韓國經營科學會誌
第19卷 第1號
1994年 4月

目標計劃法 初期解의 새로운 節次에 관한 연구†

박승헌* · 최재봉**

A New Procedure for the Initial Solution of Goal Programming

Seung Hun Park* and Jae Bong Choi**

Abstract

This study proposes a new procedure to find an initial solution to reduce the number of iterations of goal programming. The process of computing an initial solution is divided into two steps in this study.

Decision variables which satisfy feasibility using Gaussian eliminations construct an initial solution reducing the iterations in the first step. It uses LHS as a tool that decision variables construct an initial solution. The initial solution which is constructed by the first step computes the updated coefficient of the objective function in the second step. If the solution does not satisfy the optimality, the optimal solution using the Modified Simplex Method is sought.

The developed method doesn't reduce the overall computing time of goal programming problems, because time is more required for the process of constructing an initial solution. But The result of this study shows that the proposed procedure can reduce the large number of iterations in the first step effectively.

† 본 연구는 인하대학교 1992년도 연구비 지원으로 수행되었음.
* 인하대학교 산업공학과
** 세경전문대학 [illegible]

1. 서 론

목표계획법(goal programming)은 수리 모형화된 의사결정 방법 중 비교적 적용이 용이하다. 또한, 목표계획 문제의 해결방법은 수정된 심플렉스법(revised simplex method)을 사용한다. 이것은 역행렬등을 구하는 지루한 대수적인 계산과정이다.

기존의 목표계획법으로는 수정된 심플렉스 방법, LU 분해(lower upper decomposition)기법과 성김성(sparsity)기법등이 있다. 이것들은 대형 목표계획법 문제의 해결을 컴퓨터 계산시간 단축과 동시에 최적해에의 접근이 가능토록 하였다. 그러나, 이것들은 초기해(initial solution)를 찾는 데 있어 수정된 심플렉스 방법에 의존하기 때문에 역행렬을 구하는 지루한 대수과정을 거침으로써 결과적으로 전체 계산시간이 늘어나게 된다.

본 연구는 보다 효율적으로 초기해를 구하는 알고리즘을 제시함으로써 대수적 계산의 반복과정(iteration)을 줄이고 결과로써 전체 계산시간에 미치는 영향을 파악하는데 그 목적을 두었다.

새로운 기법을 구성하는 주된 방법은 가우스소거법에 의하여 제약조건을 구성하는 행열 A(여기서 행열 A는 제약조건을 구성하는 행열을 의미하며, 모든 제약조건에 해당하는 m개의 행과 결정변수(decision variable)에 해당하는 n개의 열로 되어 있다)를 재구성하고 이렇게 재구성된 것을 이용하여 기저변수에 편차변수 대신에 결정변수가 되도록 함으로써 최적해에 도달하는 데 있어서 반복과정을 줄일 수 있도록 하였다. 본 연구의 기법과 수정된 심플렉스법의 비교는 C언어로 프로그램하여 퍼스널 컴퓨터(IBM PC)상에서 행하였다.

2. 연구 방법

2.1 접근 방법

대부분의 실제적인 문제는 매우 많은 제약조건을 가지기 때문에 문제의 규모가 커지기 마련이다. 대규모 문제를 기존의 방법으로 풀기에는 계산상 많은 부담이 따른다. 이와같은 문제를 해결하는 데 있어서 발견적(heuristic)해법이 요구되며 또한 효과적인 경우가 많다. 본 연구는 단순한 알고리즘을 구성하고 가능한 계산 횟수를 줄이는 방법을 아래와 같은 착상(idea)에 기인하여 시도한다.

제약조건을 구성하는 행열 A의 원소가 영을 매우 많이 가지고(예로서 대규모의 실제적 문제들) 있으며, 또한 행열 A의 원소가 많은 부분이 비음수의 경우, 목표계획법 문제는 상당수의 편차변수 대신에 결정변수가 기저변수로 될 수 있다. 이와 같은 접근 방법은 많은 수의 편차변수 대신에 결정변수가 실행가능성을 유지하면서 기저변수로 교체가능하다면 최적해에 도달하는 데 있어서 계산상의 많은 반복과정을 줄일 수 있을 것이다.

