Mathematical Programming 10 (1976) 192–213.
North-Holland Publishing Company

A SET OF GEOMETRIC PROGRAMMING TEST PROBLEMS AND THEIR SOLUTIONS *

Ron S. DEMBO **
University of Waterloo, Waterloo, Ont., Canada

Received 26 July 1974
Revised manuscript received 4 February 1975

This paper attempts to provide a set of standard test examples for researchers working in the area of geometric programming and general nonlinear, continuous, nonconvex programming algorithms. The examples consist partly of applications of nonlinear programming that have appeared in the literature and partly of original geometric programming applications. Solutions to all the problems are provided as well as the starting points from which these solutions were computed. Other computationally important aspects such as tolerances and degree of accuracy with which these problems were solved, are also included.

Introduction

For some time now there has been a need for a set of standard test problems that would provide a basis for the comparison of the relative efficiencies of various geometric programming algorithms. Standard test problems in areas such as integer programming [9] and general nonlinear programming [4] have been available for a number of years and have been used for comparison purposes. However, the sets of problems that are available lack information on computational accuracies and tolerances that are essential for the comparison of algorithm performance. In this set of problems we have attempted to overcome this by specifying tolerances as well as the stopping criterion with which each optimal solution was calculated.

The problems we have assembled here consist partly of applications of nonlinear programming that have appeared in the literature and partly of original geometric programming applications. They may be solved by

* This work was supported in part by Canada Council Grant #S74-0148 and National Research Council of Canada Grant #084-6319-32.

** At present, Assistant Professor, Production and Management Science Area, Faculty of Business, Mc. Master University, Hamilton, Ont.

any nonlinear programming method that can be used to solve continuous, nonlinear, inequality-constrained optimization problems.

Notation

The following notation is used consistently throughout this paper. All problems are assumed to be of the form

$$\text{minimize} \quad g_0(x)\,, \tag{1}$$

$$\text{subject to} \quad g_k(x) \leqslant 1\,, \qquad k = 1, ..., \mathrm{NK}\,, \tag{2}$$

$$l_i \leqslant x_i \leqslant u_i\,, \qquad i = 1, ..., \mathrm{NVAR}\,, \tag{3}$$

where the functions $g_k(x)$ $(k = 0, ..., \mathrm{NK})$ are signomial functions of the form

$$g_k(x) = \sum_{j \in J_k} c_j \prod_{i=1}^{m} x_i^{a_{ij}}\,. \tag{4}$$

The coefficients c_j and exponents a_{ij} are given real numbers and u_i and l_i are given *positive* upper and lower bounds on the variable x_i.

The problem constraints are divided into two classes. Signomial constraints (2) and simple bounding constraints (3). The number of signomial constraints is given by NK and the number of variables by NVAR. The set J_k is used to label the terms appearing in the k^{th} signomial constraint. Terms are labeled consecutively starting with those appearing in the objective function $g_0(x)$.

Since it is expected that the collection of problems presented here might be used for purposes of comparing various codes for geometric and general nonlinear programming, a distinction is made between two different types of bounding constraints (3). The reason for this is that for certain algorithms, convergence is contingent on the compactness of the feasible region of the problem at all stages of computation toward an optimal solution (see for example [8]). On the other hand, many algorithms (see for example [2]) do not have efficient mechanisms for handling simple bounding constraints and would therefore be heavily penalized if forced to include variable bounding constraints for every problem solved. We have attempted to overcome this difficulty in the following manner. Since all the problems presented here (with the exception of Problem 8) are taken from practical applications of nonlinear programming, the variable bounding constraints that are an integral part of the system being modelled are distinguished from bounding constraints that are artificially imposed to guarantee compactness.

It is expected therefore, that when these problems are used for comparative purposes, *the bounds corresponding to the physical model of the system will always be included as part of the problem formulation* (regardless of whether they are active at the optimal solution or not). For codes requiring all variables to be bounded from above and below, the variable bounds specified with each problem must be used in order to standardize results.

The total number of terms in a problem is denoted by NTERMS. For each problem this is computed as the total number of terms appearing in the objective function and signomial constraints *plus* the terms corresponding to variable bounding constraints that result from the mathematical model of the system.

An important concept in geometric programming is the 'degrees of difficulty' (DD) of a problem. This is defined as:

$$\mathrm{DD} = \mathrm{NTERMS} - (\mathrm{NVAR} + 1)$$

For each problem the value of DD is stated and is computed according to the above formula.

Lastly, since these problems are intended for providing a standard for comparing the relative efficiency of various algorithms, some measure of the required accuracy of solution must be specified. We do this by specifying 3 tolerances, EPSCON, EPSCGP and EPSLP; where EPSCON is a constraint tolerance, EPSCGP is a convergence tolerance and EPSLP is used to determine whether an element in the LP[1] tableau is essentially zero or not. These tolerances are discussed in more detail below.

At some given solution $\hat{x}$, the constraints $g_k(x) \leqslant 1$ are considered to be satisfied if and only if:

$$\text{(A)} \qquad g_k(\hat{x}) \leqslant 1 + \mathrm{EPSCON}, \qquad k = 1, ..., \mathrm{NK}\,.$$

Since computation time is strongly dependent on the criterion used for deciding when to terminate an algorithm, we have specified the following termination criterion in an attempt to standardize results. Our criterion for primal-based algorithms is:

$$\text{(B-1)} \qquad \left| \frac{g_0(x^i) - g_0(x^{i-1})}{g_0(x^{i-1})} \right| \leqslant \mathrm{EPSCGP}\,,$$

[1] The code used to solve these problems has as its core a linear programming algorithm.

where $g_0(x^i)$ is the objective function value of the primal problem at the i^{th} iteration and $|g|$ denotes the absolute value of g. For dual-based algorithms, the tolerance EPSCGP should be used as follows:

$$\text{(B-2)} \qquad \left|\frac{g_0(x^i) - v(\delta^i)}{v(\delta^i)}\right| \leqslant \text{EPSCGP} .$$

Here $v(\delta^i)$ is the objective function value of the dual problem at the i^{th} iteration.

