

שיעור 11

NP שלמות

11.1 המחלקות NPH ו- NP

הגדרה 11.1 NP-hard

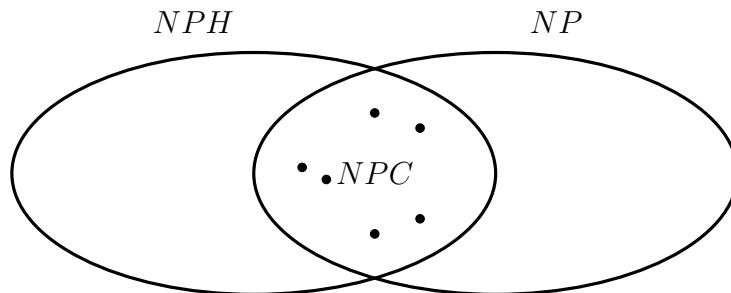
בשפה B נקראת NP קשה אם לכל בעיה $A \in NP$ קיימת רדוקציה . $A \leqslant_P B$

הגדרה 11.2 NP-complete

בשפה B נקראת NP שלמה אם

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

(2) לכל בעיה $A \in NP$ קיימת רדוקציה . $A \leqslant_p B$



משפט 11.1

אם B בעיה NP שלמה וגם $P = NP$ אז $A \leqslant_P B$ $\forall A \in NP$

הוכחה:

- הוכחנו כבר שה- . $P \subseteq NP$
- נוכח כי $NP \subseteq P$

לכל בעיה $A \subseteq NP$ קיימת רדוקציה $A \leqslant_P B$ ומכיון ש- $B \in P$, מושפט הרדוקציה מתקיים . $A \leqslant_P B$

מסקנה 11.1

אם $\bar{A} \leqslant_P \bar{B}$ אז $A \leqslant_P B$

משפט 11.2

אם $A \leqslant_p C$ ו- $B \leqslant_p C$ וגם $A \leqslant_p B$ אז

הוכחה:

משפט 11.3

תהי B בעיה NP -שלמה. אז לכל בעיה $C \in NP$, אם $B \leq_p C$ אז גם C היא NP -שלמה.

הוכחה: מכיוון ש- B היא NP -שלמה, לכל בעיה $A \in NP$ קיימת רדוקציה $B \leq_p A$. מכיוון ש- מהטרנסיטיביות מתקיים $A \leq_p C \leq_p B$ לכל בעיה $A \in NP$ ולכן C היא NP -שלמה.

11.2 בעית הספיקות**הגדרה 11.3**

נוסחת ϕ היא נוסחה בוליאנית מעל n משתנים x_1, x_2, \dots, x_n המכילה m פסוקיות כאשר כל פסוקית מכילה אוסף של ליטרלים ($x_i \setminus \bar{x}_i$) המוחברים ע"י OR (\vee) בוליאני והפסוקיות מחוברים ע"י AND (\wedge) בוליאני.

לדוגמה

$$\phi = \left(x_1 \vee \bar{x}_2 \vee x_4 \vee \bar{x}_7 \right) \wedge \left(x_3 \vee x_5 \vee \bar{x}_8 \right) \wedge \dots$$

הגדרה 11.4 נוסחת CNF ספיקה

נוסחת ϕ היא ספיקה אם קיימת השמה למשתנים x_1, x_2, \dots, x_n ע"י $T \setminus F$ כך ש- ϕ מקבלת ערך T , כלומר בכל פסוקית ישנו לפחות ליטרל אחד שקיבל ערך T .

11.3 בעית SAT**הגדרה 11.5 בעית SAT**

קלט: נוסחת ϕ , פלט: האם ϕ ספיקה?

$$SAT = \{ \langle \phi \rangle \mid \text{נוסחת CNF ספיקה } \phi \}$$

משפט 11.4

$$SAT \in NP$$

הוכחה: בניית אלגוריתם אimotoת V עבור SAT .

: על קלט $V = \langle \langle \phi \rangle, y \rangle$

1) בודק האם y היא השמה למשתנים x_1, x_2, \dots, x_n .

- אם לא $\leq 3CNF \Leftrightarrow$ דוחה.

(2) בודק האם השמה זו מספקת את ϕ .

- אם כן \Leftrightarrow מקבל.
- אם לא \Leftrightarrow דוחה.

■

11.4 משפט קוק לוין

משפט 11.5 (1973) משפט קוק לוין

הבעית SAT היא NP - שלמה.

