www.wisdom.weizmann.ac.il/~peleg

בעיות הכרעה - התשובה כן או לא.

<u>סיבוכיות</u>

אלגוריתם A

קלט x

$length(x)$

$T_A(x)$ - פונקציית זמן ריצה

$T_A(n)$ - זמן ריצה מקסימלי על ~~קלט~~ קלט בגודל n.

האלגוריתם הוא פולינומי אם קיים פולינום $p(n)$ כך ש:

$T_A(n) \le p(n)$ לכל n.

בעיה היא פולינומית אם קיים עבורה אלגוריתם פולינומי.

P : { Π | Π פולינומית }

Exp : { Π | Π אקספוננציאלית }

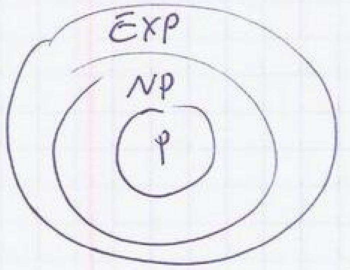
אלגוריתם לא דטרמיניסטי:

1. בתחילת התהליך רושם "ניחוש" באופן לא דטרמיניסטי
2. ממשיך באופן דטרמיניסטי - A_{det} מופעל על הקלט והניחוש.

A הוא אלגוריתם לא דטרמיניסטי עבור בעיה Π אם לכל קלט מתקיים:

1. התשובה צריכה להיות כן $=\Pi(x)$ אז קיים ניחוש אפשרי שיגרום ל-A_{det} להחזיר כן.
2. אם התשובה צריכה להיות לא $=\Pi(x)$ אז ~~לכל קיים~~ כל ניחוש ~~ב~~ יחזיר לא.

$NP = \{ \Pi \mid$ קיים ל-Π אלגוריתם לא דטרמיניסטי פולינומי $\}$



$P \subseteq NP \subseteq EXP$

הגדרה: $f: \Sigma_1^* \to \Sigma_2^*$

f העתקה פולינומית משפה $L_1 \subseteq \Sigma_1^*$ לשפה $L_2 \subseteq \Sigma_2^*$ אם:

1. f ניתנת לחישוב בזמן פולינומי (ב- $length(x)$)
2. f מעתיקה את L_1 ל- L_2, כלומר $x \in L_1 \iff f(x) \in L_2$

במקרה שקיימת העתקה פולינומית נסמן זאת כך $L_1 \propto L_2$.

<u>עובדה ב:</u> אם L_2 היא בעיה פולינומית אז גם L_1 פולינומית.

השפה L היא NP-שלמה אם:

1. $L \in NP$
2. לכל שפה $L' \in NP$ מתקיים $L' \propto L$.

3.

<u>עובדה ד:</u>

אם קיימת בעיה NP שלמה ששייכת ל- P אז $P = NP$.

<u>עובדה 4:</u>

SAT היא בעיה NP שלמה.

<u>עובדה 5:</u>

3SAT היא NP שלמה (NPC).

עובדה 6

הרדוקציות פולינומיות הן טרנזיטיביות

$L_1 \propto L_2 \ , \ L_2 \propto L_3 \Rightarrow L_1 \propto L_3$

עובדה 7

אם ידוע ש- L_1 היא NPC ורוצים להוכיח ש- L_2 שלמה מספיק להראות:

1. $L_2 \in NP$

2. $L_1 \propto L_2$.

תרגיל ח:

הוכח שבעיית 3 צביעה היא NP שלמה. (ע"י רדוקציה מבעיית 3SAT).

רדוקציית טיורינג ($\propto_T$)

$L_2 \propto_T L_1$ אם:

בהינתן פרוצדורה P שפותרת את L_1 (בזמן $O(1)$), קיים אלגוריתם פולינומי A עבור הבעיה L_2

בעיה L היא NP קשה אם לכל $L' \in NP$ מתקיים $L' \propto_T L$

עובדה: TSP bound היא NP קשה.

תרגיל:

להראות בעיית "קבוצה שולטת" ניתנת לרדוקציה עצמית.

קבוצה שולטת בגרף נתון $V \subset W$ כך: קבוצה שולטת אם לכל $w \notin V$ יש שכן ב-W

בגירסת ההחלטה, האם יש קבוצה שולטת בגודל $\le k$.

אלגוריתמים פסאודו - פולינומיים

בעיית התרמיל:

קלט: n סוגי פריטים $1,2,\dots n$. לכל סוג יש גודל s_i ומחיר c_i. תרמיל בגודל k. רווח מטרה - B.

שאלה: האם קיימים טבעיים $x_1 \dots x_n$ כך שסך נפח הפריטים לא עובר מ-k והרווח לפחות B.

תרמיל אחיד: $c_i = s_i$ וגם $k = B$.

תרמיל 0-1: $x_i \in \{0,1\}$.

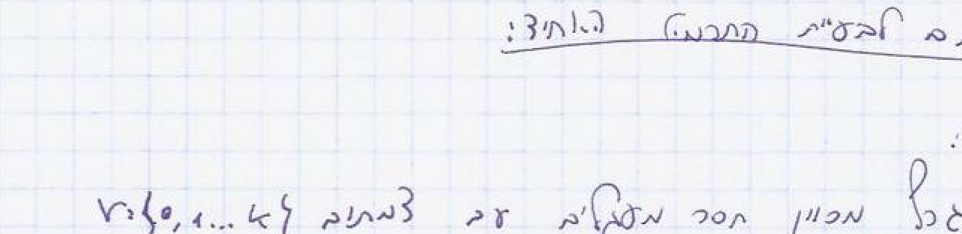
אלגוריתם לבעיית התרמיל האחיד:

שלב I:

נבנה גרף מכוון חסר מעגלים עם צמתים $V = \{0,1,\dots k\}$

והקשתות— ~~$E = \{(A,B) \mid$~~

$$E = \{(A,B) \mid 0 \le A < B \le k \,,\ \exists c_i : A + c_i = B\}$$



$k = 8$

$C = \{2, 3\}$

c_1, c_2

שלב II

נבדוק אם קיים ב-G מסלול מ-0 ל-k

נדבר על בעיות שהקלט שלהן מורכב מסדרה של מספרים.

$a_1, \ldots, a_n$.

~~נסמן~~ לקלט x, נסמן $max(x)$ = המספר הגדול ביותר בקלט

אורך הקלט בייצוג בינארי = $length(x)$.

אלגוריתם פסאודו-פולינומי $\forall x, \underbrace{T_A(x)}_{\text{זמן הריצה}} \leq p(length(x), max(x))$

בעיות NP קשות במובן החזק

בעיות שאין עבורן אלגוריתם פסאודו-פולינומי.

בע"ה Π היא NP קשה במובן החזק אם גם הצמצום המוגבל של Π_q, עדיין NP קשה, כאשר Π_q היא הבעיה Π מוגבלת לקלטים x שעבורם $max(x)$ היא לא יותר מ-$q(length(x))$ (q=פולינום). $max(x) \leq q(length(x))$.

אלגוריתם פסאודו-פולינומי ל-תרמיל 0-1.

אלגוריתם תכנות דינאמי. נמלא טבלה בגודל $n \times k$.

	0	1	2 ...	k
1				
2				
3				
⋮				
n				

ערך בנקודה $T(i,b)$ הוא 1 אם

$T(i,b) = 1$ אם ניתן למלא תרמיל בגודל b ע"י פריטים מבין הפריטים $1, \ldots, i$

$T(i,b) = 0$ אחרת.

כלל למילוי הטבלה.

$$T(i,b) = \begin{cases} 1 & i=1 \text{ ו- } b=a_1 \text{ או } b=0 \\ 0 & \text{אחרת} \end{cases}$$

נניח שמילאנו עד שורה i

$$T(i+1,b)=\begin{cases}1, & \text{אם } T(i,b)=1 \text{ או } T(i,b-c_{i+1})=1\\ 0 & \text{אחרת}\end{cases}$$

בסוף: בודקים אם $T(n,k)=1$

תכנות דינמי ל- TSP

מתחילים מעיר 1. $M=\{2,\dots,n\}$ שאר הערים.

