

THE INTEGRATION  
OF ALTERNATIVE MODELLING  
WITH DYNAMO

Hermann Krallmann

Industrieseminar der Universitaet Mannheim

Often system dynamics, and particularly the DYNAMO-language, is attacked for not integrating other modelling approaches into the field. This investigation offers alternatives that will hopefully stand up against the critics.

The first part of this paper concerns the integration of external functions into system dynamics models. Modifications of the DYNAMO simulation language and of the DYNAMO compiler are explained, and conceptional questions about the integration are discussed. By means of examples of LP programs and statistical methods, the paper shows the philosophical improvements entailed by the system dynamics method.

The same criteria are applied in the second part of this paper to the model-method integration of a system dynamics model with an input-output method, considered to be representative of a complete economic structure.

The last part of the paper explains the integration of the system dynamics model into the higher level program structure of an optimizing feedback loop. The best combination of input vector parameters is calculated in the feedback loop at any time so that the output vector follows a predetermined objective function. The overall paper contents demonstrate the flexibility of the system dynamics method.

TABLE OF CONTENTS

	Page
I. EXTERNAL FUNCTIONS AS ELEMENTS OF THE SYSTEM DYNAMICS LANGUAGE: AS DEMONSTRATED IN AN LP-PROGRAM	655
A. Extension of DYNAMO II <sub>F</sub> in the Program Technique Context	655
B. Extension of DYNAMO II <sub>F</sub> in the Language Context	658
C. An Application: LP-Model	659
D. Conceptual Aspects of the Integration of System Dynamics and Statistical or OR-Methods	661
II. HANDLING INPUT-OUTPUT ANALYSIS IN SYSTEM DYNAMICS MODELS	665
A. Modification of the DYNAMO Language	665
B. Conceptual Aspects of the Model - Method Integration	667
III. SYSTEM DYNAMICS MODEL IN A SUPERIOR PROGRAM STRUCTURE	669
A. System Dynamics Applied to a "Polycondensation"-Plant	670
B. The Optimization of a Polycondensation-Plant by Means of the Razor-Search Algorithm	675
IV. RESULTS OF THE OPTIMIZATION PROCEDURE	679
BIBLIOGRAPHY	683

I. EXTERNAL FUNCTIONS AS ELEMENTS  
OF THE SYSTEM DYNAMICS - LANGUAGE :  
- AS DEMONSTRATED IN AN LP - PROGRAM -

The integration of external functions into the system dynamics model is principally done in the following way:

The external subprograms (e.g. written in FORTRAN) are linked with the system dynamics model. Proper variables of this model are as input variables the arguments of the subroutine, which carries out the user defined computation. The result of these computations is transferred back to the section of the system dynamics model that brought the subprogram into action. To realize this concept important modifications had to be made at two different program levels. The DYNAMO language had to be extended by the new EXTRN statement. The source program of the DYNAMO II<sub>F</sub>-Compiler was modified in such a way that the new EXTRN statement could be processed.

A. Extension Of DYNAMO II<sub>F</sub>

In The Program Technique Context

The DYNAMO II<sub>F</sub>-Compiler consists of two main parts: the preprocessor and the runtime system. On condition of there are no syntax errors in the DYNAMO-code, the preprocessor package generates of the system dynamics model an equivalent FORTRAN-subroutine called ZZMOD (refer to figure 1)<sup>1)</sup> and a data file DAT in accordance with its constants and table functions. Subroutine ZZMOD, compiled by the FORTRAN compiler, thereupon is put into the temporary module library \*MOD. The linkage editor TSOSLNK links the module ZZMOD with the runtime modules stored in the

1) The flow diagram in Figure 1 illustrates the performance of the DYNAMO-

data set DYNAMOM, thereby yielding an executable program ZZDYN (see figure 1).

The runtime subprograms generally take care of the preparation and execution of tabular and graphical output, as well as the processing of RERUN-modifications. The rest of the routines of the runtime library consists of the popular DYNAMO functions to which the external functions (subroutines) are assigned at the same level.

In Figure 1, a separate data set MODULE has been created with modules of external subprograms (e.g. linear programming subroutines). To solve the problem of external subroutines the simulation language DYNAMO had to be extended by one more element called EXTRN so that function names used in the model could be declared as such.

The preprocessor system also had to be modified so that it could process the EXTRN statement. For this purpose the subroutine CRPRC of the preprocessor package which normally identifies macro definitions, equations, print and plot cards of a system dynamics model, was modified by inserting additional statements which make possible the recognition of the EXTRN command and which bring into operation the new subroutine EXTRNC. This subroutine was added to the preprocessor package to process the EXTRN statement and to include external subprograms to the runtime system.

