G-110

2 階層にまたがる頂点をもつグラフの描画アルゴリズムの改良

An improved algorithm for drawing a hierarchical graph with vertices spanning two layers

池田 浩一郎 † Koichiro Ikeda
増田 澄男 † Sumio Masuda
山口 一章 † Kazuaki Yamaguchi

1．まえがき

階層グラフは，科目間関係図や作業工程図の表示など様々な用途に用いられている．階層グラフの自動描画法に関して広く研究が行われており [1]，代表的なアルゴリズムとして Sugiyama ら [2, 3] の方法が知られている．このアルゴリズムのステップの 1 つに，各階層における頂点順序の決定があり，辺交差数の削減を目的として，重心法 [2, 3]，メディアン法 [1]，田守らの手法 [4]，的場らの手法 [5] などが提案されている．また，頂点の配置座標の決定法についても研究されており，優先度法 [2, 3]，線形計画法を用いて辺長（厳密には，辺の 2 端点の x 座標の差）の総和を最小にする手法 [6] などが知られている．

階層描画アルゴリズムに関する従来の研究のほとんどは，各頂点が 1 つの階層上に存在する場合のものであった．しかし，例えば，通年開講の科目がある場合の科目間関係図では，2 階層にまたがる頂点が現れることになる．このような頂点のことを 2 階層頂点と呼ぶ．

2 階層頂点をもつグラフの描画では，各階層における頂点順序を適切に定めないと，2 階層頂点と辺，または 2 階層頂点同士の交差が生じることがある．このような交差は，通常の辺の交差に比べ，描画を非常に見づらくする．そこで，筆者ら [7] は，2 階層頂点を含むグラフに対し，2 階層頂点と辺あるいは 2 階層頂点同士の交差を作らないという制約の下で，辺交差数が比較的少ない描画を求めるアルゴリズムを提案した．本稿では，この方法を従来法と呼ぶ．

従来法は，上記の制約を満たすようにしながら各階層における頂点の初期順序を定めた後，重心法あるいは田守らの手法と同様の処理を繰返し実行することにより，辺交差数を削減していくものである．本研究では，この方法の初期順序決定部分に工夫を加えることにより，従来法よりさらに辺交差数を少なくすることができる描画アルゴリズムを提案する．

以下，**2.** では用語及び表記の定義をした後，重心法などのいくつかの方法と従来法について簡単に説明する．**3.** では提案法について述べ，**4.** で計算機実験の結果を示す．最後に **5.** において，本研究の結果をまとめ，今後の課題について述べる．

2．準　備

本章では，**2.1** においていくつかの定義を示す．次に，**2.2**~**2.4** において，重心法 [2, 3]，田守らの手法 [4] 及び線形計画法を用いた頂点座標の決定法 [6] を簡単に紹介する．最後に **2.5** では従来法 [7] について説明する．

2.1　諸定義

$G = (V, E)$ を任意の単純な連結グラフとする．V が部分集合 $L_1, L_2, \ldots, L_h$ に分割されており，同じ集合に属する頂点同士が隣接していないとき，G を階層グラフと呼ぶ．各集合 L_i $(1 \le i \le h)$ を G の第 i 階層と呼び，h を G の階層数と呼ぶ．

本研究では，図 1(a) 中の v のように 2 つの連続した階層にまたがる頂点（2 階層頂点と呼ぶ）をもつグラフ G について考える．G 中の各 2 階層頂点を，2 つの頂点とそれらをつなぐ辺で置き換える．2 頂点のうち上の階層に置くものを上部頂点，下の階層に置くものを下部頂点と呼び，それらをつなぐ辺を接続辺と呼ぶ．G に対してこのような処理をした後，階層をまたぐ各辺に対してダミー頂点 [2, 3] を設けてできる階層グラフを G' と表し，その階層を上から順に $L_1, L_2, \ldots, L_h$ とする．図 1(b) 参照．この図では頂点 v' が上部頂点，v'' が下部頂点であり，赤線で示した辺が接続辺である．また小さな黒丸で示した d がダミー頂点である．G' の頂点のうち，2 階層頂点の上部頂点あるいは下部頂点でないものを通常頂点と呼ぶ．グラフがダミー頂点を多くもつ場合には，ダミー頂点の共有処理 [8] を行って得られる階層グラフを G' とし，それを描画することも考えられる．

