Algoritmica Grafurilor - Cursul 11

20 decembrie 2019

Teorema 1

(Karp, 1972) SM $\leqslant_P$ CH.

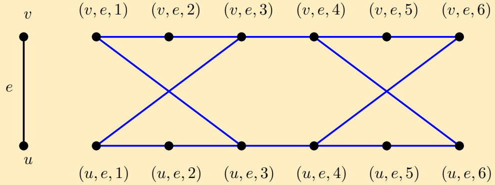
Demonstraţie. Fie $G = (V, E)$ şi $j \in \mathbb{N}$ o instanţă a problemei **SM**. Vom construi în timp polinomial (relativ la $n = |V|$) un graf H astfel încât există o mulţime stabilă S în G cu $|S| \geqslant j$ dacă şi numai dacă H este Hamiltonian.

Fie $k = |V| - j$. Să presupunem că $k > 0$ (pentru a evita cazurile banale)

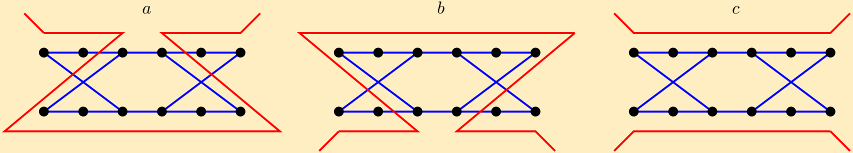
(i) Fie $A = \{a_1, a_2, \ldots, a_k\}$ o mulţime of k de noduri distincte.

(ii) Pentru fiecare muchie $e = uv \in E(G)$, considerăm graful $G'_e = (V'_e, E'_e)$ cu $V'_e = \{(w, e, i) : w \in \{u, v\}, i = \overline{1, 6}\}$ şi $E'_e = \{(w, e, i)(w, e, i+1) : w \in \{u, v\}, i = \overline{1, 5}\} \cup \{(u, e, 1)(v, e, 3), (u, e, 3)(v, e, 1), (u, e, 4)(v, e, 6), (u, e, 6)(v, e, 4)\}$.

Demonstraţie (continuare).



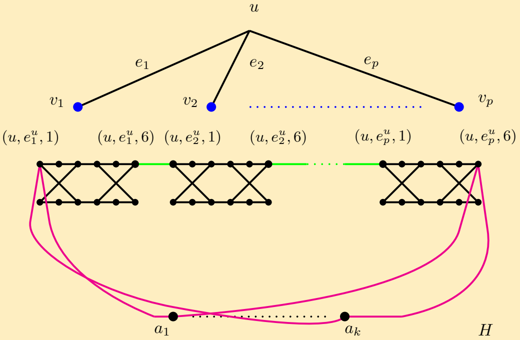
Graful are proprietatea că, dacă este un subgraf indus al unui graf Hamiltonian, H, şi nici unul dintre nodurile (w, e, i) cu $w \in \{u, v\}$ şi $i = \overline{2, 5}$ nu are alţi vecini în H, atunci singurele posibilităţi de parcurgere a lui ${G'}_e$ pe un circuit Hamiltonian sunt:



Demonstraţie (continuare). Astfel, dacă circuitul Hamiltonian "intră" în G'_e printr-un nod corespunzând lui u ($(u, e, 1)$ sau $(u, e, 6)$) atunci "părăseşte" graful G'_e de asemeni printr-un nod corespunzând lui u.

(iii) Pentru fiecare nod $u \in V$, considerăm (într-o ordine arbitrară) muchiile incidente în G cu u: $e_1^u = uv_1, e_2^u = uv_2, \ldots, e_p^u = uv_p$ ($p = d_G(u)$). Fie $E'_u{}' = \{(u, e_i^u, 6)(u, e_{i+1}^u, 1) \; : \; i = \overline{1, p-1}\}$ şi $E'_u{}'' = \{a_i(u, e_1^u, 1), a_i(u, e_p^u, 6) \; : \; i = \overline{1, k}\}$

Demonstraţie (continuare). Graful H are $V(H) = A \cup \left(\bigcup_{e \in E} V'_e \right)$ şi $E(H) = \left(\bigcup_{e \in E} E'_e \right) \cup \left(\bigcup_{u \in V} (E''_u \cup E'''_u) \right)$. Evident, poate fi construit din G în timp polinomial (în n).



