Strukturanalyse von Planungsmodellen

Prof. Dr. Eckart Zwicker

Technische Universität Berlin

Fachgebiet Unternehmensrechnung und Controlling

Andreas Pleger

In: Kurbel, K., Mertens, P., Scheer, A.-G. (Hrsg.),

Interaktive betriebswirtschaftliche Informations- und Steuerungssysteme, Berlin, New York 1989

Strukturanalyse von Planungsmodellen

Eckart Zwicker, Andreas Pleger

Inhalt

1 Vorbemerkung

Der folgende Beitrag resultiert aus einem von der Deutschen Forschungsgemeinschaft im Rahmen des Forschungsschwerpunktes 'Interaktive betriebswirtschaftliche Informations- und Steuerungssysteme' geförderten Projektes. Ziel dieses Projektes war die Strukturanalyse von Planungsmodellen. Die hier beschriebenen Ergebnisse decken sich weitgehend mit dem Projektziel: es sollte die Möglichkeit eröffnet werden, im Rahmen von Planungsmodellen auf interaktive Weise die sogenannte reduzierte Gleichung einer Modellvariablen zu ermitteln. Dieses Verfahren ist anhand von Beispielen im folgenden beschrieben. Der zweite Teil des Forschungsprojektes zielt darauf ab, sogenannte interaktive Kausalkettenanalysen auf der Grundlage bestimmter Planungsmodelle vorzunehmen. Der vorliegende Beitrag beschreibt nur einen Teil dieses Projektes: es werden sukzessive Kausalkettenanalysen behandelt. Dies ist ein Verfahren, bei welchem der Benutzer im Dialog festzulegen hat, welche Folge von Variablenbeziehungen eines Modelles untersucht werden sollen. Die im Rahmen des Projektes zu realisierende Kausalkettenanalyse sämtlicher Modellbeziehungen in einem umfassenden Kausaldiagramm wird im folgenden nicht beschrieben.

2 Aufbau von Planungssystemen zur computergestützten Unternehmensplanung

Die Budget- und Finanzplanung von Unternehmen wird heute vorwiegend mit Hilfe von computergestützten Planungssystemen betrieben.

Der Aufbau solcher Systeme ist in Abbildung 1 beschrieben.

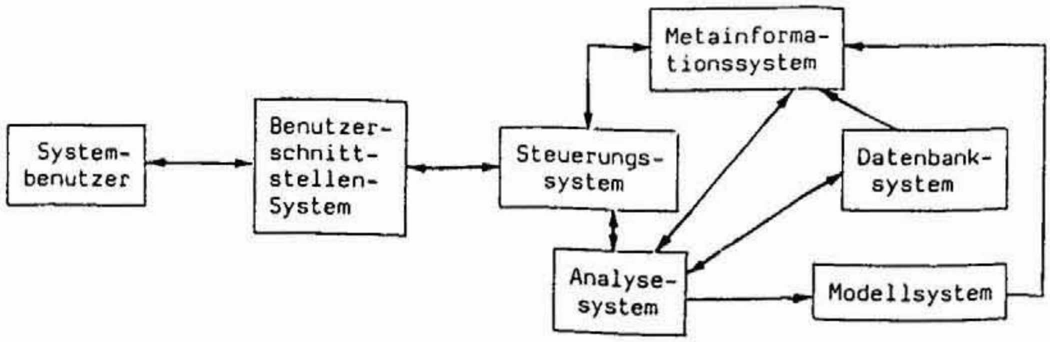

Abb. 1: Aufbau computergestützter Planungssysteme

Das Benutzer-Schnittstellensystem teilt dem Steuerungssystem die Systemeingaben mit und erhält von diesem die Systemausgaben. Das Steuerungssystem ruft das Analysesystem auf, welches mit der Datenbank und dem Modellsystem kommuniziert. Das Analysesystem dient dazu, die Implikationen des im Modellsystem niedergelegten Planungsmodelles zu ermitteln. Auch kann mit Hilfe des Analysesystems das Modell unter normativen Fragestellungen analysiert werden.

Das Metainformationssystem sammelt und liefert Informationen über den Prozeßablauf.

Das Modellsystem enthält das eigentliche Planungsmodell. Es besteht aus einem System von Differenzengleichungen und algebraischen Gleichungen.

Dieses Modellsystem dient, nachdem es entwickelt wurde, zwei Aufgaben:

- der Berechnung bestimmter Modellalternativen,
- der Modellstrukturanalyse.

Die Berechnung bestimmter Modellalternativen ist die Hauptaufgabe jedes Planungssystems. Sie wird von dem Analysesystem durchgeführt und besteht in der Bestimmung der Werte der endogenen Modellvariablen unter Vorgabe bestimmter numerischer Werte der Basisgrößen. Solche Modellberechnungen finden während der einzelnen Phasen eines Planungsprozesses statt. Sie bilden die Grundlage für Bottom-Up- und Top-Down-Rechnungen oder auch für die Berechnungen von Variatoren, sowie für Sensitivitäts- und Break-Evenanalysen.

Die Modellstrukturanalyse ist ebenfalls eine Aufgabe des Analysesystems. Sie dient dazu, dem Benutzer Einsichten über den strukturellen Aufbau des Modelles zu vermitteln. Die Modellstrukturanalyse bleibt aber, wie sich später zeigen wird, nicht nur auf der Ebene einer reinen Aufweisung struktureller Zusammenhänge stehen. Es werden unter Umständen auch bestimmte Rechnungen durchgeführt.

Solche Strukturanalysen fehlen weitgehend bei den heute in der Praxis gebräuchlichen kommerziellen Planungssystemen. Im folgenden sollen zwei Verfahren der Modellstrukturanalyse anhand von jeweils einem Beispiel beschrieben werden. Es handelt sich um die Ermittlung von sogenannten reduzierten Gleichungen und die Durchführung von Kausalkettenanalysen.

Diese beiden Formen einer Modellstrukturanalyse sind im Rahmen eines Planungssystems mit dem Namen INZPLA (Inkrementale Zielplanung) eingebettet [1].

Die in diesem System praktizierte Planungsprozedur soll im folgenden kurz beschrieben werden, um zu erkennen, wie die später behandelte Modellstrukturanalyse in dieses Planungssystem integriert wird.

Das INZPLA-System geht von einem Planungskonzept aus, welches als eine Konkretisierung des sogenannten Management durch Zielvorgabe angesehen werden kann. Es wird zwischen bestimmten Top- und Basiszielen unterschieden. Die Topziele sind die Ziele, die von der Unternehmensleitung angestrebt werden, wie beispielsweise die Eigenkapitalrentabilität oder die Umsatzrentabilität. Die Basisziele sind die Ziele, für welche bestimmte Bereiche verantwortlich gemacht werden können. Ein inkrementales Zielplanungsmodell ist ein Modell, in welchem diese Basisziele mit den Topzielen verbunden werden.

Abbildung 2 zeigt die schematische Darstellung eines Beispielmodelles, auf das im folgenden zurückgegriffen wird.

