שיעור 13
סיבוכיות מקום ושלמות ב PSPACE

13.1 הגדרה של סיבוכיות מקום

הגדרה 13.1 סיבוכיות מקום של מכונת טיורינג

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

הגדרה 13.2 המחלקה $SPACE(f(n))$

מחלקת $SPACE(f(n))$ היא אוסף כל השפות L עבורן קיימת מכונת טיורינג דטרמיניסטית M שמכריעה אותה כך ש:
על כל קלט w באורך $n=|w|$, המכונה M משתמשת לכל היותר $O(f(n))$ תאי סרט.

$$SPACE(f(n)) = \{L \mid \text{ מ"ט } \exists M \text{ שמכריעה } L \text{ ומשתמשת לכל היותר ב- } O(f(n)) \text{ תאי סרט} \} \ .$$

דוגמה 13.1

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

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

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

$M=$ על כל קלט $\langle\phi\rangle$:

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

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

א) M רושמת את מחרוזת של ההשמה $a_1a_2\cdots a_m$ על סרט העבודה.

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

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

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

מכאן אנחנו רואים כי המכונה M_1 רצה במקום ליניארי. בפרט:

- M שומרת על סרט העבודה את ההשמה $a_1\ldots a_m$ וזה נדרש $O(m)$ תאים.
- המספר המשתנים, m הוא n לכל היותר.

- לכן M רצה במקום $O(n)$.

לפיכך:
$$SAT \in SPACE(n) \ .$$

הגדרה 13.3 המחלקה $NSPACE(f(n))$

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

$$NSPACE(f(n)) = \{L \mid \exists \text{ מ"ט א"ד } N \text{ שמכריעה } L \text{ ומשתמשת לכל היותר ב- } O(f(n)) \text{ תאי סרט} \} .$$

דוגמה 13.2

תהי ALL_{NFA} השפה הבאה:
$$ALL_{NFA} = \{\langle A\rangle \mid A \text{ הוא } NFA \text{ עבורו } L(A)=\Sigma^* \} \ .$$
הוכיחו כי $ALL_{NFA} \in NSPACE(n)$.

פתרון:
הפתרון מתבוסס על זה שזה פשוט יותר לבנות אלגוריתם המכריע את השפה המשלימה:
$$\overline{ALL_{NFA}} = \{\langle A\rangle \mid \text{ קיימת } w\in\Sigma^* \text{ עבורה } A \text{ דוחה } w \} \ .$$
נשתמש במשפט מרכזי כדי לבנות אלגוריתם שמכריע את $\overline{ALL_{NFA}}$:

משפט 13.1

אם $M=(Q,\Sigma,\delta,q_0,F)$ הוא NFA וקיים מילה w שנדחת ע"י M אז האורך המילה $|w|\leqslant 2^q$ כאשר $q=|Q|$ הוא המספר המצבים של M, וקיימים אינסוף מלים שנדחות ע"י M.

לפני שנתאר את האלגוריתם עצמו, נגדיר סימון שנשתמש בו בבניית האלגוריתם. נניח ש- $M=(Q,\Sigma,\delta,q_0,F)$ היא מכונת NFA כלשהי. תהי $P(Q)$ הקבוצת החזקה של Q. עבור כל NFA הפונקצית המעברים היא מהצורה
$$\delta : Q\times\Sigma \to P(Q) \ .$$
בהינתן מילה $w=a_1a_2\cdots a_n$ כאשר $a_i\in\Sigma$ הוא התו ה- i של המילה, $1\leqslant i\leqslant n$. נגדיר את הסדרה הבאה:
$$S_0 \ , \ S_1 \ , \ S_2 \ , \quad \ldots \quad , \ S_n$$
כאשר
$$S_0 \triangleq \{q_0\} \ , \quad S_{i+1} \triangleq \delta(S_i,a_i) \ ,$$
כאשר $S_i\in P(Q)$ לכל $0\leqslant i\leqslant n$.

<u>בניית האלגוריתם</u>

נבנה אלגוריתם לא-דטרמיניסטי, N המכריע את $\overline{ALL_{NFA}}$ באופן הבא:

$N=$ "על כל קלט x:

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

- אם לא $\Leftarrow$ N תדחה.