Demonstraţie (continuare). Acum arătăm că există o mulţime stabilă în G cu cel puţin j noduri dacă şi numai dacă H este Hamiltonian.
"$\Leftarrow$" Dacă H este Hamiltonian, atunci există C un circuit Hamiltonian în H. Deoarece A este o mulţime stabilă în H, A descompune circuitul C în exact k drumuri intern disjuncte: $D_{a_{i_1} a_{i_2}}, D_{a_{i_2} a_{i_3}}, \ldots, D_{a_{i_k} a_{i_1}}$.
Fie $D_{a_{i_j} a_{i_{j+1}}}$ un astfel de drum ($j+1 = 1+(j \pmod k)$). Din construcţia lui H, urmează că primul nod după a_{i_j} pe acest drum va fi $(v_{i_j}, e_1^{v_{i_j}}, 1)$ sau $(v_{i_j}, e_p^{v_{i_j}}, 6)$, unde $p = d_G(v_{i_j})$, $v_{i_j} \in V$.
După aceasta, $D_{a_{i_j} a_{i_{j+1}}}$ va intra în componenta G'_{e_1} sau G'_{e_p} care va fi părăsită printr-un nod corespunzător lui v_{i_j}. Dacă următorul nod nu este $a_{i_{j+1}}$, va intra în componenta corespunzând următoarei muchii incidente cu v_{i_j}, care va fi părăsită de asemeni printr-un nod corespunzător lui v_{i_j}.

Demonstraţie (continuare). Urmează că fiecare drum $D_{a_{i_j}\, a_{i_{j+1}}}$ corespunde unui singur nod $v_{i_j} \in V$, astfel încât prima şi ultima muchie a lui $D_{a_{i_j}\, a_{i_{j+1}}}$ sunt $a_{i_j}(v_{i_j}, e_t^{v_{i_j}}, x)$, $a_{i_{j+1}}(v_{i_j}, e_{t'}^{v_{i_j}}, x')$, cu $t = 1$ şi $t' = d_G(v_{i_j})$, $x = 1$, $x' = 6$ sau $t = d_G(v_{i_j})$, $t' = 1$, $x = 6$, $x' = 1$.

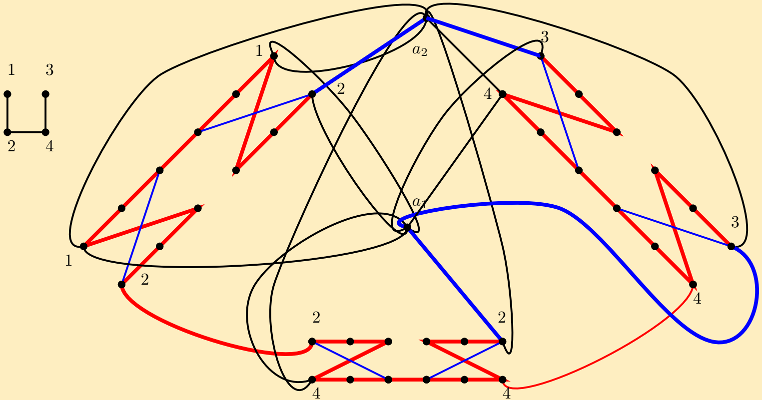
Urmează că nodurile v_{i_j} sunt distincte.

Fie $V^* = \{v_{i_1}, v_{i_2}, \ldots, v_{i_k}\}$. Deoarece C este un circuit Hamiltonian în H, urmează că, $\forall e \in E$, există un drum $D_{a_{i_j}\, a_{i_{j+1}}}$ care traversează G'_e, astfel există $v \in V^*$ incident cu e.

Deci $S = V \setminus V^*$ este o mulţime stabilă în G şi $|S| = j$.

Astfel, am arătat că, dacă H este Hamiltonian, atunci în G există o mulţime stabilă cu j noduri şi răspunsul la **SM** pentru instanţa G, j este da.

Demonstraţie (continuare).



Demonstraţie (continuare). " $\Rightarrow$ " Să presupunem că răspunsul la **SM** pentru instanţa G, j este da, astfel există S_0 o mulţime stabilă în G cu $|S_0| \geqslant j$. Există $S \subseteq S_0$ cu $|S| = j$. Fie $V^* = V \setminus S = \{v_1, v_2, \ldots, v_k\}$.

Considerăm în H pentru fiecare $e = uv \in E$:

- cele două drumuri din cazul (c) în G'_e (desenate mai sus) dacă $u, v \in V^*$.
- drumul din cazul (b) în G'_e (desenat mai sus) dacă $u \in V^*$ şi $v \notin V^*$.
- drumul din cazul (a) în G'_e (desenat mai sus) dacă $u \notin V^*$ şi $v \in V^*$.

