韓國經營科學會誌
第12券 第2號
1987年 12月

부분문제가 같은 블록대각형 선형계획법문제의 효율적인 방법*

An Efficient Algorithm for a Block Angular Linear Program with the Same Blocks

양 병 학**
박 순 달**

ABSTRACT

The objective of this paper is to develop an efficient method with small memory requirement for a feed-mixing problem on a micro computer.

First, this method uses the decomposition principle to reduce the memory requirement. Next, the decomposition principle is modified to fit the problem. Further four different variations in solving subproblems are designed in order to improve efficiency of the principle. According to the test with respect to the processing time, the best variation is such that the dual simplex method is used, and the optimal basis of a previous subproblem is used as an initial basis, and the master problem is (M+1) dimensional.

In general, the convergence of solution becomes slower near the optimal value. This paper introduces a termination criterion for a sufficiently good solution. According to the test, 5% -tolerence is acceptable with respect to the relation between the processing time and the optimal value.

1. 서 론

사료 배합문제는 전형적인 線形計劃法문제중의 하나이다. 우리나라에서는 보통 최대 50여가지의 배합 사료를 생산하는데 이들 배합사료를 생산하기 위하여 사용되는 원료는 옥수수, 소금, 약품 등 50여가지이다. 그리고 이들 원료를 이용하여 각 사료가 갖춰야 할 단백질, 회분, 인, 비타민 등 50여가지의 영양소의 최소 요구량을 맞추어 가장 값싼 사료를 생산한다. 따라서 사료 문제란 각제품의 영양성분, 원료배합비에 대한 규격과 제품별 생산 요구량을 만족시키면서 최소 비용을 얻을 수 있는 배합비를 산정하는 문제로써 이것을 數理計劃法식으로 표현하면 다음과 같다[3].

$$
\begin{aligned}
\text{Min } & \sum_{i=1}^{m} c_{io} x_{io} \\
\text{s. t. } & x_{io} - \sum_{k=1}^{l} a_k x_{ik} = 0 \quad i=1,\cdots\cdots,m \\
& \sum_{i=1}^{m} x_{ik} = 1 \qquad k=1,\cdots\cdots l \\
& \sum_{i=1}^{m} b_{ij} x_{ik} - s_{jk} = 0 \quad j=1,\cdots\cdots n \\
& \qquad\qquad\qquad\qquad k=1,\cdots\cdots l \\
& l_{io} \le x_{io} \le u_{io} \\
& l_{ik} \le x_{ik} \le u_{ik} \\
& l_{(m+j)k,} \le s_{ik} \le u_{(m+j)k}
\end{aligned} \tag{1}
$$

이때 사용된 기호는 다음과 같다.

x_{io} : 원료 i의 사용량

x_{ik} : 제품 k에서 원료 i의 배합비

s_{jk} : 제품 k에서 단위당 성분 j의 함량

c_{io} : 원료 i의 구입가

* 이 연구는 문교부 연구비로 이루어진 "PC를 이용한 사료 배합 package 개발"의 일부임.

** 서울대학교 공대 산업공학과

a_k : 제품 k의 목표 생산량

b_{ij} : 원료 i의 단위당 성분 j의 함량

l_{io}, u_{io} : x_{io}의 상·하한

l_{ik}, u_{ik} : x_{ik}의 상·하한

$l_{(m+j)k}$, $u_{(m+j)k}$: s_{ik}의 상·하한

이 문제(1)은 선형계획법 형태로 제약식의 갯수가 $m+(n+1)\times l$인데 $l=50$, $m=50$, $n=50$인 경우에는 2,600개가 되어 대형문제가 된다. 그러나 문제(1)은 블록대각구조 문제로서 분해원리를 적용할 수 있다. 특히 이 문제는 변수가 上·下限을 가진 경우인데 이런 문제의 해법은 이미 개발되어 있다[2].

그럼에도 불구하고 소형전산기로 이 문제를 풀려고 할때는 실용성이 없을 정도의 오랜 전산시간과 전산기의 많은 기억용량을 필요로 한다. 이 연구는 이 기존 해법을 토대로 하여 계산시간과 기억용량을 줄여 소형전산기(개인용)으로써 실제 문제를 풀더라도 실용성이 있도록 하고자 하는데 목적이 있다.

2. 특 성

식(1)은 전형적인 블록대각 구조를 가진 형태의 문제인데 이 문제는 우선 식(2)와 같이 표현될 수 있다[3, 그림 2]. 그런데 이 사료 배합 문제는 일반적인 블록대각 구조의 문제와는 다른 특성을 지니고 있다. 특히 식(2)는

- 모든 부분문제의 행렬은 동일하고
- 목적함수의 계수는 0블록외는 모두 영

이라는 특성을 지니고 있다. 그래서 여기에서는 이들 특성을 이용하여 일반 블록대각 구조의 일반적인 해법인 Dantzig-Wolfe의 분해원리를 이 식(2)에 알맞게 수정하기로 한다.

$$
\begin{aligned}
&Min\ C_0X_0\\
&s.t.\ A_0X_0+A_1X_1+\cdots+A_lX_l=\mathrm{b}_0\\
&\qquad B_1X_1 \qquad\qquad\quad =\mathrm{b}_1\\
&\qquad\qquad B_1X_2 \qquad\quad =\mathrm{b}_1\\
&\qquad\qquad\qquad B_1X_l =\mathrm{b}_1\\
&L_k\le X_k\le U_k \quad k=0,1,\cdots,l\\
&X_k=[x_{1k}, x_{2k}, \cdots, x_{mk}, s_{1k}, s_{2k}, \cdots, s_{nk}]^T
\end{aligned}
$$

단 대문자와 고딕은 벡타를 나타내기로 함.

$$A_0=I_{(m\times m)}$$

$$A_k=-a_k[I:0],\ k=1, 2, \cdots, l$$

$$B_1=\begin{bmatrix} 1, & 1, & \cdots\cdots 1, & 0, & 0, \cdots\cdots 0\\ b_{11}, & b_{21}, & \cdots\cdots b_{m1}, & -1, & 0,\ 0, \cdots, 0\\ b_{12}, & b_{22}, & \cdots\cdots b_{m2}, & 0, & -1,\ 0 \cdots, 0\\ \vdots & \vdots & \vdots & \vdots & \ddots \\ b_{1n}, & b_{2n}, & \cdots\cdots b_{mn}, & 0, & 0, \cdots, -1 \end{bmatrix}$$

$$C_0=[c_{10}, c_{20}, \cdots\cdots c_{mo}]$$

$$\mathrm{b}_0=[0, \cdots\cdots 0]^T$$

$$\mathrm{b}_1=[1, 0, \cdots\cdots, 0]^T$$

그러면 식(2)에 Dantzig-Wolf의 분해원리를 적용할 수 있는데 분해원리에서는 주문제를 (m+1)의 제약식으로 만드는 경우와(m+l)의 제약식으로 만드는 경우가 있다[4, 16].

