

KERNFORSCHUNGSZENTRUM KARLSRUHE

Januar 1969

Externer Bericht

8/69-1

Institut für Reaktorentwicklung  
Datenverarbeitungszentrale

Dynamische Simulatoren in der Reaktorentwicklung.  
Ein Vergleich

---

W.Frisch

G.Wilhelmi

Gesellschaft für Kernforschung mbH., Karlsruhe

Als Manuskript vervielfältigt.

Für diesen Bericht behalten wir uns alle Rechte vor.

Gesellschaft für Kernforschung mbH., Karlsruhe

KERNFORSCHUNGSZENTRUM KARLSRUHE

Januar 1969

Externer Bericht

8/69-1

Institut für Reaktorentwicklung  
Datenverarbeitungszentrale

Dynamische Simulatoren in der Reaktorentwicklung.  
Ein Vergleich

---

W.Frisch

G.Wilhelmi

Gesellschaft für Kernforschung mbH., Karlsruhe

<u>Inhalt</u>	<u>Seite</u>
1. Einleitung	1
2. Dynamische Simulatoren	1
2.1 Kriterien zur Beurteilung von Simulatoren	1
2.2 Gestellte Anforderungen	3
2.3 Entwicklungstendenzen bei Simulatoren	4
2.4 Ausgewählte Simulatoren	8
3. Problemanalyse	10
3.1 Dynamische Probleme in der Reaktorentwicklung	10
3.2 Lösungsmethoden	11
3.3 Eigenschaften der Gleichungssysteme	13
3.4 Testbeispiele	17
4. Systemvergleich	20
4.1 Systemeigenschaften	20
4.2 Prozeßvorbereitung	26
4.3 Prozeßablauf	28
5. Zukünftige Entwicklung	29
5.1 Hardware	29
5.2 Software	30
5.3 Mögliche Erhöhung der Rechengeschwindigkeit	31
6. Zusammenfassung	35
Literatur	39

## 1. Einführung

In der vorliegenden Vergleichsstudie werden verschiedene Simulatoren zum Lösen dynamischer Probleme gegenübergestellt. Die Beurteilung erfolgt mit Hilfe von Kriterien, die in Kap. 2 zusammengestellt sind, und anhand von konkreten Simulationsbeispielen aus der Kernreakorteknik (Kap. 3).

Hauptaufgabe dieser Studie ist die Gegenüberstellung von Analogrechnern (einschl. Hybridrechnern) und digitalen Simulatoren. Neben dem hybriden Analogrechner HYDAC 2000 des Instituts für Reaktorentwicklung (Kernforschungszentrum Karlsruhe) wurden 3 digitale Simulatoren, die wegen ihres grundsätzlich verschiedenen Aufbaus einen sehr großen Teil aller bisher bekannten dynam. Simulatoren repräsentieren.

Um nicht nur qualitative, sondern auch quantitative Aussagen zu ermöglichen, wurden 4 typische Simulationsbeispiele aus der Reaktorteknik ausgewählt und mit allen 4 Simulatoren dargestellt. Die Gegenüberstellung der ausgewählten Simulatoren wird durch einen Ausblick auf mögliche Verbesserungen in der nahen Zukunft ergänzt.

## 2. Dynamische Simulatoren

### 2.1 Kriterien zur Beurteilung von Simulatoren

Als dynamische Simulatoren werden Hilfsmittel bezeichnet, mit denen das Zeitverhalten von physikalischen Systemen berechnet werden kann.

Die benutzten Lösungsmethoden können sehr verschieden sein. Beim Analogrechner wird das zu untersuchende physikalische oder technische System (z.B. Kernreaktor) durch ein anderes System (normalerweise ein elektrisches Netzwerk) ersetzt, das durch die gleichen mathematischen Gleichungen beschrieben werden kann.

Bei digitaler Simulation werden die Gleichungen, die das System beschreiben, numerisch gelöst.

Simulatoren, die für spezielle Probleme, wie z.B. für Reaktorex-

kursionen [13] entwickelt wurden, und in ihrem Anwendungsbereich auf diese Problematik beschränkt sind, werden hier nicht betrachtet. Diese Studie befaßt sich mit Simulatoren, die möglichst allgemein verwendbar sind.

Bei jedem Simulator sind das Programmiersystem (Software) und die Rechenanlage (Hardware) als eine geschlossene Einheit zu betrachten, denn die Qualität eines Simulators hängt von beiden ab.

Als Simulatoren der untersten Stufe kann man einfache Analogrechner mit blockorientierter Programmierung oder Digitalrechner mit FORTRAN-Programmierung bezeichnen; sie genügen jedoch längst nicht allen Anforderungen, die an gute Simulatoren gestellt werden. Ihre Hauptaufgabe ist, den Programmieraufwand zum Lösen eines bestimmten dynamischen Problems erheblich herabzusetzen. Im Idealfall soll der Aufwand auf die reine Darstellung des Problems (z.B. in Form einer Gleichung oder eines Blockschaltbildes) reduziert werden, ohne daß zusätzliche zeitraubende Detailarbeit (z.B. Programmorganisation, Stabilitätsprüfung für Schrittweiten bei digitaler Integration usw.) erforderlich ist.

Bei der Erstellung oder Auswahl solcher dynamischer Simulatoren sind folgende Kriterien zu beachten, nach denen der Prozeß des Lösen dynamischer Probleme beurteilt wird:

1. Systemeigenschaften

- a) Darstellbare Funktionen
- b) Rechenkapazität
- c) Rechengeschwindigkeit
- d) Genauigkeit
- e) Einsatzbereitschaft und Verfügbarkeit über die Ergebnisse

2. Prozeßvorbereitung

- a) Programmieraufwand
- b) Übersichtliche Struktur der Programmiersprache
- c) Testmöglichkeit

### 3. Prozßverlauf

- a) Verfolgung der laufenden Rechnung
- b) Eingriff in die laufende Rechnung

### 2.2 Gestellte Anforderungen

Anhand der Kriterien zur Beurteilung von dynamischen Simulatoren kann man konkrete Anforderungen an diese Systeme stellen. Bei der Aufstellung einer Liste solcher Anforderungen wurde sowohl die Notwendigkeit von der Benutzerseite her als auch die in näherer Zukunft zu erwartende Realisierbarkeit (Bereitstellung geeigneter Hardware) berücksichtigt. Die Liste der folgenden Anforderungen ist durch entsprechende Nummerierung den in Kap. 2.1 aufgestellten Kriterien zugeordnet.

- 1a. Alle gebräuchlichen mathematischen Funktionen müssen leicht und mit ausreichender Genauigkeit dargestellt werden können.
- 1b. Es muß genügend Rechenkapazität zur Verfügung gestellt werden, um auch sehr große Systeme simulieren zu können (z.B. 200 gekoppelte Dgln.).
- 1c. Die von Problemen und von der Hardware her maximal mögliche Rechengeschwindigkeit soll immer erreicht werden (z.B. durch Schrittweitenautomatik bei digitaler Integration).
- 1d. Die Rechengenauigkeit, die meistens eng verknüpft ist mit der Rechengeschwindigkeit (beim Digitalrechner) oder mit der Rechenkapazität (beim Analogrechner), muß frei wählbar sein und Mindestanforderungen genügen.
- 1e. Der Simulator muß ohne lange Wartezeiten einsatzbereit sein. Ergebnisse müssen während der Rechnung verfolgt werden können und nach Beendigung der Rechnung in Tabellen- oder Kurvenform zur Verfügung stehen.
- 2a. Der Dialog mit dem Simulator muß in einer benutzernahen Sprache geführt werden können. Übermäßige Belastung mit Organisationsproblemen muß vermieden werden.
- 2b. Die Problemstruktur muß auch nach Umsetzen in die Programmiersprache noch deutlich sichtbar bleiben.
- 2c. Ein Austesten der stationären Phase des Programmes muß automatisch ohne zusätzlichen Programmieraufwand erfolgen.

- 3a. Die Rechnung muß dauernd verfolgt werden können, z.B. durch Darstellung von Variablen in Tabellen- oder Kurvenform und durch Signale, die den Betriebszustand des Programmes markieren. Die dargestellten Variablen müssen während der laufenden Rechnung ausgetauscht werden können.
- 3b. Die Rechnung muß an jeder beliebigen Stelle abgebrochen oder unterbrochen werden können. Sowohl Parameter als auch die Struktur des simulierten Systems müssen während der Rechnung (oder während einer kurzen Unterbrechung) geändert werden können.

Die zum Vergleich herangezogenen Simulatoren werden auf diese Anforderungen hin untersucht. Es kann schon vorweggenommen werden, daß keiner der Simulatoren alle gestellten Forderungen zufriedenstellend erfüllt. Lediglich moderne Hybridrechnersysteme, die in den praktischen Vergleich jedoch nicht mit einbezogen werden konnten, erfüllen die Forderungen in den meisten Punkten.

## 2.3 Entwicklungstendenzen bei Simulatoren

Bevor die ausgewählten Simulatoren beschrieben werden, ist ein kurzer Überblick über die augenblicklichen Entwicklungstendenzen erforderlich. Ohne einen Anspruch auf Vollständigkeit zu erheben, werden für die verschiedenen Simulatortypen (Analog- und Hybridrechner, Digitalsimulator) der augenblickliche Entwicklungsstand und die in näherer Zukunft zu erwartenden Entwicklungsrichtungen aufgezeigt.

### 2.3.1 Analog- und Hybridrechner

Auf das Prinzip des Analogrechnens und auf die zur Zeit von mehreren Firmen angebotenen Standard-Analogrechner soll hier nicht näher eingegangen werden.

Neben der Erhöhung der Genauigkeit und der Schnelligkeit wird bei der Weiterentwicklung der Analogrechner besonderer Wert auf eine Verbesserung der Programmvorbereitung, -eingabe und -prüfung gelegt. Außerdem kann der Programmablauf durch einen in fast jedem modernen Analogrechner enthaltenen Logik-Teil automatisiert werden. (Die mit einem digitalen Logikteil ausgerüsteten Analogrechner ver-



den oft hybride Analogrechner genannt.)

Die Programmvorbereitungszeit wird durch digitale Hilfsprogramme, mit denen Koeffizienten und Testwerte berechnet werden können, erheblich verkürzt (z.B. DIANA [27], APACHE [28]). Um die Rechenvorbereitungen auf der Maschine zu verkürzen, stehen ebenfalls Hilfsprogramme und -geräte zur Verfügung, z.B. das ADIOS-System zum automatischen Einstellen der Potentiometer, zur Durchführung verschiedener Tests und zum Ausdrucken von Ergebnissen.

Bemühungen, die auf eine Automatisierung der Steckbrettprogrammierung hinzielen, haben bisher noch keine praktischen Erfolge gebracht [46].

Durch eine allmähliche Erweiterung des Analogrechners mit digitalen Elementen (Logik, digitale Speicher) entstand als neuer Rechnertyp der Hybridrechner, eine Verknüpfung eines vollständigen Analogrechners mit einem vollständigen Digitalrechner. Über Entwicklung, Programmierung und Anwendungsmöglichkeiten solcher Hybridrechner steht eine umfangreiche Literatur zur Verfügung [38-50]. Die Entwicklung dieses Rechnertyps wurde durch die Nachteile des Analogrechners (eingeschränkte und ungenaue Darstellung von nicht-linearen Funktionen und Funktionen mehrerer Veränderlicher, keine Datenspeicherung über längere Zeit) angeregt.

Neben den beiden Rechnern (Analog- und Digital-) muß ein Hybrid-system ein umfangreiches Koppelwerk enthalten, das folgende Operationen durchführen muß:

1. Austausch von Dateninformation zwischen den beiden Rechnern.  
Die beiden wichtigsten Vorgänge sind dabei:
  - a) Umsetzung von analogen Daten in digitale und umgekehrt.
  - b) Auswahl von Daten aus einer Menge auf dem Analogrechner (paralleler Rechner) gleichzeitig anstehender Information und Einfügen dieser Daten in den serienmäßigen Ablauf des Digitalrechners und umgekehrt.

