תכנון אלגוריתמים 6.12.2020

הרצאה 15: זרימה

מרצה: אורי שטמר

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

תזכורות

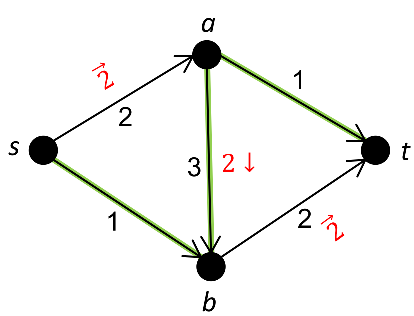
__רשת זרימה:__ $N = \left(\underbrace{G = (V,E)}_{\text{גרף מכוון}}, \underbrace{c}_{\text{קיבולים}}, \underbrace{s}_{\text{מקור}}, \underbrace{t}_{\text{יעד}} \right)$

__זרימה:__ פונקציה $f: V \times V \to \mathbb{R}$

- אילוצי קיבול: $f(u,v) \leq c(u,v)$
- אנטי-סימטריות: $f(u,v) = -f(v,u)$
- שימור זרימה לכל $u \neq s,t$: $\sum_{v \in V} f(u,v) = 0$
- גודל הזרימה: $|f| = \sum_{v \in V} f(s,v)$

__מסלול שיפור:__ בהינתן רשת זרימה N ופונקצית זרימה f, __מסלול שיפור__ הוא __סדרה__ של קודקודים $s = v_0, v_1, \dots, v_\ell = t$ כך שלכל i מתקיים $0 < \Delta_i = c(v_i, v_{i+1}) - f(v_i, v_{i+1})$

__דוגמה:__
בשחור – רשת זרימה
באדום – זרימה נוכחית
בירוק – מסלול שיפור



__רשת זרימה שיורית:__ בהינתן רשת זרימה $N = (G = (V,E), c, s, t)$ וזרימה f, הגדרנו __רשת הזרימה השיורית__ N_f באופן הבא:

$$N_f = \left(G_f = \left(V, E_f\right), c_f, s, t\right)$$

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

$$E_f = \left\{ (u,v) \in V \times V \ : \ c_f(u,v) > 0 \right\}$$

__ראינו:__ מסלול שיפור ברשת N $\iff$ מסלול מ s ל t בגרף G_f

אלגוריתם *F-F*:

(1) התחל מזרימה $f(u,v)=0$ לכל u,v

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

(3) אם אין מסלול מ- s ל- t ב- G_f עצור והחזר את f. אחרת מצא מסלול פשוט P מ- s ל- t ב- G_f.

(4) הגדר

$$c_f(P)=\min_{(u,v)\in P}\{\,c_f(u,v)\,\}\quad=\min_{(u,v)\in P}\{\,c(u,v)-f(u,v)\,\}$$

(5) הגדר

$$f(u,v)=\begin{cases}f(u,v)+c_f(P) & , \quad (u,v)\in P\\ f(u,v)-c_f(P) & , \quad (v,u)\in P\\ f(u,v) & , \quad \text{else}\end{cases}$$

(6) לך ל- (2)

בפעם שעברה התחלנו לנתח את האלגוריתם של *F-F*. קודם כל רצינו להראות שהזרימה שהוא מחזיר היא בכלל <u>חוקית</u> ובשביל זה הראנו את הטענה הבאה:

<u>**טענה 1:**</u> תהי N רשת זרימה, תהי f זרימה חוקית ב- N, ותהי f' זרימה חוקית ב- N_f. נגדיר $g=f+f'$, כלומר

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

אזי g היא זרימה <u>חוקית</u> ב- N, שגודלה

$$|g|=|f|+|f'|$$

<u>מסקנה מטענה 1:</u>

- בכל שלב באלכוריתם של *F-F* מתקיים ש- f היא זרימה <u>חוקית</u>
- בכל שלב, הזרימה גדלה ב- $0<C_f(P)$