本研究では，G' を描画する際，平面上に等間隔に引い

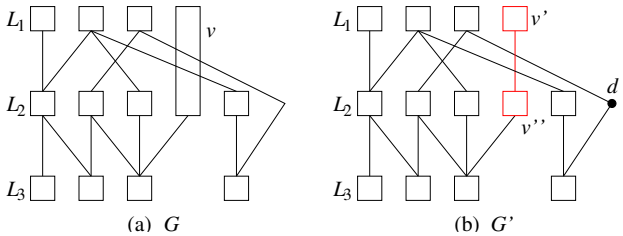


図 1　2 階層頂点を含むグラフの例

† 神戸大学，Kobe University

た h 本の水平線を考え，各階層 L_i の頂点を上から i 番目の水平線上に置くものとする．各階層 L_i の要素数を n_i と表す．

図 1(a) の例に見られる 2 階層頂点と辺の交差や，2 階層頂点の重なりが存在すると，描画が非常に見づらくなる．グラフ G' においては，2 階層頂点と辺の重なりは，接続辺と通常の辺の交差に対応する．また，2 階層頂点の重なりは，接続辺同士の交差に対応する．

G' の各頂点 v に対し，その隣接頂点のうち，1 つ上の階層に存在するものを上隣接頂点と呼ぶ．また，v の隣接頂点のうち，1 つ下の階層に存在するものを下隣接頂点と呼ぶ．$i = 1, 2, \ldots, h-1$ について，L_i の頂点と L_{i+1} の頂点との間の辺全てからなる集合を E_i と表す．

2.2 重心法

重心法 [2, 3] は，階層グラフの描画を求める際に，各階層上の頂点の配置順序を決定するための基本的な方法であり，2 つのフェーズからなる．

重心法ではまず，L_1 の頂点順序を固定し，L_2 の頂点順序を入れ替える．各頂点 $v \in L_2$ について，上隣接頂点が L_1 上で左から何番目に置かれているかという順番を求め，それらの平均値を計算する．この値のことを v の上重心と呼ぶ．そして，L_2 の頂点を上重心の非減少順に左から並べ換える．次に $i = 2, 3, \ldots, h-1$ についても同様に，L_i 上の頂点順序を固定し，L_{i+1} の頂点を上重心の順に並べ換えるという処理を行う．ここまでの処理を Down 過程と呼ぶ．次に，$i = h, h-1, \ldots, 2$ について，上とは逆に，L_i 上の頂点順序を固定して，L_{i-1} の頂点を，L_i 上の隣接頂点の順番の平均値（下重心）の順に並べ換えていく．この処理を Up 過程と呼ぶ．Down 過程と Up 過程を交互に，辺交差数が減少しなくなるまで（あるいは，あらかじめ設定した打ち切り回数に達するまで）繰り返す．以上の処理をフェーズ 1 と呼ぶ．

フェーズ 1 終了時点で，ある階層上に上重心の値が同じ頂点あるいは下重心の値が同じ頂点が存在する場合，それらの等重心の頂点をグループごとに逆順にし，フェーズ 1 と同様の処理を実行する．この処理をフェーズ 2 と呼ぶ．

