Journal of the Korean
OR/MS Society
Vol. 10, No. 1, Apr. 1985

—Research Paper—

Revised Iterative Goal Programming Using Sparsity Technique on Microcomputer*

Mitsuo Gen**
Kenichi Ida**
Sang M. Lee***

Abstract

Recently, multiple criteria decision making has been well established as a practical approach to seek a satisfactory solution to a decision making problem. Goal programming is one of the most powerful MCDM tools with satisfying operational assumptions that reflect the actual decision making process in real-world situations. In this paper we propose an efficient method implemented on a microcomputer for solving linear goal programming problems. It is an iterative revised goal simplex method using the sparsity technique. We design an interactive software package for microcomputers based on this method. From some computational experiences, we can state that the revised iterative goal simplex method using the sparsity technique is the most efficient one for microcomputers for solving goal programming problems.

1. Introduction

Recently, multiple criteria decision making(MCDM) or multiple objective decision making (MODM) has been fairly well established as a practical decision making approach with limited information, resources, and cognitive ability of the decision maker [12,13,24,25]. Goal programming(GP) is one of the most powerful MCDM tools with satisfying operational assumptions [5,14,15,18,21]. Recently, Lee, Gen, and Shim [20] presented a comprehensive state-of-the-art survey of GP which included a review of the most up-to-date references in the areas of GP solution methods, sensitivity and parametric analysis, integer GP, nonlinear GP, large-scale GP models, various other GP approaches, and a complete survey of GP applications. In their study, they systematically identified GP applications by subjects according to the *Subjective Classification for the OR/MS Index* and also concluded that GP is indeed the most widely applied MCDM method.

Kornbluth [17] first suggested a basic iterative approach of GP which solves a series of

* The research was done while the first author was a Visiting Associate Prof. at Univ. of Nebraska-Lincoln.

** Dept. of Management Engineering, Ashikaga Institute of Technology, Ashikaga, 326 Japan

*** Dept. of Management, University of Nebraska-Lincoln, Lincoln, NE 68588 U.S.A linear programming(LP) problems, from solving the highest priority problem or the initial solution to the solution of the lower priority problem. Since then, several related studies have been reported by Dauer and Krueger[7], Delft and Nijkamp [9], Ignizio and Perlis[15,16], Arthur and Ravindran[1,2], and Hwang and Masud[13]. Recently, Dauer and Krueger[8] applied their iterative approach to integer GP models for solving an assignment majorization problem. Also, Arthur and Ravindran[3] applied their iterative approach to solving an integer GP problem. Dauer[6] further developed a parametric method, based on the iterative approach, for solving linear GP problems.

Much attention has been focused on a microcomputer as an economical management tool to apply systematic approaches to decision making in Industrial Engineering problems[23]. Microcomputers are making important contributions to the operation of industrial, business, educational and nonprofit organizations. They will play an important role in management decision making processes.

In this paper we propose an efficient method that could be implemented on microcomputers for solving linear goal programming problems. The method developed here is a revised iterative goal simplex method using the sparsity technique. This method reduces the computational burden through the sparsity technique which makes a more efficient use of a microcomputer storage. We design an interactive software package for microcomputers based on this method. Then, we test some GP problems in varying structures and sizes on the microcomputer by the GP codes developed. From the view of computational experiences and storage capacity, we can state that the revised iterative goal simplex method using the sparsity technique is the most efficient method for microcomputers for solving goal programming problems.

2. The Goal Programming Model

In a summary of various techniques to traditional Industrial Engineering problems, Whitehouse, *et al.*[23] reported that 40% of the respondents expressed interest in linear programming. Also, in GP applications to real-world decision problems, a linear goal programming model is formulated in most cases. In a state-of-the-art survey of GP for decision making, Lee, Gen, and Shim[20] stated that about 89% of all GP applications used the linear goal programming models. We can also solve a linear programming problem by using our software package.

The general linear GP model with q priority levels, m_0 system constraints, m-m_0 goal constraints, and n decision variables can be expressed as follows[10]:

$$\text{min} \qquad z_0=\sum_{k=1}^{q}\sum_{i=m_0+1}^{m} p_k(w_{ki}^{-}d_i^{-}+w_{ki}^{+}d_i^{+}) \tag{1}$$

$$\text{s. t.} \qquad \sum_{j=1}^{n} a_{ij}\,x_j \qquad =b_i, \quad i=1,\cdots,\ m_0, \tag{2}$$

$$\sum_{j=1}^{n} a_{ij}x_j+d_i^{-}-d_i^{+}=b_i, \quad i=m_0+1,\cdots,\ m, \tag{3}$$

$$x_j,\ d_i^-,\ d_i^+ \geq 0,\ j=1,\cdots,\ n,\ i=1,\cdots,\ m, \tag{4}$$

where

p_k : the kth priority level, $p_k \gg Mp_{k+1}$ for all k(M is large positive number),

w_{ki}^-, w_{ki}^+ : numerical weights attached to deviational variables of the ith goal at a given priority level k,

d_i^-, d_i^+ : negative and positive deviational variables about the ith goal,

a_{ij} : technological coefficient of the jth activity for the ith goal or system constraints,

b_i : the ith goal or resource, and

x_j : the jth decision variable.