2) יהי $q = |Q|$ מספר המצבים של M. נגדיר $S_0 = \{q_0\}$.

3) N מבצעת את הלולאה הבאה:

לכל $0 \leqslant i \leqslant 2^q - 1$:

א) בוחרת באופן אי־דטרמיניסטי תו קלט $a_i \in \Sigma$.

ב) מחשבת

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

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

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

4) אם בסיום הלולאה לא היה מצב-קבלה באף אחת מן הקבוצות S_i אז N תקבל. "

אם $x \in \overline{ALL_{NFA}}$:

$\Leftarrow$ $x = \langle A \rangle$, כאשר A היא מכונת NFA. וקיימת מילה $w \in \Sigma^*$ כך ש־ A תדוחה.

$\Leftarrow$ קיימת מילה w' באורך לכל 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 תקבל.

אם $x \notin \overline{ALL_{NFA}}$ אז שני מקרים:

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

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

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

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

$\Leftarrow$ באיטרציה זו N תדחה.

$\Leftarrow$ בכל ריצה N תדחה.

$\Leftarrow$ N דוחה את x.

סיבוכיות מקום

- נסמן ב־ $n = |\langle M \rangle|$ את אורך הקלט, וב־ $q = |Q|$ את מספר המצבים של ה־ NFA.
- כל מצב וכל מעבר של M מופיעים בקידוד, מתקיים $q = O(n)$.
- במהלך כל ריצה, N שומרת רק את המידע הבא:

* הקבוצה הנוכחית $S_i \subseteq Q$ של מצבים אפשריים.
לפועל N שומרת S_i בוקטור ביטים באורך q לכל היותר.
* מונה של האיטרציות הלולאה עד 2^q, המאוחסן בייצוג בינארי ודורש $O(q)$ ביטים.
* תו קלט אחד הנבחר באפן אי-דטרמיניסטי בכל איטרציה, ומשתני עזר לחישוב S_{i+1}, הדורשים מקום קבוע או לינארי ב־q.

לפיכך סיבוכיות המקום הכוללת של N היא
$$O(q) = O(n) \ .$$
לפיכך האלגוריתם N פועל במקום לינארי.

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

13.2 משפט סביץ'

הגדרה 13.4 $CANYIELD$

בהינתן מכונת טיורנג אי-דטרמיניסטית N, מספר טבעי חיובי t, ושתי קונפיגורציות c_1, c_2 של N (ראו את ההגדרה של קונפיגורציה בהגדרה 1.3). האלגוריתם $CANYIELD$ הוא אלגוריתם דטרמיניסטי הבודק אם ניתן לעבור מ- c_1 ל- c_2 על ידי לכל היותר t צעדי חישוב של N.
התאור פסאודוקוד של $CANYIELD$ הוא כדלקמן:

$CANYIELD =$ על קלט $\langle N, c_1, c_2, t\rangle$:

1) רושם את c_1, c_2 ו- t על מחסנית.

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

- אם לא אז הוא דוחה.

3) אם $t = 1$:

- 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 הגדרה 1.4]) מקבל.
- אחרת הוא דוחה.

4) אם $t > 1$, לכל קונפיגורציה c_k של הרצה של N על w אשר משתמשת במקום $f(n)$ (כאשר w היא המילה הנקראת של הקונפיגורציה c_k):

5) מריץ $$CANYIELD\left(N, c_1, c_k, \left\lfloor \frac{t}{2} \right\rfloor\right)$$
כאשר $\left\lfloor \frac{t}{2} \right\rfloor$ הוא השלם הכי הקרוב ל- $\frac{t}{2}$ וקטן מ- או שווה ל- $\frac{t}{2}$.

6) מריץ $$CANYIELD\left(N, c_k, c_2, \left\lceil \frac{t}{2} \right\rceil\right)$$
כאשר $\left\lceil \frac{t}{2} \right\rceil$ הוא השלם הכי הקרוב ל- $\frac{t}{2}$ וגדול מ- או שווה ל- $\frac{t}{2}$.

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

8) אחרת אם לא התקבלה תשובת קבלה $\Leftarrow$ דוחה.

