H IN BUSINESS ECONOMICS
Erasmus Universiteit Rotterdam
Burgemeester Oudlaan 50
3062 PA Rotterdam
The Netherlands

Tel.: 010-145511

The object of the Centre for Research in Business Economics is:

- to start and coordinate research programs that involve more than one field of business economics;

- to stimulate and be of service to research projects of the individual departments centered on one field in particular.

The centre publishes a series of workingpapers, reports and reprints of articles.

- In a workingpaper are laid down the preliminary findings of unfinished research.
  Workingpapers are for internal purposes mainly.

- A report gives account of finished research which in principal contains some original results.

Coverdesign: Audiovisual Department EUR

A PL/I COMPUTER PROGRAM FOR I.M.G.P. USING THE M.P.S.X. PACKAGE

(first draft)

by Cok Ouwerkerk and Jaap Spronk *)

CONTENTS

*) We would like to thank Drs J. Hartog for his patient willingness to answer our questions on PL/I and on the MPSX-package.

1. INTRODUCTION

In this paper, we describe a computer-program for Interactive Multiple Goal Programming (I.M.G.P.), as described in Nijkamp and Spronk [1978a]. This method is based on a mutual and successive interplay between a decision-maker and an expert. By consequence, a computer program for the use of this method should be of the conversational type. In that case, the decision-maker has a computer-terminal to his disposal by means of which he can interfere with the program and thus articulate his preferences during each step of I.M.G.P. However, to avoid a large amount of technical details (which depend largely on the very computer system used), we here do not present a program of the conversational type. Instead, we give a program in which the concept of an 'imaginary decision-maker' (see section 2) is used. As will be shown in the final section, the latter program can easily be changed into one of the conversational type. However, the required modifications depend on the computer system used.
The present program consists of three subprograms, written in PL/I and using the MPSX-package. The first subrogram, described in section 3, reads the data and generates the input for the MPSX-package. The second subprogram, given in section 4, calculates an optimal solution for the imaginary decision-maker's objective functions. These first two subprograms can also be used for standard linear and multiple goal programming. The third subprogram forms the actual Interactive Multiple Goal Programming method. All programs have been run on the IBM 370/158 computer of the Technical University in Delft, the Netherlands. Obviously, they can also be run on other computer systems in which both PL/I and the MPSX-package are being used.
The authors realize that the described programs can be improved upon. They will be grateful for any suggestion to increase the efficiency of the programs and to correct possible errors.

2. PRELIMINARIES

In this section we discuss the formulation of the linear programming problems to be solved within the computer programs (subsection 2.1.), the characteristics of the imaginary decision-maker (subsection 2.2.) and the structure of the computer program (subsection 2.3.).

2.1. The formulation of the linear programming problems

During each iteration of IMGP, there are a number of goal variables $g_i(\underline{x})$ ($i = 1, \ldots, m$), each of which is to be minimized or maximized, given the same set of linear constraints. This set of constraints normally consists of:

(a) the set of constraints $h(\underline{x}) \leqslant \underline{h}$, describing the feasible region R of the instrumental variables x_i, $i = 1, \ldots, n$ ($\underline{x}$ in vector notation) where $\underline{h}(\underline{x}) = \underline{B}.\ \underline{x}$, and where B is a matrix of order (k x n).

(b) the set of constraints $\underline{g}(\underline{x}) = A.\ \underline{x}$, where A is a matrix of order (m x n), describing the goal variables in terms of the instrumental variables $\underline{x}$.

(c) the constraints $g_i(\underline{x}) \geqslant \gamma_i$ (or $g_i(\underline{x}) \leqslant \gamma_i$ in the case $g_i(\underline{x})$ is to be minimized), for $i = 1, \ldots, m$.

The constraints (a) and (b) remain unchanged during the whole procedure, while the set of constraints (c) changes from iteration to iteration. The latter changes occur because, each time, at least one of the right-hand side values γ_i is shifted by the decision-maker.