먼저 (m+1)차 방법을 보기로 하자. 우선 식(2)를 다음 식(3)으로 표시하기로 하자.

$$
\begin{aligned}
&Min\ C_0X_0\\
&s.t.\ A_0X_0+AX=\mathrm{b}_0\\
&\qquad\qquad BX=\mathrm{b}
\end{aligned}
\tag{3}
$$

$$L_0\le X_0\le U_0,\ L\le X\le U$$

단

$$B=\begin{bmatrix} B_1 & & 0\\ & B_1 & \\ & & \ddots \\ 0 & & B_1\end{bmatrix}, \mathrm{b}=\begin{bmatrix}\mathrm{b}_1\\ \mathrm{b}_1\\ \vdots\\ \mathrm{b}_1\end{bmatrix}, L=\begin{bmatrix}L_1\\ L_2\\ \vdots\\ L_l\end{bmatrix}, U=\begin{bmatrix}U_1\\ U_2\\ \vdots\\ U_l\end{bmatrix}, X=\begin{bmatrix}X_1\\ X_2\\ \vdots\\ X_l\end{bmatrix}$$

지금 (3)문제의 $BX=\mathrm{b}$의 가능해 집합을 S라 하자. 그러면

$$S=\{X \mid BX=\mathrm{b},\ L\le X\le U\} \tag{4}$$

라고 표현된다. 이때 S 중 정점해를 X^t라 하면 $X(\in S)$는

$$X=\sum_t X^t y^t=\sum_t (X_1^t y^t,\ X_2^t y^t,\ \cdots X_l^t y^t) \tag{5}$$

$$\sum_t y^t=1,\ y^t\ge 0$$

으로 표현된다. 이것을 (3)에 대입하여 정리하면

$$
\begin{aligned}
&Min\ C_0X_0+\sum_t d^t y^t\\
&s.t.\ A_0X_0+\sum_t P_t y_t=\mathrm{b}0\\
&\qquad\qquad \sum_t y^t=1\\
&\qquad\qquad L_0\le X_0\le U_0,\quad y^t\ge 0
\end{aligned}
\tag{6}
$$

단

$$P^t=AX^t,\ d^t=0 \tag{6}$$

이 된다. 이 식(6)을 식(3)의 주문제라고 하며 분

해원리란 문제(6)을 풀어서 문제(1)의 해를 찾는 것이다.

식(3)과 식(6)은 일반 대각구조 형태의 문제와는 목적함수에 차이가 있는데 여기에서는 단지C_0만이 존재하고 기타는 모두 零이다. 그래서 변수가 上·下限을 가진 경우의 분해원리〔2〕를 여기에 맞게 수정하면 다음과 같이 된다.

$(m+1)$차 분해원리

단계1 初期解

식(3)에 인공변수 X_s를 도입한다. 그래서 初期解는

$$X_0=A_0^{-1}b_0 \qquad (7)$$
$$X=0,\ X_s=b$$

이라 두자. 이때 주문제의 기저행렬

$M=\begin{bmatrix} A_0 & 0 \\ & \\ 0 & 1 \end{bmatrix}$이고 이것은 (m_0+1)次行列이다.

이때 만일

$$L_0 \le X_0 \le U_0 \qquad (8)$$
$$X_s=0 \qquad (9)$$

이면 이 초기해는 가능해가 되어 국면 2(phase 2)로 간다. 그렇지 않을 경우에는 다음 3가지의 경우가 생긴다.

경우 1 : 식(8)은 만족, 식(9)는 만족치 않을 경우

이 경우는 문제(6)을 푸는 단체법의 절차는 국면 2(phase2)이지만 d^t의 할인가를 구하기 위하여 부분문제를 풀때는 국면 1(phase 1)상태로 풀어야 한다.

경우 2 : 식(9)는 만족, 식(8)이 만족치 않을때

이 경우에는 문제(6)을 푸는 과정을 국면 1(phase 1)으로 풀어야 한다. 그리고 d^t를 평가할때 즉, 부분문제를 풀때는 국면 2(phase 2)상태로 풀어야 한다.

경우 3 : 식(8), 식(9) 모두 만족치 않을 경우

이 경우에는 c_{io}, d^t의 할인가를 구할때 모두 국면 1(phase 1)으로 풀어야 한다.

단계 2 단체승수

단체 승수 $\pi=d_M M^{-1}=(\pi_0,\ \pi_1)$, π_1은 스칼라이다. 그리고 d_M는 기저에 해당하는 목적함수의 계수이다. 여기서 d_M의 요소 d^t는 0이거나 c_{io}이다.

단계 3 진입변수선택

(1) 블록0

먼저 블록0에 대해 할인한다. 즉, $\bar{c}_{io}=\pi_o A_{oj}$를 구한다. 다음에

$$Min\ \ \bar{c}_{io}=\bar{c}_{so}$$

를 구한다. 만일 $\bar{c}_{so}\le 0$이면 (2)로 넘어간다. 만일 $\bar{c}_{so}>0$이면 x_{so}가 進入변수가 된다. 그리고 단계 4로 간다.

(2) 기타 블록

$\bar{d}^t$를 구한다. 이 $\bar{d}^t$를 구하기 위하여

$$Min\ \ \pi_o AX$$
$$s.t.\ BX=b \qquad (10)$$
$$L\le X\le U$$

을 풀어 그 최적해를 X^* 그리고 그때의 목적함수의 값을 Z^*라고 하자.

그러면

$$Min\ \ \bar{d}^t=\bar{d}^s=Z^*+\pi_1$$

이다. 만일 여기서 $\bar{d}^s\ge 0$이면 이 문제의 최적해를 찾은 것이다.

만일 $\bar{d}^s<0$이면 y^s이면 進入변수가 된다.

단계 4 進入列수정

(1) 進入列이 블록0에서 나올 경우

문제(6)의 블록0의 S째列은 $[A_{os}\ 0]^T$이다. 그래서 進入列을 수정하면

$$F=M^{-1}\begin{bmatrix} A_{os} \\ 0 \end{bmatrix}$$

와 같이 된다.

(2) 進入列이 블록 0가 아닐 경우

이 경우 進入列은 $[P^s,\ 1]^T$인데 이 P^s는 식(10)에서 구해진 최적해 X^*를 이용하여

$$P^s=AX^*$$

로 구해진다. 그래서 이 進入列을 수정하면

$$F=M^{-1}\begin{bmatrix} P^s \\ 1 \end{bmatrix}=M^{-1}\begin{bmatrix} AX^* \\ 1 \end{bmatrix}$$

이 된다.