The general linear GP model can be represented in a matrix form as follows :

$$\begin{aligned} \text{min} \quad & z_0 = p^T(W^-d^- + W^+d^+) && (5)\\ \text{s. t.} \quad & A_0x = b_0, && (6)\\ & \overline{A}x + d^- - d^+ = \overline{b}, && (7)\\ & x,\ d^-,\ d^+ \geq 0, && (8) \end{aligned}$$

where $p=[p_k]_q$, $W^-=[w_{ki}^-]_{qx(m-m0)}$, $W^+=[w_{ki}^+]_{qx(m-m0)}$, $d^-=[d_i^-]_{m-m0}$, $d^+=[d_i^+]_{m-m0}$, $A_0=[a_{ij}]_{m0\times n}$, $b_0=[b_i]_{m0}$, $\overline{A}=[a_{m0+i,j}]_{(m-m0)\times n}$, $\overline{b}=[b_{m0+i}]_{m-m0}$, and $x=[x_j]_n$

Assuming that each priority level has m_k goals, the general linear GP model can be restated as follows :

$$\begin{aligned} \text{min} \quad & z_0 = \sum_{k=1}^{q} p_k z_k && (9)\\ \text{s. t.} \quad & A_0x = b_0, && (10)\\ & A_kx + d^- - d^+ = b_k,\ k=1,\cdots,\ q, && (11)\\ & x, d_k^-, d_k^+ \geq 0,\ k=1,\cdots,\ q, && (12) \end{aligned}$$

where

$z_k = w_{k.}^-d^- + w_{k.}^+d^+$ ($w_{k.}^-, w_{k.}^+$: row vectors of W^-, W^+, respectively),

$A_k=[a_{\tilde{m}_{k-1}+i,j}]_{m_k\times n}$, $b_k=[b_{\tilde{m}_{k-1}+i}]_{m_k}$, $d_k^-=[d_{\tilde{m}_{k-1}+i}]_{m_k}$, $d_k^+=[d_{\tilde{m}_{k-1}+i}]_{m_k}$,

and $\tilde{m}_k = \sum_{e=0}^{k} m_e$.

We denote the index set of deviational variables at the kth priority level by J_k.

3. The Revised Iterative Approach

In the iterative approach for solving GP problems, we have to first solve the following LP problem at the 1st priority level :

$$\begin{aligned} P_1 : \quad \text{min} \quad & z_1 = w_{1.}^-d^- + w_{1.}^+d^+ && (13)\\ \text{s. t.} \quad & A_0x = b_0, && (14)\\ & A_1x + d_1^- - d_1^+ = b_1, && (15)\\ & x, d_1^-,\ d_1^+ \geq 0. && (16) \end{aligned}$$

Assuming that the rank of the coefficient of constraints (14) and (15) in the problem P_1 is $\tilde{m}_1$, the basis B_1 can be formed by any set of $\tilde{m}_1$ independent vectors from $2m_1+n$ vectors consisting of the coefficient matrix. The remaining m_1+n-m_0 vectors form the nonbasis N_1, The weight vector $c_1=[0\ w_{1.}^{-} w_{1.}^{+}]$ and the variable vector $x'=[x\ d^-\ d^+]$ are divided into c_{11} and c_{12}, x_{11} and x_{12} respectively corresponding to matrices B_1 and N_1. Since the basis B_1 is nonsingular, the problem P_1 can be represented as follows :

$$P_1^* : \min\ z_1=z_1^*-r_{12}^T x_{12} \tag{17}$$

$$\text{s. t.}\qquad x_{11}+B_1^{-1}\ N_1 x_{12}=\beta_1, \tag{18}$$

$$x_{11},\ x_{12}\qquad \geq 0, \tag{19}$$

where

$$r_{12}^T=\pi_1^T N_1-c_{12}^T, \tag{20}$$

$$\pi_1^T=c_{11}^T\ B_1^{-1}. \tag{21}$$

We denote the index sets of basic and nonbasic vectors in the problem P_1^* by I_{11} and I_{12}, respectively, and denote the index set to be eliminated at the next priority by

$$L_1=\{j \mid r_{1j}<0,\ j\varepsilon I_{12}\}. \tag{22}$$

After solving the LP problem at the k-1 th priority level iteratively, the kth problem to be solved can be represented as follows :

$$\tilde{P}_k : \min\ z_k=w_{k.}^{-}d^-+w_{k.}^{+}d^+ \tag{23}$$

$$\text{s. t.}\qquad \tilde{x}_{k-1,1}+\tilde{B}_{k-1}^{-1}\ \tilde{N}_{k-1}\ \tilde{x}_{k-1,2}=\tilde{\beta}_{k-1}, \tag{24}$$

$$A_k x+d_k^- - d_k^+ \qquad =b_k, \tag{25}$$

$$\tilde{x}_{k-1,1},\ \tilde{x}_{k-1,2},\ d_k^-, d_k^+\geq 0, \tag{26}$$

where $\tilde{B}_{k-1}^{-1}$ and $\tilde{N}_{k-1}$ are basis inverse and nonbasis, respectively, up to the k-1 th problem. Also, $\tilde{x}_{k-1,1}$ and $\tilde{x}_{k-1,2}$ correspond to $\tilde{\beta}_{k-1}$ and $\tilde{N}_{k-1}$, respectively. Denoting the index sets of basic and nonbasic vectors up to the k-1th problem by $\tilde{I}_{k-1,1}$ and $\tilde{I}_{k-1,2}$, respectively, and the index set of variables to be eliminated at the kth problem by

$$L_{k-1}=\{\ j \mid r_{k-1,j}<0,\ j\ \varepsilon\ \tilde{I}_{k-1,2}\}, \tag{27}$$

the index sets of basic and nonbasic vectors at the kth problem are as follows :

$$\tilde{I}_{k1}=\tilde{I}_{k-1,1}+J_{k1} \tag{28}$$

$$\tilde{I}_{k2}=\tilde{I}_{k-1,2}-L_{k-1}+J_{k2}, \tag{29}$$

where J_{k1} and J_{k2} are the index sets at the kth problem to be added such that

$$J_k=J_{k1}\cup J_{k2},\quad J_{k1}\cap J_{k2}=\phi. \tag{30}$$

Since the k-1th basis $\tilde{B}_{k-1}$ is nonsingular, an inverse of the argumented basis

$$\tilde{\beta}_k=\begin{pmatrix} \tilde{\beta}_{k-1} & 0 \\ A_{k1} & I \end{pmatrix} \tag{31}$$

exists in which A_{k1} is a submatrix of the coefficient matrix corresponding to $I_{k-1,1}$ and the following factors at the kth problem can be recursively calculated

