

直交順序を保存する方形の最小面積非交差再配置問題

林 邦彦^{†*} 井上美智子[†] 増澤 利光[†] 藤原 秀雄[†]

A Layout Adjustment Problem for Disjoint Rectangles Preserving
Orthogonal Order

Kunihiko HAYASHI^{†*}, Michiko INOUE[†], Toshimitsu MASUZAWA[†],
and Hideo FUJIWARA[†]

あらまし あらかじめ平面上に配置されている n 個の方形の集合に対して、直交順序を保存し、方形同士が交差しない、という制約のもとで、面積最小の再配置を求める問題について考察する。三末らはこの問題に対し、 $O(n^2)$ 時間の発見的アルゴリズムを提案した。本論文ではまず、ある面積以下で再配置可能かどうかを判定する問題が、NP 完全であることを証明する。更に、面積最小の再配置を求める $O(n^2)$ 時間の発見的再配置アルゴリズムを与え、このアルゴリズムで得られる再配置面積が、三末らの手法で得られる面積以下であることを証明する。また、ランダムな初期配置に対して計算機実験を行い、特に方形数が多い場合に、三末らの結果に比べて 15~20% の面積で再配置できることを示す。

キーワード グラフ、描画、アルゴリズム、直交順序、計算量

1. ま え が き

グラフは様々な構造や関係を表現するための基礎的なモデルであり、データ構造や問題を記述するために用いられる。グラフを視覚化すること、つまり平面上に描画することは、その構造を理解するために有効な手段であるが、人手によるグラフの描画はグラフの規模の増大に伴って困難となるため、グラフを自動的に描画する必要がある。

グラフを自動的に描画する研究として、様々な手法が提案されている [1], [2]。そのほとんどはグラフの構造のみが与えられ、新たに描画を行うものである。しかし、与えられた描画に対して、節点及び辺の挿入や削除、節点を含む情報の変更などの動的な操作が加えられた後、もとの描画の構造を保つように再描画する方法も研究されている [3]。

新たに描画を行うアルゴリズムを再描画に用いると、節点の相対位置が全く変わってしまうことがあるため、もとの描画との位置関係が把握できなくなることがある。そのため、再描画を行う際には各節点間の位置関

係が重要となる。Eades ら [4] は、メンタルマップと呼ばれる節点間の位置関係に関する指標を提案し、次のような数学的モデルを挙げている。

- 直交順序：再配置前後で節点の上下左右の関係を保存する。
 - 近接関係：近接する節点は、再描画後も近接して配置する。
 - トポロジー：再配置前後で配置によって定まる 2 次元平面上の領域間の隣接関係を保存する。
- これらの指標を満たす再描画アルゴリズムがいくつか提案されている [5], [6]。

各節点が情報を保持しているようなグラフの場合、通常節点を方形や円などで表し、内部に保持する情報を表示する。この情報に変更が加えられると、節点を拡張する必要があるため、他の節点と交差することがある。更に、描画領域は画面や紙面のように通常限界があるため、交差を解消するために節点を移動した結果、節点が領域外に押し出される可能性がある。全体の情報をできるだけ多く表示するためには、節点同士が交差せず、描画面積ができるだけ小さくなるような描画が得られることが望ましい。

本論文では、節点の交差に重点をおき、単純化のためグラフの辺の接続関係を考慮せず、節点を方形と置き換えてグラフの描画問題を方形の配置問題として取

[†] 奈良先端科学技術大学院大学情報科学研究科, 生駒市
Graduate School of Information Science, Nara Institute of
Science and Technology, Ikoma-shi, 630-0101 Japan

* 現在, 松下電器産業株式会社

り扱うものとする．すなわち，直交順序を保存し，方形の非交差を満たす面積最小の再配置を求める問題について考察する．三末ら [5] がこの問題に対する発見的手法を提案している．本論文ではまず，直交順序を保存する方形の非交差最小面積再配置問題を定義し，この問題が整数座標系で NP 完全であることを示す．更に，文献 [5] と同じ漸近的時間計算量で，同文献で得られる面積以下の再配置を求める発見的再配置アルゴリズムを示す．このアルゴリズムの時間計算量は $O(n^2)$ である．また，ランダムな初期配置に対して計算機実験を行い，方形数が多い場合に，文献 [5] の結果に対して 15 ~ 20% の面積で再配置できることを示す．

本論文の構成は以下のとおりである．まず，2. で本論文で用いる用語及び記号を定義し，3. で方形の最小面積非交差再配置問題 LADR を定義する．4. では問題 LADR が整数座標系で NP 完全であることを証明する．更に 5. では，三末らのアルゴリズムを紹介し，このアルゴリズムより面積の小さい再配置を求めるアルゴリズムを提案する．最後に，6. で本論文のまとめと今後の課題について述べる．

2. 諸 定 義

R を任意の n 個の方形 v_1, v_2, \dots, v_n からなる方形集合とする．各方形 v_i はそれぞれ正整数の x 方向の幅 w_i ， y 方向の幅 h_i をもつ．方形 v_i を $\langle w_i, h_i \rangle$ と表すことがある．ここで，方形集合 R の 2 次元空間上の配置を考える．方形集合 R の配置 π_R は，整数座標系を用いるとき写像 $\pi_R: R \rightarrow \mathcal{Z}^2$ ，実数座標系を用いるとき写像 $\pi_R: R \rightarrow \mathcal{R}^2$ で表され，各方形 $v_i \in R$ の中心の x 座標値， y 座標値がそれぞれ x_i, y_i のとき， $\pi_R(v_i) = (x_i, y_i)$ と定義する．ここで $\mathcal{Z}^2, \mathcal{R}^2$ はそれぞれ，2 次元整数座標空間，2 次元実数座標空間である．本論文で扱う配置では，方形 v_i の長さ w_i (x 方向の幅) の辺は 2 次元平面の x 軸と水平になるように置くものとし，方形の回転は許さない．

R に含まれる方形 v_i の，配置 π_R における左右の境界の x 座標をそれぞれ $left_\pi(v_i), right_\pi(v_i)$ ，上下の境界の y 座標をそれぞれ $top_\pi(v_i), bottom_\pi(v_i)$ と定義する． π_R についても同様に，次のように定義する．

$$left(\pi_R) = \min_{v_i \in R} left_\pi(v_i)$$

$$right(\pi_R) = \max_{v_i \in R} right_\pi(v_i)$$

$$top(\pi_R) = \min_{v_i \in R} top_\pi(v_i)$$

$$bottom(\pi_R) = \max_{v_i \in R} bottom_\pi(v_i)$$

更に， π_R の x 方向の幅， y 方向の幅をそれぞれ $W_x(\pi_R), W_y(\pi_R)$ と表し，次のように定義する．

$$W_x(\pi_R) = right(\pi_R) - left(\pi_R)$$

$$W_y(\pi_R) = bottom(\pi_R) - top(\pi_R)$$

方形と同様に， π_R に対して $\langle W_x(\pi_R), W_y(\pi_R) \rangle$ という表記を用いる． π_R の面積 $S(\pi_R)$ を $S(\pi_R) = W_x(\pi_R) \cdot W_y(\pi_R)$ と定義する．

3. 方形の最小面積非交差再配置問題

方形の最小面積非交差再配置問題を定義する．

[定義 1] (最小面積非交差再配置問題) 方形集合 R とその配置 π_R 及び正の整数 K が与えられる．任意の二つの方形が同じ中心座標をもたないとする．すなわち，任意の $v_i, v_j \in R (i \neq j)$ について， $\pi_R(v_i) \neq \pi_R(v_j)$ である．このとき，以下の制約 (1), (2) を満たし，かつ $S(\pi'_R) \leq K$ となる R の配置 π'_R が存在するかどうかを判定する問題を，最小面積非交差再配置問題という．ただし，以下の制約 (1), (2) では，各 $v_i \in R$ について $\pi_R(v_i) = (x_i, y_i)$ ， $\pi'_R(v_i) = (x'_i, y'_i)$ とする．

