



AgEcon SEARCH
RESEARCH IN AGRICULTURAL & APPLIED ECONOMICS

The World's Largest Open Access Agricultural & Applied Economics Digital Library

This document is discoverable and free to researchers across the globe due to the work of AgEcon Search.

Help ensure our sustainability.

Give to AgEcon Search

AgEcon Search

<http://ageconsearch.umn.edu>

aesearch@umn.edu

*Papers downloaded from **AgEcon Search** may be used for non-commercial purposes and personal study only. No other use, including posting to another Internet site, is permitted without permission from the copyright owner (not AgEcon Search), or as allowed under the provisions of Fair Use, U.S. Copyright Act, Title 17 U.S.C.*

No endorsement of AgEcon Search or its fundraising activities by the author(s) of the following work or their employer(s) is intended or implied.

Staff Paper Series

Staff Paper P81-4

January 1981

USER'S GUIDE FOR MINNLP

By:
Earl I. Fuller

**Department of Agricultural and Applied
Economics**

University of Minnesota
Institute of Agriculture, Forestry, and Home Economics
St. Paul, MN 55108

Staff Paper P81-4

January 1981

USER'S GUIDE FOR MINNLP

by
Earl I. Fuller

Staff Papers are published without formal review within the
Department of Agricultural and Applied Economics.

USER'S GUIDE FOR MINNLP

I. INTRODUCTION

- A. Identification
- B. Uses And Limits
- C. Environment

II. BUILDING A MATRIX

- A. General Matrix
- B. Mathematical Statement
- C. A Sample Matrix
- D. Notes About The Matrix

III. CREATING AND USING A DATA FILE WITH MINNLP

- A. Input Format
- B. Correcting The Data File
 - 1. While Inputting Data
 - 2. After The Data File Is Created
- C. Sample Data File

IV. USING THE PROGRAM

- A. Calling The Program
 - 1. Using An Unformatted Data File
 - 2. Using Multiple Data Files
 - 3. Using A Formatted Data File
- B. Running The Program
 - 1. Options Available
 - 2. Brief Description Of Each MINNLP Option With Examples
 - 3. Logging Off

V. INTERPRETING THE RESULTS

- A. Table 1
- B. Table 2
- C. Table 3
- D. Table 4

Preface

MINNLP in its current version is a good example of how computer software evolves over time. This program has a long history of modifications beginning from the work of Foster and Weyrick at New Hampshire published in 1968. As the below list of references will further indicate, various modifications to this work have occurred since then. A parallel development of the computer software package FMGTLP also provided useful ideas associated with how to present the output and manipulate the data. Both of these programs provide the user with easy to use input options and well labeled output in a more understandable form than most general purpose packages. While the current versions have been developed to operate on Control Data Corporation Cyber hardware, they are workable on machines with smaller word size as well. In fact, a BASIC language version of MINNLP exists for use on micro/mini-computers.

References

- Foster, Bennett B. and Richard R. Weyrick, A Modified General Simplex Method For Solving Linear Programming Problems, Agricultural Experiment Station, University of New Hampshire, Durham, New Hampshire, Station Bulletin 493, February 1968.
- Fuller, Earl I., FMGTLP - Management Linear Programming, (Write-up To Accompany A Computer Code), Department of Agricultural and Food Economics, University of Massachusetts, Amherst, Massachusetts, 1970.
- Fuller, Earl I., Manuals For Remote Access Computing - Manual 3: UMASS Linear Programming, Department of Agricultural and Food Economics, University of Massachusetts, Amherst, Massachusetts, February 1970.
- Fuller, Earl I., MASSLPCD - Compact Linear Programming, (Write-up To Accompany A Computer Code), Department of Agricultural and Food Economics, University of Massachusetts, Amherst, Massachusetts, A & FE Report 70-11, Sept. 1970.

INTRODUCTION

The code MINNLP, or Minnesota Linear Programming, is an interactive linear programming algorithm which is efficient in the use of computer storage space and computational time. It is also easy to use and contains matrix modification options.

A. Identification

1. Title: MINNLP - Compact Linear Programming
2. Programmer: Earl I. Fuller*
3. Programming Language: MNF Fortran
4. Date: December 18, 1980

B. Uses And Limits

Management linear programming generates LP solutions (output) which, when accompanied by a set of footnotes, becomes management reports understandable to laymen. The tables of output provide headings that give a summarization of the obtained solution. The output includes the original prices of commodities, the original levels of resources, the optimum level of production of the enterprises selected, the limiting resources and their marginal value, the non-limiting resources and their slack amounts, and the enterprises not in the plan and their opportunity cost. Output also includes price and resource stability information.

The present form of MINNLP will not do quadratic programming, but does contain a special option that allows the user to do resource mapping.

C. Environment

1. Hardware - CDC - CYBER time-share system.
2. Software - Code is compiled and run under NOS monitors. It presently requires a 52K field length to load. MINNLP is dimensioned for a 77 x 111 matrix (excluding slack vectors).

* With help from a number of others over the years--including Eugene Haley, Jim Kelly, Dale Nordquist, Tony Groble, Dave Maloney and Mark Kirchhoff.

A contribution to Minnesota Agricultural Experiment Station project 14-036, Management Information Systems for Farm Firms.

BUILDING A MATRIX

A. General Matrix

It is easiest to understand this algorithm by perceiving the complete tableau as represented by one matrix. This matrix has as its Nth (last) column the right hand side specification or resource limits. It has as its Mth (last) row the values of the objective function or the gross margins from the activities. This algorithm does not use slack columns. Therefore, slack resources are implicitly priced at zero.

B. Mathematical Statement

A mathematical statement of the algorithm.

1. Maximize $A_{MN} = \sum_{j=1}^{N1} (A_{Mj}) \cdot (X_j)$

- Where: A_{MN} is the final gross margin or profit
 X_j is the level at which the activity is utilized
 A_{Mj} is the gross margin returns for the enterprise, i.e. is equal to C_j as frequently denoted in L.P. modeling
N1 is the last activity. N is the last column in the matrix (N = N1 + 1)

So, the objective is to maximize the sum of the returns times the levels of all of the enterprises.

2. Subject to a set of equations relating constraint levels to activities:

$A_{iN} \geq \sum_{j=1}^{N1} \sum_{i=1}^{M1} (X_j) \cdot (A_{ij})$

- Where: A_{iN} is the level the resource is limited to, i.e. is equal to bi or the RHS as frequently denoted in L.P. modeling
 X_j is the level at which an activity is utilized
 A_{ij} is how much resource is used by the activity
N1 is the number of real activities. N is the last column in the matrix (N = N1 + 1)
M1 is the number of resource constraints (M = M1 + 1)

Thus, the objective function value is limited by the sum of the uses of resources which must be less than or equal to the amount of resource available.

And, all $X_j \geq 0$ for $j=1$ to $N1$

all $A_{iN} \geq 0$ for $i=1$ to $M1$

All activity levels must be at zero or positive levels (i.e. you can't produce negative amounts) and feasible solutions must force zero or positive levels of constraint levels (i.e. all negative RHS values must be "forced out").

C. A Sample Matrix

The example problem in words - Suppose that you are the operator of a feed grinding and mixing plant. You have a set of buildings and equipment as indicated below. Your feed grinder will grind 120 tons of feed per day. Your mixer will mix 100 tons of feed per day. You have four full-time employees who work 8 hours per day. They are each willing to work 2 extra hours at an overtime rate of \$3 per hour. You can hire extra labor by the hour if you want to; while it would only cost you \$2 an hour, they are only one-half as efficient as the full-time workers.

Your choices are (a) to grind feed for farmers which will contribute to overhead \$2 per ton ground (over cash operating costs) or (b) you can mix and grind feed for farmers with a contribution over cash operating costs of \$3 per ton. But, you will not mix feed unless you have ground it in your own mill. In addition, you can grind, mix and sell three kinds of mixed feeds. Feed A gives you \$4 per ton on the same basis; Feed B, \$8; and Feed C, \$12.50. Hours of labor required per ton for the various alternatives are as follows: grinding alone - .2; grinding and mixing - .3; grinding, mixing and preparing for sale of Feed A - .5, Feed B - .6, Feed C - .7. These hours are based on the efficiency of the full-time employees.

A Way The Problem Is Given To The Computer:

ONE Matrix Formulation Of The Feed Elevator Operation Problem

Amount or Level Specif.	Units	Cost	Sign	Price Row/Col.	R.H.S.						
					Custom Grind Feed Ton \$2 1	Mix Grind Feed Ton \$3 2	Prep. & Sell Feed A Ton \$4 3	Prep. & Sell Feed B Ton \$8 4	Prep. & Sell Feed C Ton \$12 5	Hire Over- time Hrs. -\$3 6	Hire Extra Labor Hrs. -\$2 7
Max. Grinder Capacity	120	Zero	≥	1	1	1	1	1	1		
Max. Mixer Capacity	100	Zero	≥	2	1	1	1	1	2		
Avail. Labor	32	Zero	≥	3	.2	.3	.5	.6	.7	-1	-.5
Max. Overtime	8	Zero	≥	4							1
Feed B - Control	15	Zero	≥	5				1			
Feed C - Control	0	Zero	≥	6			-.5		1		

Notes: 1 - Assumes that hiring overtime labor does not increase grinding or mixing capacity. (Hiring labor could have increased maximum capacity.)

2 - Assumes any worker can do any job (perfect substitutes).

3 - Labor bill of \$2.50 per hour x 32 hours per day must be paid if labor is used or not. If priced in the R.H.S., it would imply not paid if used unless an offsetting amount is added to cash expenses (double entry accounting).