רעיון ההוכחה:

$A \leq_p SAT, A \in NP$ לכל $w \in \Sigma^*$

$$w \in A \Leftrightarrow f(w) \in SAT,$$

$$\text{כאן } f(w) = \langle \phi_w \rangle$$

מסקנה 11.2

$$P = NP \Leftrightarrow SAT \in P.$$

11.5 גרסאות של $kSAT$

ישנן לכל היותר k ליטרלים בכל פסוקית:

$$.1SAT \in P \bullet$$

$$\phi = x_1 \wedge \bar{x}_2 \wedge x_3 \wedge \dots$$

$$.2SAT \in P \bullet$$

$$\phi = (x_1 \vee \bar{x}_2) \wedge (x_2 \vee x_4) \wedge \dots$$

$$.3SAT \in P \bullet \text{ היא } NP - \text{ שלמה.}$$

3SAT בעית 11.6

הגדרה 11.6 בעית 3SAT

קלט: נוסחת $\phi, 3CNF$

פלט: האם ϕ ספיקת?

$$3SAT = \{\langle \phi \rangle \mid \phi \text{ נוסחת } 3CNF \text{ ספיקת}\}$$

משפט 11.6 $3SAT \in NP$ שלמה. $3SAT \in NP$ שלמה.

הוכחה:

ישקיימים את השני תנאים הבאים:

(1) $3SAT \in NP$

ניתן לבנות אלגוריתם אimoto עבור $SAT \in NP$ דומה לאלגוריתם האimoto עבור SAT שבנו בהוכחה של המשפט קוק-ליין 11.5 לעלה.

(2) $3SAT \in NP$ קשה ע"י רדוקציה

$SAT \leq_p 3SAT$.

ואז בגלל ש- $SAT \in NP$ שלמה (לפי משפט קוק-ליין 11.5) ומכיון ש- $3SAT \in NP$ אז לפי משפט האסימפטוטית 11.2 גם $3SAT \in NP$ - שלמה.

קיים פונקציה הרדוקציה $SAT \leq_p 3SAT$

כעת נבנה את פונקציה הרדוקציה מ- SAT ל- $3SAT$.ראשית נזכיר כי כל נוסחה בוליאנית ϕ ניתנת לרשום בצורה CNF בזמן פולינומיAli.

בහינתנו נוסחת CNF , ϕ (הקלט של SAT) נבנה בזמן פולינומיAli נוסחת ϕ' (הקלט של $3SAT$) ואז נוכחים מתקיים

$\langle\phi'\rangle \in 3SAT \Leftrightarrow \langle\phi\rangle \in SAT$.

לכל פסוקית C ב- ϕ המכילה יותר מ- 3 ליטרלים, ניצור אוסף C' ב- ϕ' של פסוקיות כך שכל פסוקית ב- C' תכיל 3 ליטרלים. למשל בהינתן הפסוקית C הבאה של ϕ :

$$C = x_1 \vee x_2 \vee x_3 \vee \bar{x}_4 \vee \bar{x}_5$$

ניצור את הפסוקית C' הבאה ב- ϕ' :

$$C' = (x_1 \vee x_2 \vee y_1) \wedge (\bar{y}_1 \vee x_3 \vee y_2) \wedge (\bar{y}_2 \vee \bar{x}_4 \vee \bar{x}_5)$$
.

באופן כללי, לכל פסוקית a_k המכיל $k > 3$ ליטרלים, ניצור אוסף C' של פסוקיות שבו כל פסוקית מכילה 3 ליטרלים, ע"י הוספה 3 משתנים y_1, y_2, \dots, y_{k-3} :

$$C' = (a_1 \vee a_2 \vee y_1) \wedge (\bar{y}_1 \vee a_3 \vee y_2) \wedge \dots \wedge (\bar{y}_{i-2} \vee a_i \vee y_{i-1}) \wedge \dots \wedge (\bar{y}_{k-3} \vee a_{k-1} \vee a_k)$$
.

בפרט, עבור כל פסוקית $(a_1 \vee a_2 \vee \dots \vee a_k)$ נניח $a_i = 1$ הוא הלiteral הראשון שווה ל- 1. אז

• נשים $y_j = 1$ לכל $1 \leq j \leq i-2$

• ונסים $y_j = 0$ לכל $i-1 \leq j \leq k-3$

סימנו להגדר את הפונקציה הרדוקציה.

כעת נוכחים כי הפונקציה הזאת מקיימת את התנאי ההכרחי

$$\langle\phi'\rangle \in 3SAT \Leftrightarrow \langle\phi\rangle \in SAT$$
.

