Problem szczotkowania grafu bez ograniczeń pojemności

Paweł Troka 132334, Informatyka, ETI, PG

Plan prezentacji

1. Wstęp
2. Założenia
3. Definicje
4. Czyszczenie w ogólnym przypadku
    1. Górne granice
    2. W zależności od cutwidth i bisection width
5. Czyszczenie grafów będących produktem kartezjańskim
    1. Górna granica
    2. Gridy
    3. Hiperkostki
6. Czyszczenie drzew
7. Podsumowanie
8. Bibliografia

1. Wstęp (1)

- W problemie czyszczenia grafu szczotki czyszczą graf poprzez trawersowanie go zgodnie z określonymi regułami
- Pojawiają się ciekawe problemy typu wyznaczenie minimalnej liczby szczotek potrzebnej aby oczyścić cały graf. Ta wielkość jest nazywana „the brushing number” – „liczba szczotkująca grafu” (moje wolne tłumaczenie)
- W tej pracy została przedstawiona nowa odmiana problemu szczotkowania grafu w której jeden wierzchołek jest czyszczony na raz ale więcej niż jedna szczotka może trawersować brudną krawędź
- W pracy udało się wyznaczyć liczbę szczotkującą dla grafów będących wynikiem produktu kartezjańskiego, drzew jak i wyznaczono górną i dolną granicę dla przypadku ogólnego

2. Założenia (1)

- Inicjalnie wszystkie krawędzie i wierzchołki grafu są zanieczyszczone
- Szczotki poruszają się po grafie czyszcząc go jak przechodzą.
- Gdy wszystkie wierzchołki i krawędzie zostały odwiedzone przez szczotkę graf zostaje uznany za oczyszczony, *chociaż w niektórych modelach ponowne zanieczyszczenie może wystąpić*.
- Niektóre modele zezwalają na wielokrotne przechodzenie przez krawędzie lub jednokrotne ale dopuszczają wiele szczotek na raz.
- Czasami nakładane są ograniczenia typu „tylko zanieczyszczone krawędzie mogą być odwiedzone” i „każda krawędź może być odwiedzona przez maksymalnie jedną szczotkę”.
- W pracy dozwolono na trawersowanie krawędzi przez wiele szczotek ale tylko raz mogą one przez daną krawędź przejść.

2. Założenia (2)

- Dwie kluczowe kwestie rozważane w pracy to
  - Wyznaczanie liczby szczotkującej grafu
  - Opisanie odpowiadających strategii czyszczenia grafu

3. Definicje (1)

- Przykład

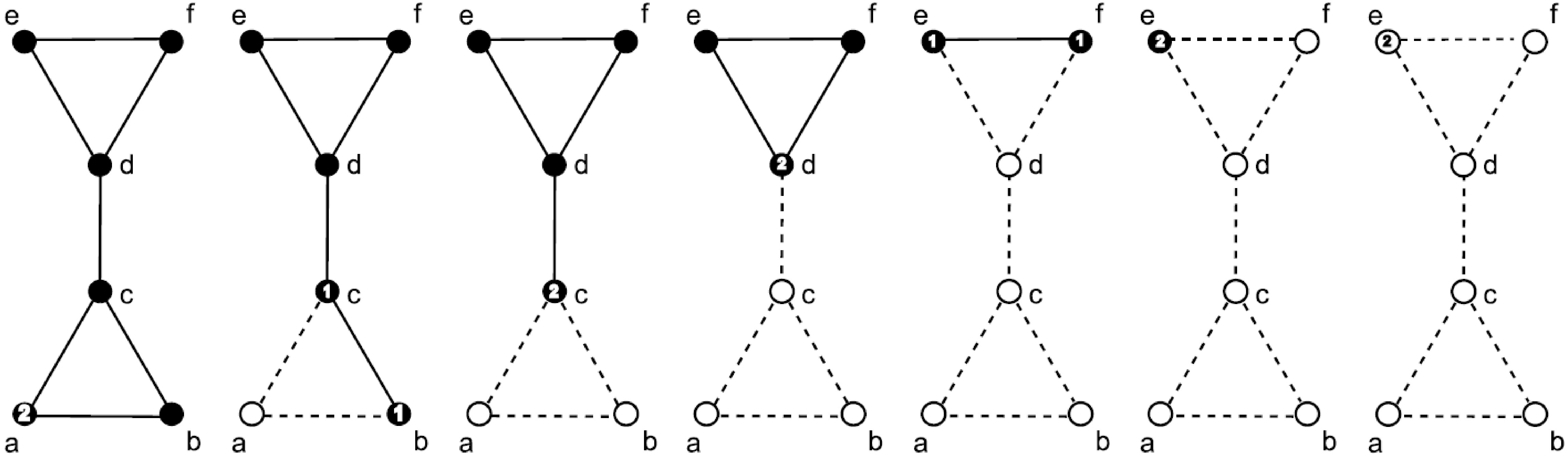


Fig. 1. A graph G with an initial configuration of 2 brushes.

3. Definicje (2) - oznaczenia

- G = (V, E) to skończony, nieskierowany graf
- Początkowa konfiguracja szczotek jest dana przez funkcję $\omega_0 : V \rightarrow N_0$, gdzie $N_0 = \{0, 1, 2, . . .\}$
- $\omega_0(v)$ jest liczbą szczotek początkowo znajdujących się na wierzchołku v
- Wszystkie wierzchołki i krawędzie są inicjalnie brudne
- W każdym kroku t procesu czyszczenia $\omega_t(v)$ oznacza liczbę szczotek znajdujących się na wierzchołku $v \in V$, i $D_t \subseteq V$ oznacza zbiór brudnych wierzchołków
- Krawędź $uv \in E$ jest brudna wtedy i tylko wtedy jeżeli zarówno u jak i v są brudne, czyli $\{u, v\} \subseteq D_t$
- N(v) oznacza sąsiedztwo v
- $D_t(v)$ oznacza liczbę brudnych krawędzi incydentnych do v w kroku t

$$D_t(v) = \begin{cases} |N(v) \cap D_t| & \text{if } v \in D_t \\ 0 & \text{otherwise} \end{cases}$$