접근방법은 2단계로 나누어 시도한다. 단계 1에서는 편차변수대신에 교체된 결정변수가 제약조건을 충족시키면서 새로운 가능해를 찾도록 한다. 단계 2에서는 새로이 구성된 가능해가 최적해인가의 여부를 확인한다. 만일 최적해가 아니라면 수정된 심플렉스법으로 최적해를 구한다.

2.2 기호의 정의

(1) 집합 J_i : 행열 A의 원소를 a_{ij}로 표현할 때, 행 i의 원소가 영이 아닌 모든 원소의 첨자 j의 집합(j=1,2,…,n).
예) 행 i의 벡터를 [1, 0, 0, 1, 1]이라고 하면 J_i={1, 4, 5}.

(2) 집합 J^+ : 행열 A의 열 벡터 a_j(j=1,2,…, n)중에서 새로운 가능해를 형성하면서 추가되는 벡터 a_j가 있을 때 첨자 j의 집합.
예) 처음에 편차변수 대신에 기저변수가 된 것이 x_1이라고 한다면 J^+={1}이 되고, 다음으로 기저변수가 된 것이 x_2라고 한다면 J^+={1,2}가 된다.

(3) 숫자 N_i : 행열 A의 i th 행 벡터 a_i (i=1,2,…,n)를 [a_{i1},a_{i2}…a_{in}]라고 할 때, i th 행 벡터 a_i의 원소중에서 영이 아닌 원소의 갯수.
예) 행 i의 벡터를 [1, 0, 0, 1, 1]이라고 하면 N_i=3.

(4) i th LHS : 편차변수 대신에 기저변수를 형성하면서 새로이 추가되는 결정 변수의 가능성을 검사. LHS를 구하는 방법은, 행열 A의 i th행의 제약조건을 구성하는 모든 결정변수 x중에서 첨자j가 집합 J^+에 포함되어 있는 결정변수 x_j의 RHS값을 i th행에 대입시킨 값이 된다. j가 집합 J^+에 포함되지 않는 변수의 대입하는 값은 영이 된다. 왜냐하면 기저변수가 아니기 때문이다.
예)i th행이(x_1+x_2+x_3+x_4+x_5+d^- −d^+) 라고 하고 집합 J^+={1, 2, 4}라고 하며 기저변수 x_1의 RHS=10, x_2의 RHS=20, x_4의 RHS=15라고 한다면 i th LHS=45가 된다.

(5) 기호 "V" : 결정변수 x_j(j=1,2,…,n)가 기저변수가 되었음을 표시(〈표 2〉상단참조).

(6) 기호 "X" : 결정변수 x_j(j=1,2,…,n)가 비기저변수임을 표시(〈표 2〉 상단 참조).

2.3 단계 1

단계 1은 〈표 1〉, 〈표 2〉로 구성된다. 〈표 1〉은 주어진 목표 계획법 문제가 있을때, 그 제약조건을 그대로 표로 옮긴 것에 불과하다. 이때, 행열 A 각각의 행에 해당하는 N_i(i=1,2,…, m)를 구하여 좌측의 비영(non−zero)원소수의 칸에 적어 넣는다. 그 다음 가장 작은 값을 갖는 N_i에 해당하는 행 i의 LHS값을 구하여 도입변수를 찾기 시작한다. 이러한 절차로 기저변수를 찾으면 〈표 1〉의 상단에 "V"표시를 하고 행 계산(row operation)을 한 후에 〈표 2〉에 옮겨 적는다. 구체적인 내용은 다음과 같다.

