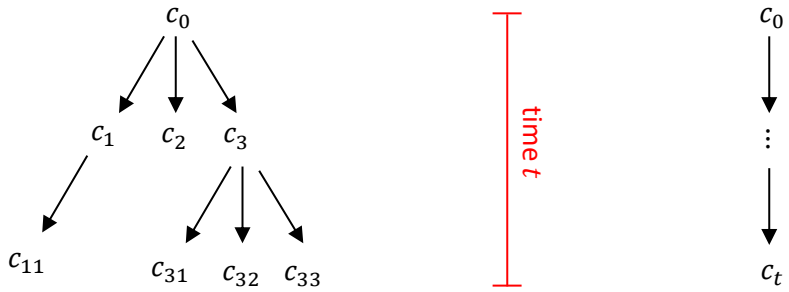
26.5.2025 ***מודלים חישוביים***

הרצאה 9: המחלקות P ו- NP

Based on previous iterations of this course, given by Nir Bitansky, Rotem Oshman, Iftach Haitner, and Omer Paneth.

מרצה: אורי שטמר

בפעם שעברה הגדרנו זמן ריצה של מ"ט / מטל"ד.

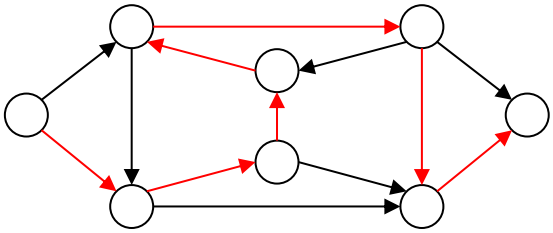


הגדרנו את המחלקה *P* המייצגת בעיות שניתנות לפתרון "פיזבילי" ואת המחלקה *NP* שנאפיין היום.

$$P \coloneqq \bigcup_{c\in\mathbb{N}} DTime(n^c) \qquad NP \coloneqq \bigcup_{c\in\mathbb{N}} NTime(n^c)$$

<u>הגדרה:</u> מסלול המילטוני בגרף מכוון G הוא מסלול שמבקר בכל צומת בדיוק פעם אחת.

לדוגמה:



<u>הגדרה:</u>

$$HAMPATH = \{\langle G,s,t\rangle : \text{גרף מכוון } G \text{ עם מסלול המילטוני מ } s \text{ ל } t\}$$

<u>אלג' נאיבי:</u> נעבור על כל המסלולים ונבדוק לכל אחד האם המילטוני מ- s ל- t. זמן ריצה $O(n!) \approx n^n$.

לא ידוע אלג' יעיל לבעיה. האלג' הטוב ביותר הידוע רץ בזמן $2^{O(n)}$, כלומר רק קצת יותר טוב מהאל' הנאיבי.

טענה: $HAMPATH \in NP$

נתאר מטל"ד N שמכריעה את $HAMPATH$ בזמן פולינומי:

N בהינתן קלט $\langle G=(V,E),s,t\rangle$:	
1. לכל i מ- 1 עד $n=\lvert V\rvert$:	$n\times$
- נבחר ("ננחש") $v\in V$ באופן לא-דטרמיניסטי, נכתוב v על הסרט ונזוז ימינה	$O(1)$
2. נבדוק באופן דטרמיניסטי האם קיבלנו מסלול המילטוני מ- s ל- t	$poly(n)$

סה"כ זמן ריצה / עומק עץ החישוב: $poly(n)$

N מכריעה את HAMPATH משום שיש עלה מקבל בעץ הריצה אם"ם יש מסלול המילטוני.

הערה: כשאנחנו מנתחים זמן ריצה של מטל"ד כזאת אנחנו צריכים להראות שכל המסלולים האפשריים בעץ החישוב חסומים. במקרה הזה כל המסלולים הם באותו אורך. בשלב הראשון בכל צעד אנחנו מתפצלים ל n אפשרויות. ככה במשך n צעדים התפצלנו ל n^n אפשרויות. ואז יש עוד חישוב דטר ללא פיצולים בעומק poly(n).

נשים לב: אמנם אנחנו לא יודעים לפתור את הבעיה הזאת באופן יעיל (ע"י מ"ט), אבל בהינתן פתרון (מסלול $w_1,w_2,\dots,w_n$) ניתן לבדוק ביעילות שהוא אכן פתרון:

אלג' הוידוא צריך לבדוק: (1) לכל i קימת קשת (w_i,w_{i+1}) בגרף. (2) כל צומת בגרף מופיע בדיוק פעם אחת במסלול.