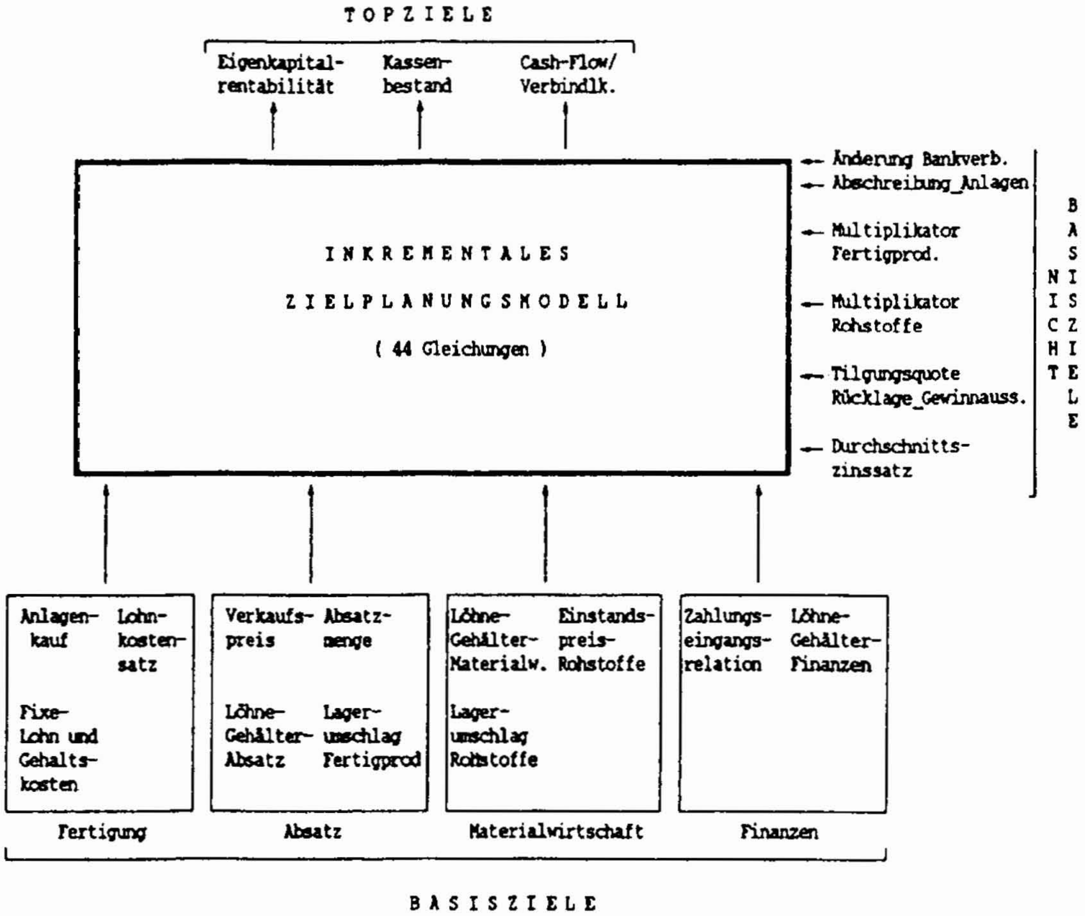

Abb. 2: Beispiel eines inkrementalen Zielplanungsmodelles

Es existieren vier Verantwortungsbereiche, deren insgesamt 12 Basisziele im einzelnen angeführt sind. Darüber hinaus enthält das Modell 5 Basisgrößen, die keine Basisziele sind. Ihre Werte sind als Schätzwerte vorzugeben. Der Planung liegen 3 Topziele zu Grunde.

Die Planung vollzieht sich in drei Stufen: der Bottom-Up-Planung, der Top-Down-Planung und der Bottom-Up-Top-Down-Konfrontation.

Während der Bottom-Up-Planung werden die *freiwilligen* Basiszielverpflichtungen der Verantwortungsbereiche zu den Topzielwerten hochgerechnet. Die Top-Down-Planung gliedert sich in zwei Schritte: Im ersten Schritt versucht die Unternehmensleitung, auf der Basis der Bottom-Up-Rechnung bestimmte Forderungen bezüglich der wünschenswerten Topziele zu formulieren. Auf dieser Grundlage bemüht sich die Unternehmensleitung (vertreten durch den Controller), Basiszielkombinationen zu finden, die die gewünschten Topziele realisieren.

Während der Bottom-Up-Top-Down-Konfrontation wird zwischen der Unternehmensleitung und der Bereichsleitung über die letztlich zu realisierenden Basisziele verhandelt.

Bottom-Up-Top-Down-Konfrontation 1985
Teilschritt:
Verantw.-bereich: 3 ABSATZ
Verantwortlich: SCHULZE

			11.70	200	17.32
B a s i s z i e l	Einh.	BZ-Wert	EIGEN-KAPITAL-RENTAB.	KASSEN-BESTAND	CASHFLOW/VERBIND-LICHKEITEN
LOEHNE_GEHAELTER_ABSATZ	DM	396	S -0.6510	S -1.9800	S -0.2190
ABSATZMENGE	Stück	1000	B 7.4243	B 12.7338	B 2.4840
DURCHSCHN_VERKAUFSPREIS	DM	11.50	B 18.9048	B 53.9692	B 6.3595
LAGERUMSCHLAG_FERTIGPRODUKTE	1/Jahr	12.35	S -0.2985	B 3.1405	S -0.0994

Abb. 3: Beispiel eines Konfrontationstableaus

Abbildung 3 zeigt ein sogenanntes Konfrontationstableau des erwähnten Modelles für den Verantwortungsbereich Absatz. Es wird während der Bottom-Up-Top-Down-Konfrontation verwendet. Die Spalte 'BZ-Wert' enthält die gerade zur Diskussion stehenden Werte der Basisziele des Absatzbereiches.

Die Spalten des Konfrontationstableaus korrespondieren mit den Topzielen, die durch die Eigenkapitalrentabilität, den Kassenbestand und das Verhältnis Cash-Flow zu Verbindlichkeiten repräsentiert werden.

Die numerischen Werte in den Matrizenfeldern repräsentieren sogenannte Variatoren. Sie zeigen, um wieviel Prozent sich der Wert des betreffenden Topzieles ändert, wenn der Wert des Basiszieles um ein Prozent erhöht wird. Im Falle der Absatzmenge beispielsweise beträgt der Variatorwert bezüglich der Eigenkapitalrentabilität 7,4 Prozent. Diese Situation kann, wie später gezeigt wird, den Ausgangspunkt für eine Strukturanalyse bilden.

3 Modellstrukturanalyse von Planungsmodellen

Im folgenden werden die bereits erwähnten Verfahren der Strukturanalyse, d.h. die Ermittlung von reduzierten Gleichungen und die Durchführung von Kausalkettenanalysen erläutert und anhand des beschriebenen Modelles illustriert.

3.1 Ermittlung von reduzierten Gleichungen

Wir beschäftigen uns, wie erwähnt, nur mit Planungssystemen, die sich durch Differenzengleichungen und algebraische Gleichungen beschreiben lassen. Diese Gleichungssysteme beschreiben bestimmte endogene Variable, wie den Gewinn oder die Eigenkapitalrentabilität. Die Erklärungsgleichungen solcher Topzielvariablen enthalten praktisch nie Modellbasisgrößen als erklärende Variable. Vielmehr werden die interessierenden Variablen zumeist nur über verschiedene *Zwischenvariable* von den Basisgrößen eines Modelles beeinflußt.

Es lassen sich Modelle finden, in welchen solche Abhängigkeiten über mehr als 30 Stufen von Zwischenvariablen laufen.

Um einen Eindruck von der Verknüpfungsstärke solcher Modelle zu geben, ist in Abbildung 4 das Kausaldiagramm eines bestimmten Verantwortungsbereiches (der Montage) in einem Industrieunternehmen angeführt. Die Verknüpfung der Variablen in diesem Subsystem erfolgt über maximal 7 Stufen.

