Comparative Study of Nonlinear Optimization Programs

Klaus Schittkowski
Institut fur Angewandte Mathematik und Statistik
Universitat Wurzberg
D-87, Wurzberg, West Germany

Purpose: At the present, we are testing and comparing optimization codes for solving the general nonlinear programming problem

$$\min f(x)$$
$$g_j(x) = 0 \ , \ j=1 \ , \ \ldots \ , \ m_1$$
$$g_j(x) \geq 0 \ , \ j=m_1+1 \ , \ \ldots \ , \ m$$
$$x_1 \leq x \leq x_u$$

where f, g_j (j=1 , ... , m) are signomial functions.

The programs: To date, 33 qualified programs have been submitted for comparison. 19 of them are in machine readible form and we hope to receive the remaining ones in the near future. 13 programs are completely tested using 240 test runs and 80 randomly chosen test problems. Only two of these codes are executed in two different versions. The test problem generator is described in (1) and intermediate test results are presented in (2) and (3) based on results of 5 or 6 programs, respectively.

The performance criteria: The randomly generated test problems with predetermined solutions allows one to measure accuracy, efficiency, global convergence and reliability of each optimization program. It is possible, using this generator, to relate the efficiency of a code (execution time, number of function and gradient calls) to the reached accuracy. (2) and (3) present details of how the performance criteria are used to evaluate the programs.

The test examples: Three major classes of test problems will be considered. The first set consists of the 80 randomly generated problems mentioned above and explained in detail in (2) and (3). The second set consists of 80 other randomly chosen test problems. These test problems possess special

structures such as ill-conditioning, indefinite Hessians, and degeneracy. A third set of test examples consists of a collection of 126 problems which are found in the literature and have been used in the past to test optimization software, see (4) and (5) for more details.

In summary, the optimization programs which will be examined in this test effort will undergo about 450 test runs in contrast to at most 30 test runs performed in earlier studies.

Publications:

(1) K. Schittkowski, Randomly generated NLP test problems with predetermined solutions, submitted for publication.

(2) ____, A numerical comparison of optimization software using randomly generated test problems - and intermediate balance. Preprint No. 43, Institut fur Angewandte Mathematik und Statistik, Universitat Wurzburg.

(3) ____, A numerical comparison of optimization programs using randomly generated test problems, to appear: Proceedings of the IFIP Working Conference on Performance Evaluation, L. D. Foskick, ed., Baden Austria, Dec. 1978, North Holland.

(4) W. Hock and K. Schittkowski, Test problems for the solution of nonlinear programming problems, Part 1, Preprint No. 44, Institut fur Angewandte Mathematik und Statistik, Universitat Wurzburg.

(5) ____, Test problems for the solution of nonlinear programming problems, Part 2, Preprint No. 45, Institut fur Angewandte Mathematik und Statistik, Universitat Wurzburg.

xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

Development and Comparative Study of Optimization Techniques for Engineering Design

Kenneth M. Ragsdell
Eric Sandgren

School of Mechanical Engineering
Purdue University
West Lafayette, Indiana

Project Goals: The major goals of this research project are to perform a technical comparison of the world's leading nonlinear programming algorithms, and to develop a new and effective algorithm. The results of this work should be of particular interest to designers of engineering systems, since many of the test problems are taken from modern engineering practice.

Comparative Study [1]:

Thirty-five codes were tested using thirty test problems. Of these thirty test problems, twenty-two came from modern engineering design applications. Several problems and codes were contributed by interested industrial colleagues. The remaining problems in the test set appear in previous comparative studies conducted by Colville [2] and Eason and Fenton [3] and were included in a spirit of continuity.

The problems of Colville and Eason and Fenton were used as an initial test for all codes, and as a result of poor performance, eleven codes were eliminated from further study. The problems range from two to forty-eight design variables and from four to seventy-five constraints. Various linear approximation techniques, and interior and exterior penalty function methods are included in the code set.

Various convergence characteristics were compiled for each code as applied to each problem, and the codes were rated on their ability to solve problems in a reasonable amount of computation time. The effect of problem parameter variation on solution time for the various algorithm classes was studied. The problem parameters which were varied are: the number of design variables, the number of inequality constraints, the number of equality constraints and the degree of nonlinearity of the objective function and constraints.