$$\tilde{B}_k^{-1}=\begin{pmatrix} \tilde{B}_{k-1}^{-1} & 0 \\ -A_{k1}\ \tilde{B}_{k-1}^{-1} & I \end{pmatrix} \tag{32}$$

$$\tilde{\beta}_k=\begin{pmatrix} \tilde{B}_{k-1} \\ \beta_k \end{pmatrix}\equiv\begin{pmatrix} \tilde{B}_{k-1} \\ b_k-A_{k1}\ \tilde{B}_{k-1} \end{pmatrix} \tag{33}$$

$$\pi_k^T=[-c_{k1}^T\ A_{k1}\ \tilde{B}_{k-1}^{-1}\quad c_{k1}^T] \tag{34}$$

$$z_k = c_{k1}{}^T \beta_k. \tag{35}$$

If any element of the basic solution β_k to be added to the $k-1$th problem is negative, we determine the most negative element among the negative elements. The nonbasic representation corresponding to the outgoing vector $\tilde{a}_{i_r}$ is calculated as follows :

$$\alpha_{r\cdot} = e_{r\cdot}\ \tilde{B}_k{}^{-1}\tilde{N}_k \tag{36}$$

where $e_{r\cdot}\tilde{B}_k{}^{-1}$ represents the rth row vector of the basis inverse $\tilde{B}_k{}^{-1}$. If all the elements of the nonbasic representation are zero or positive the original GP problem is unbounded. Otherwise, the reduced cost is calculated by

$$r_{kj} = \begin{cases} \pi_k{}^T\tilde{a}_j - c_{kj} & ; j\varepsilon J_{k2} \\ \pi_k{}^T\tilde{a}_j & ; j\varepsilon \tilde{I}_{k2} - J_{k2} \end{cases} \tag{37}$$

and an incoming vector to the basis is determined by the minimum ratio test.

$$\frac{r_{ks}}{\alpha_{rs}} = \min_j\left\{\frac{r_{kj}}{\alpha_{rj}}\,|\,\alpha_{rj}<0,\ j\varepsilon\tilde{I}_{k2}\right\}. \tag{38}$$

When there is no negative element in the basic feasible solution β_k, we update the basis by the primal revised simplex method.

4. Goal Simplex Method Using Sparsity Technique and Implementation.

The sparsity technique is used in order to save the memory capacity of the microcomputer. The revised iterative goal simplex method using the sparsity technique (GP-IRS/S) developed in this paper has an idea in solving the GP problem recursively and computing the reduced cost vector and goal function.

To compact the priority weight matrix into a small storage space, we introduce the sparsity technique in which the array is used as follows :

$v_w(\cdot)$: the numerical values of nonzero elements in the priority weight matrix,

$j_w(\cdot)$: the index number for column in the priority weight matrix,

$z_w(\cdot)$: the value of nonzero elements in each priority level.

The computational routines using the sparsity technique for the goal function z_k and the reduced cost vector $r_k.$ including π_k are referred as KGOAL and RCOST, respectively.

We summarize the revised iterative goal simplex method using the sparsity technique for solving linear GP problems as follows :

Step 1 : Solve the LP problem P_1 including the system constraint. After obtaining the optimal solution of P_1, store the following factors :

$\tilde{\beta}_1, z_1{}^*,\ \tilde{B}_1{}^{-1}, \bar{I}_{11}, \bar{I}_{12}$, and L_1.

Set $k=2$ and go to Step 2.

Step 2 : Reset the index sets of basic and nonbasic vectors at the kth priority level by(28) and(29). Recalculate the following factors at the kth pripority level by (32)-(35) in which z_k is used the sparsity technique, *i.e.*, KGOAL :

$\tilde{B}_k{}^{-1}, \beta_k, \pi_k$, and z_k.

Step 3 : If any element of β_k is negative, go to step 8. Otherwise, go to Step 4.

Step 4 : Calculate the reduced cost vector by (37) using the sparsity technique, *i.e.*,