כלומר לוודא נכונות של פתרון זה יותר קל מאשר למצוא אותו, באופן דומה להוכחות מתמטיות.
כעת נפרמל זווית ראיה זו ונראה שהיא לא מוגבלת לבעיית *HAMPATH*.

NP ווידוא פולינומי

הגדרה: תהי V מ"ט עם א"ב קלט $\Sigma\cup\{,\}$ ותהי $L\subseteq\Sigma^*$. נאמר ש- V הוא מוודא פולינומי עבור L אם:

- **נכונות: (א)** לכל $x\in L$ קיים $w\in\Sigma^*$ כך ש- $V(x,w)$ מקבל (תנאי זה נקרא "שלמות")
  (ב) לכל $x\notin L$ ולכל $w\in\Sigma^*$ מתקיים ש- $V(x,w)$ דוחה (נקרא "נאותות")
- **יעילות:** קיים פולינום $p(n)$ כך שלכל $x,w\in\Sigma^*$ זמן הריצה של $V(x,w)$ לכל היותר $p(|x|)$

נחשוב על w כך ש- $V(x,w)$ מקבלת בתור עד לכך ש- $x\in L$.

שימו לב כי V נדרש להיות פולינומי ב- $|x|$ (במקום ב- $|w|+|x|$). בפרט לכל $x\in L$ מובטח שקיים עד w באורך לכל היותר $p(|x|)$.

<u>טענה:</u> $L \in NP$ אם"ם יש ל- L מוודא פולינומי.

<u>הוכחה:</u>

$\Leftarrow$ נניח כי קיימת מט"ד N עם זמן ריצה $p(n)$ המכריעה את L. נבנה מוודא V עבור L.
יהי $b = O(1)$ חסם על דרגת עצי הקונפוגורציות של N.

<u>$V(x,w)$:</u>
- נוודא כי w מתאר כתובת בעץ הקונפ', כלומר $w \in \{1,2,\dots,b\}^i$ עבור $i \leq p(n)$. אחרת נדחה
- נחשב את הקונפ' המתאימה c_w ונקבל אם"ם מקבלת

קיים w כך ש- $v(x,w)$ מקבלת אם"ם קיים ענף מקבל ב- $T_{N,x}$, מה שקורה אם"ם $x \in L$.
זמן ריצה: $O\big(p(|x|)\big)$.

$\Rightarrow$ נניח כי קיים מוודא $V(x,w)$ עם זמן ריצה $p(|x|)$.
נבנה מט"ד N עם זמן ריצה פולינומי המכריעה את L.

<u>$N(x)$:</u>
- נבחר באופן לא-דטרמיניסטי w באורך $p(n) \geq$
- נקבל אם"ם $V(x,w)$ מקבל

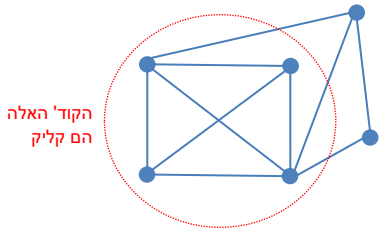
N מקבלת אם"ם קיים w באורך $p(n) \geq$ כך ש- $V(x,w)$ מקבל, מה שקורה אם"ם $x \in L$.

וידוא (דטרמיניסטי) של הוכחה זה קונספט הרבה יותר טבעי עבורנו מאשר חישוב לא דטרמיניסטי ועליו נחשוב מעתה. עכשיו נשתמש באפיון הזה כדי להראות דוגמאות נוספות לשפות ב- *NP*.

<u>דוגמה: בעיית הקליקה</u>

<u>הגדרה:</u> בהינתן גרף לא מכוון $G = (V,E)$, קבוצה $C \subseteq V$ היא <u>קליק</u> בגרף אם לכל $u \neq v \in C$ מתקיים $(u,v) \in E$.

בציור:



<u>הגדרה:</u>

$$CLIQUE = \{\langle G,k\rangle : \text{ גרף לא מכוון עם קליק בגודל } k \text{ } G\}$$

<u>טענה:</u> $CLIQUE \in NP$

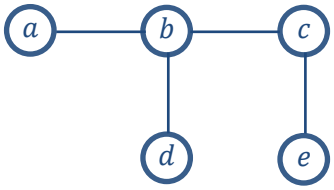
<u>רעיון הוכחה:</u> העד ל $\langle G,k\rangle \in CLIQUE$ הוא קידוד של קליק C בגודל k ב- G.
ניתן לבדוק שאכן קליק בזמן פולינומי.

