Skojarzenia Równoległe

Piotr Sankowski

Plan

- algorytm używający $O(n^\omega)$ procesorów.

Plan

- algorytm używający $O(n^\omega)$ procesorów.

RNC

Problem należy do NC^k jeżeli daję się go rozwiązać na wielomianowo wielu procesorach w czasie $O(\log^k n)$.

$\mathrm{NC} = \bigcup_{i=0}^{\infty} \mathrm{NC}^i$.

RNC to NC używające randomizacji. Algorytm z prawdopodobieństwem $< \frac{1}{2}$ może ponieść porażkę.

Problem

Możemy znajdować doskonałe skojarzenia sekwencyjnie w czasie $O(n^\omega)$ —

Mucha i Sankowski (FOCS'04).

Problem

Możemy znajdować doskonałe skojarzenia sekwencyjnie w czasie $O(n^\omega)$ —
Mucha i Sankowski (FOCS'04).

Czy możemy znajdować skojarzenia równoległe w takiej samej pracy?

Problem

Możemy znajdować doskonałe skojarzenia sekwencyjnie w czasie $O(n^\omega)$ —
Mucha i Sankowski (FOCS'04).

Czy możemy znajdować skojarzenia równoległe w takiej samej pracy?

TAK

Plan

- Historia

Plan

- Historia
- Szkielet algorytmu

Plan

- Historia
- Szkielet algorytmu
- Ważone skojarzenia i macierze wielomianów

Plan

- Historia
- Szkielet algorytmu
- Ważone skojarzenia i macierze wielomianów
- Równoległe algorytmy macierzowe

Plan

- Historia
- Szkielet algorytmu
- Ważone skojarzenia i macierze wielomianów
- Równoległe algorytmy macierzowe
- Podsumowanie

Algorytmy sekwencyjnie

- $O(n^3)$ – Edmonds (1965),

Algorytmy sekwencyjnie

- $O(n^3)$ – Edmonds (1965),
- $O(n^{2.5})$ – Micali i Vazirani (1980),

Algorytmy sekwencyjnie

- $O(n^3)$ – Edmonds (1965),
- $O(n^{2.5})$ – Micali i Vazirani (1980),
- $O(n^{\omega+1})$ – Rabin i Vazirani (1989),

Algorytmy sekwencyjnie

- $O(n^3)$ – Edmonds (1965),
- $O(n^{2.5})$ – Micali i Vazirani (1980),
- $O(n^{\omega+1})$ – Rabin i Vazirani (1989),
- $O(n^{2.5})$ – Blum (1990),

Algorytmy sekwencyjnie

- $O(n^3)$ – Edmonds (1965),
- $O(n^{2.5})$ – Micali i Vazirani (1980),
- $O(n^{\omega+1})$ – Rabin i Vazirani (1989),
- $O(n^{2.5})$ – Blum (1990),
- $O(n^{2.5} \log n)$ – Gabow i Tarjan (1991),

Algorytmy sekwencyjnie

- $O(n^3)$ – Edmonds (1965),
- $O(n^{2.5})$ – Micali i Vazirani (1980),
- $O(n^{\omega+1})$ – Rabin i Vazirani (1989),
- $O(n^{2.5})$ – Blum (1990),
- $O(n^{2.5} \log n)$ – Gabow i Tarjan (1991),
- $O(n^{\omega})$ – Mucha i Sankowski (2004).

Algorytmy sekwencyjnie

- $O(n^3)$ – Edmonds (1965),
- $O(n^{2.5})$ – Micali i Vazirani (1980),
- $O(n^{\omega+1})$ – Rabin i Vazirani (1989),
- $O(n^{2.5})$ – Blum (1990),
- $O(n^{2.5} \log n)$ – Gabow i Tarjan (1991),
- $O(n^{\omega})$ – Mucha i Sankowski (2004).

$\omega < 2.376$ – Coppersmith i Winograd (1987).

Algorytmy równoległe

- $O(n^{\omega+4})$, RNC^3 – Karp, Upfal i Wigderson (1986),

Algorytmy równoległe

- $O(n^{\omega+4})$, RNC^3 – Karp, Upfal i Wigderson (1986),
- $O(n^{\omega+3})$, RNC^2 – Mulmuley, Vazirani i Vazirani (1987),

Algorytmy równoległe

- $O(n^{\omega+4})$, RNC^3 – Karp, Upfal i Wigderson (1986),
- $O(n^{\omega+3})$, RNC^2 – Mulmuley, Vazirani i Vazirani (1987),
- $O(n^{\omega+1})$, RNC^3 – Gail i Pan (1988),

Algorytmy równoległe

- $O(n^{\omega+4})$, RNC^3 – Karp, Upfal i Wigderson (1986),
- $O(n^{\omega+3})$, RNC^2 – Mulmuley, Vazirani i Vazirani (1987),
- $O(n^{\omega+1})$, RNC^3 – Gail i Pan (1988),
- $O(n^{\omega})$, RNC^5 – ta prezentacja.

Algorytmy równoległe

- $O(n^{\omega+4})$, RNC^3 – Karp, Upfal i Wigderson (1986),
- $O(n^{\omega+3})$, RNC^2 – Mulmuley, Vazirani i Vazirani (1987),
- $O(n^{\omega+1})$, RNC^3 – Gail i Pan (1988),
- $O(n^{\omega})$, RNC^5 – ta prezentacja.

Jako produkt uboczny otrzymamy nowy sekwencyjny algorytm działający w czasie $\tilde{O}(n^{\omega})$.

Ważone skojarzenia

Rozważmy pełny graf $G = (V, E)$.

W *problemie najcięższych ważonych skojarzeń*:

- każda krawędź $uv \in E$ ma daną wagę c_{uv},
- szukamy doskonałego skojarzenia M o największej wadze $w(M) = \sum_{e \in M} c_e$.

Ważone skojarzenia

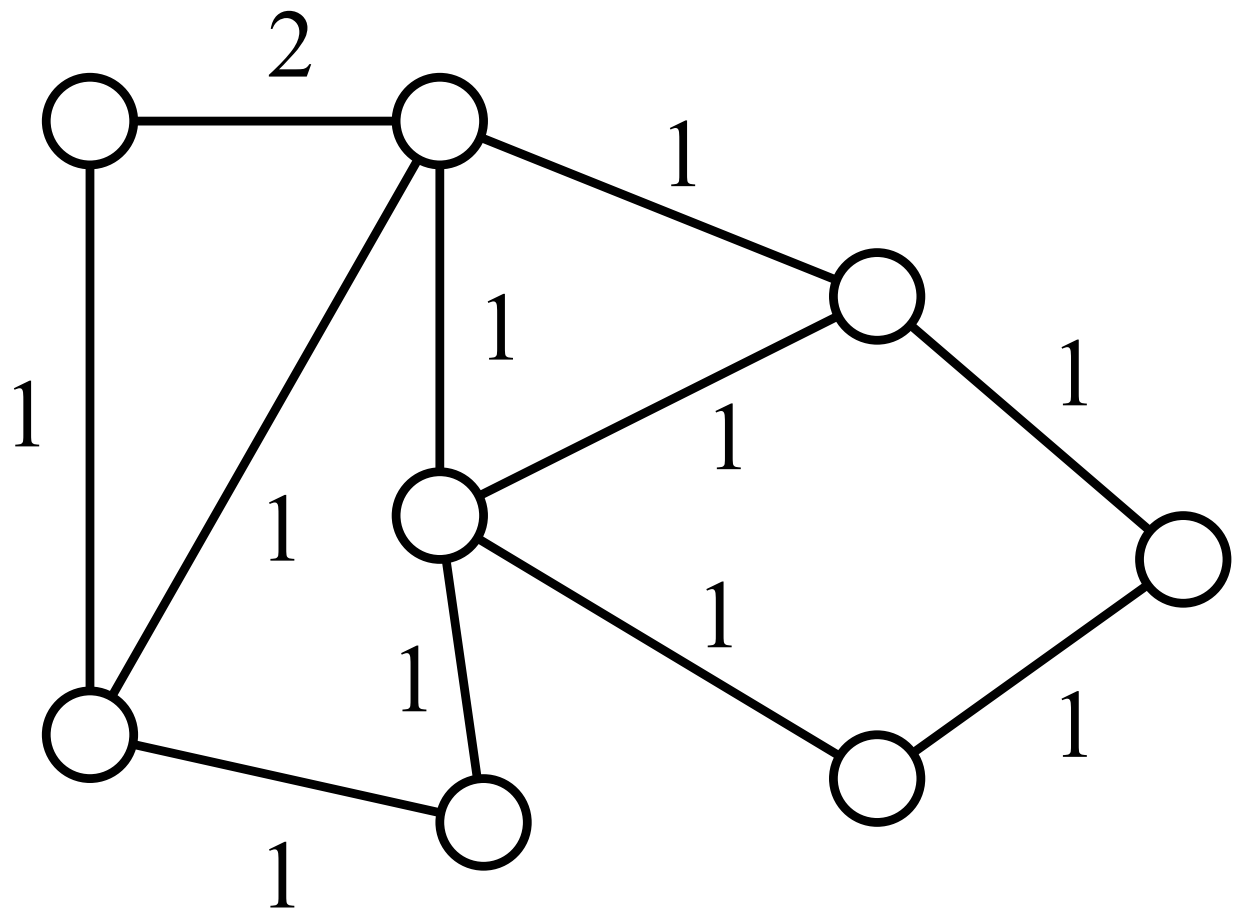
Rozważmy pełny graf $G = (V, E)$.