כיון \Leftarrow :

נניח כי $\langle\phi\rangle \in SAT$ ותהי X השמה המספקת את ϕ .
nocich שקיימת השמה X' מתאימה המספקת את ϕ' .

- בכל פסוקית C של ϕ , עבור הליטרלים a_1, a_2, \dots, a_k ניתן אותם ערכים כמו ב- X .
- מכיוון ש- X מספקת את ϕ , בכל פסוקית $C = (a_1 \vee a_2 \vee \dots \vee a_k)$ יש לפחות ליטרל אחד שקיבל ערך 1. נניח $a_i = 1$. אז על פי ההגדרה של פונקציית הרודוקציה:

$$\begin{aligned} * \text{ נשים } 1 = y_j \text{ לכל } 2 \leq j \leq i-1 \\ * \text{ ונשים } 0 = y_j \text{ לכל } i-1 \leq j \leq k-3 \end{aligned}$$

באופן זהה אנחנו נזכיר אוסף C' של פבוקיות עם המבנה הבא:

$$\begin{aligned} & \left(a_1 \vee a_2 \vee y_1 \right) \wedge \left(\bar{y}_1 \vee a_2 \vee y_2 \right) \wedge \dots \wedge \left(\bar{y}_{i-3} \vee a_{i-1} \vee y_{i-2} \right) \wedge \left(\bar{y}_{i-2} \vee a_i \vee y_{i-1} \right) \wedge \left(\bar{y}_{i-1} \vee a_{i+1} \vee y_i \right) \\ & \wedge \dots \wedge \left(\bar{y}_{k-3} \vee a_{k-1} \vee a_k \right) \end{aligned}$$

ולכן השמה זו מספקת את C' ולכן $\langle \phi' \rangle \in 3SAT$

כיוון: \Rightarrow

נניח כי $\langle \phi' \rangle \in 3SAT$ ותהי X' השמה המספקת את ϕ' .
נוכיח שקיימות השמה X המספקת את ϕ .

נסתכל על פסוקית $C = (a_1 \vee a_2 \vee \dots \vee a_k)$ נניח בשלילה שלא קיימת השמה X המספקת את C . אז בהכרח

$$a_1 = a_2 = \dots = a_k = 0$$

לפי זה, באוסף פסוקיות C' שנתקבל על פי ההגדרה של פונקציית הרודוקציה, $y_j = 1$ לכל $1 \leq j \leq k-3$ $y_j = 0$ לכל $y_1 = y_2 = \dots = y_{k-3} = 0$. כלומר מתקיים

$$C' = \left(a_1 \vee a_2 \vee y_1 \right) \wedge \left(\bar{y}_1 \vee a_3 \vee y_2 \right) \wedge \dots \wedge \left(\bar{y}_{i-2} \vee a_i \vee y_{i-1} \right) \wedge \dots \wedge \left(\bar{y}_{k-3} \vee a_{k-1} \vee a_k \right)$$

הפסוקית האחרונה $\left(\bar{y}_{k-3} \vee a_{k-1} \vee a_k \right)$ אינה מסופקת.
לכן C' אינה מסופקת, בסתיו לכך X' מספקת את ϕ' .

ולכן $\langle \phi \rangle \in SAT$

$SAT \leq 3SAT$

כעת נוכיח כי הרודוקציה זו היא זמן פולינומיAli.

סיבוכיות

הчисוב של הפונקציה מתבצע בזמן פולינומיAli. ספציפי, אם האורך של הנוסחה ϕ הוא $| \phi | = n$ אז הרודוקציה היא $O(n)$.



11.7 הוכחת משפט קוק לוין*

משפט 11.7 משפט קוק לוין

הבעית SAT היא NP - שלמה.

הוכחה:

חישפה מלאה: ההוכחה הבאה מتبוססת על ההוכחה שנותונה בספר של Sipser.

על פי הגדירה 11.2 יש להוכיח שני התנאים הבאים מתקיימים:

תנאי 1: $SAT \in NP$

תנאי 2: $A \in NP \leq_p SAT$

ראשית נוכיח כי $SAT \in NP$:

כדי להוכיח כי SAT שייכת ל- NP , נוכיח כי אישור המרכיב מהשמה מספקת עבור נוסחת קלט ϕ ניתן לאיומות בזמן פולינומייאלי.

