

Znajdowanie skojarzeń na maszynie równoległej

Kamil Lenartowicz

10 grudnia 2008

Spis treści

- 1 Wprowadzenie
- 2 Skojarzenia maksymalne
- 3 Bibliografia

Definicje

Rozważamy grafy nieskierowane $G = (V, E)$ gdzie V to zbiór wierzchołków grafu G a E to zbiór krawędzi grafu G , $|V| = n$, $|E| = m$.

Celem jest znaleźć skojarzenie maksymalne co do zawierania, doskonałe lub najliczniejsze na maszynie PRAM (parallel random access machine).

Przypomnienie: Problemy NC i P-zupełne

Problem należy do klasy NC jeśli istnieje algorytm rozwiązujący ten problem działający w czasie $O(\log(n)^l)$ na n^k procesorów gdzie n to rozmiar problemu a l i k są ustalone.

Przypomnienie: Problemy NC i P-zupełne

Problem należy do klasy NC jeśli istnieje algorytm rozwiązujący ten problem działający w czasie $O(\log(n)^l)$ na n^k procesorów gdzie n to rozmiar problemu a l i k są ustalone.

Oczywiście $NC \subseteq P$

Przypomnienie: Problemy NC i P-zupełne

Problem należy do klasy NC jeśli istnieje algorytm rozwiązujący ten problem działający w czasie $O(\log(n)^l)$ na n^k procesorów gdzie n to rozmiar problemu a l i k są ustalone.

Oczywiście $NC \subseteq P$

Niewiadomo czy $NC = P$

Przypomnienie: Problemy NC i P-zupełne

Problem należy do klasy NC jeśli istnieje algorytm rozwiązujący ten problem działający w czasie $O(\log(n)^l)$ na n^k procesorów gdzie n to rozmiar problemu a l i k są ustalone.

Oczywiście $NC \subseteq P$

Niewiadomo czy $NC = P$

Problemy P-zupełne to problemy należące do P których nie można zrównoleglić.

Skojarzenia maksymalne

Lemat

Jeśli M' jest najliczniejszym skojarzeniem w G a M jest skojarzeniem maksymalnym ze względu na zawieranie to $|M'|/|M| \leq 2$.

Algorytm sekwencyjny

```
M :=  $\emptyset$ 
while  $E \neq \emptyset$  do
begin
  wybierz dowolną krawędź  $e \in E$ 
   $M := M \cup \{e\}$ ;
   $E := E - \text{Incident}(e)$ ;
end
return  $M$ ;
```

Algorytm równoległy

```
 $M := \emptyset$   
while  $E \neq \emptyset$  do  
  begin  
     $X := \text{PartialMatch}(G)$   
     $M := M \cup X$ ;  
     $E := E - \text{Incident}(X)$ ;  
  end  
return  $M$ ;
```

EulerPartition(G)

Podział grafu G na rozłączne krawędziowo ścieżki i cykle: dla każdego wierzchołka zbiór krawędzi z nim incydentnych jest podzielony na pary - tworzona jest funkcja $Next(e)$ która dla każdej krawędzi takiej pary (e, e') przyjmuje wartości $Next(e) = e'$ i $Next(e') = e$.

$Halve(G)$

$Halve(G)$

$D := EulerPartition(G)$

wybierz kierunek dla każdego cyklu i ścieżki;
usuń co drugą krawędź;

$Halve(G)$

$Halve(G)$

$D := EulerPartition(G)$

wybierz kierunek dla każdego cyklu i ścieżki;

usuń co drugą krawędź;

Lemat

*Jeśli $G' = Halve(G)$ to $\lfloor deg_G(v)/2 \rfloor \leq deg_{G'}(v) \leq \lceil deg_G(v)/2 \rceil$.
 $Halve(G)$ może być obliczona w czasie $O(\log(n))$ przy użyciu $O(m)$ procesorów.*

Budowa lasu: $KelsenForest(G)$

```
 $KelsenForest(G)$   
while  $E \neq \emptyset$  do  
     $H := \{v : deg_G(v) = 1\};$   
    for each  $v \in H$  do in parallel  
         $father(v) := \text{sąsiad } v;$   
     $E := E - Incident(H);$   
     $V := V - H;$   
     $G = Halve(G);$   
end
```

