Econometric estimation and simulation using auto-patch system

de Bruijn, M.A.; Sylwestrowicz, J.D.

subfaculteit der econometrie

REEKS "TER DISCUSSIE"

[redacted]

Katholieke
Hogeschool
Tilburg

KATHOLIEKE HOGESCHOOL TILBURG

Reeks "Ter Discussie"

No. 79.96 December 1979

ECONOMETRIC ESTIMATION AND SIMULATION USING AUTO-PATCH SYSTEM

M.A. de Bruijn
J.D. Sylwestrowicz

FACULTEIT DER ECONOMISCHE WETENSCHAPPEN
SUBFACULTEIT DER ECONOMETRIE

M.A. de Bruijn[1)]
J.D. Sylwestrowicz[2)]

ECONOMETRIC ESTIMATION AND SIMULATION USING AUTO-PATCH SYSTEM[3)]

0. INTRODUCTION

Several hybrid implementations of continuous-time econometric systems were discussed in [2] and [3].
One of the disadvantages of analog programming is that the structure of the system has to be manually created by patching the relevant electronic circuits. Recently, at Delft University of Technology an auto-patch system has been developed and implemented, see [1]. The patching is replaced by a digitally programmable definition of the analog circuit, which is automatically created by high speed electronic switch matrices. A brief description of the auto-patch facility is given in the next section.

A small system, defined in section 3 has been recently implemented on the auto-patch installation. It is suitable for continuous-time estimation and simulation. Its application is illustrated on an example related to the model described in [3].

1. AUTO-PATCH SYSTEM

The Delft hybrid auto-patch system is implemented on the hybrid installation AD4-PDP 11/45 of the Delft University computing centre. The hardware of the system consists of a parallel processor (AD4) connected via an interface to the sequential processor PDP 11/45. For the programming of the parallel processor a simple set of statements and a compiler (SCALP) have been developed. For hybrid communication a library of subroutines is available.

The parallel processor has an arithmetical part and a boolean control part. The arithmetical elements can perform basic mathematical operations like integration with respect to time, summation, multiplication by a constant or by a variable, division, etc. All elements can operate simultaneously (parallel).

---

1) T.H. Delft, Hybrid Computing Centre,

2) K.H.Tilburg, Department of Econometrics,

3) Research connected with K.H.T.-pool-project "Hybrid Simulation of Economic Models".

They are controlled by boolean control components, which can perform, in parallel as well, basic logic functions like 'AND' and 'OR' operations and memory functions like a flag (flip-flop), timer or counter. For linkage of the arithmetical with the boolean elements, comparators and switches are present.

Figure 1 shows a global sketch of the present system:

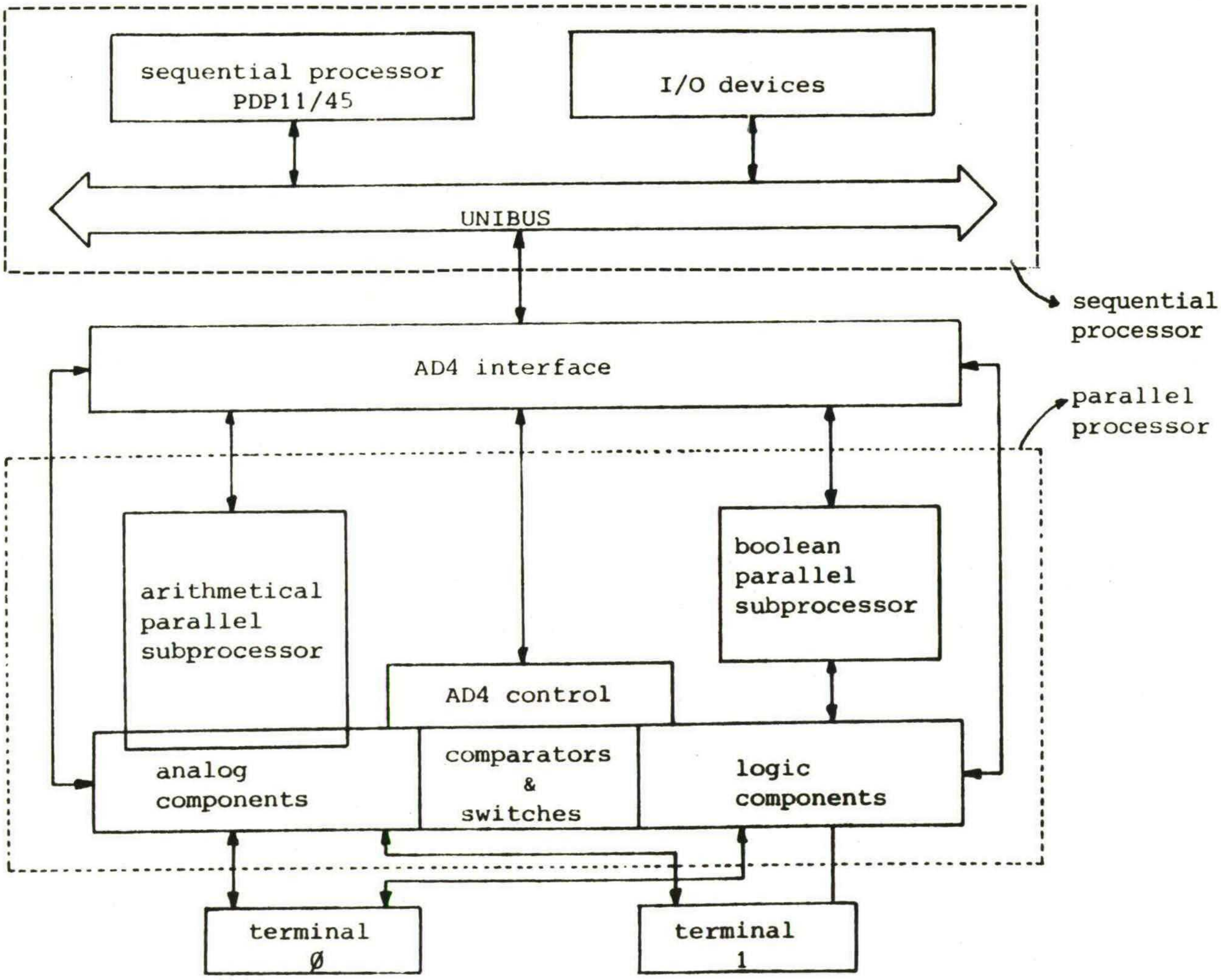

Figure 1: Present hybrid automatically programmable system.

2. PROGRAMMING THE PARALLEL PROCESSOR

Each hybrid program consists of two parts:

- The parallel part consists of the arithmetical and boolean circuits. Figure 2a depicts the flow by which a parallel load module can be obtained. Note that in certain circumstances the necessity of diagrams does not exist: the mathematical formulas can be translated directly into parallel statements.
- The sequential program coordinates all organizing, communication and sequential calculating activities. Figure 2b shows the way in which a sequential load module can be generated.

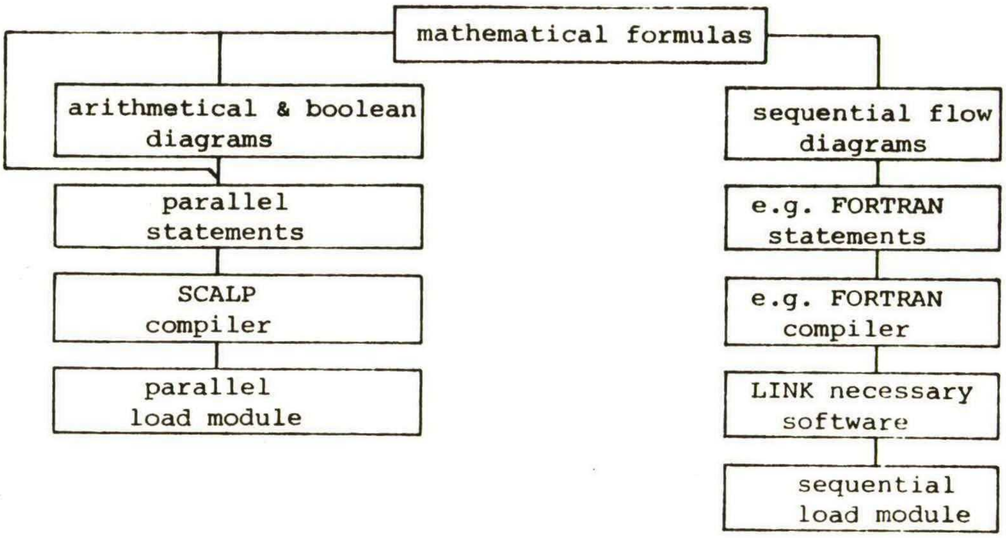

Figure 2a: Parallel programming.

Figure 2b: Sequential programming

Using the parallel statements together with the SCALP compiler [see Zegwaard,4] introduces a significant simplification in parallel programming: the parallel and sequential parts of the program can now be treated in identical ways. From figure 2a it follows that the arithmetical and boolean diagrams must translated into parallel statements. Each statement should contain all the necessary information for a complete description of one block in the diagram (block-oriented statements).
The general form of a parallel statement is

```
Xij=NAME(TYPE,inputs,parameters,control,expressions);
```

- Xij identifies the block to be programmed. The part ij is called the address, and consists of 2 or 3 digits. *All addresses are octal numbers.*
- NAME specifies roughly the character of the block, while TYPE determines the functional characteristics more in detail.
- All blocks whose outputs are to be connected with the input(s) of block Xij must be specified.
- Parametrical information can be specified (e.g. the value of a coefficient or the number of events to be counted).
- The way in which the block is to be controlled must be stated (e.g. the boolean control input of a switch).
- By means of the expressions certain operations on the inputs can be specified (e.g. a boolean expression at the SET-input of a flip-flop).

