

PQ-木を用いたグラフの平面化アルゴリズム†

小澤孝夫^{††} 高橋博之^{††}

Lempel, Even, Cederbaum のグラフの平面性判定アルゴリズムに基づき, PQ-木と呼ばれるデータ構造を用いたグラフの平面化アルゴリズムと, それをランダムグラフに適用した結果を示す. このアルゴリズムでは, まず, グラフの節点に *st*-番号と呼ばれる番号をつけ, 番号1の節点から始めて, *st*-番号順に, 節点1個ずつとそれから出る枝を付加し, その都度, 平面描写を試みる. 平面描写できないときは, 最小限の枝を除去して平面描写可能にする. 得られた平面部分グラフに次の節点とそれから出る枝を付加する. この操作を最後の節点まで続け, すべての節点を含む平面部分グラフを得る. 上記の操作中, 平面化の対象となる部分グラフは PQ-木で表現され, その平面化のため必要最小限の枝の除去は, PQ-木の葉から始め, 順次, 根に向かって最小除去葉数を計算することにより達成される. このアルゴリズムの計算複雑度は, $O(n(e+n))$ 以下である. ただし, n はグラフの節点数, e は枝数である. このアルゴリズムをランダムグラフに適用し, 平面化に関する統計的データを求めている. 適用例の範囲では, 平面化のための除去枝数は, *st*-番号を変えてもあまり変わらず, その平均と標準偏差は, 比較的少ない *st*-番号の変化回数で推定可能である. また, グラフによっても, 除去枝数はさほど変わらず, 得られる平面グラフの枝数は, もとのグラフの節点数と枝数から推定できる.

1. ま え が き

グラフが平面グラフでないとき, グラフの適当な枝を開放除去して平面グラフを得ることを, グラフの平面化という. グラフの平面化問題は, 集積回路の配線を一層にしなければならないときなどに生じ^{1), 2)}, 平面化アルゴリズムは, 工学上いろいろの応用がある. ところが, 与えられたグラフの平面部分グラフで枝数最大のものを求めよという最大平面化問題は, NP 完全であり, これを多項式オーダーで解くアルゴリズムは期待しがたい^{1), 2), 13)}. そのため, 与えられたグラフに平面性判定アルゴリズムを適用したとき, 平面性に対する必要条件が満たされなくなれば, それが満たされるように枝を除去するというアルゴリズムが試みられている³⁾⁻⁵⁾. このようなアルゴリズムでは, 除去される枝がアルゴリズムに依存し, 除去枝数もアルゴリズムによって異なりうるものが予想される.

本文では, Lempel らの平面性判定アルゴリズム⁶⁾に基づき, PQ-木と呼ばれるデータ構造⁷⁾を用いた平面化アルゴリズムを報告する. これは概略次のようになる. まず, グラフの節点に *st*-番号と呼ばれる番号付け^{8), 9)}を行う. 次に, この番号1の節点1から始めて, 番号順に1個の節点とそれから出る枝を付加し, その都度, 平面描写を試みる. 平面描写が可能でなければ,

必要最小限の枝を除去して平面描写可能にする. 得られた平面部分グラフに, 次の節点とそれから出る枝を付加する. この操作を最後の節点まで続けることにより, すべての節点を含む平面部分グラフを得る.

上記のアルゴリズムによると, 平面化のための除去枝数は, 1個のグラフについても, *st*-番号を変えれば, 異なりえ, また, グラフによっても異なる. 本文では, このアルゴリズムを, ランダムグラフ^{9), 10)}に適用して得られた結果についても報告する. 平面化のための除去枝数や平面化されたグラフは興味深い性質を示す. 最近, 効率的なアルゴリズムとして, いろいろな確率的アルゴリズムが試みられている¹¹⁾. このようなアルゴリズムの設計の際, 対象の特徴的な性質を求めることが重要であり, また, 理論的な解析が困難なとき, 実験的にアルゴリズムの評価をすることが必要となる¹¹⁾. このような見地からも, 本文で報告する結果は, 一つの資料を提供することになると思われる.

2. 準 備

(1) グラフ: 平面化の対象となるグラフを G , その節点集合を V , 枝集合を E , 節点数を n , 枝数を e , 平均節点次数を \bar{d} と記す. $\bar{d} = 2e/n$ である.