3. Definicje (2) – oznaczenia (2)

- Funkcja δ zwraca liczbę szczotek, które trawersują brudną krawędź. W najbardziej standardowym modelu, rozważanym w [3] tylko jedna szczotka mogła przejść przez konkretną krawędź; czyli, $\delta(uv) = 1$ dla każdego $uv \in E$ (zakładając, że graf został wyczyszczony, w innym wypadku $\delta(uv)$ dla każdej wyczyszczonej krawędzi)
- Nie ma żadnego zysku w zostawianiu niepotrzebnych szczotek po drodze; więc możemy zawsze założyć, że $\rho(\alpha_{t+1}) = \omega_t\,(\alpha_{t+1})$, zakładając, że istnieje przynajmniej jedna brudna krawędź incydentna do α_{t+1} w czasie czyszczenia (czyli $N(\alpha_{t+1}) \cap D_t \neq \emptyset$). Innymi słowy nierówność (1) staje się równością.

3. Definicje (3) - uogólniony proces czyszczenia $\mathfrak{P}(G, \omega_0) = \{(\omega_t, D_t)\}_{t=0}^{T}$

⓪ Initially, all vertices are dirty: $D_0 = V$; set $t = 0$.

① Let α_{t+1} be any vertex in D_t such that $\omega_t(\alpha_{t+1}) \geq D_t(\alpha_{t+1})$. If no such vertex exists, then stop the process (set $T = t$), return the **cleaning sequence** $\alpha = (\alpha_1, \alpha_2, \ldots, \alpha_T)$, the **final set of dirty vertices** D_T, the **final configuration of brushes** ω_T, and the **distribution of brushes** δ.

② Assign to each dirty edge $v\alpha_{t+1}$ incident to α_{t+1} a natural number $\delta(v\alpha_{t+1})$ such that the sum of numbers assigned is at most $\omega_t(\alpha_{t+1})$; in other words, for each dirty neighbour v of α_{t+1}, let $\delta(v\alpha_{t+1}) \in \mathbb{N}$ and

$$\rho(\alpha_{t+1}) = \sum_{v \in N(\alpha_{t+1}) \cap D_t} \delta(v\alpha_{t+1}) \leq \omega_t(\alpha_{t+1}). \qquad (1)$$

③ Clean α_{t+1} and all dirty incident edges by moving $\delta(v\alpha_{t+1})$ brushes from α_{t+1} to each dirty neighbour v of α_{t+1}. Consequently, $D_{t+1} = D_t \setminus \{\alpha_{t+1}\}$, $\omega_{t+1}(\alpha_{t+1}) = \omega_t(\alpha_{t+1}) - \rho(\alpha_{t+1})$, and for every $v \in N(\alpha_{t+1}) \cap D_t$, $\omega_{t+1}(v) = \omega_t(v) + \delta(v\alpha_{t+1})$, otherwise $\omega_{t+1}(v) = \omega_t(v)$.

④ Set $t = t + 1$ and go back to ①.

3. Definicje (4) – czy graf może być oczyszczony?

- Graf G = (V, E) może być wyczyszczony przez inicjalną konfigurację szczotek ω_0 jeżeli istnieje proces czyszczący $\mathcal{B}(G, \omega_0)$ który zwraca pusty zbiór brudnych wierzchołków (czyli $D_T = \emptyset$).
- Przyjmijmy (uogólnioną) liczbę szczotkującą grafu G, oznaczaną B(G), jako minimalną liczbę szczotek potrzebnych do wyczyszczenia G; czyli

$$B(G) = \min_{\omega_0 : V \to \mathbb{N}_0} \left\{ \sum_{v \in V} \omega_0(v) \ : G \text{ can be cleaned by } \omega_0 \right\}$$

- W standardowym modelu tego problemu liczbę szczotkującą oznacza się jako b(G). Jako, że w nim jest więcej restrykcji można wywnioskować, że **B(G) ≤ b(G)**. Dla niektórych klas grafów różnica ta może być ogromna, chociaż zdarzają się też takie, że B(G) = b(G).

3. Definicje (5) – ścieżka szczotkująca

- **Ścieżka szczotkująca** danej szczotki to ścieżka uformowana ze zbioru krawędzi oczyszczonych przez tę szczotkę
- Z definicji, G może zawierać B(G) ścieżek szczotkujących.
- Minimalna liczba (nieskierowanych) ścieżek przez które graf może być pokryty jest dolnym ograniczeniem dla B(G)
- Dlaczego jest to tylko dolne ograniczenie?
  - Ponieważ niektóre ścieżki pokrywające nie będą zgodne z założeniami procesu czyszczenia.
  - Ale za to takie ścieżki mogą być traktowane jako skierowane (z orientacją odpowiadającą chronologicznemu kierunkowi szczotek), co daje nam zupełnie inny pogląd na ten problem

4.1 Czyszczenie w ogólnym przypadku – górna granica

- Skoro $B(G) \leq b(G)$ każde górne ograniczenie dla $b(G)$ jest też górnym ograniczeniem dla $B(G)$. Co ciekawe dla niektórych grafów $b(G)-B(G)$ może być dowolnie duże ale są klasy grafów dla których jest równe.
- Dla każdego $n \in N$ mamy:
  - $B(C_n) = b(C_n) = 2$ gdzie C_n jest cyklem długości n,
  - $B(P_n) = b(P_n) = 1$ gdzie P_n jest ścieżką o długości n-1,
  - $B(K_n) = b(K_n) = \lfloor n^2/4 \rfloor$ gdzie K_n jest grafem pełnym o n wierzchołkach.

$$B(G) \leq b(G) \leq \frac{|E|}{2} + \frac{|V|}{4} - \frac{1}{4} \sum_{v \in V(G), \deg(v) \text{ is even}} \frac{1}{\deg(v)+1}$$

