頂点被覆へのリスト減少法の解析に関する一考察

飯田浩志*

概 要

頂点被覆は, $\mathcal{NP}$ 困難な組合せ最適化問題ゆえに, 多項式時間では解き得ないと考えられている．他方, 頂点被覆にはいくつかの近似解法が提案されている．これら近似解法には, 大きく分けて二種類, 即ち, 与えられたグラフの最大次数を Δ として近似率 $O(\log\Delta)$ を与えるものと近似率 2 を与えるものがある．近年, この頂点被覆に対するある近似解法が, 近似率 $\sqrt{\Delta}/2+3/2$ を与えることが示された．ここでは, その近似率を導出した解析に若干の修正を加えて, より良い近似率を導くことを試みる．

キーワード 組合せ最適化, 集合被覆, 頂点被覆, 近似アルゴリズム, 近似率

集合被覆 (Set Cover, 以下では SC と略す) は, 全体集合 U の元をその要素とする部分集合族, i.e. $S_1, S_2, \ldots, S_n$ $(\cup_{i=1}^n S_i = U)$ の内, すべての U の元をカバーし, かつそれらの重み (i.e., $c(S_i)$; $c(\cdot) : S_i \to \mathbb{Q}^+$) の総和を最小にする組を求める組合せ最適化問題である．Yu [8] が, max-min 0-1 ナップサック問題の複雑さを示すために SC からの帰着可能性を示しているように, SC は強 $\mathcal{NP}$ 困難である．

SC において, カバーされる要素各々が高々2 つの部分集合にしか属さない場合を, 重み付き頂点被覆 (weighted vertex cover) という．特に, すべての部分集合の重みが同一のとき, cardinality vertex cover, または単に, 頂点被覆 (以下, VC と略す) と呼ばれる．VC は, $\mathcal{NP}$ 困難である (例えば, Korte and Vygen [6, pp.359-60])．VC への近似解法における近似率 (近似値/最適値の上界, 精度保証ともいう) では, 次数 (接続する枝の本数) の大きい頂点から考えるという通常の貪欲法を用いた場合のそれは, 与えられたグラフ (本稿では, グラフといえば無向グラフを指すものとする) の最大次数を Δ とすれば, $H(\Delta) = 1 + 1/2 + \cdots + 1/\Delta$ であることが知られている (Chvátal [3])．

Goldschmidt et al [4] は, 貪欲法を改良することによって, 近似率 $H(\Delta) - 1/6$ を得た．提案されたアルゴリズムは, すべての頂点の次数が 2 以下になるまで貪欲法を用いた後, 未カバーの枝を点, 次数 2 の点を枝にそれぞれ対応させたグラフの最大マッチング問題 (与えられたグラフにおいて, 共通の端点を持たない枝の集合 (マッチングという) の要素数を最大にする) を exact に解いている．

また, 与えられたグラフの極大マッチング M を求め, 選ばれた枝の端点すべてを集めた集合 (頂点数は $2|M|$) を返すアルゴリズムが VC への 2-近似アルゴリズムになることは, 容易に分かる (M に含まれる枝の端点全部をもってしてもカバーされない枝があるとしたら, その枝は M に含められるので, M の極大性に反する)．M に属するすべての枝をカバーするのに最低でも $|M|$ 個の頂点が必要であることから $|M| \leq \mathrm{Opt}$ (Opt は, 最適値を表す)．したがって, $2|M| \leq 2 \cdot \mathrm{Opt}$ が成立する．特に, すべての枝がバラバラ $(\Delta = 1)$ の場合は, この近似率が tight である (近似率を表す式における等号を成立させる具体例 (あるいはその列) が存在する) ことを如実に表している (Korte

*小樽商科大学商学部 〒047-8501 北海道小樽市緑 3 丁目 5-21; E-mail: oggi@res.otaru-uc.ac.jp

and Vygen [6, p.381] あるいは Vazirani [7, p.3] を参照されたい). また, Bar-Yehuda and Even [2] のアルゴリズムも近似率 2 である.

加えて, Hochbaum [5] の提案した解法も, VC への 2-近似アルゴリズムである. Hochbaum [5] は, VC の双対問題である最大マッチング問題の LP 緩和問題を解いた際に, 接続する枝各々に対応する変数の値の総和が 1 である頂点の集合を近似解としている. この考え方は, 先の Bar-Yehuda and Even [2] のアルゴリズムと同じく, 主相補スラック条件から得られるものである. これと同じ結果は, 主問題の LP 緩和問題を解いた際に, 対応する変数の値が 1/2 以上である頂点を選択することでも得られる [7, Chapter 14].

