

平面オイラーグラフの辺素な路問題を解く並列アルゴリズム

中山慎一 NAKAYAMA Shin-ichi
増山 繁 MASUYAMA Shigeru
豊橋技術科学大学 知識情報工学系

あらまし K 個の始点 s_k と終点 t_k の対 (s_k, t_k) , $k = 1, 2, \dots, K$ が与えられたとき、 s_k から t_k への d_k 本の路 ($k = 1, 2, \dots, K$) をグラフ $G = (V, E)$ 内に互いに素にとれるかどうかを問うのが辺素な路問題である。但し、 G 内の路が互いに辺を共有しないとき、これらは互いに素であるという。本稿では、この問題の制限された一つの部分クラスである、与えられた無向グラフ G と各終点 t_k と各始点 s_k とを結び辺 ($k = 1, 2, \dots, K$) で構成されるグラフ H とを合成したグラフが平面、かつ、オイラーグラフで、始点、終点の対の数 (K) が定数個の場合に対し、CREW PRAM上で辺素な路の存在判定を $O(n^3 + m)$ 個のプロセッサを用いて $O(\log^2 n + \log m)$ 時間で、辺素な路の構成を $O(n + m)$ 個のプロセッサを用いて $O(\log^2 n + \log m)$ 時間で求める並列アルゴリズムをそれぞれ開発した (ただし、 n は節点数、 m は辺数)。

1 はじめに

辺素な路問題 [3][5][6] は、指定された始点と終点の各組 ($k = 1, \dots, K$) に対し、始点から終点への互いに辺を共有しない路を、それぞれ、要求する本数 d_k 本取れるかどうかを決定する問題であり、グラフ理論における基本的な問題の一つである (以下、本稿では、 $d_k = 1$ 、 K は定数とする)。これは、通信ネットワーク、VLSI デザイン、並列計算におけるプロセス実行のスケジューリング問題などに適用でき、幅広い応用がある。

始点と終点の組の数 (K) が 1 の場合、“ s から t への辺素な路の最大本数は、 s と t を分離する最小カットのサイズに等しい” という、いわゆる Menger の定理 [1] が知られている。

$K \geq 2$ の場合、辺素な路問題は多品種フロー問題の離散版と理解することができる。しかし、一般のグラフ G では、辺素な路問題は $K = 2$ の場合であっても NP困難であり [2]、効率の良い算法は存在しないと予想される。そのため、グラフ G の形状や (s_i, t_i) の位置に制限をおいて、多項式時間で解く逐次アルゴリズムが種々開発されている。辺素な路問題において、与えられた問題のグラフ G を供給グラフと呼び、辺集合 $F = \{(t_i, s_i) \mid i = 1, 2, \dots, k\}$ によって定まる有向グラフ $H = (V, F)$ を需要グラフと呼ぶ。 $G + H$ が有向オイラーグラフで、かつ、 $k = 2$ の場合、Frank [3] により多項式時間で解けることが示され、更に、[7] によって、CREW PRAM上で $O(m)$ 個のプロセッサを用いて、 $O(\log^2 m)$ 時間で解く並列アルゴリズムが示されている。また、 $k = 3$ の場合、茨木、Poljak [6] により多項式時間で解けることが示された。

本稿では、辺素な路問題の制限された一つの部分クラスである、 $G + H$ が平面グラフで、かつ、無向オイラーグラフ (以下、平面オイラーグラフと呼ぶ) (図 1 参照) の辺素な路の存在判定、および、辺素な路の構成に対し、それぞれ並列アルゴリズムを開発したので報告する。これらの並列アルゴリズムは、並列計算モデルのひとつである CREW PRAM [4] 上で、辺素な路の存在判定を $O(n^3 + m)$ 個のプロセッサを用いて $O(\log^2 n + \log m)$ 時間、辺素な路の構成を $O(n + m)$ 個のプロセッサを用いて

$O(\log^2 n + \log m)$ 時間で求める (ただし、 n は節点数、 m は辺数、また、与えられたグラフは多重辺を持っていてもよいものとする)。

2 定義

$G = (V, E)$ を節点集合 V 、辺集合 E からなる自己閉路を持たない無向グラフとする。グラフが平面の中に埋め込み可能である、あるいは平面的であるというのは、そのグラフを平面内に描く場合、2 つの辺が交わるのはそれらがともに接続している節点においてのみであるように描くことができることをいう。平面へ埋め込んだ平面的グラフを平面グラフという。平面グラフは、平面をいくつかの辺で囲まれた連結領域に分割するが、連結領域の閉包 (境界を含めた部分) はそれぞれ面と呼ばれる (図 1 参照)。

平面グラフ G に対する双対グラフ $G^* = (V^*, E^*)$ を以下のように定義する。 G の各面 f に対応して G^* の 1 つの節点 f^* をとり、 G の各辺 e に対応して G^* の 1 つの辺 e^* をとる。 G^* の 2 節点 $f^*, g^* \in V^*$ が G^* の辺 e^* によって結ばれるのは、対応する G の面 f と g が G の辺 e によって分離されているとき、かつそのときに限る [1]。