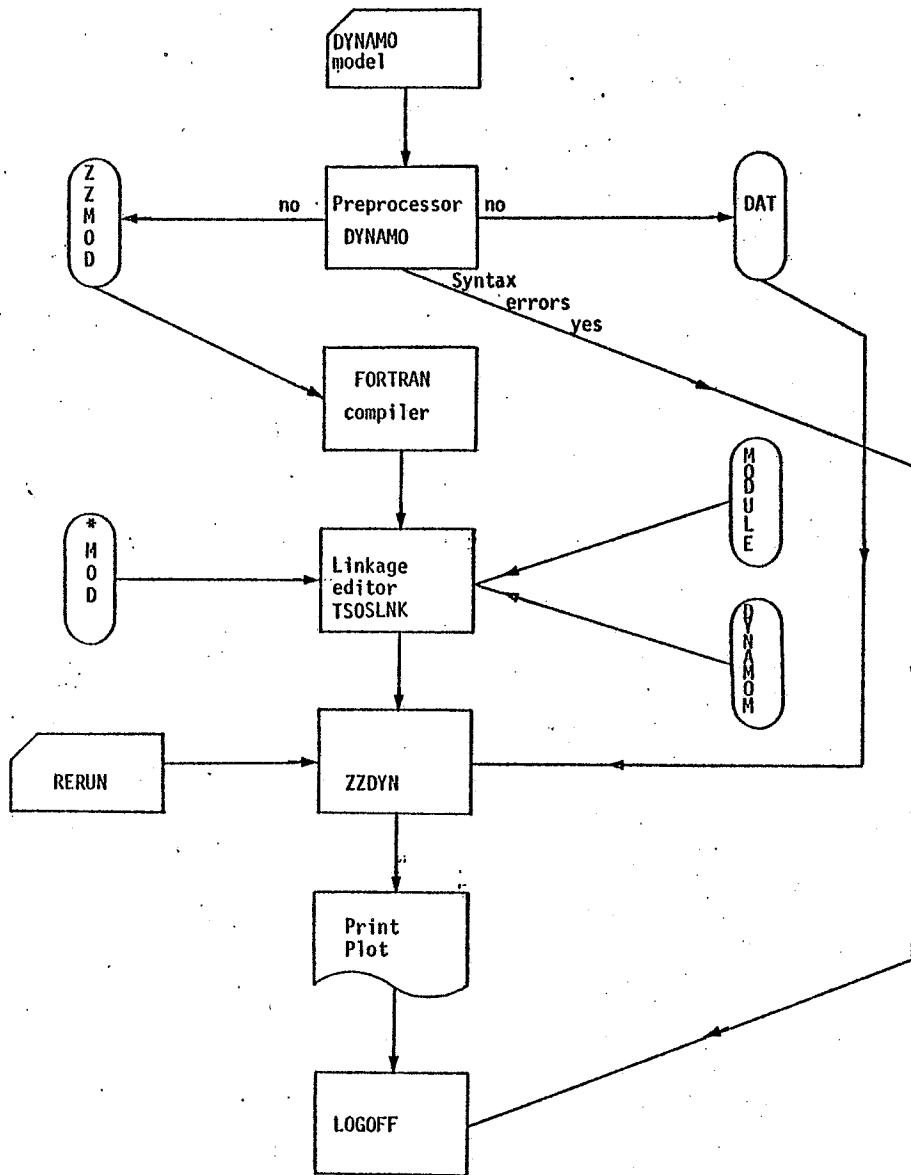


Figure 1: Flow diagram of the processing of a DYNAMO program.

### B. Extension Of DYNAMO II<sub>F</sub> In The Language Context

This technique described on the previous pages has the effect that in DYNAMO only the functional value will be assigned to an independent variable. The following rate equation<sup>2)</sup> illustrates this point:

```

EXTRN FORT
R    RATE1.KL=FORT(A1,A2,...,Ai)

```

The variable RATE1 gets the value of the function subprogram FORT at any time t<sub>i</sub>. The actual parameters A1 to A<sub>i</sub> represent information variables taken from the DYNAMO model.

In comparison to the official Pugh-Roberts<sup>3)</sup> version, a few major differences should be noted:

- a) with the EXTRN card, more than one function can be defined, for example, EXTRN NAME1, NAME2, ..., NAMEi; and
- b) the EXTRN card can be used in a DYNAMO model as often as necessary.

Inside a function subprogram, every operation allowed by FORTRAN can be executed and therefore an almost unlimited concatenation of structured programs can be achieved.

2) A rate equation is marked by the letter R; see Pugh (1973), p. 24

3) The simulation language DYNAMO respectively DYNAMO II<sub>F</sub>-precompiler is a product of Pugh-Roberts Associates, Inc., Cambridge, Mass.

C. An Application: LP - Model

The coupling principle is demonstrated on a fictitious investment problem. A company equipped with machine capacity  $MACHC = 3$  at time  $t = 0$  will be simulated.

The present production plan envisages processing five products, 1 to 5, which bring the profits 40, 70, 60, 90 and 30 VE (value units), on three machines A, B and C which are available for 195, 185 and 205 hours ( $b_i$ ) in the planning period. The objective function is to get the optimal gross margin (PROFR) associated with the subprogram name LINOPT. The variable MACHC is the actual parameter of the function LINOPT (see Figure 3, equation 3) which reads the time taken by each piece in production  $a_{ij}$ , the running time of each machine  $b_i$  as well as the coefficients of the objective function as input data.

$b_i$	machining time $a_{ij}$ for product					machine
	1	2	3	4	5	
195	5	6	3	1.5	7.5	A
185	3	7	4	9	11	B
205	0	9	3	7	16	C
0	-40	-70	-60	-90	-30	negativ coefficients of objective function

Figure 2: Basic data for the LP-Program

The subroutine LINMAX<sup>4)</sup> called by LINOPT finds the optimal gross margin per period on the basis of the observed data.

4) The subroutine LINMAX calculates the optimal gross margin with the simple Simplex algorithm, see Churchman, C.W.; Ackoff, R.L.; Arnoff, E.L. (1961)

This value will be passed on as functional value LINOPT to the DYNAMO program.

The equation 15 in the DYNAMO model describes the accumulation of optimal gross margin, TPROF, and the strategy of investment after the machine capacity has been extended by one unit if the TPROF exceeds the price for a new machine called PCOST. The machine will be put into operation where a major bottleneck in production occurs, which is determined in every time interval by LINOPT. The new machine unit is the same type as those machines already in use at that particular location. This modification of the value MACHC (see equations 4 to 7 in Figure 3) would always give an increased gross margin if fictitious boundary conditions could be assumed. A restriction is imposed by equation 10, which limits machine capacity MACHC to five units.

```

1  *  INTEGRATION OF OR-TECHNIQUES IN SD-MODELS
*****
2  EXTRN LINOPT
3  R  PROFR,KL=LINOPT(MACHC,K)
4  L  MACHC,K=MACHC,J+(DT)(MACHR,J,K)
5  N  MACHC=MACHC
6  C  MACHC=3
7  R  MACHR,KL=CLIP(1,0,TPROF,K,PCOST)(DECF,K)
8  C  PCOST=50000
9  A  DECF,K=CLIP(PCOST,0,TPROF,K,PCOST)
10 A  DECF,K=CLIP(FM1,FH2,MACHC,K,THACH)
11 C  FM1=0
12 C  FH2=1
13 C  THACH=5
14 R  EXPD,KL=(DECF,K)(DECF,K)
15 L  TPROF,K=TPROF,J+(DT)(PROFR,J,K-EXPD,J,K)
16 N  TPROF=TPROF1
17 C  TPROF1=0
18 NOTE SPECIFICATION PARAMETERS
19 C  DT=1
20 C  LENGTH=48
21 C  PLTPER=1
22 C  PRTPER=1
23 NOTE OUTPUT DATA
24 PRINT TPROF(3,3),MACHC,PROFR
25 PLOT TPROF=*(0,100000)/PROFR=P(0,100000)/MACHC=H(0,8)
26 RUN  RUN-OPT

```

Figure 3: DYNAMO-equations of the integration of an LP-program into a System Dynamics model

Results are presented in tabular form in Figure 4. The output shows, along with the typical DYNAMO output, the matrix which has been read, the intermediate result, and the result of the Simplex algorithm.

The unusual simplification used in this example serves only to present the principle of integration.

#### D. Conceptual Aspects Of The Integration Of System Dynamics And Statistical Or OR-Methods

A consideration of empirical adequacy can take place under the auspices of model formulation. In this context, the modeler may want to know according to which criteria and techniques of the observing data in the real system a system dynamics model was developed. From analysis of the model development process, techniques with dubious justification are often recognizable.

Estimation techniques have seldom if ever been used to define parameters of system dynamics models. Estimation methods are not mandatory if the required parameters are already known. For example, when behavior equations being fed into the model represent obligatory decision rules for the corresponding behavior unit which must be adhered to.<sup>5)</sup> For instance it is said in an obligatory way to make a permanent forecast of the amount of demand with the method of exponential smoothing with respect to a defined smooth constant. In this case it is reasonable to describe this behavior with the use of an information delay.