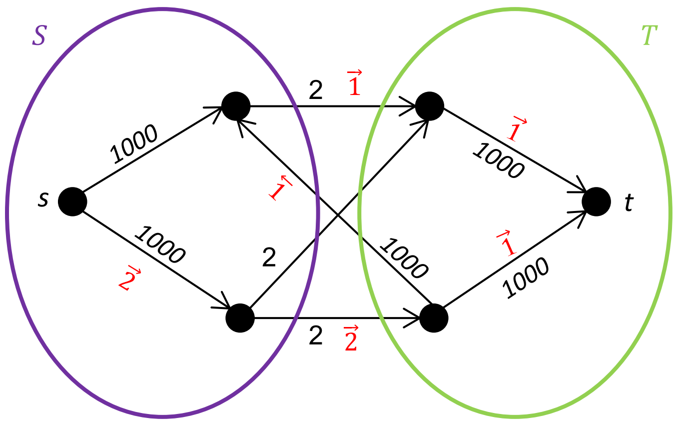
אז חרי שהוכחנו את זה כבר גלינו שלאורך כל הריצה האלג' של *F-F* מתחזק זרימה <u>חוקית</u>. בפרט, אם הוא עוצר אז הוא מחזיר זרימה חוקית. בנוסף, אנו רואים כאן שמשלב לשלב באלגוריתם הזרימה שלנו <u>גדלה</u>.

אח"כ רצינו להוכיח שאם האלגוריתם עוצר אז הזרימה שהוא מחזיר היא זרימת <u>מקסימום</u>. בשביל זה התחלנו לדבר על חתכים:

<u>חתכים:</u>

- חתך הוא זוג קבוצות S,T כך שמתקיים $S\cap T=\emptyset$,$S\cup T=V$,$s\in S$ ו- $t\in T$
- קיבול של חתך $c(S,T)=\sum_{\substack{u\in S\\ v\in T}}c(u,v)$
- זרימה בחתך $f(S,T)=\sum_{\substack{u\in S\\ v\in T}}f(u,v)$

<u>דוגמה:</u>



קיבול החתך הוא 6
הזרימה בחתך היא 2

נסתכל שוב על הדוגמה האחרונה.
בחתך שראינו הזרימה היא 2.
מה גודל הזרימה? גם 2.
בעצם, כל חתך שננסה בדוגמה הזאת נראה שגודל הזרימה בו היא 2.

בפעם שעברה הראינו שזה תמיד המצב: בכל חתך ברשת, גודל הזרימה בו שווה לגודל הזרימה במערכת:

טענה 3: לכל חתך (S,T) ולכל זרימה חוקית f מתקיים: $f(S,T)=|f|=\sum_{v\in V} f(s,v)$

ראינו גם את הטענה הבאה שאמרה שלכל חתך, הזרימה בחתך תמיד קטנה שווה מקיבול החתך:

טענה 4: לכל חתך (S,T) ולכל זרימה חוקית f מתקיים: $f(S,T)\le c(S,T)$

סוף התזכורת

עכשיו יש לנו את כל מה שאנחנו צריכים כדי להוכיח שאם האלגוריתם של *F-F* עוצר אז הוא מחזיר זרימת מקסימום.

משפט – זרימת מקסימום וקיבול חתך מינימום *(Max Flow Min Cut)*
תהי $N=(G,s,c,t)$ רשת זרימה ותהי f זרימה ברשת. התנאים הבאים שקולים:
(1) f היא זרימת מקסימום
(2) אין מסלול מ- s ל- t בגרף של הרשת השיורית N_f
(3) קיים חתך (S,T) עבורו $|f|=c(S,T)$

נזכור שתנאי העצירה של *F-F* זהה לתנאי (2) הנ"ל. לכן, אם נוכיח את המשפט הזה אז נסיים להראות שאם האלגוריתם של *F-F* עוצר אזי הוא מוצא זרימת מקסימום.