(2) PQ-木: PQ-木に関する用語は文献7)に基づく. 集合 U のある部分集合を S とする. PQ-木は, U の要素の順列の中で, S の要素が連続しているような順列のすべてを求めるために導入されたデータ構造である. PQ-木は, 1個の根から, 一般に複数

† A Graph-Planarivation Algorithm Using PQ-trees by TAKAO OZAWA and HIROYUKI TAKAHASHI (Faculty of Engineering, Kyoto University).

†† 京都大学工学部電気工学第2教室

個の葉へ順次枝分かれする木構造を持つ。葉は U の要素に対応し、葉を左から右へ順に並べたものは、一つの順列を表わし、フロンティアと呼ばれる。また、枝分かれの分岐点は、 P 節点あるいは Q 節点と呼ばれる節点である。枝分かれの際、根に近い節点が親、葉または葉に近い節点が子になる。 P 節点、 Q 節点については

- (i) P 節点の子は、任意の順序に並べかえうる
- (ii) Q 節点の子は、一つの定まった順序か、あるいはその逆の順序に並べうる

(i), (ii) によって、 PQ -木の葉の並べかえの可能性が表現され、 S が与えられると、 S の要素が連続するという制約を満たすように PQ -木を変形し、新しい PQ -木を構成することになる。この制約を満たしうるとき、 PQ -木は簡約可能であるといい、新しい PQ -木を求める操作を簡約という。なお、 PQ -木は、親を上、子を下に描き、親と子は高さ 1 だけ異なるものとする。葉も PQ -木の節点とみなす。

(3) st -番号⁶⁾: グラフ G において隣接する節点対 s と t が与えられたとき、次の条件を満たす番号を st -番号という。節点 x の番号を $f(x)$ と記すと、(i) $f(s)=1, f(t)=n$ 。(ii) $x \neq s, t$ である x に対し、 $f(y) > f(x) > f(z)$ となるような x の隣接節点 y と z が存在する。 st -番号は、 G が 2 連結なら常に存在し、これにより節点の番号が定まると、枝は、小さい番号の節点から出て、大きい番号の節点に入るように方向付けられ、 G は有向グラフとなる。以後、この番号によって節点を示すことにする。

3. 平面化アルゴリズムの概要

グラフの平面化は、その 2 連結成分に対して考えれば十分である。したがって、 G は 2 連結とする。

〈アルゴリズム PLAN〉

Step 1: G に st -番号を施し、節点 1 に対応する葉 1 個だけから成る PQ -木 T_1 を構成する。 $N \leftarrow 1$ 。

Step 2: G の節点 N に対し、1 個の P 節点とその子の葉からなる PQ -木 T_{NN} を構成する。 T_{NN} の葉は、それぞれ G において節点 N から出る枝の先端点に対応する*。 T_N において、節点 N に対応する葉を T_{NN} の根で置きかえる。得られた PQ -木を T_{N+1} とおく。 $N \leftarrow N+1$ 。

Step 3: T_N において、 G の節点 N に対応する葉

* 一般に、 G の 1 個の節点には、複数個の T の葉に対応する。葉は G の枝に対応すると考えてもよい。

のすべてからなる集合を S とする。この S に対し、 T_N が簡約可能なら Step 5 へ。

Step 4: T_N が簡約可能のように葉を除去する。この際、除去葉数が最小となるようにする。

Step 5: T_N を簡約する。 $N=n$ なら終了し、 $N < n$ なら、Step 2 へ。

上記アルゴリズムにおいて、Step 4 以外は、文献 6) の平面性判定法に文献 7) の PQ -木を用いたものにほぼ一致する。

4. 葉の除去アルゴリズム

4.1 PLAN の Step 4

まず、次のような定義をする。

S -葉、 S -木、 S -根、 S -節点: PLAN の Step 3 における集合 S に属する葉を S -葉という。 PQ -木 T_N に対し、フロンティアが S -葉をすべて含むような最小の高さの部分木(文献 7)の pertinent subtree) を、 T_N における S -木という。 S -木の根を S -根、子孫に S -葉を有する S -木の節点を S -節点という。

S -木に含まれる P 節点、 Q 節点の型として、次の 4 つを定義する。

B 型: 子孫の葉がすべて S -葉であるような節点

W 型: 子孫の葉に S -葉を全く含まないような節点

H 型: 子孫の葉からなるフロンティアにおいて、 S -葉が左端から右端にすべて連続しうような節点

A 型: 子孫の葉からなるフロンティアにおいて、 S -葉がすべて中央に連続し、その両側に S -葉でない葉が連続しうような節点

葉は B 型か W 型である。各 S -節点について、それぞれ上の 4 つの型にするために必要最小限除去しなければならない葉数を、 b, w, h, a と記す。ただし、必要に応じて、B 型、W 型は、H 型、A 型の特殊な場合とみて、除去葉数を計算する。また、各節点の子孫の葉の総数を l と記す。このとき、次の命題が成立する。