5) Zwicker (1973), p. 45

TIME	TPROF	MACHC	PROFR							
E+00	E+03	E+00	E+00							
0.00	0.000	3.00	2775.00	-----						
	READ IN MATRIX	190.00	5.00	6.00	3.00	1.50	7.50			
	READ IN MATRIX	185.00	3.00	7.00	4.00	9.00	11.00			
	READ IN MATRIX	205.00	0.00	9.00	3.00	7.00	16.00			
	READ IN MATRIX	0.00	-40.00	-70.00	-60.00	-90.00	-30.00			
	190.00	5.00	6.00	3.00	1.50	7.50	1.00	0.00	0.00	
	185.00	3.00	7.00	4.00	9.00	11.00	0.00	1.00	0.00	
	205.00	0.00	9.00	3.00	7.00	16.00	0.00	0.00	1.00	
	0.00	-40.00	-70.00	-60.00	-90.00	-30.00	0.00	0.00	0.00	
	OUTPUT DATA	51.25	2.75	0.75	0.00	-5.25	-0.75	1.00	-0.75	0
	OUTPUT DATA	46.25	0.75	1.75	1.00	2.25	2.75	0.00	0.25	0
	OUTPUT DATA	66.25	-2.25	3.75	0.00	0.25	7.75	0.00	-0.75	1
	OUTPUT DATA	2775.00	5.00	35.00	0.00	45.00	135.00	0.00	15.00	0
1.00	2.775	3.00	2775.00	-----						
	READ IN MATRIX	190.00	5.00	6.00	3.00	1.50	7.50			
	READ IN MATRIX	185.00	3.00	7.00	4.00	9.00	11.00			
	READ IN MATRIX	205.00	0.00	9.00	3.00	7.00	16.00			
	READ IN MATRIX	0.00	-40.00	-70.00	-60.00	-90.00	-30.00			
	190.00	5.00	6.00	3.00	1.50	7.50	1.00	0.00	0.00	
	185.00	3.00	7.00	4.00	9.00	11.00	0.00	1.00	0.00	
	205.00	0.00	9.00	3.00	7.00	16.00	0.00	0.00	1.00	
	0.00	-40.00	-70.00	-60.00	-90.00	-30.00	0.00	0.00	0.00	
	OUTPUT DATA	51.25	2.75	0.75	0.00	-5.25	-0.75	1.00	-0.75	0
	OUTPUT DATA	46.25	0.75	1.75	1.00	2.25	2.75	0.00	0.25	0
	OUTPUT DATA	66.25	-2.25	3.75	0.00	0.25	7.75	0.00	-0.75	1
	OUTPUT DATA	2775.00	5.00	35.00	0.00	45.00	135.00	0.00	15.00	0
20.00	5.500	4.00	2775.00	-----						
	READ IN MATRIX	190.00	5.00	6.00	3.00	1.50	7.50			
	READ IN MATRIX	370.00	3.00	7.00	4.00	9.00	11.00			
	READ IN MATRIX	205.00	0.00	9.00	3.00	7.00	16.00			
	READ IN MATRIX	0.00	-40.00	-70.00	-60.00	-90.00	-30.00			
	190.00	5.00	6.00	3.00	1.50	7.50	1.00	0.00	0.00	
	370.00	3.00	7.00	4.00	9.00	11.00	0.00	1.00	0.00	
	205.00	0.00	9.00	3.00	7.00	16.00	0.00	0.00	1.00	
	0.00	-40.00	-70.00	-60.00	-90.00	-30.00	0.00	0.00	0.00	
	OUTPUT DATA	61.97	2.12	1.73	1.00	0.00	1.73	0.42	0.00	-0
	OUTPUT DATA	97.58	2.70	-4.82	0.00	0.00	-9.82	-0.06	1.00	-1
	OUTPUT DATA	2.73	-0.91	0.55	0.00	1.00	1.55	-0.18	0.00	0
	OUTPUT DATA	3963.64	5.45	82.73	0.00	0.00	212.73	9.09	0.00	10
21.00	8.275	4.00	3963.64	-----						
	READ IN MATRIX	190.00	5.00	6.00	3.00	1.50	7.50			
	READ IN MATRIX	370.00	3.00	7.00	4.00	9.00	11.00			
	READ IN MATRIX	205.00	0.00	9.00	3.00	7.00	16.00			
	READ IN MATRIX	0.00	-40.00	-70.00	-60.00	-90.00	-30.00			

Figure 4: Results of the System Dynamics - LP model with machine capacity MACHC, the optimal gross margin PROFR and its accumulated variable TPROF

General criticism has been voiced in connection with the system dynamics techniques for defining parameters for the so-called exponential delays. These parameters are determined according to the kind of delay characteristics and the amount (length) of the average delay. The critics maintain that both criteria can then only be directly observed if in the actual system the size, defined as the input rate of a level, can be constructed in its running time so as to represent an impulse function. Since such a situation can practically never be realized, both criteria, the order of the delay and the delay constant, are nonobservable variables which have to be estimated based on the observed data of the input and output rates of the delay level in the described system.<sup>6)</sup> The wind can now be taken out of the sails of the criticism about the exponential delays by the integration of statistical method. With the use of parameter estimation techniques to determine so-called distributed lags, the order of the exponential delays as well as their average length can be calculated.<sup>7)</sup>

Further, the criticism which flamed up over Jay Forrester's views on authenticity tests in dynamic systems<sup>8)9)</sup> can be met with the help of the described integration method of external subprograms. The principal relationship between levels and rates can be modified by statistical elements. The amount of a rate may depend not only on one level of a system, but also on a certain random variable or random distribution whose density function is described and delivered by external functions.

6) Zwicker (1973), p. 46

7) Dhrymes (1971)

8) With respect to validation of dynamic simulation models see Naylor and Finger, "Verification of Computer Simulation Models", Management Science, Application 14 (1968), Series B, p. 92-101.

9) Stevin, "An Appreciation of Industrial Dynamics", Management Science 14 (1968), Theory Series, p. 383-397.

The structure of rates, which determines the transformation into actions, is shown in Figure 5.<sup>10)</sup> Rates are decisions which release certain actions according to given rules in so-called policies. A decision is made in accordance with an objective towards which the system should move. Between the objective and observed condition, which often differs from the actual realized condition, there is a discrepancy. From this discrepancy, an action takes place to eliminate the deviation.

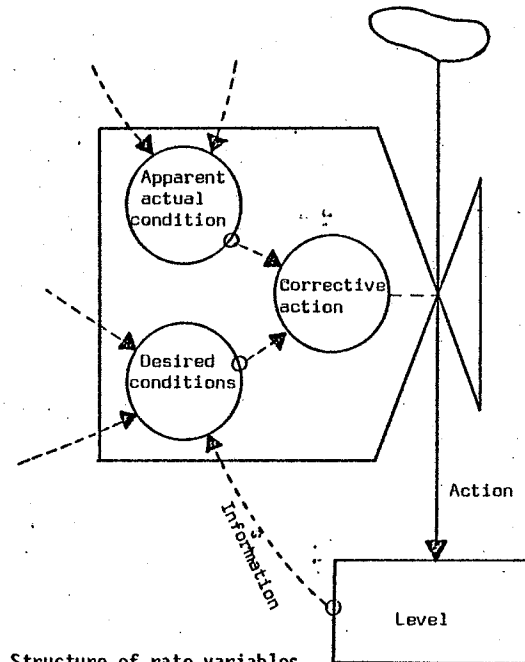


Figure 5: Structure of rate variables.

