משפט קוק-לוין

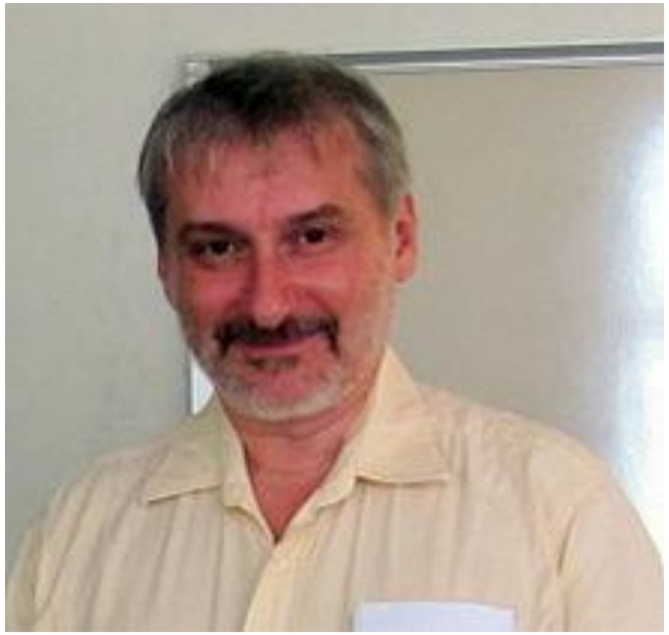
סטיבן קוק

- סטיבן קוק הוא מדען מחשב ומתמטיקאי אמריקאי-קנדי. פרופסור באוניברסיטת טורונטו .נחשב לאחד האבות המייסדים של תורת הסיבוכיות.
- במאמרו מ-1971 המורכבות של הוכחת משפט ,קוק טבע פורמלית את המושגים של רדוקציה פולינומית ובעיות NP-שלמות, והוכיח את קיומה של בעיה-NP שלמה על ידי שהראה שבעיית SAT, היא NP-שלמה. המאמר גם מציג פורמלית את שאלת P=NP.
- זכה בפרס טיורינג על תרומתו לתורת הסיבוכיות.

(מתוך ויקיפדיה)

ליאוניד לוין

- A Soviet-American mathematician and computer scientist.
- Known for his work in randomness in computing, algorithmic complexity and intractability, average-case complexity, foundations of mathematics and computer science, algorithmic theory, theory of computations, and information theory.
- He and Stephan Cook independently discovered the existence of NP-Complete problems. Levin was awarded the Knuth Prize in 2012 for his discovery of NP-completeness and the development of average case complexity .
- He is described in a chapter of the book *Out of Their Minds: The Lives and Discoveries of 15 Great Computer Scientists*.
- (Wikipedia)



SAT The Satisfiability Problem

- נגדיר α נוסחה בוליאנית בCNF
- $\alpha = (x_1 \vee x_2) \wedge (x'_2 \vee x_3 \vee x'_4)$
- נאמר ש α ניתנת לסיפוק, אם קיימת השמה למשתנים

בα שהצבתה בנוסחה נותנת TRUE.

SAT ={α | α נוסחה בוליאנית שניתנת לסיפוק }

SAT The Satisfiability Problem

שימושים לSAT:

- בינה מלאכותית
- Circuit Design
- Automatic theorem Prooving

SAT היא בעיה בNPC

שלב א בהוכחה, $\in NP$ SAT

SAT $=\{ \alpha \mid \alpha$ ניתנת לסיפוק $\}$

יש להראות שקימת מ"ט ל"ד N <u>שפועלת בזמן פולינומי בגודל הקלט</u>,

שמקבלת את SAT.

N יכולה לעצור <u>בזמן פולינומי</u>

<u>בגודל הקלט</u> אם $\alpha \in$ SAT

↓

N יכולה לעצור <u>בזמן פולינומי בגודל הקלט</u>

אם ל α קיימת השמה שנותנת TRUE

N לא תעצור <u>בזמן פולינומי</u>

<u>בגודל הקלט</u> אם $\alpha \notin$ SAT .

↓

N לא תעצור <u>בזמן פולינומי בגודל הקלט</u>

אם כל השמה ל α נותנת FALSE

שלב א בהוכחה, $SAT \in NP$

- הוכחה : נשתמש במ"ט ל"ד N – שעבור קלט של נוסחה α,

1. תנחש ערכים לכל המשתנים שבנוסחה, (במטרה לגרום לנוסחה להיות TRUE).
2. N תציב את הערכים של הניחוש בנוסחה ותבדוק למה הנוסחה שווה.
3. אם הנוסחה היא TRUE המכונה N תעצור.
4. אם הנוסחה היא FALSE, המכונה N לא תעצור (כי אם היתה

אפשרות לערכים שיגרמו לספיקות הנוסחה היינו יכולים לנחש אותם).

שלב א בהוכחה, $\in NP$ SAT זמן

- בדיקת זמן עבודת N כשהקלט שלה הוא : α
- N מנחשת ערכים בולאניים למשתני הנוסחה α. מספר המשתנים קטן או שווה לאורך הנוסחה, לכן הניחוש ליניארי בגודל הקלט.
- בשלב הבדיקה, N מציבה הערכים בנוסחה , ליניארי בגודל הקלט α.
- ס"הכ זמן העבודה של N ליניארי ב $|\alpha|$

שלב א בהוכחה, $\in$NP SAT

- מתקיים: אם SAT $\in \alpha$

⇦קיימת לפחות הצבה אחת של ערכים למשתנים של α שנותנת T.

⇦ N יכולה לנחש ערכים אלו, ולכן כשתציב אותם בα , נקבל TRUE

⇦ N תעצור.