【命題 1】 S の指定された PQ -木が簡約可能であるための必要十分条件は、 S -根が、B, H, A のいずれかの型をとることである ($S \neq \emptyset$ とする)。

上の命題に基づいて、アルゴリズム PLAN の Step 4 を次の 4 段階に分けて行う。

〈PLAN-Step 4〉

Step 4.1: S -葉から始めて、 T_N を子から親へと探索し、 S -木を求める。このとき、各 S -節点について、その子に含まれる S -節点数 n_i を求める。

Step 4.2: 後述のアルゴリズム SCAN に示す手順で、各 S -節点について b, w, h, a を計算する。

Step 4.3: S -根に対する b, h, a から最小除去葉数で T_N を簡約可能にするための S -根の型が定まる。これに応じて、親から子へ順次 S -節点のとるべき型が決定されるので、 S -節点に型を示すラベル B, W, H, A のいずれかを付け、ラベルに応じて、節点、葉の除去を行う。

PLAN の Step2 で新しく T_N を構成したとき、新しく付加する P 節点に、その親より大きい番号を付けておく。ただし、 T_1 の根の番号は 1 とする。この番号を用いて、 S -葉から順次 S -木を求める。アルゴリズムは特殊なものでないから省略する。

4.2 葉の最小除去数の計算

(1) l の計算: Step2 において T_{NN} を構成したとき、 P 節点の持つ葉数は、節点 N から出る枝数である。これを d_N とする。 T_{NN} を T_N に付加したとき、この P 節点の l を d_N とし、この P 節点の祖先である各節点の l を、 $l \leftarrow l + d_N$ と変化させる。

(2) b, w, h, a の計算: 葉から始めて、その親へと順次走査し、 b, w, h, a を計算する。このとき、節点のキューとスタックを設ける。キューは上記の走査のため、スタックは Step 4.3 において、Step 4.2 と逆の順で S -節点のラベル決定をするためのものである。

〈アルゴリズム SCAN〉

1° キュー、スタックに S -葉を入れる。各 S -葉について、 $w=1$ とおく。

2° キューの先頭の節点 X について、 $b=l-w$ から b を求める。また、 h, a を後述の方法で求める。

3° X が S -根であれば終了。

4° X の w を、その親の w に加える。

5° X の親の n_i について、 $n_i \leftarrow n_i - 1$ とする。 $n_i \geq 1$ なら 2° へ。 $n_i = 0$ なら X の親をキューの最後尾に入れるとともに、スタックにも入れる。2° へ。

(3) h, a の計算

葉については、 $h=0, a=0$ である。節点 X の子を $1, 2, \dots, m$ と番号付ける。子の b, w, h, a を b_i, w_i, h_i, a_i ($i=1, 2, \dots, m$) と記す。

[定理 1] 節点 X が P 節点であれば、

$$h = \sum_{i=1}^m \min\{w_i, b_i\} - \max_{1 \leq i \leq m} [\min\{w_i, b_i\} - h_i], \quad (1)$$

$$a = \min\{a_1, a_m\}, \quad (2)$$

ただし、 a_1, a_m は次式で与えられる。

$$a_1 = \sum_{i=1}^m \min\{w_i, b_i\} - \max_{1 \leq i \neq j \leq m} [\min\{w_i, b_i\} - h_i + \min\{w_j, b_j\} - h_j] \quad (3)$$

$$a_m = \sum_{i=1}^m w_i - \max_{1 \leq i \leq m} [w_i - a_i] \quad (4)$$

(証明) P 節点 X を除去葉数最小で H 型にするには、 X の子のうち、 $\min\{w_i, b_i\} - h_i$ が最大の子を H 型とし、ほかの子は、 $b_i \leq w_i$ なら B 型に、 $b_i > w_i$ なら W 型にすればよい。したがって、式(1)を得る。次に、 P 節点 X を A 型にするには、 X の子のうちの 2 個を H 型とし、ほかは B 型か W 型にするか、 X の子のうちの 1 個を A 型とし、ほかは W 型にするかのどちらかである。前の場合の最小除去葉数は、式(1)と同様の考察から、式(3)で与えられる。後の場合は、 $w_i - a_i$ が最大である子を A 型にすればよく、式(4)を得る。式(3)、(4)で与えられる除去葉数の小さい方を a とすればよい。