In acquiring an optimal value from an LP program on a rate, there lies a special implication for system dynamics philosophy. The LP program describes in detail with its secondary requirements a clearly defined microlevel (partial problem) and computes the op-

10) Forrester (1961), p. 93.

timal value of this partial problem. Delivery of the optimal value to the rate at any point in time  $t_i$  means that the action brought about by deviation is optimal.

With this special combination, the rate represents the optimal behavior of the real system.

## II. HANDLING INPUT-OUTPUT ANALYSIS IN SYSTEM DYNAMICS MODELS

The integration of a system dynamics model and an input-output analysis involves two different problem complexes. The simulation language DYNAMO must be modified and supplemented for application to matrix operations. And understanding the conceptual characteristics of this model-method integration can demonstrate what ever methodological expansion of the system dynamics method is realized through the integration.

### A. Modification Of The DYNAMO Language

A concept developed for the processing matrices can be used in the simulation language DYNAMO II<sub>F</sub>, as well as DYNAMO III<sub>F</sub>. An additional software package, integrated in the DYNAMO precompiler, takes on the following functions:

- a) input and output of defined arrays
- b) operations between those arrays
- c) access to an array component.

The subroutine MATRIX, having the entries MATRT and MATRF,

defines the arrays in which it generates an internal address list. MATRT copies the addresses of the array components (DYNAMO variables) over the addresses of the names of a coupled FORTRAN program. MATRF performs this process in the opposite direction. These program routines are programmed in Assembler. A matrix in the DYNAMO model is defined as follows:

- a) name of the array
- b) number of lines
- c) number of columns
- d) the  $i$ th line
- e) names of the components.

The following three dummy auxiliary equations called AA, BB and CC must be written to declare the matrix FELD(3,4):

```

A  AA.K = MATRIX(FELD,3,4,1, XB1, XB2, XB3, XB4)
A  BB.K = MATRIX(FELD,3,4,2, XA1, XA2, XA3, XA4)
A  CC.K = MATRIC(FELD,3,4,3, XC1, XC2, XC3, XC4).

```

The DYNAMO program begins with a declaration in this manner. The functioning of the modification is shown in two equivalent FORTRAN statements. The above DYNAMO equations correspond with the following DIMENSION and EQUIVALENCE statement:

```

DIMENSION  NAME1(2,2)
EQUIVALENCE NAME1(1,1),XB1),(NAME1(1,2),XB2)
              NAME1(2,1),XB3),(NAME1(2,2),XB4).

```

With the help of this technique, matrix operations can be formulated in the DYNAMO model, as for example:

$$A \text{ DD.K} = \text{MULT}(\text{FELDO}, \text{FELD1}, \text{FELD2})$$

The equation requires the multiplication of the matrices FELD1 and FELD2, and assigns the product to matrix FELDO (providing that the matrices, as in the example FELD(3,4) are defined).

Up to now, only the essential matrix operations have been admitted. A certain amount of caution is recommended with the semantics of matrix operations concerning its handling of the right sequence.

#### B. Conceptual Aspects Of The Model - Method Integration

The basis of input-output analysis is the pattern of the input-output table according to which empirical data are raised. An input-output table represents an expanded total economic activity, which in addition to the usual data separates the data into individual production sectors. The table therefore offers an insight into the economic production structure not obtainable with any other statistical instrument.

Input-output tables depict flows of goods and services between individual sectors of the economy. Reading a table by rows can determine how much a certain individual sector delivers to other sectors. In this way, the elements of the first row indicate how much the first production sector delivers to itself and to other sectors. The demand at this level reflects the sum of deliveries to the production sectors.

The remaining elements of the first row indicate which deliveries go to private households, government, investment (capital expenditures and inventories), or export - final demand sectors. The deliveries to the production and final demand sectors depict, by definition, the total production of the first sector. Reading the table by columns reveals the transfers which the individual sector receives from other sectors (inputs). Included in the so-called primary inputs are the imports which themselves include foreign raw material deliveries. The remaining primary inputs indicate what was applied in the form of capital depreciation, what flowed in the form of indirect taxes to the government sector, and what flowed in the form of income to labour and capital. The value tables can be consolidated as desired. Quantity tables can be produced by dividing the value tables by price levels.

Input-output analysis is a mechanism for depicting a complete economic structure. Applying the results and partial results of this method to the auxiliaries of a system dynamics model illustrates a picture of the real world. As proof of this statement in the following example, the so-called input coefficients of an input-output table are assigned to the auxiliaries of a system-dynamics model. The input coefficients,  $a_{ij} = \frac{x_{ij}}{x_j}$ , can be calculated from the input-output table for each production sector. They show what quantity of product  $i$  sector  $j$  used to produce one unit of output  $j$ . The input coefficients, and therefore the corresponding auxiliaries, can be interpreted as technical variables (analogous to baking recipes) or quantitative market variables which characterize the production structure of the economy underlying the table. The advantage of the estimates obtained with the input-output model integrated with a system dynamics model over other prognoses (sectorial or technological) lies in the systematic consideration of total economic interdependencies.



An application of the integration of a system dynamics model and input-output analysis is performed within the scope of a research project to judge new technologies (products and methods) of the chemical industry. This study should support governmental decision makers to find out whether these qualified chemical technologies can promote the securing and saving of raw materials. The information obtained with the input-output model can be used to project future natural resources and energy balances. The desired values gained from the computations can be compared to the estimated available raw materials and energy from the system dynamics model, revealing gaps in the raw material and energy balances. The advantage is obvious; timely, appropriate actions can be undertaken to eliminate emerging gaps (for example, by promoting appropriate technologies).

### III. SYSTEM DYNAMICS - MODEL IN A SUPERIOR PROGRAM STRUCTURE

On the one hand, system dynamics models provide information about the future behavior of problems described in the model; on the other hand, system dynamics models show how behavior can be influenced by a way of a user defined criteria. The latter demand is realized by the integration of the simulation model and a superior program structure, represented by a modified feedback loop. By means of the feedback method<sup>11)</sup>, the control system becomes identical to the system dynamics model, where output variables (state vector  $\underline{z}(\underline{s}, t)$ ) are compared with the objective function

11) Krallmann, H.: "Optimization of Simulation Models in a Feedback Loop Structure", in: Proceedings of the 1975 Summer Computer Simulation Conference (SCSC), San Francisco, July 1975, p. 1003.

of getting the deviations of the desired and actual value. The controller - the optimization algorithm - tries through the modification of the control vector  $\underline{s}(t)$  (the input variable of the system dynamics model) to minimize the deviations from the objective function. Figure 6 shows the discussed feedback structure.

