NBSIR 75-666

# Post Optimality and Parametric Analysis with the National Bureau of Standards' Linear Programming Subroutine RVSMPX

Tyrone B. Ayers

Applied Mathematics Division

February 1975

Final

Technical Report to

**Computer Services Division**
**Center for Computer Sciences and Technology**

NBSIR 75-666

# POST OPTIMALITY AND PARAMETRIC ANALYSIS WITH THE NATIONAL BUREAU OF STANDARDS' LINEAR PROGRAMMING SUBROUTINE RVSMPX


Tyrone B. Ayers

Applied Mathematics Division



February 1975

Final

Technical Report to
Computer Services Division
Center for Computer Sciences and Technology

CONTENTS



LIST OF FIGURES

ABSTRACT

This report is a sequel to NBS Report 10695 (February 1972), The National Bureau of Standards' Linear and Quadratic Programming Subroutines, which documented one phase of an effort to provide users of the facility operated by the National Bureau of Standards' Computer Services Division, with reliable clearly-described solution algorithms for selected frequently-arising classes of special mathematical problems. The present report presents subroutines which perform post-optimality analysis and parametric programming studies on linear programming problems solved by the National Bureau of Standards' RVSMPX subroutine. (The present versions of these codes use internal storage only.)

Keywords: Algorithms; post-optimality analysis; parametric programming; linear programming.

ACKNOWLEDGEMENT

The author would like to acknowledge the aid provided by W.G. Hall, R.H.F. Jackson, and P.B. Saunders of the Applied Mathematics Division in the formulation and testing of the subroutines discussed in this report.

# 1. INTRODUCTION

This report documents another phase of a continuing effort to provide users of the computer facility operated by the National Bureau of Standards' Computer Services Division with reliable, clearly-described solution algorithms for selected, frequently-arising classes of mathematical problems. This phase of the effort has increased NBS's capabilities in the area of linear programming by producing three subroutines: RANGES, a subroutine which provides post-optimality ranges for the objective function coefficients (costs) and for the constants which constitute the right-hand-sides (RHS) of the linear program's constraints; PAROBJ, a subroutine which performs a parametric study of the effects of changes in the costs; and PARRHS, a subroutine which performs a similar parametric study for changes in the RHS.

The subroutines were written in FORTRAN V, solve the problems "in core," and were designed to be used with RVSMPX, a subroutine for performing the revised simplex method of linear programming. RVSMPX is described in the previous report [2], and familiarity with that document is assumed here.

Sections 2 and 3 of this report discuss, respectively, the methods used in RANGES, and those employed for PAROBJ, and PARRHS. Section 4 presents the deck set-up, and parameter settings required to use the routines, and Section 5 provides sample outputs with explanations. The appendices contain program listings, and also the results of sample problem runs made for purposes of code verification and timing.

## 2. SENSITIVITY ANALYSIS ON COST AND RHS VECTORS (RANGES)

The input to RANGES, consisting of a linear programming problem together with its solution, is the output from RVSMPX (see [2]). The output from RANGES is a set of upper and lower bounds. An upper-lower bound pair appears for each objective function coefficient and also for each constraint right hand side, and represents the maximum range over which that quantity can vary (with all other problem data fixed) without forcing a change from the optimal basis for the original LP problem.* That is to say, the set of variables in the optimal basis will remain the same so long as no cost or RHS is changed beyond its bounds. It should be noted that in the case of a change in an RHS, although the basis remains the same, the values of the variables in the basis would change. Correspondingly, in the case of a change in a cost, the values of the variables in the basis as well as the membership of the basis will remain the same, while the value of the objective function changes.

Note, that due to round-off error the computed bound values may occasionally vary from the actual values by as much as 3 times the epsilon (EPS) computed in RVSMPX. However, the computed values, when substituted for their corresponding cost or RHS, do indeed yield the same optimal basis.

A number of computer runs were made to check the "correctness" of the code. The times for these runs (which, by the way, employed the very same problems that were used to verify RVSMPX in [2]) are reported in Appendix D. A potentially useful statistic from these runs is that the time consumed by RANGES was on the average roughly 20% of that required by RVSMPX to solve the original problem.

---

*This technique is described in detail in [1] and [3].

## 3. PARAMETRIC PROGRAMMING ON COST VECTOR (PAROBJ) OR RHS VECTOR (PARRHS)

Consider the linear programming problem given in matrix notation as

$$\max cx, \text{ subject to } Ax:b \text{ and } x \geq 0,$$

where ":" represents $\leq$, $\geq$, or =.* If B is the basis matrix, and $c_B$ and $x_B$ are the corresponding subvectors of c and x respectively, then the solution to this problem is given by $x_B = B^{-1}b$, with optimal objective value of $c_B x_B$.

Parametric programming on the cost vector is concerned with replacing c in the above problem by $c^* = c + \theta f$, where f is an arbitrary vector supplied by the user and $\theta$ is a nonnegative scalar parameter. PAROBJ solves this problem by first solving the original problem using RVSMPX, then computing the largest value of $\theta$ for which the current optimal basis remains optimal. Call this value $\theta_{max}$. A new basic solution is then calculated which is optimal in some new interval of $\theta$ values beginning with $\theta_{max}$. The upper bound of that new interval is the new value of $\theta_{max}$.

This procedure continues until either a.) the problem becomes unbounded (linear programming theory assures that it will remain so for all larger values of $\theta$), or b.) the last of the critical values of $\theta$ is reached (the same solution remains optimal for all larger values of $\theta$), or c.) a user-specified limit on $\theta$ is exceeded.

If the solution to the original problem is unbounded, it is still possible that for some positive value of $\theta$, the solution to the modified problem "becomes" bounded. If such is the case for the original linear programming problem, then PAROBJ will start the parametric study with the smallest positive value of $\theta$ for which a bounded, feasible solution is obtained.

As with RANGES, a number of sample problems were run on PAROBJ for the purpose of code verification and timing. The results of these runs are reported in appendix D.

* * *

PARRHS is a subroutine written to perform parametric programming on the RHS vector. This is the situation in which vector b of the original problem stated in the beginning of this section is replaced by $b^* = b + \theta r$, where r is an arbitrary vector supplied by the user, and $\theta$ is as before. All the comments stated above for PAROBJ with regard to boundedness, apply for PARRHS except that the concern here is with feasibility.

---

*By way of clarification, RVSMPX accepts the A matrix in non-canonical form, so that the constraints need not be all equalities or inequalities.

## 4. DECK SET-UP AND PARAMETER SETTINGS

The previous sections of this report discussed the subroutines in general terms, stressing the functions they perform. This section will provide the reader with the information needed actually to use the subroutines. The deck set-ups will be discussed first, followed by the form of the call to the subroutines, followed further by an explanation of each of the parameters employed.

RANGES is used in what can be termed a "post-RVSMPX" mode. In the user's main program, after the call to RVSMPX in the form described in [2], the user simply inserts a FORTRAN CALL statement of the following form:

```
CALL RANGES (A,MA,B,MB,MT,NT,L,X,RHSUP,RHSLOW,COEUP,COELOW,NSET),
```

and the range information is printed out.

The observant reader will note that many of the parameters in the above CALL statement appear identical to those in the CALL statement for RVSMPX. In most cases they are exactly the same. There are, however, some variations, which are listed below and should be NOTED by the user.

Both PAROBJ and PARRHS operate in a "super-RVSMPX" mode. The user-supplied main program referred to earlier and discussed in [2] should pass control to one of these routines, which then takes charge of the parametric study and uses RVSMPX as a subroutine. This, of course, means that any parameters required by RVSMPX which would normally be passed directly to it, must now be passed through the parametric programming routines. The FORTRAN CALL statements, as they should appear in the main program, are:

```
CALL PAROBJ (A,MA,B,MB,MT,NT,L,X,TOLP,INV,F,BOUND,NSET),

CALL PARRHS (A,MA,B,MB,MT,NT,L,X,TOLP,INV,BOUND,R,NSET).
```

Again it should be noted that the similarity between these CALL statements and the one for RVSMPX is not accidental. A full understanding of the parameters, especially as they relate to the various subroutines, is absolutely imperative. Listed below are the parameters that appear in RANGES, PAROBJ, and PARRHS. After each variable name, there appears an (R), an (O), or an (H) if the variable is a parameter of either RANGES, PAROBJ, or PARRHS, respectively. If the variable pertains to all three subroutines without variation, no letters will appear. But if variations are present, an explanation will be given. The m and n used with reference to dimensions have the same meaning as in the RVSMPX documentation: m is the number of constraints, and n is the number of real (original) variables.

* * *

<u>A</u> (R) This is the augmented constraint matrix. It should be dimensioned at least (m+1) by (n+1) and is unchanged by the subroutine.

(O) Because of the nature of PAROBJ, A must be dimensioned at least (m+2) by (n+1) to provide storage for the original coefficients of the objective function. A is unchanged by PAROBJ.

(H) Because of the methods used in storing the original coefficients of the objective function and in dealing with an originally infeasible problem, A must be dimensioned at least (m+2) by (n+2). A is unaltered by PARRHS.

MA This variable equals the value of the first dimension of A. It is unchanged upon exiting from the subroutines.

B (R) This matrix provides storage for the inverse of the basis matrix and should be dimensioned (m+2) by (m+2). B is unchanged by the subroutine.

(O,H) Since the subroutines perform parametric studies which cover several bases, B will contain the inverse of the last optimal basis matrix upon exiting from the subroutines. B should be dimensioned at least (m+2) by (m+2).

MB This is the value of the first dimension of B. It is not altered by the subroutines.

MT (R) This is the number of rows of information in the A matrix, i.e., (m+1). MT is not changed by RANGES.

(O,H) Recall that the A matrix for PAROBJ and PARRHS has (m+2) rows. Hence MT must equal m+2. Upon exiting from the subroutine, MT is unaltered.

NT (R,O) This variable equals the number of columns of information in the A matrix, i.e., (n+1). It is unchanged by the subroutines.

(H) Recall that the A matrix for PARRHS has (n+2) columns. Therefore, NT must equal (n+2). Upon exiting from the subroutine, NT is not changed.

L (R) This is a multipurpose vector providing work space needed by RVSMPX. L is not changed in any way by RANGES.

(O,H) In order to maintain a certain consistency in output formulation, the user-supplied values of L only function for the initial reporting of the problem, after which certain elements of L are automatically set by the subroutines. Upon exiting, L(5), L(8), L(13), and L(14) will equal 1. L(4) and L(12) will equal 0. The value of all other L(I), I $\leq$ 14, will be the same as they are upon returning from RVSMPX.

X (R) Just as in RVSMPX, X provides storage for the solution of the problem. X is not changed by RANGES.

(O,H) These subroutines perform a parametric study over a series of optimal bases. Therefore, the solution in X upon exiting from the subroutines will not be the original solution, but rather the solution corresponding to the final optimal basis of the parametric study.

TOLP (O,H) This is the variable used by RVSMPX to compute the value of EPS and CEPS. Its value on input should be exactly the same as it is for RVSMPX. TOLP is not changed by the subroutine.

INV (O,H) This is the switch allowing the user to supply RVSMPX with an initial feasible solution. As in RVSMPX, INV may still be used to indicate a user-supplied initial basis. The value of INV is altered in the subroutines.

RHSUP (R) This vector provides storage for the upper bounds on the ranges of the right-hand sides of the given problem being solved. Since each right-hand side will have an upper bound, RHSUP should be dimensioned at least m in the main program, where m is the number of rows in the unaugmented A matrix.

RHSLOW (R) This vector provides storage for the lower bounds on the ranges of the right-hand sides. Like RHSUP, RHSLOW should be dimensioned at least m in the main program.