(1) 再配置前後で直交順序を保存する．つまり，各方形 $v_i, v_j \in R$ に対して，次式が成り立つ．

$$x_i < x_j \Leftrightarrow x'_i < x'_j, \quad x_i = x_j \Leftrightarrow x'_i = x'_j$$

$$y_i < y_j \Leftrightarrow y'_i < y'_j, \quad y_i = y_j \Leftrightarrow y'_i = y'_j$$

(2) 再配置後，方形は互いに交差ししない．つまり，各方形 $v_i, v_j \in R (i \neq j)$ に対して次式が成り立つ．

$$|x'_i - x'_j| \geq \frac{w_i + w_j}{2} \quad \text{又は} \quad |y'_i - y'_j| \geq \frac{h_i + h_j}{2}$$

以下では，最小面積非交差再配置問題を LADR と表し，整数座標系を用いるとき，特に ILADR と表す．

4. 方形の最小面積非交差再配置問題の計算複雑度

本章では，問題 ILADR が NP 完全であることを示す．ILADR が NP に属するのは明らかである．ここでは，まず ILADR の NP 完全性を証明するために，NP 完全問題である 3-SAT [7] を ILADR に帰着する．

3-SAT は次のように定義される．ブール変数の集合を $X = \{x_1, x_2, \dots, x_r\}$ とする．各変数 $x_k \in X$ 又

はその否定 $\overline{x_k}$ をリテラル (literal) と呼び、三つの異なるリテラルの論理和を節 (clause) と呼ぶ。有限個の節の論理積をブール代数式 E とする。節の個数を $m (\geq 1)$ とし、 m 個の節を F_1, F_2, \dots, F_m と表し、 $E = F_1 \wedge F_2 \wedge \dots \wedge F_m$ とする。節 F_i に含まれるリテラルを $y_{i,1}, y_{i,2}, y_{i,3}$ と表し、 $F_i = y_{i,1} \vee y_{i,2} \vee y_{i,3}$ とする。3-SAT とは、与えられた E を true にするような各変数への true 又は false の割当てが存在するかどうか、つまり、 E の充足可能性を判定する問題である。

4.1 3-SAT から問題 ILADR への帰着

3-SAT から問題 ILADR への帰着性を示す。ブール代数式 E から方形集合 $R(E)$ とその初期配置 $\pi_{R(E)}$ を定義する。最も左上の方形の左上角を座標原点 $(0, 0)$ とする。 $R(E)$ は各ブール変数 x_k に対応する方形集合 $VR(x_k)$ 、各節 F_i に対応する方形集合 $CR(F_i)$ 、方形集合 R' 及び方形 R^* からなる。 $R(E)$ の初期配置 $\pi_{R(E)}$ の概要を図 1 に示す。 $\pi_{R(E)}$ は 3-SAT の各ブール変数 x_k がすべて true である場合に対応する。以下、方形集合 $R(E)$ とその初期配置 $\pi_{R(E)}$ について順に述べる。なお、方形集合 $VR(x_k)$ に対し、初期配置 $\pi_{R(E)}$ の定義域を $VR(x_k)$ に制限した初期配置 $\pi_{R(E)}|_{VR(x_k)}$ を、ここでは簡単のため単に $\pi_{VR(x_k)}$ と表記する。 $R(E)$ に含まれる他の方形集合に対しても同様の表記を用いる。

R' は $VR(x_k)$ や $CR(F_i)$ の配置を制限するためのもので、方形集合 $R'_i (i = 1, \dots, m)$ を含んでいる。 R'_i 及び R' の初期配置を図 2 に示す。図の中で用いている r は 3-SAT におけるブール変数の個数である。各変数 x_k に対応する方形集合 $VR(x_k)$ の初期配置 $\pi_{VR(x_k)}$ を図 3(a) に示す。 $VR(x_k)$ は m 個の

$\langle 2, 2(4r + 3) \rangle$ の方形 $vc_{k,i}$ を含んでおり、初期配置ではこれらは縦にすき間なく配置される。 $\pi_{R(E)}$ では、各 $VR(x_k)$ に対して $top(\pi_{VR(x_k)}) = top(\pi_{R'})$ となるように配置する (図 1 参照)。

次に、各節 F_i に対応する方形集合 $CR(F_i)$ について述べる。まず、 $CR(F_i)$ に含まれる各リテラル $y_{i,j} (j = 1, 2, 3)$ に対応する方形集合 $LR(y_{i,j})$ につい

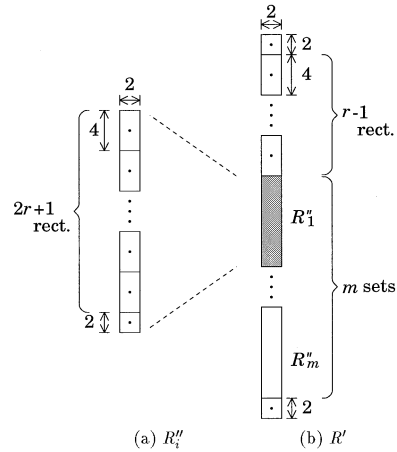


図 2 R'_i 及び R' の配置
Fig. 2 The layout of R'_i and R' .

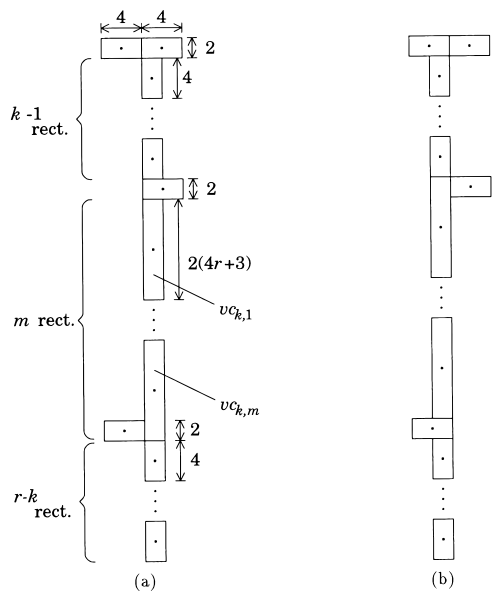


図 3 $VR(x_k)$ の配置
Fig. 3 Two layouts of $VR(x_k)$.

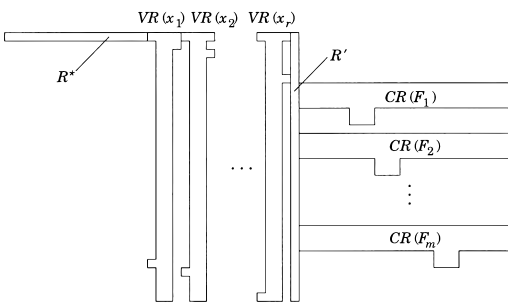


図 1 $R(E)$ の初期配置 $\pi_{R(E)}$ の概略
Fig. 1 Outline of the initial layout $\pi_{R(E)}$ of a rectangle set $R(E)$.

て述べる. $LR(y_{i,j})$ は二つの種類がある(図4, 図5). いずれの場合も, $LR(y_{i,j})$ は $\langle 2, 2 \rangle$ の方形 $vs_{i,j}$ を含んでいる. $LR(y_{i,j})$ の初期配置 $\pi_{LR(y_{i,j})}$ はそれぞれ, ある変数 x_k に対して, 対応するリテラルが $y_{i,j} = x_k$ ならば図4(a), $y_{i,j} = \bar{x}_k$ ならば図5(a)とする. $\pi_{LR(y_{i,j})}$ の上端の境界の y 座標と $vs_{i,j}$ の y 座標との差を $d_s(\pi_{LR(y_{i,j})})$ とおく. ただし, ここでは $d_s(\pi_{LR(y_{i,j})})$ を単に d_s と表すことにする.