일반적인 목표 계획법 문제를 다음과 같이 표시한다.

$$\text{Min } Z = P(C^{-}d^{-} + C^{+}d^{+}) \qquad (2-1)$$

$$\text{s.t} \quad AX + d^{-} - d^{+} = b$$

$$x,\ d^{-},\ d^{+} \geq 0$$

$$A=\begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1r} & a_{1r+1} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2r} & a_{2r+1} & \cdots & a_{2n} \\ \cdot & \cdot & & \cdot & \cdot & & \cdot \\ \cdot & \cdot & & \cdot & \cdot & & \cdot \\ \cdot & \cdot & & \cdot & \cdot & & \cdot \\ a_{r1} & a_{r2} & \cdots & a_{rr} & a_{rr+1} & \cdots & a_{rn} \\ a_{r+11} & a_{r+12} & \cdots & a_{r+1r} & a_{r+1r+1} & \cdots & a_{r+1n} \\ \cdot & \cdot & & \cdot & \cdot & & \cdot \\ \cdot & \cdot & & \cdot & \cdot & & \cdot \\ \cdot & \cdot & & \cdot & \cdot & & \cdot \\ a_{m1} & a_{m2} & \cdots & a_{mr} & a_{mr+1} & \cdots & a_{mn} \end{bmatrix} \qquad (2-2)$$

〈표 1〉 원 문제의 행열

비영 원소수	LHS	기저 변수	x_1	x_2	…	x_r	x_{r+1}	…	x_n	d_1^-	…	d_m^-	d_1^+	…	d_m^+	RHS
N_1'	LN_1		a_{11}'	0	…	0	a_{1r+1}'	…	a_{1n}'							b_1'
N_2'	L_2'		a_{21}'	a_{22}'	…	0	a_{2r+1}'	…	a_{2n}'							b_2'
·	·		·	·		·	·		·							·
·	·		·	·		·	·		·							·
·	·		·	·		·	·		·							·
N_r'	L_r		a_{r1}'	a_{r2}'	…	a_{rr}'	a_{rr+1}'	…	a_{rn}'		I_m		$-I_m$			b_r'
N_{r+1}'	L_{r+1}		a_{r+11}'	a_{r+12}'	…	a_{r+1r}'	a_{r+1r+1}'	…	a_{r+1n}'							b_{r+1}'
·	·		·	·		·	·		·							·
·	·		·	·		·	·		·							·
·	·		·	·		·	·		·							·
N_m'	L_m		a_{m1}'	a_{m2}'	…	a_{mr}'	a_{mr+1}'	…	a_{mn}'							b_m'

· $N_1' \leq N_2' \leq \cdots \leq N_r'$ · $a_{ij}' = 0 \ (i < j \leq r)$

식(2-2)에서 $N_1, N_2, \cdots, N_m$을 구하고 그것들의 값을 오름차순으로 정리한 것을 $N_1', N_2', \cdots, N_m'$라 하여 순서대로 행열 A의 행을 재배열한다. 다음으로 각각의 열을 상호교환하여 〈표 1〉좌상귀쪽의 부분행열을 하부 삼각(lower Triangular)행열이 되도록 식(2-1)을 새로이 구성하면 다음과 같은 식이 된다.

$$\text{Min } Z = P(C^- d^- + C^+ d^+) \qquad (2-3)$$
$$\text{s.t} \quad A'X + d^- - d^+ = b'$$
$$x, d^-, d^+ \geq 0$$

$$A' = \begin{bmatrix} a_{11}' & 0 & \cdots & 0 & a_{1r+1}' & \cdots & a_{1n}' \\ a_{21}' & a_{22}' & \cdots & 0 & a_{2r+1}' & \cdots & a_{2n}' \\ \cdot & \cdot & & \cdot & \cdot & & \cdot \\ \cdot & \cdot & & \cdot & \cdot & & \cdot \\ \cdot & \cdot & & \cdot & \cdot & & \cdot \\ a_{r1}' & a_{r2}' & \cdots & a_{rr}' & a_{rr+1}' & \cdots & a_{rn}' \\ a_{r+11}' & a_{r+12}' & \cdots & a_{r+1r}' & a_{r+1r+1}' & \cdots & a_{r+1n}' \\ \cdot & \cdot & & \cdot & \cdot & & \cdot \\ \cdot & \cdot & & \cdot & \cdot & & \cdot \\ \cdot & \cdot & & \cdot & \cdot & & \cdot \\ a_{m1}' & a_{m2}' & \cdots & a_{mr}' & a_{mr+1}' & \cdots & a_{mn}' \end{bmatrix} \qquad (2-4)$$

여기서 첨자 "r"은 다음에 설명할 새로운 방법에 의해 결정변수가 기저변수로 되는 임의의 수를 표시한 것이다. 식(2-3)을 표로 나타내면 〈표 1〉과 같다.