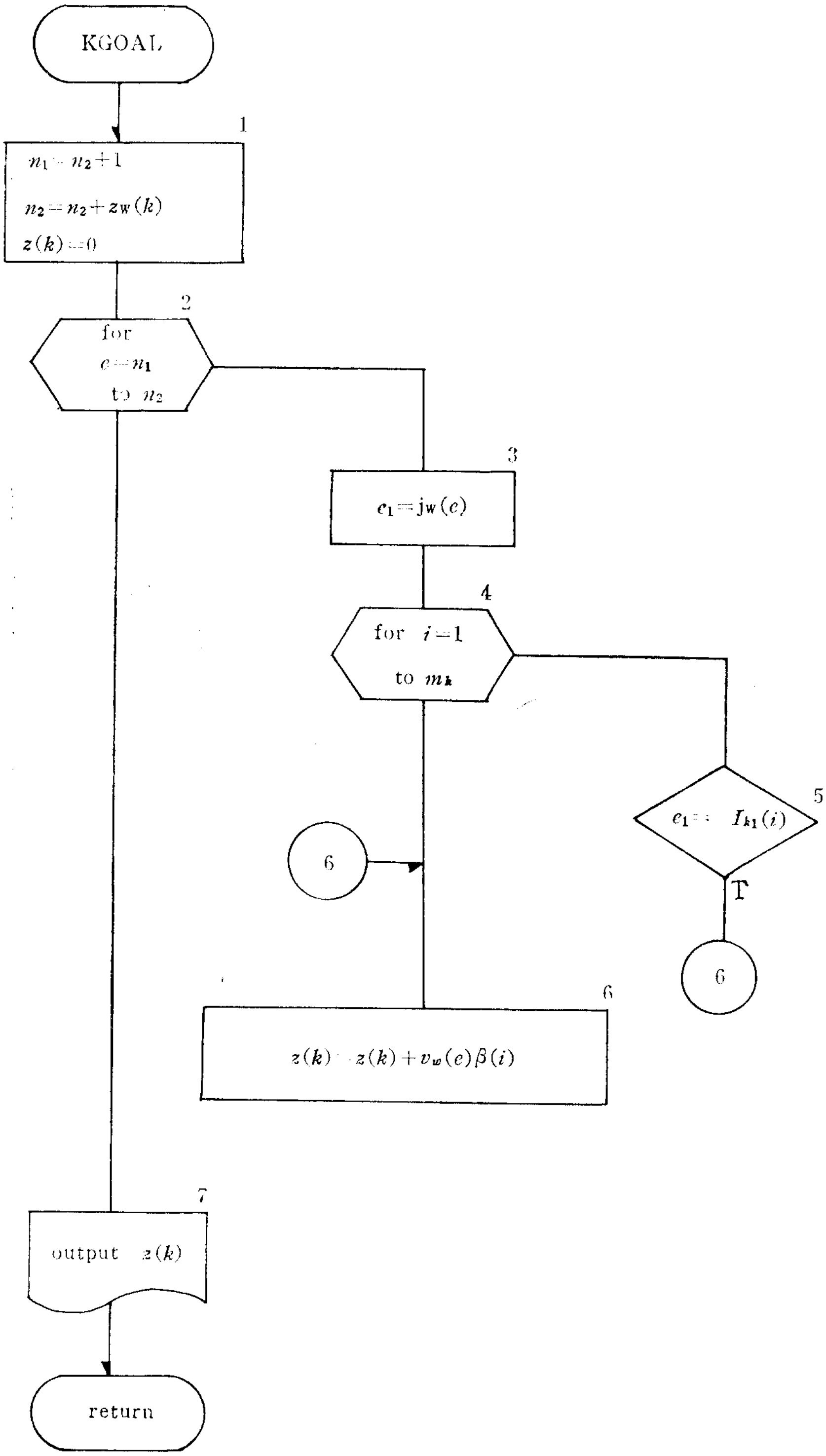

Fig. 1 Flowchart of KGOAL Routine

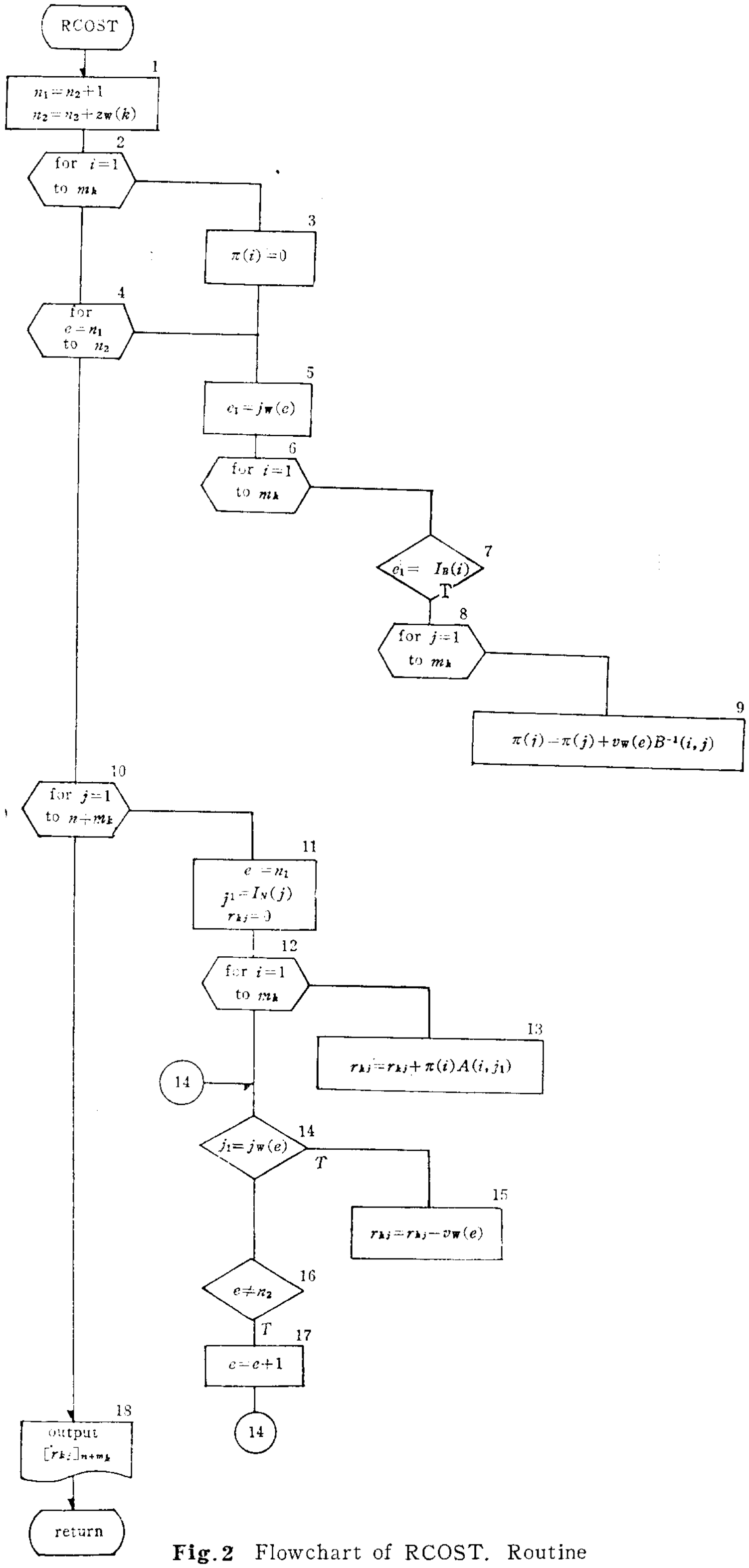

Fig. 2 Flowchart of RCOST. Routine

RCOST. Determine an incoming vector from the basis using the optimality test

$$r_{ks} = \max_{j} \{r_{kj} \mid r_{kj} < 0, \ j \in I_{k2}\} \tag{39}$$

If $s=0$, go to Step 7. Otherwise, go to Step 5.

