



Hans-Peter Prüfer (Autor)

Methoden der Künstlichen Intelligenz in einem Konzept des rechnerintegrierten Konstruierens



<https://cuvillier.de/de/shop/publications/7305>

Copyright:

Cuvillier Verlag, Inhaberin Annette Jentsch-Cuvillier, Nonnenstieg 8, 37075 Göttingen,
Germany

Telefon: +49 (0)551 54724-0, E-Mail: info@cuvillier.de, Website: <https://cuvillier.de>

Gegensätze ziehen sich an! Dieser oft gehörte und gewiß eher als banal empfundene Satz mag wohl am besten die Motivation verdeutlichen, die zu dem im Titel dieser Arbeit niedergelegten Programm geführt hat. Hier soll der Versuch gewagt werden, drei durchaus nicht immer friedlich miteinander koexistierende Bereiche der Wissenschaft einander näherzubringen. Die "feindlichen Geschwister", die es zu betrachten gilt, sind - in der Reihenfolge innerhalb des Arbeitstitels - die Informatik, vertreten durch die Künstliche Intelligenz, die Mathematik in Form der jedem methodischen Vorgehen zugrundeliegenden Logik sowie der Numerik für das Rechnen und schließlich die Ingenieurwissenschaften in ihrer wohl anschaulichsten Ausprägung als Konstruktionstechnik.

Es sollen an dieser Stelle keine alten Vorurteile einiger Adepten der unmittelbar involvierten Wissenschaften aufgewärmt werden; unbestreitbar aber gibt es offensichtlich selbst bei ihren fortschrittlichsten Vertretern durchaus Berührungspunkte, die zu einem großen Teil auf die so sehr voneinander getrennten Ausbildungsgänge zurückzuführen sein dürften. Interessant ist allerdings, daß die erwähnten Vorurteile im wesentlichen ebenso alt sind wie die gemeinsamen Wurzeln der drei zu betrachtenden Fachbereiche.

Mathematik und Logik haben gemeinsame Wurzeln in der klassischen geometrisch abstrahierenden Denkweise der Pythagoräer, aber auch in der praktisch orientierten Landmesskunst der Ägypter sowie in den astronomischen Berechnungen der Sumerer. Die induktive Logik¹ und das damit verbundene Aufstellen von Rechenmodellen bilden eine der Grundfesten der Physik, die in dieser Form über Galilei hinaus bis hin zu Archimedes, dem vielleicht ersten

¹ Logik ist hier nicht nur als aristotelische, deduktive Logik zu verstehen, sondern weitergehend als induktives Denkprinzip, worin sie dem Logikbegriff der Philosophie nahekommt. Es sei darauf verwiesen, daß die Aufstellung physikalischer Theorien ein entschieden induktiver Prozeß ist, bei dem von hinreichend vielen Einzelbeobachtungen durch Abstraktion und Generalisierung auf die allgemeingültige Theorie geschlossen wird.

modernen Physiker und Ingenieur, zurückverfolgt werden kann. Eine zweckmäßige Organisation von Denkvorgängen, also eine formale Beschreibung von Daten und ihren wechselseitigen Beziehungen, die einen bedeutenden Bestandteil der Informatik darstellt, ist andererseits ohne axiomatische Logik und das damit verbundene mathematische Rüstzeug kaum vorstellbar. Die Ingenieurwissenschaften ihrerseits fußen auf der klassischen Physik², die ohne mathematische Methoden niemals zu ihrer Blüte hätte gelangen können; nicht umsonst ist der große Physiker Newton gleichzeitig auch einer der Begründer der Infinitesimalrechnung. Auch hier findet man wiederum die umgekehrte Beziehung: Der durch die strenge aristotelische Logik geprägte Philosoph Leibnitz ist der andere Vater der infinitesimalen Methoden.