〈표 2〉는 〈표 1〉의 부분행열을 단위행열이 되도록 행 계산한 것이다.

〈표 2〉는 결정변수가 기저변수가 되는것을 시각적으로 보이기 위하여 좌상귀쪽의 부분행열이 단위행열이 되도록 재구성하였다. 그 계산 절차는 다음과 같다.

〈표 1〉 → 〈표 2〉 계산 방법

〈표 1〉의 행을 row(i), 〈표 2〉의 행을 row'(i)라 하고 L_i는 i th 행의 LHS로 표시한다. 첨자 "e"는 열 e에 있는 변수가 기저변수로 됨을 표시하고 a_{re}는 樞軸元素(pivot element)이다. 행에 관한 첨자(1, 2, …, m)은 $(N_1' \leq N_2' \leq \cdots \leq N_m')$의 순서를 뜻한다.

〈표 2〉 재구성에 따른 계산 결과

비영 원소수	LHS	기저 변수	x_1	x_2	…	x_r	x_r	…	x_n	d_1^-	…	d_m^-	d_1^+	…	d_m^+	RHS
			V	V	…	V	X	…	X							
		x_1	1	0	…	0	$\bar{a}_{1r+1}$	…	$\bar{a}_{1n}$							$\bar{b}_1$
		x_2	0	1	…	0	$\bar{a}_{2r+1}$	…	$\bar{a}_{2n}$							$\bar{b}_2$
		·	·	·		·	·		·							·
		·	·	·		·	·		·							·
		·	·	·		·	·		·							·
		x_r	0	0	…	1	$\bar{a}_{rr+1}$	…	$\bar{a}_{rn}$		B^{-1}			$-B^{-1}$		$\bar{b}_r$
		d_{r+1}	0	0	…	0	$\bar{a}_{r+1r+1}$	…	$\bar{a}_{r+1n}$							$\bar{b}_{r+1}$
		·	·	·		·	·		·							·
		·	·	·		·	·		·							·
		·	·	·		·	·		·							·
		d_m	0	0	…	0	$\bar{a}_{mr+1}$	…	$\bar{a}_{mn}$							$\bar{b}_m$

· $\bar{a}_{ii} = 1$ $(i=1, 2, \cdots, r)$, $a_{ij} = 0$ $(i \neq j \leq r)$

(1) $L_1=0$, 집합 $J^+=\varphi$ 경우.

① (집합 J_1－집합 J^+)$\neq\varphi$ 일 때, $a_{1e}=\{a_{1j}$: $e=\mathrm{Min}_{j\in J_1}\{j\}\}$

(집합 J_1－집합 J^+)$=\varphi$ 일 때, $a_{1e}=1$(d_1^- 가 기저변수가 된다).

② $row'(1)=row(1)/a_{1e}$

③ 〈표 2〉에서 도입변수에 해당하는 열 e에 "V"표시.

④ 집합 J^+의 원소에 e를 추가. 단, $(e=1, 2, \cdots, n)$.

⑤ (집합 J_1－집합 J^+)$\neq\varphi$ → (집합 J_1－집합 J^+) 원소 j의 열에 "X" 표시.

⑥ $i = 2$.

(2) $L_i = L_i=\sum_{j\in J^+}(a_{ij}\times X_j)$ 경우.

① (집합 J_i－집합 J^+) $\neq \varphi$ 일 때, $a_{ie}=\{a_{ij}$: $e=\mathrm{Min}_{j\in J_i}\{j\}\}$

(집합 J_i－집합 J^+) $= \varphi$ 일 때,

$L_i \leq$ i th RHS의 경우 d_i^- 가 기저변수가 된다. $a_{ie} = 1$

$L_i \geq$ i th RHS의 경우 d_i^+가 기저변수가 된다. $a_{ie} = -1$

② $row'(i)=\{row(i)-\sum_{j\in J^+}(a_{ij}\times row'(j))\}/a_{ie}$

③ 〈표 2〉에서 열 e에 "V"표시를 한다.

④ 집합 J^+의 원소에 e를 추가. 단, $(e=1,2,\cdots,n)$.

