

2部グラフ描画問題に対する近似アルゴリズム

山口 敦子^{†*} 杉本 晃 宏^{†,☆☆}

2部グラフ描画における辺交差数最小化問題は、NP困難であることが知られている。本稿では、この問題に対する多項式時間近似アルゴリズムを提案し、その近似精度と入力グラフの最大次数との関係を述べる。最大次数が4以下の場合、本アルゴリズムが出力する解の辺交差数と最適解の辺交差数との比は2以下になり、最大次数が大きくなるにつれてこの値は漸近的に3に近づく。また、計算機実験によって、本アルゴリズムと従来法である重心法やメジアン法とを比較し、頂点数やグラフの最大次数にかかわらず、本アルゴリズムの方が良い解を出力することを示す。

An Approximation Algorithm for the Bipartite Graph Drawing Problem

ATSUKO YAMAGUCHI^{†*} and AKIHIRO SUGIMOTO^{†,☆☆}

The minimum edge crossings problem for bipartite graphs is known to be NP-hard. This paper presents a polynomial-time approximation algorithm and the relationship between the approximation ratio of our algorithm and the maximum degrees of input graphs. When the maximum degree of a graph is not greater than four, the approximation ratio, i.e., the maximum ratio of the edge crossing number of the solution constructed by our algorithm to the minimum edge crossing number, is two, and this ratio monotonically increases to three as the maximum degree becomes high. We also present our experiments on comparing the proposed algorithm with the barycenter method and the median method. Our experiments show that the proposed algorithm constructs better solutions than the other methods for dense graphs as well as sparse graphs.

1. はじめに

グラフは、関連するものとの間の相互関係を抽象化した表現であり、多くの分野で広く利用されている。グラフを平面上の図形として描画すれば、グラフで表現された関係が織りなす構造の特徴を直観的に捉えることができる。見にくく描画されたグラフは誤解を招き、構造の把握を困難にするため、グラフで表現された関係の構造を容易に捉えるためには、頂点間のつながりや全体の特徴を直観的に把握しやすいようにグラフを描画する必要がある。このような背景のもと、与えられたグラフの頂点を点とし、辺を線として、グラフを平面上に自動的に見やすく描く技術——

グラフ描画問題とよばれている——が、ユーザインタフェースをはじめとして、様々な分野で必要になっている²⁾。

あるグラフの描画が見やすいかどうかを議論し評価するためには、見やすさに関する何らかの基準が必要であるが、この基準は最終的に個人の主観に依存する。それゆえ、見やすさを一般論として議論するのは困難である^{☆☆}。しかし、「グラフの辺の交差数が少ない」など共通に受け入れられる見やすさの基準もある。このような共通に受け入れられる基準を数学的に定式化し、描画の見やすさを評価する研究が数多く報告されている^{1),5),8)}。このような見やすさの基準として、これまで経験的に用いられているもの²⁾を以下にあげる：

- (1) 辺の交差数が少ないこと。
- (2) 頂点が平均的に散らばっていること。
- (3) 辺の長さが揃っていること。

† 日立製作所基礎研究所
Advanced Research Laboratory, Hitachi, Ltd.

* 現在、生物分子工学研究所
Presently with Biomolecular Engineering Research Institute

☆☆ 現在、京都大学大学院情報学研究所
Presently with Graduate School of Informatics, Kyoto University

☆☆☆ VLSIレイアウトの自動生成を目指したグラフ描画では、描画領域や配線長の最小化など最適化の基準が比較的明確であるが、ユーザインタフェースなど、見やすく表示することを目指したグラフ描画では、最適化の基準は必ずしも明確でない。

(4) 頂点・辺が対称的に描かれていること。

階層グラフ^{*}は頂点間の意味的構造を表現することができるので、その構造を直観的に把握するために、見やすく描画することが特に重要となる。また、有向グラフを描画する際に、階層グラフを構成しその描画を利用することが一般的であるので、階層グラフの描画は、グラフ描画で重要な位置を占める。

与えられた階層グラフに対して、 x - y 平面の y 軸方向に層をとって、各層を x 軸と平行になるように埋め込み、各頂点の位置をその頂点が含まれる層と x 座標とで表す描画を階層グラフ描画とよぶ。階層グラフの描画を考えた場合、頂点をうまく各層に散らばらせることで見やすさの基準(2)をある程度満たすように描くことができる。したがって、階層グラフの描画では、基準(1), (3), (4)を満たすようにどう描くかが問題となる。特に、辺の交差数最小化(基準(1))は、その基準の明確さから重要視されている。本稿では、見やすさの基準として、辺の交差数最小化を取りあげる。

辺の交差数最小化を見やすさの基準とする階層グラフ描画では、以下の手法が一般的に使われている⁹⁾。

Step 1: 3つ以上の層にまたがる辺に対し、辺と層が交差する場所にダミー頂点を加え、すべての辺の両端点を隣接する2つの層を含むようにする。

Step 2: 第1層から順に、隣接する2つの層(または3つの層)を選ぶ。上位層の頂点の位置を固定し、固定していない下位層の頂点の x 座標を変えることで辺交差数を減らす^{**} [3つの層の場合には、中位層を除く2つの層(上位層と下位層)の頂点の位置を固定し、中位層の頂点の x 座標を変えることで辺交差数を減らす]。

Step 3: ダミー頂点を取り除き、辺を元に戻す。

本稿では、Step 1で2つの層を選んだ場合に行う処理に焦点を当てる。2つの層からなる階層グラフは2部グラフと等価なので、扱う問題は2部グラフ描画の辺交差数最小化となる。この問題に対して、下位層の2つの頂点間の順序による辺交差数の増減に着目し、増減比を測度としたグリーディ法による多項式時間近似アルゴリズムを提案する。また、下位層に含まれる頂点の最大次数を d としたとき、このアルゴリズムの近似精度は、最適解の辺交差数の $c(d)$ 倍以下である

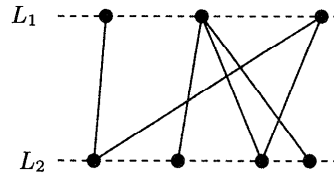


図1 2部グラフ
Fig. 1 Bipartite graph.