4 - Matrix is prepared for one "typical" day's operation.

5 - This and all linear programming is an exercise in logic full of "if this, than that" and "if and only if" type statements.

D. Notes About The Matrix

1. The columns are the activities. The last activity (hire extra labor) is the N1 column. The last column is the right-hand side or the N column. RHS value is an A_{iN} (i.e. $A_{1N} = 120$; $A_{2N} = 100$; $A_{3N} = 32$, etc.) Each column has a label and a measurement unit.
2. The rows are the resources. The last resource (Feed C - control) is the M1 row. The last row is the objective or return over cost or the M row. Each return is an A_{Mj} or C_j (i.e. $A_{M1} = C_1 = \$1,800$; $A_{M2} = C_2 = \$300$). All rows are valued at zero.
3. The body of the matrix indicates how much resource is used or supplied by each activity. Each coefficient is an A_{ij} . A_{ij} 's that are zero need not be entered.
4. Positive C_j 's indicate positive returns over costs. Negative C_j 's indicate negative returns over costs (usually purchase activities or intermediate products such as crops that are fed). Positive RHS's indicate a maximum specification of the constraint level while a negative RHS value indicates a minimum specification of the constraint level (usually a contract requirement). Positive A_{ij} 's indicate that the activity "takes from" the resource. Negative A_{ij} 's indicate that the activity "gives to" the resource.

CREATING AND USING A DATA FILE WITH MINNLP

A. Input Format

The input procedures for MINNLP follow the same rules as the batch processing code FMGTLP with some minor but important exceptions.

These exceptions are:

1. Single blank spaces are sufficient to separate variables in the read list, i.e. exact column location is not important as it is on cards.*
2. There must be at least as many variables in a line or record as items in the read list, i.e. blanks are not read as variables in the list.
3. Each line must begin with a sequence number. These numbers must be in ascending order.

* The formatted input used in FMGTLP (1970 version).

4. Alphanumeric computer words to label the problem, its resources, and constraints consist of eight or less characters in sequence in a line. Asterisks may be used to combine shorter words to make a single computer word to increase print readability and capacity.

Terminology Used

(CR) = carriage return

Ø = zero

b = space, note: a space can be used instead of a comma and vice versa.

Step 1 - Assuming you are already logged on to the computer, the following procedure should be used:

To create a new data file, type: NEW, file name (CR)

The file name is the name of the data file you wish to create. The file name can be any name of seven or fewer characters.

Step 2 - Type the word AUTO (CR)

Typing the word AUTO will have the computer print line numbers automatically starting with 100 and increasing by intervals of 10, for each new line. This will save you time since you won't have to retype line numbers for each line.

Step 3 - Type: Ø, 1, Ø (CR)

This tells the computer that the label and parameters of a new problem is the next information to be read.

Step 4 - Type in a label or identification line of up to 72 characters to describe the problematic situation.

Type: identification line (CR)

Remember to include the objective, time span, title and any other useful information in the identification line.

Step 5 - Type the number of constraints in the plan or tableau, the number 1, and the number of activities considered in the plan or tableau.

Type: # rows, 1, # columns (CR)

Step 6 - Type: Ø, 2, Ø (CR)

Remember, a comma or a blank space can be used interchangeably or together. This step tells the computer that some A_{ij} values will follow next.

Step 7 - Type in a row location, column location, and A_{ij} value set per line, for each nonzero A_{ij} value in the data set.

Format Required for Step 7

<u>Row #</u>	<u>Column #</u>	<u>A_{ij}</u>	<u>Value</u>
1	2	1	(CR)
5	4	1	(CR)

Reminder - be sure to leave at least one blank space, or comma, between each value.

Step 8 - Once all the A_{ij} values have been typed in, type: \emptyset , 3, \emptyset (CR).

This tells the computer that the RHS values will follow.

Step 9 - Type in each row number and RHS value set per line for each nonzero RHS value in the tableau. A RHS value equal to \emptyset does not need to be typed in.

Format Required for Step 9

<u>Row #</u>	<u>Row #</u>	<u>RHS Value</u>	
1	1	120	(CR)
2	2	100	(CR)

Reminder - be sure to leave one blank space, or comma, between each value.

Step 10 - When all RHS values have been typed in, type: \emptyset , 4, \emptyset (CR).

This tells the computer to now read the C_j - price/costs and column labels for each activity. (Remember that A_{Mj} values = C_j values.)

Step 11 - Type in the C_j - price/cost values and column labels for each activity, one set per line.

Enter the C_j values and labels for every column in the tableau, in the format shown below.

The column labels are divided into three fields of eight characters. These three fields must be separated by a blank space or comma.

<u>Column #</u>	<u>Column #</u>	<u>C_j-Price/Cost</u>	<u>First 8 Character Field Label</u>	<u>Second 8 Character Field Label</u>	<u>Third 8 Character Field Label</u>	
1	1	2	CUSTOM	GRINDFD	TONS	(CR)
2	2	3	MIXGRIND	FEED	TONS	(CR)

Reminder: (1) each eight character field may contain up to eight alphanumeric characters; (2) each label or character field must be separated by a blank space or a comma; (3) an asterisk may be used to connect two words to fit into the eight alphanumeric character spaces; when the data file is printed, the asterisk won't be printed in the output.

Step 12 - When all of the C_j values and labels have been typed in, type: $\emptyset, 5, \emptyset, X, X, X$ (CR). This tells the computer to now read the row labels of each constraint listed in the tableau. The three X's must be typed after the $\emptyset, 5, \emptyset$, since the computer is reading these fields and a value must be placed in each field. Failure to do so will result in an error message when the data file is called by the LP program.

Step 13 - Type in the constraint row labels shown below. The row labels are divided into three fields of eight characters. These three fields must be separated by a blank space or a comma.

Row#	Row #	Must Be A Zero \emptyset	First 8 Character Field Label	Second 8 Character Field Label	Third 8 Character Field Label
1	1	\emptyset	MAXGRND	CAPACITY	TONS (CR)
2	2	\emptyset	MAXMIXR	CAPACITY	TONS (CR)

Reminder: (1) no price/cost values on rows are allowed, a \emptyset must be typed; (2) each eight character field may contain up to eight alphanumeric characters; (3) each label or character field must be separated by a blank space or comma; and (4) an asterisk may be used to connect two words to fit into the eight alphanumeric character spaces; when the data file is printed, the asterisk won't be printed in the output.

Step 14 - When all of the row labels have been typed in, type: $\emptyset, 6, \emptyset, X, X, X$, (CR). Remember, the three X's must be typed following the $\emptyset, 6, \emptyset$ since the computer is reading these three fields, a value must be typed in each field, or an error message will result when the data file is called by the LP program.

Typing $\emptyset, 6, \emptyset$ tells the computer that all of the data for the tableau has been typed in, and to now solve the problem.

Step 15 - After the 0, 6, 0, X, X, X line has been typed, press the escape (ESC) button located on the computer terminal. This will stop the computer from printing a line number each time the carriage return is pressed.

Step 16 - Type: SAVE (CR)

When the computer responds with READY, your file is a permanent (old) file stored in your user number.

Reminder - you must save the new file that you have just created. Later you will need to recall your data file. In order to do this your data file must be a permanent (old) file.

Note - the order of control values [0, 2, 0] [0, 3, 0] [0, 4 0] [0, 5, 0] and their associated sets of data is actually unimportant to the computer. The reading routines will adjust to any order. The suggested order is simply the logical one.

B. Correcting The Data File

1. While inputting data

If two line numbers of the same value and their associated data are entered, the last line entered is the one finally stored and used in calculations. To make changes (or correct mistakes) in your data file as you are typing in data, simply press the escape key on the terminal. Type in the line number of the line you wish to correct. Leave one blank space then enter the data desired, as previously explained. Type AUTO, comma, and the line number of the next data line you wish to type. Now, the computer will continue supplying line numbers as before.

2. After the data file is created

To make changes in a file that is already saved (permanent), retype the line or use the XEDIT procedures and, when changes have been made, type REP (replace). This will make the desired changes a part of your permanent data file. Type: REP (CR).

C. Sample Data File

To illustrate the proper format your data file should follow, an unformatted data listing with the elevator problem is shown below.

Listing Of The Unformatted Data File

sentence number in sequence tells computer the ID sentence and parameters of the tableau are next ID sentence

```

00100 0 1 0
00110 IBM VERSION ELEVATOR PROBLEM; MAXIMIZING PROFITS 1 DAY TIME PERIOD
00120 ④ 1 ⑦ first value tells number of rows in tableau
00130 0 ② 0 third value tells number of columns in tableau
00140 1 1 1 tells computer that Aij values follow
00150 3 1 .2
00160 1 2 1
00170 2 2 1
00180 3 2 .3
00190 1 3 1
00200 2 3 1
00210 3 3 .5
00220 ⑥ 3 -.5
00230 ① 4 1 row numbers
00240 2 4 1
00250 3 ④ .6
00260 5 ④ 1 column numbers
00270 1 5 ①
00280 2 5 ②
00290 3 5 ⑦ Aij's in matrix
00300 6 5 1
00310 3 6 -1
00320 4 6 1
00330 3 7 -.5
00340 0 ③ 0 read RHS values
00350 1 1 120
00360 2 2 100
00370 3 3 32 RHS values
00380 4 4 8
00390 5 5 15
00400 0 ④ 0 read Cj-price/costs and column labels
00410 1 1 2 CUSTOM GRIND*FD TONS
00420 2 2 ③ MIX*GRND FEED TONS
00430 3 3 4 PREP&SEL FEED*A TONS
00440 4 4 8 PREP&SEL FEED*B TONS
00450 5 5 12.5 PREP&SEL FEED*C TONS
00460 6 6 -3 HIRE OVERTIME HOURS
00470 7 7 -2 HIRE EXTRA*LB HOURS
00480 0 ⑤ 0 X X X read row labels
00490 1 1 0 MAX*GRND CAPACITY TONS
00500 2 2 0 MAX*MIXR CAPACITY TONS
00510 3 3 0 AVAIL. LABOR HOURS
00520 4 4 0 MAX OVERTIME HOURS
00530 5 5 0 FEED*B CONTROL TONS
00540 6 6 0 FEED*C CONTROL TONS
00550 0 ④ 0 X X X tells computer all data has been read in,
also tells computer to solve problem.