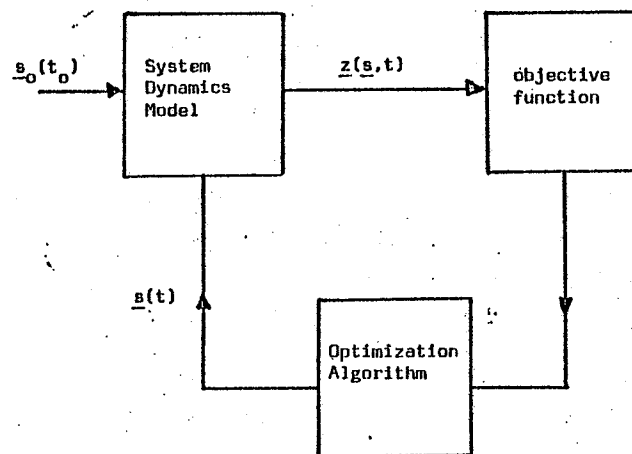


Figure 6: System Dynamics model in a feedback loop structure.

#### A. System Dynamics Applied To A "Polycondensation" - Plant

The simulation model applied in this optimization describes a chemical process inherent in the main condenser of a "TPA"-plant. The adjustments to the pressure and temperature should be computed at each time point  $t_i$  for the throughput changes from 900 kilogram per hour (kg/h) to 600 kilogram per hour (kg/h). The essential conditions in the case are that the viscosity SV at the entry of the main condenser SVE remains constant (SVE = 390 SV-units) while viscosity of the output SVA1 should run between defined limits.

The equations, written in the DYNAMO language<sup>12)</sup>, describe the structure of the problem (see Figure 9). A detailed description of the DYNAMO language is not necessary<sup>13)</sup>.

The system dynamics model can be extended by the three external FORTRAN subroutines LAST, BER and BER1, which are linked during the running time of the model.

The function LAST is a discrete delay function. The actual variable (for example, V/K) is delayed n times, dependent on the parameter DT.

The function  $f_i(t)$  shown in Figure 7 is realized by the FORTRAN subprogram BER. The graph shows the principal dependence of changes in the throughput called  $\Delta D$  to the viscosity at the output SVA1 and has the same shape for the dependence of the changes on the pressure called  $\Delta V$  to SVA1. The slope  $m$  and the magnitude  $\Delta G$  have to be replaced by the values of the corresponding relationships of  $\Delta D$  to  $\Delta SVA1$  and  $\Delta V$  to  $\Delta SVA1$ .

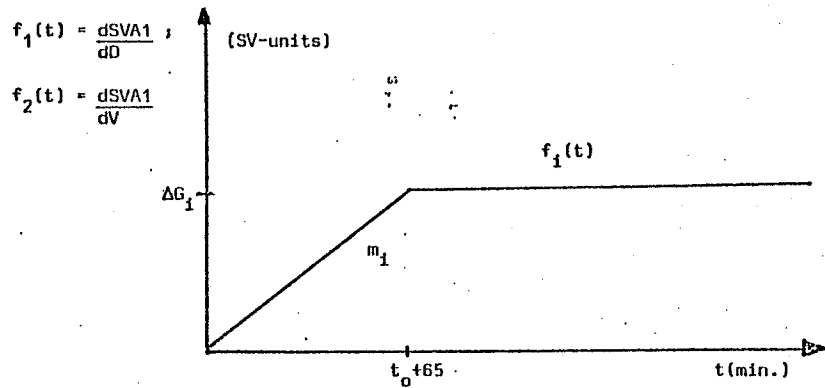


Figure 7: Viscosity SVA1 = f(ΔD); SVA1 = f(ΔV).

12) An extensive description of the simulation language DYNAMO can be found in: Pugh (1973)  
 13) For explanations of the system dynamic method, see Forrester (1968), Zahn (1970) and Krallmann (1976)

The subroutine BER1 computes the course of curve  $f_T(t)$ , shown in Figure 8. The output function  $f_T(t)$  illustrates the influence of changes in temperature on the viscosity SVA1 ( $\frac{dSVA1}{dT}$ ).

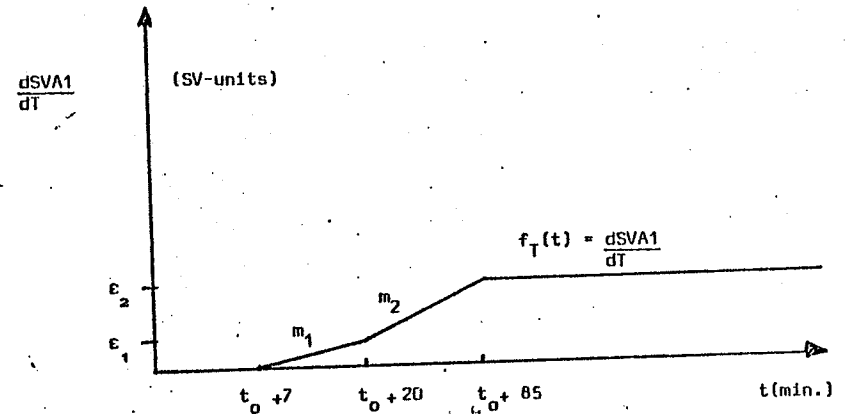


Figure 8: Output viscosity SVA1 = f(ΔT).

With regard to Figure 9, the polycondensation model must be described in some detail. The EXTRN instruction states the external subroutines BER, BER1 and LAST added for running in the DYNAMO model. The variables A and B are parameters of the control vector, varied by the optimization algorithm called razor-search in such a way that the suitable viscosity SVA1 remains constant. With the integration of the model into Figure 9 feedback loop, the initial value of the parameters A and B are replaced by the computed values of the optimization algorithm. The equations 5 - 15 describe the throughput change D from 900 kg/h to 600 kg/h at the definite time point  $t_0 = 20$ , and the functional dependence of the throughput change on the initial viscosity SVA1. The following formula illustrates the effect:  $\frac{dSVA1}{dB} = \alpha \frac{SV-units}{(kg/h)}$ . A sudden change in D requires 65 minutes to get the new level of SVA1.

```

1 * SIMULATIONSMODELL POLYKONDENSATIONSANLAGE
2 N. A=1
3 N. B=1
4 EXTRN BER, LAST, BER1
5 A D.K=CLIP(600,D1,TIME.K,ZP)
6 C D1=900
7 C ZP=20
8 C DSVAD=-0.7
9 A EDD.K=D1*DSVAD+DDD.K
10 A DDD.K=BER(ZAHLO,TIME.K,DELTAD.K,DT,DSVAD)
11 C ZAHLO=2
12 A DELTAD.K=D.K-DD.K
13 C ANFD=900
14 A DD.K=LAST(D.K,DT,ANFD,LENGTH,TIME.K,3.)
15 C D1=900
16 A V.K=A.K+V0.K
17 A V0.K=TABHL(TV0,TIME.K,0,300,60)
18 T TV0=1.8/1.8/1.8/1.8/1.8/1.8
19 C DSVAV=-39
20 A EVV.K=V1*DSVAV+VVV.K
21 A VVV.K=BER(ZAHL,TIME.K,DELTAV.K,DT,DSVAV)
22 C ZAHL=1
23 A DELTAV.K=V.K-VV.K
24 C ANFV=1.8
25 A VV.K=LAST(V.K,DT,ANFV,LENGTH,TIME.K,1.)
26 C V1=1.8
27 A T.K=B.K+100.K/KORRF
28 C KORRF=100
29 A T00.K=TABHL(TT0,TIME.K,0,300,60)
30 T TT0=295.3/295.3/295.3/295.3/295.3/295.3
31 C DSVAT=28
32 A ETT.K=TI*DSVAT+TTT.K
33 A TTT.K=BER1(ZAHL1,TIME.K,DELTAT.K,DT)
34 C ZAHL1=2
35 A DELTAT.K=T.K-TT.K
36 C ANFT=295.3
37 A TT.K=LAST(T.K,DT,ANFT,LENGTH,TIME.K,2.)
38 C TI=295.3
39 A SVA1.K=EVV.K+ETT.K+EDD.K+CC+DSVAE+SVE
40 C DSVAE=0.2
41 C SVE=390
42 C CC=-6981.2
43 NOTE OUTPUTPARAMETER
44 C PLTPER=1
45 C PRTPER=1
46 C LENGTH=250
47 C DT=1.0
48 PRINT SVA1,EDD,ETT,EVV,A,B,V,T,D
49 PLOTN SVA1*(550,750)/V=0(1,4)/T=1(290,300)/D=2(590,910)
50 RUN RUN=RAMOLLA