2.3 田守らの手法

階層グラフに重心法を適用した後，田守らの手法 [4] を実行することによって，一般に辺交差数をさらに削減することができる．この方法は，ある頂点順序を初期解として，現在の解の近傍から次の解を探索することを繰り返すものである．ここで近傍は，ある階層 L_i $(1 \le i \le h)$ の頂点のたかだか 1 つを，その階層上の他の位置に移動して得られる $O(n_i^2)$ 通りの頂点順序である．このアルゴリズムでは，以下の 3 種類の処理によって頂点順序を入れ替えていく．

Up(i): L_{i+1} の頂点順序を固定し，L_i のたかだか 1 つの頂点を移動して得られる頂点順序の中での E_i の辺の交差数が最小となるものを求める．

Both(i): L_{i-1} と L_{i+1} の頂点順序を固定し，L_i のたかだか 1 つの頂点を移動して得られる頂点順序の中で，$E_{i-1} \cup E_i$ の辺の交差数が最小となるものを求める．

Down(i): L_{i-1} の頂点順序を固定し，L_i のたかだか 1 つの頂点を移動して得られる頂点順序の中で，E_{i-1} の辺の交差数が最小となるものを求める．

田守らの手法は，Up(1), Both(2), Both(3), . . ., Both($h-1$), Down(h) という一連の処理を，描画全体の辺交差数が減らなくなるまで，繰返し実行する．

2.4 線形計画法を用いた頂点座標の決定法

この方法は，各階層上の頂点の配置順序を決定した後に，描画中の辺長（厳密には，辺の 2 端点の x 座標の差）の総和が最小となるように，頂点の x 座標を決定するものである．頂点座標をこのように決定する問題は，各頂点の x 座標を表すような変数をもつ線形計画問題に帰着することができる [6]．各階層における頂点順序が決定した通りになるように，頂点の x 座標の大小関係を指定する制約条件を作る．目的関数は辺長の総和である．

本研究では，この方法によって頂点座標を求める際，ソルバーを用いて線形計画問題を解くものとする．

2.5 従来法

本節では，2 階層頂点をもつグラフ G の描画アルゴリズムである従来法 [7] について説明する．この方法は，G から階層グラフ G' を作成した後，以下のステップ (1), (2) を実行するものであり，アルゴリズム実行中のどの時点においても，G' 中の接続辺が交差を起こさないようにしている．

(1) 各階層における頂点順序の決定

(1a) 初期順序の決定

(1b) 通常頂点に対する重心法の適用

(1c) 2 階層頂点と通常頂点の移動

(2) 頂点座標の決定

以下，各ステップについて説明する．

2.5.1 ステップ (1a) 及び (1b)

従来法のステップ (1a) では，接続辺の交差が生じないように，各階層 L_i 上の頂点の初期順序を以下のように定める．

- i が奇数のとき，左から L_i 上に存在する上部頂点，通常頂点，L_i 上に存在する下部頂点の順に配置する．

- i が偶数のとき，左から L_i 上に存在する下部頂点，通常頂点，L_i 上に存在する上部頂点の順に配置する．

どの階層 L_i においても，通常頂点を並べる順序は任意とする．また，同じ階層上に上部頂点（あるいは下部頂点）が複数存在している場合にも，それらを並べる順序は任意とする．

ステップ (1b) では，2 階層頂点を各階層の端に置いたまま，通常頂点に対してのみ重心法を適用する．こうすることで，接続辺の交差がない状態を維持したまま，辺交差数を削減することができる．

2.5.2 ステップ (1c)

ステップ (1c) では，現在の頂点順序を暫定解として記録した後，Down 過程，Up 過程と呼ぶ処理によって，辺交差数のさらなる削減を試みる．

Down 過程では，階層 L_i 上に上部頂点が存在するような 2 階層頂点の移動を行った後に L_i 上の通常頂点の移動を行うという処理を，$i = 1, 2, \ldots, h$ の順に実行する．一方 Up 過程では，L_i 上に下部頂点が存在するような 2 階層頂点の移動を行った後に L_i 上の通常頂点の移動を行うという処理を，$i = h, h-1, \ldots, 1$ の順に実行する．通常頂点の移動では，各階層 L_i 上で通常頂点のみが連続する部分ごとに，田守らの手法の頂点移動の処理を繰り返し適用する．こうすることで，接続辺の交差を起こすことなく，交差数を削減することができる．2 階層頂点の移動方法については **2.5.3** で説明する．