2. Steuerung und Koordination des Programmablaufs, wie z.B.

- a) Steuerung der Datenumsetzung
- b) Synchronisation zwischen den auf dem parallel- und dem serienorganisierten Anlagenteil ablaufenden Programmteilen.
- c) Steuerung der Betriebszustände des Gesamtsystems (Anfangsbedingungen, Rechnen, Halt).

Außer der geeigneten Hardware steht für diese Aufgaben eine umfangreiche Software von seiten der Hersteller zur Verfügung (HYTRAN [51], Real-Time-FORTRAN [52], HYTROL [53] usw.), so daß diese relativ komplizierten Abläufe durch einfache Befehle (z.B. auf einer erweiterten FORTRAN-Basis) eingeleitet und überwacht werden können.

Betrachtet man den Hybridrechner aus der Sicht des Analogrechners, so bildet er in folgenden Punkten eine wesentliche Erweiterung:

1. Automatisierung der Programmvorbereitung (Testen, Potentiometereinstellen).
2. Automatisierung des Rechenablaufs (Iteratives Rechnen, autom. Parametervariationen).
3. Speichern von Daten in großem Umfang und über beliebige Zeit.
4. Erweiterung der darstellbaren Funktionen und Operationen.
5. Erhöhung der Genauigkeit bei der Darstellung nichtlinearer Zusammenhänge.
6. Rechnen mit mehreren unabhängigen Variablen (Multi-Speed-Technik).

Diese Erweiterung erschließt dem Hybridrechner gegenüber dem Analogrechner neue Einsatzgebiete, wie z.B. komplizierte nichtlineare Systeme von Dgln., partielle Dgln. einschließlich Randwertproblemen [54], statistische Methoden zur Lösung dynamischer Probleme [12,13], Optimierungsaufgaben [55-62], Simulation von Totzeit [26] usw.

An den Digitalrechner eines Hybridsystems werden höchste Anforderungen in bezug auf Rechengeschwindigkeit gestellt, damit in Hybridrechnerprogrammen mit relativ großem Digitalanteil der Vorteil des schnellen Rechners auf der Analogseite nicht dadurch verlorengeht, daß der Digitalrechner nicht "mitkommt". Die Geschwindigkeit, mit der ein Problem auf einem Hybridrechner gelöst werden kann, wächst mit dem Ansteigen der digitalen Rechengeschwindigkeiten. Die Hybridrechnerentwicklung profitiert also in starkem Maße von der Entwicklung sehr schneller Digitalrechner.

### 2.3.2 Digitale Simulatoren

Stehen zur Lösung dynamischer Probleme mit dem Digitalrechner außer der FORTRAN-Sprache selbst keine weiteren Hilfsmittel zur Verfügung, so ist die Programmierung dynamischer Systeme sehr aufwendig und schwierig. Vor allem die Ausführung der Integrationen erfordert erheblichen Programmieraufwand. Eine Erleichterung ist es also schon, wenn Unterprogramme zur Verfügung stehen, die die Integration durchführen.

Gute Simulatoren sollen jedoch die Programmierarbeit noch weiter vereinfachen und die gesamte Organisation des Programms automatisch durchführen, so daß nur noch das zu lösende Problem in einer problemnahen Form eingegeben werden muß.

Es gibt bereits mehrere digitale Simulatoren, bei denen das zu lösende Problem entweder in Blockschaltbildform dargestellt wird (MIDAS [29,37], 1130 CSMP [30,31,32], IRE-DAS, <sup>RE)</sup> DYSAC [33], [34]) oder die zu lösenden Gleichungen in FORTRAN oder einer ähnlichen benutzernahen Form einzugeben sind (DSL/90 [35], CSSL [36], DYSYS). Bei allen Systemen, mit Ausnahme von CSMP, ist es nicht möglich, in die laufende Rechnung einzugreifen.

Ein Abschluß der Entwicklung von digitalen Simulatoren ist noch nicht abzusehen, weil einerseits die vorhandenen Systeme noch erhebliche Nachteile haben und andererseits die Hardware-Entwicklung, die einen entscheidenden Einfluß auf die Funktionsweise von Simulatoren hat (z.B. im Hinblick auf bequemere Ausgabe über Bildschirmgeräte oder Schreiber), weitere Fortschritte macht. Mit jedem Schritt der Weiter-

---

n) Digitaler Analogrechner-Simulator des Instituts für Reaktorentwicklung.

entwicklung von Hardware und Rechnerbetriebssystemen ergeben sich neue Möglichkeiten für bessere Simulatoren. In nächster Zukunft wird der Echtzeitbetrieb den Vorteil des Eingriffs in die laufende Rechnung (Dialogverkehr) bringen. Die Rechengeschwindigkeit läßt sich nur in kleinen Schritten steigern, so daß auch in Zukunft nicht damit gerechnet werden kann, daß digitale Simulatoren auch nur annähernd die Rechengeschwindigkeiten von Analogrechnern erzielen werden.

## 2.4 Ausgewählte Simulatoren

Für die Durchführung der in Kap. 3 beschriebenen Simulationen wurden neben dem Analogrechner mit digitaler Logik (sog. hybrider Analogrechner) drei digitale Simulatoren ausgewählt, die sich in ihrem Aufbau und ihrer Funktionsweise stark unterscheiden.

### 2.4.1 Analogrechner

Die verfügbare Analogrechenanlage umfaßt

- 1 Konsole PACE 231 RV mit VLG voll ausgebaut
- 1 " 231 R voll ausgebaut
- 1 " 231 R teilweise ausgebaut
- 1 " DOS 350

sowie das ADIOS-System für Test, Ein- und -Ausgabe.

Das System DOS enthält neben Logikelementen (Und-Glieder, Flip-Flops, Register usw.) 4 digitale Laufzeitspeicher mit je 256 Worten.

Die statische Genauigkeit der linearen Elemente beträgt  $10^{-4}$ , die Bandbreite der Verstärker ca. 30 kHz, die kleinste einstellbare Integrationskonstante (bei 1er-Eingang)  $10^{-4}$  sec. Ein digitales Hilfsprogramm DIANA zur Berechnung und Einstellung von Potentiometerwerten steht ebenfalls zur Verfügung 27. Die Blockschaltbild- und Steckbrett-Programmierung sowie die Funktionsweise dieses Analogrechners mit digitaler Ausbaustufe werden als bekannt vorausgesetzt.

#### 2.4.2 Continuous System Modeling Program (1130 CSMP)

1130 CSMP [31,32] ist ein digitaler Analogrechner-Simulator, der eine Konsole eines mittleren Analogrechners (25 Integratoren) auf einer IBM 1130 mit Plotter nachbildet. Das Programm ist eine Variante des Simulators Pactolus [30]. Die Eingabe ist blockorientiert, alle Rechenelemente des Analogrechners sind nachgebildet. Parameterwerte und Blockverknüpfungen werden in übersichtlichem Format entweder über Lochkarten oder Schreibmaschine eingegeben. Während des Rechenlaufs besteht eine enge Mensch-Maschine-Verbindung über die Schreibmaschine. Für die Darstellung beliebiger Funktionen (über FORTRAN-Unterprogramme) steht nur sehr wenig Speicherplatz zur Verfügung. Die Integrationen werden nach einem Integrations-Verfahren 2.Ordnung durchgeführt. Das System hat keine Schrittweitenautomatik.

#### 2.4.3 IRE - DAS

IRE - DAS [Digital Analog Simulator] ist ebenfalls ein blockschaltbildorientierter dynamischer Simulator, bei dem die wesentlichen Rechenkomponenten des Analogrechners nachgebildet sind. Zusätzlich sind häufig vorkommende Funktionen, wie Quadratwurzel, Exponentialfunktion, Potenzen, trigonometrische Funktionen in Blockform darstellbar. Außerdem können beliebige Funktionen, die in FORTRAN als Unterprogramme vorliegen müssen, eingesetzt werden. Ein Eingriff in die laufende Rechnung ist nicht möglich. Die Ausgabe erfolgt z.Zt. noch über Listen, ein Plot-Programm dazu ist in Vorbereitung. Die Integration wird mit Hilfe eines abgewandelten Runge-Kutta-Verfahrens 5.Ordnung mit Schrittweitenautomatik durchgeführt. Das Programm wurde für die hier durchgeführten Untersuchungen auf der IBM 7074 eingesetzt. Mittlerweile kann es auch auf der IBM 360/65 benutzt werden.

#### 2.4.4 DYSYS

Der Dynamic System Simulator <sup>33)</sup> DYSYS ist ein Beispiel für einen dynamischen Simulator, der nicht blockorientiert ist. Ein System von

<sup>33)</sup> Der Simulator DYSYS wurde von Herrn Dipl.-Ing. E.G. Schlechtendahl (Institut für Reaktorentwicklung) entwickelt und freundlicherweise für diese Arbeit zur Verfügung gestellt.

allgemeinen Dgln. muß in ein System von Dgln. 1. Ordnung umgewandelt werden. Dieses Gleichungssystem muß als FORTRAN-Unterprogramm geschrieben werden. Zur Integration wird ein Runge-Kutta-Verfahren 4. Ordnung benutzt. Die Schrittweitenautomatik wird über eine Vergleichsrechnung mit Hermiteschen Polynomen bis 4. Ordnung gesteuert. Die Ergebnisse werden über Listen oder Listen und Kurven (PLOT-Subroutine) ausgeliefert. Ein Eingriff in die laufende Rechnung ist nicht möglich. Das Programm wurde auf der IEM 7074 eingesetzt.

### 3. Problemanalyse

#### 3.1 Dynamische Probleme in der Reaktorentwicklung

Eine Bewertung verschiedener dynamischer Simulatoren ist nur möglich, wenn man zunächst die Probleme analysiert, die mit den Simulatoren gelöst werden sollen. Die Vor- und Nachteile eines Simulators können je nach Problem ganz verschiedenem Gewicht bekommen. So ist z.B. eine große Rechengenauigkeit bei der Simulation eines Reaktorkreislaufs mit einem Regelsystem zur Untersuchung des Lastfolgeverhaltens weniger wichtig als bei Exkursionsmodellen mit raumabhängiger Kinetik.

Zum Testen wurden Probleme ausgewählt, die in der Reaktorentwicklung, speziell bei der Entwicklung schneller Brutreaktoren, auftreten. Die erzielten Ergebnisse lassen sich jedoch auch auf andere Gebiete übertragen.

Zunächst werden die wichtigsten dynamischen Probleme, die im Rahmen der Entwicklung von schnellen Reaktoren anfallen, in einer Liste zusammengestellt:

- Thermodynamik des Kernes
- Raumabhängige Reaktorkinetik
- Exkursionsanalysen
- Dynamik von Abschaltssystemen
- Kreislaufkomponenten (z.B. Dampferzeuger)
- Kreislaufdynamik
- Reaktorregelung
- Reaktorstabilität
- Zweiphasenströmung

Na-Austreibung

Brennstabschwingungen

Balanced-Oszillator-Methode zur Bestimmung von  
Reaktivitätskoeffizienten und Stoffwerten des Brennstoffs

Dynamische Plastizitätstheorie

Kriechverhalten von Brennelementhüllen

Spaltproduktfreisetzung

Optimierungsprobleme

Zu diesen größeren dynamischen Problemen kommen noch zahlreiche alltägliche Routinearbeiten (z.B. Berechnung des Weg-Zeit-Verhaltens eines Abschaltstabes), deren Ausführung durch geeignete Simulatoren erleichtert werden muß. Diese kleineren Probleme sollen in diesem Vergleich jedoch nicht berücksichtigt werden, weil man dafür sowohl mit Digital- als auch mit Analogrechnern geeignete Simulatoren entwickeln kann.