系列 $v_0, e_1, v_1, e_2, v_2, \dots, e_p, v_p$ は、 $v_i \in V, i = 0, 1, \dots, p$ 、および、 $e_i = (v_{i-1}, v_i) \in E, i = 1, 2, \dots, p$ 、の条件を満たすとき、 v_0 から v_p への路であるという。また、 v_0, v_p をそれぞれ始点、終点、 v_1, v_2, \dots, v_{p-1} を内点という。

グラフ $G = (V, E)$ 内の路 (paths) π, π' が互いに辺を共有しないとき、これらは互いに辺素であるという。 K 個の始点 s_k と終点 t_k の対 $(s_k, t_k), k = 1, 2, \dots, K$ 、と要求量 d_k (非負整数)、 $k = 1, 2, \dots, K$ 、が与えられたとき、 s_k から t_k へのそれぞれ d_k 本の路 ($k = 1, 2, \dots, K$)、すなわち $\sum_{k=1}^K d_k$ 本の路のすべてを、 G 内に互いに辺を共有しないようにとれるかどうかを問う決定問題を、辺素な路問題 [3][5][6] という。

辺素な路問題において、与えられた問題のグラフ G を供給グラフと呼び、辺集合 $F = \{(t_i, s_i) \mid i = 1, 2, \dots, K\}$ によって定まるグラフ $H = (V, F)$ を需要グラフと呼ぶ。

2 節点 $s, t \in V$ が与えられたとき、節点集合 $X \subseteq V$ は、 $s \in X$ と $t \in V - X$ の条件を満たす時、 st -カットであるという。集合 $X \subseteq V$ に対し

$$\delta_G(X) = \{(v_i, v_j) \in E \mid v_i \in X, v_j \in V - X\} \quad (1)$$

と記す。

問題例 $G + H$ が辺素な路を持つためには、

$$d_G(X) \geq d_H(X), \quad \forall X \subseteq V \quad (2)$$

でなければならない。ただし、

$$d_G(X) = |\delta_G(X)| \quad (3)$$

$$d_H(X) = \sum_{(t_k, s_k) \in \delta_H(V-X)} d_k \quad (4)$$

である。式 (2) をカット条件と呼ぶ [5]。式 (3)、(4) の $\delta_G(X)$ 、 $\delta_H(V - X)$ は式 (1) の定義から、グラフ G (グラフ H) において X から $V - X$ ($V - X$ から X) へ橋渡ししている辺の集合である。

3 $G + H$ が平面オイラーグラフの場合の辺素な路問題

カット条件は、要求本数分の辺素な路が存在するための必要条件であるが、一般には十分条件ではない。カット条件が十分条件となれば、カット条件を調べることにより辺素な路の存在判定を行なえる。そこで、供給グラフ G と需要グラフ H がどのような条件を満たせばカット条件が十分条件となるかについて、種々の結果があきらかにされている [5]。

Seymour [8] により、次の補題が示されている。

補題 1 [8] 供給グラフ G に需要グラフ H を加えた無向グラフ $G + H = (V, E \cup F)$ が平面オイラーグラフの場合 (図 1 参照)、カット条件は辺素な路が存在するための必要十分条件である。□

本稿では、 $G + H$ が平面オイラーグラフの場合に対し、3.1 において辺素な路の存在判定の並列アルゴリズムを、また 3.2 では、辺素な路が存在するならば、それらの路を構成する並列アルゴリズムをそれぞれ述べる。

3.1 辺素な路の存在判定

補題 1 より、全ての $X \subset V$ に対してカット条件を調べれば、辺素な路が存在するかどうか判定できる。ところが、 $X \subset V$ は $2^{|V|}$ 個あるので、そのままでは効率の良いアルゴリズムは得られない。しかしながら、 $G + H$ が平面グラフなので、平面グラフの辺素な閉路の各々は、双対グラフ $(G + H)^*$ の辺素なカットの各々に対応し、更に、 $G + H$ が平面オイラーグラフの場合、その双対グラフ $(G + H)^*$ は 2 部グラフ $(V_1^*, V_2^*, E^* \cup F^*)$ になる [8] という事実を用いる。本稿では、以下、 $(G + H)^*$ の節点集合と $(G + H)^*$ における G に対応する辺集合 E^* から生成される部分グラフを $\langle E^* \rangle$ と表し、 $(G + H)^*$ の節点集合と $(G + H)^*$ における H に対応する辺集合 F^* から生成される部分グラフを $\langle F^* \rangle$ と表す (図 2 参照)。

