数理解析研究所講究録
第1734巻 2011年 85-92

複数制約付き割当問題に対する2つの釘付けテスト

Two Pegging Tests for the Multiplly Constrained Assignment Problem

防衛大学校情報工学科
横谷 大輔, 山田 武夫

YOKOYA Daisuke, YAMADA Takeo
{g47090, yamada}@nda.ac.jp
Department of Computer Science, The National Defense Academy
Yokosuka, Kanagawa 239-8686, Japan

1 はじめに

割当問題 (AP: assignment problem) についてはハンガリー法 [7] をはじめ効率的な解法が知られているが, 本稿ではこれに対し資源制約が複数本追加された**複数制約付き割当問題** (MCAP: multiplly constrained assigment problem) を考察する. 割当問題は様々な場面で応用されるが, 実際には何らかの制約が加わることが多い. そのため複数制約付き割当問題が生じる. これは次のように定式化される.

$$\textbf{MCAP}: \quad \text{minimize} \quad \sum_{i=1}^{n}\sum_{j=1}^{n} c_{ij}x_{ij} \tag{1}$$

$$\text{subject to} \quad \sum_{j}^{n} x_{ij} = 1, \quad i = 1,2,\cdots,n \tag{2}$$

$$\sum_{i}^{n} x_{ij} = 1, \quad j = 1,2,\cdots,n \tag{3}$$

$$\sum_{i}^{n}\sum_{j}^{n} r_{ij}^{k}x_{ij} \le b_k, \quad k = 1,2,\cdots,K \tag{4}$$

$$x_{ij} \in \{0,1\}, \quad \forall i,j. \tag{5}$$

(4) 式が追加された資源制約であり, (r_{ij}^{k}) は k 本目の資源制約式に対応する資源制約行列である.

MCAP で資源制約が1本の場合は Lieshout ら [8] による研究があるが, 本稿では複数本の場合について考える.

2 0-1 整数計画問題に対する釘付けテスト

0-1 整数計画問題 (BIP: binary integer problem) に対する釘付けテスト [9, 11] は周知であるが, 要点を以下にまとめる. 対象とする BIP を簡単のため P と記し, 具体的には

$$P\text{: minimize } z(x) := c^T x \quad \text{subject to } Ax = b,\ x_j \in \{0,1\},\ \forall j.$$

とし, この問題の最適解 (の一つ) を $x^\star$, 対応する最適目的関数値を $z^\star = z(x^\star)$ とする. この問題の

0-1 条件を連続緩和して得られる線形計画問題を $C(P)$ と記し，その最適解（の一つ）と対応する下界値を $\underline{x}$, $\underline{z} = z(\underline{x})$ と記す．また，P の任意の実行可能解 $\bar{x}$ は P の上界値を与えるが，これを $\bar{z} = z(\bar{x})$ とする．すなわち，

$$\underline{z} \le z^\star \le \bar{z}.$$

ここで，P にさらに制約式 $x_j = 1$ を付加した問題を $P(x_j = 1)$ とし，その下界値を $\underline{z}(x_j = 1)$ と記すと，

$$\underline{z}(x_j = 1) > \bar{z} \tag{6}$$

ならば，$x_j = 1$ であるような実行可能解からは $\bar{z}$ より良い（小さい）値は得られず，最適解においては $x_j^\star = 1$ ではあり得ない．すなわち，変数 x_j は $x_j = 0$ と**固定**（釘付け）される．同様に，