La reuniunea tuturor acestor drumuri adăugăm muchiile $a_i(v_i, e_1^{v_i}, 1)$, $(v_i, e_1^{v_i}, 6)(v_i, e_2^{v_i}, 1), \ldots, (v_i, e_{p-1}^{v_i}, 6)(v_i, e_p^{v_i}, 1)$, $(v_i, e_p^{v_i}, 6)a_{i+1}$, (cu $p = d_G(v_i)$), pentru $i = \overline{1, k}$.

Ceee ce se obţine este un circuit Hamiltonian în H. $\square$

Traveling Salesman Problem - TSP Dat $G = (V, E)$ un graf şi $d : E \to \mathbb{R}_+$ o funcţie de pondere nenegativă pe muchiile sale, să se determine un circuit Hamiltonian, H_0, a. î. suma ponderilor pe muchiile lui H_0 să fie minimă (printre toate circuitele Hamiltoniene ale lui G).

Fie graful G poate fi o reţea constând dintr-o mulţime, V, de oraşe împreună c cu o mulţime, E, de rute directe între oraşe, funcţia d oferind, pentru fiecare muchie $uv \in E$, $d(uv) =$ distanţa pe ruta directă între oraşele u şi v. Fixând un oraş de start v_0, circuitul Hamiltonian H_0 reprezintă cel mai scurt traseu care vizitează toate oraşele exact odată (cu excepţia lui v_0) pe care îl poate parcurge un comis-voiajor plecând din v_0 şi întorcându-se în v_0.

Aceasta este probabil, cea mai studiată problemă de optimizare NP-hard!

Considerăm următoarea formulare echivalentă a acestei probleme.

TSP Dat $n \in \mathbb{N}$ ($n \geqslant 3$) şi $d : E(K_n) \to \mathbb{R}_+$, să se determine un circuit Hamiltonian, H_0, în K_n, cu $d(H_0)$ minim printre toate circuitele Hamiltoniene ale lui K_n, unde

$$d(H_0) = \sum_{e \in E(H_0)} d(e).$$

Dacă graful G, pe care trebuie să rezolvăm TSP, nu este graful complet K_n, atunci putem introduce muchiile lipsă cu o pondere foarte mare, $M \in \mathbb{R}_+$, unde $M > |V| \cdot \max\limits_{e \in E(G)} d(e)$.

Aceasta este formularea problemei **TSP** simetrice, o problemă similară (asimetrică) poate fi considerată pentru cazul când G este un digraf.

În studiul complexităţii timp a acestei probleme, vom considera $d(e) \in \mathbb{N}$, pentru fiecare muchie e.

Problema de decizie asociată este

DTSP

Instanţă: $n \in \mathbb{N}(n \geqslant 3)$, $d : E(K_n) \to \mathbb{N}$ şi $B \in \mathbb{N}$.

Întrebare: Există un circuit Hamiltonian, H_0, în K_n, a. î. $d(H_0) \leqslant B$?

Teorema 2

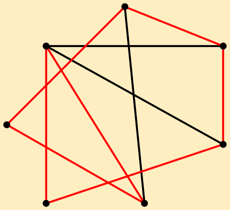
CH $\leqslant_P$ **DTSP**.

Demonstraţie. Fie $G = (V, E)$ $(|V| = n)$ o instanţă a problemei **CH**. Construim în timp polinomial o instanţă, $d : E(K_n) \to \mathbb{N}$ şi $B \in \mathbb{N}$, a problemei **DTSP** astfel încât există un circuit Hamiltonian în K_n de pondere totală cel mult B dacă şi numai dacă G este graf Hamiltonian.

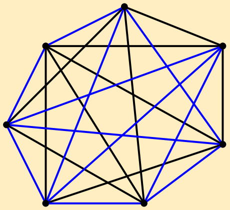
Fie

$$d(vw) = \begin{cases} 1, & \text{dacă } vw \in E(G) \\ 2, & \text{dacă } vw \in E(\overline{G}) \end{cases} \quad \text{şi } B = n.$$

Atunci, există în K_n un circuit Hamiltonian de pondere $\leqslant n$ dacă şi numai dacă există un circuit Hamiltonian C în K_n astfel încât $\forall e \in E$ $d(e) = 1$, adică, dacă şi numai dacă G are un circuit Hamiltonian:



G



weight 2
weight 1

Urmează că **TSP** este o problem ă **NP**-hard. □

O abordare posibilă este să considerăm un algoritm de aproximare $\mathcal{A}$, care să construiască, în timp polinomial, pentru fiecare instanţă **TSP**, un circuit Hamiltonian $H_{\mathcal{A}}$ al lui K_n, o aproximare a soluţiei optime H_o. Calitatea aproximării poate fi exprimată folosind următorul raport:

$$R_{\mathcal{A}}(n) = \sup_{d:E(K_n)\to\mathbb{R}_+, d(H_o)\neq 0} \frac{d(H_{\mathcal{A}})}{d(H_o)}$$

$$R_{\mathcal{A}} = \sup_{n\geqslant 3} R_{\mathcal{A}}(n).$$

Evident, algoritmul de aproximare $\mathcal{A}$ este util numai dacă $R_{\mathcal{A}}$ este finită. Din nefericire, dacă funcţia de pondere d este arbitrară, condiţia ca $R_{\mathcal{A}}$ să fie finită este la fel de dificilă ca şi rezolvarea exactă a **TSP**. Mai precis, avem următorul rezultat:

Teorema 3

Dacă există un algoritm de aproximare în timp polinomial A pentru **TSP** a. î. $R_{\mathcal{A}} < \infty$, atunci problema **CH** poate fi rezolvată în timp polinomial.

Demonstraţie. Fie A un algoritm de aproximare în timp polinomial cu $R_{\mathcal{A}} < \infty$. Există $k \in \mathbb{N}$ astfel încât $R_{\mathcal{A}} \leqslant k$.
Fie $G = (V, E)$ un graf arbitrar, instanţă a **CH**. Dacă $n = |V|$, atunci considerăm $d : E(K_n) \to \mathbb{N}$ definită prin

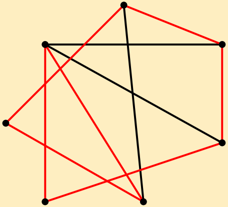
$$d(uv) = \begin{cases} 1, & \text{dacă } uv \in E(G) \\ kn, & \text{dacă } uv \notin E(G) \end{cases}.$$

Evident, G este Hamiltonian dacă şi numai dacă H_o, soluţia optimă a acestei instanţe a **TSP**, satisface $d(H_o) = n$.

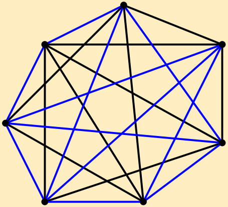
Aplicăm $\mathcal{A}$ pentru a rezolva aproximativ această instanţă a **TSP**.

- Dacă $d(H_{\mathcal{A}}) \leqslant kn$, atunci $d(H_{\mathcal{A}}) = n$ şi $H_{\mathcal{A}} = H_0$.
- Dacă $d(H_{\mathcal{A}}) > kn$, atunci $d(H_0) > n$. Într-adevăr, presupunând că $d(H_0) = n$, avem $\dfrac{d(H_{\mathcal{A}})}{d(H_0)} \leqslant k$, astfel $d(H_{\mathcal{A}}) \leqslant kd(H_0) = kn$, contradicţie.

Urmează că G este Hamiltonian dacă şi numai dacă $d(H_{\mathcal{A}}) \leqslant kn$, şi, deoarece A rulează în timp polinomial, urmează că **CH** poate fi rezolvată în timp polinomial. $\square$



G



Remarcă

Teorema 3 poate fi formulată echivalent astfel:

Teoremă

Dacă $\mathbf{P} \neq \mathbf{NP}$, atunci nu există algoritm de aproximare în timp polinomial $\mathcal{A}$ cu $R_{\mathcal{A}} < \infty$.

Teorema 4

(Christofides,1976) Dacă in **TSP** funcţia de pondere d satisface

$$\forall u, v, w \in V(K_n) \text{ distincte, } d(uv) \leqslant d(uw) + d(wv),$$

atunci există un algoritm de aproximare în timp polinomial $\mathcal{A}$ cu $R_{\mathcal{A}} = 3/2$.

