המשך Single Source Shortest Spath

[[שיניתי את path ל-spath (the 's' is silent) כדי לתמוך בסטנדרט של שאר המשפט.]]

תזכורות

$\text{Relax}(u,v)$:

- אם $d[v] > d[u] + w(u,v)$, אזי:
  - $d[v] \leftarrow d[u] + w(u,v)$

[אם מבצעים שורה זו, נקרא לקריאה לפונקציה "Relax אפקטיבי"]
[[אחרת אפשר לקרוא לה "Relax פיקטיבי", ואז Relax שהוא לא פיקטיבי הוא Relax א-פיקטיבי – או בקיצור, Relax אפקטיבי.
(הערה: אין חשיבות להערה שכתבתי כאן whatsoever; פשוט תתעלמו ממנה ותישארו עם ההגדרה המקורית של Relax אפקטיבי.)]]

<u>אלגוריתם גנרי למציאת מק"ב ממקור s</u>

- <u>אתחול</u>:
  - $d[s] \leftarrow 0$
  - $d[v] \leftarrow \infty \quad \forall v \neq s$
- <u>צעד</u>:
  - נבחר קשת (u,v) ונבצע $\text{Relax}(u,v)$.
- <u>תנאי סיום</u>:
  - נסיים כאשר לא ניתן לעשות Relax אפקטיבי עבור אף קשת.
    כלומר, לכל $(u,v) \in E$ מתקיים:

$$d[v] \leq d[u] + w(u,v)$$

- <u>סיום</u>: נחזיר את $d[v] \quad \forall v$.

<u>טענה</u>

בכל שלב, לכל $v \in V$, אם $d[v] < \infty$ אזי $d[v]$ אורך/משקל של מסלול כלשהו מ-s ל-v.

<u>משפט (נכונות של אלגוריתם גנרי – מרחקים)</u>

תנאי הסיום מתקיים $\Leftrightarrow$ לכל $v \in V$ מתקיים $d[v] = \delta(s,v)$.

מסקנה

אם תנאי הסיום מתקיים אזי $d[s] = 0$.

<u>הסבר</u>:

אם $d[s] < 0$ אזי לפי הטענה יש מסלול שלילי מ-s ל-s – כלומר מעגל שלילי.

לכן $\delta(s,v)$ לא חסום מלמטה (אפשר לחזור כרצוננו על המעגל) ולכן מתקיים $\delta(s,v)<d[v]$ ולכן, לפי משפט הנכונות, תנאי הסיום לא יכול להתקיים.

הגדרה [עץ מושרש]

עץ מושרש זה משהו שנראה כך:

[ציור]

בגרף מכוון, זה תת-גרף מכוון T שגרף התשתית שלו (שזה גרף עם אותן הקשתות, אך לא מכוונות) הוא עץ פורש, ויש קדקוד מיוחד s ("השורש") כך ש- T מכיל מסלול מכוון מ- s לכל קדקוד $v\in V$.

הגדרה: עץ מסלולים קצרים ביותר

עץ מסלולים קצרים ביותר הוא עץ מושרש עם שורש s, בו המסלולים מ- s לשאר הקדקודים הם מק"ב.

בניית מסלולים קצרים ביותר

נשים לב: לכל קדקוד, מספיק לשמור את ההורה שלו בעץ [[ע"מ לשחזר את המסלולים הקצרים ביותר]].

נשנה את האלגוריתם הגנרי בהתאם – נוסיף לכל שדה $\pi[v]=$"ההורה של v":

$\text{Relax}(u,v)$:

- אם $d[v]>d[u]+w(u,v)$, אזי:
  - $d[v]\leftarrow d[u]+w(u,v)$
  - $\pi[v]\leftarrow u$

האלגוריתם:

- אתחול:
  - $d[s]\leftarrow 0$
  - $d[v]\leftarrow\infty \quad \forall v\neq s$
  - $\pi[v]\leftarrow \text{Null} \quad \forall v\in V$
- צעד:
  - נבחר קשת (u,v) ונבצע $\text{Relax}(u,v)$.
- תנאי סיום:
  - נסיים כאשר לא ניתן לעשות Relax אפקטיבי עבור אף קשת.

    כלומר, לכל $(u,v)\in E$ מתקיים:

$$d[v]\le d[u]+w(u,v)$$

- סיום: נחזיר את $d[v] \quad \forall v$.

דוגמה
[Gentlemen, on the count of 10, draw your graphs!]

הגדרה [זמן סיום של קדקוד]
זמן סיום של v = הצעד האחרון שבו Relax עדכן את $d[v]$ (כלומר, Relax אפקטיבי אחרון עבור קשת נכנסת (u,v) כלשהי).

<u>הערה</u>: אם תנאי הסיום מתקיים אזי ל- s יש זמן סיום ראשון (זמן 0).

משפט [מציאת עץ מק"ב]
אם תנאי הסיום מתקיים אזי התת-גרף:

$$T=\left(V,\left\{\left(\pi[v],v\right)\middle|\begin{matrix}v\in V\\ v\neq s\end{matrix}\right\}\right)$$

הוא עץ מק"ב.
[[דבר לא רלוונטי שמשעשע את תומר מס' 2^7+7 : אם מחליפים את שתי האותיות הראשונות והאחרונות בביטוי *עץ מק"ב* מקבלים *קב מעץ*. ☺]]

טענת עזר
בזמן הסיום של v, הגרף T כבר מכיל מק"ב מ- s ל- v שמורכב מקדקודים עם זמני סיום שקודמים ל- v.

