תכנון אלגוריתמים 9.12.2020

הרצאה 16: זרימה

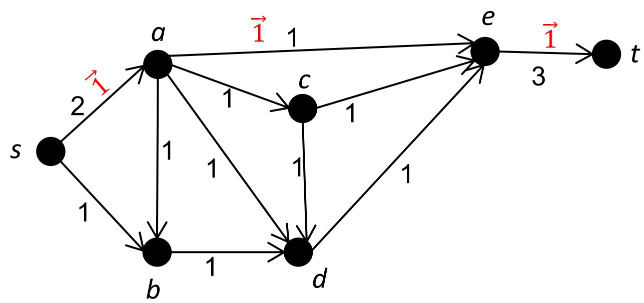
Textbook: Cormen, Leiserson, Rivest, Stein. Introduction to Algorithms.

מרצה: אורי שטמר

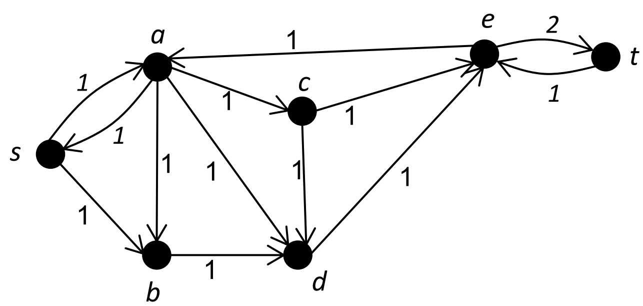
תזכורות

רשת זרימה N

וזרימה נוכחית f

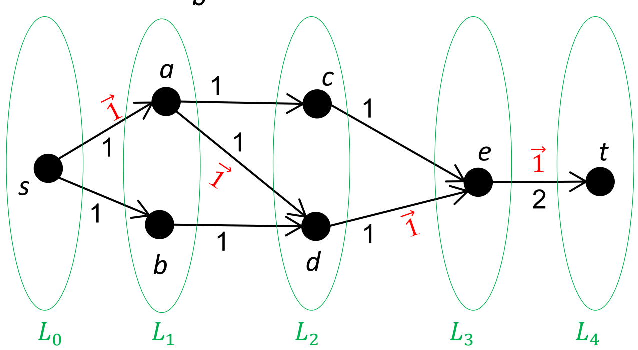


רשת שיורית N_f



רשת השכבות L_f

דוגמה לזרימה חוסמת



<u>האלגוריתם של דיניץ:</u>

(1) התחל מזרימה חוקית $f(u,v)=0$ לכל $u,v\in V$

(2) בנה את הרשת השיורית $N_f=\left(G_f=\left(V,E_f\right),c_f,s,t\right)$

(3) כל עוד יש מסלול מ- s ל- t ב- G_f בצע:

(א) בנה את רשת השכבות L_f מתוך N_f

(ב) מצא זרימה חוסמת g ב- L_f

(ג) עדכן $f(u,v)\leftarrow f(u,v)+g(u,v)$ לכל $u,v\in V$

(ד) בנה את הרשת השיורית N_f

(4) החזר את f

בפעם שעברה הכרנו את האלגוריתם של דיניץ וכבר הבנו 2 דברים לגבי האלגוריתם:

1. בכל שלב f היא זרימה חוקית
2. אם האלגוריתם עוצר אז הוא מחזיר זרימת מקסימום

היום אנחנו צריכים לברר תוך כמה זמן בדיוק האלגוריתם עוצר, כלומר אנחנו צריכים לחסום את מספר האיטרציות שיבוצעו במהלך הריצה. בדוגמת הריצה שעשינו בסוף השיעור שעבר, ראינו ש- t "התרחק" מ- s בכל איטרציה. אנחנו נראה שזה תמיד המצב, ולכן ייתכנו לכל היותר $|V|$ איטרציות.