$CR(F_i)$ は, 各 $LR(y_{i,j})$ 及び $LR'(y_{i,j})$ ($j = 1, 2, 3$), 更に $\langle 36(m+i-2), 4 \rangle$ の方形 vl_i , $\langle 36(2m-i-1)+12, 4 \rangle$ の方形 vr_i を含んでいる. $LR'(y_{i,j})$ は $\langle 4, 4 \rangle$ の方形 $ve_{i,j}$ と $\langle 6, 4 \rangle$ の方形 $vf_{i,j}$ からなる. $CR(F_i)$ の初期配置 $\pi_{CR(F_i)}$ を図6に示す. $CR(F_i)$ に含まれる各 $LR(y_{i,j})$ は, $y_{i,j}$ に対応する変数が x_k 又はその否定ならば, $top(\pi_{LR(y_{i,j})}) = bottom(\pi_{LR'(y_{i,j})}) + 4(k-1)$ となるように配置する. $LR'(y_{i,j})$ と $LR(y_{i,j})$ は, $left(\pi_{LR(y_{i,j})}) = left_\pi(ve_{i,j}), right(\pi_{LR(y_{i,j})}) = right_\pi(vf_{i,j})$ となるように配置する.

各 $VR(x_k), CR(F_i), R'$ 及び R^* から $R(E)$ の初期配置 $\pi_{R(E)}$ を図1のように作る. $\pi_{R(E)}$ では, $top_\pi(vl_i) = top(\pi_{R'_i})$ となるように $CR(F_i)$ と R' を配置する(図7参照). このとき, $\pi_{LR(y_{i,j})}$ の $vs_{i,j}$ 以外の各方形の y 座標は, R' に含まれるある方形の y 座標と等しい. $y_{i,j} = x_k$ 又はその否定であるとす

ると, $\pi_{LR(y_{i,j})}$ に含まれる $vs_{i,j}$ と $\pi_{VR(x_k)}$ に含まれる $vc_{k,i}$ の y 座標はそれぞれ,

$$\begin{aligned} & (vs_{i,j} \text{ の } y \text{ 座標}) \\ & = 2 + 4(r-1) + \{4(2r+1) + 2\}(i-1) + 4 \\ & \quad + 4(k-1) + 5 \\ & = 4(r+k) + 2(4r+3)(i-1) + 3 \end{aligned}$$

$$\begin{aligned} & (vc_{k,i} \text{ の } y \text{ 座標}) \\ & = 2 + 4(k-1) + 2 + 2(4r+3)(i-1) \\ & \quad + (4r+3) \\ & = 4(r+k) + 2(4r+3)(i-1) + 3 \end{aligned}$$

であるから, $vs_{i,j}$ と $vc_{k,i}$ は同じ y 座標をもつ. したがって, 制約(1)を満たす再配置においてもこれらは同じ y 座標をもつ. また, $CR(F_i)$ と $CR(F_{i+1})$ の配置では, それらに属する方形が交差しないように

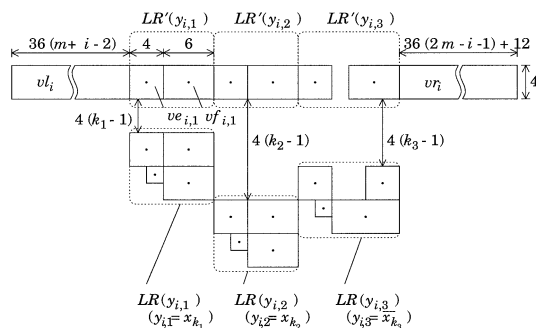


図6 $CR(F_i)$ の初期配置
Fig. 6 The initial layout of $CR(F_i)$.

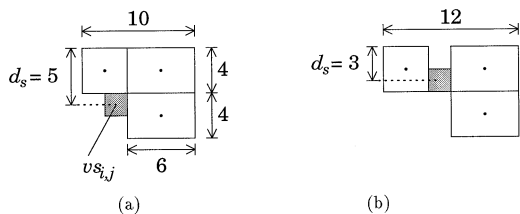


図4 $LR(y_{i,j})$ の配置 ($y_{i,j} = x_k$ の場合)
Fig. 4 Two layouts of $LR(y_{i,j})$, where $y_{i,j} = x_k$.

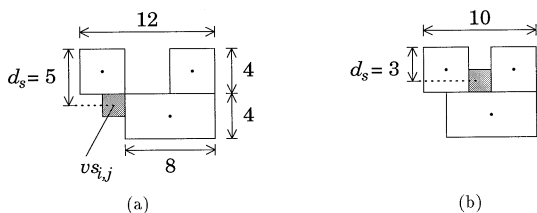


図5 $LR(y_{i,j})$ の配置 ($y_{i,j} = \bar{x}_k$ の場合)
Fig. 5 Two layouts of $LR(y_{i,j})$, where $y_{i,j} = \bar{x}_k$.

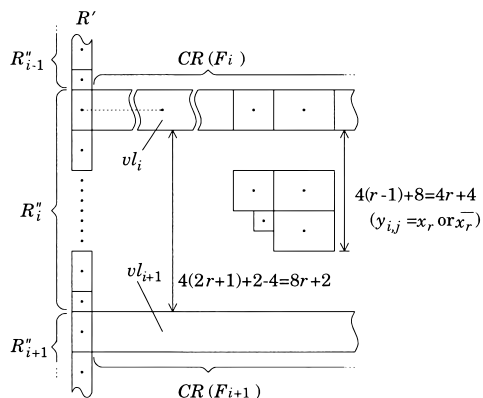


図7 R' と $CR(F_i)$ の配置
Fig. 7 The layout of $CR(F_i)$ and R' .

十分な y 座標の間隔をとっている (図 7 参照). 更に $\pi_{R(E)}$ では, $\pi_{VR(x_1)}$ の左側に, 上端の境界が同じになるように $\langle 32mr + 8r, 2 \rangle$ の方形 R^* を配置する.

各 $VR(x_k)$ 及び $CR(F_i)$ は, それぞれ r, m に關する多項式の大きさ及び個数の方形を含み, $R(E)$ はそれらをそれぞれ, r 個, m 個含んでいるので, $R(E)$ とその初期配置 $\pi_{R(E)}$ は多項式時間で作成できる.

E の各変数に true 又は false を割り当てることによって得られる $R(E)$ の再配置を $\pi'_{R(E)}$ とする. $\pi'_{R(E)}$ は直交順序を保存し, 方形の非交差を満たすように作成する.

4.2 証 明

[補題 1] E が充足可能ならばそのときに限り, $R(E)$ の配置 $\pi'_{R(E)}$ が存在し, $\pi'_{R(E)}$ は制約 (1), (2) を満たし, $W_x(\pi'_{R(E)}) \leq 32mr + 108m + 16r - 60$, $W_y(\pi'_{R(E)}) = 8mr + 6m + 4r$ である.

(証明) (\Rightarrow) まず, E が充足可能であると仮定する. このとき, E を充足可能にするような各変数への真偽割当てが存在する. 各変数 x_k の割当てに対応する $VR(x_k)$ の配置を決める. $x_k = \text{true}$ ならば図 3(a), $x_k = \text{false}$ ならば同図 (b) とし, $R(E)$ の配置 $\pi'_{R(E)}$ を考える. 図 3(a) の初期配置に対して, 同図 (b) の配置が直交順序を満たしていることに注意する. このとき, $W_x(\pi'_{VR(x_k)}) = 8$, $W_y(\pi'_{VR(x_k)}) = 8mr + 6m + 4r$ である.

各 $VR(x_k)$ の配置より, R' は各方形がすき間なく縦列に配置され, これにより各 $LR(y_{i,j})$ ($1 \leq i \leq m, j = 1, 2, 3$) に属する各方形の y 座標が決定する. $W_y(\pi'_{LR(y_{i,j})}) = 8$ であり, $d_s = 5$ 又は 3 で幅が最小となる配置は, 図 4 及び図 5 で示す配置のみである. リテラル $y_{i,j}$ に対応する変数が true であれば $d_s = 5$, false であれば $d_s = 3$ となる. このとき, $y_{i,j}$ の値が true であれば図 4(a) 又は図 5(b) のいずれかの配置となり, $W_x(\pi'_{LR(y_{i,j})}) = 10$ となる. また, $y_{i,j}$ の値が false のときは $W_x(\pi'_{LR(y_{i,j})}) = 12$ となる.