2.1 Arithmetical Parallel Subprocessor

Figure 3 shows the layout of the present arithmetical subprocessor.

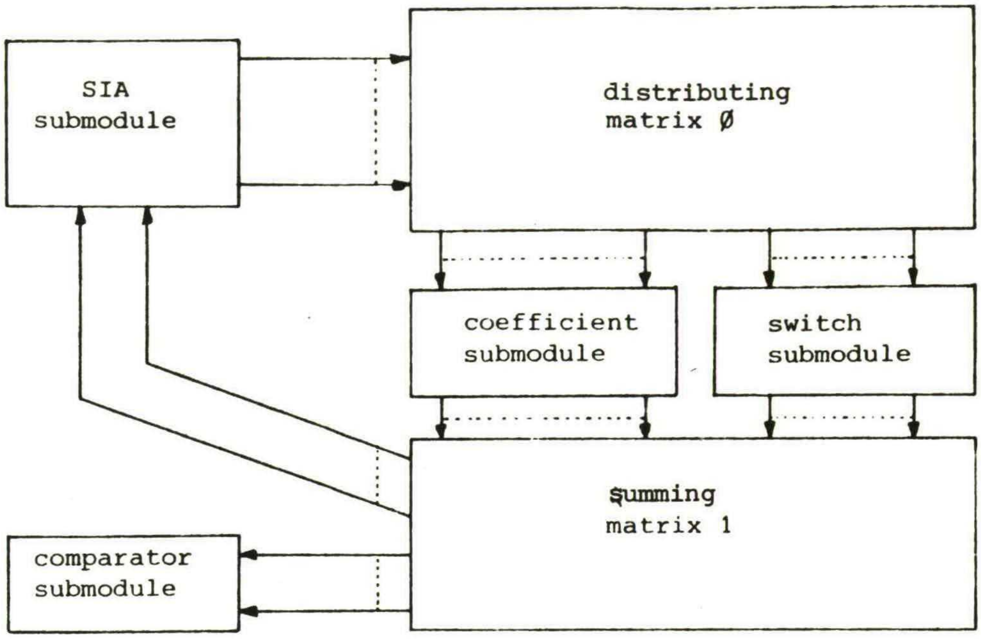

Figure 3: Present arithmetical parallel subprocessor.

Both matrices contain electronic switches which can be controlled by the sequential processor and assure complete interconnectability between the main arithmetical components. System design allows each output of matrix 1 to be the weighed sum of any number of matrix Ø inputs. There is one restriction however: each connection by means of a coefficient or switch can be used only once. There are no restrictions with respect to the number of times a matrix Ø input is used.

At the moment the SIA submodule contains 8 individually programmable integrator/ summer components. On the outputs the non-linear operations square/square root or multiplication /division can be applied. The coefficient submodule contains 12 digitally settable coefficients. The switch module has 4 switches, which are controlled from the boolean parallel processor. The comparator module contains 8 comparators. Each comparator can defect the sign of a linear combination of variables and send that information to the boolean parallel processor.

2.2 Boolean Parallel Subprocessor

In figure 4 the boolean parallel subprocessor is depicted.

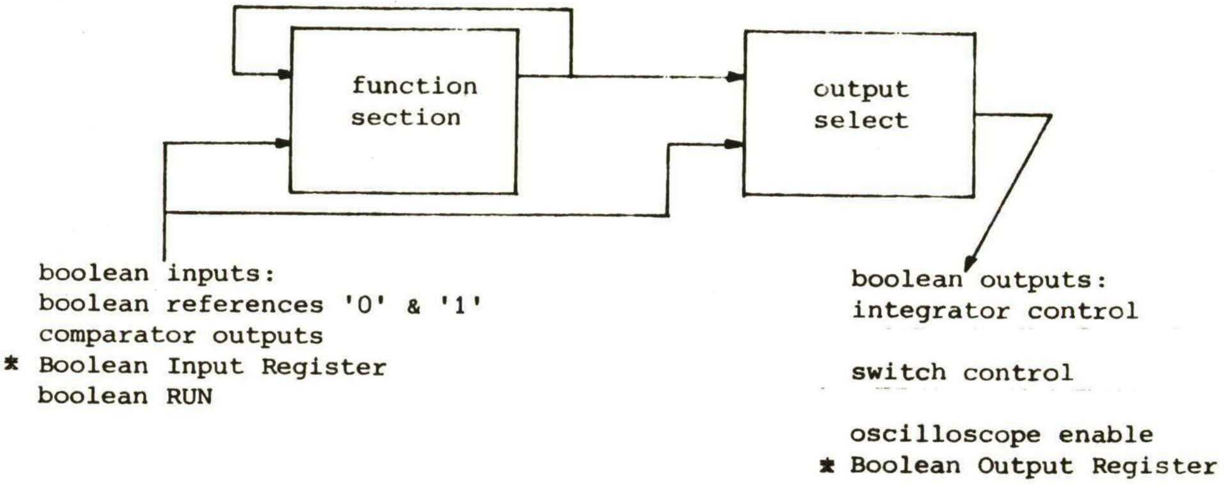

Figure 4 : Present boolean parallel subprocessor.

At this time the function section contains 6 boolean submodules which can be identified as Lij, ij being the address (ij=1Ø,11,...,15).
The boolean submodules can be programmed to operate as combinative or sequential (flip-flop, counter, timer) elements.
Three different modes can be activated for the boolean subprocessor: LOAD, STOP and RUN. In LOAD mode all sequential elements read their initial values, while in STOP mode the outputs keep their last attained values. In RUN mode the state of a sequential element is determined not only by its active inputs, but also by history (memory function). Combinative elements are mode independent: their output is always determined by their active inputs.

SCALP compiler

When the parallel program has been completed it can be punched into cards or created via the editor as a file on the disk of the sequential processor. This parallel source program can be transformed into a parallel load module by means of a compiler, called SCALP. The load module containing all necessary structural information for the parallel processor. After the translation phase it can be implemented on the parallel processor. In order to facilitate programming an extensive set of messages is created upon detection of any error during compilation.

HYBRID software

A library of hybrid software allows hybrid access of the parallel processor.
The available software can be arranged into three groups:

- implementation software, which allows on-line implementation of parallel load modules under sequential program control,
- terminal software for the communication between the sequential program and the hybrid terminals,

---

* for comminication with digital computer.

- hybrid communication software, dedicated to the transfer of data/control to and from the parallel processor. Examples are: for communication with the arithmetical part, setting of coefficients and reading of integrator outputs; for the communication with the boolean control part: setting of the boolean input register, and reading of the boolean output register.

HYBRID terminals

Once the hybrid program is running it can be controlled from:

- the console of the sequential processor. Numerical in- and output can be done and messages can be written on the video-terminal,
- one of the hybrid terminals. Numerical input can be read from the thumbwheels and control input can be obtained from the push buttons. The thumbwheels and pushbuttons are read into sequential variables so allowing to control manually the flow of the sequential program and the values of parameters. Graphical output of arithmetical parallel data can be displayed on the oscilloscope of the terminal.

3. STRUCTURE OF THE SYSTEM, DATA REPRESENTATION, ESTIMATION PROCEDURE

In the present implementation the following set of equations can be solved:

$$y(t) = \alpha_1 z_{\lambda_1}(t) + \alpha_2 z_{\lambda_2}(t) + \alpha_3 z_{\lambda_3}(t) + \alpha_4$$

$$\frac{dz_{\lambda_i}}{dt} = \lambda_i \, (z_i - z_{\lambda_i}) \quad , \quad i = 1,2,3.$$

where $y(t)$ is an endogenous variable, z_i are exogenous variables, z_{λ_i} are exponentially delayed exogenous variables (the length of the delay is given by $1/\lambda_i$).

It is assumed that the data for the variables are given in discrete form. The continuous representation for the analog calculations is automatically created. In the present version it is done by linear interpolation between the data points, as illustrated in fig. 5.

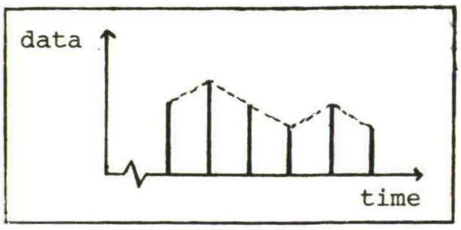

Figure 5: Reconstruction of exogenous data.

During the digital part of the estimation process we have to compare the values of the analog solution $y_i(t)$ with the data y_{ik}. Therefore the analog variables are sampled and the values $y_i(\delta k)$, where $k = 0,\ldots T$; and δ is the length of the discrete time period, are transmited to the digital computer.