Um die Problemanalyse übersichtlich zu gestalten, wurde nicht jede der in der Aufstellung aufgeführten Aufgaben einzeln analysiert. Da oft mehrere der Themen mit gleichen Lösungsmethoden behandelt werden oder die gleichen mathematischen Probleme enthalten, ist eine Aufstellung der Lösungsmethoden und der anfallenden speziellen mathematischen Probleme aufschlußreicher.

### 3.2 Lösungsmethoden

Die Lösungsmethoden lassen sich in 4 Gruppen einteilen:

- a) analytische Lösung
- b) Transformation
- c) statistische Methoden
- d) Simulation des Zeitablaufs

#### Zu a):

Analytische Lösungen sind nur in den seltensten Fällen möglich, deshalb ist man fast immer auf die 3 folgenden Lösungsmethoden angewiesen.

Zu b):

Manche Probleme lassen sich eleganter lösen, wenn man das Grundgleichungssystem einer Transformation unterzieht. Häufig gebraucht wird die Laplace-Transformation, z.B. bei der Balanced-Oszillator-Methode oder bei Stabilitätsuntersuchungen [9,10,11]. In vielen Fällen sind die transformierten Gleichungen keine Dgln., sondern algebraische Gleichungen, so daß dann eine Lösung mit dem Digitalrechner vorteilhafter ist.

Spezielle Transformationen sind immer mit Einschränkungen in der Anwendung verbunden. Systeme mit beliebigen Störfunktionen oder mit komplizierten nichtlinearen Zusammenhängen eignen sich z.B. nicht gut für die Laplace-Transformation.

Zu c):

Statistische Methoden zum Lösen von reaktordynamischen Problemen werden bisher nur vereinzelt angewendet (z.B. Monte-Carlo-Methode bei thermodynamischen Problemen [12]).

Die in den letzten Jahren erheblich erhöhten Rechengeschwindigkeiten bei Digitalrechnern und die Entwicklung großer Hybridrechnersysteme haben erst die Voraussetzungen für den sinnvollen Einsatz dieser Methoden geschaffen. Sie werden deshalb in Zukunft mehr in den Vordergrund treten.

Zu d):

Die direkte Simulation des Zeitablaufs (Integration von Dgln.) ist die am häufigsten verwendete Methode, weil sie übersichtlich und allgemein anwendbar ist. Besonders Analog- und Hybridrechner eignen sich für diese Methode, weil sie in der Lage sind, sehr schnell und kontinuierlich Integrationen durchzuführen [1-9].

Da sich bei der Lösung der meisten dynamischen Probleme die Simulation des Zeitablaufs als vorteilhaft erweist und diese Methode am häufigsten angewendet wird, dient sie auch als Grundlage der Vergleichsstudie.



### 3.3 Eigenschaften der Gleichungssysteme

Im folgenden werden die Gleichungssysteme, die die dynamischen Zusammenhänge der in Kap. 3.1 aufgeführten Probleme beschreiben, näher untersucht, wobei als Lösungsmethode immer die Simulation des Zeitablaufs vorausgesetzt wird. Die Aufschlüsselung der Eigenschaften erleichtert eine Übertragung der hier für Reaktorprobleme gefundenen Ergebnisse auf andere Anwendungsgebiete.

Die mathematische Beschreibung der zahlreichen technischen und physikalischen Probleme (s.Kap. 3.1) führt im wesentlichen immer wieder auf 6 verschiedene Gleichungstypen.

#### 3.3.1 Lineare Dgl.n. mit konstanten Koeffizienten

Nur wenige dynamische Vorgänge in der Reaktortechnik lassen sich in guter Näherung durch lineare Dgl.n. mit konstanten Koeffizienten darstellen. Diese Näherung eignet sich entweder für Voruntersuchungen komplizierter Systeme oder bei kleinen Störampplituden. Für Untersuchungen auf dem Gebiete der Reaktorregelung, Spaltproduktfreisetzung, Thermodynamik des Cores und Brennstabschwingungen [14] kommen sie zur Anwendung.

Der Analogrechner eignet sich für die Lösung dieser Gleichungssysteme besonders gut. Für größere Systeme von Dgl.n. (z.B. 50 Dgl.n. 1.Ordnung) geschieht das Lösen der Gleichungen größenordnungsmäßig 100 mal schneller als auf den schnellsten Digitalrechnern. Der Gewinn an Rechenzeit macht sich besonders dann bemerkbar, wenn entweder sehr viele Rechenläufe nacheinander durchgeführt werden müssen (z.B. Parametervariationen) oder wenn die Rechenläufe sehr lange dauern (bezogen auf die kleinste im System vorkommende Zeitkonstante, denn diese bestimmt die Rechengeschwindigkeit). Diese sehr langen Rechenläufe, bezogen auf die kleinste Zeitkonstante, treten besonders bei der Simulation der Thermodynamik eines Reaktorkernes auf. Die Zeitkonstante des Brennstoffs, die das Übergangsverhalten im wesentlichen bestimmt, liegt im Sekundenbereich. Deshalb sind z.B. zur Berechnung von Übergangsfunktionen Rechenläufe über mehrere Sekunden erforderlich. Die Zeitkonstanten der Aufheizung der Hülle und des Kühlmittels liegen jedoch im Millisekundenbereich und dürfen z.B. für eine genaue Temperatur- bzw. Reaktivitätsberechnung nicht vernachlässigt werden.

### 3.3.2 Partielle Dgln.

Alle dynamischen Vorgänge im Reaktorkern werden durch partielle Dgln. in Ort und Zeit beschrieben. Mit keinem Rechnertyp kann man diese Gleichungen direkt lösen. In seltenen Fällen ist eine analytische Lösung möglich, so z.B. in [15] mit Hilfe der Laplace-Transformation. Normalerweise ist man jedoch auf eine numerische Integration in Orts- und Zeitrichtung angewiesen [16,17,18], was zu Stabilitätsproblemen und sehr langen Rechenzeiten führen kann. Will man die Fähigkeit des Analogrechners, kontinuierliche Integrationen durchführen zu können, beim Lösen von partiellen Dgln. des Ortes und der Zeit ausnutzen, so kann dies näherungsweise durch eine örtliche Zonenaufteilung geschehen (Ortsdiskretisierung). Für jede Zone wird eine gewöhnliche Dgln. angesetzt, so daß eine partielle Dgl. durch einen Satz von gewöhnlichen Dgln. ersetzt wird [9,25]. Dadurch wird ein erheblicher Gewinn an Rechengeschwindigkeit erzielt. Eingeschränkt wird diese Methode durch die Kapazität des Analogrechners. Sie bestimmt die maximale Zonenzahl und damit die Genauigkeit der Lösung. Mit Hilfe eines Hybridrechners kann dieser Nachteil auf Kosten der Rechenzeit behoben werden, indem die gleiche Rechenschaltung mehrfach benutzt wird und die Integrationen teilweise nacheinander durchgeführt werden [5,7,19,20,21]. Außerdem bietet sich bei Hybridrechnern die Möglichkeit der kontinuierlichen Integration in einer Ortsrichtung an (Zeitdiskretisierung, serielle Lösung) [20,22,23,24].

### 3.3.3 Nichtlineare Dgln.

Nichtlineare Dgln., die nicht in erster Näherung linearisiert werden können, treten vor allem bei Neutronenkinetik, Zweiphasenströmung, Natriumaustreibung und Kreislaufdynamik auf.

Bei der numerischen Integration nichtlinearer Dgln. treten gegenüber den linearen Dgln. zusätzlich Stabilitätsprobleme bei der Bestimmung der maximalen möglichen Schrittweite auf. Es gibt kein allgemeingültiges Verfahren, das notwendige und hinreichende Bedingungen zur Bestimmung der Stabilitätsgrenze liefert. Bei den einzelnen Integrationsverfahren ist man also gezwungen, einen relativ

großen Abstand von der Stabilitätsgrenze zu halten, d.h. mit Schrittweiten zu rechnen, die weit unter der nicht bekannten Grenzschrittweite liegen.

Beim Analogrechner wirkt sich die größere Ungenauigkeit der nichtlinearen Elemente (Multiplizierer, Funktionsgeneratoren) nachteilig aus. Bei Systemen von Dgl. mit sehr vielen stark nichtlinearen Zusammenhängen kann dabei die Genauigkeit soweit zurückgehen, daß Analogrechnerlösungen nicht mehr brauchbar sind. Mit Hilfe eines Hybridrechners kann dieser Nachteil behoben werden, wenn man die nichtlinearen Zusammenhänge auf dem digitalen Teil des Rechners darstellt. Besteht ein System aus nur wenigen, aber stark nichtlinearen Zusammenhängen, so ist die Simulation auf dem Digitalrechner oft vorteilhafter als auf dem Analogrechner.

### 3.3.4 Nichtlineare Funktionen mehrerer Veränderlicher

Kühlmittelstoffwerte und Wärmeübergangszahlen hängen von mehreren Veränderlichen ab. Deshalb ist es erforderlich, bei der Thermodynamik des Cores, der Kreislaufdynamik, Reaktorregelung und Zweiphasenströmung diese Abhängigkeit darzustellen. Auf dem Analogrechner kann dies nur in vereinfachter Form (Trennung der Veränderlichen) geschehen. Die Modelle für den dampfgekühlten schnellen Reaktor haben gezeigt, daß Rechnungen mit ausreichender Genauigkeit nur in einem engen Druckbereich ( $P_0 \pm 20$  at) durchführbar sind [8]. Tritt anstelle des Analogrechners ein Hybridrechner, so können diese Schwierigkeiten weitgehend behoben werden, wenn im digitalen Teil der Maschine genügend Speicherplatz zur Darstellung der Funktionen vorhanden ist.

Beim Digitalrechner gibt es prinzipiell keine Einschränkungen bei der Darstellung von Funktionen mehrerer Veränderlicher. Die Abhängigkeiten können sowohl in Gleichungs- als auch in Tabellenform programmiert werden.

Während der Durchführung der Rechnungen zu dieser Studie war die Darstellung mehrerer komplizierter Funktionen mehrerer Veränderlicher (z.B. aller Stoffgrößen von Wasser und Wasserdampf) noch durch

die relativ geringe Speicherkapazität der IBM 7074 erheblich eingeschränkt, so daß teilweise noch mit Näherungspolynomen gearbeitet werden mußte. Mit dem Einsatz der größeren Rechenmaschinen im Kernforschungszentrum ist diese Einschränkung weitgehend aufgehoben.

Viele dynamische Zusammenhänge in der Reaktortechnik werden sowohl durch eine große Anzahl von Dgln. (vorteilhaft: Analogrechner) als auch durch nichtlineare Funktionen mehrerer Veränderlicher (vorteilhaft: Digitalrechner) beschrieben. Für solche Probleme ist ein Hybridrechner besonders gut geeignet.

### 3.3.5 Transzendente Gleichungen

Treten in einem Problem Gleichungen auf, die nicht explizit gelöst werden können (z.B. Kühlmitteltemperatur bei druckabhängiger Kühlmitteldichte [8]), so ist eine iterative Lösung erforderlich. Bei schlechter Konvergenz werden bei digitaler Lösung die Rechenzeiten sehr lang. Mit dem Analogrechner lassen sich solche Gleichungen sehr schnell ( $10^{-4}$ - $10^{-3}$  sec) lösen, indem man die implizite Gleichung künstlich zu einer Dgl. 1. Ordnung mit der kleinsten darstellbaren Zeitkonstanten (offener Verstärker) macht. In den meisten Fällen wird durch dieses kontinuierliche Lösen der impliziten Gleichung die Rechenzeit für die Lösung des gesamten Problems gar nicht oder nur unwesentlich beeinflusst.

