A-34

グラフ最適化問題に対する緩和問題の構成法の提案
A New Method to Obtain Relaxed Problems of Graph Optimization Problems

山口 一章* Kazuaki Yamaguchi　西出 知史† Tomofumi Nishide　増田 澄男* Sumio Masuda

1 まえがき

最適化問題は，ある集合 S と関数 $f(\cdot)$ に関し，$f(x^*) = \max_{x\in S} f(x)$ を満たすような $x^* \in S$ を求めよという問題として定式化できる．このような最適化問題を，二項組 (S, f) で表現することにする．S は制約条件，$f(\cdot)$ は目的関数と呼ばれる．最適化問題 (S, f) において，$x \in S$ なる任意の x を，実行可能解と呼ぶ．以降，特に断らない限り，最適化問題は最大化問題を意味するものとする．二つの最適化問題 (S, f), (S', f') に対し，S から S' への単射 $\phi : S \mapsto S'$ が存在し，任意の $x \in S$ に対し，$f(x) \le f'(\phi(x))$ が成立するとき，(S', f') を (S, f) の緩和問題と呼び，(S, f) を (S', f') の原問題と呼ぶ．任意の最適化問題 (S, f) とその任意の緩和問題 (S', f') について，$\max_{x\in S} f(x) \le \max_{x\in S'} f'(x)$ が成り立つ．

最適解を求めることがNP完全問題を解くのと同程度に難しい最適化問題はNP最適化問題 [1] と呼ばれる．NP最適化問題を多項式時間で解く方法は今のところ知られていない．しかし，良質な上界を与える緩和問題が存在すれば，それを分枝限定法と組み合わせることにより，NP最適化問題の最適解や近似解を現実的な時間で求めることができる．また，そのような緩和問題を用いることにより，何らかのヒューリスティックによって得られた近似解が最適解からどの程度離れているかを計算することができる．よって，良質な上界を与える緩和問題を得ることは，非常に重要な課題の一つである．

最適化問題の緩和問題を得る方法としては線形計画緩和やLagrange緩和 [2] がある．Lagrange緩和が与える上界は，線形計画緩和の上界に比べ悪くないことが知られている [2]．

本稿では，Lagrange緩和と，グラフアルゴリズムを基礎とする，新たな上界計算法を提案する．まず，提案手法が，多くのNP最適化問題に適用可能であることを示す．また，提案手法により良好な上界が得られることを，計算機実験により示す．

2 準備

単純な無向グラフを単にグラフと呼ぶ．どの2頂点間にも辺があるようなグラフは完全グラフと呼ばれる．あるグラフ $G = (V, E)$ に対し，完全グラフ $G' = (V', E')$ が G の部分グラフであるとき，G' を G のクリークと呼ぶ．クリーク G' の頂点集合 V' もクリークと呼ぶことにする．グラフ $G = (V, E)$ が与えられたときに，要素数が最大のクリークを求めよという問題は，最大クリーク問題と呼ばれている．この問題は，NP最適化問題の中でも最も有名なものの一つである．最大クリーク問題は，次の0-1整数計画問題として表現できる．ただし，$V = \{v_1, v_2, \cdots, v_n\}$ とする．

$$\begin{array}{lll} \max: & \sum_{i=1}^{n} x_i & \\ \text{s.t.} & x_i + x_j \le 1 & (\forall (v_i, v_j) \notin E) \\ & x_i \in \{0, 1\} & (1 \le i \le n) \end{array}$$

ここで，変数 x_i は，1かつそのときに限り，頂点 v_i が選ばれたクリークの要素であることを意味する．すなわち，この0-1整数計画問題の最適解が $x_i = X_i$ $(1 \le i \le n)$ であるとき，最大の重みを持つクリークは $\{v_i \in V | X_i = 1\}$ で表される．

$|V| = |E| = n$ かつハミルトン閉路を持つようなグラフ $G = (V, E)$ を，C_n と書く．あるグラフ G が C_k $(k \ge 4)$ を頂点誘導部分グラフとして持たないとき，G はchordal graph [3] と呼ばれる．あるグラフ $G = (V, E)$ に対し，$G' = (V, E \cup E')$ が G のchordal supergraphであるような E' は，G のfill-inと呼ばれる．できるだけ辺数の少ないfill-inを求めることについては多くの研究がなされている [4]．