שאלה: האם $CLIQUE \in P$? לא ידוע... מאמינים שלא...

דוגמה נוספת: בעיית הקבוצה הבלתי תלויה *(independent set)*

הגדרה: בהינתן גרף לא מכוון $G=(V,E)$, קבוצה $I \subseteq V$ היא בלתי תלויה בגרף אם לכל $u,v \in C$ מתקיים $(u,v) \notin E$.

בציור:



- $\{a,d,e\}$ קב' ב"ת מקסימלית בגרף
- גם $\{a,c\}$ ב"ת

הגדרה:

$$IS = \{\langle G,k\rangle : \ G \text{ גרף לא מכוון עם קב' ב"ת בגודל } k\}$$

טענה: $IS \in NP$ מה העד?

שאלה: האם $IP \in P$? לא ידוע... מאמינים שלא...

דוגמה נוספת: בעיית *FACTOR*

הגדרה:

$$FACTOR = \{\langle N,k\rangle : \ N \text{ קיים } 1 < d \leq k \text{ המחלק את}\}$$

טענה: $FACTOR \in NP$ מה העד?

שאלה: האם $FACTOR \in P$? לא ידוע... מאמינים שלא...

NP-hardness

אנחנו יודעים ש- $P \subseteq NP$. האמונה הרווחת היא ש- $P \neq NP$ ואמונה זו מגובה בכסף רב. בפרט עולם הקריפטוגרפיה נשען על הנחה זו (ולמעשה הרבה יותר מכך). אבל אנחנו לא יודעים להוכיח את זה.

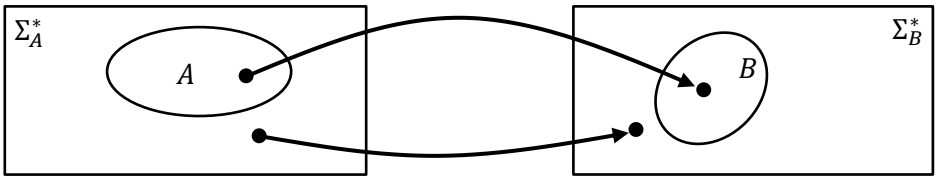
איך בכל זאת נוכל לטעון ששפות מסויימות ב *NP* הן "קשות"?

1. נמצא שפת "עוגן" $L \in NP$ ונראה ש- $L \in P$ **אם"ם** $P = NP$. שפה כזו נקראת *NP*-שלמה. כלומר, אנחנו לא נוכיח לגבי L שאין לה מכריע פולינומי. מה שכן נוכיח זה שאם יש מכריע פולינומי, אז בבת אחת יש מכריע פולינומי לכל שפה ב *NP*, כולל *CLIQUE,IS,HAMPATH*...
2. נראה שפות *NP*-שלמות נוספות בעזרת רדוקציות.

תזכורת, בהרצאה 7 הגדרנו "רדוקציית מיפוי" באופן הבא:

יהיו Σ_A, Σ_B אלפבית ויהיו $A \subseteq \Sigma_A^*$ ו- $B \subseteq \Sigma_B^*$ שפות. <u>רדוקציית מיפוי</u> מ- A ל- B היא פונקציה חשיבה $f: \Sigma_A^* \to \Sigma_B^*$ כך שלכל $x \in \Sigma_A^*$ מתקיים:

$$x \in A \quad \text{אם"ם} \quad f(x) \in B$$



<u>הגדרה:</u> נאמר שרדוקציית מיפוי f היא <u>פולינומית</u> אם היא חשיבה בזמן פולינומי.

כלומר, קיימת מ"ט M וקיים פולינום p כך שלכל קלט $x \in \Sigma_A^*$ מתקיים ש- $M(x)$ עוצרת תוך לכל היותר $p(|x|)$ צעדים ובסיום הריצה על הסרט כתוב $f(x)$.

כלומר f מתרגמת <u>ביעילות</u> קלטים עבור בעיית ההכרעה של A לקלטים עבור בעיית ההכרעה של B

<u>סימון:</u> אם יש רדוקציית מיפוי פולינומית מ- A ל- B אז נסמן $A \leq_p B$.

