DOCUMENT RESUME

ED 111 352 IR 002 400

AUTHOR Talpaz, Hovav
TITLE Computer Assistance in Teaching Dynamic-Stochastic Systems Analysis.
PUB DATE Jun 75
NOTE 20p.; Paper presented at the Conference on Computers in the Undergraduate Curricula (6th, Fort Worth, Texas, June 16-18, 1975); Related document is IR 002 399
AVAILABLE FROM Entire Proceedings; Ted Sjoerdsma, Treasurer, CCUC, 1248 Lindquist Center for Measurement, Iowa City, Iowa 52242 ($10.00; Checks payable to the University of Iowa)

EDRS PRICE MF-$0.76 HC-$1.58 Plus Postage
DESCRIPTORS *Computer Assisted Instruction; *Computer Programs; Computers; *Course Descriptions; Course Objectives; Graduate Students; Higher Education; Programing Languages; Simulation; *Systems Analysis
IDENTIFIERS CSMP 360; FORTRAN IV; Stochastic Analysis

ABSTRACT

A university level course in systems analysis with close contact and massive use of computer time was designed. The objectives of the course were primarily to teach social science graduate students, mostly from economics and agricultural economics, the basic methodological and quantitative tools of systems analysis and design. It was designed to show that the task of teaching principles of systems analysis and its applications to students with no programing background can be achieved using high level simulation language like CSMP/360. Stochastic situations represented by Monte-Carlo simulation were designed in FORTRAN with some preprogramed segments given to the students. An example of such an assignment is provided. (Author/KKC)

****
* Documents acquired by ERIC include many informal unpublished *
* materials not available from other sources. ERIC makes every effort *
* to obtain the best copy available. nevertheless, items of marginal *
* reproducibility are often encountered and this affects the quality *
* of the microfiche and hardcopy reproductions ERIC makes available *
* via the ERIC Document Reproduction Service (EDRS). EDRS is not *
* responsible for the quality of the original document. Reproductions *
* supplied by EDRS are the best that can be made from the original. *
****

ED111352

COMPUTER ASSISTANCE IN TEACHING DYNAMIC-STOCHASTIC SYSTEMS ANALYSIS

Hovav Talpaz*

Department of Agricultural Economics
Texas A&M University

U S DEPARTMENT OF HEALTH, EDUCATION & WELFARE
NATIONAL INSTITUTE OF EDUCATION
THIS DOCUMENT HAS BEEN REPRODUCED EXACTLY AS RECEIVED FROM THE PERSON OR ORGANIZATION ORIGINATING IT POINTS OF VIEW OR OPINIONS STATED DO NOT NECESSARILY REPRESENT OFFICIAL NATIONAL INSTITUTE OF EDUCATION POSITION OR POLICY

SE 002 400

ABSTRACT

The task of teaching principles of systems analysis and its applications to students with no programming background can be greatly achieved using high level simulation language like CSMP/360. Various stochastic situations represented by Monte-Carlo simulation can be designed in FORTRAN with some pre-programmed segments given to the students. Heavy load of computer homework assignments contributed to a much better understanding and to the speed of learning and higher motivation. Example of such an assignment is provided.

ERIC

COMPUTER ASSISTANCE IN TEACHING DYNAMIC-STOCHASTIC SYSTEMS ANALYSIS

Hovav Talpaz*

Introduction

The systems concept has come to play a critical role in contemporary science [2,4] and has become an indispensable component of not only engineering but also the behavioral studies, especially at the university graduate program level [3,6,7,8].

Though an old concept, systems analysis became a useful discipline only with the appearance of the electronic computer, which, in combination with systems methodology [2,7], made it possible to solve complex problems with wide areas of application. Yet, another decade was to pass before the computer became generally available as an educational tool for use in systems analysis courses.

In Agricultural Economics, the volume of computer use would probably rank research first, extension second and teaching third. It is beyond the scope of this paper to investigate the reasons for this. A few of the reasons include physical inaccessability of computer services, funding structures, lack of instructors' experience, and conservatism in teaching methods as demonstrated by class lectures and non-computer oriented textbooks.

Recently, this situation has changed due to the availability of time-sharing and telecommunication facilities and to the cost rates being lower and accessibility much improved. More computer oriented textbooks in related fields like simulation [6,8], numerical analysis [1] and the systems approach [7] provide an initial base for the instructor in the development of computer aided instruction in the area of systems analysis. The aim of this paper is to describe such an attempt made recently in the Department of Agricultural Economics, Texas A&M University.

