische Algebra. Kombinatorische Algebren haben nur gerade ein Grundgesetz, das hier angegebene Prinzip der kombinatorischen Abstraktion. Man kann es so einrichten, dass diese mathematische Struktur reich genug ist, so reich, dass sie die Objekte der klassischen Mathematik, selbst die Mengenlehre, enthält. Und naturgemäss enthält sie das Computing; die Struktur selbst ist in gewissem Sinne ein Computer. Ist die kombinatorische Algebra aber auch ein universelles Gefäss für die gesamte Mathematik? Ein Test dafür wäre die Ausführbarkeit der Anweisungen des vor vierhundert Jahren geborenen René **Descartes** (1596 - 1650) für die Wissenschaften. In der Formulierung von 1962 unseres unvergessenen Georg **Polya** (1887 - 1985) lauten sie:

Descartes's famous "Rules for the Direction of the Mind"

- First, reduce any kind of problem to a mathematical problem.
- Second, reduce any kind of mathematical problem to a problem of algebra.
- Third, reduce any problem of algebra to the solution of a single equation.

Man reduziere alle ungelösten Probleme der Welt auf Mathematik; dort stelle man das Problem als ein Gleichungssystem dar, mache daraus eine einzige Gleichung, die dann zu lösen wäre.

Mit dem nötigen Respekt und den nötigen Vorbehalten kann ich aber durchaus behaupten, dass die kombinatorische Algebra das Cartesische Programm erfüllt.

Erstens wird das gegebene Problem und dessen mathematischer Kontext in eine genügend reichhaltige kombinatorische Algebra übersetzt.

Das Resultat ist ein Gleichungssystem. Zweitens bringt man dieses Gleichungssystem mit den Regeln der kombinatorischen Algebra in reduzierte Form, nämlich gerade auf eine einzige Gleichung: zu gegebenen a und b finde man x so, dass a mal x gleich b mal x ist.

Falls \mathcal{D} eine kombinatorische Algebra ist, so lässt sich jedes endliche Gleichungssystem reduzieren auf eine einzige Gleichung mit einer Unbekannten:

$$a \cdot x = b \cdot x$$

In “genügend reichen” kombinatorischen Algebren kann man jede mathematische Struktur treu (isomorph) einbetten.

Die Mathematik des Computing als eine Theory of Everything scheint nun wohl ein bisschen utopisch. Aber durchaus ernst zu nehmende Physiker und Mathematiker haben in neuester Zeit einen solchen Ansatz mit grossem Elan verfolgt: Der Oxforder Astrophysiker David **Deutsch** präsentiert ein starkes Argument dafür, dass das Universum als eine Rechenmaschine dargestellt werden kann, nämlich als Richard **Feynmans** *quantum computer*. Der Mathematiker Sir Roger **Penrose** macht in seinen Büchern einen Erklärungsversuch für das Phänomen des Bewusstseins, ebenfalls basiert auf einem quantentheoretischen Computermodell.

Wie mein Lehrer Paul **Bernays** (1888 - 1977) vor 20 Jahren in seinem letzten Vortrag und im hohem Alter von 88 sagte: “Vieles in der mathematischen Forschung ist noch unabgeklärt, und wir können auf etliches, was wir noch zu lernen haben, neugierig sein.”



Auch ich werde mich freuen, ob all dem was ich in meinem *otium cum dignitate*, der Musse in Würde, noch lernen darf, und ich danke dafür zum voraus.