משפט 13.2 משפט סביץ'

לכל פונקציה $f:\mathbb{N}\to\mathbb{R}^+$, אם $f(n)\geqslant n$ אז

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

הוכחה:

הריעון של ההוכחה:

תהי N מ"ט אי-דטרמיניסטית שמכריעה את השפה A במקום $O\left(f(n)\right)$, כאשר n אורך הקלט w של N.
נבנה מכונת טיורינג דטרמיניסטית, M שמכריעה את A במקום $O\left(f^2(n)\right)$.
כלומר, בהינתן $N\in NSPACE\left(f(n)\right)$ המכריעה שפה A, נבנה $M\in SPACE\left(f^2(n)\right)$ המכריעה A.
כלומר, אנחנו נראה שלכל $N\in NSPACE\left(f(n)\right)$ קיימת $M\in SPACE\left(f^2(n)\right)$.
באופן הזה אנחנו נוכיח כי

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

בניית המכונה:

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

- אם ניתן לעבור מ- c_1 ל- c_2 בלכל היותר t צעדים $\Leftarrow$ $CANYIELD\left(N,c_1,c_2,t\right)$ מקבל.
- אחרת $\Leftarrow$ $CANYIELD\left(N,c_1,c_2,t\right)$ דוחה.

נגדיר מכונת טיורינג דטרמיניסטית M שמסמלצת את המכונה האי-דטרמיניסטית 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 השמאל של תוכן הסרט ו- N עוברת לקונפיגורציה c_{acc}.

נגדיר d כך ש- $2^{df(n)}$ הוא חסם עליון של מספר הקונפיגורציות שקיימות בריצות של N שדורשות $O\left(f(n)\right)$ מקום.

המכונת טיורינג הדטרמיניסטית M תוגדר כך:

$M=$ "על קלט w:

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

הוכחת הנכונות:

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

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

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

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

$\Leftarrow$ M יקבל w.

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

$\Leftarrow$ 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.

$\Leftarrow$ M 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 לשחזר אותם לאחר הקריאה הרקורסיבית הבאה.
- בגלל ש- $N \in NSPACE(f(n))$ אזי 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 $2^{df(n)}$ לכן העומק של הרקורסיה הוא
$$O\left(\log_2\left(2^{df(n)}\right)\right) = O(f(n)) \ .$$
- לכן המכום הכולל ש- M דורש הוא $O\left(f^2(n)\right)$.

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

לסיכום: הוכחנו שבהינתן מכונת אי-דטרמיניסטית N כלשהי שמכריעה שפה A כלשהי עבורה
$$N \in NSPACE(f(n)) \ ,$$
קיימת מכונת טיורינג דטרמיניסטית M שמכריעה A במקום $O\left(f^2(n)\right)$, כלומר:
$$M \in SPACE\left(f^2(n)\right) \ .$$

לפיכך:
$$NSPACE(f(n)) \subseteq SPACE\left(f^2(n)\right) \ .$$

■

13.3 המחלקה PSPACE

ההגדרה הבאה היא האנלוג של ההגדרה 10.2 של אלגוריתם זמן פולינומיאלי.

הגדרה 13.5 אלגוריתם מקום פולינומיאלי

אומרים כי אלגוריתם A מכריע בעייה במקום פולינומיאלי אם קיים קבוע $c > 0$ כך שהמקום הריצה של A על קלט w חסום ע"י $O\left(|w|^c\right)$.

התזה של צרץ' טיורינג אומר שאם קיים אלגוריתם המכריע בעייה במקום פולינומיאלי, אזי קיימת מכונת טיורינג דטרמיניסטית המכריעה את השפה השקולה לבעייה זו במקום פולינומיאלי.