[定理 2] 節点 X が Q 節点であれば、

$$h = \min_{1 \leq k \leq m} \left[\min \left\{ \sum_{i=1}^{k-1} (w_i - b_i) - b_k + \sum_{i=1}^m b_i, \sum_{i=1}^{k-1} (b_i - w_i) - w_k + \sum_{i=1}^m w_i \right\} + h_k \right] \quad (5)$$

$$a = \min\{a_1, a_m\} \quad (6)$$

である。ここに、 a_1, a_m は次式で与えられる。

$$a_1 = \sum_{i=1}^m b_i - \max_{1 \leq j < k \leq m} [y_j + z_k], \quad (7)$$

$$y_j = \sum_{i=1}^{j-1} (b_i - w_i) + b_j - h_j, \quad (8)$$

$$z_k = \sum_{i=k+1}^m (b_i - w_i) + b_k - h_k, \quad (9)$$

$$a_m = \sum_{i=1}^m w_i - \max_{1 \leq i \leq m} [w_i - a_i] \quad (10)$$

(証明) Q 節点 X を H 型とするには、 X の子のうちの 1 個を H 型とし、それより左側にある子をすべて B 型 (あるいは W 型) にし、右側にある子をすべて W 型 (あるいは B 型) にすればよい。このときの除去葉数は、

$$\sum_{i=1}^{k-1} w_i + h_k + \sum_{i=k+1}^m b_i$$

あるいは

$$\sum_{i=1}^{k-1} b_i + h_k + \sum_{i=k+1}^m w_i$$

であり、これから式 (5) が導かれる。次に、 Q 節
点 X を A 型にするには、2つの方法がある。第1は、
2個の子を H 型とし、それらの間にはさまれた子を
すべて B 型とし、それらの外側の子をすべて W 型
とする方法である。2個の子を j と k とすると、除去
葉数は、

$$\sum_{i=1}^{j-1} w_i + h_j + \sum_{i=j+1}^{k-1} b_i + h_k + \sum_{i=k+1}^m w_i$$

となり、これから最小除去葉数を与える式 (7) が導
かれる。第2の方法は、 X の子の1個を A 型とし、
その外側の子をすべて W 型にする方法である。この
ときの除去葉数は、 P 節点の場合と同様にして導ける。
式 (7), (10) で与えられる除去葉数の小さい方を a
とすればよい。

式 (1), (3), (4), (5), (10) を計算するには、
 X の子を1から m へ1回走査すればよい。また、式
(7) を計算するには、 X の子を1から m までと、 m
から1へと1回ずつ、計2回走査すればよい。このアル
ゴリズムを下に示す。 X の子を1から j まで走査し
たときの $\max y_i (1 \leq i \leq j)$ を L_j とし、 \max を与え
る子の番号 i を C_j に記録する。また、 X の子を m
から k まで走査したとき、 $L_{k-1} + z_k$ を R_k とし、
 $\max R_i (m \geq i \geq k)$ を M とおく。この \max を与え
る子の番号 i を c に記録する。

〈アルゴリズム QA1〉

1° $L_1 \leftarrow b_1 - h_1$, $C_1 \leftarrow 1$, $j \leftarrow 1$ とする。

2° $j \leftarrow j+1$ とする。 $j=m$ なら 4° へ。

3° $L_j \leftarrow L_{j-1} + h_j - w_{j-1} + b_j - h_j$ とする。

$L_j > L_{j-1}$ なら $C_j \leftarrow j$ として 2° へ。 $L_j \leq L_{j-1}$ なら、

$L_j \leftarrow L_{j-1}$, $C_j \leftarrow C_{j-1}$ として 2° へ。

4° $z_m \leftarrow b_m - h_m$, $R_m \leftarrow L_{m-1} + z_m$, $M \leftarrow R_m$, $c \leftarrow m$,
 $k \leftarrow m$ とする。

5° $k \leftarrow k-1$ とする。 $k=1$ なら、 $a_1 = (\text{節点 } X \text{ の } b) - M$ として終了。 a_1 を与える X の H 型の子の
番号は、 c と C_c である。