Keinesfalls zu unterschätzen sind Beziehungen zwischen den Ingenieurwissenschaften und der Mathematik. Die heute so sehr zum Stand der Technik avancierte Methode der Finiten Elemente wurde von praktisch arbeitenden Ingenieuren zu ihrer jetzigen Reife entwickelt, und die Aufgabe der Mathematik konnte sich im wesentlichen darauf beschränken, die theoretische Begründung für die praxisbewährte Vorgehensweise nachzuliefern³. Dem allen Mathematikern innewohnenden Hang zur Aufdeckung allgemeinerer Prinzipien folgend wurde dann die Methode der Finiten Elemente von der Bearbeitung "einfacher" elastomechanischer Aufgaben auf die Lösung von Systemen partieller Differentialgleichungen übertragen, so etwa auf Wärmeleitungs- oder Strömungsprobleme, die ihrerseits nicht nur von theoretischem Interesse

² Unter dem Begriff der klassischen Physik wollen wir hier im wesentlichen die Newtonsche Mechanik verstehen, wobei die Festigkeitslehre, die in ihrem Ansatz vollkommen "newtonisch" ist, mit eingeschlossen wird. Von eher marginalem Interesse dürfte dagegen die relativistische Mechanik sein. Aussagen über ihre Rückwirkungen auf Ingenieur Tätigkeiten gehören wenigstens zur Zeit noch in das Reich der Spekulation.

³ Ein vergleichbarer Fall ist aus der modernen Physik bekannt. Die berühmte Diracsche Deltafunktion - zuweilen auch als Einheitsimpuls bezeichnet - wurde bereits in den zwanziger Jahren in implizit richtiger Weise wie eine Funktion angewendet, obwohl sie defacto keine Funktion darstellt. Die mathematische Begründung für die Richtigkeit der intuitiv aufgestellten praktischen Regeln wurde um 1950 von Laurent Schwarz als Theorie der Distributionen veröffentlicht.

sind, sondern letztlich zu einem besseren Verständnis komplexer Maschinen⁴ führen.

Die heutige Bedeutung der Informatik als tendenziell interdisziplinäre Wissenschaft steht außer Frage. Zwar kann die Informatik bei boshafter Interpretation als mehr oder minder illegitimer Sproß der reinen Mathematik verstanden werden, dennoch hat sie sich inzwischen zu einer eigenständigen Wissenschaft emanzipieren können, die in methodischer Hinsicht in gleichem Maße von der Mathematik wie von der Konstruktionstechnik als typischem Vertreter der Ingenieurwissenschaften profitiert hat. Zwei Arbeitsgebiete mögen dies unterstreichen. So ist die Theorie der formalen Sprachen, die für den Entwurf von Compilern unverzichtbar ist, in ihren Ursprüngen zutiefst algebraisch. Die Disziplin des Software Engineering, die bereits in ihrem Namen eine ingenieurmäßige Komponente enthält, entspricht demgegenüber fast vollständig dem Konstruktionsprozeß des "klassischen" Maschinenbaus⁵. Als befruchtende Einflüsse der Informatik sollen hier die neuen rechnerorientierten Beweistechniken⁶ in der reinen Mathematik sowie die Datenbankkonzepte und die Expertensysteme, die eine

4 Im Falle der Strömungsmechanik sind dies vor allem Turbinen mit ihren geometrisch komplex geformten Umlenkkanälen; ebenfalls von praktischem Interesse sind die Strömungsverhältnisse an Überschallflugkörpern.

5 Man kann die Parallele sogar noch weiter ziehen. Die heutigen methodischen Ansätze für den Konstruktionsprozeß mit ihrem strengen, beinahe axiomatischen Vorgehen und dem darin enthaltenen Prinzip der fortgesetzten Verfeinerung, also Konkretisierung und Spezialisierung, finden ihre Entsprechung in den verschiedenen Phasen des Software Life Cycle. Die technischen Zeichnungen des Ingenieurs kann man als Pendant zu den Darstellungsmethoden wie Flußdiagramme oder Struktogramme verstehen!