To bring the computer program in line with the standard version of MPSX, and to avoid unnecessary complications of the program, we assume that each of the goal variables is to be minimized. This assumption does not deliver any problems. For instance, a goal variable $g_i(\underline{x})$ to be maximized can be transformed into $g_i'(x) = -g_i(\underline{x})$, which then has to be minimized. Another possibility is to formulate the goal restriction $g_i(\underline{x}) - y_i^- = \gamma^*$, where γ^* is a value of $g_i(\underline{x})$ which is known to be unattainable, and then to minimize y_i^-. The use of such 'deviational' variables can also be very useful in other cases (cf. Nijkamp and Spronk [1978a]).

Notice, that the above assumption simplifies the program very much. At each iteration, the problem is to minimize (consecutively and separately) a number of goal variables, subject to the same set of constraints.

Then one or more of the right-hand values is changed and the minimizations can be carried out again, while using some of the characteritics (e.g. basis or solution values) of the former solutions.

2.2. The imaginary decision maker

As mentioned above, we do not present a computer program of the conversational type here. Instead, we introduce the concept of an 'imaginary decision-maker', who makes the necessary managerial choices[1].
In order to make the program suitable for a decision-maker in reality, some modifications are necessary. However, because these modifications depend largely on the computer system which is being used, we do not give them in detail here. In the final section, we will indicate in which way the computer program should be changed in order to make it suitable for practical use.
The managerial choices to be made, both by the imaginary and the real-life decision-maker are the following. First, given a proposal solution, she (or he) has to indicate which of the goal variables need(s) to be improved in value. We assume that the preferences of the imaginary decision-maker can be described adequately by a preference function which is known to us but not to him. Nevertheless, we assume him to be able to give his judgements concerning a proposal solution, and moreover to do so in a manner which is in complete accordance with the preference function specified.
In this case, we assume, that the imaginary decision-maker's preference function is linear in the goal variables. Furthermore we assume that he, within the IMGP framework, first tries to drive the first variable to its optimal value (up to a small, predeterminal margin), then the same with the second and the third goal variable respectively, where the rank order is chosen arbitrary. In this case, the sizes of the proposed changes of the goal values are calculated within the method. Thus, the imaginary decision-maker does not formulate any aspiration levels, neither before nor during the interactive process. For the merits and the demerits of these assumptions we refer to Nijkamp and Spronk [1978b].
A second managerial judgement to be made, concerns the acceptability of a proposed shift in one or another goal value, vis-a-vis the loss of potential values of the unchanged goal variables. Given the a priori knowledge

---

[1] This concept has also been used in Nijkamp and Spronk [1978b] and [1978c].

of the imaginary decision-maker's preference function, an optimal solution of the problem at hand can be calculated in a straightforward manner. This optimal solution is then used to evaluate the proposed shifts in the value of the goal variables. The question simply becomes, whether the proposal solution excludes the optimal solution or not. If so, the proposal solution is clearly unacceptable. If not, the proposed solution is being accepted and the decision-maker again has to indicate which of the goal variables has to be raised, and so on.

2.3. <u>Structure of the computer program</u>

This computer program consists of three parts or (sub)programs. The first subprogram, described in section 3, generates the input for the MPSX-package. In each of the three programs, we are using the following 'standard' terms:

NP = number of problems, i.e. number of goal variables

DD = data set name

PP = problem name: there are always NP + 1 problem names. Problem names PP||1 up to and including PP||NP correspond with the NP goal variables. Problem name PP||NP + 1 corresponds with the imaginary decision-maker's preference function.