נמלא טבלה $T(j,S)$ עם ת"א לכל $S\subseteq M$ ולכל $j\in S$. שיכיל את אורך המסלול הקצר ביותר שמקיים את התכונות הבאות:

1. מתחיל ב-1
2. מבקר בדיוק בכל עיר מהקבוצה $S\cup\{1\}$ ומסתיים בעיר j.

תחילת המילוי:

ל- $|S|=1$ $(S=\{j\})$

$T(j,S)=d_{1j}$

נניח שמילאנו את הטבלה לכל S מגודל $i\le|S|$ ונמשיך לבנות לגודל $|S|=i+1$.

$S'=S\setminus\{j\}$

$$T(S,j)=\min_{k\in S\setminus\{j\}}\left(T(S\setminus\{j\},k)+d_{kj}\right)$$

תשובה סופית: $\min_{j=2..n}\left(T(M,j)+d_{j1}\right)$

רדוקציה:

3SAT

קלט: נוסחה ב- CNF

$$\varphi = \bigwedge_{i=1}^{m} c_i$$

כל פסוקית מהצורה $c_i = (v_1^i, v_2^i, v_3^i)$

כל v_i הינו $\{x, \bar{x}\}$

המשתנים $X = \{x_1, \dots, x_n\}$

השמת אמת. $f : X \to \{T, F\}$

האם קיימת השמת אמת שמקיימת את F.

X3SAT (exact).

האם קיימת השמת אמת f כך שבכל פסוקית בדיוק ליטרל אחד מסופק (שני האחרים מופרכים)

נניח $f(x_1) = T$

$$c = (x_1, x_2, x_3) = T$$
$$T \quad F \quad F$$

$f(x_1) = F$

$$c = (x_1, u_2, u_3)$$
$$F \quad ? \quad ?$$

$$f(u_2) \neq f(u_3)$$

$$f(u_2) = f(\bar{u}_3)$$

צורה קנונית

φ בצורה קנונית אם בכל פסוקית יש 3 משתנים שונים.

טענה: אם φ לא בצורה קנונית ניתן להעביר אותה לנוסחה φ' שקולה קלה אשר בצורה קנונית בזמן פולינומי.

הרעיון

נניח ש-φ לא בצורה קנונית. אז קיימת פסוקית C שאינה מקיימת את הדרישה.

$C=(v_1, v_2, v_3)$

1. v_1 ~~קבוע~~ קבועה (F, v_2, v_3)

אז ניתן לבטל את הפסוקית ולצבוע את המשתנים x_2 של v_2 את x_3 של v_3 בצורה מתאימה, כך ש: $f(v_2)=f(\bar{v}_3)$.
~~אם~~ מחליפים את כל המופעים של x_2 ב-x_3.

2. (T, v_2, v_3)

$\Downarrow$

מגדירים את x_2 ו-x_3 כך שהם יהיו F, ומורידים את הפסוקית.

3. רק מהמשתנים זהים (x_1, x_1, x_1) אז אפשר ונוציא מהרקורסיה.

$(x_1, x_1, \bar{x}_1) \Rightarrow x_1 = F$

$(x_1, x_1, x_2) \Rightarrow x_2 = T, x_1 = F$

אלגוריתם $X3SAT(\varphi)$

1. בחר משתנה x_i המופיע בנוסחא.
2. הפעל $b \leftarrow Test(\varphi, x_i, T)$
3. אם $b = T$ החזר T
4. אחרת להמשיך לחפש. $b \leftarrow Test(\varphi, x_i, F)$ והחזר b.

הפרוצדורה $Test(\varphi, x_i, v)$

1. קבע את ערך ההשמה של $f(x_i) = v$.
2. העבר את φ לנוסחא φ' קצרה יותר (במצב קנוני).
   - אם קיבלנו סתירה החזר F.
   - אם φ' ריקה החזר T.
3. הפעל $X3SAT(\varphi')$ והחזר את התוצאה.

חסם על זמן הריצה

זמן הריצה המקסימלי של האלגוריתם על נוסחאות בנות n משתנים - $f(n)$.

$$f(n) \le p(n) + f(n-3) + f(n-2)$$

$$f(n) = O(2^{\alpha n}) \qquad 0 \le \alpha \le 1$$

$$f(n) \le O(p(n) \cdot 2^{\frac{n}{2}})$$

אפשר להוסיף עוד כללים בכללי קבע היוריסטים תקפים יותר.

למשל: אם בשתי פסוקיות שונות, יש לכל היותר משתנה משותף אחד.

משתנה יתומי: x_i יתום אם הוא מופיע בפסוקית אחת בלבד

דרישה נוספת, בכל פסוקית יש לכל היותר משתנה חיובי אחד.

נניח ש-φ יש יותר ממשתנה חיובי אחד.

יש פסוקית $(x_1, \bar{x}_2, u_3)$

$$f(\bar{x}_2) = f(\bar{u}_3)$$

אם יש שתי פסוקיות עם יותר ממשתנה אחד חיובי

(u_1, u_2, u_3)

(u_1', u_2', u_4)

$(x_1, \bar{x}_2, \bar{x}_3)$

$(\bar{x}_1, x_2, x_4)$

דוגמא!

אם בכל פסוקית יש משתנה חיובי, צ"ל הפתרון פולינומי.

קבוצה בלתי תלויה מקסימום בגרפים מישוריים.

קבוצה בלתי תלויה זה קבוצת צמתים שלא קיימות קשתות ביניהם.

בעיית מציאת קבוצה בלתי תלויה מקסימום היא NP קשה אפילו אם נגביל את עצמנו לגרפים מישוריים.

עובדה: G מישורי אם ורק אם אין לו תת-גרף "דומה" ל-K_5 או $K_{3,3}$

מפריד זה קבוצת צמתים בגרף שהתכונה שלה שבזכותה ניתן לחתוך את הגרף לחלקים קטנים יחסית.

משפט המפריד הקטן: לכל גרף מישורי בן n צמתים, קיים מפריד ~~מגודל~~ מגודל שהוא לכל היותר $\sqrt{8n}$. ניתן לחשב את המפריד בצורה יעילה אלגוריתמית. $O(n)$.

אלגוריתם(G) MIS-PL.

1. מצא ל-G מפריד קטן S.
2. לכל קבוצה בלתי תלויה $M_0 \subseteq S$, נבצע

   א. נמחק מ-A_1 ו-A_2 את כל השכנים של צמתי M_0 ומקבלים גרף חדש, A_1' ו-A_2'

   ב. (נפתור) $M_1 \leftarrow MIS\text{-}PL(A_1')$

   $M_2 \leftarrow MIS\text{-}PL(A_2')$

   ג. $M = M_0 \cup M_1 \cup M_2$
3. החזר את ה-M הגדולה ביותר.

טענה: האלגוריתם מחזיר MIS.