COEUP (R) This vector provides storage for the upper bounds on the ranges of the coefficients of the objective function of the given problem. COEUP should be dimensioned at least n in the main program, where n is the number of columns in the unaugmented A matrix.

COELOW (R) This vector provides storage for the lower bounds on the ranges of the coefficients. Like COEUP, COELOW should be dimensioned at least n.

R (H) This is the vector supplied by the user to indicate the rates of change of each of the right-hand sides with respect to the scalar $\theta$. Therefore, R must be dimensioned at least m. R should be set up so that the ith element of R is the rate of change of the ith right-hand side. For example, if the original right-hand side vector is $b=(1,2,3.7,5)^T$ and if R is $(0,2,-3,.6)^T$, then the new right hand side vector $b^*$ varies as follows:

$$b_1^* = b_1 + 0\,\theta = 1$$

$$b_2^* = b_2 + 2\,\theta = 2 + 2\,\theta$$

$$b_3^* = b_3 - 3\,\theta = 3.7 - 3\,\theta$$

$$b_4^* = b_4 + 0.6\,\theta = 5 + 0.6\theta$$

where $\theta \geq 0$. The subroutine does not change R.

F (O) This is the vector supplied by the user to indicate the rates of change of each of the objective function coefficients with respect to the scalar $\theta$. Therefore, F must be dimensioned by at least n. F should be set up so that the jth element of F is the rate of change of the jth cost coefficient. For example, if the original cost vector is $c = (2,3,1.6,-4)$ and if F is $(2,0,-.5,5)$, then the new cost vector $c^*$ varies as follows:

$$c_1^* = c_1 + 2\theta = 2 + 2\theta$$

$$c_2^* = c_2 + 0\theta = 3$$

$$c_3^* = c_3 - 0.5\theta = 1.6 - 0.5\theta$$

$$c_4^* = c_4 + 5\theta = -4 + 5\theta$$

where $\theta \geq 0$. F is not altered by the subroutine.

<u>BOUND</u> (O,H) This is the user-specified limit on the non-negative scalar parameter $\theta$. The scalar $\theta$ is incremented up to the value of BOUND. Once the value of $\theta$ exceeds BOUND, the subroutines will terminate by reporting that the limit has been exceeded and that the current basis remains optimal at that limit. However, the subroutines may terminate before $\theta$ reaches BOUND if:

1) the largest critical value of $\theta$ is less than BOUND,
2) the problem becomes infeasible (PARRHS),
3) the problem becomes unbounded (PAROBJ).

The value of BOUND is unchanged by the subroutines.

<u>NSET</u> (R) This parameter is a user-specified switch. If the user sets NSET = 0, ranges are calculated for both right-hand sides and objective function coefficients. If NSET = 1, ranges are determined only for the right-hand sides. If NSET = 2, ranges are obtained only for the objective function coefficients.

(O,H) For these subroutines, this user-specified switch determines the output format. If NSET = 0, a detailed format is used. If NSET = 1, a compressed tabular summary format is used.

# 5. OUTPUT

This section of the report will discuss the output from the three subroutines, and illustrate them through actual output samples.

## 5.1 THE OUTPUT FROM RANGES

The printing of the output is controlled by RANGES. Information on the original right-hand side vector is printed first, followed by information on the objective function coefficient vector. The user may obtain information on both the right-hand side and the objective function coefficient vectors or either of them separately, depending on his specified value for the parameter NSET (see section 4). The output is printed in an E14.8 format as follows:

```
-.14845019+02  LE   RHS  1   LE  POS.INFINITY
 .54586695+01  LE   RHS  2   LE  POS.INFINITY
 .30275573+01  LE   RHS  3   LE  .15395222+02
 .15161557+02  LE   RHS  4   LE  POS.INFINITY

 NEG.INFINITY  LE   COEFFICIENT  1   LE -.14055693+00
-.20215248+01  LE   COEFFICIENT  2   LE -.64413020+00
 NEG.INFINITY  LE   COEFFICIENT  3   LE -.54610899+01
 NEG.INFINITY  LE   COEFFICIENT  4   LE -.40288613+01
-.12985771-01  LE   COEFFICIENT  5   LE  .57053266+00
```

Line 1, for example, reads -14.845019 $\leq$ RHS 1 $\leq$ POSITIVE INFINITY. This means that the reported final basis will remain optimal as long as RHS 1 (the first element of the right-hand side vector of the problem being solved) is greater than -14.85019. Similarly, bounds are reported for each remaining element of the right-hand side vector, i.e., RHS 2, RHS 3, and RHS 4.

Line 5 reads

NEGATIVE INFINITY $\leq$ COEFFICIENT 1 $\leq$ -.14055693.

This means that coefficient 1, the first element of the cost vector of the problem being solved, has no lower bound and has an upper bound of -.14055693. Following line 5 are the bounds on the remaining elements of the cost vector.

## 5.2 THE OUTPUT FROM PAROBJ AND PARRHS

All output is completely controlled by the subroutines, and the output formats for the two routines are nearly identical. The first item to be printed is a statement that the first critical value of $\epsilon$ is zero. With this value of $\epsilon$ , the solution to the parametric programming problem is the same as the solution to the simple linear programming problem, and that solution is printed next. A formula for the calculation of the objective function value for all values of $\theta$ from zero through the next critical value is then printed.

A cycle of output then begins. The "current" critical value of $\theta$ is printed, along with the basis change required when passing through that value for $\theta$. The "current" basis is then given, followed by the objective function calculation formula. This output grouping is repeated until one of the termination conditions in section 3 is met. This output format is illustrated in Figures 1 and 2.

A more compressed tabular format is also available. The value of NSET (see Section 4) controls the output format selection. The compressed format is illustrated in Figures 3 and 4.

```
CRITICAL VALUE (  0) OF THETA =    .00000000

EPSILON =    .364000-04  CAPITAL EPSILON =    .364000-03      0 NON-ZERO ENTRIES ARE EFFECTIVELY EQUAL TO ZERO.

********************************************************************************************
*                                                                                          *
*          END OF PHASE 1.    OBJECTIVE FUNCTION =      .74505806-08    THERE WERE   2 ITERATIONS.
*          MINIMUM PIVOT WAS   1.0000    AT ITERATION   1.   REAL OBJECTIVE FUNCTION =  -1.2676858
*                                                                                          *
********************************************************************************************

********************************************************************************************
*                                                                                          *
*          END OF PHASE 2.    OBJECTIVE FUNCTION =     -.66359763       THERE WERE   1 ITERATIONS.
*          MINIMUM PIVOT WAS   1.0000    AT ITERATION   1.                                 *
*                                                                                          *
********************************************************************************************
BASIC.VARIABLES

  X(  1)=   .2211992

  SI  1)=   6.538539       S(  2)=   10.48937       S(  3)=   12.37799       S(  5)=   .1006814

OBJECTIVE FUNCTION VALUE BETWEEN CRITICAL VALUES (  0) AND (  1) OBTAINED FROM    .44239841+00* THETA    -.66359762+00

*  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *

CRITICAL VALUE (  1) OF THETA =   .62500000+00
AT THAT VALUE, X(  4) ENTERS THE BASIS AND SI   1) LEAVES.
```

Figure 1: PAROBJ Output
Regular Format

```
BASIC VARIABLES

   X(  1)=   .5184055       X(  4)=   2.080444

   S(  2)=   12.56981       S(  3)=   9.405929       S(  5)=   2.478332

OBJECTIVE FUNCTION VALUE = -.38709863+00
OBJECTIVE FUNCTION VALUE BETWEEN CRITICAL VALUES (  1) AND (  2) OBTAINED FROM   .51976991+01* THETA   -.36356606+01

 . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .

CRITICAL VALUE (  2) OF THETA =   .86764706+00
AT THAT VALUE, X(   5) ENTERS THE BASIS AND X(   1) LEAVES.

BASIC VARIABLES

   X(  4)=   5.579681       X(  5)=   1.425615

   S(  2)=   27.47397       S(  3)=   .3338330       S(  5)=   9.736009

OBJECTIVE FUNCTION VALUE =   .87410781+00
OBJECTIVE FUNCTION VALUE BETWEEN CRITICAL VALUES (  2) AND (  3) OBTAINED FROM   .14010593+02* THETA   -.11282142+02

 . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .

CRITICAL VALUE (  3) OF THETA =   .12500000+01
AT THAT VALUE, S(   4) ENTERS THE BASIS AND S(   3) LEAVES.

BASIC VARIABLES

   X(  4)=   5.746590       X(  5)=   1.592532

   S(  2)=   28.30855       S(  4)=   .6676660       S(  5)=   10.40367

OBJECTIVE FUNCTION VALUE =   .62310992+01
OBJECTIVE FUNCTION VALUE BETWEEN CRITICAL VALUES (  3) AND (  4) OBTAINED FROM   .14678259+02* THETA   -.12116724+02

 . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .

CRITICAL VALUE (  4) OF THETA =   .16111111+01
PROBLEM GOES UNBOUNDED BEYOND THIS CRITICAL VALUE.
```

Figure 1: PAROBJ Output
Regular Format (continued)

```
CRITICAL VALUE (  0) OF THETA =   .00000000

EPSILON =   .364000-04  CAPITAL EPSILON =   .364000-03   0 NON-ZERO ENTRIES ARE EFFECTIVELY EQUAL TO ZERO.

**************************************************************************************
*                                                                                    *
*        END OF PHASE 1.   OBJECTIVE FUNCTION =   .74505806-08   THERE WERE  2 ITERATIONS.
*        MINIMUM PIVOT WAS   1.0000   AT ITERATION   1.   REAL OBJECTIVE FUNCTION =  -1.2676858
*                                                                                    *
**************************************************************************************

**************************************************************************************
*                                                                                    *
*        END OF PHASE 2.   OBJECTIVE FUNCTION =   -.66359763     THERE WERE   1 ITERATIONS.
*        MINIMUM PIVOT WAS   1.0000   AT ITERATION   1.
*                                                                                    *
**************************************************************************************
BASIC VARIABLES

  X(  1)=   .2211992

  S(  1)=   6.538539    S(  2)=   10.48937    S(  3)=   12.37799    S(  5)=   .1006814

OBJECTIVE FUNCTION VALUE BETWEEN CRITICAL VALUES (  0) AND (  1) OF THETA OBTAINED FROM  =.85714285+00*THETA   =.66359763+00

 *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *

CRITICAL VALUE (  1) OF THETA =   .58730794-01
AT THAT VALUE, X(  5) ENTERS THE BASIS AND S(  5) LEAVES.

BASIC VARIABLES

  X(  1)=   .2379794    X(  5)=   .0000000

  S(  1)=   6.521758    S(  2)=   10.43064    S(  3)=   12.68004
```

Figure 2: PARRHS Output
Regular Format

```
OBJECTIVE FUNCTION VALUE = -.71393830+00
OBJECTIVE FUNCTION VALUE BETWEEN CRITICAL VALUES (  1) AND (  2) OF THETA OBTAINED FROM  -.22500000+01*THETA    -.58179401+00

 . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .

CRITICAL VALUE (  2) OF THETA =   .10106485+01
AT THAT VALUE, X(   4) ENTERS THE BASIS AND X(   1) LEAVES.

BASIC VARIABLES

   X(  4)=   .0000000      X(  5)=   .7139383

   S(  1)=   11.75731      S(  2)=   15.19023      S(  3)=   15.53579

OBJECTIVE FUNCTION VALUE = -.28557532+01
OBJECTIVE FUNCTION VALUE BETWEEN CRITICAL VALUES (  2) AND (  3) OF THETA OBTAINED FROM  -.25000000+01*THETA    -.32913187+00

 . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .

THE CRITICAL VALUE OF THETA IS INFINITE.

 . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .
```