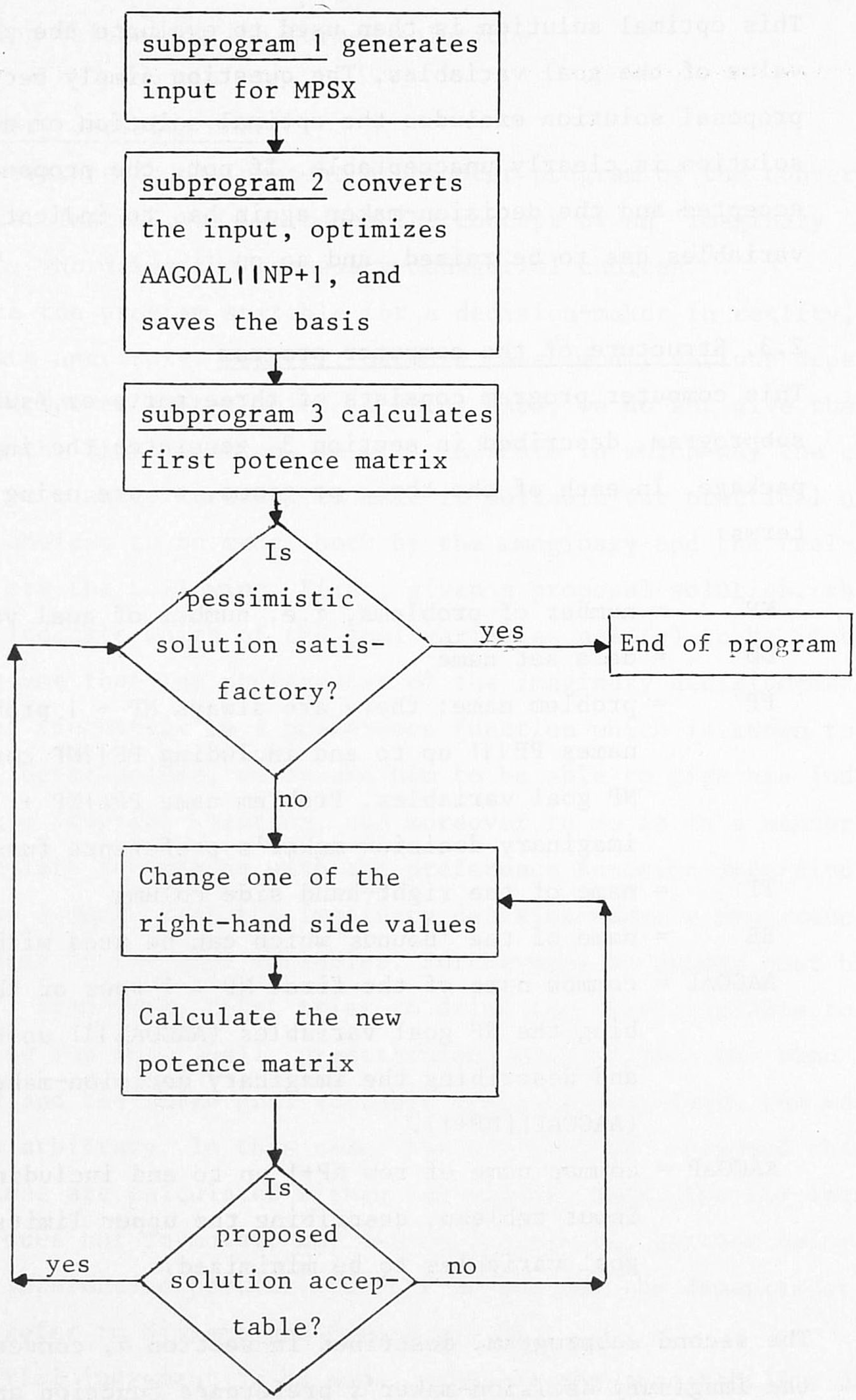
TT = name of the right-hand side column

BB = name of the 'bounds'which can be used within MPSX (see section 3.).

AAGOAL = common name of the first NP + 1 rows of the input tableau, describing the NP goal variables (AAGOAL||1 up to and including AAGOAL||NP) and describing the imaginary decision-maker's preference function (AAGOAL||NP+1).

AAGOAP = common name of row NP+1 up to and including row 2xNP+1 of the input tableau, describing the upper limits for the (NP) respective goal variables to be minimized.

The second subprogram, described in section 4, converts the input, optimizes the imaginary decision-maker's preference function and saves the final basis for use in the third subprogram. The third subprogram, described in section 5, is the actual IMGP procedure. In broad lines only, the following chart shows the structure of the total program.



A more detailed chart of the third subprogram is given in section 5.

3. THE INPUT GENERATOR

MPSX expects input in a certain format and in a certain order. In order to fulfil these requirements, we use the input generator ETMIC. Below, we only give a brief description of the card deck as required by ETMIC, together with an example of the input and of ETMIC. For further details, we refer to Lebret [1977].

Of each card, only the first 72 characters are used by ETMIC. The input card deck consists of one name definition card and a series of sections in which the linear programming tableau is defined. The format of the first card is

text   name1   name2   name3

These words must be separated by blanks. 'text' can be any string without blanks.