- Dla każdego grafu $G = (V, E)$.
- Dowód jest niekonstruktywny i znajduje się w [2]

4.2 Czyszczenie w ogólnym przypadku – w zależności od cutwidth i bisection width

- Definicja „szerokości cięcia” grafu
  - Niech $G = (V, E)$ będzie grafem n-wierzchołkowym grafem
  - Niech $f : V \rightarrow \{1, 2, . . . , n\}$ będzie bijekcją etykietującą wierzchołki grafu G na liczby opisujące ich kolejność w liniowym układzie grafu.
  - Szerokość cięcia f (dla G) jest zdefiniowana następująco:

$$cw_f(G) = \max_{1 \leq i \leq n} |\{uv \in E : f(u) \leq i < f(v)\}|$$

- Szerokość cięcia grafu G, $cw(G)$, jest minimalną możliwą szerokością cięcia dla wszystkich liniowych ułożeń wierzchołków G; czyli, $cw(G) = \min_f cw_f(G)$.
- W [3] poczyniono obserwację, że szerokość cięcia grafu ogranicza z dołu $b(G)$

4.2 Czyszczenie w ogólnym przypadku – w zależności od cutwidth i bisection width

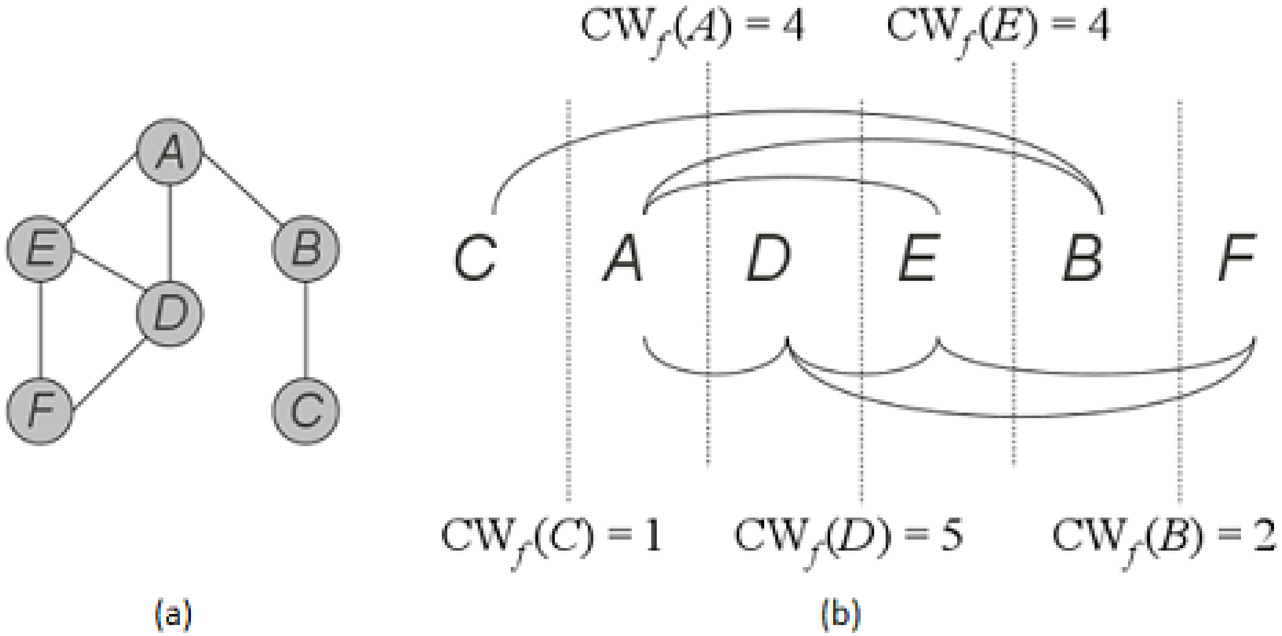


Figure 1: (a) Graph example, (b) Cutwidth of G for a labeling *f*.

4.2 Czyszczenie w ogólnym przypadku – w zależności od cutwidth i bisection width

- Twierdzenie
  - Dla każdego G, B(G) ≥ cw(G)
- Dowód:
  - Dowód jest trywialny. Niech G = (V, E) będzie dowolnym grafem o n wierzchołkach
  - Rozpatrujemy sekwencję czyszczącą $\alpha = (\alpha_1, \alpha_2, \ldots, \alpha_n)$ która zwraca B(G).
  - Zauważmy, że α może być interpretowane jako liniowe ułożenie wierzchołków $f_\alpha : V \to \{1, 2, \ldots, n\}$ zdefiniowane jako $f(\alpha) = i$ dla $i \in \{1, 2, \ldots, n\}$.
  - Aby uzyskać ograniczenie z dołu dla B(G), dla każdego $i \in \{1, 2, \ldots, n\}$, niech

$$\gamma_i = \{uv \in E : f_\alpha(u) \leq i < f_\alpha(v)\} = \{\alpha_x \alpha_y \in E : x \leq i < y\}$$

  - Czyli takie krawędzie które istnieją w grafie i w sekwencji czyszczącej jeden wierzchołek jest czyszczony przed lub w kroku i a drugi po kroku i

4.2 Czyszczenie w ogólnym przypadku – w zależności od cutwidth i bisection width

- Dowód (cd):
  - Należy zauważyć, że na koniec i-tego kroku procesu, przynajmniej jedna szczotka przebyła każdą krawędź z γ_i
  - Co więcej, każda krawędź z γ_i była trawersowana przez różne szczotki skoro jeden koniec krawędzi jest wciąż brudny
  - Więc przynajmniej $|\gamma_i|$ jest potrzebny do oczyszczenia grafu G. Skoro jest to prawdziwe dla każdego „i” otrzymujemy następujący rezultat