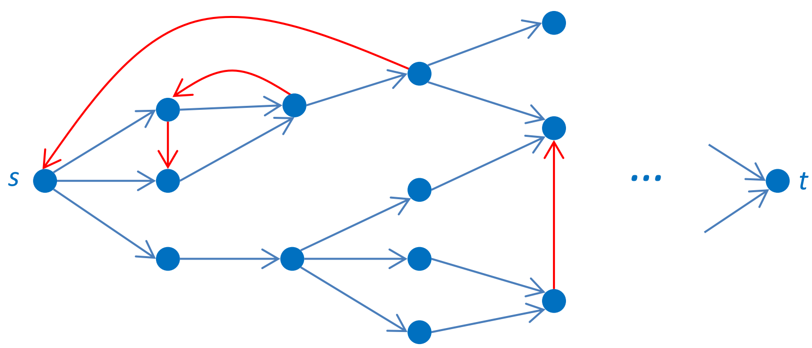
<u>משפט:</u> משלב לשלב באלגוריתם, המרחק של t מ- s ברשת השיורית גדל <u>ממש</u>. לכן יתכנו לכל היותר $|V|$ איטרציות.

<u>אינטואיציה להוכחה:</u>

נניח באיטרציה ה- k מתקיים שהמרחק של t מ- s ברשת השיורית הוא ℓ, ונניח שהרשת השיורית N_f ורשת השכבות L_f הם:

בכחול: רשת השכבות L_f

באדום: שאר הקשתות ב- N_f

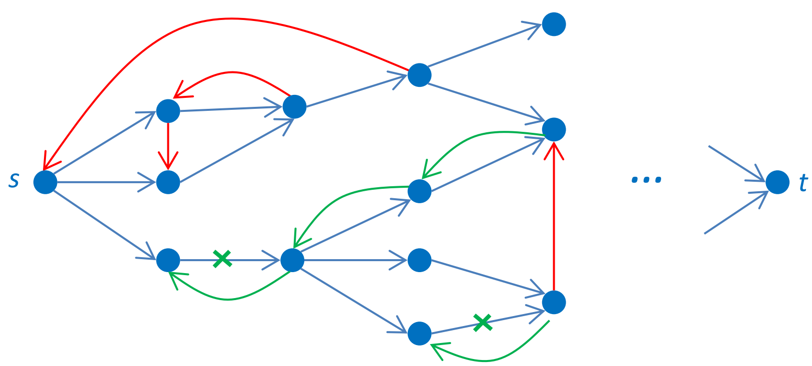


מה עשוי להשתנות בין הרשת הזיורית הזאת לזאת שאחריה (נסמנה $\widetilde{N_f}$) ?

אנחנו מוסיפים זרימה חיובית רק לאורך מסלולים <u>ברשת השכבות</u>

לכן עשויות להתווסף רק קשתות אחורה (בירוק בציור הבא) ועשויות להימחק קשתות קדימה:

הרשת השיורית הבאה $\widetilde{N_f}$



מה קורה למרחק של t מ- s ?

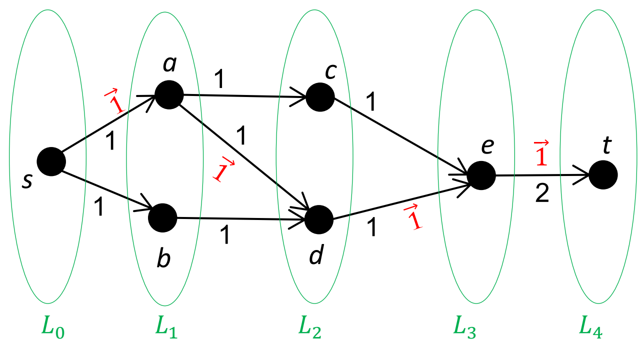
הוספת קשתות אחורה לא יכולה להקטין מרחקים. מחיקת קשתות בוודאי שלא יכולה להקטין מרחקים. לכן, ברשת השיורית הבאה $\widetilde{N_f}$ אין מסלולים קצרים מ- ℓ מ- s ל- t.

בנוסף, אין ב- $\widetilde{N_f}$ מסלולים מ- s ל- t מסלולים באורך ℓ. למה? אם יש כזה מסלול אז הוא נמצא גם ברשת השכבות הקודמת L_f ולכן הוונו לפחות צע אחת במסלול ולכן המסלול <u>לא</u> יודיע ב- $\widetilde{N_f}$. סתירה.

