TA Session 9: Approximations

רדוקציות שמשמרות יחס קירוב

<u>רדוקציה משמרת יחס קירוב בין בעיה A לבעיה B</u> היא זוג פונקציות (f,g) כך ש:

$$f: A \to B, \qquad g: S(B) \to S(A)$$

כאשר $S(X)$ מסמן את מרחב הפתרונות של X.

אם נתון אלגוריתם $Algo$ שהוא k-מקרב עבור B, אזי $g\big(Algo(f)\big)$ הוא אלגוריתם k-מקרב עבור A.

לדוגמה, עבור בעיית $3\text{-}CNF\text{-}SAT$. נשתמש ב-f כדי להעביר את הבעיה לבעיית 3-צביעה, נשתמש באלגוריתם צביעה שהוא k-מקרב, ונקבל פתרון φ. ונשתמש ב-g כדי להעביר את הפתרון ל- φ', השמה מספקת ל- $3\text{-}CNF$.

$$3\text{-}CNF\text{-}SAT \xrightarrow{f} 3\text{-}COLOR \xrightarrow{Algo} \varphi \xrightarrow{g} \varphi'$$

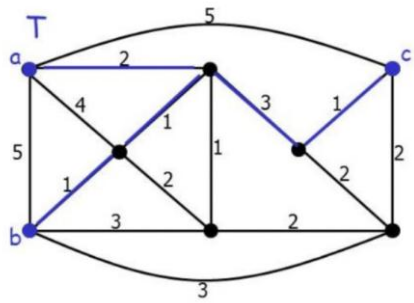
באופן כללי:

1. נקבל קלט x לבעיה A.
2. נמיר אותו לקלט $y = f(x)$ לבעיה B.
3. נפעיל אלגוריתם קירוב ל-y ונקבל פתרון s.
4. נמיר את s לפתרון $s' = g(s)$ עבור x.

עץ שטיינר

נתון גרף עם משקלים אי-שליליים על הצלעות, וחלוקה של הקודקודים ל-2 קבוצות R, S.

המטרה היא למצוא עץ מינימלי שפורש את כל R. (מותר לכלול קודקודים מ-S אבל לא חובה).



עץ שטיינר מטרי

אותה הגדרה, אבל הגרף מלא והמשקלים מקיימים את אי-שוויון המשולש. כלומר $w(uv) \le w(ux) + w(xv)$ לכל x, u, v.

נמצא רדוקציה משמרת קירוב בין עץ שטיינר (G,T) לעץ שטיינר מטרי (G',T').

$$f(G) = G', \qquad g(T') = T$$

נגדיר את $f(G)$: אותם קודקודים כמו ב-G, והמשקל של כל צלע uv הוא המרחק הכי קצר בין v ל-u ב-G. המרחק הקצר מקיים את אי"ש המשולש.

נגדיר את $g(T')$: לכל צלע ב-T' ניקח את המסלול שהיא מייצגת ב-G, ובסוף ניקח עץ פורש על מה שמתקבל. בגלל שזה עץ פורש ב-G' על כל הקודקודים של R, זה אומר שיש מסלול בין כל שני קודקודים ב-R דרך T'. אז גם ב-T יהיה מסלול.

אנחנו צריכים להוכיח:

1. $OPT(G) \ge OPT(G')$.
2. לכל עץ T' מתקיים $cost(T) \le cost(T')$.
3. יש אלגוריתם k-מקרב עבור הבעיה המטרית.

נוכיח את 1:

$OPT(G)$ זה המשקל של העפ"מ של G. מתקיים לפי תהליך הבנייה: $\mathrm{cost}_G(uv) \ge \mathrm{cost}_{G'}(uv)$, כי ניקח את המשקל הכי נמוך שמחבר ביניהם.

אז המשקל האופטימלי ב-G' הוא לכל היותר המשקל האופטימלי ב-G.

נוכיח את 2:

מכיוון שלוקחים עבור כל צלע את המסלול שהיא מייצגת (שהיא באותו משקל), המשקל של כל מסלול בין שני קודקודים נשאר זהה.

יכולות להיות צלעות שחוזרות על עצמן, ובמקרה הזה זה רק מוריד את המשקל.

בסה"כ, התחלנו עם G ועברנו ל-G' שבו יש פתרון אופטימלי יותר טוב.