Demonstraţie. Fie $\mathcal{A}$ următorul algoritm:

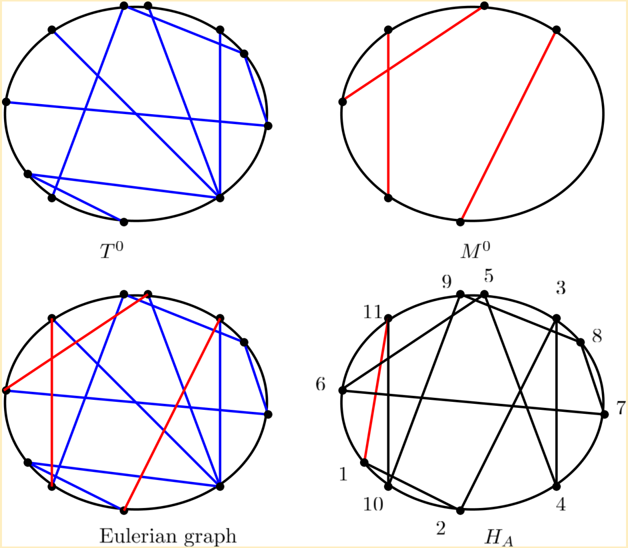
- Determină T^0, mulţimea de muchii a unui arbore parţial de cost minim din K_n (costul unei muchii e va fi $d(e)$) (acest pas ia un timp polinomial folosind orice algoritm pentru MST).
- Determină M^0, un cuplaj perfect de pondere minimă în subgraful indus în K_n de mulţimea de noduri de grad impar din arborele parţial de cost minim T^0 (acest pas ia un timp polinomial folosind orice algoritm pentru determinarea unui cuplaj de cardinal maxim).

Demonstraţie (continuare)

- În multigraful obţinut din $\langle T^0 \cup M^0 \rangle_{K_n}$, prin duplicarea muchiilor din $T^0 \cap M^0$ (este conex şi are toate nodurile de grad par) determină un parcurs Eulerian închis, $(v_{i_1}, v_{i_2}, \ldots, v_{i_1})$. Eliminăm toate apariţiile multiple ale nodurilor interne pentru a obţine un circuit Hamiltonian $H_\mathcal{A}$ in K_n cu mulţimea de muchii $H_\mathcal{A} = \{v_{j_1}v_{j_2}, v_{j_2}v_{j_3}, \ldots, v_{j_n}v_{j_1}\}$ (amândouă construcţiile se fac în $\mathcal{O}(n^2)$).

Fie $H_\mathcal{A}$ soluţia aproximativă a **TSP** dată de algoritmul lui Christofides. Fie $m = \lfloor n/2 \rfloor$ şi H_o o soluţie optimă. Arătăm (după Cornuejols & Nemhauser) că

$$\forall n \geqslant 3,\ d(H_\mathcal{A}) \leqslant \frac{3m-1}{2m} d(H_o).$$


T^0
M^0
Eulerian graph
9
5
3
11
8
6
7
1
10
4
2
H_A

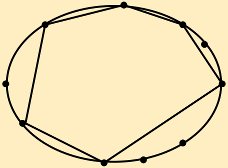
Demonstraţie (continuare) Fie $H_o = \{v_1v_2, v_2v_3, \ldots, v_nv_1\}$ (dacă e necesar, redenumim nodurile).

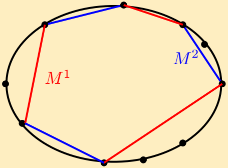
Fie $W = \{v_{i_1}, v_{i_2}, \ldots, v_{i_{2k}}\}$ mulţimea nodurilor de grad impar din $\langle T^0\rangle_{K_n}$, $i_1 < i_2 < \ldots < i_{2k}$. Fie $H = \{v_{i_1}v_{i_2}, v_{i_2}v_{i_3}, \ldots, v_{i_{2k-1}}v_{i_{2k}}, v_{i_2k}v_{i_1}\}$ circuitul generat de W în K_n. Aplicând în mod repetat inegalitatea triunghiulară, obţinem $d(H) \leqslant d(H_o)$, (ponderea fiecărei corzi, $d(v_{i_j}v_{i_{j+1}})$ este majorată de suma ponderilor de pe muchiile lui H_o care unesc extremităţile corzii $v_{i_j}v_{i_{j+1}}$).

Deoarece H este un circuit par, poate fi scris ca reuniunea a două cuplaje perfecte în $[W]_{K_n}$, $M^1 \cup M^2$. Să presupunem că $d(M^1) \leqslant d(M^2)$.