G. *TXLPGN: A Linear Programming Problem Generator*

TXLPGN is a FORTRAN program which generates linear programming problems having optional user specified structure. This code was obtained directly from a paper by A. Charnes, W. M. Raike, J. D. Stutz, and A. S. Walters entitled, "On Generation of Test Problems for Linear Programming Codes," Communications of the ACM, Vol. 17, No. 10, October 1974. Minor coding changes were necessary to make the code operational on the UNIVAC 1108 at the National Bureau of Standards. For more information about this code, contact either Karla L. Hoffman or Patsy B. Saunders, Center for Applied Mathematics, National Bureau of Standards, Washington, DC 20234.

We do not consider this list complete and would appreciate any additional information on generators and/or test problems. Please send this information to:

Richard P. O'Neill
Department of Energy
Energy Information Administration
Applied Analysis
1200 Pennsylvania Avenue, NW
Washington, DC 20461

Both generators enable automatic generation of problems with specified characteristics and (for one generator) known, unique and controllable solutions.

E. *"Randomly Generated NLP Test Problems with Predetermined Solutions."* *Klaus Schittkowski, Institut fur Angewandte Mathematic and Statistik, Universitat Wurzburg, Am Hubland, D-87 Wurzburg, West Germany.*

This paper presents a method for generating test problems for the evaluation and comparison of nonlinear programming software. The functions considered are signomials with randomly chosen coefficients and exponents. Since the optimal solution is predetermined, it is possible to study the computed accuracy, especially the number of correct digits, the error of the objective function value obtained, and the validity of the constraints. Parameters are set by the user and classify the desired problem. It is possible to generate classes of small and dense, big and sparse, or highly nonlinear test problems. Furthermore, it is possible to specify the degree of nonlinearity of each of the constraints.

F. *"RIP: A Test Problem Generator for Mixed Integer Programming Problems."* *Ronald L. Rardin, Industrial and Systems Engineering, Georgia Institute of Technology, Atlanta, GA 30334 and Benjamin Lin, Rutgers University, Department of Mechanical and Industrial, and Aerospace Engineering, Piscataway, NJ 08854.*

Problems generated by RIP are bounded integer (or mixed integer) linear programs. Thus, users may select any mix of continuous, integer, and 0-1 variables, and constraints may be any mix of equality and inequality. Coefficients in the constraint matrix can be nonnegative or of mixed sign. Among the other RIP problem parameters under user control are the density of the nonzero entries in the constraint matrix and several aspects of the relationship between the linear programming relaxation of the generated problem and its integer solution. The linear and the integer solution values are explicitly reported at the completion of problem generation.

The RIP generator is designed as a subroutine and is written in machine-independent FORTRAN. Persons interested in obtaining further information on RIP should write to Ronald L. Rardin.

The major conclusion of the study is that the linearization methods, in particular the generalized reduced gradient methods, are far superior in every way to the transformation methods tested. Of the transformation methods tested, those employing a variable metric unconstrained method were found to be superior.

A Combined Transformation/Linear Approximation Method

A new optimization algorithm was developed, which combines two well known algorithm classes; transformation and linear approximation methods. The transformation method employed was an implementation of the method of multipliers. Two linear approximation methods were studied; the generalized reduced gradient and repetitive linear approximation. The combined algorithm approach exploits the rapid movement to the vicinity of the constrained minimum associated with the method of multipliers, and the rapid convergence and sharp definition of the constrained minimum by the generalized reduced gradient method once its neighborhood has been located. The combined algorithms were applied to the test problem set and outperformed all other codes tested.

Papers planned:

Sandgren and Ragsdell, "The Utility of Nonlinear Programming Algorithms: A Comparative Study," to appear.

Ragsdell, "On Some Experiments Which Delimit the Utility of Nonlinear Programming Algorithms," invited paper for COAL session at the ORSA/TIMS national meeting, November 13-15, 1978.

Sandgren and Ragsdell, "A Combined Transformation and Reduced Gradient Algorithm," to appear.