As in [3], the estimation problem is defined in terms of non-linear least squares. θ^* is a solution of the estimation problem if it minimizes:

$$S(\theta) = \sum_{k=0}^{T} (y_i(\delta k)-y_{ik})^2 \qquad (3.1)$$

In this implementation θ can be defined as any subset of:

$$\alpha_1, \alpha_2, \alpha_3, \alpha_4, \lambda_1, \lambda_2, \lambda_3$$

The optimization procedure described in [3] is used to find the minimum of (3.1)

4. FEATURES OF THE IMPLEMENTATION

During a typical session the following activities can be undertaken:

1. Selecting variables from the data base: Specified variables are made operational for the system. The initial scaling is performed automatically, however, if necessary the user can alter the scaling at any stage of the computations. The equation specification is recorded on the line printer. A simulation run starts with the parameter values specified on the thumbwheels.
2. Display of data and the variables. At this stage, the user has the opportunity to observe on the memoscope the data, variables and the delayed variables. A hard copy of the memoscope picture can be made at any stage of the computations.
3. Manual optimization: By changing manually the parameter values, specified on the thumbwheels, and observing the reaction of the endogenous variable on the memoscope the user can attempt to improve the fit of the model. The function value and the best (so far) set of parameters can be printed on the line-printer.
4. Automatic optimization. By pressing a pushbutton it is possible to activate an automatic estimation procedure. The user specifies a subset of parameters and their initial values and then the optimization procedure attempts to locate the minimum of S. The optimization process can also be observed on the memoscope, and if, for some reason it is not converging it can be manually interrupted.
   After optimization the values of estimated parameters are recorded on the line-printer.
5. Change of specification: At any stage of the computations the user can change the specification of the equation, by eliminating and/or introducing variables into the equation or changing scaling factors.

The improvement of fit in case of price of exports equation (endogenous - pe; exogenous - w, v-a, pm) for details and notation, see [3], is illustrated on figure 6.

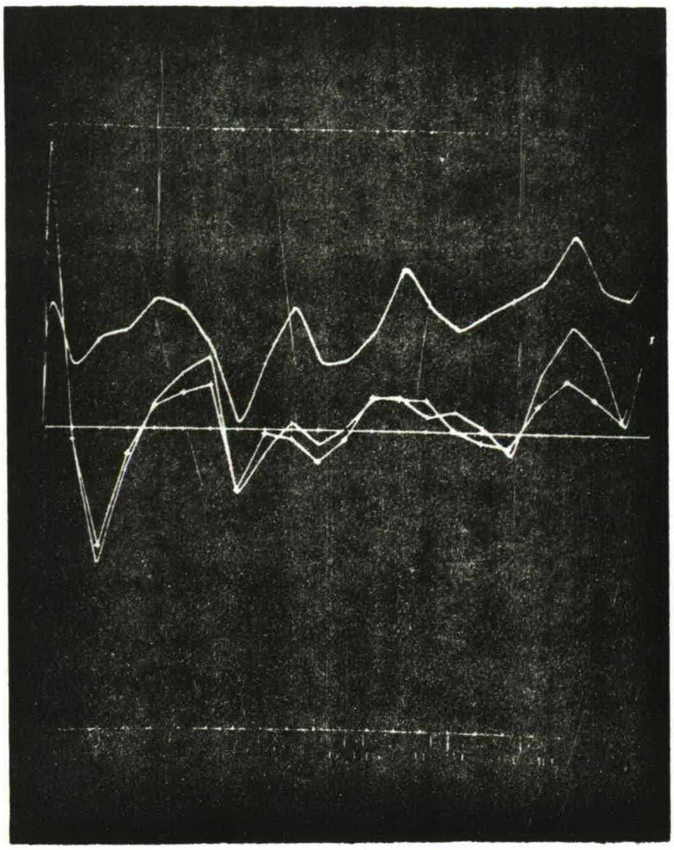

Figure 6.

Here the crosses denote discrete data, the upper curve corresponds to the starting point for optimization, the lower curve to the estimated values. The values of the parameters are given in the table below:

	α_1	α_2	α_3	α_4	λ_1	λ_2	λ_3
Start:	0.21	0.84	0.30	0.22	10.0	10.0	10.0
	-0.17	0.87	1.00	1.3	3.9	0.02	15.0

The value of S was reduced from 3.86 to 0.74.

Parallel structure diagram

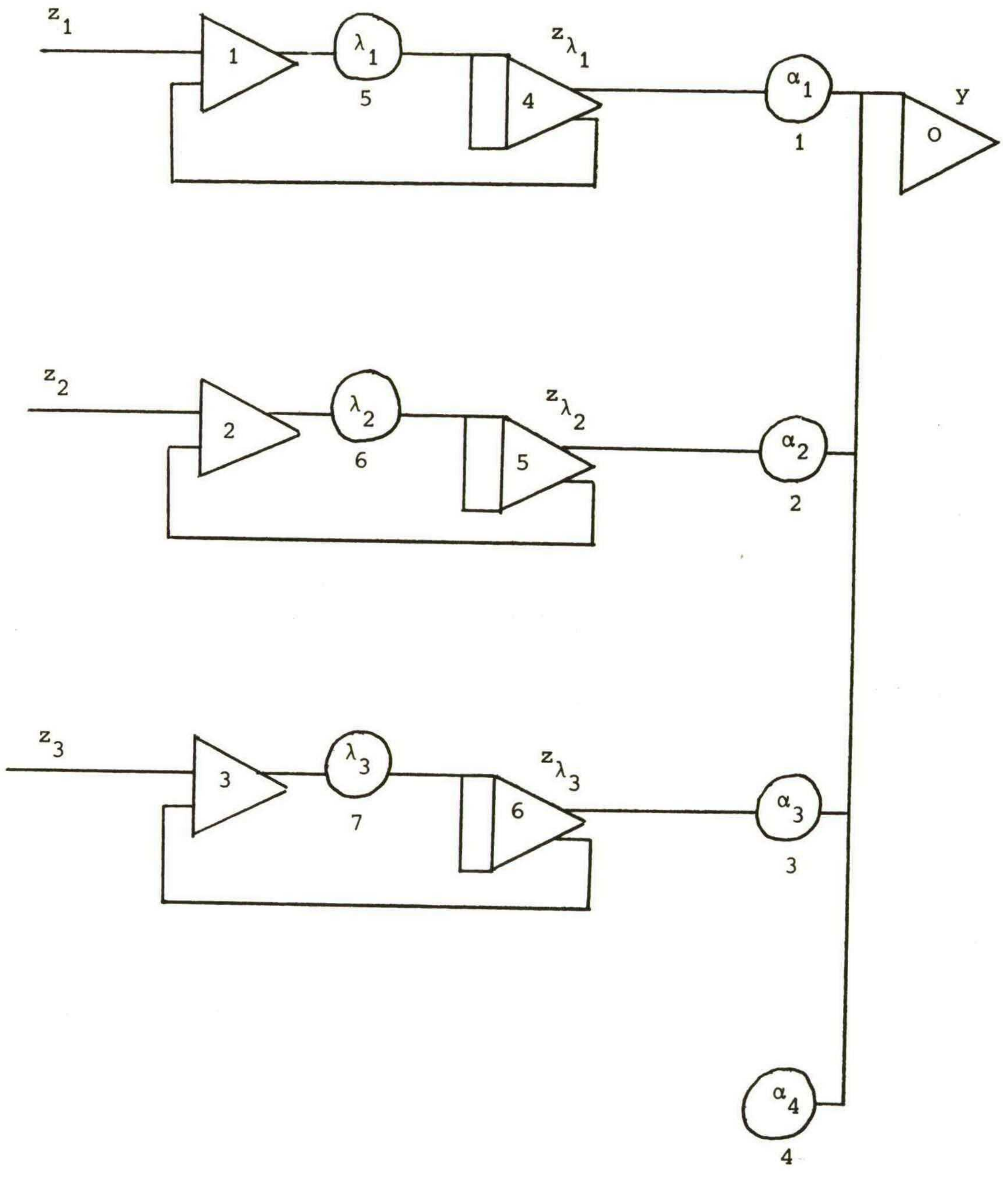

Figure 7: Definition of patching

References

[1] Brok, S.W., Kooiman, A., Llurba, R., Sips, H., Swaan Arons, H. de, 1979, "*The Delft Timeshared Hybrid Minisystem*", IMACS,Congress on Simulation of Systems, Sorrento, Italy.

[2] Sylwestrowicz, J.D., 1978, "*Application of hybrid computers in econometrics*", part I and II, Reeks "*ter discussie*" no. 78.076, Tilburg University.

[3] Sylwestrowicz, J.D., Bruijn, M.A. de, 1979, "*Hybrid computer implementation and estimation of an econometric model for the Dutch economy*", IMACS, Congress on Simulation of Systems, Sorrento, Italy.

[4] Zegwaard, J.F., 1979, "*Hybrid Computation Group Computing Centre*", Delft University of Technology.