הוכחת המשפט:
כדי להוכיח את המשפט נראה $(1)\Rightarrow(2)\Rightarrow(3)\Rightarrow(1)$

$(1)\Rightarrow(2)$
נניח בשלילה ש- (1) מתקיים ו- (2) לא מתקיים.
כלומר נניח בשלילה ש- f זרימת מקסימום, אבל כן יש מסלול מ- s ל- t ברשת השיורית N_f. איך נקבל סתירה?
נבנה את N_f ונמצא בה מסלול. על פי טענה 1, נוכל להגדיל את f בסתירה לכך ש- f זרימת מקסימום.

$(3)\Rightarrow(1)$
על פי טענות 3+4: לכל זרימה f ולכל חתך (S,T) מתקיים
$|f|=f(S,T)\le c(S,T)$
לכן, אם $|f|=c(S,T)$ עבור חתך (S,T) כלשהו, אזי f זרימת מקסימום.

$(2)\Rightarrow(3)$
תהי f זרימה כך שאין מסלול מ- s ל- t ב- N_f. נסמן ב- S את קבוצת כל הצמתים שיש אליהם מסלול מ- s ב- N_f.
מתקיים: $s\in S$ ו- $t\notin S$.

נגדיר $T = S \setminus S$.

לכל $u \in S$ ו- $v \in T$ מתקיים $(u,v) \notin E_f$ (אחרת היה מסלול ל- v ואז v היה ב- S...)

לכן, לפי ההגדרה של E_f מתקיים $c(u,v) = f(u,v)$ ⍟

למה זה? כשבנינו את הרשת השיורית, איזה צלעות הכנסנו ל- E_f?

לכל $u,v \in V$ הסתכלנו על $c_f(u,v) = c(u,v) - f(u,v)$.

אם ההפרש הזה היה אפס אז אין צלע ואם ההפרש הזה היה גדול מאפס אז יש צלע.

מסקנה:

$$|f| \underbrace{=}_{\text{לפי טענה 3}} f(S,T) = \sum_{\substack{u\in S\\ v\in T}} f(u,v) \underbrace{=}_{⍟} \sum_{\substack{u\in S\\ v\in T}} c(u,v) = c(S,T)$$

כלומר, $|f| = c(S,T)$.

מ.ש.ל. (משפט זרימת מקסימום וקיבול חתך מינימום)

<u>מסקנה ממשפט *mfmc*:</u>

אם *F-F* עוצר, אזי מוצא זרימת מקסימום!

סיכום ביניים: עד עכשיו הוכחנו שבכל שלב הזרימה f היא חוקית והוכחנו שאם האלגוריתם עוצר אזי הוא מחזיר זרימת מקסימום.

<u>עובדה:</u> אם הקיבולים אינם שלמים אזי האלגוריתם לא בהכרח עוצר.

<u>משפט:</u> אם כל הקיבולים שלמים, אזי האלגוריתם עוצר אחרי לכל היותר $|f^*|$ שלבים, כאשר $|f^*|$ היא גודל זרימת מקסימום ברשת.

<u>סקיצת הוכחת המשפט:</u>

- באינדוקציה – בכל שלב הזרימה ב- N והקיבולים ברשת השיורית N_f הם שלמים
- בכל שלב הזרימה כדלה בלפחות יחידה אחת, ולכן נקבל אחרי לכל היותר $|f^*|$ שלבים זרימת מקסימום

<u>סיבוכיות:</u>

בניית הרשת השיורית N_f מתבצעת בזמן $O(|E| + |V|)$

מציאת מסלול ב- N_f על ידי *BFS* מתבצעת בזמן $O(|E| + |V|)$

סה"כ זמן ריצה $O\big(|f^*| \cdot (|E| + |V|)\big)$

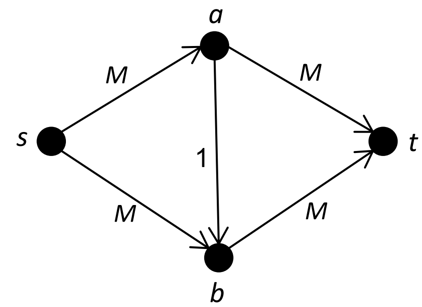
זה טוב?