נניח כי $n = |\phi|$. כאמור ב- ϕ מופיעים n ליטרלים. "א" השמה כלשהי דורשת n משתני בוליאניים לכל היוטר.

- אלגוריתם האimotoות מחליף כל משתנה בנוסחה בערך המתאים לו על פי ההשמה. השלב זהה הוא $(n)O$.

- אחר כך האלגוריתם מחשב את ערכו של הביטוי:

* נניח כי הנוסחה ϕ מכילה k דורות של סוגרים בתוך סוגרים.

* החישוב מתחילה עם החישובים של הביטויים בתוך הסוגרים הכל בפנים.

* יש n סוגרים הći-בפנים לכל היוטר, וכל אחד של סוגרים אלה מכיל n ליטרלים לכל היוטר. לכן החישוב זהה הוא $(n^2)O$.

* יש k דורות של סוגרים לכן החישוב כולל הוא $(kn^2)O$.

- בסה"כ הסיבוכיות זמן הריצה היא

$$O(n) + O(kn^2) = O(n^2)$$

לפיכך אישור של השמה כלשהי מatabase בזמן פולינומייאלי.

- אם ערכו של הביטוי הוא 1 הנוסחה ספיקה.

הוכחנו כי $SAT \in NP$. עכשו נוכיח כי $SAT \leq_p A$.

תהי N מ"ט אי-דטרמיניסטי זמני-פולינומייאלית שמכריעה שפה A כלשהי בזמן $O(n^k)$ עבור k טבעי. התרשים למטה מראה טבלה של קונפיגורציות של N . ברשימה הבאה רשומות ההגדרות של הטבלה:

- כל שורה מראה את תוכן הסרט בשלב מסוים של מסלול אחד של N .

- בשורה הראשונה יש את הקונפיגורציה ההתחלתית.

- אנחנו מניחים כי האורך של המילה, כאמור אורץ הקלט הוא n .

הסימנים w_n, \dots, w_1 מסמנים את התווים של הקלט.

- בתא הראשון בכל שורה יש #, ולאחר מכן רשומה הקונפיגורציה של N . בסוף הקונפיגורציה בכל שורה יש #.
 - אחרי #- בקצתה הימין של המילה, בכל תא ישתו רוח עד הסוף של השורה. התוויי רוח לפני המשבצת הראשונה של הקונפיגורציה לא מופיעים בטבלה.
 - האורך של כל שורה הוא בדיקת n תאים.
 - בטבלה יש בדיקת n שורות לסייע בהאה:
 - המכונת טיריניג מבצעת n צעדים לכל היותר.
 - בכל צעד המ"ט עוברת לקונפיגורציה חדשה.
 - בכל שורה רשומה קונפיגורציה.
 - בסה"כ יש n שורות עבור ה- n קונפיגורציות שונות האפשריות.

אנו אומרים כי טבלה כלשהי היא **טבלה המבלטת** אם באחת השורות יש קו ניגורציה אשר N מקבלת אותה.

בעזרת הטבלה נתאר את הרדוקציה זמו-פולינומיאלית f משפה A כלשוי L - SAT .

הפונקציה הרדוקציה f מקבלת קלט w ומחזירה נוסחה $\phi = f(w)$, אשר לפי ההגדרה של פונקציית הרדוקציה, עומדת בתנאי הבא:

$$w \in A \quad \Leftrightarrow \quad f(w) \in SAT .$$

יהיו Q קבוצת המצבים ו- Γ האלפיבית של הסרט של N . נגדיר:

$$C = Q \cup \Gamma \cup \{\#\}.$$

נסמן ב- s איבר כלשהו של C .

עבור כל תא (i, j) של הטבלת הקונפיגורציות נגידר משתנה בוליאני $x_{i,j,s}$ לכל המשתנה x_{ijs} מוגדר על פי התנאי

$$x_{ijs} = 1$$

אם בתא ה- j,i של הטבלה יש $s \in C$. למשל, אם בתא ה- $(2,5)$ של הטבלה מופיע התו a אז

$$x_{2,5,a} = 1$$

בעוד

$$x_{2,5,b} = 0 .$$

במובן זהה, התכנים של כל התאים של הטבלה מסומנים על ידי המשתנים של ϕ .

עכשו נבנה נוסחה ϕ על סמך התנאי שהשמה מספקת של ϕ תהיה מתאימה לטבלה המתקבלת של N . נגיד:

$$\phi = \phi_{\text{cell}} \wedge \phi_{\text{start}} \wedge \phi_{\text{move}} \wedge \phi_{\text{acc}} . \quad (11.1)$$

נסביר את כל הנוסחאות ϕ_{acc} , ϕ_{move} , ϕ_{start} , ϕ_{cell} אחד אחד למטה.