<u>In de Reeks ter Discussie zijn verschenen</u>:

1.H.H. Tiggelaar	Spectraalanalyse en stochastische lineaire differentievergelijkingen.	juni '75
2.J.P.C.Kleijnen	De rol van simulatie in de algemene econometrie.	juni '75
3.J.J. Kriens	A stratification procedure for typical auditing problems.	juni '75
4.L.R.J. Westermann	On bounds for Eigenvalues	juni '75
5.W. van Hulst J.Th. van Lieshout	Investment/financial planning with endogenous lifetimes: a heuristic approach to mixed integer programming.	juli '75
6.M.H.C.Paardekooper	Distribution of errors among input and output variables.	augustus '75
7.J.P.C. Kleijnen	Design and analysis of simulation Practical statistical techniques.	augustus '75
8.J. Kriens	Accountantscontrole met behulp van steekproeven.	september '75
9.L.R.J. Westermann	A note on the regula falsi	september '75
10.B.C.J. van Velthoven	Analoge simulatie van ekonomische modellen.	november '75
11.J.P.C. Kleijnen	Het ekonomisch nut van nauwkeurige informatie: simulatie van ondernemingsbeslissingen en informatie.	november '75
12.F.J. Vandamme	Theory change, incompatibility and non-deductibility.	december '75
13.A. van Schaik	De arbeidswaardeleer onderbouwd?	januari '76
14.J.vanLieshout J.Ritzen J.Roemen	Input-ouputanalyse en gelaagde planning.	februari '76
15.J.P.C.Kleijnen	Robustness of multiple ranking procedures: a Monte Carlo experiment illustrating design and analysis techniques.	februari '76
16.J.P.C. Kleijnen	Computers and operations research: a survey.	februari '76
17.J.P.C. Kleijnen	Statistical problems in the simulation of computer systems.	april '76
18.F.J. Vandamme	Towards a more natural deontic logic.	mei '76
19.J.P.C. Kleijnen	Design and analysis of simulation: practical, statistical techniques.	juni '76
20.H.H. Tigelaar	Identifiability in models with lagged variables.	juli '76
21.J.P.C. Kleijnen	Quantile estimation in regenerative simulation: a case study.	augustus '76
22.W.Derks	Inleiding tot econometrische modellen van landen van de E.E.G.	augustus '76
23.B. Diederen Th. Reijs W. Derks	Econometrisch model van België.	september '76
24.J.P.C. Kleijnen	Principles of Economics for computers.	augustus '76
25.B. van Velthoven	Hybriede simulatie van ekonomische modellen.	augustus '76.

26.F. Cole	Forecasting by exponential smoothing, the Box and Jenkins procedure and spectral analysis. A simulation study.	september '76
27.R. Heuts	Some reformulations and extensions in the univariate Box-Jenkins time series analysis.	juli '76
28.W. Derks	Vier econometrische modellen.	
29.J. Frijns	Estimation methods for multivariate dynamic models.	oktober '76
30.P. Meulendijks	Keynesiaanse theorieën van handelsliberalisatie.	oktober '76
31.W. Derks	Structuuranalyse van econometrische modellen met behulp van Grafentheorie. Deel I: inleiding in de Grafentheorie.	september '76
32.W. Derks	Structuuranalyse van econometrische modellen met behulp van Grafentheorie. Deel II: Formule van Mason.	oktober '76
33. A. van Schaik	Een direct verband tussen economische veroudering en bezettingsgraadverliezen.	september '76
34. W. Derks	Structuuranalyse van Econometrische Modellen met behulp van Grafentheorie. Deel III. De graaf van dynamische modellen met één vertraging.	oktober '76
35. W. Derks	Structuuranalyse van Econometrische Modellen met behulp van Grafentheorie. Deel IV. Formule van Mason en dynamische modellen met één vertraging.	oktober '76
36. J. Roemen	De ontwikkeling van de omvangsverdeling in de levensmiddelenindustrie in de D.D.R.	oktober '76
37. W. Derks	Structuuranalyse van Econometrische modellen met behulp van grafentheorie. Deel V. De graaf van dynamische modellen met meerdere vertragingen.	oktober '76
38. A. van Schaik	Een direkt verband tussen economische veroudering en bezettingsgraadverliezen. Deel II: gevoeligheidsanalyse.	december '76
39. W. Derks	Structuuranalyse van Econometrische modellen met behulp van Grafentheorie. Deel VI. Model I van Klein, statisch.	december '76
40. J. Kleijnen	Information Economics: Inleiding en kritiek	november '76
41. M. v.d. Tillaart.	De spectrale representatie van mutivariate zwak-stationaire stochastische processen met discrete tijdparameter.	november '76
42. W. Groenendaal Th. Dunnewijk	Een econometrisch model van Engeland	december '76
43. R. Heuts	Capital market models for portfolio selection	september '76

44. J. Kleijnen en P. Rens	A critical analysis of IBM's inventory package impact.	december '76
45. J. Kleijnen en P. Rens	Computerized inventory management: A critical analysis of IBM's impact system.	december '76
46. A. Willemstein	Evaluatie en foutenanalyse van econometrische modellen. Deel I. Een identificatie methode voor een lineair discreet systeem met storingen op input, output en structuur.	januari '77
47. W. Derks	Structuuranalyse van econometrische modellen met behulp van grafentheorie. Deel VII. Model I van Klein, dynamisch.	februari '77
48. L. Westermann	On systems of linear inequalities over $\mathbb{R}^n$.	februari '77
49. W. Derks	Structuuranalyse van econometrische modellen met behulp van Grafentheorie. Deel VIII. Klein-Goldberger model.	februari '77
50. W.v. Groenendaal en Th. Dunnewijk	Een econometrisch model van het Verenigd Koninkrijk	februari '77
51. J. Kleijnen en P. Rens	A critical analysis of IBM's inventory package "IMPACT"	februari '77
52. J.J.A. Moors	Estimation in truncated parameterspaces	maart '77
53. R.M.J. Heuts	Dynamic transfer function-noise modelling (Some theoretical considerations)	december '76
54. B.B. v.d. Genugten	Limit theorems for LS-estimators in linear regression models with independent errors.	mei '77
55. P.A. Verheyen	Economische interpretatie in modellen betreffende levensduur van kapitaalgoederen	juni '77
56. W.v.den Bogaard en J.Kleijnen	Minimizing wasting times using priority classes	juni '77
57. W. Derks	Structuuranalyse van Econometrische Modellen met behulp van Grafentheorie. Deel IX. Model van landen van de E.E.G.	juni '77
58. R. Heuts	Capital market models for portfolio selection (a revised version)	juni '77
59. A.P. Willemstein	Evaluatie en foutenanalyse van econometrische modellen. Deel II. Het Model I van L.R. Klein.	aug. '77
60. Th. Dunnewijk W. van Groenendaal	An econometric Model of the Federal Republic of Germany 1953-1973	aug. '77
61. A. Plaisier A. Hempenius	Slagen of zakken. Een intern rapport over de studieresultaten propedeuse-economie 1974/1975	aug. '77

62.	A. Hempenius	Over een maat voor de juistheid van voorspellingen	aug. '77
63.	R.M.J. Heuts	Some reformulations and extensions in the univariate Box-Jenkins time series analysis approach (a revised version)	sept. '77
64.	R.M.J. Heuts	Applications of univariate time series modelling of U.S. monetary and business indicator data	sept. '77
65.	A. Hempenius en J. Frijns	Soorten van prijsheteroskedasticiteit in marktvraagfunkties.	okt. '77
66.	H.H. Tigelaar	Identifiability in Multiple Time Series	okt. '77
67.	J. van Lieshout en P. Verheyen	Levensduur in een jaargangenmodel	nov. '77
68.	Pieter J.F.G. Meulendijks	"De macro-economische betekenis van geïnduceerde technische ontwikkeling; een meer-sektoren model met jaargangentheorie"	nov. '77
69.	R.M.J. Heuts en P.J. Rens	A Monte Carlo study to obtain the percentage points of some goodness of fit tests in testing normality, when observations satisfy a certain low order ARMA-scheme	dec. '77
70.	J.C.P. Kleijnen	Generalizing Simulation Results through Metamodels	dec. '77
71.	Th. van de Klundert	Winstmaximalisatie in het Jaargangenmodel met vaste technische coëfficiënten; een inventarisatie van de problematiek	dec. '77
72.	B.B. van der Genugten	A central limit theorem with applications in regression analysis	jan. '78
73.	A.P. Willemstein	Evaluatie en foutenanalyse van econometrische modellen. DEEL III Stochastische fluctuaties op de parameters en heteroscedasticiteit in een lineair model	jan. '78
74.	Pieter J.F.G.Meulendijks	Some reflections on macro-economic planning, policy and development in the Netherlands: 1918-1978	apr. '78
75.	J.H.J. Roemen	Overeenkomsten en verschillen in uitgangspunten tussen lognormale en Pareto-verdeling.	juli '78

:

76. J.D. Sylwestrowicz	Applications of Hybrid Computers in Econometrics - Part I.	juli '78
77. Pieter J.F.G. Meulendijks	On a disequilibrium analysis of the labour market. Review of and comments upon R.S.G. Lenderink and J.C. Siebrand, A disequilibrium analysis of the labour market, Rotterdam University Press, '76.	aug. '78
78. B.B. van der Genugten	Asymptotic normality of 2SLS-estimators in simultaneous equation systems.	aug. '78
79. J.M.G. Frijns	Imperfect competition of the labour market(s) and the adjustment of factor inputs.	okt. '78
80. H.H. Tigelaar	The identifiability problem in dynamic simultaneous equations with moving average errors.	nov. '78
81. Jack P.C. Kleijnen	Bayesian information economics: an evaluation.	dec. '78.
82. J.M.G. Frijns	Instrumental Variable Estimators Applied on Pooled Time Series Cross Section Models.	maart '79
83. M.H.C. Paardekooper	Inverse perturbations for approximations of least squares solutions.	maart '79
84. J.H.J. Roemen	Kostenverbijzondering op basis van een input-output-model voor een situatie met vaste en variabele kosten.	maart '79
85. J.H.J. Roemen	Een schattingsprocedure op basis van macro data voor de (stationaire) overgangswaarschijnlijkheden in een Markovmodel ter beschrijving van de ontwikkeling van de omvangsverdeling van ondernemingen.	april '79
86. G.J. de Nooij Th. M.M. Verhallen	Marketing mix sensitivity.	april '79
87. J. Kriens	Statistical Sampling in Auditing	april '79
88. J.D. Sylwestrowicz	Applications of hybrid computers in econometrics, Part II. An implementation of a medium size model for the Dutch economy.	mei '79