6° $z_k \leftarrow z_{k+1} + k_{k+1} - w_{k+1} + b_k - h_k$, $R_k \leftarrow L_{k-1} + z_k$
とする。 $R_k > M$ なら、 $M \leftarrow R_k$, $c \leftarrow k$ として 5° へ。
 $R_k \leq M$ なら 5° へ。

S -葉を除去しても、 S -葉でない葉を除去しても、
同じ最小除去葉数が得られる場合は、 S -葉でない葉
を除去するように節点の型を決定する。これは、アル
ゴリズム PLAN において、番号が N より大きい G
の節点に処理が進んだとき、その節点に入る枝の少な
い方が、除去葉数が少なくなりうるからである。その

ため、実際に作成した計算機プログラムは、上のアル
ゴリズムと少し異なる。

4.3 ラベル決定の手続

アルゴリズム PLAN の Step 4.3 では、アルゴリ
ズム SCAN においてスタックに入れた節点を順次取
り出し、その型を決定し、節点にラベルを付ける。こ
のラベルに応じて、その節点の子のうち除去される
べき節点が生じた場合は、これを除去し、それが葉な
ら、葉の除去リストに入れる。葉でないなら、その節
点にラベル D を付け、さらに、それが S -節点でな
いなら除去節点のキューに入れる。ラベル D の節点
は、後程、スタックまたはキューから取り出され、そ
の子の除去処理が上と同様に行われる。

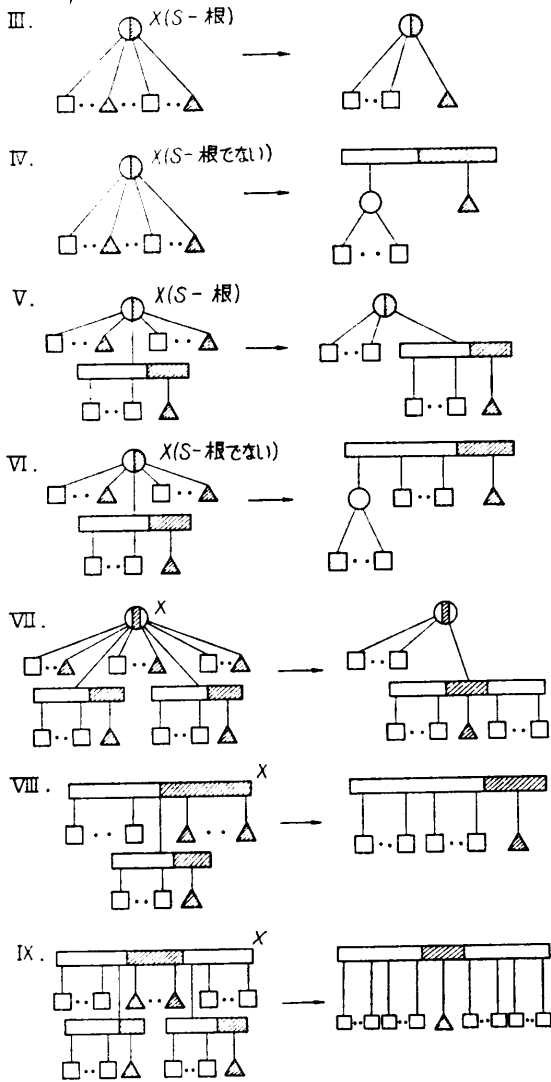
5. 簡 約

S -木に含まれる節点 X のラベルと、 X を根とする
部分 PQ -木の構造は、数種類のパターンに分類され
る。各パターンに対する簡約の方法は定まっ
ていて、公式化できる。したがって、 X に対してあてはまるパ
ターンを見つけ、公式に従って PQ -木を変形すれば
よい。この操作は、文献 7) の template matching
に相当するが、グラフの平面化に際しては、簡約の目
的が、 S -葉を1つの葉にまとめることであるから、
変形公式は、文献 7) のものより簡単化できる。図 1
にその公式を示す。

