Staff Papers Series

P86-30 JULY 1986

MICROCOMPUTER ACCURACY IN SOLVING LINEAR PROGRAMMING PROBLEMS

WITH REDUNDANT CONSTRAINTS

BY

ALVARO SOLER AND EARL I. FULLER

Department of Agricultural and Applied Economics

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

MICROCOMPUTER ACCURACY IN SOLVING LINEAR PROGRAMMING PROBLEMS WITH REDUNDANT CONSTRAINTS

BY

ALVARO SOLER AND EARL I. FULLER*

* Former graduate research assistant and professor, Department of Agricultural and Applied Economics, University of Minnesota.

The authors wish to thank Jeff Apland, Jerry Fruin, Terry Roe and Burt Sundquist for their review, comments and criticism.

Research conducted under Minnesota Agricultural Experiment Station Project No. 14-036.

ABSTRACT

This study reports on how different microcomputer systems performed in the solution of two linear programming models purposely specified with redundant vectors. Comparisons were made to a Cyber 720 that used both a Fortran and Basic version of the same primal-dual algorithm. Results are mixed. But Microsoft Basic with double precision under CP/M on a Z80A processor performed at least equally well to the Cyber 720 provided that an appropriate essential zero value was specified. Different coefficient scaling schemes were also tested. The results should be of interest to all users of matrix inversion schemes on microcomputers. Extensions of the study to new hardware and software systems are encouraged.

This paper reports on the accuracy with which five different microcomputer systems solved two redundantly specified linear programming (LP) models. Both were purposively specified with redundant characteristics and/or in a "near singular" fashion in order to make them difficult to solve. Both models represent common applications of LP in agricultural economics. The results should be of interest to all analysts using microcomputers to invert matrices. Multiple regression analysis has shown similar problems.[1/] Consequently, the results should be of interest to analysts doing multiple regression analysis because of similar numerical computations.

THE PROBLEM

The difficulties associated with near singularity and/or redundant vector specification in linear programming and the resulting solution accuracy problems incurred in their solution on digital computers have been recognized for at least 20 years. A number of causative factors and prescriptive recommendations to avoid difficulties have been noted and described. [2/]

Conditions of near singularity, where two or more of the vectors contain numeric values throughout that are similar or are in a similar ratio to one another is a major reason accuracy problems occur. Difficulties are more likely to occur as problem size increases. They are also more likely as the number of price or right-hand side mapping solutions which are computed post-solution increases.

On the software side of the problem, various techniques have been developed to avoid difficulties. Recommendations usually include: (1) keep

---

1/ Bredhal, M.E. and Ann Mylander, 1977.
2/ Fuller, Earl I., 1962.

the decimal point centered (through scaling) for each of the coefficients within the matrix. (2) Keep the number of significant digits small. Use a directed solution technique. (3) Incorporate a "stop and numerically check" technique after a specified number of interations, followed with a directed restart from the corrected basis at that point. (4) Use an algorithm which looks for numerical dividends that are close to zero and set them equal to zero before continuing. Match the essential zero value specified in the check to the internal accuracy of the computer hardware and to the numerical size of the matrix coefficients. Older generation computers have a defined word size. With machines of a larger word size and inherent accuracy, the appropriate essential zero value can be closer to zero than it should be for machines of a smaller word size. (5) Use double precision variable specifications. (6) Watch for indications of cycling during the interative solution process. If vectors cycle by entering and leaving the basis several times, the likelihood that an accuracy problem is developing increases.

On the hardware and operating system side of the problem, there is little an end user can do. When calculations are carried out on computers, an amount of rounding or truncation error is accumulated as the analysis proceeds. This is a direct consequence of hardware and software design which define a finite length to numerical values.[3/]

The capacity of computers to retain numerical values is limited. Depending on the size of the byte or number of bits per byte, numerical bounds are imposed as to how many significant digits can be retained.

3/ Nash, John C., 1981.