Ein Verfahren der Komplexitätsreduzierung für solche mehrstufigen Planungsmodelle besteht in der Ermittlung der reduzierten Gleichung einer Variablen. In diesem Fall versucht man, durch entsprechende algebraische Umformungen die interessierende Variable als Funktion der sie beeinflussenden Basisgrößen zu formulieren.

Das Verfahren sei an einem einfachen Beispiel beschrieben.

Wir gehen von dem folgenden Gleichungssystem aus

```
G = U - K                                   (1)
G - Gewinn
U - Umsatz
K - Kosten

U = P * M                                   (2)
P - Preis
M - Menge

K = FK + VSTK * M                           (3)
FK   - Fixe Kosten
VSTK - Variable Stückkosten
```

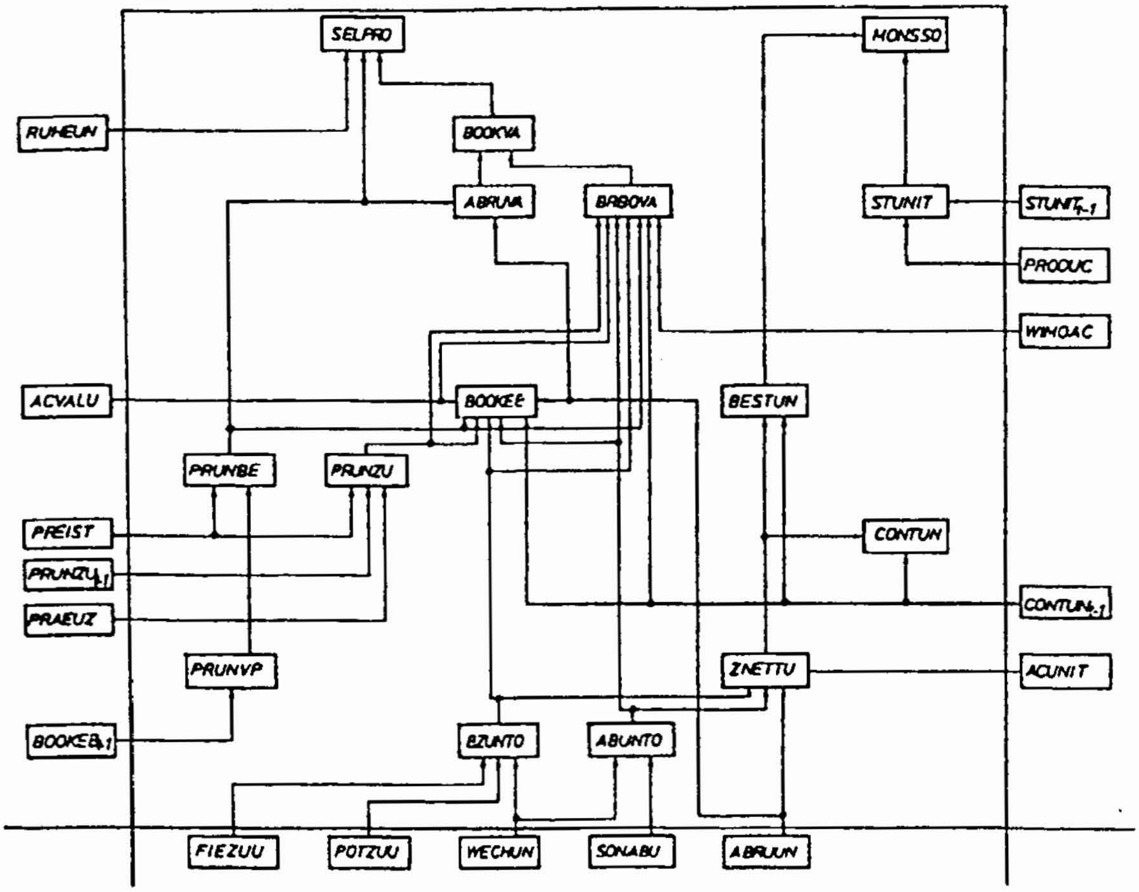

Abb. 4: Kausaldiagramm der Abhängigkeiten zwischen den Variablen eines Planungsmodelles im Verantwortungsbereich Montage (Variablen im Rechteck)

Die sogenannte vollsymbolische reduzierte Gleichung des Gewinnes G erhält man durch Einsetzen von (3) und (2) in (1), d.h.

$$G = P * M - FK - VSTK * M \tag{4}$$

Man kann verschiedene Formen einer reduzierten Gleichung unterscheiden. Eine *vollsymbolisch* reduzierte Gleichung beschreibt das obige Beispiel: sämtliche Basisgrößen sind symbolisch dargestellt. Bei einer nicht vollsymbolisch reduzierten Gleichung sind für bestimmte Basisgrößen numerische Werte eingesetzt. Wählt man beispielsweise im obigen Beispiel FK = 1000 und VSTK = 5, dann erhält man die nicht vollsymbolisch reduzierte Gleichung

$$G = P * M - 1000 - 5 * M \tag{5}$$

Beide Formen sind im Rahmen einer Strukturanalyse von Interesse. Der Extremfall einer teilsymbolischen Gleichung liegt vor, wenn alle Basisgrößen numerisch spezifiziert sind. Die Ermittlung der reduzierten Gleichung einer Variablen V würde dann zu dem Ergebnis

```
V = 'numerischer Wert'
```

führen.

Bevor der Reduktionsprozeß an einem Beispiel demonstriert wird, soll kurz auf die Voraussetzung eingegangen werden, unter der ein solcher Prozeß ablaufen kann. Sie besteht darin, daß es sich bei dem vorliegenden Gleichungssystem um ein rekursives System handelt. Dies bedeutet, daß es möglich sein muß, die Gleichungen in einer prozeduralen Form anzuordnen [2]. Dies ist bei sogenannten simultanen Modellen gerade nicht der Fall. Modelle, welche daher simultane Subsysteme enthalten, gestatten keine Ermittlung der reduzierten Gleichung einer Variablen, wenn sich eine der Zwischenvariablen in einem simultanen Gleichungssystem befindet. Liegen solche Umstände nicht vor, dann kann die reduzierte Gleichung der deklarierten Variablen durch einen Prozeß gewonnen werden, bei welchem die erklärenden Variablen der Erklärungsgleichung sukzessiv durch die sie erklärenden Ausdrücke ersetzt werden.

Die entstehenden komplexen algebraischen Ausdrücke sind dabei durch algebraische Umformung so einfach wie möglich zu gestalten [3].

Wenn die Erklärungsgleichungen der Zwischenvariablen auch nichtalgebraische Funktionen besitzen, was viele Planungssprachen zulassen, dann kann in diesem Fall keine vollständige Reduktion vorgenommen werden. Tritt bei der Reduktion beispielsweise eine Minimumsfunktion mit zwei Argumenten auf, dann erhält man als reduzierte Gleichung den Ausdruck

```
RV = MIN(A,B)
```

wobei A und B reduzierte algebraische Ausdrücke bilden. Entsprechendes gilt beispielsweise auch für Maximums- oder IF-THEN-ELSE-Funktionen.