The Course Objectives

The objectives of the course were primarily to teach social science graduate students, mostly from economics and agricultural economics, the basic methodological and the quantitative tools of systems analysis and design [2,3,7]. Applied, as well as theoretical, consideration must be given so that planning and management of "real world" problems can be studied and solved. The student is expected to develop projects and homework assignments which involve simulation or to design a system to be checked and run on the computer along with a short essay type discussion summarizing the results. An example of such a homework assignment is provided in Appendix A.

Course Content

The content of the two-semester graduate course is outlined as follows:[1]

1. Philosophy and methodology of systems science with references [2,4,8].
2. Mathematical modeling of dynamic deterministic systems with references [3,7,8].
3. Simulation techniques of dynamic-stochastic systems with references [7,9,10].
4. Control problems and optimization techniques with references [1,5,7,8,9].

The computer programs used FORTRAN IV and CSMP/360 [6] languages and were run on the Texas A&M University's 360/65 and/or 370/145 computers.

The Computer Assistance in Force

As mentioned above, emphasis was placed on having the students implement segments of systems science by using the computer. However, there was a wide range of backgrounds and experiences with regard to computer programming. This problem will probably persist at least for the next few years. Limiting the course attendance to only those who have had certain programming experience, such as competence in FORTRAN, would have jeopardized the course due to lack of potential candidates qualified to take the course.

In addition, a certain capability in computer programming and application is needed early in the game to meet the course objectives. To cope with this problem, a strategy was selected which will be discussed in three components: a) the simulation language, b) the FORTRAN extension and c) the system-algorithmic approach.

a) The Simulation Language

Careful consideration was given to the selection of the computer simulation language. It needs to have the capacity to handle dynamic and stochastic situations, and yet be easy to learn. The CSMP/360 language is well fitted for this purpose because it took only a few hours to teach the non-programmers enough features and structures of the language to allow them to advance on their own by assigning simple problems and access to the User's Manual [6]. The capacity of this language to handle dynamic-stochastic systems is well established and documented [6, pp. 72-76]. The students were introduced to CSMP by having them generate wave signals using combinations of step, ramp, pulse and impulse functions. The students demonstrated their understanding of singular function concepts, as well as computer programming, in these exercises. The second step involved the solving of systems of n order differential equations by the various Laplace equivalent function blocks. This step also allowed non-linear interpolation to be easily accomplished.[2] The third step introduced stochastic processes which could be accomplished by the use of some simple FORTRAN programs.

b) The FORTRAN Extension

Due to the need to estimate model parameters, make data transformations, do matrix operations and generate stochastic variates distributed according to specific probability functions, some FORTRAN programming was required. A few FORTRAN help-sessions provided the students with the necessary level of skill. At this point, the students were taught to generate stochastic variates, Markov Chains and Monte-Carlo Simulations [9,10]. In each case, the students were furnished with the appropriate main computer programs which sometimes required the addition of certain subroutines in order to accomplish the assignment. The FORTRAN segments were also used for the simulation of discrete time models. The project assignments were carried out by small teams of students with at least one experienced in FORTRAN programming. The inexperienced were expected to learn enough FORTRAN so that the assignment could be carried out together.

c) The System-Algorithmic Approach

The complex project assignments were time consuming to perform. Therefore, in order to reduce the time required by students and the computer cost of instruction, an algorithmic approach was adapted whereby the overall program for each problem assignment was divided into program segments as can be seen in Figure 1.

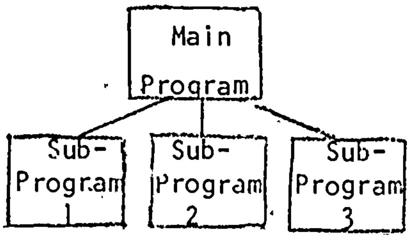


Figure 1.

In this schematic, the solution for the problem requires three different algorithms to operate in a particular sequence dynamically. The goal is to train the students by using only the design of two specific algorithms. The overall program can then be segmented into a main program connecting three subroutines (macros or procedures in the case of CSMP) where the main program and the remaining subroutines are designed and applied by the students, who in turn devote a majority of their effort and time to the main assignment. Certain plot routines, estimations, optimizations or even complicated matrix operations may be beyond the capacity of the inexperienced programmer and consume too much of his limited time. This approach requires the instructor to carefully prepare these computer routines in advance.

Some Observations

This paper described briefly an effort to design a university level course in systems analysis with close contact and massive use of computer time in training the student along the entire spectrum of activities demanded by this discipline. In this case, the computer becomes the "laboratory room" for the systems scientist in much the same way students of chemistry or physics need their laboratory experience. On the average, the students spent a ratio of 2.5:1.0 in computer design programming, debugging and analysis versus class lecture time. It should be noted, though, that most of them did not take any other course and only shared their time with their ongoing research activities.

The results were encouraging and point to the need for further effort in the direction of additional computer exercises and the removal of much of the programming burden from the students' assignments by providing them with programmed segments.