### 3.3.6 Totzeitsysteme

In Rohrleitungen, Wärmetauschern und Reaktorkernen treten reine Transportverzögerungen (Totzeiten) auf. Mit einem gewöhnlichen Analogrechner ist man auf Näherungen (Verzögerungen, Padé-Approximation) angewiesen [63]. Mit dem hybriden Analogrechner HYDAC 2000, der einen kleinen digitalen Laufzeitspeicher besitzt, ist in beschränktem Umfang eine Simulation von Totzeitgliedern möglich (maximal 4 Glieder, minimale Totzeit 0,5 sec [26]). Bei Hybrid- und Digitalrechner sind digitale Unterprogramme zur Totzeitsimulation erforderlich, die jedoch mit relativ geringem Programmieraufwand erzeugt werden können.

### 3.4 Testbeispiele

Zur quantitativen Analyse wurden die vier folgenden Testbeispiele ausgewählt. Um mit einem möglichst geringen Programmier- und Rechenaufwand auszukommen, wurden relativ kleine und einfache Probleme ausgewählt, wobei jedoch darauf geachtet wurde, daß eine Extrapolation zu größeren Systemen möglich ist. Die Testbeispiele wurden so ausgewählt, daß die wichtigsten der in 3.3 beschriebenen Probleme abgedeckt sind.

#### 3.4.1 Brennstab eines Na-gekühlten schnellen Reaktors

Als erstes Beispiel wurde ein Brennstab eines Na-gekühlten Reaktors gewählt. Die Thermodynamik wird durch partielle Dgl'n. beschrieben (3.3.2). Zur Lösung dieser Gleichungen wird die Methode der Raumdiskretisierung (Zonenaufteilung) benutzt. Unter der Annahme, daß alle Stoffwerte konstant sind, entsteht ein System von linearen Dgl'n. mit konstanten Koeffizienten (3.3.1) [9]. Die Zeitkonstanten liegen zwischen 4 sec (Brennstoff) und 4 msec (Kühlmittel). Zusätzlich wird die Transportzeit des Kühlmittels in axialer Richtung durch eine Padé-Approximation 2.Ordnung, die zwischen zwei Kühlkanalzonen geschaltet wird, berücksichtigt (3.3.6).

Zunächst war eine Aufteilung in 25 Zonen (5 axial, 5 radial, 3 in Brennstoff und je 1 in der Hülle und im Kühlkanal) vorgesehen. Das entspricht der maximal möglichen Zonenaufteilung, wenn das Problem auf einer Konsole des Analogrechners PACE 231 RV gelöst werden soll. Mit der Totzeitsimulation hätte dies zu einem System von 34 Dgl'n. 1. Ordnung geführt. Da jedoch einer der Digitalsimulatoren (GSMP) auf 25 Dgl'n. beschränkt ist, wurde die Zonenaufteilung auf 15 reduziert (5 axiale Zonen, radial je 1 Zone für Brennstoff, Hülle und Kühlmittel). Mit diesem Modell wurden sprunghafte Leistungs- und Eintrittstemperaturstörungen berechnet. Ziel war die Berechnung der wichtigsten Temperaturen in Abhängigkeit von der Zeit über 30 sec. Der Brennstab wurde in 2 Versionen untersucht:

Testbeispiel 1a: Brennstab mit Aufteilung in 15 Zonen,  
ohne Berücksichtigung der Transportzeit  
im Kühlmittel

Testbeispiel 1b: Brennstab mit Aufteilung in 15 Zonen,  
mit Berücksichtigung der Transportzeit  
im Kühlmittel (Padé-Appr. 2.Ordnung).

### 3.4.2 Brennstab eines dampfgekühlten schnellen Reaktors

Als zweites Beispiel wurde die Thermodynamik des Dampf-Core gewählt. Die Dynamik ist in stark vereinfachter Form dargestellt. Ein Brennstab steht stellvertretend für das ganze Core; die partiellen Dgln. der Thermodynamik werden wie in 3.4.1 durch Zonenaufteilung in gewöhnliche Dgln. verwandelt. Hier werden nur 3 Zonen gewählt (Brennstoff, Hülle, Kühlmittel), denn bei dieser Simulation steht nicht wie in 3.4.1 die Problematik eines Systems mit sehr vielen Dgln. im Vordergrund. Es soll hier vielmehr gezeigt werden, wie Funktionen mehrerer Veränderlicher (3.3.4) und iterative Prozesse dargestellt werden können. Diese Probleme treten in den Gleichungen des Kühlkanals ( $\xi = f(\mathcal{V}, p)$ ,  $\Delta p = f(\xi, \dot{m})$ ,  $\mathcal{V} = f(\dot{m}, \xi, cp, P)$ ) auf.

Ziel ist die Berechnung der wichtigsten Übergangsfunktionen über 30 sec.

Testbeispiel 2: Brennstab eines dampfgekühlten schnellen Brutreaktors

### 3.4.3 Reaktorkinetik mit Rückführung

Die punktreaktorkinetischen Gleichungen (6 Gruppen verzögerter Neutronen), verbunden mit einer Reaktivitätsrückwirkung über die Thermodynamik des Dampf-Cores (Testbeispiel 2) und 2 Reaktivitätskoeffizienten, wurden als einfaches Beispiel einer Core-Simulation gewählt. Damit wird die Simulation eines Systems von nichtlinearen Dgln. repräsentiert, in dem verschiedene physikalische Prozesse, die mit stark unterschiedlichen Geschwindigkeiten ablaufen, miteinander verknüpft sind. Die wichtigsten Übergangsfunktionen werden bei einer Störung von + 0,2 % über 30 sec berechnet. Es werden wieder 2 verschiedene Versionen des Modells untersucht:

Testbeispiel 3a: Neutronenkinetik mit Rückführung,  $l = 0$

3b: Neutronenkinetik mit Rückführung,  $l \neq 0$

$l =$  Lebensdauer der schnellen Neutronen

#### 3.4.4 Na-Dampfblase

Das Anwachsen einer Na-Dampfblase bei steigender Na-Temperatur wird durch extrem nichtlineare Gleichungen beschrieben (4. Grades, 2. Ordnung).

Durch 3 extreme Bedingungen ist die Simulation äußerst schwierig:

1. Die Zeitkonstanten der (linearisierten) Gleichungen für die Temperatur und den Blasenradius unterscheiden sich um 8 Zehnerpotenzen.
2. Da sich der Radius um 5-6 Zehnerpotenzen ändert, tritt bei  $r^4$  eine Veränderung um 20-25 Zehnerpotenzen auf.
3. Die Radiusgleichung enthält eine positive Rückführung, d.h. das System kann instabil werden.

Diese außergewöhnlichen Bedingungen erschweren die Simulation des Problems. Beim Analogrechner ist es schwierig, ein instabiles System über sehr lange Zeit (bezogen auf die kleinste Zeitkonstante) zu simulieren. Durch die vorhandenen Driftspannungen, die in einem instabilen System verstärkt werden, sind Rechnungen über mehr als die 1000-fache Zeitkonstante nicht möglich. Hier wird aber verlangt, daß über  $10^8$ - $10^{10}$  Zeitkonstanten integriert wird. Die Driftspannungen führen dazu, daß das instabile Verhalten des Systems (sehr starkes Anwachsen des Radius) viel zu früh auftritt.

Bei der digitalen Simulation tritt ein entgegengesetzter Effekt auf. Das System verharrt in seinem Ausgangszustand. Durch die kleinste Zeitkonstante (ca.  $10^{-9}$  sec) in der Blasengleichung wird die Zeitschrittweite bestimmt. Die Schrittweite ist aber so klein, daß in der Temperaturgleichung (Zeitkonstante 0,1 sec) während eines Schrittes Temperaturänderungen  $\Delta T = T_2 - T_1 < 10^{-8} T_1$  auftreten. Da mit der IBM 7074 nur eine Genauigkeit von  $10^{-8}$  erreichbar ist, bleibt bei  $\Delta T < 10^{-8} T_1$  immer  $T_2 = T_1$ , d.h.  $T = \text{const}$ . Eine Erhöhung der Ge-

nauigkeit würde hier zwar weiterhelfen, wegen der sehr kleinen Schrittweite wären die Rechenzeiten jedoch unerträglich hoch.

Eine Möglichkeit, die sowohl die analoge als auch die digitale Lösung mit den gegebenen Rechenmaschinen ermöglicht, ist die quasistationäre Lösung der Radiusgleichung bis zu dem Zeitpunkt, in dem der Radius sehr stark zu wachsen beginnt (Beginn der Instabilität). Das Testbeispiel wurde also auf zwei verschiedene Arten simuliert.

Testbeispiel 4a: Na-Dampfblase, genaue Simulation der Dynamik

4b: Na-Dampfblase, quasistationäre Lösung der Radiusgleichung während der stabilen Anfangsphase.

#### 4. Systemvergleich

Eine Gegenüberstellung der vier Simulatoren erfolgt nach den in Abschn. 2.1 aufgestellten Kriterien zur Beurteilung dynamischer Simulatoren. Die problembezogenen Zahlenangaben stammen aus Rechnungen mit den in Abschnitt 2.4 ausgewählten Testbeispielen.

##### 4.1 Systemeigenschaften

###### 4.1.1 Darstellbare Funktionen

Es wird geprüft, in welchem Umfang die gebräuchlichen mathematischen Funktionen mit den einzelnen Simulatoren dargestellt werden können.

Analogrechner: +, -,  $\pi$ , /, Quadratwurzel, Integration, trigonometrische Funktionen, Exponentialfunktionen, Speichern in beschränktem Umfang (1 K), Totzeit, logische Verknüpfungen, näherungsweise Darstellung beliebiger nichtlinearer Funktionen über Funktionsgeneratoren.



GSMP: +, -,  $\pi$ , /, Quadratwurzel, Integration, Speichern, spezielle stückweise lineare Funktionen (Betrag, Begrenzung) näherungsweise Darstellung beliebiger nicht-linearer Funktionen über Funktionsgeneratoren. Über FORTRAN-Unterprogramme können max. 5 beliebige Funktionen von max. 3 Variablen dargestellt werden, sie erhöhen jedoch die Rechenzeit erheblich, weil diese FORTRAN-Unterprogramme in Overlay-Technik verarbeitet werden.

IRE - DAS: +, -,  $\pi$ , /, trigonom. Funktionen, Quadratwurzel, Exponentialfunktionen, Potenzieren, Integration, logische Verknüpfungen (Verzweigung). Max. 100 beliebige Funktionen von 4 Veränderlichen sind über FORTRAN-Unterprogramme darstellbar.

DYSYS: Alle in FORTRAN darstellbaren Funktionen sind erlaubt. Andere Funktionen müssen der Programmbibliothek entnommen oder selbst programmiert werden. Eine Beschränkung der darstellbaren Funktionen ist praktisch nur über den Speicherplatz oder die Rechenzeit gegeben.

#### 4.1.2 Rechenkapazität

Da die Simulatoren in ihrem Aufbau sehr unterschiedlich sind, gibt es keine exaktes Maß für die Rechenkapazität. Beim Vergleich der blockorientierten Simulatoren ist z.B. die max. mögliche Anzahl von Blöcken kein genaues Maß, weil bei den einzelnen Simulatoren die Blöcke unterschiedlichen Aufbau und Umfang haben. Ein besseres Maß ist die max. Anzahl der Integrationsblöcke, die die max. mögliche Zahl von Dgln. 1.Ordnung angibt. Da DYSYS nicht blockorientiert ist, wird in diesem Fall ein Block einer Gleichung im DYSYS gleichgesetzt.

Die digitalen Simulatoren besitzen in FORTRAN programmierbare Unterprogramme. Ein Maß für den Umfang dieser Programme ist neben ihrer

Anzahl der freie Speicherplatz im Kernspeicher. Die Werte beziehen sich für IRE - DAS und DYSYS auf die IBM 7074, für CSMP auf die IBM 1130 mit 8K-Kernspeicher (s.u.).

Eine Erweiterung der Kapazität ist beim Analogrechner durch Anschluß weiterer Konsolen möglich, ohne daß die Rechengeschwindigkeit beeinflusst wird. Grundsätzlich ist beim digitalen Simulator wegen der seriellen Organisation des Digitalrechners eine Ausweitung der Kapazität z.B. durch Übergang auf einen Rechner mit größerem Kernspeicher, immer mit einer Verringerung der Rechengeschwindigkeit verbunden.

