〈研究論文〉

Journal of the Korean Institute of Industrial Engineers
Vol. 9, No. 2, December 1983

修正된 多重選択 背囊問題의 解法에 관한 研究

(A Study on the Modified Multiple Choice Knapsack Problem)

元 重 淵*
鄭 聖 進**

Abstract

The multiple choice knapsack problem is modified. To solve this modified multiple choice knapsack problem, Lagrangian relaxation is used, and to take advantage of the special structure of subproblems obtained by decomposing this relaxed Lagrangian problem, a modified ranking algorithm is used.

The K best rank order solutions obtained from each subproblem as a result of applying modified ranking algorithm are used to formulate restricted problems of the original problem.

The optimality for the original problem of solutions obtained from the restricted problems is judged from the upper bound and lower bounds calculated iteratively from the relaxed problem and restricted problems, respectively.

1. 序 論

Nauss〔13〕는 既存의 背囊問題(knapsack problem)에 多重選択 制限式(multiple-choice constraint)을 첨가한 問題를 제시하였다. 이러한 多重選択 背囊問題는 總 投資資本의 制約下에서 會社內의 서로 獨立된 여러 부서들이 계획하고 있는 여러 프로젝트들중 總利益이 最大가 되도록 각 부서별로 프로젝트를 하나씩 選択하는 경우에 適用될 수 있다.

Sinha 와 Zoltners〔15〕도 같은 多重選択 背囊 問題를 고려하고, 이 問題를 LP緩和(relaxation) 시켰을 때의 特性들을 밝혔으며, 이 特性들을 이용하여 效率的인 分析限界技法을 개발하였다.

Glover 와 Klingman〔7〕, Zemel〔18〕은 多重選択 背囊問題의 LP緩和된 問題에 대하여 좀 더 一般化된 特性들을 研究하고, 이에 基楚한 技法들을 提示하였다.

本 論文에서는 多重選択 制限式을 修正하여, 각 부서들이 계획하고 있는 여러 프로젝트들 중에서 각 부서별로 하나이상의 프로젝트를 選択할 수 있는 경우로의 擴張된 問題를 고려하고, 이 問題에 대한 解法을 研究하였다. 이 修正된 多重選択 背囊問題에서는 多重選択 背囊問題의 解法에 基本이 되고있는 LP緩和時의 여러 特性들이 전혀 나타나고 있지 않다.

* 京畿大學校
** 서울大學校 工科大學

2. 問題의 模型

修正된 多重選択 背嚢問題는 다음과 같이 表現될 수 있다.

目的函數 : $Max.\ \sum_{i=1}^{m} \sum_{j=1}^{n_i} p_{ij} x_{ij}$

$$s.t.\quad \sum_{i=1}^{m} \sum_{j=1}^{n_i} a_{ij} x_{ij} \leq b \quad \cdots\cdots(1)$$

$$\sum_{j=1}^{n_i} x_{ij} = r_i,\ \forall\, i=1,\cdots,m \quad (2)$$

$$x_{ij} = 0 \text{ 또는 } 1,\ \forall\ i=1,\cdots m,\ j=1,\cdots,n_i$$

각 기호에 대한 說明은 다음과 같다.

- m =多重選択集合들의 個數;
- n_i =i번째 多重選択集合에서 고려되고 있는 프로젝트들의 個數;
- p_{ij} =i번째 多重選択集合의 j번째 프로젝트에서 생기는 利益;
- a_{ij} =i번째 多重選択集合의 j번째 프로젝트에 소모되는 費用;
- b =사용가능한 總 投資額;
- r_i =i번째 多重選択集合에서 選択되어야 하는 프로젝트의 數;
- x_{ij} = 1 ; i번째 多重選択集合의 j번째 프로젝트가 選択될때, 0 ; i번째 多重選択集合의 j번째 프로젝트가 選択되지 않을 때.

여기서, 一般的으로 $p_{ij}\geq 0, a_{ij}\geq 0, b>0, 1\leq r_i<n_i$ 라고 假定할 수 있다. 앞으로 사용될 記号들은 다음과 같다.