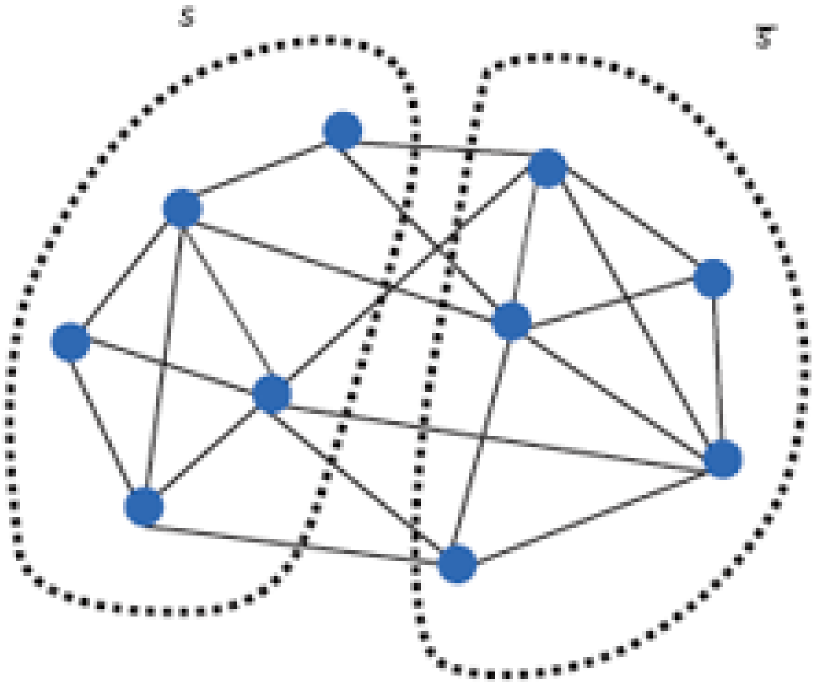
$$B(G) \geq \max_{1 \leq i \leq n} |\gamma_i| = cw_{f_\alpha}(G) \geq cw(G)$$

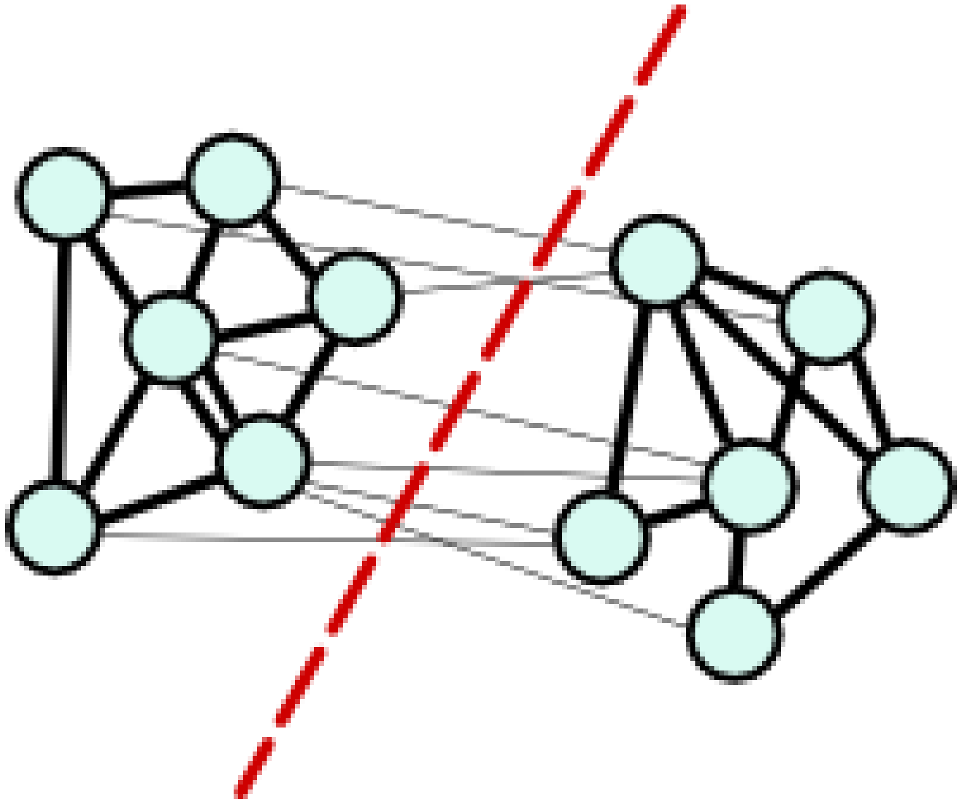
  - C.n.d.

4.2 Czyszczenie w ogólnym przypadku – w zależności od cutwidth i bisection width

- Definicja szerokości bisekcji grafu
  - Niech G = (V, E) będzie grafem z |V| = n.
  - Dla S ⊆ V, niech e(S, V \ S) oznacza liczbę krawędzi pomiędzy S i jego uzupełnieniem
  - Bisekcją V jest taki podział V na dwie części, S i V \S, że |S| = ⌊n/2⌋ i |V \S| = ⌈n/2⌉.
  - Rozmiar bisekcji (S, V \S) jest liczbą krawędzi przechodzących między częściami; czyli, e(S, V \S).
  - Minimalna bisekcja to bisekcja z jak najmniejszym rozmiarem.
  - Rozmiar bisekcji jest nazywany szerokością bisekcji i oznaczany przez bw(G).