שלב א בהוכחה, $\in$NP SAT

- מתקיים: אם SAT $\alpha \notin$

⇦לא קיימת אפילו הצבה אחת של ערכים למשתנים של α

שנותנת TRUE.

⇦ כל ניחוש של ערכים ש N ניחשה יגרור בהצבה FALSE

⇦N לא תעצור.

SAT שפה שלמה בNP

$SAT = \{ \alpha \mid$ ניתנת לסיפוק $\alpha \}$

טענה : $SAT \in NPC$

הוכחה: יש להראות כי א. $SAT \in NP$ ✓

ב. לכל $A \in NP$ מתקיים $A \le_p SAT$

שלב ב - לכל $A \in NP$ מתקיים $A \leq_p SAT$

יש להראות שלכל שפה A בNP, יש רדוקציה פולינומית לSAT .

כלומר, צריך לבנות רדוקציה פולינומית שתעביר מופע x של A

למופע R(x) של SAT, כך שיתקיים $x \in A \Leftrightarrow R(x) \in SAT$

כלומר, שאם הקלט x שייך לשפה A, אזי הנוסחה ש R תייצר

תהיה ניתנת לסיפוק ולהיפך.

שלב ב - לכל $A \in NP$ מתקיים $A \le_p SAT$

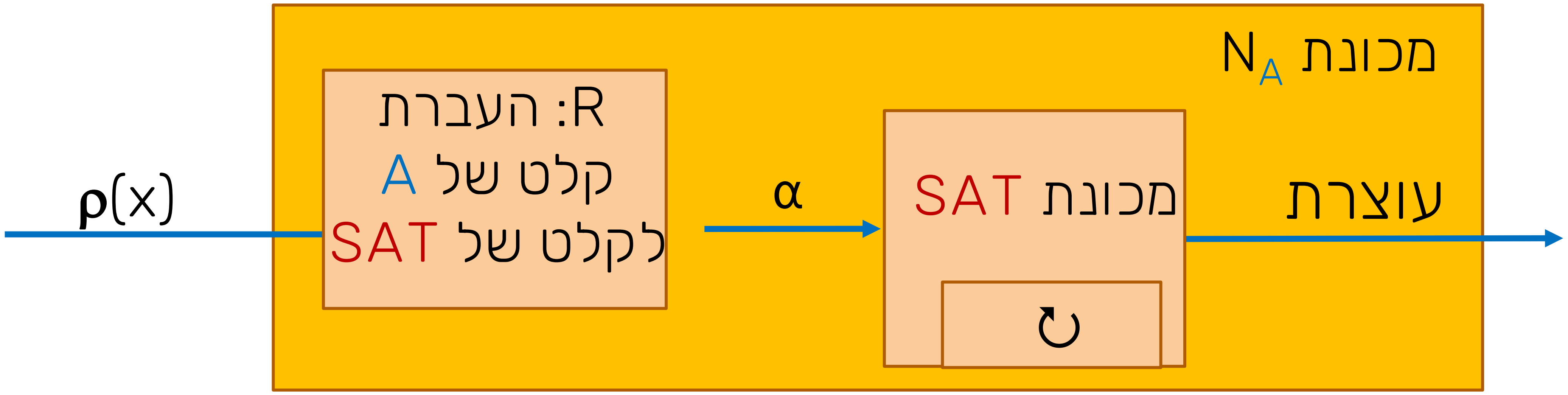
היות ש $A \in NP$ ישנה מ"ט ל"ד $N_A = (\Sigma, Q, \Delta, s, h)$ שמקבלת את

A בזמן פולינומי f_A. כלומר :

- עבור $x \in A$ קיים חישוב של N_A על w שעוצר בזמן $f_A(|x|)$.
- עבור $x \notin A$ לא קיים חישוב של N_A על w שעוצר בזמן $f_A(|x|)$.
- ניתן לומר שאורך הסרט שעליו עובדת N, הוא $len \le |x| + 2 + f(|x|)$

הרדוקציה $A \le_p SAT$

$SAT = \{ \alpha \mid$ α ניתנת לסיפוק $\}$



מה תעשה פרוצדורה R כדי להבטיח $\rho(x) \in A \Leftrightarrow \alpha \in SAT$

כלומר שיש ל α השמה שנותנת TRUE $\Leftrightarrow$ $x \in A$ משמע N_A עוצרת על x

שלב ב - לכל $A \in NP$ מתקיים $A \le_p SAT$

נכנסה ש-R תמיר את עבודת N_A לנוסחה לוגית.

הנתונים החשובים לעבודת N_A הם:

- מהו המצב בו נמצאת המכונה
- היכן עומד הראש
- איזה תו הראש רואה.

נגדיר את המשתנים הבאים:

הערך של המשתנה	המשתנה	כמות
אם בזמן i המכונה N_A נמצאת במצב q_k $\begin{cases} 1 \\ 0 \end{cases}$ אחרת	$Q[i, k]=$	$f(\|x\|) *\|Q\|$ מספר המצבים*זמן ריצה

דוגמא

המשתנה		הערך של המשתנה
$Q[i, k]=$	$\begin{cases} 1 \\ 0 \end{cases}$	אם בזמן i המכונה N_A נמצאת במצב q_k אחרת

אם הגדרנו את $S = q_0$ אז $Q[0, 0]=1$

נגדיר את המשתנים הבאים:

הערך של המשתנה	המשתנה	כמות
אם בזמן i המכונה N_A נמצאת במצב q_k $\begin{cases} 1 \\ 0 \end{cases}$ אחרת	Q[i, k]=	f(|x|) *|Q| מספר המצבים*זמן ריצה
בזמן i הראש של N_A נמצא במיקום j בסרט $\begin{cases} 1 \\ 0 \end{cases}$ אחרת	Head[i,j]=	len*f(|x|) זמן ריצה*אורך הסרט

דוגמא:

הערך של המשתנה | המשתנה

בזמן i הראש של N_A נמצא במיקום j בסרט 1
אחרת 0

$$Head[i,j]= \begin{cases} 1 \\ 0 \end{cases}$$

עבור קונפיג' התחלה (s, #ab<u>#</u>#)

$Head[0,3]=1$

$Head[0,2]=0$

נגדיר את המשתנים הבאים:

הערך של המשתנה	המשתנה	כמות
אם בזמן i המכונה N_A נמצאת במצב q_k — 1 אחרת — 0	$Q[i, k]=$	$f(\|x\|) * \|Q\|$ מספר המצבים*זמן ריצה
בזמן i הראש של N_A נמצא במיקום j בסרט — 1 אחרת — 0	$Head[i,j]=$	$len*f(\|x\|)$ זמן ריצה*אורך הסרט
בזמן i תוכן הסרט במקום j הוא תו σ_c — 1 אחרת — 0	$Symbol[i,j,c] =$	$\|\Sigma\|*len*f(\|x\|)$

דוגמא:

עבור קונפיג' התחלה (s, #ab##)
אם הגדרנו את # = σ_0

Symbol[0,0,0]=1

Symbol[0,3,0]=1

Symbol[0,1,0]=0

$$\text{Symbol}[i,j,c]=\begin{cases}1 & \text{בזמן } i \text{ תוכן הסרט במקום } j \text{ הוא תו } \sigma_c\\ 0 & \text{אחרת}\end{cases}$$

דוגמא למכונה N_A

N_A : $(s,\#) \rightarrow (search\_a, L)$

$(search\_a, a) \rightarrow (h, R)$

$\Sigma = \{\#, a, b\}$

$Q = \{ s, h, search\_a\}$

נניח קונפיגורציית התחלה $(s, \#\, ba\, \underline{\#})$

$len = 2 + 2 + 2 = 6$

הנוסחה שמייצגת את קונפיגורציית ההתחלה.

בהנתן קלט $X = x_1x_2\ldots x_{|x|}$ אזי קונפיגורציית ההתחלה היא

$(s, \# x_1x_2\ldots x_{|x|} \underline{\#})$ לכן נקבל את הנוסחה הבאה:

$Q[0, 0] \wedge head[0, |x|+1] \wedge Symbol[0, 0, |\#|] \wedge Symbol[0, 1, |x_1|] \wedge$

$Symbol[0, 2, |x_2|] \wedge \ldots \wedge Symbol[0, |x|, |x_{|x|}|] \wedge Symbol[0, |x|+1, |\#|]$

$\wedge Symbol[0, |x|+2, 0] \wedge \ldots \wedge Symbol[0, len, 0]$

E1

הנוסחה שמייצגת את קונפיג' ההתחלה -דוגמא

$\Sigma = \{\#, a, b\}$, $Q = \{ s, h, search\_a\}$, $len = 6$

בהנתן קונפיגורציית התחלה (s, # ba _#) לכן נקבל את הנוסחה הבאה:

$Q[0, 0] \wedge head[0, 3] \wedge Symbol[0, 0, 0] \wedge Symbol[0, 1, 2] \wedge$

$Symbol[0, 2, 1] \wedge \ldots \wedge Symbol[0, 3, 0] \wedge Symbol[0, 4, 0]$

$\wedge Symbol[0, 5, 0] \wedge \ldots \wedge Symbol[0, len, 0]$

E2

נוסחאות שטוענות שבכל זמן i המכונה נמצאת בדיוק במצב אחד

$$\bigwedge_{\text{for all } 0\le i \le f(|x|)} \left(Q[i, 0] \vee Q[i, 1] \vee Q[i, 2] \vee \dots \vee Q[i, |Q|]\right)$$

כלומר חייב להיות מצב למכונה בזמן i

$$\bigwedge_{\text{for all } 0\le k\neq k'<|Q| \ \&\ \text{for all } 0\le i \le f(|x|)} Q[i, k] \Rightarrow \overline{Q[i, k']}$$

$$\equiv \overline{Q[i, k]} \vee \overline{Q[i, k']}$$

אם המצב q_k נוכחי בזמן i אזי לא ייתכן שבזמן i מצב המכונה יהיה $q_{k'}$.

נוסחאות שטוענות שבכל זמן i המכונה נמצאת בדיוק במצב אחד - דוגמא

$\Sigma = \{\#, a, b\}$, $Q = \{ s, h, search\_a\}$, $len = 6$, $f(|x|) = 2$

$$\bigwedge_{\text{for all } 0 \le i \le 2} (Q[0, 0] \vee Q[0, 1] \vee Q[0, 2]) \vee (Q[1, 0] \vee Q[1, 1] \vee Q[1, 2]) \vee (Q[2, 0] \vee Q[2, 1] \vee Q[2, 2])$$

כלומר חייב להיות מצב למכונה בזמן i

E2

נוסחאות שטוענות שבכל זמן i המכונה נמצאת בדיוק במצב אחד - דוגמא

Σ= {#, a, b} , Q= { s, h, search_a}, len =6, f(|x|) =2

$$\bigwedge_{\text{for all } 0\le k\ne k'<3\text{ , for all } 0\le i\le 2} Q[0, 0] \Rightarrow \overline{Q[0, 1]} \equiv \overline{Q[0, 0]} \vee \overline{Q[0, 1]}$$