- (MMCKP) : 修正된 多重選択 背嚢問題;
- (MCKP) : 多重選択 背嚢問題;
- (PRu) : 問題(MMCKP)에서 制限式 (1)에 대해 Lagrangian 乘數 u를 이용하여 緩和한 問題;
- (SPi) : 問題(PRu)로부터 分割(decomposition)된 副問題(subproblem), i($i=1,\cdots,m$).
- $(\bar{\cdot})$: 問題(·)의 LP 緩和;
- $v(\cdot)$: 問題(·)의 最適 目的函數値;
- $\bar{u}$: 問題($\overline{\text{MMCKP}}$)의 制限式 (1) 에 대한 最適雙對解.

3. 解 法

3-1. Lagrangian 緩和

問題 (MMCKP)의 構造는 制限式 (2)가 特殊한 형태를 취하고 있으며, 따라서 상대적으로 制限式 (1)이 問題解決을 어렵게 만들고 있다. 그러므로, 制限式(1)에 대해 Lagrangian 乘數 u(≧0)를 사용하여 Lagrangian 緩和를 취하면 다음과 같은 問題(PRu)가 된다.

$$\text{(PRu)}\quad Max.\ \sum_{i=1}^{m} \sum_{j=1}^{n_i} (p_{ij} - u\, a_{ij})\, x_{ij} + u\, b$$

$$s.t.\quad \sum_{j=1}^{n_i} x_{ij} = r_i,\ \forall\ i=1,\cdots,m$$

$$x_{ij}=0 \text{ 또는 } 1, \forall\ i=1,\cdots,m,\ j=1,\cdots,n_i$$

問題 (PRu)는 (MMCKP)가 緩和된 問題이므로, v(PRu)는 모든 u(≧0)에 대해 v(MMCKP) 보다 크거나 같게된다. 그러므로 v(PRu)는 v(MMCKP)의 上限値(upper bound)가 되며, v(PRu)를 最小化시키는 Lagrangian 乘數 u값이 필요하다. 이러한 理想的인 u값은 다음 問題(D)의 最適解가 된다.

$$\text{(D)}\quad Max.\ v(PRu) \qquad s.t.\quad u\geq 0$$

問題(D)와 같은 형태의 問題를 해결하기 위하여 여러 方法들이 사용되고 있다 ([2], [3], [4]). 그러나, 問題 (PRu)에는 整數特性(Integrality Property)이 만족되고 있다. 즉, $x_{ij}(\forall_i, \forall_j)$에 부가되어 있는 整數條件을 제거하여도 v(PRu)에는 변함이 없다. 따라서, 다음과 같은 等式이 성립한다 [6].

$$v(D) = \min_{u\geq 0} v(PRu) = v(PR\bar{u})$$

여기서, $\bar{u}$는 問題 ($\overline{\text{MMCKP}}$)의 制限式(1)에 대한 最適雙對解이다. 즉, v(PRu)를 最小化 시키는 乘數u는 $\bar{u}$ 가 된다.

問題(PR$\bar{u}$)는 m개의 서로 독립적인 副問題(sub-problem)들로 分割될 수 있으며, (PR$\bar{u}$)의 最適解는 이들 m개의 副問題를 해결하므로써 쉽게 구해질 수 있다.

問題 (PR$\bar{u}$)의 最適解를 X* 라 하면 X*가 問題(MMCKP)의 最適解가 되기 위하여는 다음과 같은 條件을 만족하여야 한다. 즉,

① $\sum_{i=1}^{m} \sum_{j=1}^{n_i} a_{ij}\ x^*_{ij} \leq b$

② $\bar{u}(b-\sum_{i=1}^{m} \sum_{j=1}^{n_i} a_{ij}\ x^*_{ij}) = 0$

그러나, 이 問題에서는 一般的으로 duality gap(즉, v(D) − v(MMCKP) ≠ 0)이 존재하므로, 위의 條件中 두번째 式이 만족되기 힘들다. 여기서 우리는 다음의 3-2節과 같은 過程을 거쳐 쉽게 언어지는 最

善의 序列解(best rank order solution)들을 이용하여 問題 (MMCKP)의 最適解를 구하는 方法을 생각하였다.