Wir wenden uns nunmehr dem anfänglich beschriebenen Beispielmodell zu und gehen davon aus, daß wir uns im Planungsprozeß der Bottom-Up-Top-Down-Planung befinden. Ausgangspunkt ist das Konfrontationstableau der Abbildung 3. Der Benutzer ist an der Verknüpfung der Absatzmenge mit der Eigenkapitalrentabilität interessiert. Er erfährt anhand der Konfrontationstabelle, daß der Variator 7,42 Prozent beträgt. Er hat nunmehr die Möglichkeit, die tatsächliche funktionale Verknüpfung durch das System ermitteln zu lassen.

Vorab erhält der Benutzer, wie in Abbildung 5 dargestellt, eine Information über den Gliederungsbaum, welcher der Reduktion zu Grunde liegt.

```
MODELLSTRUKTURANALYSE - Spezifikation der Reduktionstiefe     Modell  :ZIEMOD4
                                        Planungsphase:BUJW    Planjahr:1985

Zu reduzierende Größe    EIGENKAPITALRENTABILITAET

Anzahl der Ebenen von Zwischengrößen , über
die die Basisgrößen in diese Größe eingehen :          7

Anzahl der Größen, die bei der angegebenen
Reduktionstiefe bearbeitet würden             :        34

Anzahl der  erklärenden Größen, als deren
Funktion die Größe dargestellt würde          :        16
```

Abb. 5: Übersicht zum Aufbau des Gliederungsbaumes der Reduktion

Der Reduktionsprozeß verläuft über 7 Stufen eines Gliederungsbaumes. Bei der Reduktion werden insgesamt 34 endogene Variable als erklärende Variable (Stufe für Stufe) ineinander eingesetzt. Die Zahl der Basisgrößen, welche die Eigenkapitalrentabilität beeinflussen, beträgt 16. Bis auf die Absatzmenge werden die übrigen 15 Basisgrößen numerisch konkretisiert. Die sich ergebende reduzierte Gleichung zeigt Abbildung 6.

Man erkennt, daß es sich um eine relativ komplizierte Gleichung handelt. Ein direkter Erkenntniszuwachs des Benutzers ist nur insoweit gegeben, als ihn die Kompliziertheit der Verknüpfung überraschen wird. Diese Komplexität ist darauf zurückzuführen, daß das vorliegende Modell ein sogenanntes absatzgetriebenes Modell ist, d.h. ein Modell in welchem sehr viele Modellgrößen in den verschiedensten Bereichen als eine Funktion der Absatzmenge geplant werden.

Als zweites Beispiel zur Berechnung einer reduzierten Gleichung soll wiederum die Eigenkapitalrentabilität als Referenzvariable gewählt werden. Als symbolisierte Basisgröße werden dagegen die Basisziele des Fertigungsbereiches gewählt. Aus Abbildung 2 ist zu erkennen, daß es sich um drei Basisziele handelt.

```
MODELLSTRUKTURANALYSE - Reduzierte Form der Gleichung          Modell   :ZIEMOD4
                                        Planungsphase:BUJW     Planjahr:1985

                        Reduzierte Modellgleichung

EIGENKAPITALRENTABILITAET  =
(-31177210
+0.114692*ABSATZMENGE^4
-111.7433*ABSATZMENGE^3
+10544.35*ABSATZMENGE^2
+202485.2*ABSATZMENGE
)
/(1.377363*ABSATZMENGE^3-217.9203*ABSATZMENGE^2+8619.598*ABSATZMENGE)
```

Abb. 6: Ermittlung der reduzierten Gleichung der Eigenkapitalrentabilität in einem Planungsmodell. Einzige symbolisierte Basisgröße ist die Absatzmenge (VAR^X = VAR^X)

```
MODELLSTRUKTURANALYSE - Reduzierte Form der Gleichung          Modell   :ZIEMOD4
                                        Planungsphase:BUJW     Planjahr:1985

                        Reduzierte Modellgleichung

EIGENKAPITALRENTABILITAET  =
31.72648-0.019273*FIXE_LOHN_UND_GEHALTSKOSTEN-19.22717*LOHNKOSTENSATZ
```

Abb. 7: Ermittlung der reduzierten Gleichung der Eigenkapitalrentabilität in einem Planungsmodell. Symbolisierte Basisgrößen sind die Basisziele des Verantwortungsbereiches *Fertigung*

Hier ist der Einfluß der Basisziele auf die Eigenkapitalrentabilität schon klar zu erkennen [4]. Die Erhöhung der fixen Lohn- und Gehaltskosten vermindert die Eigenkapitalrentabilität mit einem Faktor von 0,02, und der Lohnkostensatz der Fertigung wirkt sich mit einem beachtlich hohen Faktor von 19,23 aus.

3.2 Durchführung von Kausalkettenanalysen

Wie bereits erwähnt, werden bestimmte interessierende Variable nicht direkt von den Basisgrößen des Modelles beeinflußt. In ihre Erklärungsgleichungen gehen vielmehr bestimmte endogene Modellvariable ein, die wiederum von anderen Modellvariablen beeinflußt werden. Im Rahmen einer Kausalkettenanalyse ist es möglich, diese Verknüpfungen zwischen den Variablen zu verfolgen.

Ein Benutzer kann daran interessiert sein zu erfahren, in welche Erklärungsgleichung einer endogenen Variablen eine Referenzvariable als erklärende Variable eingeht. Ebenso kann er daran interessiert sein zu erfahren, welche Variablen als erklärende Variablen in der Erklärungsgleichung dieser Referenzvariablen auftreten.

Deklariert er beispielsweise das Topziel *Eigenkapitalrentabilität* als Referenzvariable, dann erhält er im Rahmen der Kausalkettenanalyse eine Ausgabe, die in Abbildung 8 wiedergegeben ist.

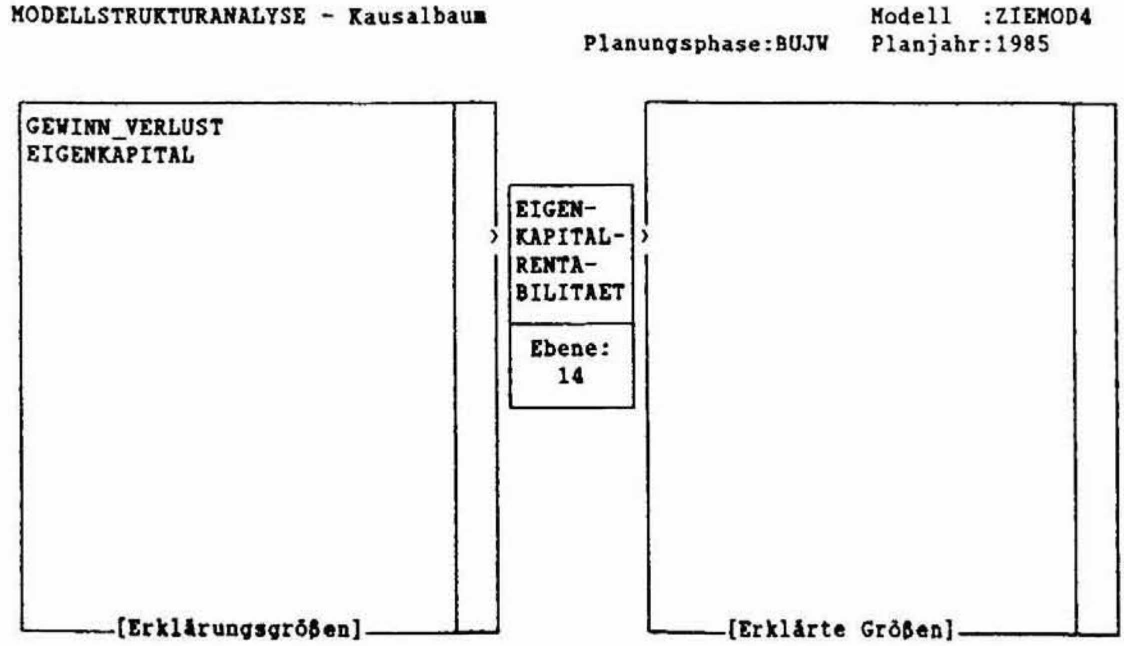

Abb. 8: Kausalkettenanalyse der Variable 'EIGENKAPITALRENTABILITÄT' in einem Planungsmodell

Die Eigenkapitalrentabilität fungiert, wie sich zeigt, in keinem Fall als erklärende Variable einer anderen Variable.

