

負の重みを許した最大重みクリーク問題に対する厳密解法

An Exact Algorithm for the Maximum Weight Clique Problem in the Graphs with Minus Weight

西藤 光[†] 山口 一章[†]
Hikaru Saito Kazuaki Yamaguchi

1. 最大重みクリーク問題

無向グラフ $G = (V, E)$ が与えられたとき、任意の 2 頂点間に辺が存在するような V の部分集合をクリークという。頂点や辺に重みが付与されているグラフに対して、頂点と辺の重みの総和が最大となるクリークを求める問題を、最大重みクリーク問題という。図 1 を例とすると、図 1 における最大重みクリークは $\{v2, v3, v4\}$ でその重みは 7 である。グラフ $G = (V, E)$ における、頂点 v の重みを $vw(v)$ 、頂点 u, v 間の辺の重みを $ew(u, v)$ と表す。 G に含まれる任意のクリーク C の重みを $W(C)$ と表し、 G における最大重みクリークの重みを $W_{opt}(G)$ と表す。 $N(v)$ は v の隣接頂点集合を表す。 V の部分集合を S とした時、 S による誘導部分グラフを $G(S)$ と表す。

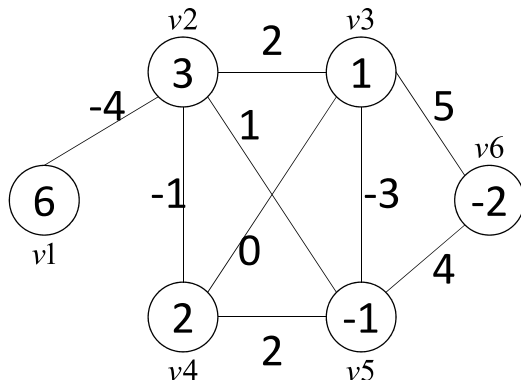


図 1: 入力グラフの例

2. まえがき

最大重みクリーク問題は NP 困難 [1] で、愚直に解こうとすると膨大な時間がかかってしまうので高速な解法が求められている。負の重みがない場合、最大重みクリークは極大なクリークであることがわかっているが、重みに負を許した問題では極大でないクリークも最大重みクリークになり得るので、より計算が困難になる。

クリークに関する問題には、最大サイズのクリークを見つける最大クリーク問題や頂点のみに重みがある最

大頂点重みクリーク問題、辺にも重みがある最大辺重みクリーク問題などがあり、その応用には、バイオインフォマティクス [2] や通信技術 [3]、ネットワーク分析などがある。最大辺重みクリーク問題に対する解法の最新の研究として清水らの MECQ [4] がある。グラフ G を完全グラフに限定した問題は MAX-2-SAT や Quadratic Unconstrained Binary Optimization (QUBO) と同等な問題となり、アニーリング型コンピュータを用いた解法も研究されている。

本研究では、負の重みを許したグラフにおける最大重みクリーク問題に対して、分枝限定法に基づいた厳密解法を提案し、提案法と、整数計画ソルバー CPLEX、負の重みを 0 として扱い従来の正の重みだけのグラフに対する解法を比較して、本アルゴリズムが負の重みを許したグラフに対して有効だと確認できた。

3. 分枝限定法

分枝限定法は分枝操作と限定操作の二つからなる。分枝操作とは、問題を部分問題に分割する操作である。分枝操作を繰り返し、部分問題を解いていくことで厳密解が得られる。限定操作とは、分割された部分問題に対して上界を計算することで、暫定解の更新がされないような部分問題の探索を打ち切ることである。限定操作を行うことで、効率的に問題を解くことができる。限定操作時の、上界計算にかかる時間と上界の良さが問題を解くのにかかる時間に大きく関わってくる。

4. 提案法

提案法では、各頂点に対して、その頂点を含む部分問題の上界計算を一括で行い分枝する頂点を決定する。

最初にまず頂点を次数の降順にソートしておく。その後、各部分問題に対して以下の処理を行う。

- ステップ 1. 頂点彩色
- ステップ 2. 各頂点の評価値と上界計算
- ステップ 3. 評価値の高い順に分枝

4.1 ステップ 1

グラフに対して頂点彩色を行い、頂点を色クラスに分割する。色クラス同士は独立集合になっている。クリークには各色クラスから高々 1 つの頂点しか含まれない。

[†] 神戸大学大学院工学研究科, Graduate School of Engineering, Kobe University

彩色法は、色数よりも速度を重視して貪欲法とした。計算量は $O(n^2)$ である。

4.2 ステップ 2

色クラスの番号が小さい頂点から順番に評価値と上界を計算する。 X_k は色クラス k の頂点集合、 x_k は X_k に含まれる頂点、 $A(v)$ は頂点 v の評価値を表すものとする。