<u>טענה:</u> $IS \leq_p CLIQUE$

<u>הגדרת עזר:</u> <u>הגרף המשלים</u> של גרף לא מכוון $G = (V, E)$ הוא הגרף $\overline{G} = (V, \overline{E})$ כאשר

$$\overline{E} = \{(u, v): \ u, v \in V \ \text{ and } \ (u, v) \notin E\}$$

<u>הוכחת הטענה:</u> נתאר רדוקציית מיפוי פולינומית מ- IS ל- $CLIQUE$:

<u>$f(x)$</u>:

- אם x לא קידוד חוקי של גרף+מספר אז נחזיר $z \notin CLIQUE$ כלשהו (למשל $z = x$)
- אם $x = \langle G = (V, E), k \rangle$ קידוד חוקי אז נחזיר $\langle \overline{G} = (V, \overline{E}), k \rangle$

כתיב מקוצר ולא מלא: $f(\langle G, k \rangle) = \langle \overline{G}, k \rangle$.
צורת הכתיבה הזאת לא מטפלת בקלטים לא חוקיים. לשם פשטות מעכשיו בהרצאות נרשה לעצמנו להשתמש בכתיב המקוצר הזה (לא בתרגילים/מבחנים!)

כתיב עוד יותר מקוצר: $f(G, k) = (\overline{G}, k)$.

מדוע הרדורציה נכונה? קב' צמתים S ב- G היא ב"ת אם"ם S קליק ב- $\overline{G}$.
הרדורציה בבירור חשיבה פולינומית.

<u>טענה:</u> אם $A \leq_p B$ ו- $B \in P$ אזי $A \in P$.

<u>הוכחה:</u> תהי M_B מ"ט המכריעה את B. נבנה M_A המכריעה את A:
לכל קלט x, המ"ט M_A מחשבת את $f(x)$, מריצה $M_B(f(x))$ ועונה כמוה.

נכונות – נובעת מההגדרות.
זמן ריצה – אם M_B רצה בזמן $p_B(n)$ ו- f בזמן $p_f(n)$ אזי M_A רצה בזמן $O\left(p_f(n) + p_B\left(p_f(n)\right)\right)$.
מ.ש.ל.

<u>הגדרות:</u>
(1) שפה L היא <u>*NP*-קשה</u> אם לכל $L' \in NP$ מתקיים $L' \leq_p L$
(2) שפה L היא אם היא <u>*NP*-שלמה</u> וגם $L \in NP$

<u>סימון:</u> נסמן ב- *NPC* את מחלקת השפות ה- *NP*-שלמות.

<u>מסקנה מהטענה הקודמת:</u> אם קיימת $L \in NPC$ כך ש- $L \in P$ אז $P = NP$.

שפה ראשונה ב- *NPC*

<u>הגדרה:</u>

$$ACC_{NP} = \left\{ \langle M, x, 1^t \rangle : \begin{array}{c} M \text{ מ"ט וקיים } w \text{ כך ש} \\ M(x,w) \text{ מקבלת בזמן } t \end{array} \right\}$$

<u>טענה:</u> $ACC_{NP} \in NPC$

<u>הוכחה:</u>

<u>ACC_{NP} היא *NP*-קשה:</u>

תהי $L \in NP$ ויהי V_L מוודא פולינומי עבור L עם זמן ריצה $p(n)$.
נגדיר

$$f(x) = \langle V_L, x, 1^{p(|x|)} \rangle$$

מתקיים:
$x \in L$ אם"ם קיים w כך ש- $V_L(x,w)$ מקבל תוך $p(|x|)$ צעדים, מה שקורה אם"ם
$f(x) = \langle V_L, x, 1^{p(|x|)} \rangle \in ACC_{NP}$

$\underline{ACC_{NP}}$ שייכת ל *NP*: נגדיר מוודא:

$\underline{\boldsymbol{V(\langle M, x, 1^t\rangle, w)}}$:

- נסמלץ t צעדים של $M(x,w)$ ונקבל אם"ם M קיבלה.