הוכחה: תהי M^* המקסימום.

$$M^* = M_0^* \cup M_1^* \cup M_2^*$$

זמן ריצה מקסימלי על כל מופעי בגודל n: $T(n)$

$$T(n) \le c_1 \cdot n + 2^{\sqrt{8n}} \cdot \left(c_2 n + 2 T\left(\frac{2}{3}n\right)\right)$$

נוכיח:

$$T(n) \le 2^{c_3\sqrt{n}}$$

נוכיח באינדוקציה:

$$T(n) \le c_1 n + 2^{\sqrt{8n}} \cdot \left(c_2 n + 2 \cdot 2^{c_3\sqrt{\frac{2}{3}n}}\right)$$

הנחת האינדוקציה

$$\le 2^{c_3\sqrt{n}}$$

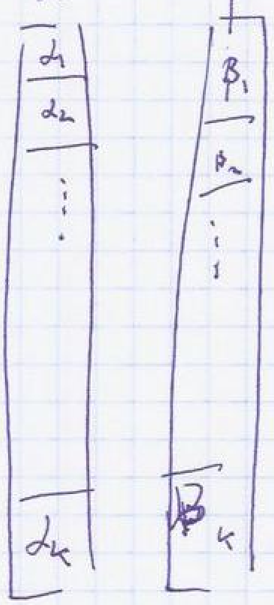
תרמיל 0-1

אלגוריתם עם $T = O(n \cdot 2^{\frac{n}{2}})$. מבוסס פ.ו.ו.

$$A_\alpha = \{a_1 \dots a_{\frac{n}{2}}\}$$

$$A_\beta = \{a_{\frac{n}{2}+1} \dots a_n\}$$

$$K = 2^{\frac{n}{2}}$$



$I = 0 \ , \ J = k$

$\alpha_I + \beta_J = B$?

אם $\alpha_I + \beta_J <$ ← $I+1$

אם $\alpha_I + \beta_J >$ ← $J-1$

סיבוכיות: $T = O(n \cdot 2^{\frac{n}{2}})$

אם בונים בצורה חכמה וממיינים.

<u>נכונות:</u>

לפי השאלה. נניח שיש פתרון (I^*, J^*) והאלגוריתם החזיר "לא"!

נניח שהאלגוריתם בדק את $I = I^*$ ראשון.

$\alpha_I + \beta_J > \alpha_{I^*} + \beta_{J^*} = B$

לכן האלגוריתם יקטין את J

~~בהכרח~~ בהכרח יגיע ל $J = J^*$ ← סתירה.

<u>אלגוריתם נוסף</u>

1. נבנה את Q תת קבוצה $Q \subset A_p$

2. לכל Q, $q = \sum_{a_i \in Q} a_i$

$r \leftarrow B - q$

3. חפש ב-B תת עם סך r.

4. מצאת ← החזר כן,

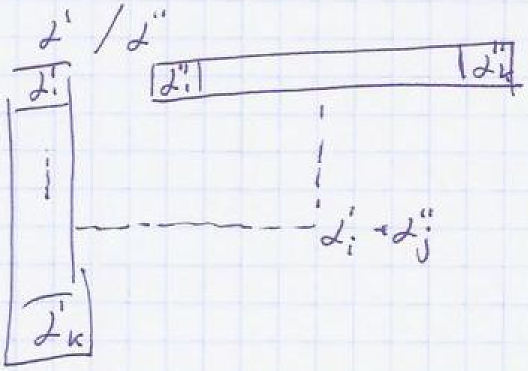
אלגוריתם נוסף

$T = O(n 2^{\frac{n}{2}})$

$S = O(2^{\frac{n}{4}})$

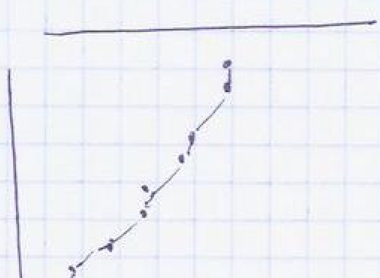
$A\alpha' = \{a_1 \dots a_{\frac{n}{4}}\}$

$A\alpha'' \{a_{\frac{n}{4}+1} \dots a_{\frac{n}{2}}\}$ $\qquad K = 2^{\frac{n}{4}}$



PQ - priority queue

- insertion
- remove min/max



צביעה בגרפים דו-צדדיים

$V = U \cup W$

בעיה פולינומיאלית בגרף דו-צדדי.

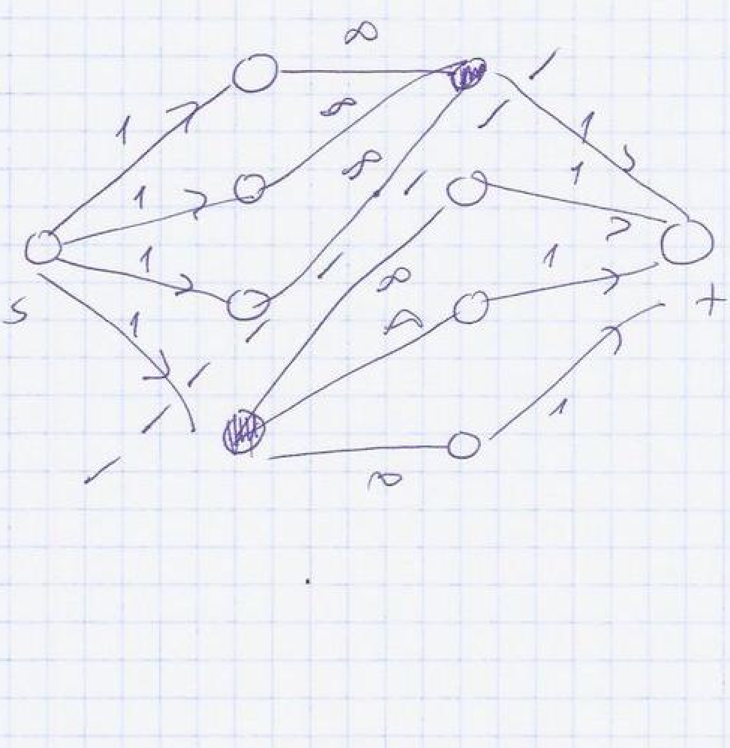
כיסוי בצמתים בגרף דו-צדדי

קבוצת צמתים $Z \subseteq V$ היא כיסוי בצמתים אם לכל קשת $(x,y) \in E$ $x \in Z$ או $y \in Z$.

בעיית כיסוי בצמתים קטן ביותר (מינימום) היא NP-Hard.

מוסיפים צומת מקור וצומת בור ויוצרים רשת זרימה בגרף דו-צדדי. לקשתות החדשות משקל 1 ולשאר משקל ∞.

בוחרים את כל הצמתים של הגרף המקורי שנוגעים בקשתות של ה min cut שמשלים את בעיית הכרומה.



בעיות עם פרמטר Fixed parameter tractability (FPT).

בנוסף לקלט יש עוד פרמטר מספרי אחד שמשפיע על הסיבוכיות. לדוגמה, כיסוי בצמתים.

כיסוי בצמתים - גרסת ההחלטה.

קלט, גרף G ו-k

השאלה, האם יש ב-G כיסוי בצמתים בגודל לא יותר מ-k.

אפשרות: חיפוש שלם. עוברים על כל הקבוצות מגודל k.

$O(n^k \cdot |E|)$.

פרוצדורה רקורסיבית

$P(k, E)$ E - קבוצת הקשתות שלא כוסו.

1. אם $E=\emptyset$ החזר "כן".
2. אם $k=0$ החזר "לא"
3. בחר קשת $e=(x,y)$ מהקבוצה E.
4. $E_x = \{e' \mid$ לא נוגעת ב-x $e'\}$

   $E_y = \{e' \mid$ לא נוגעת ב-y $e'\}$
5. $b_x \leftarrow P(k-1, E_x)$

   $b_y \leftarrow P(k-1, E_y)$
6. מחזירים $b_x \vee b_y$.

בעיית מסלול ארוך

קלט: גרף $G=(V,E)$

פרמטר k

שאלה: האם G מכיל שביל פשוט מאורך $\geq k$?

רעיון לפתרון - טכניקה רנדומית. לצבוע את הצמתים של הגרף באופן רנדומי ב-k צבעים. (לא חייבת להיות צביעה חוקית).

שביל ססגוני - לכל צומת יש צבע אחר

ההסתברות לקבל שביל ססגוני היא בערך $\frac{1}{e^k}$

אם נחזור על הניסוי $e^k \log n$ פעמים, ההסתברות שנכשל בכל הפעמים היא $\frac{1}{n}$.