4.2 Czyszczenie w ogólnym przypadku – w zależności od cutwidth i bisection width





4.2 Czyszczenie w ogólnym przypadku – w zależności od cutwidth i bisection width

- Niech G = (V, E) będzie grafem na n wierzchołkach
- Dla każdego k ∈ {1, 2, . . . , n}, niech

$$B_k = \min_{S \subseteq V, |S|=k} e(S, V \setminus S)$$

- Wtedy, $B(G) \geq \max_k B_k$. W szczególności, $B(G) \geq b_{\lfloor n/2 \rfloor} = bw(G)$.
- Mini dowód:
  - Wystarczy zauważyć, że dla każdego ułożenia f grafu G mamy

$$\max_{1 \leq k \leq n} |\{uv \in E : f(u) \leq k < f(v)\}| \geq \max_{1 \leq k \leq n} B_k$$

4.2 Czyszczenie w ogólnym przypadku – w zależności od cutwidth i bisection width

- Ograniczona szerokość cięcia grafu G = (V, E) na n wierzchołkach, $cw_p(G)$, jest minimalną szerokością cięcia dla wszystkich możliwych liniowych ułożeń wierzchołków G które tworzą ścieżkę; czyli,

$$cw_p(G) = \min\left\{cw_f(G) : f^{-1}(i)f^{-1}(i+1) \in E \text{ for all } i \in \{1, 2, \ldots, n-1\}\right\}$$

- Oczywiście tylko jeżeli graf ma ścieżkę Hamiltona. W innym wypadku, $cw_p(G) = \infty$.

4.2 Czyszczenie w ogólnym przypadku – w zależności od cutwidth i bisection width

- Twierdzenie:
  - Dla każdego grafu G, $B(G) \leq cw_p(G)$
- Dowód:
  - Jeśli G nie jest hamiltonowski to dowód jest trywialny. W inny wypadku, niech f będzie liniowym ułożeniem wierzchołków G które zwraca $cw_p(G)$.
  - Czyścimy G używając sekwencji czyszczącej powiązanej z f . Krawędzie $f^{-1}(i)f^{-1}(i + 1)$, $i = 1, 2, \ldots, n - 1$, ścieżki Hamiltona wyznaczone przez f są nazywane specjalnymi krawędziami, reszta natomiast zwykłymi
  - Zaczynamy z $\hat{b} := cw_p(G) = cw_f(G)$ szczotek umieszczonych w wierzchołku $f^{-1}(1)$. W każdym kroku procesu wysyłamy dokładnie jedną szczotkę przez normalną, brudną krawędź; pozostałe są wysyłane przez specjalne, brudne krawędzie które stanowią szkielet grafu. Oczywistym jest, że taka strategia czyści graf. Na końcu danego kroku czasowego t, liczba krawędzi pomiędzy brudnymi i czystymi wierzchołkami jest $|\gamma_t|$, gdzie

$$\gamma_t = \{uv \in E : f(u) \leq t < f(v)\}$$

  - Liczba normalnych krawędzi tego typu wynosi $|\gamma_t| - 1$, tyle samo ile szczotek trawersowało je. Wszystkie inne szczotki są dostępne na początku kroku czasowego t a liczba szczotek które zostały wysłane przez specjalną krawędź $f^{-1}(t)f^{-1}(t + 1)$ jest równa $\hat{b} - (|\gamma_t| - 1) \geq 1$. Proces może być kontynuowany natomiast dowód jest skończony.

4.2 Czyszczenie w ogólnym przypadku – w zależności od cutwidth i bisection width

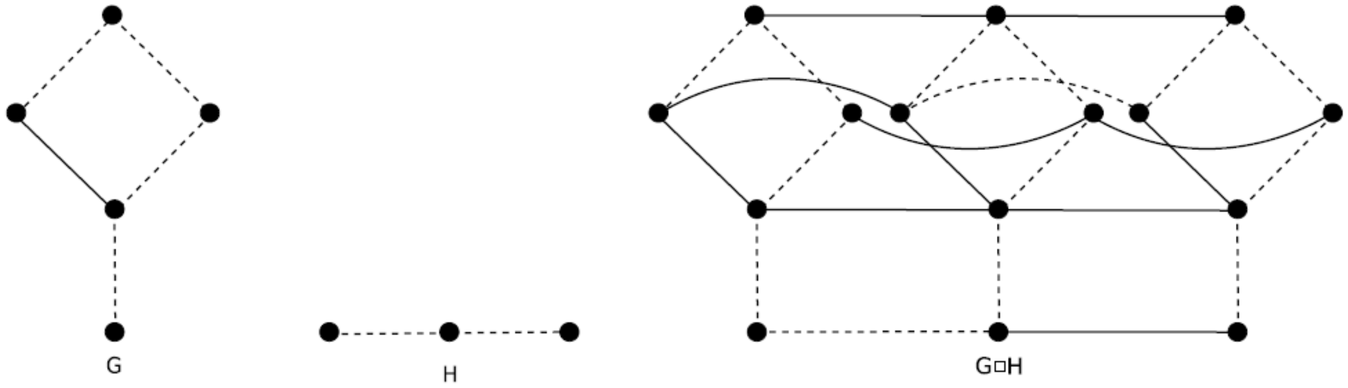


Fig. 2. The layout used to clean $G\Box H$.

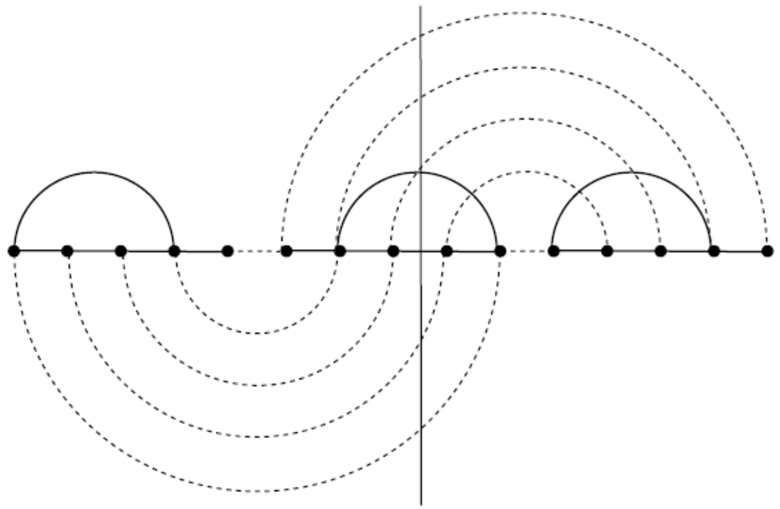


Fig. 3. The layout used to clean $G\Box H$.

