

高性能近似アルゴリズムの系統的設計法とその応用に関する研究

研究代表者 研究員 浅野孝夫 (理工学部情報工学科)

共同研究者 研究員 築山修治 (理工学部電気電子情報通信工学科)

1 はじめに

情報ネットワークや VLSI の物理設計等で生じる自然な問題は NP-困難であることが多く、厳密解を求めるのは長い計算時間を要する。そこで近似解を求めて利用することになるが、その際重要になるのが解の品質である。厳密解に匹敵する高品質な解を求める研究が近似アルゴリズムの研究であるが、最近数計画法に基づく系統的設計法の有用性が注目を浴びてきている。一般に NP-困難な問題の多くは整数計画問題をして定式化できる。その整数条件を外して線形計画問題や半正定値計画問題に緩和して解き、その最適解の値を元の整数計画問題の最適解の値の下界あるいは上界として用いて、解の品質を保証するというものが、数計画法に基づく近似アルゴリズム設計法である。緩和問題にすることにより、数計画法の双対理論に基づいた手法が、近似アルゴリズムでも適用可能になり、従来の近似性能が最近大幅に改善されてきている。

本研究では、上記の研究背景に基づいて、高性能近似アルゴリズムの系統的設計法の有用性を明らかにすることを目的としている。この目的を達成するための研究を実行してきたが、今回は、施設配置問題に対して代表的なアルゴリズムの実際の性能評価を与える。

2 施設配置問題

入力として、 n_f 個の開設候補の施設集合 F と n_c 人の利用者集合 C からなる完全 2 部グラフ ($n = n_f + n_c$, $m = n_f \times n_c$)、施設 i の正整数の開設コスト f_i 、施設 i と利用者 j との間の接続コスト c_{ij} (非負対称で三角不等式を満たす) が与えられる。このとき、適切に施設を開設し、すべての利用者を開設した施設に接続するが、総コスト (開設コストと接続コストの合計) が最小になるように施設の開設と接続を求める問題が施設配置問題である。

図 1, 2 はこの問題の入力例と出力例 (最適解) である (開設コスト f_i は一定、施設数 $n_f = 100$, 利用者数 $n_c = 100$, 接続コストは 2 点間のユークリッド距離)。なお、 \square は開設候補の施設 (図 2 では未開設施設), \bullet は開設された施設, \circ は利用者, そして実線分は接続を表している。

施設配置問題は NP-困難であり、1997 年に初めての定数近似アルゴリズム [10] が提案されて以来、様々な手法

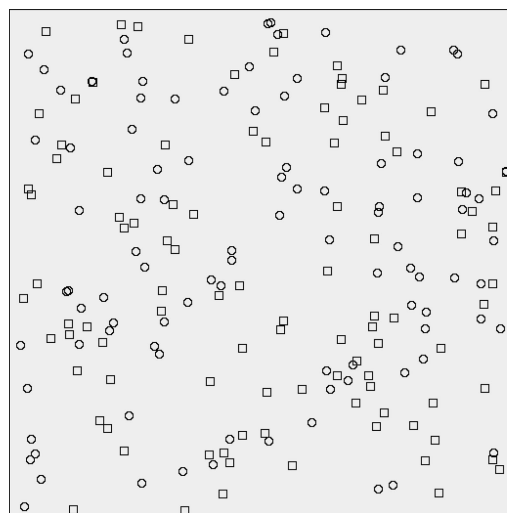


図 1 入力例

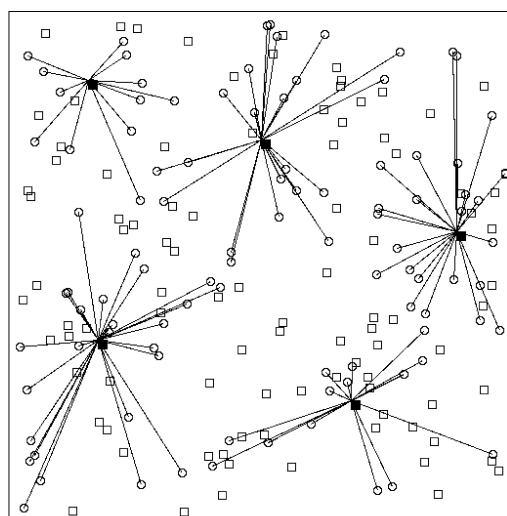


図 2 出力例

に基づく近似アルゴリズムが提案されている。これを表 1 にまとめている。本稿では、Mahdian-Markakis-Saberi-Vazirani の 1.861-近似アルゴリズム [5], 1.861-近似アルゴリズムの改善である Jain-Mahdian-Saberi の 1.61-近似アルゴリズム [4], 1.61-近似アルゴリズムを用いて初期解を求める Mahdian-Ye-Zhang の 1.52-近似アルゴリズム [9], そして 1.52-近似アルゴリズムを局所改善したもの (以下、改善案という) の 4 つアルゴリズムを C++ 言語で実装して得られた実験的性能評価を与える。