Step 5 : Compute the basis representation corresponding to the incoming vector a_s by

$$\alpha_s = \tilde{B}^{q-1} \tilde{a}_s. \tag{40}$$

Determine an outgoing vector from the basis using the minimum ratio test

$$\frac{\beta_r}{\alpha_{rs}} = \min_{i} \left\{ \frac{\beta_i}{\alpha_{is}} \mid \alpha_{is} > 0, \ i=1, \cdots, \tilde{m}_k \right\}. \tag{41}$$

Go to Step 6.

Step 6 : Update the following factors at the kth priority level by the elementary matrix :

$\bar{\pi}_k$, $\bar{B}_k^{-1}$, $\bar{\beta}_k$, and $\bar{z}_k$.

Also, update the index sets of basic and nonbasic vectors, respectively, by

$$\begin{aligned} \bar{I}_{k1} &= \tilde{I}_{k1} - \{i_r\} + \{s\} \\ \bar{I}_{k2} &= \tilde{I}_{k2} - \{s\} + \{i_r\}. \end{aligned} \tag{42}$$

Return to Step 2.

Step 7 : If $k=q$, stop with the optimal solution $\tilde{\beta}_q$ and goal function z_k^*, $k=1, 2, \cdots, q$. Otherwise, store $\tilde{\beta}_k^{-1}$, $\tilde{B}_k$, z_k^*, $\tilde{I}_{k1}$, $\tilde{I}_{k2}$, and L_k. Reset $k=k+1$ and return to Step 2.

Step 8 : Determine an outgoing vector $\tilde{a}_{i_r}$ from the basis by

$$\beta_r = \min_{i} \{\beta_i \mid \beta_i < 0, \ i=\tilde{m}_{k-1}+1, \cdots, \tilde{m}_k\}. \tag{43}$$

If all the elements of $\alpha_{r\cdot}$ are positive or zero, stop with the unbounded GP problem. Otherwise, go to Step 9.

Step 9 : Compute the reduced cost by (37) using RCOST routine. Determine an incoming vector a_s to the basis by the minimum ratio test (38). Calculate the basic representation α_s corresponding to the incoming vector by (40). Return to Step 6.

Now we show the flowcharts of KGOAL and RCOST routines included in the revised iterative goal simplex method in Fig. 1 and 2, respectively.

Microcomputer implementation of programming projects requires careful consideration of every aspect of program design in order to fully exploit the limited resources available. A good program is defined that it is correct, *i.e.*, provides desired results, fast, accurate, hardware independent, efficient with storage, and easily modified. Thensen[22] suggests that operations analysis software should place heavy emphasis on “user impact” or ease of use. In the software package GP-IRS/S based on the revised iterative goal simplex method using the sparsity technique we designed an interactive and user-friendly system. Also, in order to prevent the problems of degeneracy and cycling we adopted *Bland's method* [4]

5. Numerical Example and Computational Experience

To illustrate the revised iterative goal simplex method using the sparsity technique on a microcomputer, the following GP problem with five priority levels, five system and six goal

Table 1. GP Test Problem

Goal Programming : GP—IRS/S

goal functions :

Z 1 DN 1 : 1.00
Z 2 DN 2 : 1.00
Z 3 DN 3 : 15.00 DN 4 : 17.00
Z 4 DN 5 : 1.00
Z 5 DN 6 : 1.00
Z 0 AF 1 : 1.00

Constraints :

	X 1	X 2	X 3	X 4	X 5	DN 1	DN 2	DN 3	DN 4	DN 5
	DN 6	DP 1	DP 2	DP 3	DP 4	DP 5	DP 6	AF 1	SK 1	SK 2
	SK 3	SK 4	SP 1	RHS						
G 1	−10.000	0.000	0.000	50.000	47.000	0.000	0.000	0.000	0.000	0.000
	0.000	0.000	0.000	0.000	0.000	0.000	0.000	1.000	0.000	0.000
	0.000	0.000	−1.000	200.000						
G 2	− 1.000	2.000	3.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	1.000	0.000
	0.000	0.000	0.000	100.000						
G 3	0.000	−1.000	0.000	1.000	0.000	0.000	0.000	0.000	0.000	0.000
	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
	0.000	0.000	10.000							
G 4	0.000	0.000	−1.000	0.000	1.000	0.000	0.000	0.000	0.000	0.000
	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
	1.000	0.000	0.000	20.000						
G 5	0.000	2.000	1.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
	0.000	1.000	0.000	400.000						
G 6	10.000	0.000	0.000	50.000	47.000	1.000	0.000	0.000	0.000	0.000
	0.000	−1.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
	0.000	0.000	0.000	1200.000						
G 7	0.000	10.000	5.000	20.000	12.000	0.000	1.000	0.000	0.000	0.000
	0.000	0.000	−1.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
	0.000	0.000	200.000							
G 8	0.000	1.000	0.000	−1.000	0.000	0.000	0.000	1.000	0.000	0.000
	0.000	0.000	0.000	0.000	−1.000	0.000	0.000	0.000	0.000	0.000
	0.000	0.000	0.000	0.000	0.000					
G 9	0.000	0.000	1.000	0.000	−1.000	0.000	0.000	0.000	1.000	0.000
	0.000	0.000	0.000	0.000	−1.000	0.000	0.000	0.000	0.000	0.000
	0.000	0.000	0.000	0.000						
G 10	1.000	−2.000	−3.000	0.000	0.000	0.000	0.000	0.000	0.000	1.000
	0.000	0.000	0.000	0.000	0.000	−1.000	0.000	0.000	0.000	0.000
	0.000	0.000	0.000	0.000						
G 11	0.000	10.000	5.000	20.000	12.000	0.000	0.000	0.000	0.000	0.000
	1.000	0.000	0.000	0.000	0.000	0.000	−1.000	0.000	0.000	0.000
	0.000	0.000	0.000	9000.000						