האלרוגיתם הזה הוא <u>לא</u> פולינומי. הוא מבצע $|f^*|$ שלבים.

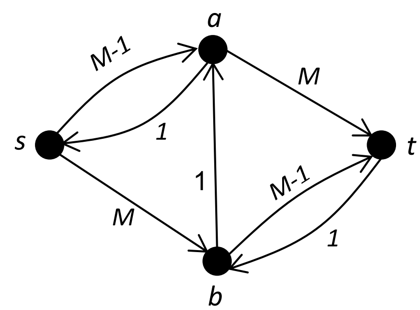
כדי שזה יהיה אלגוריתם יעיל הוא צריך להיות פולינומי בגודל <u>הייצוג</u> של $|f^*|$, כלומר תלוי ב- $\log|f^*|$.

(הערה: $|f^*|$ הוא בכלל חלק <u>מהפלט</u> ולא <u>מהקלט</u>, אבל אנחנו יכולים לחסום אותו על ידי סכום הקיבולים...)

<u>דוגמה לבעיתיות:</u>
נתבונן ברשת הבאה, כאשר $M \gg 1$ הוא מספר גדול מאוד, למשל $M = 2^{64}$.



נריץ את *F-F* והוא צריך למצוא מסלול שיפור. נניח הוא מוצא את *s,a,b,t* ומזרים בו 1.
נבנה את הרשת השיורית:



ונניח שעכשיו האלגוריתם בוחר את המסלול *s,b,a,t* ומזרים בו 1.
כך יתכן שנמשיך בדרך הזו ומספר האיטרציות שידרשו הוא *2M* כאשר *M* מספר גדול ולכן לא ניתן לומר שהאלגוריתם רץ בזמן פולינומי.

הערה: נשים לב שהבעייה מהדוגמה האחרונה נפתרת אם היינו מחפשים בכל פעם מסלול שיפור <u>קצר ביותר</u>.

<u>משפט *(Edmonds&Karp 1972)*:</u> אם בכל שלב נמצא מסלול שיפור קצר ביותר (עם מספר קטן ביותר של צלעות) אז האלגוריתם עוצר אחרי לכל היותר $O(|V| \cdot |E|)$ איטרציות ומחזיר זרימת מקסימום.

<u>סיבוכיות:</u> מציאת מסלול שיפור קצר ביותר מתבצע בזמן $O(|V| + |E|)$ על ידי *BFS* ולכל סה"כ סיבוכיות $O(|E|^2 \cdot |V|)$.

(הערה: באלגוריתם של *EK* אין את הדרישה שהקיבולים שלמים...).

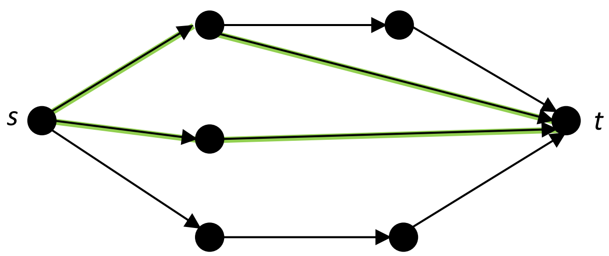
לא נפרט יותר על האלגוריתם והניתוח של *Edmonds&Karp*. אנחנו נראה אלגוריתם יעיל יותר – האלג' של דיניץ.

אלגוריתם דיניץ

אמרנו שבאלג' של *E&K* הרעיון הוא למצוא בכל שלב מסלול שיפור קצר ביותר.

<u>הרעיון באלגוריתם של דיניץ:</u> בכל שלב נמצא <u>כמה שיותר</u> מסלולי שיפור קצרים ביותר <u>בבת אחת</u>.