• הנוסחה ϕ_{cell}

כפי שמצוין לעיל, אם המשתנה $x_{i,j,s}$ "דולק", כלומר אם $x_{i,j,s} = 1$, זאת אומרת שיש סימן s בתא ה- i,j של הטבלה. אנחנו רוצים להבטיח שהשמה כלשהי בנוסחה אשר מתאימה לקונפיגורציה של הטבלה, מדliquה בדיק משותה אחד לכל תא של הטבלה. למטרה זו נגיד ϕ_{cell} כך:

$$\phi_{\text{cell}} = \bigwedge_{1 \leq i, j \leq n^k} \left[\left(\bigvee_{s \in C} x_{i,j,s} \right) \wedge \left(\bigwedge_{\substack{s,t \in C \\ s \neq t}} (\bar{x}_{i,j,s} \vee \bar{x}_{i,j,t}) \right) \right] \quad (11.2)$$

* האיבר הראשון בסוגרים מרובעים, מבטיח שלכל תא של הטבלה, לפחות משתנה אחד דולק.

* האיבר השני מבטיח שעבור כל תא של הטבלה, משתנה אחד לכל היוטר Dolk.

לפיכך כל השמה מספקת עומדת בתנאי שיהיה בדיק סימן אחד, s , בכל תא של הטבלה.

• הנוסחה ϕ_{start}

נוסחה ϕ_{start} מבטיחה ששורה הראשונה של הטבלה היא הקונפיגורציה ההתחלתית של N על הקלט w :

$$\begin{aligned} \phi_{\text{start}} = & x_{1,1,\#} \wedge x_{1,2,q_0} \wedge x_{1,3,w_1} \wedge x_{1,4,w_2} \\ & \wedge \dots \wedge \\ & x_{1,n+2,w_n} \wedge x_{1,n+3,_} \wedge \dots \wedge x_{1,n^k-1,_} \wedge x_{1,n^k,\#} \end{aligned} \quad (11.3)$$

• הנוסחה ϕ_{acc}

הנוסחה ϕ_{acc} מבטיחה שקיים טבלה קונפיגורציה אשר המ"ט N מקבל אותה. בפרט ϕ_{acc} מבטיחה שהסימן q_{acc} מופיע בתא אחד של הטבלה דרך התנאי שלפחות אחד המשתנים $x_{i,j,q_{\text{acc}}}$ דולק:

$$\phi_{\text{acc}} = \bigvee_{1 \leq i, j \leq n^k} x_{i,j,q_{\text{acc}}} \quad (11.4)$$

• הנוסחה ϕ_{move}

הנוסחה ϕ_{move} מבטיחה שכל שורה של הטבלה היא "שורה חוקית".
כלומר בכל שורה, הקונפיגורציה היא כך שאפשר הגיע אליה על ידי תזוזה חוקית של N מהקונפיגורציה הקודמת שMOVEDה בשורה אחרת מעלה.

תזוזה חוקית בין כל שתי קונפיגורציות נקבעת על ידי הפונקציה המעברים של המ"ט N .
בשפה פורמלית, אם c_i הקונפיגורציה של שורה i , ו- c_{i+1} הקונפיגורציה של השורה $i+1$ אחת למטה, אז
 ϕ_{move} מבטיחה כי לכל $1 \leq i \leq n^k$ מתקיים

$$c_i \vdash_N c_{i+1} .$$

במנוחי הטבלה, אפשר להגיד תזוזה חוקית בין כל שתי שורות על ידי תת-טבלה מסדר 3×2 שמכילה 3 תאים מתאימים של שתי שורות שכנות.
מכאן ואילך אנחנו נקרא לתת-טבלה כזאת "חלון".

