הרצאה 10

מסלולים קצרים ביותר בגרף ממושקל, בהתחלה ממקור בודד דייקסטרה ובלמן פורד, ואח"כ מכל מקור לכל מקור.

ראינו בינתיים דרך אחת - שהיתה למצוא מסלולים באורך 1 ואז לנסות להגדיל ל2 אם זה עוזר וכו וכו וכו עד שמגיעים למספר הצמתים פחות 1. הסיבוכיות היתה $O(|V|^4)$

והצלחנו לשפר ע"י שימוש בדרך שדומה לכפל מטריצות ל $O(|V|^3 \log|V|)$

ננסה לשפר עוד

Floyd Warshall

גם הפעם יש מטריצות שמחושבות לפי הסדר על פי הכלל הבא. האינדקס העליון הוא מספר המטריצה, והרעיון הוא שבכל פעם ננסה למצוא מסלולים באופן הבא - במטריצה 0 - מסלולים ישירים.

במטריצה 1 - מסלולים שעוברים אך ורק דרך הקודקוד ש*מספרו* 1.

במטריצה 2 - מסלולים שיכולים לעבור דרך הקודקוד שמספרו 1 ודרך הקודקוד שמספרו 2

במטריצה k - יכולים לעבור רק דרך הקודקודים שמספרים k,....,1

נשקל

$$d_{ij}^{(k)} = \begin{cases} w_{ij} & k=0 \\ \min\left(d_{ij}^{(k-1)},\ d_{ik}^{(k-1)} + d_{kj}^{(k-1)}\right) & 1,2,\ldots \end{cases}$$

$k-1$: $i \xrightarrow{1,2,\ldots,k-1} j$

k: $i \xrightarrow{1\ldots k-1} k \xrightarrow{1\ldots k-1} j$

הפסאודו קוד

FLOYD-WARSHALL(W)

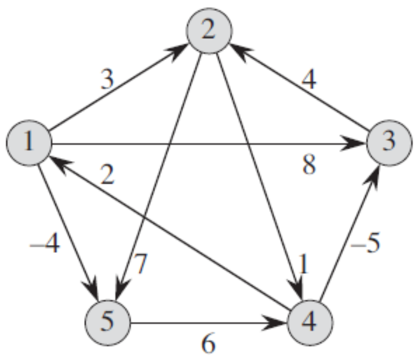
$n = W.rows$
$D^{(0)} = W$
for $k = 1$ **to** n
    let $D^{(k)} = \left(d_{ij}^{(k)}\right)$ be a new $n \times n$ matrix
    for $i = 1$ **to** n
        for $j = 1$ **to** n
            $d_{ij}^{(k)} = \min\left(d_{ij}^{(k-1)}, d_{ik}^{(k-1)} + d_{kj}^{(k-1)}\right)$
return $D^{(n)}$

באלגוריתם אפשר לבנות תוך כדי מטריצה פאי Π עם המסלולים עצמם - במטריצה בצומת π_{ij}^{k}

יהיה הצומת הקודם לj במסלול מi עם צמתים $1,2,...,k$

סיבוכיות $O(|V|^3)$

למשל



$$D^{(0)} = \begin{array}{c|c} & \begin{matrix} 1 & 2 & 3 & 4 & 5 \end{matrix} \\ \begin{matrix} 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{matrix} & \begin{pmatrix} 0 & 3 & 8 & \infty & -4 \\ \infty & 0 & \infty & 1 & 7 \\ \infty & 4 & 0 & \infty & \infty \\ 2 & \infty & -5 & 0 & \infty \\ \infty & \infty & \infty & 6 & 0 \end{pmatrix} \end{array} \qquad \Pi^{(0)} = \begin{pmatrix} \text{NIL} & 1 & 1 & \text{NIL} & 1 \\ \text{NIL} & \text{NIL} & \text{NIL} & 2 & 2 \\ \text{NIL} & 3 & \text{NIL} & \text{NIL} & \text{NIL} \\ 4 & \text{NIL} & 4 & \text{NIL} & \text{NIL} \\ \text{NIL} & \text{NIL} & \text{NIL} & 5 & \text{NIL} \end{pmatrix}$$