'name1' is the name of the data-deck for MPS (the NAME card of MPS).

'name2' is the name of the RHS column.

'name3' may be used (not necessarily) to define the RANGES column for MPS.

The rest of the input is organized in 'sections'. There are five kinds of sections: ROW, SOS, BOUNDS, INTEGERS and RHS. The sections 'SOS' and 'INTEGERS' are only used in (mixed) integer programming problems. The use of the 'BOUNDS' section is optional. However, the sequence of the sections must be conform the above list. Each section begins with an identification card and ends with an END card.

The following examples show how each kind of section must be specified.

4. OPTIMIZATION OF THE IMAGINARY DECISION-MAKER'S PREFERENCE FUNCTION

Below, we show the subprogram which converts the input, optimizes the imaginary decision-maker's preference function (i.e. minimizes AAGOAL||NP+1), and saves the basis for subsequent problems. The program should be self-explanatory. As noted above, this program can also be used to solve ordinary linear programming problems. In that case, define NP=0 and define the target row as AAGOAL1.

Computer program

```
//ECHRSPNK  JOB xxx,'OUWERKERK-SPRONK',REGION=570K,TIME=(,25)
//   EXEC MPSECL
//PLI.SYSIN  DD  *
 FIRST: PROC OPTIONS(MAIN);
 %INCLUDE DPLINIT;
 DCL NP FIXED BIN,(DD,PP,TT,BB) CHAR(7),HULP PIC'9';
 GET LIST(NP,DD,PP,TT,BB);
 A: DO;
      HULP=NP+1;
      XDATA=DD;
      XPBNAME=PP||HULP;
      CALL CONVERT('FILE','ORIG');
      XBOUND=BB;
      CALL SETUP;
      XOBJ='AAGOAL'||HULP;
      XRHS=TT;
      XOPSAVE='BAS'||HULP;
      CALL OPTIMIZE;
      CALL SOLUTION;
 END A;
 END FIRST;
//GO.PROBFILE  DD  DSN=ECHR.PPRODPL,DISP=(NEW,CATLG),
//   UNIT=DISK,VOL=SER=DISK12
//GO.ORIG  DD  DSN=ECHR.DPRODPL,DISP=SHR
//GO.SYSIN  DD  *
 5,'DPROD','PPROD','TPROD','BUFFER'
//
```

5. THE SUBPROGRAM FOR I.M.G.P.

In this section, we give a listing of the subprogram for an imaginary decision-making using I.M.G.P. Before doing so, we explain the most important names used within the program, followed by a flow chart of the program. Some further remarks will be made in the final section.

<u>Explanation of names used within the program (in order of occurrance)</u>

SOLOPT = vector of goal values for the optimal solution of the imaginary decision-maker's preference function