W *problemie najcięższych ważonych skojarzeń*:

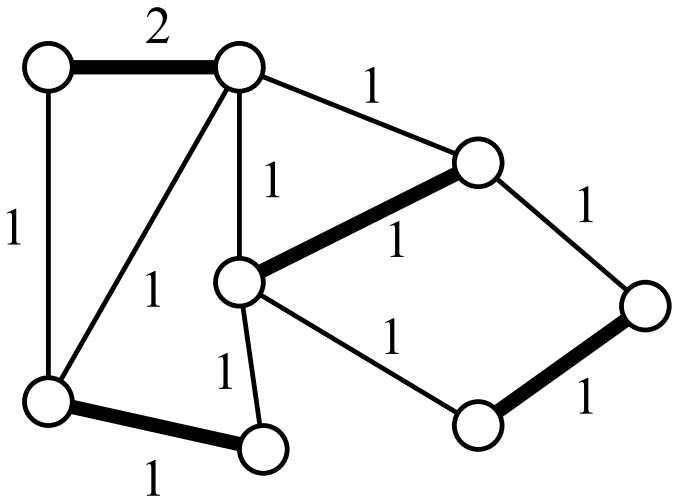
- każda krawędź $uv \in E$ ma daną wagę c_{uv},
- szukamy doskonałego skojarzenia M o największej wadze $w(M) = \sum_{e \in M} c_e$.

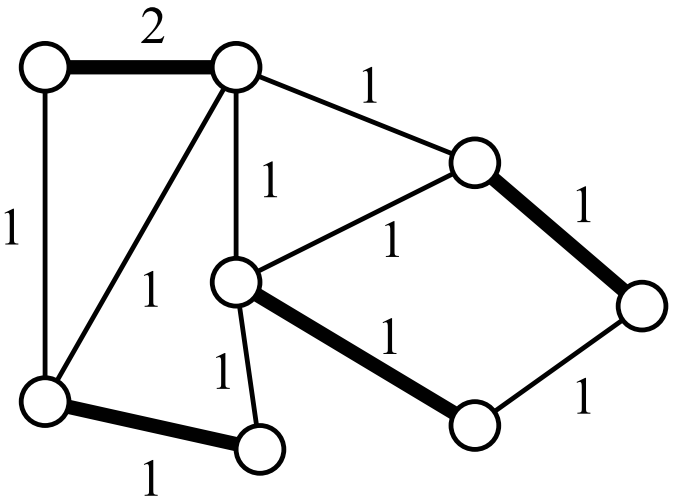
Dozwolona krawędź to będzie krawędź należąca do jakiegoś najcięższego skojarzenia.