Figure 2: PARRHS Output
Regular Format (continued)

```
THE R VECTOR IS AS FOLLOWS
R(   1) =  2.00
R(   2) =  1.00
R(   3) =  4.00
R(   4) =  2.00
R(   5) =  2.00

BOUND=  6.00


CRITICAL VALUE (  0) OF THETA =   .00000000


EPSILON =    .364000-04  CAPITAL EPSILON =    .364000-03      0 NON-ZERO ENTRIES ARE EFFECTIVELY EQUAL TO ZERO.


*************************************************************************************************
*                                                                                               *
*           END OF PHASE 1.    OBJECTIVE FUNCTION =      .74505806-08    THERE WERE   2 ITERATIONS. *
*           MINIMUM PIVOT WAS   1.0000    AT ITERATION   1.   REAL OBJECTIVE FUNCTION =  -1.2676858 *
*                                                                                               *
*************************************************************************************************


*************************************************************************************************
*                                                                                               *
*           END OF PHASE 2.    OBJECTIVE FUNCTION =     -.66359763       THERE WERE   1 ITERATIONS. *
*           MINIMUM PIVOT WAS   1.0000    AT ITERATION   1.                                     *
*                                                                                               *
*************************************************************************************************
BASIC VARIABLES

  X(  1)=     .221199

  S(  1)=    6.538539    S(  2)=    10.489368    S(  3)=    12.377992    S(  5)=     .100681

OBJECTIVE FUNCTION VALUE BETWEEN CRITICAL VALUES (  0) AND (  1) OF THETA OBTAINED FROM  -.85714285+00*THETA    -.66359763+00

 *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *
```



Figure 3: PAROBJ Output
Compressed Format

| CRIT VAL NO. | CRIT VAL | BASIS INFORMATION ENTERS | BASIS INFORMATION LEAVES | OBJ FUNC VAL | OBJ FUNC VAL BETWEEN THIS AND NEXT CRIT VAL |
|---|---|---|---|---|---|
| 1 | .6250 | X( 4) | S( 1) | -.3871 | 5.1977* THETA -3.6357 |
| 2 | .8676 | X( 5) | X( 1) | .8741 | 14.0106* THETA -11.2821 |
| 3 | 1.2500 | S( 4) | S( 3) | 6.2311 | 14.6783* THETA -12.1167 |
| 4 | 1.6111 | NO FURTHER INFORMATION-PROBLEM GOES UNBOUNDED BEYOND THIS CRITICAL VALUE | | | |

Figure 3: PAROBJ Output
Compressed Format (continued)

```
THF F VECTOR IS AS FOLLOWS
F(   1) =  2.00
F(   2) =  1.00
F(   3) =  4.00
F(   4) =  2.00
F(   5) =  2.00
BOUND=  6.00


CR,TICAL VALUE (  0) OF THETA =    .00000000


EPSILON =    .364000-04  CAPITAL EPSILON =    .364000-03      0 NON-ZERO ENTRIES ARE EFFECTIVELY EQUAL TO ZERO.


****************************************************************************************************
*                                                                                                  *
*            END OF PHASE 1.    OBJECTIVE FUNCTION =      .74505806-08    THERE WERE   2 ITERATIONS.  *
*            MINIMUM PIVOT WAS   1.0000    AT ITERATION   1.   REAL OBJECTIVE FUNCTION =  -1.2676858   *
*                                                                                                  *
****************************************************************************************************


****************************************************************************************************
*                                                                                                  *
*            END OF PHASE 2.    OBJECTIVE FUNCTION =     -.66359763       THERE WERE   1 ITERATIONS.  *
*            MINIMUM PIVOT WAS   1.0000    AT ITERATION   1.                                        *
*                                                                                                  *
****************************************************************************************************
BASIC VARIABLES

  X(  1)=     .221199

  S(  1)=    6.538539    S(  2)=   10.489368    S(  3)=   12.377992    S(  5)=     .100681

OBJECTIVE FUNCTION VALUE BETWEEN CRITICAL VALUES (  0) AND (  1) OBTAINED FROM    .44239841+00* THETA    -.66359762+00
```

Figure 4: PARRHS Output
Compressed Format

| CRIT VAL NO. | CRIT VAL | BASIS INFORMATION ENTERS | LEAVES | OBJ FUNC VAL | OBJ FUNC VAL BETWEEN THIS AND NEXT CRIT VAL | |
|---|---|---|---|---|---|---|
| 1 | .0587 | X( 5) | S( 5) | -.7139 | -2.2500* THETA | -.5818 |
| 2 | 1.0106 | X( 4) | X( 1) | -2.8558 | -2.5000* THETA | -.3291 |

CRITICAL VALUE ( 3) OF THETA IS INFINITE.

* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *

Figure 4: PARRHS Output
Compressed Format (continued)

## 6. REFERENCES


1. Hadley, G., Linear Programming, Addison-Wesley, Reading, Mass., 1962.

2. Hall, W.G., Jackson, R.H.F., Saunders, P.B., The National Bureau of Standards' Linear and Quadratic Programming Subroutines, NBS Report No. 10695, February 1972.