Another significant step forward may be expected as computer satellite terminals become more available to students. Although both FORTRAN and CSMP are available through such terminals, the inclusion of interactive languages, such as APL, may provide additional convenience and flexibility in performing the course exercises and projects.

ERIC

Footnotes

*Assistant Professor, Department of Agricultural Economics, Texas A&M University.

[1]The interested reader could get a detailed view of the course content by following the cited references or by contacting the author.

[2]See Appendix A for an example of a homework assignment which was typical at this level.

REFERENCES

[1] Carnahan, Brice, H. A. Luther and James Wilkes (1969), Applied Numerical Methods, John Wiley & Sons, Inc., New York.

[2] Churchman, C. W. (1968), The Systems Approach, Delacorte Press, New York.

[3] Cooper, G. R., C. D. McGillem (1967), Methods of Signal and System Analysis, Holt, Rinehart and Winston, Inc., New York.

[4] Emery, F. E. (1969), Systems Thinking, Penguin Books Ltd., Harmondsworth, Middlesex, England.

[5] Gaver, D. P. and G. Thompson (1973), Programming and Probability Models in Operation Research, Wadsworth Pub. Co., Belmont, California.

[6] I. B. M., System/360 Continuous System Modeling Program (1972), User's Manual, GH20-0367-4, White Plains, New York.

[7] Kochenburger, Ralph J. (1971), Computer Simulation of Dynamic Systems, Prentice Hall, Inc., New York.

[8] Manetch, T. J. and G. L. Parks (1973), Systems Analysis and Simulation With Application to Economic and Social Systems, Michigan State University, Preliminary Edition.

[9] Mihram, G. A. (1972), Simulation Statistical Foundations and Methodology, Academic Press, New York.

[10] Naylor, Thomas, Joseph L. Balintfy, Donald S. Burdick, Kong Chu (1968), Computer Simulation Techniques, John Wiley and Sons, Inc., New York.

APPENDIX A

This appendix presents an example of a project assignment, requiring students to simulate a simplified case of pest population dynamics under control. The problem statement is given first, followed by a student's solution in terms of a block diagram, CSMP program list and portion of its output.

Pest Control Simulation

The Problem Statement

Consider a single crop, single pest population (P(t)) interaction. The population dynamics is influenced by the following characteristics:

1. Initial infestation level P_o.
2. Natural mortality: $M_1 \cdot P(t)$

   where Mo = Pretreatment natural mortality rate

   M1 = Post 1st treatment natural mortality rate
3. Eggs lay: $E_1 \cdot P(t)$

   the emergence of adult pests occur with 20 days delay.
4. Adults migration

   the number of pests migrated into the field is a random variable normally distributed

   $$I_p = \eta(\mu,\sigma^2)$$
5. Number of pests killed due to treatment

   $$K_p = K_d(x) \cdot K_r(k) \cdot P(t)$$

   where $K_d(x) = 1 - e^{-ax\lambda}$

   $K_r(k) = f(k)$ (arbitrary function given below)

   K = 1,2,3 are days since treatment

6. Damage value is proportional to P(t) and given by

$$DMG = P(t) \cdot D \cdot \beta \text{ (per day)}$$

where D = per insect per day physical damage

β = product price

7. Cost of treatment is given by

$$COST \begin{cases} C + \alpha x & \text{if } x > 0 \\ 0 & \text{if } x = 0 \end{cases}$$

where

C = setup cost per treatment

α = pesticide purchasing price plus direct application costs.

Total accumulated cost = $N(C+\alpha x)$ where N is the number of treatments. The goal is to develop a simulation system capable of making sensitivity analysis.

Execute this assignment in two separate parts:

Part A. Feasibility study including all steps up to programming (detailed block diagram included).

Part B. After clearing your block diagram (a must), program it and run it with the following initial conditions and parameters:

$P_o = 2500.0$; P_c (population threshold for treatment) = 5000.0; $M_o = 0.125$; $M_1 = 0.100$; $E_1 = 0.15$; $\mu = 150.0$; $a = 75.0$; $a = 1.044$; $\lambda = 1.025$; $C = 3.0$; $\beta = 0.7$; $\alpha = 1.5$; $x = 3.8$; $f(1) = 0.7$; $f(2) = 0.5$; $f(3) = 0.3$.

let the final time = 100; DELT = 1.0; OUTDEL = 1.0

Print at least the variables P(t) and total cost. Print-plot the variables P(t) and K_p. Method of integration RECT. Maximize (approximately) the profit by parameterization of the P_c and x variables. Make more of your own assumptions if needed.