כלומר אם נקבע מצב q_k בזמן i אזי לא ייתכן שבאותו זמן מצב המכונה יהיה $q_{k'}$.

$$(\overline{Q[0, 0]} \vee \overline{Q[0, 1]}) \wedge (\overline{Q[0, 0]} \vee \overline{Q[0, 2]}) \wedge$$

$$(\overline{Q[1, 0]} \vee \overline{Q[1, 1]}) \wedge (\overline{Q[1, 0]} \vee \overline{Q[1, 2]}) \wedge (\overline{Q[1, 1]} \vee \overline{Q[1, 0]}) \wedge (\overline{Q[1, 1]} \vee \overline{Q[1, 2]}) \wedge$$

$$(\overline{Q[1, 2]} \vee \overline{Q[1, 0]}) \wedge (\overline{Q[1, 2]} \vee \overline{Q[1, 1]}) \wedge$$

$$(\overline{Q[2, 0]} \vee \overline{Q[2, 1]}) \wedge (\overline{Q[2, 0]} \vee \overline{Q[2, 2]}) \wedge (\overline{Q[2, 1]} \vee \overline{Q[2, 0]}) \wedge (\overline{Q[2, 1]} \vee \overline{Q[2, 2]}) \wedge$$

$$(\overline{Q[2, 2]} \vee \overline{Q[2, 0]}) \wedge (\overline{Q[2, 2]} \vee \overline{Q[2, 1]})$$

E3

נוסחאות שטוענות שבכל זמן i ראש המכונה נמצא בדיוק במקום אחד

$$\bigwedge_{\text{for all } 0\le i \le f(|x|)} \left(Head[i, 0] \vee Head[i, 1] \vee \dots \vee Head[i, len] \right)$$

כלומר הראש חייב להיות באחד המקומות בסרט בזמן i

$$\bigwedge_{\text{for all } 0\le i \le f(|x|) \;\&\; \text{for all } 0\le j \ne j' < len} Head[i, j] \Rightarrow \overline{Head[i,j']} \equiv \overline{Head[i,j]} \vee \overline{Head[i,j']}$$

אם הראש נמצא במיקום j בזמן i אזי לא ייתכן שבאותו זמן מיקום הראש יהיה גם j’.

E3

נוסחאות שטוענות שבכל זמן i ראש המכונה נמצא בדיוק במקום אחד - דוגמא

$\Sigma = \{\#, a, b\}$, $Q = \{ s, h, search\_a\}$, len =6, $f(|x|) = 2$

$$\bigwedge_{\text{for all } 0 \le i \le 2} \begin{aligned} &(Head[0, 0] \vee Head[0, 1] \vee Head[0, 2] \vee Head[0, 3] \\ &\quad \vee Head[0, 4] \vee Head[0, 5] \vee Head[0, 6]\,) \\ &(Head[1, 0] \vee Head[1, 1] \vee Head[1, 2] \vee Head[1, 3] \\ &\quad \vee Head[1, 4] \vee Head[1, 5] \vee Head[1, 6]\,) \\ &(Head[2, 0] \vee Head[2, 1] \vee Head[2, 2] \vee Head[2, 3] \\ &\quad \vee Head[2, 4] \vee Head[2, 5] \vee Head[2, 6]\,) \end{aligned}$$

כלומר הראש חייב להיות באחד המקומות בסרט בזמן i

E3

נוסחאות שטוענות שבכל זמן i ראש המכונה נמצא בדיוק במקום אחד - דוגמא

$\Sigma = \{\#, a, b\}$, $Q = \{ s, h, search\_a\}$, len =6, $f(|x|) = 2$

$$\bigwedge \text{ for all } 0 \le i \le f(|x|) \ \& \text{ for all } 0 \le j \ne j' < \text{len } Head[0, 0] \Rightarrow \overline{Head[0, 1]} \equiv \overline{H[0, 0]} \vee \overline{H[0, 1]}$$

כלומר אם הראש נמצא במיקום j בזמן i אזי לא ייתכן שבזמן i מיקום הראש יהיה j’.

$$(\overline{H[0, 0]} \vee \overline{H[0, 1]}) \wedge (\overline{H[0, 0]} \vee \overline{H[0, 2]}) \wedge (\overline{H[0, 0]} \vee \overline{H[0, 3]}) \wedge (\overline{H[0, 0]} \vee \overline{H[1, 4]}) \wedge$$
$$(\overline{H[0, 0]} \vee \overline{H[0, 5]}) \wedge (\overline{H[0, 0]} \vee \overline{H[0, 6]}) \wedge$$

$$(\overline{H[1, 0]} \vee \overline{H[1, 1]}) \wedge (\overline{H[1, 0]} \vee \overline{H[1, 2]}) \wedge (\overline{H[1, 0]} \vee \overline{H[1, 3]}) \wedge (\overline{H[1, 0]} \vee \overline{H[1, 4]}) \wedge (\overline{H[1, 0]} \vee$$
$$\overline{H[1, 5]}) \wedge (\overline{H[1, 0]} \vee \overline{H[1, 6]}) \wedge$$

$$(\overline{H[2, 0]} \vee \overline{H[2, 1]}) \wedge (\overline{H[2, 0]} \vee \overline{H[2, 2]}) \wedge (\overline{H[2, 0]} \vee \overline{H[2, 3]}) \wedge (\overline{H[2, 0]} \vee \overline{H[2, 4]}) \wedge (\overline{H[2, 0]}$$
$$\vee \overline{H[2, 5]}) \wedge (\overline{H[2, 0]} \vee \overline{H[2, 6]})$$

E4

נוסחאות שטוענות שבכל זמן i במיקום j בסרט של המכונה יש תו אחד

$\bigwedge$ for all $0 \le i \le f(|x|)$ & for all $0 \le j \le len$ $(Symbol[i,j,0] \vee Symbol[i,j,1] \vee \ldots \vee Symbol[i,j,|\Sigma|])$

כלומר בזמן i ובמקום j בסרט, חייב להיות תו כלשהו.