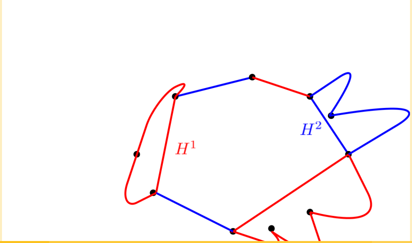
Din modul de alegere al lui M^0, avem $d(M^0) \leqslant d(M^1) \leqslant (1/2)[d(M^1) + d(M^2)] = (1/2)d(H) \leqslant (1/2)d(H_o)$. Fie $\alpha \in \mathbb{R}_+$ a. î. $d(M^0) = \alpha d(H_o)$. Evident, $0 < \alpha \leqslant 1/2$.

Demonstraţie (continuare)





Descompunem H_o în $H^1 \cup H^2$ luând în H^i muchiile din H_o care conectează extremităţile fiecărei corzi din M^i: $(v_{i_j}v_{i_{j+1}} \in M^i \Rightarrow v_{i_j}v_{i_j+1}, \ldots v_{i_{j+1}-1}v_{i_{j+1}} \in H^i)$.



Demonstraţie (continuare) Din inegalitatea triunghiulară, $d(H^i) \geqslant d(M^i)$, $i = 1, 2$. Cel puţin unul dintre H^1 sau H^2 are cel mult $m = \lfloor n/2 \rfloor$ muchii. Să presupunem că H^1 are această proprietate. Deoarece $d(H^1) \geqslant d(M^1) \geqslant d(M^0) = \alpha d(H_o)$, urmează că există $e \in H^1$ astfel încât $d(e) \geqslant (\alpha/m) d(H_o)$.

Fie T un arbore parţial de cost minim obţinut din H_o prin ştergerea unei muchii de pondere maximă. Avem $d(T) = d(H_o) - \max\limits_{e \in E(H_o)} d(e) \leqslant d(H_o) - (\alpha/m) d(H_o)$. Deoarece T^0 este arbore parţial de cost minim în K_n urmează că $d(T^0) \leqslant d(H_o)(1 - \alpha/m)$.

Folosind inegalitatea triunghiulară obţinem

$$d(H_{\mathcal{A}}) \leqslant d(T^0) + d(M^0) \leqslant d(H_o) \left(1 - \frac{\alpha}{m}\right) + \alpha d(H_o) =$$

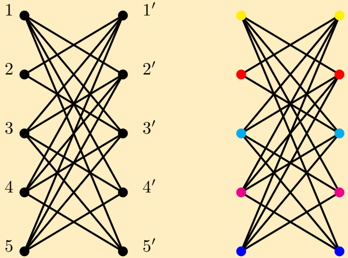
$$= \left(1 + \frac{\alpha(m-1)}{m}\right) d(H_o) \overset{\alpha \leqslant 1/2}{\leqslant} \frac{3m-1}{2m} d(H_o), \forall n \geqslant 3. \ \square$$

Fie $G = (V, E)$ un graf, $V = \{1, 2, \ldots, n\}$ şi fie π o permutare a lui V.
Construim o colorare a nodurilor $c : V \to \{1, \ldots, \chi(G, \pi)\}$.

$c(\pi_1) \leftarrow 1$; $\chi(G, \pi) \leftarrow 1$; $S_1 \leftarrow \{\pi_1\}$;
for $(i = \overline{2, n})$ **do**
  $j \leftarrow 0$;
  repeat
    $j{+}{+}$; $v \leftarrow$ primul nod (în π), din S_j a. î. $\pi_i v \in E(G)$;
    if $(\exists v)$ **then**
      $\mathit{first}(\pi_i, j) \leftarrow v$;
    else
      $\mathit{first}(\pi_i, j) \leftarrow 0$; $c(\pi_i) \leftarrow j$; $S_j \leftarrow S_j \cup \{\pi_i\}$;
    end if
  until $(\mathit{first}(\pi_i, j) = 0$ sau $j = \chi(G, \pi))$
  if $(\mathit{first}(\pi_i, j) \neq 0)$ **then**
    $c(\pi_i) \leftarrow j + 1$; $S_{j+1} \leftarrow \{\pi_i\}$; $\chi(G, \pi) \leftarrow j + 1$;
  end if
end for

Să observă m că numărul de culori folosite de algoritm nu este mai mare decât $1 + \Delta(G)$. Urmează că $\chi(G) \leqslant 1 + \Delta(G)$.