3. Simonnard, M., Linear Programming, Prentice Hall, Englewood Cliffs, N.J., 1966.

APPENDIX A: Listing of RANGES

```
@FOR,IS RANGES
FOR 01DA-09/21/73-21:06:00 (,0)

   SUBROUTINE RANGES     ENTRY POINT 001274

   STORAGE USED: CODE(1) 001344; DATA(0) 000234; BLANK COMMON(2) 000000

   EXTERNAL REFERENCES (BLOCK, NAME)

    0003   NWDU$
    0004   N102$
    0005   NEXP1$
    0006   NERR3$

   STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

    0001   000331 1L       0001   000453 106L     0000   000033 11F      0001   000120 12L      0001   000473 120L
    0000   000047 13F      0001   000534 130L     0000   000053 14F      0001   000166 144G     0000   000064 15F
    0001   000555 150L     0001   000174 151G     0001   000312 16L      0001   000227 167G     0001   000322 17L
    0001   000605 170L     0001   000351 18L      0001   000654 180L     0001   000665 190L     0001   000214 2L
    0001   000714 200L     0001   000712 201L     0001   000726 220L     0001   000771 230L     0001   000416 234G
    0001   000424 237G     0001   000447 250G     0001   001012 250L     0001   000470 261G     0001   000510 267G
    0001   001047 270L     0001   001122 280L     0001   001133 290L     0001   001161 300L     0001   001157 301L
    0001   000550 305G     0001   001216 310L     0000   000100 311F     0001   000577 315G     0001   001226 320L
    0000   000112 321F     0001   001243 322L     0001   001235 330L     0000   000124 331F     0001   000723 352G
    0001   000750 360G     0001   001025 375G     0001   001034 405G     0000   000135 500F     0001   000247 7L
    0001   000037 8L       0001   000061 9L       0000 1 000016 1        0000   000145 INJP$    0000 1 000011 J
    0000 1 000023 JTEL     0000 1 000022 J1       0000 1 000024 J2       0000 1 000025 J3       0000 1 000030 J4
    0000 1 000010 K        0000 1 000020 KL       0000 1 000014 KU       0000 1 000006 L1       0000 1 000005 L1OFF
    0000 1 000007 L2       0000 1 000013 L2OFF    0000 1 000000 M        0000 1 000003 MP1      0000 1 000004 MP2
    0000 1 000021 M2       0000 1 000001 N        0000 1 000032 NDWN     0000 1 000012 NPLPG    0000 1 000002 NP1
    0000 1 000031 NUPP     0000 R 000017 RATIO    0000 R 000015 RATMIN   0000 R 000026 XX       0000 R 000027 YY


00101      1*           SUBROUTINE RANGES(A,MA,B,MB,MT,NT,L,X,RHSUP,RHSLOW,COEUP,COELOW,NS  RG1000
00101      2*          1ET)                                                                 RG1005
00103      3*           DIMENSION A(MA,1),B(MB,1),L(1),X(1),RHSUP(1),RHSLOW(1)              RG1010
00104      4*           DIMENSION COEUP(1),COELOW(1)                                        RG1020
00105      5*           IF(L(3).EQ.0.OR.L(3).EQ.1) GO TO 8                                  RG1030
00107      6*           WRITE(6,15)                                                         RG1040
00111      7*           GO TO 322                                                           RG1050
00112      8*         8 M=MT-1                                                              RG1060
00113      9*           N=NT-1                                                              RG1070
00114     10*           NP1=N+1                                                             RG1080
00115     11*           MP1=M+1                                                             RG1090
00116     12*           MP2=M+2                                                             RG1100
00117     13*           L1OFF=14                                                            RG1110
00120     14*           L1=0                                                                RG1120
00121     15*           L2=0                                                                RG1130
00122     16*           K=NP1                                                               RG1140
00123     17*         9 J=L(L1OFF+K)                                                        RG1150
00124     18*           IF(J.EQ.3) GO TO 12                                                 RG1160
```

```
00126         IF(J.EQ.1) L1=L1+1
00130         IF(J.EQ.2) L2=L2+1
00132         IF((L1+L2).EQ.M) GO TO 12
00134         K=K+1
00135         GO TO 9
00136      12 NPLPG=M+L1+L2
00137         L2OFF=L1OFF+M+N+L2
00140         NPLPG=M+L1+L2
00141         IF(NSET.EQ.2) GO TO 18
00141   C ****************************************************
00141   C **BEGIN RHS UPPER BOUND COMPUTATION PROCEEDURE**
00141   C ****************************************************
00143         DO 1 J=1,M
00146         KU=0
00147         RATMIN=1E35
00147   C **CHECK FOR NEGATIVE VALUES IN INVERSE OF BASIS MATRIX**
00150         DO 2 I=1,M
00153         IF (B(I,J).GE.0.0) GO TO 2
00155         KU=1
00155   C **RATIO OF OPTIMAL BASIC SOLUTION TO INVERSE OF BASIS MATRIX**
00156         RATIO=-B(I,MP2)/B(I,J)
00156   C **FIND MINIMUM RATIO**
00157         IF(RATIO.LT.RATMIN) RATMIN=RATIO
00161       2 CONTINUE
00161   C **COMPUTE UPPER BOUND ON RIGHT HAND SIDE**
00163         RHSUP(J)=RATMIN+A(J,NP1)
00163   C ****************************************************
00163   C **BEGIN RHS LOWER BOUND COMPUTATION PROCEEDURE**
00163   C ****************************************************
00164       6 KL=0
00165         RATMIN=1E35
00165   C **CHECK FOR POSITIVE VALUES IN INVERSE OF BASIS MATRIX**
00166         DO 7 I=1,M
00171         IF (B(I,J).LE.0.0) GO TO 7
00173         KL=1
00173   C **RATIO OF OPTIMAL BASIC SOLUTION TO INVERSE OF BASIS MATRIX**
00174         RATIO=B(I,MP2)/B(I,J)
00174   C **FIND MINIMUM RATIO**
00175         IF(RATIO.LT.RATMIN) RATMIN=RATIO
00177       7 CONTINUE
00177   C **COMPUTE LOWER BOUND ON RIGHT HAND SIDE**
00201         RHSLOW(J)=A(J,NP1)-RATMIN
00201   C ********************************
00201   C **BEGIN PRINTING OF RHS BOUNDS**
00201   C ********************************
00202      10 IF(KL.EQ.0.AND.KU.GT.0) GO TO 16
00204         IF(KL.GT.0.AND.KU.EQ.0) GO TO 17
00206         WRITE(6,11)RHSLOW(J),J,RHSUP(J)
00213         GO TO 1
00214      16 WRITE(6,13)J,RHSUP(J)
00220         GO TO 1
00221      17 WRITE(6,14)RHSLOW(J),J
00225       1 CONTINUE
00227         WRITE(6,500)
00231         IF(NSET.EQ.1) GO TO 322
00231   C ************************************************************
00231   C **BEGIN OBJECTIVE FUNCTION UPPER BOUND COMPUTATION PROCEDURE
00231   C ************************************************************
```

```
00233       18 DO 330 J=1,N
00236          DO 100 I=1,M
00241      100 B(MP2,I)=0.0
00243          M2=0
00243    C *******************************************************
00243    C **STORE APPROPRIATE ROW OF B INVERSE IN ROW M+2 OF B**
00243    C *******************************************************
00244          IF(X(J).LT.0.0) GO TO 120
00246          M2=1
00247          DO 105 J1=1,M
00252          K=L(L2OFF+J1)
00253          JTEL=J1
00254          IF(K.EQ.J) GO TO 106
00256      105 CONTINUE
00260      106 DO 110 J2=1,M
00263      110 B(MP2,J2)=B(JTEL,J2)
00263    C **COMPUTE INDICATORS AND DENOMINATORS**
00265      120 RATMIN=1E35
00266          DO 190 J2=1,NPLPG
00271          IF(X(J2).GE.0.0) GO TO 190
00273          J3=L(L1OFF+J2)
00274          IF(J3.GT.0) GO TO 170
00274    C **REAL VARIABLES**
00276          XX=0.0
00277          IF(J2.EQ.J) XX=-1.0
00301          IF(M2.EQ.1) GO TO 130
00303          GO TO 150
00304      130 DO 140 I=1,M
00307      140 XX=XX+B(MP2,I)*A(I,J2)
00311      150 IF(XX.GE.0.0) GO TO 190
00313          YY=A(MP1,J2)
00314          DO 160 I=1,M
00317      160 YY=YY+B(MP1,I)*A(I,J2)
00321          GO TO 180
00322      170 J4=J2-N
00322    C **SLACK AND SURPLUS VARIABLES**
00323          XX=0.0
00324          IF(M2.EQ.1) XX=(-1)**(J3+1)*B(MP2,J4)
00326          IF(XX.GE.0.0) GO TO 190
00330          YY=(-1)**(J3+1)*B(MP1,J4)
00330    C **COMPUTE RATIO OF INDICATORS TO DENOMINATORS**
00331      180 RATIO=YY/XX
00331    C **FIND MINIMUM RATIO**
00332          IF(RATIO.GE.RATMIN) GO TO 190
00334          RATMIN=RATIO
00335      190 CONTINUE
00337          NUPP=0
00340          IF(RATMIN.GT.1E34) NUPP=1
00342          IF(NUPP.EQ.1) GO TO 201
00342    C **COMPUTE UPPER BOUND ON COEFFICIENT**
00344          COEUP(J)=RATMIN+A(MP1,J)
00345          GO TO 200
00346      201 COEUP(J)=RATMIN
00346    C ****************************************************************
00346    C **BEGIN OBJECTIVE FUNCTION LOWER BOUND COMPUTATION PROCEDURE
00346    C ****************************************************************
00347      200 IF(M2.EQ.0) GO TO 220
00347    C **CHANGE SIGN OF ROW M+2 OF B**
```

```
00351              DO 210 JI=1,M
00354          210 B(MP2,JI)=-B(MP2,J))
00354        C **COMPUTE INDICATORS AND DENOMINATORS**
00356          220 RATMIN=1E35
00357              DO 290 J2=1,NPLPG
00362              IF(X(J2).GE.0.0) GO TO 290
00364              IF(J3.GT.0) GO TO 270
00364        C **REAL VARIABLES**
00366              XX=0.0
00367              IF(J2.EQ.J) XX=1.0
00371              IF(M2.EQ.1) GO TO 230
00373              GO TO 250
00374          230 DO 240 I=1,M
00377          240 XX=XX+B(MP2,I)*A(I,J2)
00401          250 IF(XX.GE.0.0) GO TO 290
00403              YY=A(MP1,J2)
00404              DO 260 I=1,M
00407          260 YY=YY+B(MP1,I)*A(I,J2)
00411              GO TO 280
00412          270 J4=J2-N
00412        C **SLACK AND SURPLUS VARIABLES**
00413              XX=0.0
00414              IF(M2.EQ.1) XX=(-1)**(J3+1)*B(MP2,J4)
00416              IF(XX.GE.0.0) GO TO 290
00420              YY=(-1)**(J3+1)*B(MP1,J4)
00420        C **COMPUTE RATIO OF INDICATORS TO DENOMINATORS**
00421          280 RATIO=YY/XX
00421        C **FIND MINIMUM RATIO**
00422              IF(RATIO.GE.RATMIN) GO TO 290
00424              RATMIN=RATIO
00425          290 CONTINUE
00427              NDWN=0
00430              IF(RATMIN.GT.1E34) NDWN=1
00432              IF(NDWN.EQ.1) GO TO 301
00432        C **COMPUTE LOWER BOUND ON COEFFICIENT**
00434              COELOW(J)=-RATMIN+A(MP1,J)
00435              GO TO 300
00436          301 COELOW(J)=-RATMIN
00436        C ***********************************************
00436        C **BEGIN PRINTING OF OBJECTIVE FUNCTION BOUNDS**
00436        C ***********************************************
00437          300 IF(NUPP.EQ.0.AND.NDWN.EQ.1) GO TO 310
00441              IF(NUPP.EQ.1.AND.NDWN.EQ.0) GO TO 320
00443              WRITE(6,331) COELOW(J),J,COFUP(J)
00450              GO TO 330
00451          310 WRITE(6,311) J,COEUP(J)
00455              GO TO 330
00456          320 WRITE(6,321) COELOW(J),J
00462          330 CONTINUE
00464           11 FORMAT(E14.8,2X,'LE   RHS',I3,'   LE',E14.8)
00465           13 FORMAT('  NEG.INFINITY  LE   RHS',I3,'   LE',E14.8)
00466           14 FORMAT(E14.8,2X,'LE   RHS',I3,'   LE  POS.INFINITY')
00467           15 FORMAT(' NO RANGING DONE DUE TO EITHER INFEASIBILITY OR UNBOUNDEDN
00467             1ESS.')
00470          311 FORMAT('  NEG.INFINITY  LE   COEFFICIENT',I3,'   LE',E14.8)
00471          321 FORMAT(E14.8,2X,'LE   COEFFICIENT',I3,'   LE  PO#.INFINI,Y')
00472          331 FORMAT(E14.8,2X,'LE   COEFFICIENT',I3,'   LE',F14.8)
00473          500 FORMAT(/////)
```

```
00474    193*       322 RETURN                                        RG2890
00475    194*           END                                           RG2900

      END OF COMPILATION:        NO  DIAGNOSTICS.

@ FIN
```