仮定より, 各 F_i の三つのリテラルのうち少なくとも一つは true であるので, $W_x(\pi'_{LR(y_{i,j})}) \leq 34$ となり, $W_x(\pi'_{CR(F_i)}) \leq 108m - 62$ となる. よって $R(E)$ 全体は, $W_x(\pi'_{R(E)}) \leq 32mr + 16r + 2 + (108m - 62) = 32mr + 108m + 16r - 60$, $W_y(\pi'_{R(E)}) = W_y(\pi'_{VR(x_k)}) = 8mr + 6m + 4r$ となる. 図 8 に $E = (x_1 \vee x_2 \vee x_3) \wedge (\overline{x_2} \vee \overline{x_3} \vee \overline{x_4}) \wedge (\overline{x_1} \vee \overline{x_4} \vee x_2)$ の場合の $R(E)$ の配置例を示す.

(\Leftarrow) 次に, $W_x(\pi'_{R(E)}) \leq 32mr + 108m + 16r - 60$,

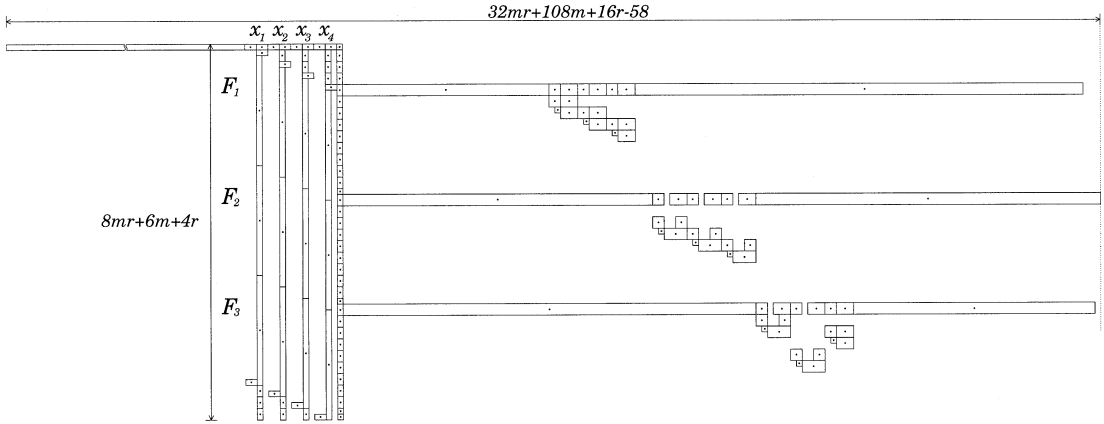
$W_y(\pi'_{R(E)}) = 8mr + 6m + 4r$ の $R(E)$ の配置 $\pi'_{R(E)}$ が存在すると仮定する.

F_i に含まれるリテラルが x_k とその否定 $\overline{x_k}$ の両方を含んでいるならば, F_i は任意の割当てで true である. F_i に含まれるリテラルがすべて異なる変数の場合を考える. 各 $y_{i,j}$ ($j = 1, 2, 3$) は x_{k_j} 又は $\overline{x_{k_j}}$ とする. $VR(x_{k_j})$ と $LR'(y_{i,j})$ の $\pi'_{R(E)}$ における幅の和は, $W_x(\pi'_{R(E)}) - \{\sum_{k \neq k_1, k_2, k_3} W_x(\pi_{VR(x_k)}) + W_x(\pi_{R'}) + (vl_{i,j} \text{の幅}) + (vr_{i,j} \text{の幅})\}$ であるから, i にかかわらず $(32mr + 108m + 16r - 60) - \{32mr + 8r + 8(r-3) + 2 + 36(m+i-2) + 36(2m-i-1) + 12\} = 58$ 以下である. よって, 少なくとも一つの j に対して, $VR(x_{k_j})$ と $LR'(y_{i,j})$ の W_x の和が 19 以下である. それぞれ $W_x(\pi'_{VR(x_{k_j})}) \geq 8$, $W_x(\pi'_{LR'(y_{i,j})}) \geq 10$ なので, $W_x(\pi'_{VR(x_{k_j})}) \leq 9$, $W_x(\pi'_{LR'(y_{i,j})}) \leq 11$ である.

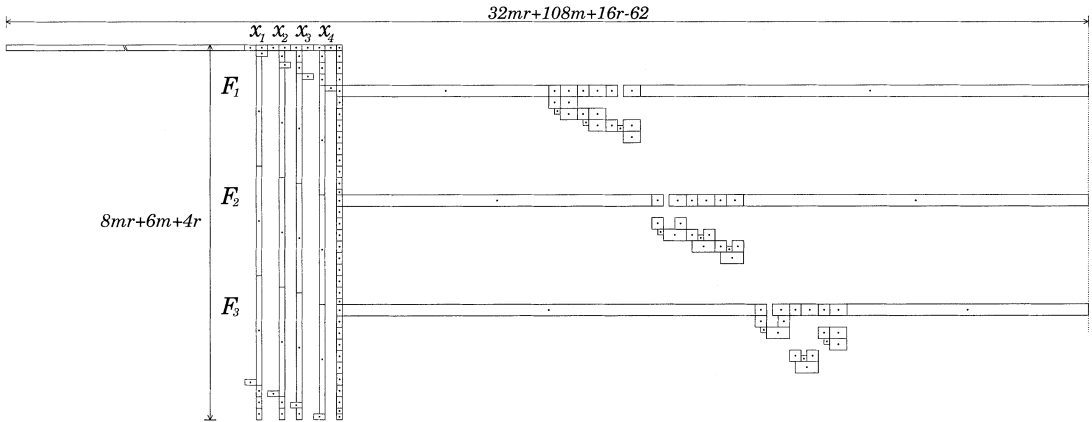
初期配置 π では, $W_y(\pi_{VR(x_{k_j})}) = 8mr + 6m + 4r$ である (図 3). 再配置 π' では $W_y(\pi'_{R(E)}) = W_y(\pi'_{VR(x_{k_j})})$ かつ $W_x(\pi'_{VR(x_{k_j})}) \leq 9$ であり, 図 3 より, $vc_{k_j,1}, \dots, vc_{k_j,m}$ は必ず y 方向に間隔が開くことなく配置され, それぞれの y 座標は図 3(a) 又は (b) のいずれかである. また R' に関して, $W_y(\pi_{R'}) = 8mr + 6m + 4r = W_y(\pi'_{R(E)})$ であることから, 再配置 π'_R では R' は y 方向に間隔が開くことなく配置され, このことにより各 $LR(y_{i,j})$ ($1 \leq i \leq m, j = 1, 2, 3$) に属する $vs_{i,j}$ 以外の方形の y 座標は初期配置と変わらず, また $W_y(\pi'_{LR(y_{i,j})}) = 8$ となる.

$y_{i,j}$ が x_k 又は $\overline{x_k}$ であるとする. このとき, $vs_{i,j}$ と $vc_{k,i}$ の y 座標が等しいことより, $VR(x_k)$ に属する $vc_{k,1}, \dots, vc_{k,m}$ の y 座標が図 3(a) ならば, $LR(y_{i,j})$ に対して $d_s = 5$, 同図 (b) ならば $d_s = 3$ となる (図 4, 図 5 参照). $d_s = 5$ のとき $y_{i,j} = \overline{x_k}$ ならば, $W_x(\pi'_{LR(y_{i,j})}) \geq 12$ となり, このとき $W_x(\pi'_{LR'(y_{i,j})}) \geq 12$ となる. また, $d_s = 3$ のとき $y_{i,j} = x_k$ ならば, 同様に $W_x(\pi'_{LR'(y_{i,j})}) \geq 12$ となる. したがって, $W_x(\pi_{LR'(y_{i,j})}) \leq 11$ であれば, $d_s = 5$ のとき $y_{i,j} = x_k$, $d_s = 3$ のとき $y_{i,j} = \overline{x_k}$ である.

