共正値計画問題に対する
単体細分アルゴリズムとその拡張

数理情報学専攻 48166228　平手 一成
指導教員　平井 広志 准教授

1　はじめに

$n \times n$ の実対称行列 A が $x^\top Ax \geq 0$ $(\forall x \in \mathbb{R}^n_+)$ を満たすとき，A は共正値であるという．ただし，$\mathbb{R}^n_+$ は n 次元非負実数空間を表す．最大クリーク数を求める問題や標準 2 次計画問題などの様々な組合せ最適化問題や 2 次計画問題が共正値計画問題で表せることが知られている．共正値計画問題を解くのは NP-hard であるため，近似解法が主に研究されている．また，ある行列が共正値であるかそうでないか判定する問題は，共正値判定問題と呼ばれ，co-NP-complete である．共正値計画問題を共正値判定を行うことで解く方法もさかんに研究されている．

本論文では，Bundfuss と Dür [1] が考案した単体細分を用いた共正値判定アルゴリズムおよび Sponsel ら [4] の拡張を元にした研究を行った．この先行研究について第 2 章で説明した後，第 3 - 5 章で本研究の成果を述べる．

2　先行研究

単体細分を用いた共正値判定アルゴリズムを考える上で，まず，標準単体と呼ばれる単体 $\Delta_{\mathrm{st}} = \{x \in \mathbb{R}^n_+ : x_1 + x_2 + \cdots + x_n = 1\}$ を用いて，実対称行列 $A \in \mathcal{S}$ が共正値である条件の $x^\top Ax \geq 0$ $(\forall x \in \mathbb{R}^n_+)$ の $\forall x \in \mathbb{R}^n_+$ の部分を $\forall x \in \Delta_{\mathrm{st}}$ に置き換える．

単体細分を用いた共正値判定アルゴリズムでは，この Δ_{st} を $\mathrm{int}\Delta^i \cap \mathrm{int}\Delta^j = \emptyset$ $(i \neq j)$ となるような $\{\Delta^1, ..., \Delta^m\}$ を用いて，

$$\Delta_{\mathrm{st}} = \bigcup_{i=1}^{m} \Delta^i$$

と単体 Δ_{st} を分割し，$\forall x \in \Delta_{\mathrm{st}}$ でなく，$\forall x \in \Delta^i$ $(i = 1, \ldots, m)$ の様に小さい単体に分けて共正値の条件を考える．ここで，単体 Δ 上の点 x についての共正値の条件 $x^\top Ax \geq 0$ $(\forall x \in \Delta)$ は，単体 Δ の頂点をならべた行列 V_Δ を用いて，$x^\top V_\Delta^\top AV_\Delta x \geq 0$ $(\forall x \in \Delta_{\mathrm{st}})$ と言い換えることができることに注意する．共正値錐 $\mathcal{C} = \left\{A \in \mathcal{S} : x^\top Ax \geq 0 \text{ for all } x \in \mathbb{R}^n_+\right\}$ を用いると，「A が共正値」$\Leftrightarrow A \in \mathcal{C} \Leftrightarrow V_{\Delta^i}^\top AV_{\Delta^i} \in \mathcal{C}$ $(i = 1, \ldots, m)$ と表せることがわかる．

このままでは 1 つの行列の共正値判定が複数の行列の共正値判定に置き換わっただけであるが，ある行列が共正値か判定するのは co-NP-complete であるため近似錐 $\mathcal{M} \subset \mathcal{C}$ を用いることを考えると，単体分割の意義が現れる．$\mathcal{M}$ と共正値行列 $A \in \mathcal{S}$ に適当な仮定を入れれば，Δ が十分小さい（1 番長い辺が十分に短い）ならば，$V_\Delta^\top AV_\Delta \in \mathcal{M}$ が示される．Δ が十分小さいならば，条件 $V_\Delta^\top AV_\Delta \in \mathcal{M}$ と条件 $V_\Delta^\top AV_\Delta \in \mathcal{C}$ がほとんど変わらないことを意味している．

Sponsel ら [4] は $V_\Delta^\top AV_\Delta \notin \mathcal{M}$ であるような単体 Δ を分割し続けるというアルゴリズムを提案した．このアルゴリズムは Bundfuss と Dür [1] が考案した単体細分を用いた共正値判定アルゴリズムの拡張であり，Bundfuss と Dür [1] が考案したアルゴリズムは，近似錐として非負対称行列錐を当てはめたものとなっている．

3　頂点番号のランダマイズ化

Bundfuss と Dür [1] が提案したアルゴリズムを用いて最大クリーク数を求める際に，グラフの頂点番号をランダムに入れ替えたグラフにも用いることによって，最大クリーク数のより良い下限が求まることを発見した．そして，複数の頂点番号をランダムに入れ替えたグラフについて提案されたアルゴリズムを適用することで，[1] の手法を用いた [2] で求まっている解よりも良い解を求めることに成功した．実験結果をまとめたものが表 1. となっている．実験で用いたグラフは，The Second DIMACS Implementation Challenge [5] の最大クリーク問題のベンチマークである．クリーク数は，グラフに含まれるクリークの中で最も頂点数が大きいクリークの頂点数である．実験は 100 個の頂点番号をランダムに入れ替えたグラフに対して行った．