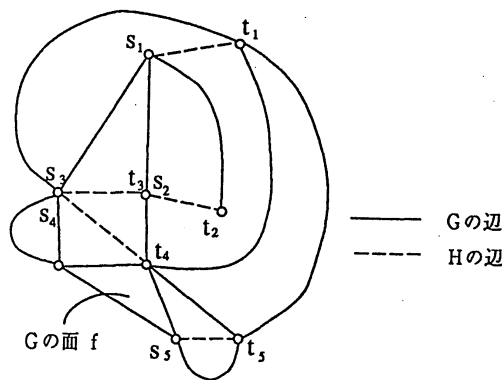


図 1: $G + H$ が平面オイラーグラフの場合

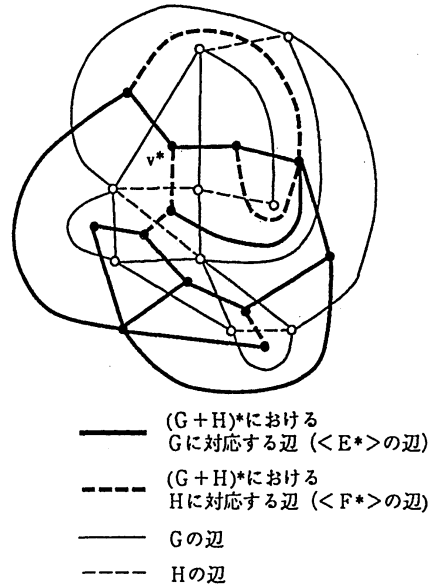


図 2: $G + H$ の双対グラフ $(G + H)^*$

これらより、補題1は次のように示せる。

補題2 [8] $(G+H)^*$ が2部グラフとする。 $\langle F^* \rangle$ の辺をただか1つ含むような辺素なカットが K 個存在するための必要十分条件は、 $(G+H)^*$ の全ての閉路上において $\langle E^* \rangle$ の辺数が $\langle F^* \rangle$ の辺数以上であることである □

補題2より、 $(G+H)^*$ に $\langle E^* \rangle$ の辺数が $\langle F^* \rangle$ の辺数より多いような閉路が1つも存在しないかどうか調べればよい。そのため、 $(G+H)^*$ の節点集合と $\langle E^* \rangle$ の辺のみからなる $(G+H)^*$ の部分グラフ $\langle E^* \rangle$ 上での各節点対間の最短距離と、 $(G+H)^*$ の節点集合と $(G+H)^*$ において H に対応する辺からなる $(G+H)^*$ の部分グラフ $\langle F^* \rangle$ 上での各節点対間の最長距離をそれぞれ求め、各節点対間における両者の差をとり、もしその差が負である、すなわち $\langle F^* \rangle$ の辺数のほうが多い節点対が存在すれば、 $\langle E^* \rangle$ の辺数が $\langle F^* \rangle$ の辺数以上である閉路が存在したことになり、補題2より辺素な路は存在しないことになる。各節点対間の $\langle E^* \rangle$ の辺による最短距離は、並列アルゴリズム [4] で求まる。一方、 $\langle F^* \rangle$ の辺のみを用いた最長距離は、次のようにして求められる。無向グラフ $\langle F^* \rangle$ が木ならば、その辺集合による各節点対間の距離が最長距離になる。そこで、 $\langle F^* \rangle$ に閉路が存在するかどうか調べ、閉路が存在するならば、補題2より辺素な路は存在しない。一方、閉路が存在しないならば、 $\langle F^* \rangle$ はいくつかの木からなる森となるので、各節点対間の距離が求まる。以上をまとめると、求める並列アルゴリズムは次のようになる。なお、入力データである $G+H$ に対して、その双対グラフ $(G+H)^*$ は、多項式オーダー個のプロセッサを用いて、対数の多項式オーダー時間で求まる [4]。そこで、このデータは、あらかじめ与えられているものとする。

procedure feasibility

{ 入力: 双対グラフ $(G+H)^*$ }

{ 出力: 辺素な道が存在するとき Yes、しないとき No }

- (1) $\{(G+H)^*$ における2節点間に $\langle F^* \rangle$ の辺のみからなる多重辺が存在するかどうか調べる。
 $\langle F^* \rangle$ の辺 (i, j) が格納されている配列 $\langle F^* \rangle _EDGE(i)$, $i = 1, \dots, K$ を、辞書式順序 $\{(i, j) < (k, l) \text{ if } i < k \text{ or } (i = k \text{ and } j < l)\}$ にソートする。

```
for all  $i, 1 \leq i \leq K$  in parallel do
  if  $\langle F^* \rangle \_EDGE(i) = \langle F^* \rangle \_EDGE(i+1)$ 
  then (7) ←
```

- (2) $\{(G+H)^*$ の節点集合と $(G+H)^*$ における H の辺に対応する辺のみからなる部分グラフ $\langle F^* \rangle$ に閉路が存在するかどうか調べる }

グラフ $\langle F^* \rangle$ の連結成分の個数 k を求める [4]。森であることの必要十分条件である、 $K = |V^{**}| - k$ が成立しているかどうか調べる (ただし、 K は $\langle F^* \rangle$ の辺数、 V^{**} は、 $\langle F^* \rangle$ の節点集合 V_1^*, V_2^* より孤立点を除いた節点集合)。成立していなければ、(7) ←。

- (3) $\langle F^* \rangle$ の各節点対間の距離を求め [4]、行列 $\langle F^* \rangle_{ij}$, $i, j = 1, \dots, n$ に、求めた各節点 i, j 間の距離を代入。ただし、節点対間に経路、または、辺が存在しない節点対に対しては0を代入。

- (4) { 初期化 }

```
for all  $i, j \in V^*$  in parallel do
  if  $\langle E^* \rangle$  の辺が節点  $i, j$  間に存在する
  then  $\langle E^* \rangle_{ij} \leftarrow 1$ 
  else  $\langle E^* \rangle_{ij} \leftarrow 0$ 
```

- (5) $(G+H)^*$ の節点集合と $(G+H)^*$ における G の辺に対応する辺のみからなる部分グラフ $\langle E^* \rangle$ 上での各節点対間の最短経路を求める [4]。節点対間に経路が存在しない節点対に対しては、最短経路長を ∞ とする。