In ihre Erklärungsgleichung aber gehen der Gewinn und Verlust, sowie das Eigenkapital als Erklärungsgrößen ein.

Wir wollen annehmen, daß der Benutzer mit Hilfe eines Lichtbalkens den *Gewinn und Verlust* als nächste Referenzgröße deklariert.

In diesem Falle erhält man Abbildung 9.

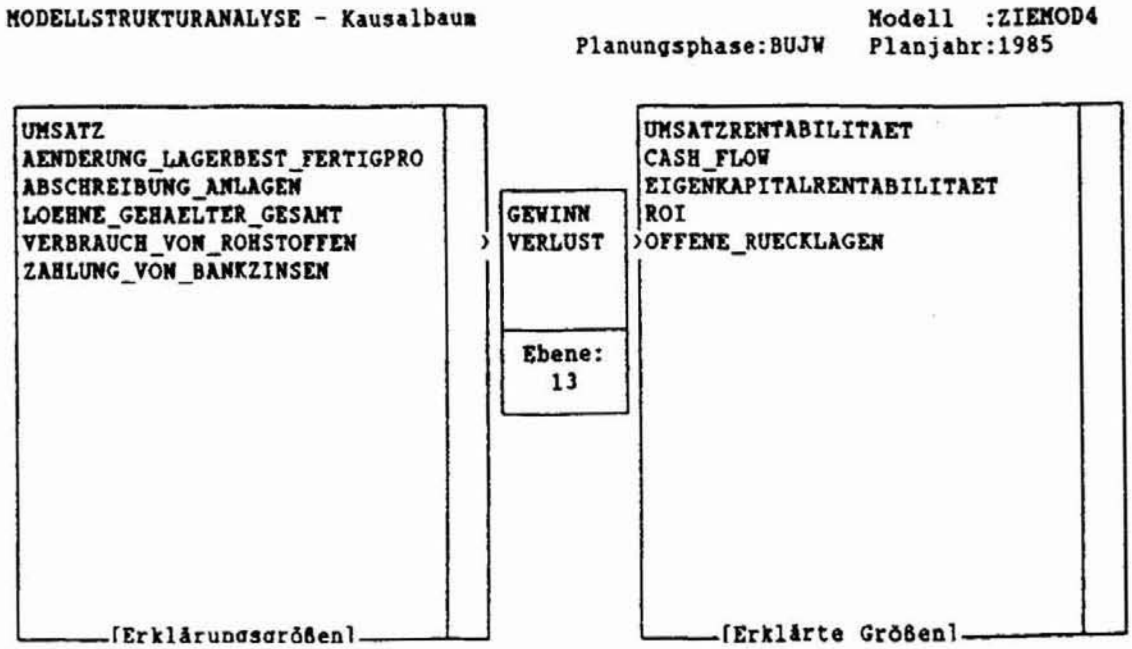

Abb. 9: Kausalkettenanalyse der Variable 'GEWINN VERLUST' in einem Planungsmodell

Die Kausalkettenanalyse einer Variablen wird durch eine 'Gleichungserklärung' ergänzt. Diese besteht darin, daß dem Benutzer die Erklärungsgleichung der betreffenden Variablen ausgegeben wird, sowie die numerischen Werte, die Variablenerklärung (Langnamen) und die Einheiten der auftretenden Variablen.

Abbildung 10 zeigt die 'Gleichungserklärung' für die Variable Gewinn und Verlust.

MODELLSTRUKTURANALYSE - Detailinformation Modell :ZIEMOD4
Planungsphase:BUJW Planjahr:1985

Name	Langname	Wert	Einheit
[Erklärte Größe]			
GEWINN_VERLUST	GEWINN_VERLUST	607.61	DM
[Erklärende Größen]			
UMSATZ	UMSATZ	11500.00	DM
AENDERUNG_LAGERBEST_FERTIGPRO	AENDERUNG_LAGERBEST_FERTIGPRO	-21.53	DM

Modellgleichung

```
GEWINN_VERLUST.J=UMSATZ.J+AENDERUNG_LAGERBEST_FERTIGPRO.J- &
ABSCHREIBUNG_ANLAGEN.J-LOEHNE_GEHAELTER_GESAMT.J- &
VERBRAUCH_VON_ROHSTOFFEN.J-ZAHLUNG_VON_BANKZINSEN.J
```

Abb. 10: Gleichungserklärung der Variable 'GEWINN_VERLUST' in einem Planungsmodell

In dem Feld *Erklärende Größen* stehen nur zwei Zeilen für die Kennzeichnung von zwei erklärenden Modellgrößen zur Verfügung. Im vorliegenden Beispiel gibt es aber sechs erklärende Modellgrößen. Diese können aber durch Rollen in das Feld *Erklärende Größen* eingespielt werden.

Das beschriebene Verfahren bietet dem Modellbenutzer die Möglichkeit, die Variablenbeziehungen komplexer Modelle zu *durchwandern*, um auf diese Weise Informationen über die Verknüpfungsweise und Einflußstärke der Variablen zu erhalten.

4 Realisierung der Modellstrukturanalyse

Das Gesamtsystem zur Modellstrukturanalyse gliedert sich in zwei Teilsysteme: die Gleichungsreduktion und die Kausalkettenanalyse.

Für beide Aufgaben benötigt das System detaillierte Informationen über die Größen des Modelles und deren algebraische Zusammenhänge. Diese Informationen werden aus den *internen Strukturen* des Modell-Precompilers gewonnen. Es wird also nicht unmittelbar auf die vom Benutzer - in der Planungssprache des INZPLA-Systems - formulierten Gleichungen zurückgegriffen, sondern der Precompiler des INZPLA-Systems führt zunächst eine Syntax-, sowie eine Kontextanalyse durch und überführt das Modell in eine Baumstruktur [5]. Diese Baumstruktur dient als Grundlage für die Routinen der Gleichungsreduktion und die Kausalkettenanalyse.

Für die Kausalkettenanalyse werden optional einige *System-Dateien* benötigt; sie liefern zusätzliche Informationen über die Modellgrößen z.B. deren Einheiten und Klassifikationen.

Wurde das Modell bereits durchgerechnet, so ist es in beiden Teilsystemen möglich, gezielt auf die Werte einer bestimmten Planungsperiode und -phase zuzugreifen.

4.1 Kausalkettenanalyse

Die Kausalkettenanalyse besteht aus zwei Teilschritten. Zunächst werden die Modellgrößen bestimmten *Ebenen zugeordnet*. Alsdann erhält der Benutzer die Möglichkeit, über den so gebildeten *Kausalbaum* zu *traversieren*.