단계 5 탈락변수선택

우선 문제(6)의 우변상수[b 1]^{T}를 수정하여야 한다. 이것은

$$R = M^{-1}\begin{bmatrix} b \\ 1 \end{bmatrix}$$

로 구해진다. 그 다음 일반 單體法에서와 같이 比率검정(Ratio test)으로써 기저탈락변수를 선택한다.

단계 6 M^{-1}의 수정

進入변수와 탈락변수가 결정되면 보편적인 일반 單體法에서와 같이 M^{-1}를 수정한다.

이상에서 $(m+1)$차 방법을 보았는데 또한 이 $(m+1)$차 분해원리를 조금만 수정하면 식(2)에 알맞는 $(m+l)$차 분해원리로 발전시킬 수 있다.

여기서 특히 관심이 있는 부분은 단계 3의 식(10)을 푸는 부분이다. 식(3)에서 보는 바와 같이 B가 독립적인 블록대각구조를 가지고 있기 때문에 식(10)의 해 X는 다음식

$$\begin{aligned} &Min\ \pi_0 A_k X_k \\ &s.t.\ B_1 X_k = b_1 \qquad\qquad (11) \\ &\quad L_k \le X_k \le U_k \end{aligned}$$

을 풀어 이들 해를 합한 것과 같다. 그래서 식 (11)의 최적해를 X^*_k, 이때 목적함수의 값을 Z^*_k라고 할 때

$$X^* = \begin{bmatrix} X^*_1 \\ X^*_2 \\ \vdots \\ X^*_l \end{bmatrix}$$

$$Z^* = \sum Z^*_k$$

이다.

지금 식(11)을 부분문제k라고 부르기로 한다. 그리고

$$S_k = \{X_k \mid B_1 X_k = b_1,\ L_k \le X_k \le U_k\} \qquad (13)$$

이라고 두자. 그런데 부분문제의 기저해집합을 S^1_k라고 두면 모든 부분문제의 기저 집합은 동일하다. 즉 한부분 문제의 기저를 구하면 이것이 곧 다른 부분문제의 기저가 된다. 그렇다고 한 부분문제의 기저가능해가 다른 부분문제의 가능해가 될 필요는 없다.

그리고 쌍대해에 대해서는 다음 정리가 성립된다.

정리 부분문제의 쌍대가능기저는 다른 부분문제의 쌍대가능이다.

증명 부분문제 k를 보자. 여기서 쌍대가능기저를 D라고 하자.

그리고 이 기저에 대해 식(6)과 같은 主문제를 만들면 목적함수 계수는 식(6)′에서 보듯이

$$C_k = \pi_0 A_k$$

가 된다. 단 C_{kD} : C_k에서 기저 D에 해당하는 부분

c_{kj} : C_k의 j항

이라고 두면

$$\bar{c}_{kj} = c_{kj} - (C_{kD} D^{-1})(B_1)_j$$

단 $(B_1)_j$는 B_1의 j열

이 된다. 여기서 D가 쌍대가능이기 때문에

$$\bar{c}_{kj} \ge 0 \quad \forall j$$

이다.

한편 임의의 부분문제 g를 보자. 기저D는 이부분문제g에서도 기저해이다. 이제 이 D가 쌍대가능인지의 여부를 보자. 부분문제g에서

$$C_g = \pi_0 A_g$$

라고 두면 식(2)에서 보는 바와 같이

$$A_g = -a_g[I : 0]$$

이기 때문에

$$C_g = \frac{a_g}{a_k} \pi_o A_k = \frac{a_g}{a_k} C_k$$

가 성립한다. 그래서 부분문제g에서 목적함수 계수를 할인하면

$$\begin{aligned} \bar{c}_{gj} &= c_{gj} - (C_{gD} D^{-1})(B_1)_j \\ &= \frac{a_g}{a_k} c_{kj} - \left(\frac{a_g}{a_k} c_{kD} D^{-1}\right)(B_1)_j \\ &= \frac{a_g}{a_k}\{c_{kj} - (c_{kD} D^{-1})(B_1)_j\} \end{aligned}$$

가 된다. 그런데 괄호{ }은 비음이고 a_g, a_k는 동일부호이기 때문에 따라서 모든 j에 대해서

$$\bar{c}_{gj} \ge 0 \quad \forall j$$

가 되어 부분문제 k의 쌍대가능기저D는 다른 모든 부분문제의 쌍대가능기저가 된다.

3. 부분문제 해법

앞에서 서술했던 計算方法을 적용할때 가장 많은 시간이 소요되는 부분이 식(10)을 푸는 부분이다. 진입열을 선정하기 위해서는 항상 이 식(10)을 풀어야 하는데 이 식(10)을 푼다는 것은 결국 l개의 부분문제를 푼다는 것과 같다. 그래서 여기에서는 전술한 특성을 이용하여 4가지의 해법을 만들어 실제로 얼마나 효율적인지 실험해 보도록 한다.

부분문제의 해법을 설계할때 고려하여야 할 점은 첫째로 l개의 부분문제는 긴밀한 상호관계가 있기 때문에 한 부분문제의 기저해를 구하여 다른 문제의 기저해로 사용할 수 없는가 하는 것이다. 매회 l개의 부분문제를 풀 때 그 때마다 새로 풀기보다 이미 구해진 다른 부분문제의 해를 사용할 수 없는가 하는 것이다.

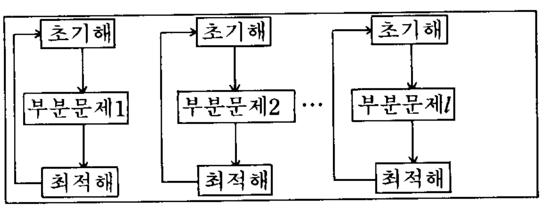
둘째로 l개의 부분문제를 매회 되풀이 하여 풀게되는데 같은 부분문제에 있어서 전번 회의 기저해를 이용할 수 없는가이다. 만일 전번 회의 기저해를 사용할 수 있으면 매회 다시 처음부터 풀 필요가 없어 시간이 절약될 수 있기 때문이다.

이런 두가지 관점에서 다음의 네가지 해법을 고려해보도록 한다.

해법 1 : 각 부분문제는 독자적으로 풀어가되 다른 부분문제의 기저해를 사용하지 않는다.