- (6) { 各節点対の間における $\langle E^* \rangle$ の辺数と $\langle F^* \rangle$ の辺数の差 }

```
for all  $i, j \in V^*$  in parallel do
  begin
     $\langle E^* \rangle_{ij} \leftarrow \langle E^* \rangle_{ij} - \langle F^* \rangle_{ij}$ 
    if  $\langle E^* \rangle_{ij} < 0$ 
    then (6) ←
    else 辺素な路は存在するので Yes を出力。
    停止。
  end
```

- (7) 辺素な路は存在しないので No を出力。停止。 □

補題3 procedure feasibility により、 $G+H$ に K 個の辺素な路が存在するかどうか正しく判定できる。 □

(証明) (1)で、双対グラフ $(G+H)^*$ の2節点間に $\langle F^* \rangle$ の辺のみからなる多重辺が存在すると分かると、その節点間で $\langle F^* \rangle$ の辺のみからなる閉路が構成されるので、補題2より辺素な路は存在しないことがいえる。

(2)で、 $(G+H)^*$ の節点集合と $(G+H)^*$ における H の辺に対応する辺のみからなる部分グラフ $\langle F^* \rangle$ に閉路が存在するならば、補題2より辺素な路は存在しないことがいえる。

(1)、(2)により $\langle F^* \rangle$ に閉路は含まれないことがいえたので、以下において $\langle F^* \rangle$ はいくつかの木からなる森である。

(6)は、(3)で求めた $(G+H)^*$ の節点集合と $(G+H)^*$ における H の辺に対応する辺のみからなる部分グラフ $\langle F^* \rangle$ 上での各節点対間の距離と、(5)で求めた $(G+H)^*$ の節点集合と $(G+H)^*$ における G の辺に対応する辺のみからなる部分グラフ $\langle E^* \rangle$ 上での各節点対間の最短距離との差をとることにより、その差が負になる節点対が存在すれば、 $\langle F^* \rangle$ の辺数が $\langle E^* \rangle$ の辺数以上である閉路が存在することになり、補題2より辺素な路は存在しない。そうでなければ辺素な路は存在する。 □

次に、計算時間とプロセッサ数の解析を行なう。

入力データである $G+H$ に対して、その双対グラフ $(G+H)^*$ は、多項式オーダー個のプロセッサを用いて、対数の多項式オーダー時間で求まる [4]。このデータは、あらかじめ与えられているものとする。

- (1): 辞書式順序は、ソーティングアルゴリズムを用いて求まる。 $O(m)$ 個のプロセッサを用いて $O(\log m)$ 時間 [4] (多重辺をもたないグラフならば、 $G+H$ は平面グラフなので m は $O(n)$ となり、 $O(n)$ 個のプロセッサを用いて $O(\log n)$ 時間となる)。
- (2): $O(m)$ 個のプロセッサを用いて $O(\log^2 n)$ 時間 [4]。
- (3): [4] の木に関する並列アルゴリズムを用いる。 $O(n)$ 個のプロセッサを用いて $O(\log n)$ 時間。
- (4): $O(m)$ 個のプロセッサを用いて、 $O(1)$ 時間。
- (5): 最短経路を求める並列アルゴリズム [4] を用いる。 $O(n^3)$ 個のプロセッサを用いて、 $O(\log^2 n)$ 時間。
- (6): 各配列 $\langle E^* \rangle_{ij}, \langle F^* \rangle_{ij}, i, j = 1, \dots, n$ にプロセッサを割り当て、計算し、負の値が存在するか調べる。 $O(n^2)$ 個のプロセッサを用いて $O(1)$ 時間。

以上より、procedure feasibility は、全体として $O(n^3 + m)$ 個のプロセッサを用いて $O(\log^2 n + \log m)$ 時間で解を与えることができる。

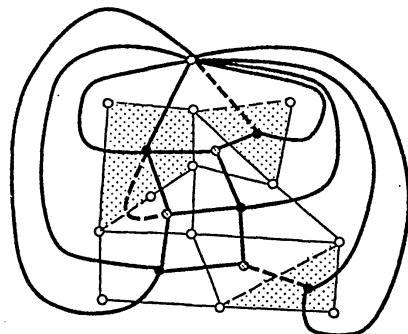
定理1 procedure feasibility は、CIRCUIT PRAM 上で全体として、 $O(n^3 + m)$ 個のプロセッサを用いて $O(\log^2 n + \log m)$ 時間で解を与える。 □

3.2 辺素な路の構成

ここでは、先の procedure feasibility で $G+H$ に要求する K 個の辺素な路が存在することが確かめられた後、実際に K 個 (K は定数) の辺素な路を構成する並列アルゴリズムを述べる。

$G+H$ において、 H の辺をそれぞれ1つだけ含むような閉路を K 個構成することは、補題2に示されているように、 $G+H$ の双対グラフ $(G+H)^*$ において、 $\langle F^* \rangle$ の辺をたかだか1つ含むような辺素なカットを K 個みつめることに対応する。よって、 $(G+H)^*$ において $\langle F^* \rangle$ の辺をたかだか1つ含む辺素なカットをみつけ、そのカットに対応する $G+H$ の辺を出力することにより、辺素な路を逐次構成していく (ここで、 K が定数としていることに注意)。