表 1 これまでの主な研究

年	近似率	主な手法	文献
1997	3.16	LP 丸め法	[10]
1998	2.408	LP 丸め法, 貪欲改善法	[3]
1998	1.736	LP 丸め法	[2]
1998	5+ ϵ	局所探索法	[7]
1999	3	主双対法	[5]
1999	1.853	主双対法, 貪欲改善法	[1]
2001	1.861	双対フィット法	[8]
2001	1.61	双対フィット法	[4]
2002	1.582	LP 丸め法	[11]
2002	1.52	貪欲改善法	[9]

3 LP 緩和と双対問題

表 1 の代表的施設配置アルゴリズムは, いずれも整数計画問題 (IP) としての定式化とその LP 緩和と双対問題を用いているので, それを説明する。

施設 i に利用者 j が接続している ($x_{ij} = 1$) かいない ($x_{ij} = 0$) かを表す変数 $x_{ij} \in \{0, 1\}$ と, 施設 i が開設している ($y_i = 1$) かいない ($y_i = 0$) かを表す変数 $y_i \in \{0, 1\}$ を用いて施設配置問題は以下のように IP として定式化できる。

$$\begin{aligned}
\min \quad & \sum_{i \in F} f_i y_i + \sum_{i \in F} \sum_{j \in C} c_{ij} x_{ij} \\
\text{s.t.} \quad & \sum_{i \in F} x_{ij} = 1 \quad (\forall j \in C) \\
& x_{ij} \leq y_i \quad (\forall i \in F, j \in C) \\
& x_{ij} \in \{0, 1\} \quad (\forall i \in F, j \in C) \\
& y_i \in \{0, 1\} \quad (\forall i \in F)
\end{aligned} \tag{1}$$

この定式化 IP(1) で, 制約式 $x_{ij} \leq y_i$ は, 利用者 j が接続されている施設 i は必ず開設されなければならないことを示している。また, 制約式 $\sum_{i \in F} x_{ij} = 1$ は利用者 j がどれかの施設に必ず接続されなければならないことを示している。したがって, 目的関数 $\sum_{i \in F} f_i y_i + \sum_{i \in F} \sum_{j \in C} c_{ij} x_{ij}$ は, 開設コストと接続コストの合計になる。

IP(1) の整数制約を $0 \leq x_{ij} \leq 1$ と $0 \leq y_i \leq 1$ に緩和すると, $x_{ij} \leq 1$ と $y_i \leq 1$ は冗長であるので, IP(1) の LP 緩和は以下のように書ける。

$$\begin{aligned}
\min \quad & \sum_{i \in F} f_i y_i + \sum_{i \in F} \sum_{j \in C} c_{ij} x_{ij} \\
\text{s.t.} \quad & \sum_{i \in F} x_{ij} = 1 \quad (\forall j \in C) \\
& x_{ij} \leq y_i \quad (\forall i \in F, j \in C) \\
& x_{ij} \geq 0 \quad (\forall i \in F, j \in C) \\
& y_i \geq 0 \quad (\forall i \in F)
\end{aligned} \tag{2}$$

以下は LP 緩和 (2) の双対問題である。

$$\begin{aligned}
\max \quad & \sum_{j \in C} \alpha_j \\
\text{s.t.} \quad & \sum_{j \in C} \beta_{ij} \leq f_i \quad (\forall i \in F) \\
& \alpha_j - \beta_{ij} \leq c_{ij} \quad (\forall i \in F, j \in C) \\
& \alpha_j \geq 0 \quad (\forall j \in C) \\
& \beta_{ij} \geq 0 \quad (\forall i \in F, j \in C)
\end{aligned} \tag{3}$$

この双対変数に対しては, α_j は利用者 j の全負担費用で, β_{ij} は施設 i を開設するときの j の貢献分 (負担額) である。すなわち $\beta_{ij} = \max(\alpha_j - c_{ij}, 0)$, と考えることができる。