מוצאים פתרון k-מקרב T' ב-G' ומעבירים אותו ל-T, והמשקל של T נמוך יותר מ-T'. אז:

$$cost(T) \leq cost(T') \leq k \cdot OPT(G') \leq k \cdot OPT(G)$$

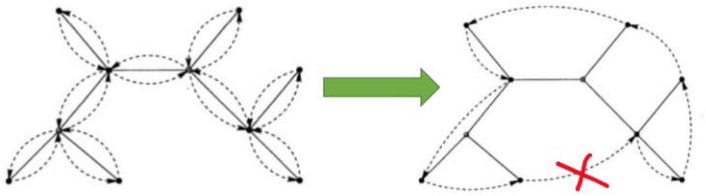
אלגוריתם מקרב לעץ שטיינר מטרי

נמצא עץ פורש מינימלי ב-G'. נקרא לו T.

<u>טענה</u>: האלגוריתם הזה הוא 2-מקרב. ($cost(T) \leq 2OPT(G)$).

<u>הוכחה</u>: יהי T^* פתרון אופטימלי, במשקל $OPT(G)$.

נכפיל את הצלעות של T^*, ונקבל הילוך אויילר:



נעשה קיצורי דרך בהילוך כך שלא יהיו קודקודים שחוזרים על עצמם ושלא יהיו קודקודים של S (אפשר, כי זה גרף מלא).

נקבל מעגל המילטון. נסיר ממנו צלע שרירותית ונקבל עץ פורש T' (לא בהכרח מינימלי) של $G[R]$.

$$cost(T) \leq cost(T') \leq 2cost(T^*) = 2OPT(G)$$

<u>טענה</u>: האלגוריתם $\left(2\left(1-\frac{1}{|R|}\right)\right)$-מקרב.

<u>הוכחה</u>: יהי T^* פתרון אופטימלי, במשקל OPT. נכפיל את הצלעות של T^* ונקבל הילוך אויילר.

נעשה קיצורי דרך בהילוך כך שלא יהיו קודקודים שחוזרים על עצמם או קודקודים מ-S.

נקבל מעגל המילטון C, נסיר ממנו את הצלע <u>בעלת המשקל הכי גדול</u> ונקבל עץ פורש T' (לא בהכרח מינימלי) של $G[R]$.

$$cost(T) \leq cost(T') \overset{\text{א}}{\leq} cost(C) - \frac{cost(C)}{|C|} \overset{\text{ב}}{=} cost(C) - \frac{cost(C)}{|R|} \overset{\text{ג}}{=}$$
$$= cost(C) \cdot \left(1 - \frac{1}{|R|}\right) \overset{\text{ד}}{\leq} 2 \cdot cost(T^*) \cdot \left(1 - \frac{1}{|R|}\right) \overset{\text{ה}}{=} 2 \cdot OPT(G) \cdot \left(1 - \frac{1}{|R|}\right)$$

א. שובך היונים, יש צלע במשקל לפחות $cost(C)/|C|$.
ב. C הוא מעגל המילטון על הקודקודים של R.
ג. גורם משותף.
ד. המשקל של C הוא לכל היותר פעמיים המשקל של T^*.
ה. בהגדרה, T^* הוא במשקל אופטימלי.