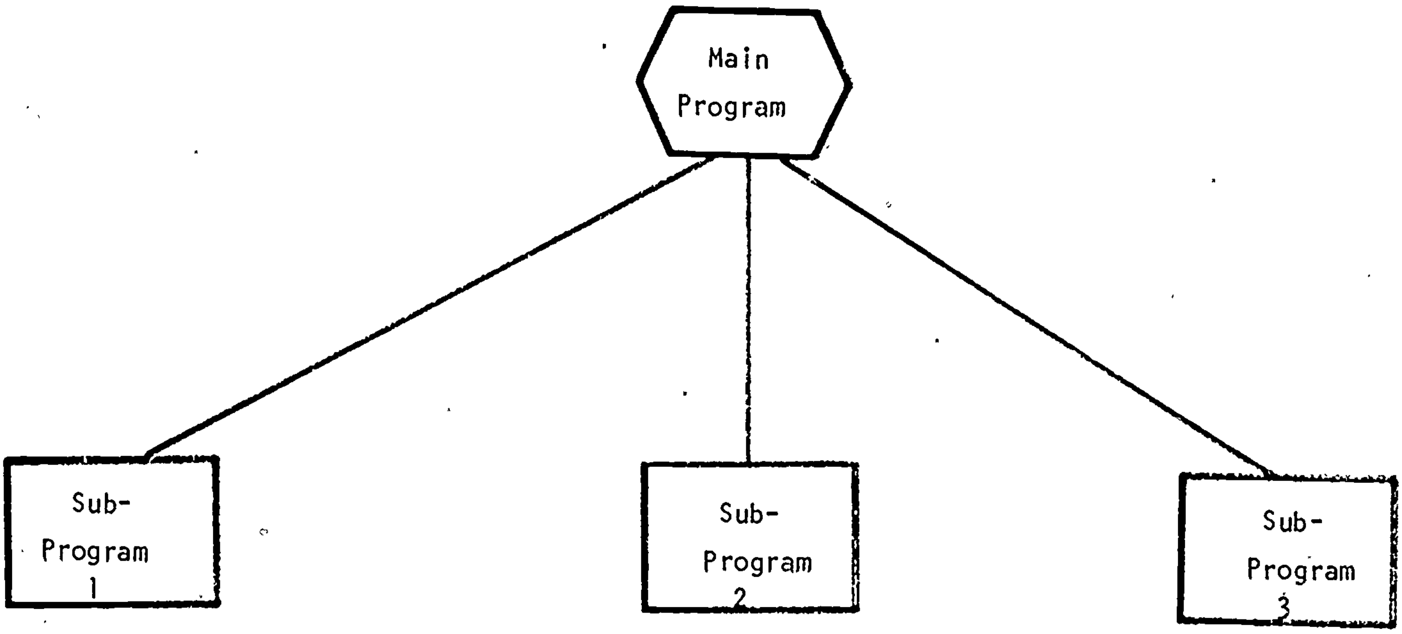


Figure 1.

```
****CONTINUOUS SYSTEM MODELING PROGRAM****

          *** VERSION 1.3 ***

TITLE   P E S T   C O N T R O L   S I M U L A T I O N
*
**********************************************************************************
INITIAL
**********************************************************************************
*
PARAMETER PC=2500.0,MC=.125,M1=.1,PC=5000.0,X=(1.,4.,8. ).E1=.15
CONSTANT B=.7,D=.001,C=3.0,S=1.5,A=1.044,W=1.025
*
**********************************************************************************
DYNAMIC
**********************************************************************************
*
PT=INTGRL(PO,RR)
RR=EL-KP-YY-KPDD-KPD+IP
YY=PT*Y
Y=FCNSW(L,MO,MO,M1)
Y1=COMPAR(PT,PC)
KP=Y1*KD*.7*PT
Y2=X**W
KD=1.0-EXP(-A*Y2)
KPD1=Y1*KD*.5*PT
KPD=DELAY(1,1.0,KPD1)
KPD2=Y1*.3*KD*PT
KPDD=DELAY(2,2.0,KPD2)
E2=(E1)*(PT)
EL=DELAY(20,20.0,E2)
LL=(C+S*X)*Y1*2.0
L=INTGRL(0.0,LL)
ZZ=PT*D*B
Z=INTGRL(0.0,ZZ)
AC=Z+L
IP=GAUSS(1,150.,75.)
*
**********************************************************************************
TERMINAL
**********************************************************************************
TIMER FINTIM=100.0, DELT=1.,OUTDEL=2.0
*
PRINT  E2 , KD,KP
PRTPLT  PT,AC , EL ,  LL , ZZ , KPD2 , KPDD , RR , L , IP , KP
METHOD TRAPZ
END
STOP
```