References:

[1] Sandgren, E., "The Utility of Nonlinear Programming Algorithms," Ph.D. Thesis, School of Mechanical Engineering, Purdue University, Dec. 1977.

[2] Colville, A. R., "A Comparative Study of Nonlinear Programming Codes," Proceedings-Princeton Symposium on Math. Prog., Kuhn, ed., 1970, pp. 487-501.

[3] Eason, E. D. and Fenton, R. G., "A Comparison of Numerical Optimization Methods for Engineering Design," Journal of Engineering and Industry, Trans. ASME, Series B, Vol. 96, No. 1, February 1974, pp. 196-200.

xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

A Computational Study of Floyd's Algorithm

D. R. Shier
Center for Applied Mathematics
National Bureau of Standards
Washington, DC 20234

Floyd's algorithm provides a simple and elegant technique for finding all shortest paths in a network. We study the empirical computational complexity of five different computer codes for realizing this algorithm, using a class of randomly generated test problems. The focus of the present work is to investigate objective and reproducible measures of computational effort, as well as to indicate how rather substantial reductions in this computational effort (as much as 30%, asymptotically) can be achieved. Both CPU time (obtained using totally dedicated computer runs) and predicted computation time (based on cycle times for elemental computer instructions) are discussed and compared. These results indicate a strong interaction between code, computer, compiler, and some empirical measures of computational effort. Moreover, it is found that matrix-subscripting operations (often ignored in performing "operation counts" for assessing computational complexity) are profoundly important in explaining observed computational behavior.

capability of generating the same problem on different machines.

The user has control over the following parameters: the density, distribution, and type of the generated values in the constraint matrix; the type of constraints (inequality/equality) and the number that are not binding in the optimal solution; the type of optimal solution (i.e. degenerate, unique, or nonunique) and the status of the variables in the solution vector (e.g., at upper bound); and, the decision to generate or supply the constraint matrix and/or the solution vector. The problem can be written on an external device in MPS or SHARE format. Accompanying MPGNR is a subroutine that will evaluate the function and its gradients, freeing the user from writing a routine to perform those functions.

The user's manuals, and a paper describing the system in detail and providing some test results may be obtained by writing to the author. The code may be obtained by including a magnetic tape.

C. *"NETGEN: A Program for Generating Large Scale (Un)capacitated assignment, transportation, and minimum cost flow network problems." D. Klingman, A. Napier, and J. Stutz.*

This code can generate capacitated and uncapacitated transportation and minimum cost flow network problems, and assignment problems. In addition to generating structurally different classes of network problems, the code permits the user to vary structural characteristics within a class.

Darwin Klingman is about to enhance and revise NETGEN and he has asked that any suggestions on new problem classes (e.g., fixed charge) or additional features be sent to him. His address is University of Texas, BEB-600, Austin, TX 78712.

D. *"POLY1 AND POLY2: Generators for Polynomial Approximation Problems in Testing Linear L_1 Codes." Paul Domich and Douglas R. Shier, National Bureau of Standards, Center for Applied Mathematics, Washington, DC 20234.*

Polynomial approximation problems represent a class of specially structured problems which are frequently encountered in empirical curve-fitting. Two generators for creating such problems have been developed, implemented, and used in the testing of linear L_1 codes.

E. *"A Collection of Test Problems for Discrete Linear L_1 Data Fitting."* *Patsy B. Saunders and Douglas R. Shier, National Bureau of Standards, Washington, DC 20234.*

This paper assembles 27 test problems representing a variety of situations in which least absolute deviations (or L_1) data fitting occurs. These problems were collected from the literature, from the authors of several L_1 codes, and from examples encountered in practice. Optimal L_1 solutions to these problems (objective function value and solution vector) were obtained using a double-precision computer code designed for checking the Kuhn-Tucker conditions and for performing an accurate reinversion of the optimal basis. Special problem characteristics such as alternative optima, degeneracy, and rank loss are also noted. This set of test problems can be useful in evaluating and improving the performance of L_1 codes as well as providing suggestions as to types of problem structures which might be mimicked by problem generators.

