選數制約 線型背囊問題의 解法研究 *

- An Algorithm for a Cardinality Constrained Linear Programming Knapsack Problem -

원 중 연**

Won, Joong-Yeon

Abstract

An algorithm for solving the cardinality constrained linear programming knapsack problem is presented. The algorithm has a convenient structure for a branch-and-bound approach to the integer version, especially to the 0-1 collapsing knapsack problem. A numerical example is given.

1. 서 론

選數制約 線型背囊問題(cardinality constrained linear programming knapsack problem)는 다음과 같이 표현된다. [1,2]

(P) *Maximize* $\Sigma_{j \in N} q_j x_j$ (1)

subject to $\Sigma_{j \in N} a_j x_j \leq T,$ (2)

$\Sigma_{j \in N} x_j = k,$ (3)

$0 \leq x_j \leq 1, \ j \in N.$ (4)

여기서 $N=\{1, \ldots, n\}, \ 1 \leq k \leq n, \ 0 < a_j \leq T, \ 0 < q_j, \ j \in N,$ 이다.

문제 (P)는 부제약식(side constraints)을 갖는 集合分割問題(set partitioning problem)를 풀 때 부문제로서 발생한다.[1,2] 이 문제는 다음과 같은 0-1 縮化背囊問題 (0-1 collapsing knapsack problem; CK)[4]의 연속완화(continuous relaxation) 상한을 구하는데도 사용되고 있다.

(CK) *Maximize* $\{ \Sigma_{j \in N} q_j x_j : \ \Sigma_{j \in N} a_j x_j \leq h(\Sigma_{j \in N} x_j), \ x_j = 0 \text{ or } 1, \ j \in N \}$

여기서 함수 $h : N \rightarrow$ N은 비선형 비증가함수이며, N은 양의 정수집합이다. 문제 (CK)의 연속완화 상한문제(CCK)는 다음과 같이 정의된다.[3,5]

* 이 논문은 1995년도 경기대학교 교내연구비 지원에 의하여 연구되었음.

**경기대학교 산업공학과

(CCK) $\quad Maximize_{k\in N} \quad \{ \Sigma_{j\in N} q_j x_j \quad : \quad \Sigma_{j\in N} a_j x_j \le h(k), \quad \Sigma_{j\in N} x_j = k, \quad 0 \le x_j \le 1, \quad j\in N\}$

문제 (CCK)는 임의의 k에 대해 $h(k) = T$로 설정하면 바로 문제 (P)가 되며 따라서 문제 (CCK)의 최적해는 모든 k ($k\in N$)에 대한 문제 (P)들을 반복적으로 풀므로써 구해진다. 문제 (P)는 一般 多重選擇 線型背囊問題[6]의 특수한 경우로서도 발생하며 연구[6]의 제시된 문제에서 고려되는 선택집합의 수가 단지 한 개일 경우의 문제에 해당된다.

연구[1]은 문제 (P)의 최적해를 구하는 복잡도 $O(n^3)$의 해법을 제시하였고, 연구[2]는 제약식 (3)을 완화, n개의 線型背囊問題로 변환하여 풀므로써 복잡도 $O(n^2)$의 해법을 제시하였다. 그러나 연구[2]의 해법은 먼저 열등한 변수들을 제거하고 나머지 모든 변수의 계수가 양의 값을 갖도록 변환된 후 적용가능하므로 정수문제 (CK)를 풀기위한 분지한계해법에 반복적으로 적용되기 힘들다는 단점이 있다. 문제 (P)는 연구[6]의 해법과정중 부문제로 활용될 수 있으므로 연구[6]의 해법을 문제 (P)에 적용할 경우 복잡도는 $O(n^3)$으로 나타나고 있다.

본 연구에서는 문제 (P)에 대한 해법으로서 열등변수의 제거없이 제약식 (2)의 우변상수 T에 대한 모수분석을 이용하여 모든 k ($\in N$)에 대한 우변상수의 변화에 대응되는 해를 쉽게 찾으므로써, 정수문제 (CK)를 위한 분지한계해법에 적합히 활용될 수 있는 해법을 제시하고자 한다. 이 해법은 해의 탐색과정에서 자료를 크기순으로 미리 배열하고 여기에 이진탐색법을 적용하여 효율성을 높임으로써 복잡도 $O(n^2 \log n)$을 갖는다.