תהליך החיפוש

מניחים שיש שביל ססגוני.

תכנות דינאמי

באיטרציה i: בונים טבלה $Q_i(v)$, $\forall v$.

$Q_i(v) = \{ C \mid$ C קבוצה של i צבעים שונים, ~~ויש~~ וקיים בגרף G שביל מאורך i בדיוק שמסתיים ב-v ומשתמש בצבעים בדיוק C $\}$

הכרעה: אם $Q_k(v)=\emptyset$ לכל v אז נכשלנו.

אתחול הבניה:

$Q_1(v): \{\{c\}\}$, $c=\{j\}$, j הצבע של v.

נניח שבנינו $Q_{i-1}(v)$, $\forall v$

נבנה $Q_i(v)$ לכל v.

הבנייה: לכל צומת v, לכל זוג צבעים ($C \in Q_{i-1}(v)$)

לכל שכן w של v עם צבע j:

אם $j \notin C$ אז הוסף ל-$Q_i(w)$ את $C \cup \{j\}$.

סה"כ סיבוכיות ($O(2^k n^2 e^k \log n) \approx O(e^{2k} n^2)$

בעיית תת עץ בינארי מלא.

קלט: G, k

שאלה: האם יש ב-G עץ בינארי מלא מעומק k.

$|V(T)| = K = 2^{k+1}-1$

בעיית הצביעה

ב-K צבעים. לצבוע את כל קודקודי הגרף

ב-K צבעים, ומחפשים עץ ססגוני.

בעיית החיפוש

מבטיחה שעם ההגדלה בצבעים, אם העץ ב-K צבעים

שונים, נמצא אותו.

ברקורסיה: $Q_i(v) = \{C \mid |C| = 2^{i+1}-1,\ C \subseteq \{1,\dots,k\}$:

יש ב-G עץ T מעומק i שמושרש ב-v, וצבועים בצבעים

בצבעים של C.

בסיום! אם יש V ש-$Q_k(v) \neq \emptyset$ אז מצאנו עץ כנדרש

.

<u>תהליך הבנייה:</u>

בהנחה שבנינו את $Q_{i-1}(v)$ $\forall v$, נבנה את $Q_i(v)$ באופן הבא:

לכל $C_1 \in Q_{i-1}(v_1)$

ולכל $C_2 \in Q_{i-1}(v_2)$

ולכל שכן משותף, w, של v_1 ו-v_2, שצבעו j, אם

$C_1 \cap C_2 = \phi$

ו- $j \notin C_1 \cup C_2$

אז הוסף את $C_1 \cup C_2 \cup \{j\}$ ל- $Q_i(w)$.

<u>תרגיל:</u>

Cheap 3SAT

קלט: נוסחה φ ב- 3CNF

רוצים כמה שיותר קטנים. מחיר השמת אמת: מס' ה-x-ים שניתנו ערך 1

האם יש ל-φ השמת אמת שמחירה לא יותר מ-k.

קליק בגודל k

קלט: גרף G

פרמטר k

שאלה: האם יש ב-G קליק בגודל k.

משפט: אם קיים אלג' פולינומי למציאת קליק מגודל $k = \log n$, אז קיים גם אלג' למציאת קליק מגודל k בסיבוכיות $2^{\sqrt{k}\log n}$.

הוכחה:

נניח שקיים אלג' A ~~[illegible]~~ לפי ההנחה. נבנה אלג' B שפותר בעיית קליק של k.

אלג' B $\quad B[G,k]$

1. נבנה גרף $\tilde{G}(\tilde{V},\tilde{E})$

$$\tilde{V} = \{ w \subset V \mid |w| = \sqrt{k} \}$$

$$\tilde{E} = \{ (w,w') \mid w \cap w' = \phi,\ w \cup w' \text{ קליק ב-G} \}$$

$$\tilde{n} = |\tilde{V}| = \binom{n}{\sqrt{k}} < 2^{\sqrt{k}\log n}$$

2. הפעל את ~~[illegible]~~ $A(\tilde{G},\sqrt{k})$ והחזר את התשובה.

$\sqrt{k} \le \log \tilde{n}$?

$\log \tilde{n} \geq c\sqrt{k}\log n \geq \sqrt{k}$

הבחנה: קיים קליק מגודל $\sqrt{k}$ ב-$\tilde{G}$ אם ורק אם קיים קליק בגודל k ב-G המקורי.

הערות:

כנ"ל ל-mis.

המשפט לא עובד לגרפים מושרים.

אם $k \ge \frac{n}{4}$ אז אפשר לענות מיידית על MIS.

אלגוריתמי קירוב

בעיית אופטימיזציה Π

קלט I

ערך האופטימום $f^*(I)$

אלגוריתם A משתמש $f_A(I)$

(1) $$\rho = \frac{|f_A(I) - f^*(I)|}{f^*(I)}$$

(2) לבעיי מקסימיזציה: $\frac{f^*(I)}{f_A(I)} \le \rho$

לאלגוריתם מינימיזציה: $\frac{f_A(I)}{f^*(I)} \le \rho$

TSP דקס

משפט: אם קיים עד TSP (ללא אי שוויון המשולש) אלגוריתם קירוב בפקטור ρ (עד 2^{n^c}) אז $P = NP$.

הוכחה: בשלילה

נניח שקיים אלגוריתם קירוב A. נשתמש ב-A לבניית אלגוריתם פולינומי לפתרון בעיית המילטוני. כיון יש סתירה.

אלגוריתם B

1. בהינתן גרף G, נבנה קלט $\{d_{ij}\}$ ל-TSP, $d_{ij} = \begin{cases} 1 & (i,j) \in E(G) \\ \rho \cdot n + 2 & \text{אחרת} \end{cases}$.
2. נפעיל את A על הקלט התקבל
3. אם A החזיר מסלול TSP מאורך n אז התשובה "כן" אחרת לא

<u>טענה:</u> B מחזיר כן אם ורק אם יש ב-G מעגל המילטוני.

<u>הוכחה:</u> !! מיידי

נניח שקיים מעגל המילטוני אז $f^*(I) = n$.

נניח בשלילה שהאלגוריתם B ענה "לא", זה אומר שהאלג' A נותן ערך שהוא לא n. זה אומר ש- $f_A(I) \geq 1(\rho n+1) + (n-1)$

זה אומר

$$\frac{f_A(I)-f^*(I)}{f^*(I)} = \frac{[(n-1)+(\rho n+1)]-n}{n} = \frac{\rho n}{n} > \rho$$

<u>סתירה להנחה.</u>

<u>קירוב ל- TSP</u>

השכן הקרוב ביותר. אלגוריתם חמדן. $\rho = \log n$

שכן קרוב גלובלי - מתחילים מעגל של הערים שבקרבתי בהן. כל קודקוד מכניסים למעגל בין שתי הנקודות הקרובות ביותר (מתחשבים בכל שלב לכל הקודקודים).

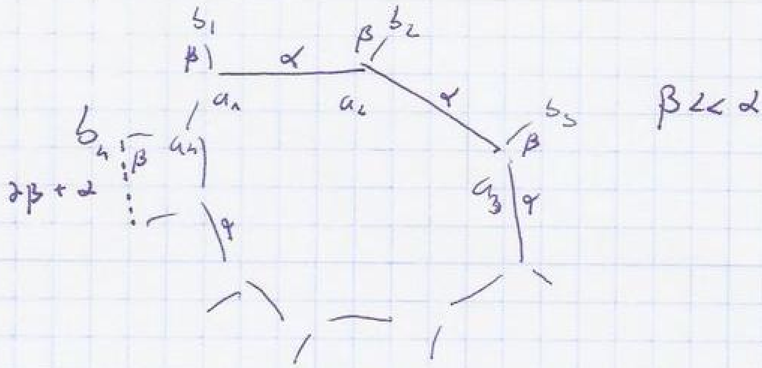
<u>אלגוריתם העץ</u>

1. בונים עץ פורש מינימלי. MST.
2. מטיילים טיול אוילרי (DFS) על העץ
3. כאשר חוזרים לנקודה שהיינו בה ואחר עוברים לנקודה חדשה, נחבר את הנקודה הישנה לנקודה החדשה. (קיצורי דרך)

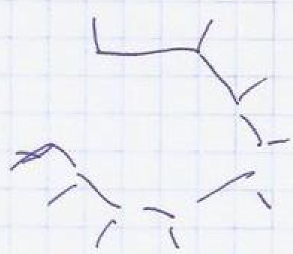
את המעגל החדש אנו מחזירים כפיתרון.