3 提案手法

提案手法は，多くのグラフ最適化問題の緩和問題を与えるものである．提案手法を一般的な表現で記述すると複雑になるので，ここでは，最大クリーク問題を例に，緩和問題作成の手順を示す．最大クリーク問題の緩和問題を作る際，まず，最大重みクリーク問題が多項式時間で解けるようなグラフのクラスを一つ選ぶ．ここでは，chordal graphを選んだとする．次に，入力である $G = (V, E)$ に対し，$E = \cap_{i=1}^{h} E_i$ を満たすような，G のchordal supergraph $G_1 = (V, E_1)$, $G_2 = (V, E_2)$, $\cdots$, $G_h = (V, E_h)$ を求める（h は任意）．次に，$E_1, E_2, \cdots, E_h$ を用い，Lagrange分解によって以下の緩和問題を作る．ここでは，等式を緩和するため，Lagrange乗数 λ_{ik} $(1 \le i \le n,\ 2 \le k \le h)$ は任意の実数とする．

$$\begin{array}{lll} \max: & \sum_{i=1}^{n} x_{i1} + \sum_{k=2}^{h} \sum_{i=1}^{n} \lambda_{ik}(x_{i1} - x_{ik}) & \\ \text{s.t.} & x_{ik} + x_{jk} \le 1 & (\forall (v_i, v_j) \notin E_k,\ 1 \le k \le h) \\ & x_{ik} \in \{0, 1\} & (1 \le i \le n,\ 1 \le k \le h) \end{array}$$

この最適化問題が最大クリーク問題の緩和問題であることは，全ての x_{ik} に x_{i1} を代入すれば確認できる．この緩和問題は以下のように変形することができる．

$$\begin{array}{lll} \max: & \sum_{i=1}^{n} (1 + \sum_{k=2}^{n} \lambda_{ik}) x_{i1} + \sum_{k=2}^{h} \sum_{i=1}^{n} (-\lambda_{ik}) x_{ik} & \\ \text{s.t.} & x_{ik} + x_{jk} \le 1 & (\forall (v_i, v_j) \notin E_k,\ 1 \le k \le h) \\ & x_{ik} \in \{0, 1\} & (1 \le i \le n,\ 1 \le k \le h) \end{array}$$

上記の緩和問題は，各頂点 x_{i1} に重み $(1 + \sum_{k=2}^{n} \lambda_{ik})$ が付けられたchordal graph G_1 の最大重みクリーク問題と，各頂点 x_{ik} に重み $-\lambda_{ik}$ が付けられた $h-1$ 個のchordal graph G_k の最大重みクリーク問題を，独立に解くことによって解が得られる．chordal graphに対する最大クリーク問題は多項

*神戸大学工学部
†神戸大学大学院自然科学研究科

表 1: 各手法による上界の計算結果（頂点数=100）

辺数	100	150	200	300	400	500	700	1000	1500	2000
従来法 1	5.34	6.71	7.90	10.08	12.12	14.12	18.13	23.75	32.93	42.02
従来法 2	49.00	49.00	49.00	49.00	49.00	49.00	49.00	49.00	49.00	49.00
提案法	3.00	3.70	4.29	5.92	7.07	8.11	10.58	13.58	18.09	22.22
最適解	2.81	3.03	3.18	4.02	4.61	5.28	6.80	9.52	14.86	21.35

式時間で求められるので [3]，提案手法の緩和問題の解は多項式時間で得られる．緩和問題は，Lagrange 乗数 λ_{ik} がどのような値であっても上界を与えるが，λ_{ik} を調整することにより，より良い上界が得られる．

上記の手順は，他のグラフの最適化問題にも用いることができる．例えば，最大独立頂点集合問題に対する緩和問題を作る手順は以下の通りである．

1. 最大重み独立頂点集合問題が多項式時間で解けるようなグラフのクラスを一つ選ぶ（ここでは chordal graph を選んだとする）．
2. 何らかのヒューリスティックにより，$E = \cup_{i=1}^{h} E_i$ を満たすような，G の chordal subgraph $G_1 = (V, E_1)$, $G_2 = (V, E_2), \cdots, G_h = (V, E_h)$ を求める．
3. 最大クリーク問題の時と同様，Lagrange 分解により緩和問題を作成し，上界を得る．