$\underline{4 \to 2!}$

$4 \to 2 = \infty$

$4 \to 1 + 1 \to 2 =$
$2 + 3 = 5$

$$D^{(1)} = \begin{pmatrix} 0 & 3 & 8 & \infty & -4 \\ \infty & 0 & \infty & 1 & 7 \\ \infty & 4 & 0 & \infty & \infty \\ 2 & \boxed{5} & -5 & 0 & -2 \\ \infty & \infty & \infty & 6 & 0 \end{pmatrix} \qquad \Pi^{(1)} = \begin{pmatrix} \text{NIL} & 1 & 1 & \text{NIL} & 1 \\ \text{NIL} & \text{NIL} & \text{NIL} & 2 & 2 \\ \text{NIL} & 3 & \text{NIL} & \text{NIL} & \text{NIL} \\ 4 & 1 & 4 & \text{NIL} & 1 \\ \text{NIL} & \text{NIL} & \text{NIL} & 5 & \text{NIL} \end{pmatrix}$$

$$D^{(2)} = \begin{pmatrix} 0 & 3 & 8 & 4 & -4 \\ \infty & 0 & \infty & 1 & 7 \\ \infty & 4 & 0 & 5 & 11 \\ 2 & 5 & -5 & 0 & -2 \\ \infty & \infty & \infty & 6 & 0 \end{pmatrix} \qquad \Pi^{(2)} = \begin{pmatrix} \text{NIL} & 1 & 1 & 2 & 1 \\ \text{NIL} & \text{NIL} & \text{NIL} & 2 & 2 \\ \text{NIL} & 3 & \text{NIL} & 2 & 2 \\ 4 & 1 & 4 & \text{NIL} & 1 \\ \text{NIL} & \text{NIL} & \text{NIL} & 5 & \text{NIL} \end{pmatrix}$$

$\underline{4 \to 2!}$

$4 \to 2 = 5$

$4 \to 3 + 3 \to 2 =$
$-5 + 4 = -1$

$$D^{(3)} = \begin{pmatrix} 0 & 3 & 8 & 4 & -4 \\ \infty & 0 & \infty & 1 & 7 \\ \infty & 4 & 0 & 5 & 11 \\ 2 & \boxed{-1} & -5 & 0 & -2 \\ \infty & \infty & \infty & 6 & 0 \end{pmatrix} \qquad \Pi^{(3)} = \begin{pmatrix} \text{NIL} & 1 & 1 & 2 & 1 \\ \text{NIL} & \text{NIL} & \text{NIL} & 2 & 2 \\ \text{NIL} & 3 & \text{NIL} & 2 & 2 \\ 4 & 3 & 4 & \text{NIL} & 1 \\ \text{NIL} & \text{NIL} & \text{NIL} & 5 & \text{NIL} \end{pmatrix}$$

$$D^{(4)} = \begin{pmatrix} 0 & 3 & -1 & 4 & -4 \\ 3 & 0 & -4 & 1 & -1 \\ 7 & 4 & 0 & 5 & 3 \\ 2 & -1 & -5 & 0 & -2 \\ 8 & 5 & 1 & 6 & 0 \end{pmatrix} \qquad \Pi^{(4)} = \begin{pmatrix} \text{NIL} & 1 & 4 & 2 & 1 \\ 4 & \text{NIL} & 4 & 2 & 1 \\ 4 & 3 & \text{NIL} & 2 & 1 \\ 4 & 3 & 4 & \text{NIL} & 1 \\ 4 & 3 & 4 & 5 & \text{NIL} \end{pmatrix}$$

$$D^{(5)} = \begin{pmatrix} 0 & 1 & -3 & 2 & -4 \\ 3 & 0 & -4 & 1 & -1 \\ 7 & 4 & 0 & 5 & 3 \\ 2 & -1 & -5 & 0 & -2 \\ 8 & 5 & 1 & 6 & 0 \end{pmatrix} \qquad \Pi^{(5)} = \begin{pmatrix} \text{NIL} & 3 & 4 & 5 & 1 \\ 4 & \text{NIL} & 4 & 2 & 1 \\ 4 & 3 & \text{NIL} & 2 & 1 \\ 4 & 3 & 4 & \text{NIL} & 1 \\ 4 & 3 & 4 & 5 & \text{NIL} \end{pmatrix}$$

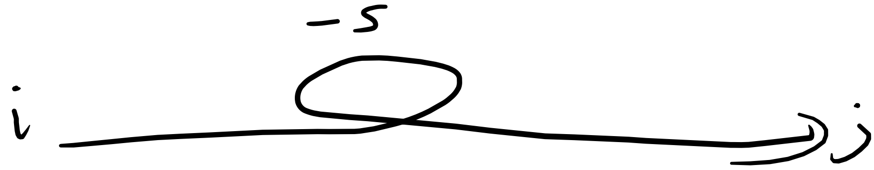
אלגוריתם נוסף

Johnson

עובד אם הגרף דליל - כלומר אין בו הרבה קשתות. האלגוריתם מבצע דייקסטרה מכל צומת, אם המשקלים אי שליליים.