6 Neben den automatischen Beweissystemen für gewisse Teilbereiche der Geometrie, die selbständig ein Axiomensystem auszuwerten vermögen, sind auch die Untersuchungen an endlichen Problemen zu nennen, die ohne Rechnerunterstützung nicht mit vertretbarem Zeitaufwand zu lösen wären. Zwar wird dieses Vorgehen von manchen fundamentalistischen Mathematikern abgelehnt, aber es spricht unter dem Aspekt der erreichbaren Erfolge wenig dagegen, eine Lücke in der Beweiskette, wie sie etwa bei der Bearbeitung des Vierfarben-Problems auftrat, durch die Verwendung symbolisch arbeitender Programmpakete zu schließen, zumal ein "eleganter" allgemeingültiger, in sich geschlossener Beweis kaum zur Erhellung beiträgt.

neuartige Organisation ingenieurmäßigen Wissens erlauben, erwähnt werden.

Beispiele für Querverbindungen der geschilderten Art lassen sich in beliebiger Zahl angeben. Sie zeigen deutlich, daß die Trennung der drei uns hier unmittelbar interessierenden Wissenschaften offensichtlich ein Artefakt darstellt. Den Umkehrschluß daraus zu ziehen bedeutet letztlich, daß eine "Wiedervereinigung" im Sinne einer intensiveren Zusammenarbeit nicht nur möglich ist, sondern darüber hinaus durch Synergieeffekte⁷ neue, fortschrittliche Ansätze für Entwicklungen in Forschung und Praxis initiieren wird.

Die als Ausgangspunkt der angestellten Betrachtungen genannten Berührungsängste oder gar Aversionen lassen sich im Grunde nur als Unverständnis oder Mangel an Selbstbewußtsein interpretieren. Gerade diese letztgenannte Eigenschaft erklärt möglicherweise die oft so heftig hervorbrechende gegenseitige Abneigung. Warum sollte ein Mathematiker sich scheuen, die vermeintlichen Niederungen praxisbezogener Forschung aufzusuchen, wenn er doch selbst davon profitieren kann? Warum haben manche Ingenieure einen durch lautstarke Verachtung kompensierten Minderwertigkeitskomplex gegenüber der Mathematik? Warum sitzen Informatiker zuweilen zwischen allen Stühlen, warum werden sie im Extremfall weder von Mathematikern noch von Ingenieuren ernstgenommen, so daß sie sich schmollend auf eine Insel der Seligen zurückziehen wollen? Es sei die Hypothese gewagt, daß durch eine stärker interdisziplinär geprägte Ausbildung - bei der von den Lehrern selbstverständlich das gegenseitige Verständnis schon vorausgesetzt werden muß - derartige Erscheinungen endgültig der Vergangenheit angehören dürften. Daß dies möglich und sinnvoll ist, sollen die in den folgenden Kapiteln beschriebenen Anwendungen und Konzepte demonstrieren.

Dem Autor dieser Zeilen, der die Bereiche Mathematik und Maschinenbau aufgrund seiner akademischen Ausbildung

⁷ Das Modewort Synergie bedeutet im Grunde nichts anderes als daß das Ganze mehr ist als die Summe seiner Teile. Zur Ausnutzung tatsächlicher und vermeintlicher Gegensätze bietet sich daneben das altbewährte dialektische Prinzip an, mit dessen Hilfe aus These und Antithese eine beide übertreffende Synthese erzeugt wird.

und seiner Forschungsarbeiten kennt, dem die Informatik durch den beinahe täglichen Umgang mit eigenen und fremden Softwareprodukten wenigstens zum Teil vertraut ist, ist es beinahe ein Herzensanliegen, diese drei Bereiche miteinander zu "verheiraten" in der Hoffnung, daß sich das so entstehende Spannungsfeld als fruchtbar erweisen möge. Als Motto sei daher dem Leser in leicht abgewandelter Form zugerufen:

"Gerant bella alii, tu felix scientia nube!"

Ein Fahrplan

Hüte Dich vor den Lehren jener
Spekulanten, deren Überlegungen nicht
von Erfahrungen bestätigt sind.

Diente der vorausgegangene Prolog noch der Erläuterung der Motivation, die zu der Erstellung einer Arbeit wie dieser führt, so wollen wir in diesem kurzen Abschnitt ein Exposé des Inhaltes geben. Wir verlassen daher die Wunschvorstellung einer idealen, vollkommenen Welt, und wenden uns einer realistischeren Weltsicht zu.