```

will print a blank space for each asterisk

C_j's

column labels

row labels

Dummy data to fill read list requirement

If the proper identification sentence or label for your tableau is listed, press the (CR).

If you typed in the wrong file name, an error report will result. You must now go back to step 17 and start over.

Assuming the correct unformatted data file has been typed in, and no additional files are to be called, press the (CR) when the computer again responds:

DATA FILE NAME (CR = DATA IS COMPLETE) ? ELEV

Since the data file is unformatted, the computer will respond with a series of questions.

DO YOU WANT MINNLP MATRIX OUTPUT (CR = FMGTLP) ? YES

WANT TO CHANGE THE PROBLEM DISCRIPTION LINE (CR = NO) ? NO

NAME OF LP FORMATED DATA FILE (CR = SKIP IT)
? ELEV1

Step 20 - You will want to type yes if you wish to have line numbers in your formatted data file; if you type no or (CR), there will be no line numbers in your formatted data file.

Step 21 - For the second question, type yes (CR), if you want to change the identification sentence or label of your data file.

The computer will respond with a (?). Type the new identification sentence or label and press (CR).

Type No or (CR) if you do not wish to change your label.

Step 22 - When the computer responds with the third question, the computer is asking for a new name for your data file. Your original data file has been reformatted by the computer so that your original data file can be read by the computer and inserted into the LP program. The formatted file is a new, permanent, separate data file, which must have its own unique file name.

If you want to run your unformatted data file without creating a permanent formatted data file, press the (CR) when the computer asks for the name of the LP formatted data file. The formatted file just created by the MINNLP program is only temporary and will be destroyed as soon as you stop the MINNLP program, or call another data file.

2. Using Multiple Data Files

You may combine separate data files to create one new data file. To illustrate this, a simple example is shown below.

Suppose we want to change the RHS value in the ELEV1 data file from 32 available labor hours to 80 available labor hours. Create and name a new data file like this:

```
NEW,ELEV1A
:AUTO
00100 0 3 0
00110 3 3 80
00120 *DEL*
SAVE
```

Now, call the MINNLP data file. The computer responds and you can type in your name and address. Then answer yes or no to divert output, then type in the name of the original formatted data file you have created.

```
L.P. TEST VERSION 11/10/80 CONTACT EARL FULLER 612-373-1145
WHAT IS YOUR NAME ? JOHN FARMER
WHAT IS YOUR ADDRESS? RURAL MINN.
```

```
WANT TO DIVERT THE MINNLP RESULTS (CR = NO) ?
```

```
DATA FILE NAME (CR = DATA IS COMPLETE) ? ELEV1
```

When the computer responds:

```
1 IBM 5100 VERSION ELVATOR PROBLEM; MAXIMIZING PROFITS 1 DAY TIME PERIOD
DATA FILE NOW READ IN
```

```
DATA FILE NAME (CR = DATA IS COMPLETE) ? ELEV1A
```

Type in the name of the data file you wish to add. In this case data file ELEV1A was typed in.

Now proceed as previously instructed.

```
DATA FILE NAME (CR = DATA IS COMPLETE) ?          press (CR)

DO YOU WANT MINNLP MATRIX OUTPUT (CR = FMGTLP)    ? YES          type yes
                                                                or (CR)
WANT TO CHANGE THE PROBLEM DISCRIPTION LINE (CR = NO) ?          type yes
                                                                or no
NAME OF LP FORMATED DATA FILE (CR = SKIP IT)
? ELEV1B
```

Type in the name of the new LP formatted data file if you want the new combination data file to be permanent. Otherwise press the (CR).

3. Using A Formatted Data File

If you already have a formatted data file, type in the name of your formatted data file, step 19 mentioned earlier.

By typing in the name of your formatted data file, steps 20, 21 and 22 will be omitted.

Notice the difference between the original data file and the formatted data file listed below.

Formatted Data File

new line numbers

notice the increased space between columns

OLD,ELEV1

:LIST

Flags the reading procedures to expect a formatted file

Unused parameters, except for FMGTLP

note how the decimal points all line up

each field is 8 characters long

Dummy characters to complete the read list requirements

10000	-1	1		
10020	6	1	7	1
10030	0	2		0
10040	1	1		1.000
10050	3	1		.200
10060	1	2		1.000
10070	2	2		1.000
10080	3	2		.300
10090	1	3		1.000
10100	2	3		1.000
10110	3	3		.500
10120	6	3		-.500
10130	1	4		1.000
10140	2	4		1.000
10150	3	4		.600
10160	5	4		1.000
10170	1	5		1.000
10180	2	5		2.000
10190	3	5		.700
10200	6	5		1.000
10210	3	6		-1.000
10220	4	6		1.000
10230	3	7		-.500
10240	0	3		0
10250	1	1	120.000	
10260	2	2	100.000	
10270	3	3	32.000	
10280	4	4	8.000	
10290	5	5	15.000	
10300	0	4	0	
10310	1	1	2.000	CUSTOM GRIND-FDTONS
10320	2	2	3.000	MIX-GRNDFEED TONS
10330	3	3	4.000	PREP&SELFEED-A TONS
10340	4	4	8.000	PREP&SELFEED-B TONS
10350	5	5	12.500	PREP&SELFEED-C TONS
10360	6	6	-3.000	HIRE OVERTIMEHOURS
10370	7	7	-2.000	HIRE EXTRA-LB HOURS
10380	0	5	0	: : :
10390	1	1	0	MAX-GRNDCAPACITYTONS
10400	2	2	0	MAX-MIXRCAPACITYTONS
10410	3	3	0	AVAIL. LABOR HOURS
10420	4	4	0	MAX OVERTIMEHOURS
10430	5	5	0	FEED-B CONTROL TONS
10440	6	6	0	FEED-C CONTROL TONS
10450	0	6	0	: : :
10460	0	0	0	: : :

B. Running The Program

1. Options available

Step 23 - The computer responds

```
NEXT MINNLP OPTION. (-1 = LIST OPTIONS) ?
```

If you wish to see a list of the options available in this program, type in -1.

The computer will list the options available.

OPTIONS AVAILABLE:

```
0 = CR = END OF LINEAR PROGRAM RUN,  
1 = READ A(NOTHER) DATA FILE FOR USE,  
2 = PRINT ALTERNATIVES & CONSTRAINTS LIST,  
3 = PRINT THE CURRENT PROBLEM MATRIX,  
4 = MODIFY THE MATRIX (WITH SUB-OPTIONS),  
5 = BUDGET/CHECK AN ANSWER/SOLUTION,  
6 = SOLVE THE CURRENT MATRIX/PROBLEM,  
7 = MAKE POST-SOLUTION PRICE/RHS ASSUMPTION CHANGES
```

2. Brief description of each MINNLP option with examples

```
NEXT MINNLP OPTION. (-1 = LIST OPTIONS) ?
```

You should type in the number of the option you desire, followed by a (CR). If you would like to have another list of the options available, type in -1.

Option 0 = CR = end of linear program run

To terminate the program, type a 0 or (CR) when the computer asks for the next MINNLP option.

NEXT MINNLP OPTION. (-1 = LIST OPTIONS) ? 0 (CR)

You are now free to call an old data file, or to create a new data file.

Option 1 = read a(nother) data file for use

Option 1 will allow you to call a new data file to run with the MINNLP program. This saves you time since you don't have to rerun the MINNLP program every time you want to use a new data file.

NEXT MINNLP OPTION. (-1 = LIST OPTIONS) ? 1 (CR)

The computer will respond:

COMPUTER DECISION AIDS	AGRICULTURAL ECONOMICS PROGRAM:
EXTENSION FARM MANAGEMENT	MINNESOTA LINEAR PROGRAMING ANALYSIS
AGRI. EXTENSION SERVICE	RESULTS FOR: JOHN FARMER user's name
UNIVERSITY OF MINNESOTA	09-04-80 RURAL MINN. and address

1 IBM 5100 VERSION ELVATOR PROBLEM; MAXIMIZING PROFITS 1 DAY TIME PERIOD

title of the current
data file

The computer responds:

WANT TO DIVERT THE MINNLP RESULTS (CR = NO) ?

Another yes or no (CR). The computer then asks for the name of the new data file you wish to use. Type in the name of the new data file.

DATA FILE NAME (CR = DATA IS COMPLETE) ? FURNLP type in new
data file name
followed by (CR)

The computer then tells you the new data file has been read in, and lists the title of the new data file.

The computer asks if you want to enter any other data files, and do you want to see a list of the MINNLP options that are available or to simply choose an option.

10010 THE FURNITURE 2X2 PROBLEM title of the new data file
DATA FILE NOW READ IN
DATA FILE NAME (CR = DATA IS COMPLETE) ?
NEXT MINNLP OPTION. (-1 = LIST OPTIONS) ?