ことを示す。ここで、 $c(d)$ は、2と3の間の値をとる、 d に関する広義単調増加関数である。また、計算機実験により、本アルゴリズムを重心法やメジアン法と比較し、入力グラフの頂点数や辺の疎密にかかわらず、本アルゴリズムの方が良い解を出力することを示す。

2. 2部グラフ描画の辺交差数最小化問題

本章では、辺交差数最小化を見やすさの基準とする2部グラフ描画問題を定式化する。

定義 2.1 (2部グラフ) 2部グラフとは、2つの互いに素な頂点集合 L_1, L_2 と $L_1 \times L_2$ の部分集合 A とからなるグラフ $G = (L_1, L_2, A)$ である(図1)。ただし、 A の要素の端点の1つは L_1 に、もう1つは L_2 に含まれるものとする。□

本稿では、2部グラフに対して、 L_1 を第1層(上位層)、 L_2 を第2層(下位層)とよぶ。

L_1 に含まれる頂点の順序が x_1 、 L_2 に含まれる頂点の順序が x_2 のときの辺の交差数を $cross(G, x_1, x_2)$ と記す。このとき、2部グラフの辺交差数を最小化する問題は次のように定式化される。

定義 2.2 (2部グラフ描画の辺交差数最小化問題)
入力: 2部グラフ $G = (L_1, L_2, A)$ 、および、 L_1 に含まれる頂点の順序 x_1 。

問題 $cross(G, x_1, x_2)$ を最小とする L_2 上の順序 x_2 を求めよ。□

この問題の最適解は、 L_1 の頂点の順序が x_1 のときの辺交差数の最小値

$$opt(G, x_1) := \min \{ cross(G, x_1, x_2) \mid x_2 \text{は } L_2 \text{上の順序} \}$$

2部グラフ描画の辺交差数最小化問題はEadesら³⁾により、NP困難であることが示されており^{***}、そのヒューリスティクスについては、重心法⁹⁾やメジアン法⁴⁾などが知られている。

重心法⁹⁾は、下位層の各頂点に対し、それと隣接する上位層の頂点の x 座標の重心をその頂点の x 座標

^{*} 階層グラフはグラフ $H = (L_1, \dots, L_m, A)$ である。 L_i は頂点集合、 A は L_j の頂点と L_k ($j < k$)の頂点を結ぶ辺の集合である。

^{**} 上位層の頂点の位置も固定されていないとする場合もあるが、上位層の頂点の位置は固定されているとする方が主流である。

^{***} L_2 に含まれる頂点の最大次数を2以下に制限したグラフに対してもNP困難である。

とする手法である。Eadesら⁴⁾は、重心法の近似精度は、上位層の頂点数が n 、下位層の頂点の最小次数が d 以上の場合、最適解の辺交差数の $O(n/d)$ 倍以下であり、一般の2部グラフの場合、最適解の辺交差数の $O(\sqrt{n})$ 倍以下であることを示した。しかし、最適解の辺交差数の $O(\sqrt{n})$ 倍の解を与える2部グラフの例を構成することができる⁴⁾ことも知られており、重心法はときに悪い解を出しうる手法であるといえる。

メジアン法⁴⁾は、下位層の各頂点に対して、それと隣接する上位層の頂点の x 座標のメジアンを計算し、その値をその頂点の x 座標とする手法である。Eadesら⁴⁾は、メジアン法の近似精度は最適解の辺交差数の3倍以下であり、特に、辺が密なグラフに対しては、最適解にかなり近い精度を出すことを示した。たとえば、上位層、下位層の頂点数をともに n 、辺の数を cn^2 ($0 < c < 1$) とおくと、最適解の辺交差数に対する近似解の辺交差数の比は $\frac{3-c^2+O(1/n)}{1+c^2-O(1/n^2)}$ となり、頂点数が多くなり、辺の数が完全グラフの辺数に近くなるほど(辺が密になるほど)、この値は1に近くなる。これに対し、辺が疎なグラフの近似精度については、詳細に調べられていない。

一方、Mäkinen⁷⁾やJüngerら⁶⁾は、計算機実験で重心法とメジアン法を比較し、両アルゴリズムにより求めた解の辺交差数にはあまり差はないが、重心法の方が平均的に良い結果を出力することを示している。

3. 近似アルゴリズムとその精度

本章では、2部グラフ描画の辺交差数最小化問題に対し、多項式時間近似アルゴリズムを与え、その近似精度を評価する。 L_1 に含まれる端点を t 、 L_2 に含まれる端点を u とする辺を (t, u) で表す。 $u, v \in L_2$ に対し、 c_{uv} を次のように定義する：

$$c_{uv} := |\{ \langle (t, u), (w, v) \rangle \in A \mid t, w \in L_1, x_1(t) > x_1(w) \}|.$$

3.1 アルゴリズム

ここでは、第2層の2つの頂点間の順序による辺交差数の増減に着目し、増減比を測度としたグリーディ法による多項式時間近似アルゴリズムを提案する。

具体的なアルゴリズムは以下のようになる：

入力：2部グラフ $G = (L_1, L_2, A)$ 、および、 L_1 に含まれる頂点の順序 x_1 。

出力： L_2 上の順序 x_2 。

Step 0. $L := L_2$, $i := 1$.