A solution $\hat{x}$ (or $\hat{\delta}$) is considered optimal when *both* the above criterian (A) and (B) are satisfied

Computation times

The solutions given here were all computed on an IBM 370/158 computer, using the code GGP written by the author and based on the algorithm described in [6,8]. In all cases *double precision* arithmetic was used and the tolerance EPSLP set to 10^{-11}. The computation times using GGP with 2 different sets of tolerances are given in Table 0.1. For comparison purposes a 'standardized' time is also quoted. The standardized time is the ratio of the CPU time in seconds required for

Table 0.1
Computation times using GGP

Problem number	EPSCON, EPSCGP as specified in problem		EPSCON = 0.001, EPSCGP = 0.001	
	CPU seconds	Standardized time	CPU seconds	Standardized time
1A	6.950	0.2747	3.190	0.1261
1B	6.860	0.2711	2.870	0.1134
2	0.060	0.0024	0.063	0.0025
3	2.097	0.0829	0.993	0.0392
4A	7.099	0.2806	1.973	0.0780
4B	3.350	0.1324	1.973	0.0780
4C	0.540	0.0213	0.337	0.0133
5	3.176	0.1255	0.967	0.0382
6	8.286	0.3275	3.963	0.1566
7	6.080	0.2403	11.209	0.4430
8A	2.413	0.0954	1.399	0.0553
8B	2.416	0.0955	1.399	0.0553
8C	2.003	0.0792	1.193	0.0472

the solution of the problem, to the time taken to execute Colville's standard timing program [4]. For the above computer the execution of Colville's timing program required 25.30 seconds of CPU time.

The problems

The problems documented here have been carefully selected from a large number of problems solved by the author over a number of years of computational trials with the code GGP [6,8].

Every effort has been made to compile as interesting a set of problems as possible from the computational point of view, without resorting to artificially constructed problems. All the problems (with the exception of Problem 8) are mathematical programming models of 'real-world' processes. They are for the main part taken from the field of optimal engineering design.

Problem 1. *Multiphase chemical equilibrium calculation*

This problem is the geometric programming dual of the Gibbs free energy minimization model formulated by Dantzig, De Haven and Sams [5].

Chemical equilibrium problems form a very important class of dual geometric programming problems. Apart from their practical importance, they are also very interesting numerically because of the scaling difficulties they present.

The optimal dual variables and hence the optimal values of the corresponding primal terms usually differ by many orders or magnitude resulting in the necessity for very accurate solution techniques. In actual fact in the particular application solved here, no more than estimates of orders of magnitude can be obtained for some of the primal terms and their corresponding dual variables. This is because some of the primal terms are as small as 10^{-22} at the optimum and in order for these terms to be significant we would require a constraint tolerance of 10^{-22} or smaller, something which is computationally infeasible at present.

Two versions of this problem are included. The first (Problem 1A) is an unscaled version and the second (Problem 1B) has been scaled so that all the variables lie between 1 and 10 at the optimal solution. In both cases, because of the extremely large value of the objective function at the optimal solution, the objective function given here is actually the original objective function to the one tenth power.

Statistics

NK	$= 3$,	DD	$= 18$,
NVAR	$= 12$,	EPSCON	$= 10^{-6}$,
NTERMS	$= 31$,	EPSCGP	$= 10^{-4}$.

Objective function and constraints

$$g_0(x) = c_1 x_1^{a_{11}} x_2^{a_{21}} x_3^{a_{31}} x_4^{a_{41}} x_5^{a_{51}} x_6^{a_{61}} x_7^{a_{71}} x_8^{a_{81}} x_9^{a_{91}} x_{10}^{a_{10,1}} x_{11}^{a_{11,1}}$$

$$g_1(x) = c_2 x_1 + c_3 x_2 + c_4 x_3 + c_5 x_4 x_5 ,$$

$$g_2(x) = c_6 x_1 + c_7 x_2 + c_8 x_3 + c_9 x_4 x_{12} + c_{10} x_5 x_{12}^{-1} + c_{11} x_6 x_{12}^{-1} + c_{12} x_7 x_{12} + c_{13} x_4 x_5 + c_{14} x_2 x_5 x_{12}^{-1} + c_{15} x_2 x_4 x_5 + c_{16} x_2 x_4^{-1} x_5 x_{12}^{-2} + c_{17} x_{10} x_{12}^{-1} ,$$

$$g_3(x) = c_{18} x_1 + c_{19} x_2 + c_{20} x_3 + c_{21} x_4 + c_{22} x_5 + c_{23} x_6 + c_{24} x_8 + c_{25} x_4 x_5 + c_{26} x_2 x_5 + c_{27} x_2 x_4 x_5 + c_{28} x_2 x_4^{-1} x_5 + c_{29} x_9 + c_{30} x_1 x_9 + c_{31} x_{11} .$$

The objective function exponents $a_{i,1}$ ($i = 1, ..., 12$) and the term coefficients c_j ($j = 1, ..., 31$) for Problems 1A and 1B, are given in Tables 1.1 and 1.2 respectively. An initial *infeasible* starting point, upper and lower bounds on the variables and the optimal solutions for Problems 1A and 1B, are given in Table 1.3 and Table 1.4 respectively. All of the variable bounds shown in Tables 1.3 and 1.4 are artificially imposed.