$G+H$ の双対グラフ $(G+H)^*$ の節点 $v^* \in V_1^*, V_2^*$ は、 $G+H$ の面に対応する。よって、 $\langle F^* \rangle$ の辺が1本しか接続していないような v^* においては、 v^* に対応する $G+H$ の面は、 v^* に接続する辺 (v^*, u_i^*) , $i = 1, \dots, p$ に対応する H の辺をたかだか1つ含んだ $G+H$ の辺からなる閉路に囲まれている。また、 $G+H$ は平面オイラーグラフなので、葉 (次数1の節点) や橋 (それを取り除くと連結度がひとつ増える辺) を持たず、いくつかの面によりグラフ全体が構成されており、更に、 H の辺を1つ含んだ面どうしが互いに辺を共有しなければ、 K 個の面で K 個の辺素な路を構成することができる。よって、 $(G+H)^*$ は2部グラフ $(V_1^*, V_2^*, E^* \cup F^*)$ であることに注意すると、 V_1^* または、 V_2^* のいずれかの節点集合に K 個の $\langle F^* \rangle$ の辺が1本接続している節点 v^* が存在すれば、それぞれの v^* に対応する面で辺素な路が構成できる (図3参照)。このような場合、 K 個の閉路を並列に同時に構成できる。



— ($G+H$)^{*} における G に対応する辺 ($\langle E^* \rangle$ の辺)
 - - - ($G+H$)^{*} における H に対応する辺 ($\langle F^* \rangle$ の辺)
 — G の辺
 - - - H の辺
 ● V_1
 ○ V_2

図3: $|F|$ 個の v^* がとれる場合

ところが、周囲に H の辺を2本以上含む $G+H$ の面が存在する場合、 V_1^* または、 V_2^* のいずれかに $\langle F^* \rangle$ の辺が1本接続している v^* が K 個とれない場合が生じる。このような場合、それを先に構成しても他の辺素な路をうまくとると要求本数分、辺素な路をとることができるようなひとつの閉路をまず構成し、その閉路を $G+H$ から取り除く。

そのために、まず、後述の procedure 中の (2) の (α), (β) に該当しない v^* をひとつ選び、 v^* に接続する辺に対応する $G+H$ の1つの閉路を構成する。つぎに、 v^* に接続した $\langle E^* \rangle$ の辺 (v^*, u_i^*) , $i = 1, \dots, p$ 、および、 $\langle F^* \rangle$ の辺 (v^*, u^*) を取り除き、 v^* に接続している $\langle E^* \rangle$ の辺 (v^*, u_i^*) の他方の端点 u_i^* をすべて u^* にする。この処理により、 v^* に対応する $G+H$ の面 (H の辺を1つ含んだ閉路) が除去される。

以上の処理を定数回 ($O(K)$ 回)、繰り返す。

辺素な路を要求本数分構成する並列アルゴリズムは、次のようになる。

procedure disjoint path construction

{ 入力: 与えられたグラフ $G+H$ とその双対グラフ $(G+H)^*$ }
 { 出力: K 個の辺素な路 }

- (1) $(G+H)^* = (V_1^*, V_2^*, E^* \cup F^*)$ の各節点 $v^* \in V_1^*, V_2^*$ が、 V_1^* または、 V_2^* のいずれかに属するか調べる。
- (2) V_1^* または、 V_2^* のいずれかに、 $\langle F^* \rangle$ の辺が1本のみ接続した節点 v_k^* が K 個存在するかどうか調べる。
 存在するならば、 K 個の各 v_k^* , $k = 1, \dots, K$ に接続する $\langle E^* \rangle$ の辺 (v_k^*, w_i^*) , $i = 1, \dots, q$ に対応する G の辺 (v_k, w_i) を出力する。 K 個の辺素な路が構成されている。

停止。

存在しなければ、次の (α), (β) に該当しないような節点 v^* を1つ選択する。

(α) 節点 v^* に $\langle F^* \rangle$ の辺が2本以上接続している (図4(a)参照)

(β) v^* に接続している $\langle F^* \rangle$ の辺は1本 (u^*, v^*) のみだが、接続している $\langle E^* \rangle$ の辺 (v^*, w_j^*), $j = 1, \dots, p$ の他方の端点 w_j^* に接続している $\langle F^* \rangle$ の辺が2つ以上存在する (但し、 $w_j^* \neq u^*$, すなわち、 $\langle E^* \rangle$ の辺 (v^*, u^*) は除く)(図4(b)参照)

(3) { 路の出力 } v^* に接続する $\langle E^* \rangle$ の辺 (v^*, u_i^*) に対応する G の辺 (v, u_i) を出力

(4) $\{(G+H)^*$ の辺の除去、すなわち、 $G+H$ の閉路の除去}

v^* に接続していた $\langle F^* \rangle$ の辺の他方の端点を u^* とする。
 for all v^* に接続する辺 (v^*, u^*) in parallel do
 $(v^*, u_i^*) \leftarrow \text{null}$
 for all 端点に u_i^* をもつ辺 $(*, u_i^*)$ in parallel do
 $(*, u_i^*) \leftarrow (*, u^*)$