Es ist offenkundig, daß eine vollständige Exploration der drei Wissenschaftsbereiche Konstruktionstechnik, Mathematik und Informatik und ihrer Gemeinsamkeiten innerhalb des hier zur Verfügung stehenden Raumes noch viel weniger möglich ist als für nur einen dieser Bereiche, deren Beschreibung gesamte Bibliotheken zu füllen vermöchte. Wir können und werden uns deshalb auf Sonderfälle exemplarischen Charakters beschränken, was jedoch der Allgemeingültigkeit der anzustellenden Überlegungen keinerlei Abbruch tut.

In den Kapiteln über die Rechnerunterstützung in den Ingenieurwissenschaften und die algorithmische Behandlung des Konstruktionsprozesses werden wir Aufschluß über die Bedeutung und den Umfang dieses Teilgebietes erlangen, wobei wir uns gleichzeitig mit dem aktuellen Stand der Technik vertraut machen. Wir werden uns dann dem Bereich zuwenden, der bei den bisherigen Konzepten zur Rechnerunterstützung trotz seiner impliziten Logik am wenigsten berücksichtigt worden ist, nämlich der Methodik des Konstruktionsprozesses selbst. Von diesem Bereich ist es ein kleiner Schritt zur Prädikatenlogik, zur Physik und zur Mathematik in ihrer Ausprägung als Arithmetik und Analysis.

Die sich daran anschließenden Kapitel schlagen eine Brücke zur Informatik, indem einerseits die Analogie zwischen Konstruktionsprozeß und Softwareentwicklung beschrieben wird, andererseits auf ebendieser Basis ein Konzept zur Rechnerunterstützung entwickelt wird. Der Kristallisationskeim aller dieser Gedanken ist das Prinzip der Übertragungsfunktion, welches wir als zentral für den gesamten Konstruktionsprozeß ansehen. Eine umfassende Beschreibung funktionaler Prozesse stellt damit einen

gangbaren Weg zur fachübergreifenden methodischen Integration dar.

Die Richtigkeit dieser zunächst theoretischen Überlegungen verifizieren wir in zweifacher Weise. Zum einen beschreiben wir das Prinzip der Implementierung eines rechnerorientierten Unterstützungssystems, zum anderen wird anhand zweier typischer Aufgabenstellungen das methodische Vorgehen dargestellt und in Regeln gefaßt, die sich für die Implementierung eignen. Gleichzeitig können wir den Vorteil der neuartigen Vorgehensweise gegenüber den konventionellen Rechnerhilfsmitteln zeigen. Insbesondere ist es denkbar, den ingenieurmäßig orientierten Anwender in weiten Bereichen von der fehlerträchtigen Programmierarbeit zu befreien, indem wir ihm ein Unterstützungssystem bereitstellen, welches nicht nur "seine Sprache" spricht, sondern darüber hinaus kontextsensitiv die Semantik der Formelzeichen und Symbole seines Vorrates an Berechnungsformeln, also letztlich seines Erfahrungsschatzes auswertet.

Wir werden dazu den inneren Aufbau des notwendigen Softwaresystems eingehend beschreiben und dabei begründen, auf welche typischen Benutzerschwierigkeiten ein solches Konzept Rücksicht zu nehmen hat. Es schließt sich eine wohlbegründete Wahl der sinnvoll einsetzbaren Softwarewerkzeuge an, die insbesondere jedem interessierten Anwender ohne übergroßen Aufwand zugänglich ist, da keine kostspieligen Hard- oder Softwarevoraussetzungen gefordert werden.

Die zu bearbeitenden Aufgaben sind dabei zum einen eine Optimierungsaufgabe, bei der ein umfangreiches benutzer-spezifisches Regelwerk zu berücksichtigen ist, und zum anderen die Betrachtung einer Funktionsstruktur, die im wesentlichen die graphische Darstellung einer Gesamtübertragungsfunktion ist. Beide Aufgabenstellungen werden anhand ihrer zugeordneten Axiomensysteme beschrieben, so daß die gesamte Systemphilosophie vollständig nachvollziehbar wird.