In der folgenden Tabelle werden die gesamte Blockzahl, die Anzahl der Integrationsblöcke (DYSYS: Anzahl der Dgln.) und der freie Speicherplatz in K-Worten für die FORTRAN-Programme angegeben.

Die für den Analogrechner angegebene Zahl von Speicherplätzen bezieht sich auf den digitalen Umlaufspeicher, in dem nur Daten über den A/D-Wandler gespeichert werden können.

	Analogrechner	CSMP	IRE-DAS	DYSYS
Max. Anzahl von Blöcken	350 <sup>*)</sup>	75	200	50
Max. Anzahl von Integrationsblöcken	68	25	100	50
Max. Speicherplatz für FORTRAN-Unterprogramme	1 K	0,4 K	1,9 K	4 K

\*) Anzahl der Analogrechner-Komponenten mit Ausnahme der Potentiometer, weil diese in den digitalen Simulatoren auch nicht als Blöcke bezeichnet sind.

#### 4.1.3 Rechengeschwindigkeit

An dieser Stelle sollen keine Aussagen über max. mögliche Rechengeschwindigkeiten gemacht werden, vielmehr werden die Rechenzeiten, die für die Berechnung der Testbeispiele (Abschn. 2.4) auftraten, gegenübergestellt. Für die Beispiele 1-3 lief die Rechnung über 30 sec (Problemzeit), für Beispiel 4 (instabiles System) über 2,5 sec. Die Rechenzeiten sind in der Tabelle auf Seite 24 zusammengefaßt. Für den Analogrechner werden 2 Zeiten angegeben, die eine entspricht der Echtzeitsimulation, die zweite (in Klammern) wird erreicht, wenn man auf den relativ trägen Schreiber verzichten kann, wenn also nur zeitliche Endwerte von Interesse sind oder eine Darstellung auf dem Oszillographen gewählt wird. Bei den Digitalsimulatoren ist neben der Rechenzeit die Anzahl der Rechenschritte in eckigen Klammern angegeben.

Lange Rechnungen (Beispiel 1b, 3b) mit den Digitalsimulatoren wurden nach einer Stunde abgebrochen. Mit der beim Abbruch eingestellten Schrittweite wurde die Rechenzeit auf 30 sec Problemzeit extrapoliert.

Im Beispiel 1 (lineares System mit vielen Dgln. unterschiedlicher Zeitkonstanten) zeigt sich der Vorteil des Analogrechners in bezug auf Rechengeschwindigkeit am deutlichsten. Verglichen mit der Echtzeitsimulation auf dem Analogrechner ist der schnellste Digitalsimulator (IRE-DAS) im Beispiel 1a um den Faktor 26 und im Beispiel 1b um den Faktor 290 langsamer. Vergleicht man mit der auf unserem Analogrechner max. möglichen Rechengeschwindigkeit, so wird der Unterschied noch um eine Zehnerpotenz größer.

Im Beispiel 2 (nur 2 Dgln.) wird der Unterschied geringer. Er wäre noch geringer, wenn bei den Digitalsimulatoren die Dampfstoffgrößen so vereinfacht dargestellt worden wären wie beim Analogrechner. Da jedoch bei IRE-DAS und DYSYS die Möglichkeit gegeben war, exaktere und über größere Bereiche gültige Dampffunktionen einzusetzen, wurde diese Möglichkeit, die eine größere Genauigkeit (besonders bei großen Störungen) auf Kosten der Rechengeschwindigkeit liefert, auch ausgenutzt. CSMP führt mit Ausnahme von Testbeispiel 2 und 3a zu unerträglich langen Rechenzeiten (bis zu 140 Std.).

Simulator Beispiel	Analog- rechner	CSMP	IRE-DAS	DYSYS
1a	30 sec (3)	32000 sec [30000]	780 sec [2050]	1000 sec [2840]
1b	30 sec (3)	80000 sec [60000]	8600 sec [17000]	14800 sec [70000]
2	30 sec (0.3)	800 sec [600]	150 sec [207]	390 sec [144]
3a	30 sec (0.3)	1000 sec [600]	240 sec [268]	420 sec [180]
3b	30 sec (30)	500000 sec [300000]	nicht durchgeführt	38000 sec [97000]
4a	Rechnung wegen zu großen Unterschieden in den Zeitkonstanten nicht durchführbar			
4b	Simulation noch nicht durchgeführt	Lösung wegen konstanter Schrittweite nicht möglich	nicht durchführbar wegen konstanter Schrittweite beim Ausdrucken	180 sec [260]

( ) = max. mögliche Rechengeschwindigkeit [ ] = Anzahl der Rechenschritte

Rechengeschwindigkeit der verschiedenen Simulatoren

Die Lösung von Beispiel 4b ist mit IRE-DAS nicht möglich. Eine Simulation auf dem Analogrechner führt voraussichtlich zu ähnlichen Rechenzeiten wie sie mit DYSYS erzielt wurden. Diese Simulation wurde noch nicht durchgeführt.

#### 4.1.4 Genauigkeit

Der Genauigkeitsvergleich wurde anhand von Testbeispiel 1a und 1b durchgeführt, und zwar wurden die Variablenwerte nach 30 sec Problemzeit benutzt und mit der exakten Lösung (Genauigkeit ca.  $10^{-6}$ ) verglichen.

Folgende mittlere Abweichungen vom wahren Wert wurden für 3 Variable (Brennstoff-, Can- und Kühlmitteltemperatur) aus 2 Rechenläufen (Temperatur- und Leistungsstörung) ermittelt:

Simulator	Analogrechner	CSMP	IRE-DAS	DYSYS
mittlerer relativer Fehler	$10^{-3}$	kein Ergebnis	$2 \cdot 10^{-4}$	$10^{-4}$

Bei CSMP konnte die Genauigkeit nach 30 sec Problemzeit nicht bestimmt werden, weil kein 30 sec-Lauf durchgeführt werden konnte. Für die Digitalsimulatoren war als Genauigkeitsparameter  $10^{-3}$  angegeben worden. Grundsätzlich hätte die Genauigkeit noch um 1-2 Zehnerpotenzen verbessert werden können, allerdings auf Kosten der Rechengeschwindigkeit.

#### 4.1.5 Einsatzbereitschaft und Verfügbarkeit über die Ergebnisse

Da beim Analogrechner und CSMP die Rechenmaschine vom Benutzer über längere Zeit belegt werden (Stunden oder einige Tage), sind die Wartezeiten unterschiedlich. Da sie jedoch im Einzelfall im voraus bekannt sind, ist dies kein schwerwiegender Nachteil. Nach Bereitstellung der Rechenmaschine fallen die Ergebnisse schon während der Rechnung in Kurvenform (beim CSMP auch in Tabellenform) an. Beim Analogrechner können gleichzeitig 3 Variable mit dem Plotter in Kurvenform

in Abhängigkeit von der Zeit dargestellt werden. Am Ende der Rechnung stehen die Endwerte aller Variablen zur Verfügung. Beim CSMP können eine Variable in Kurvenform und 5 Variable in Tabellenform dargestellt werden, am Ende eines Rechenlaufs stehen ebenfalls alle Endwerte der Variablen zur Verfügung.

Die beiden Digitalsimulatoren IRE-DAS und DYSYS müssen in den Betriebsplan der IBM 7074 eingefügt werden. Der Beginn der eigentlichen Rechnung ist nicht bekannt; da man nicht in die laufende Rechnung eingreifen kann, ist er aber auch nicht von Interesse. Hier wird also nicht unterschieden zwischen einer Wartezeit bis zur Bereitstellung der Maschine zum Rechnen und einer Wartezeit bis zur Auslieferung der Ergebnisse. Nur die Gesamtzeit zwischen Abgabe des Laufs und Rückgabe der Ergebnisse ist meßbar und von Interesse. Sie beträgt je nach Rechenzeit (Expresßlauf, Normallauf, Langlauf) 3 Stunden bis einige Tage. Verlangt man eine Kurvendarstellung der Ergebnisse, so kommen nochmals 1-2 Tage dazu. Diese langen Wartezeiten sind besonders in der Aufbauphase eines Programms, wenn viele Testläufe notwendig sind, von Nachteil. Mit der neuen IBM 360/65 werden geringere Umlaufzeiten erwartet.

## 4.2 Prozeßvorbereitung

### 4.2.1 Programmieraufwand

Beim Analogrechner und bei CSMP und IRE-DAS ist die Eingabe des Problems blockorientiert. Für alle 3 Simulatoren können nahezu die gleichen Blockschaltbilder verwendet werden. Grundsätzlich ist die Kenntnis von FORTRAN nicht erforderlich (mit Ausnahme der frei programmierbaren Unterprogramme bei CSMP und IRE-DAS). Die Umsetzung des Blockschaltbildes in die Programmiersprache ist in allen 3 Fällen ungefähr gleich aufwendig (Analogrechner: Stecken der Verknüpfungen und Erstellung eines Lochstreifens mit der Koeffizientenliste, IRE-DAS und CSMP: Darstellung der Verknüpfungen und Koeffizienten in vorgegebenem Format auf Lochkarten).

Bei DYSYS müssen die Dgln. in Dgln. 1. Ordnung umgewandelt werden und als FORTRAN-Statements in ein Unterprogramm eingesetzt werden. Diese Darstellungsweise ist nicht so übersichtlich wie die Blockdarstellung,

der Programmieraufwand ist jedoch dann wesentlich geringer, wenn viele Dgln. den gleichen Aufbau oder gar gleiche Koeffizienten haben (z.B. beim Aufteilen eines Brennstabes in mehrere axiale Zonen).

Zur reinen Programmierarbeit kommt der Aufwand für die Organisation des Programmablaufs. Beim Analogrechner sind dies im wesentlichen die Wahl des Rechenzustandes (Initial Condition, Operate, Hold, Pot-set) und die Auswahl der darzustellenden Variablen.

Die Digitalsimulatoren sind so aufgebaut, daß sie mit wenigen Kontrollparametern zur Organisation der Eingabe, Rechnung und Ausgabe auskommen. Nicht berücksichtigt sind hierbei die Job-Control-Karten, Klartext (Überschriften) und die Blocktypenkennzeichnung, die zur eigentlichen Eingabe gezählt wird.

Kontrollparameter	CSMP	IRE-DAS	DYSYS
für Eingabe	3	8	4
für Rechnung	12	5	12
für Ausgabe	7+n	1+n	12+n

n = Anzahl der ausgegebenen Variablen

IRE-DAS ist von den Kontrollparametern her am einfachsten zu handhaben, es bietet jedoch auch den geringsten Komfort (z.B. keine Plot-Ausgabe).

#### 4.2.2 Übersichtliche Struktur der Programmiersprache

Bei den 3 blockorientierten Simulatoren (Analogrechner, CSMP, IRE-DAS) ist die Problemdarstellung sehr übersichtlich, weil alle Verkopplungen graphisch dargestellt sind und deshalb der Signalfluß leicht verfolgt werden kann. In der programmierten Form ist die Steckbrettdarstellung übersichtlicher als der Lochkartenstapel für CSMP

und IRE-DAS, weil auf dem Steckbrett jedem Rechenelement ein fester Platz zugeordnet ist und die Verknüpfungen zwischen den Elementen sichtbar bleiben.

#### 4.2.3 Testmöglichkeiten

Da die Teststrategie stark problemabhängig ist, gibt es keine allgemeinen, vollständigen Testprogramme. Verschiedene Testhilfen, die die häufig vorkommenden Testoperationen vornehmen, sind jedoch für den Analogrechner vorhanden. Mit dem digitalen Hilfsprogramm DIANA und dem Steuersystem ADIOS lassen sich die Testwerte der Variablen ausrechnen und mit den Analogrechner-Werten vergleichen. Eine weitere Hilfe ist die Möglichkeit, die Ableitung der Variablen an den Integratoren zu überprüfen. Für die Digitalsimulatoren stehen keine Testhilfen zur Verfügung, man muß sie sich im Einzelfall selbst erstellen.