그러나 같은 부분문제에 있어서는 횟수가 거듭함에 따라 단지 목적함수만이 달라지기 때문에 지난 회의 최적 기저해를 다음 회의 초기해로 사용하기로 한다. 이렇게 하기 위해서는 항상 그 회의 l개의 부분 문제의 최적해를 보관하여야 한다.

이 방법을 그림으로 표시하면 〔그림 1〕과 같다.



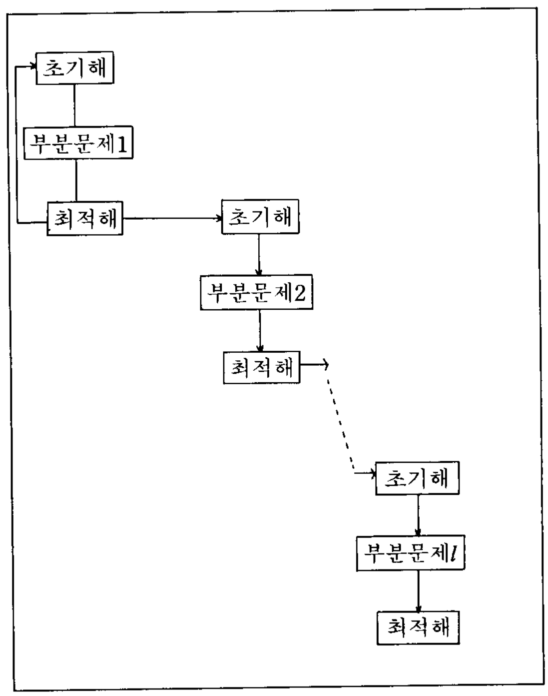
〔그림 1〕 해법 1의 개념도

해법 2 : 각 부분문제의 기저해는 동일하다. 이 점을 이용하여 첫번째 부분문제의 최적기저해를 다음 부분문제의 기저로 이용하는 방법이다. 이 때 이 기저가 다른 부분문제의 가능해일 보장은 없으므로 국면 1(phase 1)을 수행해야 할 지 모른다. 이와 같이 첫번째 부분문제부터 축차적으로 기저를 이월시키면서 모든 부분문제를 해결한다. 이 때 부분문제는 국면 1을 수행해야 할 위험은 있으나 하나의 최적 기저만 보관하면 되기 때문에 기억용량의 소요는 감소한다.

해법 3 : 해법 2와 같이 첫번째 부분문제의 최적기저를 다음번 부분문제가 받아들여서 축차적으로 모든 부분문제를 해결한다. 이 때 첫번째 부분문제는 최적해를 구했으므로 쌍대가능해가 된다. 정리 1에 의해 이 쌍대가능성은 다른 부분문제에서도 성립하게 된다. 이점을 이용해서 첫번째 부분문제를 제외한 나머지 부분문제는 쌍대단체법으로 해결한다.

지금까지는 분해해법 중에서 $(m+1)$차 방법을 사용하였다. 다음 마지막 방법은 $(m+l)$차 방법을 사용하기로 한다.

해법 4 : 여기에서는 $(m+1)$차 분해원리를 사용



〔그림 2〕 해법 2, 3, 4의 개념도

한다. 그리고 부분문제의 해결은 해법 3처럼 쌍대단체법을 사용한다.

이상의 4개의 해법에 대해서 수행시간을 측정했다. 사용한 전산기는 8088−processor를 내장한 16−bit전산기를 이용했고 문제는 (10×20×20), (19×20×20), (10×40×40), (19×40×40)의 네가지군에 대해서 각각 10개씩의 총 40문제에 대해서 실험했다. 실험결과는 〔표 1〕과 같다. 각 해법간에 수행시간에 차이가 있는지를 검정하기 위해서 수행시간의 평균치에 대한 쌍체비교 검정을 실시한 결과 각 문제의 크기별로 〔표 2〕의 결과가 나왔다.

〔표 1〕 각 해법의 수행시간의 평균치

문제의 크기	해법 1	해법 2	해법 3	해법 4
10×20×20	14.55.66	5.51.08	4.03.04	5.01.01
19×20×20	24.18.07	9.35.01	6.21.01	9.54.07
10×40×20	56.49.00	38.12.09	23.51.03	27.03.09
9×40×40	1.44.35.07	1.10.13.09	44.03.09	52.54.00

(단위 시간, 분, 초 1/100초, 각 문제의 크기별로 10개씩 실험)

〔표 2〕 각 문제의 쌍체비교 검정 결과

		해법 2	해법 3	해법 4
해법 1	a	***	***	***
	b	**	***	***
	c	**	***	***
	d	***	***	***
해법 2	a		**	*
	b		***	−
	c		***	***
	d		***	***
해법 3	a			***
	b			***
	c			***
	d			*

단, ***는 유의 수준 0.001
**는 유의 수준 0.01
*는 유의 수준 0.05
−는 차이가 있다고 할 수 없음.
a는 (10×20×20) 문제
b는 (19×20×20) 문제
c는 (10×40×40) 문제
d는 (19×40×40) 문제

〔표 1〕을 보면 수행시간에서 부분문제가 독립적이라고 본 해법 1이 나머지 해법들보다 열등한 것으로 나타났으며 부분문제간에 기저를 주고받는 해법 2, 3, 4가 더 우수한 것으로 나타났다. 또한 해법2, 3, 4중에서는 해법3이 가장 우수하다.

해법2, 3, 4는 정리 1을 사용하여 쌍대가능인 기저를 이용하는데 이 쌍대가능기저해가 원가능성만 유지되면 바로 최적이 된다. 그런데 제품별로 요구하는 성분 함량의 상·하한과 원료 배합비의 상·하한에 대한 규격이 유사한 제품이 많아서 많은 부분문제에서 최적기저가 동일하여 해법 2, 3, 4의 경우에 수행시간이 단축되고 있다.

소형전산기에서는 기억용량의 요구량도 주요한 문제가 된다. 각 해법간에 요구하는 기억용량을 분석해보면 〔표 3〕과 같다. 〔표 3〕에서 (50×50×50) 문제에 대해 요구되는 실제 기억용량을 구해보면 〔표 4〕와 같다.