משקל עץ פורש $\cdot 2 \leq$ פתרון אופטימלי

דוגמא:



$\Downarrow$ ϕ_γ



$\Downarrow$ קיצורי דרך

אלגוריתם כריסטופידס

בונים עץ פורש, ומוסיפים קשת לכל זוג קשתות שאין להן דרגה זוגית.
את הקשתות שמוסיפים עושים לפי המינימום.
על הגרף החדש נמצא מסלול אוילרי, וע"י קיצורי דרך נעבור למסלול אופטימלי.

MIS - קבוצה בלתי תלויה גדולה ביותר

קלט: G מגודל n צמתים

שאלה: מצא קבוצה בלתי תלויה גדולה ביותר.

$1 \le |M^*| \le n$

ידוע: 1. אלגוריתם קירוב עם יחס $\frac{n(\log\log n)^2}{\log^3 n}$

2. אין קירוב בפקטור $n^{1-\varepsilon}$

<u>אלגוריתם החמדן A</u>

$S \leftarrow \emptyset$

כל עוד G לא ריקה בחר צומת v מדרגה מינימלית

- הוסף אותו ל-S

- מחק מ-G את v ואת שכניו.

אלגוריתם זה לא טוב!

<u>למה 1:</u> אם ב-G הדרגה המקסימלית Δ, אז האלגוריתם החמדן יחזיר S בגודל לפחות $\frac{n}{\Delta+1}$

<u>נימוק:</u>

כל איטרציה ימחקו לכל היותר $1+\Delta$ צמתים. יהיו לפחות $\frac{n}{\Delta+1}$ איטרציות. ובכל איטרציה מוסיפים קודקוד ל-S.

<u>למה 2:</u>

אם G ניתן לצביעה ב-k צבעים אז האלגוריתם החמדן יחזיר קבוצה בלתי תלויה בגודל לפחות $\frac{1}{2}\log_k n$.

אלגוריתם קירוב עם יחס $\frac{n}{\log n}$

1. חלק את V ל- $V_1 \cup V_2 \cup \dots \cup V_{\frac{n}{\log n}}$

הקבוצות זרות כש. $|V_i| = \log n$.

2. בכל $G_i = G(V_i)$, מצא קבוצה בלתי תלויה מקסימלית בחיפוש מלא, M_i.

3. החזר את הגדולה ביניהן.

משיקולי מנייה, יש i ש- $|M^* \cap V_i| \ge \frac{|M^*|}{n/\log n}$

מאותו שיקול יש את $|M_i| \ge |M^* \cap V_i| \ge \frac{|M^*|}{n/\log n}$

$$\rho \le \frac{|M^*|}{|M_i|} \le \frac{n}{\log n}$$

אלגוריתם בקירוב $\frac{n(\log\log n)^2}{\log^2 n}$

$|M^*| = \frac{n}{k}$

א. $k \ge \log^2 n$

$|M^*| \le \frac{n}{\log^2 n}$

האלג' A_1:

החזר צומת כלשהו.

I. $\frac{\log n}{c \cdot \log\log n} \le k \le \log^2 n$

אלגוריתם A_2

1. נחלק את V ל- $\frac{n}{kt}$ קבוצות. $t = \frac{\log n}{\log\log n}$

כולן בגודל kt כל אחת.

2. בכל קבוצה $G(V_i)$ ~~נחשב~~ נחפש קבוצה בלתי תלויה בגודל t.

3. נחפש אותה כך:

$$\binom{kt}{t} \le (kt)^{t} \le (\log^3 n)^{\frac{\log n}{\log\log n}}$$

$$= \left(\tfrac{3}{2}\log\log n\right)^{\frac{\log n}{\log\log n}} = 2^{3\log n} = n^3$$

$$|M^*| = \frac{n}{k}$$

משובך היונים, קיים i, $m_i \ge |m^* \cap V_i| \ge \frac{n/k}{n/kt} = t$

יחס הקירוב:

$$|m^*| \le \frac{n}{k} < \frac{n \cdot c\log\log n}{\log n}$$

$$|m_{A_2}| = t = \frac{\log n}{\log\log n}$$

$$\frac{|m^*|}{|m_{A_2}|} \le \frac{n \cdot c \cdot (\log\log n)^2}{\log^2 n}$$

גרפים ללא משולשים:

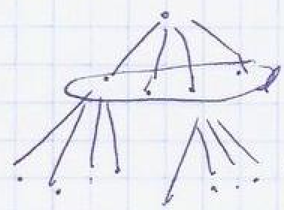
אלגוריתם B לקירוב ביחס $\sqrt{n}$.

1. אם $\Delta(v) \le \sqrt{n}$ אז הפעל אלגוריתם חמדן.

2. <u>$\sqrt{n} < \Delta(v)$:</u>

- בחר צומת עם דרגה מקסימלית Δ.
- החזר את שכניו. $\Gamma(G)$.

דוגמה הדוקה:



$\ell - 1$ $\quad n \sim \ell^2$

<u>גרפים ללא קליק-k ומעלה</u> $\quad n^{1-\frac{1}{k-1}} \quad B(k,G)$.

<u>k=2</u>: החזר את כל הצמתים

<u>k=3</u>: הפעל את האלגוריתם הקודם.

<u>k>3</u>:

1. אם $\Delta < n^{1-\frac{1}{k-1}}$ הפעל אלגוריתם חמדן.

2. - בחר v עם דרגה Δ.
   - הפעל $B(k-1, \Gamma(v))$
   - החזר תוצאות.

<u>טענה:</u>

על G ללא קליק k, $B(k,G)$ יחזיר קבוצה Z מגודל
$Z < c \cdot n^{\frac{1}{k-1}}$

הוכחה באינדוקציה.

ד. k קבוע $|M^*| = \frac{n}{k}$

$n = 2k$, $\varepsilon = \frac{1}{k}$

$|M^*| = (\frac{1}{k} + \varepsilon) n$

האלגוריתם שלנו:

1. הפעל אלגוריתם $Free\text{-}k(G)$ למציאת k-קליקים. מתקבל G'
2. הפעל $B(k, G')$.

הפרוצדורה $Free\text{-}k(G)$

כל עוד יש ב-G קליק k
- בחר קליק k
- מחק מ-G את צמתיו ואת הקשתות שנוגעות בהן.

טענה:

~~הפעלת~~ כשהאלגוריתם free מסיים עדיין נשאר זיווג על G' מגודל לפחות εn.

נימוק

נניח שהיו ~~באלגוריתם~~ free x איטרציות. סה"כ נמחקו kx צמתים.

$kx \le n$

$x \le \frac{n}{k}$

בכל איטרציה נמחק לכל היותר צומת אחד מ-m^*.