4　$\mathcal{C}^1$ への適用

Sponsel ら [4] が提案した共正値判定アルゴリズムを de Klerk と Pasechnik [3] によって提案されている共正値錐の近似錐を用いることで拡張した．これによっ

表 1. 先行研究と本研究で求まったクリーク数の下限

グラフ名	Brock200_1	Brock200_2	Keller4
頂点	200	200	171
辺	14834	9876	9435
クリーク数	21	12	11
先行研究	13	10	8
本研究	14	11	11

表 2. Brock16 のクリーク数での共正値判定結果

手法名	単体数	反復回数	時間 (s)
$\mathcal{C}^1$-23-in	8899531	16255602	1335.065831
$\mathcal{C}^1$-23-out	8829635	17659269	1258.007799
$\mathcal{C}^1$-32-in	17829614	33648926	2612.610559
$\mathcal{C}^1$-32-out	16872170	33744339	2564.303332
既存手法	56062305	112124609	3784.309601

て，[2] よりもアルゴリズムの実行時間及び反復回数や生じる単体が少ない手法を提案することができた．

de Klerk と Pasechnik [3] の考えた近似錐は次の様に定義される:

$$\mathcal{C}^r = \{A \in \mathcal{S} : \left(\sum_{i,j=1}^{n} A_{ij}x_ix_j\right)\left(\sum_{i=1}^{n} x_i\right)^r \text{ のすべての単項の係数が非負 } \}.$$

先行研究の $\mathcal{M}$ に $\mathcal{C}^1$ を当てはめると，１つの頂点についての条件と２つの頂点についての条件（１つの辺についての条件）と３つの頂点についての条件（３つの辺についての条件）が出てくる．２つの頂点についての条件が破れていたときは先行研究と同様の単体分割を行い，３つの頂点についての条件が破れていたときは以下の分割方法を考案した．３つの頂点が成す３角形の内部に３角形上の点の中で最小となる点が存在するならば，その点をとってきて３つに分割，そうでないならば，３つの辺の条件の中から，条件を破っている辺をとってきてその辺について２つに分割，という方法である．

実験は The Second DIMACS Implementation Challenge [5] のグラフジェネレーターを用いて生成したグラフのクリーク数についての共正値判定の実験を行った．実験結果の内の１つが表２である．実験で用いた Brock16 は頂点１６辺７１クリーク数５のグラフである．この実験で扱われた $\mathcal{C}^1$ から始まる分割手法が本研究の手法で複数あるのは条件をみる順番等のためである．単体数は実験終了時にできていた単体の数を表す．本研究の手法はどの項目においても元となった手法より優れているといえる．

5 新しい近似錐

３つ目は，単体分割と近似錐が与えられると定義される新しい近似錐を考案したことである．この近似錐は，単体分割によって定義されるが，単体分割の単体の情報は必要とせず，頂点集合と辺集合から定義される近似錐となっている．この新しい近似錐を用いることによって，既存研究とは違った近似錐を用いて，１つ１つの単体をみる必要のない共正値判定アルゴリズムを考案することができた．

6 今後の課題

第４章の内容について，最大クリークの下限しか求まらないような難しい問題に対して，[2] よりも良い結果が得られるような，もしくは頂点番号についてロバストな結果が得られるような細分規則の考案が今後の課題といえる．また，深さ優先探索でなく，切る辺を決めその辺を持つすべての単体をその辺について分割する手法を当てはめることが可能と考えられるので，実際に当てはめることも今後の課題である．

第５章の内容について，新しい近似錐を使って実際にアルゴリズムを組む際に，入力として [4] で述べられている１つの近似錐を採用したが，他の近似錐を入力の近似錐として採用することが考えられる．また，単体細分の規則について，この新しい近似錐に適した規則を考えることがあげられる．

参考文献

[1] S. Bundfuss and M. Dür: Algorithmic copositivity detection by simplicial partition. *Linear Algebra and its Applications*, 428(2008), pp. 1511–1523.

[2] M. Dür and J. Žilinskas: Depth-first simplicial partition for copositivity detection, with an application to Maxclique. *Optimization Methods and Software*, 26(2011), pp. 499–510.

[3] E. de Klerk and D. V. Pasechnik: Approximation of stability number of a graph via copositive programming. *SIAM Journal on Optimization*, 12(2002), pp. 875–892.

[4] J. Sponsel, S. Bundfuss and M. Dür: An improved algorithm to test copositivity. *Journal of Global Optimization*, 52(2012), pp. 537–551.

[5] The Second DIMACS Implementation Challenge, http://dimacs.rutgers.edu/Challenges/