Step 1. すべての $u, v \in L_2$ に対し、 c_{uv} を計算する。

Step 2. $\sum_{v \in L} c_{uv} / \sum_{v \in L} \min\{c_{vu}, c_{uv}\}$ が最小となる $u^* \in L$ を見つける。

$$x_2(u^*) := i, L := L - \{u^*\}.$$

Step 3. $L \neq \emptyset$ ならば、 $i := i + 1$ として Step 2へ。 $L = \emptyset$ ならば、終了。

Step 1では、 L_2 に含まれる各頂点 u, v に対し、 c_{uv} を計算する。Step 2では、各反復において、残った頂点の集合 $L \subseteq L_2$ の中で $u \in L$ を最左においた場合の u に関する辺交差数と、 u に関する辺交差数の最小値との比を比較し、比が最も小さい頂点 u^* を最左におき、 L から u^* を取り除く。 L_2 に含まれるすべての頂点の順序が決定すれば、アルゴリズムは停止する。

このアルゴリズムは明らかに多項式時間で計算可能である。Step 1は各 $u, v \in L$ に対し、 u を端点とする辺と v を端点とする辺とを比較すればよいので、 $O(|A|^2)$ 時間の計算を要する。また、Step 2では、 $\sum_{v \in L_2} c_{uv}$ と $\sum_{v \in L_2} \min\{c_{vu}, c_{uv}\}$ とをあらかじめ計算しておけば、定数時間で $\sum_{v \in L_2} c_{uv} / \sum_{v \in L_2} \min\{c_{vu}, c_{uv}\}$ を計算でき、各反復の $\sum_{v \in L} c_{uv}$ と $\sum_{v \in L} \min\{c_{vu}, c_{uv}\}$ の更新は、 c_{u^*v} と c_{vu^*} ($u^* \in L \subseteq L_2$) を参照すれば、 $O(|L|) \leq O(|L_2|)$ 時間ですむ。したがって、Step 2全体にかかる時間は $O(|L_2|) \times |L_2| = O(|L_2|^2)$ である。ゆえに、このアルゴリズム全体は $\max\{O(|A|^2), O(|L_2|^2)\}$ 時間を要する。

注意 3.1 重心法やメジアン法は $O(|A|)$ の計算量である。したがって、計算時間に着目すると本アルゴリズムは重心法やメジアン法に比べて遅い。

3.2 近似精度の評価

与えられた d に対して、 $c(d)$ は以下の式をみたす最小の数とする：任意の正整数 $d_1, d_2 \leq d$ に対し、

$$\left(\left\lceil \frac{c(d)-1}{c(d)+1} d_1 \right\rceil - 1 \right) \times \left(\left\lceil \frac{c(d)-1}{c(d)+1} d_2 \right\rceil \right) + d_1 \times \left(\left\lfloor \frac{2}{c(d)+1} d_2 \right\rfloor \right) \leq \frac{c(d)}{c(d)+1} d_1 d_2, \\ \text{かつ } c(d) \geq 2.$$

本節では、 L_2 に含まれる頂点の最大次数を d 以下に制限したとき、上に述べたアルゴリズムの近似精度が最適解の辺交差数の $c(d)$ 倍以下であることを示す。

c_{uv} の定義より、

$$\text{cross}(G, x_1, x_2) = \sum_{(u,v) \in \{u,v \in L_2 \mid x_2(u) < x_2(v)\}} c_{uv}, \\ \text{opt}(G, x_1) \geq \sum_{u,v \in L_2} \min(c_{uv}, c_{vu})$$

が成立するので、 L_2 上のある順序 x_2 に対して、ある定数 B が存在して、「任意の u, v に対し、

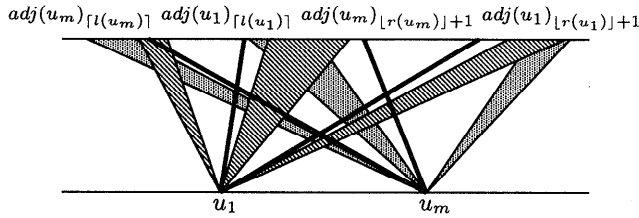


図2 $u_1, u_m \in L_2$ の隣接頂点の配置
Fig. 2 Adjacent nodes of u_1 and u_m .

$x_2(u) < x_2(v)$ ならば, $c_{uv} \leq B \cdot c_{vu}$ が成り立てば, 「 $B \cdot \text{opt}(G, x_1) \geq \text{cross}(G, x_1, x_2)$ 」が成立することが分かる. したがって, 上記アルゴリズムの近似精度が最適解の辺交差数の $c(d)$ 倍以下であることを示すには, 「任意の u, v に対し, $x_2(u) < x_2(v)$ のとき, $c_{uv} \leq c(d) \cdot c_{vu}$ 」を示せばよい. Step 2 の各反復では, $\sum_{v \in L} c_{uv} / \sum_{v \in L} \min\{c_{vu}, c_{uv}\}$ が最小となる u を L の中で最左にしているのだから, 結局, 「任意の $L \subseteq L_2$ に対し, ある $u \in L$ が存在して, 任意の $v \in L - \{u\}$ に対し, $c_{uv} \leq c(d) \cdot c_{vu}$ 」を示せばよい.

これを示す準備として次の補助定理を用意する. なお, 頂点 u ($\in L_2$) に対し, その次数を d_u で表し, 隣接頂点を左から $\text{adj}(u)_1, \text{adj}(u)_2, \dots, \text{adj}(u)_{d_u}$ とする.

補助定理 3.1 d が与えられたとき, $u_1, \dots, u_m \in L$ を, $d_{u_i} \leq d$ ($i = 1, 2, \dots, m$) かつ $c_{u_j u_{j+1}} > c(d) \cdot c_{u_{j+1} u_j}$ ($j = 1, 2, \dots, m-1$) なる頂点の列とする. また, 頂点 u に対し, $l(u) = \frac{c(d)-1}{c(d)+1} d_u$, $r(u) = \frac{2}{c(d)+1} d_u$ とおく ($l(u) + r(u) = d_u$ である). このとき, 以下の (1), (2) がともに成立する (図 2 参照).

- (1) $\text{adj}(u_1)_{\lceil l(u_1) \rceil}$ は $\text{adj}(u_m)_{\lceil l(u_m) \rceil}$ と同じ位置かそれより右 (x 座標の値が大きい).
- (2) $\text{adj}(u_1)_{\lfloor r(u_1) \rfloor + 1}$ は $\text{adj}(u_m)_{\lfloor r(u_m) \rfloor + 1}$ と同じ位置かそれより右 (x 座標の値が大きい).