$$\text{אלגוריתם מכריע} \equiv \text{מכונת טיורינג דטרמיניסטית} \ .$$

הגדרה 13.6 המחלקה $PSPACE$

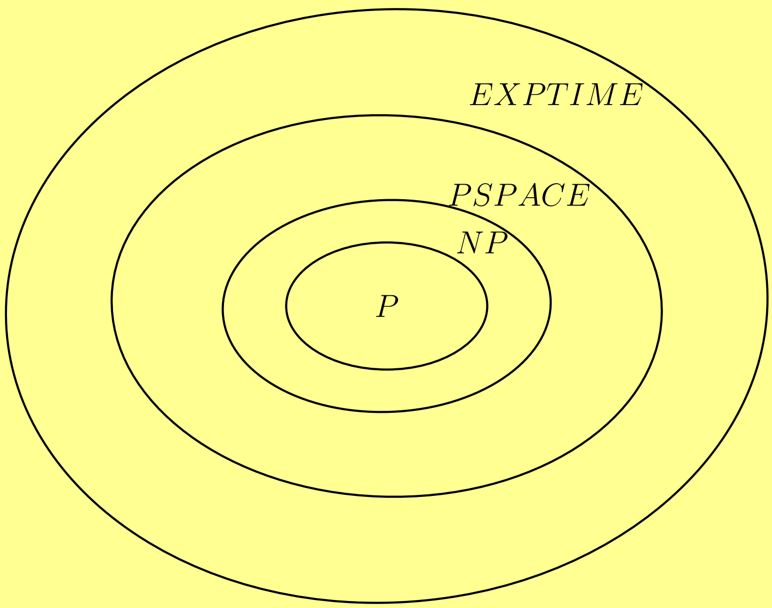
המחלקה $PSPACE$ היא אוסף כל הבעיות (השפות) שקיים עבורן אלגוריתם (מכונת טיורנג) דטרמיניסטי המכריע אותן במקום פולינומיאלי.

הגדרה 13.7 המחלקה $NPSPACE$

$NPSPACE$ היא אוסף כל הבעיות (השפות) שקיים עבורן אלגוריתם (מכונת טיורנג) אי-דטרמיניסטי המכריע אותן במקום פולינומיאלי.

משפט 13.3

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



13.4 שלמות ב- $PSPACE$

הגדרה 13.8 $PSPACE$ קשה

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

הגדרה 13.9 $PSPACE$ שלמות

בעייה B נקראת $PSPACE$ שלמה אם השני התנאים הבאים מתקיימים:

1) $B \in PSPACE$

2) לכל בעייה $A \in PSPACE$ קיימת רדוקציה זמן פולינומיאלית מ- A ל- B. כלומר: $A \leqslant_p B$.

13.5 נוסחאות בוליאניות עם כמתים

בפרקים 11 ו- 12 הגדרנו נוסחה בוליאנית כביטוי מתמטי שבנוי מהמרכיבים הבאים:

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

$\wedge$	וגם
$\vee$	או
$\neg$	לא

כעת נכליל את ההגדרה הזו לסוג היותר מורחב של נוסחה בוליאנית: נוסחה בוליאנית עם כמתים.

הגדרה 13.10 נוסחה בוליאנית עם כמתים - QBF

בנוסחת בוליאנית עם כמתים מופיעה אחת מהשני כמתים העיקריים:

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

דוגמה 13.3 דוגמאות של נוסחאותת בוליאניות עם כמתים

בדוגמאות הבאות x,y הם משתנים בוליאנים. כלומר $x,y \in \{0,1\}$.

(1

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

בדוגמה זו $\phi = 1$.

(2

$$\phi = \forall\ x\ \ [x \vee \bar{x}] \qquad \to \qquad \phi = 1\ .$$

(3

$$\phi = \exists\ x\ \ (x \wedge \bar{x}) \qquad \to \qquad \phi = 0\ .$$

הגדרה 13.11 $TQBF$

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

$$TQBF = \left\{ \langle\phi\rangle \ \middle|\ \text{נוסחת בוליאנית עם כמתים ו-}\ \phi = 1 \quad \phi \right\}\ .$$

הערה 13.1

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

משפט 13.4

$$SAT \subseteq TQBF\ .$$

הוכחה: תרגיל בית. ■

משפט 13.5

השפה $TQBF$ היא NP שלמה.

הוכחה: נראה כי

1) $TQBF \in PSPACE$
2) לכל $A \in PSPACE$ מקתיים $A \leqslant_P TQBF$.

נבנה אלגוריתם רקורסיבי $A \in PSPACE$ שמכריע את $TQBF$ באופן הבא.