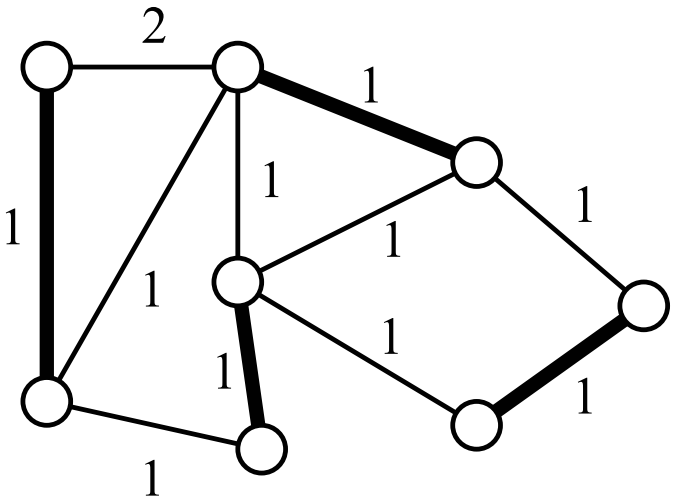
Dozwolone krawędzie



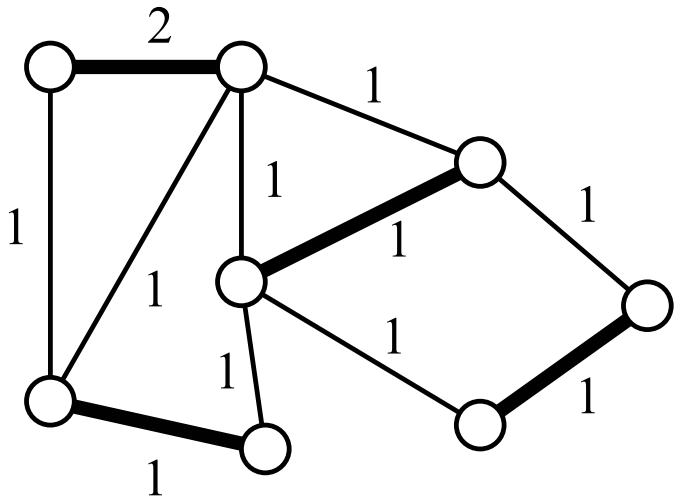
Dozwolone krawędzie

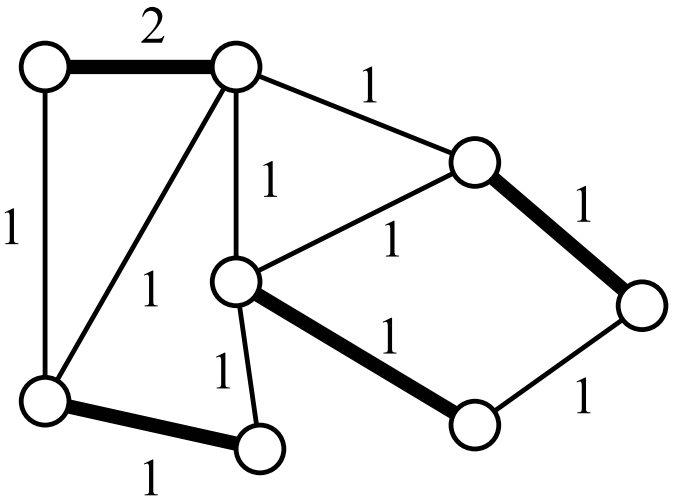


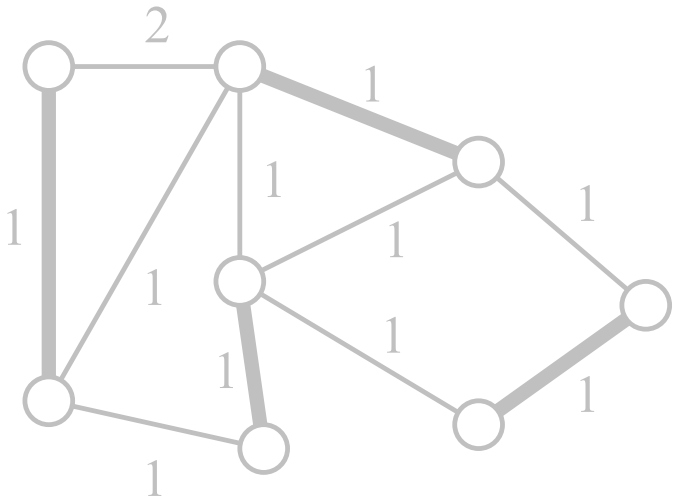




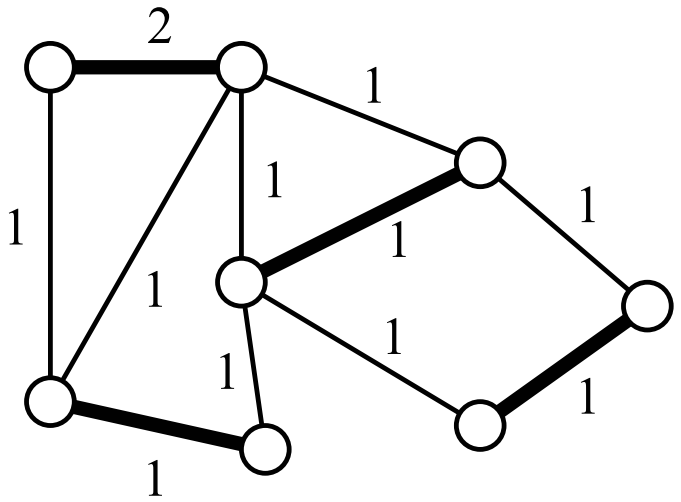
Dozwolone krawędzie

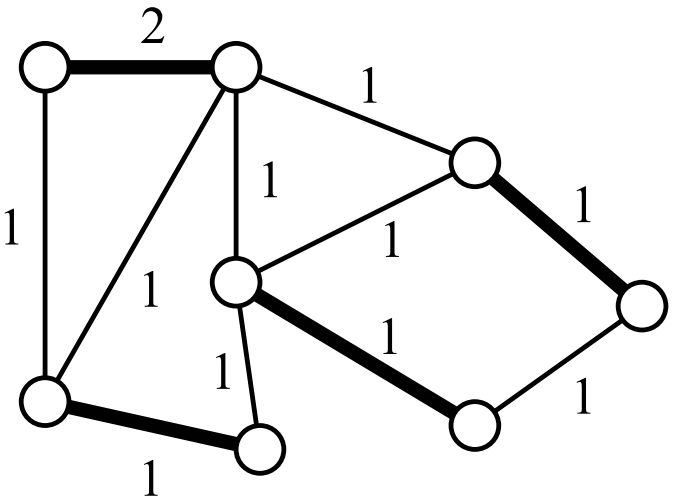


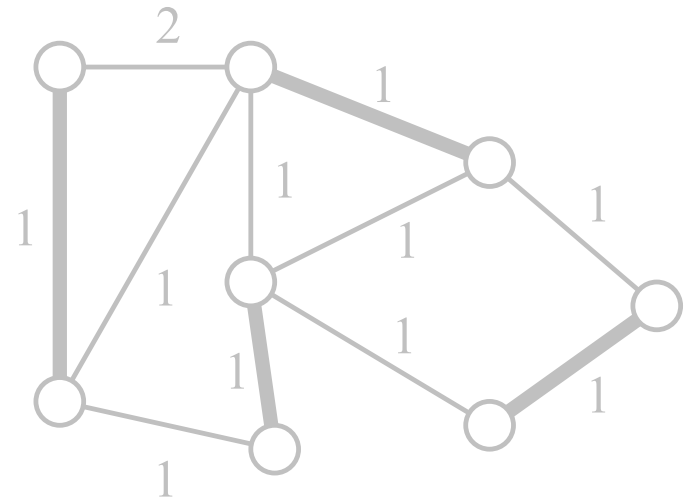


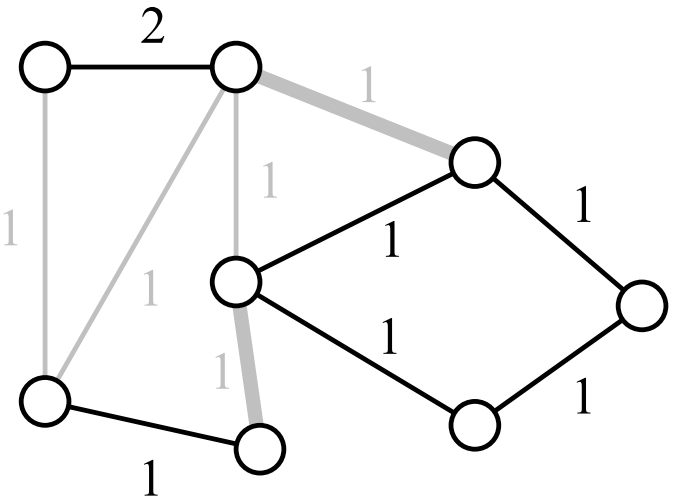


Dozwolone krawędzie









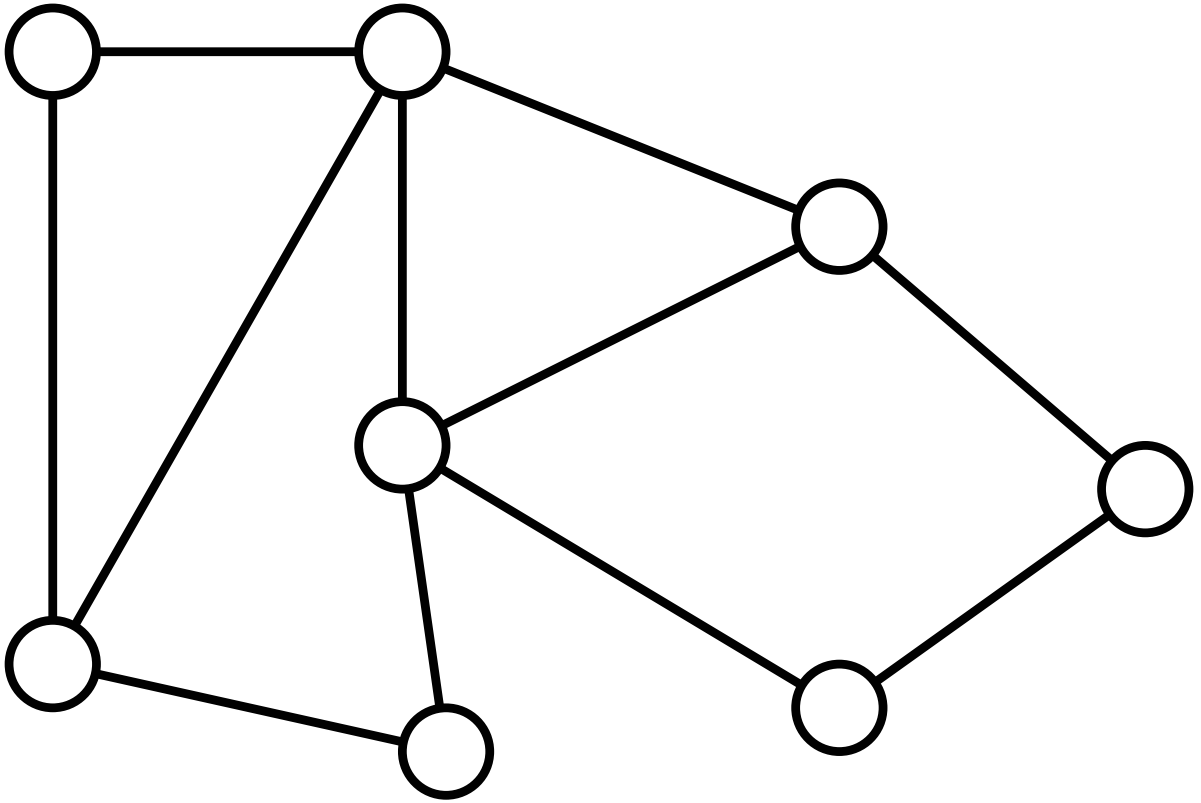
Pomysł

FIND-PERFECT-MATCHING(V,E):

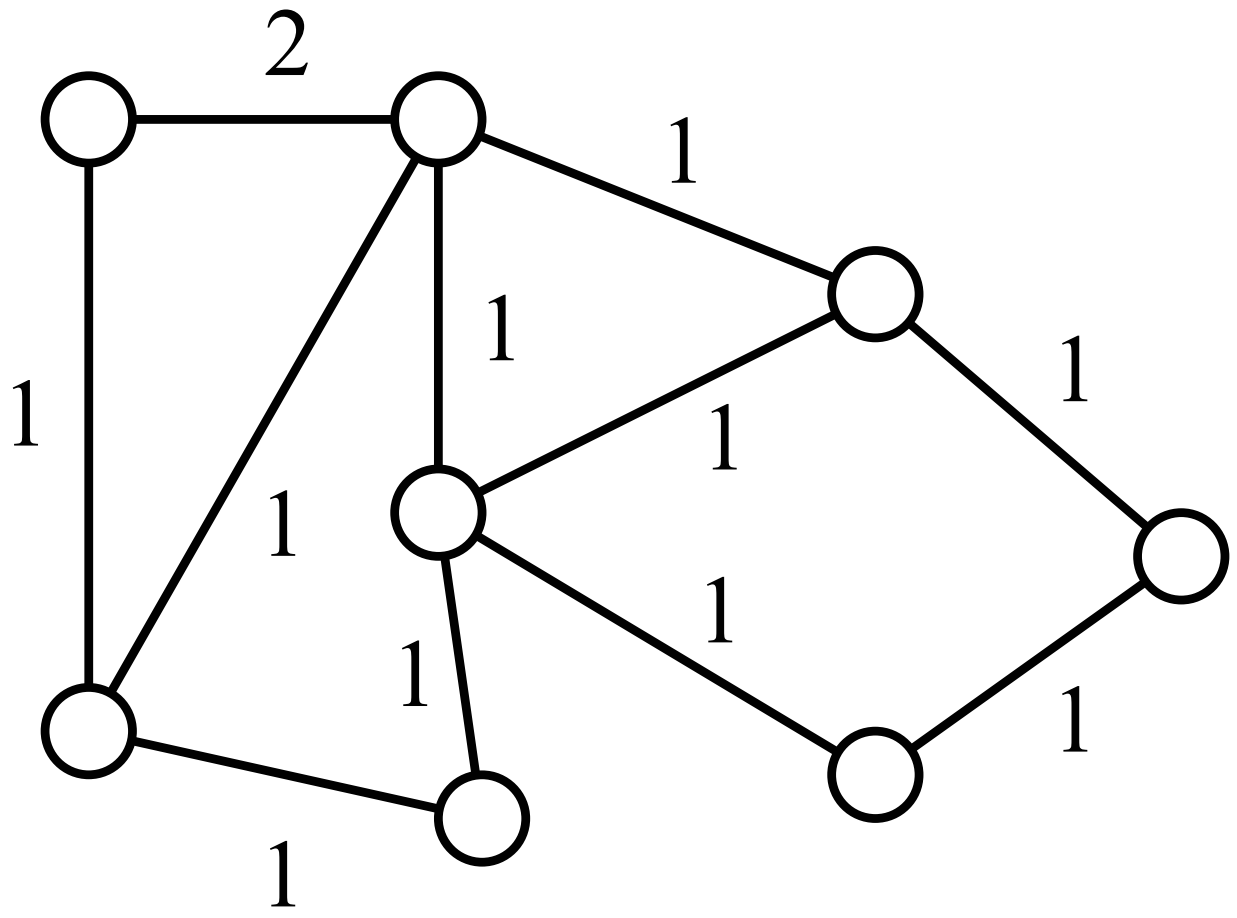
- **for** $i := 1$ **to** $24\lceil \log(n) \rceil$ **do**
  - wygeneruj losowe wagi 0,1 dla krawędzi (na rysunkach jest 1 bądź 2),
  - znajdź zbiór dozwolonych krawędzi E_A,
  - usuń krawędzie nie należące do E_A.
- **if** $|E| = \frac{n}{2}$ **then**
  - $M := E$,
- **else**
  - nie udało się.

Jeżeli algorytm się powiedzie to M jest doskonałym skojarzeniem.