TEST PROBLEM GENERATORS

A. *"L1GNR: A Test Problem Generator for Testing Linear L_1 Codes."* *Karla L. Hoffman and Douglas R. Shier, National Bureau of Standards, Washington, DC 20234.*

This paper describes a procedure that generates test problems for L_1 estimation of the model $y = Xb + u$. The procedure allows user control of problem dimensions, solution vector, residual distribution, column rank, row repetitions, and degeneracy. An important feature is that problems are guaranteed to have unique solutions whenever X has full rank.

B. *"MPGNR: A Generator of linear and nonlinear test problems with quadratic and linear functions."* *Richard P. O'Neill, Department of Energy, Energy Information Administration, Applied Analysis, 1200 Pennsylvania Avenue, Washington, DC 20461.*

MPGNR is a subroutine system used to generate linear and nonlinear programs with quadratic and linear functions of variable size with predetermined optimal solutions. MPGNR is written in ANSI FORTRAN and uses a machine independent pseudo-random number generator to provide the

NOTICES

A principle activity of COAL is encouraging people to conduct well-designed computational studies, and to serve as a focal point for disseminating information about those efforts. To this end, COAL sponsors sessions at professional meetings.

(1) "Recent and Future Developments in Mathematical Programming Systems" at the Joint National TIMS/ORSA Meeting in New Orleans, April 30, 1979. This session will be jointly chaired by Richard H. F. Jackson and Richard P. O'Neill. The following papers were presented:

"Recent and Future Developments in Sperry Univac's FMPS Product," Edward H. McCall, Sperry Univac, P. O. Box 43942, St. Paul, MN 55164.

"Database Methods in Honeywell's MPS Software," G. L. Despain, Honeywell Information System, Inc., Phoenix, AZ.

"APEX: Past and Future," C. B. Krabek, R. J. Sjoquist, and D. C. Sommer, Control Data Corp., Minneapolis, MN 55440.

"The Extended Control Language of MPSX/370," Kurt Spielberg, IBM, White Plains, NY 10604.

"MP Developments on Burroughs Computers," David M. Carstens, Burroughs Corporation, Radnor, PA 19087.

"Mathematical Programming Systems, Subsystems, and Supersystems," John S. Bonner, Bonner and Moore, Houston, TX 77002.

"Recent and Future Developments in MPSIII," David S. Hirshfeld, Ketron, Inc., Arlington, VA 22209.

"The Matrix Generator/Generator SCICONIC Mathematical Programming System," Roy Harvey, Control Analysis Corporation, Palo Alto, CA 94304 and J. Forrest, Scicon Computer Services, U. K.

"OMNI - A Versatile Language for Math Programming and Other Structured Problems," C. A. Haverly, Haverly Systems, Inc., Denville, NJ.

A PANEL DISCUSSION ON THE SAME TOPIC FOLLOWED THESE PAPERS. The panelists include: Thomas White, Shell Development Corporation; Harvey Greenberg, Department of Energy; Milt Gutterman, Standard Oil of Indiana; Eli Hellerman, Bureau of the Census; G. L. Despain, Honeywell Information System, Inc.; C. B. Krabek, Control Data Corp.; R. J. Sjoquist, Control Data Corp.; Kurt Spielberg, IBM: David M. Carstens, Burroughs Corp.; John S. Bonner, Bonner and Moore; David S. Hirshfeld, Ketron, Inc.; Roy Harvey, Control Analysis Corp.; J. Forrest, Scicon Computer Services; and C. A. Haverly, Haverly Systems, Inc.

(2) "Testing and Validating Mathematical Programming Software" at TIMS XXIV International Meeting in Hawaii, June 18-22, 1979. Richard H. F. Jackson and Richard P. O'Neill will jointly chair this session. The following papers will be presented:

"A Survey of Efforts to Test and Validate Mathematical Programming Software," Richard H. F. Jackson, Center for Applied Mathematics, National Bureau of Standards, Washington, DC 20234.

"Processing Time: An Accurate Measure of Code Performance?," Karla L. Hoffman and Richard H. F. Jackson, Center for Applied Mathematics, National Bureau of Standards, Washington, DC 20234.