<u>**בניית האלגוריתם**</u>

$A =$ "על קלט $\langle \psi \rangle$ כאשר

$$\psi = Q_1 x_1 Q_2 x_2 \ldots Q_n x_n \phi\left(x_1, \ldots, x_n\right)\ ,$$

כאשר לכל $1 \leqslant i \leqslant n$, Q_i הוא כמת $\forall$ או $\exists$, x_i משתנה בוליאני ו- $\phi\left(x_1, \ldots, x_n\right)$ נוסחה בוליאנית בלי כמתים:

1) <u>מקרה הבסיס:</u>
אם $n=0$ אז $\phi\left(x_1, \ldots, x_n\right)$ הוא קבוע ומעריך אותה.
2) <u>מקרה הרקורסיבי:</u>
   - אם $Q_1 = \exists$:
     - מריץ $A\Big(\psi\left(x_1=0\right)\Big)$.
     - מריץ $A\Big(\psi\left(x_1=1\right)\Big)$.
     - אם **לפחות אחד מהם** התקבל אז מקבל.
   - אם $Q_1 = \forall$:
     - מריץ $A\Big(\psi\left(x_1=0\right)\Big)$.
     - מריץ $A\Big(\psi\left(x_1=1\right)\Big)$.
     - אם **שניהם** התקבלו אז מקבל."

<u>**הוכחת הנכונות**</u>

ניתן להוכיח הנכונות של A ע"י אינדוקציה.

<u>שלב הבסיס: $n=0$.</u>

אם $\psi = 1 \Leftarrow$ בשלב (1) A מחשב את $\phi(x_1, \ldots, x_n) = 1 \Leftarrow A$ מקבל.

אם $\psi = 0 \Leftarrow$ בשלב (1) A מחשב את $\phi(x_1, \ldots, x_n) = 0 \Leftarrow A$ דוחה.

<u>שלב המעבר: $n=k$.</u>

נניח ש- A מכריע נוסחה כלשהי עם כמתים $\psi = Q_1x_1Q_2x_2\ldots Q_kx_k\phi(x_1,\ldots,x_n)$ במקרה $n=k$. זוהי ההנחת האינדוקציה שלנו. נוכיח כי A גם מכריע נוסחה ψ כלשהי במקרה ש- $n=k+1$ באופן הבא:

אם $\psi(x_1,x_2,\ldots,x_k,x_{k+1})=1 \Leftarrow$ שני מקרים:

מקרה 1: $Q_1=\exists$

$\Leftarrow$ $\psi(x_1=0)=1$ או $\psi(x_1=1)=1$ ושתיהן QBF בעלי k משתנים וכמתים.

$\Leftarrow$ $A\big(\psi(x_1=0)\big)$ התקבל או $A\big(\psi(x_1=1)\big)$ התקבל.

$\Leftarrow$ A מקבל $\psi(x_1,\ldots,x_{k+1})$.

מקרה 2: $Q_1=\forall$

$\Leftarrow$ $\psi(x_1=0)=1$ וגם $\psi(x_1=1)=1$ ושתיהן QBF בעלי k משתנים וכמתים.

$\Leftarrow$ $A\big(\psi(x_1=0)\big)$ התקבל וגם $A\big(\psi(x_1=1)\big)$ התקבל.

$\Leftarrow$ A מקבל $\psi(x_1,\ldots,x_{k+1})$.

אם $\psi(x_1,x_2,\ldots,x_k,x_{k+1})=0 \Leftarrow$ שני מקרים:

מקרה 1: $Q_1=\exists$

$\Leftarrow$ $\psi(x_1=0)=0$ וגם $\psi(x_1=1)=0$ ושתיהן QBF בעלי k משתנים וכמתים.

$\Leftarrow$ $A\big(\psi(x_1=0)\big)$ דוחה וגם $A\big(\psi(x_1=1)\big)$ דוחה.

$\Leftarrow$ A דוחה $\psi(x_1,\ldots,x_{k+1})$.

מקרה 2: $Q_1=\forall$

$\Leftarrow$ $\psi(x_1=0)=0$ או $\psi(x_1=1)=0$ ושתיהן QBF בעלי k משתנים וכמתים.