```

Figure 9: Simulation Model of a "Polycondensation"-Plant

The next ten equations (16 - 25) express the dependence of pressure changes on the viscosity SVA1. During a longer running period of the main condenser, the normal pressure is 1.8 Torr (equation 17), which can be modified in desired limits fixed by the physical conditions of the plant. The variable VVV determines the amount of influence of pressure fluctuations on SVA1.

The second component of the control vector  $\underline{s}(t)$  is the transformer temperature, with a standard value of  $T_0 = 295.3^\circ\text{C}$ . The equations 27 - 38 explain the influence of temperature and variations in temperature on viscosity SVA1. The constant KORFF normalizes the control parameter B at its permissible critical values (to be explained later). The subroutine BER1 computes the effects of temperature fluctuations on SVA1 in accordance with the function  $f_T(t)$  (see Figure 8). Referring to Figure 8, 7 minutes are required to adapt the condenser cover to a temperature change ( $T_0 \rightarrow T_1$ ) and another 13 minutes are needed to adjust the melt.

Subsequently, a chemical reaction runs off corresponding to the following formula:  $\frac{dSVA1}{dt} = \beta \frac{SV\text{-units}}{CO\text{-min}}$ . Equation 32 expresses the total impact of temperature changes on SVA1. During the simulation of the model, the viscosity at the input of the main condenser is assumed to be constant (SVE=390 SV-units).

Equation 39 calculates SVA1 as a function of the pressure, temperature, throughput, and viscosity SVE. The length of simulation of the chemical process of a main condenser is 250 periods (minutes).

B. The Optimization Of A Polycondensation Plant By Means Of The Razor-Search Algorithm

The system dynamics model of the polycondensation plant has been integrated as a control system into the feedback loop illustrated in Figure 6. The control vector s(t) - the input variable to the model - consists of the components temperature T and pressure V. The viscosity SVA1 is the state vector z(s,t) controlled by the optimization algorithm.

The optimization algorithm acting as controller is the so-called razor-search method developed by Bandler and Macdonald<sup>14)</sup>. The algorithm is a modified pattern-search method, belonging to the direct search and climbing procedures<sup>15)</sup> for optimizing multidimensional problems.

The pattern-search<sup>16)</sup> method is an improved direct search procedure, which in this problem has the main task of directing the output variables z\_i(s,t) of the model into the user defined solution space described in the object function U(s,t). To do so, the modeler must vary the parameters of the control vector s(t) by exploratory moves.

Experience with pattern-search method indicates that the procedure is very efficient in reaching an optimum also in circum-

14) Bandler, J.W., Macdonald, P.A., "Optimization of Microwave Networks by Razor Search", in: IEEE Transaction on Microwave Theory and Techniques, Vol. MIT-17, No. 8, August 1969, pp. 552-562.

15) Wilde and Beightler (1967), pp. 271-273.

16) Wilde (1964), p. 146. "In visualizing what is meant by a "pattern" it is helpful to think of an arrow, its base at one end and its head at the other".

stances where the feasible region<sup>17)</sup> for the control vector has fairly narrow valleys in it. Classical methods (such as steepest descent-, generalized Newton-Raphson-, Fletcher-Powell method etc.) slow down or even fail to get an optimum in such cases.

An important modification of the pattern-search procedure is the so-called razor-search method. This routine overcomes the difficulties of discontinuous partial derivatives with respect to the control variables. Otherwise efficient search methods fail to converge, particularly when the objective function's hyper-space includes narrow curved valleys in the vicinity of the path of discontinuous partial derivatives.

The razor search routine normally overcomes these difficulties by a search strategy that begins with a version of pattern-search<sup>18)</sup> and then applies this until it fails.

Then, the procedure automatically selects a random point in the immediate neighborhood. The random point is selected so that:

$$s_i(t) = s_i^0(t) + \rho \cdot R(n) \cdot \epsilon$$

where s\_i(t) is the new value of the ith control variable, s\_i^0(t) is the old value of the ith control variable, rho is a scale factor, R(n) produces random numbers between -1, and +1, and epsilon represents the current value of the exploratory increment.

17) In general, there will be constraints that must be satisfied either during optimization or by the optimum solution. Each parameter might be constrained explicitly by an upper and lower bound as follows: s\_ei <= s\_i <= s\_ui i=1,2,...,k where s\_e and s\_u are lower and upper bounds, respectively. Any vector s which satisfies the constraints is termed feasible; it lies in a feasible region.

18) Hooke and Jeeves (1961)

When the pattern-search fails again the same valley (or boundary) is assumed to be responsible, and an attempt is undertaken to establish a new pattern in the direction of the minimum. The process is automatically repeated until any of several possible terminating criteria are satisfied.

The razor-search presented in this paper has two further essential characteristics<sup>19)</sup>:

- 1) The exploratory increments depend on the total progress made between the previous two base points. Therefore, they automatically increase or decrease in accordance with previous successes or failures, respectively.
- 2) When a pattern move plus exploratory moves fail, the pattern is not immediately discarded. Instead, the same procedure is repeated closer to the base point. If this effort also proves unsuccessful, the procedure is attempted in the opposite direction.

The razor-search method has been successfully applied to microwave optimization.

The optimization algorithm, the razor-search procedure, is responsible for varying the parameters of the control vector A and B so that the state variable SVA1 follows a user defined objective function.