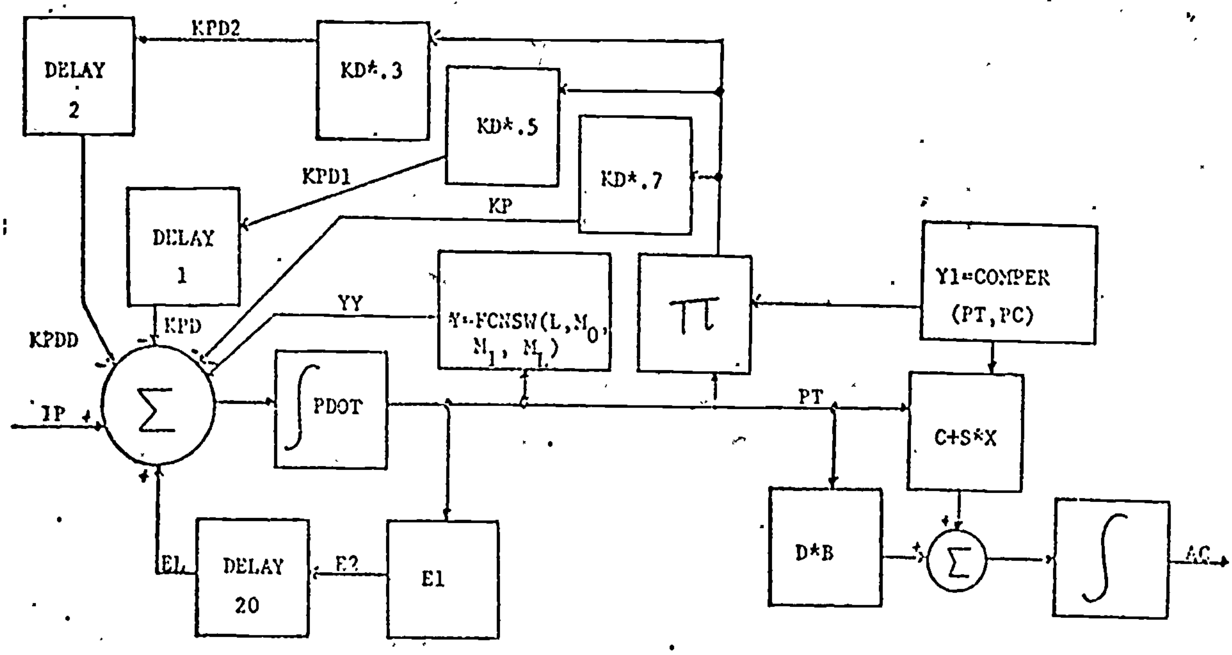


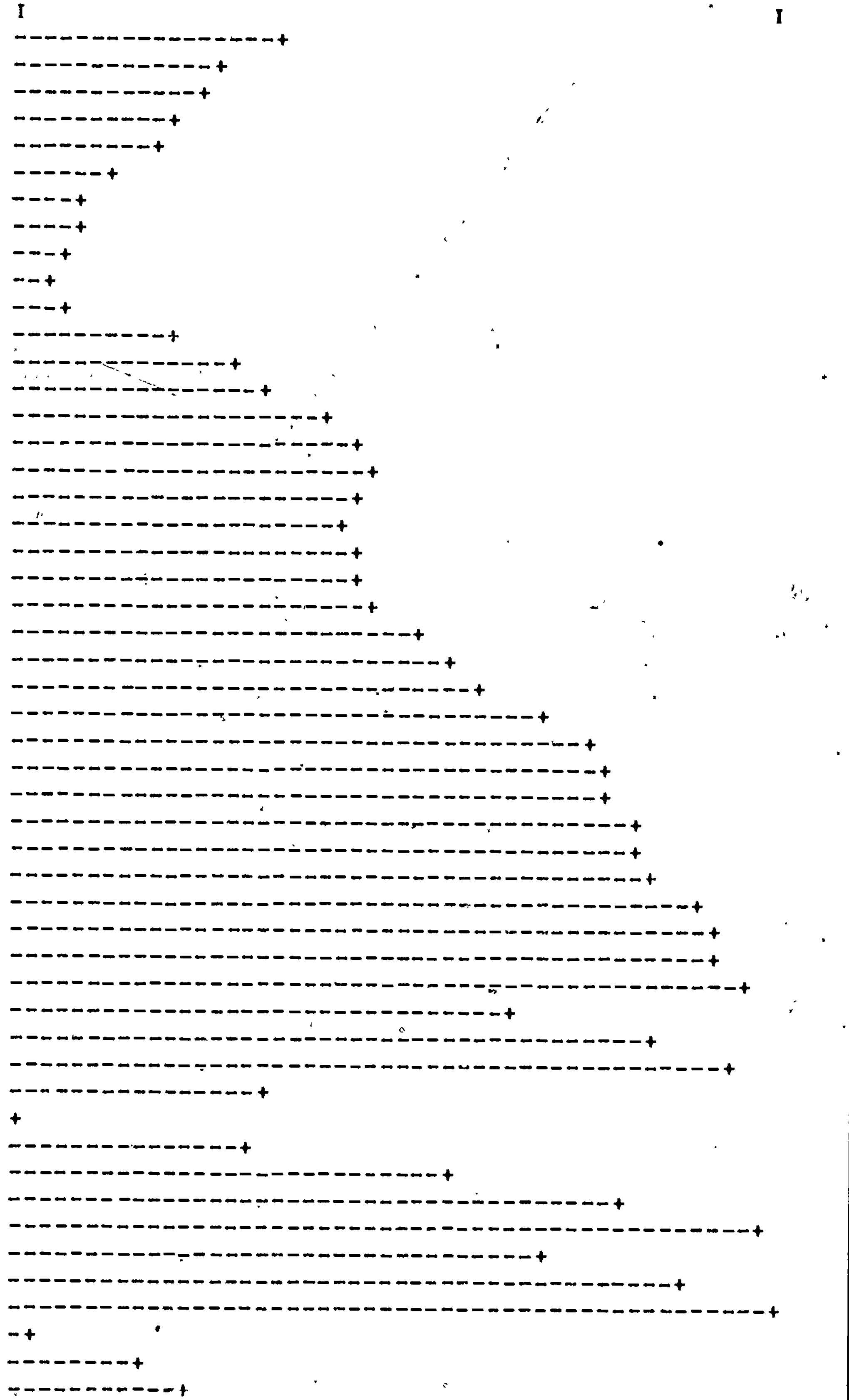
Figure 2. The block diagram solution

PROBLEM DURATION 0.0 TO 1.0000E 02

VARIABLE	MINIMUM	TIME	MAXIMUM	TIME
PT	1.2721E 03	1.9000E 01	5.0629E 03	9.4000E 01
AC	0.0	0.0	2.5603E 02	1.0000E 02
EL	0.0	0.0	7.5493E 02	9.9000E 01
LL	0.0	0.0	9.0000E 00	7.9000E 01
ZZ	8.9050E-01	1.9000E 01	3.5440E 00	9.4000E 01
KPD2	0.0	0.0	9.8416E 02	9.4000E 01
KPDD	0.0	0.0	9.8416E 02	9.6000E 01
RR	-1.8796E 03	9.4000E 01	5.4541E 02	9.9000E 01
L	0.0	0.0	2.7000E 01	9.5000E 01
IP	-9.2499E 01	0.0	2.8040E 02	7.9000E 01
KP	0.0	0.0	2.2964E 03	9.4000E 01