ここで, $vc_{k,1}, \dots, vc_{k,m}$ の y 座標が図 3(a) 又は (b) のいずれかである x_k に対して, 図 3(a) であれば $x_k = \text{true}$, 同図 (b) であれば $x_k = \text{false}$ の割当てを考える. このとき, 各 F_i に対してある j が



(a) 初期配置 ($x_1 = x_2 = x_3 = x_4 = \text{true}$)



(b) 再配置 ($x_1 = x_2 = \text{true}, x_3 = x_4 = \text{false}$)

図8 $R(E)$ の配置例

Fig. 8 An example of the layout of $R(E)$.

存在し, $W_x(\pi'_{VR}(x_{k_j})) \leq 9$, $W_x(\pi'_{VR}(x_{k_j})) \leq 11$, $y_{i,j} = x_{k_j}$ 又は $\overline{x_{k_j}}$ である. $W_x(\pi'_{VR}(x_{k_j})) \leq 9$ より, $vc_{k_j,1}, \dots, vc_{k_j,m}$ の y 座標は, 図 3 (a) 又は (b) である. 図 3 (a) ならば, $x_{k_j} = \text{true}$ 及び $d_s = 5$ となり, $W_x(\pi'_{LR}(y_{i,j})) \leq 11$ より, $y_{i,j} = x_{k_j}$, $y_{i,j} = \text{true}$ である. また, 同図 (b) ならば同様に, $x_{k_j} = \text{false}$, $d_s = 3$ であり, $y_{i,j} = \overline{x_{k_j}}$, $y_{i,j} = \text{true}$ である.

よって, 各 F_i の少なくとも一つのリテラルは true となり, E は true となるので充足可能である. □

[補題 2] E が充足可能ならばそのときに限り, 配置 $\pi'_{R(E)}$ が存在し, $\pi'_{R(E)}$ は制約 (1), (2) を満たし, $S(\pi'_{R(E)}) \leq (32mr+108m+16r-60)(8mr+6m+4r)$ である.

(証明) (\Rightarrow) $S = (32mr + 108m + 16r - 60)(8mr + 6m + 4r)$ とおく. E が充足可能ならば, 補題 1 より, 配置 $\pi'_{R(E)}$ が $S(\pi'_{R(E)}) \leq S$ ができる.

(\Leftarrow) $S(\pi'_{R(E)}) \leq S$ ならば $R(E)$ の定義より $W_x(\pi'_{R(E)}) \geq 32mr+108m+16r-64$, $W_y(\pi'_{R(E)}) \geq 8mr+6m+4r$ である. $W_y(\pi'_{R(E)}) > 8mr+6m+4r$ ならば,

$$\begin{aligned} S(\pi'_{R(E)}) &\geq (32mr+108m+16r-64)(8mr+6m+4r+1) \\ &= (32mr+108m+16r-60)(8mr+6m+4r) \\ &\quad -4(8mr+6m+4r)+32mr+108m+16r-64 \\ &= S+84m-64. \end{aligned}$$

$m \geq 1$ なので $84m - 64 > 0$ となり, $S(\pi'_{R(E)}) > S$ となる.

よって, $S(\pi'_{R(E)}) \leq S$ ならば, $W_y(\pi'_{R(E)}) = 8mr + 6m + 4r$ であり, $W_x(\pi'_{R(E)}) \leq 32mr + 108m + 16r - 60$ であるので, 補題 1 より E は充足可能である. \square

ILADR が NP に属することは明らかなので, 補題 2 より次の定理が導かれる.

[定理 1] ILADR は NP 完全である.

5. 再配置アルゴリズム

文献 [5] では, 方形集合とその配置が与えられたときに, 直交順序を保存した面積最小の非交差再配置を求める発見的手法として, アルゴリズム PFS (Push Force-Scan) が提案されている. 本章では, PFS アルゴリズムをもとにした, 面積最小の再配置を求める発見的アルゴリズム PFS' を示す.

5.1 Push Force-Scan アルゴリズム

アルゴリズム PFS は方形間の交差を除くために, 方形間力と呼ばれる値を用いる. 方形間力は任意の二つの方形間で定義されるベクトルである. 方形 v_i, v_j 間の方形間力 $f_{i,j}$ は, これらの方形が交差するとき, v_i に対して v_j を相対的に $f_{i,j}$ だけ移動すれば, v_i, v_j が交差しないことを保証する. $f_{i,j}$ の向きは, v_i, v_j 間の交差を除くだけでなく, 過去の実験などで得られた経験に従って, 直交順序を保存してできるだけ配置面積が小さくなるように選ばれる.

ここではまず, 方形間力及び本節で用いる記号について定義する. 方形集合 R 及びその配置 π_R は与えられているものとする.

配置 π_R における方形 $v_i (\in R)$ の中心座標を (x_i, y_i) と表す. すなわち, $\pi_R(v_i) = (x_i, y_i)$ とする. 方形 $v_i, v_j (\in R)$ 間の座標の差 $\Delta x_{i,j}, \Delta y_{i,j}$ を次のように定義する.

$$\Delta x_{i,j} = x_j - x_i$$

$$\Delta y_{i,j} = y_j - y_i$$

方形 v_i, v_j が交差しているならばそのときに限り,

$$|\Delta x_{i,j}| < \frac{w_i + w_j}{2} \quad \text{かつ} \quad |\Delta y_{i,j}| < \frac{h_i + h_j}{2}$$

が成り立つ. L を v_i 及び v_j の中心座標を結ぶ直線とする. 交差を解消するために, 直交座標を満たす範囲で, v_i, v_j が交差せずに接触するまで, L に沿っ

て v_j を移動させるとする. このとき, 移動後の v_j の座標を (x'_j, y'_j) とする. 方形間力 $f_{i,j} = (f_{i,j}^x, f_{i,j}^y)$ は, (x_j, y_j) から (x'_j, y'_j) へのベクトルとして, 以下のように定義する. $g_{i,j}$ を L のこう配, すなわち, $g_{i,j} = \Delta y_{i,j} / \Delta x_{i,j}$ とする. $\Delta x_{i,j} = 0$ ならば $g_{i,j} = \infty$ とする. また, $G_{i,j}$ を $(h_i + h_j) / (w_i + w_j)$ とする.

(a) v_i, v_j が y 方向の境界で接触する場合, すなわち, $G_{i,j} \geq g_{i,j} > 0, -G_{i,j} \leq g_{i,j} < 0, g_{i,j} = 0$ のいずれかの場合,

$$f_{i,j}^x = \frac{\Delta x_{i,j}}{|\Delta x_{i,j}|} \left(\frac{w_i + w_j}{2} - |\Delta x_{i,j}| \right), \quad f_{i,j}^y = f_{i,j}^x \cdot g_{i,j}$$

(b) v_i, v_j が x 方向の境界で接触する場合, すなわち, $(G_{i,j} < g_{i,j})$ かつ $(g_{i,j} > 0)$, 又は $(-G_{i,j} > g_{i,j})$ かつ $(g_{i,j} < 0)$ の場合,

$$f_{i,j}^y = \frac{\Delta y_{i,j}}{|\Delta y_{i,j}|} \left(\frac{h_i + h_j}{2} - |\Delta y_{i,j}| \right), \quad f_{i,j}^x = f_{i,j}^y / g_{i,j}$$

次に, PFS の詳細について述べる. PFS は制約 (1), (2) を満たす再配置を $O(n^2)$ 時間で求める ($n = |R|$). PFS では, まず x 方向の移動である Horizontal-PFS, 次に y 方向の移動である Vertical-PFS を実行する. Horizontal-PFS では, 任意の方形対 v_i, v_j に対し, v_i から見て v_j を少なくとも $f_{i,j}$ 移動させることを保証する. Vertical-PFS は適用する方向を除いて Horizontal-PFS と同じであるため, ここでは Horizontal-PFS についてのみ説明する.