4 1.861 近似アルゴリズム [8]

図 3 の Mahdian-Markakis-Saberi-Vazirani の 1.861-近似アルゴリズム [5] は主双対法に基づいているといえる。アルゴリズムは, 双対問題の解 (α, β) を時刻ともに更新しながら, 主問題の実行可能整数解 (施設配置問題の解) (x, y) を求める。得られる双対問題の解 (α, β) は, 必ずしも双対問題の実行可能解ではない。そこで, 双対フィット法を用いて解析をする。すなわち, 双対問題の解 (α, β) をある定数 $R > 1$ 倍縮小して $(\alpha/R, \beta/R)$ が実行可能解になるようにする。この縮小する値 R が近似率になる。

未接続の利用者の費用 α_j をすべて同じ割合で増加しながら, $\beta_{ij} = \max\{\alpha_j - c_{ij}, 0\}$ と定め, 以下の (a), (b) を繰り返す。すべての利用者が接続されたら終了とする。

- (a) if (ある未開設施設 i とある未接続の利用者 j に対して, $\sum_j \beta_{ij} = f_i$) then
施設 i を開設し, $\alpha_j \geq c_{ij}$ であるすべての未接続の利用者 j を施設 i に接続する。
- (b) if (ある開設施設 i とある未接続の利用者 j に対して, $\alpha_j = c_{ij}$) then
利用者 j を施設 i に接続する。

図 3 1.861-近似アルゴリズム

このアルゴリズムの (a) は, 幾人かの未接続の利用者による未開設施設 i への貢献の合計が i の開設コストと等しくなったら, i を開設として, i に貢献している未接続の利用者は i に接続することを意味する。(b) は, 未接続の利用者 j の全費用が j とある開設施設との接続コストに等しくなったら, j をこの施設に接続することを意味する。また, (a), (b) から一度接続された利用者の接続は不変であることがわかる。

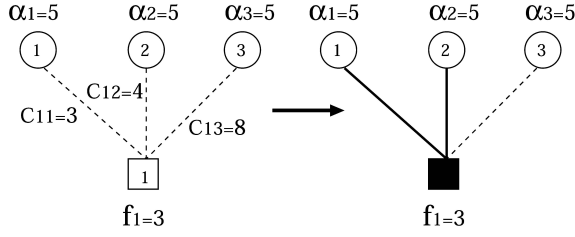


図4 アルゴリズムの (a) の実行

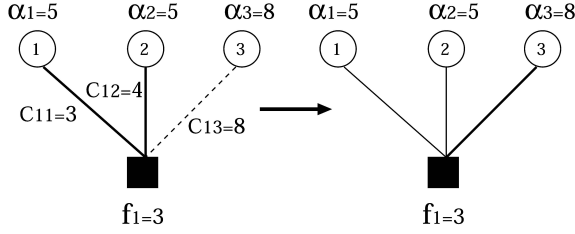


図5 アルゴリズムの (b) の実行

1つの入力例に対する実行 (a) を図4に、実行 (b) を図5に示す。なお、以下のアルゴリズムの実行例を示す図において、左が実行前、右が実行後を表していて、明記していない接続コストは、距離の性質を満たす大きい値としている。

5 1.61-近似アルゴリズム [4]

1.861-近似アルゴリズムでは一度決めた接続は不変であったが、総コストが減少するなら接続を変更するとしたのが Jain-Mahdian-Saberi の 1.61-近似アルゴリズムである (図6)。アルゴリズムの流れは 1.861-近似アルゴリズムと同じであるが、利用者 j から施設 i への貢献 β_{ij} の定義を以下のように変更している。

- if (利用者 j は未接続) then
 $\beta_{ij} = \max(\alpha_j - c_{ij}, 0)$ とする。

未接続の利用者の費用 α_j をすべて同じ割合で増加しながら、利用者 j が未接続なら $\beta_{ij} = \max\{\alpha_j - c_{ij}, 0\}$ と定め、利用者 j は既に施設 i' と接続なら $\beta_{ij} = \max\{c'_{ij} - c_{ij}, 0\}$ と定めて、以下の (a), (b) を繰り返す。すべての利用者が接続されたら終了とする。