2. 해법 및 분석

문제 (P)의 기저가능해가 갖는 특성은 다음과 같다. 즉, 한 기저가능해에서 모든 변수들은 정수값을 갖거나 또는 최대로 두 개의 변수만이 분수값을 가질 수 있고, 이 때 두 변수의 합은 항상 1을 취하게 된다. 이것은 제약식 (3)에서 k가 정수이고 제약식 (4)에 의해 각 변수들은 0과 1사이의 값을 가지기 때문이다. 이 특성에 의하여 우변상수 T에 대한 모수분석 형태의 해법이 가능하다.

비율 $\theta(j_1, j_2)$를 다음과 같이 정의한다. 여기서 $a_{j_1} = a_{j_2}$이면 $\theta \equiv \infty$로 정의한다.

$$\theta(j_1, j_2) = (q_{j_2} - q_{j_1})/(a_{j_2} - a_{j_1})$$

문제 (P)의 제약식 (2)에서 계수 a_j가 가장 작은 k개 변수들을 찾고 해당하는 지수집합을 J라 할 때 다음과 같은 해를 고려하자. 단, 동일한 a_j가 여러 개 있을 경우는 각 해당되는 계수 q_j가 큰 변수를 우선 선택한다.

$$x_j = 1, \; j\in J, \; x_j = 0, \; j\in N \setminus J$$

$\overline{T} = \Sigma_{j\in J} a_j$ 라 할 때 $\overline{T} = T$ 이면 이 해는 최적해가 된다. 그러나, $\overline{T} > T$ 이면 제약식 (2)를 만족시키는 해는 없으므로 문제 (P)는 비가해이다. $\overline{T} \le T$일 경우 다음 보조정리 1은 더 향상된 해를 찾을 수 있는 모수분석방법을 제시한다.

보조정리 1 우변상수 $\overline{T}$가 증가하면서 발생하는 기저가능해에서 분수값을 갖는 변수의 지수 f_1, f_2, 및 쌍대변수값 θ는 다음과 같이 결정된다.

$$\theta(f_1, f_2) = \max_{j_1\in J} \max_{j_2\in N\setminus J,\, j_2 > j_1} \{\theta(j_1, j_2)\}$$

(증명) 우변상수가 $\overline{T}$일 때의 해에서 1의 값을 취하고 있는 변수들의 지수집합은 J 이다. 우변상수가 α(α>0, α는 충분히 작은 수)만큼 더 커짐에 따라 이를 충족시키기 위한 해의 변화는 다음과 같다. 즉, J에 속한 한 지수 j_1에 해당하는 변수가 1에서 감소하여 양의 분수값을 취하고, J에 속하지 않은 다른 한 지수 j_2 ($j_2 > j_1$, $j_2 \in N \setminus J$)에 해당하는 변수가 0에서 양의 분수값으로 증가하여야 한다. 제약식 (2), (3)의 열벡타만으로 이루어 지는 2x2 축소 기저행열 B를 정의하자. 분수값을 갖는 기저변수는 x_{j_1}, x_{j_2}, $j_1 \in J$, $j_2 \in N \setminus J$, $j_2 > j_1$,의 형태로 발생이 되므로 축소 기저행열 B는 기저벡타 (x_{j_1}, x_{j_2})에 대응된다. 이 때의 목적함수치 z는 다음과 같이 증가한다.

$$
\begin{aligned}
z(\overline{T}+\alpha) &= z(\overline{T}) + \alpha c_B B^{-1} e_1 \\
&= z_i(\overline{T}) + \alpha (q_{j_2} - q_{j_1})/(a_{j_2} - a_{j_1}) \\
&= z_i(\overline{T}) + \alpha \theta(j_1, j_2)
\end{aligned}
$$