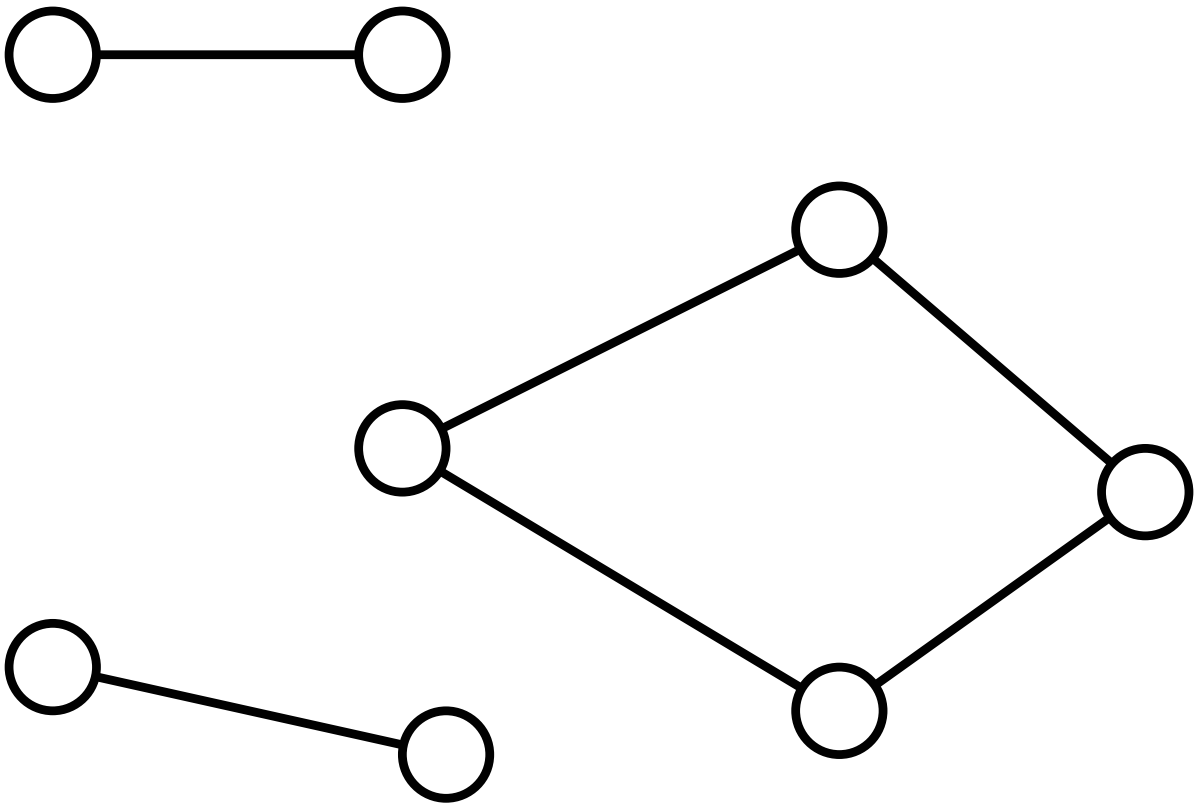
Algorytm



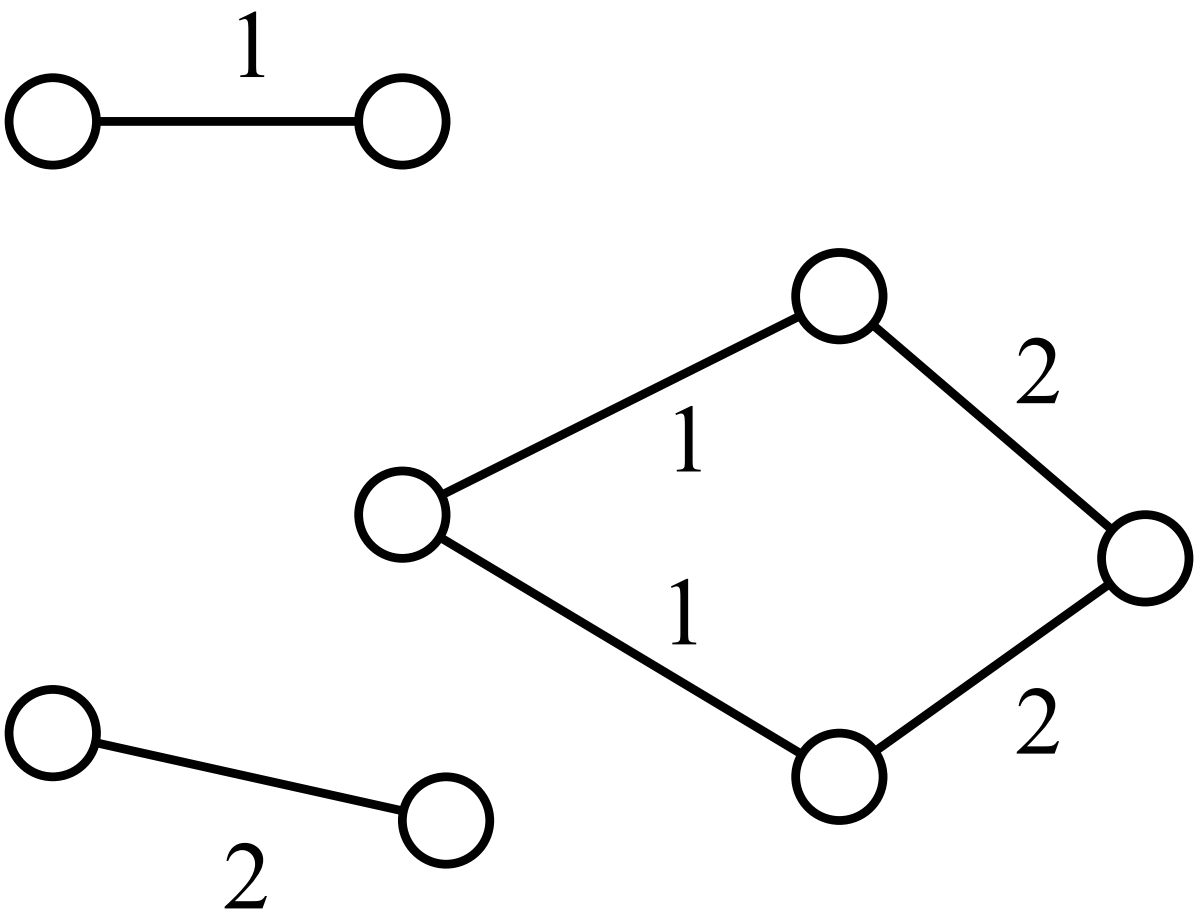
Algorytm



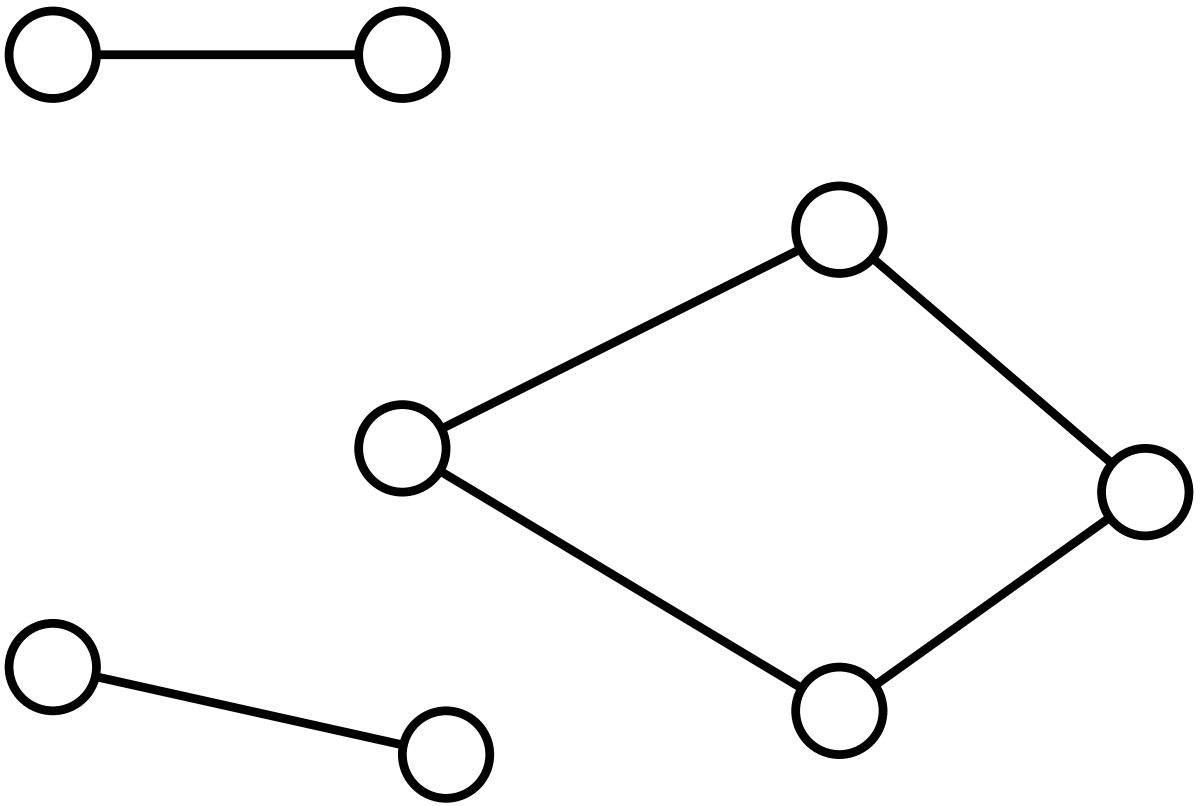
Algorytm



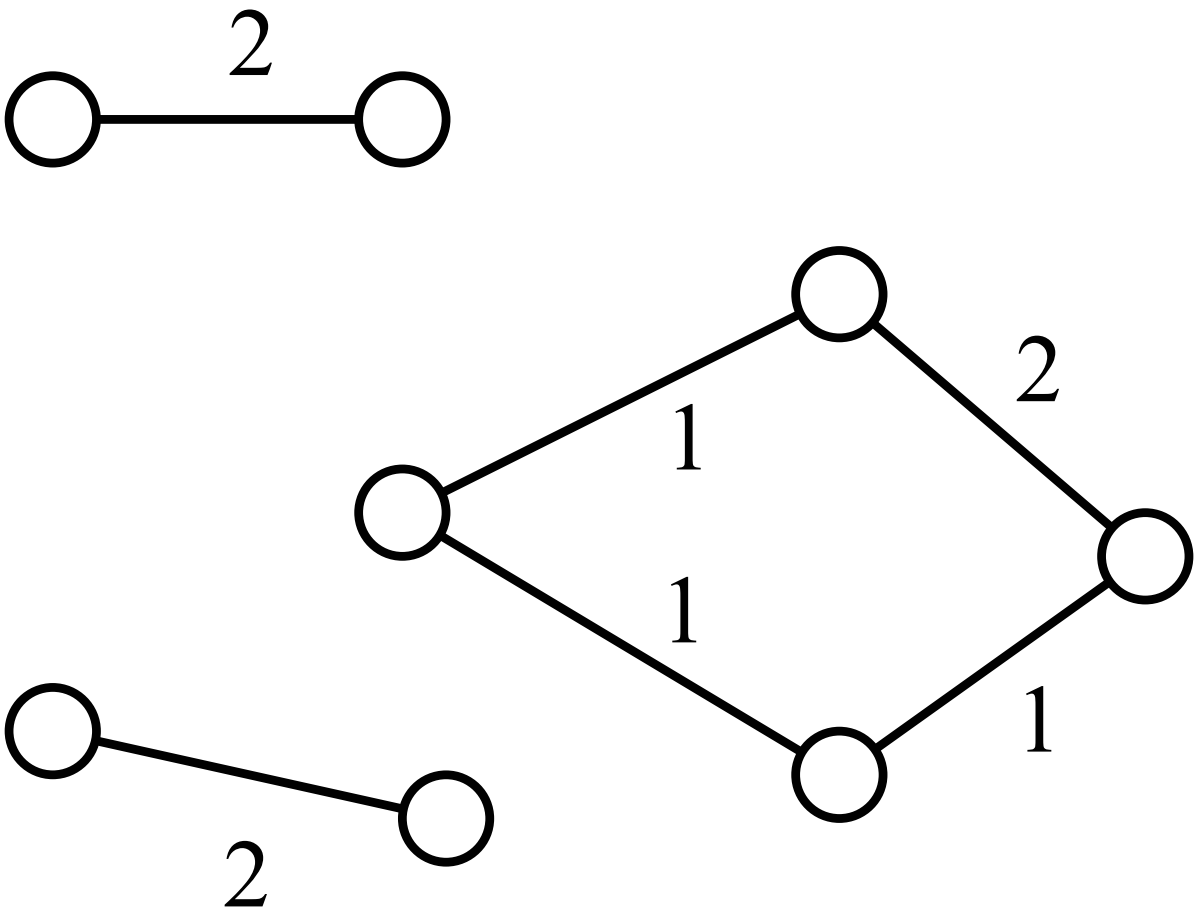
Algorytm



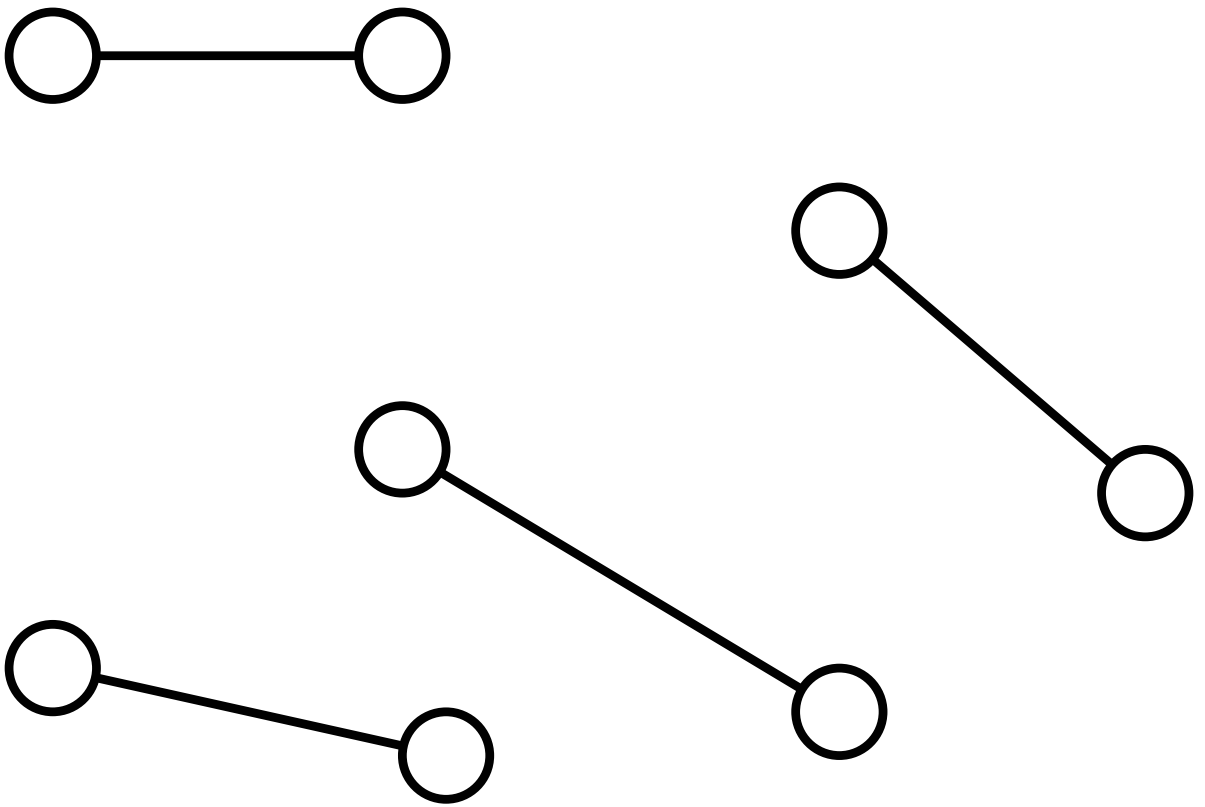
Algorytm



Algorytm



Algorytm



Musimy pokazać

Musimy pokazać:

- że $24\lceil \log(n) \rceil$ iteracji wystarcza,

Musimy pokazać

Musimy pokazać:

- że $24\lceil \log(n) \rceil$ iteracji wystarcza,
- jak wyznaczyć dozwolone krawędzie w RNC na $O(n^{\omega})$ procesorach.

Liczba iteracji

Lemma 1 *Rozważmy wierzchołek* $v \in G$, *jeżeli* $\deg(v) > 1$ *to prawdopodobieństwo, że stopień* v *zmniejsza się o połowę w każdej iteracji jest* $> \frac{1}{4}$.

Liczba iteracji

Lemma 1 *Rozważmy wierzchołek* $v \in G$, *jeżeli* $\deg(v) > 1$ *to prawdopodobieństwo, że stopień* v *zmniejsza się o połowę w każdej iteracji jest* $> \frac{1}{4}$.

Waga krawędzi określa, czy będzie ona dozwolona czy nie.

Liczba iteracji

Lemma 1 *Rozważmy wierzchołek $v \in G$, jeżeli* $\deg(v) > 1$ *to prawdopodobieństwo, że stopień v zmniejsza się o połowę w każdej iteracji jest* $> \frac{1}{4}$.

Waga krawędzi określa, czy będzie ona dozwolona czy nie.

Wagi krawędzi wybierane są niezależnie.

Liczba iteracji

Lemma 2 *Rozważmy wierzchołek* $v \in G$, *jeżeli* $\deg(v) > 1$ *to prawdopodobieństwo, że stopień* v *zmniejsza się o połowę w każdej iteracji jest* $> \frac{1}{4}$.

Liczba iteracji

Lemma 2 *Rozważmy wierzchołek $v \in G$, jeżeli* $\deg(v) > 1$ *to prawdopodobieństwo, że stopień v zmniejsza się o połowę w każdej iteracji jest* $> \frac{1}{4}$.

- Rozważmy krawędzie E_v sąsiednie do v.

Liczba iteracji

Lemma 2 *Rozważmy wierzchołek $v \in G$, jeżeli* $\deg(v) > 1$ *to prawdopodobieństwo, że stopień v zmniejsza się o połowę w każdej iteracji jest* $> \frac{1}{4}$.

- Rozważmy krawędzie E_v sąsiednie do v.
- Krawędź $e \in E_v$ należy do E_A jeżeli należy do pewnego najcięższego doskonałego skojarzenia w G.

Liczba iteracji

- Ustalmy wagi w dla $uv \in G - E_v$.