$$\underline{z}(x_j = 0) > \bar{z} \tag{7}$$

の場合には，$x_j = 1$ としてよい．

しきい値を

$$\theta_{j1} := \underline{z}(x_j = 1) - \underline{z}, \tag{8}$$

$$\theta_{j0} := \underline{z}(x_j = 0) - \underline{z} \tag{9}$$

と定義し，$gap = \bar{z} - \underline{z}$ とすると，(6)，(7) はそれぞれ下のようになる．

$$\theta_{j1} > gap \tag{10}$$

$$\theta_{j0} > gap \tag{11}$$

以下では，これらのしきい値の迅速な計算方法を説明する．

$C(P)$ の最適単体表 [2] が

$$\bar{b}_i = x_{B(i)} + \sum_{j \in N} \alpha_{ij} x_j, \tag{12}$$

$$z = \underline{z} + \sum_{j \in N} \alpha_{0j} x_j, \tag{13}$$

で与えられているとする．ここで，N は**非基底変数**に対する添字の集合であり，$B(i)$ は i 番目の**基底変数**を表す．最適性より，

$$\alpha_{0j} \ge 0,\ \forall j, \quad 0 \le \bar{b}_i \le 1,\ \forall i$$

を満たす．各々の i に対して

$$PU_i := \min\{-\alpha_{0j}/\alpha_{ij} \mid j \in N, \alpha_{ij} < 0\}(1 - \bar{b}_i), \tag{14}$$

$$PL_i := \min\{\alpha_{0j}/\alpha_{ij} \mid j \in N, \alpha_{ij} > 0\}\bar{b}_i. \tag{15}$$

とすると，しきい値は基底変数 $x_{B(i)}$ については $\theta_{B(i)1} = PU_i$，$\theta_{B(i)0} = PL_i$ で，非基底変数 x_j については $\theta_{j1} = \alpha_{0j}$ で与えられる．したがって，次が P に対する釘付けテストを与える．

定理 1 [9]

(i) 基底変数 $x_{B(i)}$ に対して，

$$PU_i > \bar{z} - \underline{z} \Rightarrow x^\star_{B(i)} = 0, \tag{16}$$

$$PL_i > \bar{z} - \underline{z} \Rightarrow x^\star_{B(i)} = 1. \tag{17}$$

(ii) 非基底変数 x_j $(j \in N)$ に対して,

$$\alpha_{0j} > \bar{z} - \underline{z} \;\Rightarrow\; x_j^\star = 0. \tag{18}$$

つまり PU_i, PL_i, α_{0j} をしきい値として釘付けを行う.

3 ラグランジュ緩和

MCAP の連続緩和問題 C(MCAP) を解いて得られる下界値を $\underline{z}_C$ と記す. 同時に, 制約式 (2), (3), (4) に対応する**最適双対変数** $u_i^\dagger$, $v_j^\dagger$, $\lambda_k^\dagger$ も得られる.

(4) 式にラグランジュ乗数 $\lambda_k \geq 0$ を乗じて (1) 式に繰り込むと, 次の**ラグランジュ緩和問題** [3] となる.

$$L(\lambda) : \text{minimize} \quad \sum_{i=1}^{n}\sum_{j=1}^{n}(c_{ij} + \sum_{k=1}^{M}\lambda_k r_{ij}^k)x_{ij} - \sum_{k=1}^{m}\lambda_k b_k$$
$$\text{subject to} \qquad (2), (3), (5)$$

これは $\lambda = (\lambda_k)$ を固定すると, 通常の割当問題となる. この最適関数値を $\underline{z}(\lambda)$ とすると次のことが成り立つ.

命題 1 (i) 任意の $\lambda \geq \mathbf{0}$ に対して $\underline{z}(\lambda)$ は MCAP の下界値を与える.

(ii) $\underline{z}(\lambda)$ は λ について区分的に線形な凹関数である.

(iii) λ において, $\underline{z}(\lambda)$ が微分可能であるなら, $\partial\underline{z}(\lambda)/\partial\lambda_k = \sum_i \sum_j r_{ij}^k x_{ij} - b_k$ である.

(iv) すべての k で $\partial\underline{z}(\lambda)/\partial\lambda_k \leq 0$ なら $L(\lambda)$ の解は MCAP の実行可能解を与える.

命題 1 の (i), (ii) に関連して, 最良の下界値を得るために次の**ラグランジュ双対問題**を考える.

$$\text{maximize} \quad \underline{z}(\lambda)$$
$$\text{subject to} \quad \lambda \geq \mathbf{0}$$

連続緩和問題で得られた $\lambda^\dagger$ はこの問題の最適解を与え, $\lambda = \lambda^\dagger$ としたときの下界値を $\underline{z}_L$ とすると, 次が成り立つことが証明できる [10].

