

שיעור 13

סיבוכיות מקום ושלמות ב PSPACE

13.1 הגדרה של סיבוכיות מקום

הגדרה 13.1 סיבוכיות מקום של מבנה טיורינג

הסיבוכיות מקום של מ"ט M על קלט w היא פונקציה $(|w|) f$ השווה למספר התאי סרט לכל היותר של המוכנה M שבהם נעשה שימוש בחישוב של M על w .

הגדרה 13.2 המחלקה $SPACE(f(n))$

מחלקת $(f(n))$ היא אוסף כל השפות L עבורן קיימת מבנה טיורינג דטרמיניסטי M שמכריעה אותה כך ש: על כל קלט w באורך $|w| = n$, המוכנה M משתמשת לכל היותר $O(f(n))$ תאים סרט. $SPACE(f(n)) = \{L \mid L \text{ שמכריעה } L \text{ ומשתמשת לכל היותר ב- } O(f(n)) \text{ תאים סרט.}\}$

דוגמה 13.1

נראה כי ניתן לפתר את הבעיה SAT ע"י אלגוריתם שהוא רץ במקום ליניארי. כלומר:

$$SAT \in SPACE(n).$$

תהי ϕ נוסחה בוליאנית כלשהי. נסמן $|\phi| = n$ ונסמן ב- m את מספר המשתנים ב- ϕ . נגדיר מבנה M שפועלת כך:

$$\text{על כל קלט } \langle\phi\rangle = M$$

1 M רושמת את המחרוזת $\langle\phi\rangle$ על סרט הקלט.

2 לכל השמה a_1, a_2, \dots, a_m (כאשר $a_i \in \{0, 1\}$) הוא הערך הנוכחי של x_i :

a M רושמת את מהירות של ההשמה $a_1 a_2 \dots a_m$ על סרט העבודה.

b M מחשבת את הערך של ϕ עבור ההשמה הנוכחית a_1, \dots, a_m ע"י סריקה של הקלט $\langle\phi\rangle$ שרשום על סרט הקלט.

ג אם מתקיים $\phi(a_1, \dots, a_m) = 1$ אז M מקבלת.

3 אם עבור כל ההשומות התקבל $0 = \phi(a_1, \dots, a_m)$ אז M דוחה.

מכאן אנחנו רואים כי המוכנה M_1 רצתה במקום ליניארי. בפרט:

- M שומרת על סרט העבודה את ההשמה $a_1 \dots a_m$ וזה נדרש $O(m)$ תאימים.

- המספר המשתנים, m הוא n לכל היותר.

- לכן M רצה במקום $O(n)$.

לפיכך:

$$SAT \in SPACE(n) .$$

הגדלה 13.3 המחלוקת $NSPACE(f(n))$

מחלקה $NSPACE(f(n))$ היא אוסף כל השפות L עבורן קיימת מכונה טירונג אי-דטרמיניסטיבית N שמכריעה אותה כך ש: על כל קלט w באורך $|w| = n$ המכונה N משתמשת לכל היותר $O(f(n))$ תאי סרט מתוך כל המסלולי חישוב של N .

$$NSPACE(f(n)) = \{L \mid \text{קיימים } L \text{ שפה } L \text{ ומשתמשת לכל היותר ב- } O(f(n)) \text{ תאי סרט.}\}$$

דוגמה 13.2

תהי השפה הבאה:

$$ALL_{NFA} = \{\langle A \rangle \mid L(A) = \Sigma^* \text{ עבורו } NFA \text{ הוא } A\} .$$

הוכיחו כי $ALL_{NFA} \in NSPACE(n)$.**פתרון:**

הפתרון מהתבסס על זה שזה פשוט יותר לבנות אלגוריתם המכريع את השפה המשילמה:

$$\overline{ALL_{NFA}} = \{\langle A \rangle \mid w \in \Sigma^* \text{ עבורו } A \text{ דוחה } w\} .$$

נשתמש במשפט מרכזי כדי לבנות אלגוריתם שמכريع את $\overline{ALL_{NFA}}$:

משפט 13.1

אם $M = (Q, \Sigma, \delta, q_0, F)$ הוא NFA וקיים מילה w שנדחה ע"י M אז האורך המילה $|w| \leq 2^q$ כאשר $q = |Q|$