Liczba iteracji

- Ustalmy wagi w dla $uv \in G - E_v$.
- Niech $W(uv)$ będzie wagą najcięższego doskonałego skojarzenia w $G - u - v$.

Liczba iteracji

- Ustalmy wagi w dla $uv \in G - E_v$.
- Niech $W(uv)$ będzie wagą najcięższego doskonałego skojarzenia w $G - u - v$.
- Waga ta jest wyznaczona przez ustalone wagi w.

Liczba iteracji

- Niech

$$m = \max_{e \in E_v} W(e).$$

Liczba iteracji

- Niech

$$m = \max_{e \in E_v} W(e).$$

- Niech

$$E'_v = \{e : e \in E_v \text{ and } W(e) = m\}.$$

Liczba iteracji

- Niech

$$m = \max_{e \in E_v} W(e).$$

- Niech

$$E'_v = \{e : e \in E_v \text{ and } W(e) = m\}.$$

- Wygeneruj wagi dla krawędzi w E_v.

Liczba iteracji

- Niech

$$m = \max_{e \in E_v} W(e).$$

- Niech

$$E'_v = \{e : e \in E_v \text{ and } W(e) = m\}.$$

- Wygeneruj wagi dla krawędzi w E_v.

Jeżeli jakakolwiek waga krawędzi w E'_v jest 1, wtedy z E'_v nie usuwane są krawędzie które mają wagę 1.

Liczba iteracji

- Załóżmy, że $|E_v| > 1$.

Liczba iteracji

- Załóżmy, że $|E_v| > 1$.
- Jeżeli $|E'_v| = 1$ to wtedy prawdopodobieństwo że stopień v się zmniejszy dwukrotnie jest $\geq \frac{1}{2}$.

Liczba iteracji

- Załóżmy, że $|E_v| > 1$.
- Jeżeli $|E'_v| = 1$ to wtedy prawdopodobieństwo że stopień v się zmniejszy dwukrotnie jest $\geq \frac{1}{2}$.
- Załóżmy w takim razie, że $|E'_v| \geq 2$.

Liczba iteracji

Jest $2^{|E'_v|}$ możliwości wybrania wag dla E'_v, wśród których

Liczba iteracji

Jest $2^{|E'_v|}$ możliwości wybrania wag dla E'_v, wśród których

- jedna jest taka, że wszystkie wagi są 0,

Liczba iteracji

Jest $2^{|E'_v|}$ możliwości wybrania wag dla E'_v, wśród których

- jedna jest taka, że wszystkie wagi są 0,
- w $2^{|E'_v|-1}$ więcej niż połowa krawędzi ma wagę 1,

Liczba iteracji

Jest $2^{|E'_v|}$ możliwości wybrania wag dla E'_v, wśród których

- jedna jest taka, że wszystkie wagi są 0,
- w $2^{|E'_v|-1}$ więcej niż połowa krawędzi ma wagę 1,
- w $2^{|E'_v|-1} - 1$ mniej niż połowa krawędzi ma wagę 1 i jest co najmniej jedna z wagą 1.