証明: $1 \leq i \leq m-1$ に対し, $\text{adj}(u_i)_{\lceil l(u_i) \rceil}$ が $\text{adj}(u_{i+1})_{\lceil l(u_{i+1}) \rceil}$ より左にあると仮定する. このとき, $c_{u_i u_{i+1}}$ と $c_{u_{i+1} u_i}$ は以下の式をみたとす (図 3).

$$c_{u_i u_{i+1}} \leq \lceil l(u_i) \rceil \cdot (\lceil l(u_{i+1}) \rceil - 1) + (d_{u_i} - \lceil l(u_i) \rceil) \cdot (d_{u_{i+1}} - \lceil l(u_{i+1}) \rceil + 1)$$

$$c_{u_{i+1} u_i} \geq d_{u_i} d_{u_{i+1}} - \lceil l(u_i) \rceil \cdot (\lceil l(u_{i+1}) \rceil - 1) - (d_{u_i} - \lceil l(u_i) \rceil) \cdot (d_{u_{i+1}} - \lceil l(u_{i+1}) \rceil + 1)$$

また, $d_{u_i}, d_{u_{i+1}} \leq d$ より, $c(d)$ の定義から,

$$(\lceil l(u_{i+1}) \rceil - 1) \times \lceil l(u_i) \rceil + d_{u_{i+1}} \times \lfloor r(u_i) \rfloor \leq \frac{c(d)}{c(d)+1} d_{u_i} d_{u_{i+1}}$$

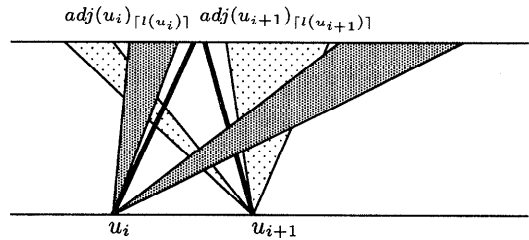


図3 $\text{adj}(u_i)_{\lceil l(u_i) \rceil}$ が $\text{adj}(u_{i+1})_{\lceil l(u_{i+1}) \rceil}$ より左にあるときの c_{u_i} と $c_{u_{i+1}}$ の関係
Fig. 3 Relationship between c_{u_i} and $c_{u_{i+1}}$ when $\text{adj}(u_i)_{\lceil l(u_i) \rceil}$ is allocated to the left-hand side of $\text{adj}(u_{i+1})_{\lceil l(u_{i+1}) \rceil}$.

が成り立つ. よって,

$$c(d) \cdot c_{u_{i+1} u_i} - c_{u_i u_{i+1}} \geq c(d) \cdot \{d_{u_i} d_{u_{i+1}} - \lceil l(u_i) \rceil \cdot (\lceil l(u_{i+1}) \rceil - 1) - (d_{u_i} - \lceil l(u_i) \rceil) \cdot (d_{u_{i+1}} - \lceil l(u_{i+1}) \rceil + 1) - \lceil l(u_i) \rceil \cdot (\lceil l(u_{i+1}) \rceil - 1) - (d_{u_i} - \lceil l(u_i) \rceil) \cdot (d_{u_{i+1}} - \lceil l(u_{i+1}) \rceil + 1)\} = c(d) d_{u_i} d_{u_{i+1}} - (c(d)+1) \{ \lceil l(u_i) \rceil \cdot (\lceil l(u_{i+1}) \rceil - 1) + (d_{u_i} - \lceil l(u_i) \rceil) \cdot (d_{u_{i+1}} - \lceil l(u_{i+1}) \rceil + 1) \} \geq c(d) d_{u_i} d_{u_{i+1}} - (c(d)+1) \frac{c(d)}{c(d)+1} d_{u_i} d_{u_{i+1}} + (c(d)+1) (d_{u_i} - \lceil l(u_i) \rceil) (\lceil l(u_{i+1}) \rceil - 1) = (c(d)+1) (d_{u_i} - \lceil l(u_i) \rceil) (\lceil l(u_{i+1}) \rceil - 1) \geq 0$$

これは $c_{u_i u_{i+1}} > c(d) \cdot c_{u_{i+1} u_i}$ に反する. したがって, 任意の i ($1 \leq i \leq m-1$) について, $\text{adj}(u_i)_{\lceil l(u_i) \rceil}$ は $\text{adj}(u_{i+1})_{\lceil l(u_{i+1}) \rceil}$ と同じ位置かそれより右にある. ゆえに (1) が成立する.

(2) についても (1) と同様に示すことができる. □
補助定理 3.1 を用いて次の定理を得る.

定理 3.1 L に含まれる頂点の最大次数が d 以下のとき, 任意の $L \subseteq L_2$ に対し, ある $u \in L$ が存在して, 任意の $v \in L - \{u\}$ に対し, $c_{uv} \leq c(d) \cdot c_{vu}$.

証明: 頂点列 u_0, \dots, u_m に対し, u_i の隣接頂点を左から $\text{adj}(u_i)_1, \text{adj}(u_i)_2, \dots, \text{adj}(u_i)_{d_{u_i}}$ とする. 頂

点 $adj(u_{i+1})_j$ と $adj(u_{i+1})_{j+1}$ の間にある u_i の隣接頂点の数を h_j^i で表し, u_i の隣接頂点に $adj(u_{i+1})_j$ が含まれるとき $h_j^{i'} = 1$, そうでないとき $h_j^{i'} = 0$ とする.

ある $L \subseteq L_2$ において, 任意の $u \in L$ に対し, ある $v \in L - \{u\}$ が存在して, $c_{uv} > c(d) \cdot c_{vu}$ とする. このとき, 有向グラフ $G = (L, E)$, $E = \{(u, v) \mid c_{uv} > c(d) \cdot c_{vu}\}$ を考える. 仮定より, $G = (L, E)$ には, あるサイクル $C(u_0, \dots, u_m)$ が存在する. 以下では, サイクル C の存在が矛盾を引き起こすことを示す.