"The Importance of Performance Indicators in Code Testing," Harlan T. Crowder, IBM Research Center, Yorktown Heights, NY 10598 and Patsy B. Saunders, Center for Applied Mathematics, National Bureau of Standards, Washington, DC 20234.

"A Comparison of Real-World Linear Programs and their Randomly Generated Analogs," Richard P. O'Neill, Office of Analytic Methods, Department of Energy, Washington, DC 20461.

"Testing and Packaging of Scientific and Engineering Software to Achieve Portability," M. K. Butler, H. S. Edwards, L. R. Eyberger, P. L. Johnson, M. Legan, L. L. Reed, and A. J. Strecok, National Energy Software Center, Argonne National Laboratory, Argonne, IL 60439.

(3) "Progress in Evaluating Mathematical Programming Algorithms" at Mathematical Programming Symposium X in Montreal, Canada, August 1979. This session is chaired by John M. Mulvey, School of Engineering and Applied Science, Princeton University, Princeton, NJ 08544.

"Can a General Purpose NLP Code Find Happiness in a GP World?," J. E. Fattlar, G. V. Reklaitis, and K. M. Ragsdell, Purdue University, West Lafayette, IN.

"A Methodology for Testing Mathematical Programming Software," P. D. Domich, K. L. Hoffman, R. H. F. Jackson, P. B. Saunders, and D. R. Shier, U. S. National Bureau of Standards, Washington, DC 20234.

"Testing Mathematical Programming Algorithms on Randomly Generated Polyhedra," Jan Telgen, Erasmus University, Rotterdam.

"Performance Evaluation of Non-linear Programming Codes as a Multi-Criterion Decision Problem," Freerk Lootsma, University of Technology, Delft, The Netherlands.

"Processing Time: An Accurate Measure of Code Performance," K. L. Hoffman and R. H. F. Jackson, U. S. National Bureau of Standards, Center for Applied Mathematics, Washington, DC 20234.

"On Some Experiments Which Delimit the Utility of Nonlinear Programming Methods for Engineering Design," Eric Sandgren, IBM Office Products Division, Lexington, KY and K. M. Ragsdell, School of Mechanical Engineering, Purdue University, West Lafayette, IN 47907.

at several points against published results and a validation process was used to check the values of the objective function, constraints, and gradients.

The problems were collected from various sources and many of them have been used by other authors in published results of their algorithm testing. This report should also be useful in an educational setting to provide students with experience in nontrivial problems. Listings of the IBM FORTRAN code are included in this report.

C. *"Test Examples for the Solution of Nonlinear Programming Problems, Part I." W. Hock and K. Schittkowski, Mathematische Institut der Julius-Maximilians-Universitat, Wurzburg, West Germany.*

The numerical solution of constrained nonlinear programming problems with continuously differentiable functions requires the development of an extensive set of test examples. The purpose of this paper is to give a classification and documentation scheme for such test problems and to present a user's guide to implement the FORTRAN subroutines which contain these problems. This paper presents a collection of 120 test examples which are found in the literature and which could be used for further experimentations in mathematical programming.

The coding of the test problem functions together with their gradients requires about 10,000 FORTRAN statements making it impossible to publish the corresponding listings. The authors are willing to provide any interested user with all subroutines on magnetic tape.

D. *"Test problems used in a study of optimization techniques for Engineering Design." Kenneth Ragsdell and Eric Sandgren.*

A goal of this research was to perform a technical comparison of the world's leading nonlinear programming algorithms. Thirty-five codes were tested using thirty test problems. Of these thirty test problems, twenty-two came from modern engineering design applications. Several of the problems were contributed by interested industrial colleagues. The remaining problems in the test set appear in previous comparitive studies conducted by Colville, and Eason and Fenton, and were included in the spirit of continuity.

For more information about these test problems, contact Kenneth M. Ragsdell, School of Mechanical Engineering, Purdue Univ., West Lafayette, IN.

TEST PROBLEMS AND TEST PROBLEM GENERATORS

Currently, COAL is undertaking an effort to collect information on test problems and generators for math programming algorithms. This will include an address list of those who distribute test problems or generators. To ate, the collection consists of the SHARE test problems, several Department of Energy applications, and the problems presented below. Several of the commercial math programming vendors have promised to contribute their non-confidential problems. We appreciate this cooperation.