Horizontal-PFS の概要を図 9 に示す. ここで, $x_1 \leq x_2 \leq \dots \leq x_n$ と仮定する. Horizontal-PFS では v_1, v_2, \dots, v_n の順で各方形の再配置における x 座標を決める. 同じ x 座標をもつ方形の x 座標は同時に決まる. 同じ x 座標をもつ方形 v_i, \dots, v_k の x 座標を決めるとき, これらの方形だけを動かすのではなく, すべての $v_j (i \leq j \leq n)$ を同時に移動させる. この移動距離は, while ループの 1 回前に移動した方形 $v_m (l \leq m < i)$ と $v_j (i \leq j \leq n)$ 間の方形間力 $f_{m,j}^x$ の最大値とする.

PFS は正方向のみの移動に制限しているが, この制限を解消するために, 三末らは別の手法である Push-Pull Force-Scan アルゴリズムを提案している. このアルゴリズムは負方向の移動を許しているが, 非交差を保証していない. よって, ここではこのアルゴリズムについては議論しない.

[Algorithm Horizontal-PFS]

```

begin
  i := 1;
  l := 1;
  while (i ≤ n) do begin
    k := max{j | x_i = x_j}; /* x_i = x_{i+1} = ... = x_k */
    δ := 0;
    if (x_i > x_1) then
      δ := max(0, max_{l ≤ m < i ≤ j ≤ n} f_{m,j}^x);
    for j := i to n do
      x_j := x_j + δ;
    l := i;
    i := k + 1;
  end;
end.

```

図9 アルゴリズム Horizontal-PFS
Fig. 9 Algorithm Horizontal-PFS.

5.2 Push Force-Scan アルゴリズムの改良

PFS ではいくつかの場合において、方形間に余分なすき間を生じる。図 10 の例において、同図 (a) は初期配置、(b) は PFS による再配置を表している。この例では、 v_2 と v_3 は $f_{1,3}^x$ によって右に移動し、その後、 v_3 が $f_{2,3}^x$ によって再び移動するため、 v_1 と v_2 の間に余分なすき間を生じる。

ここでは、PFS で得られた配置面積以下の配置を得るアルゴリズム PFS' を提案する。PFS' の計算量は PFS と同じく $O(n^2)$ である。PFS' は PFS と同様に Horizontal-PFS'、Vertical-PFS' の順で実行する。ここでは Horizontal-PFS' についてのみ説明する。

再び、図 10 の例について考える。この場合、 v_2 に対しては $f_{1,2}^x$ による移動、 v_3 に対しては $\max\{f_{1,2}^x + f_{2,3}^x, f_{1,3}^x\}$ による移動を考えれば十分である。PFS' はこの考え方を一般化したものである。 $x_1 \leq x_2 \leq \dots \leq x_n$ と仮定する。Horizontal-PFS' の概要を図 11 に示す。関数 $Lbnd(v_i, x_i)$ は方形 v_i が座標 x_i をもつときの左端の x 座標である。Horizontal-PFS' では v_1, v_2, \dots, v_n の順で再配置における方形の x 座標を決める。同じ x 座標をもつ方形の x 座標は同時に決まる。同じ x 座標をもつ方形 v_i, \dots, v_k の x 座標を決めるとき、移動距離はある特別な場合を除いて v_1, v_2, \dots, v_k にのみ依存する。これが PFS とは異なる部分である。ここでは v_i, \dots, v_k の x 座標を決める方法について述べる。 v_1, v_2, \dots, v_{i-1} の x 座標が既に決まっている

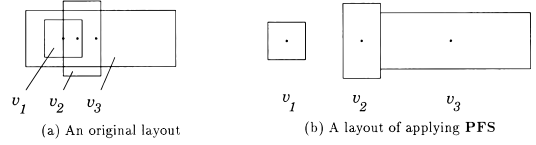


図 10 すき間が生じる例
Fig. 10 An example of a needless gap.

[Algorithm Horizontal-PFS']

```

begin
  i := 1;
  σ := 0;
  lmin := 1;
  while (i ≤ n) do begin
    k := max{j | x_i = x_j}; /* x_i = x_{i+1} = ... = x_k */
    γ := 0;
    if (x_i > x_1) then
      γ := -∞;
    for m := i to k do begin
      γ'' := max_{1 ≤ j ≤ i-1} (γ_j + f_{j,m}^x);
      γ' := {
        σ if Lbnd(v_m, x_m) + γ''
            < Lbnd(v_lmin, x_lmin)
        γ'' otherwise
      };
      γ := max(γ, γ');
    end;
    for m := i to k do begin
      γ_m := γ;
      x_m := x_m + γ_m;
      if Lbnd(v_m, x_m) < Lbnd(v_lmin, x_lmin) then
        lmin := m;
    end;
    σ := σ + max(0, max_{i ≤ m ≤ k < j ≤ n} f_{m,j}^x);
    i := k + 1;
  end;
end.

```

図 11 アルゴリズム Horizontal-PFS'
Fig. 11 Algorithm Horizontal-PFS'.

と仮定する。 γ_j を PFS' によって v_j が x 方向に移動する距離とする。後で述べる特別な場合を除いて、Horizontal-PFS' は $\gamma_m (i \leq m \leq k)$ を $\gamma_j + f_{j,m}^x$ の最大値として決定する ($1 \leq j < i \leq m \leq k$)。

特別な場合について述べる。 σ_m を、PFS において $v_m (i \leq m \leq k)$ が右に移動した距離とする。PFS' では、方形 v_m が γ_m だけ移動することによって、 v_m

の左境界が x 座標の決まっている v_m 以外の方形より左になるように置かれることがある。このとき、配置面積は PFS' よりも大きくなる可能性があるため、このような場合に限って、移動距離として γ_m の代わりに PFS で用いられる σ_m を用いることにする。

5.3 アルゴリズムの有効性

ここでは、PFS' が制約 (1), (2) を満たし、PFS' による配置面積が PFS による配置面積以下であることを示す。

π'_R と π''_R をそれぞれ PFS, PFS' を実行した後の R の配置とする。 x_i, x'_i, x''_i をそれぞれ初期配置, π'_R, π''_R における v_i の x 座標とおく ($1 \leq i \leq n$)。PFS における v_i の x 移動距離 σ_i は、

$$\sigma_i = \delta_0 + \delta_1 + \dots + \delta_{i-1}$$

$$\delta_i = \begin{cases} 0 & \text{if } i = 0 \text{ or } x_i = x_{i+1} \\ \max(0, \max_{1 \leq m \leq i < j \leq n} f_{m,j}^x) & \text{if } x_i < x_{i+1} \end{cases}$$

l は $x_i = x_m$ を満たす最小の m である。一方、PFS' における v_i の移動距離 γ_i は以下のように計算する。

$$\gamma_i'' = \max_{1 \leq j < l} (\gamma_j + f_{j,i}^x)$$

$$\gamma_i' = \begin{cases} \sigma_i \text{ if } \text{Lbnd}(v_i, x_i) + \gamma_i'' < \min_{j < l} \text{Lbnd}(v_j, x_j + \gamma_j) \\ \gamma_i'' \text{ otherwise} \end{cases}$$

$$\gamma_i = \max_{x_i = x_m} \gamma_m'$$

l は $x_i = x_m$ を満たす最小の m である。

[補題 3] すべての i ($1 \leq i \leq n$) について、(a) $\sigma_i \geq \gamma_i''$ 及び (b) $x'_i \geq x''_i$ が成り立つ。

(証明) i に関する帰納法で証明する。まず、 $x_1 = x_i$ である i について、 $\gamma_i = \sigma_i'' = \sigma_i = 0$ が成り立つ。ゆえに、 $x'_i = x_i + \sigma_i = x_i + \gamma_i = x''_i$ が成り立つ。 x_1, \dots, x_k を同じ x 座標をもつ最大の列とする。 $j < l$ である j について、 $x'_j \geq x''_j$ 、すなわち、 $\sigma_j \geq \gamma_j$ と仮定する。 $l \leq i \leq k$ であるすべての i について、 x_i の x 座標はすべて等しいので、 $\gamma_i = \gamma_l$ 及び $\sigma_i = \sigma_l$ が成り立つ。よって、 $\sigma_i \geq \gamma_i$ を示せば十分である。