<u>השלמת הפרטים החסרים</u>

<u>סימון:</u> עבור $u \in V$ נסמן ב- $L_f(u)$ את השכבה של u ב- L_f.

דוגמה: ברשת השכבות הבאה מתקיים $L_f(s) = 0, \quad L_f(a) = 1, \quad L_f(e) = 3$



<u>אבחנה 1:</u>
אם $(u,v) \in E_f$, כלומר קשת שנמצאת ברשת השיורית, אזי

$$(1) \quad L_f(v) \le L_f(u) + 1$$
$$(2) \quad L_f(v) = L_f(u) + 1 \quad \Leftrightarrow \quad (u,v) \in E_L$$
$$(3) \quad L_f(v) < L_f(u) + 1 \quad \Leftrightarrow \quad (u,v) \notin E_L$$

<u>סימון:</u> נסמן ב- ℓ_k את השכבה של t ברשת השכבות בשלב ה- k של האלגוריתם.

<u>משפט:</u> $\ell_k < \ell_{k+1}$

<u>הוכחה:</u>
בהוכחה נשתמש בסימונים הבאים:

$= f$	זרימה בתחילת השלב ה- k (ממנה נבנה רשת שיורית N_f ו- ℓ_k זה המרחק של t מ- s ב- N_f)
$= f'$	זרימה בתחילת השלב ה- $k+1$ (בסימונים של האלגוריתם, $f' = f + g$)
$= P$	מסלול קצר ביותר מ- s ל- t ב- $N_{f'}$ אורך המסלול הוא ℓ_{k+1}

נחלק ל- 2 מקרים:

מקרה א: המסלול P נמצא גם ב- N_f

המסלול P נמצא ב- $N_{f'}$. לכן, כל הקשתות במסלול <u>לא</u> מורוות ע"י f', כלומר <u>לכל</u> קשת $(u,v) \in P$ מתקיים:

$$c(u,v) > f'(u,v)$$

אם המסלול היה ב- L_f אז היינו מרווים לפחות צלע אחת במסלול (כי מצאנו זרימה חוסמת). פורמלית,

- היינו מוצאים תוספת זרימה g כך שעבור לפחות צלע אחת (u,v) במסלול מתקיים:

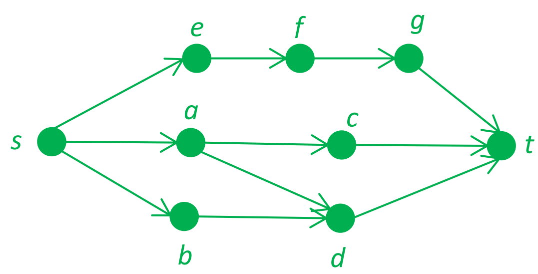
* ואז היינו מעדכנים:

$$g(u,v) = c_f(u,v) = c(u,v) - f(u,v)$$

$$f'(u,v) = f(u,v) + g(u,v) = c(u,v)$$

מסקנה: המסלול P <u>איננו</u> ב- L_f. כלומר הוא <u>לא</u> מסלול קצר ביותר ב- N_f מ- s ל- t.
לכן אורך המסלול הוא לפחות $\ell_k + 1$

דוגמה למקרה א:
אם N_f הוא הגרף משמאל, אזי המסלול
(s,e,f,g,t) <u>לא</u> היה ב- L_f



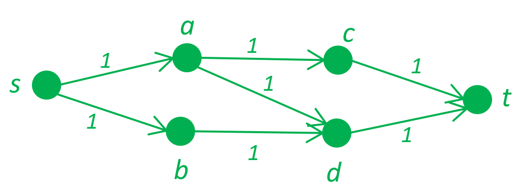
מקרה ב: המסלול P לא נמצא ב- N_f

כלומר, קיימת ב- P לפחות צלע אחת (u,v) <u>שלא</u> מופיעה ב- N_f.
איך זה קרה? ברשת N_f הופיעה הקשת (v,u) והזרמנו עליה זרימה חיובית. פורמלית,