W ostatnim przypadku stopień v zmniejsza się co najmniej dwukrotnie.

Liczba iteracji

Jest $2^{|E'_v|}$ możliwości wybrania wag dla E'_v, wśród których

- jedna jest taka, że wszystkie wagi są 0,
- w $2^{|E'_v|-1}$ więcej niż połowa krawędzi ma wagę 1,
- w $2^{|E'_v|-1} - 1$ mniej niż połowa krawędzi ma wagę 1 i jest co najmniej jedna z wagą 1.

W ostatnim przypadku stopień v zmniejsza się co najmniej dwukrotnie.

Liczba iteracji

Prawdopodobieństwo, że stopień zmniejszy się o co najmniej połowę wynosi

$$\frac{2^{|E'_v|-1}-1}{2^{|E'_v|}} = \frac{1}{2} - \frac{1}{2^{|E'_v|}} \geq \frac{1}{4},$$

ponieważ $|E'_v| > 1$.

Liczba iteracji

Lemma 3 *Rozważmy wierzchołek v w G. Prawdopodobieństwo, że stopień v nie jest równy 1 po $24\lceil \log n \rceil$ iteracji algorytmu jest $O(\frac{1}{n^2})$.*

Liczba iteracji

Lemma 3 *Rozważmy wierzchołek v w G. Prawdopodobieństwo, że stopień v nie jest równy 1 po $24\lceil \log n \rceil$ iteracji algorytmu jest $O(\frac{1}{n^2})$.*

- Stopień v nie jest równy 1 jeżeli jego stopień zmniejszany był dwukrotnie mniej niż $\lceil \log n \rceil$ razy.

Liczba iteracji

Lemma 3 *Rozważmy wierzchołek v w G. Prawdopodobieństwo, że stopień v nie jest równy 1 po $24\lceil \log n \rceil$ iteracji algorytmu jest $O(\frac{1}{n^2})$.*

- Stopień v nie jest równy 1 jeżeli jego stopień zmniejszany był dwukrotnie mniej niż $\lceil \log n \rceil$ razy.
- Każda iteracja jest niezależna, więc mamy tutaj próby Bernoulli'ego.

Liczba iteracji

Prawdopodobieństwo $\Pr(k,s)$, że w k próbach otrzymamy mniej niż s sukcesów jest ograniczone przez nierówność Chernoffa

$$\Pr(k,s) < \exp\left(-\frac{(pk-s)^2}{2pk}\right),$$

gdzie p to prawdopodobieństwo sukcesu.

Liczba iteracji

Stosując tą nierówność do naszego przypadku otrzymujemy

$$\Pr(24\lceil\log n\rceil, \lceil\log n\rceil) <$$

$$< \exp\left(-\frac{(\frac{1}{4}24\lceil\log n\rceil - \lceil\log n\rceil)^2}{2\frac{1}{4}24\lceil\log n\rceil}\right) =$$

$$= \exp\left(-\frac{25}{12}\lceil\log n\rceil\right) < \exp\left(\log n - 1\right)^{-2} =$$

$$= \left(\frac{n}{e}\right)^{-2}.$$

Liczba iteracji

Lemma 4 *Prawdopodobieństwo, że algorytm poniesie porażkę jest* $O(\frac{1}{n})$.

Liczba iteracji

Lemma 4 *Prawdopodobieństwo, że algorytm poniesie porażkę jest* $O(\frac{1}{n})$.

- Wierzchołek po $24\lceil \log n \rceil$ iteracjach ma stopień większy niż 1 z prawdopodobieństwem $O(\frac{1}{n^2})$.

Liczba iteracji

Lemma 4 *Prawdopodobieństwo, że algorytm poniesie porażkę jest* $O(\frac{1}{n})$.

- Wierzchołek po $24\lceil \log n \rceil$ iteracjach ma stopień większy niż 1 z prawdopodobieństwem $O(\frac{1}{n^2})$.
- Czyli wśród n wierzchołków jest taki o stopniu wyższym niż 1 z prawdopodobieństwem $O(\frac{1}{n})$.

Liczba iteracji

Lemma 4 *Prawdopodobieństwo, że algorytm poniesie porażkę jest* $O(\frac{1}{n})$.

- Wierzchołek po $24\lceil \log n \rceil$ iteracjach ma stopień większy niż 1 z prawdopodobieństwem $O(\frac{1}{n^2})$.
- Czyli wśród n wierzchołków jest taki o stopniu wyższym niż 1 z prawdopodobieństwem $O(\frac{1}{n})$.

Musimy więc znajdować dozwolone krawędzie $O(\log n)$ razy.

Maximum Matching Weight

Zdefiniujmy dla grafu G oraz wag krawędzi $w : E \to N$ macierz

$$\tilde{A}(G, w, x)_{i,j} = \begin{cases} z_{i,j} x^{w(ij)} & \text{jeżeli } (i,j) \in E \text{ i } i < j, \\ -z_{j,i} x^{w(ij)} & \text{jeżeli } (i,j) \in E \text{ i } i > j, \\ 0 & \text{wpp.} \end{cases}$$

Maximum Matching Weight

Zdefiniujmy dla grafu G oraz wag krawędzi $w : E \to N$ macierz

$$\tilde{A}(G,w,x)_{i,j} = \begin{cases} z_{i,j}x^{w(ij)} & \text{jeżeli } (i,j) \in E \text{ i } i < j, \\ -z_{j,i}x^{w(ij)} & \text{jeżeli } (i,j) \in E \text{ i } i > j, \\ 0 & \text{wpp.} \end{cases}$$

Lemma 5 (Karp, Upfal i Widgerson '86) *Stopień x w $\det(\tilde{A}(G,w,x))$ jest równy dwukrotnej wadze doskonałego skojarzenia M w G,*

$$\deg(\det(\tilde{A}(G,w,x))) = 2w(M).$$

Macierz minorów

Macierz minorów dla macierzy A jest zdefiniowana jako

$$\mathrm{adj}(A)_{i,j} = (-1)^{i+j} \det(A^{j,i}),$$

gdzie $A^{j,i}$ to macierz z usuniętym j-tym wierszem i i-tą kolumną.

Jeżeli A jest macierzą sąsiedztwa grafu G to $A^{j,i}$ koduje graf G z usuniętymi krawędziami wychodzącymi z j i wchodzącymi do i.

Dozwolone krawędzie

Dla krawędzi $uv \in E$ niech $M(uv)$ będzie najcięższym doskonałym skojarzeniem w $G - \{u, v\}$.

Lemma 6 *Niech $G = (V, E)$ będzie grafem z wagami krawędzi danymi przez w, wtedy*

$$\deg(\mathrm{adj}(\tilde{A}(G, w, x))_{i,j}) = w(M) + w(M(ij)).$$

Dozwolone krawędzie

Powyższy lemat pozwala nam scharakteryzować krawędzie tworzące najcięższe doskonałe skojarzenia w następujący sposób.

Wniosek 1 *Mając dany graf* $G = (V, E)$ *oraz wagi krawędzi* $w : E \to N$. *Krawędź* $ij \in E$ *jest dozwolona wttw*

$$\deg(\det(\tilde{A}(G, w, x))) =$$

$$= \deg(\mathrm{adj}(\tilde{A}(G, w, x))_{i,j}) + \deg(\tilde{A}(G, w, x)_{i,j}).$$

Dozwolone krawędzie

Powyższy lemat pozwala nam scharakteryzować krawędzie tworzące najcięższe doskonałe skojarzenia w następujący sposób.

Wniosek 2 *Mając dany graf* $G = (V, E)$ *oraz wagi krawędzi* $w : E \to N$. *Krawędź* $ij \in E$ *jest dozwolona wttw*

$$\deg(\det(\tilde{A}(G, w, x))) =$$

$$= \deg(\mathrm{adj}(\tilde{A}(G, w, x))_{i,j}) + \deg(\tilde{A}(G, w, x)_{i,j}).$$

$$w(M) = w(M(ij)) + w(ij).$$

Zippel-Schwartz

Lemma 7 *Jeżeli $p(x_1, \ldots, x_m)$ jest niezerowym wielomianem stopnia d o współczynnikach z ciała S, to wtedy prawdopodobieństwo, że wartość p wynosi 0 dla losowego wartościowania $(s_1, s_2, \ldots, s_m) \in S^m$ wynosi co najwyżej $d/|S|$. Takie zdarzenie nazywamy fałszywym zerem.*

Zippel-Schwartz

Lemma 7 *Jeżeli $p(x_1, \ldots, x_m)$ jest niezerowym wielomianem stopnia d o współczynnikach z ciała S, to wtedy prawdopodobieństwo, że wartość p wynosi 0 dla losowego wartościowania $(s_1, s_2, \ldots, s_m) \in S^m$ wynosi co najwyżej $d/|S|$. Takie zdarzenie nazywamy fałszywym zerem.*

Jeżeli wielomian stopnia n jest obliczony dla losowych wartości modulo pewna liczba pierwsza p długości $(1 + c) \log n$, to prawdopodobieństwo fałszywego zera wynosi co najwyżej $\frac{1}{n^c}$, dla każdego $c > 0$.

Pochodne

W celu wyznaczenia stopnie wielomianu p możemy policzyć jego wszystkie pochodne w pewnym punkcie t

$$p^{(k)} = \frac{d^k}{dx^k} p \Big|_{x=t}.$$

Pochodne

W celu wyznaczenia stopnie wielomianu p możemy policzyć jego wszystkie pochodne w pewnym punkcie t

$$p^{(k)} = \frac{d^k}{dx^k} p \Big|_{x=t}.$$

Najwyższy niezerowy $p^{(k)}$ daje stopień p.