(5) $K \leftarrow K - 1$ (K は定数)
 (2)へ。□

以下では、アルゴリズムの正当性の証明、および計算時間とプロセッサ数の解析を行なう。

アルゴリズム全体の正当性の証明の前に、(2)で、 V_1^* または、 V_2^* のいずれかに、 $\langle F^* \rangle$ の辺が1本のみ接続している節点 v_i^* が K 個存在しない場合、それを先に選択しても後で他の閉路がとれる $(G+H)^*$ の節点を選択しなければならないが、この時 procedure disjoint path construction の (2) の (α), (β) に該当する節点は選ぶことができず、かつ、 $\langle F^* \rangle$ の辺が接続している節点 v^* において、それを先に選択できないのは、その2通りの場合のみであることをいう。

補題4 K 個の辺素な路が存在する場合、 $\langle F^* \rangle$ の辺が接続している節点 v^* を先に選択できないのは、アルゴリズムの (2) で述べた (α), (β) の2通りの場合のみである。□

(証明) アルゴリズムの (2) の (α) を満たす v^* に対しては、 v^* に対応する $G+H$ の面に H の辺が2つ以上存在することになるので、その面は H の辺を丁度1つ含んだ閉路を構成できない。

辺素な路が存在すれば、補題3の証明で述べたように、 $\langle F^* \rangle$ はいくつかの木からなるので、 $\langle F^* \rangle$ の辺が接続している節点で (α) を満たすもの以外は、木の葉にあたる部分、すなわち $\langle F^* \rangle$ の辺が1本のみ接続している節点である。この節点 v^* に接続する $\langle E^* \rangle$ の辺 (v^*, w_j^*), $j = 1, \dots, p$ の他方の端点に接続する辺が $\langle E^* \rangle$ の辺のみの場合、 w_j^* に対応する $G+H$ の面は、 G の辺のみから構成され、かつ、 $G+H$ がオイラーグラフなので、 (v^*, u^*) に対応する $\langle F^* \rangle$ の辺と (v^*, w_j^*) に対応

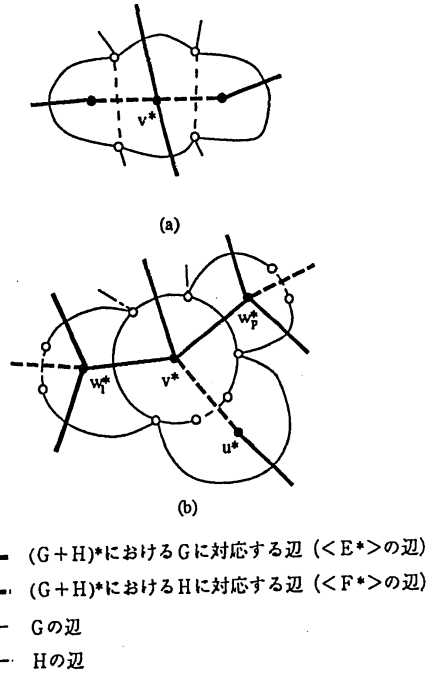


図4: procedure disjoint path construction のステップ2で選択できない節点

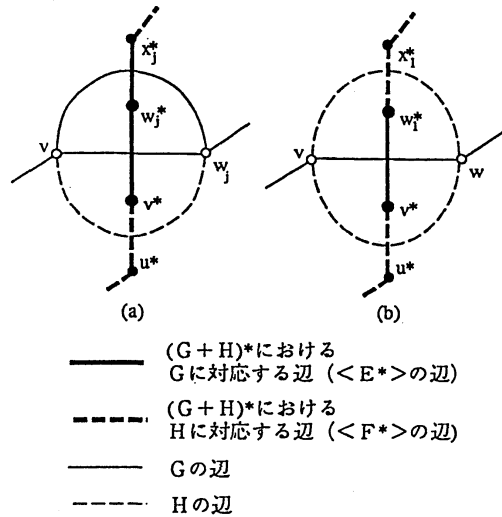


図5: 補題4の図示

する G の辺 (v, w_j) で閉路を構成しても、他の閉路は v, w_j に接続している他の G の辺を用いばよい (図 5(a) 参照)。

v^* に接続する $\langle E^* \rangle$ の辺 (v^*, w_j^*) の他方の端点に接続する辺が、 $\langle F^* \rangle$ の辺 1 本 (w_i^*, x_i^*) のみの場合も、 v^* を選択したとしても他の辺素な路が存在するにもかかわらずそれがとれなくなることはない。なぜならば、 w_i^*, v^* に対応する $G+H$ の面のいずれを用いて閉路を構成しようとしても、 $G+H$ はオイラーグラフで、かつ、辺素な路が存在することは既に示されているので、 (v^*, w_i^*) に対応する G の辺 (v, w) の節点 v, w からは、他の G の辺が必ず出ていなければならない、 $(u^*, v^*), (v^*, w_i^*)$ を選んだ、すなわち、 $(u^*, v^*), (v^*, w_i^*)$ に対応する H, G の辺で閉路を構成したとしても、残りの $\langle F^* \rangle$ の辺 (w_i^*, x_i^*) に対応する H の辺に対しては、残りの v, w に接続している G の辺で路を構成すればよいので (図 5(b) 参照)、他方の辺素な路が構成できなくなることはない。