constraints, and five decision variables is considered [11] :

$$
\begin{array}{ll}
\min & z=p_1d_1^-+p_2d_2^-+p_3\,(15d_3^-+17d_4^-)+p_4d_5^-+p_5d_6^- \\
\text{s.t.} & -10x_1 \qquad\qquad +50\;x_4+47\;x_5 \geq 200 \\
 & -x_1+2\;x_2+3\;x_3 \leq 100 \\
 & -\;x_2\;+\;x_4 \leq 10 \\
 & -\;x_3\;+\;x_5 \leq 20 \\
 & 2x_2+\;x_3 \leq 400 \\
 & -10\;x_1 \qquad +50\;x_4+47\;x_5+d_1^--d_1^+=1200 \\
 & 10x_2+5\;x_3+20\;x_4+12\;x_5+d_2^--d_2^+=2000 \\
 & x_2\;-\;x_4\;+d_3^--d_3^+=\;0 \\
 & x_3\;-\;x_5+d_4^--d_4^+=\;0 \\
 & x_1-2\;x_2-3\;x_3\;+d_5^--d_5^+=\;0 \\
 & 10x_2+5\;x_3+20\;x_4+12\;x_5+d_6^--d_6^+=9000 \\
 & x_j,\;d_i^-,\;d_i^+\geq 0,\;j=1,2,\cdots,\;5,\;i=1,2,\cdots,6.
\end{array}
$$

After interactively entering data of the GP problem from the keyboard, the GP data are displayed on CRT as shown in Table 1. The user can confirm the GP problem prior to its use and detect errors if confirmed those. In the software package GP-IRS/S we involved menu-driven selections where various choices are displayed along with simple commands. Table 2 shows three arrays of the priority weight matrix of the GP problem. We can easily save microcomputer storaging capacity using the sparsity technique. Table 3 and Table 5 represent the initial and final computational processes by GP-IRS/S. In Table 3 the first problem consisting of five system and one goal constraints was solved at the 3rd iteration. Table 4 shows basic and nonbasic variables, basic solution, goal function and reduced cost at each problem. Table 5 presents optimal basis inverse, basic solution, goal function, and reduced cost identifying optimality test to the 5th problem at the 12th iteration. An optimal solution,

Table 2. Arrays for the Sparsity Technique

k/e	1	2	3	4	5	6
z_w(k)	1	1	2	1	1	
v_w(e)	1.0	1.0	15.0	17.0	1.0	1.0
j_w(e)	12	13	14	15	16	17

Table 3. Computational Process at the 1st Problem

```
basic variables :
   AF 1    SK 1    SK 2    SK 3    SK 4
nonbasic variables :
    X 1     X 2     X 3     X 4     X 5     SP 1
basis inverse :
               1.000        0.000        0.000        0.000        0.000
               0.000        1.000        0.000        0.000        0.000
               0.000        0.000        1.000        0.000        0.000
               0.000        0.000        0.000        1.000        0.000
               0.000        0.000        0.000        0.000        1.000
```

```
basic solution :
              200.000      100.000      10.000      20.000      400.000
goal function :
Z 0=          200.000
pricing vector :
                1.000        0.000       0.000       0.000        0.000
reduced cost :
S=4 :         -10.000        0.000       0.000      50.000       47.000      -1.000
iteration : 1
pivot column :
R=1 :          50.000        0.000       1.000       0.000        0.000
basic variables :
    X 4    SK 1    SK 2    SK 3    SK 4
nonbasic variables :
    X 1    X 2    X 3    AF 1    X 5    SP 1
basis inverse :
                0.020        0.000       0.000       0.000        0.000
                0.000        1.000       0.000       0.000        0.000
               -0.020        0.000       1.000       0.000        0.000
                0.000        0.000       0.000       1.000        0.000
                0.000        0.000       0.000       0.000        1.000
basic solution :
                4.000      100.000       6.000      20.000      400.000
goal function :
Z 0=            0.000
pricing vector :
                0.000        0.000       0.000       0.000        0.000
reduced cost :
S=0 :           0.000        0.000       0.000      -1.000        0.000       0.000
subproblem : 1
eliminated variables :
   AF 1
basic variables :
    X 4    SK 1    SK 2    SK 3    SK 4    DN 1
nonbasic variables :
    X 1    X 2    X 3    X 5    SP 1    DP 1
basis inverse :
                0.020        0.000       0.000       0.000        0.000       0.000
                0.000        1.000       0.000       0.000        0.000       0.000
               -0.020        0.000       1.000       0.000        0.000       0.000
                0.000        0.000       0.000       1.000        0.000       0.000
                0.000        0.000       0.000       0.000        1.000       0.000
               -1.000        0.000       0.000       0.000        0.000       1.000
basic solution :
                4.000      100.000       6.000      20.000      400.000    1000.000
goal function :
Z 1=         1000.000
pricing vector :
               -1.000        0.000       0.000       0.000        0.000       1.000
```

