אלגוריתמים - שיעור 6

$d(s,u) \le d(u)$ לכל u.

עץ המרחקים

במהלך האלגוריתם מתעדכנים מצביעי π, בכל Relax משפר.

(שארית זמן אקו כמו ב)

שעובר בכל כשלב שמעדכן, מצביעי π יוצרים עץ מכוון לפי השרשרת, והשורש הוא s

(בהנחה שאין מעגלים שליליים).

הוכחה: נכון בתחילת האלגוריתם המינימום שמעליו הוא ריק ומכיל רק את s.

הבעיה המורכבת ממצביעי π הינו קשיר ומתסר מעגלים (בגלל אי נתיבים מהכיוון).

לכל מצביע u עם מצביע $\pi \neq NIL$, קיים מסלול המתקבל מ-u דרך המצביעים.

$u \to u_2 \to u_3 \to \dots$

① אם המסלול מסתיים בצומת u_k $\pi(u_k)=NIL$ אז $u_k \equiv s$.

$\pi(u_{k-1})=u_k$. ההשמה הזו ל-$\pi(u_{k-1})$ קרתה בעקבות Relax משפר

על הקשת (u_k, u_{k-1}) בזמן כלשהו קודם $d(u_k)+w(u_k,u_{k-1})<d(u_{k-1})$ (I)

בפרט $d(u_k)<\infty$ ובאותו רגע $u_k \neq s$

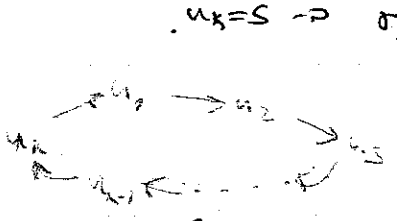
בפעם הראשונה ש-$d(u_k)$ מקבל ערך סופי, זה קרה ב- Relax משפר את

$d(u_k)$ ואז $\pi(u_k)$ מקבל ערך $\neq NIL$ ולכן לא יקבל NIL.

(לפני כן $u_k \neq s$) לכן המסלול חייב להתיקים ב- $u_k=s$.

* הקשתות הללו הולכות בכיוון ההפוך למצביעי π.

② אין מעגל: נניח בשלילה כי יש מעגל:



נניח בלי הגבלת הכלליות שבהשמה $\pi(u_{k-1})=u_k$ הינו האחרון שהתבצע במעגל

באותו רגע כל ההצבות: $\begin{cases}\pi(u_i)=u_{i+1} \\ \pi(u_k)=u_1\end{cases}$ קיימות.

נסתכל על המצב רגע לפני ההשמה $\pi(u_{k-1}) \leftarrow u_k$

כלומר, עומדים לבצע relax משפר על הקשת (u_k,u_{k-1}). כלומר, בגלל ש-

$d(u_{k-1})>d(u_k)+w(u_k,u_{k-1})$.

לכל i אחר, ההשמה $\pi(u_i) \leftarrow u_{i+1}$ התבצעה ב- Relax משפר קודם.

ומיד אחריו קיים השוויון $d(u_i)=d(u_{i+1})+w(u_{i+1},u_i)$.

לא שינוי שם הרגע הנוכחי, לפני ה- relax שמעדכן את המעגל

"חבון מתוה relax משפר מאותה יחד" (*)

(ברור כי לפרק הזמן יחד..)

בגלל (*):

אם כן כרגע נכון יחד לומר כי:

$d(u_i) \ge d(u_{i+1})+w(u_{i+1},u_i)$ (II) לכל ה-i-ים האחרים.

$$\sum_{i=1}^{k} d(u_i) > \sum_{i=1}^{k} d(u_i) + \sum_{i=1}^{k} w(u_{i+1},u_i)$$

(אלו אינדקסים האחרים)

נתבונן את כלפי:
את $(\Sigma)+(\Sigma)$

$$\Longrightarrow 0 > \sum_{i=1}^{k} w(u_i,u_i) \Longrightarrow$$ קיים מעגל שלילי!
לכן גרף המינימום

* הכאנו קשירות, כלומר מתוקרים ל-s. לכן הנתון שיש מעגל
⇐ אין מעגלים.

מצב לא אפשרי: מעגלים אחד לכל צומת.

⇐ אין מעגלים π ⇐ לכן גם לא מחזורים.

ברור שהאלגוריתם מסתיים, שהסיום $d(u)=\delta(s,u)$ לכל צומת u.