5.1 Czyszczenie grafów będących produktem kartezjańskim – górna granica

- Graf G×H jest produktem kartezjańskim grafów G i H.
- Zawiera zbiór wierzchołków V(G)×V(H) gdzie $(u, v) \in V(G \times H)$ sąsiaduje z $(u', v') \in V(G \times H)$ kiedy albo $u = u'$ i $vv' \in E(H)$ albo $v = v'$ i $uu' \in E(G)$
- Łatwo można zauważyć, że G×H składa się z |V(G)| kopii H lub |V(H)| kopii G
- Tą własność można użyć aby uzyskać górne ograniczenie dla B(G×H)
- W [3] pokazano, że $b(G \times H) \leq |V(H)|b(G) + |V(G)|b(H)$, i to ograniczenie może być użyte dla B(G) też
- Jednakże ograniczenie z góry, które jest znacznie silniejsze może być wyznaczone dla B(G)

5.1 Czyszczenie grafów będących produktem kartezjańskim – górna granica

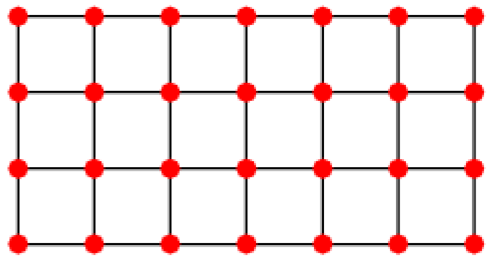
- $B(G \times H) \le cw_p(G \times H) \le |V(G)| cw_p(H) + cw_p(G)$
- Przez symetrię mamy też:
- $B(G \times H) \le cw_p(G \times H) \le |V(H)| cw_p(G) + cw_p(H)$
- Dowód nie jest trywialny

5.1 Czyszczenie grafów będących produktem kartezjańskim – górna granica - dowód

- We will show that cwp(GH) ≤ |V(G)|cwp(H)+cwp(G), from which the result follows since B(GH) ≤ cwp(GH)
- Note that if either G or H has no Hamiltonian path, then by definition we have |V(G)|cwp(H)+cwp(G) =∞and the result holds. Thus, we assume that G and H both have Hamiltonian paths.
- Let (v1, v2, . . . , vn), n = |V(G)|, be a Hamiltonian path in G induced by the linear layout g which yields cwp(G). Also, let (w1,w2, . . . ,wk), k = |V(H)|, be a Hamiltonian path in H induced the linear layout h of H that yields cwp(H). Consider the Cartesian product of G and H.
- Let us colour edges of each copy of H red and edges of each copy of G green. Copies of G are ordered according to the layout h (that is, the ith copy of G, Gi, has vertices (v1,wi), (v2,wi), . . . , (vn,wi)). In order to provide an upper bound of cwp(GH) we consider the following layout of GH. Use g applied to G1, then the inverse of g (that is Hamiltonian path (vn, vn−1, . . . , v1)) applied to G2, then g again but applied to G3, etc. (See Fig. 2 in which dashed edges outline Hamiltonian paths.)
- Formally, take the following Hamiltonian path in GH: ((v1,w1), (v2,w1), . . . , (vn,w1), (vn,w2), (vn−1,w2), . . . , (v1,w2), (v1,w3), (v2,w3), . . . , (vn,w3), . . .), and denote by f the linear layout of GH which induces this path. Then, for every s (1 ≤ s ≤ nk) we have {(vj1 ,wi1)(vj2 ,wi2) ∈ E : f ((vj1 ,wi1)) ≤ s < f ((vj2 ,wi2))} ≤ n ・ cwp(H) + cwp(G), since there are always at most cwp(G) green edges and at most n ・ cwp(H) red edges. For example, Fig. 3 gives a linear layout of the graph GH from Fig. 2. Here, the solid edges are the edges of the copies of G (called ‘green’ in the proof) and the dashed edges are the edges of the copies of H (called ‘red’). The vertical line corresponds to the restricted cutwidth of this layout.
- In particular, there are |V(G)|cwp(H) edges of H and cwp(G) edges of G crossing the vertical line. Hence cwp(GH) ≤ cwf (GH) ≤ n ・ cwp(H) + cwp(G). (In fact, cwf (GH) = n ・ cwp(H) + cwp(G).) The proof of the theorem is finished.

5.2 Czyszczenie grafów będących produktem kartezjańskim – gridy

- Grid to graf będący produktem kartezjańskim dwóch skończonych ścieżek
- Szerokość bisekcji dla $P_n \times P_n$ jest dobrze znana



$$B_{\lfloor n^2/2 \rfloor} = bw(P_n \square P_n) = \begin{cases} n & \text{if } n \text{ is} \\ n+1 & \text{otherwise} \end{cases}$$

- Dla nieparzystego n, to wystarczy aby ustalić ograniczenie z dołu dla $B(P_n \times P_n)$
- Dla parzystego $n \geq 4$, zauważmy, że $B[(n^2/2)-1] = B[(n^2/2)+1] = n + 1$
- Korzystając z tego, że $B(G) \geq bw(G)$ mamy, że $B(P_n \times P_n) \geq n + 1$
- Aby otrzymać ograniczenie z góry korzystamy z faktu, że $cw_p(P_n) = 1$ i wzoru dla produktów kartezjańskich

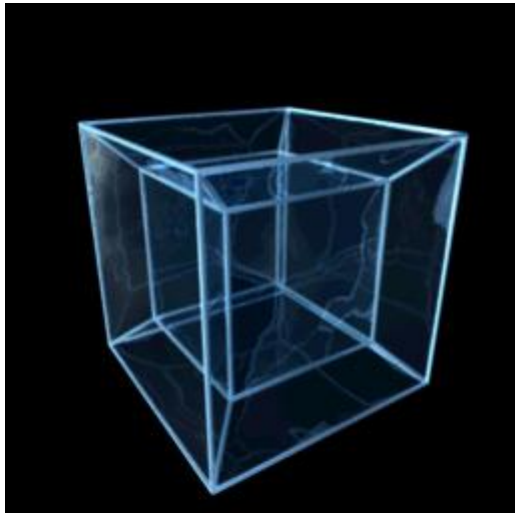
5.2 Czyszczenie grafów będących produktem kartezjańskim – gridy