לכן נמחקו מ-m^* רק x צמתים סה"כ שזה לא יותר מ-$\frac{n}{k}$. ולכן נשארו εn צמתים

$|m^*| = \frac{n}{k} + \varepsilon n$

סה״כ הוכ׳ קיום קבוצה בלתי תלויה S מגודל

$$|S| \geq (\varepsilon n)^{\frac{1}{k-1}}$$

<u>הערה למקרה ד.</u>

$$|S| \geq \underbrace{\left(\frac{1}{k}\right)^{\left(\frac{1}{k-1}\right)}}_{1 < \| < 2} \cdot \cancel{\varepsilon} n^{\left(\frac{1}{2k-1}\right)}$$

בעיית הצביעה

משפט: אם קיים אלגוריתם קירוב ל- MIS עם יחס קירוב $\rho(n)$ (פונקציה לא יורדת), אז קיים אלגוריתם קירוב לצביעה עם יחס $\rho(n)\log n$.

$\cancel{\rho(n) = \frac{n}{\log n}}$

משפט: אם קיים אלגוריתם קירוב ל- MIS עם יחס קירוב $\rho(n) = \frac{n}{\log^\alpha n}$, אז קיים אלגוריתם קירוב לצביעה עם יחס קירוב $3\rho(n)$.

הוכחה: בהינתן אלג' A לקירוב MIS ביחס $\rho(n) = \frac{n}{\log^\alpha n}$, נבנה אלגוריתם B לקירוב צביעה.

אלג' B

1. $i = 1$
2. כל עוד $|V| > \frac{n}{\log^\alpha n}$ בצע:
   - א. הפעל אלגוריתם A ומצא קבוצה בלתי תלויה I ב-G
   - ב. צבע את צמתי I בצבע i;
   - ג. נתק מהגרף G את I והקשתות שלו
   - 3. $i = i + 1$

m איטרציות ← m צבעים

3. צבע כ"א מהצמתים הנותרים בצבע חדש.

$m' < \frac{n}{\log^\alpha n}$ צבעים

סה"כ $m + m'$ צבעים.

V_i: קבוצת הצמתים בתחילת איטרציה i:

$G_i = G(V_i)$ הגרף שנשאר

x_i - מספר הצבעים הנדרש לצביעת G_i

n_i - מספר הצמתים $n_i = |V_i|$

I_i - הקבוצה שהאלגוריתם מצא

I_i^* - הקבוצה הבלתי תלויה הגדולה ביותר בגרף G_i.

<u>טענת עזר I</u>: $|I_i^*| \geq \frac{n_i}{x_i}$

<u>נימוק:</u> $x_i \leq x \Rightarrow$ קיימת קבוצת צמתים S (קבוצת צבע) כך ש

$$\frac{n_i}{x} \leq \frac{n_i}{x_i} \leq |S| \leq |I_i^*| .$$

$$* \qquad \frac{|I_i^*|}{I_i} \leq \rho(n_i)$$

** עבור $n_i \geq \frac{n}{\log^\alpha n}$, ל-n גדול מספיק, מתקיים

$\log^\alpha n_i \geq \frac{1}{2}\log^\alpha n$

<u>טענת עזר 2:</u> $|I_i| \geq \frac{\log^\alpha n}{2x}$

<u>נימוק:</u> $|I_i|^* \geq \frac{|I_i^*|}{\rho(n_i)} \geq \frac{n_i/x}{n_i/\frac{1}{2}\log^\alpha n}$

איטרציה הראשונה: $n_1 = n$

איטרציה m-1: $\frac{\log^\alpha n}{2x} < \frac{n}{\log^\alpha n} \leq n_{m-1} \leq$

$$\leq n - (m-1)\left(\frac{\log^\alpha n}{2x}\right)$$

$$m \cdot \frac{\log^\alpha n}{2x} \leq n$$

$$m \leq \frac{2n}{\log^\alpha n} x$$

$$x_{alg} = m + m' \leq \frac{3n}{\log^\alpha n} x$$

אלגוריתם DEG.

נסמן $\Delta(G)$ - דרגה מקסימלית ב-G.

$C=\{1,\dots,\Delta(G)+1\}$ קבוצת צבעים.

לכל צומת בודקים את הצבעים של השכנים שלו, וצובעים בצבע פנוי.

אם G גרף 3 צביע, אז לכל צומת v, קבוצת השכנים שלו $(N(v))$ משרה גרף דו צדדי.

אלגוריתם לצביעת גרף 3 צביע: משתמש ב- $3\sqrt{n}$ צבעים.

א. $i \leftarrow 1$

ב. כל עוד $\Delta(G) \geq \sqrt{n}$ בצע:

1. בחר צומת v עם דרגה מקסימלית
2. צבע אותו ב-$i+2$
3. צבע את השכנים בצבעים i, $i+1$
4. מחק את v ואת שכניו
5. $i \leftarrow i+2$

ג. צבע את הגרף הנותר ע"י אלגוריתם DEG

הוכחות

1. צעד ג' משתמש ב-$\sqrt{n}$ צבעים.
2. צעד ב' משתמש בשני צבעים בכל איטרציה - סה"כ $2m$.
3. בכל איטרציה מורידים לפחות $\sqrt{n}$ צמתים, לכן מס' ~~מספר השלבים~~ האיטרציות $m \leq \frac{n}{\sqrt{n}} \leq \sqrt{n}$

ע"כ סה"כ השתמשנו ב-$3\sqrt{n}$ צבעים.

שיפור ל- $\sqrt{\frac{n}{\log n}}$ צבעים.

1. כל עוד $\Delta_G \ge \sqrt{\frac{n}{\log n}}$ בצע:

א. $G' \leftarrow G$

ב. $S \leftarrow \emptyset$

ג. כל עוד $G' \neq \emptyset$ ו- $|S| \le \log n$

- בחר v בדרגה מקסימלית
- $S \leftarrow S \cup \{v\}$
- מחק $\Gamma'(v) \cup \{v\}$ מ-G'.

ד. עבור על S את הקבוצה $Z \subseteq W$

נבחר את W הגדולה ביותר ש: $(\Gamma(W))$ 3 צביע.

ה. צבע את צמתי W בצבע אחד, $\Gamma(W)$ בשני צבעים נוספים

ו. הסר מ-G את $\Gamma(W) \cup W$

2. צבע את הגרף הנותר ע"י DEG.

נשים לב ההבדל: מס' איטרציות

$$m \le 3\sqrt{\frac{n}{\log n}}$$

<u>הטענה:</u>

כל איטרציה מוחקת לפחות $\frac{1}{3}\sqrt{n\log n}$ צמתים.

<u>נימוק:</u>

נשים לב שבכל איטרציה נמצא $|W| \ge \frac{\log n}{3}$

כי $G(S \cup \Gamma(S))$ 3-צביע, בפרט $G(S)$ 3-צביע.

W בצבע 1, אז $\Gamma(W)$ צבועים בצבע $\neq 1$.

$$|\Gamma(W)| \ge |W| \cdot \sqrt{\frac{n}{\log n}} \ge \frac{\log n}{3}\sqrt{\frac{n}{\log n}} =$$

$$= \frac{1}{3}\sqrt{n\log n}$$

תרגיל:

תנו אלגוריתם פולינומי לבעיה הבאה:

קלט: גרף G עם n צמתים, 3-צביע. קבוצת צמתים W, שגודלה $|W| \le \log n$

$\log n$, שנתון שהינה בלתי תלויה ושלמה.

פלט: לצבוע את G ב- $O(\log\log n)$ צבעים.

אלגוריתמים חמדניים

מסגרת:

עולם $I = \{1, \dots, m\}$

תתי-קבוצות $P_1, P_2 \dots P_n \subseteq I$

$J = \{1, \dots, n\}$

כיסוי קבוצות:

אוסף (מינימלי) של תתי-קבוצות $S \subset I$ כך שהאיחוד שלהם שווה לעולם $\bigcup_{j \in S} P_j = I$.

הבעיה: כיסוי קבוצות קטן/גדול ביותר

הבעיה הפוכה: $S \subset I$ כך ש: $P_j \cap S \ne \emptyset$, $\forall j$

אלגוריתם חמדן

1. $S \leftarrow \emptyset$
2. כל עוד $I \ne \emptyset$ בצע:

א. בחר k כך ש: $|P_k|$ מקסימלי.