Das Austesten eines Programms kann beim Analogrechner und beim CSMP wesentlich schneller erfolgen als beim IRE-DAS und DYSYS. Grund für die schnelle Testmöglichkeit ist die enge Mensch-Maschine-Verbindung bei Analogrechner und CSMP. Jede Variable kann zur Fehlersuche augenblicklich ausgewählt werden. Notwendige Korrekturen können sofort durchgeführt werden. Bei IRE-DAS und DYSYS liegt zwischen Fehlererkennung und Korrektur mindestens die Umlaufzeit für ein Digitalprogramm (bei kurzen Rechenläufen mindestens 3 Stunden). Das Austesten eines mittelgroßen Programms (eine Analogrechnerkonsole) dauert im Durchschnitt beim Analogrechner weniger als einen halben Tag. Bei IRE-DAS und DYSYS ist bei einem gleich großen Problem auf der IBM 7074 mit Testzeiten von mehr als 1 Woche zu rechnen.

#### 4.3 Prozeßablauf

##### 4.3.1 Verfolgung der laufenden Rechnung

Nur beim Analogrechner und beim CSMP kann der Verlauf der Variablen während der Rechnung verfolgt werden. Beim Analogrechner können die zu beobachtenden Variablen auch während des Rechenlaufs ausgetauscht werden, beim CSMP ist dazu eine kurze Unterbrechung nötig. Bei IRE-DAS und DYSYS ist keine Beobachtung der laufenden Rechnung möglich.



#### 4.3.2 Eingriff in die laufende Rechnung

Sowohl beim Analogrechner als auch beim CSMP ist es möglich, in die laufende Rechnung einzugreifen, und zwar durch Abbrechen der Rechnung oder Anhalten, um Parameter oder die Struktur zu ändern.

Bei IRE-DAS und DYSYS kann während der Rechnung nicht eingegriffen werden. Ein vorprogrammierter Eingriff ist jedoch möglich. Bei DYSYS können sowohl Parameter als auch die Struktur nach Erfüllung vorgegebener Bedingungen variiert werden. Bei IRE-DAS sind solche Variationen nur in beschränktem Maße über die IF-Blöcke möglich. Da der Zeitverlauf der Variablen oft nur sehr schwer abgeschätzt werden kann, ist die Möglichkeit des vorprogrammierten Eingriffs kein gleichwertiger Ersatz für die Möglichkeit, zu jeder beliebigen Zeit jede beliebige Änderung von Struktur und Parametern durchzuführen.

Die Möglichkeit zur Verfolgung der laufenden Rechnung, verbunden mit der Möglichkeit des Eingriffs (beim Analogrechner und CSMP) führen in der Praxis zu einer starken Reduzierung unnötiger Rechenzeit, weil die Rechnung sofort abgebrochen werden kann, wenn sie genügend Information geliefert hat oder wenn aus dem Verlauf der Variablen sichtbar wird, daß ein Programm- oder Eingabefehler aufgetreten ist.

### 5. Zukünftige Entwicklung

Die durchgeführten Untersuchungen basieren auf einer Hardware und Software, wie sie im Herbst 1968 zur Verfügung stand.

In einem kurzen Überblick soll gezeigt werden, wie sich die Hardware weiterentwickelt hat und welche Möglichkeiten sich in den nächsten 2-3 Jahren infolge der verbesserten Hardware für eine Weiterentwicklung der Software bieten.

#### 5.1 Hardware

Bereits die Inbetriebnahme der IBM 360/65 im Kernforschungszentrum Karlsruhe brachte eine erhebliche Verbesserung der beiden Digitalsimulatoren IRE-DAS und DYSYS. Durch wesentlich größere Kernspeicher ist die Rechenkapazität (Kap. 4.1.2) erheblich vergrößert worden. So kann z.B. für FORTRAN-Unterprogramme die Anzahl der Speicherplätze von 1,9 K (bzw. 4 K) ohne Schwierigkeiten auf 30-40 K (Worte) erhöht

werden. Auch die max. mögliche Anzahl von Blöcken wird zur Zeit erhöht.

Erste Vergleiche in bezug auf Rechengeschwindigkeit (Kap. 4.1.3) ergaben auf der IBM 360/65 eine Verbesserung um den Faktor 6-10 gegenüber der IBM 7074. Ein weiterer Vorteil ist die kürzere Umlaufzeit, die zur Zeit bei kleineren Problemen unter einer Stunde liegt.

Mit der erwarteten Einführung des Telefunken-Systems TR 440, der Satellitenrechner TR 86 und der Sichtgeräte STG 100 ergeben sich weitere Verbesserungen, besonders im Hinblick auf einen engeren Mensch-Maschine-Kontakt (Direktzugriffssystem), so daß das Verfolgen und Beeinflussen der laufenden Rechnung möglich wird.

Die Analog- und Hybridrechnerentwicklung hat in den vergangenen Jahren erhebliche Fortschritte gemacht, so daß die Daten des hier benutzten hybriden Analogrechners als überholt gelten. Moderne transistorisierte Analog- und Hybridrechner [43,44,49] sind um mehr als den Faktor 10 schneller und um den Faktor 2-5 genauer. Die Erweiterung der Analogrechner zu Hybridrechnern hat vor allem den Nachteil der beschränkten Funktionsdarstellung aufgehoben.

## 5.2 Software

Mit der Verbesserung der Hardware ist auch eine Verbesserung der Software möglich geworden. So ist es jetzt z.B. möglich, für das IRE-DAS eine Plot-Subroutine einzusetzen, damit die Ausgabe der Ergebnisse auch in Kurvenform geschehen kann.

IRE-DAS und DYSYS sind eine gute Grundlage für einen weiteren Ausbau eines größeren Simulator-Systems. Sie sind besonders dann gut geeignet, wenn hohe Anforderungen an die Genauigkeit gestellt werden. Durch die verschiedenartige Problemdarstellung ergänzen sich die Programme gut. Eine Aufgabe der Software-Entwicklung ist die Bereitstellung von Test-Subroutinen für IRE-DAS und DYSYS, um das Testen ähnlich wie beim Analogrechner zu erleichtern. Zusätzlich zu diesen beiden Simulatoren besteht noch Bedarf an einem einfacher aufgebauten Simulator, der ein weniger aufwendiges Integrationsverfahren besitzt und deshalb wesentlich schneller sein kann. Dabei kann eine geringe Genauigkeit in Kauf genommen werden.

Verbesserungen der Software müssen in größerem Umfang durchgeführt werden, wenn das Telefunken-Direktzugriffssystem zum Einsatz kommt. Dann ist es vor allem erforderlich, in den beiden Simulatoren IRE-

DAS und DYSYS die Möglichkeiten zur Verfolgung und Beeinflussung der laufenden Rechnung vorzusehen.

Für Hybridrechner steht schon jetzt eine umfangreiche Programmier- und Test-Software zur Verfügung [51,52,53]. Da diese Software stark maschinenabhängig ist, wurde sie von den Rechner-Herstellern entwickelt und wird beim Kauf einer Anlage mitgeliefert.

### 5.3 Mögliche Erhöhung der Rechengeschwindigkeit

Beim Vergleich von Analogrechner und Digitalsimulatoren sind die Unterschiede bei zwei Kriterien besonders hervorgetreten. Einmal ist es die fehlende Eingriffsmöglichkeit in die laufende Rechnung (bei IRE-DAS und DYSYS), dieser Nachteil kann jedoch beseitigt werden, wenn die Programme auf Maschinen mit Direktzugriffssystem eingesetzt werden. Der zweite wesentliche Unterschied tritt bei der Rechengeschwindigkeit auf. Die Rechenelemente moderner Analogrechner haben Bandbreiten von 300 kHz und mehr, sie sind also mindestens um den Faktor 10 schneller als der Analogrechner PACE 231. Auch die Digitalrechner sind in ungefähr dem gleichen Maß schneller geworden, der relative Unterschied bleibt also bestehen.

Auch in der weiteren zukünftigen Entwicklung ist damit zu rechnen, daß der Geschwindigkeitsunterschied zwischen beiden Rechnertypen sich nur wenig zu Gunsten des Digitalrechners ändert, wenn man die sehr teuren Multiprozessor-Großrechner, bei denen Operationen in beschränktem Rahmen parallel durchgeführt werden können, ausgeklammert.

Bei der Bewertung der Digitalrechner zum Lösen dynamischer Probleme ist nicht nur der Vergleich mit dem Analogrechner, sondern auch die Frage, ob mit einem Digitalrechner Echtzeitsimulation durchgeführt werden kann, von Bedeutung. Mit wenigen einfachen Gleichungen ist das heute ohne Schwierigkeiten möglich; die Gleichung  $y + \dot{y} = x$  kann z.B. mit DYSYS auf der IBM 7074 50mal schneller als Echtzeit gelöst werden. Bei größeren Gleichungssystemen mit zahlreichen nicht-linearen Zusammenhängen und großen Unterschieden in den Zeitkonstanten (große erforderliche Bandbreite) steigt die Rechenzeit schnell erheblich an.

Betrachten wir als typisches Beispiel ein System von 30 Dgln. 1. Ordnung, wobei 5 nichtlinear sind mit 5 Koeffizienten  $\neq 0$  und 25 linear mit im Mittel 3 Koeffizienten  $\neq 0$ . Die Zeitkonstanten sollen zwischen 0,01 und 5 sec liegen. Gleichungssysteme dieser Art treten z.B. bei der Simulation von Brennstäben auf.

Das Gleichungssystem kann auf einer Konsole eines mittleren Analogrechners dargestellt werden.

Mit einer kleinsten einstellbaren Zeitkonstanten von  $10^{-4}$  sec (bei der dynamische Fehler  $< 0,1\%$  gewährleistet sind) und der kleinsten Zeitkonstanten des Gleichungssystems von  $10^{-2}$  sec ist eine gegenüber Echtzeit 100mal schnellere Lösung des Gleichungssystems möglich. Ist das Gleichungssystem stabil und interessieren nur die Werte eines neuen stationären Zustandes infolge einer sprungförmigen Störung, so sind noch größere Rechengeschwindigkeiten möglich.