3-2. 解의 序列化 過程(ranking procedure)

問題 (PR$\bar{u}^*$)는 다음과 같은 m개의 副問題(SPi)들로 分割될 수 있다. ($i=1, \cdots, m$)

$$(SPi): Max.\ \sum_{j=1}^{n_i} (p_{ij} - \bar{u}\, a_{ij})\, x_{ij}$$
$$s.\ t.\quad \sum_{j=1}^{n_i} x_{ij} = r_i \quad \cdots\cdots\cdots\cdots(3)$$
$$x_{ij} = 0 \text{또는 } 1,\ \forall j=1, \cdots, n_i$$

制限式(3)의 構造가 特殊하므로, 이것을 이용하여 각 副問題(SPi)에서 K_i 개의 最善의 序列解를 쉽게 구할 수 있으며 최종적으로는 이들 K_i 개의 解를 이용하여 (MMCKP)의 最適解를 구할 수 있다.

(SPi)에서 最善의 序列解들을 效率的으로 구하기 위하여, 既存의 序列技法(ranking algorithm) ([10], [12])을, 현재 고려되고 있는 각 副問題에 알맞게끔 아래와 같이 修正하여 適用시켰다.

修正된 序列技法

段階 0 : 副問題(SPi)의 目的函數係數들을 큰것 부터 작은 順으로 分類整理한다.

段階 1 : 副問題 (SPi)의 最適解를 구한다. 이 解가 最善의 序列解들 중에서 제일좋은 解(best 1 solution)가 된다. 이 解를 $X_{i\cdot}^1 = (x_{i1}^1, \cdots, x_{in_i}^1)$이라 하자. 目的函數의 係數가 이미 큰것 부터 작은 順으로 정리 되었으므로, 이 最適解는 매우 쉽게 구할 수 있다.

段階 2 : 이미 K_i개의 最善의 序列解가 구해졌으면 作業을 멈춘다.

段階 3 : k번째로 좋은 最善의 序列解 $X_{i\cdot}^k = (x_{i1}^k, \cdots, x_{in_i}^k)$가 구해졌다고 假定하자. 이 解로부터 다음과 같이 變數들을 고정시켜 n_i개의 候補問題(CP$_k^l$)를 만든다. ($l=1, \cdots, n_i$)

$(CP_k^1): x_{i1} = 1 - x_{i1}^k$

$(CP_k^2): x_{i1} = x_{i1}^k,\ x_{i2} = 1 - x_{i2}^k$

$(CP_k^3): x_{i1} = x_{i1}^k,\ x_{i2} = x_{i2}^k,\ x_{i3} = 1 - x_{i3}^k$

⋮

$(CP_k^{n_i}): x_{i1} = x_{i1}^k,\ x_{i2} = x_{i2}^k,\ x_{i3} = x_{i3}^k, \cdots\cdots x_{in_{i-1}} = x_{in_{i-1}}^k,\ x_{in_i} = 1 - x_{in_i}^k$

이들 n_i개 候補問題들 중에서 式 $\sum_{j=1}^{n_i} x_{ij} \leq r_i$를 만족시키는 候補問題들만 고려하여 그 最適解와 目的函數値를 계산한다. 이 制限式을 만족시키지 않는 候補問題들은 實行不可能(infeasible)한 問題 들이기 때문이다.

階段 4 : 段階 3에서 얻어진 解들중 다음과 같은 解는 제거한다.

1. 이미 前 過程에서 구해져 目錄에 첨가되었던 解가 다시 생긴 경우의 解.
2. 段階 3에서 고려되었던 k번째로 좋은 序列解의 目的函數値보다 더 큰 目的函數値를 갖는 解.

段階 5 : 段階 4에서 구한 解들과 目的函數値를 目錄에 첨가한다.