6. 計算の手間

グラフ G の各節点に対し、 S -木を求め、簡約する。
このために必要な処理のうち、主要なものは、除去葉
数を計算して節点のラベルを定める処理と、節点に変
形公式を適用する処理である。これらの処理は、 S -
節点について繰り返されるが、 S -節点とその子の数
は、グラフによって異なるので、その上限を考えてみ
る。 S -木を含む PQ -木 T_N は、処理の対象となる
 G の部分グラフを表わす⁹⁾。 T_N の葉は部分グラフの
枝の一部に対応し、各 Q 節点に対応する部分は、葉に
対応しない枝を少なくとも1本含むので、 T_N の葉数
と Q 節点数の和は、部分グラフの枝数を越えない。
 P 節点数 (Q 節点数も) は、部分グラフの節点数を越
えない。したがって、 T_N の葉を含む総節点数は、部
分グラフの枝数と節点数の和を越えず、これは $e+n$
以下である。 S -木の処理はただか n 回繰り返され
るだけなので、処理の総手間は、 $O(n(e+n))$ を越えな
い。アルゴリズムに含まれるほかの処理も、上記のオ

- I. XのラベルがW → 変形しない
- II. XのラベルがB → 1つのS-葉を置きかえる



- △: S-葉
- : ラベルWのP節点
- ⊙: ラベルHのP節点
- ⊕: ラベルAのP節点
- : ラベルWの節点 (葉, P節点, Q節点)
- ▨: ラベルHのQ節点
- ▩: ラベルAのQ節点

図1 簡約公式
Fig. 1 Reduction formulae.

ーダ以下であり, PLAN の計算複雑度は, $O(n(e+n))$ を越えない.

7. 計算結果

ランダムグラフを文献 9) による方法で発生し, そ

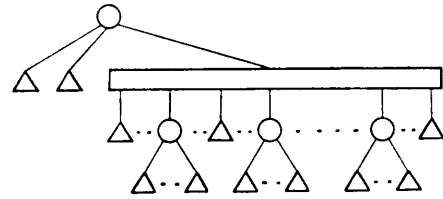


図2 大きな節点番号 N に対する PQ-木 T_N
Fig. 2 PQ-tree T_N for large vertex number N.

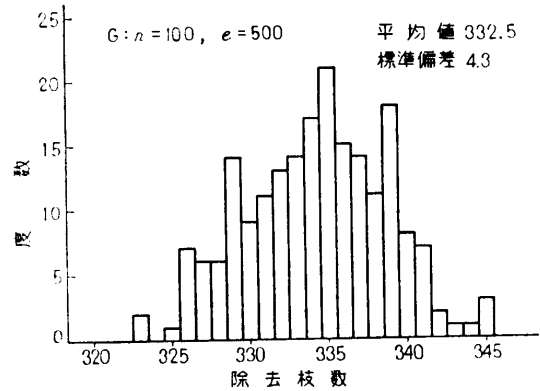


図3 st -番号を変えた際の除去枝数の度数分布
Fig. 3 Frequency distribution of the number of deleted edges for various st -numberings.

のうち2連結なグラフに対してアルゴリズム PLAN を適用した. 節点数 $n=30\sim 150$, 平均節点次数 $\bar{d}=6, 8, 10, 12$ のグラフについて, 次のような結果が得られた. ($\bar{d} \ll n$ である.)

7.1 一般的结果

グラフは, st -番号に従った節点の順に処理されるが, PQ-木 T_N は, 節点番号 N が大きくなると, ほとんど図2に示されるような構造をとる. この図のQ節点は, 1個の大きな2連結平面部分グラフを示す. このように大きな1個の2連結部分グラフが生じる理由は, 節点数の大きな2個の2連結部分グラフがあったとしても, それらに含まれる節点と節点Nの間に枝が存在する確率が大きく, それらはすぐ1個にまとめられるからと考えられる. また, PQ-木が図2のような構造をとると, S-葉はQ節点の下に散在し, 平面化の際, ほとんど除去され, たいてい1~3個が残るだけとなる. (平面グラフの平均節点次数は3以下である.)

7.2 st -番号による除去枝数の変化

平面化のための除去枝数は, st -番号を変化させると異なりうる. st -番号を200回変えて得られた除去

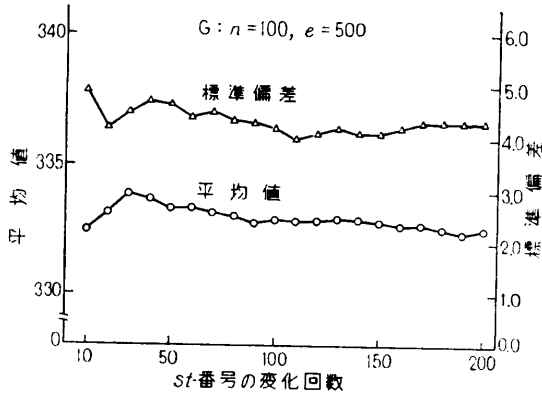


図4 除去枝数の平均値と標準偏差の変動

Fig. 4 Variation of the mean and the standard deviation of the number of deleted edges.