לפנינו שנותאר את האלגוריתם עצמו, נגדיר סימון שנשתמש בו בבניית האלגוריתם. נניח ש- $(Q, \Sigma, \delta, q_0, F)$ היא מכונית NFA כלשהי. תהי $P(Q)$ הקבוצה החזקה של Q . עבר כל הפונקציית המעברים היא מהצורה

$$\delta : Q \times \Sigma \rightarrow P(Q) .$$

בהתנחת מילה $w = a_1 a_2 \dots a_n$ אשר Σ הוא התו ה- i של המילה, $n \leq i \leq 1$. נגדיר את הסדרה הבאה:

$$S_0, S_1, S_2, \dots, S_n$$

כך אשר

$$S_0 \triangleq \{q_0\}, \quad S_{i+1} \triangleq \delta(S_i, a_i),$$

כך אשר $S_i \in P(Q)$ לכל $0 \leq i \leq n$.

בנייה האלגוריתם

בנייה אלגוריתם לא-דטרמיניסטי, N המכريع את $\overline{ALL_{NFA}}$ באופן הבא:על כל קלט $x = N$:

1) בודקת אם $\langle M \rangle$, כאשר $M = (Q, \Sigma, \delta, q_0, F)$ היא מכונת NFA.

• אם לא $\Leftarrow N$ תדחה.

2) יי' $|Q| = q$ מספר המ מצבים של M . נגדיר $S_0 = \{q_0\}$.

3) N מבצעת את הלולאה הבאה:

$$\text{לכל } 0 \leq i \leq 2^q - 1$$

א) בוחרת באופן אידטרמיניסטי תו קלט Σ $a_i \in$.

ב) מחשבת

$$S_{i+1} = \delta(S_i, a_i).$$

ג) אם $N \Leftarrow S_{i+1} \cap F \neq \emptyset$ תדחה.

בפועל N בודקת את התנאי הזה ע"י לסמן את כל המ מצבים שב- S_{i+1} . אם אחד מה מצבים המסומנים הוא מצב קבלה אז N תדחה.

4) אם בסיום הלולאה לא הייתה מצב-קבלה באף אחת מן הקבוצות S_i אז N מקבל.

אם $x \in \overline{\text{ALL}_{NFA}}$

כasher A היא מכונת NFA. וקיימת מילה $w \in \Sigma^*$ כך ש- A תדחה.

\Leftarrow קיימת מילה w' באורך כל היותר 2^q ש- A תדחה.

\Leftarrow קיימת ריצה של N שבה N בוחרת את התווים של w' בלולאה.

\Leftarrow במהלך הריצה של A על w' , אין מצב קבלה באף אחת מן הקבוצות S_i .

\Leftarrow N לא דחתה עד סוף הלולאה.

\Leftarrow בסופה N מקבל.

אם $x \notin \overline{\text{ALL}_{NFA}}$ אז שני מקרים:

מקרה 1) $N \Leftarrow x \neq \langle A \rangle$ תדחה בשלב 1.

מקרה 2) $L(A) = \Sigma^*$ ו- $x = \langle A \rangle$

\Leftarrow לכל מילה $w \in \Sigma^*$, קיים שלב שבו A נמצא במצב קבלה.

\Leftarrow בכל ריצה של N , קיימת איטרציה i עבורה $\emptyset \neq \cap S_i \cap F \neq \emptyset$.

\Leftarrow באיטרציה זו N תדחה.

\Leftarrow בכל ריצה N תדחה.

\Leftarrow N דוחה את x .

סיבוכיות מקומ

• נסמן ב- $| \langle M \rangle | = n$ את אורך הקלט, וב- $|Q| = q$ את מספר המ מצבים של ה- NFA.

• כל מצב וכל מעבר של M מופיעים בקיים, מתקיים $O(n) = q$.

• במהלך כל ריצה, N שומרת רק את המידע הבא:

- * הקבוצה הנווחית $Q \subseteq S_i$ של מצבים אפשריים.
- לפועל N שומרת S_i בוקטור ביטים באורך q לכל היוטר.
- * מונה של האיטרציות הלולאה עד 2^q , המאושר ביצוג בינארי ודורש (q) ביטים.
- *תו קלט אחד הנבחר באופן א-דטרמיניסטי בכל איטרציה, ומשתני עזר לחישוב S_{i+1} , הדורשים מקום קבוע או בינארי ב- q .

לפיכך סיבוכיות המקום הכלולת של N היא

$$O(q) = O(n).$$

לפיכך האלגוריתם N פועל במקום לינארי.

משמעותו לב: N לינארי במקום אף על פי שזמן הריצה שלו עלול להיות אקספוננציאלי.

