Tema 3

Renghiuc Bianca Elena, Chichirău Claudiu-Constantin 2A2

05.01.2023

1 Execiţiul 1

a) Componentele conexe ale grafului parțial $G' = (S \cup T, M \cup M')$ sunt o combinație de cicluri și de arbori. Știind că graful bipartit $G = (S, T; E)$ este format din două seturi de noduri ce satisfac condiția $A \subseteq S(M)$ și $B \subseteq S(M')$, înseamnă că fiecare nod din mulțimea A este conectat la cel puțin un nod din T prin intermediul unei muchii din mulțimea M iar fiecare nod din mulțimea B este conectat la cel puțin un nod din S prin intermediul unei muchii din mulțimea M'.

Ceea ce înseamnă că între fiecare nod din mulțimea S și între fiecare nod din mulțimea T există un drum în graful parțial $G' = (S \cup T, M \cup M')$. Drum, care poate fi ori ciclu, ori arbore, în funcție de modul în care muchiile din M și M' sunt conectate între ele.

b) Pentru a arăta că există un cumplaj M_0 care saturează toate nodurile din $A \cup B$ ($A \cup B \subseteq S(M_0)$) trebuie să găsim un cuplaj M_0 astfel încât fiecare nod din mulțimea $A \cup B$ să fie conectat la un not din mulțimea S sau T.

Astfel, considerăm toate muchiile din M care conectează nodurile din mulțimea A la nodurile din mulțimea T să facă parte din cuplajul M_0. În acest caz am reușit ca toate nodurile din mulțimea A să fie conectate la noduri din mulțimea T.

În continuare dacă adăugăm în cuplajul M_0 muchiile din M' care conectează nodurile din mulțimea B la nodurile din mulțimea S, obținem că toate nodurile din mulțimea $A \cup B$ sunt conectate la noduri din partițiile T sau S în cuplajul M_0.

Astfel, am arătat că există un cuplaj M_0 care saturează toate nodurile mulțimii $A \cup B$.

2 Exerciţiul 2

a) **I.** Pentru a demonstră că toate clasele V_i din D-partiția $(V_1, V_2, ..., V_p)$ sunt mulțimi stabile în G trebuie să arătăm că dacă $\exists v \in V_i$ și $\exists u \in V_j$, unde $(i \neq j)$ atunci muchia {v,u} $\notin E$.

Știind că $d^-_{\vec{G}}(v) = 0$, putem trage concluzia că nu există niciun drum de la nodul v la nodul u în orientarea aciclică $\vec{G}$. Dar, dacă am avea o muchie de la partiția V_i la V_j ar însemna că există un drum de la V_j la V_i în orientarea aciclică $\vec{G}$, ceea ce înseamnă că există o muchie (de la V_j la V_i) care intră în $\vec{G}$. Astfel, se contrazice faptul că V_i conține doar noduri fără muchii care intră în $\vec{G}$ - $\bigcup_{i-1}^{j=1} V_j$.

Același lucru se aplică și pentri cazul când avem o muchie de la V_j la V_i în orientarea aciclică $\vec{G}$. Dacă am avea o muchie de la partiția V_j la V_i ar însemna că există un drum de la V_i la V_j în orientarea aciclică $\vec{G}$, ceea ce înseamnă că există o muchie (de la V_i la V_j) care intră în $\vec{G}$. Astfel, se contrazice faptul că V_j conține doar noduri fără muchii care intră în $\vec{G}$ - $\bigcup_{i-1}^{j=1} V_j$.

Astfel, dacă avem două noduri în G, unul în partiția V_i și unul în partiția V_j, unde $(i \neq j)$, atunci ele nu sunt conectate prin o muchie în $G \rightarrow$ toate clasele V_i sunt mulțimi stabile în G.

II. Pentru a arăta că p = c($\vec{G}$) + 1 > χ(G), trebuie să demonstrăm inițial că χ(G) = numărul de mulțimi V_i din D-partiție. Prima dată vom colora nodurile în G folosind culorile mulțimilor V_i. Deoarece am demonstrat că mulțimile V_i sunt stabile în G, dacă avem 2 noduri adiacente $v \in V_i$ și $u \in V_j$, unde $(i \neq j)$, atunci înseamnă că ele vor avea culori diferite. Deoarece numărul cromatic al lui G reprezintă minimul de culori necesare pentru a putea colora nodurile astfel încât orice două noduri adiacente să aibă culori diferite, acest lucru demonstreaza afirmația de mai sus.