Die Rechenzeiten beim Digitalrechner können näherungsweise vorausberechnet werden, wenn das Integrationsverfahren vorgegeben ist. Am Beispiel von Verfahren, die sich bereits bewährt haben, wird unter Berücksichtigung der Rechenzeiten sehr schneller Rechenmaschinen die minimale Rechenzeit für das oben beschriebene Problem abgeschätzt. Ein Runge-Kutta-Verfahren 4. Ordnung das in DYSYS eingesetzt wird, und die Multistepmethode von Hamming [65] liegen den Berechnungen zugrunde. Beim Runge-Kutta-Verfahren 4. Ordnung gilt für die Rechenzeit eines Zeitschrittes folgende Näherung:

$$J = [4Z + m(6a + 8b)] \cdot A \cdot B \quad (1)$$

$J$  [sec] = Rechenzeit je Schritt

$Z$  [sec] = Zeit für die Berechnung aller Variablen zu einem Zeitpunkt

$A$  = Faktor, der den Zeitaufwand für Schrittweitemathematik und Genauigkeitsabfrage berücksichtigt

$B$  = Faktor, der den Zeitaufwand für die Organisation berücksichtigt

$m$  = Ordnung des Gleichungssystems

a  $\sqrt{\text{sec}}$  = Zeit für eine Multiplikation (Gleitkomma)

b  $\sqrt{\text{sec}}$  = Zeit für eine Addition (Gleitkomma)

Für das oben beschriebene Beispiel ist

$$Z = q \cdot r ( a + b ) + p^2 ( c + b ) \quad (2)$$

q = Anzahl der linearen Dgln. = 25

r = Anzahl der Koeffizienten der linearen Dgln. = 3

p = Anzahl der nichtlinearen Dgln. = 5

c  $\sqrt{\text{sec}}$  = Mittlere Rechenzeit für eine nichtlineare Operation  
(z.B. Wurzelziehen)

Für die IBM 7074 und 360/65 gelten folgende Rechenzeiten für die einzelnen Operationen:

Rechner	a $\sqrt{\text{sec}}$	b $\sqrt{\text{sec}}$	c $\sqrt{\text{sec}}$
7074	$45 \cdot 10^{-6}$	$20 \cdot 10^{-6}$	$700 \cdot 10^{-6}$
360/65	$4,4 \cdot 10^{-6}$	$2,5 \cdot 10^{-6}$	$60 \cdot 10^{-6}$

Die Faktoren für zusätzlichen Zeitaufwand kann man näherungsweise folgendermaßen ansetzen:

$$A = 2$$

$$B = 2$$

Für das oben beschriebene System von 30 Dgln. 1. Ordnung sind nach Gl. 1 und 2 folgende Rechenzeiten J je Rechenschritt zu erwarten:

Runge-Kutta 4. Ordnung	7074	360/65
J $\sqrt{\text{sec}}$	0,416	0,039
$\bar{t} = \frac{\text{Rechenzeit}}{\text{Problemzeit}}$	83,2	7,8

Aus Stabilitätsgründen ist die Schrittweite begrenzt. Der theoretische Maximalwert ist der doppelte Wert der kleinsten Zeitkonstanten  $T_{\min}$  (hier:  $T_{\min} = 0,01 \text{ sec}$ ). Aus bisherigen Rechnungen kann man für Integrationsverfahren mit Schrittweitenautomatik als Mittelwert für die Schrittweite  $0,5 T_{\min}$  erwarten. Für das gegebene Problem ist der Mittelwert  $x = 0,005 \text{ sec}$ . Für 1 sec Problemzeit sind also 200 Rechenschritte erforderlich. Mit der IBM 7074 werden dazu 83,2 sec benötigt, mit der IBM 360/65 7,8 sec.

Für das gegebene System von Dgln. müßte also ein Digitalrechner um den Faktor 10-20 schneller rechnen als die IBM 360/65, um eine Echtzeitsimulation zu erlauben.

Benutzt man anstelle eines Runge-Kutta-Verfahrens ein implizites Verfahren, z.B. die Multistepmethode nach Hamming [65], so kommt man zu ähnlichen Ergebnissen:

$$J = [2 Z + m (18a + 6b)] \cdot A \cdot B$$

Setzt man die gleichen Zahlenwerte wie beim Runge-Kutta-Verfahren für  $Z$ ,  $A$ ,  $B$ ,  $a$ ,  $b$  und die mittlere Schrittweite ein, so erhält man:

Multistep- methode	7074	360/65
$J \text{ [sec]}$	0,294	0,028
$\tau = \frac{\text{Rechenzeit}}{\text{Problemzeit}}$	58,8	5,6

Aufgrund der Unsicherheiten die in den Faktoren  $A$  und  $B$  und der Größe der mittleren Schrittweite liegen, können die errechneten Zahlen nicht genau sein. Die tatsächlichen Rechenzeiten mit einem gegebenen Simulator (z.B. DYSYS) können sich durchaus um den Faktor 2 - 3 von den hier errechneten unterscheiden.

Eine Herabsetzung der Rechenzeit ist von der Software her möglich durch eine Zeitoptimalisierung der Organisation (Faktor  $A$ ), was aber voraussetzt, daß der Simulator in Maschinencode programmiert werden

muß. Für bestimmte Probleme (wenn sich z.B. die Variablen nicht sehr stark ändern) kann auch eine Reduzierung der Ordnung des Integrationsverfahrens zu einer Verkürzung der Rechenzeit führen. Alle diese Maßnahmen reichen jedoch nicht aus, die Simulatoren so schnell zu machen, daß eine Echtzeitsimulation möglich ist. Diese ist nur über eine Verbesserung der Rechner selbst zu erreichen. Die überschlägig ermittelten Zahlenwerte lassen erkennen, daß eine Echtzeitsimulation für durchschnittliche Analogrechenprobleme (für 1 Konsole eines mittelgroßen Rechner) mit Digitalrechnern erst dann möglich werden, wenn die Rechner 10 - 50mal schneller rechnen als die IBM 360/65. Aufgrund der Fortschritte in der Technik integrierter Schaltkreise ist damit zu rechnen, daß in einigen Jahren Rechner hergestellt werden, die diesen Anforderungen genügen.

Diese erhöhten Geschwindigkeiten reichen aber immer noch nicht aus, wenn wesentlich umfangreichere Systeme in Echtzeit oder schneller gelöst werden sollen.

## 6. Zusammenfassung

Der Vergleich der vier dynamischen Simulatoren

hybrider Analogrechner	HYDAC 2000
CSMP	(für IBM 1130)
IRE-DAS	(für IBM 7074)
DYSYS	(für IBM 7074)

mit typischen Simulationsbeispielen aus den Aufgabengebieten der Reaktorentwicklung hat Vor- und Nachteile der einzelnen Simulatoren deutlich hervortreten lassen:

1. Der Digitalsimulator CSMP ist ungefähr 10mal langsamer als die beiden anderen Digitalsimulatoren. Das liegt zum Teil an der Organisation, zum Teil an der geringeren Rechengeschwindigkeit der IBM 1130. Er ist eine genaue Nachbildung eines mittleren Analogrechners einschließlich dessen Nachteile (z.B. Funktionsgeneratoren). Wegen der beschränkten Rechenkapazität und den sehr niedrigen Rechengeschwindigkeiten (im günstigsten Beispiel das 25-fache der Echtzeit) ist er für unsere Probleme unbrauchbar. Sein

einzigster Vorteil gegenüber den anderen Digitalsimulatoren ist die Möglichkeit des Eingriffs in die laufende Rechnung. Dabei ist jedoch wieder von Nachteil, daß die Maschine in den Pausen, die zwangsläufig beim Eingreifen in die Rechnung entstehen, nicht arbeitet. Ein digitaler Simulator mit der Möglichkeit des Eingreifens in die laufende Rechnung kann nur auf einer Maschine mit Direktzugriffssystem wirtschaftlich sinnvoll eingesetzt werden. In den zwangsläufig entstehenden Pausen können dann andere Aufträge abgewickelt werden.

2. Mit dem Analogrechner sind alle Probleme mindestens in Echtzeit, zum Teil sogar um den Faktor 100 schneller lösbar. Der Analogrechner ist bei den gewählten Beispielen im Durchschnitt ca. 1000mal schneller als die Digitalsimulatoren IRE-DAS und DYSYS. Bei der Darstellung von Funktionen mehrerer Veränderlicher sind jedoch grobe Vereinfachungen nötig. Der Einsatz eines Hybridrechners würde eine bessere Funktionsdarstellung ermöglichen. Es ist jedoch zu berücksichtigen, daß beim Hybridrechner die Rechengeschwindigkeiten gegenüber dem Analogrechner wesentlich geringer sein können, wenn die digitalen Rechnungen so umfangreich sind und so lange dauern, daß die maximal mögliche Rechengeschwindigkeit des Analogrechners nicht ausgenutzt werden kann. Es sind Fälle aus der Reaktordynamik bekannt, in denen die Lösung mit Hilfe eines Hybridrechners nur noch um den Faktor 3-10 schneller ist als eine rein digitale Lösung.
3. Die beiden digitalen Simulatoren IRE-DAS und DYSYS sind nahezu gleichwertig. Wenn nicht zu hohe Anforderungen gestellt werden (z.B. zu große Unterschiede in den Zeitkonstanten), sind die Rechenzeiten auch in erträglichen Grenzen. Berücksichtigt man, daß die beiden Simulatoren auf der IBM 360/65 ca. 6 - 10 mal schneller laufen als auf der IBM 7074, so sind sie "nur" noch um den Faktor 100 - 200 langsamer als der Analogrechner.

Der Vergleich zeigt, daß viele Probleme der Reaktorentwicklung besonders gut für eine Lösung mit dem Analogrechner geeignet sind. Ein Ersetzen des Analogrechners durch eine moderne Hybridrechenanlage würde wesentlich genauere Ergebnisse liefern können. Gegenüber



Digitalsimulatoren wie DYSYS und IRE-DAS oder einer speziellen digitalen Lösung eines bestimmten Problems wäre ein moderner Hybridrechner je nach Problem um den Faktor 5 bis 100 schneller.

Die Ergebnisse haben auch gezeigt, daß digitale Simulatoren sich neben Analogrechnern behaupten können. Es erscheint deshalb zweckmäßig, sowohl die digitale als auch analoge Simulation weiterzubetreiben und weiter auszubauen.

Im Rahmen der Entwicklung von Digitalsimulatoren für die Reaktorentwicklung stehen folgende konkreten Aufgaben für die nähere Zukunft an:

1. Zusammenfassung aller vorhandenen dynamischen Digitalprogramme zu einem Programmsystem (dynamic package).
2. Entwicklung von 2 weiteren Digitalsimulatoren, die IRE-DAS und DYSYS ergänzen:
  - a) ein Simulator, der sehr einfach zu handhaben ist und mit einem schnellen Integrationsverfahren arbeitet. Er soll für Überschlagsrechnungen und bei neuen Problemen für erste Lösungsansätze geeignet sein.
  - b) ein Simulator, der für umfangreiche lineare Differentialgleichungssysteme geeignet ist und die Laplace-Transformation benutzt (Übertragungsfunktion).
3. Erstellung von Unterprogrammen zur Totzeitsimulation. Die Unterprogramme müssen von IRE-DAS oder DYSYS aufgerufen werden können.
4. Einbau einer Möglichkeit zum Eingriff in die laufende Rechnung bei IRE-DAS und DYSYS. Diese Möglichkeit ist jedoch erst dann von Interesse, wenn Rechner mit Direktzugriffssystem zur Verfügung stehen.
5. Anpassung der Simulatoren an Betriebssysteme mit modernen Peripheriegeräten (z.B. Bildschirmgeräte für Ausgabe der Lösung in Kurvenform).

Bei der Weiterentwicklung des Hybridrechners als dynamischer Simulator stehen folgende Punkte im Vordergrund:

1. Die Programmierung des Koppelwerkes zwischen Digitalrechner und Analogrechner ist naturgemäß wegen der unterschiedlichen Arbeitsweise der beiden Rechner kompliziert. Es sollten jedoch von der Hardware her noch weitere Anstrengungen unternommen werden, eine einfachere Programmierung zu ermöglichen (übersichtliche Anordnung, Zusammenfassung von Funktionsgruppen usw.)
2. Die Betriebssoftware, vor allem für das Koppelwerk muß weiter ausgebaut werden, so daß bei häufig wiederkehrenden Operationen im Bereich des Koppelwerkes auf fertige Unterprogramme oder Schaltbilder zurückgegriffen werden kann.
3. Hybridrechner werden meistens in einer analogrechnerorientierten Form betrieben. Wenn man wesentliche Vereinfachungen bei der Programmierung erreichen will oder wenn Hybridrechner mit sehr großen Digitalrechnern zusammenarbeiten sollen, muß der Hybridrechner ganz vom Digitalrechner aus betrieben werden können.

Wenn man davon ausgeht, daß eine Automatisierung der Steckbrettprogrammierung mit einem vernünftigen technischen Aufwand nicht möglich ist, dann ist zunächst das folgende Ziel anzustreben: Eingabe und Steuerung des Programmablaufs eines Hybridrechners erfolgt vom Digitalrechner, die analogen Rechenschaltungen werden wie Unterprogramme aufgerufen, die vorher bereitgestellten Steckbretter werden nach Anforderung vom Operator eingesetzt, so wie es z.B. heute mit Datenbändern geschieht.

