6B−5

B&Bによる最適系列分割問題における改善と拡張

加地太一　　加地郁夫　　大内　東
北海道情報大学　北海道工業大学　北海道大学

1．はじめに

ノードが連続的な番号を保持し、始点が初期番号、終点が最終番号となり、両端点が異なるグラフを系列グラフG（N，E）と呼ぶ。本論文における最適系列グラフ分割は系列グラフに対して、各ノードに与えられた重みの総和がブロックサイズP（＞0）以下であり、かつ、部分集合のノード番号が連続的に保持される条件のもとで、カットされるエッジのコストの和が最小となるよう分割する問題である。

以上の問題に対して、Branch−and−Bound法（B&B）を用いて従来、開発した手法を新たに改良改善し、また新しい問題の拡張を試みる。

2．問題の設定

ノードN＝｛1、2、・・・、n｝、エッジE＝｛e_1、e_2、・・・、e_r｝からなる　無向グラフをG＝｛N、E｝とする。各エッジは非負のコストC＝｛c_1、c_2、・・・、c_r｝をもつ。この時Gの各ノードは｛w_1、w_2、・・・、w_n｝の重みをもち、各々$0<w_i \leq P$となる。ここで、Pはブロックサイズと呼ばれる正の数である。また、ノード集合Sの重さは$|S|=\Sigma_{i\in S} w_i$となる。

Gをk個のサブ集合G_1、G_2、・・・、G_kに、以下の制約のもとで、カットされるエッジのコストの総和が最小になるよう分割する。

（1）$|G_i|$＝部分集合のノードの重みの総和≦P

（2）任意のG_iの各ノード番号は連続的な数をもつ

部分集合G_iはブロックと呼ばれG_i＝｛j，j＋1，・・・，m−1，m｝となる。G_iでの一番小さなノードの名前をb_iとして、ブレイクポイントと呼ぶこととする。b_i＝jはノードj−1とノードjの間でカットすることを意味する。　集合｛b_1，b_2，・・・、b_k｝は一意的に分割｛G_1，G_2，・・・，G_k｝を表現する。

ここで、グラフGが有向グラフであり、かつ（i，j）と（j，i）のエッジを含んでいるのなら、i＜jである単一エッジ（i，j）におきかえる。その時のコスト$C'ij=Cij+Cji$となる

3．Branch−and−Bound法（B&B）

この問題に対して以前に示したB&B法の概略を以下の6つのパラメーターによる構成で表す。

（1）ブランチングルールB

ブランチングルールBは分割問題の派生過程を定義する。派生過程において、いかに探索空間を小さくするかが問題である。ブランチングルールBとしてはノード系列の初期値を開始点として次の2つの性質により順次展開していく。

性質1．各ブロックの大きさがPを越えてはならない。

性質2．あるブロックを細分化すればコストへの寄与が大きくなる。ゆえに探索木の中で、あるブロックとそのブロックを細分化した状態が生じるとき、細分化した状態をコスト最小化の立場で無視してよい。

1）と2）の規則より次の不等式が導ける。

$$BP_{i-1}+P \geq BP_i > BP_{i-2}+P$$

（ノードウェイト＝1のときの不等式）

BP_i　：　i番目のブレイポイント

P　：　ブロックサイズ

以上の不等式にしたがってブランチングノードを生成する。

（2）セレクションルールS

セレクションルールSは現在のアクティブノードから次のブランチングノードを選択する探索戦略である。ここではこの問題に効率のよい最良下限探索を採用する。

最良下限探索は探索過程の現時点でコストの値が最も低いノードを選んで進んでいくバックトラック法である。その時点までに得られているノードの中から、最も目標に近いノードを選んで展開する。

（3）ドミナンスリレイションD

部分解π_1と部分解π_2に対して、π_1、π_2を先祖とする完全解の中でコストが最小なものを比較し、2つのノードの優位性をみる。

もし　min｛π_1を先祖とする完全解のコスト集合｝＜min｛π_2を先祖とする完全解のコスト集合｝なら$\pi_1 D \pi_2$となる。この問題において以下の2つのドミナンスリレイションが成立する。

1）（部分解π_k）D（部分解π_kのブロックG_iを細分化した部分解）

2）確定部の長さが等しい部分解π_a、π_bにおいて、π_aのコストがπ_bより低ければ$\pi_a D \pi_b$

（4）下限値関数L

L（π）は各部分解πに対しての完全解のコスト集合の

Improvement and Extension for Sequential Partitions of Graphs

Taich Kaji, Ikuo Kaji, Azuma Ohuti

Hokkaido Information University, Hokkaido Institute of Technology, Hokkaido University

下限となる実数値を表わす。ここでは各部分集合の下限値は確定部においてブレイクポイントによって切られるエッジのコストと未確定部におけるブロックサイズPを越えるエッジのコストの総和とする。

（５）上限値U

Uは今現在知られている完全解のなかで最良なコストの値である。

（６）削除ルールE

Eは上限値やドミナンスリレイションを用いてノードを削除する方を定義する。探索過程において、$L(\pi_1) > U$ または $\pi_2 D \pi_1$ の場合、π_1を削除し、探索空間を縮小させる。