VC に対して, Avis and Imamura [1] は, list-decreasing heuristic の近似率が高々$\sqrt{\Delta}/2+3/2$ であることを示した. list-decreasing heuristic(以下, LDH と略す) とは, 頂点を次数の降順に並べた後, まだカバーされていない枝が接続する頂点をこの順に選択していくというものであり, 最初に決めた頂点の並び (次数の降順) を動かさないところがユニークである. 通常の貪欲法では, 頂点が選択されて枝がカバーされるごとに, 残りの頂点の中から, 未カバーの枝に関して次数最大のものを動的に選択する. また, Avis and Imamura [1] は, 最初にリストを固定する手法では, 近似率 $\sqrt{\Delta}/2$ が限界であることも示している.

Avis and Imamura [1] の解析は, 大雑把に言えば, 次のようになる. 頂点数 n, 枝数 m のグラフ $G=(V,E)$ が与えられる. 各頂点 i の次数を d_i と書き, $\Delta=d_1\geq d_2\geq\cdots\geq d_n$ とする. また, $p:=\min\{j\,|\,\sum_{i=1}^{j}d_i\geq m\}$ とする. 以下では, LDH が選択する頂点の番号の集合を C_D と書く. 頂点 i $(i>p)$ を C_D に入れるにあたり, カバーされる枝 (少なくとも一本はある) の内一本の重みを $1/d_i$ とし, 残りすべての枝の重みを $1/\Delta$ とする. 以降, 重み $1/d_i$ を付された枝を, special な枝と呼ぶ. Special な枝の総数 (C_D の中で, p より大きいものの個数) を s とする.

補題 (Avis and Imamura, 2007) $d_1,\ldots,d_s$ が正の整数で, それらの和が m 以下ならば,

$$\sum_{i=1}^{s}\frac{1}{d_i}\geq\frac{s^2}{m}.$$

今, $\sum_{i=1}^{n}d_i=2m$ であることから, $\sum_{i=p+1}^{n}d_i\leq m$. したがって補題から, special な枝に付された重みの総和は s^2/m 以上である. また, 最適解を構成する頂点の集合における次数の総和は m 以上であることから $p\leq\mathrm{Opt}$, かつ $|C_D|\leq p+s$ から, $s\geq|C_D|-\mathrm{Opt}$ を得る. 以上の準備の上で, 弱双対性を表す式, 即ち, $\mathrm{Opt}\geq\sum_{e\in E}y_e$ を評価することで $|C_D|/\mathrm{Opt}\leq\sqrt{\Delta}/2+3/2$ を得ている.* ここに, y_e は, 各枝 $e\in E$ に付された重みを表す.

しかしながら, C_D に頂点 i を入れる時, それが p より大きいか否かにかかわらず常に, カバーされる枝の内一本の重みを $1/d_i$ とし, 残りすべてを $1/\Delta$ としても, 双対問題の有理解は依然として可能解である.† この操作によって, $\sum_{e\in E}y_e$ の値をより大きくできることから, より精密な解析になることが期待される. 以下では, この方針に沿って, LDH の解析を試みる.

まず, 明らかに $\sum_{i=1}^{p}d_i\leq m-1+\Delta$ が成立する. C_D の中で p 以下のものの個数を k とすると, 補題から, C_D に頂点 i $(i\leq p)$ を入れる際に枝に付した重み $1/d_i$ の総和 (枝 k 本分) は $k^2/(m-1+\Delta)$

*Avis and Imamura [1] の証明中にあるように, 正確には近似率 $\Delta/(2\sqrt{\Delta}-1)+1$ を得ている. $\sqrt{\Delta}/2+3/2$ にすることで, $1/4-1/(8\sqrt{\Delta}-4)$ だけ損をしている.

†ある枝の端点の次数を k,l $(k\geq l)$ とすると, その枝には $1/k$ の重みを付すことができる. LDH では, 次数の大きい頂点から考えるということが, 決定的に効いている.