Acum vom demonstra că c($\vec{G}$) + 1 > χ(G). Știm deja că numărul cromatic al lui G este minumul de culori necesare pentru a putea colora nodurile din G astfel încât oricum am alege două noduri adiacente diferite, acestea să aibă culori diferite. În acest caz, dacă vom folosi culorile mulțimilor V_i din *D-partiție*, vom putea colora nodurile astfel încât orice două noduri adiacente să aibă culori diferinte deoarece am demonstrat că partițiile V_i sunt stabile în G. Astfel, χ(G) $\leq$ numărul mulțimilor V_i din *D-partiție*.

Știm că c($\vec{G}$) reprezintă lungimea maximă a unui drum direct în $\vec{G}$. Deoarece $\vec{G}$ este o orientare aciclică a lui G, acest drum nu poate forma un ciclu. Astfel c($\vec{G}$) reprezintă numărul maxim de mulțimi V_i pe care trebuie să le traversăm pentru a ajunge de la un nod la altul în $\vec{G}$.

Orice mulțime V_i pe care o vom adăuga în plus la acest drum vom avea nevoie de o culoare suplimentară pentru a colora nodul final în G. Astfel, c($\vec{G}$) + 1 > χ(G).

b) Vom pleca de la existența unei colorări optime a nodurilor lui G, care corespunde unei partiții (V_1, V_2, ..., V_p) cu mulțimi stabile în G.

Construim o orientare aciclică $\vec{G}$ a lui G astfel încât: arcele $\vec{xy}$ din G, unde $x \in V_i$ și $y \in V_j$, $i < j$, vor avea orietarea de la x la y în $\vec{G}$; iar arcele $\vec{xy}$ din G, unde x, y $\in$V$_i$, vor avea orientarea, în $\vec{G}$, care respectă ordinea topologică din urma colorării. Orientarea $\vec{G}$ este aciclică, deoarece pentru a exista un ciclu trebuie să existe un arc de la o mulțime V_i la o multime V_j, cu $i > j$, ceea ce duce la contradicție (arcele sunt orientate de la mulțimele V_i la V_j, cu $i < j$).

Dacă există două noduri care au aceleași muțimi stabile, atunci aceste două noduri pot fi colorate cu aceeași culoare, ceea ce înseamnă că nu avem nevoie de o culoare în plus pentru a colora aceste noduri. Prin urmare, avem nevoie de mai puține culori decât în cazul în care cele două noduri ar avea mulțimi stabile diferite. Deci, pentru a minimiza numărul de culori necesare pentru a colora graful G, vom avea întotdeauna cel puțin o mulțime V_j care conține cel puțin două noduri care au aceleași mulțimi stabile.

Astfel, avem nevoie de cel puțin o culoare în plus pentru a colora aceste noduri, deci numărul cromatic al lui G este cel puțin c($\vec{G}$) + 1.

Deci există o orientare aciclică $\vec{G}$ a lui G astfel încât $\chi(G) >$ c($\vec{G}$) + 1.

Prin urmare, numărul cromatic al lui G este egal cu minimul dintre c($\vec{G}$) + 1, unde $\vec{G}$ aparține familiei orientărilor aciclice ale lui G.

3 Exercițiul 3

a) Pentru a demonstra că $v(x) = \sum_{t \in Y} (\sum_{vt \in E} x_{vt} - \sum_{tu \in E} x_{tu})$ vom folosi faptul că $m - fluxul$ se conservă în fiecare nod din afara mulțimii $X \cup Y$. Aceasta înseamnă că suma fluxurilor care intră într-un nod din afara mulțimii $X \cup Y$ este egală cu suma fluxurilor care ies din acel nod.

Mai concret, dacă luăm orice nod s din mulțimea X, putem scrie:

$$\sum_{su \in E} x_{su} - \sum_{vs \in E} x_{vs} = 0$$

Aceasta înseamnă că suma fluxurilor care intră în nodul s este egală cu suma fluxurilor care ies din acel nod.

Acum, pentru a arăta că $v(x) = \sum_{t\in Y}(\sum_{vt\in E} x_{vt} - \sum_{tu\in E} x_{tu})$ să presupunem că avem un $m - flux$ x în rețeaua $R = (G, X, Y, c)$ și să notăm cantitățile care apar în ambele expresii pentru v(x) astfel:

$$v(x) = \sum_{s\in X}(\sum_{su\in E} x_{su} - \sum_{vs\in E} x_{vs}) = \sum_{s\in X}(A - B)$$

$$iar$$

$$\sum_{t\in Y}(\sum_{vt\in E} x_{vt} - \sum_{tu\in E} x_{tu}) = \sum_{t\in Y}(C - D)$$

Pentru a arăta că $v(x) = \sum_{t\in Y}(\sum_{vt\in E} x_{vt} - \sum_{tu\in E} x_{tu})$, trebuie să arătăm că $A - B = C - D$.