$c_{u_i u_{i+1}}$ と $c_{u_{i+1} u_i}$ を h_j^i ($0 \leq j \leq d_{u_i}$), $h_j^{i'}$ ($1 \leq j \leq d_{u_i}$) を用いて表すと,

$$\begin{aligned} c_{u_i u_{i+1}} &= (h_1^i + h_2^{i'}) \times 1 + (h_2^i + h_3^{i'}) \times 2 + \dots \\ &\quad + (h_{d_{u_i}-1}^i + h_{d_{u_i}}^{i'}) \times d_{u_i-1} \\ &\quad + h_{d_{u_i}}^i \times d_{u_i} \\ &= \sum_{j=1}^{d_{u_i}} h_j^i \times j + \sum_{j=2}^{d_{u_i}} h_j^{i'} \times (j-1), \quad (1) \\ c_{u_{i+1} u_i} &= h_0^i \times d_{u_i} + (h_1^{i'} + h_1^i) \times (d_{u_i} - 1) + \dots \\ &\quad + (h_{d_{u_i}-1}^{i'} + h_{d_{u_i}-1}^i) \times 1 \\ &= \sum_{j=0}^{d_{u_i}-1} h_j^i \times (d_{u_i} - j) \\ &\quad + \sum_{j=1}^{d_{u_i}-1} h_j^{i'} \times (d_{u_i} - j) \quad (2) \end{aligned}$$

となる. G には辺 (u_0, u_1) が存在し, $c_{u_0 u_1} > c(d) \cdot c_{u_1 u_0}$ が成り立つ. これを, $h_j^i, h_j^{i'}$ を用いて書き下す:

$$\begin{aligned} &\sum_{j=\lceil \frac{c(d)}{c(d)+1} d_{u_1} \rceil}^{d_{u_1}} \{(c(d)+1)j - c(d)d_{u_1}\} \times h_j^1 \\ &\quad + \sum_{j=\lceil \frac{c(d)d_{u_1}+1}{c(d)+1} \rceil}^{d_{u_1}} \{(c(d)+1)j - c(d)d_{u_1} - 1\} \times h_j^{1'} \\ &> \sum_{j=0}^{\lfloor \frac{c(d)}{c(d)+1} d_{u_1} \rfloor} \{c(d)d_{u_1} - (c(d)+1)j\} \times h_j^1 \\ &\quad + \sum_{j=0}^{\lfloor \frac{c(d)d_{u_1}+1}{c(d)+1} \rfloor} \{c(d)d_{u_1} - (c(d)+1)j + 1\} \times h_j^{1'}. \quad (3) \end{aligned}$$

頂点 $adj(u_m)_j$ と $adj(u_m)_{j+1}$ の間に含まれる u_0 の隣接頂点の数を h_j^* で表し, u_0 の隣接頂点に $adj(u_m)_j$ が含まれるとき $h_j^{*'} = 1$, そうでないとき $h_j^{*'} = 0$ と

する. ここで, パス (u_1, \dots, u_m) に対して補助定理 3.1 を適用する. $adj(u_1)_{\lceil l(u_1) \rceil}$ は $adj(u_m)_{\lceil l(u_m) \rceil}$ と同じ位置かそれより右 (補助定理 3.1 (1)) なので,

$$\begin{aligned} &h_0^1 + h_1^{1'} + h_1^1 + \dots + h_{u_{\lceil l(u_1) \rceil}-1}^1 + h_{u_{\lceil l(u_1) \rceil}}^{1'} \\ &\geq h_0^* + h_1^{*'} + h_1^* + \dots \\ &\quad + h_{u_{\lceil l(u_m) \rceil}-1}^* + h_{u_{\lceil l(u_m) \rceil}}^{*'} \quad (4) \end{aligned}$$

同様に, $adj(u_1)_{\lfloor r(u_1) \rfloor+1}$ は $adj(u_m)_{\lfloor r(u_m) \rfloor+1}$ と同じ位置かそれより右 (補助定理 3.1 (2)) なので,

$$\begin{aligned} &h_{\lfloor r(u_m) \rfloor+1}^* + h_{\lfloor r(u_m) \rfloor+2}^{*'} + h_{\lfloor r(u_m) \rfloor+2}^* + \dots \\ &\quad + h_{u_{d_{u_m}-1}}^* + h_{u_{d_{u_m}}}^{*'} + h_{u_{d_{u_m}}}^* \\ &\geq h_{u_{\lfloor r(u_1) \rfloor+1}}^1 + h_{u_{\lfloor r(u_1) \rfloor+2}}^{1'} + h_{u_{\lfloor r(u_1) \rfloor+2}}^1 + \dots \\ &\quad + h_{u_{d_{u_1}-1}}^1 + h_{u_{d_{u_1}}}^{1'} + h_{u_{d_{u_1}}}^1 \\ &> \frac{1}{d_{u_1}} \left[\sum_{j=0}^{\lfloor \frac{c(d)}{c(d)+1} d_{u_1} \rfloor} \{c(d) \cdot d_{u_1} - (c(d)+1)j\} \times h_j^1 \right. \\ &\quad \left. + \sum_{j=0}^{\lfloor \frac{c(d)d_{u_1}+1}{c(d)+1} \rfloor} \{c(d) \cdot d_{u_1} - (c(d)+1)j + 1\} \times h_j^{1'} \right]. \quad (5) \end{aligned}$$

ここに, 中辺と右辺の不等号は式 (3) による. ここで, 式 (5) の右辺を計算すると

$$\begin{aligned} \text{式 (5) の右辺} &\geq h_0^1 + h_1^{1'} + h_1^1 + \dots \\ &\quad + h_{u_{\lceil l(u_1) \rceil}-1}^1 + h_{u_{\lceil l(u_1) \rceil}}^{1'} \end{aligned}$$

となるので, これと式 (4) とを組み合わせることにより, 次式を得る:

$$\begin{aligned} &h_{\lfloor r(u_m) \rfloor+1}^* + h_{\lfloor r(u_m) \rfloor+2}^{*'} + h_{\lfloor r(u_m) \rfloor+2}^* + \dots \\ &\quad + h_{u_{d_{u_m}-1}}^* + h_{u_{d_{u_m}}}^{*'} + h_{u_{d_{u_m}}}^* \\ &> h_0^* + h_1^{*'} + h_1^* + \dots \\ &\quad + h_{u_{\lceil l(u_m) \rceil}-1}^* + h_{u_{\lceil l(u_m) \rceil}}^{*'} \quad (6) \end{aligned}$$