定理 2 連続緩和による下界値とラグランジュ緩和による下界値は一致する. すなわち,

$$\underline{z}_C = \underline{z}_L$$

以下この下界値を $\underline{z}$ と記す.

4 ラグランジュ緩和を介在した釘付け

MCAP は n^2 変数, $2n - 1 + K$ 制約式の BIP で, 例えば $n = 1000, K = 100$ の場合には, 第 2 節の釘付けテストをそのまま適用しようとすると, 2100×10^6 の大きさの単体表を保有する必要があるので, 現実的でない. そこで, 前節のラグランジュ緩和を導入し, その上で釘付けテストを実施する. すなわち, 問題 $L(\lambda^\dagger)$ にさらに制約式 $x_{ij} = 1$ を付加した問題 $L(\lambda^\dagger, x_{ij} = 1)$ の下界値を $\underline{z}(\lambda^\dagger, x_{ij} = 1)$ と記すと,

$$\underline{z}(\lambda^\dagger, x_{ij} = 1) \leq \underline{z}(x_{ij} = 1) \tag{19}$$

なので,

$$\underline{z}(\lambda^{\dagger}, x_{ij} = 1) > \bar{z} \tag{20}$$

の場合には $x^{\star}_{ij} = 0$ と釘付けできる.

この場合，しきい値 $\underline{z}(\lambda^{\dagger}, x_{ij} = 1) - \underline{z}$ の計算は問題 $L(\lambda^{\dagger})$ が割当問題なので，一般の場合と比べはるかに容易になる．以下では，このしきい値の計算の仕方について，(i) ソフトな釘付けと (ii) ハードな釘付けの 2 つを提示する.

4.1 ソフトな釘付け

この方法は，$\underline{z}(\lambda^{\dagger}, x_{ij} = 1) - \underline{z}$ を通常の釘付けテストと同様に近似的に評価する方法で，そのときの値（しきい値）を θ^{S}_{ij} と記す．ただし，緩和問題 $L(\lambda^{\dagger})$ は割当問題なので，その特徴を生かすことにより計算が大幅に簡略化される．$L(\lambda^{\dagger})$ はハンガリー法等によって解き，得られた最適解をもとに問題を

$$\begin{aligned} Bx_B + Nx_n &= 1, \\ c_B x_B + c_N x_N &= z, \end{aligned}$$

と書き直す．ここで，x_B は基底変数，x_N は非基底変数を表し，B は基底変数に対応する列，N は非基底変数に対応する列である．これにより，最適解に対応する最適単体表は次のようになる.

$$\begin{aligned} x_B \quad + B^{-1}Nx_N &= B^{-1}\mathbf{1}, \\ (c_N - c_B B^{-1}N)x_N &= z - c_B B^{-1}\mathbf{1}. \end{aligned}$$

$n \times n$ の割当問題において，基底部分の行列 B とその逆行列 B^{-1} を得たとする．n が非常に大きい場合，行列 N のサイズは $(2n-1) \times (n^2 - 2n + 1)$ となるため，$B^{-1}N$ の全成分を計算するのは一般に多く記憶容量と計算時間を必要とする．ところが割当問題の場合，釘付けテストに必要なのは $B^{-1}N$ のごく一部のみであり，このことが計算の大幅なスピードアップを可能にする.

(12), (13) で与えられる最適単体表を考える．割当問題のユニモジュラー性 [1] から，行列 (B,N) の成分は,

$$\alpha_{ij} \in \{-1, 0, 1\},\ \forall i, \forall j \in N, \tag{21}$$

$$\bar{b}_i \in \{0, 1\},\ \forall i. \tag{22}$$

となっている.

$$N^{+} := \{j \in N \mid \alpha_{0j} > \bar{z} - \underline{z}\},\ N^{-} := \{j \in N \mid \alpha_{0j} \le \bar{z} - \underline{z}\}. \tag{23}$$

とすると，次が成り立つ.