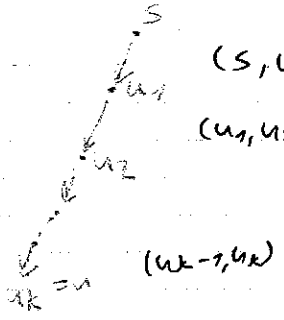
נפרוש על הלולאה G_π שקשתותיה הן מעגלי π (אין מעגלים שליליים). לולאה מכיוון
ששורשה s ולכן: ① הצמתים שלה הם בדיוק אלו הנגישים מ-s.

② לכל צומת u בלולאה, המסלול מ-s אליו הוא מקסימלי.

המסלול מ-u לשורש מורכב ממעגלי π ⇐ המסלול ההפוך מהשורש ל-u מורכב
מקשתות הלולאה.

הוכחה

① נניח v נגיש מ-s, כלומר $\delta(s,v)<\infty$ לפי ההנחה $d(v)=\delta(s,v)<\infty$ (בסיום האלגוריתם)
$\pi(v)$ לא $\neq NIL$ אחרת לכן v בלולאה.



הן הקשתות בלולאה G_π

ולהיפך באותו האופן.

② נוכיח באינדוקציה על k.

$k=0$ לכל השורש s. המסלול הוא ריק ומשקלו אפס, ואז אומנם מקסימלי מ-s ל-s.

נניח נכונות עד $k-1$ ונבדוק ל-k:

בסיום $\pi(u_i)=u_{i-1}$ לכן כמו קודם קיים בסיום $d(u_i) \geq d(u_{i-1}) + w(u_{i-1},u_i)$

אחרי השינוי יתקיים שוויון בין האלפים. לכן מתקיימים: $d(u_1) \geq d(s) + w(s,u_1)$

$$+\begin{cases} d(u_1) \geq d(s) + w(s,u_1) \\ d(u_i) \geq d(u_{i-1}) + w(u_{i-1},u_i) \\ \vdots \\ d(u) = d(u_k) \geq d(u_{k-1}) + w(u_{k-1},u_k) \end{cases}$$

$$\delta(s,u) = d(u) \geq \underbrace{\sum_{i=1}^{k} w(u_{i-1},u_i)}_{\text{לפי הנחה}} = \text{אורך מסלול מסוים מ-s ל-u.}$$

לכן המסלול הוא מקסימלי. ⇐

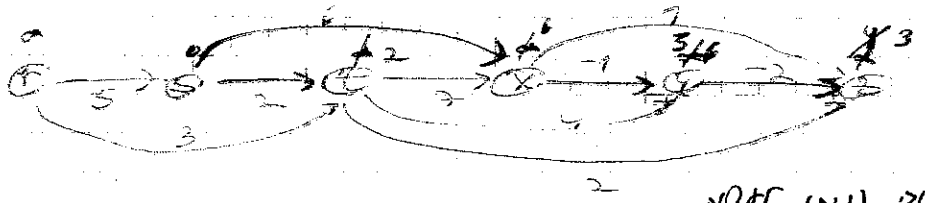
התשובות של כל המקסימום היא שהוא מספק קידוד לכל המקסימום מ-s לכל
הצמתים הנגישים מ-s.

בפרט ב-BFS היינו בריצים קידוד כזה, כי שמירת המסלולים לצמתים שונים מפרטים
עלולה לקחת $O(|V|^2)$ מקום.

למצוא את כל המסלולים בזמן רק $O(|V|)$ מקום

האלגוריתם: תישרה מלמעלה ב- DAG, שלא מכוון אי-דיק.

הרעיון נמצא בתוכנית DAG-Shortest-Paths



(1) נבצע מיון טופולוגי ונמצא לתשמ

צאת הכניסה יינאר- $O(|E|+|V|)$. המיון הטופולוגי של G = סדר בלתי של הצמתים

כך שכל קשת ב- G מכוונת מצומת קטן, לצומת גדול יותר.

נחל להתקדם מהצמתים הקודמים ל-S.

(2) אתחל נבצע k איטרציות. (כשלב S נשים 0 ולאחר אינסוף)

(3) עובר על כל הצמתים לפי סדר המיון הטופולוגי ומבצע מתבצע Relax על

הקשתות שיוצאות ממנו.