APPENDIX B: Listing of PAROBJ

```
@FOR,IS PAROBJ
FOR 010A-09/20/73-11:43:01 (,0)

   SUBROUTINE PAROBJ    ENTRY POINT 002743

   STORAGE USED: CODE(1) 003156; DATA(0) 000533; BLANK COMMON(2) 000000

   EXTERNAL REFERENCES (BLOCK, NAME)

    0003   RVSMPX
    0004   NWDU$
    0005   NI02$
    0006   NI01$
    0007   NEXP1$
    0010   NERR3$

   STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

    0000   000126 10F      0001   000415 100L     0001   001736 1001L    0001   002021 1002L    0001   002073 1003L
    0000   000277 1004F    0000   000320 1005F    0000   000337 1006F    0000   000354 1007F    0000   000362 1009F
    0000   000366 1010F    0001   002411 1021G    0001   002323 1099L    0001   000157 11L      0001   002335 1100L
    0001   002617 1101G    0001   002346 1101L    0000   000371 1102F    0000   000406 1103F    0000   000413 1104F
    0000   000420 1105F    0001   002660 1123G    0001   002707 1130G    0001   000532 130L     0001   000130 143G
    0001   000131 146G     0001   000203 160G     0001   000217 167G     0000   000127 201F     0001   000335 226G
    0001   000670 240L     0001   000411 254G     0001   000753 260L     0001   000470 261G     0001   000476 267G
    0001   000764 270L     0001   000525 275G     0001   001016 300L     0001   000551 307G     0001   000602 320G
    0001   000623 327G     0001   001134 330L     0001   000626 333G     0001   001202 340L     0001   000657 345G
    0001   001221 350L     0001   000733 361G     0000   000051 4F       0001   001250 400L     0000   000143 401F
    0000   000157 402F     0001   001011 407G     0001   001054 416G     0001   001105 427G     0001   001126 437G
    0000   000062 5F       0001   001340 500L     0001   001333 520G     0001   001361 526G     0001   001457 543G
    0001   001463 547G     0001   001541 557G     0001   001573 566G     0000   000063 6F       0001   001422 600L
    0001   001703 617G     0001   001477 620L     0000   000064 7F       0001   001545 700L     0001   002114 706G
    0001   001614 710L     0001   001755 720L     0001   001762 730L     0001   001772 740L     0001   002004 750L
    0000   000204 751F     0001   002130 760L     0001   002277 762G     0001   000351 8L       0001   002304 800L
    0001   002425 810L     0000   000101 9F       0001   002624 900L     0000   000220 901F     0000   000230 902F
    0000   000241 903F     0000   000254 904F     0000   000266 905F     0000   000270 906F     0001   002632 910L
    0001   002640 920L     0001   002646 930L     0001   002664 999L     0000 R 000016 CRIT     0000 R 000040 DENOM
    0000 R 000037 ENUM     0000 R 000021 EPS      0000 I 000024 I        0000   000456 INJP$    0000 I 000030 ISG
    0000 I 000025 J        0000 I 000033 J1       0000 I 000035 J2       0000 I 000044 J5       0000 I 000017 KT
    0000 I 000020 KT1      0000 I 000032 K1       0000 I 000042 K3       0000 I 000046 LEAVE    0000 I 000047 LET1
    0000 I 000050 LET2     0000 I 000006 L1       0000 I 000014 L1OFF    0000 I 000007 L2       0000 I 000015 L2OFF
    0000 I 000010 L3       0000 I 000001 M        0000 I 000002 MP1      0000 I 000003 MP2      0000 I 000031 M2
    0000 I 000000 N        0000 I 000011 NPL      0000 I 000013 NPLPG    0000 I 000012 NPLP1    0000 I 000004 NP1
    0000 I 000041 NTER     0000 R 000026 R        0000 R 000036 RATIO    0000 R 000034 RATMIN   0000 R 000027 S
    0000 R 000043 SMALL    0000 R 000005 STAR     0000 R 000045 TEST     0000 R 000022 XX       0000 R 000023 YY


00101      1*          SUBROUTINE PAROBJ(A,MA,B,MB,MT,NT,L,X,TOLP,INV,F,BOUND,NSET)       OB1000
00103      2*          DIMENSION A(MA,1),B(MB,1),L(1),X(1),F(1)                            OB1010
00104      3*          N=NT-1                                                               OB1020
00105      4*          M=MT-2                                                               OB1030
00106      5*          MP1=M+1                                                              OB1040
```

```
00107                 MP2=MP1+1                                            OB1050
00110                 NP1=NT                                               OB1060
00111                 MT=MP1                                               OB1070
00112                 STAR='  *'                                           OB1071
00113                 L1=L(1)                                              OB1080
00114                 L2=L(2)                                              OB1090
00115                 L3=L(3)                                              OB1100
00116                 L(5)=1                                               OB1110
00117                 L(8)=1                                               OB1120
00120                 L(12)=1                                              OB1130
00121                 L(13)=1                                              OB1140
00122                 L(14)=1                                              OB1150
00123                 NPL=N+L1                                             OB1160
00124                 NPLP1=N+L1+1                                         OB1170
00125                 NPLPG=N+L1+L2                                        OB1180
00126                 L1OFF=14                                             OB1190
00127                 L2OFF=L1OFF+M+N+L2                                   OB1200
00130                 CRIT=0.0                                             OB1210
00131                 KT=1                                                 OB1211
00132                 KT1=0                                                OB1212
00133                 EPS=TOLP                                             OB1220
00134                 IF(EPS.GT.0.0) GO TO 11                              OB1230
00136                 XX=1.0E-5                                            OB1240
00137                 IF(EPS.LT.0.0) XX=-EPS                               OB1250
00141                 YY=0.0                                               OB1260
00142                 DO 1 I=1,M                                           OB1270
00145                 DO 1 J=1,M                                           OB1280
00150                 EPS=A(I,J)                                           OB1290
00151                 IF(EPS.LT.0.0) EPS=-EPS                              OB1300
00153               1 YY=YY+EPS                                            OB1310
00156                 EPS=XX*YY/(M*N)                                      OB1311
00156           C  **STORE ORIGINAL CJ'S**                                 OB1320
00157              11 DO 3 J=1,N                                           OB1330
00162               3 A(MP2,J)=A(MP1,J)                                    OB1340
00164                 WRITE(6,1007)                                        OB1341
00166                 DO 1008 I=1,N                                        OB1342
00171            1008 WRITE(6,1009) I,F(I)                                 OB1343
00176                 WRITE(6,1010) BOUND                                  OB1344
00201                 WRITE(6,6)                                           OB1345
00203                 WRITE(6,4) KT1,CRIT                                  OB1350
00207                 WRITE(6,5)                                           OB1360
00207           C  **SOLVE ORIGNAL PROBLEM**                               OB1370
00211                 CALL RVSMPX(A,MA,B,MB,MT,NT,L,X,EPS,INV,0)           OB1380
00212                 WRITE(6,6)                                           OB1390
00214                 IF(L(3).EQ.1) WRITE(6,7)                             OB1400
00217                 IF(L(3).EQ.2) GO TO 910                              OB1410
00221                 IF(L(3).EQ.4) GO TO 920                              OB1420
00221           C **COMPUTE C + THETA*F FOR OBJECTIVE FUNCTION VALUE**     OB1430
00223                 R=0.0                                                OB1440
00224                 S=0.0                                                OB1450
00225                 DO 8 I=1,N                                           OB1460
00230                 IF (X(I).LE.0.0) GO TO 8                             OB1470
00232                 R=R+A(MP2,I)*X(I)                                    OB1480
00233                 S=S+F(I)*X(I)                                        OB1490
00234               8 CONTINUE                                             OB1500
00236                 ISG='+'                                              OB1510
00237                 IF(R.LT.0.0) ISG=' '                                 OB1520
00241                 WRITE(6,9) KT1,KT,S,ISG,R                            OB1530
```

```
00250        WRITE(6,10)                                                  OB1540
00250     C **COMPUTE FR*B IN ORDER OF BASIC VARIABLES**                   OB1550
00252        WRITE(6,905) (STAR,I=1,40)                                   OB1551
00260     100 DO 110 I=1,M                                                OB1560
00263     110 B(MP2,I)=0.0                                                OB1570
00265        M2=0                                                         OB1580
00266        DO 130 J=1,M                                                 OB1590
00271        K1=L(L2OFF+J)                                                OB1600
00272        IF(K1.GT.N) GO TO 130                                        OB1601
00274        DO 120 J1=1,M                                                OB1610
00277     120 B(MP2,J1)=B(MP2,J1)+(F(K1)*B(J,J1))                         OB1620
00301     130 CONTINUE                                                    OB1630
00301     C *******************************************************       OB1640
00301     C **COMPUTE INDICATORS, DENOMINATORS, AND MINIMUM RATIO**       OB1650
00301     C **         FOR UNROUNDED ORIGINAL PROBLEM               **    OB1660
00301     C *******************************************************       OB1670
00303        IF(L(3).NE.3) GO TO 300                                      OB1680
00305        RATMIN=1E35                                                  OB1690
00306        DO 270 J=1,NPLPG                                             OB1700
00311        IF(X(J).GE.0.0) GO TO 270                                    OB1710
00313        J1=L(L1OFF+J)                                                OB1720
00314        IF(J1.GT.0) GO TO 240                                        OB1730
00314     C **REAL VARIABLES**                                            OB1740
00316        YY=A(MP1,J)                                                  OB1750
00317        DO 200 I=1,M                                                 OB1760
00322     200 YY=YY+B(MP1,I)*A(I,J)                                       OB1770
00324        IF(YY.LE.EPS) GO TO 270                                      OB1780
00326        DO 220 I=1,M                                                 OB1790
00331        XX=0.0                                                       OB1800
00332        DO 210 J1=1,M                                                OB1810
00335     210 XX=XX+B(I,J1)*A(J1,J)                                       OB1820
00337        IF(XX.GT.0.0) GO TO 270                                      OB1830
00341     220 CONTINUE                                                    OB1840
00343        XX=-F(J)                                                     OB1850
00344        DO 230 I=1,M                                                 OB1860
00347     230 XX=XX+B(MP2,I)*A(I,J)                                       OB1870
00351        IF(XX.LE.0.0) GO TO 270                                      OB1880
00353        GO TO 260                                                    OB1890
00353     C **SLACK AND SURPLUS VARIABLES**                               OB1900
00354     240 J2=J-N                                                      OB1910
00355        YY=(-1)**(J1+1)*B(MP1,J2)                                    OB1920
00356        IF(YY.LE.EPS) GO TO 270                                      OB1930
00360        DO 250 I=1,M                                                 OB1940
00363        XX=(-1)**(J1+1)*B(I,J2)                                      OB1950
00364        IF(XX.GT.0.0) GO TO 270                                      OB1960
00366     250 CONTINUE                                                    OB1970
00370        XX=(-1)**(J1+1)*B(MP2,J2)                                    OB1980
00371        IF(XX.LE.0.0) GO TO 270                                      OB1990
00371     C **FIND MINIMUM RATIO**                                        OB2000
00373     260 YY=YY/XX                                                    OB2010
00374        IF(YY.GE.RATMIN) GO TO 270                                   OB2020
00376        RATMIN=YY                                                    OB2030
00377     270 CONTINUE                                                    OB2040
00401        IF(RATMIN.LE.1E34) GO TO 400                                 OB2050
00403        WRITE(6,201)                                                 OB2060
00405        WRITE(6,905) (STAR,I=1,40)                                   OB2061
00413        GO TO 999                                                    OB2070
00413     C *******************************************************       OB2080
```

```
004,3    C **COMPUTE INDICATORS, DENOMINATORS, AND MINIMUM RATIO**      OB2090
004,3    C **          FOR ROUNDED ORIGINAL PROBLEM               **      OB2100
004,3    C ******************************************************      OB2110
004,4      300 RATMIN=1E35                                             OB2120
004,5          DO 350 J=1,NPLPG                                        OB2130
00420          IF(X(J).GE.0.0) GO TO 350                               OB2140
00422          J1=L(L1OFF+J)                                           OB2150
00423          IF(J1.GT.0) GO TO 330                                   OB2160
00423    C **REAL VARIABLES**                                          OB2170
00425          XX=-F(J)                                                OB2180
00426          DO 310 I=1,M                                            OB2190
00431      310 XX=XX+B(MP2,I)*A(I,J)                                   OB2200
00433          IF(XX.GE.0.0) GO TO 350                                 OB2210
00435          YY=A(MP1,J)                                             OB2220
00436          DO 320 I=1,M                                            OB2230
00441      320 YY=YY+B(MP1,I)*A(I,J)                                   OB2240
00443          GO TO 340                                               OB2250
00443    C **SLACK AND SURPLUS VARIABLES**                             OB2260
00444      330 J2=J-N                                                  OB2270
00445          XX=(-1)**(J1+1)*B(MP2,J2)                               OB2280
00446          IF(XX.GE.0.0) GO TO 350                                 OB2290
00450          YY=(-1)**(J1+1)*B(MP1,J2)                               OB2300
00450    C **FIND MINIMUM RATIO**                                      OB2310
00451      340 RATIO=YY/XX                                             OB2320
00452          IF(RATIO.GE.RATMIN) GO TO 350                           OB2330
00454          RATMIN=RATIO                                            OB2340
00455          ENUM=YY                                                 OB2350
00456          DENOM=XX                                                OB2360
00457          NTER=J                                                  OB2370
00460      350 CONTINUE                                                OB2380