Table 1.1
Objective function exponents, Problems 1A and 1B

i	$a_{i,1}$	i	$a_{i,1}$
1	−0.001331720	7	−0.008092
2	−0.002270927	8	−0.005
3	−0.002485460	9	−0.000909
4	−4.67	10	−0.00088
5	−4.671973	11	−0.00119
6	−0.008140		

Table 1.2
Term coefficients; Problems 1A and 1B

Term number j	Coefficient c_j Problem 1A	Problem 1B
1	1.0000000 E-70	1.00000000 E+5
2	5.3637300 E 4	5.36737300 E-2
3	2.1863746 E 3	2.1863746 E-2
4	9.7733533 E 4	9.7733533 E-2
5	6.6940803 E 15	6.6940803 E-3
6	1.0	1.0 E-6
7	1.0	1.0 E-5
8	1.0	1.0 E-6
9	1.0	1.0 E-10
10	1.0	1.0 E-8
11	1.0	1.0 E-2
12	1.0	1.0 E-4
13	1.0898645 E 17	1.0898645 E-1
14	1.6108052 E 9	1.6108052 E-4
15	1.0	1.0 E-23
16	1.9304541 E-3	1.9304541 E-6
17	1.0	1.0 E-3
18	1.0	1.0 E-6
19	1.0	1.0 E-5
20	1.0	1.0 E-6
21	1.0	1.0 E-9
22	1.0	1.0 E-9
23	1.0	1.0 E-3
24	1.0	1.0 E-3
25	1.0898645 E 17	1.0898645 E-1
26	1.6108052 E 9	1.6108052 E-5
27	1.0	1.0 E-23
28	1.9304541 E-3	1.9304541 E-8
29	1.0	1.0 E-5
30	1.1184059 E 7	1.1184059 E-4
31	1.0	1.0 E-4

Problem 2. Colville's test problem #3

This problem is taken from Colville's nonlinear programming study [4]. The problem has been included in this selection of problems because of the apparent ease with which it was solved, despite the fact that it has a relatively high degree of difficulty and contains both negative and positive terms. In fact, if the standardized times for this problem (see Table 0.1) are compared with those obtained by Colville in his study [4], it is seen that GGP solves this problem approximately 3 times faster than any of the codes in [4].

Table 1.3
Starting values, variable bounds and optimal solution for Problem 1A

Variable	Starting value	Upper bound	Lower bound	Optimal solution
$g_0(x)$	1.E11	1.E11	1.E9	4.8904620 E+9
x_1	4.E-6	1.E-4	1.E-7	2.5229712 E-6
x_2	4.E-5	1.E-3	1.E-6	2.5288262 E-5
x_3	4.E-6	1.E-4	1.E-7	7.6566135 E-6
x_4	4.E-9	1.E-8	1.E-11	1.1853913 E-9
x_5	4.E-9	1.E-6	1.E-9	7.6971796 E-9
x_6	4.E-3	1.E-1	1.E-4	1.2919459 E-3
x_7	4.E-3	1.E-1	1.E-4	4.2615451 E-3
x_8	4.E-3	1.E-1	1.E-4	2.7863431 E-3
x_9	4.E-5	1.E-3	1.E-6	1.7093851 E-5
x_{10}	4.E-4	1.E-2	1.E-5	2.0389545 E-4
x_{11}	4.E-4	1.E-2	1.E-5	6.6285938 E-4
x_{12}	4.0	1.E+2	1.E-1	6.5581875 E-1

Table 1.4
Starting values, variable bounds and optimal solution for Problem 1B

Variable	Starting value	Upper bound	Lower bound	Optimal solution
$g_0(x)$	10.0	10.0	0.1	3.1682133
x_1	4.0	100.0	0.1	2.5179680
x_2	4.0	100.0	0.1	2.5391937
x_3	4.0	100.0	0.1	7.6570348
x_4	4.0	100.0	0.1	1.2219265
x_5	4.0	100.0	0.1	7.4670724
x_6	4.0	100.0	0.1	1.2916003
x_7	4.0	100.0	0.1	4.2830877
x_8	4.0	100.0	0.1	2.7816976
x_9	4.0	100.0	0.1	1.7870736
x_{10}	4.0	100.0	0.1	2.0016703
x_{11}	4.0	100.0	0.1	6.4961066
x_{12}	4.0	100.0	0.1	6.4496897

Statistics

$$\begin{array}{llll} \text{NK} & = 6\,, & \text{DD} & = 26\,, \\ \text{NVAR} & = 5\,, & \text{EPSCON} & = 10^{-5}\,, \\ \text{NTERMS} & = 32\,, & \text{EPSCGP} & = 10^{-4}\,. \end{array}$$

Objective function and constraints

$$g_0(x) = c_1 x_3^2 + c_2 x_1 x_5 + c_3 x_1 + c_4\,,$$

$$g_1(x) = c_5 x_3 x_5 + c_6 x_2 x_5 + c_7 x_1 x_4 ,$$

$$g_2(x) = c_8 x_2 x_5 + c_9 x_1 x_4 + c_{10} x_3 x_5 ,$$

$$g_3(x) = c_{11} x_2^{-1} x_5^{-1} + c_{12} x_1 x_5^{-1} + c_{13} x_2^{-1} x_3^2 x_5^{-1} ,$$

$$g_4(x) = c_{14} x_2 x_5 + c_{15} x_1 x_2 + c_{16} x_3^2 ,$$

$$g_5(x) = c_{17} x_3^{-1} x_5^{-1} + c_{18} x_1 x_5^{-1} + c_{19} x_4 x_5^{-1} ,$$

$$g_6(x) = c_{20} x_3 x_5 + c_{21} x_1 x_3 + c_{22} x_3 x_4 .$$

The coefficients $c_1, ..., c_{22}$ are given in Table 2.1. A feasible starting point, upper and lower bounds on the problem variables and an optimal solution are given in Table 2.2.

The upper and lower bounds on the variables $x_1, ..., x_5$ are all part of the original mathematical programming model given in [4].