Notăm cu E_+ și E_- mulțimile de arce de intrare și de ieșire ale unui nod s respectiv t, astfel:

E_+(s) $= \{e \in E : e = (u, s)$ pentru un nod u $\}$
E_-(s) $= \{e \in E : e = (s, u)$ pentru un nod u $\}$

Iar acum avem mulțimile:
$A = \sum_{su\in E_+(s)} x_{su}$
$B = \sum_{vs\in E_-(s)} x_{vs}$
$C = \sum_{vt\in E_+(t)} x_{vt}$
$D = \sum_{tu\in E_-(t)} x_{tu}$

Notăm cu G_+ și G_- mulțimile de arce de intrare și de ieșire ale grafului G, astfel:
$G_+ = \{(u, v) \in E : u \neq v\}$
$G_- = \{(v, u) \in E : u \neq v\}$

Astfel, avem:
E_+(s) $\subseteq G_+$
E_-(s) $\subseteq G_-$
E_+(t) $\subseteq G_+$
E_-(t) $\subseteq G_-$

Deci:
$A = \sum_{su\in E_+(s)} x_{su} \leq \sum_{(u,v) \in G_+} x_{uv}$

$$B = \sum_{vs \in E_-(s)} x_{vs} \leq \sum_{(v,u) \in G_-} x_{vu}$$
$$C = \sum_{vt \in E_+(t)} x_{vt} \leq \sum_{(u,v) \in G_+} x_{uv}$$
$$D = \sum_{tu \in E_-(t)} x_{tu} \leq \sum_{(v,u) \in G_-} x_{vu}$$

Deoarece $m - fluxul$ se conservă în fiecare nod din afara mulțimii $X \cup Y$, avem:

$$\sum_{(u,v) \in G_+} x_{uv} = \sum_{(v,u) \in G_-} x_{vu}$$

Deci:
$$A \leq \sum_{(u,v) \in G_+} x_{uv} = \sum_{(v,u) \in G_-} x_{vu} = B \; C \leq \sum_{(u,v) \in G_+} x_{uv} = \sum_{(v,u) \in G_-} x_{vu} = D$$

Prin urmare:
$A - B \leq 0$
$C - D \leq 0$

De unde rezultă că $A - B = C - D$. Deci, am arătat că:

$$v(x) = \sum_{s \in X} \Big(\sum_{su \in E} x_{su} - \sum_{vs \in E} x_{vs} \Big) = \sum_{t \in Y} \Big(\sum_{vt \in E} x_{vt} - \sum_{tu \in E} x_{tu} \Big)$$

b) Un algoritm eficient pentru a determina un $m - flux$ de valoare maximă în R prin reducerea la cazul cu o singură sursă și o singură destinație ar putea avea următorii pași:

1. Adăugăm un nod nou, a, și îl legăm de toate nodurile din mulțimea X prin arce de capacitate infinită.
2. Adăugăm un nod nou, b, și îl legăm de toate nodurile din mulțimea Y prin arce de capacitate infinită.
3. Calculăm $m - fluxul$ de valoare maximă între nodul a și nodul b în graful obținut (G', X', Y', c'), folosind un algoritm de parcurgere în adâncime **DFS**.
4. M-fluxul de valoare maximă între nodurile a și b în graful pe care l-am obținut (G', X', Y', c') este $m - fluxul$ de valoare maximă în graful inițial (G, X, Y, c)

Complexitatea timp a acestui algoritm este $O(|V| \times |E|)$, deoarece parcurgerea în adâncime **DFS** are această complexitate.

Pentru a eficientiza mai bine algoritmul, putem calcula $m - fluxul$ de valoare maximă folosind algoritmul lui Ford-Fulkerson. În acest caz, *complexitatea timp* a algoritmului ar fi $O(|V| \times |E|^2)$, deoarece algoritmul lui Ford-Fulkerson are această complexitate.

4 Exerciţiul 4

a) Problema **FOREST** este o problemă de decizie din clasa **NP**, dacă îndeplinește următoarele două condiții:

   I. Problema **FOREST** trebuie rezolvată într-un număr polinomial de pași pentru o instanță H = (W,F).

   II. Trebuie să verificăm într-un număr *polinomial* de pași pentru o instanță H = (W,F) dacă o soluție dată este corectă.

   I. Va trebui sa gasim o partiție a lui W în k subgrafuri (toate k subgrafurile sunt păduri). Pentru a reuși să facem asta, vom folosi un *algoritm backtracking* care ne va da toate posibilele soluții. În acest mod, vom reuși să gasim o soluție într-un număr polinomial de pași pentru o instanță H = (W,F).