$\chi(G, \pi)$ numărul de culori returnate de algoritmul de colorare greedy, poate fi cu oricât mai mare decât $\chi(G)$. De exemplu, fie G graful obţinut din graful complet bipartit $K_{n,n}$, cu mulţimea de noduri $\{1, 2, \ldots, n\} \cup \{1', 2', \ldots, n'\}$, prin ştergerea muchiilor $11', 22', \ldots, nn'$.



Dacă $\pi = (1, 1', 2, 2', \ldots, n, n')$, atunci algoritmul de colorare greedy returnează $c(1) = c(1')$, $c(2) = c(2') = 2$, $c(n) = c(n') = n$. Astfel, $\chi(G, \pi) = n$, pe când $\chi(G) = 2$.

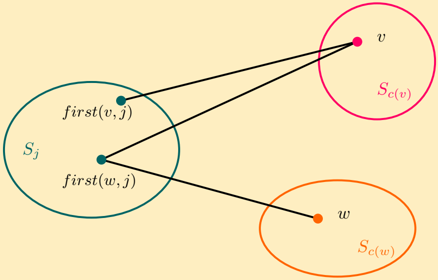
Pe de altă parte, pentru orice graf G, există o permutare π a nodurilor sale astfel încât $\chi(G, \pi) = \chi(G)$.

Într-adevăr, fie $S_1, S_2, \ldots, S_{\chi(G)}$ clasele de colorare ale unei colorări optimale, astfel încât fiecare S_i este o mulţime stabilă maximală (relativ la incluziune) în $G - \bigcup_{j=1}^{i-1} S_j$. Fie π o permutare care induce o ordonare astfel încât nodurile apar în ordinea crescătoare a culorilor lor. Atunci, $\chi(G, \pi) = \chi(G)$.

O condiţie suffcientă pentru corectitudinea algoritmului de colorare greedy:

Teoremă

Dacă, $\forall vw \in E$ şi $\forall j < \min\{c(v), c(w)\}$ astfel încât $\mathit{first}(v,j) < \mathit{first}(w,j)$ în ordonarea dată de π, avem $\mathit{first}(w,j)v \in E$, atunci $\chi(G,\pi) = \chi(G)$.



Condiţia de mai sus poate fi testată în $\mathcal{O}(m)$ ca un ultim pas al algoritmului. Dacă este îndepinită, atunci avem un certificat pentru optimalitatea numărului găsit de culori.

Teorema de mai sus se poate demonstra arătând că dacă condiţia enunţată are loc, atunci $\chi(G, \pi) = \omega(G) \leqslant \chi(G)$. Pe de altă parte, se ştie că există grafuri G pentru care $\chi(G) - \omega(G)$ este arbitrar de mare; pentru un astfel de graf G nici o permutare π nu satisface condiţia din teoremă.

Exerciţiul 1. Regiunile (inclusiv cea infinită) formate de n cercuri în plan pot fi colorate cu două culori astfel ca două regiuni care au în comun un arc de cerc (nedegenerat) să fie colorate diferit.

Exerciţiul 2. Fie $G = (V, E)$ un graf fără circuite impare disjuncte. Arătaţi că G este 5-colorabil.

Exerciţiul 3. Pentru un graf dat H definim **gradul mediu** ca $ad(H) = \frac{2|E(H)|}{|V(H)|}$. Pentru un graf G, **gradul mediu maxim** este

$$mad(G) = \max\{ad(H) \ :\ H \text{ subgraf indus al lui } G\}$$

Fie $k \geqslant 3$ un întreg. Arătaţi că, dacă un graf G are numărul cromatic strict mai mare decât k iar gradul său mediu maxim cel mult k, atunci G conţine un subgraf indus k-regulat.

Exerciţiul 4.

(a) Arătaţi că orice graf poate colorat (pe noduri) cu $\Delta(G) + 1$ culori.

(b) Considerăm următorul algoritm recursiv pentru colorarea nodurilor unui graf 3-colorabil cu n noduri:

color(G) {
if ($\Delta(G) \leqslant \sqrt{n}$) **then**
 colorează toate nodurile lui G cu ($\Delta(G) + 1$) culori noi;
else
 fie $x_0 \in V(G)$ a. î. $d_G(x_0) = \Delta(G)$;
 colorează x_0 cu o culoare nouă;
 colorează nodurile lui $[N_G(x_0)]_G$ cu două culori noi;
 $G \leftarrow G - (\{x_0\} \cup N_G(x_0))$;
 color(G);
end if }

Arătaţi că algoritmul de mai sus foloseşte $\mathcal{O}(\sqrt{n})$ culori.