Different computers' internal representation of floating point numbers vary with the design although the general structure--with varying arrangements--is specified by five characteristics: 4/

(a) sign of the number

(b) mantissa (with up to a certain fixed number of radix digits)

(c) an assumed position for the radix point (either immediately before or after the first non-zero digit of the mantissa)

(d) exponent (either signed or in excess of some shift value)

(e) whether final digits are specified by truncation or rounding

The ordering of the components of the general structure may vary from one system to another, but this is not relevant to the end results of calculations. What is relevant to the end results is the way stored values are operated upon. This is a system characteristic that cannot really be changed by the user except possibly by declaring double precison variables. Nevertheless, the user can still use the structure described above to determine the limits imposed on the results by the representation of floating point numbers in a particular system.5/

THE APPROACH

Interaction between the various factors can make it difficult to completely quantify all of the cause and effect relationships under all possible conditions for all possible factors. It would be extremely costly in time and computing resources to empirically determine the potential for difficulties and/or the likelihood of avoiding them for the possible combinations of computer operating

---

4/ Nash, John C., 1981.
5/ ibid.

systems, linear programming algorithms and all the other factors mentioned above. However, it was possible to do some testing under some conditions and, consequently, to report here the experience for the benefit of future potential users of linear programming on microcomputers.

Given these considerations, computing trials were designed to observe the solution characteristics of the Apple II computer with its 8 bit byte and 56K memory, using both the 6502 and the Z80A central processor units with one algorithm and two source languages. Applesoft Basic and Microsoft Basic, the latter in both single and double precision. The Vector 3005 computer with its 8 bit byte also at 56K uses the Z80A central processor unit chip with the same algorithm and one source language, Microsoft Basic. Two precision levels were also used. Comparisons were made to a CDC 720 Cyber mainframe computer using Fortran and Basic versions of the same algorithm operating in single precision.

The software package used was MINNLP and its microcomputer version SMALLP, an interactively controlled linear programming procedure. It is a primal-dual procedure and, consequently, is supposedly less likely to encounter an essential zero rounding problem than are simplex techniques. [6/] It operates on inequalities without slacks or artificials. All versions listed the pivoting vector and the objective value function at each interation. Forward and backward checks on the value of the objective function were calculated.

The complete set of trials included the solution to three different versions of a 41 x 50 low density farm planning model and a smaller but very dense feed mix model. For the larger model, six different hardware-software systems were tried. A seventh system was added for the dense model.

6/ Fuller, Earl I., 1981a.

The CDC Cyber 720 computer Fortran version was declared to be the comparison benchmark.

THE ANALYSIS

Table 1 summarizes the results of an analysis of each system's accuracy following Nash's discussion.[7/]

The larger and less dense model was a corn and soybean crop scheduling model. It controls a proper sequencing of operations. It also accounts for the impact certain field operations may have on total revenue if they are delayed in as much as yields decline decine as planting or harvest is delayed, etc.. The model can be regarded as a good test case since it considers most of the kind of activities a farm planner might include in a scheduling type model.[8/]

Two sets of equations in this model provide the redundancy of specifications. Labor time available for field work constitutes one set. Machine capacity by season provides the other. People are likely to over specify models in similar ways. This was purposely done in this case to provide a near singular matrix which would make the problem prone to cycling and more difficult to solve.

Cycling was defined here as the need for at least twice as many iterations as the number of rows in the matrix before reaching a solution. The model can be considered difficult not only because of the similarity of certain vectors, but also because of the size of the matrix.

The second model was a least cost hog ration formulation. The matrix for the feed mix model was 13 rows by 18 columns. It was more dense and tended to

---

7/ Fuller, Earl I., 1981a.

8/ The authors are indebted to Dr. Jeff Apland who suggested how a set of machine capacity constraints can be redundant to a set of labor or field time constants in this type model.