$PartialMatch(G)$

$PartialMatch(G)$

$F := KelsenForest(G);$

$M := MaxForestMatch(F);$

end

$PartialMatch(G)$

$PartialMatch(G)$

$F := KelsenForest(G);$

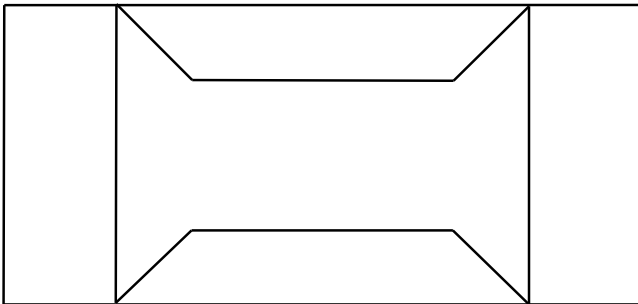
$M := MaxForestMatch(F);$

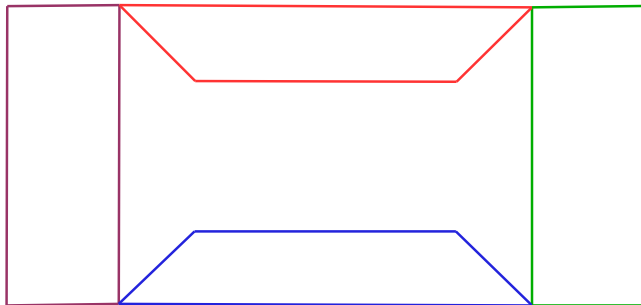
end

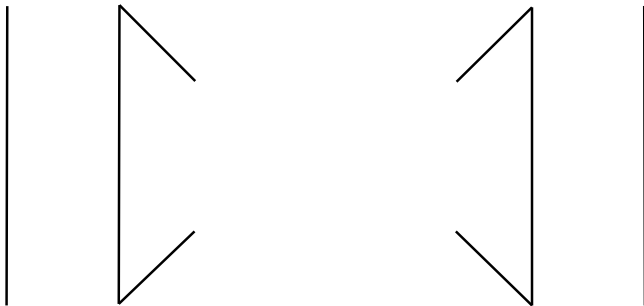
Lemat

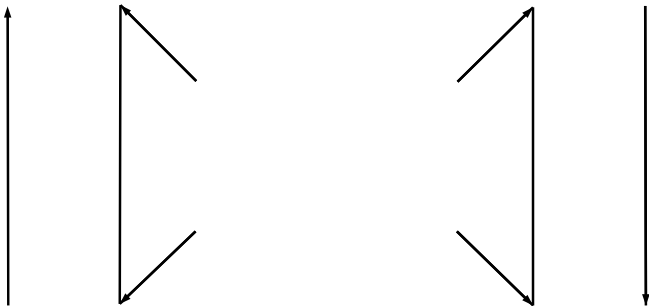
Dla powyższego skojarzenia M zachodzi własność:

$|Incident(M)| \geq |E|/3.$

$PartialMatch(G)$ 

$PartialMatch(G)$ 

$PartialMatch(G)$ 

$PartialMatch(G)$ 

$PartialMatch(G)$

|

|

$PartialMatch(G)$



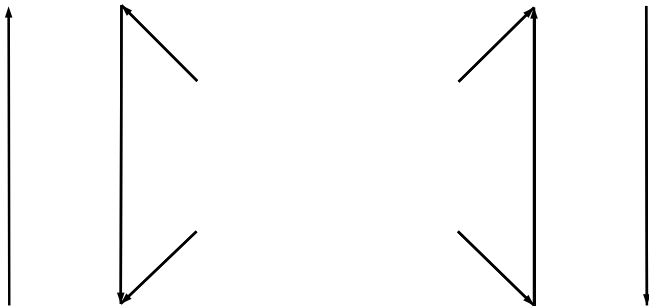
$PartialMatch(G)$

|

|

$PartialMatch(G)$



$PartialMatch(G)$ 

$PartialMatch(G)$



Dowód lematu

Lemat

Dla skojarzenia M będącego wynikiem $\text{PartialMatch}(G)$ zachodzi własność: $|\text{Incident}(M)| \geq |E|/3$.

Twierdzenie

$$\deg_G(w) \geq \frac{1}{2} \sum_{v \in \text{Sons}(w)} \deg_G(v)$$

Dowód lematu

Lemat

Dla skojarzenia M będącego wynikiem $\text{PartialMatch}(G)$ zachodzi własność: $|\text{Incident}(M)| \geq |E|/3$.