ב. הוסף את k לפתרון

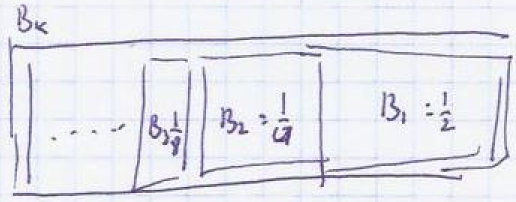
ג. מחק את איברי P_k מ- I. ומכל קבוצה P_j שנשארה

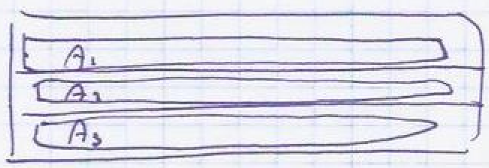
דוגמה רעה הדוקה:

I $\qquad m = 3 \cdot 2^k$

a_{11}	a_{12}		a_{12^k}
a_{21}	a_{22}		a_{22^k}
a_{31}	a_{32}		a_{32^k}

הקבוצות $m = k + 3$





greedy $c = k \sim \log m$

$c^* = 3 \quad (A_1 \cup A_2 \cup A_3)$

אלגוריתם חמדן - מסגרת

עולם $U = \{u_1, u_2 \dots u_n\}$

מחיר $c_i = c(u_i)$

פתרון: $S \subseteq U$

אילוצים

פתרון פיזיבילי - פתרון שמקיים את כל האילוצים.

מטרה - למצוא פתרון פיזיבילי זול ביותר.

מדד למרחק מפיזיביליות: פונקציית קנס $P(S)$

S פיזיבילי $\longrightarrow P(S) = 0$

נניח ש- $S=U$ פיזיבילי. $P(U) = 0$.

דרישה (A_1) מונוטוניות:

לכל S_1, S_2 אם $S_1 \subseteq S_2$ אז $P(S_1) \geq P(S_2)$

השיפור: $\Delta(S, u_i) = P(S) - P(S \cup \{u_i\})$.

האלגוריתם החמדן:

1. $S \leftarrow \emptyset$
2. כל עוד $P(S) > 0$ בצע
   - בחר u_i עם ערך מקסימלי של $\frac{\Delta(S, u_i)}{c_i}$
   - $S \leftarrow S \cup \{u_i\}$
3.

דרישה (A_3): חישוב P פולינומי.

דרישה (A_2):

S^*

$\bar{S} = S^* \setminus S$

$$\sum_{u \in \bar{S}} \Delta(S, u) \geq P(S)$$

ניתוח טיב הקרוב

$c^* = c(S^*)$

אופטימום

לכל קבוצה S, $U(S)$ = האלמנט שהאלגוריתם החמדן היה מוסיף ל-S לו פיזיבילי ~~מחזור~~ מחירו $c(U(S))$

למה:

$$\frac{\Delta(S,U(S))}{c(U(S))} \ge \frac{P(S)}{c^*}$$

הוכחה:

נניח $\bar{S} = S^* \setminus S = \{U_{i_1}, U_{i_2} \dots U_{i_\ell}\}$

$$\ge \frac{\sum_{j=1}^{\ell} \Delta(S,U_{i_j})}{\sum_{j=1}^{\ell} c_{i_j}} = \frac{\sum_{j=1}^{\ell} \Delta(S,U_{i_j})}{c(\bar{S})} \overset{(A_2)}{\ge} \frac{P(S)}{c^*}$$

$$\frac{\Delta(S,U(S))}{c(U(S))} \ge$$

כי $\frac{\Delta(S,U_{i_j})}{c_{i_j}} \le \frac{\Delta(S,U(S))}{c(U(S))}$ לכל j

למה ב:

C_g - חמדני $\quad C_g \le H(P(\emptyset)) \cdot c^*$

כאשר $H(k) = 1 + \frac{1}{2} + \frac{1}{3} + \dots + \frac{1}{k} \approx \ln k$

הוכחה: בריצת האלגוריתם

U_i - האלמנט שנבחר באיטרציה i.

c_i - המחיר של c_i

p_i - הקנס בתחילת איטרציה i

Δ_i - השיפור באיטרציה i.

$P_1 = P(\phi)$

$P_{i+1} = P_i - \Delta_i$

$P_{m+1} = 0$ נניח שיש m איטרציות

$$C_g = \sum_{i=1}^{m} c_i \le \sum_{i=1}^{m} \frac{c^* \cdot \Delta_i}{P_i} =$$

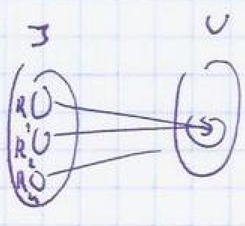
$$= c^* \sum_{i=1}^{m} \frac{\Delta_i}{P_i} \le c^* \sum_{i=1}^{m} \left(\frac{1}{P_i} + \frac{1}{P_i - 1} + \frac{1}{P_i - 2} + \dots + \frac{1}{P_i - \Delta_i + 1} \right) =$$

$$= c^* \sum_{i=1}^{m} \left(\frac{1}{P_i} + \frac{1}{P_i - 1} + \dots + \frac{1}{P_{i+1} + 1} \right) = c^* \sum_{j=P_1}^{P_{m+1}+1} \frac{1}{j} = c^* \cdot H(P_1)$$

דרישה (A_2'): לכל S_1, S_2, U עם $S_1 \subseteq S_2$

$S_1 \subseteq S_2 \Rightarrow \Delta(S_1, U) \ge \Delta(S_2, U)$

$(A_2') \Rightarrow (A_2)$ (תרגיל: להוכיח)



דוגמא: קבוצה פוגעת

U - אלמנטים

J - קבוצות.

$$P(S) = |\{ j \in J : S \cap R_j = \phi \}|$$

תרגיל:

קבוצה k פוגעת - S כך ש $\forall i \ |S \cap R_i| \ge k$

קבוצות קדימה מינימליות: קלט: $I, R_1 \dots R_n$ $(1..m)$

דרישה: אוסף מינימלי קבוצות מ-J

$J = \{x \subseteq \{1..m\}\}$ כך שלכל $i, j \in I$ קיימת ב-X קבוצה R_k המכילה בדיוק אחד מבין i ו-j.

<u>הבעיה הדואלית:</u>

קבוצה פוגעת דואלית:

<u>קלט:</u> I , $R=\{R_1, R_2 \dots R_n\}$, p

<u>נדרש:</u> $S \subset I$ מגודל $p = |S|$ שפוגעת בכמה שיותר קבוצות ב- R.

נראה יחס קירוב לאלגוריתם החמדן (p איטרציות)

האופטימום בוחר S^* $|S^*| = p$

ומשיג x פגיעות

(פוגע ב- $R^* \subset R$ $x = |R^*|$)

x_g - x החמדן , $x_g \geq \frac{x}{2}$.