$\Leftarrow$ $A\big(\psi(x_1=0)\big)$ דוחה או $A\big(\psi(x_1=1)\big)$ דוחה.

$\Leftarrow$ A דוחה $\psi(x_1,\ldots,x_{k+1})$.

סיבוכיות מקום של האלגוריתם

כדי לחשב את הסיבוכיות מקום, נסמן השימוש מקום ב- $S(n,m)$ כאשר n המספר המשתנים של ψ ו- $m=|\psi|$ הוא האורך של ψ. אז יש לנו את היחס רקורסיבי הבא:

$$S(0,m)=O(m)\ ,\qquad S(n,m)=S(n-1,m)+O(m)\ .$$

מכאן

$$S(n,m)=O(nm)\ .$$

הוכחה כי השפה $TQBF$ היא $PSPACE$ קשה: בניית הרדוקציה $L\leqslant_P TQBF$.

נראה כי לכל שפה $L\in PSPACE$ מתקיים

$$L\leqslant_P TQBF\ .$$

תהי L שפה השייכת ל- $PSPACE$ ותהי M מכונת טיורינג המכריעה את L במקום פולינומיאלי $S(n)$. נוכיח שקיימת פונקציה f כך ש:

$$x\in L \iff f(x)\in TQBF\ .$$

ז"א $f(x)=\psi$ כאשר ψ היא נוסחה בוליאנית עם כמתים כך ש:

$$\begin{aligned} x\in L &\quad\Rightarrow\quad \psi=1 \quad , \\ x\notin L &\quad\Rightarrow\quad \psi=0 \quad . \end{aligned}$$

כדי להגדיר את הפונקצית הרדוקציה, קודם נגדיר את הנוסחאה הבוליאנית הבאה. יהיו c,c' שתי קונפיגורציות של המכונה M ויהי t מספר שלם חיובי.

$$\phi\left(c,c',t\right) \triangleq \begin{cases} 1 & : \text{אפשר לעבור מ- } c \text{ ל- } c' \text{ עם } t \text{ צעדים לכל היותר} \\ 0 & : \text{אחרת} \end{cases} .$$

הצורה המפורשת של הנוסחה $\phi\left(c,c',t\right)$ עצמה בנויה רקורסיבי באופן הבא.

המקרה $t>1$

לכל $t>1$ ולכל קונפיגורציות c,c':

$$\phi\left(c,c',t\right)=\exists w\ \forall x,y \quad (x=c\wedge y=w)\vee(x=w\wedge y=c') \quad\Rightarrow\quad \phi\left(x,y,\left\lfloor\frac{t}{2}\right\rfloor\right) .$$

כאן w מסמן קונפיגורציה ביניים בין הקונפיגורציה c לקונפיגורציה c'. היחס הרקורסיבי הזה של $\phi\left(c,c',t\right)$ אומר ש- M יכול לעבור מ- c ל- c' ב- t צעדים לכל היותר אם קיימת קונפיגורציה ביניים w כך ש- M יכול לעבור מ- c ל- w בכל היותר $\lfloor\frac{t}{2}\rfloor$ צעדים, ואחר כך M יכול לעבור מ- w ל- c' בכל היותר $\lfloor\frac{t}{2}\rfloor$ צעדים. הנוסחאות $\phi\left(c,w,\lfloor\frac{t}{2}\rfloor\right)$ ו- $\phi\left(w,c',\lfloor\frac{t}{2}\rfloor\right)$ הן נבנות בשלב הבא של הרקורסיה באותו אופן. התתליך הזה ממשיך עד שנגיע לנוסחה $\phi(x,y,t)$ שבה $t=0$ או $t=1$.