SOLACC = vector of pessimistic goal values

SOLOPT = vector of ideal goal values

DELTA = vector of step sizes for the shifts in SOLACC

ACCUR = vector of accuracies

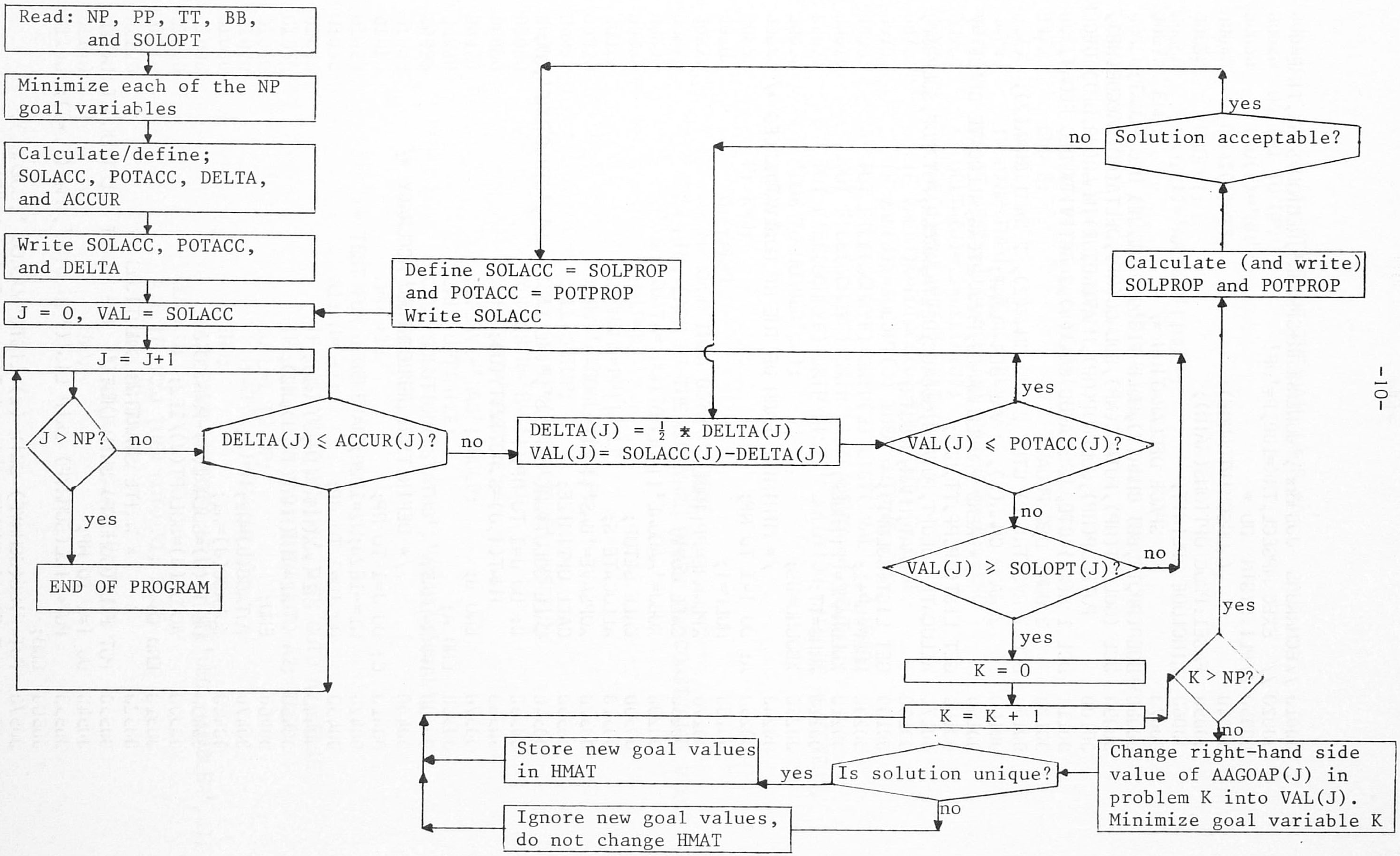
VAL = vector of changed right-hand side values for the AAGOAP restrictions

HMAT = matrix of goal values obtained during the respective minimizations of the NP goal variable

SOLPROP = vector of pessimistic goal values after the proposed shift in one of the AAGOAP restrictions