<u>ניתוח:</u>

החמדן בוחר i_1 ומרוויח g_1 .. i_p ומרוויח g_p

$$x_g = \sum_{i=1}^{p} g_i$$

$$S^* = \{v_1^* \dots v_p^*\}$$

$$g_1 \geq x/p$$

כמה נשארו ב- R^* שלא נפגעו ? $x_2 \geq x - g_1$

$$g_2 \geq \frac{x - g_1}{p}$$

$$g_3 \geq \frac{x - g_1 - g_2}{p}$$

$$g_p \geq \frac{x - g_1 - g_2 - \dots - g_{p-1}}{p}$$

$p g_1 \geq x$

$p g_2 + g_1 \geq x$

$p g_3 + g_2 + g_1 \geq x$

$\vdots$

$p g_p + g_{p-1} \dots + g_1 \geq x$

$p g_p + \dots + (2p-2) g_2 + (2p-1) g_1 > p x$

$\overset{\text{(ii)}}{\leq}$

$(2p-1)(g_1 + g_2 + \dots g_p) \geq p x$

$x_g \geq \frac{p}{2p-1} x \geq \frac{x}{2}$

עיגול רנדומי

כיסוי קבוצות:

ILP WSC $x_1 \dots x_n$

$$x_i = \begin{cases} 1 & \text{נבחרה} \\ 0 & \text{לא נבחרה} \end{cases}$$

$$\text{minimize} \sum_j c_j x_j$$

$$\text{s.t.} \quad A\bar{x} \geq 1$$

$$a_{ij} = \begin{cases} 1, & i \in S_j \\ 0 & \text{אחרת} \end{cases}$$

$$x_i \in \{0, 1\}.$$

אלג' עיגול A

1. פתור את התוכנית הלינארית המתאימה עם $0 \leq x_i \leq 1$, פתרון $\hat{x}$
2. נבצע השמה רנדומית לכל משתנה $x_i^A \xleftarrow{R} \begin{cases} 1 \\ 0 \end{cases}$ בהסתברות של $\hat{x}_i$.

ניתוח:

$$\mathbb{E}(f(\bar{x}^A)) = \mathbb{E}\left(\sum_i c_i x_i^A\right) = \sum_i c_i \cdot \mathbb{E} x_i^A =$$

$$= \sum_i c_i \hat{x}_i = f(\hat{x}) \leq f(x^*)$$

חוקיות:

$$S_i = \{j \mid j \in S_j\}$$

$$\sum_{i \in J_i} x_i \geq 1 \qquad |J_i| = k$$

$$P[i \text{ לא מכוסה}] = \prod_{j \in S_i} (1 - \hat{x}_j) \leq \left(1 - \frac{1}{k}\right)^k = \frac{1}{e}$$

אלגוריתם עיגול B

1. נריץ את אלג' A במשך $2\ln n$ פעמים.
2. ניקח לפתרון את כל הקבוצות שנבחרו בכל אחת מהאיטרציות.

$$P[\text{אילוץ } i \text{ לא מתקיים}] \le \left(\frac{1}{e}\right)^{2\ln n} = \frac{1}{n^2}$$

$$P[\text{איזשהו } i \text{ לא יתקיים}] \le \frac{1}{n}$$

$$f(\bar{x}^B) \le 2\ln n \; f(\bar{x}^A) \le 2\ln n \cdot f(\bar{x}^*)$$

אלגוריתם C

כמו A, אבל $x_i^c \xleftarrow[R]{} \begin{cases} 1, & \hat{x}_i \cdot 2\ln n \text{ בהסתברות} \\ 0, & \text{אחרת} \end{cases}$

$$P(\text{אילוץ } i \text{ לא מתקיים}) \le \left(1 - \frac{2\ln n}{k}\right)^k \le \left(\frac{1}{e}\right)^{2\ln n} = \frac{1}{n^2}$$

סכמת קירוב פולינומיאלית PTAS.

A_ε סכמת קירוב פולינומית, אם לכל קלט I, לכל פרמטר ε מתקיים

$$\frac{C_\varepsilon^A(I) - C^*(I)}{C^*(I)} \le \varepsilon$$

וזמן הריצה של A_ε חסום ע"י פולינום ב-n $O(n^{(\frac{1}{\varepsilon})^2})$.

סכמת קירוב פולינומית מלאה FPTAS

אם זמן הריצה פולינומי הן ב-n והן ב-$\frac{1}{\varepsilon}$.

לדוגמה $n^2\left(\frac{1}{\varepsilon}\right)^3$.

בעיית התרמיל

קלט: n פריטים (p_i, a_i) [illegible]

רווח, חסם נפח b

דרישה: $S \subseteq \{1,2 \ldots n\}$

כך ש: $P_S = \sum_{i \in S} p_i$ מקסימום.

ו- $A_S = \sum_{i \in S} a_i \; \cancel{[illegible]} \le b = 1$

אלגוריתם קרוב עם פקטור $\frac{1}{2}$ A.

1. ממיינים את הפריטים כך ש: $\frac{p_1}{a_1} \ge \frac{p_2}{a_2} \ldots$

2. קח את $1, 2 \ldots j$ כך ש:

$$\sum_{i=1}^{j} a_i \le b$$

$$\sum_{i=1}^{j+1} > b$$

הרווח $P_0 = \sum_{i=1}^{j} p_i$

יהי i_{max} האינדקס ש- $p_{i_{max}}$

טענה: $P^A \ge \frac{1}{2} P^*$

גירסה מרובה, אפשר לקחת חלק של פריט.

$$P_0 \le \hat{P} < P_0 + p_{j+1}$$

$$P^* \le \hat{P} < P_0 + p_{j+1} \le P_0 + p_{i_{max}} \le 2P^A$$

אלגוריתם תיכנות דינאמי:

פתרון: (P, A, S)

$P_s \qquad A_s \qquad S \subseteq \{1 \dots n\}$

בהתחלה $\{0, 0, \emptyset\}$ ~~$0, \emptyset, \emptyset$~~

$\{(0,0,\emptyset), (P_1, A_1, \{1\})\}$.

באיטרציה i בודקים את האפשרות להוסיף את פריט i.

בתחילת האיטרציה L: {פתרונות עם תת-קבוצה של $\{1 \dots i-1\}$}

בזמן איטרציה i:

אם $(P, A, S) \in L$ מייצרים פתרון חדש

$(P+p_i, A+a_i, S \cup \{i\})$

אם $\leq b$ נשמור את הפתרון.

בדיקת שליטה (נשלטות).

הפתרון (P, A, S) שולט על (P', A', S') אם

$P \geq P'$ ו- $A \leq A'$.

סיבוכיות: $O(|L| \cdot n^2)$

P, ~~A~~

P_1, A_1, S_1

$\vee \quad \vee \quad \vee$

P_2, A_2, S_2

$\vee \quad \vee \quad \vee$

P_3, A_3, S_3

$|L| \le p^*$

Scaling

מחירים חדשים: $p_1 \rightarrow q_1 = \lfloor \frac{p_i}{k} \rfloor$

טענה 1: לכל פתרון S, $0 \le P_S - k \cdot Q_S \le k \cdot |S|$

$Q_S = \sum_{i \in S} Q_i$

נימוק:

$q_i \le \frac{p_i}{k} < q_i + 1$

$kq_i \le p_i \le kq_i + k$

$0 \le p_i - kq_i < k$

$0 \le \sum_{i \in S} (\quad) < k \cdot |S|$

טענה 2:

אם $k < \frac{\varepsilon p^*}{|S^*|}$

אז האלגוריתם יתן קירוב ε.

אלגוריתם קירוב

1. קבע $k = \frac{\varepsilon p^*}{|S^*|}$
2. חשב מחירים $q_i = \lfloor \frac{p_i}{k} \rfloor$
3. הפעל את התוכנית הדינאמית על ה-q_i-ים
5. החזר פתרון המתאים.

תיקון לבחירת x

נפעיל תיקון 2 ונמצא $\frac{p^*}{2} \le \tilde{p} \le p^*$

$$k = \frac{\varepsilon \tilde{p}}{n} < \frac{\varepsilon p^*}{|S^*|}$$

$\Downarrow$

נשים קרוב ε H

זה טוב וכו'..

$$|L| \le Q^* \le \frac{P^*}{k}$$

$$= \frac{p^*}{\varepsilon \tilde{p}/n} = \frac{p^*}{\tilde{p}} \cdot \frac{n}{\varepsilon} \le \frac{2n}{\varepsilon}$$

הסיבוכיות היא $O(|L| \cdot n^2) = O\left(n^3 \cdot \frac{1}{\varepsilon}\right)$