最大独立頂点集合問題の場合，いくつかの chordal subgraph の和が元のグラフになるようにすることに注意されたい．最大クリーク問題では，グラフに辺を付け加えると条件が緩和されるので chordal supergraph を用いたが，最大独立頂点集合問題では，辺を取り除くことで条件が緩和されるので，chordal subgraph を用いる．

提案手法は，Dominating Set 問題のような最小化問題にも同様に用い得るし（その場合は下界が得られる），重み付きグラフに対する問題にも用いることができる．なお，chordal graph に対する Dominating Set 問題に対する多項式時間のアルゴリズムは知られていないので，この場合は，interval graph のような，他のグラフを用いる必要がある．

4 計算機実験

頂点数が 100 の連結なグラフについて，辺数を 100 から 2000 まで変化させ，各々を 100 個ずつ作成した．それらを入力として，最大クリーク問題の上界を，二つの従来法と提案法で得られる上界の平均値を比較した．提案法では，緩和問題を作成する際，例で示したのと同じく，chordal supergraph を用いた．詳細は省略するが，chordal supergraph を得る際，単純なヒューリスティックを用いた．従来法としては，以下の二つのものを用いた．

従来法 1　次数が k 以上の頂点が k 個以下しかないとき，最大クリークのサイズは k 以下である．k として，できるだけ小さな値を選び，それを上界とする．

従来法 2　最大クリーク問題の制約条件の不等式を全て，Lagrange 緩和により緩和し次の緩和問題を作る．ただし，λ_{ij} は Lagrange 乗数である．不等式を緩和するので，Lagrange 乗数は非負実数であるとする．

$$\begin{array}{lll} \max: & \sum_{i=1}^{n} x_i + \sum_{(v_i,v_j) \notin E} \lambda_{ij}(1 - x_i - x_j) & \\ \text{s.t.} & x_i \in \{0,1\} & (1 \le i \le n) \end{array}$$

なお，従来法 2，提案法ともに，原問題の解が整数であることを利用し，得られた緩和解の値そのものではなく，小数点以下を切り捨てた値を上界として用いた．また，Lagrange 乗数の更新には劣勾配法 [2] を用いた．分枝限定法により得られた最適解の値も併せて示す．

表 1 から，頂点の次数から得られる上界（従来法 1）は，最適解とは大きく異なっていることが分かる．単に全ての制約条件を緩和した場合（従来法 2），得られる上界は何ら役に立たない．それに比べ，提案法による上界は，最適解からさほど離れておらず，質の良い上界が得られたことが分かる．

5 まとめ

グラフの最適化問題に関し，その subgraph あるいは supergraph を用いることによって，良質な上界あるいは下界を与える緩和問題を作成する方法を示した．また，単に Lagrange 緩和を施しただけでは意味のない上界しか得られないことがあることも同時に示した．今回の提案は，Lagrange 緩和を適用する可能性を広げたといえる．

今後の課題はいくつか考えられるが，最も本質的かつ重要な課題は，与えられたグラフを，どのようなグラフの積あるいは和として表現するのが良いか調べることである．本稿では詳細は省略したが，最大クリーク問題の場合，用いる chordal supergraph は少ないほうが良く，また，各 chordal supergraph は，辺が少ないほど良いことが分かった．実験では，chordal supergraph を得る際，単純なヒューリスティックを用いたため，入力のグラフ G が密なとき，G を表現するために非常に多くの辺を持つ chordal supergraph を多数使うことになった．その結果，Lagrange 緩和における変数の数が増え，値が収束するまで長い時間がかかってしまった．

各問題に対する適切なグラフのクラスの選択，及び，適切な supergraph や subgraph を求めるヒューリスティックを，今後，明らかにしたいと考えている．

参考文献

[1] G.Ausiello et al., Complexity and Approximation, Springer, 1999.

[2] C.Reeves, モダンヒューリスティックス –組合せ最適化の先端手法–, 日刊工業新聞社, 1997.

[3] M.Golumbic, Algorithmic Graph Theory and Perfect Graphs, Academic Press, 1980.

[4] A.Natanzon, R.Shamir and R.Sharan, “A polynomial approximation algorithm for the minimum fill-in problem,” SIAM J. Comput., vol.30, no.4, pp.1067–1079, 2000.