- (a) if (ある未開設施設 i とある未接続の利用者 j に対して、 $\sum_j \beta_{ij} = f_i$) then
施設 i を開設し、 $\alpha_j \geq c_{ij}$ であるすべての未接続の利用者 j を施設 i に接続する。
- (b) if (ある開設施設 i とある未接続の利用者 j に対して、 $\alpha_j = c_{ij}$) then
利用者 j を施設 i に接続する。

図6 1.61-近似アルゴリズム

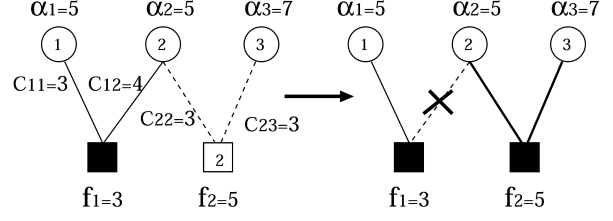


図7 接続を変更する実行

(初期解) スケーリングパラメータを $\delta = 1.504$ とする。このとき、開設コストを一律に δ 倍した問題に対して、図6の 1.61-近似アルゴリズムを適用する。

(貪欲改善) 開設コストを一律に $\frac{1}{\delta}$ 倍した問題 (つまりもとの問題) に対して、総コストが減少する ($gain(i) > 0$ である) かぎり以下を繰り返す。

- if (ある未開設施設 i の $\frac{gain(i)}{f_i}$ が最大) then
 i を開設して、利用者は最も近い開設施設に接続を変更する。

図8 1.52-近似アルゴリズム

- if (利用者 j は既に施設 i' と接続) then
 $\beta_{ij} = \max(c'_{ij} - c_{ij}, 0)$ とする。

これから、既に施設に接続している利用者においても貢献を考えることができる。これに伴い、1.861 近似のアルゴリズムの (a) の実行において、既に接続している利用者 j も考える。すなわち、 j が既に別の施設 i' に接続しているなら、 j の接続を i' から i に変更する (図7)。この操作では $\beta_{i'j}$ は不変であり、 β_{ij} のみが変わる。なお、接続を変更することで i' へ接続している利用者がいなくなったら、 i' は未開設とする (これ以降のアルゴリズムでも同様)。

6 1.52-近似アルゴリズム [9]

図8の Mahdian-Ye-Zhang のアルゴリズム [9] は、コストスケリング後に Jain らの 1.61-近似アルゴリズムを用いることで、貢献を多く集めることのできる経済的な施設のみを開設しておき、その後の貪欲改善によりできるだけ多く総コストを減少できる未開設施設を開設するアルゴリズムである。

このアルゴリズムにおいて、総コストが減少する値として $gain$ を用いる。もし未開設施設 i を開設することで接続コストが c から c' になるなら、 $gain(i) = c - c' - f_i$ と定義する。

貪欲改善の条件において、 $gain(i)$ ではなく $\frac{gain(i)}{f_i}$ を用

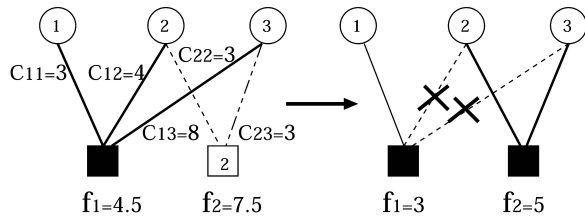


図 9 貪欲改善の実行

いるのは, $gain(i)$ が大きいだけでなく f_i も小さい施設を開設したいためである。

図 9 は貪欲改善の実行例である (左は初期解を求めた直後を表し, 便宜上 $\delta = 1.5$ としている)。

7 改善案 (1.52 近似の局所改善)[6]

九里-浅野は, 1.52-近似アルゴリズムと同じ方法で初期解を求め, 1.52-近似アルゴリズムの貪欲改善において施設を開設することだけではなく開設施設を削除する (未開設にする) ことも考慮するヒューリスティックを提案している [6] (この考え方は [1] でも提案されている)。すなわち, もし i が未開設なら開設し, そうではなく i が開設なら削除する。そしてその後に, i 以外の開設施設を削除すると総コストが減少するなら, 繰り返し削除することを考える。なお, このとき最も総コストが減少する値を, $gain(i)$ と定義する。九里-浅野の貪欲改善では, このような $gain$ をすべての施設で考え, $gain(i) > 0$ であるかぎり, $\frac{gain(i)}{f_i}$ が最大の i に対して開設か削除, もしくはその両方を実行する。

8 計算機実験 [6]