$A(x_k)$ を以下のように定義する。

$$A(x_k) = vw(x_k) + \sum_{i=1}^{k-1} P(x_k, X_i)$$

ただし、

$$P(x_k, X_i) = \begin{cases} \max_{v \in N(x_k) \cap X_i} \{ew(x_k, v)\} & (\geq 0) \\ 0 & (\textit{otherwise}) \end{cases}$$

$A(x_k)$ は頂点 x_k の重みに、番号が k より小さい各色クラスの頂点集合から、頂点 x_k との辺の最大の辺の重みの最大値をそれぞれ足したものを表している。ただし、辺の重みの最大値が負の場合 0 を足すこととする。

次にグラフ $G_k = G(\{x_k\} \cup \bigcup_{i=1}^{k-1} X_i)$ における、 x_k を含む最大重みクリーク C_k の重みの上界を計算する。上界は以下のように計算する。

$$UB(W(C_k)) = vw(x_k) + \sum_{i=1}^{k-1} Q(x_k, X_i)$$

ただし、

$$Q(x_k, X_i) = \begin{cases} \max_{v \in X_i} \{ew(x_k, v) + A(v)\} & (\geq 0) \\ 0 & (\textit{otherwise}) \end{cases}$$

$UB(W(C_k))$ は頂点 x_k の重みに、番号が k より小さい各色クラスの頂点集合から、頂点 x_k との辺の重みと評価値の和の最大値をそれぞれ足したものを表している。ただし、辺の重みと評価値の和の最大値が負の場合 0 を足すこととする。

辺の重みと評価値の和が負の値になるまでは、辺重みと頂点重みが負でも許容されるので、負の重みを 0 として扱う方法よりは悪くならないと考えられる。

次に、 $C_k = \{y_1, \dots, y_m\}$ ($n \leq k$) とおき、 $y_m = x_k$ とする。そして y_i を含む色クラスの頂点集合を Y_i と定義する。

最大重みクリーク C_k の重みは

$$W(C_k) = \sum_{i=1}^m vw(y_i) + \sum_{i=1}^m \sum_{j=1}^{i-1} ew(y_i, y_j)$$

と表される。

このとき、

$$\begin{aligned} & UB(W(C_k)) \\ &= vw(x_k) + \sum_{i=1}^{k-1} \begin{cases} Q(x_k, X_i) & (\geq 0) \\ 0 & (\textit{otherwise}) \end{cases} \\ &\geq vw(y_m) + \sum_{i=1}^{m-1} \begin{cases} Q(y_m, Y_i) & (\geq 0) \\ 0 & (\textit{otherwise}) \end{cases} \\ &\geq vw(y_m) + \sum_{i=1}^{m-1} Q(y_m, Y_i) \\ &\geq vw(y_m) + \sum_{i=1}^{m-1} (ew(y_m, y_i) + A(y_i)) \\ &= vw(y_m) \\ &\quad + \sum_{i=1}^{m-1} \left(ew(y_m, y_i) + vw(y_i) + \sum_{j=1}^{i-1} \max_{v \in N(y_i) \cap Y_j} ew(y_i, v) \right) \\ &= \sum_{i=1}^m vw(y_i) \\ &\quad + \sum_{i=1}^{m-1} \left(ew(y_m, y_i) + \sum_{j=1}^{i-1} \max_{v \in N(y_i) \cap Y_j} ew(y_i, v) \right) \\ &\geq \sum_{i=1}^m vw(y_i) + \sum_{i=1}^{m-1} \left(ew(y_m, y_i) + \sum_{j=1}^{i-1} ew(y_i, y_j) \right) \\ &= \sum_{i=1}^m vw(y_i) + \sum_{i=1}^m \sum_{j=1}^{i-1} ew(y_i, y_j) \\ &= W(C_k) \end{aligned}$$