89. Bob Kaper	Comparative Statics on nonlinear input-output models.	juli '79
90. W.J. Oomens	Vertragingsverdelingen voor het overloopeffect van marketing-uitgaven.	Aug. '79
91. B.B. v.d. Genugten	Central limit theorems for the OLS-estimator in linear regression models including lagged variables.	Sept. '79
92. J.J.A. Moors	Momentproblems and truncated binomial estimation	Sept. '79
93. H.H. Tigelaar	A general approach to identification, predictability and the problem of minimal informative sample size.	Sept. '79
94. G.J. de Nooij J. de Schouwer	Positionering en segmentatie in de popmuziekmarkt.	Nov. '79
95. Jack P.C. Kleijnen	Analysis of simulation with common random numbers: a note on Heikes et al. (1976). version 2.	Nov. '79

```
C-----------------------------------------------------------------------
C               HYBRID ESTIMATION OF ONE EQUATION ECONOMETRIC MODELS
C               MAIN PROGRAM
C
C               MICHIEL DE BRUIJN, RC, TH DELFT
C               JERZY SYLWESTROWICZ, KH TILBURG
C
C               IMPLEMENTATION ON DELFT HYBRID AUTOPATCH SYSTEM
C               INTERACTIVE EXPERIMENTING FACILITY VIA HYBRID TERMINAL #0
C
C               19-OCT-79, FILE SYLECO.FOR
C               PARALLEL PROGRAM IN FILE SYLECO.SRC
C
C               SUBROUTINE CALLS:
C                       DR417=DATA READING
C                       OS417=WRITE FRAME ON MEMOSCOPE
C                       FU417=FUNCTION EVALUATION
C                       GR417=WRITE GRAPH ON MEMOSCOPE
C
C               HYBRID COMMUNICATION LIBRARY: HLIB.OBJ
C                       ACCESS     FINDFL     TERMNL     SAWTIM
C                       FRESH                 THUMBW     COSAWS
C                       SPEED                 IPUSHB     COSAWC
C                       SETUPD
C                       GAIN                  GRAPH      TBSTIM
C
C-----------------------------------PUSHBUTTON FUNCTIONS
C     0 - OPTIMIZATION
C     1 - GRAPHICAL DISCRETE DISPLAY
C         THUMBWHEEL 0 DETERMINES VARIABLES
C         0 - ENDOGENOUS
C         1 - EXOGENOUS 1
C         2 - EXOGENOUS 2
C         3 - EXOGENOUS 3
C     2 - LINEAR INTERPOLATION
C     3 - SHORTENING DELAYS
C     4 - INTERMEDIATE OPTIMIZATION RESULTS
C     5 - GO OUT OF OPTIMIZATION PROCEDURE
C     6 - GO TO NEW SPECIFICATION
C     7 - PRINT CURRENT RESULTS
C-----------------------------------------------------------------------
C      ONE EQUATION SYSTEM, AN AUTO-PATCH IMPLEMENTATION
C-----------------------------------------------------------------------
      DIMENSION VAR(30,40),SFC(40),XNAM(40),PAR(10),RPAR(10)
      DIMENSION ENDO(30),EX01(30),EX02(30),EX03(30),SEN(30)
      DIMENSION NPAX(10)
      DIMENSION POP(10)
      DIMENSION IDELAY(3)
      DIMENSION IS0(3),IS1(3)
      DIMENSION XX(10)
      DIMENSION IPATCH(685)
      DATA IS0/0,0,0/
      DATA IS1/1,1,1/
      DATA IDELAY/"04,"05,"06/
      COMMON ENDO,EX01,EX02,EX03,SEN
      COMMON NPAX,MRES,I1,J1,NSTAR,NFIN
C-----------------------------------INITIALIZATION
      CALL ASSIGN(2,'SYLDAT.DAT')
      CALL ASSIGN(16,'MS1:')
      CALL ACCESS
      CALL FRESH
      CALL FINDFL(IPATCH,'SYLECO')
      CALL IMPLDF(IPATCH)
      CALL GRAPH(0)
```

```
      CALL TERMNL(0)
      CALL SETUPD("036,1)
      CALL SETUPD("040,1)
      CALL SAWTIM(1.)                        !----TIMEBASE
      CALL COSAWS("00,1)                     !----    "
      CALL COSAWC("036)                      !----    "
      CALL COSAWC("040)                      !----    "
      CALL COSAWC("042)                      !----    "
      I1=0
      J1=0
        LCAR=2                               !----I/O
        LREAD=2
        LMEMO=16
        LPRIN=6
        LCONR=5
      FP=0.0
      FN=0.0
        LCONW=7
        MEM=16
        IPUS=0
        IMODE=0
        ISIZE=70
      DO 500 I=1,30                          !----DATA ARRAYS
         DO 500 J=1,40
            VAR(I,J)=0.0
  500 CONTINUE
      NP=7
      NSUC=0
      EE=10000.0
      READ(LREAD,1003) IDAY,IWEEK,ILAT,IDISP
      ILAT=ILAT+1900
      WRITE(LPRIN,9377) IDAY,IWEEK,ILAT
        REWIND LPRIN
 9377 FORMAT(1H1,//,' EXPERIMENTS PERFORMED ON=    'I2,'.',I2,'.',I4,//)
      READ(LREAD,1003) NDAT,IT,NENDO,NEXO,NSTAR,NFIN
      PERIOD=NFIN-NSTAR
      PERIOD=22.0
      CALL TBSTIM(PERIOD)
      NYMAX=NFIN
      NST=NSTAR
 1003 FORMAT(20I4)
 1014 FORMAT(20A4)
      NVAR=NENDO+NEXO
      NEQ=NENDO
C--------------------------------------READ DATA
      CALL DR417(NDAT,IT,VAR)
      READ(LREAD,1014) (XNAM(J), J=1,40)
C--------------------------------------DETERMINING SCALING FACTORS
      DO 400 J=1,NDAT
         AMAX=ABS(VAR(NSTAR-1,J))
         DO 401 I=NSTAR,NFIN
            ABV=ABS(VAR(I,J))
            IF(AMAX-ABV) 450,401,401
  450       AMAX=ABV
  401    CONTINUE
         SFC(J)=AMAX
  400 CONTINUE
      DO 402 J=1,NDAT
         SCAL=0.0
         SF=SFC(J)
  451    SCAL=SCAL + 5.0
         IF(SCAL-SF) 451,452,452
  452    SFC(J)=SCAL
  402 CONTINUE
      SFC(12)=55.0
```