00462          IF(RATMIN.GT.1E34) GO TO 900                            OB2390
00464          IF(RATMIN.GT.EPS) GO TO 400                             OB2400
00466          M2=1                                                    OB2410
00467          RATMIN=-((1.2*EPS)-ENUM)/DENOM                          OB2420
00467    C **ACCUMULATE CRITICAL VALUE**                               OB2430
00470      400 CRIT=CRIT+RATMIN                                        OB2440
00471          IF(L(3).EQ.3) WRITE(6,401) CRIT                         OB2450
00475          IF(CRIT.LE.BOUND) GO TO 500                             OB2460
00477          WRITE(6,402) BOUND                                      OB2470
00502          ISG='+'                                                 OB2480
00503          IF(R.LT.0.0) ISG=' '                                    OB2490
00505          KT1=KT                                                  OB2500
00506          KT=KT+1                                                 OB2501
00507          WRITE(6,9) KT1,KT,S,ISG,R                               OB2502
005,6          WRITE(6,905) (STAR,I=1,40)                              OB2503
005,4          GO TO 999                                               OB2510
00524    C *******************************************************      OB2520
00524    C **UPDATE OBJECTIVE FUNCTION WITH NEW THETA AND F VECTOR**    OB2530
00524    C *******************************************************      OB2540
00525      500 DO 510 J=1,N                                            OB2550
00530          A(MP1,J)=A(MP1,J)+(RATMIN*F(J))                         OB2560
00531      510 CONTINUE                                                OB2570
00533          IF(L(3).EQ.3.OR.M2.EQ.1) GO TO 800                      OB2580
00533    C ***************************************************      OB2590
00533    C **GET APPROPRIATE Y(J) AND STORE IN COLUMN MP1 OF B**      OB2600
00533    C ***************************************************      OB2610
00535          K3=L(L1OFF+NTER)                                        OB2620
00536          IF(K3.EQ.0) GO TO 600                                   OB2630
```

```
00540           IF(K3.EQ.1.OR.K3.EQ.2) GO TO 620                    OB2640
00542       600 DO 610 I=1,M                                        OB2650
00545           B(I,MP1)=0.0                                        OB2660
00546           DO 610 J1=1,M                                       OB2670
00551       610 B(I,MP1)=B(I,MP1)+B(I,J1)*A(J1,NTER)                OB2680
00554           GO TO 700                                           OB2690
00555       620 J2=NTER-N                                           OB2700
00556           DO 630 I=1,M                                        OB2710
00561       630 B(I,MP1)=(-1)**(K3+1)*B(I,J2)                       OB2720
00561     C ********************************************            OB2730
00561     C **FIND THE LEAVING AND ENTERING VARIABLES**             OB2740
00561     C ********************************************            OB2750
00563       700 SMALL=1E35                                          OB2760
00564           J5=0                                                OB2770
00565           DO 710 J=1,M                                        OB2780
00570           IF(B(J,MP1).LE.0.0) GO TO 710                       OB2790
00572           J5=1                                                OB2800
00573           TEST=B(J,MP2)/B(J,MP1)                              OB2810
00574           IF(TEST.GT.SMALL) GO TO 710                         OB2820
00576           SMALL=TEST                                          OB2830
00577           LEAVE=L(L2OFF+J)                                    OB2840
00600       710 CONTINUE                                            OB2850
00602           KT1=KT                                              OB2860
00603           KT=KT+1                                             OB2861
00604           IF(NSET.EQ.1.AND.J5.EQ.0) WRITE(6,5)                OB2862
00607           IF(NSET.EQ.1.AND.J5.EQ.0) WRITE(6,1105)KT1,CRIT     OB2863
00614           IF(NSET.EQ.1.AND.J5.EQ.0) WRITE(6,905) (STAR,I=1,40) OB2864
00623           IF(NSET.EQ.1.AND.J5.EQ.0) GO TO 999                 OB2865
00625           IF(NSET.EQ.1) GO TO 1001                            OB2866
00627           WRITE(6,4) KT1,CRIT                                 OB2867
00633           IF(J5.EQ.0) GO TO 930                               OB2870
00633     C **SET X VECTOR FOR RVSMPX WITH ADVANCED START**         OB2880
00635      1001 X(LEAVE)=-1.0                                       OB2890
00636           X(NTER)=1.0                                         OB2900
00637           IF(NTER.GT.N) GO TO 720                             OB2910
00641           LET1=' X('                                          OB2920
00642           GO TO 730                                           OB2930
00643       720 LET1=' S('                                          OB2940
00644           NTER=NTER-N                                         OB2950
00645       730 IF(LEAVE.GT.N) GO TO 740                            OB2960
00647           LET2=' X('                                          OB2970
00650           GO TO 750                                           OB2980
00651       740 LET2=' S('                                          OB2990
00652           LEAVE=LEAVE-N                                       OB3000
00653           IF(NSET.EQ.1) L(13)=0                               OB3002
00655       750 IF(NSET.EQ.1) GO TO 1002                            OB3004
00657           WRITE(6,751) LET1,NTER,LET2,LEAVE                   OB3010
00665      1002 L(1)=L1                                             OB3020
00666           L(2)=L2                                             OB3030
00667           L(3)=0                                              OB3040
00670           L(4)=0                                              OB3050
00671           L(12)=0                                             OB3051
00672           INV=1                                               OB3060
00673           CALL RVSMPX(A,MA,B,MB,MT,NT,L,X,EPS,INV,0)          OB3070
00674           WRITE(6,6)                                          OB3080
00676           IF(NSET.EQ.1) GO TO 1003                            OB3082
00700           WRITE(6,906) X(M+N+L2+1)                            OB3084
00703      1003 R=0.0                                               OB3090
```

```
00704              S=0.0                                                    OB3100
00705              DO 760 I=1,N                                             OP3110
00710              IF(X(I).LE.0.0) GO TO 760                                OB3120
00712              R=R+A(MP2,I)*X(I)                                        OB3130
00713              S=S+F(I)*X(I)                                            OB3140
00714          760 CONTINUE                                                 OB3150
00716              ISG='+'                                                  OB3160
00717              IF(R.LT.0.0) ISG=' '                                     OB3170
00721              IF(NSET.EQ.1.AND.KT1.EQ.1) WRITE(6,1004)                 OB3172
00724              IF(NSET.EQ.1.AND.KT1.EQ.1) WRITE(6,1005)                 OB3174
00727              IF(NSET.EQ.1.AND.KT1.EQ.1) WRITE(6,5)                    OB3176
00732              IF(NSET.EQ.1) WRITE(6,1006)KT1,CRIT,LET1,NTFR,LET2,LEAVE,X(M+N+L2+ OB3177
00732             11),S,ISG,R                                               OB3178
00747              IF(NSET.EQ.1) GO TO 100                                  OB3179
00751              WRITE(6,9) KT1,KT,S,ISG,R                                OB3180
00760              WRITE(6,905) (STAR,I=1,40)                               OB3190
00766              GO TO 100                                                OB3200
00767          800 KT1=KT                                                   OB3210
00770              KT=KT+1                                                  OB3211
00771              IF(NSET.EQ.1) GO TO 1099                                 OB3212
00773              WRITE(6,4) KT1,CRIT                                      OB3213
00777         1099 L(1)=L1                                                  OB3220
01000              L(2)=L2                                                  OP3230
01001              IF(NSET.EQ.1) GO TO 1100                                 OB3231
01003              L(3)=L3                                                  OB3240
01004         1100 INV=1                                                    OB3242
01005              IF(NSET.NE.1) GO TO 1101                                 OB3244
01007              L(3)=0                                                   OB3246
01010              L(4)=0                                                   OB3248
01011              L(12)=0                                                  OB3250
01012              L(13)=0                                                  OB3252
01013         1101 CALL RVSMPX(A,MA,B,MB,MT,NT,L,X,EPS,INV,0)               OB3260
01014              WRITE(6,5)                                               OB3270
01016              R=0.0                                                    OB3280
01017              S=0.0                                                    OB3290
01020              DO 810 I=1,N                                             OB3300
01023              IF(X(I).LE.0.0) GO TO 810                                OB3310
01025              R=R+A(MP2,I)*X(I)                                        OB3320
01026              S=S+F(I)*X(I)                                            OB3330
01027          810 CONTINUE                                                 OB3340
01031              ISG='+'                                                  OB3350
01032              IF(R.LT.0.0) ISG=' '                                     OB3360
01034              IF(NSET.EQ.1.AND.KT1.EQ.1) WRITE(6,1004)                 OB3361
01037              IF(NSET.EQ.1.AND.KT1.EQ.1) WRITE(6,1005)                 OB3362
01042              IF(NSET.EQ.1.AND.KT1.EQ.1) WRITE(6,5)                    OB3363
01045              IF(NSET.EQ.1) WRITE(6,1102) KT1,CRIT,X(M+N+L2+1),S,ISG,R OB3364
01056              IF(NSET.EQ.1) WRITE(6,1103)                              OB3365
01061              IF(NSET.EQ.1) WRITE(6,1104)                              OB3366
01064              IF(NSET.EQ.1) GO TO 100                                  OB3367
01066              WRITE(6,9) KT1,KT,S,ISG,R                                OB3370
01075              WRITE(6,10)                                              OB3380
01077              WRITE(6,905) (STAR,I=1,40)                               OB3381
01105              GO TO 100                                                OB3390
01106          900 WRITE(6,901)                                             OB3400
01110              GO TO 999                                                OB3410
01111          910 WRITE(6,902)                                             OB3420
01113              GO TO 999                                                OB3430
01114          920 WRITE(6,903)                                             OB3440
```

```
01116    296*          GO TO 999                                                      0B3450
01117    297*      930 WRITE(6,904)                                                   0B3460
01121    298*          WRITE(6,905) (STAR,I=1,40)                                     0B3461
01127    299*      999 DO 1000 J=1,N                                                  0B3470
01132    300*          A(MP1,J)=A(MP2,J)                                              0B3480
01133    301*     1000 CONTINUE                                                       0B3490
01135    302*        4 FORMAT(1H0,'CRITICAL VALUE (',I3,') OF THETA = ',E14.8)        0B3500
01136    303*        5 FORMAT(//)                                                     0B3510
01137    304*        6 FORMAT(/)                                                      0B3520
01140    305*        7 FORMAT(' NUMERICAL DIFFICULTIES IN RVSMPX.  QUESTION RESULTS OF PA 0B3530
01140    306*         1RAMETRIC STUDY')                                               0B3540
01141    307*        9 FORMAT(' OBJECTIVE FUNCTION VALUE BETWEEN CRITICAL VALUES (',I3,') 0B3550
01141    308*         1 AND (',I3,') OBTAINED FROM ',E15.8,'* THETA 'A1,E15.8)        0B3560
01142    309*       10 FORMAT(///)                                                    0B3570
01143    310*      201 FORMAT(' PROBLEM REMAINS UNBOUNDED THROUGHOUT RANGE OF CRITICAL VA 0B3580
01143    311*         1LUES')                                                         0B3590
01144    312*      401 FORMAT(' WHEN THETA =',F15.10,'YOUR PROBLEM BECOMES BOUNDED AND PR 0B3600
01144    313*         1OCEEDS')                                                       0B3610
01145    314*      402 FORMAT(' CRITICAL VALUE OF THETA BYPASSES YOUR LIMITATION (BOUND), 0B3620
01145    315*         1 THEREFORE THIS BASIS REMAINS OPTIMAL AT YOUR LIMIT OF ',F10.4) 0B3630
01146    316*      751 FORMAT(' AT THAT VALUE,'A3,I4,') ENTERS THE BASIS AND'A3,I4,') LEA 0B3640
01146    317*         1VES.')                                                         0B3650
01147    318*      901 FORMAT(' THE CRITICAL VALUE OF THETA IS INFINITE.')            0B3660
01150    319*      902 FORMAT(' PROBLEM IS INFEASIBLE.  NO STUDY PERFORMED')          0B3670
01151    320*      903 FORMAT(' SYSTEM ERROR ENCOUNTERED BY RVSMPX.  NO STUDY PERFORMED') 0B3680
01152    321*      904 FORMAT(' PROBLEM GOES UNBOUNDED BEYOND THIS CRITICAL VALUE.')  0B3690
01153    322*      905 FORMAT(/40A3)                                                  0B3691
01154    323*      906 FORMAT(' OBJECTIVE FUNCTION VALUE =', E14.8)                   0B3692
01155    324*     1004 FORMAT(' CRIT VAL     CRIT VAL     BASIS INFORMATION     OBJ FUNC  0B3693
01155    325*         1VAL     OBJ FUNC VAL BETWEEN THIS')                            0B3694
01156    326*     1005 FORMAT('    NO.                         ENTERS     LEAVES',15X,'   0B3695
01156    327*         1    AND NEXT CRIT VAL')                                        0B3696
01157    328*     1006 FORMAT(I6,F15.4,5X,A3,I4,')',3X,A3,I4,')',F14.4,6X,F10.4,'* THETA  0B3697
01157    329*         1',A1,F10.4)                                                    0B3698
01160    330*     1007 FORMAT(' THE F VECTOR IS AS FOLLOWS')                          0B3700
01161    331*     1009 FORMAT(' F(',I4,') =',F6.2)                                    0B3710
01162    332*     1010 FORMAT(' BOUND=',F6.2)                                         0B3720
01163    333*     1102 FORMAT(I6,F15.4,5X,' NO REPORT - THETA',F14.4,6X,F10.4,'* THETA ', 0B3730
01163    334*         1A1,F10.4)                                                      0B3740
01164    335*     1103 FORMAT(26X,' LESS THAN EPS OR ')                               0B3750
01165    336*     1104 FORMAT(26X,' ORIG PROB UNBOUNDED')                             0B3760
01166    337*     1105 FORMAT(I6,F15.4,5X,' NO FURTHER INFORMATION-PROBLEM GOES UNBOUNDED
01166    338*         1 BEYOND THIS CRITICAL VALUE')
01167    339*          MT=MT+1                                                        0B3770
01170    340*          RETURN                                                         0B3780
01171    341*          END                                                            0B3790

        END OF COMPILATION:        NO  DIAGNOSTICS.


@XQT
MAP 0023-09/20-11:43
```