Den Abschluß bildet der übliche Ausblick auf die Zukunft, der aber hier an konkreten Voraussetzungen festgemacht wird und damit vielmehr als "Auftragsbuch" zu verstehen ist, welches nur noch in die Tat umgesetzt werden muß.

1 DIE RECHNERUNTERSTÜTZUNG IN DEN INGENIEURWISSENSCHAFTEN - EINE BESTANDSAUFNAHME

Honni soit qui mal y pense.

Die Ingenieurwissenschaften moderner Prägung sind ohne den Einsatz rechnerorientierter Hilfsmittel kaum mehr denkbar. Der Rechner hat ausgehend von den kaufmännischen Anwendungen einen Marsch durch die Institutionen angetreten, der schier unaufhaltsam erscheint. Man findet das Instrumentarium der elektronischen Datenverarbeitung in den Labors und an den Prüfständen ebenso wie in den Konstruktionsbüros, in denen der Rechner allmählich die Zeichenbretter und Rechenschieber/Taschenrechner ablöst. Trotz dieser außerordentlichen Verbreitung von Rechnern der unterschiedlichsten Leistungsklassen wird der interessierte Beobachter mit der erstaunlichen Tatsache konfrontiert, daß die überwiegende Mehrheit der EDV-Komponenten einschließlich der darauf installierten Softwareprodukte völlig beziehungslos nebeneinander steht. Damit ist keineswegs gemeint, daß kein Datenaustausch zwischen den verschiedenen Systemen stattfindet - rein technisch betrachtet gibt es hier dank des Aufkommens der sogenannten Lokalen Netzwerke kaum mehr Hindernisse - sondern es ergibt sich im Gegenteil die absurde Situation, daß zwar tendenziell jeder in einer vernetzten Umgebung befindliche Rechner auf die Daten seiner Partner zugreifen könnte, diese aber in der Regel nicht zu nutzen versteht, da bei der Erzeugung der Daten keine Rücksicht auf gemeinsame Nutzung gelegt worden ist.

Man kann diese Situation mit der während des Turmbaus zu Babel entstandenen vergleichen. Alle beteiligten Rechnerkomponenten sind in der Lage, Informationen zu erzeugen und zur Verwendung freizugeben. Da aber die Systeme "unterschiedliche Sprachen sprechen", bleiben die Botschaften ungehört, es sei denn, man verfügt über geeignete Dolmetscher, die in der Welt der Rechner durch sogenannte Schnittstellenprogramme repräsentiert werden. Derartige Hilfsmittel bilden aber eher die Ausnahme und müssen auch von Fall zu Fall individuell bereitgestellt werden; eine signifikante Verbreitung von Schnittstellen zum Datenaustausch findet man allenfalls im Umfeld der technisch-graphischen Datenverarbeitung.

Was offensichtlich fehlt, ist eine "lingua franca" auf der Ebene der Daten oder auch der Dateien, die einen mühelosen Informationsaustausch gestattet. Es ist ebenso klar, daß hierzu die Möglichkeit einer vollständigen Übertragung der Gesamtheit aller relevanten Daten wünschenswert ist, denn nur allzu oft entpuppt sich ein Minimalkonsens als die leere Menge. Wollen wir ein tieferes Verständnis für derartige Fragestellungen entwickeln, so ist es gewiß unerlässlich, an erster Stelle den derzeitigen Stand der Softwarewelt im technischen Bereich unter Einschluß der jeweiligen Datenhaushalte einer näheren, wenn auch keineswegs erschöpfenden Betrachtung zu unterziehen.

Nun bedeutet Rechnerunterstützung - und hier ist schon das Generalthema der Rechnerintegration mitangesprochen - gewiß mehr als nur Datenaustausch. Miteinander gekoppelt werden ja nicht nur zu verarbeitende Objekte, sondern auch die Methoden, die auf den Objekten operieren. So können etwa komplexe Synthese-/Analysealgorithmen als Verkettung elementarer Methoden beschrieben werden. Dabei sollten im Sinne eines objektorientierten Konzeptes Methoden und Daten als Einheit betrachtet werden, denn - abgesehen von Seiteneffekten - entstehen bei allen algorithmischen Vorgängen aus vorgegebenen Eingangsdaten die gewünschten Ausgangsdaten, die Resultate. Erst die integrale Auffassung bietet daher die Möglichkeit der semantischen Kontrolle aller beteiligten Daten: Die Bedeutung von Daten wird durch die auf ihnen operierenden Funktionen festgelegt.