Table 2.1
Coefficients for Problem 2

j	c_j	j	c_j	j	c_j
1	5.35785470	9	0.00009395	17	2275.132693
2	0.83568910	10	−0.00033085	18	−0.26680980
3	37.239239	11	1330.32937	19	−0.40583930
4	−40792.1410	12	−0.42002610	20	0.00029955
5	0.00002584	13	−0.30585975	21	0.00007992
6	−0.00006663	14	0.00024186	22	0.00012157
7	−0.00000734	15	0.00010159		
8	0.000853007	16	0.00007379		

Table 2.2
Starting point, Variable bounds and an optimal solution for Problem 2

Variable	Initial value (F)	Upper bound	Lower bound	Optimal solution
$g_0(x) - c_4$	15000.0	20000.0	1000.0	10126.64252
x_1	78.62	102.0	78.0	78.0
x_2	33.44	45.0	33.0	33.0
x_3	31.07	45.0	27.0	29.99551065
x_4	44.18	45.0	27.0	45.0
x_5	35.22	45.0	27.0	36.77517397

Problem 3. Alkylation process optimization

This problem has been taken from Bracken and McCormick [3]. The original version of this problem contained some equality constraints resulting from mass balance considerations. These have been eliminated and the problem has been reformulated as a signomial optimization problem.

Statistics

NK	$= 14$,	DD	$= 50$,
NVAR	$= 7$,	EPSCON	$= 10^{-5}$,
NTERMS	$= 58$,	EPSCGP	$= 10^{-4}$,

Objective function and constraints

$$g_0(x) = c_1 x_1 + c_2 x_1 x_6 + c_3 x_3 + c_4 x_2 + c_5 + c_6 x_3 x_5 \,,$$

$$g_1(x) = c_7 x_6^2 + c_8 x_1^{-1} x_3 + c_9 x_6 \,,$$

$$g_2(x) = c_{10} x_1 x_3^{-1} + c_{11} x_1 x_3^{-1} x_6 + c_{12} x_1 x_3^{-1} x_6^2 \,,$$

$$g_3(x) = c_{13} x_6^2 + c_{14} x_5 + c_{15} x_4 + c_{16} x_6 \,,$$

$$g_4(x) = c_{17} x_5^{-1} + c_{18} x_5^{-1} x_6 + c_{19} x_4 x_5^{-1} + c_{20} x_5^{-1} x_6^2 \,,$$

$$g_5(x) = c_{21} x_7 + c_{22} x_2 x_3^{-1} x_4^{-1} + c_{23} x_2 x_3^{-1} \,,$$

$$g_6(x) = c_{24} x_7^{-1} + c_{25} x_2 x_3^{-1} x_7^{-1} + c_{26} x_2 x_3^{-1} x_4^{-1} x_7^{-1} \,,$$

$$g_7(x) = c_{27} x_5^{-1} + c_{28} x_5^{-1} x_7 \,,$$

$$g_8(x) = c_{29} x_5 + c_{30} x_7 \,,$$

$$g_9(x) = c_{31} x_3 + c_{32} x_1 \,,$$

$$g_{10}(x) = c_{33} x_1 x_3^{-1} + c_{34} x_3^{-1} \,,$$

$$g_{11}(x) = c_{35} x_2 x_3^{-1} x_4^{-1} + c_{36} x_2 x_3^{-1} \,,$$

$$g_{12}(x) = c_{37} x_4 + c_{38} x_2^{-1} x_3 x_4 \,,$$

$$g_{13}(x) = c_{39}x_1x_6 + c_{40}x_1 + c_{41}x_3 ,$$

$$g_{14}(x) = c_{42}x_1^{-1}x_3 + c_{43}x_1^{-1} + c_{44}x_6 .$$

The problem coefficients $c_1, ..., c_{44}$ are given in Table 3.1. A feasible starting point, upper and lower bounds on the variables and an optimal solution are given in Table 3.2. *All* the bounding constraints on the variables $x_1, ..., x_7$ are part of the model describing the alkylation process.

Table 3.1
Coefficients for Problem 3

j	c_j	j	c_j	j	c_j
1	1.715	16	−0.19120592 E−1	31	0.00061000
2	0.035	17	0.56850750 E+2	32	−0.0005
3	4.0565	18	1.08702000	33	0.81967200
4	10.0	19	0.32175000	34	0.81967200
5	3000.0	20	−0.03762000	35	24500.0
6	−0.063	21	0.00619800	36	−250.0
7	0.59553571 E−2	22	0.24623121 E+4	37	0.10204082 E−1
8	0.88392857	23	−0.25125634 E+2	38	0.12244898 E−4
9	−0.11756250	24	0.16118996 E+3	39	0.00006250
10	1.10880000	25	5000.0	40	0.00006250
11	0.13035330	26	−0.48951000 E+6	41	−0.00007625
12	−0.00660330	27	0.44333333 E+2	42	1.22
13	0.66173269 E−3	28	0.33000000	43	1.0
14	0.17239878 E−1	29	0.02255600	44	−1.0
15	−0.56595559 E−2	30	−0.00759500		

Table 3.2
Starting point, variable bounds and an optimal solution for Problem 3

Variable	Initial value	Upper bound	Lower bound	Optimal solution
$g_0(x)$	2000.0	2000.0	1000.0	1227.1831610
x_1	1745.0	2000.0	1.0	1698.5276698
x_2	110.0	120.0	1.0	53.5257212
x_3	3048.0	5000.0	1.0	3031.5798057
x_4	89.0	93.0	85.0	90.0909228
x_5	92.0	95.0	90.0	95.0
x_6	8.0	12.0	3.0	10.5192394
x_7	145.0	162.0	145.0	153.5353546

Problem 4. *Optimal reactor design*

This problem is taken from Rijckaert [10]. Two formulations of the problem are given. The first (Problems 4A and 4B) has a signomial objective function and posynomial constraints and is Rijckaert's original formulation [10]. The second formulation (Problem 4C) is a posynomial approximation of Problem 4A.

Despite the fact that this problem is 'smaller' than the previous one, it appears to require more time to reach an optimal solution (see Table 0.1). This could be due to the fact that for this problem we start at an infeasible point whereas for Problem 3 we use a feasible starting point.

Another interesting aspect to the problem is the large difference in computation times between the posynomial version (Problem 4C) and signomial versions (Problems 4A and 4B) of the problem. This is unexpected since Problems 4A and 4B are 'almost' prototype geometric programs. That is, they have only 2 negative out of a total of 16 terms. One explanation for this is the slow convergence of the iteration procedure of Avriel and Williams [1]. upon which GGP is based.