3)

```
  102 CONTINUE
      DO 499 I=1,10
  499    NPAX(I)=I
C-------------------------------------TERMINAL COMMUNICATION
      PAUSE ' CHOOSE VARIABLES'
      READ(LCONR,1003) IEND,IEX1,IEX2,IEX3
      WRITE(LCONW,1014) XNAM(IEND),XNAM(IEX1),XNAM(IEX2)
     1,                 XNAM(IEX3)
      WRITE(LCONW,1003)IEND,IEX1,IEX2,IEX3
      WRITE(LCONW,1004)SFC(IEND),SFC(IEX1),SFC(IEX2)
     1,                  SFC(IEX3)
C
      PAUSE ' RESCALE VARIABLE'
      READ(LCONR,1003) KSC,ISCA
      WRITE(LCONW,1003)KSC,ISCA
      IF(KSC.EQ.0) GO TO 530
      SFC(KSC)=ISCA
  530 CONTINUE
      WRITE(LPRIN,1015) XNAM(IEND),XNAM(IEX1),XNAM(IEX2)
     1,                 XNAM(IEX3)
      REWIND LPRIN
 1004 FORMAT(1H ,10F4.1)
 1015 FORMAT(' ENDOGENOUS:',A4/' EXOGENOUS:',3A4)
C-------------------------------------SCALING OF ARRAYS
      SFE=SFC(IEND)
      SF1=SFC(IEX1)/SFE
      SF2=SFC(IEX2)/SFE
      SF3=SFC(IEX3)/SFE
      SF4=1.0/SFE
      DO 501 I=NSTAR,NFIN
         ENDO(I)=VAR(I,IEND)/SFE
         EXO1(I)=VAR(I,IEX1)/SFC(IEX1)
         EXO2(I)=VAR(I,IEX2)/SFC(IEX2)
         EXO3(I)=VAR(I,IEX3)/SFC(IEX3)
  501 CONTINUE
C-------------------------------------READ THUMBWHEELS
      DO 502 I=1,7
  502    PAR(I)=THUMBW(I)
  101 CONTINUE
      CALL OS417(23)                    !----WRITE FRAME ON MEMOSCOPE
      GAI=1.
      CALL GAIN(IDELAY,IS0,3)
      IF(IPUSHB(3).EQ.0) GO TO 111
      GAI=10.
      CALL GAIN(IDELAY,IS1,3)
 111  CONTINUE
            RPAR(1)=PAR(1)/SF1
C-------------------------------------PARAMETER RECALCULATION
      RPAR(2)=PAR(2)/SF2
      RPAR(3)=PAR(3)/SF3
      RPAR(4)=PAR(4)/SF4
      RPAR(5)=PAR(5)*GAI
      RPAR(6)=PAR(6)*GAI
      RPAR(7)=PAR(7)*GAI
C-------------------------------------INITIALLY ONE FUNCTION EVALUATION
      CALL FU417(NP,PAR,FV)
      WRITE(LPRIN,1001) NSUC,FV,FN
      WRITE(LPRIN,1002) (PAR(I),I=1,7)
      WRITE(LPRIN,1002) (RPAR(I),I=1,7)
      REWIND LPRIN
      IX=-10000
      IY=-11000
      JX=-9500
      JY=-10500
      CALL GRAPH(1)
```

```
      WRITE(MEM,1007)IMODE,ISIZE,IX,IY,(PAR(I),I=1,7)
      REWIND MEM
      CALL GRAPH(0)
 1001 FORMAT(1H ,I5,'    F=',2F12.3)
 1006 FORMAT(2I2,2I6,'  P1      P2      P3      P4      D1      D2      D3')
 1007 FORMAT(2I2,2I6,7F7.3)
 1002 FORMAT(1H ,7F7.3)
      FP=FV
C-------------------------------------DYNAMIC SECTION
  100 CONTINUE
      NP=7
      DO 510 I=1,7
  510    PAR(I)=THUMBW(I)
      CALL FU417(NP,PAR,FN)
      IF(FN.GT.FP) GO TO 520
      FP=FN
      DO 511 I=1,7
  511    POP(I)=PAR(I)
      NSUC=NSUC+1
  520 CONTINUE
C-------------------------------------DIGITAL DISPLAY ON MEMOSCOPE
      IF(IPUSHB(1).EQ.0) GO TO 521
      CALL GR417(3,IT,VAR,SFC,IEND,IEX1,IEX2,IEX3,XNAM, IDAY,IWEEK,
     1          ILAT)
  521 CONTINUE
C
      IF(IPUSHB(7).EQ.IPUS) GO TO 522
      IPUS=IPUSHB(7)
      DO 512 I=1,7
  512    PAR(I)=POP(I)
      GO TO 101
  522 CONTINUE
C
      IF(IPUSHB(6).EQ.0) GO TO 110
      GO TO 102
  110 CONTINUE
C
      IF(IPUSHB(0).EQ.0) GO TO 100
      DO 125 I=1,7
  125    XX(I)=PAR(I)
C-------------------------------------OPTIMIZATION
      PAUSE ' READ NR. OF PARAMETERS FOR OPTIMIZATION'
      READ(LCONR,1019) NPAR
 1019 FORMAT(I1)
      PAUSE ' READ INDEXIES AND PARAMETER VALUES'
      DO 121 I=1,NPAR
         READ(LCONR,1016) NPAX(I),XX(I)
         IF(NPAX(1).EQ.0) GO TO 124
  121 CONTINUE
  124 IF(NPAX(1).NE.0) GO TO 127
      NPAX(1)=1
      XX(1)=PAR(1)
  127 CONTINUE
 1016 FORMAT(I1,F10.2)
      NTEST=10
      CALL COPTI(NPAR,XX,NTEST,FM,VARI,NFUN,NSUC,FVAL)      !----OPT. ROUTINE
      DO 122 I=1,NPAR
         KK=NPAX(I)
         PAR(KK)=XX(I)
  122 CONTINUE
      IF(FVAL.GE.FP) GO TO 101
      FP=FVAL
      DO 123 I=1,7
  123    POP(I)=PAR(I)
      DO 498 I=1,10
```

```
498 CONTINUE
    GO TO 101
    END
```