$O(V^2 \log V + VE)$

הבעיה - אם המשקלים שליליים. קודם כל נריץ בלמן פורד לוודא שאין מעגל שלילי בגרף. אם יש מעגל שלילי בגרף - אז לבעיה אין פתרון.



שאלה - מה זה גרף דליל - מה המקסימום של קשתות שיכול להיות סדר גודל של $O(V^2)$

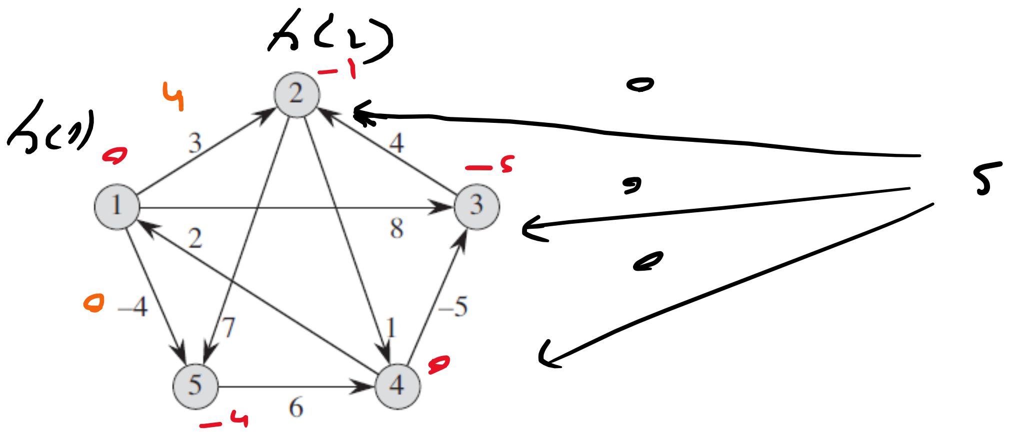
דליל מבחינתנו זה כל מספר שיוריד את הסיבוכיות הכתובה למעלה מ V^3

כדי שנוכל להריץ דייקסטרה נרצה להפוך את המשקלים לאי שליליים. ראיתם דוגמא שבה תוספת של קבוע - ראיתם שזה יכול לשנות את המסלולים כי כתלות באורך המסלול לכל מסלול יתווסף קבוע אחר.

נרצה לבחור פונקציית משקל חדשה

1. לא משנה את המסלולים הקיימים
2. המשקלים אי שליליים

מגדירים פונקציה חדשה על הקשתות באופן הבא - קודם כל לכל צומת נגדיר h(v) מספר



ובעזרת המספר הזה נגדיר פונקציית משקל חדשה. הדרך לעשות זאת - מוסיפים קודקוד חדש בשם s ומחברים אותו לכל הקודקודים הישנים בקשתות במשקל 0. ונגדיר לכל קודקוד v

$h(v) = \delta(s, v)$

כלומר המרחק הקצר ביותר מs לv. לכל קשת נגדיר עכשיו משקל חדש (u,v)

$$\hat{w}(u,v) = w(u,v) + h(u) - h(v)$$

טענה: המסלולים לא ישתנו, המשקלים יהיו אי שליליים.