Option 2 = print alternatives and constraints list

Option 2 will print the heading for each column vector with its associated C_j value, and the heading for each row vector will be printed with its associated RHS value. To have the computer print a listing of the row and column vectors, type a 2 when the computer asks for the next MINNLP option.

NEXT MINNLP OPTION. (-1 = LIST OPTIONS) ? 2 (CR)

The computer responds:

COMPUTER DECISION AIDS
EXTENSION FARM MANAGEMENT
AGRI. EXTENSION SERVICE
UNIVERSITY OF MINNESOTA

AGRICULTURAL ECONOMICS PROGRAM:
MINNESOTA LINEAR PROGRAMING ANALYSIS
RESULTS FOR: JOHN FARMER
09-04-80 RURAL MINN.

1 IBM 5100 VERSION ELVATOR PROBLEM; MAXIMIZING PROFITS 1 DAY TIME PERIOD

PROCESS / ALTERNATIVES AND UNIT PRICE/COSTS title of the formatted file

101	CUSTOM GRIND-FD TONS	\$	2.000	}	column vectors with C_j values
102	MIX-GRNDFEED TONS	\$	3.000		
103	PREP&SELFEED-A TONS	\$	4.000		
104	PREP&SELFEED-B TONS	\$	8.000		
105	PREP&SELFEED-C TONS	\$	12.500		
106	HIRE OVERTIME HOURS	\$	-3.000		
107	HIRE EXTRA-LB HOURS	\$	-2.000		

RESOURCE/CONSTRAINTS AND RHS LEVELS

1	MAX-GRND CAPACITY TONS	"	120.000	}	row vectors with RHS values
2	MAX-MIXR CAPACITY TONS	"	100.000		
3	AVAIL. LABOR HOURS	"	32.000		
4	MAX OVERTIME HOURS	"	8.000		
5	FEED-B CONTROL TONS	"	15.000		
6	FEED-C CONTROL TONS	"	0		

Option 3 = print the current problem matrix

Option 3 will print the formatted data file in MINNLP matrix form.
To have the computer print the initial matrix, type a 3 when the computer asks for the next MINNLP option.

NEXT MINNLP OPTION. (1 = LIST OPTIONS) ? 3 (CR)

The computer will respond:

COMPUTER DECISION AIDS
 EXTENSION FARM MANAGEMENT
 AGRI. EXTENSION SERVICE
 UNIVERSITY OF MINNESOTA

AGRICULTURAL ECONOMICS PROGRAM:
 MINNESOTA LINEAR PROGRAMING ANALYSIS
 RESULTS FOR: JOHN FARMER
 09-04-80 RURAL MINN.

1 IBM 5100 VERSION ELVATOR PROBLEM; MAXIMIZING PROFITS 1 DAY TIME PERIOD
 ENTER 3 VALUES; (THE MATRIX HAS 8 COLUMNS)
 ENTER (1)-1ST COL. & LAST COL. TO PRINT + NO OF COLS PER PAGE? 1,8,10

The name and address of the user is listed as well as the title of the data file in use.

The computer then asks you to type in 3 numbers separated by a comma or blank space. The first number you type in should be the number of the first column of the matrix that you wish to have printed. The second number you type in should be the number of the last column in the matrix that you wish to have printed. The third number you type in should be the number of columns that you want printed per page.

These features allow you to print select portions of the matrix without having to print the whole matrix. By regulating the number of columns per page, the MINNLP matrix output can be printed on any size computer terminal or printer.

Below is an example listing of the initial matrix.

title of the formatted data file

1 IBM 5100 VERSION ELVATOR PROBLEM; MAXIMIZING PROFITS 1 DAY TIME PERIOD

ROW NO.	PROCESS (COLUMN) NO.	CUSTOM GRIND-FD TONS	MIX-GRND FEED TONS	PREP&SEL FEED-A TONS	PREP&SEL FEED-B TONS	PREP&SEL FEED-C TONS	HIRE OVERTIME HOURS	HIRE EXTRA-LB HOURS	SPECIFIED RESOURCE AMOUNTS
	PROCESS UNIT PRICE COST \$	2.000	3.000	4.000	8.000	12.500	-3.000	-2.000	0
1	MAX-GRND CAPACITY TONS *	1.000	1.000	1.000	1.000	1.000	0	0	120.000
2	MAX-MIXR CAPACITY TONS *	0	1.000	1.000	1.000	2.000	0	0	100.000
3	AVAIL. LABOR HOURS *	.200	.300	.500	.600	.700	-1.000	-.500	32.000
4	MAX OVERTIME HOURS *	0	0	0	0	0	1.000	0	8.000
5	FEED-B CONTROL TONS *	0	0	0	1.000	0	0	0	15.000
6	FEED-C CONTROL TONS *	0	0	-.500	0	1.000	0	0	0

Option 4 = modify the matrix (with sub-options)

This option allows you to use the MINNLP program to make changes in your formatted data file. To use this option, type a 4 when the computer asks for the next MINNLP option.

NEXT MINNLP OPTION. (-1 = LIST OPTIONS) ? 4 (CR)

The computer responds with the title of the data file, and a question concerning the types of changes to be made.

1 IBM 5100 VERSION ELVATOR PROBLEM; MAXIMIZING PROFITS 1 DAY TIME PERIOD
HIT CR TO CHANGE SINGLE VALUES; 1 FOR OTHER REVISIONS.
?

If you only want to change A_{ij} , RHS or price/cost values in your formatted data file, press (CR). The computer will respond:

THE PRICE/COST ROW = M = 7 THE R.H.S. COLUMN = N =, 8
USE ACTUAL COLUMN LOCATION NUMBERS; NOT COLUMN CODE NUMBERS
" 3,8,50
↑ ↑ ↙
row column value

Notice that the price/cost row and the RHS column are numbered. These row and column values must be used in making each individual matrix change.

To change single values in the formatted data file, simply type the row number, column number and new value after each question mark that appears. You can only type one row, column and value set per line. Press (CR) after each line has been typed in. When no further changes are desired, press the carriage return and the computer responds with a series of questions.

DO YOU WANT TO REPLACE THE DATA FILE ? YES
DO YOU WANT MINNLP MATRIX OUTPUT (CR = FMGTLP) ? YES
WANT TO CHANGE THE PROBLEM DISCRPTION LINE (CR = NO) ?
NAME OF LP FORMATED DATA FILE (CR = SKIP IT) -
? E1
↑
name of new data file

Type yes (CR) if you want to replace the old data file with the updated data file (the file containing the changes you have just made).

If you type no (CR) the computer will respond:

NEXT MINNLP OPTION. (-1 = LIST OPTIONS) ?

Type yes (CR) if you want your updated data file to be printed in MINNLP matrix output format.

Type yes (CR) if you want to change the identification sentence or label of your data file. The computer will respond with a ? after which you should type the new identification sentence or label and press (CR).

Type no or (CR) if you do not wish to change your label.

The computer is now asking for a new name for your formatted data file. The original formatted data file has been changed by the new values that have just been inputted. This formatted file is a new, separate data file which must have its own unique file name.

If you press (CR) when the computer asks for the name of the new data file, the changes you have just made will only be temporary, not permanent. Once you have called a new data file or have logged off the computer, only the data file that existed before any changes were made will remain.

To make other kinds of matrix revisions

When the computer responds:

```
1 IBM 5100 VERSION ELVATOR PROBLEM; MAXIMIZING PROFITS 1 DAY TIME PERIOD
  HIT CR TO CHANGE SINGLE VALUES; 1 FOR OTHER REVISIONS.
? 1
```

Type a 1 instead of a (CR). The computer will respond.

```
THE PRICE/COST ROW = M = 7 THE R.H.S. COLUMN = N =, 8
USE ACTUAL COLUMN LOCATION NUMBERS; NOT COLUMN CODE NUMBERS
NEED MATRIX ACTION IN TRUCTIONS          ? YES
```

Type no or (CR) if you do not wish to see a listing of the options available.

Type yes if you want a listing of the options available. The computer lists the following options:

THE QUERY, NEXT ACTION, REQUIRES ONE OF SEVERAL" RESPONSES;
TYPING THE FIRST THREE LETTERS OF EACH" WILL SUFFICE
THEY ARE:
INSERT...A ROW OR COLUMN INSERTING PROCEDURE WHERE DESIRED.
DELETE...A ROW OR COLUMN DELETION PROCEDURE WHERE DESIRED.
BOUND... AN ACTIVITY LEVEL LIMITING PROCEDURE.
SCALE... A ROW OR COLUMN SCALING (OR ROW CHANGING) PROCEDURE.
REORDER...A MATRIX ROW OR COLUMN REORDERING PROCEDURE.
NEXT MATRIX ACTION (CR=NONE) ?

Explanation

Sub-option:	Purpose/result:
Insert	This option allows you to insert another row or column in the original matrix.
Delete	This option allows you to delete either a row or column from the original matrix.
Bound	This option allows you to set a maximum or minimum limit on the number of times a particular activity comes into the final solution.
Scale	This option allows you to change every coefficient in a row or column by a constant amount.
Reorder	This option allows you to rearrange the order of the columns or rows of the original matrix.

When the computer responds:

NEXT MATRIX ACTION (CR=NONE) ?

Type in the first three letters of the option you wish to use. A few brief examples are shown below to help explain the proper procedures that must be used with these options.