MINIMUM 1.0865E 03 PT VERSUS TIME X = 8.0000E 00 MAXIMUM 5.0629E 03

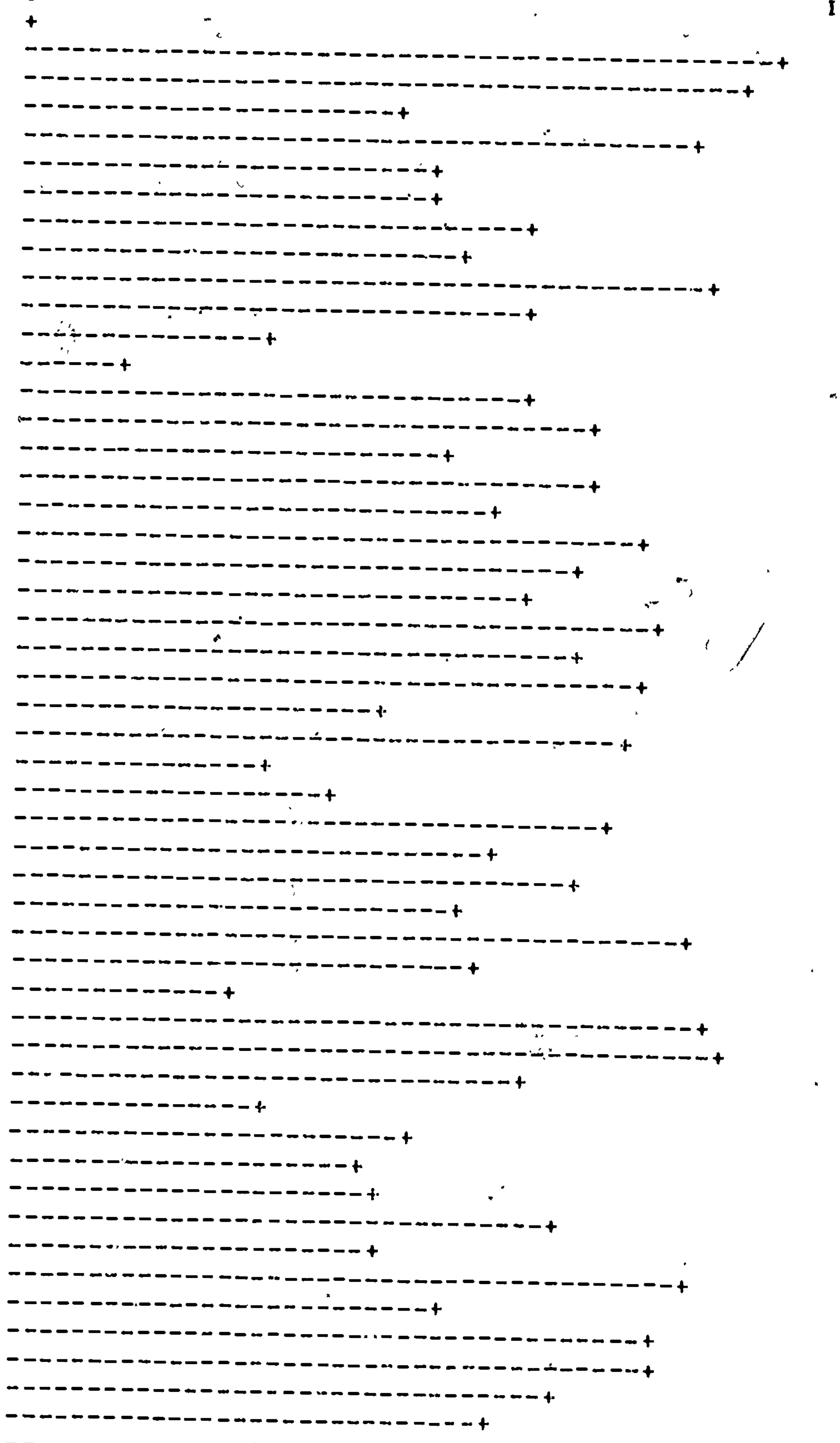
TIME	PT
0.0	2.5000E 03
2.0000E 00	2.1694E 03
4.0000E 00	2.0827E 03
6.0000E 00	1.9649E 03
8.0000E 00	1.8202E 03
1.0000E 01	1.6053E 03
1.2000E 01	1.4600E 03
1.4000E 01	1.4596E 03
1.6000E 01	1.3471E 03
1.8000E 01	1.3037E 03
2.0000E 01	1.4034E 03
2.2000E 01	1.9606E 03
2.4000E 01	2.2341E 03
2.6000E 01	2.4328E 03
2.8000E 01	2.7468E 03
3.0000E 01	2.8863E 03
3.2000E 01	2.9484E 03
3.4000E 01	2.8649E 03
3.6000E 01	2.7792E 03
3.8000E 01	2.8861E 03
4.0000E 01	2.8563E 03
4.2000E 01	2.9542E 03
4.4000E 01	3.1735E 03
4.6000E 01	3.3233E 03
4.8000E 01	3.5423E 03
5.0000E 01	3.8335E 03
5.2000E 01	4.0523E 03
5.4000E 01	4.1107E 03
5.6000E 01	4.1609E 03
5.8000E 01	4.2766E 03
6.0000E 01	4.3250E 03
6.2000E 01	4.3736E 03
6.4000E 01	4.6126E 03
6.6000E 01	4.6715E 03
6.8000E 01	4.7245E 03
7.0000E 01	4.8977E 03
7.2000E 01	3.6744E 03
7.4000E 01	4.3666E 03
7.6000E 01	4.7862E 03
7.8000E 01	2.4084E 03
8.0000E 01	1.0865E 03
8.2000E 01	2.2867E 03
8.4000E 01	3.3653E 03
8.6000E 01	4.2104E 03
8.8000E 01	4.9052E 03
9.0000E 01	3.8180E 03
9.2000E 01	4.5284E 03
9.4000E 01	5.0500E 03
9.6000E 01	1.1814E 03
9.8000E 01	1.7237E 03
1.0000E 02	1.9977E 03



```
                                  MINIMUM                  AC      VERSUS TIME                  MAXIMUM
                                   0.0                     X       = 1.0000E 00               2.7392E 02
 TIME             AC                I                                                             I
 0.0              0.0               +
 2.0000E 00       3.1303E 00        +
 4.0000E 00       6.1054E 00        -+
 6.0000E 00       8.9478E 00        -+
 8.0000E 00       1.1503E 01        --+
 1.0000E 01       1.3933E 01        --+
 1.2000E 01       1.6072E 01        --+
 1.4000E 01       1.8104E 01        ---+
 1.6000E 01       2.0072E 01        ---+
 1.8000E 01       2.1885E 01        ---+
 2.0000E 01       2.3670E 01        ----+
 2.2000E 01       2.6115E 01        ----+
 2.4000E 01       2.9097E 01        -----+
 2.6000E 01       3.2315E 01        -----+
 2.8000E 01       3.5953E 01        ------+
 3.0000E 01       3.9939E 01        -------+