いずれの過程においても，接続辺の交差がなく，暫定解よりも辺交差数が少ない頂点順序が見つかったとき，その並びを新たな暫定解として記録する．ステップ (1c) では，辺交差数が削減されなくなるまで Down 過程と Up 過程を交互に繰返し実行する．

2.5.3 2 階層頂点の移動

階層 L_i 上に上部頂点 v' が，L_{i+1} 上に下部頂点 v'' が存在する 2 階層頂点 v を移動させるものとする．L_i 及び L_{i+1} 上の他の頂点の順序を一旦固定しておき，接続辺の交差が生じない範囲で，

$C_v \triangleq$ (v' と上隣接頂点をつなぐ辺の交差総数)
+(v'' と下隣接頂点をつなぐ辺の交差総数)

の値が最小となる位置に v', v'' を移動する．

v' 以外の L_i 上の頂点を左から順に $u_1, u_2, \ldots, u_{n_i-1}$ とし，v'' 以外の L_{i+1} 上の頂点を左から順に $w_1, w_2, \ldots, w_{n_{i+1}-1}$ とする．L_i 上で，u_1 の左の位置に 0，u_s（$1 \le s \le n_i - 2$）と u_{s+1} の間の位置に s，u_{n_i-1} の右の位置に $n_i - 1$ とそれぞれ番号を付ける．L_{i+1} 上の位置に対しても同様に，$0, 1, \ldots, n_{i+1} - 1$ と番号を付ける．このとき，v', v'' の配置位置として $n_i \cdot n_{i+1}$ 通りの組合せが考えられるが，これら全てに対して接続辺の交差の有無と C_v の値の計算をすることは効率が悪い．

$s = 1, 2, \ldots, n_i - 1$ に対して

$a_s \triangleq \max\{k \mid u_1, \ldots, u_s$ のいずれかが w_k に隣接 $\}$

と定義し，$t = 1, 2, \ldots, n_{i+1} - 1$ に対して

$b_t \triangleq \max\{k \mid w_1, \ldots, w_t$ のいずれかが u_k に隣接 $\}$

と定義する．図 2 に簡単な例を示す．このとき，L_i 上の位置 p ($1 \le p \le n_i - 1$) に v' を，L_{i+1} 上の位置 p' ($1 \le p' \le n_{i+1} - 1$) に v'' をそれぞれ配置して接続辺の交差が生じないためには，$a_p \le p'$ かつ $b_{p'} \le p$ であることが必要である（図 2 の例では $a_2 = 3, b_3 = 2$ である．L_i 上の位置 2 に v' を，L_{i+1} 上の位置 3 に v'' を配置したとき，接続辺 (v', v'') は他の辺と交差しない）．詳細は省略するが，従来法はこの性質に基づいて，v', v'' の配置位置の組合せを絞って調べている．

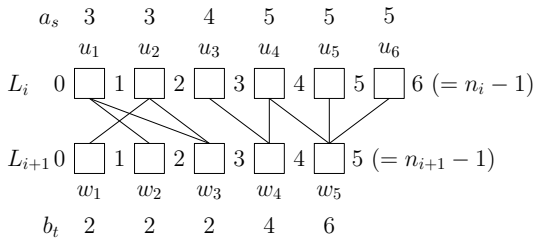


図 2　a_s と b_t の例

2.5.4 ステップ (2)

従来法のステップ (2) では，**2.4** で説明した方法と同様，辺長（厳密には，辺の 2 端点の x 座標の差）の総和が最小となるように，頂点座標を決定する．この問題を線形計画問題に帰着して解くのであるが，各 2 階層頂点を垂直に表示するために，