First example, deleting a row or column

```
NEXT MATRIX ACTION (CR=NONE) ? DEL
DELETE A ROW OR COLUMN      ? ROW
DELETED ROW NUMBER ? 4
DELETE A ROW OR COLUMN      ?      (CR) = no more
NEXT MATRIX ACTION (CR=NONE) ?      (CR)

DO YOU WANT TO REPLACE THE UPDATED FILE ? YES
DO YOU WANT MINNLP MATRIX OUTPUT (CR = FMGTLP) ? YES
WANT TO CHANGE THE PROBLEM DISCRIPTION LINE (CR = NO) ?      (CR)
NAME OF LP FORMATED DATA FILE (CR = SKIP IT)
? ELEV2 ← new, separate data file
```

In this example, one row (row 4) was deleted from the original matrix. When deleting more than one row or column, remember to keep track of the rows and columns that have already been deleted. As soon as a row or column is deleted, the row or column is gone from the original matrix. It is very important to count carefully only the remaining rows and columns in the matrix in order to put a new row or column number on the next row or column to be deleted.

Second example, inserting a row or column

NEXT MATRIX ACTION (CR=NONE) ? INS

NEED FURTHER INSERT INSTRUCTIONS ? YES

INSERT WILL SHOVE THE INDICATED ROW OR COLUMN" INTO THE INDICATED LOCATION
MOVING ALL OTHER ROWS" DOWN OR COLUMNS TO THE LEFT

CR=DONE WITH THIS SECTION

INSERT A ROW\; A COLUMN\ (OR #CR#) ? ROW

NUMBER OF INSERTED ROW ? 4

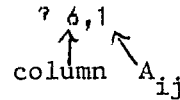
ENTER 2-8 COL WORDS+ UNITS FOR NEW ROW ? MAX

OVERTIME HOURS

ENTER COLUMN _ VALUE PAIRS 1 SET PER LINE AS NEW ROW

? 8,8

must be typed in formatted style,
each field 8 characters long



INSERT A ROW\; A COLUMN\ (OR #CR#) ? (CR) = no more

NEXT MATRIX ACTION (CR=NONE) ? (CR)

DO YOU WANT TO REPLACE THE UPDATED FILE ? YES

DO YOU WANT MINNLP MATRIX OUTPUT (CR = FMGTLP) ? YES

WANT TO CHANGE THE PROBLEM DISCRIPTION LINE (CR = NO) ?

NAME OF LP FORMATED DATA FILE (CR = SKIP IT)

? ELEV3 ← new, separate data file

In this example, one row (row 4) was inserted into the original matrix.

When inserting more than one row or column, remember to keep track of the rows and columns that have already been inserted. As soon as the row or column is inserted, the row or column becomes a part of the original matrix. It is very important to count carefully the rows or columns in the original matrix and also each row or column that has been inserted. By doing this, you can insert all new rows or columns in their proper location in the matrix.

Third example, bounding an activity labels

NEXT MATRIX ACTION (CR=NONE) ? BOU

ENTER A COLUMN NUMBER FOLLOWED BY THE MAXIMUM(+) OR
MINIMUM(-) QUANTITY LEVEL IT IS TO BE BOUNDED OR LIMITED AT.
?? 1 30

THE BOUND LABEL NOW READS MAX-CNTLCUSTOM TONS
IF THIS IS NOT ACCEPTABLE ENTER A NEW (FORMATED) ONE. (CR=OK)
?

TO ADD ANOTHER COL TO THE SAME BOUND LIMIT,
INDICATE THE COL (CR=NO)
?

BOUND ANOTHER COLUMN AS ABOVE? (CR=DONE)

ENTER A COLUMN NUMBER FOLLOWED BY THE MAXIMUM(+) OR
MINIMUM(-) QUANTITY LEVEL IT IS TO BE BOUNDED OR LIMITED AT.
??

NEXT MATRIX ACTION (CR=NONE) ?

DO YOU WANT TO REPLACE THE UPDATED FILE ? YES

DO YOU WANT MINNLP MATRIX OUTPUT (CR = FMGTLP) ? YES

WANT TO CHANGE THE PROBLEM DISCRPTION LINE (CR = NO) ?

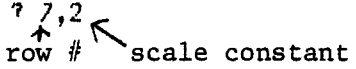
NAME OF LP FORMATED DATA FILE (CR = SKIP IT)
? ELEV4

In this example, the activity in column one (max. grinder capacity, tons) was limited to 30 tons. By putting an upper bound of 30 tons on column one, no more than 30 tons of grinder capacity can be used.

Notice how the MINNLP program placed a constraint on the amount of grinder capacity available in the matrix listed below. Look at row 7 in the matrix. Notice the upper limit of 30 tons of grinder capacity in row 7.

Fourth example - scaling a row or a column

NEXT MATRIX ACTION (CR=NONE) ? SCA
NEED FURTHER SCALE INSTRUCTIONS ? YES
SCALE WILL MULTIPLY THE INDICATED ROW OR COLUMN" BY THE DESIRED CONSTANT
(TO CHANGES ALL SIGNS IN A ROW THPE ROWNO.&-1.)
COMPLETION IS INDICATED BY A CR=NO MORE
SCALE A ROW\; A COLUMN\ (OR #CR#) ? ROW
ENTER ROW NO. _ SCALE CONSTANT (CR= ROWS DONE) ? 7,2
TYPE NEW ROW UNIT NAME ? DOLLARS
ENTER ROW NO. _ SCALE CONSTANT (CR= ROWS DONE) ?
SCALE A ROW\; A COLUMN\ (OR #CR#) ? (CR) = no more
NEXT MATRIX ACTION (CR=NONE) ? (CR) = no more
DO YOU WANT TO REPLACE THE UPDATED FILE ? YES
DO YOU WANT MINNLP MATRIX OUTPUT (CR = FMGTLP) ? YES
WANT TO CHANGE THE PROBLEM DISCRIPTION LINE (CR = NO) ?
NAME OF LP FORMATED DATA FILE (CR = SKIP IT)
? ELEV5 ← new, separate data file



In this example, each C_j value in row 7 was multiplied by the coefficient
Z. This doubled the value of the C_j 's in the process unit price cost row.

Fifth example, reordering a matrix row or column

```

NEXT MATRIX ACTION (CR=NONE) ? REO
REORDER ROWS OR COLUMNS:      ? COLUMNS
WHICH ONE DO YOU WANT TO MOVE? 2
WHERE DO YOU WANT IT (AFTER WHICH ONE) ? 0
REORDER ROWS OR COLUMNS:      ?
NEXT MATRIX ACTION (CR=NONE) ?

DO YOU WANT TO REPLACE THE UPDATED FILE ? YES

DO YOU WANT MINNLP MATRIX OUTPUT (CR = FMGTLP)      ? YES
WANT TO CHANGE THE PROBLEM DISCRPTION LINE (CR = NO)      ?

NAME OF LP FORMATED DATA FILE (CR = SKIP IT)
? ELEV6
    
```

In this example, the order of columns 1 and 2 was reversed. Notice the example listed below.

The positions of columns 1 and 2 are reversed (compare to matrix on page 20).

1 IBM 5100 VERSION ELVATOR PROBLEM; MAXIMIZING PROFITS 1 DAY TIME PERIOD

		positions switched		MIX-GRND	CUSTOM	PREP&SEL	PREP&SEL	PREP&SEL	HIRE	HIRE	SPECIFIED
		FEED	GRIND-FD	FEED-A	FEED-B	FEED-C	OVERTIME	EXTRA-LB	RESOURCE		
		TONS	TONS	TONS	TONS	TONS	HOURS	HOURS	AMOUNTS		
	PROCESS UNIT PRICE COST \$	3.000	2.000	4.000	8.000	12.500	-3.000	-2.000	0		
ROW											
NO.	PROCESS (COLUMN) NO.	101	102	103	104	105	106	107	108		
1	MAX-GRND CAPACITY TONS *	1.000	1.000	1.000	1.000	1.000	0	0	120.000		
2	MAX-MIXR CAPACITY TONS *	1.000	0	1.000	1.000	2.000	0	0	100.000		
3	AVAIL. LABOR HOURS *	.300	.200	.500	.600	.700	-1.000	-.500	32.000		
4	MAX OVERTIME HOURS *	0	0	0	0	0	1.000	0	8.000		
5	FEED-B CONTROL TONS *	0	0	0	1.000	0	0	0	15.000		
6	FEED-C CONTROL TONS *	0	0	-.500	0	1.000	0	0	0		

Option 5 = budget/check an answer/solution

This option allows you to set the desired level of each activity in the optimal solution. To use this option, type a 5 when the computer asks for the next MINNLP option.

NEXT MINNLP OPTION. (-1 = LIST OPTIONS) 7 5 (CR)

The computer asks you to input the process code number and the desired level of each activity wanted in the solution. Type one value set separated by a space or comma per line.

INPUT THE PROCESS CODE NUMBER AND THE DESIRED LEVEL OF EACH
ACTIVITY WANTED IN THE RESULTS (CR = NO MORE)

?? 101 5
?? 102 8
?? 103 10
?? 104 15
?? 105 20
?? 106 3
?? 107 5
??

comma or space

desired level in final solution

process code number

The order in which process code numbers and the data (desired level of each activity) is typed in is unimportant. Erroneous entries may be corrected by retyping the data line to provide a new process code level.

The budgeted results are listed below.

COMPUTER DECISION AIDS	AGRICULTURAL ECONOMICS PROGRAM:
EXTENSION FARM MANAGEMENT	MINNESOTA LINEAR PROGRAMING ANALYSIS
AGRI. EXTENSION SERVICE	RESULTS FOR: JOHN FARMER
UNIVERSITY OF MINNESOTA	12-06-80 RURAL MINN.