定理 3 (i) $\bar{b}_i = 1$ かつ $\{j \in N^{-} \mid \alpha_{ij} = 1\} = \emptyset \Rightarrow x^{\star}_{B(i)} = 1,$

(ii) $\bar{b}_i = 0$ かつ $\{j \in N^{-} \mid \alpha_{ij} = -1\} = \emptyset \Rightarrow x^{\star}_{B(i)} = 0.$

この定理の言っていることは，釘付けテストを行う際に必要なのは，N^{-} に含まれる列のみであり，$\{j \in N^{-} \mid \alpha_{ij} = \pm 1\} = \emptyset$ が成り立つかどうかをチェックすれば良いということである．$|N^{-}|$ はたいてい $|N|$ よりはるかに小さく，定理 3 による釘付けは定理 1 を直接適用するよりもかなり速い.

4.2 ハードな釘付け

4.1 節においては $L(\lambda^\dagger, x_{ij}=1)$ の最適値を最適単体表から導かれる下界値により近似的に評価したが，本節では $L(\lambda^\dagger, x_{ij}=1)$ の最適値を厳密に計算したときのしきい値を

$$\theta^H_{ij} = \underline{z}(\lambda^\dagger, x_{ij}=1) - \underline{z} \tag{24}$$

とし，これを高速に求める方法を示す．$L(\lambda^\dagger)$ の最適解を $x^\star = (x^\star_{ij})$，最適双対解を $u^\star = (u^\star_i), v^\star = (v^\star_j)$ とする．また，$x^\star$ に対応する**完全マッチング**を $M^\star = \{(i,\alpha_i) \mid i = 1,2,\cdots,n\}$ とする．すなわち，$x^\star_{ij} = 1 \Leftrightarrow \alpha_i = j$．ここで，節点 i と j' 間の '重み' を

$$d^0_{ij} = c_{ij} + \gamma^\dagger_{ij} - u^\star_i - v^\star_j \tag{25}$$

とすると，$x^\star$ の最適性より

$$d^0_{ij} \geq 0,\ i,j = 1,2,\cdots,n \tag{26}$$

で，$M^\star$ の枝では

$$d^0_{ij} = 0,\ i = 1,2,\cdots,n \tag{27}$$

となる．また，これを用いると，$L(\lambda^\dagger)$ は次のように書き換えられる．

$$\begin{array}{ll} \text{minimize} & \displaystyle\sum_{i=1}^{n}\sum_{j=1}^{n} d^0_{ij}x^k_{ij} + \underline{z}^k \\ \text{subject to} & \displaystyle\sum_{j=1}^{n} x^k_{ij} = 1, \quad \forall i, \\ & \displaystyle\sum_{i=1}^{n} x^k_{ij} = 1, \quad \forall j, \\ & x^k_{ij} \in \{0,1\}, \quad \forall i,j. \end{array}$$

ここで，$(i,j) \notin M^\star$ として，$L(\lambda^\dagger, x_{ij}=1)$ を考えると，これは枝 (i,j') を含む**最短交互サイクル**を求めることに帰着する．

ここまでは節点数が $2n$ の 2 部グラフで考えたが，同じことを節点数 n の有向完全グラフ上でも議論できる．この場合，グラフは $G = (V,E)$ で節点集合は $V = \{(i,\alpha_i) | i = 1,2,\cdots,n\}$ であり，枝 (i,α_i) と (j,α_j) の間の重みは

$$d_{ij} = c_{i\alpha_j} + \gamma^\dagger - u^\star_i - v^\star_{\alpha_j} \tag{28}$$

となる．このようにして得られる距離行列を $D = (d_{ij})$ と表記し，節点 (i,α_i) から (j,α_j) への G での最短距離 f_{ij} を次のように導入する．

$$f_{ij} = \mathrm{dist}(v_i, v_j). \tag{29}$$

$F = (f_{ij})$ は $D = (d_{ij})$ に対してワーシャル・フロイド等 [4] の全点対最短距離アルゴリズムを適用することで得られる．すると，次のことがわかる．