段階 6 : 目錄에 있는 解들중 이미 選択된 解를 제외한 나머지 解들중 目的函數値가 가장 큰 解를 그 다음의 ($k+1$)번째로 좋은 序列解로 選択한다.

k에 1을 더하고 段階 2로 간다.

3-3. 解　法

각 副問題 (SPi)에서 구한 K_i 개의 最善의 序列解들을 이용하여 새로운 多重選択 背囊問題를 형성 할 수 있으며, 이 問題는 다음과 같이 된다.

$$(MCKP): Max.\ \sum_{i=1}^{m} \sum_{k=1}^{K_i} f_{ik}\, y_{ik}$$
$$s.t.\quad \sum_{i=1}^{m} \sum_{k=1}^{K_i} g_{ik}\, y_{ik} \leq b$$
$$\sum_{k=1}^{K_i} y_{ik} = 1,\ \forall i=1, \cdots, m$$
$$y_{ik} = 0 \text{ 또는 } 1,\ \forall i=1, \cdots, m$$
$$\forall k=1, \cdots, K_i$$

여기서 $f_{ik} = \sum_{j=1}^{n_i} p_{ij}\, x_{ij}^k$, $g_i^k = \sum_{j=1}^{n_i} a_{ij}\, x_{ij}^k$ 이다.

이 (MCKP)問題는 (MMCKP)의 縮小問題(restricted problem)가 된다([3], [4]).

그러므로, v(MCKP)는 問題 (MMCKP)의 下限値(lower bound)가 되며, 앞에서 언급한 바와 같이 v(PR$\bar{u}$)는 (MMCKP)의 가장 좋은 上限値가 된다. 여기서, v(PR$\bar{u}$)와 v(MCKP)의 차이를 CRI라 하고 각副問題 (SPi)에서 가장 좋은 序列解와 K_i번째로 좋은 序列解들의 目的函數値들의 차이를 DEL(i)라 하자. 다음과 같은 Lemma[16]을 適用하여 問題 (MMCKP)에 대한 最適解 與否를 判別할 수 있다.

Lemma. 모든 i에 대하여($i=1, \cdots, m$), DEL(i)≥

CRI이면, 問題 (MCKP)의 最適解에 해당하는 問題 (MMCKP)의 解가 바로 (MMCKP)의 最適解가 된다.

(證明) 問題 (MCKP)를 형성하는데에 사용된 한 副問題 ($SP\bar{i}$)의 $K_{\bar{i}}$개 最善의 序列解들 중에 포함되지 않은 序列解를 $X_{\bar{i}}^{\bar{k}}$ ($\bar{k} > K_{\bar{i}}$)라 하고 實行可能한 解 X가 $X_{\bar{i}}^{\bar{k}}$를 포함한다고 假定하자. 問題(MMCKP)에서 $\bar{X}$의 目的函數값을 Z라 하면,

$$
\begin{aligned}
Z &\le \sum_{\substack{i=1 \\ i \ne \bar{i}}}^{m} \sum_{j=1}^{n_i} (p_{ij} - \bar{u}\, a_{ij}) x_{ij}^{1} + \sum_{j=1}^{n_{\bar{i}}} (p_{\bar{i}j} - \bar{u}\cdot a_{\bar{i}j}) x_{\bar{i}j}^{\bar{k}} + \bar{u} b \\
&\le v(PR\bar{u}) + \{ \sum_{j=1}^{n_{\bar{i}}} (p_{\bar{i}j} - \bar{u}\, a_{\bar{i}j}) x_{\bar{i}j}^{\bar{k}} - \sum_{j=1}^{n_{\bar{i}}} (p_{\bar{i}j} - \bar{u}\, a_{\bar{i}j}) x_{\bar{i}j}^{1} \} \\
&= v(PR\bar{u}) - DEL(\bar{i}) \\
&\le v(PR\bar{u}) - \min_{i} \{ DEL(i) \} \\
&\le v(PR\bar{u}) - CRI \\
&= v(PR\bar{u}) - v(PR\bar{u}) + v(\mathrm{MCKP}) \\
&= v(\mathrm{MCKP})
\end{aligned}
$$

즉, 問題 (MCKP)를 형성하는데에 사용되지 않은 序列解 $X_{\bar{i}}^{\bar{k}}$를 포함하는 實行可能한 解 X의 目的函數値는 항상 v(MCKP)보다 적게된다. 따라서, 問題 (MCKP)의 最適解에 해당하는 (MMCKP)의 解가 (MMCKP)의 最適解가 된다.