 3.2000E 01       4.4012E 01        --------+
 3.4000E 01       4.8091E 01        --------+
 3.6000E 01       5.2016E 01        ---------+
 3.8000E 01       5.6002E 01        ----------+
 4.0000E 01       6.0022E 01        ----------+
 4.2000E 01       6.4041E 01        -----------+
 4.4000E 01       6.8339E 01        ------------+
 4.6000E 01       7.2870E 01        -------------+
 4.8000E 01       7.7712E 01        --------------+
 5.0000E 01       8.2841E 01        ---------------+
 5.2000E 01       8.8431E 01        ----------------+
 5.4000E 01       9.4145E 01        -----------------+
 5.6000E 01       9.9895E 01        ------------------+
 5.8000E 01       1.0582E 02        -------------------+
 6.0000E 01       1.1183E 02        --------------------+
 6.2000E 01       1.1794E 02        ---------------------+
 6.4000E 01       1.2419E 02        ----------------------+
 6.6000E 01       1.3072E 02        -----------------------+
 6.8000E 01       1.3733E 02        -------------------------+
 7.0000E 01       1.4398E 02        --------------------------+
 7.2000E 01       1.5479E 02        ----------------------------+
 7.4000E 01       1.6119E 02        -----------------------------+
 7.6000E 01       1.7197E 02        -------------------------------+
 7.8000E 01       1.7813E 02        --------------------------------+
 8.0000E 01       1.8887E 02        ----------------------------------+
 8.2000E 01       1.9258E 02        -----------------------------------+
 8.4000E 01       1.9707E 02        -----------------------------------+
 8.6000E 01       2.0286E 02        --------------------------------------+
 8.8000E 01       2.1406E 02        ----------------------------------------+
 9.0000E 01       2.2024E 02        -----------------------------------------+
 9.2000E 01       2.3112E 02        -------------------------------------------+
 9.4000E 01       2.3772E 02        --------------------------------------------+
 9.6000E 01       2.4711E 02        ----------------------------------------------+
 9.8000E 01       2.5083E 02        ----------------------------------------------+
 1.0000E 02       2.5603E 02        -----------------------------------------------+
```