４．高速化のための問題点

おおまかに以下の５つの点においてアルゴリズムおよびデータ構造の工夫をほどこし高速化に対処した。

１）グラフデータの表現

任意のノードから距離が違い順に整列された流出辺をリストにつなぎ、同様に任意のノードへ距離が違い順に整列された流入辺をリストにつなぐ。これによって、スパース部への非参照およびコスト計算の効率化が行われる。

２）増分コスト計算

前回までの増分コスト計算は $C(x, y) = \sum_{\substack{y \le i < x \\ j \ge x}} c_{ij}$ の計算式を用いていた。これに対して今回は

$$C(x, y) = C(x-1, y) + (C_{x-1,x} + C_{x-1,x+1} + \cdots + C_{x-1,n}) - (C_{y,x-1} + C_{y+1,x-1} + \cdots + C_{x-2,x-1})$$

を用いることによって、漸化的に計算し、効率化を計る。また、このとき上記１）に示すデータ構造を有効に用いることによって効率の向上を計る。

３）探索空間のノード状態の表現

探索空間の各ノードの状態を逆順リストで表現することによって、データのアクセスを迅速化する。

４）ＡＳリスト処理の高速化

小さい順に並べる整列一方向リストを構成するＡＳリストと、あわせて同レベル要素を高速に参照するために付表処理を用いる。またＡＳリストへの展開ノード集合の挿入法として付表の書換機能をもつマージ法を採用する。これらにより、アクティブノード集合への高速化が計られる。

５）セレクションルールの順序

実験結果より以下の構成順が最良であることが判明した。

１：細分化によるドミナンス
２：同レベルによるドミナンス
３：上限値による選択

以上の５つの点を考慮したアルゴリズムでのノード数、ブロックサイズの変化に対する影響を図１、２に示す。

５．問題の拡張

削除ルールおよびセレクションルールの拡張によって、問題の多様化を計る。以下の２点を考察中である。

１：すべての最適解を求める。
２：ある性質を持つノードを同じブロック内に、および異なるブロックに分割する。

参考文献

[1]Brian.W.Kernighan: Optimal Sequential Partitions of Graphs, J.ACM,Vol.18,No.1,pp.34-40(1971).

[2]Walter.H.Kohler: Characterization and Theoretical Comparison of Branch-and-Bound Algorithms for Permutation Problems, J.ACM,Vol21,No.1,pp.140-156(1974).

[3]加地、大内、加地: Branch-and-Bound法によるグラフの最適系列分割問題、ＯＲ学会春季研究発表会,pp198-199(1990).

[4]加地、大内、加地: グラフの最適系列分割アルゴリズムの改善、情報処理学会第４０回全国大会,pp61-62(1990).

[5]加地、大内: 分枝限定法による最適系列分割問題、情報処理学会研究報告、90-AL-16,pp.69-76(1990).

[6]加地、大内: 最適系列分割問題に対するＤＰとＢＢアルゴリズム、 情報処理学会第４１回全国大会、pp89-80(1990).

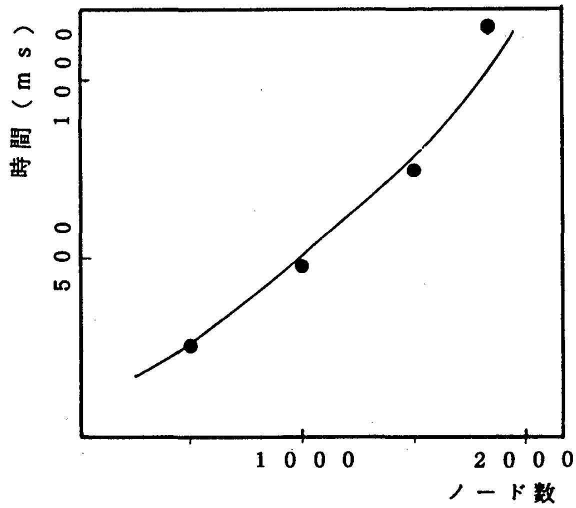

図１． ノード数と計算時間

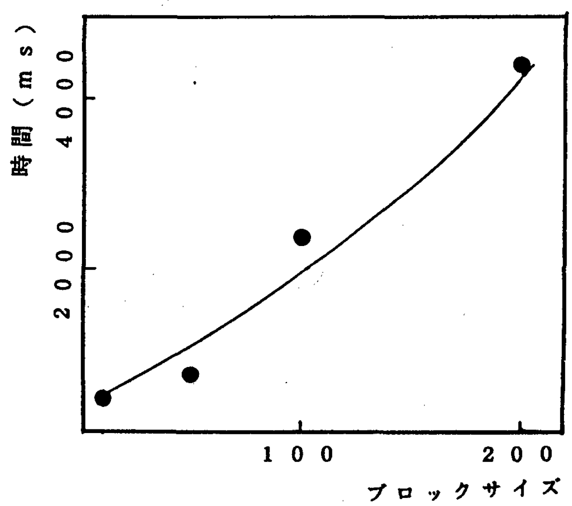

図２． ブロックサイズと計算時間