PROCESS/ALTERNATIVES	UNIT PRICE/COSTS	LEVEL IN BUDGET
1 CUSTOM GRIND FD TONS	\$ 2.000	5.000
2 MIX GRND FEED TONS	\$ 3.000	8.000
3 PREP&SEL FEED A TONS	\$ 4.000	10.000
4 PREP&SEL FEED B TONS	\$ 8.000	15.000
5 PREP&SEL FEED C TONS	\$ 12.500	20.000
6 HIRE OVERTIME HOURS	\$ -3.000	3.000
7 HIRE EXTRA LB HOURS	\$ -2.000	5.000

BUDGETED PROFIT/COST VALUE \$ 425.00

RESOURCE/CONSTRAINTS	ORGINAL LEVELS	CALCULATED	SLACK REMAINING
1 MAX GRND CAPACITY TONS	* 120.000	58.000	62.000
2 MAX MIXR CAPACITY TONS	* 100.000	73.000	27.000
3 AVAIL. LABOR HOURS	* 32.000	25.900	6.100
4 MAX OVERTIME HOURS	* 8.000	3.000	5.000
5 FEED B CONTROL TONS	* 15.000	15.000	0
6 FEED C CONTROL TONS	* 0	15.000	-15.000

NEGATIVE VALUES IN THE LAST COLUMN INDICATE THE BUDGET OR PLAN DOES NOT ABIDE BY THE SPECIFIED PLAN LIMITS. POSITIVE VALUES INDICATE UNUSED RESOURCES OR SLACK IN THE MIX SPECIFICATIONS.

Special note: After the problem/matrix has been solved, option 5 can be used to check the solution. To do this type a 5 when the computer asks for the next MINNLP option. Remember, to check the solution, option 6 must be run first; otherwise option 5 will only budget a solution.

NEXT MINNLP OPTION. (-1 = LIST OPTIONS) ? 5 (CR)

The computer responds:

WANT TO CHECK THE SOLUTION ? YES

Type yes if you want to check the solution. If you type no, the computer will let you budget a desired solution without checking the solution.

If you typed yes, the computer responds:

WANT TO SAVE THE SOLUTION MATRIX 1ST FOR FUTURE USE ? NO

Type yes if you want to save the solution matrix, type no if you do not. The computer then responds with a check of the final solution.

COMPUTER DECISION AIDS AGRICULTURAL ECONOMICS PROGRAM:
EXTENSION FARM MANAGEMENT MINNESOTA LINEAR PROGRAMING ANALYSIS
AGRI. EXTENSION SERVICE RESULTS FOR: JOHN FARMER
UNIVERSITY OF MINNESOTA 12-17-80 RURAL MINN.

PROCESS/ALTERNATIVES	UNIT PRICE/COSTS	LEVEL IN BUDGET
1 CUSTOM GRIND FD TONS	\$ 2.000	41.250
3 PREP&SEL FEED A TONS	\$ 4.000	42.500
4 PREP&SEL FEED B TONS	\$ 8.000	15.000
5 PREP&SEL FEED C TONS	\$ 12.500	21.250
6 HIRE OVERTIME HOURS	\$ -3.000	8.000
7 HIRE EXTRA LB HOURS	\$ -2.000	26.750

BUDGETED PROFIT/COST VALUE \$ 560.63

RESOURCE/CONSTRAINTS	ORGINAL LEVELS	CALCULATED	SLACK REMAINING
1 MAX BRND CAPACITY TONS	* 120.000	120.000	0
2 MAX MIXR CAPACITY TONS	* 100.000	100.000	0
3 AVAIL. LABOR HOURS	* 32.000	32.000	.000
4 MAX OVERTIME HOURS	* 8.000	8.000	0
5 FEED B CONTROL TONS	* 15.000	15.000	0
6 FEED C CONTROL TONS	* 0	0	0

NEGATIVE VALUES IN THE LAST COLUMN INDICATE THE BUDGET OR PLAN DOES NOT ABIDE BY THE SPECIFIED PLAN LIMITS. POSITIVE VALUES INDICATE UNUSED RESOURCES OR SLACK IN THE MIX SPECIFICATIONS.

NEXT MINMLP OPTION. (-1 = LIST OPTIONS) ?

Option 6 = solve the current matrix/problem

Option6 is comprised of four sub-options. To solve the current matrix/
problem, type a 6 when the computer asks for the next MINNLP option.

NEXT MINNLP OPTION. (-1 = LIST OPTIONS)? 6 (CR)

The computer then asks if you want to see a list of the sub-options that
are available. To see a list of the sub-options that are available, type
a -1 (CR).

OUTPUT SUB-OPTION (-1 = LIST OPTIONS)? -1 (CR)

The computer lists the sub-options that are available.

**SOLUTION DISPLAY SUB OPTIONS;
1 = DISPLAY PLAN & INPUTED VALUE TO RESOURCES,
2 = OPTION 1 + SENSITIVITY ANALYSIS,(130 SPACES),
3 = DISPLAY FINAL MATRIX (MAY BE VOLUNINOUS),
4 = DISPLAY ALL MATRICIES (VOLUNININOUS)
OUTPUT SUB-OPTION (-1 = LIST OPTIONS)?**

Sub-option 1 = display plan and inputted value to resources

Sub-option 1 will display the optimal plan for the matrix problem. The
solution will be listed showing the income over direct cash expenses for the
plan, the activities selected and the number of units, the processes not in
solution, the limiting constraints, and the resources not completely used. To
have the computer solve the problem, type a 1 when the computer asks for the
next output sub-option.

OUTPUT SUB-OPTION (-1 = LIST OPTIONS)? 1 (CR)

The computer will respond:

BY STEPS...VARIABLE IN...VARIABLE OUT OF SOLUTION..PROFIT VALUE

1(105)	PREP&SELFEED C	(6)	FEED C CONTROL	0
2(103)	PREP&SELFEED A	(3)	AVAIL. LABOR	385.882
3(106)	HIRE OVERTIME	(4)	MAX OVERTIME	458.353
4(107)	HIRE EXTRA LB	(2)	MAX MIXRCAPACITY	478.500
5(104)	PREP&SELFEED B	(5)	FEED B CONTROL	511.125
6(101)	CUSTOM GRIND FD	(1)	MAX GRNDCAPACITY	560.625

COMPUTER DECISION AIDS AGRICULTURAL ECONOMICS PROGRAM:
 EXTENSION FARM MANAGEMENT MINNESOTA LINEAR PROGRAMING ANALYSIS
 AGRI. EXTENSION SERVICE RESULTS FOR: JOHN FARMER
 UNIVERSITY OF MINNESOTA 12-06-80 RURAL MINN.

10010 00110 IBM VERSION ELEVATOR PROBLEM; MAXIMIZING PROFITS 1 DAY TIME

INCOME OVER DIRECT CASH EXPENSES (GROSS MARGIN) = \$ 560.62

ACTIVITIES INCLUDED IN SELECTED PLAN AND NUMBER OF UNITS

101	CUSTOM GRIND FD	AT \$	2.000 / TONS	:	41.250
103	PREP&SELFEED A	AT \$	4.000 / TONS	:	42.500
104	PREP&SELFEED B	AT \$	8.000 / TONS	:	15.000
105	PREP&SELFEED C	AT \$	12.500 / TONS	:	21.250
106	HIRE OVERTIME	AT \$	-3.000 / HOURS	:	8.000
107	HIRE EXTRA LB	AT \$	-2.000 / HOURS	:	26.750

ACTIVITIES NOT INCLUDED IN PLAN AND COST OF FORCING A UNIT INTO THE PLAN

102	MIX GRNDFEED	AT \$	3.000 / TONS	\$	1.925
-----	--------------	-------	--------------	----	-------

LIMITING RESOURCES AND VALUE OF ANOTHER UNIT ORIGINAL AMOUNT SPECIFIED

1	MAX GRNDCAPACITY TONS	\$	1.200	120.000
2	MAX MIXRCAPACITY TONS	\$	2.525	100.000
3	AVAIL. LABOR HOURS	\$	4.000	32.000
4	MAX OVERTIME HOURS	\$	1.000	8.000
5	FEED B CONTROL TONS	\$	1.875	15.000
6	FEED C CONTROL TONS	\$	3.450	0

TO SAVE THE FINAL BASIS FOR RERUNS, NAME IT:(CR=NO)

?

Sub-option 2 = as in option 1 plus sensitivity analysis

Sub-option 2 will show the solution to the problem/matrix (like sub-option 1) plus a sensitivity analysis of the optimal solution. To have the computer solve the problem with a sensitivity analysis, type a 2 when the computer asks for the next MINNLP sub-option.

OUTPUT SUB-OPTION (1 = LIST OPTIONS) 2 (CR)

The computer responds:

BY STEP.. VARIABLE IN...VARIABLE OUT OF SOLUTION..PROFIT VALUE

COMPUTER DECISION AIDS AGRICULTURAL ECONOMICS PROGRAM
 EXTENSION FARM MANAGEMENT MINNESOTA LINEAR PROGRAMING ANALYSIS
 AGRI. EXTENSION SERVICE RESULTS FOR. JOHN FARMER
 UNIVERSITY OF MINNESOTA 09-15-80 RURAL MINN.