이 Lemma에 基礎하여 다음과 같은 修正된 多重選択 背囊問題의 解法을 考案 하였다.

修正된 多重選択 背囊問題의 解法

段階 1 : 問題(MMCKP)에서 制限式(1)에 해당하는 雙對最適解 $\bar{u}$를 구한다. 이 $\bar{u}$를 Lagrangian 乘數로 하여 問題 ($PR\bar{u}$)및 m개의 副問題 (SPi)를 형성한다.

段階 2 : 各 副問題 (SPi)에 修正된 序列技法을 適用하여 K_i개의 最善의 序列解들을 구한다. DEL(i) 및 $v(PR\bar{u})$를 計算한다.

段階 3 : 段階 2에서 얻어진 K_i개의 最善의 序列解들을 사용하여 問題 (MCKP)를 형성하고, v(MCKP) 및 CRI를 구한다.
이 問題(MCKP) 가 實行不可能이면 段階 5로 간다.

段階 4 : 모든 i에 대해 ($i = 1, \cdots, m$), DEL(i) $\ge$ CRI이면 (MCKP) 의 最適解가 願하는(MMCKP)의 最適解가 된다. 모든 i에 대해 DEL(i)$\ge$CRI가 아니면, 段階 5로 간다.

段階 5 : 問題(MCKP) 가 實行不可能이면, 모든 i에 대해 K_i값을 증가시킨다. 위의 경우가 아니면 DEL(i)<CRI 인 i에 해당하는 副問題 (SPi)에 대한 K_i 값만을 증가시킨다. 段階 2로 간다.

4. 結論 및 討議

修正된 多重選択 背囊問題에서는 多重選択 背囊問題의 解法에 基本이 되고 있는 LP緩和(relaxation)時의 여러 特性들이 전혀 나타나고 있지 않다. 여기서, 修正된 多重選択 背囊問題를 Lagrangian 緩和하였을때 생기는 整數特性(Integrality Property) 을 이용하여 最小上限値(least upper bound)를 구하고, 分割(decomposition)된 副問題들의 最善의 序列解 (best rank order solution)들로 새로이 형성한 多重選択 背囊問題에서 下限値들을 반복적으로 구하므로써, 최종적으로는 이들 값으로부터 原問題의 最適解를 判別하였다. 새로운 多重選択 背囊問題의 형성시 각 副問題에 알맞도록 修正한 序列技法(ranking algorithm)을 適用하므로써 不必要한 候補問題를 미리 제거하여 努力을 줄일 수 있었다.

무작위로 형성한 12개의 問題에 대해 本 硏究의 解法을 適用하였으며 MV8000 콤퓨터에 遂行한 結果를 다음 表1에 나타내었다.

問題(MMCKP)의 最小上限値는 고정되어 있으므로, (MMCKP)의 下限値가 증가할수록 CRI는 작아지며 따라서 더욱 신속히 最適解를 찾을 수 있다. (MMCKP)의 下限値는 序列解의 수가 증가 할 수록 커지나, 이 序列解의 수는 (MCKP)의 行의 數가 되므로, 그만큼 (MCKP)의 最適解를 찾는데에 시간이 걸리게 된다.

問題解決過程에는 多重選択集合의 數, 選択되어야 하는 프로젝트의 個數, 最善의 序列解들의 初期値 및 每 段階에서 증가되어야 할 數 等의 여러 要因이 作用하고 있으므로, 이 解法의 特性을 完全히 分析하기 위하여는 多樣한 형태의 여러 問題들을 충분히 分析하는 것이 필요할 것이다.