כנדרש. ■

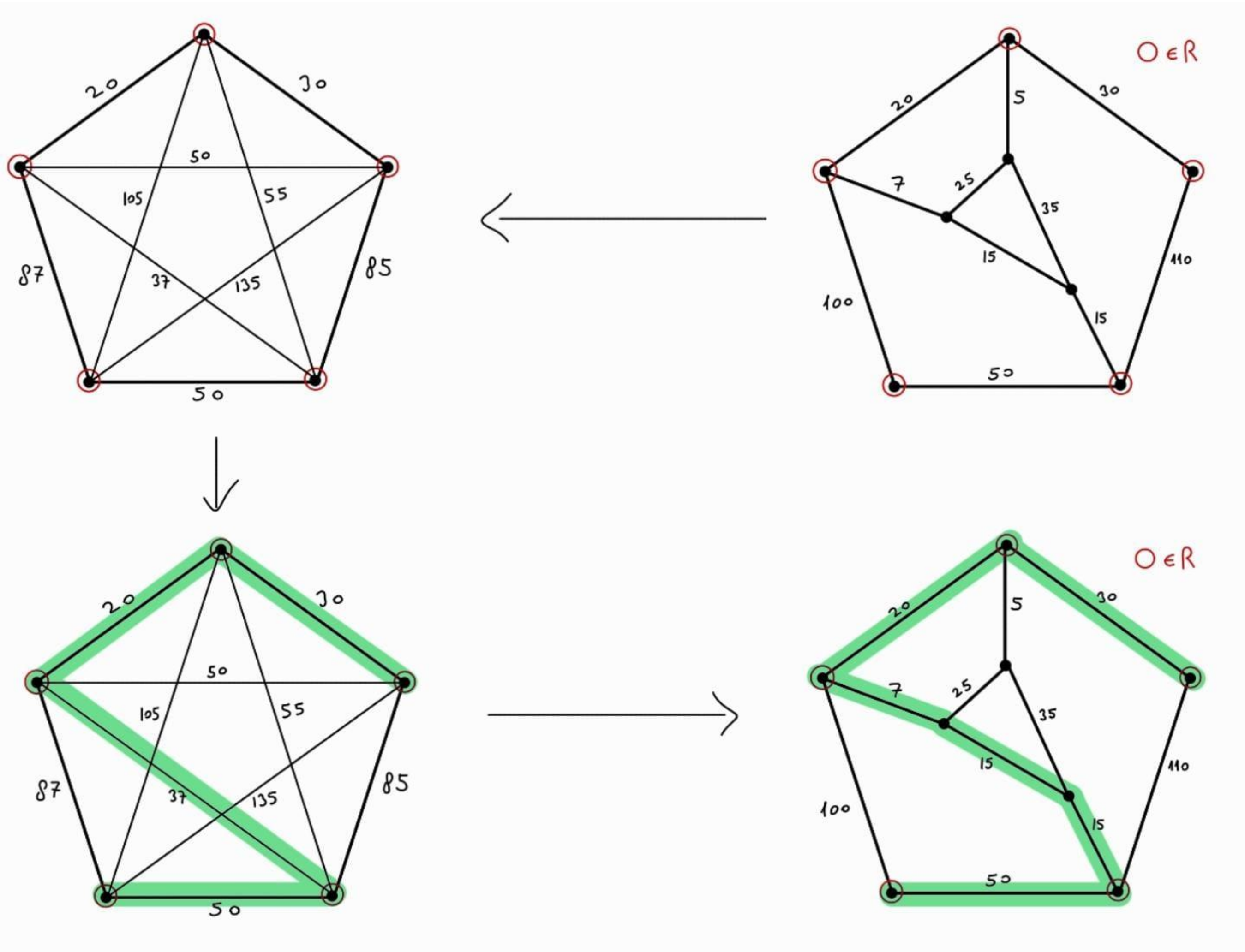
TA Session 9: Approximations

דוגמה

נריץ את האלגוריתם על הגרף:

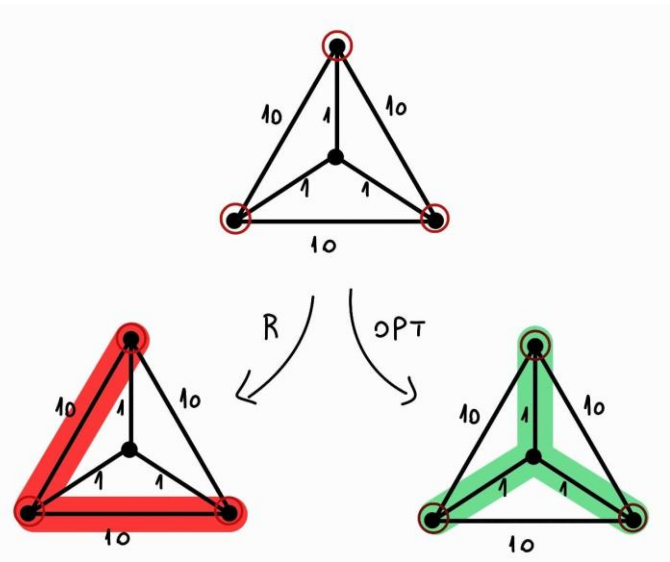
במעבר לקליקה, לא ציירנו את הקודקודים של S כי לא צריך אותם. העץ יהיה רק בין קודקודים של R.