אינטואיציה: נניח שהרשת השיורית בשלב כלשהו נראית כך:



נרצה למצוא את שני המסלולים הירוקים

תאור לא פורמלי של האלגוריתם של דיניץ:

(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) מצא "כמה שיותר" מסלולים קצרים ביותר מ- s ל- t ב- G_f

(ב) הזרם "כמה שאפשר" במסלולים שמצאת. סמן זאת כ- g

(ג) עדכן $f\leftarrow f+g$

(ד) בנה את הרשת השיורית N_f

(4) החזר את f

עכשיו אנחנו צריכים להבין מה שלב (3) אומר בדיוק.

הגדרה (רשת שכבות):
בהינתן רשת שיורית $N_f=\left(G_f=\left(V,E_f\right),c_f,s,t\right)$ נבנה <u>רשת שכבות</u> L_f באופן הבא:

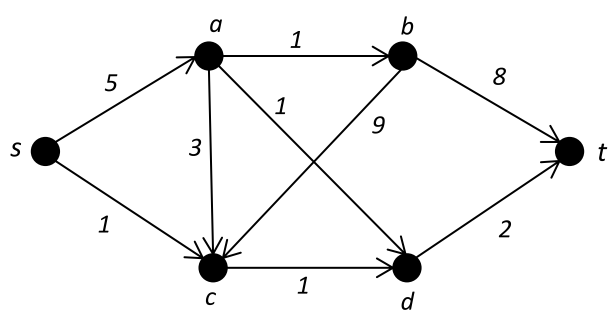
$$L_f=\left(G_L=(V,E_L),c_f,s,t\right)$$

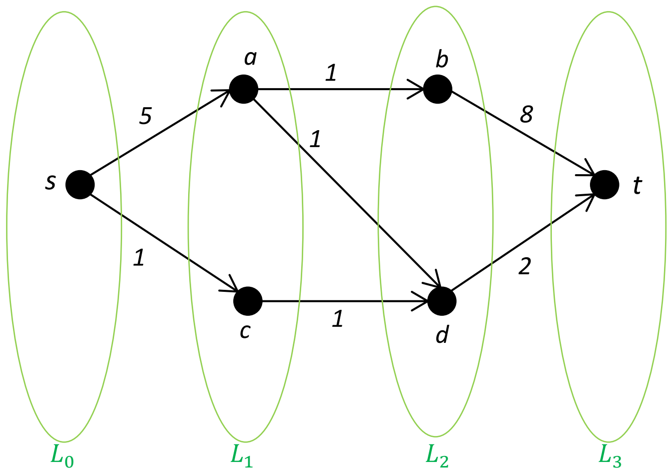
$$L_i=\left\{v\ :\ \begin{matrix}\text{מסלול קצר ביותר מ } s \text{ ל } v \text{ ב } G_f\\ \text{אורכו (במספר צלעות) בדיוק } i\end{matrix}\right\}$$

$$E_L=\left\{(u,v):\ \begin{matrix}(u,v)\in E_L \text{ וגם קיים } i\\ \text{כך ש } u\in L_i \text{ וגם } v\in L_{i+1}\end{matrix}\right\}$$

אינטואיטיבית: רשת השכבות L_f מכילה את כל המסלולים הקצרים ביותר מ- s ל- t ב- G_f.

דוגמה: רשת שיורית:





שימו לב:
הקשת *(b,c)* לא מופיעה כאן כי היא קשת <u>אחורה</u>
הקשת *(a,c)* לא מופיעה כי היא קשת באותה שכבה

קשתות ב- E_f שמתקדמות לשכבה הבאה יהיו ב- E_L וקשתות באותה שכבה או קשתות לשכבה קטנה יותר לא יהיו ב- E_L.

שאלה: האם יתכן שברשת השכבות מהדוגמה האחרונה תהייה קשת קדימה "שקופצת" מעל שכבה, למשל קשת מ- *a* ל- *t* ?