表 1. 12개의 문제에 대한 遂行結果

m	$(n_1, \cdots, n_m)$	$(r_1, \cdots, r_m)$	$(K_1, \cdots, K_m)$	문 제 번 호	CPU Time (sec)
2	(10, 10)	(2, 2)	(13, 13)	1	3.101
				2	3.466
				3	3.570
3	(8, 8, 7)	(3, 3, 2)	(11, 11, 4)	4	3.874
3	(8, 8, 7)	(3, 2, 2)	(11, 11, 4)	5	3.877
			(16, 7, 6)	6	5.947
			(15, 6, 6)	7	5.658
3	(10, 10, 9)	(2, 2, 2)	(13, 13, 10)	8	5.805
				9	6.900
				10	6.999
4	(7, 7, 7, 7)	(3, 3, 2, 2)	(10, 10, 6, 6)	11	6.775
				12	7.337

参 考 文 獻

1. Balas, E., and Zemal, E. ,"An Algorithm for Large Zero-One Knapsack Problem," *Operations Research*, Vol. 28, pp. 1130-1154, 1980.
2. Fisher, M. L., and Shapiro, J.F., "Constructive Duality in Integer Programming", *SIAM Journal of Applied Mathematics*, Vol. 27, pp. 31-52, 1974.
3. Fisher, M. L., Northup, W. D., and Shapiro, J. F., "Using Duality to Solve Discrete Optimization Problems;Theory and Computational Experience,"*Mathematical Programming Study*, Vol. 3, pp. 56-94, 1975.
4. Fisher, M. L., "The Lagrangian Relaxation Method for Solving Integer Programming Problems,"*Management Science*, Vol. 27, pp. 1-18, 1981.
5. Geoffrion, A. M., and Marsten, R. E., "Integer Programming Algorithms;A Framework and State-of-the-Art Survey,"*Management Science*, Vol. 18, pp. 465-491, 1972.
6. Geoffrion, A. M., "Lagrangean Relaxation for Integer Programming," *Mathematical Programming Study*, Vol. 2, pp. 82-114, 1974.
7. Glover, F. and Klingman, D., "A O(nlogn) Algorithm for LP Knapsacks with GUB Constraints," *Mathematical Programming*, Vol. 17, pp. 345-361, 1979.
8. Ibaraki, T., "Approximate Algorithms for the Multiple-Choice Continuous Knapsack Problems,"*Journal of the Operations Research Society of Japan*, Vol. 23, pp. 28-63, 1980.
9. Kochman, G.A., "Decomposition in Integer Programming," *Technical Report SOL 76-21*, Dept. of Operations Research, Stanford University, Stanford, California September, 1976.
10. Lowler, E. L., "A Procedure for Computing the K-best Solutions to Discrete Optimization Problems," *Management Science*, Vol. 18, pp. 401-406, 1972.
11. __________, "Fast Approximation Algorithms for Knapsack Problems," *Mathematics of Operations Research*, Vol. 4, pp. 339-356, 1979.

12. Murty, K. G., *Linear and Combinatorial Programming*, John Wiley & Sons Inc., New York, 1976.

13. Nauss, R. M., "The 0-1 Knapsack Problem with Multiple Choice Constraints," *European Journal of Operations Research*, pp. 125-131, 1978.

14. Salkin, H. M., *Integer Programming*, Addison-Weseley Publishing Company, Inc., 1975.

15. Sinha, P., and Zoltners, R. A., "The Multiple Choice Knapsack Problem," *Operations Research*, Vol. 27, pp. 503-515, 1979.

16. Sweeny, D. J., and Murphy, R. A., "A Method of Decomposition for Integer Programming," *Operations Research*, Vol 27, pp. 1128-1141, 1979.

17. ___________ and ___________, "Branch and Bound for Multi-item Scheduling," *Operations Research*, Vol. 29, pp. 853-864, 1981.

18. Zemel, E., "The Linear Multiple Choice Knapsack Problem," *Operations Research*, Vol. 28, pp. 1412-1423, 1980.