The following contains brief descriptions and a contact for each test problem set and problem generator currently in the COAL database.

TEST PROBLEMS

A. *"Testing unconstrained software." Gorge Moré, Burton S. Garbow, and Kenneth E. Hillstrom. Argonne National Laboratory, Argonne, Illinois 60439.*

Much of the testing of optimization software is inadequate because the number of test functions is small or the starting points are close to the solution. In addition, there has been too much emphasis on measuring the efficiency of the software and not enough on testing reliability and robustness. To address this need, we have produced a relatively large but easy-to-use collection of test functions and designed guidelines for testing reliability and robustness of unconstrained software.

Copies of this report can be obtained by writing to the authors.

B. *"Test problems for Constrained Nonlinear Mathematical Programming Algorithms." Larry W. Cornwell, Patricia A. Hutchison, Michael Minkoff, and Hilbert K. Schultz. Applied Mathematics Division, Argonne National Laboratory, Argonne, Illinois 60439.*

The report presents a collection of 32 constrained nonlinear programming problems for use in testing optimization algorithms. The problems vary in size from two variables to one hundred variables with various combinations of linear/nonlinear constraints and objective functions. IBM FORTRAN IV programs have been written to provide function values and gradients for the objective function and constraints. Each coded problem has been checked

"Performance Indicators for Evaluating Math Programming Software," H. T. Crowder, IBM Research Center, Yorktown Heights, NY 10598 and P. B. Saunders, U. S. National Bureau of Standards, Center for Applied Mathematics, Washington, DC 20234.

(4) "Testing and Evaluation of Math Programming Systems" at the Mathematical Programming Symposium X in Montreal, August 1979. This session is jointly chaired by Richard H. F. Jackson and Richard P. O'Neill.

The papers in this session will center on the methods used in testing and evaluating large-scale commercial Math Programming Systems. The speakers in this session include C. A. Haverly, Haverly Systems, Denville, NJ; David M. Carstens, Burroughs Corporation, Radnor, PA; J. B. Creegan, Ketron, Inc., Arlington, VA; and Michel Couthier, IBM, France.

(5) "Mathematical Programming Software" at the Joint ORSA/TIMS Meeting in Milwaukee, October 15-17, 1979. This session is chaired by Michael Minkoff, Argonne National Laboratory, Argonne, IL. Papers in this session include:

"Implementation and Evaluation of a New Algorithm for Minimization," Danny C. Sorensen, University of Kentucky, Dept. of Mathematics, Lexington, KY 40506.

"Evaluation of Mathematical Software," Harlan Crowder, IBM Research Center, Yorktown Heights, NY.

"Implementation and Testing of Optimization Software," Gorge Moré, Argonne National Laboratory, Argonne, IL.

"Design and Implementation of an Experimental Large-Scale Optimization System," Gordon H. Bradley and Gerald G. Brown, Naval Postgraduate School, Monterey, CA and Glenn W. Graves, University of California at Los Angeles, Los Angeles, CA.

(6) On December 11-15, 1978 a Working Conference on Performance Evaluation of Numerical Software, organized by the IFIP, Working group on Numerical Software was held at Baden (Austria). One morning sesssion was devoted to performance evaluation in optimization and nonlinear equations. During a panel discussion the activities of COAL were discussed, such as the guidelines for reporting computational results (prepared for MPS, ORSA, and ACM-TOMS) and the proposal for classification and documentation of test problems in the field of nonlinear programming. Proceedings of this conference can be obtained by writing L. D. Foskick, ed., IFIP Working Conference on Performance Evaluation, Baden, Austria.

EDITORIAL

To All Friends of COAL:

As mentioned previously, the purpose of this newsletter is to provide an informal forum for the exchange of ideas related to MP algorithm and software evaluation. Toward this end, Harlan Crowder and Patsy Saunders have designed a questionnaire to determine the attributes and characteristics which people consider important when performing such evaluations.

Because I believe this survey will help further COAL's goal of determining proper methodology for assessing and comparing mathematical programming algorithms, I encourage each of you to complete and return this questionnaire.

