

## 2部グラフ描画に対する近似アルゴリズム

山口 敦子      杉本 晃宏

日立製作所 基礎研究所

{atsuko, sugimoto}@harl.hitachi.co.jp

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となる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よ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

Advanced Research Laboratory, Hitachi, Ltd.

Hatoyama, Saitama 350-0395, Japan

The minimum edge crossing 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 4, the approximation ratio, i.e., the maximum ratio of the crossing number of the solution constructed by our algorithm to the minimum crossing number, is two, and this ratio monotonically increases to three as the maximum degree becomes high. We also presents our experiments on comparing our algorithm with the barycenter method or the median method. Our experiments shows that our 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。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。これに対し、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。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易く描く技術—グラフ描画問題とよばれている—が、ユーザインターフェイスをはじめとして、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易いかどうかを議論し評価するためには、見易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、見易さを一般論として議論するのは困難である<sup>1</sup>。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易さの基準もある。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、描画の見易さを評価する研究が数多く報告されている (例えば, [1, 5, 8])。このような見易さの基準として、これまで経験的に用いられているもの [2] を以下に挙げる:

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。

<sup>1</sup>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、ユーザインターフェイスなど、見易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。

3. 辺の長さが揃っていること.
4. 頂点・辺が対称的に描かれていること.

階層グラフ<sup>2</sup>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、見易く描画することがとくに重要となる。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易さの基準 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、見易さの基準として、辺の交差数最小化をとりあげる。

辺の交差数最小化を見易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: 三つ以上の層にまたがる辺に対し、辺と層とが交差する場所にダミー頂点を加え、すべての辺の終点と始点が隣接する二つの層に含まれるようにする。

Step 2: 第 1 層から順に、隣接する二つの層 (または三つの層) 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  $x$  軸方向の位置を変えることで辺交差数を減らす<sup>3</sup> (三つの層の場合には、中位層を除く二つの層 (上位層と下位層) の頂点の位置を固定し、中位層の頂点の  $x$  軸方向の位置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 2 で二つの層を選んだ場合に行う処理に焦点を当てる。二つの層からなる階層グラフは 2 部グラフと等価であるので、扱う問題は 2 部グラフ描画の辺交差数最小化となる。

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ックスについては、重心法 [9] やメジアン法 [4] などが知られている。また、遺伝的アルゴリズムを使った研究 [7] も報告されている。

重心法 [9] 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  $x$  座標の重心を計算し、その値をその頂点の  $x$  座標とする手法である。Eades ら [4] は、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得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、Mäkinen [6] は、計算機実験で重心法とメジアン法を比較した。その結果、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よい結果を出力することを示した。

本稿では、下位層の二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。 $d \leq 4$  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 3 に近づく。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粗密にかかわらず、本アルゴリズムの方がよ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

本節では、辺交差数最小化を見易さの基準とする 2 部グラフ描画問題を定式化する。

2 層グラフに対して、第 1 層 (上位層) を  $L_1$ 、第 2 層 (下位層) を  $L_2$  で表す。

<sup>2</sup>階層グラフとは  $H = (L_1, \dots, L_m, A)$  である。 $L_i$  は頂点集合、 $A$  は  $L_j$  の頂点から  $L_k (j < k)$  の頂点への有向辺の集合である。

<sup>3</sup>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。

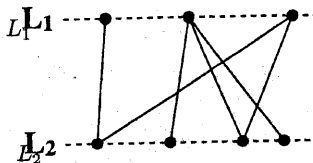


図 1: 2部グラフ

**定義 2.1 (2部グラフ)** 2部グラフとは、二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  $A$  とからなる有向グラフ  $G = (L_1, L_2, A)$  である (図1)。ただし、 $A$  の要素の始点は  $L_1$  に、終点は  $L_2$  に含まれるものとす。  
□

$L_1$  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。なお、Eadesら[3]は、この問題は、 $L_2$  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困難であることを示した。

**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、 $L_1$  に含まれる頂点の順序が  $x_1$  のときの辺の交差数の最小値  $opt(G, x_1)$  で以下のように表される。

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

### 3 近似アルゴリズムとその精度

本節では、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。始点を  $u$ , 終点を  $v$  とする辺を  $(u, v)$  で表し、 $u, v \in L_2$  に対して、 $c_{uv}$  を次のように定義する:

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

#### 3.1 アルゴリズム

ここでは、第2層の二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。

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

**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、比がもっとも小さい頂点  $u^*$  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  $\sum_{v \in L_2} \min\{c_{vu}, c_{uv}\}$  とを予め計算しておけば、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。

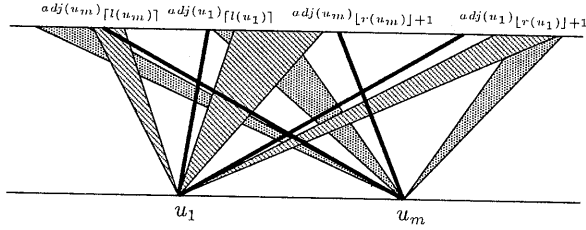


図 2:  $u_1, u_m \in L_2$  の隣接頂点の配置

### 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,

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

本節では,  $L_1$  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,

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

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,  $x_2(u) < x_2(v)$  ならば,  $c_{uv} \leq B \cdot c_{vu}$ 」が成り立てば, 「 $B \cdot \text{opt}(G, x_1) \geq \text{cross}(G, x_1, x_2)$ 」が成立することがわかる. 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, 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  $v \in L$  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  $adj(u)_1, adj(u)_2, \dots, adj(u)_{d_u}$  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,  $c_{u_i u_{i+1}} > c(d) \cdot c_{u_{i+1} u_i}$  かつ  $d_{u_i} \leq d$  なる頂点の列とする ( $i = 1, 2, \dots, m-1$ ). また, 頂点  $u$  に対して,  $l(u) = \frac{c(d)-1}{c(d)+1} d_u$ ,  $r(u) = \frac{2}{c(d)+1} d_u$  とおく. このとき, 以下の (1), (2) がともに成立する (図 2 参照).

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

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

この性質は, 2 頂点  $u_i, u_{i+1}$  が (1), (2) のいずれかをみたさない場合,  $c_{u_i u_{i+1}}$  が  $c_{u_{i+1} u_i}$  の  $c(d)$  倍を超えないことより, 容易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  $v \in L$  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  $adj(u_i)_1, adj(u_i)_2, \dots, adj(u_i)_{d_{u_i}}$  とする. 頂点  $adj(u_{i+1})_j$  と  $adj(u_{i+1})_{j+1}$  の間に含まれる  $u_i$  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  $v \in L$  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,

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

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

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

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

$u_0$  の隣接頂点を  $u_m$  の隣接頂点で分割したものを、 $h_0^*, h_1^{*'}, h_1^*, \dots, h_{d_{u_m}}^{*'}, h_{d_{u_m}}^*$  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,

$$h_0^1 + h_1^{1'} + \cdots + h_{u_{\lceil l(u_1) \rceil}}^{1'} \geq h_0^* + h_1^{*'} + \cdots + h_{u_{\lceil l(u_m) \rceil}}^{*'} \quad (4)$$

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,

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

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

$$\text{式 (5) の右辺} \geq h_0^1 + h_1^{1'} + \cdots + h_{u_{\lceil l(u_1) \rceil}}^{1'}$$

となるので、これと式 (4) とを組み合わせるにより、次式を得る:

$$h_{\lfloor r(u_m) \rfloor + 1}^* + \cdots + h_{u_{d_{u_m}}}^{*'} + h_{u_{d_{u_m}}}^* > h_0^* + h_1^{*'} + \cdots + h_{u_{\lceil l(u_m) \rceil}}^{*'} \quad (6)$$

以上の準備のもと、式 (1), (2) を用いて  $c(d) \cdot c_{u_0, u_m} - c_{u_m, u_0}$  を書き下し、図 3 に示す性質を利用すると、