Table 1. Comparative Internal Accuracy Of Several Computer Systems

Hardware and Software Combinations	Digits Of Internal Machine Precision	Approximate Numerical Precision (Min. Value)
Cyber Fortran	49	3.55271E-15
Cyber Basic	48	7.10543E-15
Vector 3005 - Microsoft DP*	57	138.77788E-15**
Apple II - Microsoft DP	57	138.77788E-15
Vector 3005 - Microsoft SP*	25	0.05960E-10**
Apple II - Microsoft SP	25	0.05960E-10
Apple II - Applesoft	23	2.32831E-10

Note: All systems displayed two radix digits in the mantissa and truncated the results to the indicated accuracy level.

* DP = double precision; SP = single precision
** A Radio Shack Model I provided identical results.

be redundant in the activities, grain sources, as well as in the constraints, protein and amino acid specifications.

RESULTS

The results showed that the use of the CP/M Operating System utilizing the Z80A microprocessor and microsoft Basic with double precision provided, in these cases, solutions as accurate as those obtained by the larger, more sophisticated and costly computer system used as a benchmark.

The use of single precision on the CP/M Operating System or the use of the Applesoft system cannot be recommended when conditions are as extreme as in the larger model used for this trial. CP/M with single precision yielded unusable results in all runs. Applesoft yielded a few acceptable results but showed a poor overall performance. When conditions are less extreme as in the smaller models, CP/M with single precision still performed poorly but the Applesoft system provided acceptable results for essential zero values within the range from 1E-06 to 1E-04.

The results also showed the importance of adjusting essential zero values and scaling the coefficients. The essential zero value seems to have a greater importance in achieving accuracy while scaling appeared to affect the number of iterations required to get a solution. Scaling will not correct the use of too large an essential zero value. However, scaling used in conjunction with large essential zero values disallowed inaccurate solutions; the algorithm proclaimed the situation to be infeasible.

Five different essential zero values were tried. There was enough evidence to recommend the use of values not larger than 1E-05 since the solutions are not satisfactory when an essential zero value equal to or larger than 1E-04 was used.

However, this varies with the systems as CP/M with double precision showed signs of being more accurate then the Cyber in obtaining solutions, thus being less affected in its accuracy by essential zero values as large as 1E-04. For first trials, a value of 1E-06 is suggested.

Other system differences were also highlighted by the results. Most noticeable is the difference in the total number of iterations needed to solve the models. It is beyond the scope of this paper to totally explain this phenomenon, but is was noted that total iterations varied between systems and runs when solving both the large and small problems. These differences in the total number of iterations did not effect the solutions obtained when using the CP/M Operating System.

CONCLUSIONS

The economic significance of post solution analysis makes it worthwhile to pay special attention to certain factors which can influence the results. The results of this study also showed that:

(a) Users should be aware that the solution to complex LP models may differ depending on the computer system, the solution software and the essential zero value utilized. Furthermore, for any one combination of these factors, the solution may differ depending how the model's coefficients are numerically expressed.

(b) When running large models on microcomputer systems, it is best to use double precision when this feature is available.

(c) If a new or untried model is to be solved, it would be worthwhile to verify the constraint set subcalculations using different essential zero values. Start with a value somewhere in the range from 1E-07 to 1E-05; then solve again for a value larger than the largest value in that range and for a value smaller than the smallest value in the above range.

(d) This study did not provide enough evidence upon which to base sound conclusions about scaling. It is probable that different types of scaling will affect differently the solution results for any given model. Nevertheless, it has shown that scaling may help preclude infeasible results and scaling should be considered whenever coefficients in an equation vary greatly in magnitude.

(e) If possible, use a solution algorithm which has an option to print out for each interation the vector coming into the solution and the vector leaving the solution and the intermediate value of the objective function, as this will help in the detection of cycling. If vectors enter and leave the basis several times, the likelihood that cycling is occurring increases. Also, make sure that either the system or the software package itself checks for and warns a division by zero error.