The control variables pressure and temperature may only be

modified within fixed boundaries. The current limits for the pressure are  $1.0 \leq V \leq 4.0$  (Torr). With regard to the standard pressure  $V_0 = 1.8$  Torr, the range of parameter A is  $0.556 \leq A \leq 2.221$ . The lower boundary of the temperature is given as  $290^\circ\text{C}$  and the upper is  $300^\circ\text{C}$ . Consequently, within a standard temperature of  $295.3^\circ\text{C}$ , the range of parameter B is  $0.9821 \leq B \leq 1.0169$ . The tolerance zone for the viscosity SVA1 is  $\text{LIM1} \leq \text{SVA1} \leq \text{LIM2}$  (see Figure 10). With knowledge of the chemical reaction process which take place by a sudden throughput changes D from 900 kg/h to 600 kg/h, the objective function can be illustrated for SVA1 in a more precise and time-variable manner.

The objective function, which is used to optimize the polycondensation plant, is written in FORTRAN in the following equation:

$$U = \text{GEWA} \cdot \text{ABS}(\text{LIM}(1) - A) + \text{GEWA} \cdot \text{ABS}(\text{LIM}(2) - A) \\ + \text{GEWB} \cdot \text{ABS}(\text{LIM}(3) - B) + \text{GEWB} \cdot \text{ABS}(\text{LIM}(4) - B) \\ + \text{GEWS} \cdot \text{ABS}(\text{SVA1} - \text{CONST})$$

- U - value of the objective function
  - GEWA - weighting factor of the variable A
  - LIM(1) - lower boundary of A
  - LIM(2) - upper boundary of A
  - A - parameter of the control variable pressure
  - GEWB - weighting factor of the variable B
  - LIM(3) - lower boundary of B
  - LIM(4) - upper boundary of B
  - B - parameter of the control variable temperature
  - GEWS - weighting factor of the variable SVA1
  - SVA1 - viscosity at the output of main condensor
  - CONST - desired value of SVA1
- with  $\text{GEWA} = \text{GEWB} = 100$  and  $\text{GEWS} = 1$

19) Bandler and Macdonald, "Optimization ..., op. cit., p. 554.

The variable CONST is represented by the dotted line in Figure 10.

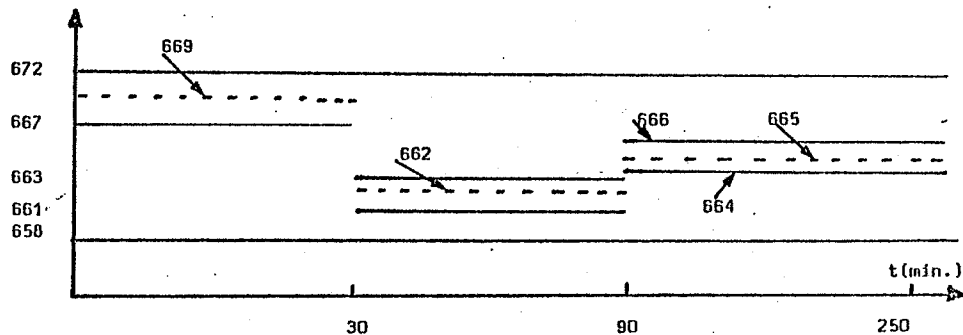


Figure 10: Objective function of the output viscosity SVA1.

At each time point  $t_i$ , the optimization algorithm razor-search tries to vary the parameters A and B of the control variables pressure and temperature within the defined boundaries so that the state variable SVA1 follows the desired value SVA1 as closely as possible<sup>20)</sup>.

#### IV. RESULTS OF THE OPTIMIZATION PROCEDURE

Optimization of the simulation model has been executed on a SIEMENS-computer 4004/151 in the computer center at the University of Mannheim.

Figure 11 shows the results of the optimization. In columns

20) The theoretical foundations and discussions surrounding the organization of computer programs using a system dynamic model with razor-search procedures as well as the dynamics within the feedback loop, are explained fully in: Krallmann, *op. cit.*, pp. 183-190.

6 and 7, the values of parameters A and B<sup>21)</sup> of the control vector  $\underline{s}(t)$  ( $s_1$  = pressure and  $s_2$  = temperature) are printed over 36 time periods. The next two columns represent the corresponding orders of magnitude of pressure and temperature.

Without any impact from outside the following chemical process would take place in the polycondensation plant: at time point  $t_0$  a throughput change from 900 kg/h to 600 kg/h causes the viscosity SVA1 to rise to a level of 875 SV-units during the time interval of  $t_0 + 65$  minutes. The pressure V could be increased to 4.0 Torr at time point  $t_0$ ; consequently, the viscosity SVA1 would reach a level of 794.2 SV-units after  $t_0 + 65$  minutes. During the optimization procedure, the critical time period is 65 minutes after the throughput change.

The optimization algorithm razor-search succeeds in controlling the viscosity SVA1 with continuous and simultaneous modifications of the control variables, pressure and temperature, so that SVA1 follows the desired value CONST within the defined boundaries<sup>22)</sup>. Figure 12 shows the changes in pressure and temperature to stabilize viscosity SVA1 in the fixed range<sup>23)</sup>.

- 21) The parameter B has to be divided by 100 to get the normal value of 1.
- 22) At the critical time point  $t_0 + 65$ , SVA1 is tangent to the upper boundary.
- 23) The CPU-time used in the optimization of this SD-model is 1410.44 sec. on the SIEMENS-computer 4004/151. This considerable CPU-time can be explained by the numerical amount of time to compute the functions BER and BER1. However, with regard to size and complexity of the total model, the order of magnitude of the CPU-time is justifiable.

TIME	SVA1				Parameters		Pressure	Temperature	Throughput
		A	B		A	B			
0.00	664.99	-630.00	8268.39	-70.20	1.00	100.00	1.60	295.50	900.00
2.00	665.35	-630.00	8268.39	-69.84	0.82	100.18	1.48	295.83	900.00
4.00	UPHO =	515	3472		PHO =	0.82	0346	100.	799
	UPHO =	515	0347		PHO =	0.82	0346	100.	799
6.00	665.74	-630.00	8268.39	-69.45	0.82	100.18	1.48	295.83	900.00
8.00	UPHO =	514	7300		PHO =	0.82	0346	100.	799
	UPHO =	514	5771		PHO =	0.82	0346	100.	799
10.00	666.13	-630.00	8268.39	-69.07	0.82	100.18	1.48	295.83	900.00
12.00	UPHO =	514	8857		PHO =	0.82	0346	100.	799
	UPHO =	515	1982		PHO =	0.82	0346	100.	799
14.00	666.61	-630.00	8268.49	-68.68	0.82	100.18	1.48	295.83	900.00
16.00	667.06	-630.00	8268.71	-68.45	0.97	100.03	1.75	295.34	900.00
18.00	666.35	-630.00	8268.95	-69.39	1.59	99.41	2.86	293.55	900.00
20.00	664.82	-630.00	8269.18	-71.15	1.89	99.11	3.40	292.67	900.00
22.00	662.81	-630.00	8269.40	-73.39	2.11	98.89	3.88	292.02	900.00
24.00	660.07	-630.00	8269.24	-75.97	2.19	98.26	3.95	290.15	900.00
26.00	UPHO =	515	0972		PHO =	2.2	4053	98.2	728
	UPHO =	515	626.77		PHO =	2.20	98.25	3.97	290.12
28.00	659.85	-626.77	8268.39	-78.57	2.20	98.25	3.97	290.12	600.00
30.00	664.20	-620.31	8268.91	-81.21	2.22	98.21	4.00	290.02	600.00
32.00	666.65	-613.05	8267.54	-83.85	2.22	98.21	4.00	290.02	600.00
34.00	UPHO =	524	0281		PHO =	2.2	0870	98.2	086
	UPHO =	524	607.38		PHO =	2.21	98.22	3.98	290.05
36.00	668.49	-607.38	8265.55	-86.47	2.21	98.22	3.98	290.05	600.00
38.00	670.71	-600.92	8263.52	-88.69	1.99	98.44	3.58	290.70	600.00
40.00	669.09	-594.46	8257.57	-90.82	1.98	98.45	3.57	290.72	600.00
42.00	666.80	-588.00	8250.94	-92.94	1.98	98.45	3.57	290.72	600.00
44.00	UPHO =	524	3064		PHO =	1.98	0897	98.4	073
	UPHO =	524	581.51		PHO =	1.98	98.45	1.57	290.72
46.00	666.12	-581.51	8245.91	-95.05	1.98	98.45	1.57	290.72	600.00
48.00	UPHO =	523	0564		PHO =	1.98	0897	98.4	073
	UPHO =	523	9109		PHO =	1.98	0897	98.4	073
50.00	663.49	-575.08	8238.94	-97.17	1.98	98.45	3.57	290.72	600.00