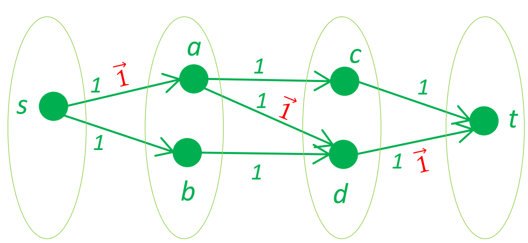
- (u,v) מופיעה ב- $N_{f'}$ ולכן $f'(u,v) < c(u,v)$
- (u,v) לא מופיעה ב- N_f ולכן $f(u,v) = c(u,v)$

אבל $f' = f + g$ ולכן $g(u,v) < 0$, כלומר $g(v,u) > 0$.
זאת אומרת שהקשת (v,u) הופיעה ב- L_f והזרמנו עליה זרימה חיובית.

דוגמה: נתחיל מ- N_f:

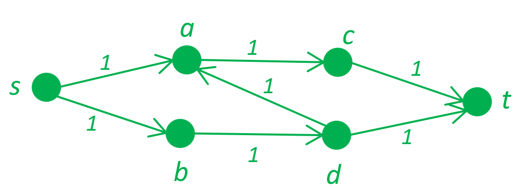


רשת השכבות L_f וזרימה חוסמת



בשלב הבא $N_{f'}$ נראית כך:

נשים לב שהקשת (d,a) שמופיעה כאן
לא הופיעה ברשת השיורית הקודמת
(זאת קשת שהתהפכה...)



נסמן את המסלול שלנו $P=\left(s=v_0,v_1,v_2,\dots,v_{\ell_{k+1}}=t\right)$ (זהו מסלול קצר ביותר מ- s ל- t ב- $N_{f'}$)
ותהי $\left(v_j,v_{j+1}\right)$ קשת במסלול P.

- אם $\left(v_j,v_{j+1}\right)$ מופיעה ב- N_f אזי לפי אבחנה 1 מתקיים $\boldsymbol{L_f\left(v_{j+1}\right)\leq L_f\left(v_j\right)+1}$.
- אם $\left(v_j,v_{j+1}\right)$ <u>לא</u> מופיעה ב- N_f אזי $\left(v_{j+1},v_j\right)$ הופיעה ב- N_f, והזרמנו בה זרימה חיובית, ולכן היא גם מופיעה ב- L_f. לכן, לפי אבחנה 1 מתקיים $L_f\left(v_j\right)=L_f\left(v_{j+1}\right)+1$, כלומר $\boldsymbol{L_f\left(v_{j+1}\right)=L_f\left(v_j\right)-1}$.

כלומר, תמיד מתקיים $L_f\left(v_{j+1}\right)\leq L_f\left(v_j\right)+1$ וקיים אינדקס j עבורו $L_f\left(v_{j+1}\right)=L_f\left(v_j\right)-1$.

נסתכל על הסדרה הבאה:

$$\underbrace{L_f(v_0)}_{\substack{=L_f(s)\\=0}}\ ,\ L_f(v_1)\ ,\ L_f(v_2)\ ,\qquad \dots\ ,\ \underbrace{L_f\left(v_{\ell_{k+1}}\right)}_{\substack{=L_f(t)\\=\ell_k}}$$

בסדרה הזאת בין כל 2 איברים עוקבים גדלים בלכל היותר 1, וקיים מקום בו קטנים ב- 1.

דוגמה לסדרה כזאת: *0,1,2,3,2,3,4*

מספר האיברים בסדרה הוא לפחות ℓ_k+3

לכן, אורכו של P (שהוא מסלול קצר ביותר ב- $N_{f'}$ מ- s ל- t) הוא לפחות ℓ_k+2.

בפרט, $\ell_{k+1}>\ell_k$.

<u>**מסקנה:**</u> ישנם כל היותר $|V|$ שלבים באלגוריתם.

<u>**סיבוכיות האלגוריתם של דיניץ:**</u>
הלולאה מתבצעת $O(|V|)$ פעמים. בכל איטרציה של הלולאה:

(א) *BFS* בזמן $O(|V|+|E|)$
איך עם *BFS*? אלג' *BFS* מחזיר לנו לכל קודקוד v את המרחק שלו מ- s, שנסמנו ע"י $dist(v)$.
לכן, נעתיק קשת (u,v) לרשת L_f אם ורק אם $dist(v)=dist(u)+1$.