以上より、 $UB(W(C_k))$ が $W(C_k)$ の上界となることが証明できた。

上界が暫定解を超えるような頂点の集合を U とおく。このとき、 $W_{opt}(G(V \setminus U))$ は暫定解を超えないことがわかる。ゆえに、 U に含まれる頂点に対してのみ分枝操作を行えばよい。

4.3 ステップ 3

まず、 U の頂点を評価値の降順にソートし、列 $U_s = (u_1, u_2, \dots, u_{|U|})$ を作る。そして U_s の頂点に対して順番に分枝操作を行い部分問題を作成する。 u_i に対して分枝操作を行うとする。このとき、作成する部分問題の頂点集合 V' は、 $V' = N(u_i) \cap ((V \setminus U) \cup \{u_{i+1}, \dots, u_{|U|}\})$ とし、その順番は $V \setminus U$ に含まれる頂点を色クラスの番号が小さい順に並べたあとに、 U に含まれる頂点を評価値が小さい順に並べる。その後、 V' の各頂点の重みに、 $ew(u_i, v)$ を足して部分問題を作成する。

以下に提案法のアルゴリズムを示す。

Algorithm 1 提案法 branch

Require: グラフ $G' = (V', E')$, 頂点の重み $vw'[\cdot]$, 所持クリーク C , 暫定解 W_{cur}

Ensure: 最大重み W_{max}

```
1: if  $W(C) > W_{cur}$  then
2:    $W_{cur} \leftarrow W(C)$ 
3:    $W_{max} \leftarrow W_{cur}$ 
4: end if
5: 各頂点の評価値  $A \leftarrow vw'$ 
6: 上界が暫定解を超える頂点の集合  $U \leftarrow \emptyset$ 
7:  $\Psi \leftarrow V'$ を彩色して得られた独立集合  $\{X_1, \dots, X_c\}$ 
8: for  $i = 1$  to  $c$  do
9:   for  $x_i \in X_i$  do
10:     $UB \leftarrow W(C)$ 
11:     $s \leftarrow 0$ 
12:    for  $j = 1$  to  $i$  do
13:       $Q \leftarrow \max_{v \in X_j} \{ew(x_i, v) + A(v)\}$ 
14:      if  $Q > 0$  then
15:         $UB \leftarrow UB + Q$ 
16:      end if
17:       $P \leftarrow \max_{v \in X_j} \{ew(x_i, v)\}$ 
18:      if  $P > 0$  then
19:         $s \leftarrow s + P$ 
20:      end if
21:    end for
22:    if  $UB > W_{cur}$  then
23:      add  $x_i$  to  $U$ 
24:    end if
25:     $A[x_i] \leftarrow A[x_i] + s$ 
26:  end for
27: end for
28:  $U_s \leftarrow U$ を  $A$ の大きい順に並べた列  $(u_1, u_2, \dots, u_{|U|})$ 
29: for  $i = 1$  to  $|U|$  do
30:    $C' \leftarrow C \cup u_i$ 
31:    $V'' \leftarrow N(u_i) \cap ((V' \setminus U) \cup \{u_{i+1}, \dots, u_{|U|}\})$ 
32:    $V''$ の並び替え
33:   for  $v \in V''$  do
34:      $vw''[v] \leftarrow vw'[v] + ew(u_i, v)$ 
35:   end for
36:   branch( $G'(V''), vw'', C', W_{cur}$ )
37: end for
```

5. 計算機実験

提案法と、負の重みを0として負の重みのない最大重みクリーク問題に対する分枝限定法、そして整数計画ソルバー CPLEX の3つで解が得られた時間を比較した。

負の重みの割合は $m\%$ と表す。入力に用いるグラフは DIMACS とし、頂点 v_i の重みは $(i \bmod 200 + 1)$ とし、 $(i \bmod 100 < m)$ のとき重みを負にする。辺 (v_i, v_j) の重みは $(i+j) \bmod 200 + 1$ とし、 $(i+j) \bmod 100 < m$ のとき重みを負にする。負の重みの割合を 20%, 50%, 80%, 各条件でグラフを生成しその平均実行時間を記録する。実行時間が 300 秒を超えた場合は探索を中止する。