נניח P מסלול מתקיים

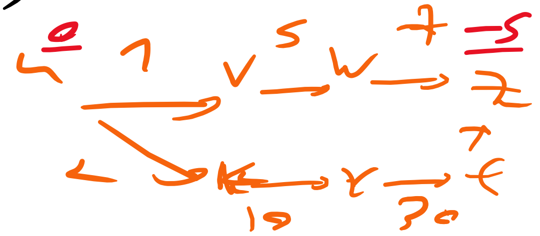
$$\hat{w}(p) = w(p) + h(u) - h(z)$$

$$p: \quad u \xrightarrow{1} v \xrightarrow{2} w \xrightarrow{3} z$$

$$w(u,v) + h(u) - \cancel{h(v)} +$$

$$w(v,w) + \cancel{h(v)} - h(w) +$$

$$w(w,z) + \cancel{h(w)} - h(z)$$



$$p_1 = 2+5+7 = 13$$

$$p_2 = 2+10+30 = 42$$

במשקלים החדשים - שני המסלולים נותרים במשקל שהיו בו קודם + (5-)-0

המסלולים הישנים ישתנו כולם רק בתוספת h(u)-h(z)

זה לא משפיע על המסלולים אלא תלוי רק בנקודת ההתחלה והסיום

*42 - התשובה לשאלה הגדולה של החיים היקום וכל השאר - לשאלות - ספרי מדריך הטרמפיסט לגלקסיה (דאגלס אדמס)