הוכחת המשפט
- T עץ פורש: כי לכל $s \neq v$ יש הורה, ומס' הקשתות $|V|-1$.
  <u>הסבר</u>: לכל v יש מסלול מכוון מ- s ל- v בגרף (אמרנו בתחילת הנושא שלנו שאנו מניחים זאת ושניתן לבדוק זאת עם BFS); אם ל- v אין הורה אזי לא בוצע Relax אפקטיבי, ועדיין מתקיים $d[v]=\infty$, בסתירה למשפט הנכונות-מרחקים.

  כמו-כן, בזמן הסיום של v קיים מסלול מכוון מ- s ל- v ב- T, ומסלול זה לא משתנה אחרי זמן הסיום של v כי ערכי $\pi[\cdot]$ של שאר הקדקודים במסלול כבר עודכנו בפעם האחרונה.

- T עץ מק"ב: מאותה סיבה – יש מסלול מכוון מ- s ל- v ב- T בזמן הסיום של v, והוא נשאר ב- T עד סוף האלגוריתם.

הוכחת טענת העזר
באינדוקציה על זמני הסיום.

- <u>בסיס האינדוקציה</u>:

זמן הסיום הראשון הוא של s, ולפי המסקנה הנ"ל $d[s]=0$, וזהו המרחק מ-s ל-s, וזה גם אורך המסלול המנוון (s) שנמצא ב-T.

- <u>צעד אינדוקטיבי:</u>
תהי (u,v) קשת אחרונה ש-$\text{Relax}(u,v)$ עבורה מעדכן את v.
כיוון שה-Relax היה אפקטיבי, בסופו התקיים $d[v]=d[u]+w(u,v)$.
הערך $d[v]$ לא משתנה יותר (לפי ההנחה [[שזו הפעם האחרונה שהתבצע Relax אפקטיבי]]), אבל אם נניח בשלילה ש-$d[u]$ עוד מתעדכן בהמשך, אזי בסוף האלגוריתם יתקיים $d[v]>d[u]+w(u,v)$, בסתירה לתנאי הסיום.

לכן u סיים לפני v.
לפי הנחת האינדוקציה, באותו זמן היה מסלול מ-s ל-u שלא יכול להתעדכן יותר (וזמני הסיום של קדקודיו קודמים ל-u ובפרט ל-v), ומסלול זה הוא מק"ב מ-s ל-u.
נקרא למסלול זה p.
ה-$\text{Relax}(u,v)$ בתחילת הצעד הוסיף קשת (u,v) ל-T ועכשיו קיים מסלול $p'=p\circ(u,v)$ ב-T, שאורכו:

משפט הנכונות-מרחקים הנחת האינדוקציה

$$w(p')=w(p)+w(u,v)=\delta(s,u)+w(u,v)=d[u]+w(u,v)=d[v]=\delta(s,v)$$

ולכן p' מק"ב מ-s ל-v.

$\square$

תת-נושא: מק"ב עם מקור יחיד ומשקלות אי-שליליים

האלגוריתם של Dijkstra [דַיְקְסְטְרַה]

[ציור של חלל הפה ע"י המרצה בשביל להסביר כיצד לבטא את השם]

<u>האלגוריתם:</u>

- <u>נתחזק:</u>
  - S – קבוצת קדקודים שמרחקם חושב סופית
  - Q – שאר הקדקודים, ממוינים לפי ערך ה-$d[\,]$ שלהם
- <u>אתחול:</u>

- $d[s] \leftarrow 0$
- $d[v] \leftarrow \infty \quad \forall v \neq s$
- $\pi[v] \leftarrow \text{Null} \quad \forall v \neq s$
- $S \leftarrow \varnothing$
- $Q \leftarrow V$

- <u>צעד</u>:
  - כל עוד $Q \neq \varnothing$ ($S \neq V$):
    - נבחר $u \in Q$ עם $d[u]$ מינימלי
    - $Q \leftarrow Q \setminus \{u\}$
    - $S \leftarrow S \cup \{u\}$
    - לכל קשת (u,v) כך ש- $v \notin S$ נבצע:
      - $\text{Relax}(u,v)$
- <u>סיום</u>:
  - נחזיר את כל ה- $d[v]$, $\pi[v]$

דוגמה

[עיגולים עם קווים וחצים ומספרים]

הוכחת נכונות

- <u>אבחנה</u>: לכל קדקוד u, אחרי כניסת u ל- S, הערך $d[u]$ לא מתעדכן יותר.
- <u>משפט</u>: האלגוריתם של דַיְקסטרה מחזיר ערכים $d[v] = \delta(s,v)$ לכל $v \in V$.
- <u>טענת עזר</u>: בזמן כניסת u ל- S מתקיים $d[u] = \delta(s,u)$.

<u>הוכחת המשפט</u>:

האלגוריתם לא נתקע, כי כל עוד יש $v \notin S$ גם יש קשת (x,y) לאורך מסלול מ- s ל- v שיוצאת מ- S, ו- $\text{Relax}(x,y)$ גורם לכך ש- $d[y] < \infty$ [[ולכן גם כל קדקוד מתישהו נכנס ל- S]].

כמו-כן, לכל $u \in V$, $d[u] = \delta(s,u)$ בזמן כניסת u ל- S, ולפי האבחנה, ערך זה לא משתנה במהלך שאר האלגוריתם.

□

את טענת העזר נוכיח באמצעות:

- <u>"טענת עזר"-עזר</u>: לכל מסלול קצר ביותר שמסתיים בקשת (x,y) כך ש- $x \in S$ וְ- $y \notin S$, בתחילת/סוף צעד כלשהו מתקיים $d[y] = \delta(s,y)$.

הסבר ציורי:
[ציור הסברי]