Twierdzenie

$$\deg_G(w) \geq \frac{1}{2} \sum_{v \in \text{Sons}(w)} \deg_G(v)$$

Stąd $|\text{Incident}(M)| \geq \frac{1}{2}|E - \text{Incident}(M)|$

Dowód twierdzenia

$d_i(w)$ - stopień w przed $i + 1$ aplikacją $Halve(G)$

$k_i(w)$ - ilość liści doczepionych do w w tym samym momencie

$level(w)$ - ilość potrzebnych operacji $Halve(G)$ aby w stał się liściem

z definicji: $2^{level(w)} \geq \frac{1}{2}deg_G(w)$.

Dowód twierdzenia

$d_i(w)$ - stopień w przed $i + 1$ aplikacją $Halve(G)$

$k_i(w)$ - ilość liści doczepionych do w w tym samym momencie

$level(w)$ - ilość potrzebnych operacji $Halve(G)$ aby w stał się liściem
z definicji: $2^{level(w)} \geq \frac{1}{2}deg_G(w)$.

Wtedy:

$$d_{i+1}(w) \leq (d_i(w) - k_i(w))/2 + \frac{1}{2}$$

dla $0 \leq i < level(w)$

Dowód twierdzenia

$d_i(w)$ - stopień w przed $i + 1$ aplikacją $Halve(G)$

$k_i(w)$ - ilość liści doczepionych do w w tym samym momencie

$level(w)$ - ilość potrzebnych operacji $Halve(G)$ aby w stał się liściem
z definicji: $2^{level(w)} \geq \frac{1}{2}deg_G(w)$.

Wtedy:

$$d_{i+1}(w) \leq (d_i(w) - k_i(w))/2 + \frac{1}{2}$$

dla $0 \leq i < level(w)$

stąd:

$$d_0(w) \geq \sum_{i=0}^{level(w)-1} 2^i k_i(w) = \sum_{v \in Sons(w)} 2^{level(v)}$$

Dowód twierdzenia

$d_i(w)$ - stopień w przed $i + 1$ aplikacją $Halve(G)$

$k_i(w)$ - ilość liści doczepionych do w w tym samym momencie

$level(w)$ - ilość potrzebnych operacji $Halve(G)$ aby w stał się liściem
z definicji: $2^{level(w)} \geq \frac{1}{2}deg_G(w)$.

Wtedy:

$$d_{i+1}(w) \leq (d_i(w) - k_i(w))/2 + \frac{1}{2}$$

dla $0 \leq i < level(w)$

stąd:

$$d_0(w) \geq \sum_{i=0}^{level(w)-1} 2^i k_i(w) = \sum_{v \in Sons(w)} 2^{level(v)}$$

inaczej:

$$deg_G(w) \geq \frac{1}{2} \sum_{v \in Sons(w)} deg_G(v)$$

Skojarzenie maksymalne

Uogólnienie: aby znaleźć skojarzenie w grafie dowolnym podzielić zbiór jego krawędzi tak aby otrzymać $O(\log(n))$ zbiorów rozłącznych w sumie dających E , każdy zbiór zawiera krawędzie dla których pozycja pierwszej od prawej różniąca cyfra w zapisie binarnym etykiet łączonych wierzchołków jest taka sama. Skojarzenie może być wyliczane kolejno dla powstałych grafów.

Skojarzenie maksymalne

Uogólnienie: aby znaleźć skojarzenie w grafie dowolnym podzielić zbiór jego krawędzi tak aby otrzymać $O(\log(n))$ zbiorów rozłącznych w sumie dających E , każdy zbiór zawiera krawędzie dla których pozycja pierwszej od prawej różniąca cyfra w zapisie binarnym etykiet łączonych wierzchołków jest taka sama. Skojarzenie może być wyliczane kolejno dla powstałych grafów.

Twierdzenie

Skojarzenie maksymalne co do zawierania może być wyliczone w czasie $O(\log^4(n))$ przy użyciu $O(m)$ procesorów na maszynie EREW PRAM.

Bibliografia

- Marek Karpinski, Wojciech Rytter (1998) “Fast Parallel Algorithms for Graph Matching Problems: Combinatorial, Algebraic, and Probabilistic Approach”, rozdziały 1,7,9