〔표 3〕 각 해법에 따라서 기억시켜야 하는 변수의 수

해법	실 변 수	정수변수	정 검해	기 저
해법 1	$7l+22m+10n+mn+3ml+3nl+m^2+l^2$	$6m+2n$	$(m+1)(m+n+1)l$	$l[4(n+m+1)+(m+1)(m+1)]$
해법 2	$7l+22m+10n+mn+3ml+3nl+m^2+l^2$	$6m+2n$	$(m+1)(m+n+1)l$	$4(n+m+1)+(m+1)(m+1)$
해법 3	$8l+23m+10n+3ml+3nl+m^2+l^2$	$6m+2n$	$(m+1)(m+n+1)l$	$4(n+m+1)+(m+1)(m+1)$
해법 4	$12l+21m+10n+mn+5ml+3nl+m^2+l^2$	$6m+2n+2l$	$(m+1)(m+n+1)$	$4(n+m+1)+(m+1)(m+1)$

〔표 4〕 해법에 따라서 요구되는 기억용량

해 법	실 변 수	정수 변수	정 점 해	기 저
해 법 1	97.8K	0.8K	1030K	601K
해 법 2	97.8K	0.8K	1030K	12K
해 법 3	98.2K	0.8K	1030K	12K
해 법 4	126.6K	0.9K	40.4K	12K

〔표 4〕에서 보듯이 해법 1의 경우에는 전체 요구량이 1M-byte가 넘어서 반드시 대용량의 보조기억장치가 요구된다. 해법 2, 3의 경우에는 정규적인 보조기억장치로 해결 가능하고 해법 4의 경우에는 256K-byte의 주기억 장치로도 수행 가능하다.

4. 종료 기준

분해원리로 문제를 풀어갈때 목적함수의 값은 횟수가 거듭할수록 최적해에 수렴해 간다. 그런데 목적함수의 값은 최적해에 가까워질수록 수렴속도가 늦다. 그래서 적절한 허용오차로써 계산을 중단시킬 필요가 있다. 여기에서는 인위적으로 계산을 중단시킬때 가장 적절한 허용오차에 대해 보기로 한다.

목적함수의 값은 갈수록 감소해 가는데 그 下限값은 다음과 같이 주어진다 [16, p164].

$$Min\ Z \geq Z_M + Min\ \bar{d}^t$$

단 $Min\ Z$: 식(6)의 최적해의 값

Z_M : 식(6)의 현재의 값

그리고 $\underset{t}{Min}\ \bar{d}^t \leq 0$이고 최적에 이르면

$$Min\ Z = Z_M$$
$$Min\ \bar{d}^t = 0$$

가 된다. 이 $\underset{t}{Min}\ \bar{d}^t$는 횟수가 거듭할수록 항상 감소하는 것은 아니지만 감소 추세에 있기 때문에 현재의 해가 최적해에 얼마나 수렴했는지의 여부는 이 값으로 어느정도 알아 볼 수 있다.

그런데 최적해에 대한 수렴도는 절대값보다는 목적함수 값에 대한 상대적인 값이 더욱 뜻이 있을 것이기 때문에 현재 해의 수렴도는 $Min\ \bar{d}^t$이 식(10)에서 보듯이 $Z^* + \pi_1$이기 때문에 다음 값으로 나타내기로 한다.

$$\nu = \frac{\text{식(10)의 최적해} + \pi_1}{Z_M}$$

그래서 허용오차를 ε이라고 할때 $\nu \leq \varepsilon$가 되면 계산을 끝내게 된다.

그런데 여기서 ε는 어떤값을 줄것인가가 문제가 된다. 본 실험에서는 (17×26×31)크기의 10개의 문제에 대해 ε를 5%, 2.5%, 1%로 주었을때 수행시간과 목적함수값을 구했다. 그 결과는 다음과 같다.

〔표 5〕 허용 오차와 수행 시간 및 목적함수값과의 관계 (크기(17×26×31)의 10개 문제에 대한 시간과 목적함수값의 평균)

허용오차		해 법 1	해 법 2	해 법 3	해 법 4
5%	수행시간	32.44	18.13	10.46	10.54
	목 적 값	2.712346	2.709020	2.696451	2.702051
2.5%	수행시간	37.27	20.00	11.18	11.32
	목 적 값	2.705886	2.702059	2.696157	2.699583
1%	수행시간	43.01	28.04	14.40	15.20
	목 적 값	2.686159	2.688329	2.685191	2.682994

단, 수행 시간의 단위는 분·초, 목적값의 단위는 10^8

〔표 5〕를 살펴보면 허용 오차를 감소시킴에 따라 수행시간은 증가하고 목적함수 값은 감소함을 쉽게 알 수 있다. 수행시간의 증가율과 목적함수 값의 개선율을 살펴보면 다음과 같다.

〔표 6〕 수행 시간의 증가율과 목적 함수의 감소율

오차구간		해법 1	해법 2	해법 3	해법 4
5% ~ 2.5%	수행시간	14.4	9.79	4.95	5.81
	목 적 값	0.230	0.257	0.0109	0.0913
2.5% ~ 1%	수행시간	14.9	40.3	27.8	32.9
	목 적 값	0.729	0.508	0.407	0.615

〔표 6〕에 의하면 허용오차 5%에서 목적함수 값을 1%미만으로 개선시키는데 수행시간은 14%에서 5%정도까지 증가했고 허용오차 2.5%에서는 수행시간의 증가폭이 40%에서 15%까지 증가하게 되

었다. 즉, 허용오차 5%이상 부터는 목적함수 값을 개선시키기 위해 많은 수행 시간이 걸리게 되는 것을 알 수 있는데 참고로 허용오차 1%와 비교한 수행시간의 백분율과 목적함수 값의 성취도는 다음과 같다.

〔표 7〕 허용오차 1%를 기준으로한 수행 시간의 백분률과 목적 함수값의 성취도

허용오차		해법1	해법2	해법3	해법4
5 %	수행시간	76.095	64.905	73.409	71.087
	목적값	99.025	99.230	99.581	99.290
2.5 %	수행시간	87.059	71.259	77.045	75.213
	목적값	99.266	99.490	99.592	99.382

〔표 7〕에 의하면 허용오차 5%를 주었을때 이미 목적함수 값은 99%이상 성취하게 되는데 수행시간은 65%−76%정도선에서 멈추어져 허용오차 5%정도면 충분히 만족할 만한 해를 적정시간내에 구한다고 볼 수 있다.

5. 결론

본 논문에서는 소형전산기를 이용한 사료문제의 효율적인 해법을 찾아보고 해의 최적판정에 대한 종료기준을 제시하였다.

(1) 사료 문제에 대해 4가지 해법을 제시했으며 수행시간과 기억용량의 요구량에 대한 효율성 검정을 실시하였다. 실험 결과에 따르면 기저를 공유하여 쌍대단체법을 수행한 해법이 수행시간과 기억용량면에서 가장 우수한 결과가 나왔다.

(2) 연산 수행중에 수렴도 ν를 구하여

$$\nu \leq \varepsilon$$

이면 최종해로 받아들이는 종료기준을 제시했으며 ε는 0.05가 수행시간과 목적함수 값의 관계에서 적절한 것으로 분석되었다.