本実験で使用した計算機の CPU は Intel(R) Core(TM) i7-8700 CPU 3.20GHz, メモリは 8GB, OS は Windows10, 使用言語は Java8 である。

実験結果は表 1 に示す。

6. まとめ

提案法がほぼすべてのグラフにおいて、ほかの手法に対して一番実行時間が短くなった。特に C125.9 のように辺密度が大きなグラフほど、実行時間の差が大きくなった。また、負の重みの割合が多いほど、実行時間の比は小さくなった。これからは、CPLEX のようにマルチスレッドによる並列化処理を試みたり、さらなる改良を目指す。

参考文献

- [1] M.R. Garey and D.S. Johnson, Computers and Intractability - A Guide to the Theory of NP-Completeness, Freeman, NewYork, 1979.
- [2] R. Samudrala, J. Moulton, "A graph-theoretic algorithm for comparative modeling of protein structure", Journal of Molecular Biology, 279 (1): 287-302, (1998)
- [3] Z. Prihar, "Topological properties of telecommunications networks", Proceedings of the IRE, 44 (7): 927-933, (1956)
- [4] S. Shimizu, K. Yamaguchi, and S. Masuda, "A maximum edge-weight clique extraction algorithm based on branch-and-bound", Discrete Optimization, Vol. 37. (2020)

表 1: 実験結果

| Instance | 頂点数 | 辺密度 | 負の重みの割合 | 提案法 (s) | 正の厳密解法 (s) | CPLEX(s) |
|------------|-----|------|---------|---------|------------|----------|
| brock200_1 | 200 | 0.75 | 20% | 24.48 | 29.77 | over |
| | | | 50% | 12.80 | 18.45 | over |
| | | | 80% | 2.08 | 7.60 | over |
| brock200_2 | 200 | 0.50 | 20% | 0.05 | 0.06 | over |
| | | | 50% | 0.04 | 0.06 | over |
| | | | 80% | 0.02 | 0.04 | 82.45 |
| brock200_3 | 200 | 0.61 | 20% | 0.43 | 0.56 | over |
| | | | 50% | 0.30 | 0.40 | over |
| | | | 80% | 0.12 | 0.24 | 221.94 |
| brock200_4 | 200 | 0.66 | 20% | 1.68 | 2.05 | over |
| | | | 50% | 0.74 | 1.04 | over |
| | | | 80% | 0.25 | 0.57 | over |
| c-fat200-1 | 200 | 0.08 | 20% | 0.01 | € | 6.06 |
| | | | 50% | € | € | 4.97 |
| | | | 80% | € | € | 2.16 |
| c-fat200-2 | 200 | 0.16 | 20% | € | € | 6.27 |
| | | | 50% | 0.01 | 0.01 | 6.89 |
| | | | 80% | € | 0.01 | 5.89 |
| c-fat500-1 | 500 | 0.04 | 20% | 0.02 | 0.02 | 56.45 |
| | | | 50% | 0.01 | 0.01 | 83.88 |
| | | | 80% | € | 0.01 | 37.38 |
| c-fat500-2 | 500 | 0.07 | 20% | 0.01 | € | over |
| | | | 50% | 0.01 | 0.01 | over |
| | | | 80% | 0.01 | 0.05 | over |
| C125.9 | 125 | 0.90 | 20% | 39.19 | 51.21 | over |
| | | | 50% | 22.86 | 59.18 | over |
| | | | 80% | 5.42 | 87.20 | over |
| DSJC500_5 | 500 | 0.50 | 20% | 11.85 | 14.53 | over |
| | | | 50% | 8.44 | 11.34 | over |
| | | | 80% | 3.05 | 4.90 | over |
| hamming6-2 | 64 | 0.90 | 20% | 0.01 | 0.01 | 0.47 |
| | | | 50% | 0.08 | 0.13 | 0.69 |
| | | | 80% | 0.01 | 0.04 | 1.02 |
| hamming6-4 | 64 | 0.35 | 20% | € | € | 0.56 |
| | | | 50% | € | € | 0.38 |
| | | | 80% | € | € | 0.08 |
| hamming8-4 | 256 | 0.64 | 20% | 1.58 | 1.86 | over |
| | | | 50% | 3.09 | 4.26 | over |
| | | | 80% | 0.82 | 2.10 | over |