I would also like to take this opportunity to again request that you send me any summaries of work-in-progress and discussion papers related to MP algorithm and software evaluation.

Karla L. Hoffman
Editor of Newsletter

Chairman's Comments

I was delighted by the response to our solicitation in the first newsletter for "friends of COAL". We have received almost 200 names from countries as distant as Chili, Japan, South Africa, and Australia. Evidently, there is broad interest in evaluating mathematical programming algorithms. This newletter is planned as an informal outlet among this global audience. So continue to respond. We are especially interested in suggestions about the content of future newsletters.

Besides the journal, the major preoccupation of the Mathematical Programming Society is the International Symposium. COAL will be active in the upcoming August symposium in Montreal. The events are described on page 13.

Looking further into the future, COAL is planning a 2-3 day workshop on evaluating mathematical programming techniques. By bringing together people who have performed computational analyses (either MP or general computations), we hope to improve our understanding of how MP algorithms should be compared and evaluated. The tentative date for this workshop is January 1981 in Boulder, Colorado.

COMMITTEE ON ALGORITHMS of the MATHEMATICAL PROGRAMMING SOCIETY

The Tenth International Symposium on Mathematical Programming

The Mathematical Programming Society announces that the Tenth International Symposium on Mathematical Programming will be held August 27-31, 1979, on the campus of McGill University in Montreal, Quebec, Canada.

Its subject matter will be the whole range of theoretical, computational, and applicational aspects of mathematical programming. In addition to many research and expository papers in these areas, "mini-courses" of several hours duration will be given in the subjects of computational complexity in optimization, fixed-point calculation, and multi-objective optimization.

A general meeting of the Society will be held in Montreal on August 29. Members are invited to propose agenda items to the Executive Committee of the Society (Chairman: Dr. A. C. Williams, Computer Science Department, Mobil Technical Center, Box 1025 Princeton, New Jersey 08540 U.S.A.)

For registration, information, and all other matters, address the

Symposium Secretariat
Mathematical Programming X
772 Sherbrooke Street West
Montreal, Quebec, Canada H3A 1G1

xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

The Eleventh International Symposium in Mathematical Programming

The Mathematical Programming Society is beginning its planning for the Eleventh International Symposium in Mathematical Programming, to be held in 1982 at a suitable site *not* in North America. All parties who might consider acting as hosts to this event are asked to make their interest known to the Society by communicating with:

Professor Alex Orden
Chairman, MPS Symposium Advisory Committee
Graduate School of Business
University of Chicago
5836 Greenwood Avenue
Chicago, Illinois 60637

xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

APPLICATION FOR MEMBERSHIP

Mail to: MATHEMATICAL PROGRAMMING SOCIETY
c/o International Statistical Institute
428 Prinses Beatrixlaan
2270 AZ Voorburg, The Netherlands

Cheques or money orders should be made payable to The Mathematical Programming Society in one of the currencies indicated below.

☐ I wish to enroll as a member of the Society. My dues payment for 1979, which covers subscription to volumes 16 and 17 of MATHEMATICAL PROGRAMMING is enclosed: Dfl. 92 (or $40.00 or £22.00 or Sw.Fr.70.00 or FF185.00 or DM85.00).

☐ As a member of the Society I wish to subscribe to the serial edition of MATHEMATICAL PROGRAMMING STUDIES, Nos. 10, 11 and 12. Payment is enclosed: Dfl. 49 (or $23.00 or £12.00 or Sw.Fr.37.00 or FF99.00 or DM45.00).

My subscription(s) is (are) for my personal use and not for the benefit of any library or other institution.

Name (printed): .... Signature: ....

Mailing address (please print): ....

MATHEMATICAL PROGRAMMING SOCIETY
Committee on Algorithms
NEWSLETTER

June 1979 — Karla L. Hoffman, Editor

Contents:

ENCLOSURE: Survey on MP Software Performance Indicators

John M. Mulvey
School of Engineering/Applied Science
Princeton University
Princeton, New Jersey 08544

THIRD-CLASS
PRINCETON N.J. JUN 13'79
U.S. POSTAGE 33
U.S. POSTAGE 20
PB METER 336030