נוכיח את נכונות האלגוריתם: כאשר הלולאה המרכזית מגיעה לצומת u, מתקיים: $d(u)=\delta(s,u)$

הוכחה באינדוקציה על סדר המיון הטופולוגי.

נכון ל-S בשלב האיתחול $d(s)=0=\delta(s,s)$.

נניח נכונות עד צומת u כלשהו ונבדוק u. אם u לא נגיש מ-S אז $d(u)=\infty$

כי הרי אין $d(u) \geq \delta(s,u)$. (לפי האלגוריתם $\delta(s,u)=\infty$)

נניח שנגיש: $\underbrace{s \to u_1 \to u_2 \to \dots \to u_{k-1}}_{\text{מופיעים לפני } u \text{. מקיים } \delta(s,u_{k-1})} \to u_k = u$

כי הם מקיים המשלו מקיים

נכון לפי הנחת האינדוקציה $d(u_{k-1}) = \delta(s,u_{k-1})$ כשמגיעים ל- u_k (אם מבצעים

Relax על כל שכניו, כולל $u=u_k$. לאחר Relax זה מתקיים

$$d(u) \leq \underbrace{d(u_{k-1})}_{\delta(s,u_{k-1})} + w(u_{k-1},u)$$

$$\Leftarrow d(u) \leq \underbrace{\delta(s,u_{k-1}) + w(u_{k-1},u)}_{\delta(s,u_k)}$$

$\Leftarrow$ לכן: $d(u) \geq \delta(s,u)$ ומנגד קיים $d(u) \leq \delta(s,u)$

לכן מתקיים שוויון גם לצומת u.

הוכחנו באינדוקציה, לכן מתקיים לכל u.

$\Leftarrow$ האלגוריתם בסיומו, לצומת כל $d(u_k)=\delta(s,u_k)$, ומוצאים למשל את המסלולים הקצרים ביותר.

האלגוריתם של דיג'סטרה (גם בהוכחה)

קלט: גרף מכוון. משקלות אי שליליים. (אין מעגלים שליליים).

אתחול

לולאה – לכל קודקוד קיימת קשת (u,v) עם Relax משפר נקבע אותן

את האלגוריתם עובר פעם קיים $d(u)=\delta(s,u)$ לכל u.

הוכחה: נניח שלא, וקיימים בעיות v שבורם $d(v)\neq\delta(s,v)$ הפעם.

ניקח מעגל בעיות v עם מספר מינימלי של קשתות במקרה מ-s ל-v

(מתוך כל הבעיות הי"מקרקיים).

בגלל s מקרה

המקרה

לכן u בעיות לא מקרקית, כי לקחנו את v שהיו הכי קצר מקרקית,

ו-u קצר ממנו. $d(u)=\delta(s,u) \Leftarrow$

שכן שני המעגלים (*) מקרה $d(v)>\delta(s,v)=\delta(s,u)+w(u,v)=$

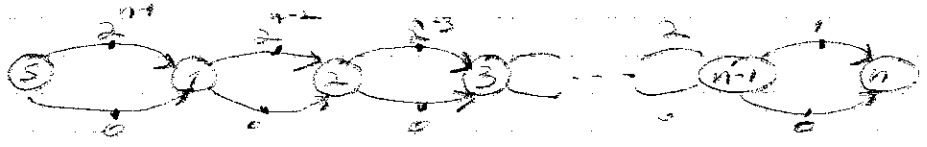
$$d(v)>d(u)+w(u,v)$$

לכן עכשיו קיים Relax משפר על (u,v). ואת בסתירה להנחה

שבאלגוריתם עבר כבר הוא עובר שבכל אין עוד Relax המשפרים.

לכן האלגוריתם עובר בוודאי $d(u)=\delta(s,u)$ לכל u.

לעומת זאת, מן הרבה יכול להיות אינם ונכון.



סדר Relax שבו נעשה מספר רב של פעמים מחלק יתנהגו ~ 2^n פעמים.

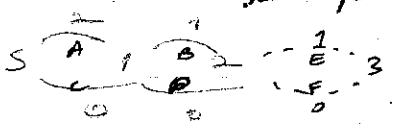
ישנם 2^n מסלולים אפשריים מ-s ל-n ומשקלם הוא $0,1,2,\dots,2^n-1$

נעשה Relax על הקשת המסומנת (s,1). כתוצאה נרים את האלגוריתם

על הבעיות (1...n) נעשה Relax על הקשת המסומנת (s,1) וכתוצאה

על כל השאר.