   II. Dacă ni se dă o soluție și trebuie să verificăm corectitudinea sa, atunci va trebui să verificăm dacă fiecare subgraf H_i=(W_i, F_i) din partiția (W_1, W_2, ..., W_k) este o pădure. Pentru a face acest lucru, într-un număr *polinomial* de pași pentru o instanță H = (W,F), vom verifica dacă fiecare subgraf este aciclic si neconectat(nu există muchii între arborii din subgraf). Pentru a realiza acest lucru, în timp polinomial, vom folosi un algoritm de căutare în adâncime, astfel verificând dacă există cicluri în graful H_i.

   În concluzie, problema **FOREST** este o problemă de decizie din clasa **NP**.

b) Pentru a demonstra că graful G este $3-$*colorabil* dacă și numai dacă H poate fi partiționat în trei păduri, trebuie să arătăm că:

   1. Dacă G este **3-colorabil**, atunci H poate fi partiționat în **trei păduri**.
   2. Dacă H poate fi partiționat în **trei păduri**, atunci G este **3-colorabil**.

   I. Demonstrăm că dacă graful G poate fi colorat în **trei culori** astfel încât niciun nod nu are aceeași culoare ca vecinii săi, atunci graful H poate fi partiționat în **trei păduri**, atfel:

   1. Colorăm fiecare nod al lui G într-una dintre cele trei culori (1, 2, sau 3).
   2. Creăm partițiile W_1, W_2, W_3, astfel:
      (a) W_1 conține toate nodurile colorate în culoarea 1, plus nodul x_1.
      (b) W_2 conține toate nodurile colorate în culoarea 1, plus nodul x_2.
      (c) W_3 conține toate nodurile colorate în culoarea 1, plus nodul x_3.
   3. Deoarece toate nodurile din G sunt conectate printr-un arc la cel puțin un nod x_i, și deoarece arcele $x_1\vec{x}_2$, $x_2\vec{x}_3$, $x_3\vec{x}_1$ formează un ciclu în H, fiecare nod din G va fi conectat la cel puțin un nod din W_1, W_2, sau W_3.

4. Deoarece fiecare nod din G este colorat într-o singură culoare și fiecare nod din W_1, W_2, sau W_3 are o singură culoare asociată (culoarea 1, 2 sau 3), atunci fiecare nod din G va fi conectat numai la noduri cu aceeași culoare din W_1, W_2, sau W_3.
5. Astfel, fiecare subgraf $[W_i]_H$ este un graf fără cicluri (o pădure), ceea ce înseamnă că H poate fi partiționat în trei păduri.

II. Demonstrăm că dacă H poate fi partiționat în **trei păduri**, atunci G este **3-colorabil**, atfel:

1. Dacă graful H poate fi partiționat în trei păduri, atunci putem alege o partiție (W_1, W_2, W_3) astfel încât orice subgraf $[W_i]_H$ să fie o pădure.
2. Dacă facem acest lucru, atunci fiecare nod din G va fi conectat numai la noduri din W_1, W_2, sau W_3 care au *aceeași culoare*.
3. Astfel, putem colora fiecare nod din G cu una dintre cele trei culori astfel încât să respectăm condiția ca fiecare nod să fie conectat numai la noduri care au aceeași culoare.
4. În acest caz, graful G poate fi colorat cu trei culori astfel încât să nu existe două noduri vecine care să aibă aceeași culoare.

În concluzie, am arătat că dacă G poate fi colorat în **trei culori** astfel încât să nu existe două noduri vecine care au aceeași culoare, atunci H poate fi partiționat în **trei păduri**, iar dacă H poate fi partiționat în **trei păduri**, atunci G poate fi colorat în **trei culori**.

c) Pentru a demonstra ca problema **FOREST** este **NP-completă**, trebuie să arătăm că este în **NP** și că poate fi redusă la o altă problemă **NP-completă**:

I. Problema **FOREST** este **NP**: Am demonstrat la punctul a).

II. Problema **FOREST** poate fi redusă la o altă problemă NP-completă (**3COL**): Pentru problema **3COL** avem instanța G=(V,E), iar pentru problema **FOREST** construim instanța H=(W,F).

• Dacă G are o 3-colorare, atunci vom putea crea o partiție (W_1, W_2, W_3) a lui W, astfel încât fiecare subgraf $[W_i]_H$ să fie o pădure. Acest lucru este demonstrat în demonstrația I de la punctul b).

• Dacă avem o partiție (W_1, W_2, W_3) a lui W, astfel încât fiecare subgraf $[W_i]_H$ să fie o pădure, atunci G are o 3-colorare. Acest lucru este demonstrat în demonstrația II de la punctul b).

Astfel, am arătat că dacă G are o 3-colorare, atunci H are o partiție care respectă condiția de a fi o pădure și, invers, dacă H are o partiție care respectă condiția de a fi o pădure, atunci G are o 3-colorare. Asta înseamnă că FOREST poate fi redus la **3COL**, de unde rezultă că **FOREST** este o problemă **NP-completă**.