$l \leq i \leq k$ である i について、

$$\gamma_i'' = \max_{1 \leq j < l} (\gamma_j + f_{j,i}^x) \leq \max_{1 \leq j < l} (\sigma_j + f_{j,i}^x)$$

が成り立つ。 l_j 及び k_j をそれぞれ、 $x_{l_j} = x_j$ 及び $x_{k_j} = x_j$ を満たす最小及び最大の添字とおく。

$$f_{j,m}^x \leq \max_{l_j \leq j' \leq k_j} f_{j',m}^x$$

$$\leq \max_{l_j \leq j' \leq k_j < m' \leq n} f_{j',m'}^x \quad (\because k_j < m)$$

$$\leq \max(0, \max_{l_j \leq j' \leq k_j < m' \leq n} f_{j',m'}^x) = \delta_{k_j}$$

$\sigma_j \leq \sigma_{k_j}$ より、 $\sigma_i \geq \gamma_i''$ ($l \leq i \leq k$) であることを示す。

$$\gamma_i'' \leq \max_{1 \leq j < l} (\sigma_j + f_{j,i}^x)$$

$$\leq \max_{1 \leq j < l} (\sigma_j + \delta_{k_j})$$

$$\leq \max_{1 \leq j < l} (\sigma_{k_j} + \delta_{k_j})$$

$$\leq \sigma_{k_{j+1}} \leq \sigma_l = \sigma_i$$

$\sigma_i \geq \gamma_i''$ より、 $\sigma_i \geq \gamma_i'$ が成り立つ。 $\sigma_l = \dots = \sigma_k$ より、 $\gamma_i = \max_{l \leq m \leq k} \gamma_m' \leq \max_{l \leq m \leq k} \sigma_m = \sigma_i$ である。よって、 $x'_i \geq x''_i$ が成り立つ。□

[補題 4] π'_R 及び π''_R は次の式を満たす。

$$W_x(\pi'_R) \geq W_x(\pi''_R) \quad \text{かつ} \quad W_y(\pi'_R) \geq W_y(\pi''_R)$$

(証明) $W_x(\pi'_R) \geq W_x(\pi''_R)$ についてのみ証明する。 π'_R の左境界をなす方形を $v_{l'}$ 、右境界をなす方形を $v_{r'}$ とおく。同様に、 π''_R に対しても、 $v_{l''}, v_{r''}$ を定義する。このとき、

$$\text{left}_{\pi'}(v_{l'}) \leq \text{left}_{\pi''}(v_{l''}) \quad \text{かつ}$$

$$\text{right}_{\pi'}(v_{r'}) \geq \text{right}_{\pi''}(v_{r''})$$

を示せば十分である。

$x_{l''} = x_1$ ならば、 $\gamma_{l''} = \sigma_{l''} = 0$ である。このとき、 $\text{left}_{\pi'}(v_{l'}) \leq \text{left}_{\pi''}(v_{l''}) = \text{left}_{\pi''}(v_{l''})$ が成り立つ。 $x_{l''} \neq x_1$ の場合を考える。方形 $v_{l''}$ は同じ x 座標をもつ方形 $v_{l_1''}, \dots, v_{k_{l''}''}$ の中で最も幅の大きいものである。 l_{\min}'' を PFS' が σ_i ($i = 1, \dots, l_{l''} - 1$) を決めた後の変数 l_{\min} の値とおく。 $\text{left}_{\pi''}(v_{l''}) \leq \text{left}_{\pi''}(v_{l_{\min}''})$ より、PFS' は $\text{Lbnd}(v_{l''}, x_{l''}) + \gamma_{l''} < \text{Lbnd}(v_{l_{\min}'}, x_{l_{\min}'}) + \gamma_{l_{\min}'}'$ であることを検出し、 $\gamma_{l''} = \sigma_{l''}$ とする。これにより、 $\text{left}_{\pi'}(v_{l'}) \leq \text{left}_{\pi''}(v_{l''} = \text{left}_{\pi''}(v_{l''}))$ である。

補題 3 より、 $\text{right}_{\pi'}(v_{r'}) \geq \text{right}_{\pi''}(v_{r''})$ が成り立つ。よって、 $\text{right}_{\pi'}(v_{r'}) \geq \text{right}_{\pi''}(v_{r''}) \geq \text{right}_{\pi''}(v_{r''})$ である。□

次に、アルゴリズム PFS' による再配置が二つの制約を満たすことを示す。

[補題 5] 任意の二つの方形 $v_i, v_j \in R$ ($i \leq j$) に対して、 $\gamma_j - \gamma_i \geq f_{i,j}^x$ が成り立つ。

(証明) $x_i = x_j$ ならば, $\gamma_i = \gamma_j$ かつ $f_{i,j}^x = 0$ であるので, 与式は成り立つ. $x_i < x_j$ の場合を考える. l 及び k をそれぞれ, $x_l = x_j$ 及び $x_k = x_j$ を満たす最小及び最大の添字とおく. $l \leq m \leq k$ であるすべての m に対して,

$$\gamma_m'' = \max_{1 \leq i' < l} (\gamma_{i'} + f_{i',m}^x) \geq \gamma_i + f_{i,j}^x$$

が成り立つ. 補題 3 より, $\gamma_m'' \leq \sigma_m$ が成り立ち, 更に $\gamma_j = \max_{l \leq m \leq k} \gamma_m'$ かつ $\gamma_m' = (\sigma_m \text{ 又は } \gamma_m'')$ である. よって, $\gamma_m'' \leq \gamma_j$ であるから, $\gamma_j \geq \gamma_i + f_{i,j}^x$ が成り立つ. \square

[補題 6] アルゴリズム PFS' は直交順序を保存する (制約 (1)).

(証明) $x_i = x_j$ ならば, $x_i'' = x_j''$ が成り立つ. ここでは $x_i \neq x_j$ の場合を考えるが, 一般性を失うことなく $x_i < x_j$ と仮定してもよい. 補題 5 より, $\gamma_j - \gamma_i \geq f_{i,j}^x$ が成り立つ. $f_{i,j}^x$ の定義より, $x_i \leq x_j + f_{i,j}^x$. よって,

$$x_i'' = x_i + \gamma_i \leq x_i + \gamma_j - f_{i,j}^x \leq x_j + \gamma_j = x_j''$$

が成り立つ. \square

$i < j$ である二つの方形 v_i, v_j に対して, 補題 5 及び $f_{i,j}^x$ の定義より, PFS' による配置は制約 (2) を満たすことが言え, 次の補題が成り立つ.

[補題 7] アルゴリズム PFS' は方形の非交差を保証する.

以上より, 次の定理が成り立つ.

[定理 2] PFS' は時間計算量 $O(n^2)$ で再配置を行い, 再配置後の方形の配置は制約 (1), (2) を満たし, 配置面積は PFS による再配置後の配置面積以下である.

5.4 配置例と実験結果

まず, ある方形集合 R とその配置 π_R に対する PFS と PFS' の配置例を示す. 図 12 (a) はもとの配置 π_R , 同図 (b), (c) がそれぞれ PFS, PFS' 実行後の配置 π_R' である.