MINIMUM -9.2499E 01 | IP VERSUS TIME | X = 1.0000E 00 | MAXIMUM 2.8040E 02

TIME	IP
0.0	-9.2499E 01
2.0000E 00	2.5345E 02
4.0000E 00	2.4296E 02
6.0000E 00	7.9815E 01
8.0000E 00	2.1826E 02
1.0000E 01	9.7427E 01
1.2000E 01	1.0066E 02
1.4000E 01	1.3995E 02
1.6000E 01	1.1524E 02
1.8000E 01	2.2386E 02
2.0000E 01	1.4487E 02
2.2000E 01	2.3454E 01
2.4000E 01	-4.4735E 01
2.6000E 01	1.4583E 02
2.8000E 01	1.6993E 02
3.0000E 01	1.0592E 02
3.2000E 01	1.7018E 02
3.4000E 01	1.2643E 02
3.6000E 01	1.9516E 02
3.8000E 01	1.6292E 02
4.0000E 01	1.4181E 02
4.2000E 01	2.0374E 02
4.4000E 01	1.6490E 02
4.6000E 01	1.9506E 02
4.8000E 01	7.7000E 01
5.0000E 01	1.9085E 02
5.2000E 01	2.3495E 01
5.4000E 01	5.2914E 01
5.6000E 01	1.8258E 02
5.8000E 01	1.2585E 02
6.0000E 01	1.6528E 02
6.2000E 01	1.1207E 02
6.4000E 01	2.1540E 02
6.6000E 01	1.2181E 02
6.8000E 01	9.5806E 00
7.0000E 01	2.2311E 02
7.2000E 01	2.3228E 02
7.4000E 01	1.4183E 02
7.6000E 01	2.5742E 01
7.8000E 01	8.6624E 01
8.0000E 01	6.5055E 01
8.2000E 01	7.3984E 01
8.4000E 01	1.5810E 02
8.6000E 01	7.8198E 01
8.8000E 01	2.2057E 02
9.0000E 01	1.0637E 02
9.2000E 01	2.0098E 02
9.4000E 01	1.9842E 02
9.6000E 01	1.5566E 02
9.8000E 01	1.2707E 02
1.0000E 02	2.3737E 01



ERIC Full Text Provided by ERIC