כמה מעין מסלול-אלגוריתם. כמו כאן כלל. לכן הוא לא כך מובטח.



מעין דוגמת הרבה לאלגוריתם

A (BD) C (BD)

A BEFDEF C

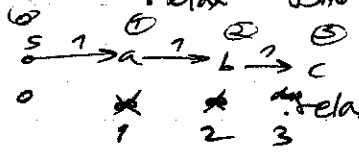
(*) (*)

אלגוריתם של בלמן-פורד BELLMAN-FORD

אם אין מעגלים שליליים, הסיבוכיות קיים מקסימום פשוט. לכן במקרה של M-1 קשתות

האלגוריתם - קיים בהנחה: $s \to u \to v$

(3-5) בכל איטרציה הוא עובר על כל הקשתות ומבצע relax.



סדר סריקה: $(s,a), (a,b), (b,c)$ איטרציה אחת

סדר ה-relax $\longrightarrow$

סדר לא: $(s,a), (a,b), (b,c) \to$ 3 איטרציות

כאן ריצה- $O(|V|\cdot|E|)$.

אבל הרי טוב אם היינו יודעים את "הסדר הנכון". מכיוון שאותנו לא יודעים, נשתמש

על פנינו על כל הצמתים, ועל כל הקשתות.

<u>נטענה:</u> בסיום $d(u) = \delta(s,u)$ לכל u. אם נסתכל בדרך מ-s ל-u שנבנה מ-s, ונסתכל

ה במקרה פשוט מ-s ל-u. $\underbrace{s \to u_1 \to u_2 \to \dots \to u_{k-1}}_{\text{בה במקרה לא } k-1 \text{ קשתות}} \to u$:k קשתות

נראה באינדוקציה על k שאחרי k איטרציות $d(u) = \delta(s,u)$

באיטרציה ה-k עוברים על כל הקשתות כולל (u_{k-1}, u_k) ועושים Relax

ואחריו מתקיים: $d(u) = d(u_k) \le d(u_{k-1}) + w(u_{k-1}, u_k) =$

$= \delta(s, u_{k-1}) + w(u_{k-1}, u_k) =$

$= \delta(s, u_k) = \delta(s,u)$

אבל כן: $d(u) \le \delta(s,u)$ מתמיד קיים אי-השוויון ההפוך, לכן השוויון מתקיים.

* שורת (ד-5) - אם יש מעגלים שליליים האלגוריתם שם את קשורת (ד-5).

שורה (6) בה ה- test ש- relax מבצע. כלומר אם (6) מתקיים, אזי קיים

עוד relax אשר, האלגוריתם יודע ב-(7) False

False ← יש מעגלים שליליים.

True ← אין מעגלים שליליים. והתכונה נרצה

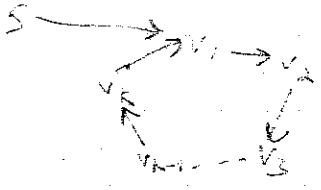
<u>נבדוק עכשיו נכונות</u> האלגוריתם יודע FALSE ⇔ קיים מעגל שלילי נגיש מ-s.

<u>הוכחה</u> (⇐) אם אין מעגל שלילי, הכוונה שבכל אחרי שנבנה מתקיים $d(u) = \delta(s,u)$

לכל u, ובפרט אין relax אשר לכן יחזיר TRUE.

(⇒) להיפך, נניח שקיים מעגל שלילי הנגיש מ-s. קל לראות שאחרי

$v-1$ איטרציות לכל הצמתים במעגל ש $d(v_i)$ כולו נגיש מ-s.



נלכן שקיים Relax אשר.

נניח שלא, ובפרט כל האילוצים המלא נבדקו והגענו אל מהן לא היה

Relax משפר. לכן,

$$+\begin{cases} d(v_2) \le d(v_1) + w(v_1, v_2) \\ d(v_3) \le d(v_2) + w(v_2, v_3) \\ \vdots \\ d(v_k) \le d(v_{k-1}) + w(v_{k-1}, v_k) \\ d(v_1) \le d(v_k) + w(v_k, v_1) \end{cases}$$

$$0 \le w(v_1, v_2) + \dots + w(v_k, v_1)$$

לכן זה אינו מעגל שלילי. סתירה

לכן הוכחנו את הכיוון השני.

* הלולאה מעגל לאחר מעגל שלילי.