Grundlage beider Teilschritte ist die Erstellung zweier Listen, die jeweils eine Richtung der Kausalzusammenhänge abbilden. Diese beiden Listen werden durch sequentielles Abarbeiten aller Modellgleichungen erzeugt. Sie liefern auch die Basis für die Traversion über die Kausalstruktur, wobei Zusatzinformationen aus Systemdateien eingespielt werden.

Bei der Berechnung der Ebenen wird für jede Modellgröße, die von keiner anderen Modellgröße abhängt, also für alle exogenen Modellgrößen, eine rekursive Routine aufgerufen, die die von ihnen abhängigen endogenen Modellgrößen bearbeitet. Dabei überprüft die Routine alle direkt abhängigen Modellgrößen, ob deren gerade untersuchte Beeinflussung von höherer Ebene ist, als bisher festgehalten; in diesem Fall wird die Modellgröße ebenso bearbeitet, woraus sich die Rekursion ergibt.

4.2 Gleichungsreduktion

4.2.1 Steuerung durch den Benutzer

Ist die interessierende Modellgröße (Topziel) ausgewählt worden, so hat der Benutzer festzulegen, welche der erklärenden Basisgrößen der reduzierten Gleichung symbolisch erhalten bleiben sollen und welche numerisch zu konkretisieren sind, also durch die Werte der zuvor festgelegten Planungsperiode und -phase zu ersetzen sind.

Aus diesen Benutzereingaben resultiert für jede in die Reduktion involvierte Modellgröße ein Status, der in einem Vektor abgelegt wird. Hier wird später auch festgehalten, ob eine Modellgröße als Zwischenergebnis bereits berechnet wurde, so daß keine erneute Berechnung erforderlich wird.

4.2.2 Zum Begriff eines vereinfachten reduzierten Terms

Ziel der Gleichungsreduktion ist die Ermittlung einer *vereinfachten* reduzierten Gleichung. Darunter versteht man eine reduzierte Gleichung, deren Erklärungsausdruck ein vereinfachter reduzierter Term ist. Der Begriff des vereinfachten reduzierten Terms soll im folgenden erläutert werden.

Ein *reduzierter* Term ist daran zu erkennen, daß er als Operanden lediglich numerische Werte enthält oder solche Modellgrößen, die Basisgrößen sind und für die der Benutzer festgelegt hat, sie symbolisch zu belassen.

Bezüglich der *Vereinfachung* wurde ein besonderes Augenmerk auf eine übersichtliche kanonische Form gelegt. Deshalb wurde als Grundstruktur die Form einer *ausmultiplizierten gebrochenrationalen Funktion* gewählt. Hier läßt sich auch bei großen Termen die Struktur vom Benutzer verhältnismäßig gut erkennen. Auch für eine analytische Optimierung bietet sie erhebliche Vorteile [6]. Im einzelnen kann ein Term folgende Zustände annehmen:

(1) Real-Zahl
Eine Real-Zahl darf positiv und negativ sein.

(2) Größe
Eine Größe gilt nur dann als vereinfachter reduzierter Term, wenn es sich um eine nicht numerisch zu konkretisierende Basisgröße handelt.

(3) Funktion
Jede Funktion besteht aus dem sie identifizierenden Namen und einer beliebig langen Parameterliste. Dabei ist jeder Parameter für sich wieder ein vereinfachter reduzierter Term. Für eine endliche Zahl vordefinierter Funktionen soll, wenn alle Parameter Real-Zahlen (1) sind, die Funktion durch ihren Wert als Real-Zahl dargestellt werden. Sind mehr als einer, aber nicht alle Parameter Real-Zahlen, so sind diese nach Möglichkeit zusammenzufassen.

Alle Ausdrücke und Konstrukte, die sonst nicht zulässig wären, werden als Funktion formuliert (z.B. IF-Statements oder Potenzen mit nicht-numerischen Exponenten).

(4) Potenz
Als Basis einer Potenz kommen Größen (2) und Funktionen (3) in Frage, der Exponent darf nur eine reelle Zahl größer 0 und ungleich 1 sein. Andere Potenzen sind auszumultiplizieren /-dividieren bzw. als Funktion darzustellen.

(5) Produkt
Jedes Produkt besteht aus mindestens zwei Faktoren, davon darf nur einer eine Real-Zahl (1) sein, deren Wert weder 0 noch 1 betragen darf. (Der Wert -1 ist zulässig.)

Die weiteren Faktoren können Größen (2), Funktionen (3) und Potenzen (4) sein. Dabei darf keine Größe oder Funktion mit einer Basis oder zwei Basen miteinander übereinstimmen; gegebenenfalls sind diese Faktoren miteinander zu verrechnen.

Alle Faktoren sind in der Reihenfolge Real-Zahl - Größe - Funktion zu sortieren, wobei die Gruppen untereinander alphabetisch zu sortieren sind. Die Potenzen werden ihrer Basis gemäß einsortiert.

Produkte, die neben einer Real-Zahl (1) nur einen weiteren Faktor enthalten, werden im folgenden als Elementarprodukte bezeichnet.

(6) Summe
Eine Summe besteht aus mindestens zwei Summanden, davon maximal eine Real-Zahl (1), deren Wert verschieden von 0 sein muß. Als weitere Summanden kommen Größen (2), Funktionen (3), Potenzen (4) und Produkte (5) in Betracht. Hierbei ist zu beachten, daß sich alle enthaltenen Produkte durch mehr unterscheiden als ihre Real-Zahl. Auch Größen, Funktionen und Potenzen müssen untereinander und zu dem entsprechenden Faktor eines Elementarprodukts verschieden sein.

Die Sortierung erfolgt in der gleichen Reihenfolge, wie bei den Produkten, jedoch müssen Potenzen mit gleicher Basis (dies gibt es in Produkten nicht) nach abfallenden Exponenten sortiert werden. Während die Elementarprodukte entsprechend ihrem nicht-numerischen Faktor geordnet werden, folgen alle übrigen Produkte als weitere Gruppe. Die Sortierung dieser Produkte erfolgt nach den gleichen Kriterien, wie für die einfachen Summanden, zunächst für den ersten Faktor, dann für den zweiten usw. Die Real-Faktoren bleiben in der Sortierfolge unberücksichtigt.

(7) Quotient
Ein Quotient besteht aus einem Zähler und einem Nenner. Der Zähler darf ein beliebiger vereinfachter reduzierter Term sein (1) - (6), jedoch kein Quotient. Sein Nenner kann eine Größe (2), eine Funktion (3), eine Potenz (4), ein Produkt (5) ohne Real-Zahl als Faktor oder eine Summe (6) sein, deren nicht-numerische Summanden, falls sie alle Produkte sind, nicht den gleichen numerischen Faktor enthalten dürfen. Weiterhin ist grundsätzlich Teilerfremdheit zu fordern, wobei sich dies zunächst auf Identitätsvergleiche zwischen einer Größe (2), einer Funktion (3), der Basis einer Potenz (4) und den Faktoren eines Produkts (5) im Zähler bzw. Nenner des Quotienten beschränken soll. Eine weitere Möglichkeit zu kürzen, besteht dann, wenn Zähler und Nenner als Summen nur aus Produkten bestehen, deren numerische Faktoren innerhalb der Summe gleich sind, wenn sich die Summen bis auf diese numerischen Faktoren gleichen. Existieren keine numerischen Faktoren, so kann die ganze Summe verglichen werden.