00110 .5M VESPIGN ELEVATOR PROBLEM; MAXIMIZING PROFITS 1 DAY TIME PERIOD

INCOME OVER DIRECT CASH EXPENSES (GROSS MARGIN) = \$ 560.62

ACTIVITIES INCLUDED IN THE DERIVED PLAN

FIG. SENSITIVITY ANALYSIS OF THE OPTIMUM PLAN

COL NO	SELECTED ALTERNATIVES	LEVEL OR AMOUNT	UNIT PRICE	LOWEST PRICE	LIMITING VARIABLE	HIGHEST PRICE	LIMITING VARIABLE
101	CUSTOM GRIND FD TONS	41.25	2.00	.80	MAX GRND CAPACITY	3.37	MAX MIXR CAPACITY
103	PREP&SEL FEED A TONS	42.50	4.00	15	MIX GRND FEED	7.45	FEED C CONTROL
104	PREP&SEL FEED B TONS	15.00	8.00	6.13	FEED B CONTROL	10000.00	SPECIFIED LIMIT
105	PREP&SEL FEED C TONS	21.25	12.50	5.50	FEED C CONTROL	20.00	FEED B CONTROL
106	HIRE OVERTIME HOURS	3.00	-5.00	-4.00	MAX OVERTIME	9997.00	SPEC.FED LIMIT
107	HIRE EXTRA LB HOURS	25.75	-2.00	-5.00	MAX GRND CAPACITY	-1.50	MAX OVERTIME

PRICE SENSITIVITY OF RESOURCES NOT COMPLETELY USED

ROW NO	UNUSED RESOURCES	AMOUNT LEFT	ORIGINAL AMOUNT	LOWEST PRICE	LIMITING VARIABLE	HIGHEST PRICE	LIMITING VARIABLE
--------	------------------	-------------	-----------------	--------------	-------------------	---------------	-------------------

SENSITIVITY OF PLAN TO REJECTED ALTERNATIVES.

COL NO	ALTERNATIVES NOT SELECTED	COST OF INCLUSION	UNIT PRICE	LOWER LIMIT	BOUND VARIABLE	UPPER LIMIT	BOUND VARIABLE
102	MIX GRND FEED TONS	1.92	3.00	-76.43	HIRE EXTRA LB	10000.00	PROBLEM LIMIT

RESOURCE SENSITIVITY RANGE AROUND ORIGINAL VALUES.

ROW NO	LIMITING RESOURCES	RESIDUAL MARGINAL VALUE	ORIGINAL AMOUNT	LOWER LIMIT	BOUND VARIABLE	UPPER LIMIT	BOUND VARIABLE
1	MAX GRND CAPACITY TONS	1.20	120.00	78.75	CUSTOM GRIND FD	10120.00	PROBLEM LIMIT
2	MAX MIXR CAPACITY TONS	2.52	100.00	51.36	HIRE EXTRA LB	155.00	CUSTOM GRIND FD
3	HIRE OVERTIME HOURS	4.00	32.00	-9987.00	PROBLEM LIMIT	45.38	HIRE EXTRA LB
4	MAX OVERTIME HOURS	1.00	8.00	0	HIRE OVERTIME	21.23	HIRE EXTRA LB
5	FEED B CONTROL TONS	1.87	15.00	0	PREP&SEL FEED B	100.00	PREP&SEL FEED A
6	FEED C CONTROL TONS	3.45	0	-42.50	PREP&SEL FEED C	42.50	PREP&SEL FEED A

TO SAVE THE FINAL BASIS FOR RETURNS, MAKE IT,(CR=NO)

?

Sub-option 3 = display final matrix (may be voluminous)

Sub-option 3 will list the steps used in solving the problem. The final or solution matrix, as generated by the MINNLP program, will then be displayed. To have the computer display the final matrix, type a 3 when the computer asks for the next output sub-option.

OUTPUT SUB-OPTION (-1 = LIST OPTIONS)? 3 (CR)

The computer will respond:

BY STEPS...VARIABLE IN...VARIABLE OUT OF SOLUTION..PROFIT VALUE

HOW MANY COLUMNS PER PAGE(1 TO 10) ? 8

00110 IBM VERSION ELEVATOR PROBLEM; MAXIMIZING PROFITS 1 DAY TIME PERIOD

			MAX GRND	MIX GRND	AVAIL.	FEED B	FEED C	MAX	MAX MIXR	CURRENT
			CAPACITY	FEED	LABOR	CONTROL	CONTROL	OVERTIME	CAPACITY	PLAN
			TONS	TONS	HOURS	TONS	TONS	HOURS	TONS	AMOUNTS
	INPUTED SHADOW PRICES	\$	1.200	1.925	4.000	1.875	3.450	1.000	2.525	560.625
904										
40.	PROCESS (COLUMN) NO.		1	102	3	5	6	4	2	108
101	CUSTOM GRIND FD TONS	*	1.000	.250	0	-.250	.500	0	-.750	41.250
107	HIRE EXTRA LB HOURS	*	.400	.350	-2.000	.250	-.100	-2.000	.550	26.750
103	PREP&SEL FEED A TONS	*	0	.500	0	-.500	-1.000	0	.500	42.500
106	HIRE OVERTIME HOURS	*	0	0	0	0	0	1.000	0	8.000
104	PREP&SEL FEED B TONS	*	0	0	0	1.000	0	0	0	15.000
105	PREP&SEL FEED C TONS	*	0	.250	0	-.250	.500	0	.250	21.250

TO SAVE THE FINAL BASIS FOR RERUNS, NAME IT;(CR=NO) ?

Sub-option 4 = display all matrices

Caution: For sub-option 4 to work properly, it must be chosen and run before sub-options 1, 2 or 3.

Sub-option 4 will display the initial matrix iteration and all subsequent iterations until the final iteration is shown. To have the computer display all matrices, type a 4 when the computer asks for the next output sub-option.

OUTPUT SUB-OPTION (-1 = LIST OPTIONS)? 4 (CR)

The example shown here is a small and very simple matrix. As the matrix gets larger and larger, more iterations are needed to solve the problem.

The computer responds:

BY STEPS...VARIABLE IN...VARIABLE OUT OF SOLUTION..PROFIT VALUE

HOW MANY COLUMNS PER PAGE(1 TO 10) ? 3

10010 THE FURNITURE 2X2 PROBLEM title of the data file

ROW NO.	PROCESS (COLUMN) NO.	PARSONS TABLES EACH	AVABLE ASSEMBLY HOURS	CURRENT PLAN AMOUNTS	
	INPUTED SHADOW PRICES \$	-8.000	4.000	240.000	} first iteration
102	DEACONS BENCH EACH *	.500	.167	10.000	
2	AVAILBLE FINISHNG TIME(HRS*	3.000	-.333	12.000	

1 (102) DEACONS BENCH (1) AVAPLE ASSEMBLY 240.000

10010 THE FURNITURE 2X2 PROBLEM

ROW NO.	PROCESS (COLUMN) NO.	AVAILBLE FINISHNG TIME(HRS	AVABLE ASSEMBLY HOURS	CURRENT PLAN AMOUNTS	
	INPUTED SHADOW PRICES \$	2.667	3.111	272.000	} second and final iteration
102	DEACONS BENCH EACH *	-.167	.222	8.000	
101	PARSONS TABLES EACH *	.333	-.111	4.000	

2 (101) PARSONS TABLES (2) AVAILBLEFINISHNG 272.000

final basis

TO SAVE THE FINAL BASIS FOR RERUNS, NAME IT;(CR=NO) ?

For this example, a very small matrix was used. With a larger matrix, the number of iterations and matrices displayed could be a large amount. The computer performs the first iteration in the matrix and then displays the matrix. The second iteration is performed and the matrix is displayed. The computer keeps doing this until it reaches the final iteration and displays the final basis.

Special note: After every sub-option in option 6, the computer asks if you want to save the final basis for re-runs and, if you do, to give the final basis (data file) a name. Don't forget to purge this file after you have re-run it, otherwise you might confuse it with your original data file.

Option 7 = make post-solution price/RHS assumption changes

Caution: Before option 7 can be used, option 6 must be selected to solve the problem/matrix. Once the problem/matrix has been solved, option 7 can be used.

Option 7 permits you to map the range of a specified price that allows an activity to come into solution, or to map the range of a constraint/resource level to determine the point at which a constraint/resource level becomes limiting, or you can modify several prices and constraints at the same time.

Usefulness of option 7 - example

Price mapping - as the price level of one of the activities is varied, "breaking points" are observed. Such points are the price stability bounds to a particular solution and indicate where the solution will change, i.e., where an activity appearing in solution will be replaced by another and the solution (plan or basis) changes. The new solution will usually differ from the old both by the change in the two activities and in the level of other activities appearing in solution. The price of the mapping activity at which the solution changes is denoted as the "breaking price".

To delineate all possible solutions as the price of the activity being mapped is changed, it is only necessary to determine the solution at all "breaking prices". The price mapping feature of this LP code does

this, i.e. it determines all "breaking prices" and their associated optimum solutions as the price changes between some specified price range. In the price mapping operation, the first step is determination of the optimum solution at the algebraic lower specified price. For this original solution, the upper "breaking price" is determined. A second solution at this price is determined, which automatically provides the second "breaking price". These last two steps are repeated until a "breaking price" is found which is larger than the upper specified price. At this point the last solution is determined at the upper specified price instead of the higher breaking price.

The same logic applies to mapping resource/constraint levels and also when mapping prices and resource/constraint levels at the same time.