תשובה: לא, כי אז *t* היה צריך להיות בשכבה הקודמת...

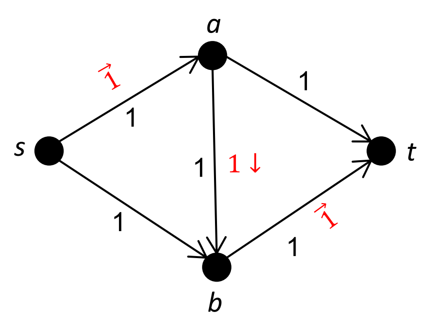
האינטואיציה: אמרנו שאנחנו רוצים למצוא "כמה שיותר" מסלולים קצרים ביותר מ- s ל- t ברשת השיורית. הנה הם. ככה אנחנו מוצאים אותם – בעזרת רשת השכבות. אז התוכנית שלנו זה מהרשת השיורית N_f לבנות את רשת השכבות L_f ועכשיו אמרנו שאנחנו רוצים להזרים כאן "כמה שאפשר". מה זה אומר בדיוק? נזכר באלגוריתם הכושל שנסינו לפני *F-F*.

<u>הגדרה:</u> f היא <u>זרימה חוסמת</u> ברשת $N=(G,c,s,t)$ אם בכל מסלול מ- s ל- t יש קשת רווייה, כלומר קשת (u,v) שעבורה $c(u,v)=f(u,v)$.

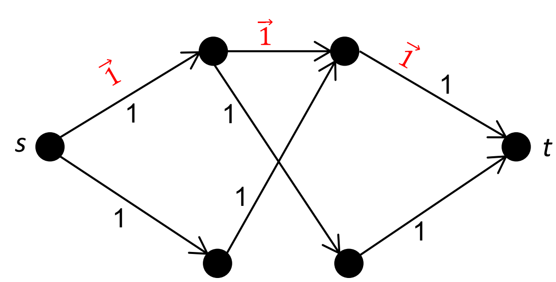
כלומר בהגדרת זרימה חוסמת מסתכלים רק על מסלולים ב- G ולא על מסלולי שיפור או על מסלולים ב- G_f!

<u>דוגמה:</u>

זרימה חוסמת <u>שאינה</u> זרימת מקסימום.
בכל מסלול מ- *s* ל- *t* יש קשת רווייה.



דוגמה נוספת לזרימה חוסמת שאינה זרימת מקסימום:



עכשיו נוכל להשלים את הפרטים החסרים מהתיאור של האלגוריתם של דיניץ:

<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

<u>עובדה:</u> g זרימה <u>חוקית</u> ב- N_f

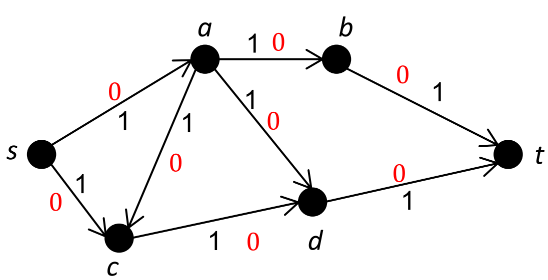
<u>מסקנה:</u> בכל שלב f זרימה חוקית (סכום של זרימה חוקית ב- N וזרימה חוקית ב- N_f היא זרימה חוקית ב- N)

<u>מסקנה:</u> קיבלנו בחינם שאם האלגוריתם של דיניץ עוצר אזי f זרימת מקסימום!