למטה יש דוגמאות של חלונות חוקיים:

<table border="1"><tr><td>a</td><td>q_1</td><td>b</td></tr><tr><td>q_2</td><td>a</td><td>c</td></tr></table>	a	q_1	b	q_2	a	c	<table border="1"><tr><td>a</td><td>q_1</td><td>b</td></tr><tr><td>a</td><td>a</td><td>q_2</td></tr></table>	a	q_1	b	a	a	q_2	<table border="1"><tr><td>a</td><td>a</td><td>q_1</td></tr><tr><td>a</td><td>a</td><td>b</td></tr></table>	a	a	q_1	a	a	b
a	q_1	b																		
q_2	a	c																		
a	q_1	b																		
a	a	q_2																		
a	a	q_1																		
a	a	b																		
<table border="1"><tr><td>#</td><td>b</td><td>a</td></tr><tr><td>#</td><td>b</td><td>a</td></tr></table>	#	b	a	#	b	a	<table border="1"><tr><td>a</td><td>b</td><td>a</td></tr><tr><td>a</td><td>b</td><td>q_2</td></tr></table>	a	b	a	a	b	q_2	<table border="1"><tr><td>b</td><td>b</td><td>b</td></tr><tr><td>c</td><td>b</td><td>b</td></tr></table>	b	b	b	c	b	b
#	b	a																		
#	b	a																		
a	b	a																		
a	b	q_2																		
b	b	b																		
c	b	b																		

החלונות האלה למטה הם דוגמאות לחלונות לא חוקיים:

<table border="1"><tr><td>a</td><td>b</td><td>a</td></tr><tr><td>a</td><td>a</td><td>a</td></tr></table>	a	b	a	a	a	a	<table border="1"><tr><td>a</td><td>q_1</td><td>b</td></tr><tr><td>q_1</td><td>a</td><td>a</td></tr></table>	a	q_1	b	q_1	a	a	<table border="1"><tr><td>b</td><td>q_1</td><td>b</td></tr><tr><td>q_2</td><td>b</td><td>q_2</td></tr></table>	b	q_1	b	q_2	b	q_2
a	b	a																		
a	a	a																		
a	q_1	b																		
q_1	a	a																		
b	q_1	b																		
q_2	b	q_2																		

הנוסחה ϕ_{move} קובעת שכל חלון של הטבלה חוקי. בפרט, כל חלון מכיל 6 תאים. לכן ϕ_{move} קובעת שהתכנים של ה-6 תאים של כל חלון מהוות חלון חוקי. ז"א

$$\phi_{\text{move}} = \bigwedge_{\substack{1 \leq i \leq n^k \\ 1 \leq j \leq n^k}} (\text{חלון } i, j \text{ חוקי}) \quad (11.5)$$

אנטונו מציבים בטקסט "חלון ה- i, j חוקי" את נוסחה הבאה, כאשר a_6, \dots, a_1 מסמנים את התכנים של ה-6 תאים של כל חלון:

$$\bigvee_{\substack{\{a_1, a_2, a_3, a_4, a_5, a_6\} \\ \text{חלון חוקי}}} (x_{i,j-1,a_1} \wedge x_{i,j,a_2} \wedge x_{i,j+1,a_3} \wedge x_{i+1,j-1,a_4} \wedge x_{i+1,j,a_5} \wedge x_{i+1,j+1,a_6}) \quad (11.6)$$

עד כה הוכחנו שקיים רדוקציה מכל שפה $.SAT \rightarrow A \in NP$ כעת נוכיח כי הרדוקציה זו חישובית בזמן-פולינומייאלי.

הטבלה של N היא מסדר $n^k \times n$ ולכן היא מכילה n^{2k} תאים.

נחשב את הסיבוכיות של כל הנוסחים $\phi_{\text{move}}, \phi_{\text{acc}}, \phi_{\text{start}}, \phi_{\text{cell}}$.

• הנוסחה ϕ_{cell}

הנוסחה (11.2) של ϕ מכילה n^{2k} נוסחאות עם 3 ליטרלים. לכן

$$\phi_{\text{cell}} = O(n^{2k}) .$$

• הנוסחה ϕ_{start}

הנוסחה (11.3) של ϕ מכילה בדיק n^k ליטרלים. לכן

$$\phi_{\text{start}} = O(n^k) .$$

• הנוסחה ϕ_{acc}

הנוסחה (11.4) של ϕ מכילה בדיק n^k ליטרלים. לכן

$$\phi_{\text{acc}} = O(n^k) .$$

• הנוסחה ϕ_{move}

הנוסחה (11.6,11.5) של ϕ מכילה n^{2k} נוסחאות עם 6 ליטרלים. לכן

$$\phi_{\text{move}} = O(n^{2k}) .$$

לכן בסה"כ

$$\phi = O(n^{2k}) + O(n^k) + O(n^k) + O(n^{2k}) = O(n^{2k}) .$$

לפיכך קיימת רדוקציה חישובית בזמן פולינומייאלי מכל שפה L - SAT $\in NP$

