אלגוריתמים מתקדם – תרגול 13/11/2005

הוכן ע"י: קושרובסקי יורי

בעיית GAP (מינימיזציה):

בהינתן בעיה L, בעיית ה GAP המקבילה היא להבחין בין הקלטים:

- YES: הערך האופטימלי ≤ a
- NO: הערך האופטימלי ≥ b

הערה:אין שום דרישה לגבי קלטים בין a ו-b .

טענה 1: אם קיים קירוב של הבעיה המקורית $\frac{b}{a}$, אזי קיים פתרון לבעיית GAP.

הוכחה: נניח כי נתון YES instance כלומר, הערך האופטימלי של הפתרון ≤ a.

כעת, נריץ את אלגוריתם הקירוב ונקבל תשובה עם ערך ≤ $b=\frac{b}{a}a$. אזי, ניתן לקבוע בוודאות שלא קיבלנו No instance (הערה: לא ניתן לקבוע שקיבלנו YES instance, מאחר ואלגוריתם הקירוב נותן לנו חסם עליון של $\frac{b}{a}$, אך ייתכן כי קיבלנו קלט בין YES ולבין NO, ודווקא עליו האלגוריתם רץ טוב יותר וכך קיבלנו תשובה b ולא תשובה בעלת ערך גדול מ- b).לפיכך, כתשובה לבעיית GAP נענה YES. במקרה שקיבלנו תשובה ≥ b - נענה NO.

מ.ש.ל.

משפט 1: אם בעיית GAP היא NP-hard אזי גם קירוב של $\frac{b}{a}$ הינו NP-hard.

הוכחה: נניח בשלילה כי קירוב $\frac{b}{a}$ אינו NP-hard, אזי לפי הטענה הקודמת, על-ידי קירוב כנ"ל ניתן למצוא פתרון לבעיית GAP, בסתירה להיותה *NP-hard*.

מ.ש.ל.

הגדרה: מעגל המילטוני בגרף הינו מעגל שמתחיל בקודקוד v , עובר בכל קודקוד בדיוק פעם אחת ומסתיים ב-v.

הגדרה: בעיית הסוכן הנוסע: בהינתן גרף G=(V,E) ופונקציית משקל על הצלעות $c:V\rightarrow\Re^{+}$
מטרה: למצוא מעגל המילטוני בעל משקל מינימלי.

נוכיח כי בעיית TSP (הסוכן הנוסע) , ללא שימוש באי-שוויון המשולש, קשה לקירוב לכל קבוע>0 .
תחילה נבצע רדוקציה ממעגל המילטוני:
בהינתן $G=(V,E)$ נגדיר משקולות עבור ה – TSP באופן הבא:

$$w(i,j)=\begin{cases}1 & \text{(i,j)} \in \text{E} \\ \infty & \text{(i,j)} \notin \text{E}\end{cases}$$

אם בגרף המקורי היה מעגל המילטוני, אזי, מאחר ומבקרים בכל קודקוד בדיוק פעם אחת, קיים TSP בגרף הנתון בגודל |V|.
אחרת, פתרון ה- TSP עובר דרך "אין צלע" וערך הפתרון הינו ∞ .

מסקנה: בעיית ה GAP הבאה היא NP-קשה:

- YES: האם ערך הפתרון ≤ |V|.
- NO: האם הערך הוא ∞ .

מסקנה 2: קירוב $\frac{\infty}{|V|}=\infty$ הוא NP קשה.

<u>MAX-3-SAT:</u> נתונה נסחה בצורת CNF בעלת פסוקיות בגודל 3.
המטרה: למצוא השמה שממקסמת את מספר הפסוקיות המסופקות.

<u>משפט 2:</u> לכל $\varepsilon>0$, קשה לקרב את MAX-3-SAT עד כדי $\frac{8}{7}-\varepsilon$. (כלומר, לא ניתן למצוא אלגוריתם המבטיח שיוכל למצוא יותר מ- $\frac{7}{8}$ מכלל הפסוקיות הניתנות לסיפוק)

<u>הוכחה:</u> נתחיל מבעיה אחרת ונוכיח שהיא הינה קשה. לאחר מכן, נבצע רדוקציה ל - *MAX-3-SAT*:

MAX-E3-LIN: נתונה מערכת משוואות לינאריות עם n משתנים ו- m משוואות מעל F (mod 2) עם 3 משתנים בדיוק בכל משוואה.
מטרה: למקסם את מספר המשוואות המסופקות.