נכונות נובעת מההגדרות

- עבור $\langle M, x, 1^t\rangle \in ACC_{NP}$, לפי הגדרת ACC_{NP}, קיים $w_{M,x}$ כך ש- $M(x, w_{M,x})$ מקבלת בזמן t. עבור ה $w_{M,x}$ הזה מתקיים ש- $V(\langle M, x, 1^t\rangle, w_{M,x})$ מקבלת.
- עבור $\langle M, x, 1^t\rangle \notin ACC_{NP}$, לפי הגדרת ACC_{NP}, <u>לא</u> קיים w כך ש- $M(x, w_{M,x})$ מקבלת בזמן t. לכן <u>לכל</u> w נקבל ש- $V(\langle M, x, 1^t\rangle, w)$ דוחה.

זמן ריצה: סימלוץ t צעדים של $M(x)$ הוא פולינומי ב t וב- $\langle M\rangle$.

הערה: אפשר לבצע את הסימולציה הזאת ע"י U_{timer} שדיברנו עליה בשיעור שעבר בזמן $\text{poly}(|\langle M\rangle|) \cdot t \log t$
אבל מכיוון שעכשיו אנחנו פחות עם האצבע על הדופק מבחינת זמן הריצה (כל מה שאכפת לנו כרגע זה זמן פולינומי ב t) אז אפשר להצדיק את זמן הריצה שלנו בצורה הרבה יותר קלה, למשל ע"י סימולציה עם מ"ט מרובת סרטים, או אפילו סימולציה בפייתון: מה יהיה זמן הריצה של תוכנית בפייתון אשר בהינתן קלטים $\langle M, x, 1^t\rangle, w$ מסמלצת את $M(x,w)$ למשך t צעדים? כל אחד מצעדי החישוב נוכל לבצע ע"י מציאת המעבר המתאים בתיאור של $\langle M\rangle$ ולכן סה"כ זמן ריצה $\text{poly}(|\langle M\rangle|, t)$ ולכן גם זמן הריצה של V יהיה $\text{poly}(|\langle M\rangle|, t)$.

מ.ש.ל.

עכשיו כשאנחנו יודעים ש- ACC_{NP} היא *NP*-קשה, היינו רוצים להשתמש בה "כעוגן" על מנת להראות ששפות נוספות הן *NP*-קשות.

<u>טענה:</u> אם $A \leq_p B$ ו- A היא *NP*-קשה אז גם B היא *NP*-קשה.

<u>הוכחה:</u> עלינו להראות כי לכל $L \in NP$ מתקיים $L \leq_p B$.
אז תהי $L \in NP$. מכיוון ש- A היא *NP*-קשה, קיימת רדורציה פולינומית f_{LA} מ- L ל- A.
בנוסף, נתון כי קיימת רדוקציה פולינומית f_{AB} מ- A ל- B.
נגדיר:

$$f_{LB}(x) = f_{AB}(f_{LA}(x))$$

מה זמן הריצה?
נסמן ב- p_{LA} וב- p_{AB} את הפולינומים החוסמים את זמני הריצה של חישוב f_{LA}, f_{AB} בהתאמה. לכן חישוב $y = f_{LA}(x)$ אורך זמן $p_{LA}(|x|)$. בפרט ז"א ש- $|y| \leq p_{LA}(|x|)$ כי בזמן t אפשר לרשום לכל היותר t תווים.
לכן, חישוב $f_{AB}(y)$ אורך לכל היותר $p_{AB}(p_{LA}(|x|)) \leq p_{AB}(|y|)$ זמן.
סה"כ זמן ריצה

$$O(p_{LA}(|x|) + p_{AB}(p_{LA}(|x|))$$

שהוא פולינומי ב- $|x|$.

אז עכשיו כדי להראות ששפה נוספת L היא NP-קשה מספיק להראות רדוקציה מ $\mathrm{ACC}_{\mathrm{NP}}$ ל- L. הבעיה היא ש- $\mathrm{ACC}_{\mathrm{NP}}$ היא שפה "מסובכת" ויהיה לנו מאוד קשה להראות רדוקציות ממנה לבעיות "טבעית". למשל, איך אפשר להראות רדוקציה $\mathrm{ACC}_{\mathrm{NP}} \leq_{\mathrm{p}} \mathrm{CLIQUE}$? בשביל זה נהיה צריכים איכשהו לתרגם קלט מהצורה $\langle \mathrm{M}, \mathrm{x}, 1^{\mathrm{t}} \rangle$ לגרף G כך שאם ל M, x קיים עד מתאים w אז בגרף G יהיה קליק גדול. זה נראה מסובך...