6. 감사의 글

이 논문의 미흡한 점을 바로잡아 준 심사위원들에게 심심한 감사를 드린다.

참 고 문 헌

1. 김태호, "블록대각구조를 지닌 2단계 확률계획법의 분해해법에 관한 연구", 서울대 공학석사학위논문, 1985
2. 박순달, "변수가 상, 하한을 가진 블록 대각 구조 문제의 분해원리에 관한 소고", 한국경영과학회지, 제 10권, 제 2호, 1985
3. 박순달, "D사의 사료배합 계산문제", 경영과학의 응용, 1984, 10월.
4. 박순달, 선형계획법 및 그 관련분야, 제 8장, 대영사, 1987
5. 양병학, "소형전산기에서 부분문제의 행렬계수가 동일한 분해해법의 효율성 제고에 관한 연구", 서울대 공학석사학위논문, 1987
6. I. Alder And A. Ulkucu, "On The Number of Iterations in Dantzig－Wolfe Decomposoition," in : D. M. Himmelblau, ed., *Decomposotion of Large Scale Problems,* North－holland, Amsterdam, 1973, p181－187
7. E. Balas, "An infeasibility－Pricing Decomposotion Method For Linear Programs", *O. R.* vol. 14, No. 5, 1966
8. M. S. Bazara and J. J. Jarvis, *Linear Programming and Network Flows,* John Wiley & Sons Inc., New York, 1977
9. E. M. L. Beale, P. A. B. Huges and R. E. Small, "Experiances in Using a Decomposition Program", *The Computer Journal,* Vol. 8, 1965, p13－18
10. R. M. Burton, W. W. Damon and D. W. Loughridge, "The Economics of Decomposotion : Resource Allocation vs Transfer Pricing", *Decision Sciences,* Vol. 5, No. 3, July, 1974
11. G. B. Dantzig and P. Wolfe, "Decomposition Pricing for Linear Programs", *O. R.,* Vol 8, 1960, p101－111
12. G. B. Dantzig, *Linear Programming and Extentions,* Princeton University Press, Princeton, N. J., 1963.
13. A. M. Geoffrion, "Elements of Large－Scale Mathematical Programming", *Mana. Sci.,* Vol. 16, No. 11, July, 1970
14. J. K. Ho and E. Loute, "An Advanced Implementation of the Dantzig－Wolfe Decomposition Algorithm for Linear Programming", *Math. Prog.* 20, 1981
15. J. K. Ho and E. Loute, "Computational Experiance with Advanced Implementation of Decomposition Algorithm for Linear Programming, *Math. Prog. 27,* 1983
16. L. S. Lasdon, *Optimizing Theory for Large System,* The Macmillan Company, New York, 1970
17. O. B. G. Madsen, "The Connection between Decomposition Algorithms and Optimal Degree of Decomposition", in : D. M. Himmelblau, ed., *Decomposition of Large Scale Problems,* North－holland, Amsterdam, 1973, p241－250
18. K. G. Murty, *Linear Programming,* John Wiley & Sons Inc., New York, 1983
19. C. H. Papadimitriou and K. Steigtz, *Combinatorial Optimization Algorithms and Complexity,* Prentice－Hall, inc., Englewood Cliffs, New Jersey, 1982

Journal of the Korean
MS Society
Vol. 12 No. 2, Dec 1987

—Research Paper—

A Goal Programming Model for Reverse Resource Allocation

N. K. Kwak*
Carol B. Diminnie**

ABSTRACT

This paper deals with the development and analysis of a quantitative model for reducing operating budgets of the academic units of a small private university, while reflecting the diverse goals of the university. A zero-one goal programming approach is used to design and implement the model for budgetary decision-making. The goal programming model can facilitate academic planning and decision-making by providing valuable information.

1. INTRODUCTION

The subject of reverse resource allocation (i.e., budget cut) continues to receive increasing emphasis in colleges and universities as the environment of resource limitation permeates many institutions of higher learning. Some major factors contributing to the financial predicament of the university are enrollment shifts from humanities to business, reassessment of the value of a college education, enrollment declines, and rapidly rising costs. In addition to these, both the federal and state governments have cut back their support to academia, thus swithching of priorities to more expedient social problems. Research grants have been reduced or eliminated by both foundations and private industry.

In recent years, there has been tendency to place much emphasis on "balancing the budget" at the expense of the academic goals of the institution. The crucial issue in the university administration is not just financial efficiency. The operational policy must be based on the combined philosophies of many diverse groups within the university. The very purpose, concept, and function of the administrators, faculty and students must be embodied in budgeting decisions.

The purpose of this paper is to develop and analyze a quantitative model for reducing the operating budgets of the academic units of a university while reflecting diverse goals of the university and allowing some degree of autonomy in decision-making with respect to budgetary matters. To achieve

* St. Louise University
** St. Bonaventure University

this objective, a goal programming (GP) model was developed based on the data obtained from a small private university in the State of New York to reflect the multiple competitive goals of the university. Specifically, the model employs a zero-one (0-1) goal programming process to reflect the indivisibility of the budget packages.

There are a number of goal programming studies which have appeared in the academic journals [2], [3], and [6]. However, this present study differs from the existing studies in that it addresses the problem of budget reductions rather than expansions. It is large in scope, encompassing every level of academic units within the university, incorporating input from administration at all levels of the academic hierarchy.

2. A GOAL PROGRAMMING MODEL

The GP model data is drived from a private co-educational university offering undergraduate programs in natural sciences, social sciences, humanities, business, and education, The undergraduate population is approximately 2,400 students. The GP model is primarily concerned with establishing optimal portfolios of budget-cut packages for each school in a university, and optimal packages of budget-cut items for each department within the schools.

Overview of the Model

The GP model in this paper involves three administrative levels : academic vice-president, deans of schools or divisions, and chairmen of departments. The model will allow decision maker at each level to select the portfolio of budget cut-backs which has the least negative impact on his multiple objectives.

The budget process is sequential in nature, using top-down goal decomposition approach. The academic vice-president initially determines goals of the university (i.e., the highest-level goals). These goals are generally quite broad in nature rather than very detailed. These goals are then decomposed into the goals of the deans of the schools. This decomposition results in the formulation of subgoals which contribute to the goals of the higher level. A dean may add one or more additional goals which are not in direct conflicts with those of the academic vice-president. Finally, the same process is repeated by chairmen at the department level.

After the goals are established at the department level, the information flow is reversed. Each department formulates budget-cut alternatives with its dollar savings amounts and establishes priority ranking of its goals. Department chairman assesses the adverse impact each alternative has on the development goals using paired-comparison and eigenvalue prioritization techniques [4], [5], [7]. Each department sets up its own GP model in which the zero-one (0-1) decision variables correspond to budget-cut alternatives. The model is solved for several possible budget level cuts, usually between three and five in number. Corresponding to the budget levels, optimal sets of budget-cut alternatives, called portfolios, are communicated upward to the dean of the school.