⑤ (집합 J_i－집합 J^+) $\neq \varphi$ → (집합 J_i－집합 J^+) 원소 j의 열에 "X" 표시.

⑥ $i=i+1$. i가 m이 될 때까지 (2)를 반복한다.

여기서 i th 행의 LHS와 RHS간의 차이가 존재하는 것은 현재의 기저변수 대신 비기저변수가 기저변수로 교체될 수 있는 가능성이 있음을 의미한다. 만일 LHS가 RHS보다 크다면

〈표 3〉 수정된 심플렉스 법에 의한 적용 결과

									C_1-	…	C_m-	C_1+	…	C_m+		
C_J	기저변수	x_1	x_2	…	x_r	x_{r+1}	…	x_n	d_{1-}	…	d_{m-}	d_{1+}	…	d_{m+}	RHS
	x_1	1	0	…	0	$\bar{a}_{1r+1}$	…	$\bar{a}_{1n}$							$\bar{b}_1$
	x_2	0	1	…	0	$\bar{a}_{2r+1}$	…	$\bar{a}_{2n}$							$\bar{b}_2$
	.	.	.		.	.		.							.
	.	.	.		.	.		.							.
	.	.	.		.	.		.							.
	x_r	0	0	…	1	$\bar{a}_{rr+1}$	…	$\bar{a}_{rn}$		B^{-1}			$-B^{-1}$		$\bar{b}_r$
C_{r+1}	d_{r+1}	0	0	…	0	$\bar{a}_{r+1r-1}$	…	$\bar{a}_{r+1n}$							$\bar{b}_{r+1}$
.	.	.	.		.	.		.							.
.	.	.	.		.	.		.							.
.	.	.	.		.	.		.							.
C_m	d_m	0	0	…	0	$\bar{a}_{mr+1}$	…	$\bar{a}_{mn}$							$\bar{b}_m$
	P_k	0	0	…	0										
	.	.	.		.										
	.	.	.		.		$C_BB^{-1}a_j$		$C_BB^{-1}-C_j$			$-C_BB^{-1}-C_{j+}$			$C_BB^{-1}b$
	.	.	.		.										
	P_1	0	0	…	0										

i th행 도입변수의 계수가 음수여야 한다. 그 반대의 경우, 계수가 양수이여야 한다. 이러한 방법으로 도입변수를 결정하면 도입변수는 항상 음수가 되지 않는다. 따라서 위의 계산절차는 항상 가능해를 유지하면서 초기해를 탐색할 수 있다.

2.4 단계 2

단겨 1에서 생성한 〈표 2〉를 이용하여 그것이 최적해인가를 확인한다. 최적해가 아니라면 수정된 심플렉스법을 이용하여 최적해를 구한다.

〈표 3〉의 C_j 열의 $[C_{r+1},\cdots, C_m]$은 편차변수 $[d_{r+1},\cdots, d_m]$에 해당하는 목적값을 의미하고 $[P_k,\cdots, P_1]$은 우선순위가 낮은 순위부터 순서대로 되어 있음을 의미한다.

3. 수치 예 및 비교 분석

3.1 반복 과정의 수

단계 1의 〈표 2〉와 같이 r개의 결정변수가 편차변수 대신에 기저변수로 되는 경우, 수정된 심플렉스법은 각 반복과정마다 편차변수 대신에 기저변수가 될 수 있는 가능한 결정변수를 하나씩 대체시켜 준다. 그 예로서 〈표 2〉와 같이 기저변수에 결정변수를 r개를 포함한다면

〈표 4〉 예제의 형태

예제번호	행의 수			열의 수			성 김 성
	제약조건	우선순위	합 계	결정변수	편차변수	합 계	
1	4	4	8	2	6	8	62.5(%)
2	4	4	8	2	8	10	67.5(%)
3	8	5	13	3	16	19	82.9(%)
4	17	5	22	8	34	42	89.6(%)
5	25	8	33	12	50	62	93.4(%)
6	31	10	41	12	62	74	94.8(%)
7	38	13	51	15	76	91	95.3(%)

수정된 심플렉스법은 적어도 r번의 반복과정이 일어나게 된다.