(5)

```
C-----------------------------------------------------------------------
C                   HYBRID ESTIMATION OF ONE EQUATION ECONOMETRIC MODELS
C                   SUBROUTINE FOR DATA READING
C
C                   MICHIEL DE BRUIJN, RC, TH DELFT
C                   JERZY SYLWESTROWICZ
C
C                   16-OCT-79, FILE SYLDRD.FOR
C
C                   N=NUMBER OF DATA VECTORS (OUTPUT)
C                   IT=NUMBER OF YEARS (OUTPUT)
C                   X=DATA MATRIX (OUTPUT)
C-----------------------------------------------------------------------
      SUBROUTINE DR417(N,IT,X)
             DIMENSION X(30,40)
      LREAD=2
      LPRIN=3
      DO 1 I=1,N
         READ(LREAD,1000) (X(J,I),J=1,IT)
    1 CONTINUE
 1000 FORMAT(9F8.0)
 1001 FORMAT(1H ,I5,10F10.3)
 1002 FORMAT(1H1,///,' DATA SET')
 1003 FORMAT(1H0,////)
C-------------------------------------DATA TRANSFORMATIONS
      DO 6 J=1,IT
         X(J,25)=0.171*X(J,31)-0.074*X(J,25)
         X(J,27)=0.067*X(J,27)-0.008*X(J,32)
    6 CONTINUE
      RETURN
      END
```

```
C-----------------------------------------------------------------------
C                 HYBRID ESTIMATION OF ONE EQUATION ECONOMETRIC MODELS
C                 SUBROUTINE FOR WRITING A FRAME ON THE MEMOSCOPE
C
C                 MICHIEL DE BRUIJN, RC, TH DELFT
C                 JERZY SYLWESTROWICZ
C
C                 15-OCT-79, FILE SYLMOS.FOR
C
C                 NYMAX=MAX NUMBER OF YEARS OBSERVED
C
C                 HYBRID COMMUNICATION ROUTINES
C                        GRAPH
C                        LINE
C-----------------------------------------------------------------------
      SUBROUTINE OS417(NYMAX)
C-------------------------------------DEFINITION OF FRAME COORDINATES
      IFX=0
      IFY=0
      IDFX=10000
      IDFY=10000
      IDDY=100
      ISHY=IDDY/2.0
      IFMXX=IFX+IDFX
      IFMNX=IFX-IDFX
      IFMXY=IFY+IDFY
      IFMNY=IFY-IDFY
      IDDX=FLOAT((IFMXX-IFMNX))/(NYMAX-1)
C-------------------------------------WRITE FRAME
      CALL GRAPH(1)
      CALL LINE(IFMNX,IFMXY,IFMXX,IFMXY)
      CALL LINE(IFMNX,IFY  ,IFMXX,IFY  )
      CALL LINE(IFMNX,IFMNY,IFMXX,IFMNY)
      CALL LINE(IFMNX,IFMNY,IFMNX,IFMXY)
      CALL LINE(IFMXX,IFMNY,IFMXX,IFMXY)
      NYEAR=NYMAX-2
      IY=IFMXY
      DO 600 I=1,3
         IX=IFMNX
         DO 500 IYEAR=1,NYEAR
            IX=IX+IDDX
            IY2=IY-IDDY
  500       CALL LINE(IX,IY,IX,IY2)
  600    IY=IY-IDFY+ISHY
      CALL GRAPH(0)
      RETURN
      END
```

```
C----------------------------------------------------------------------------
C               HYBRID ESTIMATION OF ONE EQUATION ECONOMETRIC MODELS
C               SUBROUTINE FOR FUNCTION EVALUATION
C
C               MICHIEL DE BRUIJN, RC, TH DELFT
C               JERZY SYLWESTROWICZ
C
C               16-OCT-79, FILE SYLFUN.FOR
C
C               NP=NUMBER OF PARAMETERS (INPUT)
C               PAR=PARAMETER VECTOR (INPUT)
C               FN=VALUE OF FUNCTION (OUTPUT)
C
C               SUBROUTINE CALLS:
C                       RS417=RESETTING OF PARAMETERS
C
C               HYBRID COMMUNICATION ROUTINES
C                       SETCOF
C                       SYSRUN
C                       SYSLD
C                       QUIET
C                       SAMPLE
C                       SETBIT
C                       BORWT
C                       SAW
C                       TBSE
C----------------------------------------------------------------------------
      SUBROUTINE FU417(NP,PAR,FN)
      DIMENSION PAR(10)
      DIMENSION ENDO(30),EXO1(30),EXO2(30),EXO3(30),SEN(30)
      DIMENSION NPAX(10)
      COMMON ENDO,EXO1,EXO2,EXO3,SEN
      COMMON NPAX,MRES,I1,J1,NSTAR,NFIN
C------------------------------------HYBRID DYNAMIC SECTION
      CALL RS417(NP,PAR)
C------------------------------------PREPARATION
      NT=NSTAR
      CALL SETCOF("036,EXO1(NT+1))
      CALL SETCOF("037,EXO1(NT))
      CALL SETCOF("040,EXO2(NT+1))
      CALL SETCOF("041,EXO2(NT))
      CALL SETCOF("042,EXO3(NT+1))
      CALL SETCOF("043,EXO3(NT))
      CALL SYSRUN
      CALL SYSLD
      CALL QUIET(10)
C------------------------------------SAMPLE AND PLAY-BACK
      CALL SAW(1)
      CALL TBSE(1)
      CALL SYSRUN
  210 CONTINUE
      CALL SAMPLE("00,SEN(NT),1)
      CALL SETBIT(0,1)
      CALL SETBIT(0,0)
      NT=NT+1
      IF(NT.GT.NFIN) GO TO 200
      CALL SETCOF("036,EXO1(NT+1))
      CALL SETCOF("037,EXO1(NT))
      CALL SETCOF("040,EXO2(NT+1))
      CALL SETCOF("041,EXO2(NT))
      CALL SETCOF("042,EXO3(NT+1))
      CALL SETCOF("043,EXO3(NT))
      CALL BORWT(0)
      GO TO 210
```

```
  200 CALL SYSEQ
      CALL SAW(Ø)
      CALL TBSE(Ø)
C---------------------------------SUM OF SQUARES
      SSE=Ø.Ø
      DO 579 I=NSTAR,NFIN
  579 SSE=SSE + (ENDO(I) - SEN(I))**2
      FN=SSE
      RETURN
      END
```

```
C---------------------------------------------------------------------------
C                 HYBRID ESTIMATION OF ONE EQUATION ECONOMETRIC MODELS
C                 SUBROUTINE FOR RESETTING OF PARAMETERS
C
C                 MICHIEL DE BRUIJN, RC, TH DELFT
C                 JERZY SYLWESTROWICZ
C
C                 17-OCT-79, FILE SYLRES.FOR
C
C                 N=NUMBER OF PARAMETERS TO BE CHANGED
C                 PAX=VALUES OF PARAMETERS
C         COMMON  NPAX=INDEX OF THE PARAMETER
C         COMMON  MRES=ERROR ARGUMENT
C
C                 HYBRID COMMUNICATION ROUTINE
C                       SETCOF
C---------------------------------------------------------------------------
      SUBROUTINE RS417(N,PAX)
      DIMENSION PAX(10),NPAX(10)
      DIMENSION ENDO(30),EXO1(30),EXO2(30),EXO3(30),SEN(30)
C
      COMMON ENDO,EXO1,EXO2,EXO3,SEN
C
      COMMON NPAX,MRES,I1,J1,NSTAR,NFIN
      MRES=0
      DO 10 I=1,N
         K=NPAX(I)
         VAL=PAX(I)
         IF(K.LT.5) GO TO 14
         IF(VAL.LT.0.) GO TO 11
 14      CONTINUE
         IF(VAL.LT.-1.5) GO TO 11
         IF(VAL.GT.1.5) GO TO 12
         GO TO(1,2,3,4,5,6,7),K
  1      CALL SETCOF("030,VAL)
         GO TO 10
  2      CALL SETCOF("031,VAL)
         GO TO 10
  3      CALL SETCOF("032,VAL)
         GO TO 10
  4      CALL SETCOF("010,VAL)
         GO TO 10
  5      CALL SETCOF("033,VAL)
         GO TO 10
  6      CALL SETCOF("034,VAL)
         GO TO 10
  7      CALL SETCOF("035,VAL)
 10   CONTINUE
      RETURN
 11   MRES=-K                               !----NEGATIVE BOUNDARY REACHED
      RETURN
 12   MRES=K                                !----POSITIVE BOUNDARY REACHED
      RETURN
      END
```

```
C-----------------------------------------------------------------------
C               HYBRID ESTIMATION OF ONE EQUATION ECONOMETRIC MODELS
C               SUBROUTINE FOR WRITING A GRAPH OF A VARIABLE ON THE MEMOSCOPE
C
C               MICHIEL DE BRUIJN, RC, TH DELFT
C               JERZY SYLWESTROWICZ
C
C               16-OCT-79, FILE SYLGRA.FOR
C
C               NS=FIRST YEAR
C               IT=LAST YEAR
C               VAR=VARIABLE TO BE DISPLAYED
C               SFC=SCALING FACTOR
C               IEND=POINTER TO ENDOGENOUS VARIABLE
C               IEX1=POINTER TO FIRST EXOGENOUS VARIABLE
C               IEX2=POINTER TO SECOND EXOGENOUS VARIABLE
C               IEX3=POINTER TO THIRD EXOGENOUS VARIABLE
C               XNAM=
C               IDAY=
C               IWEEK=
C               ILAT=
C
C               SUBROUTINE CALLS:
C                       CR417=WRITE CROSS ON MEMOSCOPE
C
C               HYBRID COMMUNICATION ROUTINES
C                       GRAPH
C                       THUMBW
C                       LINE
C-----------------------------------------------------------------------
      SUBROUTINE GR417(NS,IT,VAR,SFC,IEND,IEX1,IEX2,IEX3,XNAM,IDAY,
     1                 IWEEK,ILAT)
      DIMENSION VAR(30,40),SFC(40)
      DIMENSION XNAM(40)
      LMEMO=16
      CALL GRAPH(1)
      NVAR=IEND
      ITHUMB=THUMBW(0)*10
      IF( ITHUMB.EQ.1) NVAR=IEX1
      IF(ITHUMB.EQ.2) NVAR=IEX2
      IF(ITHUMB.EQ.3) NVAR=IEX3
      IF(ITHUMB.EQ.0) NVAR=IEND
      ISTEP=20000.0/(IT-NS)
      FIFTA=10000.0
      IX=-10000
      IDEL=100
      DO 1 I=NS,IT
      IY=FIFTA*VAR(I,NVAR)/SFC(NVAR)
      CALL CR417(IX,IY,IDEL)
      IX=IX+ISTEP
    1 CONTINUE
      IF(IPUSHB(2).EQ.0) GO TO 11
      IY=FIFTA*VAR(NS,NVAR)/SFC(NVAR)
      NSS=NS+1
      IX=-10000
      DO 2 I=NSS,IT
      IX1=IX+ISTEP
      IY1=FIFTA*VAR(I,NVAR)/SFC(NVAR)
      CALL LINE(IX,IY,IX1,IY1)
      IX=IX1
      IY=IY1
    2 CONTINUE
   11 CONTINUE
      CALL GRAPH(0)
```

```
RETURN
END
```

```
C-------------------------------------------------------------------------
C                HYBRID ESTIMATION OF ONE EQUATION ECONOMETRIC MODELS
C                SUBROUTINE FOR WRITING A CROSS ON THE MEMOSCOPE
C
C                MICHIEL DE BRUIJN, RC, TH DELFT
C                JERZY SYLWESTROWICZ
C
C                16-OCT-79, FILE SYLCRS.FOR
C
C                IX=HORIZONTAL POSITION
C                IY=VERTICAL POSITION
C                IDEL=CROSS SIZE
C
C                HYBRID COMMUNICATION ROUTINE
C                        LINE
C-------------------------------------------------------------------------
      SUBROUTINE CR417(IX,IY,IDEL)
      IXD=IX+IDEL
      IYD=IY+IDEL
      IXM=IX-IDEL
      IYM=IY-IDEL
      CALL LINE(IXM,IY,IXD,IY)
      CALL LINE(IX,IYD,IX,IYM)
      RETURN
      END
```

```
C-----------------------------------------------------------------------------------
C               HYBRID ESTIMATION OF ONE EQUATION ECONOMETRIC MODELS
C               SUBROUTINE FOR OPTIMIZATION ALONG COORDIANTE DIRECTIONS
C
C               MICHIEL DE BRUIJN, RC, TH DELFT
C               JERZY SYLWESTROWICZ
C
C               19-OCT-79, FILE SYLOPT.FOR
C
C               NP=NUMBER OF PARAMETERS
C               PAR=PARAMETER VECTOR
C               NTEST=
C               FM=
C               VARI=
C               NFUN=NUMBER OF FUNCTION EVALUATIONS
C               NSUC=NUMBER OF SUCCESSFUL FUNCTION EVALUATIONS
C               FVAL=FUNCTION VALUE
C
C               SUBROUTINE CALLS
C                       FU417=FUNCTION EVALUATION
C                       SU417=DETERMINATION OF SUCCESS OR FAILURE
C
C               HYBRID COMMUNICATION ROUTINE
C                       IPUSHB
C-----------------------------------------------------------------------------------
      SUBROUTINE COPTI(NP,PAR,NTEST,FM,VARI,NFUN,NSUC,FVAL)
      DIMENSION PAR(1Ø)
      DIMENSION X(1Ø)
      DIMENSION ENDO(3Ø),EXO1(3Ø),EXO2(3Ø),EXO3(3Ø),SEN(3Ø)
      DIMENSION NPAX(1Ø)

      DIMENSION FVEC(5Ø),BU(1Ø)
      COMMON ENDO,EXO1,EXO2,EXO3,SEN
      COMMON NPAX,MRES,I1,J1,NSTAR,NFIN
      LCONW=7
      LPRIN=6
      STEP=.Ø1
      IMPR=Ø
      IFUN=Ø
      ISUC=Ø
      NEVAL=NTEST
  810 CONTINUE
      S=Ø.Ø
      SS=Ø.Ø
C--------------------------------APPROXIMATING VARIANCE OF A FUNCTION EVALUATIC
      DO 1 I=1,NEVAL
         CALL FU417(NP,PAR,FV)
         FVEC(I)=FV
         S=S+FV
         SS=SS + FV*FV
    1 CONTINUE
      FM=S/NEVAL
      VARI=SS/NEVAL - FM*FM
      SSS=Ø.Ø
      DO 2 I=1,NEVAL
    2    SSS=SSS + (FVEC(I) - FM)**2
      VARI=SSS/(NEVAL-1)
      SERR=SQRT(VARI)
      IF(SERR.LE.Ø.ØØØØØ1) SERR=Ø.ØØØØØ1
      WRITE(LCONW,11Ø6)SS,SERR,NEVAL,NTEST
      WRITE(LPRIN,11ØØ)
      WRITE(LPRIN,11Ø6)FM,VARI,(NPAX(I),I=1,NP)
 11Ø6 FORMAT(1H ,2F1Ø.7,7I8)
 11ØØ FORMAT(' OPTIMIZATION STARTS WITH:')
```

```
      WRITE(LPRIN,1101) FM, (PAR(I),I=1,NP)
 1101 FORMAT(' FUNCTION VALUE=' F8.3/'PARAMETERS'/7F8.3///)
      REWIND LPRIN
C---------------------------------------OPTIMIZATION ALONG COORDINATES
      FP=FM
      TAU=0.5*(1.0+SQRT(5.0))
      TAU4=1.0/TAU
      TAU3=(TAU-1.0)*TAU4
      DO 300 I=1,NP
  300    X(I)=PAR(I)
      DO 301 I=1,NP
         ALPL=0.
         ALPR=0.
         BU(I)=0.
         KK=NPAX(I)
         STEPX=STUN
C---------------------------------------DETERMINING INITIAL SEARCH INTERVAL
  304    CONTINUE
         X(I)=PAR(I)-STEPX
         CALL FU417(NP,X,F1)
         IF(IPUSHB(5).NE.0) GO TO 811
         IFUN=IFUN+1
         IF(MRES ) 302,303,302
  302    IMRE=IMRE+1
         IF(MRES.LT.0) GO TO 801
         X(I)=1.499
         GO TO 888
  801    CONTINUE
         X(I)=-1.499
         IF(KK.LT.5) GO TO 888
         X(I)=0.002
         GO TO 888
  303    CONTINUE
         CALL SU417(F1,FP,SERR,LOGI)
         IF(LOGI) 306,306,307
  306    STEPX=2.0*STEPX
         GO TO 304
  307    ALP1=-STEPX
  305    STEPX=STUN
  404    CONTINUE
         X(I)=PAR(I)+STEPX
         CALL FU417(NP,X,F2)
         IF(IPUSHB(5).NE.0) GO TO 811
         IFUN=IFUN+1
         IF(MRES ) 402,403,402
  402    IMRE=IMRE + 1
         IF(MRES.LT.0) GO TO 802
         X(I)=1.499
         GO TO 888
  802    CONTINUE
         X(I)=-1.499
         IF(KK.LT.5) GO TO 888
         X(I)=0.002
         GO TO 888
  403    CONTINUE
         CALL SU417(F2,FP,SERR,LOGI)
         IF(LOGI) 406,406,407
  406    STEPX=STEPX*2.0
         GO TO 404
  407    ALP2=STEPX
C---------------------------------------THE GOLDEN SECTION SEARCH
         ALPL=ALP1
         ALPR=ALP2
         ALP3=(ALP2-ALP1)*TAU3 + ALP1
         ALP4=(ALP2-ALP1)*TAU4 + ALP1
         X(I)=PAR(I)+ALP1
```

```
      CALL FU417(NP,X,F1)
      IF(IPUSHB(5).NE.0) GO TO 811
      X(I)=PAR(I)+ALP2
      CALL FU417(NP,X,F2)
      IF(IPUSHB(5).NE.0) GO TO 811
      X(I)=PAR(I)+ALP3
      CALL FU417(NP,X,F3)
      IF(IPUSHB(5).NE.0) GO TO 811
      X(I)=PAR(I)+ALP4
      CALL FU417(NP,X,F4)
      IF(IPUSHB(5).NE.0) GO TO 811
      IFUN=IFUN+4
420   CONTINUE
      AL21=ALP2-ALP1
      IF(AL21.LT.0.005) GO TO 422
      CALL SU417(F3,F4,SERR,LOGI)
      IF(LOGI) 421,422,423
423   CONTINUE
      ISUC=ISUC+1
      ALP1=ALP3
      F1=F3
      ALP3=ALP4
      F3=F4
      ALP4=(ALP2-ALP1)*TAU4 + ALP1
      X(I)=PAR(I)+ALP4
      CALL FU417(NP,X,F4)
      IF(IPUSHB(5).NE.0) GO TO 811
      IFUN=IFUN+1
      GO TO 420
421   CONTINUE
      ISUC=ISUC+1
      ALP4=ALP3
      F4=F3
      ALP2=ALP4
      F2=F4
      ALP3=(ALP2-ALP1)*TAU3 + ALP1
      X(I)=PAR(I)+ALP3
      CALL FU417(NP,X,F3)
      IF(IPUSHB(5).NE.0) GO TO 811
      IFUN=IFUN+1
      GO TO 420
422   CONTINUE
      ALP=(ALP1+ALP2)*0.5
      X(I)=PAR(I)+ALP
888   CONTINUE
      CALL FU417(NP,X,FP)
      IF(IPUSHB(5).NE.0) GO TO 811
      IFUN=IFUN+1
      BU(I)=(ALP2-ALP1)*0.5
      IF(IPUSHB(4).EQ.0) GO TO 805
      WRITE(LCONW,1005) I,FM,FP,X(I),ALPL,ALPR,BU(I),SERR,IFUN,ISUC,IMRE
1005  FORMAT(1H ,I3,7F7.2,3I4)
      REWIND LCONW
805   CONTINUE
      CALL SU417(FP,FM,SERP,LOGI)
      IF(LOGI) 430,431,431
430   PAR(I)=X(I)
      GO TO 301
431   FP=FM
      X(I)=PAR(I)
  301 CONTINUE
C
 1102 FORMAT(' OPTIMIZATION ENDED WITH:')
  811 CONTINUE
      WRITE (LPRIN,1102)
      WRITE(LPRIN,1101) FP,(PAR(I),I=1,NP)
```

```
REWIND LPRIN
      NFUN=IFUN
FVAL=FP
IF(IPUSHB(0).NE.0) GO TO 810
RETURN
END
```