Häufig benutzte Analogrechenschaltungen sollen in Form einer Unterprogramm-Bibliothek immer zur Verfügung stehen und ähnlich wie digitale Unterprogramme aufrufbar sein, so daß für den Programmierer kein großer Unterschied zwischen einem digitalen und einem analogen Unterprogramm besteht.

Es ist zu erwarten, daß die Einführung des Großrechners TR 440 gute Voraussetzungen für den Einsatz und die Weiterentwicklung der Hybridrechentechnik bietet.

Die Programmierung der Simulationsbeispiele wurde zum größten Teil von den Herren D. Janßen, F. Katz und G. Kurze durchgeführt.

Literatur

- [1] Braffort, P.:  
The Dynamic Problems of Power Reactors and Analog  
Computers  
Proc. of the 1st Int.Conf. of the Peaceful Uses of  
Nuclear Energy, Genf, 1956
- [2] Nesterenko:  
Simulation Methods of Transient Thermal Processes in  
Gas-cooled Reactors on Analog Computers  
3.Int. Conf. Peaceful Uses of Nuclear Energy, Genf, 1964
- [3] Bryant L.T., et al.:  
A Hybrid Computer Program for Transient Temperature  
Calculations on Treat Fast Reactor Safety Experiments  
ANL 7374, Sept. 1967
- [4] Caillet C., Deat M.:  
Possibilities and Limitations of Analogue Calculations  
for Dynamic Study of Nuclear Power Stations  
Proc. 2nd UN Int.Conf. Peacef. Uses of Atomic Energy,  
Genf, 1958, Vol 11
- [5] Sanathanan C.K., et al.:  
The Application of a Hybrid Computer to the Analysis of  
Transient Phenomena in a Fast Reactor Core  
Nucl. Sc. a. Eng. 28, 1967
- [6] Bryant L.T.:  
Transient Temperature Calculations Using a Hybrid Computer  
AED-Conf. 1967, 069-039
- [7] Sanathanan C.K., et al.:  
Advances in the Hybrid Simulation of Nuclear Reactor  
Transients  
International Simposium on analog and hybrid Computation  
applied to Nuclear Energy, Versailles 1968

- [8] Frisch W., Woite G.:  
Analogrechenmodell für dampfgekühlte schnelle Reaktoren  
mit Direktkreislauf  
KFK 657, Dez. 1967
- [9] Frisch W., Schönfeld E.:  
Rechenprogramme für Dynamik und Stabilität eines  
schnellen Leistungsreaktors  
KFK 465, Juni 1966
- [10] Calderola L., et al.:  
The Balanced Oscillator Tests in SEFOR  
KFK 650, August 67
- [11] Audoux M., et al.:  
Automatic Control Systems for Balanced Oscillator Tests  
KFK 667, September 1967
- [12] Cimorelli L., et al.:  
Hybrid stochastic Monte Carlo Methodes for solving heat  
transfer problems in nuclear reactors  
Proceedings of the International Symposium on Analog and  
Hybrid Computation applied to Nuclear Energy, Versailles  
1968
- [13] Handler H.:  
Monte Carlo Solution of partial Differential Equations  
using a Hybrid Computer  
IEEE Transactions on electronic computers, Vol EC-16, Okt.1967
- [14] Kadlec J.:  
Rechnungsverfahren für die Bestimmung der Eigenfrequenzen  
der Normalfunktionen und der Dehnungsfunktionen quer-  
schwingender Brennstäbe  
Externer Bericht 8/66-9, Sept. 1966
- [15] Calderola L., Schlechtendahl E.G.:  
Reactor Temperature Transients with Spatial Variables,  
First Part: Radial Analysis  
KFK 223, Mai 1964

- [16] Kessler G.:  
Zur numerischen Lösung der ortsabhängigen dynamischen Gleichungen schneller Brutreaktoren mit Hilfe des Variationsprinzips  
KFK 781/I 1968
- [17] Fischer M.:  
Zur Dynamik der Wellenausbreitung in der Zweiphasenströmung unter Berücksichtigung von Verdichtungsstößen  
Externer Bericht, Kernforschungszentrum Karlsruhe,  
8/67-4, Aug. 1967
- [18] Hornyik K.:  
Models, Methods and Digital Computer Codes for the Analysis in Reactor Dynamics with Emphasis on Fast Breeder Reactors and Compressible Single Phase Coolants  
KFK 799, Dezember 1968
- [19] Bohn P.F.:  
Time sharing nonlinear analog equipment  
Simulation, Febr. 1968
- [20] Vichnevetsky R.:  
Analog/Hybrid Solution of Partial Differential Equations in the Nuclear Industry  
International Symposium on Analog and Hybrid Computation Applied to Nuclear Energy, Versailles Sept. 1968
- [21] Nomura T., Deiters R.:  
Improving the analog simulation of partial differential equations by hybrid computation  
Simulation August 1968
- [22] Di Bartolomeo M., et al.:  
Time Discretization Hybrid Model for Nuclear Reactor Cores Simulation  
International Symposium on Analog and Hybrid Computation Applied to Nuclear Energy, Versailles Sept. 1968

- [23] Neuenhahn P.:  
On the Hybrid Solution of Partial Differential Equations of  
the Hyperbolic Type  
International Symposium on Analog and Hybrid Computation  
Applied to Nuclear Energy, Versailles Sept. 1968
- [24] Vichnevetsky R.:  
A new stable computing method for the serial hybrid computer  
integration of partial differential equations  
AFIPS-Conference, Vol 32, 1968
- [25] Smidt D., et al.:  
Systemanalyse eines 1000 MW<sub>e</sub> natriumgekühlten schnellen  
Leistungsreaktors  
Kernforschungszentrum Karlsruhe  
Externer Bericht 8/66, Dez. 1966
- [26] Schönfeld E. und Merk W.:  
Unfalldynamische Untersuchungen an Kühlkreisläufen schneller  
Brutreaktoren  
AICA-Conf. 1967 Lausanne
- [27] Frisch W.:  
DIANA-Digitalprogramm zur Koeffizientenberechnung, Poten-  
tiometereinstellung und Prüfung von Analogprogrammen  
(noch nicht veröffentlicht!)
- [28] APACHE: Analog Programming and Checking:  
Programmers Manual  
EUR 2437 e
- [29] Petersen H.E., et al.:  
MIDAS - How it works and how it is worked  
Fall Joint Computer Conference 1964  
Spartan Books Inc., Baltimore, Md.
- [30] Brennan R.D., Sano H.:  
"PACTOLUS" - A digital analog simulator program for the  
IBM 1620  
Fall Joint Computer Conference 1964

- [31\_] IBM Application Program  
1130 Continuous System Modeling Program (1130-CX-13X)  
Program Reference Manual H20 - 0282 - 0
- [32\_] IBM Application Program  
1130 Continuous System Modeling Program (1130-CX-13X)  
System Manual H20 - 0284 - 1
- [33\_] Hurley J.R. and Skiles J.J.:  
DYSAC - A digitally simulated analog computer  
Spring Joint Computer Conf. May 1963
- [34\_] Trauboth H.:  
Programmsystem zur Simulierung allgemeiner Regelsysteme  
auf dem Digitalrechner  
Regelungstechnik Heft 1, 1966
- [35\_] Syn W.M., Linebarger R.N.:  
DSL/90 - A digital simulation program for continuous  
system modeling  
AFIPS Conference Proceedings Vol 28, 1966 SJCC
- [36\_] The SCi Continuous System Simulation Language (CSSL)  
Simulation, Vol 9, Nr. 6, Dez. 1967
- [37\_] Harnett R.T.:  
MIDAS - An Analog Approach to Digital Computation  
Simulation, May 1964

- [38] Korn G.A., Korn T.M.:  
Electronic Analog and Hybrid Computers  
Mc Grwa<sup>o/w</sup> Hill, New York 1964
- [39] Marston G.P., Mac Donald J.S.:  
A medium-scale hybrid interface  
Simulation, May 1968
- [40] Korn G.A.:  
Computer Interface Design for Simulation and Instrumentation  
in Nuclear Engineering  
Proceedings of the International Symposium on Analog and  
Hybrid Computation applied to Nuclear Energy, Versailles  
1968
- [41] Truitt T.D.:  
Hybrid computation - What is it? -  
Who needs it?  
Proceedings of the Spring Joint Computer Conference 1964
- [42] Vichnevetsky R.:  
Applying Hybrid Computers  
Southern Engineering, April 1968
- [43] Bub W.:  
Hybrid-Rechenanlagen  
Elektronische Datenverarbeitung, Heft 1, 1966
- [44] Bub W.:  
Das hybride Rechensystem EAI 8900  
Elektronische Datenverarbeitung, Heft 1, 1966
- [45] Hortenbach K.:  
Arbeitsweise und Einsatzmöglichkeiten hybrider  
Analogrechner  
Elektronische Datenverarbeitung, Heft 3, 1968



- [46] Giloi W.:  
Entwicklungstendenzen im hybriden Rechnen  
Elektron. Rechenanlagen 10, 1968 H.1
- [47] Sutton A.J.:  
Second generation language orientated hybrid computers  
5.Intern. Congress of AICA 1967, Lausanne
- [48] Becker D., et al.:  
Die Programmierung hybrider Rechenanlagen auf der  
Grundlage formeller Sprachen  
5.Intern. Congress of AICA 1967, Lausanne
- [49] AEG - Telefunken  
Das hybride Rechnersystem HRS 900  
AEG-Telefunken Informationstechnik FBS 048/0468
- [50] AEG - Telefunken  
Die Programmierung des hybriden Koppelwerks HKW 900  
AEG-Telefunken Rechnertechnik, Hybride Systeme ABS 0088 0968
- [51] Ocker W., Teger S.:  
HYTRAN - A software system to aid the analog programmer  
Proceedings of the Fall Joint Computer Conference 1964
- [52] AEG - Telefunken  
Real-Time-FORTRAN für das hybride Rechnersystem HRS 900  
in Vorbereitung
- [53] AEG - Telefunken  
HYTROL - Kontroll-und Testprogramm für das hybride  
Rechnersystem HRS 900  
AEG-Telefunken Informationstechnik, FBS 046/0468
- [54] Showalter R.E.:  
The solution of boundary-value problems in ordinary  
differential equations employing iterative differential  
analyzer equipment  
Simulation, Dezember 1965

- [55] Kucera J.:  
Zur Anwendung von Analogrechnern für Optimierungsaufgaben  
in der Industrie  
Messen, Steuern, Regeln Nr. 8, H.10, 1965
- [56] Gonzalez R. and Muterspaugh M.:  
Optimisation by parameter-perturbation correlation  
Simulation, August 1968
- [57] Wiesenthal P.:  
Die Lösung von Optimierungsaufgaben mit dem Gleichspannungs-  
analogrechner  
Elektronische Datenverarbeitung, Heft , 1966
- [58] Fogarty L.E., Howe R.M.:  
Trajectory optimization by a direct descent process  
Simulation, September 1968
- [59] Blake R.G., et al.:  
Hybrid Computer Optimization of a Nuclear Power Station  
Control System  
Proceedings of the International Symposium on Analog and  
Hybrid Computation applied to Nuclear Energy, Versailles  
1968
- [60] Kahne S.J.:  
Computational questions in optimal control, quasilinearization,  
and hybrid computers  
Simulation, August 1968
- [61] Paul R. and Legge C.G.:  
The solution of an optimal control problem by hybrid  
methods  
5.Intern. Congress of AICA 1967, Lausanne

- [62] Miura T. and Tsuda J.:  
Synthesis of Optimal Control Problems Using Hybrid  
Computers  
5. Internat. Congress of AICA 1967, Lausanne
- [63] Fifer S.:  
Analogue Computation  
Mc Graw-Hill, New York 1961
- [64] Starr D. and Jonsson J.:  
The design of an automatic patching system  
Simulation, Juni 1968
- [65] Ralstone W.:  
A first course in numerical analysis  
Mc Graw-Hill, New York 1965