כשנמיר חזרה למסלולים בעץ המקורי, נשתמש גם בקודקודים של S.



דוגמה 2

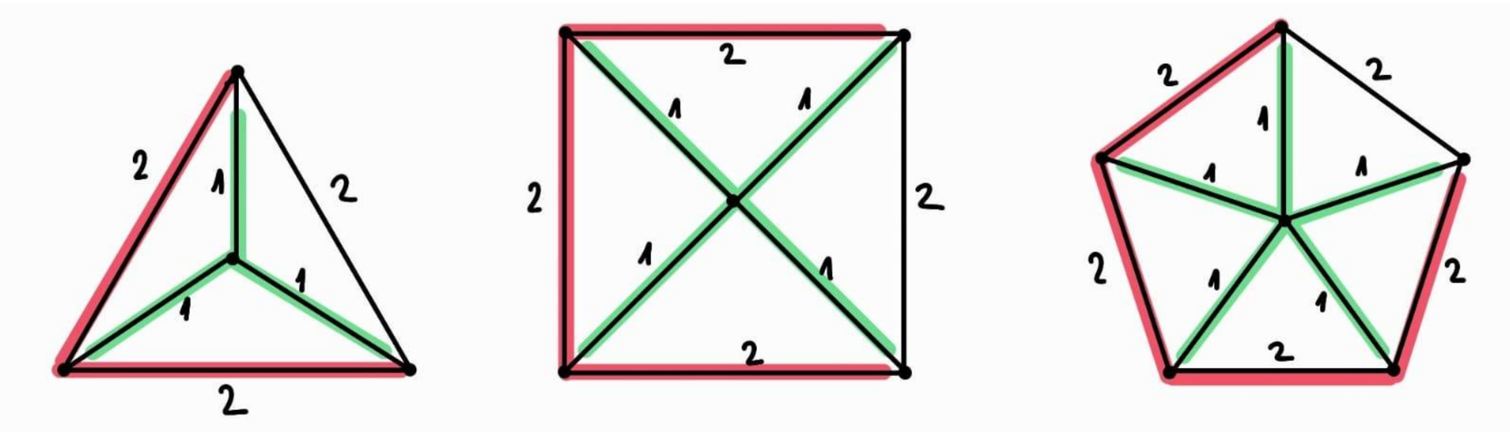
בבעיה המטרית, MST על הקודקודים של R נותן פתרון 2-מקרב. במקרה הכללי (לא מטרי) זה לא עובד:



דוגמה 3

ניתן 3 דוגמאות לבעיה <u>המטרית</u> כך ש:

1. $S \neq \emptyset$
2. ל-MST על R יש משקל גבוה יותר מהפתרון האופטימלי.



תרגיל

נתון גרף קשיר G עם משקלים אי-שליליים על הצלעות. הקודקודים מחולקים ל-2 קבוצות: R – לקוחות, S – ספקים.

צריך למצוא תת גרף במשקל מינימלי, כך שלכל לקוח יש לפחות ספק אחד ברכיב הקשירות שלו.

נמצא אלגוריתם 2-מקרב: נעשה רדוקציה לעץ שטיינר, ונשתמש באלגוריתם עבור שטיינר.

נוסיף ל-G קודקוד v נוסף, שמחובר לכל הקודקודים ב-S, עם צלע במשקל 0. נקרא לגרף החדש G'.

נפעיל את האלגוריתם ה-2-מקרב של שטיינר על G', כאשר $R' \coloneqq R \cup \{v\}$. יהי T העץ המתקבל. הפתרון יהיה $T' \coloneqq T \setminus \{v\}$.

האלגוריתם ימצא עץ פורש של R', שזה כולל את v. והוא מחובר לכל S, אז כל קודקוד ב-R יהיה עם קודקוד מ-S ברכיב קשירות.

<u>נוכיח שהוא 2-מקרב</u>: יהי OPT הפתרון האופטימלי לבעיה.

נשים לב ש $E(OPT) \cup \{(v,s): s \in S\}$ הוא תת גרף במשקל של OPT. ואם נמחק צלעות בין S ל-v נקבל עץ T^* שהוא עץ שטיינר כלשהו. אז:

$$\mathrm{OPT}_{steiner} \leq E(OPT) \cup \{(v,s): s \in S\} = OPT$$

אז:

$$T' \leq T^* \leq 2 \cdot \mathrm{OPT}_{steiner} \leq 2 \cdot OPT$$

הגדרה: קבוצה שלטת היא קבוצה של קודקודים כך שכל קודקוד בגרף הוא שכן של הקבוצה (או בקבוצה עצמה).