(上部頂点の x 座標) – (下部頂点の x 座標) = 0

という制約条件を，全ての 2 階層頂点に対して設けている．

3．提案法

従来法で得られる描画の辺交差数は，ステップ (1a) で定める初期順序に依存する．提案法では，主に初期順序の決定法の改良を行うことで，接続辺の交差を起こすことなく，辺交差数のさらなる削減を行うものである．

従来法と同様，提案法は以下のステップからなる．前述のとおり，従来法は，アルゴリズム実行中の任意の時点において，接続辺が交差を起こさないようにしている．これに対し，提案法は，ステップ (1a') の実行途中で一時的に接続辺の交差ができることを許している．

(1) 各階層における頂点順序の決定

(1a') 初期順序の決定

(1b') 通常頂点に対する重心法の適用

(1c) 2 階層頂点と通常頂点の移動

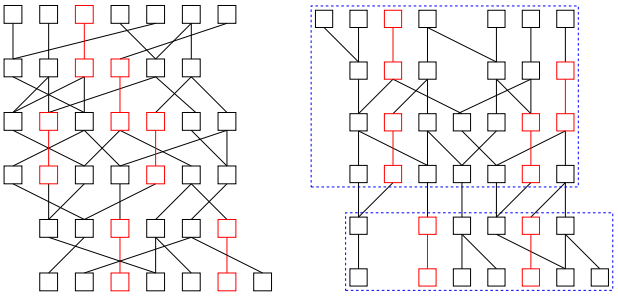
(2) 頂点座標の決定

ステップ (1c) 及び (2) は従来法と同じである．以下では，ステップ (1a') と (1b') について説明する．

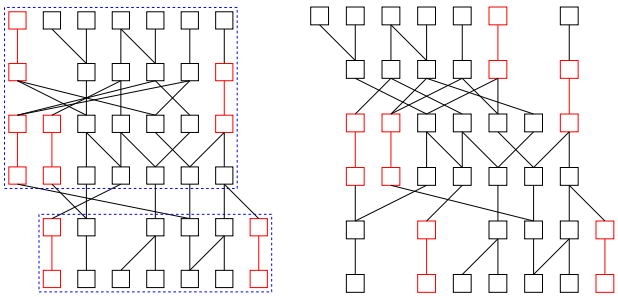
3.1 ステップ (1a')

ステップ (1a') では，グラフ G' に対して，まず頂点の仮配置を行う．これは各階層上で，頂点を任意の順序で並べるものである．次に，G' に重心法を適用する．ここまでの処理では，上部頂点及び下部頂点を通常頂点と同様に扱っており，一般に接続辺の交差が生じる．

図 3 は，2 階層頂点をもったある 6 階層グラフに対してステップ (1a') を実行したときに，各階層の頂点順序が変化していく様子を示したものである．同図 (a) が仮配置終了時，同図 (b) が重心法終了時の状態を示しており，これらの図では接続辺の交差が生じている．



(a) 仮配置終了時　(b) 重心法終了時



(c) 端寄せ処理終了時　(d) 2 階層頂点移動後

図 3　提案法のステップ (1a') の実行例

重心法の適用後，接続辺の交差がない状態にするために，端寄せと呼ぶ処理によって 2 階層頂点のみを移動する．以下，この処理について述べる．まず，G' において，以下の条件を満たす階層の集合 $\{L_i, L_{i+1}, \ldots, L_{i+k}\}$ $(k \geq 1)$ を全て求め，それらを $S_1, S_2, \ldots$ とする．

- L_i には通常頂点と上部頂点のみが存在する．
- L_{i+k} には通常頂点と下部頂点のみが存在する．
- $L_{i+1}, \ldots, L_{i+k-1}$ には上部頂点と下部頂点が存在する．