APPENDIX C: Listing of PARRHS

```
@FOR,IS PARPHS
FOR C10A-09/21/73-19:31:09 (,0)

   SUBROUTINE PARRHS     ENTRY POINT 002027

   STORAGE USED: CODE(1) 002233; DATA(0) 000447; BLANK COMMON(2) 000000

   EXTERNAL REFERENCES (BLOCK, NAME)

    0003    RVSMPX
    0004    NWDU$
    0005    NI02$
    0006    NI01$
    0007    NFXPI$
    0010    NFRR3$

   STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

    0000    000060 10F       0000    000276 1001F     0000    000304 1003F     0000    000310 1004F     0000    000313 1005F
    0001    001063 1006L     0001    001206 1007L     0001    001556 1008L     0000    000330 1009F     0000    000351 1010F
    0001    000141 133G      0000    000101 14F       0001    000155 140G      0001    001716 15L       0001    000174 151G
    0000    000103 16F       0000    000121 17F       0000    000132 18F       0000    000144 19F       0001    000622 199L
    0001    000626 200L      0001    000323 206G      0001    000345 217G      0001    000354 224G      0001    000726 230L
    0000    000164 231F      0001    000441 254G      0001    000524 266G      0001    000547 277G      0000    000045 3F
    0001    001130 300L      0000    000212 301F      0001    001676 302L      0001    000557 304G      0001    001173 310L
    0001    000615 320G      0001    000664 330G      0001    000670 334G      0001    000707 345G      0000    000056 4F
    0001    001057 414G      0001    001106 423G      0001    001166 453G      0001    001243 470G      0000    000057 5F
    0001    001273 501G      0001    001314 511G      0001    001427 540G      0001    001322 540L      0001    001454 546G
    0001    001370 550L      0001    001403 560L      0000    000236 561F      0001    001434 570L      0001    001711 642G
    0001    001745 657G      0001    001772 664G      0001    000446 8L        0001    000454 9L        0001    001505 900L
    0001    001512 910L      0001    001522 920L      0001    001527 930L      0000    000253 931F      0000    000251 950F
    0000    000267 951F      0001    001751 999L      0000 R  000026 CBY       0000 R  000030 COMPUT    0000 R  000007 CRIT
    0000 I  000021 I         0000    000403 INJP$     0000 I  000031 ISG       0000 I  000022 J         0000 I  000041 JI
    0000 I  000044 J2        0000 I  000027 K         0000 I  000035 KLEAVE    0000 I  000023 KT        0000 I  000024 KTI
    0000 I  000037 LEAVE     0000 I  000032 LETI      0000 I  000034 LET2      0000 I  000010 LI        0000 I  000017 LIOFF
    0000 I  000011 L2        0000 I  000020 L2OFF     0000 I  000012 L3        0000 I  000000 M         0000 I  000003 MPI
    0000 I  000002 MP2       0000 I  000001 N         0000 I  000016 NMGI      0000 I  000013 NPL       0000 I  000015 NPLPG
    0000 I  000014 NPLPI     0000 I  000004 NPI       0000 I  000005 NP2       0000 I  000033 NTER      0000 R  000036 RATIO
    0000 R  000040 RATMAX    0000 R  000025 RATMIN    0000 R  000006 STAR      0000 R  000042 XX        0000 R  000043 YY


00101       1*          SUBROUTINE PARRHS(A,MA,B,MB,MT,NT,L,X,TOLP,INV,BOUND,R,NSET)        RH1000
00103       2*          DIMENSION A(MA,1),B(MB,1),L(1),X(1),R(1)                            RH1010
00104       3*          M=MT-2                                                              RH1020
00105       4*          N=NT-2                                                              RH1030
00106       5*          MP2=M+2                                                             RH1040
00107       6*          MPI=M+1                                                             RH1050
00110       7*          NPI=N+1                                                             RH1060
00111       8*          NP2=N+2                                                             RH1070
00112       9*          STAR='  *'                                                          RH1071
00113      10*          CRIT=0.0                                                            RH1080
00114      11*          LI=L(1)                                                             RH1090
```

```
00115            L2=L(2)                                              RH1100
00116            L3=L(3)                                              RH1110
00117            L(5)=1                                               RH1120
00120            L(8)=1                                               RH1130
00121            L(12)=1                                              RH1140
00122            L(13)=1                                              RH1150
00123            L(14)=1                                              RH1160
00124            NPL=N+11                                             RH1170
00125            NPLP1=NPL+1                                          RH1180
00126            NPLPG=NPL+L2                                         RH1190
00127            NMG1=N+M+L2+1                                        RH1200
00130            LJOFF=14                                             RH1210
00131            L2OFF=L1OFF+M+N+L2                                   RH1220
00132            DO 1 I=1,M                                           RH1230
00135          1 A(I,NP2)=A(I,NP1)                                    RH1240
00137            DO 2 J=1,NP1                                         RH1250
00142          2 A(MP2,J)=A(MP1,J)                                    RH1260
00144            KT=1                                                 RH1270
00145            KT1=0                                                RH1271
00146            WRITE(6,1001)                                        RH1272
00150            DO 1002 I=1,M                                        RH1273
00153       1002 WRITE(6,1003) I,R(I)                                 RH1274
00160            WRITE(6,5)                                           RH1275
00162            WRITE(6,1004) BOUND                                  RH1276
00165            WRITE(6,5)                                           RH1277
00167            WRITE(6,3) KT1,CRIT                                  RH1278
00173            WRITE(6,4)                                           RH1280
00173      C ************************                                 RH1290
00173      C **SOLVE ORIGINAL PROBLEM**                                RH1300
00173      C ************************                                 RH1310
00175            MT=MT-1                                              RH1320
00176            NT=NT-1                                              RH1330
00177            CALL RVSMPX(A,MA,B,MB,MT,NT,L,X,TOLP,INV,0)          RH1340
00200            WRITE(6,5)                                           RH1350
00202            IF(L(3).LT.2) GO TO 200                              RH1360
00204            WRITE(6,950) (STAR,I=1,40)                           RH1361
00212            IF(L(3).EQ.3) GO TO 15                               RH1370
00212      C **INFEASIBILITY METHOD FOR INFEASIBLE ORIGINAL PROBLEM** RH1380
00214            NT=NT+1                                              RH1390
00215            A(MP1,NP1) =-1.0                                     RH1400
00216            DO 6 J=1,N                                           RH1410
00221          6 A(MP1,J)=0.0                                         RH1420
00223            DO 7 I=1,M                                           RH1430
00226            A(I,NP1)=-R(I)                                       RH1440
00227          7 CONTINUE                                             RH1450
00231            L(1)=L1                                              RH1460
00232            L(2)=L2                                              RH1470
00233            L(3)=L3                                              RH1480
00234            INV=0                                                RH1490
00235            CALL RVSMPX(A,MA,B,MB,MT,NT,L,X,TOLP,INV,0)          RH1500
00236            WRITE(6,14)                                          RH1510
00240            NT=NT-1                                              RH1520
00241            IF(L(3).EQ.2) GO TO 9                                RH1530
00243            RATMIN=X(NP1)                                        RH1540
00244            CRIT=RATMIN                                          RH1550
00245            IF(CRIT.LE.BOUND) GO TO 8                            RH1560
00247            WRITE(6,10) CRIT                                     RH1570
00252            WRITE(6,950) (STAR,I=1,40)                           RH1571
```

```
00260          GO TO 9                                                         RH1580
00261        8 WRITE(6,19) CRIT                                                RH1590
00264        9 A(MP1,NP1)=A(MP2,NP1)                                           RH1600
00265          DO 11 I=1,M                                                     RH1610
00270       11 A(MP1,J)=A(MP2,J)                                               RH1620
00272          IF(L(3).EQ.2) GO TO 15                                          RH1630
00274          IF(CRIT.GT.BOUND) GO TO 999                                     RH1640
00276          DO 12 I=1,M                                                     RH1650
00301       12 A(I,NP1)=A(I,NP2)                                               RH1660
00303          DO 13 I=1,M                                                     RH1670
00306          A(I,NP1)=A(I,NP1)+RATMIN*R(I)                                   RH1680
00307       13 CONTINUE                                                        RH1690
00311          L(1)=L1                                                         RH1700
00312          L(2)=L2                                                         RH1710
00313          L(3)=L3                                                         RH1720
00314          INV=0                                                           RH1730
00314    C **SOLVE PROBLEM WITH THETA FROM INFEASIBILITY METHOD**              RH1740
00315          CALL RVSMPX(A,MA,B,MB,MT,NT,L,X,TOLP,INV,0)                     RH1750
00316          WRITE(6,950) (STAR,I=1,40)                                      RH1760
00316    C ******************************************                         RH1770
00316    C **COMPUTE Y AND STORE IN COLUMN MP1 OF B**                          RH1780
00316    C ******************************************                         RH1790
00324          GO TO 200                                                       RH1791
00325      199 KT1=KT                                                          RH1792
00326          KT=KT+1                                                         RH1793
00327      200 DO 220 I=1,M                                                    RH1800
00332          B(I,MP1)=0.0                                                    RH1810
00333          DO 210 J=1,M                                                    RH1820
00336          B(I,MP1)=B(I,MP1)+(B(I,J)*R(J))                                 RH1830
00337      210 CONTINUE                                                        RH1840
00341      220 CONTINUE                                                        RH1850
00343          CBY=0.0                                                         RH1860
00344          DO 230 J=1,M                                                    RH1870
00347          K=L(L2OFF+J)                                                    RH1880
00350          IF(K.GT.N) GO TO 230                                            RH1890
00352          CBY=CBY+(A(MP2,K)*B(J,MP1))                                     RH1900
00353      230 CONTINUE                                                        RH1910
00355          COMPUT=X(NMG1)-(CBY*CRIT)                                       RH1920
00356          ISG='+'                                                         RH1930
00357          IF(COMPUT.LT.0.0) ISG=' '                                       RH1940
00361          IF(NSET.EQ.1.AND.KT1.GT.0) WRITE(6,1005)KT1,CRIT,LET1,NTER,LET2,KL  RH1942
00361         1EAVE,X(NMG1),CBY,ISG,COMPUT                                     RH1944
00376          IF(NSET.EQ.1.AND.KT1.GT.0) WRITE(6,4)                           RH1946
00401          IF(NSET.EQ.1.AND.KT1.GT.0) GO TO 1006                           RH1948
00403          WRITE(6,231) KT1,KT,CBY,ISG,COMPUT                              RH1950
00412          WRITE(6,950) (STAR,I=1,40)                                      RH1960
00412    C **FIND MINIMUM RATIO AND LEAVING VARIABLE**                         RH1970
00420     1006 K=0                                                             RH1980
00421          RATMIN-1E35                                                     RH1990
00422          DO 300 I=1,M                                                    RH2000
00425          IF(B(I,MP1).GE.0.0) GO TO 300                                   RH2010
00427          RATIO=-B(I,MP2)/B(I,MP1)                                        RH2020
00430          IF(RATIO.GT.RATMIN) GO TO 300                                   RH2030
00432          RATMIN=RATIO                                                    RH2040
00433          K=I                                                             RH2050
00434          LEAVE=I                                                         RH2060
00435      300 CONTINUE                                                        RH2070
00437          KLEAVE=L(L2OFF+LEAVE)                                           RH2080
```

```
00440          CRIT=CRIT+RATMIN                                                  RH2090
00441          IF(K.EQ.0) GO TO 302                                              RH2100
00443          IF(CRIT.LE.ROUND) GO TO 310                                       RH2110
00445          WRITE(6,301) KT,ROUND                                             RH2120
00451          WRITE(6,950) (STAR,I=1,40)                                        RH2121
00457          GO TO 999                                                         RH2130
00460      310 IF(NSET.EQ.1) GO TO 1007                                          RH2135
00462          WRITE(6,3) KT,CRIT                                                RH2140
00462    C *************************************************************        RH2150
00462    C **COMPUTE INDICATORS AND DENOMINATORS FOR ENTERING VARIABLES**        RH2160
00462    C *************************************************************        RH2170
00466    1007 RATMAX=-1E35                                                       RH2180
00467          DO 560 J=1,NPLPG                                                  RH2190
00472          IF(X(J).GE.0.0) GO TO 560                                         RH2200
00474          J1=L(LOFF+J)                                                      RH2210
00475          IF(J1.GT.0) GO TO 540                                             RH2220
00475    C **REAL VARIABLES**                                                    RH2230
00477          XX=0.0                                                            RH2240
00500          DO 510 I=1,M                                                      RH2250
00503      510 XX=XX+B(LEAVE,I)*A(I,J)                                           RH2260
00505          IF(XX.GE.0.0) GO TO 560                                           RH2270
00507          YY=A(MP1,J)                                                       RH2280
00510          DO 520 I=1,M                                                      RH2290
00513      520 YY=YY+B(MP1,I)*A(I,J)                                             RH2300
00515          GO TO 550                                                         RH2310
00515    C **SLACK AND SURPLUS VARIABLES**                                       RH2320
00516      540 J2=J-N                                                            RH2330
00517          XX=(-1)**(J1+1)*B(LEAVE,J2)                                       RH2340
00520          IF(XX.GE.0.0) GO TO 560                                           RH2350
00522          YY=(-1)**(J1+1)*B(MP1,J2)                                         RH2360
00523      550 RATIO=-YY/XX                                                      RH2370
00524          IF(RATIO.LE.RATMAX) GO TO 560                                     RH2380
00526          RATMAX=RATIO                                                      RH2390
00527          NTER=J                                                            RH2400
00530      560 CONTINUE                                                          RH2410
00532          IF(RATMAX.GT.-1E35) GO TO 570                                     RH2420
00534          WRITE(6,561)                                                      RH2430
00536          WRITE(6,950) (STAR,I=1,40)                                        RH2431
00544          GO TO 999                                                         RH2440
00544    C **************************                                            RH2450
00544    C **CHANGE RIGHT HAND SIDES**                                           RH2460
00544    C **************************                                            RH2470
00545      570 DO 580 I=1,M                                                      RH2480
00550          A(I,NP1)=A(I,NP1)+(RATMIN*R(I))                                   RH2490
00551      580 CONTINUE                                                          RH2500
00551    C **SET X VECTOR FOR RVSMPX WITH ADVANCED START**                       RH2510
00553          X(NTER)=1.0                                                       RH2520
00554          X(KLEAVE)=-1.0                                                    RH2530
00555          L(1)=L1                                                           RH2540
00556          L(2)=L2                                                           RH2550
00557          L(3)=0                                                            RH2560
00560          L(4)=0                                                            RH2570
00561          L(12)=0                                                           RH2571
00562          INV=1                                                             RH2580
00563          IF(NTER.GT.N) GO TO 900                                           RH2590
00565          LET1=' X('                                                        RH2600
00566          GO TO 910                                                         RH2610
00567      900 LET1=' S('                                                        RH2620
```

```
      NTER=NTER-N
  910 IF(KLEAVE.GT.N) GO TO 920
      LET2=' X('
      GO TO 930
  920 LET2=' S('
      KLEAVE=KLEAVE-N
  930 IF(NSET.EQ.1) L(13)=0
      IF(NSET.EQ.1) GO TO 1008
      WRITE(6,931) LET1,NTER,LET2,KLEAVE
      WRITE(6,5)
 1008 CALL RVSMPX(A,MA,B,MB,MT,NT,L,X,TOLP,INV,0)
      WRITE(6,5)
      IF(NSET.EQ.1.AND.KT1.EQ.0) WRITE(6,1009)
      IF(NSET.EQ.1.AND.KT1.EQ.0) WRITE(6,1010)
      IF(NSET.EQ.1.AND.KT1.EQ.0) WRITE(6,4)
      IF(NSET.EQ.1) GO TO 199
      WRITE(6,951) X(NMG1)
      GO TO 199
  302 WRITE(6,18) KT
      WRITE(6,950) (STAR,I=1,40)
      GO TO 999
   15 IF(L(3).EQ.2) WRITE(6,16)
      IF(L(3).EQ.3) WRITE(6,17)
      WRITE(6,950) (STAR,I=1,40)
  999 DO 1000 I=1,M
 1000 A(I,NP1)=A(I,NP2)
    3 FORMAT(1H0,'CRITICAL VALUE (',I3,') OF THETA = ',E14.8)
    4 FORMAT(//)
    5 FORMAT(/)
   10 FORMAT(' WHEN THETA CRITICAL =', F15.5, 'YOUR PROBLEM BECOMES FEAS
     1IBLE BUT THIS BYPASSES YOUR LIMIT.')
   14 FORMAT(//////)
   16 FORMAT(' PROBLEM REMAINS INFEASIBLE THROUGHOUT RANGE OF THETA.  NO
     1 STUDY PERFORMED.')
   17 FORMAT(' PROBLEM IS UNBOUNDED.  NO STUDY PERFORMED.'1
   18 FORMAT('      CRITICAL VALUE (',I3,') OF THETA IS INFINITE.'1
   19 FORMAT(' WHEN THETA CRITICAL =', F15.5, 'YOUR PROBLEM BECOMES FEAS
     1IBLE AND THE STUDY CONTINUES.')
  231 FORMAT(' OBJECTIVE FUNCTION VALUE BETWEEN CRITICAL VALUES (',I3,')
     1 AND (',I3,') OF THETA OBTAINED FROM',E15.8,'*THETA ',A1,E15.8)
  301 FORMAT('      CRITICAL VALUE (',I3,') BYPASSES YOUR LIMITATION, THE
     1REFORE BASIS REMAINS OPTIMAL AT YOUR LIMIT OF',F10.5)
  561 FORMAT(' PROBLEM BECOMES INFEASIBLE BEYOND THIS CRITICAL VALUE.'1
  950 FORMAT(/40A3)
  931 FORMAT(' AT THAT VALUE,'A3,I4,') ENTERS THE BASIS AND'A3,I4,') LEA
     1VES.')
  951 FORMAT(' OBJECTIVE FUNCTION VALUE =', E14.8)
 1001 FORMAT(' THE R VECTOR IS AS FOLLOWS')
 1003 FORMAT(' R('I4,') ='F6.2)
 1004 FORMAT(' BOUND='F6.2)
 1005 FORMAT(I6,F15.4,5X,A3,I4,')',3X,A3,I4,')',F14.4,6X,F10.4,'* THETA
     1',A1,F10.4)
 1009 FORMAT(' CRIT VAL     CRIT VAL      BASIS INFORMATION      OBJ FUNC
     1VAL     OBJ FUNC VAL BETWEEN THIS')
 1010 FORMAT('   NO.                        ENTERS      LEAVES',15X,'
     1    AND NEXT CRIT VAL')
      MT=MT+1
      NT=NT+1
```