Obliczanie wagi skojarzenia

- wybierz liczbę pierwszą p długości $\Theta(\log n)$,

Obliczanie wagi skojarzenia

- wybierz liczbę pierwszą p długości $\Theta(\log n)$,
- podstaw losowe wartości ze zbioru $\{1, \ldots, p\}$ za zmienne w $\tilde{A}(G, w, x) \to A$,

Obliczanie wagi skojarzenia

- wybierz liczbę pierwszą p długości $\Theta(\log n)$,
- podstaw losowe wartości ze zbioru $\{1, \ldots, p\}$ za zmienne w $\tilde{A}(G, w, x) \to A$,
- wybierz losową liczbę $t \in \{1, \ldots, p\}$,

Obliczanie wagi skojarzenia

- wybierz liczbę pierwszą p długości $\Theta(\log n)$,
- podstaw losowe wartości ze zbioru $\{1, \ldots, p\}$ za zmienne w $\tilde{A}(G, w, x) \to A$,
- wybierz losową liczbę $t \in \{1, \ldots, p\}$,
- gdy $\det(A|_{x=t}) = 0$ to G nie ma doskonałego skojarzenia,

Obliczanie wagi skojarzenia

- wybierz liczbę pierwszą p długości $\Theta(\log n)$,
- podstaw losowe wartości ze zbioru $\{1, \ldots, p\}$ za zmienne w $\tilde{A}(G, w, x) \to A$,
- wybierz losową liczbę $t \in \{1, \ldots, p\}$,
- gdy $\det(A|_{x=t}) = 0$ to G nie ma doskonałego skojarzenia,
- policz $\frac{d^k}{dx^k} \det(A)\Big|_{x=t}$, dla wszystkich $0 \leq k \leq n$,

Obliczanie wagi skojarzenia

- wybierz liczbę pierwszą p długości $\Theta(\log n)$,
- podstaw losowe wartości ze zbioru $\{1, \ldots, p\}$ za zmienne w $\tilde{A}(G, w, x) \to A$,
- wybierz losową liczbę $t \in \{1, \ldots, p\}$,
- gdy $\det(A|_{x=t}) = 0$ to G nie ma doskonałego skojarzenia,
- policz $\frac{d^k}{dx^k} \det(A)\Big|_{x=t}$, dla wszystkich $0 \leq k \leq n$,
- znajdź najwyższe k takie, że $\frac{d^k}{dx^k} \det(A)\Big|_{x=t} \neq 0$,

Obliczanie wagi skojarzenia

- wybierz liczbę pierwszą p długości $\Theta(\log n)$,
- podstaw losowe wartości ze zbioru $\{1, \ldots, p\}$ za zmienne w $\tilde{A}(G, w, x) \to A$,
- wybierz losową liczbę $t \in \{1, \ldots, p\}$,
- gdy $\det(A|_{x=t}) = 0$ to G nie ma doskonałego skojarzenia,
- policz $\frac{d^k}{dx^k} \det(A)\Big|_{x=t}$, dla wszystkich $0 \leq k \leq n$,
- znajdź najwyższe k takie, że $\frac{d^k}{dx^k} \det(A)\Big|_{x=t} \neq 0$,
- z dużym pr. k jest równe podwojonej wadze najcięższego doskonałego skojarzenia w G.

Dozwolone krawędzie

- policz $d = \deg(\det(\tilde{A}(G, w, x)))$,

Dozwolone krawędzie

- policz $d = \deg(\det(\tilde{A}(G, w, x)))$,
- wybierz liczbę pierwszą p długości $\Theta(\log n)$,

Dozwolone krawędzie

- policz $d = \deg(\det(\tilde{A}(G, w, x)))$,
- wybierz liczbę pierwszą p długości $\Theta(\log n)$,
- podstaw losowe wartości ze zbioru $\{1, \ldots, p\}$ za zmienne w $\tilde{A}(G, w, x) \to A$,

Dozwolone krawędzie

- policz $d = \deg(\det(\tilde{A}(G, w, x)))$,
- wybierz liczbę pierwszą p długości $\Theta(\log n)$,
- podstaw losowe wartości ze zbioru $\{1, \ldots, p\}$ za zmienne w $\tilde{A}(G, w, x) \to A$,
- wybierz losową liczbę $t \in \{1, \ldots, p\}$,

Dozwolone krawędzie

- policz $d = \deg(\det(\tilde{A}(G, w, x)))$,
- wybierz liczbę pierwszą p długości $\Theta(\log n)$,
- podstaw losowe wartości ze zbioru $\{1, \ldots, p\}$ za zmienne w $\tilde{A}(G, w, x) \to A$,
- wybierz losową liczbę $t \in \{1, \ldots, p\}$,
- policz $\frac{d^d}{dx^d} \operatorname{adj}(A)\Big|_{x=t}$ oraz $\frac{d^{d-1}}{dx^{d-1}} \operatorname{adj}(A)\Big|_{x=t}$,

Dozwolone krawędzie

- policz $d = \deg(\det(\tilde{A}(G, w, x)))$,
- wybierz liczbę pierwszą p długości $\Theta(\log n)$,
- podstaw losowe wartości ze zbioru $\{1, \ldots, p\}$ za zmienne w $\tilde{A}(G, w, x) \to A$,
- wybierz losową liczbę $t \in \{1, \ldots, p\}$,
- policz $\frac{d^d}{dx^d} \operatorname{adj}(A)\Big|_{x=t}$ oraz $\frac{d^{d-1}}{dx^{d-1}} \operatorname{adj}(A)\Big|_{x=t}$,
- krawędź ij jest dozwolona jeżeli $(\frac{d^{d-w(ij)}}{dx^{d-w(ij)}} \operatorname{adj}(A)\Big|_{x=t})_{i,j} \neq 0$.

Pochodne macierzy

Pokażemy jak obliczyć w RNC na $O(n^\omega)$ procesorach

- $\frac{d^k}{dx^k} \det(A)\Big|_{x=t}$, dla wszystkich k takich, że $0 \leq k \leq n$,

Pochodne macierzy

Pokażemy jak obliczyć w RNC na $O(n^\omega)$ procesorach

- $\frac{d^k}{dx^k} \det(A)\Big|_{x=t}$, dla wszystkich k takich, że $0 \leq k \leq n$,
- $\frac{d^k}{dx^k} \mathrm{adj}(A)\Big|_{x=t}$, dla danego k takiego, że $0 \leq k \leq n$.

Algorytmy macierzowe

Będziemy potrzebować następujących wyników

- wyznacznik oraz odwrotność macierzy

Algorytmy macierzowe

Będziemy potrzebować następujących wyników

- wyznacznik oraz odwrotność macierzy $O(n^\omega)$ procesorów, RNC^2 – Kaltofen i Pan (1991),

Algorytmy macierzowe

Będziemy potrzebować następujących wyników

- wyznacznik oraz odwrotność macierzy $O(n^\omega)$ procesorów, RNC^2 – Kaltofen i Pan (1991),
- wielomian charakterystyczny

Algorytmy macierzowe

Będziemy potrzebować następujących wyników

- wyznacznik oraz odwrotność macierzy
  $O(n^\omega)$ procesorów, RNC^2 –
  Kaltofen i Pan (1991),
- wielomian charakterystyczny
  $O(n^\omega)$ procesorów, RNC^4 – Giesbrescht (1995),

Algorytmy macierzowe

Będziemy potrzebować następujących wyników

- wyznacznik oraz odwrotność macierzy
  $O(n^\omega)$ procesorów, RNC^2 –
  Kaltofen i Pan (1991),
- wielomian charakterystyczny
  $O(n^\omega)$ procesorów, RNC^4 – Giesbrescht (1995),
- obliczenie wartości wielomianu na macierzy

Algorytmy macierzowe

Będziemy potrzebować następujących wyników

- wyznacznik oraz odwrotność macierzy
  $O(n^\omega)$ procesorów, RNC^2 –
  Kaltofen i Pan (1991),
- wielomian charakterystyczny
  $O(n^\omega)$ procesorów, RNC^4 – Giesbrescht (1995),
- obliczenie wartości wielomianu na macierzy
  $O(n^\omega)$ procesorów, RNC^4 – Giesbrescht (1995),

Algorytmy macierzowe

Będziemy potrzebować następujących wyników

- wyznacznik oraz odwrotność macierzy
  $O(n^\omega)$ procesorów, RNC^2 –
  Kaltofen i Pan (1991),
- wielomian charakterystyczny
  $O(n^\omega)$ procesorów, RNC^4 – Giesbrescht (1995),
- obliczenie wartości wielomianu na macierzy
  $O(n^\omega)$ procesorów, RNC^4 – Giesbrescht (1995),
- rozwiązywanie liniowych rekurencji

Algorytmy macierzowe

Będziemy potrzebować następujących wyników

- wyznacznik oraz odwrotność macierzy
  $O(n^\omega)$ procesorów, RNC^2 –
  Kaltofen i Pan (1991),
- wielomian charakterystyczny
  $O(n^\omega)$ procesorów, RNC^4 – Giesbrescht (1995),
- obliczenie wartości wielomianu na macierzy
  $O(n^\omega)$ procesorów, RNC^4 – Giesbrescht (1995),
- rozwiązywanie liniowych rekurencji
  $O(n^\omega)$ procesorów, RNC^2.