השפה SAT

<u>פסוק CNF</u> מעל משתנים $x_1, \dots, x_n$ הוא נוסחה מהצורה

$$\phi = \underbrace{(x_1 \vee \overline{x}_2 \vee x_3)}_{\text{פסוקית}} \wedge (\overline{x}_3 \vee x_5 \vee x_6 \vee \overline{x}_7) \wedge (\underbrace{x_3 \vee \overline{x}_2}_{\text{ליטרלים}})$$

(משתנים בוליאניים/שלילתם)

כלומר נוסחת CNF זה "וגם" של "פסוקיות או". פורמלית,

<u>הגדרה:</u>
פסוק/נוסחת CNF מעל משתנים $x_1, \dots, x_n$ מוגדר באופן הבא:

- <u>ליטרל</u> = משתנה או שלילתו
- <u>פסוקית</u> = OR בין ליטרלים
- <u>פסוק CNF</u> = AND בין פסוקיות

<u>הגדרה:</u> פסוק CNF הוא ספיק אם קיימת השמה בוליאנית למשתנים כך שערך הנוסחה הוא 1.

<u>הגדרה:</u>

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

למשל,

$$(x_1 \vee x_2) \wedge (\overline{x}_1 \vee \overline{x}_2) \wedge (x_1 \vee \overline{x}_2) \wedge (\overline{x}_1 \vee x_2) \notin SAT$$

$$(x_1 \vee x_2) \wedge (\overline{x}_1 \vee \overline{x}_2) \wedge (x_1 \vee \overline{x}_2) \in SAT$$

<u>משפט:</u> $SAT \in NPC$

נשים לב: זה ש- $SAT \in NP$ זה ברור – העד יהיה השמה מספקת. החלק הקשה במשפט הזה הוא להוכיח ש SAT היא NP-קשה.

הערה היסטורית: המשפט הזה הוכח במקביל ע"י קוק בארה"ב ולוין בברה"מ בתחילת שנות ה 70, אך זה התגלה רק כמה שנים מאוחר יותר.

את המשפט הזה נוכיח בהרצאה הבאה. עכשיו נשתמש בו כדי להוכיח (בעזרת רדוקציות) ששפות נוספות הן NP-שלמות.

<u>הגדרה:</u> נוסחת kCNF היא נוסחת CNF בה בכל פסוקית יש k ליטרלים (לא בהכרח שונים).

דוגמא לנוסחת 3CNF: $(x_1 \vee x_2 \vee x_3) \wedge (\overline{x}_1 \vee x_2 \vee \overline{x}_1)$

הערה: בתרגולים/עבודות בית נבחן וריאנט של ההגדרה הזו בו אסורות חזרות בתוך אותה פסוקית.

<u>הגדרה:</u>

$$kSAT = \{\langle \phi \rangle : \text{ספיקה kCNF היא נוסחת } \phi\}$$

<u>טענה:</u> $SAT \leq_p 3SAT$

הערות:
- זה מראה ש- 3SAT היא NP-קשה. מכיוון שהיא בבירור ב NP, זה מראה שהיא NP-שלמה
- נשים לב ש- 3SAT היא מקרה פרטי של SAT. למרות זאת, הטענה הזאת מראה שהיא קשה כמו SAT!

<u>הוכחה:</u> נראה רדוקציה המקבלת פסוק CNF ϕ ומחזירה פסוק 3CNF ψ.
הרעיון: נפצל פסוקיות ארוכות לפסוקיות באורך 3 ע"י הוספת משתנים.

דוגמא לקיצור פסוקיות:

$$(x_1 \vee x_2 \vee x_3 \vee x_4) \Longrightarrow (x_1 \vee x_2 \vee t) \wedge (\overline{t} \vee x_3 \vee x_4)$$

הסבר: המשתנה החדש t לא יכול "לעזור" לשתי הפסוקיות אלא רק לאחת מהן, והשנייה צריכה "להסתדר" בכוחות עצמה...