여기서 $c_B = (q_{j_1}, q_{j_2})$, $e_1 = (1, 0)^t$이다. 따라서 목적함수치가 최대로 증가되는 쌍대변수값 및 분수값을 취하게 되는 변수들의 지수 f_1, f_2는 다음과 같이 구해진다.

$$
\theta(f_1, f_2) = \max_{j_1 \in J} \max_{j_2 \in N \setminus J,\ j_2 > j_1} \{\theta(j_1, j_2)\}
$$

따라서 본 정리가 성립된다. □

보조정리 1에서 α가 0으로부터 $(a_{f_2} - a_{f_1})$만큼 증가하게 되면 양의 분수값을 취하던 변수 x_{f_1}은 0이 되고 변수 x_{f_2}는 1의 값을 갖게 되어 모든 변수는 정수값을 취한다. 여기서 1의 값을 취하는 변수들의 지수집합 J에는 f_2가 포함되고 f_1은 탈락이 된다.

다음은 보조정리 1의 반복적용에 의해 초기해로부터 출발 우변상수 T에 이르기까지 발생하는 최적해를 찾는 해법을 제시한다. 우변상수는 매 회마다 선정되는 비율 $\theta(j_1, j_2)$에 해당하는 $(a_{j_2} - a_{j_1})$ 만큼씩 증가하고 이에따라 목적함수치는 쌍대변수값의 비율로 증가한다.

해 법

0. $J \leftarrow \phi$, $L \leftarrow \phi$.
1. 계수 a_j가 가장 작은 k개의 변수들을 찾고 해당되는 지수집합을 J라 한다. 동일한 a_j에 대해서는 q_j가 큰 변수를 우선 선택한다. J의 원소들을 비감소하는 크기순으로 배열하고, $\overline{T} \leftarrow \Sigma_{j \in J} a_j$ 라 놓는다.
2. $\overline{T} = T$이면 최적해는 다음과 같다. 해법과정을 끝낸다.

   $x_j = 1$, $j \in J$, $x_j = 0$, $j \in N \setminus J$

   $\overline{T} > T$이면 문제 (P)는 비가해이므로 해법과정을 끝낸다.

   $\overline{T} < T$이면 단계 3으로 간다.
3. 각 지수 j_1, j_2간의 비율 $\theta(j_1, j_2)$들을 계산하고 비증가하는 순으로 배열하여 목록 L에 넣는다.

   $\theta(j_1, j_2) = (q_{j_2} - q_{j_1})/(a_{j_2} - a_{j_1})$, $j_1 < j_2$, $j_1 \in N$, $j_2 \in N$

4. 목록 L에서 첫번째에 위치한 가장 큰 비율 $\theta(j_1, j_2)$를 선택한다.

$j_1 \in J$, $j_2 \notin J$이면, $\overline{T} \leftarrow \overline{T} + (a_{j_2} - a_{j_1})$를 계산한다.

$\overline{T} \geq T$이면 $f_1 \leftarrow j_1$, $f_2 \leftarrow j_2$, $\overline{e} \leftarrow \overline{T} - T$라 하고 단계 6으로 간다.

$\overline{T} < T$이면 $J \leftarrow J \cup \{j_2\} \setminus \{j_1\}$라 하고 단계 5로 간다.

$j_1 \notin J$이거나 $j_2 \in J$이면 단계 5로 간다.

5. $L \leftarrow L \setminus \{\theta(j_1, j_2)\}$로 수정하고 단계 4로 간다.
6. 최적해는 다음과 같다.

$$x_{f_1} = \overline{e} / (a_{f_2} - a_{f_1}),\ x_{f_2} = (a_{f_2} - a_{f_1} - \overline{e}) / (a_{f_2} - a_{f_1}),$$

$$x_j = 1,\ j \in J \setminus \{f_1\},\ x_j = 0,\ j \in N \setminus J \setminus \{f_2\}$$

정리 2 문제 (P)에 대한 해법의 계산상 복잡도는 최악상황하에서 $O(n^2 \log n)$이다.