以上である．この時, Avis and Imamura [1] の解析とほぼ同様にして,

$$
\begin{aligned}
\mathrm{Opt} &\geq \sum_{e\in E} y_e \\
&\geq \frac{m-k-s}{\Delta} + \frac{k^2}{m-1+\Delta} + \frac{s^2}{m} \\
&= \frac{(m-1+\Delta)+m+1-\Delta-2|C_D|}{2\Delta} + \frac{k^2}{m-1+\Delta} + \frac{s^2}{m} \\
&\geq 2\sqrt{\frac{k^2}{2\Delta}} + 2\sqrt{\frac{s^2}{2\Delta}} + \frac{1-\Delta-2|C_D|}{2\Delta} \qquad (a+b\geq 2\sqrt{ab}) \\
&= \left(\frac{\sqrt{2}}{\sqrt{\Delta}} - \frac{1}{\Delta}\right)|C_D| + \frac{1-\Delta}{2\Delta}
\end{aligned}
$$

整理して,

$$
|C_D| \leq \frac{\Delta}{\sqrt{2\Delta}-1}\mathrm{Opt} + \frac{\Delta-1}{2\sqrt{2\Delta}-2} < \frac{\Delta}{\sqrt{2\Delta}-1}\mathrm{Opt} + \frac{\sqrt{2\Delta}+1}{4}
$$

を得る．Avis and Imamura [1] の得た近似率 $\sqrt{\Delta}/2+3/2$ の主要項は $\sqrt{\Delta}/2$ であり, ここで得た結果のそれは $\sqrt{\Delta}/\sqrt{2}$ なので, Δ が大きいほど, Avis and Imamura [1] の得た近似率の方が有利である．実際, 数値実験では, ここで得た近似率の方が小さくなるのは, Δ が 2 から 19 までの場合に限られる (巻末の付録を参照).

以上, あまり良い結果は得られなかった．原因としては, 式 $\sum_{i=1}^{p} d_i \leq m-1+\Delta$ の評価の甘さが挙げられよう．特に Δ が大きいほど, この評価式のギャップは大きい．一つ, $\Delta=1$ の時には, この甘さが解消される (しかもこの時, LDH は最適解を与える) にもかかわらず, ここで得た近似率が $\Delta/(\sqrt{2\Delta}-1)=\sqrt{2}+1$ なのは奇妙である．Avis and Imamura [1] の得た近似率も, $\Delta=1$ の時には $\sqrt{\Delta}/2+3/2=2$ であり, これは, 解析を進める過程自体に潜む問題ではないかと推察される．

最後に, 頂点被覆の近似率は, 今のところ 2 が最小であり, これより小さくできるか否かは, 未解決の問題である [7, p.334]. 本稿に興味を持ち, この分野に参入する人が多少なりとも増えることを祈念して, 結びとしたい．

参考文献

[1] D. Avis and T. Imamura, A list heuristic for vertex cover. *Operations Research Letters* **35**(2) 201-4 (Mar 2007).

[2] R. Bar-Yehuda and S. Even, A linear-time approximation algorithm for the weighted vertex cover problem. *Journal of Algorithms* **2**, 198-203 (1981).

[3] V. Chvátal, A greedy heuristic for the vertex cover problem. *Mathematics of Operations Research* **4**, 233-5 (1979).

[4] O. Goldschmidt, D.S. Hochbaum and G. Yu, A modified greedy heuristic for the set covering problem with improved worst case bound. *Information Processing Letters* **48**(6) 305-10 (20 Dec 1993).

[5] D.S. Hochbaum, Approximation algorithms for the set covering and vertex cover problems. *SIAM Journal on Computing* **11**(3) 555-6 (Aug 1982).

[6] B. Korte and J. Vygen, *Combinatorial Optimization: Theory and Algorithms (Algorithms and Combinatorics 21)*, 3rd ed., Springer 2005 (4th ed. has already been shipped in 2007).

[7] V.V. Vazirani, *Approximation Algorithms*, corrected 2nd printing, Springer 2003.

[8] G. Yu, On the max-min 0-1 knapsack problem with robust optimization applications. *Operations Research* **44**(2) 407-15 (1996).

付録: 以下は, PowerBook G4 (Mac OS X 10.4.11) 上で行った.

```
$ cat dist.c
#include <stdio.h>
#include <math.h>

main(ac, av)
char *av[];
{
        int d;

        for (d=1; d<=20; d++)
                printf("d = %2d: %g\n", d,
                        (sqrt((double)d)+3)/2 - d/(sqrt(d*2.0)-1));
}
$ cc dist.c
$ ./a.out
d =  1: -0.414214
d =  2: 0.207107
d =  3: 0.296332
d =  4: 0.312327
d =  5: 0.305658
d =  6: 0.28978
d =  7: 0.269676
d =  8: 0.247547
d =  9: 0.224484
d = 10: 0.201067
d = 11: 0.177618
d = 12: 0.154322
d = 13: 0.131285
d = 14: 0.108568
d = 15: 0.0862026
d = 16: 0.0642043
d = 17: 0.0425776
d = 18: 0.0213203
d = 19: 0.000426065
d = 20: -0.0201142
$
```