以上の準備のもと, 式 (2) を用いて $c(d) \cdot c_{u_0 u_m} - c_{u_m u_0}$ を書き下し, 図 4 に示す性質を利用すると,

$$\begin{aligned} &c(d) \cdot c_{u_0 u_m} - c_{u_m u_0} \\ &= c(d) \left\{ \sum_{j=0}^{d_{u_m}} h_j^* \times j + \sum_{j=1}^{d_{u_m}} h_j^{*'} \times (j-1) \right\} \\ &\quad - \left\{ \sum_{j=0}^{d_{u_m}} h_j^* \times (d_{u_m} - j) \right. \\ &\quad \left. + \sum_{j=0}^{d_{u_m}} h_j^{*'} \times (d_{u_m} - j) \right\} \end{aligned}$$

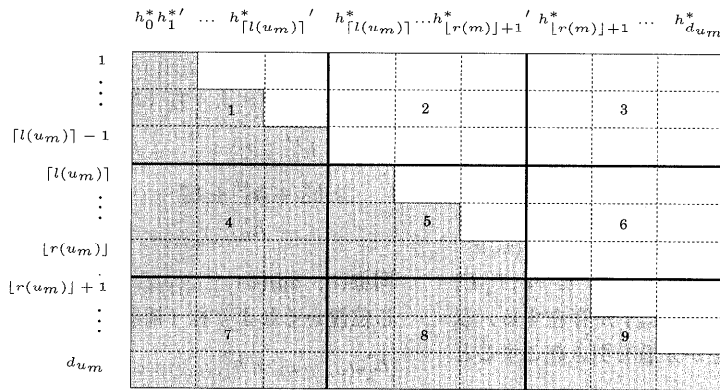


図4 $c(d) \cdot c_{u_0 u_m}$ と $c_{u_m u_0}$ との大小評価のために用いる性質（縦の長さが d_{u_m} 、横の長さが d_{u_0} であるような長方形を構成し、そして、縦方向には上から1ずつ、横方向には $h_{i-1}^* + h_i^{*'}$ ($i = 1, 2, \dots, m$) ずつ目盛を付与する。したがって、白部分の面積は u_0 を u_m の左においた場合の辺交差数に相当し、黒部分の面積は u_0 を u_m の右においた場合の辺交差数と、 u_0 と u_m が共有する隣接頂点の数の和に相当する。太線は u_m の隣接辺の左から数えて $[l(u_m)]$ 番目と $[r(u_m)]$ 番目による区切りである。図より、白部分の面積 $\geq 2, 3, 6$ の部分の面積の和、黒部分の面積 $\leq 1, 4, 5, 7, 8, 9$ の部分の面積の和であることが分かる。）

Fig. 4 Property for comparing the greatness between $c(d) \cdot c_{u_0 u_m}$ and $c_{u_m u_0}$.

$$\begin{aligned}
 &\geq \left\{ c(d) \sum_{j=[l(u_m)]+1}^{[r(u_m)]+1} (h_{j-1}^* + h_j^{*'}) \times [l(u_m)] \right. \\
 &\quad \left. + c(d) \sum_{j=[r(u_m)]+2}^{d_{u_m}} (h_{j-1}^* + h_j^{*'}) \times [r(u_m)] \right\} \\
 &\quad - \left\{ \sum_{j=0}^{[l(u_m)]} (h_{j-1}^* + h_j^{*'}) \times d_{u_m} \right. \\
 &\quad \left. + \sum_{j=[l(u_m)]+1}^{[r(u_m)]+1} (h_{j-1}^* + h_j^{*'}) \times [r(u_m)] \right. \\
 &\quad \left. + \sum_{j=[r(u_m)]+2}^{d_{u_m}} (h_{j-1}^* + h_j^{*'}) \times [l(u_m)] \right\} \tag{7}
 \end{aligned}$$

となる。次に、 $[a] - [b] > a - b$ (a, b は実数) を式(7)の右辺の第1項と第4項に、 $[a] + 1 - [b] > a - b - 1$ (a, b は実数) を第2項と第5項にそれぞれ適用し、 $c(d) \geq 2, d_{u_m} \geq 1$ を用いると、

$$\begin{aligned}
 \text{式(7)の右辺} &\geq \sum_{j=[r(u_m)]+2}^{d_{u_m}} (h_{j-1}^* + h_j^{*'}) \times d_{u_m} \\
 &\quad - \sum_{j=0}^{[l(u_m)]} (h_{j-1}^* + h_j^{*'}) \times d_{u_m}
 \end{aligned}$$

を得る。したがって、式(6)から、 $c(d) \cdot c_{u_0 u_m} - c_{u_m u_0} \geq 0$ が成り立つ。しかし、これは矛盾である。

なぜなら、仮定より G には辺 (u_m, u_0) が存在するので、 $c_{u_m u_0} > c(d) \cdot c_{u_0 u_m}$ が成立するからである。□
定理 3.1 から次の定理を得る。

定理 3.2 上記アルゴリズムの近似精度は、最適解の辺交差数の $c(d)$ 倍以下である。 □

L_2 に含まれる頂点の最大次数 d が小さいとき、 $c(d)$ の値は 2 に近く、 d が大きくなるにつれて（広義）単調に増加し、漸近的に 3 に近づく。具体的には、 $c(2) = c(3) = c(4) = 2, c(5) = c(6) = 7/3 + \epsilon$ (ϵ は任意の正数) となるので、以下の系が得られる。

系 3.1 上記アルゴリズムの近似精度は、第2層に含まれる頂点の最大次数が 4 以下のとき、最適解の辺交差数の 2 倍以下であり、最大次数が 6 以下のとき、最適解の辺交差数の $(7/3 + \epsilon)$ 倍 (ϵ は任意の正数) 以下である。 □

d を偶数とし、入力グラフの第2層の最大次数を d 以下としたとき、メジアン法が出力する解の辺交差数が最適解の辺交差数の $(3d - 2)/d$ 倍となる例を構成することができる。 $d = 2$ のとき、この値は 2 となり、 $d = 4$ のとき 2.5、 $d = 6$ のときには $8/3$ となる。 d が奇数のときも同様のことがいえる。したがって、最大次数が小さいとき、本アルゴリズムの方が理論的に良い近似精度を与えることが分かる。