(f) If cycling is indicated, check for near singularity or redundancy in the matrix. Try to avoid vectors which are redundant or which include essentially the same coefficients in almost the same numerical ratio from one to another vector.

POTENTIALS FOR EXTENSION OF THIS WORK

The rapid acquisition of microcomputers by agricultural economists and other analysts worldwide suggest that extensions of this work to other systems are in order.

It is difficult to empirically determine all of the possible cause-effect relationships between the factors influencing accuracy in the solution to a linear programming model. However, further testing should be carried out. Two areas that best lend themselves for future research are those concerned with:

(a) The influence of different hardware and software designs on accuracy. In these trials, the CP/M Operating System yielded identical results regardless of hardware, although differences in the time needed for the solution were consistently detected between computers.

(b) Influence of variations in applications software, essential zero value and scaling. The limited range of essential zero values utilized in these trials yielded enough evidence to support the equality in accuracy between the Cyber Fortran and the Vector CP/M and the Apple CP/M double precision systems in the solution to a large model when using small (1E-07 to 1E-05) essential zero values. The question remains whether even smaller essential zero values will not only maintain the accuracy of the different systems, but also improve other performance factors such as the number of iterations needed to reach the final basis.

In addition to these factors, there are some others that may also be taken into consideration. These trails did not show a large difference between Fortran and Basic, but will it be the same if some other source language is utilized?

Yet another issue is related to the forthcoming generations of 16-bit and 32-bit byte microcomputers using other Operating Systems. Will they also match or exceed the accuracy attained on the Cyber and the CP/M double precision systems?

Finally, questions on how these factors influence the results obtained from the use of matrix inversion based statistical packages deserve empirical testing. Bohem, Menkhaus and Penn,[9/] Bredahl and Mylander, [10/] and Weingarten [11/] among

---

9/ Bohem, William T., D.J. Menkhaus and J.B. Penn, 1976.
10/ Bredahl, Maury E. and Ann Mylander, 1977.
11/ Weingarten, Hyman, 1978.

others, have published results of accuracy tests for different least squares computer algorithms. These studies emphasize the difference between algorithms.

Given the computational similarities between linear programming and multiple regression and, given the growing trend towards the use of microcomputers, it would be worthwhile to test those algorithms in trials similar to those in this work. If microcomputer versions of those algorithms are not available then the trials might include some of the microcomputer commerical or public good statistical packages already available.

REFERENCES

Bohem, William T., D.J. Menkhaus and J.B. Penn. "Accuracy Of Least Squares Programs: Another Reminder." American Journal of Agricultural Economics. 58 (1976): 757-60.

Bredahl, Maury E., and Ann Mylander. The Accuracy Of Regression Programs - University of Minnesota. Institute of Agriculture, Forestry and Home Economics. Staff Paper P77-2, 1977.

Fuller, Earl I. Farm Oriented MINN-AIDS. CDA 60.0-60.5. Department of Agricultural and Applied Economics, University of Minnesota, 1981a.

________, and P.R. Hasbargen. GENMTX User's Guide. CDA 418. Department of Agricultural and Applied Economics, University of Minnesota, 1972.

________, Linear Programming Operating Procedures For The 1620 Using LP 10.1.006 Card Output Option, Unnumbered Mimeo, University of Massachusetts, 1962.

________, User's Guide For MINNLP, Department of Agricultural and Applied Economics, University of Minnesota, 1981b.

Nash, John C. "The Micromathematician: Determining Floating-point Properties." Interface Age, May 1981, pp. 30-33.

Soler, Alvaro J. An Evaluation Of Microcomputer Accuracy In The Solution Of Difficult Linear Programming Problems., Plan B Paper, Department of Agricultural and Applied Economics, University of Minnesota, 1982.

Weingarten, Hyman. A Note On The Accuracy Of Computer Algorithms For Least Squares. Agricultural Economics Research. U.S. Department of Agriculture, Econ., Stat., and Coop. Svce., V.30(1), 1978:49-52.