v^* に接続する $\langle E^* \rangle$ の辺 (v^*, w_j^*) の他方の端点に接続する辺に、 $\langle F^* \rangle$ の辺が 2 本以上存在する場合、 $(G+H)^*$ の v^* に対応する $G+H$ の面に、 G の辺を共有する H の辺を含んだ他の面が 2 個以上存在する。(図 4(b) 参照)。 G の辺を共有する H の辺を含んだ 2 つ以上の他の面 (図 4(l) の w_i^*, w_p^* に対応する面) が、この v^* に対応する面の G の辺を用いなければ閉路を構成できない場合、 v^* に対応する面を先に閉路として構成すると、 G の辺を共有していた他の 2 つ以上の閉路が必要とする G の辺を用いるため、後に、他の 2 つ以上の閉路を構成できなくなる。よって、このような v^* に対応する $G+H$ の閉路は先に構成できない。この閉路に対応する、 $(G+H)^*$ における節点 v^* は、アルゴリズムの (2) で述べた (β) を満たす。

以上より、 K 個の辺素な路が存在する場合、 $\langle F^* \rangle$ の辺が接続している節点 v^* で選べないのは、 v^* が (α) 、又は (β) を満たす場合のみである。□

補題 5 K 個の辺素な路が存在するならば、procedure disjoint path construction で K 個の辺素な路が構成される。□

(証明) (2) で、 $(G+H)^*$ の V_1^* または、 V_2^* のいずれかに、 $\langle F^* \rangle$ の辺が 1 本のみ接続した節点 v^* が K 個存在した場合、 K 個の辺素なカットが存在したことになるので、そのカットに対応した $G+H$ の辺を選ぶことにより、 K 個の辺素な路が構成できる。

V_1^* または、 V_2^* のいずれかに、 v^* が K 個存在しなかった場合、常にアルゴリズムの (2) の (α) 、 (β) に該当しないような節点が $V_1^* \cup V_2^*$ に存在することをいう。

まず初期状態でアルゴリズムの (2) の (α) 、 (β) に該当しないような節点 v^* が 1 つも存在しないとすると、 v^* が (α) をみたくものみだとすると、全ての v^* が $\langle F^* \rangle$ の辺で接続されるので、 $\langle F^* \rangle$ の辺による閉路を構成することになり、補題 2 に矛盾する。

v^* が (β) を満たすということは、 v^* に接続する $\langle E^* \rangle$ の辺 (v^*, w_i^*) 、 $i = 1, \dots, p$ に接続する辺 (w_i^*, x_i^*) の中に、少なくとも 1 本は $\langle F^* \rangle$ の辺が含まれていることを意味する。この $\langle F^* \rangle$ の辺 (w_i^*, x_i^*) を含む $\langle F^* \rangle$ の辺による部分グラフの連結成分 $\langle F^* \rangle_j$ 、 $j = 1, \dots, q$ は辺素な路が K 個存在するならば、補題 3 の証明で示したように、それぞれ木である。よっ

て、 (β) を満たすためには、その木の葉にあたる節点も、 $\langle E^* \rangle$ の辺 1 本で他の $\langle F^* \rangle_j$ の辺と接続していなければならない。

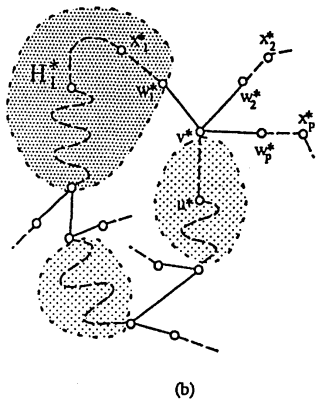
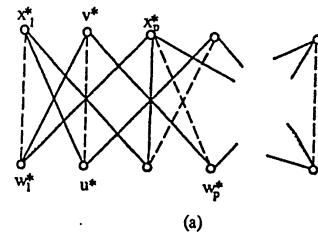
$\langle F^* \rangle_j$ が、全て $\langle F^* \rangle$ の辺 1 本の場合、全ての $\langle F^* \rangle_j$ の各辺 (w_i^*, x_i^*) の両端点は他の 2 つ以上の $\langle F^* \rangle_j$ と $\langle E^* \rangle$ の辺で接続してはならず、また、 $(G+H)^*$ は 2 部グラフなので、 K 個の辺素なカットが存在し (図 6(a) 参照)、(2) で K 個の辺素な路が存在すると判定される。

$\langle F^* \rangle_j$ 中に $\langle F^* \rangle$ の辺 2 本以上からなる木が 1 つでも存在するならば、 $\langle F^* \rangle$ の辺数が $\langle E^* \rangle$ の辺数以上の閉路が構成されることになり (図 6(b) 参照)、補題 2 に矛盾する。

よって、 V_1^* または、 V_2^* のいずれかに、 v^* が K 個存在しなくても、常に (α) 、 (β) に該当しないような節点が存在する。

また、(2) 以下の処理においては、補題 4 より、それを選んで他の構成すべき閉路がとれなくなるような閉路を選んでいるので、そのような閉路を $G+H$ から取り除くことができる (すなわち、 $G+H$ の面に対応する $(G+H)^*$ の節点、および接続する辺を除去できる)。