ある集合 S_j が 3 つ以上の階層 $\{L_i, L_{i+1}, \ldots, L_{i+k}\}$ からなるときには，以下の処理 putEnd1 により，2 階層頂点の配置を定める．一方，S_j が 2 階層 L_i, L_{i+1} のみを含むときには，処理 putEnd2 を用いる．

putEnd1： $L_i \sim L_{i+k-1}$ 上に上部頂点をもつ各 2 階層頂点を，上の階層から順に，左端，右端，左端, ... と交互に置いていく方法と，右端，左端，右端, ... の順に置いておく方法を試し，辺交差数が少なくなる方を選ぶ．

putEnd2： L_i 上に上部頂点をもつ各 2 階層頂点を，階層の左端もしくは右端のうち，辺交差数が少なくなる方に移動させる．

これらで計算する辺交差数は，S_j の階層中に存在する各上部頂点と上隣接頂点との間の辺，及び，各下部頂点と下隣接頂点との間の辺の交差数の総和である．図 3(c) は，同図 (b) の頂点順序に対して端寄せ処理を実行した結果である．ここでは，上の 4 階層に対して putEnd1 を，下の 2 階層に対して putEnd2 を，それぞれ適用している．

提案法のステップ (1a') では，最後に 2 階層頂点の移動を行う．端寄せ処理終了後の頂点順序と辺交差数を暫定解として記録した後，従来法のステップ (1c) と同様に，以下の Down 過程と Up 過程を交互に，辺交差数が削減されなくなるまで繰返し実行する．いずれの過程においても，接続辺の交差がなく，暫定解よりも辺交差数が少ない頂点順序が見つかった場合，その並びを新たな暫定解として記録する．

Down 過程： $i = 1, 2, \ldots, h-1$ の順に，階層 L_i 上に上部頂点が存在するような 2 階層頂点の移動を行う．

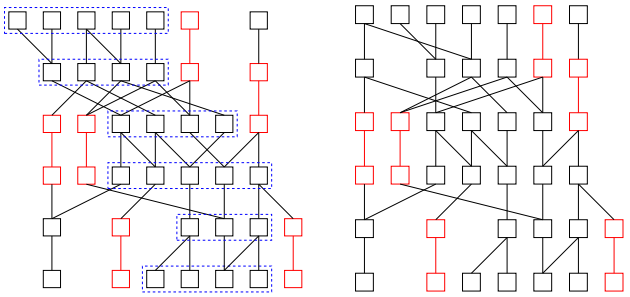
Up 過程： $i = h, h-1, \ldots, 2$ の順に，L_i 上に下部頂点が存在するような 2 階層頂点の移動を行う．

2 階層頂点の移動場所を探す方法は，**2.5.3** で述べたものと同じである．図 3(d) は，同図 (c) の頂点順序に対して，2 階層頂点移動処理を行った結果である．

3.2 ステップ (1b')

提案法のステップ (1b') では，従来法のステップ (1b) と同様に，通常頂点に対して重心法を適用する．ただし，従来法のステップ (1a) 終了時は，各階層において全ての通常頂点が連続して並んでいたが，提案法のステップ (1a') 終了時には必ずしもそうなっていない．そこでステップ (1b') では，接続辺の交差を作らないようにするために，まず各階層ごとに 2 個以上の通常頂点が連続して並んでいる部分を抽出する．そして，重心法の処理により，そのような各部分ごとに通常頂点の並び替えを行う．