17

```
C-----------------------------------------------------------------------
C                  HYBRID ESTIMATION OF ONE EQUATION ECONOMETRIC MODELS
C                  SUBROUTINE FOR DETERMINATION OF SUCCESS OR FAILURE
C
C                  MICHIEL DE BRUIJN, RC, TH DELFT
C                  JERZY SYLWESTROWICZ
C
C                  16-OCT-79, FILE SYLSUC.FOR
C
C                  FN=
C                  FP=
C                  SER=
C                  LOGI=
C-----------------------------------------------------------------------
      SUBROUTINE SU417(FN,FP,SER,LOGI)
      FL=FP-2.0*SER
      FH=FP+2.0*SER
      IF(FN-FL)1,2,2
    1 LOGI=-1
      RETURN
    2 IF(FN-FH) 3,3,4
    4 LOGI=1
      RETURN
    3 LOGI=0
      RETURN
      END
```

```
C-----------------------------------------------------------------------------
C               HYBRID ESTIMATION OF ONE EQUATION ECONOMETRIC MODELS
C               SUBROUTINE FOR RANDOM NUMBER GENERATION
C
C               MICHIEL DE BRUIJN, RC, TH DELFT
C               JERZY SYLWESTROWICZ
C
C               16-OCT-79, FILE SYLRAN.FOR
C
C               N=
C               X=
C-----------------------------------------------------------------------------
      SUBROUTINE RN417(N,X)
      DIMENSION X(10)
      DIMENSION ENDO(30),EXO1(30),EXO2(30),EXO3(30),SEN(30)
      DIMENSION NPAX(10)
      COMMON ENDO,EXO1,EXO2,EXO3,SEN
      COMMON NPAX,MRES,I1,J1,NSTAR,NFIN
      DO 2 K=1,N
      SS=0.0
      DO 1 I=1,12
      XR=RAN(I1,J1)
    1 SS=SS+XR
      X(K)=SS-6.0
    2 CONTINUE
      XN=0.0
      DO 3 I=1,N
    3 XN=XN + X(I)**2
      XN=SQRT(XN)
      DO 4 I=1,N
    4 X(I)=X(I)/XN
      RETURN
      END
```

```
;-----------------------------------------------------------------------------
;               HYBRID ESTIMATION OF ONE EQUATION ECONOMETRIC MODELS
;               PARALLEL PROGRAM
;
;               MICHIEL DE BRUIJN, RC, TH DELFT
;               JERZY SYLWESTROWICZ, KH, TILBURG
;
;               19-OCT-79, FILE SYLECO.SRC
;-----------------------------------------------------------------------------
      A00=SUM(IN=C030*A04 + C031*A05 + C032*A06 + C010);
;-----------------------------------EXOGENOUS VARIABLES
      A04=INT(IN=C033*A01);
      A05=INT(IN=C034*A02);
      A06=INT(IN=C035*A03);
;-----------------------------------DATA RECONSTRUCTION
      A01=SUM(IN=-S01*A04 + C036 +   C037*A07);
      A02=SUM(IN=-S02*A05 + C040 + C041*A07);
      A03=SUM(IN=-S03*A06 + C042 + C043*A07);
      A07=SUM(IN=1.0 + S00);
;-----------------------------------SAMPLE AND PLAY-BACK CONTROL
      L11=COM(A=LR1,P=S-0,BEX=A);
      L10=FFP(A=L11,B=BIR00,SET=A,CLR=B) ;
      TSH00=L10;
      BOR00=L10;
      UPC=L11;
      L12=COM(A=RUN,BEX=A);
      BOR04=L12;
      L13=FFP(A=L11,SET=A,CLR=A);
      BOR03=L13;
;-----------------------------------ALL SWITCHES CLOSED FOR 1-1 CONNECTION
      S00=LR1;
      S01=LR1;
      S02=LR1;
      S03=LR1;
```