(3)、(4)、(5) の後、(2) に戻り、 $K-1$ 個の辺素なカットが存在するか調べる。これを定数回 ($O(K)$ 回) 繰り返している。□



— (G+H)* における G に対応する辺 ($\langle E^* \rangle$ の辺)
 - - - (G+H)* における H に対応する辺 ($\langle F^* \rangle$ の辺)

図 6: 補題 5 の図示

次に、計算時間とプロセッサ数の解析を行なう。

(1): まず、 $(G+H)^*$ の各節点が V_1^* 、 V_2^* のいずれに属するかを求める。そのため、まず、 $(G+H)^*$ の全域木を求める [4]。次に、この木の節点番号の最小な節点を根とした、根付き木の深さ優先探索を求める [4]。探索順序の奇数番目に訪れた節点を節

点集合 V_1^* に属するとし、偶数番目に訪れた節点を節点集合 V_2^* に属するとする。 $O(m)$ 個のプロセッサを用いて、 $O(\log^2 n)$ 時間 [4]。

(2) : 各辺にプロセッサを割り当て、1本だけ $\langle F^* \rangle$ の辺を持つ節点 v_k^* , $k = 1, \dots, r$ が、 V_1^* 、 V_2^* にいくつ存在するかを求める。 $O(m)$ 個のプロセッサを用いて、 $O(\log m)$ 時間 (多重辺をもたないグラフならば、 $G+H$ は平面グラフなので m は $O(n)$ となり、 $O(n)$ 個のプロセッサを用いて、 $O(\log n)$ 時間となる)。

K 個の v_k^* が V_1^* 、または V_2^* のいずれかに存在したならば、 v_k^* に接続する $\langle E^* \rangle$ の辺、 (v_k^*, w_i^*) , $i = 1, \dots, q$ に対応する G の辺 (v_k, w_i) を出力する。出力は、 $O(m)$ 個のプロセッサを用いて、 $O(1)$ 時間。

存在しなかった場合、アルゴリズムの (2) の (α)、(β) に該当しない節点を探す。1本の $\langle F^* \rangle$ の辺 (u^*, v^*) をもつ v_k に接続する $\langle E^* \rangle$ の辺 (v_k^*, w_j^*) , $j = 1, \dots, p$ の、 w_j^* に接続する辺 (w_j^*, x_l^*) , $l = 1, \dots, r'$ に、 $\langle F^* \rangle$ の辺が2本以上含まれていないかどうか調べる (但し、 $x_l^* \neq u^*$ は除く)。この条件を満たす v_k^* の中で、節点番号の最小なものを v^* として選ぶ。 $O(m)$ 個のプロセッサを用いて、 $O(\log m)$ 時間。

(3) : (2) 同様、出力は $O(m)$ 個のプロセッサを用いて、 $O(1)$ 時間。

(4) : v^* に接続している各辺 (v^*, u_i^*) , $i = 1, \dots, q'$ にプロセッサを割り当て、各辺 (v^*, u_i^*) を (v^*, u^*) とする (ただし、 u^* は v^* に接続していた $\langle F^* \rangle$ の辺の他方の端点)。 $O(m)$ 個のプロセッサを用いて、 $O(1)$ 時間。

(5) : $O(1)$ 個のプロセッサを用いて、 $O(1)$ 時間。

(2) ~ (5) まで、 $O(n+m)$ 個のプロセッサを用いて、 $O(\log^2 n + \log m)$ 時間。これを、最悪 $O(K)$ 回 (定数回) 繰り返す。

よって、全体として $O(n+m)$ 個のプロセッサを用いて、 $O(\log^2 n + \log m)$ 時間で解を与える。

定理 2 procedure disjoint path construction は、CREW PRAM 上で全体として、 $O(n+m)$ 個のプロセッサを用いて $O(\log^2 n + \log m)$ 時間で解を与える。□

参考文献

- [1] J. A. Bondy and U. S. R. Murty : "Graph Theory with Applications", North-Holland, 1976
(立花俊一、奈良知恵、田澤新成 共訳 : "グラフ理論への入門", 共立出版、(1991)).
- [2] S. Fortune, J. Hopcroft and J. Wyllies : "The directed subgraph homeomorphism problem", Theoretical Computer Science, 10, pp.111-121, 1980.
- [3] A. Frank : "On connectivity properties of Eulerian digraphs", Annals of Discrete Mathematics, 41, North-Holland, pp.179-194, 1991.
- [4] A. Gibbons and W. Rytter : "Efficient Parallel Algorithms", Cambridge University Press, 1988.
- [5] 茨木俊秀 : "グラフの disjoint paths について", 第一回数理計画研究会シンポジウム論文集, pp.3-13, 1989.
- [6] T. Ibaraki and S. Poljak : "Weak three-linking in Eulerian digraphs", SIAM Journal of Discrete Mathematics, 4, 1, pp.99-106, 1991.
- [7] 中山慎一、増山繁 : "2 端子対をもつ有向オイラーグラフの辺素な路問題を解く並列アルゴリズム", 信学技報 COMP91-73, pp.49-57, 1991.
- [8] P. D. Seymour : "On odd cuts and plane multicommodity flows", Proc.London Math.Society, 42, pp.178-192, 1981.