נשתכנע שהמשקלים החדשים אי שליליים - מאי שוויון המשולש

$$\underset{h(v)}{d(s,v)} \le \underset{h(u)}{d(s,u)} + w(u,v)$$

$$h(v) \le h(u) + w(u,v)$$

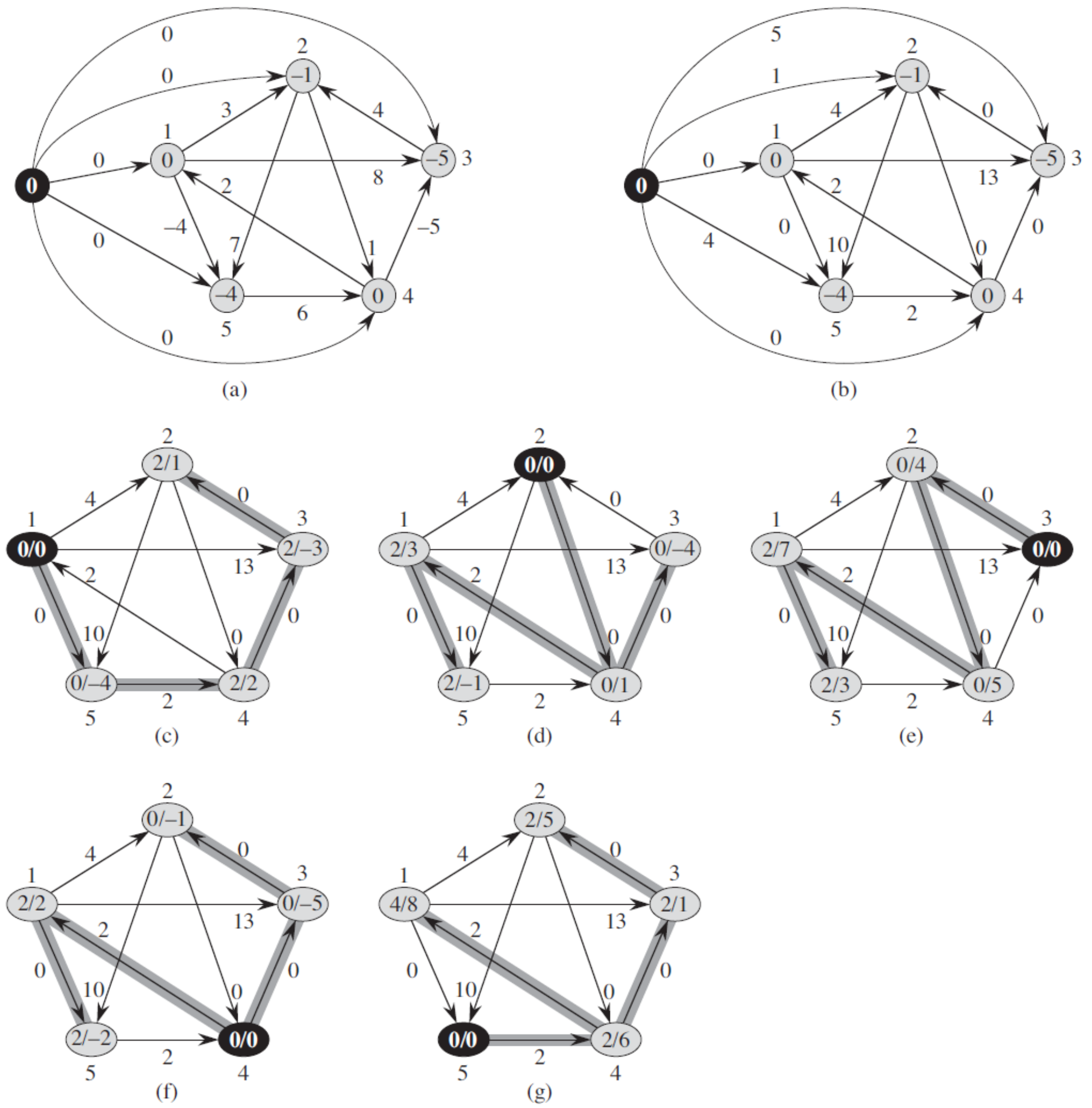
$$w(u,v) + h(u) - h(v) \ge 0$$

ואלו המשקלים החדשים.

הסדר בין המסלולים ישמר, למשל -

$-5 \longrightarrow 2$

$-1 \longrightarrow 3$



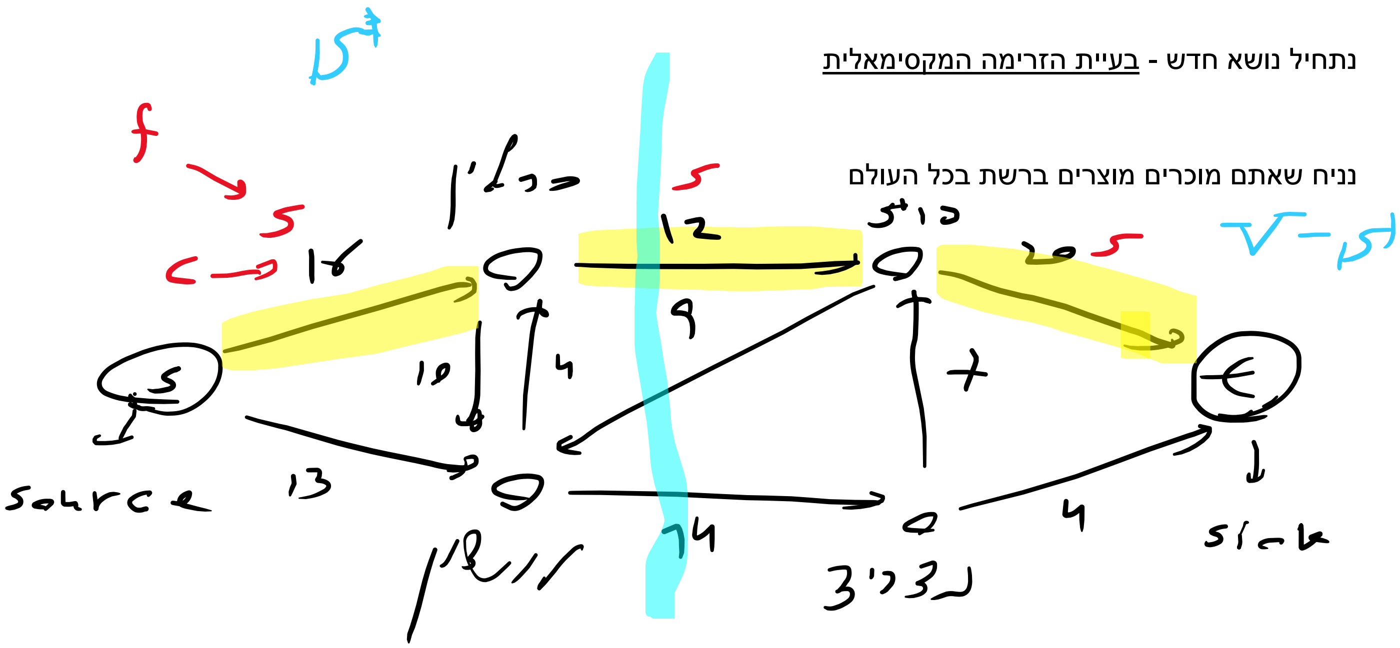
פסאודו-קוד

```
JOHNSON(G, w)
compute G', where G'.V = G.V ∪ {s},
      G'.E = G.E ∪ {(s, v) : v ∈ G.V}, and
      w(s, v) = 0 for all v ∈ G.V
if BELLMAN-FORD(G', w, s) == FALSE
      print "the input graph contains a negative-weight cycle"
else for each vertex v ∈ G'.V
            set h(v) to the value of δ(s, v)
                  computed by the Bellman-Ford algorithm
      for each edge (u, v) ∈ G'.E
            ŵ(u, v) = w(u, v) + h(u) − h(v)
      let D = (d_uv) be a new n × n matrix
      for each vertex u ∈ G.V
            run DIJKSTRA(G, ŵ, u) to compute δ̂(u, v) for all v ∈ G.V
            for each vertex v ∈ G.V
                  d_uv = δ̂(u, v) + h(v) − h(u)
      return D
```