따라서, 본 연구의 알고리즘은 r이 크면 클수록 수정된 심플렉스법에 비하여 반복과정을 많이 줄일 수 있는 가능성이 커진다.

〈표 4〉에서와 같은 예제들을 실행시켜 심플렉스법과 새로운 방법의 반복과정의 수를 비교하였다. 그 결과는 〈표 5〉와 같다.

〈표 5〉 반복과정수의 비교

예제번호	반복과정의 수	
	수정된 심플렉스법	새로운 방법
1	3	2
2	5	4
3	10	6
4	15	5
5	25	4
6	19	13
7	41	13

〈표 5〉의 결과로 새로운 방법이 수정된 심플렉스법보다 전반적으로 반복과정의 수가 적어지는 것을 확인할 수 있다. 또한, 새로운 방법은 제약조건이 많아지거나 성김성이 커질수록 수정된 심플렉스법에 월등하게 반복과정의 수가 줄어드는 경향도 확인되었다.

3.2 계산 시간

새로운 방법은 결정변수 r이 많아질수록 반복과정의 수는 줄일 수 있으나 반면 〈표 2〉를 재구성하여 계산하는데 걸리는 시간은 늘어나게 된다. 여기에서는 이렇게 늘어난 시간이 전체 계산시간에 미치는 영향을 검토한다.

사칙 연산 횟수는 수정된 심플렉스법을 r번 반복한 것에 기준을 두었고 새로운 방법은 〈표 3〉까지 계산한 것에 기준을 두었다. 이 두 방법의 사칙연산 계산과정은 번잡함을 피하기 위해 본 지면에서는 생략하였다.

〈표 4〉의 수치 예를 사용한 두 방법의 계산 시간 결과는 〈표 6〉과 같다.

새로운 방법은 단계 1에서 매우 많은 시간을 소비하였다. 그 이유는 반복과정의 수를 줄일 수 있는 임의의 초기해를 산출하기 위하여 〈표 2〉를 계산하는 시간이 많이 소요되기 때문이다.

〈표 6〉 계산시간의 비교

예제 번호	계산 시간(초)			
	수정된 심플렉스법	새로운 방법		
		합 계	단 계 1	단 계 2
1	1	2	1	1
2	1	2	1	1
3	3	4	2	2
4	13	14	8	6
5	47	51	33	18
6	53	76	45	31
7	157	188	121	67

특히 LHS 계산시간과 제약조건의 행 계산시간에 많은 시간이 소요되었기 때문에 반복과정의 수는 줄었음에도 불구하고 전체 계산시간이 줄어들지 않았음을 확인할 수 있다.

3.3 예 제

여기에서는 앞의 2장에서 설명한 본 연구의 새로운 방법에 대한 계산절차를 다음과 간단한 예제를 통해 구체적으로 표현하였다.

$$\text{Min } Z = p_1 d_{1-} + p_2 d_{4+} + 5p_3 d_{2-} + 3p_3 d_3 + p_4 d_{1+}$$

$$\begin{aligned} \text{s.t} \quad & x_1 + x_2 + d_{1-} + d_{1+} = 80 \\ & x_1 + d_2 + d_{2+} = 70 \\ & x_2 + d_{3-} + d_{3+} = 45 \\ & x_1 - x_3 + d_{4-} + d_{4+} = 10 \\ & x_1, x_2, x_3, d_{i-}, d_{i+} \geq 0 \ (i=1,2,3,4) \end{aligned}$$

(1) 단계 1

$N_1=2$, $N_2=1$, $N_3=1$, $N_4=2$

〈표 7〉 표 1을 계산한 결과

비영원소수	LHS	기저변수	x_1	x_2	x_3	d_{1-}	d_{2-}	d_{3-}	d_{4-}	d_{1+}	d_{2+}	d_{3+}	d_{4+}	RHS
2	115		1	1		1				-1				80
1	0		1				1				-1			70
1	0			1				1				1		45
2	70		1		-1				1				1	10