13.2 משפט סביז'

הגדלה 13.4 CANYIELD

בהתנחת מוכנות טירוגן א-דטרמיניסטי N , מספר טבעי חיובי t , ושתי קונפיגורציות c_1, c_2 של N (ראו את ההגדלה של קונפיגורציה בהגדלה 1.3). האלגוריתם $CANYIELD$ הוא אלגוריתם דטרמיניסטי הבודק אם ניתן לעבור מ- c_1 ל- c_2 על ידי לכל היוטר t צעדי חישוב של N . התאזר פסודוקוד של $CANYIELD$ הוא כדלקמן:

$$\langle N, c_1, c_2, t \rangle \text{ על קלט } = CANYIELD$$

1) רושם את c_1 ו- c_2 על מחסנית.

2) בודק אם N היא מוכנת טירוגן, c_1, c_2 קונפיגורציות ו- t מספר טבעי חיובי.

• אם לא אז הוא דוחה.

: $t = 1$ **3)**

• אם $c_1 = c_2$ אז הוא מקבל.

• אחרת אם $c_1 \vdash_N c_2$ (אם אפשר לעבור מ- c_1 ל- c_2 בצעד אחד [ראו הגדלה 1.4]) מקבל.

• אחרת הוא דוחה.

4) אם $t > 1$, לכל קונפיגורציה c_k של הריצה של N על w אשר משתמשת במקום

(כאשר w היא המילה הנקרהת של הקונפיגורציה (c_k)):

$$CANYIELD\left(N, c_1, c_k, \left\lfloor \frac{t}{2} \right\rfloor\right) \text{ מרץ} \quad (5)$$

כאשר $\left\lfloor \frac{t}{2} \right\rfloor$ הוא השלם הכי הקרוב ל- $\frac{t}{2}$ וקטן מ- $\frac{t}{2}$ או שווה ל- $\frac{t}{2}$.

$$CANYIELD\left(N, c_k, c_2, \left\lceil \frac{t}{2} \right\rceil\right) \text{ מרץ} \quad (6)$$

כאשר $\left\lceil \frac{t}{2} \right\rceil$ הוא השלם הכי הקרוב ל- $\frac{t}{2}$ וגדול מ- $\frac{t}{2}$ או שווה ל- $\frac{t}{2}$.

7) אם שתי הרצות בשלבי 4) ו- 5) הסתיימו בקבלה \Leftarrow מקבל.

8) אחרת אם לא התקבלה תשובה קבלה \Leftarrow דוחה.

משפט 13.2 משפט סבץ'

לכל פונקציה $f(n) \geq n$, אם $f : \mathbb{N} \rightarrow \mathbb{R}^+$

$$NSPACE(f(n)) \subseteq SPACE(f^2(n)).$$

הוכחה:

הריינו של ההוכחה:

תהי N מ"ט אי-דטרמיניסטי שמכריעה את השפה A במקומות $O(f(n))$, כאשר n אורך הקלט w של N .
נבנה מכונת טיריניג דטרמיניסטית, M שמכריעה את A במקומות $O(f^2(n))$.
כלומר, בהינתן $N \in NSPACE(f(n))$ המכריעה שפה A , נבנה $M \in SPACE(f^2(n))$ המכריעה A .
כלומר, אנחנו נראה שלכל $N \in NSPACE(f(n))$ קיימת $M \in SPACE(f^2(n))$ כזו ש M המכריעה A במקומות $O(f^2(n))$.

$$NSPACE(f(n)) \subseteq SPACE(f^2(n)).$$

בנייה המכונה:

תהי N מכונת טיריניג אי-דטרמיניסטי שמכריעה השפה A .
תהי w מחרוזת שהיא הקלט של N .
בהינתן שתי קונפיגורציות c_1, c_2 של N ומספר טבעי t .

- אם ניתן לעבור מ- c_1 ל- c_2 בכל היותר t צעדים $\Leftarrow CANYIELD(N, c_1, c_2, t)$ מקבל.
- אחרות $\Leftarrow CANYIELD(N, c_1, c_2, t)$.

נגידר מכונת טיריניג דטרמיניסטי M שמסמלצת את המכונה האי-דטרמיניסטי N באופן הבא.

ראשית נסמן ב- n את אורך הקלט w של N .

תהי c_0 הקונפיגורציה ההתחלתית.

נתקן את N כך שלאחר כל ריצה הראש חוזר לקצה השמאלי של תוכן הסרטט ו- N עוברת לkonfiguracji c_{acc} .