POTPROP = vector of ideal goal values corresponding with SOLPROP

Flow chart of the subprogram for IMGP



```
//ECHRKUNS  JOB xxx ,'OUWERKERK-SPRONK',REGION=576K,TIME=4
// EXEC HPSECL,TRK=180,P='OF'
//PLI.SYSIN  DD *
           (SUBSCRIPTRANGE):
 FASE1:PROC OPTIONS(MAIN);
 %INCLUDE DPLINIT;
           /* SPACE DECLARATION */
 DCL (PP,TT,BB) CHAR(7),HULP PIC'9',(W1,W2) DEC FLOAT;
 DCL (SOLOPT(NP),POTACC(NP),SOLACC(NP),DELTA(NP),ACCUR(NP),
      POTPROP(NP),SOLPROP(NP),HMAT(NP,NP),HMAR(NP,NP)) DEC FLOAT CTL;
 DCL 1 S(XJ) CTL, 2 STATUS CHAR(4),2 ACTIVITY DEC FLOAT,
      2 DUAL DEC FLOAT;
 DCL 1 RVSSTR(NP) CTL, 2 IND CHAR(2), 2 NAM1 CHAR(7),
      2 NAM2 CHAR(8), 2 VAL DEC FLOAT;
           /* READ PROBLEMNAMES,PARAMETERS,ALLOCATE SPACE */
 GET LIST(NP,PP,TT,BB);
 ALLOCATE SOLOPT,POTACC,SOLACC,DELTA,ACCUR,POTPROP,SOLPROP,
           HMAT,HMAR,RVSSTR;
 GET LIST(SOLOPT);
 HULP=NP+1;
 XOLDNAME=PP||HULP;
 XRHS=TT;
 XBOUND=BB;
           /* MINIMIZE EACH OF THE NP GOALVARIABLES */
 A: DO I=1 TO NP;
    HULP=I;
    XPBNAME=PP||HULP;
    CALL COPY;
    XOBJ='AAGOAL'||HULP;
    CALL SETUP;
    ALLOCATE S;
    XOPSAVE='BAS'||HULP;
    CALL OPTIMIZE;
    CALL SOLUTION('STATUS','VALUE','DUAL','STRUCTURE',S);
    B: DO J=1 TO NP;
       HMAT(I,J)=S.ACTIVITY(J);
    END B;
 END A;
 HMAR=HMAT;
           /* DEFINITION OF REQUIRED ACCURACY */
 C: DO J=1 TO NP;
    W1=-1E20;W2=1E20;
    DO IH=1 TO NP;
       W1=MAX(HMAT(IH,J),W1);
       W2=MIN(HMAT(IH,J),W2);
    END;
    POTACC(J)=W2;
    SOLACC(J)=W1;
    DELTA(J)=SOLACC(J)-POTACC(J);
    ACCUR(J)=SOLOPT(J)/1E2;
 END C;
           /* WRITE STARTING SOLUTIONS */
 PUT FILE(SCHRYF) SKIP LIST('  THE FIRST SOLUTIONMATRIX');
 DO I=1 TO NP;
    PUT FILE(SCHRYF) SKIP LIST('        ',HMAT(I,*));
 END;
 PUT FILE(SCHRYF) SKIP (2) LIST('SOLACC ',SOLACC);
 PUT FILE(SCHRYF) SKIP LIST('SOLOPT ',SOLOPT);
 PUT FILE(SCHRYF) SKIP LIST('POTACC ',POTACC);
```

```
                 /* START OF THE MAIN LOOP */
      DO I=1 TO NP;
         IND(I)='RH';
         NAM1(I)=TT;
         HULP=I;
         NAM2(I)='AAGOAP'||HULP;
      END;
                 /* CHOICE OF THE GOAL VALUE TO BE CHANGED */
      D: J=0;
         E: DO II=1 TO NP;
            VAL(II)=SOLACC(II)+(9E-6)*SOLACC(II);
         END E;
         J=J+1;
         IF J>NP THEN GOTO Z; ELSE
         G: IF DELTA(J)<=ACCUR(J) THEN GOTO E; ELSE
            DELTA(J)=.5*DELTA(J);
            VAL(J)=SOLACC(J)-DELTA(J);
            IF VAL(J)<=POTACC(J) THEN GOTO G; ELSE
            IF VAL(J)>=SOLOPT(J) THEN GOTO P; ELSE
            PUT FILE(SCHRYF) SKIP(2) LIST('VAL    ',VAL);
            PUT FILE(SCHRYF) SKIP LIST('DELTA   ',DELTA);
            PUT FILE(SCHRYF) SKIP LIST(' SOLUTION WILL BE REJECTED',
            'DUE TO NUMBER',J);
            GOTO G;
            P: K=0;
               Q: K=K+1;
                  IF K>NP THEN GOTO R2;ELSE
               /* MIN. EACH OF THE GOALVARS GIVEN NEW CONSTRAINT VALUE */
                  R1: XDATA='DUMMY';
                      HULP=K;
                      XPBNAME=PP||HULP;
                      XOBJ='AAGOAL'||HULP;
                      CALL SETUP;
                      CALL MODIFY('STRUCTURE',RVSSTR);
                      XOPRESTR='BAS'||HULP;
                      XOPSTART='RESTORE';
                      XOPSAVE='BAS'||HULP;
                      CALL OPTIMIZE;
                      CALL SOLUTION('STATUS','VALUE','DUAL',
                      'STRUCTURE',S);
                      M=NP+1;
               /* TEST FOR DEGENERACY */
                      A1: M=M+1;
                          IF S.STATUS(M)¬='*L' THEN GOTO A2; ELSE
                          IF S.DUAL(M) > 0. THEN GOTO A2; ELSE
                          DO L=1 TO NP;
                             HMAT(K,L)=HMAR(K,L);
                          END;
                          PUT FILE (SCHRYF) SKIP LIST('DEG. PROBLEM',K);
                      GOTO Q;
                      A2: IF M<XJ THEN GOTO A1; ELSE
                          DO L=1 TO NP;
                             HMAT(K,L)=S.ACTIVITY(L);
                          END;
                  GOTO Q;
```

```
           /* DETERMINATION OF POTENCE MATRIX */
              R2: DO L=1 TO NP;
                    W1=-1E20;W2=1E20;
                  DO IH=1 TO NP;
                      W1=MAX(HMAT(IH,L),W1);
                      W2=MIN(HMAT(IH,L),W2);
                  END;
                  POTPROP(L)=W2;
                  SOLPROP(L)=W1;
              END R2;
              PUT FILE(SCHRYF) SKIP (2) LIST('SOLPROP',SOLPROP);
              PUT FILE(SCHRYF) SKIP LIST('POTPROP',POTPROP);
              PUT FILE(SCHRYF) SKIP LIST('SOLOPT ',SOLOPT);
              PUT FILE(SCHRYF) SKIP LIST('DELTA  ',DELTA);
              PUT FILE(SCHRYF) SKIP LIST
                  ('IS SOLUTION ACCEPTABLE?','            PROBLEM NUMBER',J);
              T: DO K=1 TO NP;
                  IF(SOLPROP(K)<(SOLOPT(K)-(9E-6)*SOLOPT(K)))THEN GOTO G;
                  IF(SOLOPT(K)<(POTPROP(K)-(9E-6)*POTPROP(K)))THEN GOTO G;
              END T;
              HMAR=HMAT;
              SOLACC=SOLPROP;
              POTACC=POTPROP;
              PUT FILE(SCHRYF) SKIP LIST('SOLUTION ACCEPTED');
              PUT FILE(SCHRYF) SKIP LIST('SOLACC ',SOLACC);
          GOTO D;
  Z: PUT FILE(SCHRYF) SKIP LIST('END OF PROGRAM');
  END FASE1;
//GO.PROBFILE DD DISP=OLD,DSN=ECHR.PPRODPL
//GO.SYSPRINT DD DUMMY
//GO.OLDPFILE DD DISP=OLD,DSN=ECHR.PPRODPL
//GO.SCHRYF DD SYSOUT=A,DCB=(RECFM=FA,LRECL=133)
//GO.SYSIN DD *
 5,'PPROD','TPROD','BUFFER'
 186776.63761,3958.09928,11355.28498,
 264.38750,8700.29052
//
```