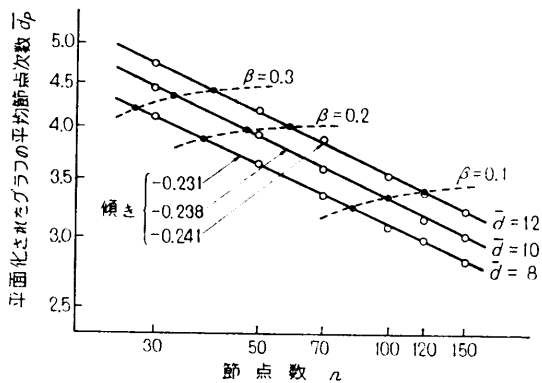


図5 平面化されたグラフの平均節点次数

Fig. 5 Average vertex degree of planarized graphs.

枝数の度数分布を図3のヒストグラムに示す。この例および節点数 50, 80, 100, 120 の計 15 のグラフの例については、*st*-番号を変化させたときの除去枝数がほとんど変わらない。すなわち、除去枝数の平均値に比べ、標準偏差が非常に小さい。さらに、この平均値と標準偏差は、変化させた *st*-番号の回数がかんり小さいときから大きいときまで、あまり変動しない。図4に例を示す。したがって、比較的少ない *st*-番号の回数によって、除去枝数の度数分布が予測でき、アルゴリズム PLAN によって得られる除去枝数の限界の見当がつけられる。次に、*st*-番号を 10 回変化させたときの除去枝数の変動を、いろいろの節点数と平均節点次数のグラフについて求めたものを表1に示す。表中、上段が除去枝数の変動範囲、下段がその平均値である。

7.3 グラフによる除去枝数の変化

1 個のグラフにつき、*st*-番号を 10 回変えて除去枝数の平均をとり、それをさらに、同じ節点数と平均節

表1 *st*-番号を変えたときの除去枝数の変動
上段：変動範囲，下段：平均値

Table 1 Variation of the number of deleted edges for 10 *st*-numberings.

\bar{d} \ n	30	50	70	100	120	150
6	32~43 38	71~79 76	102~115 109	158~162 160	207~210 209	—
8	54~62 59	106~116 113	156~166 162	241~250 245	289~305 298	383~394 388
10	77~87 83	151~160 155	217~228 223	325~338 334	402~413 410	514~531 524
12	106~112 109	191~203 196	278~290 286	419~433 425	511~523 517	658~668 663

表2 グラフを変えたときの除去枝数の変動
上段：変動範囲，下段：平均値

Table 2 Variation of the number of deleted edges for 10 graphs.

\bar{d} \ n	30	50	70	100	120	150
8	57~60 58	106~113 111	162~166 164	241~249 245	296~305 300	383~391 387
10	82~84 83	149~155 151	222~227 224	330~337 333	405~412 408	518~527 523
12	108~110 109	193~197 195	283~287 286	420~428 424	512~519 517	655~664 660

点次数を持つ 10 個の異なるグラフについて調べた結果を表2に示す。上段は、グラフを変えたとき、除去枝数の平均が変動する範囲、下段は、その平均値を示す。この表は、平面化のための除去枝数がグラフによってあまり変わらないことを示している。このとき得られた平面部分グラフの平均節点次数 \bar{d}_p を 10 個のグラフについて平均し、それを節点数 *n* についてプロットしたものを図5に示す。パラメータは、もとのグラフ *G* の平均節点次数 \bar{d} と、節点数が同じ完全グラフの枝数に対する *G* の枝数の比 $\beta = \bar{d}/(n-1)$ である。この図を用いると、平面部分グラフの枝数を、もとのグラフの節点数と枝数から推定することができる。

7.4 計算時間

平面化に要した計算時間の平均と節点数の関係を図6に示す。パラメータは *G* の平均節点次数 \bar{d} である。平均は、10 個の異なるグラフの各々について、*st*-番号を 10 回変えて平面化したときの所要時間の平均 (10×10=100 回の平均) である。この図からは、計算複雑度は、同じ節点次数のグラフについて、 $O(n^p)$, $p=1.41\sim 1.44$ であることがわかる。計算機は、京都大学大型計算機センター FACOM M-200 を使用した。