4.2.3 Erzeugung der vereinfachten reduzierten Gleichung

Das Grundprinzip der Erzeugung eines vereinfachten Termes ist nun, daß die Gleichungen abgearbeitet werden, als sollten sie numerisch berechnet werden, dies jedoch mit *symbolischen Rechenroutinen*. Neben diesen wird also noch ein Algorithmus benötigt, der die Abfolge der Rechenoperationen steuert. Hierzu dient der *Simple-Precedence-Algorithmus*. Diese Vorgehensweise bewirkt einen vereinfachten reduzierten Term unter der Voraussetzung, daß der Berechnung nur Real-Zahlen oder Größen, die nicht weiter zu ersetzen sind (also Terme der Form (1) und (2) zugeführt werden oder auch Zwischenergebnisse, die bereits als vereinfachte reduzierte Terme (1) bis (7) errechnet wurden, womit impliziert ist, daß die Rechenroutinen aus zwei vereinfachten Termen nur einen vereinfachten Term erzeugen dürfen.

4.2.4 Der Simple-Precedence-Algorithmus und die Reduktion

Zur Festlegung der Rangfolge der Operatoren und zur Steuerung des Abspeicherns von Zwischenergebnissen liegt dem Simple-Precedence-Algorithmus folgende *Vorrangsmatrix* zugrunde (vergleiche Abbildung 11).

Anhand eines Beispieles soll in Abbildung 12 die Wirkungsweise kurz aufgezeigt werden. Der besseren Darstellbarkeit wegen wurde ein rein numerisches Beispiel gewählt.

Auf der rechten Seite jedes Diagramms, ist die *Eingabe* in Infix-Notation dargestellt, d.h. die Operatoren befinden sich zwischen den Operanden. Die Eingabe wird nun von links nach rechts abgearbeitet, also bildlich gesehen nach links geschoben. Liegt ein Operand zur Bearbeitung vor, so wird dieser immer sofort auf den *Operanden-Stack* (oben im Diagramm) geschoben, weshalb dieser Schritt nicht jeweils gesondert dargestellt ist.

Zeigt die Vorrangsmatrix (siehe Abbildung 11), daß der Operator auf dem Stack eine kleineren Rang hat als der gerade anstehende, so wird letzterer auf den *Operatoren-Stack* geschoben.

Kann gerechnet werden, so werden die für die *Berechnung* benötigten Operanden vom Operanden-Stack geholt und das Ergebnis dorthin zurück geschrieben.

		aktueller Operator											
		gr(	fk(	(	±	*/÷	**	(-)	)	,	)fk	)gr	⊣
Operator v. Stack	⊢	<	<	<	<	<	<	<	E	E	E	E	⊣
	gr(	<	<	<	<	<	<	<	E	E	E	S	E
	fk(	<	<	<	<	<	<	<	E	<	>	E	E
	(	<	<	<	<	<	<	<	=	E	E	E	E
	±	<	<	<	>	<	<	<	>	>	>	>	>
	*/÷	<	<	<	>	>	<	<	>	>	>	>	>
	**	<	<	<	>	>	<	<	>	>	>	>	>
	(-)	<	<	<	>	>	>	=	>	>	>	>	>
	,	<	<	<	<	<	<	<	E	<	>	E	E

Legende:

- < Der aktuelle Operator hat höhere Priorität, deshalb muß zunächst weitergelesen werden, um ihn zuerst ausführen zu können.
- = Die Operatoren haben die gleiche Priorität und können ohne Aktion gelöscht werden.
- > Der aktuelle Operator hat niedrigere Priorität, d. h. der Operator vom Stack darf ausgeführt werden.
- S Eine Größe wurde vollständig bestimmt und kann als Zwischenergebnis abgespeichert werden
- ⊣ Der Ausdruck ist vollständig ausgewertet.
- E Fehler, unzulässige Reihenfolge.

Abb. 11: Vorrangsmatrix für Simple-Precedence-Algorithmus

Üblicherweise wird der Algorithmus derart formuliert, daß er selbst den Input in linearer Infix-Notation einliest und Operanden auf dem Operanden-Stack ablegt, während er bei Operatoren prüft, ob der vorhergehende zur Ausführung gebracht werden darf. Diese Vorgehensweise hätte bedeutet, daß zunächst eine vollständige Reduktion durchzuführen gewesen wäre, was verhindert hätte, die Zwischenergebnisse einer bereits berechneten Größe wieder zu verwenden. Umgekehrt sollte die zur Reduktion erforderliche Rekursion nicht in den Simple-Precedence-Algorithmus integriert werden, um das Verfahren nicht zu kompliziert werden zu lassen. Deshalb wurde der Algorithmus derart implementiert, daß eine Operatoren verarbeitende Routine entstand.