המקרה $t=1$

$$\phi(c,c',t=1)=\begin{cases} 1 & : \text{אם } M \text{ יכול לעבור מ- } c \text{ ל- } c' \text{ בצעד חישוב בודד} \\ 0 & : \text{אחרת} \end{cases}$$

הצורה המפורשת של $\phi(c,c',t=1)$ היא כדלקמן. תהיינה c,c' שתי קונפיגורציות כלשהן. נגדיר טבלת הקונפיגורציות:

#	c_1	c_2	$\ldots$	c_{N-1}	c_N	#
#	c'_1	c'_2	$\ldots$	c'_{N-1}	c'_N	#

נניח כי $M=(Q,\Sigma,\Gamma,\delta,q_0,q_{\text{acc}},q_{\text{rej}})$ ותהי $C=Q\cup\Gamma\cup\{\#\}$. עבור כל תא ה- (i,j) של הטבלת הקונפיגורציות נגדיר משתנה בוליאני $x_{i,j,s}$ לכל $1\leqslant i\leqslant 2$ ולכל $1\leqslant j\leqslant N$ כך ש:

$$x_{ijs}=\begin{cases} 1 & : \text{הסימן } s\in C \text{ כתוב בתא } ij \\ 0 & : \text{הסימן } s\in C \text{ לא כתוב בתא } ij \end{cases}$$

למשל אם בתא $(2,5)$ כתוה a אז $x_{2,5,\text{a}}=1$ בעוד $x_{2,5,\text{b}}=0$. הנוסחה $\phi\left(c,c',t=1\right)$ מוגדרת:

$$\phi\left(c,c',t=1\right)=\phi_{\text{cell}}\wedge\phi_c\wedge\phi_{\text{move}}\wedge\phi_{c'} .$$

- $\phi_{\text{cell}}=1$ אם ורק אם יש רק סימן אחד בדיוק שכתוב בכל תא ו- 0 אחרת. בפרט:

$$\phi_{\text{cell}}=\bigwedge_{\substack{1\leqslant j\leqslant 2\\ 1\leqslant j\leqslant n^k}}\left[\left(\bigvee_{s\in C}x_{i,j,s}\right)\wedge\left(\bigwedge_{\substack{s,t\in C\\ s\neq t}}\left(\overline{x_{i,j,s}}\vee\overline{x_{i,j,t}}\right)\right)\right] \tag{13.1}$$

* $\bigvee_{s\in C} x_{i,j,s}$ מבטיח שלכל תא לפחות משתנה אחד דולק.

* $\bigwedge_{\substack{s,t\in C\\ s\neq t}} \left(\overline{x_{i,j,s}} \vee \overline{x_{i,j,t}}\right)$ מבטיח שלכל תא משתנה אחד לכל היותר דולק.

לפיכך ϕ_{cell} מסופקת אם ורק אם יש בדיוק סימן אחד כתוב בכל תא. בניגוד זה אם שני משתנים של אותו תא דולקים בגלל שהסימן s וגם הסימן t כתובים בו זמנית בתא ה- ij אז $x_{i,j,s}=1$ וגם $x_{i,j,t}=1$ ואז ϕ_{cell} תהיה שווה ל- 0.

- הנוסחה ϕ_c ($\phi_{c'}$) קובעת כי המשתנים הספציפים של הקונפיגורציות c (c') הם דולקים:

$$\begin{aligned}
\phi_c =& x_{1,0,\#} \wedge x_{1,1,c_1} \wedge x_{1,2,c_2} \wedge \ldots \wedge x_{1,N-1,c_{N-1}} \wedge x_{1,N,c_N} \wedge x_{1,N+1,\#}\ ,\\
\phi_{c'} =& x_{2,0,\#} \wedge x_{2,1,c'_1} \wedge x_{2,2,c'_2} \wedge \ldots \wedge x_{2,N-1,c'_{N-1}} \wedge x_{2,N,c'_N} \wedge x_{2,N+1,\#}\ .
\end{aligned} \tag{13.2}$$

- $\phi_{\text{move}}=1$ אם אפשר להגיע מ- c ל- c' על ידי תזוזה חוקית אחת של M ו- 0 אחרת. תזוזה חוקית בין כל שתי קונפיגורציות מוגדרת על פי סהפונקצית המעברים של M. במונחי הטבלה, אפשר להגדיר תזוזה חוקית בין השתי שורות על ידי תת-טבלה מסדר 2×3 שמכילה 3 עמודות.