The same procedures are repeated by each dean based on the set of mutually exclusive portfloios received from his departments, along with any budget-cut options he may wish to add. Dean's budget-cut alternatives are then communicated upward to the academic vice-president. The academic vice-president uses the model to make a final selection of portfolios so as to best achieve the highest-level goals for the known total value of budget cuts and allocates the budget cuts.

The following assumptions are made for the model development.

1. Any goal within a hierarchical level can be represented by a linear combination of the budget-cut alternatives.
2. The goals of an academic unit can be identified.
3. It is possible to prioritize goals.
4. A budget item is either not cut at all, or is cut only at one of the levels specified in the model.
5. Administrators are able to formulate and rank budget-cut alternatives with respect to their expressed goals.

Mathematical Statement of the Model

The mathematical statement of the GP model is presented below.

Phase 1: Department Level

If there are s structural constraints, f functional constraints and r levels of g goal constraints, the model can be stated as follows:

$$\text{Minimize } V = P_1 \sum_{i=1}^{s} (d_i^{+} + d_i^{-}) + P_2 \sum_{i=s+1}^{s+f} (d_i^{+} + d_i^{-}) + P_3 (d_{s+f+1}^{+} + d_{s+f+1}^{-}) + \sum_{j=1}^{r} \sum_{i=s+f+2}^{s+f+r} P_j \alpha_{ij} d_i^{+} \quad (1)$$

Subject to

$$f_i(x) + d_i^{-} - d_i^{+} = a_i \qquad i=1, \cdots, s \text{ (structural constraints)} \quad (2)$$

$$h_i(x) + d_i^{-} - d_i^{+} = a_i \qquad i=s+1, \cdots, s+f \text{ (functional constraints)} \quad (3)$$

$$\sum_{j=1}^{n} F(X_j) X_j + d_i^{-} - d_i^{+} = b \qquad i=s+f+1 \text{ (budget-cut constraint)} \quad (4)$$

$$\sum_{j=1}^{n} W(X_i) X_j - d_i^{+} = 0 \qquad i=s+f+2, \cdots, s+f+r \text{ (other goal constraints)} \quad (5)$$

and

$$X_j = \begin{cases} 1 & \text{if alternative is chosen} \\ 0 & \text{otherwise} \end{cases}$$

$d_i^{-}, d_i^{+} \geq 0.$

where α_i = weights within a priority level

$f_i(X)$, $h_i(X)$ = linear combinations of decision variables corresponding to the structural and functional characteristics of the problem

$F(X_j)$ = the dollar savings of budget alternative X_j

X_j = a department budget-cut decision variable, $j=1,\cdots, n$

$W_i(X_j)$ = the relative impact of X_j on goal i

a_i = contsants (usually 0 or 1) determined uniquely by the specific situation being modeled

b = a budget-cut parameter. The model will be solved for three to five different values of b

d^+ = positive deviation from goal

d^- = negative deviation from goal

In the objective function (1), the highest priority, P_1, is placed on structural constraints (2) with equal weighting, because these constraints must be satisfied in order to assure a feasible solution. The functional constraints (3) are given the second priority level, P_2, with equal weights in the objective function. The decision variables in (3) correspond to interrelated activities and are mutually dependent upon one another. The budget-cut constraint (4) is placed at the third priority level, P_3. This is the first goal constraint specifying the dollar amount of budget-cut requirement to be satisfied as closely as possible. In other goal constraints (5), $W(X)$ is a linear function of decision variables with weights derived from the eigenvalue technique. The right-hand side value of 0 represents the goal's least negative impact when all variables are zero. These goal constraints are prioritized by the department chairmen with P_4, through P_n, priorities in accordance with their importance.

Phase 2: Division Level

The model is similar to that of Phase 1, substituting Y for X and B_j for $F(X_j)$, since the Y_i will yield savings of B_j.

Phase 3: University Level

The model is similar to that of Phase 2, substituting Z for Y. This moded is solved only once, since the value of B here is the total amount of budget cuts.

3. A ZERO-ONE GP MODEL FOR DEPARTMENT A1

There are 26 GP models in this study: 20 departmental, 5 divisional, and 1 university models. For illustration, a 0-1 GP model for Department A1 is only presented below as an example.

$$\text{Minimize } V=P_1\left(\sum_{i=1}^{6} d_{i,1}^{-}+d_{i,1}^{+}\right)+P_2(d_{7,1}^{-}+d_{7,1}^{+})+P_3 d_{8,1}^{+}+P_4 d_{9,1}^{+}$$

$$\begin{aligned}\text{subject to } & X_1(1)+X_1(2)+W_1+d_{1,1}^{-}-d_{1,1}^{+}=1\\ & X_3(1)+X_3(2)+W_2+d_{2,1}^{-}-d_{2,1}^{+}=1\\ & X_4(1)+X_4(2)+W_3+d_{3,1}^{-}-d_{3,1}^{+}=1\\ & X_5(1)+X_5(2)+W_4+d_{4,1}^{-}-d_{4,1}^{+}=1\\ & X_6(1)+X_6(2)+W_5+d_{5,1}^{-}-d_{5,1}^{+}=1\end{aligned}$$

$$X_7(1)+X_7(2)+W_6+d_{6,1}^{-}-d_{6,1}^{+}=1$$
$$300X_1(1)+1300X_1(2)+500X_2(1)+200X_3(1)+650X_3(2)+500X_6(1)+1000X_6(2)$$
$$+500X_7(1)+2000X_7(2)+d_{7,1}^{-}-d_{7,1}^{+}=\text{b}$$
$$35X_1(1)+63X_1(2)+49X_2(1)+8X_3(1)+211X_3(2)+144X_4(1)+206X_4+127X_5(1)$$
$$+20X_5(2)+18X_6(1)+54X_6(2)+50X_7(1)+44X_7(2)-d_{8,1}^{+}=0$$
$$129X_1(1)+64X_1(2)+62X_2(1)+11X_3(1)+15X_3(2)+16X_4(1)+18X_4(2)+18X_5(1)$$
$$+214X_5(2)+169X_2(1)+52X_6(2)+47X_7(1)+185X_7(2)-d_{9,1}^{+}=0$$

and

$$X_j=\begin{cases}1 \text{ if alternative is chosen}\\0 \text{ otherwise}\end{cases}$$

$$d_1^{+},\ d_i^{-}\geq 0$$

The model was solved using a modified 0-1 GP package [8]. For each of the 20 departments at the university studied, a separate portfolio of budget reductions was established corresponding to a 10%, a 15%, or a 20% cut in that department's operating expense budget. The five division models were run corresponding to 5%, 10%, 15%, or 20% reduction in that division's budget. Finally, at the highest academic level, the model was run once corresponding to an overall reduction of 10%. Thus, a total of 81 runs were made on a Prime 550 Computer. A CPU time for these runs required almost nine minutes. All of the structural, functional, and budget-cut goals were completely satisfied on 81 runs. The academic unit goals were not completely satisfied on any of the runs.