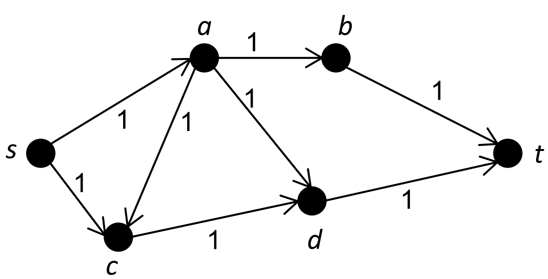
דוגמת ריצה

הַרשת N:

מתחילים עם זרימה זהותית אפס



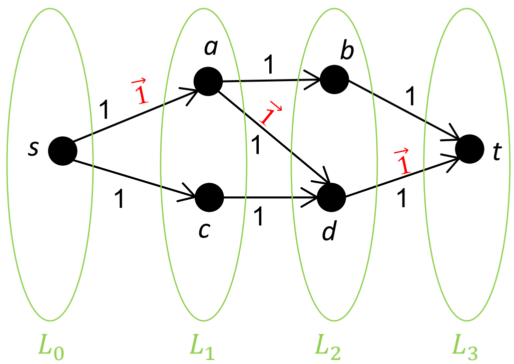
רשת שיורית N_f באיטרציה 1:



רשת שכבות L_f באיטרציה 1:

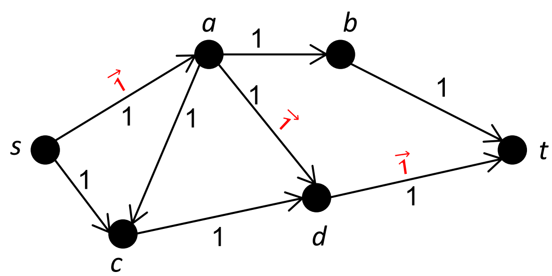
נשים שכאן $t \in L_3$

נמצא זרימה חוסמת ברשת השכבות,
למשל כמו שמסומן כאן באדום

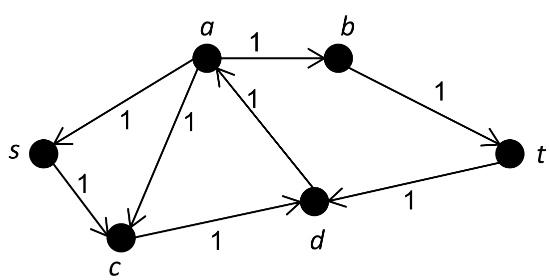


עכשיו מעדכנים את הזרימה ברשת המקורית.

הַרשת N:



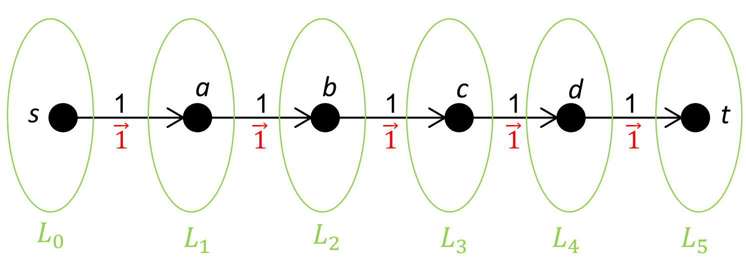
רשת שיורית N_f באיטרציה 2:



רשת שכבות L_f באיטרציה 2:

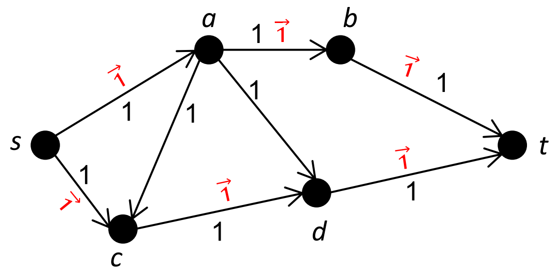
נשים שכאן $t \in L_5$

נמצא זרימה חוסמת



נחזור לרשת המקורית ונעדכן שם את הזרימה.

הַרשת N:



ברשת השיורית הבאה שנבנה לא יהיו קשתות שיוצאות מ- s (כי כל הקשתות שיוצאות מ- s בגרף הנ"ל רוויות) ולכן לא יהיה מסלול מ- s ל- t ולכן האלגוריתם יעצור.