ステップ (1b') の実行例を図 4 に示す．同図 (a) の頂点順序は図 3(d) と同じであり，各階層上で 2 個以上の通常



(a) 実行前（交差数：15）　(b) 実行後（交差数：14）

図 4　提案法のステップ (1b') の実行例

頂点が連続している部分を青の破線で示している．図 4(b) は重心法による並び替えが終了した状態である．

4．計算機実験

提案法の辺交差数削減効果を確認するため，従来法との比較実験を行った．本章では，その方法と結果について述べる．

4.1　実験方法

実験データとして，2 階層頂点を含むグラフ $G = (V, E)$ をランダムに作成した．G の階層数 h，頂点数 $|V|$，辺数 $|E|$，及び 2 階層頂点の個数 N の組合せ $(h, |V|, |E|, N)$ として $(6, 35, 40, 5)$, $(8, 55, 60, 5)$ の 2 通りを考え，それぞれについて 100 個のグラフを作成した．これらの各グラフから，**2.1** で述べた方法（ダミー頂点の共有処理 [8] を含む）によって階層グラフ G' を作成した．そして，従来法と提案法を実行し，(1a), (1b), (1c)（提案法では (1a'), (1b'), (1c)）の各ステップ終了時点での辺交差数，接続辺の交差数，及び実行時間を求めた．

使用した計算機の CPU は AMD A10-7800 Radeon R7,12 Compute Cores 4C，メモリは 8GB，OS は Windows 10 であり，プログラミング言語は Java SE8 である．

4.2　実験結果

実験結果を表 1 に示す．表中の各値は 100 個のデータに対する平均値である．参考のため，同じグラフ G' に対して，重心法，及び重心法と田守らの手法を実行した結果も表に載せている．

表 1 より，提案法は従来法と同様，接続辺の交差のない頂点順序を求めることができている．また，提案法はステップ (1a') の終了時点で，辺交差数を，従来法のステップ (1c) 終了時点と近い値にすることができている．このことから，ステップ (1a') で行っているように，全頂点に対して重心法を適用し，接続辺の交差を解消した後に 2 階層頂点の移動をする方法が有効であることが分かる．提案法は，ステップ (1b'), (1c) の処理でさらに交差数を削減できており，最終的な描画の辺交差数は従来法よりもかなり少なくなっている．

表 1: 実験結果

(a) $(h, |V|, |E|, N) = (6, 35, 40, 5)$ のグラフに対する結果

手　法		辺交差数	接続辺の交差数	実行時間 [ms]
従来法	(1a) 終了時	154.23	0.00	0.08
	(1b) 終了時	49.29	0.00	18.93
	(1c) 終了時	32.88	0.00	28.05
提案法	(1a') 終了時	36.46	0.00	23.06
	(1b') 終了時	32.37	0.00	26.07
	(1c) 終了時	26.40	0.00	34.52
重心法		24.43	2.35	18.32
重心法+田守らの手法		23.33	2.18	26.13

(b) $(h, |V|, |E|, N) = (8, 55, 60, 5)$ のグラフに対する結果

手　法		辺交差数	接続辺の交差数	実行時間 [ms]
従来法	(1a) 終了時	345.67	0.00	0.13
	(1b) 終了時	80.88	0.00	73.01
	(1c) 終了時	53.17	0.00	96.48
提案法	(1a') 終了時	53.69	0.00	92.60
	(1b') 終了時	49.17	0.00	99.62
	(1c) 終了時	38.77	0.00	122.70
重心法		42.52	2.60	79.95
重心法+田守らの手法		35.97	2.50	95.52

参考として挙げた 2 つの方法のうち，重心法は，6 階層のデータでは辺交差数を提案法より少なくすることができているが，8 階層のデータではより多くの辺交差を作っている．一方，重心法の後で田守らの方法を実行した場合には，いずれのデータに対しても，提案法より辺交差数を若干少なくすることができている．ただし，これらの方法はいずれも接続辺の交差を起こしている．