נגידר d כך ש- $2^{df(n)}$ הוא חסם עלין של מספר הקונפיגורציות שקיימות בריצות של N שדורשות $O(f(n))$ מוקום.

המכונת טיריניג הדטרמיניסטי M תוגדר כך:

M על קלט w :

1) מריצה $CANYIELD$ על הקלט $\langle N, c_0, c_{acc}, 2^{df(n)} \rangle$ ועונה כמוות.

הוכחת נכונות:

נניח $N \in NSPACE(f(n))$ ו- $w \in L(N)$.

\Leftarrow לפי ההגדרה של d , ל- N יש $2^{df(n)}$ לכל היותר.

\Leftarrow קיימים מסלול חישוב N על w מ- c_0 ל- c_{acc} .

\Leftarrow האלגוריתם $CANYIELD$ על הקלט $\langle N, c_0, c_{acc}, 2^{df(n)} \rangle$ יקבל.

$M \Leftarrow \text{יקבל } w.$

$N \in NSPACE(f(n))$ ו- $w \notin L(N)$.

\Leftarrow לפי ההגדרה של d , ל- N יש $2^{df(n)}$ לכל היותר.

\Leftarrow לא קיים מסלול חישוב של N על w מ- c_0 ל- c_{acc} .

\Leftarrow האלגוריתם $CANYIELD$ על הקלט $\langle N, c_0, c_{\text{acc}}, 2^{df(n)} \rangle$ ידחה.

$M \Leftarrow \text{ידחה } w.$

סיבוכיות מקום:

- כל פעם ש- $CANYIELD$ מפעיל את עצמו באופן רקורסיבי, הוא רושם את c_1, c_2 ו- t על מחסנית, כך שניתן יהיה לשזר אותו לאחר הקריאה הרקורסיבית הבאה.

- בכלל ש- $N \in NSPACE(f(n))$ אז הכתיבה של c_1, c_2 ו- t על המחסנית דורשת $O(f(n))$ מקום.

- בכל שלב של הרקורסיה, האלגוריתם $CANYIELD$ מחלק את t ב- 2.

- הערך ההתחלתי של t הוא $2^{df(n)}$ שכן העומק של הרקורסיה הוא

$$O(\log_2(2^{df(n)})) = O(f(n)).$$

- שכן המקום הכלול ש- M דורש הוא $O(f^2(n))$.

לפיכך

$$M \in SPACE(f^2(n)).$$

לסיכום: הוכחנו שבהינתן מכונת א-טרמיניסטית N כלשהי שמכריעה שפה A כלשי עבורה

$$N \in NSPACE(f(n)),$$

קייםת מכונת טיריניג דטרמיניסטית M שמכריעה A במקום $O(f^2(n))$, כלומר:

$$M \in SPACE(f^2(n)).$$

לפיכך:

$$NSPACE(f(n)) \subseteq SPACE(f^2(n)).$$



13.3 המחלוקת PSPACE

ההגדרה הבאה היא האנלוג של הגדרה 10.2 של אלגוריתם זמן פולינומייאלי.

הגדרה 13.5 אלגוריתם מקום פולינומייאלי

אומרים כי אלגוריתם A מכריע בעיה במקומות פולינומייאלי אם קיים קבוע $c > 0$ כך שהמקומות הריצה של A על קלט w חסום ע"י $O(|w|^c)$.

התזה של צרץ' טיריניג אומר שאם קיים אלגוריתם המכריע בעיה במקומות פולינומייאלי, אז קיימת מכונת טיריניג דטרמיניסטית המכricaעת את השפה השקולה לעביה זו במקומות פולינומייאלי.