(ב) איך מוצאים זרימה חוסמת ??

(ג) $O(|E|)$

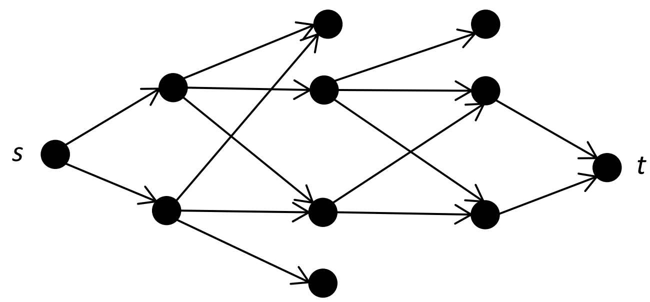
(ד) $O(|E|)$

<u>**נסיון ראשון למציאת זרימה חוסמת ע"י אלגוריתם חמדן:**</u>
נוכל למצוא זרימה חוסמת על ידי אלגוריתם חמדן שבכל פעם מוצא מסלול לא רווי בגרף ומזרים בו כמה שאפשר. מציאת מסלול מתבצעת ע"י *BFS* בזמן $O(|V|+|E|)$ ויתכנו לכל היותר $O(|E|)$ מסלולים לפני שנמצא זרימה חוסמת. לכן נקבל אלגוריתם בסיבוכיות $O\left(|E|^2+|V|\cdot|E|\right)$ למציאת זרימה חוסמת.

מציאת זרימה חוסמת ברשת השכבות בצורה יעילה

באלגוריתם החמדן הנ"ל היו לנו $O(|E|)$ איטרציות בהן חיפשנו מסלול, וכל חיפוש מסלול התבצע בזמן $O(|E|+|V|)$. מה שנשפר עכשיו זה את הזמן שלוקח למצוא מסלול ברשת שכבות. במקום $O(|E|+|V|)$ זה יהיה בזמן $O(|V|)$ וסה"כ נקבל אלג' בסיבוכיות $O(|E|\cdot|V|)$ למציאת זרימה חוסמת.

<u>הרעיון:</u> נתחיל מ- t "ונלך" אחורה.



<u>האלגוריתם:</u>

<u>קלט:</u> רשת שכבות $L_f=\left(G_L=(V,E_L),c_f,s,t\right)$

<u>אתחול:</u> $g(u,v)=0$ לכל $u,v\in V$
$F\leftarrow E_L$

<u>צעד:</u> כל עוד $0<in\text{-}degree(t)$ בצע:

(א) מתחילים מ- t והולכים אחורה בגרף (V,F) עד שנתקע ב- s. מצאנו מסלול P.

(ב) חשב: $c_g(P)=\min\limits_{(u,v)\in P}\left\{c_f(u,v)-g(u,v)\right\}$

(ג) עדכן זרימה: לכל (u,v) במסלול P בצע:
$g(u,v)\leftarrow g(u,v)+c_g(P)$
$g(v,u)\leftarrow g(v,u)-c_g(P)$

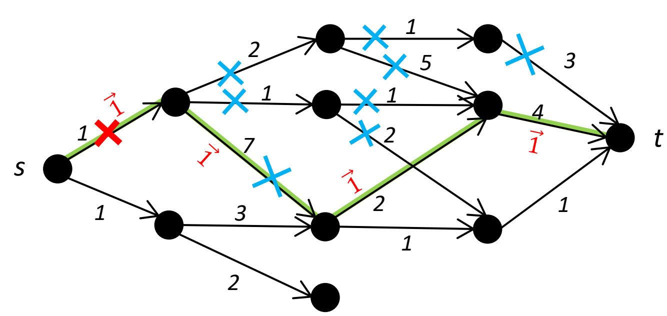
(ד) נקה את הגרף: לכל (u,v) במסלול P, אם $c_f(u,v)=g(u,v)$ אזי בצע:
- $F\leftarrow F\setminus\{(u,v)\}$
- אם $in\text{-}degree(v)=0$ אזי בצע $\text{CleanForward}(v)$

<u>הפרוצדורה $\mathbf{CleanForward}(v)$</u>

לכל $(v,w)\in F$ בצע:
- $F\leftarrow F\setminus\{(v,w)\}$
- אם $in\text{-}degree(w)=0$ אזי בצע $\text{CleanForward}(w)$

המטרה של הפרוצדורה הזאת היא לשמור על הנכונות של הטענה הנשמרת הבאה:

<u>טענה נשמרת:</u> לכל צומת $v\neq s$ בגרף (V,F) מתקיים: אם דרגם היציאה שלו גדולה ממש מאפס אז גם דרגת הכניסה שלו גדולה ממש מאפס.