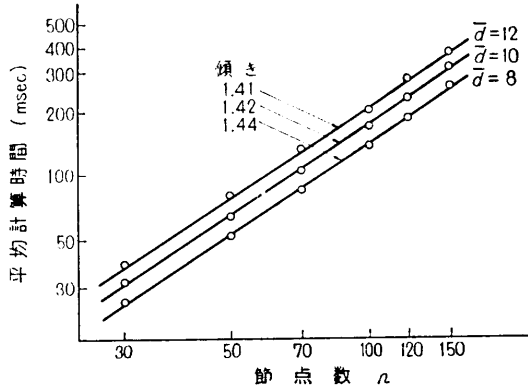


図 6 平面化のための平均計算時間

Fig. 6 Average computational time for planarization.

8. むすび

PQ-木を用いたグラフの平面化アルゴリズムでは、平面化と同時に、得られた平面部分グラフの平面描写が、PQ-木によって示される⁶⁾。この平面描写は、文献 4), 5) などのパスによる描写と異なり、節点、枝の描写がただちに得られるという利点がある。アルゴリズムをランダムグラフに適用した場合、 st -番号を変化させたときの除去枝数の度数分布 (図 3 など) から、適用例では、平面化のための最小除去枝数に近い除去枝数が得られているのではないかと推測される。もしそうなら、PLAN は、複雑な問題とされている最大平面化問題の効率よい近似解法であるといえる。平面化のための除去枝数が st -番号やグラフによってあまり変わらないことは、ランダムグラフの特質ではないかと推測される。文献 5) で得られた除去枝数は、PLAN で得られたそれよりやや大きい、大差ないといえる。

謝辞 この研究は、文部省科学研究費補助金「総合研究 A」, 435013「大規模システムにおける解析・設計・制御に関する基礎研究」の援助を受けた。

参考文献

- 1) 可児, 大附: 設計自動化におけるグラフ理論と組み合わせ算法 (2), 情報処理, Vol. 16, No.

- 6, pp. 526-537 (1975).
- 2) 白川 功: 実装設計におけるグラフ理論の応用, 電子通信学会誌, Vol. 62, No. 7, pp. 780-789 (1979).
- 3) Fisher, G.J. and Wing, O.: Computer Recognition and Extraction of Planar Graphs from the Incidence Matrix, IEEE, Trans. Cir. Th. Vol. CT-13, pp. 154-163 (1966).
- 4) van Cleemput, W. M.: On the Topological Aspects of the Circuit Layout Problem, Proc., 13th Design Automation Conf. pp. 441-450 (1976).
- 5) Chiba, T., Nishioka, I. and Shirakawa, I.: An Algorithm of Maximal Planarization of Graphs, 1979 Intern. Symp. Cir. and Syst. Proc., pp. 649-653 (1979).
- 6) Lempel, A., Even, S. and Cederbaum, I.: An Algorithm for Planarity Testing of Graphs, Theory of Graphs: International Symposium; Rome, July 1966, Rosenstiel, P. Edit. pp. 215-232, Gordon and Breach, N. Y. (1967).
- 7) Booth, K. S. and Lueker, G. S.: Testing for the Consecutive Ones Property, Interval Graphs and Graph Planarity Using PQ-tree Algorithms, J. Computer and Syst. Scie., Vol. 13, pp. 335-379 (1976).
- 8) Even, S. and Tarjan, R.: Computing an st -numbering, Theor. Comp. Scie., Vol. 2, pp. 339-344 (1976).
- 9) 高見沢, 滝内, 西関, 斎藤: ランダムグラフの統計解析, 情報処理, Vol. 19, No. 8, pp. 742-748 (1978).
- 10) Ozawa, T. and Nishizeki, T.: Properties of Certain Type of Random Graphs, 1979 Intern. Symp. Cir. Syst. Proc. pp., 88-91 (1979).
- 11) 五十嵐善英: 確率的アルゴリズムの概観, 情報処理, Vol. 21, No. 1, pp. 13-18 (1980).
- 12) 小沢, 高橋: PQ-木を用いたグラフの平面化アルゴリズム, 電子通信学会技術研究報告, 回路とシステム CAS 79-150 (1980).
- 13) Garey, M.R. and Johnson, D. S.: Computers and Intractability, W. N. Freeman and Co. San Francisco, p. 197 (1979).

(昭和 55 年 3 月 14 日受付)

(昭和 55 年 9 月 18 日採録)