更に, ランダムに生成した方形集合に対して PFS 及び PFS' を適用する実験を行った. 実験には Sun Ultra-2 を用いた. 図 13 に実験結果を示す. 初期配置は, 200×200 の範囲にランダムに方形を配置したものを用いる. 同図 (a) は, 横軸に方形の個数をとった場合の PFS に対する PFS' の面積比である. 方形サイズは 1 辺が $50 \sim 100$ の範囲でランダムに選んだものである. これらの値は, 50 の方形集合を独立に作

成して実験し, 結果の平均値をとったものである. 実験に要した計算時間は, 50 回の試行に対していずれも 1 秒以下である. この結果より PFS' は, 特に方形の個数が多い場合に対して, PFS による配置の面積の 15 ~ 20% の面積で再配置できることがわかる. 同図

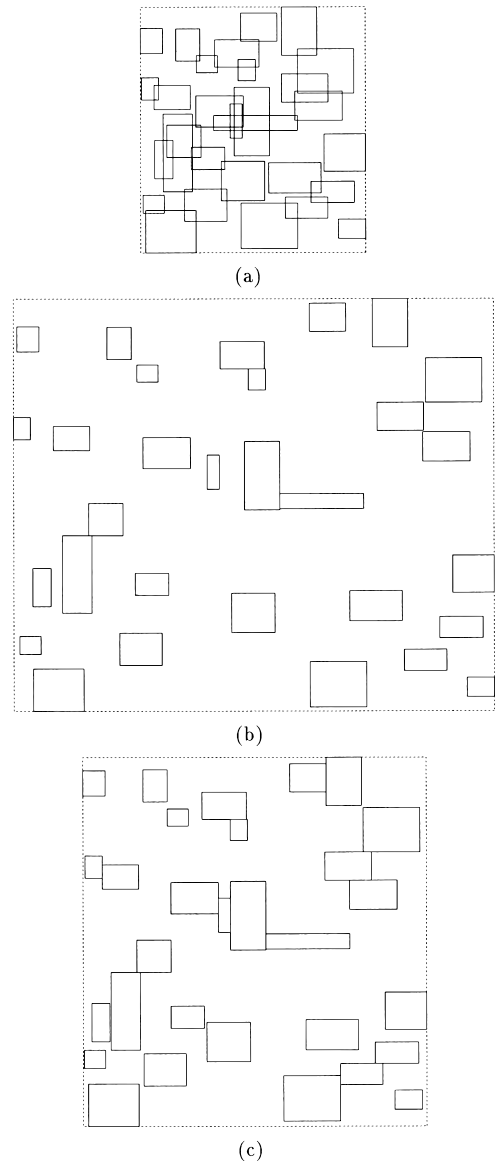


図 12 アルゴリズムの実行情例 ($n = 30$). (a) もとの配置 π_R (b) PFS の結果 (c) PFS' の結果

Fig. 12 An example of applying the algorithms, (a) an original layout π_R , (b) a result of applying PFS, and (c) a result of applying PFS'.

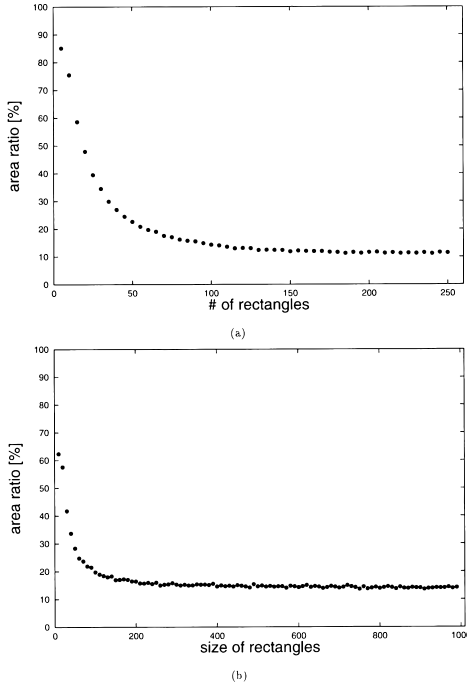


図 13 実験結果 . (a) 方形数を变化させた場合, (b) 方形サイズを变化させた場合
 Fig. 13 Experimental results, (a) the case of changing the number of rectangles, (b) the case of changing the size of rectangles.

(b) は, 横軸に方形サイズをとった場合の PFS に対する PFS' の面積比である . 方形の個数を 50, 方形サイズは 1 辺が (値) ± 10 の範囲でランダムに選んだものである . このことから, 方形の個数が同じ場合には, 方形のサイズの変化は余り影響しないことがいえる .

6. む す び

本論文では, 直交順序を保存する方形集合に対して, 方形が交差しない最小面積再配置問題を定義し, この問題が整数座標系では NP 完全であることを示した . 更に, 方形の非交差を満たす面積最小化のための発見的再配置手法の一つとして, 文献 [5] のアルゴリズムと同じ漸近的計算量で, 面積が小さいか等しい再配置を求める方法を示し, 実験によって多くの場合に本手法による再配置の面積が小さくなることを確認した . 今後の課題としては, 他の制約を考慮した場合の NP 完全性について考察すること, より効率的な発見的アルゴリズムを作ることが考えられる .

謝辞 本研究に対し, 御助言, 御討論いただいた,

広島市立大学情報科学部井上智生助教授ならびに奈良先端科学技術大学院大学情報論理学講座諸氏に深謝致します .

文 献

- [1] G. Di Battista, P. Eades, R. Tamassia, and I.G. Tollis, "Algorithms for drawing graphs: An annotated bibliography," Computational Geometry Theory and Applications, vol.4, pp.235-282, 1994.
- [2] R. Tamassia, G. Di Battista, and C. Batini, "Automatic graph drawing and readability of diagrams," IEEE Trans. on System, Man and Cybernetics, vol.18, no.1, pp.61-79, 1988.
- [3] 杉山公造, "グラフ自動描画法とその応用 — ビジュアルヒューマンインタフェースのための基礎技法," 計測自動制御学会出版, 1993.
- [4] P. Eades, W. Lai, K. Misue, and K. Sugiyama, "Preserving the mental map of a diagram," Proc. of COMPUGRAPHICS'91, pp.34-43, 1991.
- [5] K. Misue, P. Eades, W. Lai, and K. Sugiyama, "Layout adjustment and the mental map," Journal of Visual Languages and Computing, vol.6, pp.183-210, 1995.
- [6] M-A.D. Storey and H.A. Müller, "Graph layout adjustment strategies," Proc. Graph Drawing'95, LNCS 1027, pp.487-499. Springer-Verlag, 1995.
- [7] M.R. Garey and D.S. Johnson, "Computers and Intractability—A Guide to the Theory of NP-Completeness," W.H. Freeman and Company, New York, 1979.

(平成 10 年 11 月 2 日受付)



林 邦彦 (学生員)

平 6 神戸大・工・電子卒 . 平 8 同大大学院博士前期課程了 . 平 11 奈良先端大博士後期課程了 . 同年松下電器産業 (株) に入社, 現在に至る . 在学中, グラフ描画アルゴリズム, グラフ理論, 計算複雑度に関する研究に従事 .



井上美智子 (正員)

昭 62 阪大・基礎工・情報卒 . 平 1 同大大学院博士前期課程了 . 同年富士通研究所 (株) 入社 . 平 7 阪大大学院博士後期課程了 . 現在奈良先端大情報科学助手 . 分散アルゴリズム, グラフ理論, テスト容易化設計, 高位合成の研究に従事 . 工博 . IEEE, 情報処理学会, 人工知能学会各会員 .



増澤 利光 (正員)

昭 57 阪大・基礎工・情報卒。昭 62 同大大学院博士後期課程了。同年同大情報処理教育センター助手。同大基礎工助教授を経て、平 6 奈良先端大情報科学助教授、現在に至る。平 5 コーネル大客員準教授(文部省在外研究員)。分散アルゴリズム、並列アルゴリズム、テスト容易化設計、テスト容易化高位合成に関する研究に従事。工博。ACM, IEEE, EATCS, 情報処理学会各会員。



藤原 秀雄 (正員)

昭 44 阪大・工・電子卒。昭 46 同大大学院博士後期課程了。阪大工学部助手、明治大理工学部教授を経て、現在奈良先端大情報科学教授。昭 56 ウォータールー大客員助教授。昭 59 マッギル大客員準教授。論理設計、高信頼設計、設計自動化、テスト容易化設計、テスト生成、並列処理、計算複雑度に関する研究に従事。著書“Logic Testing and Design for Testability”(The MIT Press)など。大川出版賞。工博。IEEE, 情報処理学会各会員。IEEE Fellow. IEEE Golden Core Member.