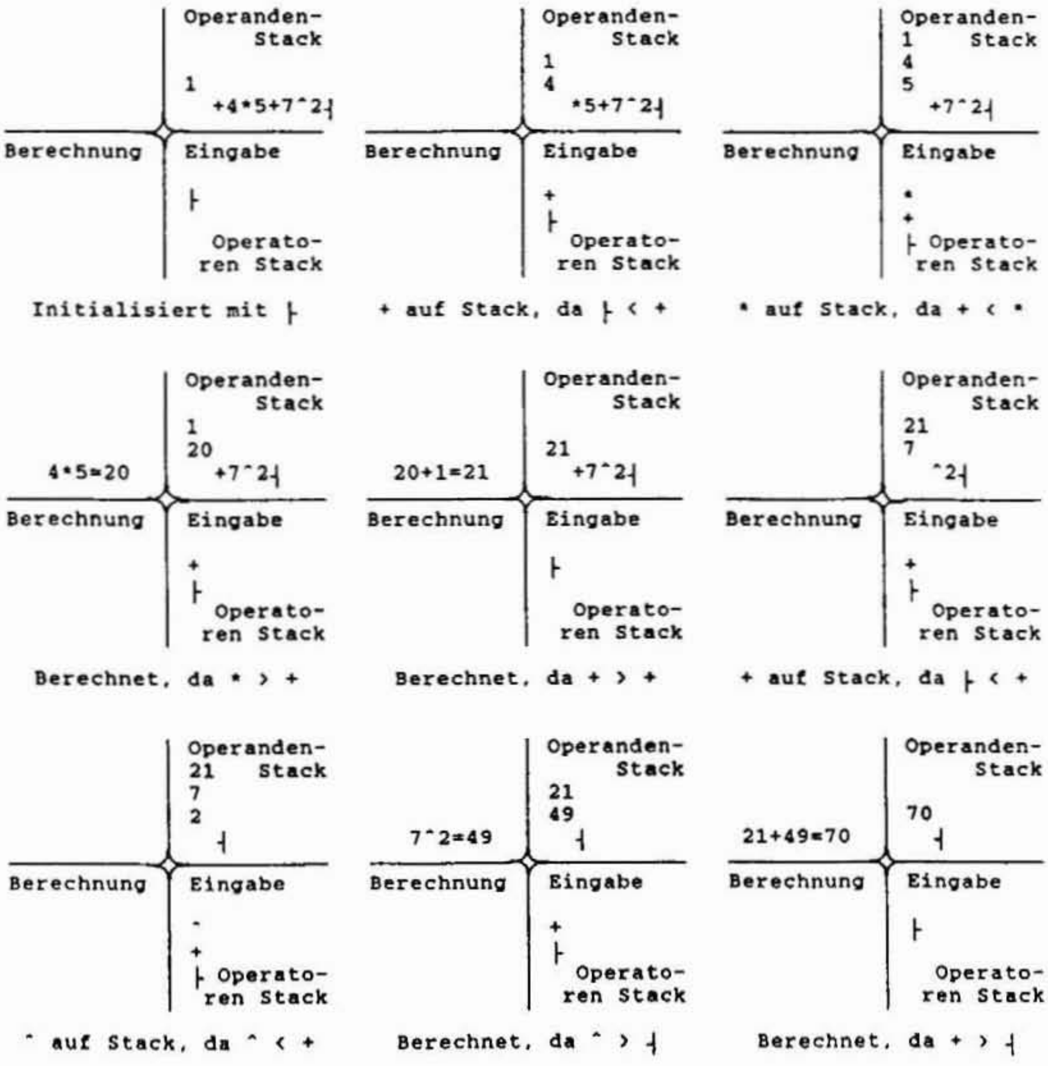

Abb. 12: Beispiel für die Anwendung des Simple-Precedence-Algorithmus

So beginnt das System die Modellgleichung (bzw. Compiler-Struktur) der zu betrachtenden Größe zu lesen und ruft jeweils Routinen zur Operator- und Operandenbehandlung auf. Bei der *Operatorenbehandlung* handelt es sich um den Simple-Precedence-Algorithmus. Dagegen wird bei der *Operandenbehandlung*, bevor der Operand auf den Operanden-Stack gelegt wird, geprüft, ob der Operand eine Zahl ist, eine numerisch zu konkretisierende, bzw. symbolisch zu belassende Basisgröße oder eine weiter zu reduzierende endogene Größe. Im letzten Fall wird die gleichungslesende Routine rekursiv aufgerufen. Wurde die Größe bereits berechnet, wird das ermittelte Ergebnis ohne neue Berechnung auf dem Operanden-Stack abgelegt.

Kommt der Simple-Precedence-Algorithmus an einen Zustand, in dem gerechnet werden kann, so wird eine Routine aufgerufen, die die Operanden vom Operanden-

Stack abruft und die Ausführung der entsprechenden Operation veranlaßt. Das Ergebnis wird dann wieder auf dem Operanden-Stack abgelegt, der Speicherplatz der Operanden wird freigegeben. Im folgenden soll noch auf die symbolischen Rechenroutinen eingegangen werden.

4.2.5 Die symbolischen Rechenroutinen

Aufgabe dieser Routinen ist es, aus jeweils zwei vereinfachten reduzierten Termen einen vereinfachten reduzierten Term zu bilden. Die dabei anzustellenden Überlegungen sollen *exemplarisch an der Addition* demonstriert werden, sind aber für alle Operationen erforderlich.

Zunächst ergibt sich bei rein syntaktischer Betrachtung eine Matrix, die alle Termtypen einander gegenüberstellt und in deren Feldern die Syntax des Ergebnisses dargestellt werden kann. Fast die Hälfte der Felder läßt sich durch Kommutativität herleiten (jedoch gilt dies nicht für alle Operationen). Die tatsächliche Abarbeitung ist aber komplexer, da der vereinfachte Term nicht nur durch seine Syntax definiert wird. So sind beispielsweise zwei gleiche Größen, die addiert werden sollen, zum Produkt zusammenzufassen (a+a=2*a). Teilweise kann in den einzelnen Fällen auch auf andere Operationen zurückgegriffen werden. Ein Beispiel für die über die Syntax hinaus anzustellenden Überlegungen soll anhand der Addition einer Größe (2) als Summand_1 mit einem Produkt (5) als Summand_2 gegeben werden.

Von den Routinen, welche die Rekursion durchführen, werden dem Simple-Precedence-Algorithmus und damit den Rechenroutinen als Operanden nur reelle Zahlen (1) und Größen (2) zugeleitet. Da die Rechenroutinen als Ergebnis zweier vereinfachter Terme wieder einen vereinfachten Term erzeugen, genügen auch die in sie eingehenden Zwischenergebnisse den Forderungen an einen solchen Term. Durch *vollständige Induktion* läßt sich daraus schließen, daß als *Ergebnis-Term* e in vereinfachter reduzierter Term entsteht.

```
a) Summand₂ ist kein Elementarprodukt
   Ergebnis: Summenterm mit
      reeller_Summand = 0
      1. Operand = Summand₁
      2. Operand = Summand₂
b) sonst
   ba) Summand₁ = Operand von Summand₂
       baa) reeller_Faktor_Summand₂ = -1
            Ergebnis: Realterm mit
                Wert = 0
          bab) sonst
            Ergebnis: Produktterm mit
               reeller_Faktor = reeller_Faktor_Summand₂ + 1
               Operand = Operand von Summand₂
   bb) sonst
        bba) Summand₁ > Summand₂
            Ergebnis: Summenterm mit
                    reeller_Summand = 0
                    1. Operand = Summand₂
                    2. Operand = Summand₁
          bbb) sonst
            Ergebnis: Summenterm mit
                    reeller_Summand = 0
                    1. Operand = Summand₁
                    2. Operand = Summand₂
```

Abb. 13: Symbolische Addition einer Größe (2) und eines Produktes (5) zu einem vereinfachten Term

5 Ausblick

Die Anwendung von Modellstrukturanalysen erweist sich als eine sinnvolle Erweiterung der Verfahren, die im Rahmen einer computergestützten Unternehmensplanung praktiziert werden. Die Ermittlung der reduzierten Gleichung kann darüber hinaus dazu dienen, die Leistungsfähigkeit einer Planungssprache zu erhöhen. So können erst auf der Grundlage einer reduzierten Gleichung, d.h. einer kanonischen Form, bestimmte Algorithmen der linearen und nichtlinearen Optimierung im Rahmen einer Planungssprache eingesetzt werden. Durch die Verwendung von reduzierten Gleichungen kann darüber hinaus in manchen Fällen zur Berechnung bestimmter endogener Variablen die erforderliche Rechenzeit bei großen Modellen beachtlich reduziert werden.

Anmerkungen

[1] Zwicker (1988), S. 341-354.

[2] Mathematisch gesehen bedeutet dies, daß die Strukturmatrix des Gleichungssystems durch eine endliche Zahl von Permutationen in eine blocktriangulare Form überführt werden kann.

[3] Siehe hierzu Abschnitt 4.3.2.

[4] Das Basisziel Anlagenkauf hat keinen Einfluß auf die Eigenkapitalrentabilität, da es sich bilanztechnisch nur um einen sogenannten Aktivtausch handelt.

[5] Diese Baumstruktur bildet jedoch nicht die von uns im Rahmen der Kausalkettenanalyse angestrebte Modellgleichungshierarchie ab, sondern behandelt die einzelnen Modellgleichungen auf einer Ebene, um sie bei der auf die Precompilation folgenden Code-Erzeugung in prozeduraler Reihenfolge als einzelne Statements abzusetzen.

[6] Siehe Abschnitt 5.

Literatur

Davenport, J.H., Siret, Y., Tournier, E.: Computer Algebra: Systems and Algorithm for algebraic Computation; London 1988.

Rosenkranz, F.: An Introduction to Corporate Modelling; Durham 1979.

Zwicker, E.: INZPLA - ein Konzept der computergestützten Unternehmensgesamtplanung; in: Lücke, W. (Hrsg.): Betriebswirtschaftliche Steuerungs- und Kontrollprobleme; Wiesbaden 1988.