劣モジュラシステムの分割問題に関する研究

平山　剛史

（東京大学大学院情報理工学系研究科数理情報学専攻　現所属：東芝ソリューション（株））
指導教員：牧野和久 准教授

1. グラフの k カット問題と劣モジュラシステムの k 分割問題

グラフの k カット問題とは，重み付き無向グラフ $G=(V,E,c\colon E\to\mathbb{R}_+)$ が与えられたときに，重み最小の k カットを求める問題である．ここで，重み付き無向グラフ G の k カットは頂点集合 V の k 分割と定義され，k カット $\mathcal{P}_k=\{V_1,\ldots,V_k\}$ の重み $\Phi(\mathcal{P}_k)$ は，グラフ G のカット関数 $c\colon 2^V\to\mathbb{R}_+$ を用いて $\Phi(\mathcal{P}_k)=\sum_{i=1}^k c(V_i)$ で定められる．この問題は，巡回セールスマン問題を分枝カット法で解くときに用いる切除平面の生成，ネットワークの信頼性解析，VLSI の設計，グラフの強度計算など多くの応用があり，組合せ最適化分野における重要な問題の一つである．

劣モジュラシステムの k 分割問題とは，台集合 V と劣モジュラ関数（すなわち $f(X)+f(Y)\geq f(X\cap Y)+f(X\cup Y)$ $(\forall X,Y\subseteq V)$ が成り立つ）$f\colon 2^V\to\mathbb{R}$ の組である，劣モジュラシステム (V,f) が与えられたときに，重み最小の V の k 分割を求める問題である．ただし，k 分割 $\mathcal{P}_k=\{V_1,\ldots,V_k\}$ の重みは $f(\mathcal{P}_k)=\sum_{i=1}^k f(V_i)$ で定められる．グラフのカット関数 $c\colon 2^V\to\mathbb{R}_+$ は劣モジュラ性をもつので，グラフの k カット問題は，劣モジュラシステムの k 分割問題として記述可能である．劣モジュラシステムの k 分割問題は，グラフの k カット問題の拡張であるハイパーグラフの k カット問題なども含み，表現力が高い．また，良く知られているように，劣モジュラ性は組合せ最適化においては重要な性質であるため，劣モジュラシステムの k 分割問題は，応用上はもちろん理論的にも重要な問題である．

2. 先行研究

本節では，前節で導入した二つの問題の先行研究について述べる．

グラフの k カット問題は，k を入力に含めると NP 困難であることが 1988 年に Goldschmidt and Hochbaum[1] によって示され，さらに同じ論文により，$\mathrm{O}(n^{k^2/2-3k/2+4}T(n,m))$ 時間のアルゴリズムが与えられた．ただし，$T(n,m)$ は，頂点数が n，枝数が m であるグラフ中の最大フローを求めるために必要な計算時間である．このアルゴリズムは定数 k に対する最初の多項式時間アルゴリズムとなる．これより後，グラフの k カット問題に対するより高速なアルゴリズムがいくつか提案され，計算時間は Kamidoi, Yoshida and Nagamochi [2] により $\mathrm{O}(n^{4k/(1-1.71/\sqrt{k})-34}T(n,m))$ 時間，Xiao [9] により $\mathrm{O}(n^{4k-\lg k})$ 時間，Thorup [8] により $\tilde{\mathrm{O}}(mn^{2k-2})$ 時間と改善された．一般の k に対しては Thorup [8] のアルゴリズムがこれまでのところ最速である．一方で，小さな k に対しては，より高速なアルゴリズムがいくつも提案されている．Nagamochi and Ibaraki [4] は，$k=3,4$ に対する $\tilde{\mathrm{O}}(mn^k)$ 時間アルゴリズムを与え，Nagamochi, Katayama and Ibaraki [5] は，このアイデアを拡張することにより，$k=5,6$ に対する $\tilde{\mathrm{O}}(mn^k)$ 時間アルゴリズムを与えた．Yeh, Wang and Su [10] は上の二つのアルゴリズム中で用いるサブルーチンの列挙アルゴリズムを改善することによって計算時間が $\tilde{\mathrm{O}}(mn^{k-1})$ $(k=3,4,5,6)$ となることを示した．$k=3,4,5,6$ に対しては現在これらが最速である．

劣モジュラシステムの k 分割問題に対してはあまり多くの結果は知られていない．$k=2$ の場合は対称劣モジュラ関数最小化問題と等価であるので，Queyranne [7] のアルゴリズムによって多項式時間で解ける．$k=3$ に対しては，Okumoto, Fukunaga and Nagamochi [6] が最初の多項式時間アルゴリズムが与えた．しかし，$k\geq 4$ に対して多項式時間アルゴリズムが存在するかはこれまでのところ知られていない．