$\bigwedge$ for all $0 \le c \ne c' \le |\Sigma|$ & for all $0 \le i \le f(|x|)$ & for all $0 \le j \le len$ $Symbol[i,j,c] \Rightarrow \overline{Symbol[i,j,c']} \equiv \overline{Symbol[i,j,c]} \vee \overline{Symbol[i,j,c']}$

כלומר אם במקום j בסרט בזמן i נמצא תו c, אזי לא ייתכן שבאותו זמן ובאותו מיקום יהיה גם תו c'.

E4

נוסחאות שטוענות שבכל זמן i במיקום j בסרט של המכונה יש תו אחד -דוגמא

$\Sigma = \{\#, a, b\}$, $Q = \{ s, h, search\_a\}$, $len = 6$, $f(|x|) = 2$

בזמן i ובמקום j בסרט, חייב להיות תו כלשהו.

$(Symbol[0,0,0] \vee Sym[0,0,1] \vee Sym[0,0,2]) \wedge$

$(Symbol[0,1,0] \vee Sym[0,1,1] \vee Sym[0,1,2]) \wedge$

:

$(Symbol[0,6,0] \vee Sym[0,6,1] \vee Sym[0,6,2]) \wedge$

$(Symbol[1,0,0] \vee Sym[1,0,1] \vee Sym[1,0,2]) \wedge$

$(Symbol[1,1,0] \vee Sym[1,1,1] \vee Sym[1,1,2]) \wedge$

:

$(Symbol[1,6,0] \vee Sym[1,6,1] \vee Sym[1,6,2]) \wedge .....$

E4

נוסחאות שטוענות שבכל זמן i במיקום j בסרט של המכונה יש תו אחד -דוגמא

$\Sigma = \{\#, a, b\}$, $Q = \{ s, h, search\_a\}$, len =6, $f(|x|) = 2$

$$\bigwedge Sym[i,j,c] \Rightarrow \overline{Sym[i,j,c']} \equiv \overline{Sym[i,j,c]} \vee \overline{Sym[i,j,c']}$$

כלומר אם במקום j בסרט בזמן i נמצא תו c , אזי לא ייתכן שבזמן i ובמיקום j יהיה גם תו c’ .

$$(\overline{Sym[0,0,0]} \vee \overline{Sym[0,0,1]}) \wedge (\overline{Sym[0,0,0]} \vee \overline{Sym[0,0,2]}) \wedge$$

$$(\overline{Sym[0,1,0]} \vee \overline{Sym[0,0,1]}) \wedge (\overline{Sym[0,1,0]} \vee \overline{Sym[0,0,2]}) \wedge$$

:

$$(\overline{Sym[0,6,0]} \vee \overline{Sym[0,6,1]}) \wedge (\overline{Sym[0,6,0]} \vee \overline{Sym[0,6,2]}) \wedge$$

$$(\overline{Sym[1,0,0]} \vee \overline{Sym[1,0,1]}) \wedge (\overline{Sym[1,0,0]} \vee \overline{Sym[1,0,2]}) \wedge$$

:

$$(\overline{Sym[1,6,0]} \vee \overline{Sym[1,6,1]}) \wedge (\overline{Sym[1,6,0]} \vee \overline{Sym[1,6,2]}) \wedge \ldots\ldots$$

E5

פסוקית שמייצגת הגעה למצב עצירה

$Q[f(|x|) , |h|]$

היות שהמכונה מקבלת את x בזמן $f(|x|)$, היא צריכה להגיע למצב מקבל בזמן זה.

לדוגמא עבור: $\Sigma = \{\#, a, b\}$, $Q= \{ s, h, search\_a\}$, len =6, $f(|x|) =2$

נוסיף את הפסוקית $Q[2, 1]$

E6

נוסחאות שטוענות שהקונפיג' של N_B בזמן i+1 נקבעת
לפי הפעלה אחת של מעבר מ Δ על הקונפיגורציה בזמן i

לכל מעבר ב Δ מהסוג: $\Delta(q_k, \alpha) = (q_{k'}, \beta)$

$\wedge$ for all $0 \le i \le f(|x|)$ and for all $0 \le j \le len$

$(\, Q[i, k] \wedge Head[i, j] \wedge Symbol[i, j, |\alpha|\,]) \Rightarrow Symbol[\underline{i+1}, j, |\beta|]\,)$

$\wedge\, (Q[i, k] \wedge Head[i, j] \wedge Symbol[i, j, |\alpha|] \Rightarrow Q[i+1, k']$

$\wedge\, (Q[i, k] \wedge Head[i, j] \wedge Symbol[i, j, |\alpha|] \Rightarrow Head[i+1, j]$

נוסחאות שטוענות שהקונפיג' של N_B בזמן i+1 נקבעת
לפי הפעלה אחת של מעבר מ Δ על הקונפיגורציה בזמן i