<u>משפט 3:</u> בעיית ה- GAP הבעה היא NP קשה: (יש לשים לב כי זוהי בעיית מקסימיזציה ולכן ההגדרה הפוכה ממה שראינו קודם)
YES: הפתרון האופטימלי מספק $(1-\varepsilon)M$ (או יותר) מהמשוואות.
NO: הפתרון האופטימלי מספק $\frac{M}{2}$ מהמשוואות.
<u>הוכחה:</u> ההוכחה קשה ולא נוכיח אותה.

<u>רדוקציה (מ- MAX-E3-LIN ל- MAX-3-SAT):</u> נתרגם כל מערכת משוואות ל- 4M פסוקיות ו- n משתנים.

• בהינתן משוואה: a+b+c = 1 נתרגמה ל –

$(a \vee b \vee c) \wedge (a \vee \overline{b} \vee \overline{c}) \vee (\overline{a} \vee b \vee \overline{c}) \vee (\overline{a} \vee \overline{b} \vee c)$

• בהינתן משוואה: a+b+c = 0 נתרגמה ל –

$(\overline{a} \vee \overline{b} \vee \overline{c}) \vee (\overline{a} \vee b \vee c) \vee (a \vee \overline{b} \vee c) \vee (a \vee b \vee \overline{c})$

נשים לב שבהינתן השמה נכונה, כל 4 הפסוקיות מסתפקות. אחרת, 3 פסוקיות מסתפקות.

GAP עבור MAX-3-SAT:

YES: $\underbrace{3M\varepsilon}_{\text{מש' שלא הסתפקו}} + \underbrace{4(1-\varepsilon)M}_{\text{מש' שהסתפקו}} = 4M - \varepsilon M$

NO: $\frac{M}{2}4 + \frac{M}{2}3 = 3.5M$

אם היינו יודעים לפתור את בעיית ה GAP הזאת , היינו יודעים לפתור את בעיית ה GAP הקודמת. אך לפי המשפט 3 בעייה זו הינה NP קשה לכן, גם בעייה זו הינה קשה. לפיכך, לפי טענה 1, קירוב זה: $\frac{4M - \varepsilon M}{3.5M} = \frac{8}{7} - \frac{2}{7}\varepsilon$ הינו NP קשה.

מ.ש.ל משפט 2

<u>הגדרה:</u>

<u>PTAS</u>: לכל $\varepsilon > 0$ **קבוע**, יש לנו קירוב $1+\varepsilon$ והאלגוריתם הינו פולינומי בגודל הקלט.

<u>FPTAS</u>: לכל $\varepsilon > 0$ **קבוע**, יש לנו קירוב $1+\varepsilon$ והאלגוריתם הינו פולינומי בגודל הקלט וב- $\frac{1}{\varepsilon}$.

<u>Knapsack:</u> נתון שק בגודל B. $S=\{a_1 \ldots a_n\}$ $|S|=n$ קבוצה של פריטים. לכל פריט ישנם שני מאפיינים – רווח $profit(a_i)$ ומשקל $size(a_i)$.

המטרה: למצוא קבוצה $S` \subseteq S$ כך ש: $\sum_{a_i \in S`} size(a_i) \le B$ ו- $\sum_{a_i \in S`} profit(a_i)$ מקסימלי.

נזכיר בכמה מילים כיצד רץ האלגוריתם הדינאמי:

$$P := \max_i \ \text{profit}(a_i)$$

נבנה טבלא A בגודל nP x n. תא A(p,i) יסמל את קבוצת הפריטים מתוך $\{a_1 \ldots a_i\}$ במשקל מינימלי שערכה בדיוק p. נעבור על הטבלא מלמלה למטה, לשמאל לימין, כאשר כלל המעבר יהיה: $A(p,i+1) = \min \begin{Bmatrix} A(p,i) \\ A(p - profit(a_i), i) + size(a_{i+1}) \end{Bmatrix}$. ניתן להסביר זאת באופן הבא:

בכל שלב קיימת בדיוק אחת מן האפשרויות הבאות: או שהאיבר האחרון נמצא בפתרון **או** שהאיבר האחרון אינו נמצא בפתרון. אזי, מאחר וכבר פתרנו את הבעיות הקטנות יותר, נבחר באופציה הטובה (המינימלית) מבין השתים.

הבעיה היא בכך שהאלגוריתם הנ"ל פותר את הבעיה ב $O(nnP)$ כלומר פולינומי בערך ה*רווח* המירבי, וזהו מצב שאנו רוצים להמנע ממנו.