定理 4 [6] しきい値 θ^H_{ij} $(1 \leq i,j \leq n)$ は，

$$\theta^H_{ij} = d_{i\beta_j} + f_{\beta_j i}. \tag{30}$$

で与えられる．

5 数値実験

本節ではソフトな釘付けとハードな釘付けの数値実験を行う．前節までのアルゴリズムを ANSI C 言語で実装し，Dell Precision T7400 computer (CPU: Xeon(R) X5482, 3.20GHz Quad-Core, 6.5GB RAM) 上で実験を行った．使用したソルバーは CPLEX 11.100 [5] である．

5.1 実験計画

例題は次のように生成した．c_{ij} は $[1, 1000]$ の一様乱数で生成し，付加制約 (r^k_{ij}) は c_{ij} に対し弱相関を持たせた上で，同じく $[1, 1000]$ の一様乱数で生成した．このとき付加制約に対しては次の 2 通りの例題を考えた．

密： 係数行列 (r^k_{ij}) の成分がすべて密に埋まっているもの．
疎： 1 本目の付加制約は密の場合と同じであるが，2 本目以降は (r^k_{ij}) の成分が 75%は 0 で，残りの 25%が $[1, 1000]$ の一様乱数のもの．

5.2 CPLEX による直接解

表 1, 2 はそれぞれ密，疎な例題タイプに対して問題を直接 CPLEX で解いたときの計算時間である．表の各行は 10 例題の平均である．密な例題においては $n = 200, 400, 600$，$K = 2, 4$ とし，疎な例題においては $n = 200, 400, 600$，$K = 4, 16$ として実験を行った．密，疎いずれの例題においても n, K の値が大きくなると多くの時間を要している．

表 1: 密なときの計算時間（秒）

n	K	
	2	4
200	15.16	293.62
400	121.12	2363.52
600	331.53	2536.75

表 2: 疎なときの計算時間（秒）

n	K	
	4	16
200	5.03	11.27
400	22.84	31.39
600	2915.50	2916.30

5.3 ソフトな釘付けとハードな釘付けの比較

表 3, 4 は密な例題，表 5, 6 は疎な例題に対するソフトな釘付けテストとハードな釘付けテストの実験結果の比較である．表の各項目は次を表す．この表も各行は 10 例題の平均である．

n： 問題サイズ
K： 付加制約の数
$\bar{\theta}^S, \bar{\theta}^H$： ソフト，ハードでのしきい値の平均
#fix0： 0 に固定された変数の個数
#fix1： 1 に固定された変数の個数
#unfix： 未固定の（残った）変数の個数
#row： 縮小後の問題の行数
#col： 縮小後の問題の列数

CPU： 計算時間の秒表示
#solved： 仮想釘付けテストおいて最適性の保証が得られた回数（10回中）

仮想釘付けテストにおける仮のギャップ（*vgap*）は密な例題では $vgap = 20$，疎な例題では $vgap = 5$ で実験を行った．どちらのタイプの例題においてもしきい値はハードな釘付けのほうが大きい値を取っていることがわかる．その結果#row，#col を見ると，縮小後の問題サイズはハードの方がより小さくなっており，特に列数は大幅に縮小されている．

6 結論

本報告では複数制約付き割当問題に対してソフトな釘付けテストとハードな釘付けテストの比較実験を行った．釘付け率はしきい値が大きいほど高くなるが，実際にハードな釘付けの方がソフトな釘付けよりもしきい値が大きく，釘付け率も高かった．このように，問題の一部に割当問題の構造が含まれている場合にはそれを活用したハードな釘付けが有効であると考えられる．

参考文献

[1] R.K. Ahuja, T.L. Magnanti and J.B. Orlin: *Network Flows: Theory, Algorithms, and Applications* (Prentice Hall, Englewood Criffs, 1993).

[2] G.B. Danzig: *Linear Programming and Extensions* (Princeton Univ. Press, Princeton, 1963).

[3] M. Fisher: The Lagrangian relaxation method for solving integer programming problems. *Management Science*, **50** (2004), 1861-1871.