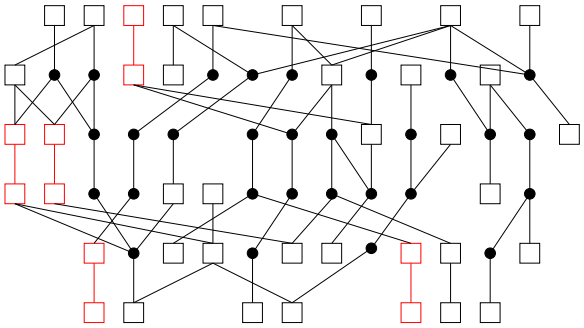
図 5 に，$(h, |V|, |E|, N) = (6, 35, 40, 5)$ のあるグラフに対する従来法と提案法の描画例を示す．このグラフに対しては，提案法による辺交差数が従来法より大幅に少なくなっている．

5．あとがき

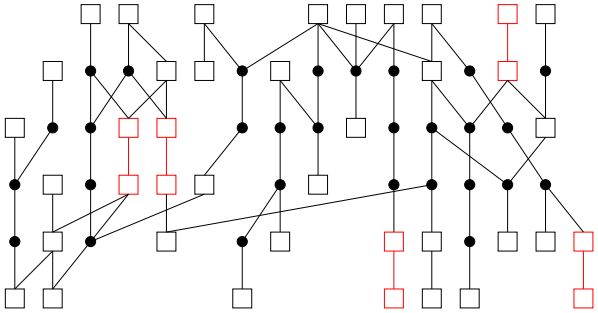
本研究では，2 階層頂点を含むグラフに対して，接続辺の交差を起こさないようにしながら，辺交差数が少ない描画を求めるアルゴリズムを提案した．提案法は，従来法 [7] における頂点の初期順序決定部分を改良したものである．

今後の課題として以下の 2 つが考えられる．

課題 1：　接続辺の交差を起こさないようにしながら，辺交差数をさらに削減できる方法について検討すること．



(a) 従来法（交差数：42）



(b) 提案法（交差数：13）

図 5　従来法と提案法の描画例

課題 2：　3 階層以上にまたがる頂点を含むグラフに対しても適用できるようにアルゴリズムを拡張すること．

2 階層頂点をもたない通常の階層グラフの場合，辺交差数を削減するために，道の移動処理と呼ばれる処理 [5] が有効であることが知られている．課題 1 に対する 1 つの方法として，2 階層頂点が存在する場合にも適用できるよう，この処理を拡張することが考えられる．

参考文献

[1] O. Bastert and C. Matuszewski, “Layered drawings of digraphs,” Drawing Graphs, Lecter Notes in Computer Science, vol.2025, pp.87-120, Springer, Berlin, 2001.

[2] K. Sugiyama, Graph Drawing and Applications - For Software and Knowledge Engineering, World Scientific, Singapore, 2002.

[3] K. Sugiyama, S. Tagawa, and M. Toda, “Methods for visual understanding of hierachical system structures,” IEEE Trans. Systems, Man and Cybernetics, vol.SMC-11, pp.109-125, 1981.

[4] 田守健太郎, 山口一章, 増田澄男, “局所探索法による階層的描画の辺交差数削減,” 電子情報通信学会論文誌 (A), vol.J92-A, pp.55-61, 2009.

[5] 的場郁典, 増田澄男, 荒木徹也, 斎藤寿樹, 山口一章, “階層グラフ描画における道の移動処理を用いた頂点順序決定法,” 電子情報通信学会論文誌 (A), vol. J98-A, pp. 152-164, 2015.

[6] E.R. Gansner, E. Koutsofios, S.C. North and K.-P. Vo, “A technique for drawing directed graphs,” IEEE Trans. Software Engineering, vol.19, pp.214-230, 1993.

[7] 池田浩一郎, 増田澄男, 山口一章, “2 階層にまたがる頂点をもつグラフの描画アルゴリズム” 平成 29 年電気関係学会関西連合大会, G10-7, 2017.

[8] 荒木徹也, 増田澄男, 山口一章, “階層グラフ描画におけるダミー頂点の共有,” 電子情報通信学会論文誌 (A), vol. J94-A, pp. 950-959, 2011.