Mit der folgenden Übersicht soll gezeigt werden, wie sehr die Ingenieurwissenschaften inzwischen auf die verschiedensten Formen der Rechnerunterstützung angewiesen sind. Dabei werden uns angesichts der Zielsetzung unserer Untersuchungen vor allem die Softwarekomponenten interessieren, die einen unmittelbaren Bezug zum Maschinenbau und dort insbesondere zur Konstruktionstechnik im engeren Sinne aufweisen.

1.1 Kritik des reinen CAD: Wunsch und Wirklichkeit

Die wohl bekannteste Form der Rechnerunterstützung innerhalb der Ingenieurertätigkeiten ist unter dem Kürzel CAD - Computer Aided Design - bekannt. Dieser Begriff wird gerne als Rechnerunterstütztes Konstruieren interpretiert, was

auch durchaus einer wörtlichen Übersetzung der anglo-amerikanischen Version entspricht. Vergleicht man allerdings Anspruch und Wirklichkeit, so wird man rasch eines Besseren belehrt. Das Konstruieren ist zweifelsohne eine Tätigkeit, die von der motivierenden Grundidee über den Entwurf in immer weitergehenden Konkretisierungen bis hin zur endgültigen Bemessung reicht. Demgegenüber findet man unter dem Oberbegriff CAD zumeist solche Systeme subsummiert, die zwar eine erhebliche Erleichterung bei der Erstellung von Zeichnungen bieten, an darüber hinausgehenden Fähigkeiten bestenfalls über die Möglichkeit von parametrisierten Befehlen zur Variantenkonstruktion verfügen, zuweilen sogar noch die Erzeugung von Befehlen für die NC-Fertigung - genannt sei hier das Stichwort APT¹ - unterstützen, bei so wichtigen und ständig benötigten Ingenieurtätigkeiten wie der Ausführung von Berechnungen aber den Anwender im Stich lassen. Die heute eher als polemisch empfundene Interpretation von CAD als Computer Aided Drafting - Rechnerunterstütztes Zeichnen - beschreibt dennoch die Gesamtheit der verfügbaren Leistungsmerkmale von CAD-Software nur allzu treffend².

1 Automatically Programmed Tools: Es handelt sich hierbei um das erste sogenannte Teileprogrammiersystem, mit dessen Hilfe die Geometrie eines Bauteiles in einer für die NC-Bearbeitung geeigneten Form beschrieben werden kann [1].

2 Zum Verständnis sollte man sich in diesem Zusammenhang die Entwicklungsgeschichte des CAD vergegenwärtigen. Es gibt hier zwei voneinander vollkommen unabhängige Entwicklungslinien. Die eine geht auf die Erstellung elektrischer Schaltpläne - auch Platinenlayouts - zurück, die insofern recht einfach zu implementieren sind als im wesentlichen nur waagerechte und senkrechte Linien auftreten, die an einen kleinen Vorrat von Schaltsymbolen geknüpft werden. Angesichts dieser Einfachheit ist es nicht weiter verwunderlich, daß derartige Systeme bereits recht früh zu Anfang der sechziger Jahre kommerziell verfügbar wurden. Die andere Entwicklungslinie geht von der technischen Zeichnung als einem bedeutsamen Zwischenprodukt des Konstruktionsprozesses aus. Solche Zeichnungen sind jedoch erheblich komplexer und vielseitiger als Schaltpläne; man denke hierzu nur an Schraffur, Bemaßung sowie die Vielzahl von Normteilen. Wenn man nun weiter bedenkt, wie aufwendig die Spezifikation und Implementierung allein dieser Bereiche ist, so wird es klar, daß bereits die Generierung und Verwaltung der reinen "Geometrie" eine überaus bemerkenswerte Leistung darstellt, zumal diese CAD-Pakete ohne Unterstützung durch die seinerzeit noch nicht so weit verbreitete Informatik erstellt worden sind.