[4] R.W. Floyd, Algorithm 97, Shortest path. Communications of the ACM 5(1962) 345.

[5] ILOG CPLEX 11.100, http://ilog.com/products/cplex, 2008.

[6] G. Kindervater, A. Volgenant, G. de Leve, V. van Gijlswijk, On dual solutions of the linear assignment problem, European Journal of Operational Research 19 (1985) 76-81.

[7] H.W. Kuhn: The Hungarian method for the assignment problem. *Naval Research Logistics Quarterly* **2** (1955), 83-97.

[8] P.M.D. Lieshout and A. Volgenant, "A branch-and-bound algorithm for the singly constrained assignment problem", *European Journal Operational Research*, Vol. 176, pp. 151-164. 2007.

[9] R.M. Nauss: *Parametric Integer Programming* (Univ. Missouri Press, Columbia, MI, 1979).

[10] L.A. Wolsey: *Integer Programming* (John Wiley & Sons, New York, 1998).

[11] B.-J. You and T. Yamada, "A virtual pegging approach to the precedence constrained knapsack problem", *European Journal Operational Research*, Vol. 183, pp. 618-632. 2007.

表 3: 密-ソフト（$vgap = 20$）

n	K	$\overline{\theta}^S$	#fix0	#fix1	#unfix	#row	#col	CPU	#solved
200	2	549.28	39170.6	16.5	812.9	369.0	812.9	2.91	9
	4	499.92	38905.1	4.7	1090.2	394.6	1090.2	1.76	10
400	2	563.04	157722.1	17.2	2260.7	767.6	2260.7	15.26	10
	4	516.28	156452.6	0.7	3546.7	802.6	3546.7	172.81	10
600	2	566.32	355800.3	16.7	4183.0	1168.6	4183.0	49.06	10
	4	519.74	352792.8	0.1	7207.1	1203.8	7207.1	1580.90	10

表 4: 密-ハード（$vgap = 20$）

n	K	$\overline{\theta}^H$	#fix0	#fix1	#unfix	#row	#col	CPU	#solved
200	2	561.21	39416.8	42.8	540.4	316.4	540.4	1.76	9
	4	506.74	39138.2	12.2	849.6	379.6	849.6	1.48	10
400	2	572.32	158433.9	50.5	1515.6	701.0	1515.6	12.53	10
	4	520.57	157106.0	3.2	2890.8	797.6	2890.8	150.53	10
600	2	574.80	357213.8	51.8	2734.4	1098.4	2734.4	39.91	10
	4	523.36	354008.9	0.7	5990.4	1202.6	5990.4	1668.06	10

表 5: 疎-ソフト（$vgap = 5$）

n	K	$\overline{\theta}^S$	#fix0	#fix1	#unfix	#row	#col	CPU	#solved
200	4	490.50	39523.5	50.7	425.8	302.6	425.8	0.20	7
	16	490.41	39533.1	53.3	413.6	309.4	413.6	0.36	3
400	4	495.36	158574.7	39.5	1385.8	725.0	1385.8	1.42	10
	16	495.35	158585.4	40.3	1374.3	735.4	1374.3	1.55	10
600	4	496.70	357137.0	24.9	2838.1	1154.2	2838.1	4.65	10
	16	496.69	357212.8	26.7	2760.5	1162.6	2760.5	5.34	10

表 6: 疎-ハード（$vgap = 5$）

n	K	$\overline{\theta}^H$	#fix0	#fix1	#unfix	#row	#col	CPU	#solved
200	4	497.96	39668.8	110.5	220.7	183.0	220.7	0.14	7
	16	496.74	39637.4	96.0	266.6	224.0	266.6	0.27	3
400	4	499.27	159008.2	108.3	883.5	587.4	883.5	0.90	10
	16	499.09	159004.2	107.1	888.7	601.8	888.7	1.05	10
600	4	499.51	357990.3	77.6	1932.1	1048.8	1932.1	2.99	10
	16	499.34	358069.5	88.7	1841.8	1038.6	1841.8	3.64	10