$$\begin{aligned} c(d) \cdot c_{u_0, u_m} - c_{u_m, u_0} &= c(d) \left\{ \sum_{j=0}^{d_{u_m}} h_j^* \times j + \sum_{j=1}^{d_{u_m}} h_j^{*'} \times (j-1) \right\} - \left\{ \sum_{j=0}^{d_{u_m}} h_j^* \times (d_{u_m} - j) + \sum_{j=0}^{d_{u_m}} h_j^{*'} \times (d_{u_m} - j) \right\} \\ &\geq \left\{ c(d) \sum_{j=\lceil l(u_m) \rceil + 1}^{\lfloor r(u_m) \rfloor + 1} (h_{j-1}^* + h_j^{*'}) \times (\lceil l(u_m) \rceil) + c(d) \sum_{j=\lfloor r(u_m) \rfloor + 2}^{d_{u_m}} (h_{j-1}^* + h_j^{*'}) \times \lfloor r(u_m) \rfloor \right\} \\ &\quad - \left\{ \sum_{j=0}^{\lfloor l(u_m) \rfloor} (h_{j-1}^* + h_j^{*'}) \times d_{u_m} + \sum_{j=\lceil l(u_m) \rceil + 1}^{\lfloor r(u_m) \rfloor + 1} (h_{j-1}^* + h_j^{*'}) \times (\lfloor r(u_m) \rfloor) \right. \\ &\quad \left. + \sum_{j=\lfloor r(u_m) \rfloor + 2}^{d_{u_m}} (h_{j-1}^* + h_j^{*'}) \times \lceil l(u_m) \rceil \right\} \end{aligned} \quad (7)$$

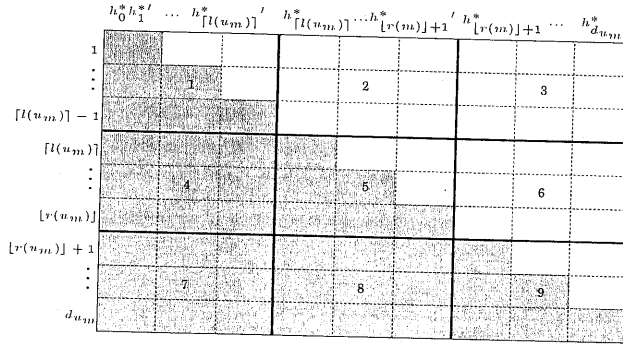


図 3:  $c(d) \cdot c_{u_0 u_m}$  と  $c_{u_m u_0}$  との大小評価のために用いる性質 (白の面積  $\leq 2, 3, 6$  の部分の面積の和, 黒の面積  $\geq 1, 4, 5, 7, 8, 9$  の部分の面積の和)

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、 $[a] + 1 - [b] > a - b - 1$  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、

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

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、 $c_{u_m u_0} > c(d) \cdot c_{u_0 u_m}$  が成立しているからである。□

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、 $c(2) = c(3) = c(4) = 2, c(5) = c(6) = 7/3 + \epsilon$  ( $\epsilon$  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 ( $\epsilon$  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。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よい近似精度を与えることがわかる。

#### 4 計算機実験による従来法との比較

二種類の計算機実験を行って、前節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。

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、 $n_1 = 2 \cdot n_2$  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 の三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 回の繰り返しによる平均値と標準偏差を求めた。この結果を図 4 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 の三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回の繰り返しによる平均値を表1に示す。

図4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わかる。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層に含まれる頂点の最大次数が少ない場合、提案手法の方が優れていることがわかる。

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よいことがわかる。しかし、辺の数が多くなるにしたがい、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、メジアン法の近似精度が上がること[4]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粗密にかかわらず、提案アルゴリズムの方が重心法やメジアン法よりよい解を出力することがわかった。

## 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、二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近似精度がよい。

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よ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。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よい。