לכל מעבר ב Δ מהסוג: $\Delta(q_k, \alpha) = (q_{k'}, \beta)$

$\wedge$ for all $0 \le i \le f(|x|)$ and for all $0 \le j \le len$

$$(\overline{Q[i,k]} \vee \overline{Head[i,j]} \vee \overline{Symbol[i,j,|\alpha|]} \vee Symbol[i+1, j, |\beta|])$$

$$\wedge (\overline{Q[i,k]} \vee \overline{Head[i,j]} \vee \overline{Symbol[i,j,|\alpha|]} \vee Q[i+1, k'])$$

$$\wedge (\overline{Q[i,k]} \vee \overline{Head[i,j]} \vee \overline{Symbol[i,j,|\alpha|]} \vee Head[i+1, j])$$

E6

נוסחאות שטוענות שהקונפיג' של N_B בזמן $i+1$ נקבעת
לפי הפעלה אחת של מעבר מ Δ על הקונפיגורציה בזמן i

לכל מעבר ב Δ מהסוג: $\Delta(q_k, \alpha) = (q_{k'}, L)$

$\wedge$ for all $0 \le i \le f(|x|)$ and for all $0 \le j \le len$

$(Q[i, k] \wedge Head[i, j] \wedge Symbol[i, j, |\alpha|]) \Rightarrow Symbol[i+1, j, |\alpha|]) =$

$\wedge (Q[i, k] \wedge Head[i, j] \wedge Symbol[i, j, |\alpha|] \Rightarrow Q[i+1, k']$

$\wedge (Q[i, k] \wedge Head[i, j] \wedge Symbol[i, j, |\alpha|] \Rightarrow Head[i+1, j-1]$

נוסחאות שטוענות שהקונפיג' של N_B בזמן i+1 נקבעת לפי
הפעלה אחת של מעבר מ Δ על הקונפיגורציה בזמן i - דוגמא

$(s,\#) \rightarrow (search\_a, L)$: N_A

$\Sigma = \{\#, a, b\}$ $Q = \{ s, h, search\_a\}$ len = 6 f(2) =2

לכל מעבר ב Δ מהסוג: $\Delta(s, \#) = (search\_a, L)$

$\wedge$ for all $0 \le i \le 2$ & for all $0 \le j \le 6$

$(Q[0, 0] \wedge Head[0, j] \wedge Symbol[0, j, 0]) \Rightarrow Symbol[1, j, 0]$

$\wedge(Q[0, 0] \wedge Head[0, j] \wedge Symbol[0, j, 0]) \Rightarrow Q[1, 2]$

$\wedge (Q[0\ 0] \wedge Head[0, j] \wedge Symbol[0, j, 0] \Rightarrow Head[1, j-1]$

$\wedge(Q[1, 0] \wedge Head[1, j] \wedge Symbol[1, j, 0]) \Rightarrow Symbol[2, j, 0]$
$\wedge(Q[1, 0] \wedge Head[1, j] \wedge Symbol[1, j, 0]) \Rightarrow Q[2, 2]$
$\wedge (Q[1, 0] \wedge Head[1, j] \wedge Symbol[1, j, 0]) \Rightarrow Head[2, j-1]$

E6

נוסחאות שטוענות שהקונפיג' של N_B בזמן $i+1$ נקבעת
לפי הפעלה אחת של מעבר מ Δ על הקונפיגורציה בזמן i

לכל מעבר ב Δ מהסוג: $\Delta(q_k, \alpha) = (q_{k'}, R)$

$\wedge$ for all $0 \leq i \leq f(|x|)$ and for all $0 \leq j \leq len$

$(Q[i, k] \wedge Head[i, j] \wedge Symbol[i, j, |\alpha|]) \Rightarrow Symbol[\underline{i+1,} j, |\alpha|])$

$\wedge (Q[i, k] \wedge Head[i, j] \wedge Symbol[i, j, |\alpha|] \Rightarrow Q[\underline{i+1}, k']$

$\wedge (Q[i, k] \wedge Head[i, j] \wedge Symbol[i, j, |\alpha|] \Rightarrow Head[\underline{i+1}, j+1]$

סיכום הרדוקציה

הנוסחה שהרדוקציה בונה מתוך הקלט x והמכונה N_A שמקבלת את שפה A היא

$$\boldsymbol{\alpha} = E1 \wedge E2 \wedge E3 \wedge E4 \wedge E5 \wedge E6$$

זמן: ראינו שכל קבוצת פסוקיות E מכילה מספר פסוקיות פולינומי ב x ובו N_A שנחשב קבוע כי הוא בגודל סופי ואינו תלוי בגודל X הקלט.

$\boldsymbol{\alpha}$ = E1 **∧** E2 **∧** E3 **∧** E4 **∧** E5 **∧** E6

$(s,\#) \rightarrow (search\_a, L) \quad : \; N_A$

$(search\_a, a) \rightarrow (h, R)$

$Q[0, 0] \land head[0, 3] \land Sym[0, 0, 0] \land Sym[0, 1, 2] \land Sym[0, 2, 1] \land Sym[0, 3, 0] \ldots$

$\land\, Symbol[0, len, 0] \land \land (Q[0, 0] \lor Q[0, 1] \lor Q[0, 2]) \lor (Q[1, 0] \lor Q[1, 1] \lor Q[1, 2]) \lor (Q[2, 0] \lor$

$Q[2, 1] \lor Q[2, 2]) \boldsymbol{\land} (\overline{Q[0, 0]} \lor \overline{Q[0, 1]}) \land (\overline{Q[0, 0]} \lor \overline{Q[0, 2]}) \land \ldots ((\overline{Q[2, 0]} \lor \overline{Q[2, 1]}) \land (\overline{Q[2, 0]} \lor \overline{Q[2, 2]}) \land$