この節は, 九里-浅野の予備的な計算機実験からの報告 [6] をまとめたものである。1.52-近似アルゴリズムと改善案のパラメータ δ の値は, δ を変化した場合の性能を観察した以下の実験に基づいて, 近似率が最も良くなる $\delta = 1.504$ を用いている。

入力データとして, 施設と利用者を座標 $(0, 0)$ から $(500, 500)$ の間にそれぞれランダムに 100 個ずつ ($n_f = n_c = 100$) 発生し, 開設コストはすべての施設で一定 ($f_i = 1000$), 接続コストは施設と利用者のユークリッド距離としたものを用いる。図 10 の近似率は, それぞれの δ において実験を 20 回行ったときの平均値である (横軸は δ の値を表し, 縦軸は $\delta = 1.504$ の近似率に対する比率 (倍率) の平均を表している)。

8.1 実験的性能評価

上記の 4 つの近似アルゴリズムに対して様々な計算機実

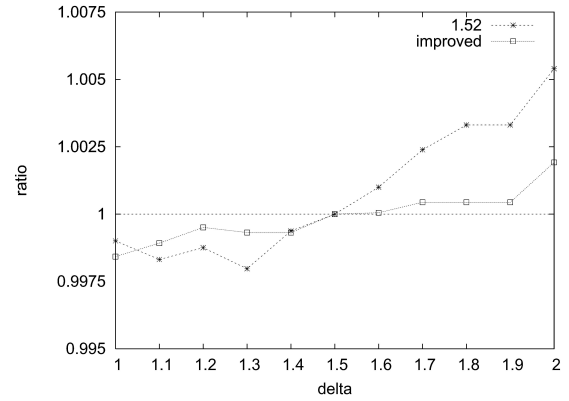


図 10 パラメータ δ による性能評価

験を行い, 性能を観察した。最適解の値は, LP 緩和 (2) を線形計画ソフト XPRESS-MP で解いて得られた目的関数値を用いている。入力データは,

- 施設と利用者は, 座標 $(0, 0)$ から $(500, 500)$ の間でランダムに 100 個ずつ発生,
- 開設コストは, 1 から 1000 の間のランダムな正整数,
- 施設と利用者の接続コストは, 2 点間 (施設と利用者の間) のユークリッド距離

としたものを用い, この入力データ (以下, 標準入力データ) の一部を変更することで様々な実験を行った。以下の図の縦軸は, それぞれの横軸の値で実験を 20 回行ったときの近似率の平均値を表している。

実験 1: 施設数と利用者数による影響

- 施設数 n_f , 利用者数 n_c を増加する場合

(標準入力データの n_f, n_c をそれぞれ同じ割合で増加した場合の性能評価)

- 利用者数 n_c のみを増加する場合

(標準入力データの n_f を $n_f = 100$ と固定して, n_c のみを増加した場合の性能評価)

1.861-近似アルゴリズムは利用者数 n_c が多い場合, 性能がわずかに悪くなるが, その他のアルゴリズムは施設数と利用者数による影響を受けないことが観察された。

実験 2: 開設コストと接続コストによる影響

- 開設コスト f_i を増加する場合

(標準入力データの開設コスト f_i を, ランダムではなくすべての施設で一定とし, この値を増加した場合の性能評価) 結果を図 11 に示している (横軸は開設コストを表している)。