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。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] C. Batini, L. Furlani, and E. Nardelli: What Is a Good Diagram?: A Pragmatic Approach, *Proc. of the 4th International Conference on Entity-Relationship Approach*, pp. 312-319, 1985.
- [2] G. Di Battista, P. Eades, R. Tamassia, and I. G. Tollis: Algorithms for Drawing Graphs: An Annotated Bibliography, *Computational Geometry*, 4 (1994), pp. 235-282.
- [3] P. Eades, B. D. McKay, and N. Wormald: On an Edge Crossing Problem, *Proc. of the 9th Australian Computer Science Conference*, pp. 327-334, 1986.
- [4] P. Eades and N. Wormald: Edge Crossings in Drawings of Bipartite Graphs, *Algorithmica*, 11 (1994), pp. 379-403.
- [5] C. Esposito: Graph Graphics: Theory and Practice, *Computers and Mathematics with Applications*, 15 (1988), 4, pp. 247-253.
- [6] E. Mäkinen: Experiments on Drawing 2-level Hierarchical Graphs, *International Journal of Computer Mathematics*, 36 (1990), pp. 175-181.
- [7] E. Mäkinen and M. Sieranta: *Genetic Algorithms for Drawing Bipartite Graphs*, Technical Report A-1994-1, Department of Computer Science, University of Tampere, Finland, 1994.
- [8] H. C. Purchase, R. F. Cohen, and M. James: Validating Graph Drawing Aesthetics, *Proc. of Symposium on Graph Drawing, GD'95 (Lecture Notes in Computer Science, Vol. 1027)*, pp. 435-446, Springer, 1996.
- [9] K. Sugiyama, S. Tagawa, and M. Toda: Methods for Visual Understanding of Hierarchical Systems, *IEEE Transactions on Systems, Man, and Cybernetics*, 11 (1981), 2, pp. 109-125.

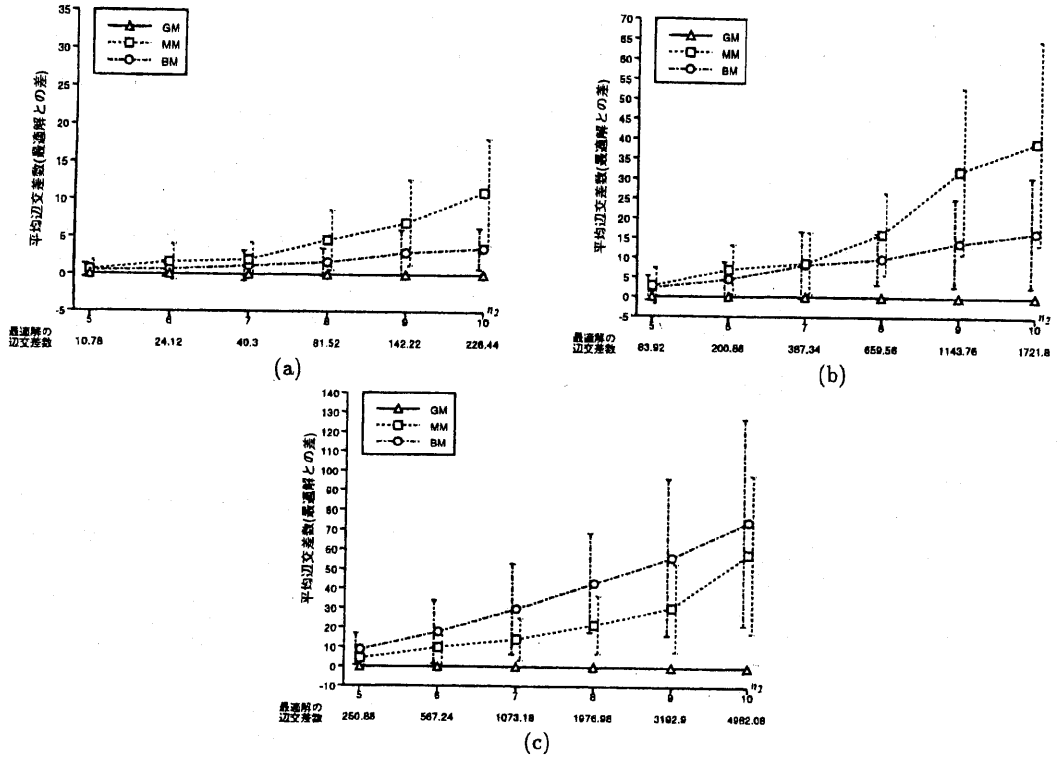


図 4: 頂点数が少ないグラフ ( $5 \leq n_2 \leq 10, n_1 = 2n_2$ ) に対する各アルゴリズムの出力結果の辺交差数と最適解の辺交差数との差の平均値 (BM は重心法, MM はメジアン法, GM は提案アルゴリズムを表し, 縦線は標準偏差を表す; (a) では  $p = 0.2$ , (b) では  $p = 0.5$ , (c) では  $p = 0.8$ )

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)

| $p$ | $n_2$ | 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 | $10^7$            |