$(\overline{Q[2, 1]} \lor \overline{Q[2, 0]}) \land (\overline{Q[2, 1]} \lor \overline{Q[2, 2]}) \land (\overline{Q[2, 2]} \lor \overline{Q[2, 0]}) \land (\overline{Q[2, 2]} \lor \overline{Q[2, 1]}) \boldsymbol{\land} (Head[0, 0] \lor \ldots \lor H[0, 6]) \land ..$

$\land (H[2, 0] \lor \ldots \lor H[2, 6]) \boldsymbol{\land} \overline{H[0, 0]} \lor \overline{H[0, 1]}) \land \ldots \land (\overline{H[0, 0]} \lor \overline{H[0, 6]}) \land .. \land (\overline{H[2, 0]} \lor \overline{H[2, 1]}) \land \ldots (\overline{H[2, 0]} \lor \overline{H[2,6]})$

$\land (Symbol[0,0,0] \lor Sym[0,0, 1] \lor Sym[0,0,2]) \land \ldots \land (Symbol[1,6,0] \lor Sym[1,6, 1] \lor Sym[1,6,2]) \land \ldots..$

$\land (Sym[0,0,0] \lor Sym[0,0,1]) \land (Sym[0,0,0] \lor Sym[0,0,2]) \land \ldots \land (Sym[1,6,0] \lor Sym[1,6,2]) \land \ldots \land Q[2 , 1]$

$\land (\overline{Q[0, 0]} \lor \overline{H[0, j]} \lor \overline{Sym[0, j, 0])} \lor Sym[1, j, 0] \;\; \land (\overline{Q[0, 0]} \lor \overline{H[0, j]} \lor \overline{Sym[0, j, 0])} \lor Q[1, 2]$

$\land (\overline{Q[0\ 0]} \lor \overline{H[0, j]} \lor \overline{Sym[0, j, 0]} \lor H[1, j-1]$

$\land (\overline{Q[1, 2]} \lor \overline{H[1, j]} \lor \overline{Sym[1, j, 1])} \lor Sym[2, j, 1] \;\; \land (\overline{Q[1, 2]} \lor \overline{H[1, j]} \lor \overline{Sym[1, j, 1])} \lor Q[2, 1]$

$\land (\overline{Q[1\ 2]} \lor \overline{H[1, j]} \lor \overline{Sym[1, j, 1]} \lor H[2, j+1]$

$(s,\#) \rightarrow (search\_a, L) \; : \; N_A$

$(search\_a, a) \rightarrow (h, R)$

$\boldsymbol{\alpha} = E1 \wedge E2 \wedge E3 \wedge E4 \wedge E5 \wedge E6$

$$Q[0,0] \wedge head[0,3] \wedge Sym[0,0,0] \wedge Sym[0,1,2] \wedge Sym[0,2,1] \wedge Sym[0,3,0] \ldots$$

$$\wedge\, Symbol[0,6,0] \wedge \wedge (Q[0,0] \vee Q[0,1] \vee Q[0,2]) \vee (Q[1,0] \vee Q[1,1] \vee Q[1,2]) \vee (Q[2,0] \vee Q[2,$$

$$1] \vee Q[2,2]) \wedge (\overline{Q[0,0]} \vee \overline{Q[0,1]}) \wedge (\overline{Q[0,0]} \vee \overline{Q[0,2]}) \wedge \ldots ((\overline{Q[2,0]} \vee \overline{Q[2,1]}) \wedge (\overline{Q[2,0]} \vee \overline{Q[2,2]}) \wedge (\overline{Q[2,}$$

$$\overline{1]} \vee \overline{Q[2,0]}) \wedge (\overline{Q[2,1]} \vee \overline{Q[2,2]}) \wedge (\overline{Q[2,2]} \vee \overline{Q[2,0]}) \wedge (\overline{Q[2,2]} \vee \overline{Q[2,1]}) \wedge (Head[0,0] \vee \ldots \vee H[0,6]) \wedge .. \wedge$$

$$(H[2,0] \vee \ldots \vee H[2,6]) \wedge \overline{H[0,0]} \vee \overline{H[0,1]}) \wedge \ldots \wedge (\overline{H[0,0]} \vee \overline{H[0,6]}) \wedge .. \wedge (\overline{H[2,0]} \vee \overline{H[2,1]}) \wedge \ldots (\overline{H[2,0]} \vee \overline{H[2,6]}) \wedge$$

$$(Symbol[0,0,0] \vee Sym[0,0,1] \vee Sym[0,0,2]) \wedge \ldots \wedge (Symbol[1,6,0] \vee Sym[1,6,1] \vee Sym[1,6,2]) \wedge \ldots..$$

$$\wedge (\overline{Sym[0,0,0]} \vee \overline{Sym[0,0,1]}) \wedge (\overline{Sym[0,0,0]} \vee \overline{Sym[0,0,2]}) \wedge \ldots \wedge (\overline{Sym[1,6,0]} \vee \overline{Sym[1,6,2]}) \wedge \ldots \wedge Q[2,1]$$

$$\wedge (\overline{Q[0,0]} \vee \overline{H[0,j]} \vee \overline{Sym[0,j,0]}) \vee Sym[1,j,0] \;\; \wedge (\overline{Q[0,0]} \vee \overline{H[0,j]} \vee \overline{Sym[0,j,0]}) \vee Q[1,2]$$

$$\wedge (\overline{Q[0\;0]} \vee \overline{H[0,j]} \vee \overline{Sym[0,j,0]} \vee H[1,j-1]$$

$$\wedge (\overline{Q[1,2]} \vee \overline{H[1,j]} \vee \overline{Sym[1,j,1]}) \vee Sym[2,j,1] \;\; \wedge (\overline{Q[1,2]} \vee \overline{H[1,j]} \vee \overline{Sym[1,j,1]}) \vee Q[2,1]$$

$$\wedge (\overline{Q[1\;2]} \vee \overline{H[1,j]} \vee \overline{Sym[1,j,1]} \vee H[2,j+1]$$