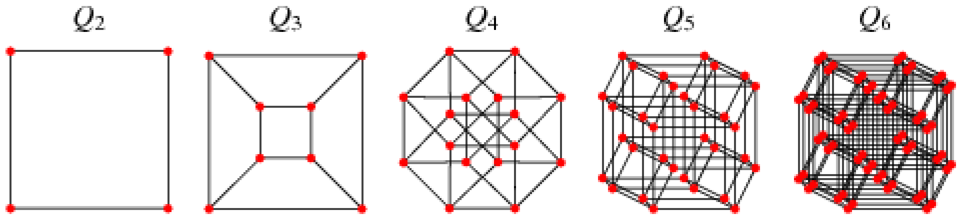
- Otrzymujemy dolne i górne ograniczenie równe n+1, więc
- Dla każdego $n \in N \setminus \{1, 2\}$, $B(P_n \times P_n) = n + 1$
- Wynik może być rozszerzony do kartezjańskich produktów nieizomorficznych ścieżek
- Dla każdego $m > n \geq 2$, $B(P_m \times P_n) = n + 1$

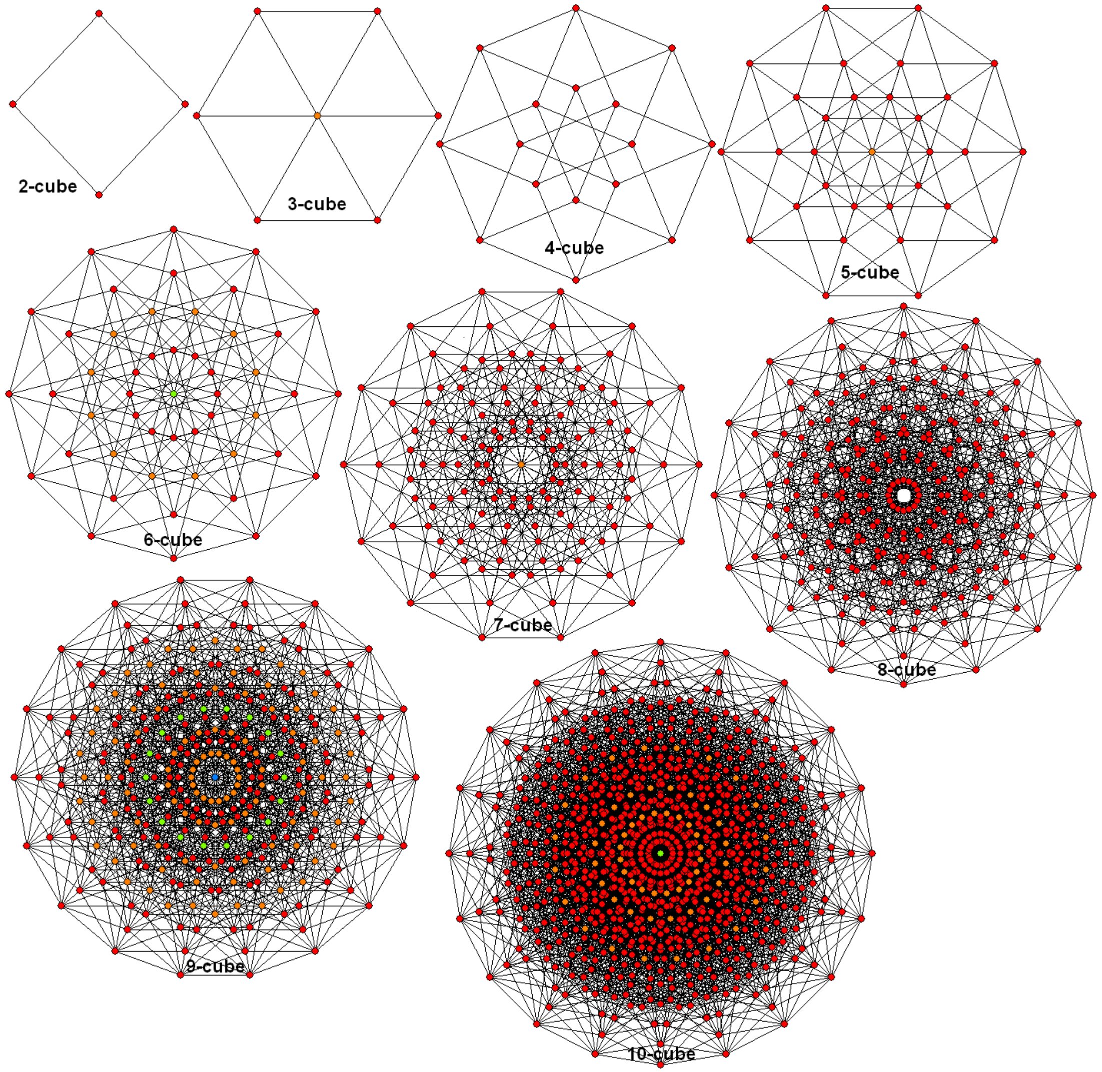
5.3 Czyszczenie grafów będących produktem kartezjańskim – hiperkostki

- Wierzchołki hiperkostki Q_n są punktami w n-wymiarowej przestrzeni. Są sąsiadami w Q_n wtedy i tylko wtedy gdy różnią się dokładnie na jednej współrzędnej.
- Mamy np. $Q_1 = P_2$, $Q_2 = C_4$, i Q_3 jest sześcianem w 3-wymiarowej przestrzeni.
- Dla każdego $n \in N$ mamy własność $Q_{n+1} = Q_1 Q_n$
- W [4] pokazano, że $cw(Q_n) = 2/3\ (2n - a)$ gdzie $a = 1$ jeśli n jest parzyste i $a = ½$ w innym wypadku
- Wiemy, że $B(Q_n) \geq cw(G)$ co daje nam ograniczenie z dołu
- Pokażemy, że ograniczenie z dołu w postaci $cw_p(G)$ wynosi dokładnie tyle samo