Table 4. Optimal Solution and Goal Functions to Each Subproblem

```
  The 1st problem at iteration 3 :
 basic variables :
      X 4    SK 1    SP 1    SK 3    SK 4     X 2
 nonbasic variables :
      X 1    DN 1     X 3     X 5    SK 2    DP 1
 basic solution :
                24.000      72.000     1000.000       20.000      372.000       14.000
 goal function :
 Z 1=            0.000
 reduced cost :
 S =0 :          0.000      -1.000        0.000        0.000        0.000        0.000
---------------------------------------------------------------------------------------
  The 2nd problem at iteration 5 :
 basic variables :
      X 4    SK 2    SP 1    SK 3    SK 4     X 2     X 1
 nonbasic variables :
    DN 2     X 3     X 5    SK 1    DP 1    DP 2
 basic solution :
    46.667      70.000     1000.000       20.000      186.667      106.667      113.333
 goal function :
 Z 2=            0.000
 reduced cost :
 S=0 :          -1.000       0.000        0.000        0.000        0.000        0.000
---------------------------------------------------------------------------------------
  The 3rd problem at iteration 7 :
 basic variables :
      X 4    SK 2    SP 1    SK 3    SK 4     X 2     X 1     X 3    DP 1
 nonbasic variables :
    DN 3     X 5    SK 1    DP 1    DP 2    DP 3    DN 4
 basic solution :
                58.095      10.000     1000.000       71.429      232.381       58.095
 goal function :
  Z 3=           0.000                                170.476       51.429       51.429
 reduced cost :
 S=0 :         -15.000       0.000        0.000        0.000        0.000        0.000
                                                                              -17.000
---------------------------------------------------------------------------------------
  The 4th problem at iteration 8 :
 basic variables :
      X 4    SK 2    SP 1    SK 3    SK 4     X 2     X 1     X 3    DP 4    SK 1
 nonbasic variables :
      X 5    DN 5    DP 1    DP 2    DP 3    DP 5
 basic solution :
                62.857      10.000     1000.000       42.857      251.429       62.857
 goal function :
 Z 4=            0.000                   194.286       22.857       22.857      100.000
 reduced cost :
 S=0 :           0.000      -1.000        0.000        0.000        0.000        0.000
```

Table 5. Computational Process at the 5th Problem

```
iteration : 12
pivot column :
R=1 :         0.500    0.000    0.000    0.000   -2.000    0.500   -2.200   -0.200   -1.000   -0.200
              2.000
basic variables :
     X 3     SK 2     SP 1     SK 3     DP 2      X 2      X 1     DP 5      X 5     SK 1     DN 6
nonbasic variables :
    DP 4     DP 1     SK 4     DP 3      X 4     DP 6
basis inverse :
              0.000    0.000    0.000    0.000    1.000    0.000    0.000   -2.000    0.000    0.000
              0.000
              0.000    0.000    1.000    0.000    0.000    0.000    0.000    1.000    0.000    0.000
              0.000
             -1.000    0.000    0.000    0.000    0.000    1.000    0.000    0.000    0.000    0.000
              0.000
              0.000    0.000    0.000    1.000    0.000    0.000    0.000    0.000    1.000    0.000
              0.000
              0.000    0.000    0.000    0.000   17.000    0.000   -1.000  -24.000  -12.000    0.000
              0.000
              0.000    0.000    0.000    0.000    0.000    0.000    0.000    1.000    0.000    0.000
              0.000
              0.000    0.000    0.000    0.000    4.700   -0.100    0.000   -9.400   -4.700    0.000
              0.000
              0.000    0.000    0.000    0.000    1.700   -0.100    0.000   -5.400   -4.700   -1.000
              0.000
              0.000    0.000    0.000    0.000    1.000    0.000    0.000   -2.000   -1.000    0.000
              0.000
              0.000    1.000    0.000    0.000    1.700   -0.100    0.000   -5.400   -4.700    0.000
              0.000
              0.000    0.000    0.000    0.000  -17.000    0.000    0.000   24.000   12.000    0.000
              1.000
basic solution :
            400.000   10.000 1000.000   20.000 4800.000    0.000 1760.000  560.000  400.000  660.000
           2200.000
goal function :
Z 5=       2200.000
pricing vector :
              0.000    0.000    0.000    0.000  -17.000    0.000    0.000   24.000   12.000    0.000
              1.000
reduced cost :
S=0 :       -12.000    0.000  -17.000  -24.000   -4.000   -1.000
```

goal functions, row informations of system and goal constraints, and computing time are shown in Table 6.

Table 6. Optimal Solution and Goal Functions to Original GP Test Problem

```
number of iterations : 12
optimal solution
   X  1           1760.0000
   X  2              0.0000
   X  3            400.0000
   X  4              0.0000
   X  5            400.0000
goal functions
   Z  0              0.0000
   Z  1              0.0000
   Z  2              0.0000
   Z  3              0.0000
   Z  4              0.0000
   Z  5           2200.0000
row information of sys. constraints
                    G(X)            SK (or SP)          RHS
   G  1          1200.0000          -1000.0000        200.0000
   G  2          -560.0000            660.0000        100.0000
   G  3             0.0000             10.0000         10.0000
   G  4             0.0000             20.0000         20.0000
   G  5           400.0000              0.0000        400.0000
row information of goal constraints
                    G(X)                DN               DP                 RHS
   G  6          1200.0000            0.0000           0.0000           1200.0000
   G  7          6800.0000            0.0000        4800.0000           2000.0000
   G  8             0.0000            0.0000           0.0000              0.0000
   G  9             0.0000            0.0000           0.0000              0.0000
   G 10           560.0000            0.0000         560.0000              0.0000
   G 11          6800.0000         2200.0000           0.0000           9000.0000
computing time 00 : 05 : 41
```

To test the computational efficiency for the three algorithms, i.e., GP-RS, GP-RS/S, and GP-IRS/S, we selected seven GP test problems of various sizes, structures, and complexity as shown in Table 7[19]. Table 8 presents the computational results obtained when the test

Table 7. GP Test Problems[19]

Problem Number	Problem Size				Constraint Type
	q	m	n_S	n_D	
P 1	3	5	5	6	GS
P 2	4	5	2	11	Gd
P 3	5	5	3	10	G
P 4	5	8	3	17	Gd
P 5	8	9	3	19	Gd
P 6	5	11	5	12	GS
P 7	5	12	3	10	GS

q : the number of priority levels,
m : the number of constraints (including both structural and goal constraints),
ns : the number of decision variables,
n_D : the number of negative and positive deviational variables,
G : the problem consisting of only goal constraints,
GS: the problem consisting of both goal and structural constraints,
Gd: the problem consisting of only goal constraints in which there are additional goal constraints with respect to deviational variables.