4. 計算機実験による従来法との比較

2 種類の計算機実験を行って、前章で提案したアルゴリズムを重心法やメジアン法と比較する。本実験は

入力となるグラフの頂点数や辺の疎密さを変化させた際の、平均的な近似精度を調べることを目的としたため、計算時間を測定していない。

いずれの実験も、以下の手続きに従った。 L_1 に含まれる頂点数 n_1 , L_2 に含まれる頂点数 n_2 , および、辺の生成確率 p ($0 \leq p \leq 1$) をパラメータとして、2部グラフを生成した。具体的には、 L_1 の頂点を n_1 個、 L_2 の頂点を n_2 個生成し、すべての組 $(u, v) (u \in L_1, v \in L_2)$ に対して、確率 p で辺 (u, v) を生成する、という手順で2部グラフを生成した。次に、生成したグラフの L_1 の頂点の位置を固定し、提案アルゴリズム、重心法、メジアン法をそれぞれ独立に適用し、出力結果の辺交差数を求めた。設定した n_1, n_2, p に対して、「2部グラフの生成、各アルゴリズムの適用、出力結果の辺交差数の数えあげ」を50回繰り返した。なお、本実験では、 $n_1 = 2n_2$ とした。

第1の実験では、 $5 \leq n_2 \leq 10$ とした。そして、それぞれの n_2 に対して、辺の生成確率 p が 0.2, 0.5, 0.8 の3種類の場合について、実験を行った。なお、この実験では、最適解の辺交差数を求め、それぞれのアルゴリズムの出力結果の辺交差数と最適解の辺交差数との差を計算した。そして、50回の繰返しによる平均値と標準偏差を求めた。この結果を図5に示す。

第2の実験では、10から60の間で n_2 を10ずつ変化させ、それぞれの n_2 に対して、辺の生成確率 p が 0.2, 0.5, 0.8 の3種類の場合について実験を行った。この実験では、最適解の辺交差数を求めず、それぞれのアルゴリズムの出力結果の辺交差数を直接比較した。提案アルゴリズムの出力結果の辺交差数に対する重心法やメジアン法の出力結果の辺交差数の比を求めた。50回の繰返しによる平均値を表1に示す。

図5から、グラフの頂点数が少ない場合、辺の生成確率にかかわらず、提案アルゴリズムはほとんど毎回最適解を出力していることが分かる。また、頂点数に依存せず、安定して最適解を出力している。これに対し、重心法やメジアン法の出力結果には、辺の疎密にかかわらず、最適解からのずれがあり、頂点数が大きくなるにつれて、このずれは大きくなる。したがって、第2層に含まれる頂点数が少ない場合、辺の疎密にかかわらず、提案手法の方が優れていることが分かる。

また、表1から、グラフの頂点数を多くしても、やはり、辺の疎密にかかわらず、提案アルゴリズムの出力結果の方が良いことが分かる。しかし、辺の数が多くなるに従い、他手法に対する提案アルゴリズムの優位性が小さくなる。辺が密になるにつれて、また、辺の生成確率が一定のときに頂点数が多くなるにつれて、

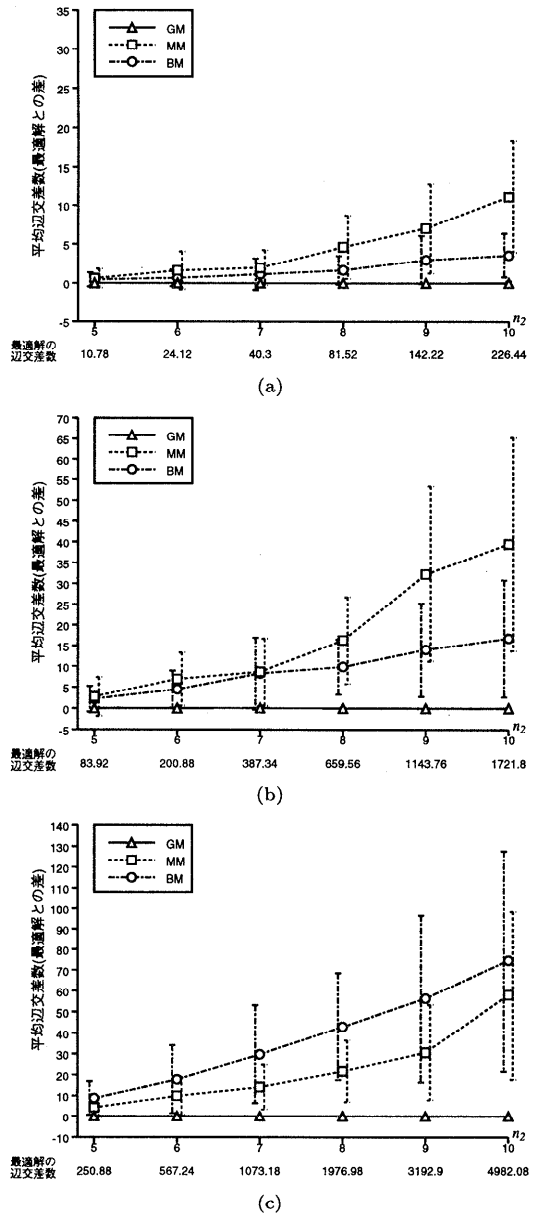


図5 頂点数が少ないグラフ ($5 \leq n_2 \leq 10, n_1 = 2n_2$) に対する各アルゴリズムの出力結果の辺交差数と最適解の辺交差数との差の平均値 (BMは重心法, MMはメジアン法, GMは提案アルゴリズムを表し、縦線は標準偏差を表す; (a)では $p = 0.2$, (b)では $p = 0.5$, (c)では $p = 0.8$)

Fig. 5 Average of the differences of the edge crossings obtained by three algorithms from the minimum edge crossing numbers for graphs with small-number nodes.