Exerciţiul 5. Considerăm următoarea problemă:

Set Cover

Instanţă: O mulţime nevidă X, o familie de submulţimi ale lui X: $\mathcal{F} = \{X_1, \ldots, X_n\}$ şi $k \in \mathbb{N}^*$.

Întrebare: X poate fi acoperit cu cel mult k submulţimi din $\mathcal{F}$?

Considerăm următoarea heuristică (de tip greedy) pentru rezolvarea acestei probleme

$\mathcal{F}' \leftarrow \varnothing$;
while ($\exists x \in X$ neacoperit de $\mathcal{F}'$) **do**
  fie X_i din $\mathcal{F}$ cu un număr maxim de elemente neacoperite;
  $\mathcal{F}' \leftarrow \mathcal{F}' \cup \{X_i\}$;
end while
return $\mathcal{F}'$;

Arătaţi că dacă m este cardinalul unei subfamilii optime a lui $\mathcal{F}$, atunci algoritmul de mai sus returnează o subfamilie cu cel mult $m \ln n$ submulţimi ($n = |X|$).

Exerciţii pentru seminarul din următoarea săptămână

Exerciţiul 6. Fie $G = (V, E)$ un graf cu n noduri şi m muchii. O ordonare $\{x_1, x_2, \ldots, x_n\}$ a nodurilor lui G este ***k*-mărginită** dacă în digraful $\vec{G}$, obţinut din G prin înlocuirea fiecărei muchii $x_i x_j$ cu arcul $x_{\min\{i,j\}}, x_{\max\{i,j\}}$, avem $d^+_{\vec{G}}(x) \leqslant k, \forall\, x \in V$.

(a) Descrieţi un algoritm care să testeze în $\mathcal{O}(n+m)$ dacă G, are o ordonare k-mărginită, pentru un $k \in \mathbb{N}^*$.

(b) Folosiţi algoritmul de mai sus pentru a determina în $\mathcal{O}((n+m)\log n)$ numărul

$$o(G) = \min\{k \in \mathbb{N} \;:\; G \text{ are o ordonare } k\text{-mărginită.}\}$$

(c) Arătaţi că orice graf G are o colorare a nodurilor cu $o(G)+1$ culori.

Exerciţii pentru seminarul din următoarea săptămână

Exerciţiul 7. Un algoritm greedy pentru colorarea nodurilor grafului $G = (V, E)$ este următorul: mai întâi, alege o D-ordonare a nodurilor $V = \{v_1, v_2, \ldots, v_n\}$, i. e., $d_G(v_1) \geqslant d_G(v_2) \geqslant \ldots \geqslant d_G(v_n)$, apoi v_i primeşte cea mai mică culoare nefolosită de vecinii săi deja coloraţi. Considerăm următoarea problemă de decizie

3GCOL

instanţă: $G = (V, E)$ un graf.

întrebare: Are G o D-ordonare a. î. euristica de mai sus foloseşte cel mult 3 culori?

Arătaţi că **3COL** $\leqslant_P$ **3GCOL**.

Exerciţiul 8. Fie $G = (V; E)$ un graf cu $V = \{1, 2, \ldots, n\}$ şi $\omega(G) = 2$. Definim un graf nou $M(G)$ considerând reuniune disjunctă a lui G cu $K_{1,n}$ (a cărui bipartiţie este $(\{0\}, \{1', 2', \ldots, n'\})$) şi adăugând toate muchiile $\{i'j, ij' : ij \in E(G)\}$.

(a) Arătaţi că $\omega(M(G)) = 2$ şi $\chi(M(G)) = \chi(G) + 1$.

(b) Arătaţi că, pentru orice $p \in \mathbb{N}^*$, există un graf K_3-free cu numărul cromatic p.

Exerciţiul 9. Fie $\mathcal{S}$ o societate formată din n indivizi. Fiecare individ, $i \in \mathcal{S}$, cunoaşte o submulţime $c(i) \subseteq \mathcal{S} \setminus \{i\}$ formată din alţi indivizi. Doi indivizi diferiţi $i, i' \in \mathcal{S}$ nu pot face parte din acelaşi juriu dacă unul dintre ei îl cunoaşte pe celălalt (putem avea jurii formate dintr-un singur membru).

Arătaţi că dacă fiecare individ cunoaşte cel mult k alţi indivizi ($|c(i)| \leqslant k$, $\forall i \in \mathcal{S}$), atunci există o familie de cel mult $(2k + 1)$ jurii disjuncte care acoperă întreaga societate.