- 接続コストを増加する場合

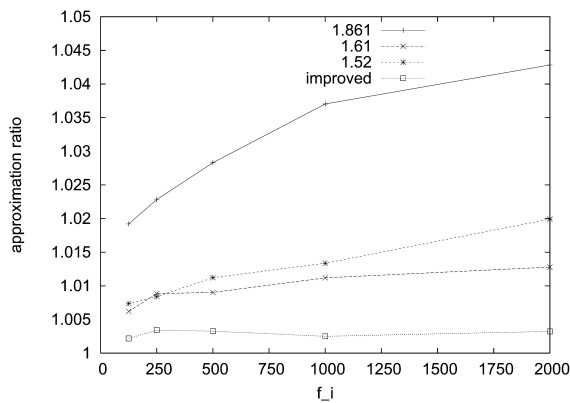


図 11 開設コスト f_i を増加した結果

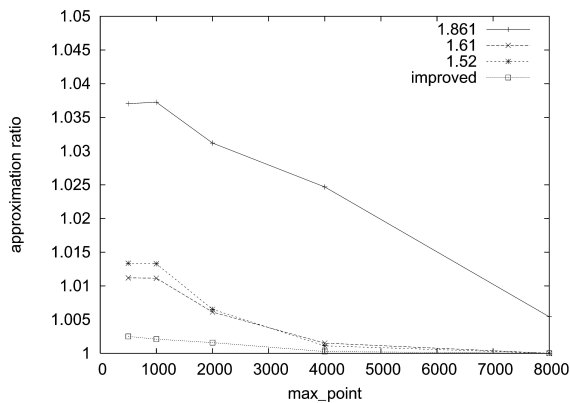


図 12 接続コストを増加した結果

(標準入力データの開設コストを一定 ($f_i = 1000$) とし、最大座標 (x, x) を $x = 500$ ではなくさらに増加した場合の性能評価) この結果を図 12 に示している (横軸は最大座標を表している)。

図 11, 12 より、開設コストによる影響は接続コストの影響に比べて大きいことが観察された。また、開設コストを一定にすると改善案も性能が悪くなることが観察された。これは、初期解で開設される施設の多くが最適な開設施設とは異なるため、その後の貪欲改善により最適な開設施設に改善されにくいことが原因であると思われる。

実験 3: 接続コストを最短パスの長さとする場合の影響 (接続コストが距離の性質を満たすが、ユークリッド距離ではない場合)

施設と利用者の 2 部グラフのそれぞれの辺に 0 から 707 ($500 \times \sqrt{2} \approx 707$) の間のランダムな正整数を与え、このときの 2 部グラフの最短パス長を接続コストとした。このように標準入力データの接続コストを変更し、 n_f, n_c を同じ割合で増加した場合の性能評価を行った。この結果、1.861 近似はかなり性能が悪くなることが観察された。こ

れは、1.861 近似では接続を変更できないことが大きく影響したためであり、他のアルゴリズムでは接続を変更することで近似率の良い解が求まるとされる。

9 計算機実験のまとめ

計算機実験では、理論的な近似率よりもかなり良い近似率を得ることが観察できた。また、削除を加えた改善案は、他のアルゴリズムよりもかなり性能が良く、最適解もしくはそれにかなり近い解を求めることが観察できた。

今後は、最悪の近似率となるインスタンスを考え、このインスタンスに対しても性能評価を行うことが課題であると思われる。また、1.463 近似より良いアルゴリズムは存在しないと考えられている [3] ため、新たな手法のアルゴリズムの提案により近似率を 1.463 に近づけることも今後の重要な課題である。

参 考 文 献

- [1] M. Charikar and S. Guha: Improved combinatorial algorithms for facility location and k -median problems, *FOCS*, 1999, pp. 378–388.
- [2] F.A. Chudak: Improved approximation algorithms for uncapacitated facility location, *IPCO*, 1998, pp. 180–194.
- [3] S. Guha, S. Khuller, Greedy strikes back: Improved facility location algorithms, *SODA*, 1998, pp. 649–657.
- [4] K. Jain, M. Mahdian and A. Saberi: A new greedy approach for facility location problems, *STOC*, 2002, pp. 731–740.
- [5] K. Jain and V.V. Vazirani: Primal-dual approximation algorithms for metric facility location and k -median problems, *FOCS*, 1999, pp. 2–13.
- [6] 九里史郎, 浅野孝夫: 代表的施設配置近似アルゴリズムの実験的性能評価, 情報処理学会アルゴリズム研究会報告 AL88-2, pp.9–16.
- [7] M. Korupolu, G. Plaxton, and R. Rajaraman: Analysis of a local search heuristic for facility location problems, *SODA*, 1998, pp.1–10.
- [8] M. Mahdian, E. Markakis, A. Saberi and V.V. Vazirani: A greedy facility location algorithm analyzed using dual fitting, *5th International Workshop on Randomization and Approximation Techniques in Computer Science*, Vol. 2129, 2001, pp.

127–137.

- [9] M. Mahdian, Y. Ye and J. Zhang: Improved approximation algorithms for metric facility location problems, <http://www.mit.edu/~mahdian/pub.html>, 2002.
- [10] D.B. Shmoys, E. Tardos and K.I. Aardal: Approximation algorithms for facility location problems, *STOC*, 1997, pp. 265–274.
- [11] M. Sviridenko: An Improved Approximation algorithm for the metric uncapacitated facility location problems, *IPCO*, 2002, pp. 230–239.