נקרא תת-טבלה כזה "חלון". אלה הם דוגמאות של חלונות חוקיים:

a	q_1	b
q_2	a	c

a	q_1	b
a	a	q_2

a	a	q_1
a	a	b

#	b	a
#	b	a

a	b	a
a	b	q_2

b	b	b
c	b	b

בעוד אלה הם דוגמאות של חלונות לא חוקיים:

a	b	a
a	a	a

a	q_1	b
q_1	a	a

b	q_1	b
q_2	b	q_2

$\phi_{\text{move}}=1$ אם כל חלון של הטבלה חוקי ו- 0 אם לא. בפרט, כל חלון מכיל 6 תאים. החלון המכיל את ה- 3 עמודות i, $i+1$ ו- $i+2$ תקרא החלון-i. לכן:

$$\begin{aligned}
\phi_{\text{move}} &= \bigwedge_{1\leqslant i\leqslant N-2} \left(\text{חלון- } i \text{ חוקי}\right)\\
&= \bigwedge_{1\leqslant i\leqslant N-2} \left(x_{1,i,c_i} \wedge x_{1,i+1,c_{i+1}} \wedge x_{1,i+2,c_{i+2}} \wedge x_{2,i,c'_i} \wedge x_{2,i+1,c'_{i+1}} \wedge x_{2,i+2,c'_{i+2}}\right)
\end{aligned} \tag{13.3}$$

המקרה $t=0$

אם $t=0$ אז

$$\phi\left(c,c',t=0\right)=\begin{cases}1 & : c=c'\\ 0 & : c\neq c'\end{cases}\ .$$

כעת נחזור להגדרה עצמה של הרדוקציה פולינומיאלית. תהי L שפה כריעה ע"י מכונת טיורינג M במקום $O\left(n^k\right)$, אז קיימת רדוקציה $f(x)=\psi$ כך ש:

$$\psi=\phi\left(c_{\text{start}},c_{\text{acc}},h\right)$$

כאשר $h=2^{df(n)}$ ו- $d>0$ מספר ממשי חיובי הנבחר כך של- M יש לכל היותר $2^{df(n)}$ קונפיגורציות בהינתן קלט של אורך n, ו- $f(n)=n^k$.

הוכחת הנכונות

אם $x \in L$

$\Leftarrow$ M תקבל x

$\Leftarrow$ M יכולה לעבור מ- c_{start} ל- c_{acc} במספר צעדים פחות מ- או שווה ל- h

$\Leftarrow$ $\psi = 1$

$\Leftarrow$ $\psi \in TQBF$.

אם $x \notin L$

$\Leftarrow$ M תדחה x

$\Leftarrow$ לא קיימים צעדים של M מ- c_{start} ל- c_{acc}.

$\Leftarrow$ $\psi = 0$

$\Leftarrow$ $\psi \notin TQBF$.

סיבוכיות זמן של הרדוקציה

בפונקצית הרדוקציה מחשבת $f(x) = \psi = \phi\left(c_{\text{start}}, c_{\text{acc}}, h\right)$ כאשר $h = 2^{df(n)}$ ו- $f(n) = n^k$, באופן רקורסיבי. לרקורסיה יש מספר שלבים השווה ל:

$$\log_2\left(2^{df(n)}\right) = df(n) = dn^k \ .$$

- הנוסחה (13.1) של ϕ_{cell} מכילה n^{2k} נוסחאות עם 3 ליטרלים. לכן הסיבוכיות זמן של ϕ_{cell} היא:
$$O\left(n^{2k}\right) \ .$$
- הנוסחאות ϕ_c ו- $\phi_{c'}$ במשואה (13.2) מכילות בדיוק n^k ליטרלים. לכן הסיבוכיות זמן של ϕ_c ושל $\phi_{c'}$ הן:
$$O\left(n^{k}\right) \ .$$
- הנוסחה (13.3) של ϕ_{move} מכילה n^{2k} נוסחאות עם 6 ליטרלים. לכן הסיבוכיות זמן של ϕ_{move} היא
$$O\left(n^{2k}\right) \ .$$

לפיכך בסה"כ הרדוקציה ניתנת לחישוב בזמן $O\left(n^{2k}\right)$ ולכן $f \in TIME\left(n^{2k}\right)$ ולכן הרדוקציה היא רדוקציה פולינומיאלית.

■

13.6 המחלקה L

13.7 המחלקה NL

13.8 שלמות ב- NL

13.9 שיוויון NL ו- coNL