メジアン法の近似精度が上がることを考えると、提案アルゴリズムの近似精度は、辺の疎密や頂点数にかかわらず、安定して優れているといえる。

以上より、入力グラフの頂点数や辺の疎密にかかわ

表1 頂点数が多いグラフ ($10 \leq n_2 \leq 60, n_1 = 2n_2$) に対する各アルゴリズムの出力結果 (提案アルゴリズムの出力結果の辺交差数に対する他手法の出力結果の辺交差数の比の平均値)

Table 1 Average of ratios of the edge crossings obtained by the other algorithms to those of our algorithm for graphs with large-number nodes.

p	n_2	BM/GM	MM/GM	得られた辺交差数の平均値のオーダー
0.2	10	1.0169	1.0391	10^2
	20	1.0057	1.0303	10^3
	30	1.0026	1.0264	10^4
	40	1.0016	1.0204	10^4
	50	1.0011	1.0191	10^5
	60	1.0009	1.0166	10^5
0.5	10	1.0113	1.0212	10^3
	20	1.0033	1.0165	10^4
	30	1.0019	1.0129	10^5
	40	1.0010	1.0108	10^5
	50	1.0077	1.0095	10^6
	60	1.0006	1.0081	10^7
0.8	10	1.0155	1.0138	10^3
	20	1.0058	1.0077	10^4
	30	1.0028	1.0063	10^5
	40	1.0018	1.0052	10^6
	50	1.0014	1.0044	10^6
	60	1.0011	1.0040	10^7

らず、提案アルゴリズムの方が重心法やメジアン法より良い解を出力することが分かった。

5. おわりに

2部グラフ描画の辺交差数最小化問題に対し、2つの頂点間の順序による辺交差数の増減に着目した近似アルゴリズムを提案した。また、そのアルゴリズムの近似精度について述べた。すなわち、第2層の頂点の最大次数を d とすると、このアルゴリズムが出力する解の辺交差数は、最適解の辺交差数の $c(d)$ 倍以下であることを示した。 $c(d)$ は、2と3の間の値をとる、 d に関する広義単調増加関数である。 $d \leq 4$ の場合に $c(d) = 2$ となり、 d が大きくなるにつれて、 $c(d)$ の値は漸近的に3に近づく。メジアン法の近似精度は、最適解の辺交差数の3倍以下であり、実際に3倍に近い例を構成することができるので、第2層の頂点の最大次数が小さい場合、提案アルゴリズムはメジアン法より理論的に近似精度が良い。

また、計算機実験により、本アルゴリズムを重心法やメジアン法と比較し、頂点数や辺の疎密にかかわらず、本アルゴリズムが重心法やメジアン法より良い解を求めることを示した。グラフの頂点数が少ない場合、重心法やメジアン法の出力結果には、辺の疎密にかかわらず、最適解からのずれがあるが、本アルゴリズム

は、ほとんどの場合、最適解を出力することが分かった。また、グラフの頂点数を多くしても、やはり、辺の疎密にかかわらず、本アルゴリズムの出力結果の方が良いことが分かった。

今後の課題として、第2層の頂点の最大次数に依らない近似精度を示すことがあげられる。また、近似精度の上界だけではなく、MAX SNP 困難性など、近似精度の下界についての考察も必要である。さらに、近似精度だけではなく、計算時間について詳しく実験することも今後に残された課題である。また、今回は2層の階層グラフに焦点を当てたが、今後は3層の場合についても考える予定である。3層の場合、重心法やメジアン法の近似精度がひどく悪い例の構成が容易である。3層の場合、頂点間の順序による辺交差数の増減に着目したグリーディ法の近似精度が、どの程度になるかを検討する予定である。

参考文献

- 1) Batini, C., Furlani, L. and Nardelli, E.: What Is a Good Diagram?: A Pragmatic Approach, *Proc. 4th International Conference on Entity-Relationship Approach*, pp.312-319 (1985).
- 2) Di Battista, G., Eades, P., Tamassia, R. and Tollis, I.G.: Algorithms for Drawing Graphs: An Annotated Bibliography, *Computational Geometry*, 4, pp.235-282 (1994).
- 3) Eades, P., McKay, B.D. and Wormald, N.: On an Edge Crossing Problem, *Proc. 9th Australian Computer Science Conference*, pp.327-334 (1986).
- 4) Eades, P. and Wormald, N.: Edge Crossings in Drawings of Bipartite Graphs, *Algorithmica*, Vol.11, No.4, pp.379-403 (1994).
- 5) Esposito, C.: Graph Graphics: Theory and Practice, *Computers and Mathematics with Applications*, Vol.15, No.4, pp.247-253 (1988).
- 6) Jünger, M. and Mutzel, P.: 2-Layer Straight-line Crossing Minimization: Performance of Exact and Heuristic Algorithms, *Journal of Graph Algorithms and Applications*, Vol.1, No.1, pp.1-25 (1997).
- 7) Mäkinen, E.: Experiments on Drawing 2-level Hierarchical Graphs, *International Journal of Computer Mathematics*, Vol.36, pp.175-181 (1990).
- 8) Purchase, H.C., Cohen, R.F. and James, M.: Validating Graph Drawing Aesthetics, *Proc. Symposium on Graph Drawing (GD'95)*, Lecture Notes in Computer Science, Vol.1027, pp.435-446, Springer (1996).
- 9) Sugiyama, K., Tagawa, S. and Toda, M.:

Methods for Visual Understanding of Hierarchical Systems, *IEEE Trans. Systems, Man, and Cybernetics*, Vol.11, No.2, pp.109-125 (1981).

(平成 10 年 10 月 26 日受付)

(平成 11 年 9 月 2 日採録)



山口 敦子 (正会員)

1991 年九州大学理学部数学科卒業。1993 年同大学大学院総合理工学研究科情報システム学専攻修士課程修了。同年日立製作所基礎研究所に入社。1999 年生物分子工学研究所に出向。現在に至る。グラフ描画, 近似アルゴリズム, 生物情報解析等の研究に従事。



杉本 晃宏 (正会員)

1987 年東京大学工学部計数工学科卒業。1989 年同大学大学院工学系研究科修士課程修了(数理工学専攻)。同年日立製作所基礎研究所に入社。1991 年~95 年, ATR に出向。1999 年京都大学大学院情報学研究科講師, 現在に至る。博士(工学)。視覚情報処理や離散システム・アルゴリズム等に興味を持ち, 数理的手法に基づいたコンピュータビジョンの研究に従事。日本応用数理学会, 電子情報通信学会等会員。