$L2=0$

$row'(2)=row(2)/1$

$L3=0\times 70$

$row'(3)=\{row(3)-0\times row'(2)\}/1$

$L1=1\times 70+1\times 45$

$row'(1)=\{row(1)-1\times row'(2)-1\times row'(3)\}/-1$

$L4=1\times 70+0\times 45$

〈표 8〉 표 2를 계산한 결과

V V V

비영원소수	LHS	기저변수	x_1	x	x	d	d	d	d	d	d	d	d	RHS
		d_{1-}				1	1	1		1	1	1		35
		x_1	1				1				-1			70
		x		1				1				1		45
		x			1		1		1		1		-1	60

(2) 단계 2

〈표 9〉 단계 2를 계산한 결과

P_1 $5P$ $3P$ P P

C_j	기저변수	x_1	x_2	x	d_1	d	d	d	d	d	d	d_{1-}	RHS
P	d_{1+}				-1	1	1		1	-1	1		35
	x_1	1				1				-1			70
	x		1				1				1		45
	x			1		1		-1		1		1	60
	P				-1	1	1			-1	-1		35
	P					5	-3						
	P											-1	
	P				-1								

4. 결론 및 향후 연구 과제

1970년대 말부터 관심이 집중된 목표계획법은 다수의 상충되는 목적을 해결하는 실제적인 의사결정 도구로서 많은 역할을 해왔다. 그러나 실질적인 목표계획법 문제는 규모가 클 뿐만 아니라 영을 많이 포함하고 있는 성김성 행렬을 가지고 있다. 지금까지 많은 연구가 있어왔듯이 이런 문제일수록 계산노력을 줄일 수 있도록 방법을 강구하여야 할 것이다.

본 연구에서는 초기해를 구하는 새로운 절차를 제시하여 기존의 수정된 심플렉스 방법보다 계산 반복과정의 수를 줄일 수 있었다. 특히, 제약조건의 수가 많거나 성김성이 큰 의사결정 문제일수록 반복과정의 수 면에서는 보다 효율적이었다. 그러나, 계산 반복과정의 수가 줄어드는 것이 목표계획법 문제의 전체 계산시간을 줄이는 것과 직결되지 않음을 확인할 수 있었으며 그 이유는 밝힐 수 있었다.

본 연구의 새로운 방법은 목표계획법 문제의 초기해를 구성할 때 도입변수 선택에 있어서 기존의 심플렉스법 판단기준 대신에 LHS를 이용할 수 있는 가능성을 보였다. 그러나, 이것을 이용하여 반복과정의 수는 물론 전체 계산시간까지도 줄일 수 있는 알고리즘의 개발에 대한 연구가 앞으로의 과제라 하겠다.

참 고 문 헌

[1] Arthur, J. L. and A. Ravindran, "An Efficient Goal Programming Algorithm Using Constraint Partitioning and Variable Elimination", *Management Science*, Vol.24, No.8(1978).

[2] Harris, P. M. J., "Pivot Selection Methods of the Devex Lp Code", *Math. Programming*, Vol.5(1973), pp.1-28.

[3] Hellerman, E. and D. Rarick, "Reinversion with the Preassinged Pivot Procedure", *Math. Programming*, Vol.1 (1971), pp.195-216.

[4] Ignizio, J. P., "A Review of Goal Programming : A tool for Multiple Objective Analysis", *Journal of Operational Research Society*, Vol.29(1978), pp.1109-1119.

[5] Ignizio, J. P., "Generalized Goal Programming. An Overview", *Computer & Operational Research*, Vol.10, No.4 (1983), pp.277-289.

[6] Ignizio, J. P., "Solving Large-Scale Problems : A Venture into a New Dimension", *Journal of Operational Research Society*, Vol.31(1980), pp.217-225.

[7] Lee, S. M., M. Gen and B. H. Rho, "A Revised Iterative Algorithm for Decomposition Goal Programming", *Internal Journal of Systems Science*, Vol.14, No.12(1983), pp.1383-1393.

[8] Richard, D. M, "A Spike Collective Dynamic Factorization Algorithm for the Simplex Method", *Management Science*, Vol.24(1978), No.10.

[9] Schniederjans, M. J. and N. K. Kwak, "An Alternative Solution Method for Goal Programming Problem : a Tutorial", *Journal of Operational Research Society*, Vol.33(1982), pp.247-251.