(증명) 단계 1에서 집합 J에 해당하는 지수들을 찾는데 $O(kn)$의 계산이 필요하다. J의 원소들을 크기순으로 배열하는데 $O(k \log k) \leq O(n \log n)$의 계산이 소요된다. 단계 2에서는 상수회의 계산이 필요하고 단계 3에서 모든 비율 계산에 $O(n^2)$과 비증가하는 순서로 배열하여 L을 작성하는데 $O(n^2 \log n)$이 필요하다. 단계 4에서 집합 J의 원소는 이미 크기순으로 배열이 되었으므로 j_1 및 j_2가 각각 J에 속하는 지를 판단하는데 $O(\log k) \leq O(\log n)$의 계산이 소요된다. 또한 $\overline{T} < T$일 경우 집합 J에서 j_1을 삭제하고 j_2를 크기 순으로 첨가하는데 $O(\log n)$이 소요된다. 단계 5에서 L의 원소수는 최대로 $O(n^2)$이다. 따라서 주 회전단계인 단계 4 및 단계 5의 최대계산양은 $O(n^2 \log n)$이다. 단계 6은 상수회에 계산된다. 이상으로부터 해법의 최악상황하 복잡도는 $O(n^2 \log n)$이다. □

3. 수치예제

다음 문제 (P)의 최적해를 구한다.

(P) *Maximize* $\Sigma_{j \in N} q_j x_j$

$$\text{s.t.}\quad \Sigma_{j \in N} a_j x_j \leq 20,$$

$$\Sigma_{j \in N} x_j = 2,$$

$$0 \leq x_j \leq 1,\ j \in N = \{1, \ldots, 10\}.$$

여기서, 계수 q_j 및 a_j의 값은 다음 표와 같다.

q_i, a_i \ i	1	2	3	4	5	6	7	8	9	10
q_i	7	10	11	13	16	17	19	18	21	22
a_i	2	4	6	9	13	16	20	22	26	31

<1회>

1. $J = \{1, 2\}$, $\overline{T} = 6$
2. $\overline{T} < T\ (=20)$이므로 단계 3으로 간다.

3. $L=\{\ \theta(1,2),\ \theta(1,3),\ \theta(1,4),\ \theta(1,5),\ \theta(4,5)=\theta(8,9),\ \theta(1,6)=\theta(3,5),$
$\theta(1,7)=\theta(2,5)=\theta(3,4),\ \theta(2,4)=\theta(3,6),\ \theta(1,9)=\theta(2,6),$
$\theta(3,7)=\theta(4,6),\ \theta(2,7),\ \theta(1,8),\ \theta(4,7),\ \theta(4,9),\ \theta(1,10),$
$\theta(2,3)=\theta(2,9)=\theta(3,9)=\theta(6,7),\ \theta(2,8)=\theta(2,10)=\theta(8,10),\ \theta(3,10),$
$\theta(3,8),\ \theta(5,7),\ \theta(4,10),\ \theta(6,9),\ \theta(4,8)=\theta(5,9),\ \theta(5,6)=\theta(5,10)=$
$\theta(6,10)=\theta(7,9),\ \theta(7,10),\ \theta(5,8),\ \theta(9,10),\ \theta(6,8),\ \theta(7,8)\}$

4. $\theta(1,2)$ 선택, $1\in J$, $2\in J$이므로 단계 5로 간다.
5. $L=L\setminus\{\theta(1,2)\}$, 단계 4로 간다.

<2회>

4. $\theta(1,3)$ 선택, $1\in J$, $3\notin J$이므로 $\overline{T}=6+4=10$.
$\overline{T}<T$이므로 $J=\{2,3\}$, 단계 5로 간다.
5. $L=L\setminus\{\theta(1,3)\}$, 단계 4로 간다.

<3회>

4. $\theta(1,4)$ 선택, $1\notin J$이므로 단계 5로 간다.
5. $L=L\setminus\{\theta(1,4)\}$, 단계 4로 간다.

<4회>

4. $\theta(1,5)$ 선택, $1\notin J$이므로 단계 5로 간다.
5. $L=L\setminus\{\theta(1,5)\}$, 단계 4로 간다.