Problem 4B is included to show that GGP requires *one half* the computation time to come within 99.9% the value of the optimal solution computed in Problem 4A.

Problem 4A

Statistics

NK	= 4 ,	DD	= 7 ,
NVAR	= 8 ,	EPSCON	$= 10^{-5}$,
NTERMS	= 16 ,	EPSCGP	$= 10^{-4}$,

Objective function and constraints

$$g_0(x) = 0.4x_1^{0.67}x_7^{-0.67} + 0.4x_2^{0.67}x_8^{-0.67} + 10.0 - x_1 - x_2 ,$$

$$g_1(x) = 0.0588x_5x_7 + 0.1x_1 ,$$

$$g_2(x) = 0.0588x_6x_8 + 0.1x_1 + 0.1x_2 ,$$

$$g_3(x) = 4x_3x_5^{-1} + 2x_3^{-0.71}x_5^{-1} + 0.0588x_3^{-1.3}x_7 ,$$

$$g_4(x) = 4x_4x_6^{-1} + 2x_4^{-0.71}x_6^{-1} + 0.0588x_4^{-1.3}x_8 .$$

The variable bounds (all artificially imposed) are given by:

$$0.1 \leqslant x_i \leqslant 10, \qquad i = 1, ..., 8\ ,$$

and the objective function $g_0(x)$, is bounded (artificially) by

$$1.0 \leqslant g_0(x) \leqslant 4.2\ .$$

Problem 4B

Identical to 4A except for EPSCGP = 10^{-3}.

Problem 4C

Statistics

NK	= 5 ,	DD	= 5 ,
NVAR	= 9 ,	EPSCON	= 10^{-5} ,
NTERMS	= 15 ,	EPSCGP	= 10^{-4} .

Objective function and constraints

$$g_0(x) = x_0\ ,$$

$g_1(x), g_2(x), g_3(x), g_4(x)$ as in Problem 4A,

$$g_5(x) = 0.144x_0^{-0.313}x_1^{0.168}x_2^{-0.185}x_7^{-0.670} + 0.144x_0^{-0.313}$$

$$x_1^{-0.502}x_2^{0.485}x_8^{-0.670} + 3.6x_0^{-0.313}x_1^{-0.502}x_2^{-0.185}\ .$$

An infeasible starting point and optimal solutions for Problems 4A, 5B and 4C computed using this point are given in Table 4.1.

Table 4.1
Starting point and optimal solutions for Problems 4A, 4B and 4C

Variable	Starting point (infeasible)	Problem 8A	Optimal solution Problem 4B	Problem 4C
$g_0(x) = x_0$	4.2	3.9516982	3.9561968	3.9520666
x_1	6.0	6.3450905	6.1016207	6.3658386
x_2	3.0	2.3427973	2.5741491	2.3317880
x_3	0.4	0.6701581	0.6765178	0.6725790
x_4	0.2	0.5966619	0.5959532	0.5935613
x_5	6.0	5.9528907	5.9935248	5.9494125
x_6	6.0	5.5291597	5.5315385	5.5272775
x_7	1.0	1.0441714	1.1061818	1.0388592
x_8	0.5	0.4036023	0.4071500	0.4007536

Problem 5. Heat exchanger design

This problem was first formulated as a generalized geometric programming problem by Avriel and Williams [1].

Statistics

$$\begin{aligned} &\text{NK} = 6\,, && \text{DD} = 10\,,\\ &\text{NVAR} = 8\,, && \text{EPSCON} = 10^{-5}\,,\\ &\text{NTERMS} = 19\,, && \text{EPSCGP} = 10^{-4}\,. \end{aligned}$$

Objective function and constraints

$$g_0(x) = c_1 x_1 + c_2 x_2 + c_3 x_3\,,$$

$$g_1(x) = c_4 x_1^{-1} x_4 x_6^{-1} + c_5 x_6^{-1} + c_6 x_1^{-1} x_6^{-1}\,,$$

$$g_2(x) = c_7 x_2^{-1} x_5 x_7^{-1} + c_8 x_4 x_7^{-1} + c_9 x_2^{-1} x_4 x_7^{-1}\,,$$

$$g_3(x) = c_{10} x_3^{-1} x_8^{-1} + c_{11} x_5 x_8^{-1} + c_{12} x_3^{-1} x_5 x_8^{-1}\,,$$

$$g_4(x) = c_{13} x_4 + c_{14} x_6\,,$$

$$g_5(x) = c_{15} x_5 + c_{16} x_7 + c_{17} x_4\,,$$

$$g_6(x) = c_{18} x_8 + c_{19} x_5\,.$$

The problem coefficients c_1, ..., c_{19} are given in Table 5.1.

An infeasible starting point, upper and lower bounds on the problem

Table 5.1
Coefficients for test problem 5

j	c_j	j	c_j
1	1.0	11	1.0
2	1.0	12	−2500.0
3	1.0	13	0.0025
4	833.33252	14	0.0025
5	100.0	15	0.0025
6	−83333.333	16	0.0025
7	1250.0	17	−0.0025
8	1.0	18	0.01
9	−1250.0	19	−0.01
10	1250000.0		

Table 5.2
Starting point, variable bounds and an optimal solution for Problem 5

Variable	Initial value (infeasible)	Upper bound	Lower bound	Optimal solution
$g_0(x)$	15000.0	20000.0	3000.0	7049.324305
x_1	5000.0	10000.0	100.0	572.852745
x_2	5000.0	10000.0	1000.0	1361.497867
x_3	5000.0	10000.0	1000.0	5114.973425
x_4	200.0	1000.0	10.0	181.476181
x_5	350.0	1000.0	10.0	295.402778
x_6	150.0	1000.0	10.0	218.524281
x_7	225.0	1000.0	10.0	286.075426
x_8	425.0	1000.0	10.0	395.402602

variables (all of which are artificially imposed) and an optimal solution to the problem are given in Table 5.2.

Problem 6. A 3-stage membrane separation process

This problem is a mathematical programming model of a 3-stage membrane separation process. The derivation of the model may be found in Dembo [6].