3. 本研究の成果

以下では，本研究における主結果を三つ述べる．

定理 3.1. $k=7,8$ に対するグラフの k カット問題は，$\tilde{\mathrm{O}}(mn^{k-1})$ 時間で解くことができる．ここで，n,m はそれぞれグラフの頂点数，枝数である．

$k=7,8$ に対するこの計算量は，これまでところ最

表 1 グラフの k カット問題における現在最良の計算時間

k：一般	$\tilde{\mathrm{O}}(mn^{2k-2})$	Thorup [8]
$k=2$	$\tilde{\mathrm{O}}(mn)$	Nagamochi-Ibaraki [3]
$k=3,4$	$\tilde{\mathrm{O}}(mn^{k-1})$	Nagamochi-Ibaraki [4], Yeh et al. [10]
$k=5,6$	$\tilde{\mathrm{O}}(mn^{k-1})$	Nagamochi et al. [5], Yeh et al. [10]
$k=7,8$	$\tilde{\mathrm{O}}(mn^{k-1})$	本研究
$k \geq 9$	$\tilde{\mathrm{O}}(2^{(k+1)(k+2)/2}mn^{2k-9})$	本研究

表 2 劣モジュラシステムの k 分割問題の計算複雑度

$k=2$	P	対称劣モジュラ関数最小化問題と等価
$k=3$	P	Okumoto et al. [6]
$k=4$	P	本研究
$k \geq 5$：定数	不明	
k：一般	NP 困難	Goldschmidt-Hochbaum [1]

速である．

定理 3.2. 整数 $k \geq 9$ に対するグラフの k カット問題は，$\tilde{\mathrm{O}}(2^{(k+1)(k+2)/2}mn^{2k-9})$ 時間で解くことができる．

この計算時間は，これまで最速である Thorup [8] の結果を，$k = \mathrm{O}((\log n)^{1/2-\epsilon})$ に対して改善する．

定理 3.3. 劣モジュラシステムの 4 分割問題は，$\mathrm{O}(n^7 F(n))$ 時間で解くことができる．ここで，n は台集合のサイズであり，$F(n)$ は劣モジュラ関数最小化問題を解くために必要な計算時間である．

この結果は，劣モジュラシステムの 4 分割問題に対する最初の多項式時間アルゴリズムを与えたことを意味する．

グラフの k カット問題の現在最良の計算時間，および，劣モジュラシステムの k 分割問題の計算複雑度を表 1, 2 に示す．

参考文献

[1] O. Goldschmidt and D. S. Hochbaum: A polynomial algorithm for the k-cut problem for fixed k, *Mathematics of Operations Research*, **19**, 24–37, 1994.

[2] Y. Kamidoi, N. Yoshida and H. Nagamochi: A deterministic algorithm for finding all minimum k-way cuts, *SIAM Journal on Computing*, **36**, 1329–1341, 2006.

[3] H. Nagamochi and T. Ibaraki: Computing edge connectivity in multigraphs and capacitated graphs, *SIAM Journal on Discrete Mathmatics*, **5**, 54–66, 1992.

[4] H. Nagamochi and T. Ibaraki: A fast algorithm for computing minimum 3-way and 4-way cuts, *Mathematical Programming*, **88**, 507–520, 2000.

[5] H. Nagamochi, S. Katayama and T. Ibaraki: A faster algorithm for computing minimum 5-way and 6-way cuts in graphs, *Journal of Combinatorial Optimization*, **4**, 151–169, 2000.

[6] K. Okumoto, T. Fukunaga and H. Nagamochi: Divide-and-conquer algorithms for partitioning hypergraphs and submodular systems, *Algorithmica*, **62**, 787–806, 2012.

[7] M. Queyranne: Minimizing symmetric submodular functions, *Mathematical Programming*, **82**, 3–12, 1998.

[8] M. Thorup: Minimum k-way cuts via deterministic greedy tree packing, *Proceedings of the 40th Annual ACM Symposium on Theory of Computing* (STOC), pp. 159–166, 2008.

[9] M. Xiao: An improved divide–and–conquer algorithm for finding all minimum k-way cuts, *Lecture Notes in Computer Science*, **5369**, 147–158, 2008.

[10] L.-P. Yeh, B.-F. Wang and H.-H. Su: Efficient algorithms for the problems of enumerating cuts by nondecreasing weights, *Algorithmica*, **56**, 297–312, 2010.