<5회>

4. $\theta(4,5)$ 선택, $4\notin J$이므로 단계 5로 간다.
5. $L=L\setminus\{\theta(4,5)\}$, 단계 4로 간다.

<6회>

4. $\theta(8,9)$ 선택, $8\notin J$이므로 단계 5로 간다.
5. $L=L\setminus\{\theta(8,9)\}$, 단계 4로 간다.

<7회>

4. $\theta(1,6)$ 선택, $1\notin J$이므로 단계 5로 간다.
5. $L=L\setminus\{\theta(1,6)\}$, 단계 4로 간다.

<8회>

4. $\theta(3,5)$ 선택, $3\notin J$이므로 단계 5로 간다.
5. $L=L\setminus\{\theta(3,5)\}$, 단계 4로 간다.

<9회>

4. $\theta(1,7)$ 선택, $1\notin J$이므로 단계 5로 간다.
5. $L=L\setminus\{\theta(1,7)\}$, 단계 4로 간다.

<10회>

4. $\theta(2,5)$ 선택, $2\in J$, $5\notin J$이므로 $\overline{T}=10+9=19$.
$\overline{T}<T$이므로 $J=\{3,5\}$, 단계 5로 간다.
5. $L=L\setminus\{\theta(2,5)\}$, 단계 4로 간다.

<11회>

4. $\theta(3,4)$ 선택, $3\in J$, $4\notin J$이므로 $\overline{T}=19+3=22$.

$\overline{T} > T$이므로 $f_1=3$, $f_2=4$, $\overline{e}=22-20=2$, 단계 6으로 간다.

6. 최적해는 다음과 같다.

$x_3=2/3$, $x_4=1/3$, $x_5=1$, $x_j=0$, $j \neq 3,4,5$.

4. 결 론

選數制約 線型背囊問題[1,2]는 集合分割問題의 부문제로 발생하고 있으며, 또한 0-1 縮化背囊問題(0-1 collapsing knapsack problem)[4]의 연속상한[3,5]을 구하는데 활용되고 있다. 이 문제는 一般 多重選擇 線型背囊問題[6]에서 선택집합의 수가 하나일 경우의 부문제로도 발생하고 있다.

연구[2]는 해법의 활용성을 높이기 위하여 열등변수를 제거하여 풀었으므로 0-1 縮化背囊問題를 위한 분지한계해법에 반복적으로 사용되기 힘들다는 단점이 있다. 본 연구에서는 변수의 제거없이 다만 우변상수의 변화에 따른 해의 생성을 구조적으로 파악하므로써 분지한계해법에서 반복적으로 발생하는 후보문제들의 최적해를 쉽게 구하는 해법을 제시하였다. 본 연구에서 제시한 해법은 해의 탐색에서 자료를 미리 크기순으로 배열하므로써 효율성을 높여 연구[6]이 본 문제에 적용될 때의 복잡도보다 효율적인 $O(n^2 \log n)$을 갖는다.

참 고 문 헌

1. Campello, R. E. and N.F. Maculan, "Lagrangean Relaxation for a Lower Bound to a Set Partitioning Problem with Side Constraints: Properties and Algorithms", *Discrete Appl. Math.* 18, 119-136, 1987.
2. Dudzinski, K., "On a Cardinality Constrained Linear Programming Knapsack Problem", *Opns. Res. Letters* 8, 215-218, 1989.
3. Posner, M. E., "The Continuous Collapsing Knapsack Problem", *Math. Progr.* 26, 76-86, 1983.
4. Posner, M. E. and M. Guignard, "The 0-1 Collapsing Knapsack Problem", *Math. Progr.* 15, 155-161, 1978.
5. Posner, M. E. and H. Suzuki, "A Dual Approach for the Continuous Collapsing Knapsack Problem", *Math. Progr.* 39, 207-214, 1987.
6. 원중연, "일반 다중선택 선형배낭문제의 신속한 해법연구", 대한산업공학회지, 제21권, 제4호, 519-527. 1995.