This model is representative of a large number of chemical flow processes with many recycle streams. One of the difficulties of this sort of model is to simply compute a feasible solution.

Statistics

$$\begin{aligned} &\text{NK} = 13\,, && \text{DD} = 39\,,\\ &\text{NVAR} = 13\,, && \text{EPSCON} = 10^{-6}\,,\\ &\text{NTERMS} = 53\,, && \text{EPSCGP} = 10^{-4}\,. \end{aligned}$$

Objective function and constraints

$$g_0(x) = c_1 x_{11} + c_2 x_{12} + c_3 x_{13}\,,$$

$$g_1(x) = c_4 x_8 x_{11}^{-1} + c_5 x_1 x_8 x_{11}^{-1}\,,$$

$$g_2(x) = c_6 x_9 x_{12}^{-1} + c_7 x_2 x_9 x_{12}^{-1}\,,$$

$$g_3(x) = c_8 x_{10} x_{13}^{-1} + c_9 x_3 x_{10} x_{13}^{-1}\,,$$

$$g_4(x) = c_{10} x_2 x_5^{-1} + c_{11} x_2 + c_{12} x_2^2 x_5^{-1}\,,$$

$$g_5(x) = c_{13}x_3x_6^{-1} + c_{14}x_3 + c_{15}x_3^2x_6^{-1} \, ,$$

$$g_6(x) = c_{16}x_1x_5^{-1}x_7^{-1}x_8 + c_{17}x_4x_5^{-1} + c_{18}x_4x_5^{-1}x_7^{-1}x_8 \, ,$$

$$g_7(x) = c_{19}x_2x_9 + c_{20}x_5x_8 + c_{21}x_6 + c_{22}x_5 + c_{23}x_1x_8 + c_{24}x_6x_9 \; ;$$

$$g_8(x) = c_{25}x_2^{-1}x_3x_9^{-1}x_{10} + c_{26}x_2^{-1}x_6 + c_{27}x_9^{-1} + c_{28}x_9^{-1}x_{10} + c_{29}x_2^{-1}x_6x_9^{-1} \, ,$$

$$g_9(x) = c_{30}x_2^{-1} + c_{31}x_{10} + c_{32}x_2^{-1}x_3x_{10} \, ,$$

$$g_{10}(x) = c_{33}x_2x_3^{-1} \, ,$$

$$g_{11}(x) = c_{34}x_1x_2^{-1} \, ,$$

$$g_{12}(x) = c_{35}x_7 + c_{36}x_8 \, ,$$

$$g_{13}(x) = c_{37}x_1x_4^{-1} + c_{38}x_1 + c_{39}x_1^2x_4^{-1} \, .$$

The coefficients $c_1, ..., c_{36}$ are given in Table 6.1.

An infeasible starting point, upper and lower bounds on the problem variables and an optimal solution are given in Table 6.2.

The following bounds are derived from the mathematical model of the 3-stage membrane separation process [6].

Table 6.1
Coefficients for Problem 6

j	c_j	j	c_j	j	c_j
1	1.0	14	0.975000	27	500.0
2	1.0	15	−0.009750	28	−1.0
3	1.0	16	1.0	29	−500.0
4	1.262626	17	1.0	30	0.9
5	−1.231059	18	−1.0	31	0.002
6	1.262626	19	0.002	32	−0.002
7	−1.231059	20	0.002	33	1.0
8	1.262626	21	1.0	34	1.0
9	−1.231059	22	1.0	35	0.002
10	0.034750	23	−0.002	36	−0.002
11	0.975000	24	−0.002	37	0.034750
12	−0.009750	25	1.0	38	0.975000
13	0.034750	26	1.0	39	−0.009750

Table 6.2
Starting point, variable bounds and an optimal solution for Problem 6

Variable	Starting point (infeasible)	Upper bound	Lower bound	Optimal solution
$g_0(x)$	250.0	250.0	50.0	97.591034
x_1	0.50	1.0	0.1	0.803773
x_2	0.80	1.0	0.1	0.900000
x_3	0.90	1.0	0.9	0.900000
x_4	0.10	0.1	0.0001	0.100000
x_5	0.14	0.9	0.1	0.190837
x_6	0.50	0.9	0.1	0.900000
x_7	489.0	1000.0	0.1	574.099615
x_8	80.0	1000.0	0.1	74.099636
x_9	650.0	1000.0	500.0	500.000000
x_{10}	450.0	500.0	0.1	0.100000
x_{11}	150.0	150.0	1.0	20.239117
x_{12}	150.0	150.0	0.0001	77.336450
x_{13}	150.0	150.0	0.0001	0.015467

Upper bounds on variables $x_1, x_2, x_3, x_4, x_5, x_6, x_{10}$.
Lower bounds on variables $x_1, x_2, x_3, x_5, x_6, x_9, x_{10}$.

Problem 7. A 5-stage membrane separation process

This problem is essentially an extension of the previous one and is also developed in detail in Dembo [6].

Statistics

$$\begin{array}{llll} \text{NK} & = 19\,, & \text{DD} & = 68\,, \\ \text{NVAR} & = 16\,, & \text{EPSCON} & = 10^{-5}\,, \\ \text{NTERMS} & = 85\,, & \text{EPSCGP} & = 10^{-3}\,. \end{array}$$