# APPENDIX D: Timing Considerations

# APPENDIX D

## TIMING CONSIDERATIONS

RANGES SAMPLES: Time Per Right-Hand Side

and Objective Function Coefficient

| PROBLEM SIZE | | ELAPSED TIME RANGES (SEC.) | TIME PER R.H.S. | TIME PER COEFFICIENT |
|---|---|---|---|---|
| m | n | | | |
| 10 | 20 | .1388 | .0031 | .0054 |
| 15 | 20 | .1674 | .0031 | .0060 |
| 20 | 20 | .2216 | .0035 | .0075 |
| 20 | 30 | .2750 | .0032 | .0071 |
| 20 | 80 | .9258 | .0033 | .0108 |
| 50 | 150 | 4.0446 | .0039 | .0257 |

APPENDIX D (continued)

PAROBJ SAMPLES: Time Per Critical Value

| PROBLEM SIZE | | ELAPSED TIME PAROBJ (SEC.) | NO. OF COMPUTED CRITICAL VALUES | TIME PER CRITICAL VALUE |
|---|---|---|---|---|
| m | n | | | |
| 10 | 20 | .7608 | 3 | .2536 |
| 15 | 20 | .0554 | 1 | .0554 |
| 20 | 20 | 14.1665 | 5 | 2.8333 |
| 20 | 30 | 12.5100 | 9 | 1.3900 |
| 20 | 80 | 9.4508 | 8 | 1.1814 |
| 50 | 150 | 190.2708 | 23 | 8.2726 |

APPENDIX D (continued)

PARRHS SAMPLES: Time Per Critical Value

| PROBLEM SIZE m | PROBLEM SIZE n | ELAPSED TIME PARRHS (SEC.) | NO. OF COMPUTED CRITICAL VALUES | TIME PER CRITICAL VALUE |
|---|---|---|---|---|
| 10 | 20 | .7930 | 2 | .3965 |
| 15 | 20 | 1.8704 | 3 | .6235 |
| 20 | 20 | 4.3180 | 10 | .3925 |
| 20 | 30 | 1.9914 | 1 | 1.9914 |
| 20 | 80 | 4.7322 | 1 | 4.7322 |
| 50 | 150 | 73.2580 | 2 | 36.6290 |

USCOMM NBS DC

NBS-114A (REV. 7-73)

| U.S. DEPT. OF COMM. **BIBLIOGRAPHIC DATA SHEET** | 1. PUBLICATION OR REPORT NO. NBSIR 75-666 | 2. Gov't Accession No. | 3. Recipient's Accession No. |
|---|---|---|---|
| 4. TITLE AND SUBTITLE "Post Optimality and Parametric Analysis with the National Bureau of Standards' Linear Programming Subroutine" | | | 5. Publication Date February 1975 <br> 6. Performing Organization Code |
| 7. AUTHOR(S) Tyrone B. Ayers | | | 8. Performing Organ. Report No. |
| 9. PERFORMING ORGANIZATION NAME AND ADDRESS **NATIONAL BUREAU OF STANDARDS DEPARTMENT OF COMMERCE WASHINGTON, D.C. 20234** | | | 10. Project/Task/Work Unit No. <br> 11. Contract/Grant No. |
| 12. Sponsoring Organization Name and Complete Address *(Street, City, State, ZIP)* Same as No. 9 | | | 13. Type of Report & Period Covered Final <br> 14. Sponsoring Agency Code |

15. SUPPLEMENTARY NOTES

16. ABSTRACT *(A 200-word or less factual summary of most significant information. If document includes a significant bibliography or literature survey, mention it here.)*

This report is a sequel to NBS Report 10695 (February 1972), "The National Bureau of Standards' Linear and Quadratic Programming Subroutines," which documented one phase of an effort to provide users, of the facility operated by the National Bureau of Standards' Computer Services Division, with reliable, clearly-described solution algorithms for selected frequently-arising classes of special mathematical problems. The present report presents subroutines which perform post-optimality analysis and parametric programming studies on linear programming problems solved by the National Bureau of Standards' RVSMPX subroutine. (The present versions of these codes use internal storage only.)

17. KEY WORDS *(six to twelve entries; alphabetical order, capitalize only the first letter of the first key word unless a proper name; separated by semicolons)*

Algorithms; linear programming; parametric programming; post-optimality analysis.

| 18. AVAILABILITY [X] Unlimited | 19. SECURITY CLASS (THIS REPORT) | 21. NO. OF PAGES |
|---|---|---|
| [ ] For Official Distribution. Do Not Release to NTIS | UNCLASSIFIED | 47 |
| [ ] Order From Sup. of Doc., U.S. Government Printing Office Washington, D.C. 20402, SD Cat. No. C13 | 20. SECURITY CLASS (THIS PAGE) | 22. Price |
| [X] Order From National Technical Information Service (NTIS) Springfield, Virginia 22151 | UNCLASSIFIED | $ 3.75 |

USCOMM-DC 29042-P74