. אלגוריתם מכריע \equiv מכונת טיריניג דטרמיניסטית

הגדרה 13.6 המחלקה $PSPACE$

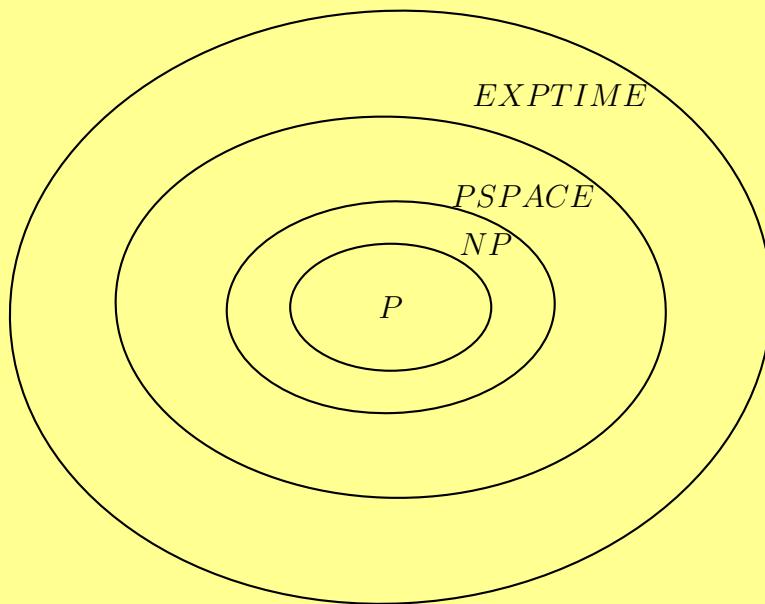
המחלקה $PSPACE$ היא אוסף כל הבעיה (השפות) שקיים עבורן אלגוריתם (מכונת טירונג) דטרמיניסטי המכריע אותו במקומות פולינומילי.

הגדרה 13.7 המחלקה $NPSPACE$

המחלקה $NPSPACE$ היא אוסף כל הבעיה (השפות) שקיים עבורן אלגוריתם (מכונת טירונג) אי-דטרמיניסטי המכריע אותו במקומות פולינומילי.

משפט 13.3

$$P \subseteq NP \subseteq PSPACE \subseteq EXPTIME .$$

**13.4 שלמות ב- $PSPACE$** **הגדרה 13.8 $PSPACE$ קשה**

בעיה B נקראת $PSPACE$ קשה אם לכל בעיה $A \in PSPACE$ קיימת רדוקציה זמן פולינומיאלית מ- A ל- B . כלומר: $A \leq_P B$.

הגדרה 13.9 שלמות $PSPACE$

בעיה B נקראת $PSPACE$ שלמה אם שני התנאים הבאים מתקיימים:

$$B \in PSPACE \quad (1)$$

(2) לכל בעיה $A \in PSPACE$ קיימת רדוקציה זמן פולינומיאלית מ- A ל- B . כלומר: $A \leq_p B$.

13.5 נוסחאות בוליאניות עם כמתים

בפרק 11 ו- 12 הגדרנו נוסחה בוליאנית כביטוי מתמטי שבינוי מהמרכיבים הבאים:

- משתנים בוליאניים, שמקבלים את הערכים 0 ו- 1 (לעתים מסומנים F ו- T).
- אופרטורים בוליאניים עיקריים

ונם	\wedge
או	\vee
לא	\neg

כעת נכליל את החגדרה זו לסוג היותר מורחב של נוסחה בוליאנית: נוסחה בוליאנית עם כמתים.

הגדרה 13.10 נוסחת בוליאנית עם כמתים - QBF

בנוסחת בוליאנית עם כמתים מופיעעה אחת מהשני כמתים העיקריים העיקריים:

"לכל" (נקרא גם "כמת כולל")	\forall
"קיים" (נקרא גם "כמת יש")	\exists

דוגמה 13.3 דוגמאות של נוסחאות בוליאניות עם כמתים

בדוגמאות הבאות y , x הם משתנים בוליאניים. קלומר $x, y \in \{0, 1\}$.

$$\phi = \forall x \exists y [(x \vee y) \wedge (\bar{x} \vee \bar{y})] . \quad (1)$$

בדוגמה זו $\phi = 1$.

$$\phi = \forall x [x \vee \bar{x}] \rightarrow \phi = 1 . \quad (2)$$

$$\phi = \exists x (x \wedge \bar{x}) \rightarrow \phi = 0 . \quad (3)$$

הגדרה 13.11 $TQBF$

$\langle \phi \rangle$ בשפה $TQBF$ אם ϕ נוסחת בוליאנית עם כמתים והנוסחה מעורכת לאמת.

$$TQBF = \{ \langle \phi \rangle \mid \phi \text{ נוסחת בוליאנית עם כמתים ו- } \phi = 1 \} .$$

הערה 13.1

בניגוד ל- SAT עבורה השאלה היא האם האם קיימת הצבת אמת, ב- $TQBF$ לכל נוסחה יש ערך אמת או שקר ייחיד, מכיוון שאין משתנים חופשיים.

משפט 13.4 $SAT \subseteq TQBF$.

הוכחה: תרגיל בית.

13.6 המחלקה L**13.7 המחלקה NL****13.8 שלמות ב- NL****13.9 שיוויון NL ו- coNL**