Objective function and constraints

$$g_0(x) = c_1 x_{12} + c_2 x_{13} + c_3 x_{14} + c_4 x_{15} + c_5 x_{16} + c_6 x_1 x_{12}$$

$$+ c_7 x_2 x_{13} + c_8 x_3 x_{14} + c_9 x_4 x_{15} + c_{10} x_5 x_{16}\,,$$

$$g_1(x) = c_{11} x_1 x_6^{-1} + c_{12} x_1 + c_{13} x_1^2 x_6^{-1}\,,$$

$$g_2(x) = c_{14} x_2 x_7^{-1} + c_{15} x_2 + c_{16} x_2^2 x_7^{-1}\,,$$

$$g_3(x) = c_{17} x_3 x_8^{-1} + c_{18} x_3 + c_{19} x_3^2 x_8^{-1}\,,$$

$$g_4(x) = c_{20} x_4 x_9^{-1} + c_{21} x_4 + c_{22} x_4^2 x_9^{-1} ,$$

$$g_5(x) = c_{23} x_5 x_{10}^{-1} + c_{24} x_5 + c_{25} x_5^2 x_{10}^{-1} ,$$

$$g_6(x) = c_{26} x_6 x_7^{-1} + c_{27} x_1 x_7^{-1} x_{11}^{-1} x_{12} + c_{28} x_6 x_7^{-1} x_{11}^{-1} x_{12} ,$$

$$g_7(x) = c_{29} x_7 x_8^{-1} + c_{30} x_7 x_8^{-1} x_{12} + c_{31} x_2 x_8^{-1} x_{13}$$
$$+ c_{32} x_{13} + c_{33} x_1 x_8^{-1} x_{12} ,$$

$$g_8(x) = c_{34} x_8 + c_{35} x_8 x_{13} + c_{36} x_3 x_{14} + c_{37} x_9 + c_{38} x_2 x_{13}$$
$$+ c_{39} x_9 x_{14} ,$$

$$g_9(x) = c_{40} x_3^{-1} x_9 + c_{41} x_3^{-1} x_4 x_{14}^{-1} x_{15} + c_{42} x_3^{-1} x_{10} x_{14}^{-1}$$
$$+ c_{43} x_3^{-1} x_9 x_{14}^{-1} + c_{44} x_3^{-1} x_8 x_{14}^{-1} x_{15} ,$$

$$g_{10}(x) = c_{45} x_4^{-1} x_5 x_{15}^{-1} x_{16} + c_{46} x_4^{-1} x_{10} + c_{47} x_{15}^{-1}$$
$$+ c_{48} x_{15}^{-1} x_{16} + c_{49} x_4^{-1} x_{10} x_{15}^{-1} ,$$

$$g_{11}(x) = c_{50} x_4^{-1} + c_{51} x_{16} + c_{52} x_4^{-1} x_5 x_{16} ,$$

$$g_{12}(x) = c_{53} x_{11} + c_{54} x_{12} ,$$

$$g_{13}(x) = c_{55} x_{11}^{-1} x_{12} ,$$

$$g_{14}(x) = c_{56} x_4 x_5^{-1} ,$$

$$g_{15}(x) = c_{57} x_3 x_4^{-1} ,$$

$$g_{16}(x) = c_{58} x_2 x_3^{-1} ,$$

$$g_{17}(x) = c_{59} x_1 x_2^{-1} ,$$

$$g_{18}(x) = c_{60} x_9 x_{10}^{-1} ,$$

$$g_{19}(x) = c_{61} x_8 x_9^{-1} .$$

Table 7.1
Coefficients for Problem 7

j	c_j	j	c_j	j	c_j
1	1.262626	22	−0.00975	43	−500.0
2	1.262626	23	0.03475	44	−1.0
3	1.262626	24	0.975	45	1.0
4	1.262626	25	−0.00975	46	1.0
5	1.262626	26	1.0	47	500.0
6	−1.231060	27	1.0	48	−1.0
7	−1.231060	28	−1.0	49	−500.0
8	−1.231060	29	1.0	50	0.9
9	−1.231060	30	0.002	51	0.002
10	−1.231060	31	0.002	52	−0.002
11	0.034750	32	−0.002	53	0.002
12	0.975	33	−0.002	54	−0.002
13	−0.00975	34	1.0	55	1.0
14	0.034750	35	0.002	56	1.0
15	0.975	36	0.002	57	1.0
16	−0.00975	37	1.0	58	1.0
17	0.03475	38	−0.002	59	1.0
18	0.975	39	−0.002	60	1.0
19	−0.00975	40	1.0	61	1.0
20	0.03475	41	1.0		
21	0.975	42	500.0		

Table 7.2
Starting point, variable bounds and an optimal solution for Problem 7

Variable	Starting point (infeasible)	Upper bound	Lower bound	Optimal solution
$g_0(x)$	250.0	250.0	50.0	174.788807
x_1	0.8	0.9	0.1	0.8037724
x_2	0.83	0.9	0.1	0.8175130
x_3	0.85	0.9	0.1	0.9
x_4	0.87	0.9	0.1	0.9
x_5	0.90	1.0	0.9	0.9
x_6	0.10	0.1	0.0001	0.0999996
x_7	0.12	0.9	0.1	0.1078842
x_8	0.19	0.9	0.1	0.1908369
x_9	0.25	0.9	0.1	0.1908369
x_{10}	0.29	0.9	0.1	0.1908369
x_{11}	512.0	1000.0	1.0	505.664787
x_{12}	13.1	500.0	0.000001	5.6650580
x_{13}	71.8	500.0	1.0	72.475185
x_{14}	640.0	1000.0	500.0	500.0
x_{15}	650.0	1000.0	500.0	500.0
x_{16}	5.7	500.0	0.000001	0.000001

The coefficients $c_1, ..., c_{61}$ are given in Table 7.1.

An infeasible starting point, upper and lower bounds on the problem variables and an optimal solution to the problem are given in Table 7.2.

The following bounds are derived from the mathematical model of the 5-stage membrane separation process [6].

Upper bounds on variables $x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8, x_9, x_{10}, x_{12}, x_{13}, x_{16}$.

Lower bounds on variables $x_1, x_2, x_3, x_4, x_5, x_7, x_8, x_9, x_{10}, x_{14}, x_{15}$.

Problem 8. *Beck and Ecker's problem*

This problem is a prototype geometric programming problem taken from Beck and Ecker [2]. It has been included here for a number of reasons. Firstly, it appears to be sensitive with respect to the exponent α in the first term of the objective function. Also, Beck and Ecker [2] report large variations in computation times for different α, whereas this phenomenon is not observed when the problem is solved using GGP.