Ślady macierzy

Lemma 8 *Niech $A \in K^{n \times n}$ będzie macierzą. Ślady $\mathrm{tr}(A^i)$, dla każdego $0 \leq i < n$ mogą zostać policzone w czasie $O(\log^4 n)$ na $O(n^\omega)$ procesorach.*

Ślady macierzy

Lemma 8 *Niech $A \in K^{n \times n}$ będzie macierzą. Ślady $\mathrm{tr}(A^i)$, dla każdego $0 \leq i < n$ mogą zostać policzone w czasie $O(\log^4 n)$ na $O(n^\omega)$ procesorach.*

Zapiszmy,

$$\det(A - \lambda I) =$$

$$= \lambda^n - p_1\lambda^{n-1} - p_2\lambda^{n-2} - \cdots - p_n.$$

Ślady macierzy

Mamy

$$\begin{aligned} p_1 &= \operatorname{tr}(A), \\ 2p_2 &= \operatorname{tr}(A^2) - p_1 \operatorname{tr}(A), \\ 3p_3 &= \operatorname{tr}(A^3) - p_1 \operatorname{tr}(A^2) - p_2 \operatorname{tr}(A), \end{aligned}$$

Ślady macierzy

Mamy

$$\begin{aligned} p_1 &= \operatorname{tr}(A), \\ 2p_2 &= \operatorname{tr}(A^2) - p_1 \operatorname{tr}(A), \\ 3p_3 &= \operatorname{tr}(A^3) - p_1 \operatorname{tr}(A^2) - p_2 \operatorname{tr}(A), \end{aligned}$$

oraz dla każdego k,

$$kp_k = \operatorname{tr}(A^k) - \sum_{i=1}^{k-1} p_i \operatorname{tr}(A^{k-i}).$$

Ślady macierzy

Otrzymujemy następującą rekurencje na $\mathrm{tr}(A^k)$,

$$\mathrm{tr}(A^k) = kp_k + \sum_{i=1}^{k-1} p_i \,\mathrm{tr}(A^{k-i}).$$

Ślady macierzy

Otrzymujemy następującą rekurencje na $\mathrm{tr}(A^k)$,

$$\mathrm{tr}(A^k) = kp_k + \sum_{i=1}^{k-1} p_i \,\mathrm{tr}(A^{k-i}).$$

- możemy najpierw policzyć współczynniki wielomianu charakterystycznego p_k,

Ślady macierzy

Otrzymujemy następującą rekurencje na $\mathrm{tr}(A^k)$,

$$\mathrm{tr}(A^k) = kp_k + \sum_{i=1}^{k-1} p_i \,\mathrm{tr}(A^{k-i}).$$

- możemy najpierw policzyć współczynniki wielomianu charakterystycznego p_k,
- następnie możemy rozwiązać tą rekurencję.

Pochodne wyznacznika

Niech $K^k[x]$ to będzie zbiór wielomianów x o stopniu mniejszym niż k.

Theorem 1

Pochodne wyznacznika

Niech $K^k[x]$ to będzie zbiór wielomianów x o stopniu mniejszym niż k.

Theorem 1

- *Mając dane* $A \in K^2[x]^{n \times n}$ *oraz* $t \in K$,

Pochodne wyznacznika

Niech $K^k[x]$ to będzie zbiór wielomianów x o stopniu mniejszym niż k.

Theorem 1

- *Mając dane* $A \in K^2[x]^{n \times n}$ *oraz* $t \in K$,
- *takie, że* $A|_{x=t}$ *jest nieosobliwa.*

Pochodne wyznacznika

Niech $K^k[x]$ to będzie zbiór wielomianów x o stopniu mniejszym niż k.

Theorem 1

- *Mając dane* $A \in K^2[x]^{n \times n}$ *oraz* $t \in K$,
- *takie, że* $A|_{x=t}$ *jest nieosobliwa.*
- *Pochodne* $\frac{d^k}{dx^k} \det(A)\Big|_{x=t}$, *dla każdego* $0 \leq k \leq n$, *mogą zostać obliczone w czasie* $O(\log^4 n)$ *na* $O(n^\omega)$ *procesorach, z dużym prawdopodobieństwem.*

Pochodne wyznacznika

Mamy

$$\frac{d^k}{dx^k}\det(A) = \frac{d^{k-1}}{dx^{k-1}}\left(\det(A)\,\mathrm{tr}\left(A^{-1}\frac{dA}{dx}\right)\right),$$

gdzie użyliśmy równości

$$\frac{d}{dx}\det(A) = \det(A)\,\mathrm{tr}\left(A^{-1}\frac{dA}{dx}\right).$$

Pochodne wyznacznika

$$\frac{d^k}{dx^k}\det(A) = \frac{d^{k-1}}{dx^{k-1}}\left(\det(A)\,\mathrm{tr}\left(A^{-1}\frac{dA}{dx}\right)\right) =$$

$$= \sum_{i=0}^{k-1}\binom{k-1}{i}\left(\frac{d^i}{dx^i}\det(A)\right)\left(\frac{d^{k-i-1}}{dx^{k-i-1}}\,\mathrm{tr}\left(A^{-1}\frac{dA}{dx}\right)\right).$$

Pochodne wyznacznika

$$\sum_{i=0}^{k-1} \binom{k-1}{i} \left(\frac{d^i}{dx^i} \det(A)\right) \left(\frac{d^{k-i-1}}{dx^{k-i-1}} \operatorname{tr}\left(A^{-1}\frac{dA}{dx}\right)\right) =$$

mamy $\frac{d}{dx}A^{-1} = -A^{-1}\frac{dA}{dx}A^{-1}$ and $\frac{d^2}{dx^2}A = 0$, więc

$$= \sum_{i=0}^{k-1} \frac{(k-1)!(-1)^{k-i-1}}{i!} \left(\frac{d^i}{dx^i} \det(A)\right) \operatorname{tr}\left(\left(A^{-1}\frac{dA}{dx}\right)^{k-i}\right)$$

Pochodne wyznacznika

$$\frac{d^k}{dx^k} \det(A)\Big|_{x=t} =$$

$$= \sum_{i=0}^{k-1} \frac{(k-1)!(-1)^{k-i-1}}{i!} \left(\frac{d^i}{dx^i} \det(A)\Big|_{x=t} \right)$$

$$\operatorname{tr}\left(\left(\left(A\Big|_{x=t} \right)^{-1} \frac{dA}{dx}\Big|_{x=t} \right)^{k-i} \right).$$

Pochodne wyznacznika

$$\frac{d^k}{dx^k} \det(A)\Big|_{x=t} =$$

$$= \sum_{i=0}^{k-1} \frac{(k-1)!(-1)^{k-i-1}}{i!} \left(\frac{d^i}{dx^i} \det(A)\Big|_{x=t} \right)$$

$$\operatorname{tr}\left(\left(\left(A\Big|_{x=t} \right)^{-1} \frac{dA}{dx}\Big|_{x=t} \right)^{k-i} \right).$$

- możemy najpierw obliczyć ślady,

Pochodne wyznacznika

$$\frac{d^k}{dx^k} \det(A)\Big|_{x=t} =$$

$$= \sum_{i=0}^{k-1} \frac{(k-1)!(-1)^{k-i-1}}{i!} \left(\frac{d^i}{dx^i} \det(A)\Big|_{x=t} \right)$$

$$\operatorname{tr}\left(\left(\left(A\Big|_{x=t} \right)^{-1} \frac{dA}{dx}\Big|_{x=t} \right)^{k-i} \right).$$

- możemy najpierw obliczyć ślady,
- a następnie rozwiązać rekurencję.

Pochodna Adjoint'a

Theorem 2

- *Mając dane* $A \in K^2[x]^{n \times n}$, $t \in K$ *oraz* $0 \leq k \leq n$,

Pochodna Adjoint'a

Theorem 2

- *Mając dane* $A \in K^2[x]^{n \times n}$, $t \in K$ *oraz* $0 \leq k \leq n$,
- *takie, że* $A|_{x=t}$ *jest nieosobliwa.*

Pochodna Adjoint'a

Theorem 2

- *Mając dane* $A \in K^2[x]^{n \times n}$, $t \in K$ *oraz* $0 \leq k \leq n$,
- *takie, że* $A|_{x=t}$ *jest nieosobliwa.*
- *Pochodna* $\frac{d^k}{dx^k} \operatorname{adj}(A)\Big|_{x=t}$, *może być obliczona w czasie* $O(\log^4 n)$ *na* $O(n^\omega)$ *procesorach, z dużym prawdopodobieństwem.*

Pochodna Adjoint'a

$$\frac{d^k}{dx^k}\operatorname{adj}(A) = \frac{d^k}{dx^k}\left(\det(A)A^{-1}\right) =$$

$$= \sum_{i=0}^{k} \binom{k}{i} \left(\frac{d^i}{dx^i}\det(A)\right)\left(\frac{d^{k-i}}{dx^{k-i}}A^{-1}\right).$$

Pochodna Adjoint'a

$$= \sum_{i=0}^{k} \binom{k}{i} \left(\frac{d^i}{dx^i} \det(A) \right) \left(\frac{d^{k-i}}{dx^{k-i}} A^{-1} \right) =$$

mamy $\frac{d}{dx} A^{-1} = -A^{-1} \frac{dA}{dx} A^{-1}$ and $\frac{d^2}{dx^2} A = 0$, więc

$$= \sum_{i=0}^{k} \binom{k}{i} \left(\frac{d^i}{dx^i} \det(A) \right)$$

$$\left((k-i)!(-1)^{k-i} \left(A^{-1} \frac{dA}{dx} \right)^{k-i} A^{-1} \right).$$

Pochodna Adjoint'a

$$\frac{d^k}{dx^k}\operatorname{adj}(A)\Big|_{x=t} =$$

$$= \left(\sum_{i=0}^{k} \frac{k!(-1)^{k-i}}{i!}\left(\frac{d^i}{dx^i}\det(A)\Big|_{x=t}\right)\right.$$

$$\left.\left(\left(A\Big|_{x=t}\right)^{-1}\frac{dA}{dx}\Big|_{x=t}\right)^{k-i}\right)\left(A\Big|_{x=t}\right)^{-1}.$$

Pochodna Adjoint'a

$$\frac{d^k}{dx^k}\operatorname{adj}(A)\Big|_{x=t} =$$

$$= \left(\sum_{i=0}^{k} \frac{k!(-1)^{k-i}}{i!}\left(\frac{d^i}{dx^i}\det(A)\Big|_{x=t}\right)\right.$$

$$\left.\left(\left(A\Big|_{x=t}\right)^{-1}\frac{dA}{dx}\Big|_{x=t}\right)^{k-i}\right)\left(A\Big|_{x=t}\right)^{-1}.$$

Możemy obliczyć ten wielomian w RNC na $O(n^\omega)$ procesorach.

Podsumowanie

Pokazaliśmy

- efektywny pod względem liczby procesorów algorytm na znajdowanie doskonałych skojarzeń,

Podsumowanie

Pokazaliśmy

- efektywny pod względem liczby procesorów algorytm na znajdowanie doskonałych skojarzeń,
- alternatywny algorytm sekwencyjny działający w czasie $\tilde{O}(n^{\omega})$.