עבור קבוצת קודקודים כלשהי, נגדיר:

- קודקוד בקבוצה – שחור,
- קודקוד עם שכן בקבוצה – אפור,
- קודקוד לא בקבוצה ובלי שכן בקבוצה – לבן.

ונסמן $w_S(v)$ את מספר השכנים הלבנים של v.

עבור קבוצה שלטת, אין קודקודים לבנים בגרף.

אלגוריתם מקרב עבור קבוצה שלטת

1. נאתחל $S = \emptyset$.
2. כל עוד יש קודקודים לבנים בגרף,
   a. נבחר את הקודקוד עם $w_S(v)$ הכי גדול ונוסיף ל-S.
3. נחזיר את S.

נוכיח שהוא $O(\ln(\Delta(G)))$-מקרב:

נגדיר לכל קודקוד את ה"עלות" של הפיכה מלבן לאפור או שחור. אם בחרנו קודקוד v:

- אם v לבן, לכל קודקוד לבן שנהפך לאפור/שחור בעקבות הבחירה ניתן עלות $1/(w_S(v)+1)$.
- אם v אפור, לכל קודקוד לבן שנהפך לאפור בעקבות הבחירה ניתן עלות $1/(w_S(v))$.

נסמן את העלות הזו $ac(v)$.

אבחנה: $|S| = \sum_{v\in V(G)} ac(v)$, כי בכל בחירה של קודקוד הוספנו בדיוק 1 לסכום.

יהי $S^* := \{v_1, v_2, \ldots, v_t\}$ הפתרון האופטימלי.

נסמן $S_1, S_2, \ldots, S_t$ את הכוכבים שהמרכזים שלהם הם $v_1, \ldots v_t$. הכוכבים לא בהכרח זרים.

מספיק להוכיח ש

$$ac(S_i) := \sum_{u\in S_i} ac(u) = O(\ln(\Delta(G)))$$

למה? כי אם נכפיל הכל ב-t:

$$|S| = \sum_{v\in V(G)} ac(v) \le \sum_{i=1}^{t} ac(S_i) = t\cdot O(\ln(\Delta(G)))$$

במקרה הגרוע בריצה של האלגוריתם, כל קודקוד אפור נצבע אפור ע"י בחירה שונה של קודקוד שהפכנו לשחור.

כלומר בכל בחירה כיסינו רק עוד קודקוד אחד. נניח שזה המצב.

עבור S_i כלשהו, יהיו $w_1, \ldots, w_\ell$ קודקודי הכוכב. ויהיו $u_1, \ldots, u_\ell$ הבחירות שהפכו את $w_1, \ldots, w_\ell$ לאפורים.

כלומר $\ell = |V(S_i)| = \deg_{S_i}(v_i) + 1$.

נסמן T_j את הרגע לפני בחירת הקודקוד u_j שהפך את w_j לאפור או שחור.

בזמן הזה יש בדיוק $j-1$ קודקודים אפורים/שחורים.

מצד אחד האלגוריתם חמדן אז הוא צובע את הקודקוד עם המספר המקסימלי של שכנים לבנים. כלומר:

$$w_{T_j}(u_j) \ge w_{T_j}(v_i)$$

מצד שני,

$$w_{T_j}(v_i) \ge \deg(v_i) - (j-1) = \deg(v_i) - j + 1$$

אזי,

$$ac(w_j) \le \frac{1}{w_{T_j}(u_j)} \le \frac{1}{w_{T_j}(v_i)} \le \frac{1}{\deg(v_i) - j + 1}$$

ומתקיים:

$$ac(S_i) = \left(\sum_{i=1}^{\deg(v)} ac(w_i)\right) + ac(v_i) \le \left(\sum_{i=1}^{\deg(v)} \frac{1}{\deg(v_i) - j + 1}\right) + 1 \le \left(\sum_{k=1}^{\deg(v)} \frac{1}{k}\right) + 1 = O(\ln(\deg(v))) \le O(\ln(\Delta(G)))$$

כנדרש. ■