דוגמה: נתבונן ברשת השכבות הבא.
נניח מצאנו את המסלול הירוק והזרמנו בו 1.
קשתות רוויות אנחנו מוחקים
קודקודים עם דרגת כניסה 0 אנחנו "מחסלים".

<u>טענה:</u> האלגוריתם מוצא זרימה חוסמת

<u>רעיון ההוכחה:</u>
נניח בשלילה שבסוף הריצה יש מסלול <u>לא רווי</u> מ- s ל- t:
(s, a, b, c, d, t)
אז הקשת הראשונה (s, a) לא נמחקה בכל הריצה
(כי אנחנו מוחקים רק קשתות רוויות או כאלה שלא נגישות מ- s)
לכן גם הקשת השניה (a, b) לא נמחקה
וכן הלאה, עד שנגיע למסקנה שגם הקשת (d, t) לא נמחקה, כלומר דרגת הכניסה של t גדולה מאפס, בסתירה לכך שהאלגוריתם עצר.

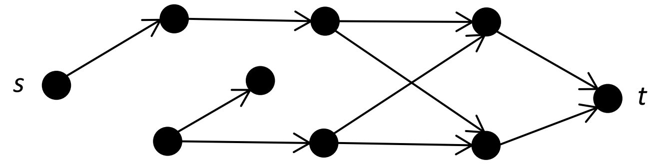
<u>ניתוח זמן ריצה:</u>
- מציאת מסלול לוקחת זמן כאורך המסלו. המסלול פשוט ולכן מציאת מסלול מתבצעת בזמן $O(|V|)$.
  יהיו $O(|E|)$ מסלולים לכל היותר, כי כל פעם שנמצא מסלול נתלוש לפחות קשת אחת מ- F.
  סה"כ מציאת מסלולים $O(|E| \cdot |V|)$

- נבצע ניתוח גלובלי של מחיר כל פעולות ה *CleanForward* ביחד:
  מחיר פעולת $\text{CleanForward}(v)$ ללא הקריאות הרקורסיביות הוא $O\big(out\text{-}degree(v)\big)$.
  כל קשת נוריד לכל היותר פעם אחת.
  סה"כ מחיר כל פעולות ה *CleanForward* הוא $O(|E|)$

- נעשה את אותו חשבון בצורה אחרת: אמרנו שמחיר פעולת $\text{CleanForward}(v)$ הוא $O\big(out\text{-}degree(v)\big)$. לכן המחיר הכולל הוא
$$\sum_{v \in V} O\big(out\text{-}degree(v)\big) = O(|E_L|) = O(|E|)$$

לכן, סה"כ סיבוכיות מציאת זרימה חוסמת היא
$$O\Bigg(\underbrace{|E| \cdot |V|}_{\text{מציאת מסלולים}} + \underbrace{|E|}_{\text{ניקויי}}\Bigg) = O(|E| \cdot |V|)$$

<u>הערה:</u> האלג' למציאת זרימה חוסמת עובד לכל רשת חסרת מעגלים, אבל ייתכן שהטענה הנשמרת לא מתקיימת בתחילת הריצה ולכן צריך להתחיל את הריצה מ- *CleanForward*.



<u>הוכחנו:</u> האלגוריתם של דיניץ מוצא זרימת מקסימום בזמן $O(|E| \cdot |V|^2)$:
$O(|V|)$ שלבים
$O(|E| \cdot |V|)$ זמן בכל שלב

<u>הערה:</u> האלגוריתם הטוב ביותר הידוע כיום רץ בזמן $O(|E| \cdot |V|)$