Department A1's model has 13 0-1 decision variables, six 0-1 artificial variables, nine constraints, and four priority levels. The decision variables and their respective dollar savings are presented in Table 1.

TABLE 1

Variable	Description	Savings($)
$X_1(1)$	Student work program	300
$X_1(2)$		1,300
$X_2(1)$	Postage	500
$X_3(1)$	Memberships	200
$X_3(2)$		650
$X_4(1)$	Travel	500
$X_4(2)$		1,000
$X_5(1)$	Lectures	700
$X_5(2)$		1,100
$X_6(1)$	Office supplies	500
$X_6(2)$		1,000
$X_7(1)$	Library, books	500
$X_7(2)$		2,000

As the table indicates, seven budget categories are included in this model. Two reductions are made in each of six of these categories, while one category (postage) has only one reduction. An artificial variable is necessary each time when two or more reduction, in the same category occur. Consequently, six artificial variables are required.

Six of the nine constraints are of a structural nature, and hence appear at the first priority level. These constraints arise due to the mutually exclusiveness of variables $X_i(1)$ and $X_i(2)$, $i=1$, 3, 4, 5, 6, 7.

In order to use a standard 0-1 goal programming computer code, these structural constraints must be converted to goal format. This can be done by placing all structural constraints on the first priority level and rewriting the constraints in the form

$$\sum_{i=i}^{C} X_i(j)+W_i+\bar{d_i}-\overset{+}{d_i}=1.$$

W_i is an artificial zero-one variable which is necessary to allow at most one of the $X_i(J)$ to be chosen at a particular i-level.

The seventh constraint is the budget constraint. The coefficient of each variable indicates the exact amount of the proposed reduction in that category. The right-hand-side constant b will vary for each of three runs of the program, corresponding to a 10%, 15%, or 20% reduction in the department's budget. For this department, the three values of b, corresponding to three separate runs, are \$1,750, \$2,625, and \$3,500. The budget constraint appears at the second priority level.

Constraints 8 and 9 correspond to the two goals of the department. These constraints appear at the third and fourth priority levels, respectively. The coefficients of the variables in these equations are the weights calculated by the eigenvalue prioritization technique.

A comparison is made between the actual percentage allocation of the university's academic budget under existing budgetary practice and the percentage allocations resulting from the model when a 10% reduction in the operating budget is required. It is presented in Table 2. While there does not appear to be great differences in the distribution of funds among the various categories in this department, a word of caution should be made here. In the current budget process, chairmen indicated that it was difficult to increase any budget item substantially even if compensating decreases were made in other categories. If this tactic were taken, the results were that the items reduced were left at those lower levels and that category which was increased was cut back to its original budgeted vlaue. Chairmen thus found it advantageous to try to increase all budget items, even when not necessary. A redistribution of resources was difficult. Under the model, the chairman is able to decide the best (in terms of satisfying departmental goals) distribution of alloted funds among all budget line-items.

TABLE 2: Comparison of Actual and Model Budget Allocations for Department A1

Line-item	Actual Allocation (%)	Model Allocation(%)
Student Work	23.99	23.79
Postage	2.49	2.73
Memberships	3.13	2.38
Travel	10.42	13.32
Lectures	3.74	3.57
Repair of Equipment	1.25	0.00
Research	1.87	1.19
Supplies	28.15	26.85
Library	24.95	26.17

4. USER EVALUATION OF THE MODEL

The 0-1 GP model has been tested and evaluated by various administrators. Twelve chairmen favored using the model to determine operationg budgets, with only two votes against it. The majority of chairmen were uncertain as to the model's use for developing alternative budget allocations (7 yes, 10 unceratin and 3 no). Eleven chairmen would use the model to defend their budget recommendations, with 6 uncertain and 3 voting no. Most chairmen did not complain of excessive work or time involvement with the model (13 no, 4 yes, 3 uncertain). Most chairmen also provided favorable comments on the model use.

The deans of the five divisions were asked for their reactions to the results of the model. In all cases, favorable comments were made. These administrators realized that the fund should be allocated to those areas where most students were registered or where most of the research was being done. They all believed the model allocation did that.

The Academic Vice-President was very enthusiastic about the model results. He is presently using the results as a planning tool in next year's budget allocation.

5. SUMMARY

This paper presented a 0-1 GP model for optimal reverse allocation of budgetary resources at a private university. The model was developed and tested for budgetary decision-making. Tests of the model to an actual decision situation show that the model is indeed promising, thus bridging the gap between theory and practice.

A survey of users revealed a definite preference for the model results over the existing budgetary procedure. While it was developed specifically for the opearting budgets of academic units within a university, the model can be easily extended to and used by any organization experiencing a similar problem. It can be a viable decision-making tool for short-term financial planning.

REFERENCES

1. Diminnie, Carol B. and N. K. Kwak, "A Hierarchical Goal Programming Approach to Reverse Resource Allocation in Institutions of Higher Learning," *Journal of the Operational Research Society,* 37, 59-66 (1986).
2. Keown, A. J., B. W. Taylor, and J. M. Pinkerton, "Multiple Objective Capital Budgeting Within the University," *Computers & Operations Research,* 8, 59-70 (1981).
3. Lee, S. M. and E. R. Clayton, "A Goal Programming Model for Academic Resource Allocation," *Management Science,* 18, B395～408 (1972).
4. Saaty, T. L., *The Analytic Hierarchy Process,* McGraw-Hill Book Co., New York, 1980.
5. Saaty, T. L. and P. C. Rogers, "Higher Education in the United States (1985－2000) : Scenario Construction Using a Hierarchical Framework with Eigenvalue Weighting," *Socio-economic Planning Sciences,* 10, 251－263 (1976).
6. Schroeder, R. G., "Resource Planning in University Management by Goal Programming," *Operations Research,* 22,700－710 (1974).
7. Wilkinson, J. H., *The Algebraic Eigenvalue Problem,* Clarendon Press, Oxford, 1965.
8. *Zero-One Goal Programming Computer Program,* School of Business Administration, University of Nebraska-Lincoln, Lincoln, Nebraska, 1983.