To use this option, type a 7 when the computer asks for the next MINNLP option.

```
NEXT MINNLP OPTION. (-1 = LIST OPTIONS) ? 7 (CR)
```

The computer responds:

```
1 = RANGE A SPECIFIED PRICE
2 = RANGE A CONSTRAINT/RESOURCE LEVEL
3 = MODIFY (SEVERAL) PRICES, CONSTRAINTS OR BOTH
CR = ALL DONE = RETURN TO MAJOR OPTIONS?
```

Because the output from option 7 is too voluminous to be listed here, examples will be shown using the correct method to enter data for option 7. The resulting output is not listed.

Sub-option 1 = range a specified price

```
1 = RANGE A SPECIFIED PRICE
2 = RANGE A CONSTRAINT/RESOURCE LEVEL
3 = MODIFY (SEVERAL) PRICES, CONSTRAINTS OR BOTH
CR = ALL DONE = RETURN TO MAJOR OPTIONS? 1
```

↑
sub-option 1 selected

The computer responds:

A DISCRIPTIVE TITLE LINE NOTEING VARIABLE(S) & MODIFICATIONS
? RANGING THE PRICE OF FEED-A FROM \$0 TO \$15



type in descriptive title line

The computer responds:

TYPE (1) THE RANGING VARIABLE CODE NUMBER
THEN (2) THE ALGEBRAICALLY LOWER VALUE (WITH SIGN?)
AND (3) THE UPPER LIMIT OF THE DESIRED VALUE RANGE
ENTER ALL THREE ITEMS SEPARATED BY A SPACE OR COMMA

? 103 0 15

↑
the column number
of the activity
you want to price
range

↘ algebraic
lower value

↗ algebraic
upper limit

The computer proceeds to list the output.

Sub-option 2 = range a constraint/resource level

1 = RANGE A SPECIFIED PRICE
2 = RANGE A CONSTRAINT/RESOURCE LEVEL
3 = MODIFY (SEVERAL) PRICES, CONSTRAINTS OR BOTH
CR = ALL DONE = RETURN TO MAJOR OPTIONS? 2

sub-option 2 selected ↗

The computer responds:

A DISCRIPTIVE TITLE LINE NOTEING VARIABLE(S) & MODIFICATIONS
? RANGING THE AVAILABLE MIXER CAPACITY FROM 50 TONS TO 200 TONS



type in a descriptive title line

The computer responds:

TYPE (1) THE RANGING VARIABLE CODE NUMBER
THEN (2) THE ALGEBRAICALLY LOWER VALUE (WITH SIGN?)
AND (3) THE UPPER LIMIT OF THE DESIRED VALUE RANGE
ENTER ALL THREE ITEMS SEPARATED BY A SPACE OR COMMA

? 2 50,200
↑
the row number of the constraint/
resource level to be ranged
algebraic lower value
algebraic upper limit

The computer proceeds to list the output.

Sub-option 3 = modify (several) prices, constraints or both

- 1 = RANGE A SPECIFIED PRICE
- 2 = RANGE A CONSTRAINT/RESOURCE LEVEL
- 3 = MODIFY (SEVERAL) PRICES, CONSTRAINTS OR BOTH
- CR = ALL DONE = RETURN TO MAJOR OPTIONS? 3

sub-option 3 selected

The computer responds:

A DESCRIPTIVE TITLE LINE NOTEING VARIABLE(S) & MODIFICATIONS
? CHANGING ASSEMBLY HRS FROM 24 TO 100, CHANGING TABLE PRICE FROM \$32 TO \$80

↑
type in a descriptive title line

The computer responds:

INPUT IS (1) A CHANGE VARIABLE CODE NUMBER
 THEN (2) THE DESIRED NEW VALUE (WITH SIGN?)
 (CR = CODE NO. OF ZERO = NO MORE CHANGES DESIRED)
 WANT A VECTOR CODE LISTING FIRST? (CR = NO) ? YES

PROCESS / ALTERNATIVES AND UNIT PRICE/COSTS

101 AVAILBLEFINISHNG TIME(HRS \$ 20.000
 102 AVABLE ASSEMBLY HOURS \$ 24.000

RESOURCE/CONSTRAINTS AND RHS LEVELS

1 DEACONS BENCH EACH " 60.000
 2 PARSONS TABLES EACH " 32.000

OK; NOW ENTER BOTH ITEMS AS XXX , XX.XX? 102 100

OK; NOW ENTER BOTH ITEMS AS XXX , XX.XX? 2 80 → new value for table price
 ↓ code #

OK; NOW ENTER BOTH ITEMS AS XXX , XX.XX?

WANT SENSITIVITY ANALYSIS OUTPUT OPTION (CR = NO) ? YES

code #
new value for assembly hours

The computer proceeds to list the output.

3. Logging off

When no further options are desired, type a 0 when the computer asks for the next MINNLP option.

To log off the computer, type BYE (CR). The computer will respond as shown below.

BYE

4530309 LOG OFF 14.44.42.
 ABFA205 SRU 7.897 UNTS.

After the computer stops typing, hang up the phone and turn the power switch off.

INTERPRETING THE RESULTS

When option 6, sub-option 2 is chosen, the computer lists the optimal solution and a price sensitivity analysis of this solution.

The solution and price sensitivity of the optimal plan is listed in four different tables. A brief discussion of each of the four tables is shown below.

Table 1

PRICE SENSITIVITY ANALYSIS OF THE OPTIMUM PLAN

COL NO.	SELECTED ALTERNATIVES	LEVEL OR AMOUNT	UNIT PRICE	LOWEST PRICE	LIMITING VARIABLE	HIGHEST COST	LIMITING VARIABLE
------------	--------------------------	--------------------	---------------	-----------------	----------------------	-----------------	----------------------

Selected Alternatives - alternative in solution, and unit.

Level Or Amount - number of units of alternative in optimal solution.

Unit Price - price or (cost) per one unit.

Lowest Price - lowest price or highest cost of a unit of the alternative, before the optimal plan changes.

Limiting Variable - The resource restricting the level of the alternative, or the activity that will enter the plan if the price of the selected alternative falls below the lowest price.

Highest Price - the highest price or lowest cost of a unit of the alternative before the optimal plan changes.

Limiting Variable - the resource restricting the level of the selected alternative, or the alternative that will leave the plan if the price of the selected alternative increases above the highest price.

Table 2

PRICE SENSITIVITY OF RESOURCES NOT COMPLETELY USED

ROW NO.	UNUSED RESOURCES	AMOUNT LEFT	ORIGINAL AMOUNT	LOWEST PRICE	LIMITING VARIABLE	HIGHEST PRICE	LIMITING VARIABLE
	resource not completely used	unused amount	original amount	the lowest price or highest cost	the alternative or resource that is limiting	the highest price or lowest cost	the alternative or resource that is limiting

Table 3

SENSITIVITY OF PLAN TO REJECTED ALTERNATIVES.

COL. NO.	ALTERNATIVES NOT SELECTED	COST OF INCLUSION	UNIT PRICE	LOWER LIMIT	BOUND VARIABLE	UPPER LIMIT	BOUND VARIABLE
----------	---------------------------	-------------------	------------	-------------	----------------	-------------	----------------

Alternatives Not Selected - alternatives that were not selected.

Cost Of Inclusion - loss incurred by forcing into solution (RMC).

Unit Price - per unit price of the rejected alternative.

Lower Limit - below this level, the cost of including this alternative in the optimal plan (RMC) would decrease.

Bound Variable - the alternative or resource that determines the lower limit in the plan.

Upper Limit - number of times you could force this alternative or resource into solution and only lose the original RMC amount.

Bound Variable - this variable puts a constraint on how high the upper limit can go before you would incur a greater loss each time you forced this alternative or resource into solution.

Table 4

RESOURCE SENSITIVITY RANGE AROUND ORIGINAL VALUES.

ROW NO.	LIMITING RESOURCES	RESIDUAL MARGINAL VALUE	ORIGINAL AMOUNT	LOWER LIMIT	BOUND VARIABLE	UPPER LIMIT	BOUND VARIABLE
---------	--------------------	-------------------------	-----------------	-------------	----------------	-------------	----------------

Limiting Resources - limiting resource in the optimal plan.

Residual Marginal Value - value you would be willing to pay for one more unit of this resource.

Original Amount - original amount.

Lower Limit - the lowest resource level before the residual marginal value product would increase.

Bound Variable - at the lowest resource level, the alternative or resource that will leave the plan.

Upper Limit - the amount where the limiting resource is no longer a constraint, another resource becomes the limiting factor.

Bound Variable - at the upper limit, this alternative or resource will leave the plan, and another alternative or resource will come into solution.

Additional Comments

The value in the column entitled "low price" is the price or cost the alternative can drop to without affecting the optimal plan. The value in the column entitled "high price" is the price or cost an alternative can increase to without affecting the optimal plan. The difference between these two values is called the "range". If the price or cost of an alternative changes and falls outside the "range", then a new optimal farm plan results. Changes can occur in prices within the "range" and the chosen farm plan is still the best.

If the range is very wide the optimal plan is "stable"; that is, the price or cost of an alternative could change drastically and the plan would still be the best.

If the range is very narrow, the optimal plan is "unstable"; that is, only a small change would be needed for a new optimal plan.

The price range given for each of the selected alternatives is valid only if all other prices remain unchanged.

This range can be very useful for your "peace of mind". If the range is wide, this would mean that even if your estimate of the cost or price of an alternative changes drastically, the plan is still the best.

Note To Potential Adopters Of MINNLP

MINNLP is a public good. It is available for transfer to other computer systems for a fee sufficient to cover the cost of transfer. No guarantees covering maintenance, enhancement or errors in use are made. The recipient is expected to share any modifications made or errors found with the author.

A technical appendix to this paper exists to aid in the process of installation on other hardware.