6. SOME FINAL REMARKS

In this final section, we indicate how to transform the described program into one of the conversational type. Furthermore, we discuss some other possible improvements of the program.

When a program of the conversational type is being made, it should be realized that a decision-maker in reality may make some simple errors (like typing errors) which have to be corrected. This makes it necessary to use a subprogram for the detection, report, and to make it possible to reset these errors. Such a subprogram may have many forms, depending on the computer system used, the availability of other error detecting codes, and so on. Therefore, we do not present such a subprogram here.

In order to make the described program of the conversational type, the following parts of the program should be adapted. First, when an accepted solution is presented to the decision-maker, he must have the opportunity to state whether this solution is satisfactory or not. In the latter case, he must also have the opportunity which of the goal variables should be improved in value, and by which amount. In order to so, the statements numbered[2)] 690 up to and including 760 should be replaced by some input statements. The second problem to be solved by the decision-maker, is whether a proposal solution is satisfactory or not. In our program, this is done in the statements numbered 790 up to and including 840 and in 1310 up to and including 1350. Consequently, these statements should be adapted for the conversational use of the program.

At the end of this paper, we should mention a possible improvement of the program. This may be found by using only one single linear programming problem for the whole procedure, instead of NP+1 problems used above. Furthermore, in using the outcomes of one optimization as a point of departure for the next optimization, one may consider to use the old activity levels instead of the old bases, which was done in our program. At the time of writing this paper, these possibilities had not been investigated in detail.

---

2) The statement numbers have been added for illustrative purpose only.