תיאור פורמלי של הרדוקציה:
בהינתן פסוק

$$\phi = c_1 \wedge c_2 \wedge \cdots \wedge c_m$$

כאשר

$$c_i = \ell_{i,1} \vee \ell_{i,2} \vee \cdots \vee \ell_{i,n_i}$$

נבנה פסוק

$$\psi = \bigwedge_i \psi_i$$

כאשר כל ψ_i הוא <u>פסוק</u> CNF3 (נשים לב כי ψ הוא אכן פסוק CNF3 במקרה זה)

תיאור ψ_i:

$$\psi_i = (\ell_{i,1} \vee \ell_{i,2} \vee t_{i,1}) \wedge (\overline{t}_{i,1} \vee \ell_{i,3} \vee t_{i,2}) \wedge (\overline{t}_{i,2} \vee \ell_{i,4} \vee t_{i,3}) \wedge \cdots \wedge (\overline{t}_{i,n_i-3} \vee \ell_{i,n_i-1} \vee \ell_{i,n_i})$$

כאשר $t_{i,1}, t_{i,2}, \ldots, t_{i,n_i-3}$ הם משתנים חדשים המופיעים רק ב ψ_i.

(הערה: טכנית הנחנו כאן ש- $n_i \geq 3$, כלומר שבפסוקית c_i יש לפחות 3 ליטרלים. זה בלי הגבלת הכלליות כי אחרת אפשר פשוט להשאיר את c_i כמו שהיא אם היא באורך 3 או להאריך אותה באופן מלאכותי אם היא קצרה יותר ע"י חזרות)

ניתן לבנות את ψ בזמן פולינומי. כלומר – הרדוקציה יעילה.

נכונות: כיוון 1: נניח $\phi \in SAT$.
כלומר קיימת הצבה המספקת את ϕ ובפרט מספקת כל פסוקית c_i של ϕ.
נרחיב את ההצבה הזאת להצבה שתספק כל ψ_i ולכן תספק את $\psi = \wedge \psi_i$.

מזה נרחיב? את ההצבות של המשתנים המקוריים נשאיר כמו שהם ורק נוסיף הצבות עבור כל ה-t-ים.

מכיוון שכל משתנה חדש מופיע ב- ψ_i יחיד, אז אפשר לטפל בכל ψ_i בנפרד.

ההצבה מספקת את c_i כלומר מספקת לפחות ליטרל אחד $\ell_{i,j}$ ב- c_i.

בציור:
$$\psi_i = (\ell_{i,1} \vee \ell_{i,2} \vee t_{i,1}) \wedge \cdots \wedge (\overline{t}_{i,j-2} \vee \ell_{i,j} \vee t_{i,j-1}) \wedge \cdots \wedge (\overline{t}_{i,n_i-3} \vee \ell_{i,n_i-1} \vee \ell_{i,n_i})$$

הפסוקית האדומה מסתפקת "בזכות" $\ell_{i,j}$

נגדיר הצבה מורחבת באופן הבא:
- $t_{i,s} = T$ לכל $1 \le s \le j-2$
- $t_{i,s} = F$ לכל $j-1 \le s \le n_i - 2$

ההצבה הזאת מספקת את $(\overline{t}_{i,j-2} \vee \ell_{i,j} \vee t_{i,j-1})$ כי $\ell_{i,j}$ מסתפק. כל הפסוקיות עם אינדקס קטן יותר מסתפקות כי $t_{i,s}$ מסתפק. כל הפסוקיות עם אינדקס גדול יותר מסתפקות כי $\overline{t}_{i,s}$ מסתפק.

לכן $f(\phi) = \psi \in 3SAT$.

כיוון 2: נניח $f(\phi) = \psi \in 3SAT$

כלומר קיימת הצבה למשתני ψ המספקת את ϕ.
מספיק להראות שהיא מספקת כל c_i ולכן מספקת את $\phi = \wedge c_i$:

- אם $t_{i,1} = F$ אזי $\ell_{i,1} \vee \ell_{i,2} = T$. כלומר ההצבה מספקת את c_i
- אם $t_{i,n_i-3} = T$ אזי $\ell_{i,n_i-1} \vee \ell_{i,n_i} = T$ ושוב ההצבה מספקת את c_i
- אחרת $t_{i,1} = T$ וגם $t_{i,n_i-3} = F$. לכן קיים אינדקס j כך ש- $t_{i,j} = T$ וגם $t_{i,j+1} = F$.
  נסתכל על הפסוקית (שמסתפקת):
  $$(\overline{t}_{i,j} \vee \ell_{i,j+2} \vee t_{i,j+1})$$
  לכן $\ell_{i,j+2} = T$ ושוב הפסוקית c_i מסתפקת.

כלומר בשלושת המקרים c_i מסתפקת ולכן כל ϕ מסתפק ולכן $\phi \in SAT$.

מ.ש.ל.