5.3 Czyszczenie grafów będących produktem kartezjańskim – hiperkostki







5.3 Czyszczenie grafów będących produktem kartezjańskim – hiperkostki

- Dla każdego $n \in N$, $B(Q_n) = r_n$, gdzie
- $r_n = \begin{cases} \frac{2}{3}(2^n - 1), & n = 2k \\ \frac{2}{3}\left(2^n - \frac{1}{2}\right), & n = 2k+1 \end{cases}$
- Dowód:
  - Jak zostało powiedziane wystarczy pokazać, że $cw_p(Q_n) \leq r_n$ dla każdego $n \in N$
  - Dowód jest przez indukcję dla n
  - Dla n=1 $cw_p(Q_1) = cw_p(K_2) = 1$
  - Dla n>=2 korzystamy ze wzoru dla produktów kartezjańskich

5.3 Czyszczenie grafów będących produktem kartezjańskim – hiperkostki

- Dowód c.d.
- $cw_p(Q_n) = cw_p(Q_{n-1}K_2) \leq 2cw_p(Q_{n-1}) + 1$, ponieważ $cw_p(K_2)=1$
- Założenie, że: $cw_p(Q_n) \leq 2cw_p(Q_{n-1})$ dla $n=2k$

$$r_n = \begin{cases} 2r_{n-1} + 1 & \text{if } n \text{ is odd} \\ 2r_{n-1} & \text{otherwise} \end{cases}$$

- $cw_p(Q_n) \leq r_n$ dla każdego $n \in N$
- Teraz wystarczy pokazać, że dla każdego parzystego $n \in N$, $cw_p(Q_n)$ jest parzyste

5.3 Czyszczenie grafów będących produktem kartezjańskim – hiperkostki

- Pokażemy, że $|\gamma_t|$ jest parzyste dla każdego t

$$\gamma_t = \{uv \in E : f(u) \leq t < f(v)\}$$

- Skoro każdy wierzchołek ma stopień n, $|\gamma_1| = n$ jest parzyste. Załóżmy teraz, że $|\gamma_t|$ jest parzyste i są 3 krawędzie z wierzchołków
- f (u) < t + 1 do wierzchołka v z f (v) = t + 1. Z tego wynika, że jest n − k krawędzi z wierzchołka v z f (v) = t + 1 do wierzchołków z f (u) > t + 1

$$|\gamma_{t+1}| = |\gamma_t| - k + (n - k) = |\gamma_t| - 2k + n$$

- Co je parzyste, ponieważ każdy człon jest parzysty

7. Podsumowanie (1)

- W ogólnym przypadku:
  - $cw(G) \ \leq \ B(G) \leq \frac{|E|}{2} + \frac{|V|}{4} - \frac{1}{4}\sum_{v\in V(G),\deg(v) is\ even} \frac{1}{\deg(v)+1}$
  - $bw(G) \ \leq \ B(G) \leq cw_p(G)$
- Dla produktów kartezjańskich:
  - $B(G \times H) \leq \ cw_{p}(G \times H) \leq \ |V(G)| cw_{p}(H) + \ cw_{p}(G)$
- Dla gridów
  - $B(G) = \ B(K_m) = \ b(K_m) = \ \left\lfloor \frac{m^2}{4} \right\rfloor = \frac{m^2}{4} + \ O(1)$
- Dla hiperkostek
  - $B(Q_n) = \begin{cases} \frac{2}{3}(2^n - 1), & n = 2k \\ \frac{2}{3}\left(2^n - \frac{1}{2}\right), & n = 2k + 1 \end{cases}$
- Dla drzew (o parzystej liczbie liści):
  - $\frac{d_l(T)}{2} \leq \ B(T) \leq \frac{d_l(T)}{2} + 1$
- Dla drzew (o nieparzystej liczbie liści)
  - $B(T) = \frac{d_l(T)+1}{2}$

7. Podsumowanie (2)

- Problem jest niezwykle ciekawy i stosunkowo niezbadany / mało znany (różne modele, różne wyniki)
- Dla konkretnych grafów udaje się wyznaczyć dokładnie liczbę szczotkującą grafu lub przynajmniej górną/dolną granicę
- Pozornie proste drzewa okazały się najbardziej nietrywialne z rozpatrywanych przypadków
- Skoro zostało ustalone, że drzewa z parzystą liczbą wierzchołków stopnia 1 mogą być podzielone na dwa zbiory zależnie czy $B(T) = d_{\ell}(T)/2$ czy wynosi o 1 więcej jest naturalnym zapytać się o **cechy charakterystykę drzew przynależących do danego zbioru**
- Można też zdefiniować następujący problem decyzyjny: „**Czy dla drzewa T dla którego $d_{\ell}(T)$ jest parzyste, $B(T) = d_{\ell}(T)/2$”** ? Autorzy artykułu pozostawiają jako pytanie otwarte złożoność tego problemu.

8. Bibliografia

1. Darryn Bryant, Nevena Francetic, Przemyslaw Gordinowicz, David A. Pike, Pawel Pralat: Brushing without capacity restrictions. Discrete Applied Mathematics 170: 33-45 (2014)
2. N. Alon, P. Prałat, N. Wormald, Cleaning regular graphs with brushes, SIAM J. Discrete Mathematics 23 (2008) 233–250.
3. M.-E. Messinger, R.J. Nowakowski, P. Prałat, Cleaning a network with brushes, Theoret. Comput. Sci. 399 (2008) 191–205.
4. K. Nakano, Linear layout of generalized hypercubes, Internat. J. Found Comput. Sci. 14 (2003) 137–156.