Figure 11: Results of the simulation run (0 ≤ t ≤ 36) of the "Polycondensation" plant model.

In reality, all processes elapse with time delays. Taking these delays into consideration during model construction, the para-

meter changes to the control vector  $\underline{s}(t)$  have a delaying action on the behavior of the state variables  $z_i(\underline{s}, t)$ . Because of this inertness, the razor-search procedure has to be modified by an additional logical step and a program change. To register the impact of the parameter changes on  $\underline{s}(t)$ , the system dynamics model has to be simulated ahead with a definite time interval dependent on the longest delay constant of the model. If the parameters of  $\underline{s}(t)$  result in an optimization of  $z_i(\underline{s}, t)$  at the final time point of the "advance simulation", the values of  $\underline{s}(t)$  are accepted. Otherwise, the procedure tries to find a new set of parameter constellations for  $\underline{s}(t)$ , with which the "advance simulation" is repeated<sup>24)</sup>.

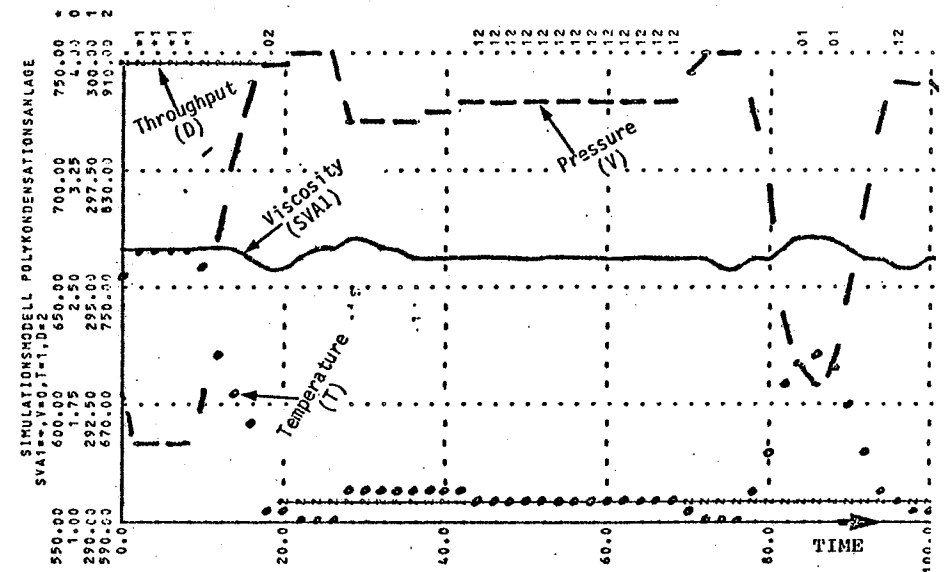


Figure 12: Plotted results of the Polycondensation plant model (0 ≤ t ≤ 100).

24) Krallmann, *op. cit.*, pp. 183-186.

The time interval of the "advance simulation" during the optimization of the polycondensation plant is 10 time units. The simulation runs on two levels. On the first level, at each time point  $t_i$ , the "advance simulation" of the model is executed with 10 time units; on the second level the set of system dynamics equations is computed step by step ( $t_{i+1} = t_i + DT$ ) with the parameters of  $\underline{s}(t)$  confirmed by the "advance simulation", therefore, this parameter constellation optimizes the  $z_i(\underline{s}, t)$  with regard to the objective function.

BIBLIOGRAPHY

Ansoff, H.I.; Slevin, D.P. "An Appreciation of Industrial Dynamics", in: MS 14(1968), Theory Series, pp. 383-397

Bandler, J.W.; Macdonald, P.A. "Optimization of Microwave Networks by Razor Search", IEEE Transactions of Microwaves Theory and Techniques, Vol. MIT-17, No. 18, 1969, pp.552-562

Dhrymes, P.J. Distributed Lags, Problem of Formulation and Estimation, San Francisco 1971

Forrester, J.W. Industrial Dynamics, Cambridge Mass. 1961  
Principles of Systems, Cambridge, Mass. 1968  
Urban Dynamics, Cambridge, Mass. 1969  
World Dynamics, Cambridge, Mass. 1971

Krallmann, H. Heuristische Optimierung von Simulationsmodellen mit dem Razor Search Algorithmus, Basel und Stuttgart 1976

Krallmann, H. "Optimization of Simulation Models in a Feedback Loop Structure", in Proceedings of the SCSC, San Francisco 1975

Lehmann, G. Wirtschaftswachstum im Gleichgewicht, Stuttgart 1975

Milling, P. Der technische Fortschritt beim Produktionsprozeß, Wiesbaden 1974

Naylor, T.H.; Finger, J.M. "Verification of Computer Simulation Models", in: MS, Application 14 (1968), Ser. B, pp. 92-101

Pugh, A.L.III DYNAMOII, User's Manual, fourth edition, Cambridge, Mass. 1973

Wilde, D.J. Optimum Seeking Methods, Englewood Cliffs 1964

Wilde, D.J.; Beightler, C.W. Foundations of Optimization, Englewood Cliffs 1967

Zahn, E. Das Wachstum industrieller Unternehmen, Wiesbaden 1970

Zwicker, E. Dynamische Systemanalyse I, Kurshandbuch, Berlin 1973

Churchman, C.W.; Ackoff, R.L. Operations Research - Eine Einführung in die Unternehmensforschung, Wien, München 1971

Arnoff, E.L.