Three versions of the problem have been included here, each identical except for the parameter α.

Statistics

NK	$= 4$,	DD	$= 10$,
NVAR	$= 7$,	EPSCON	$= 10^{-6}$,
NTERMS	$= 18$,	EPSCGP	$= 10^{-4}$.

Objective function and constraints

$$g_0(x) = c_1 x_1 x_2^{-1} x_4^2 x_6^{-3} x_7^{\alpha} + c_2 x_1^{-1} x_2^{-2} x_3^1 x_4^1 x_5^{-1} x_7^{-1/2}$$

$$+ c_3 x_1^{-2} x_2 x_4^{-1} x_5^{-2} x_6 + c_4 x_1^2 x_2^2 x_3^{-1} x_5^{1/2} x_6^{-2} x_7 ,$$

$$g_1(x) = c_5 x_1^{1/2} x_3^{-1} x_6^{-2} x_7 + c_6 x_1^3 x_2 x_3^{-2} x_6^1 x_7^{1/2}$$

$$+ c_7 x_2^{-1} x_3 x_4^{-1/2} x_6^{2/3} x_7^{1/4} ,$$

$$g_2(x) = c_8 x_1^{-1/2} x_2 x_3^{-1} x_5^{-1} x_6 + c_9 x_3 x_4^{-1} x_5^{-1} x_6^2$$

$$+ c_{10} x_1^{-1} x_2^{1/2} x_4^{-2} x_5^{-1} x_6^{1/3} ,$$

$$g_3(x) = c_{11}x_1x_3^{-3/2}x_5x_6^{-1}x_7^{1/3} + c_{12}x_2x_3^{-1/2}x_5x_6^{-1}x_7^{-1/2} + c_{13}x_1^{-1}x_2x_3^{1/2}x_5 + c_{14}x_2^{-2}x_3x_5x_6^{-1}x_7 \,,$$

$$g_4(x) = c_{15}x_1^{-2}x_2x_4^{-1}x_5^{1/2}x_7^{1/3} + c_{16}x_1^{1/2}x_2^2x_3x_4^{1/3}x_5^{-2/3}x_7^{1/4} + c_{17}x_1^{-3}x_2^{-2}x_3x_5x_7^{3/4} + c_{18}x_3^{-2}x_4x_7^{1/2} \,.$$

The variable bounds (all artificial) are given by

$$0.1 \leqslant x_i \leqslant 10 \,, \qquad i = 1, ..., 6 \,,$$

$$0.01 \leqslant x_7 \leqslant 10 \,,$$

and the objective function is bounded (artificially) by

$$100 \leqslant g_0(x) \leqslant 3000 \,.$$

The problem coefficients $c_1, ..., c_{18}$ are given in Table 8.1.

An *infeasible* starting point and the corresponding optimal solutions computed for $\alpha = -\frac{1}{4}$ (Problem 8A), $\alpha = \frac{1}{8}$ (Problem 8B), $\alpha = \frac{1}{2}$ (Problem 8C) are given in Table 8.2.

Table 8.1
Coefficients for Problem 8

j	c_j	j	c_j	j	c_j
1	10.0	7	0.2	13	1.0
2	15.0	8	1.3	14	0.65
3	20.0	9	0.8	15	0.2
4	25.0	10	3.1	16	0.3
5	0.5	11	2.0	17	0.4
6	0.7	12	0.1	18	0.5

Table 8.2
Starting point and optimal solutions for Problems 8A, 8B, 8C

Variable	Starting value	Optimal solution Problem 8A	Problem 8B	Problem 8C
$g_0(x)$	2500.0	1809.7615	911.87957	543.66638
x_1	6.0	2.8566276	3.8955214	4.3919085
x_2	6.0	0.61083257	0.80868472	0.8546317
x_3	6.0	2.1503944	2.6626285	2.8416293
x_4	6.0	4.7171337	4.2983005	3.4013674
x_5	6.0	1.0002048	0.85357785	0.72275344
x_6	6.0	1.3487370	1.0953123	0.87052969
x_7	6.0	0.03160686	0.02730898	0.02464651

Acknowledgments

I would like to acknowledge the invaluable aid of Professor Mordecai Avriel in assisting with the formulation of some of the problems in this paper. I would also like to thank the Department of Chemical Engineering at the Technion and the Department of Management Sciences at the University of Waterloo for providing the computer time for this study.

References

[1] M. Avriel and A.C. Williams, "An extension of geometric programming with applications in engineering optimization", *Journal of Engineering Mathematics* 5 (3) (1971) 187–194.

[2] P.A. Beck and J.G. Ecker, "A modified concave simplex algorithm for geometric programming", *Journal of Optimization Theory and Applications,* to appear.

[3] J. Bracken and G.P. McCormick, "Selected applications in nonlinear programming" (Wiley, New York, 1968) pp. 37–45.

[4] A.R. Colville, "A comparative study of nonlinear programming codes", IBM NYSC Rept. 320–2949 (1968).

[5] G.B. Dantzig, J.C. de Haven and C.P. Sams, in: *Proceedings of the 4th Berkeley symposium on mathematical statistics and probability* (University of California Press, Berkeley, Calif., 1960) pp. 181–196.

[6] R.S. Dembo, "Solution of complementary geometric programming problems", M.Sc. Thesis, Technion, Haifa (1972).

[7] R.S. Dembo, "GGP – A program for solving generalized geometric programs – Users Manual", Department of Chemical Engineering Rept. 72/59, Technion (1972).

[8] R.S. Dembo, M. Avriel and U. Passy, "An algorithm for the solution of generalized geometric programs", *International Journal for Numerical Methods in Engineering,* to appear.

[9] J. Haldi, "25 integer programming test problems", Stanford University, Graduate School of Business, Working Paper no. 43 (December 1964).

[10] M.J. Rijckaert, "Engineering applications of geometric programming", in: M. Avriel, M.J. Rijckaert and D.J. Wilde, eds., *Optimization and design* (Prentice Hall, Englewood Cliffs, N.J., 1973) pp. 196–220.