Table 8. Computational Results (MBC-2000)

Problems	Algorithms : Time, minutes & seconds (Iterations)		
	GP-RS	GP-RS/S	GP-IRS/S
P 1	1′54″ (4)	1′25″ (4)	50″
P 2	2′12″ (4)	1′39″ (4)	53″
P 3	3′18″ (5)	2′07″ (5)	49″
P 4	12′48″ (10)	7′34″ (10)	2′05″
P 5	21′10″ (9)	8′53″ (9)	3′06″
P 6	30′39″ (13)	16′43″ (13)	5′39″
P 7	35′05″ (14)	14′19″ (14)	4′39″

GP-RS : the revised goal simplex method,
GP-RS/S : the revised goal simplex method using the sparsity technique,
GP-IRS/S : the iterative revised goal simplex method using the sparsity technique.

problems were solved by GP-RS. GP-RS/S, and GP-IRS/S. in terms of computational time, we can easily determine that the revised iterative goal simplex method using the sparsity technique, i.e., GP-IRS/S, is the most efficient technique for GP test problem. We can state the GP-IRS/S package could be a useful tool on microcomputers for solving GP problems encountered in many small organizations.

6. Conclusions

Goal programming is one of the most powerful management techniques for multiple criteria decison making problems. GP has performed well for analyzing a wide variety of realworld decision problems. On the other hand, a microcomputer is emerging as an important tool for the operation of industrial, business, educational, and nonprofit organizations.

In this paper we have proposed a revised iterative goal simplex method using the sparsity technique on microcomputer. We have tested three GP algorithms on the microcomputer. The computational experience, based on the application of these algorithms to seven GP problems, indicated that the revised iterative goal simplex method using the sparsity technique, i.e., GP-IRS/S, is preferable over algorithms that are based on the goal simplex methods, i.e., GP-RS and GP-RS/S. We can state the GP-IRS/S software package could be a useful tool on microcomputers for solving linear goal programming problems encountered in many fields in Industrial Engineering and Management Science.

References

1. J.L. Arthur & "A. Ravindran, "An efficient goal programming algorithm using constraint partitioning and variable elimination", *Manage. Sci.* 24, 867-868, 1978.
2. J.L. Arthur & A. Ravindran, PAGP : "A partitioning algorithm for (linear) goal programming problems", *ACM Trans. on Math. Software* 6, 378-386, 1980.
3. J.L. Arthur & A. Ravindran, "A branch and bound algorithm with constraint partitioning for integer goal programming problems", *European, J. Oper. Res.* 4, 421-425, 1980.
4. R.G. Bland, "New finite pivoting rules for the simplex method', *Math. of Oper. Res.* 2, 103-107, 1977.
5. A. Charnes & W.W. Cooper, "Goal programming and multiple objective optimization", *European J. Oper. Res.* 1, 39-54, 1977.
6. J.P. Dauer, "A parametric programming algorithm for the solution of goal programs", Working Paper, Dept. of Math. & Statistics, Univ. of Nebraska-Lincoln, 1981.
7. J.P. Dauer & R.J. Krueger, "An iterative approach to goal programming", *Oper. Res. Quart.* 28, 671-681, 1977.
8. J.P. Dauer & R.J. Krueger, "Mathematical programming techniques in majorization", *J. Optim. Theory& Appl.* 25, 361-373, 1978.
9. A. van Delft & P. Nijkamp, "Multi-criteria analysis and regional decision making", *J. Regional Sci.* 17, 195-205, 1977.
10. M. Gen, "Goal programming Ⅰ, Ⅱ, Ⅲ", *Basic Math.* 14, 23-31, 51-57, 58-64, 1981, in Japanese.
11. R.V. Hartley, *Operations Research : A Managerial Emphasis,* Goodyear Pub., Pacific Palisades, Calif, 1976.
12. J.K. Ho, "Holistic preference evaluation in multiple criteria optimization", in G.B. Dantzig, *et al.* eds., *Large-Scale Linear Programming*, IIASA, Laxenblug, Austria, 976-1024, 1981.
13. C.L. Hwang & A.S.M. Masud, *Multiple objective Decision Making Methods and Applications : A State-of-the-Art Survey*, Springer-Verlag, New York, 1979.
14. J.P. Ignizio, *Goal Programming and Extention*, Lexington Books, Lexington, 1976.
15. J.P. Ignizio, "A review of goal programming : A tool for multiple objective analysis, *J. Oper. Res. Soc.* 29, 1109-1119, 1978.
16. J.P. Ignizio & J.H. Perlis, "Sequential linear goal programming : Implementation via MPSX", *Computer & Oper. Res.* 6, 141-145, 1979.
17. J.S.H. Kornbluth, "A survey of goal programming", *OMEGA* 1, 193-205, 1973.
18. S.M. Lee, *Goal Programming for Decision Analysis,* Auerbach Pub., Philadelphia, 1972.
19. S.M. Lee & M. Gen, "Computational experience of goal programming algorithms on microcomputers", presented at *The EURO-TIMS XXV Conf.*, Lausanne, 1982.
20. S.M. Lee, M. Gen, & J.P. Shim, "Goal programming for decision making : A state-of-the-art survey", Working Paper, Dept. of Manage., Univ. of Nebraska-Lincoln, 1982.

21. S.M. Lee, M. Gen, & B.H. Rho, "A revised iterative algorithm for decomposition goal programming, *Int. J. Systems Sci.*, Vol. 14, No. 12, 1383-1393, 1983.

22. J. Spronk, *Interactive Multiple Goal Programming*, Martinus Nijhoff Pub., Boston, 1981.

23. A. Thensen, *Computer Methods in Operations Research*, Academic Press, New York, 1978.

24. G.E. Whitehouse, C. S. Bauer, Y.A. Hosni, & C.S. Park, "Use of microcomputers to solve traditional Industrial Engineering problems", *Proc. of AIIE Spring Annual Conf.*, Detroit, 47-54, 1981.

25. M. Zeleny, *Multiple Criteria Decision Making*, McGraw-Hill, New York, 1981.

26. S. Zionts, "Multiple criteria decision making : An overview and several approaches", Working Paper, School of Manage., State Univ. of New York at Buffalo, 1980.