REFERENCES

IBM Mathematical Programming System Extended/370(MPSX/370) Program Reference Manual (3rd edition) 1977.

Lebret, M.D., ETMIC-Equation To MPS - Input Conversion, Rekencentrum T.H.-Delft, User's Guide RC-TWA-77004, Delft, April 1977.

Nijkamp, P. and J. Spronk, Interactive Multiple Goal Programming, Report 7803/A, Centre for Research in Business Economics, Erasmus University, Rotterdam, 1978a.

Nijkamp, P. and J. Spronk, Interactive Multiple Goal Programming; Method and Application, Report 7812/F, Centre for Research in Business Economics, Erasmus University, Rotterdam, 1978b.

Nijkamp, P. and J. Spronk, Three Cases in Multiple Criteria Decision-Making: An Interactive Multiple Goal Programming Approach, Report 7822/A, Centre for Research in Business Economics, Erasmus University, Rotterdam, 1978c.

REPORTS 1978
============

A - General Business Economics
F - Business Finance and Portfolio Investment
M - Marketing
O - Organization, Production
ACC - Accountancy

7801/A Nijkamp, P. and J. Spronk, Analysis of production and location decisions by means of multi-criteria analysis, February 1978.

7802/M Zwan, A. van der, 'Neo-professionalisering' - integraal verslag van een onderzoek onder drie groepen beroepsbeoefenaren, maart 1978.

7803/A Nijkamp, Peter and Jaap Spronk, Interactive Multiple Goal Programming, April 1978.

7804/F Jonkhart, Marius J.L., Bond Risk Differentials, the Term Structure of Interest Rates and the Risk of Default, April 1978.

7805/A Keus, J., Investeringsbeslissingen van ondernemers in macro-economisch perspectief, april 1978, discussienota.

7806/F Herst, A.C.C., Leasen of kopen: een kwestie van druk? juni 1978.

7807/F Jonkhart, Marius J.L., Bond Risk Differentials, the term structure of interest rates and the risk of default: A correction and an extension, May 1978.

7808/M Wijst, D. van der, Enige aspekten van het onderzoek t.b.v. de psychologische positionering van produkten, mei 1978.

7809/A Moerland, P.W., Prijsgedrag in theorie en praktijk, juni 1978.

7810/F Jonkhart, Marius J.L., On the Order of Debt Claims, July 1978.

7811/F Diepenhorst, A.I., Vennootschappelijke of Particuliere Schuld-aanvaarding, gevold door
Elkaar Compenserende Fouten, augustus 1978.

7812/F Nijkamp, Peter and Jaap Spronk, Interactive multiple goal programming: method and application, June 1978.

7813/F Diepenhorst, A.I., A multi-period cost of capital concept and its impact on the formulation of financial policy, August 1978.

7814/F Ballendux, F.J. and J.K. van Vliet, Capital Budgeting under Uncertainty: a neglected institutional constraint? August 1978.

7815/F Ballendux, F.J. and J.K. van Vliet, Firm Effects and Project Values, September 1978.

7816/F Vliet, J.K. van, Discriminant Analysis as an Alternative Capital Budgeting Rule, September 1978.

7817/F Jonkhart, M.J.L., Determinants of coporate borrowing: a reinterpretation, November 1978.

7818/F Verboom, P.M., Het valutarisico in de internationale onderneming, oktober 1978 (discussienota).

7819/M Zwan, A. van der, m.m.v. H. Wijnberger, 'De markt voor het marktonderzoek' - Een integraal verslag van de uitkomsten van een enquête onder nederlandse (markt-)onderzoekbureaus 1946-1975, december 1978.

7820/M Zwan, A. van der, m.m.v. H. Wijnberger, 'De markt voor het marktonderzoek' - Een integraal verslag van de uitkomsten van een enquête onder nederlandse bedrijven naar het gebruik en de toepassing van marktonderzoek 1977, december 1978.

7821/M Zwan, A. van der, The industrialization of market research - The disintegration of its professional structure, December 1978.

7822/A Nijkamp, Peter and J. Spronk, Three Cases in Multi Criteria Decision Making: An Interactive Multiple Goal Programming Approach, December 1978.

7823/A Ouwerkerk, C. and J. Spronk, A PL/I Computer Program for I.M.G.P. Using the M.P.S.X. Package, December 1978.