אם נסכם - פגשנו אלגוריתם תכנון דינאמי שכל פעם מגדיל את גודל המסלול, אח"כ שיפרנו אותו שיפרנו עוד עם Floy-Warshall ואם הגרף היה דליל שיפרנו עוד עם Johnson.

נתחיל נושא חדש - <u>בעיית הזרימה המקסימאלית</u>

נניח שאתם מוכרים מוצרים ברשת בכל העולם



s - המקור (משם יוצאים המוצרים), t - הבור - לשם מגיעים המוצרים.

לכל קשת - מה הקצב המקסימלי של הזרמת מוצרים שיכול לעבור בקשת הזאת

המטרה - מציאת קצב מקסימאלי של מוצרים מs לt כאשר עומדים בדרישות הקיבול של כל קשת

f - זרימה

C - קיבול

<u>יותר פורמלית</u> גרף מכוון $G(V,E)$

1. לכל זוג צמתים יש קיבול

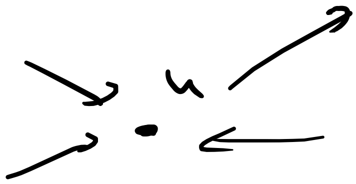
$$c(u,v) \geq 0$$

2. הזרימה צריכה להיות קטנה יותר מהקיבול

$$f(u,v) \leq c(u,v)$$

3. חוק השימור - מה שנכנס מצומת חייב לצאת (לא נשארים דברים בדרך)