$\boldsymbol{\alpha}$ = E1 ∧ E2 ∧ E3 ∧ E4 ∧ E5 ∧ E6

(s,#) → (search_a, L) : N_A
(search_a, a) → (h, R)

Q[0, 0] ∧ head[0, 3] ∧ Sym[0, 0, 0] ∧ Sym[0, 1, 2] ∧ Sym[0, 2, 1] ∧ Sym[0, 3, 0] … ∧ Symbol[0, 6, 0] ∧ ∧ (Q[0, 0] V Q[0, 1] V Q[0, 2]) V(Q[1, 0] VQ[1, 1]V Q[1, 2])V (Q[2, 0]V Q[2, 1] VQ[2, 2]) ∧ $(\overline{Q[0,0]} \vee \overline{Q[0,1]}) \wedge (\overline{Q[0,0]} \vee \overline{Q[0,2]}) \wedge \ldots (\overline{Q[2,0]} \vee \overline{Q[2,1]}) \wedge (\overline{Q[2,0]} \vee \overline{Q[2,2]}) \wedge (\overline{Q[2,1]} \vee \overline{Q[2,0]}) \wedge (\overline{Q[2,1]} \vee \overline{Q[2,2]}) \wedge (\overline{Q[2,2]} \vee \overline{Q[2,0]}) \wedge (\overline{Q[2,2]} \vee \overline{Q[2,1]})$ ∧ (Head[0, 0] V … V H[0, 6]) ∧ .. ∧ (H[2, 0] V… V H[2, 6]) ∧ $\overline{H[0,0]} \vee \overline{H[0,1]}) \wedge \ldots \wedge (\overline{H[0,0]} \vee \overline{H[0,6]}) \wedge .. \wedge (\overline{H[2,0]} \vee \overline{H[2,1]}) \wedge \ldots (\overline{H[2,0]} \vee \overline{H[2,6]})$ ∧ (Symbol[0,0,0] V Sym[0,0, 1] V Sym[0,0,2]) ∧ … ∧(Symbol[1,6,0] V Sym[1,6, 1] V Sym[1,6,2]) ∧ …..

∧ $(\overline{Sym[0,0,0]} \vee \overline{Sym[0,0,1]}) \wedge (\overline{Sym[0,0,0]} \vee \overline{Sym[0,0,2]}) \wedge \ldots \wedge (\overline{Sym[1,6,0]} \vee \overline{Sym[1,6,2]}) \wedge \ldots \wedge$ Q[2 , 1]

∧ $(\overline{Q[0,0]} \vee \overline{H[0,j]} \vee \overline{Sym[0,j,0]}) \vee Sym[1,j,0]$ ∧ $(\overline{Q[0,0]} \vee \overline{H[0,j]} \vee \overline{Sym[0,j,0]}) \vee Q[1,2]$

∧ $(\overline{Q[0\ 0]} \vee \overline{H[0,j]} \vee \overline{Sym[0,j,0]} \vee H[1,j-1]$

∧ $(\overline{Q[1,2]} \vee \overline{H[1,j]} \vee \overline{Sym[1,j,1]}) \vee Sym[2,j,1]$ ∧ $(\overline{Q[1,2]} \vee \overline{H[1,j]} \vee \overline{Sym[1,j,1]}) \vee Q[2,1]$

∧ $(\overline{Q[1\ 2]} \vee \overline{H[1,j]} \vee \overline{Sym[1,j,1]} \vee H[2,j+1]$

מתקיים:

אם $A \in x$, למ"ט N_A שמקבלת את A בזמן $f(|x|)$, קיים חישוב שעוצר

על קלט x תוך $f(|x|)$ צעדי ריצה ומגיע למצב h.

נציב ערכי אמת למשתני Symbol[]. Head[], Q[] בE1 לפי הקלט

ולאחר מכן נעקוב על מעברי N_A במסלול חישוב זה, לפי זמני

הפעילות של N_A וניתן ערכי אמת מתאימים למשתנים.

$\Leftarrow$השמה זו תספק את הנוסחה, כי בנינו נוסחה שמתארת ריצה

חוקית של $N_A \Leftarrow$ SAT $\alpha \in$

מתקיים:

אם $A \notin x$, למ"ט N_B שמקבלת את A בזמן $f(|x|)$, לא קיים חישוב שעוצר על קלט x תוך $f(|x|)$ צעדי ריצה ומגיע למצב h.

גם אם נציב ערכי אמת למשתני []Symbol. []Q[], Head ב E1 לפי הקלט ולאחר מכן נעקוב על מעברי N_A בכל מסלול חישוב שננסה, לפי זמני הפעילות של N_A וניתן ערכי אמת מתאימים למשתנים לא נוכל לתת TRUE ל $Q[f(|x|), |h|]$, כי באותו זמן המכונה N_A לא נמצאת ב h.

$\Leftarrow$ לא קיימת השמה שתספק את הנוסחה $\Leftarrow$ $\alpha \notin SAT$

סיכום הוכחה

- הראנו שאם $A \in x \Leftrightarrow SAT \in \alpha$
- כאשר R עבדה בזמן פולינומי בגודל הקלט x ולכן הראנו $SAT \leq_p A$.

הראנו כי א. $SAT \in NP$ ✓

וכעת הראנו ב. לכל $A \in NP$ מתקיים $SAT \leq_p A$

לכן $SAT \in NPC$