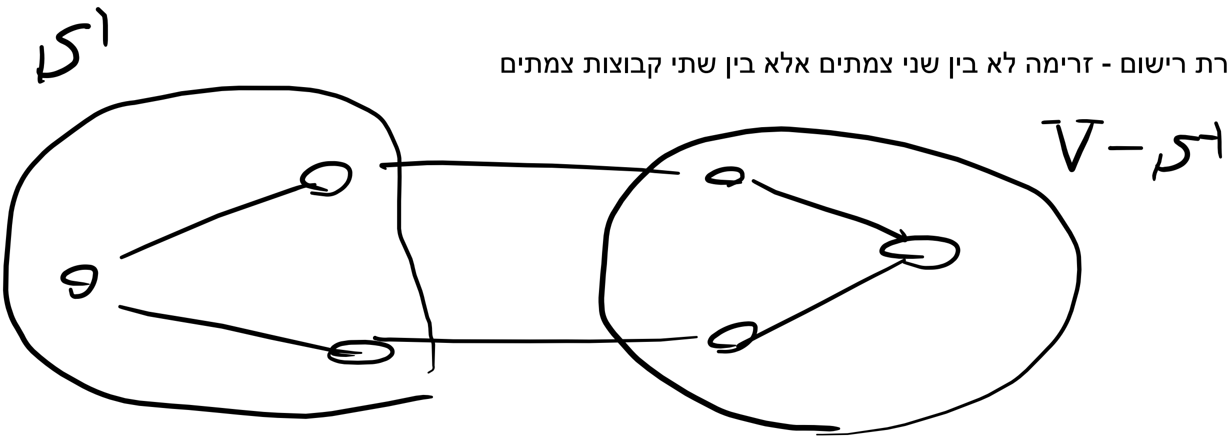
$$\sum_{v \in V} f(u,v) = 0$$



ערך הזרימה בגרף הוא מה שיוצא מs

$$|f| = \sum_{v \in V} f(s,v)$$

צורת רישום - זרימה לא בין שני צמתים אלא בין שתי קבוצות צמתים



הזרימה בין חלק שמאל של הגרף לחלק הימני תסומן כך

$$f(S, V-S)$$

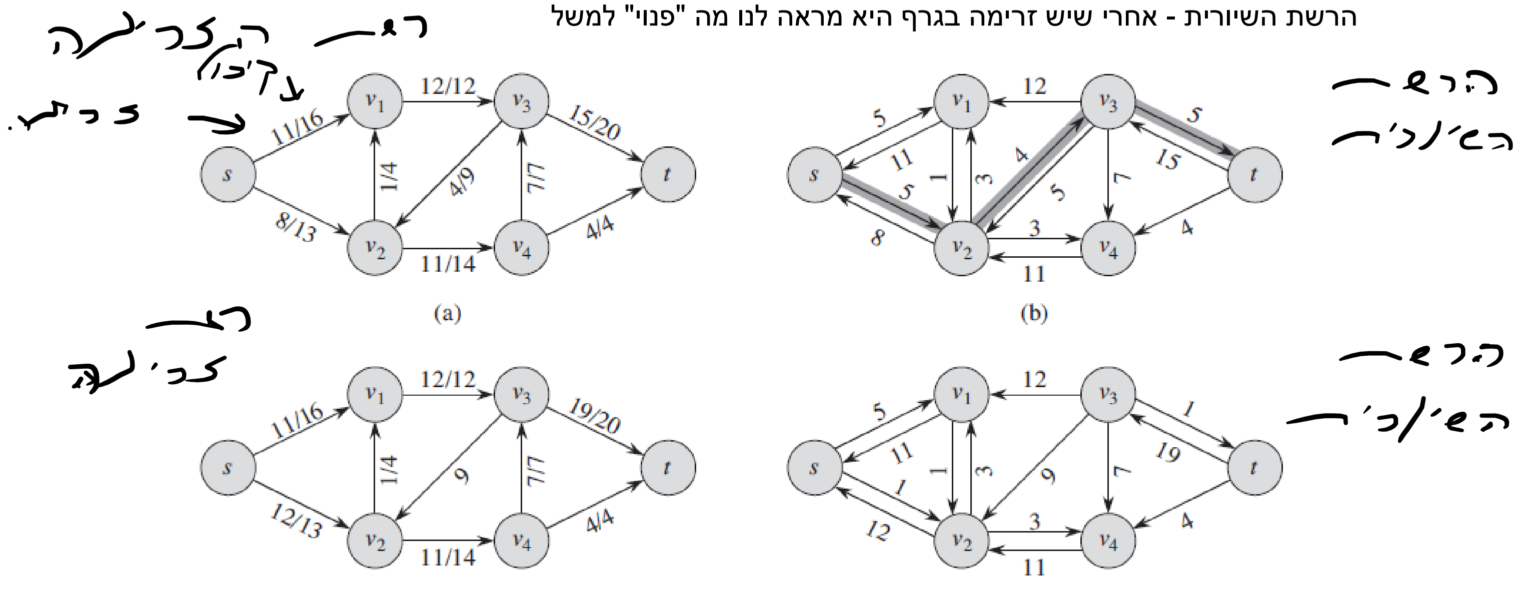
מתקיים

$$|f| = f(s,V) = f(V,t)$$

<u>פורד פולקרסון</u>

מתחילים ב$f(u, v)=0$ בכל איטרציה מחפשים מסלול שיפור (augmenting path) מוסיפים את הזרימה במסלול הזה והיא מגדילה את ערך הזרימה לאורך המסלול וחוזר חלילה עד שלא ניתן למצוא יותר מסלולי שיפור (המחשה עוד מעט) - זה נעשה בעזרת משהו שנקרא הרשת השיורית - residual network

הרשת השיורית - אחרי שיש זרימה בגרף היא מראה לנו מה "פנוי" למשל



(a) (b) (c) (d)

<u>מסלולי שיפור Augmenting paths</u>

מסלול מs ל t ברשת השיורית G_f

ניתן להגדיל את הזרימה במסלול P עד הקיבול המינימלי במסלול

$$c_f(p) = \min\{c_f(u,v) \mid (u,v) \in p\}$$

אם נסמן את הזרימה שהוספנו במסלול בfp הזרימה החדשה היא

f’ = f+ fp

$$|f'| = |f| + |f_p| > |f|$$

כלומר שיפרנו.

רשת השיורית   רשת הזרימה