מבוא לחלק I

בעיית העצירה:

בהינתן קוד M וקלט x, האם M מסיימת את ריצתה על X?

הוכחה שזוהי בעיה לא פתירה:

נניח בשלילה שיש תוכנית A שפותרת את הבעיה. נבנה תוכנית B שעל קלט w פועלת כך: תריץ את A על הקלט (w,w) (כלומר: W הוא גם המכונה וגם הקלט של המכונה).

אם $A(w,w)=true$ אז B נכנסת ללולאה אינסופית.

אם $A(w,w)=false$ אז B עוצרת מייד.

עבור הקלט B: אם B עצרה, סימן ש- $A(B,B)=false$ (A טוענת ש- B לא עוצרת), סתירה.

אם B לא עצרה, אז $A(B,B)=true$ (A טוענת ש- B עוצרת), שוב סתירה.

לכן, לא קיימת תוכנית A שפותרת את בעיית העצירה.

דוגמא לבעיות לא פתירות:

- בעיית העצירה.
- בדיקת שקילות של תוכניות.
- בדיקת שקילות של דקדוקים חסרי הקשר.
- בהינתן אוסף (סופי) של מטריצות ריבועיות, האם קיימת מכפלה שנותנת את מטריצת האפס?

מכונת טיורינג:

מכונת טיורינג בנויה מ-3 מרכיבים: סרט המחולק לתאים שכל אחד מהם יכול להכיל תו בודד, כך שהוא אינסופי לכיוון ימין אך סופי לכיוון שמאל (ניתן למספר את תאיו ב-0,1,2,...); ראש קורא וכותב הנע על הסרט ויכול לנוע לכל היותר צעד אחד ימינה או שמאלה; וקבוצת מצבי בקרה סופית. נראה בהמשך שכוחו של המודל לא ישתנה באופן מהותי אם נרשה מספר רב של סרטים, סרט אינסופי לשני הכיוונים, ראש שיכול לנוע מספר צעדים לצדדים וכן הלאה. לעומת זאת, סופיותה של קבוצת מצבי הבקרה היא קריטית.

חישוב כל המכונה מתבצע כך: המכונה מתחילה את החישוב כאשר הראש נמצא בתא השמאלי ביותר, ועל הסרט כתובה מילה סופית כלשהי החל מהתא השמאלי ביותר, ושאר הסרט פרט למילה ריק. המכונה מבצעת סדרה של צעדים כאשר בכל צעד היא פועלת בהתאם למצב הבקרה הנוכחי שלה ולתוכן התא שמעליו נמצא הראש הקורא. בכל צעד המכונה יכולה לשנות את תוכן התא שמעליו נמצא הראש הקורא, לשנות את מצב הבקרה שלה, ולהזיז את הראש הקורא צעד אחד ימינה או שמאלה (או להותיר אותו במקום). המכונה מסיימת את החישוב שלה אם היא נכנסת למצב בקרה המסומן כמצב סופי, ובמקרה זה הפלט שלה על מילת הקלט הוא המילה שכתובה החל מהתא השמאלי ביותר בסרט ועד לתא שמשמאל לראש (אם הראש נמצא בתא השמאלי ביותר, הפלט הוא "המילה הריקה"; הגדרת הפלט מהונדסת בכוונה כדי להבטיח שניתן יהיה להוציא את המילה הריקה כפלט).

הגדרת המכונה: זוהי שביעייה $M=(Q,q_0,F,\Gamma,\Sigma,b,\delta)$.

Q: קבוצה סופית של מצבי בקרה (אם היא לא תהיה סופית, היא תוכל לעשות כל חישוב).

- $q_0 \in Q$: מצב הבקרה ההתחלתי.
- $F \subseteq Q$: קבוצת מצבי הבקרה הסופיים (כאשר המכונה נכנסת אליהם, החישוב נגמר).

Γ : קבוצה סופית של תווים (א"ב העבודה/ א"ב הסרט).

- $\Sigma \subseteq \Gamma$: א"ב הקלט.
- $b \in \Gamma \setminus \Sigma$ נקרא רווח או בלנק (תו לסימון תא ריק). במפורש דורשים ש- b לא תהיה חלק מא"ב הקלט כדי שיהיה ניתן לקבוע חד-משמעית היכן הקלט נגמר.

$\delta : (Q \setminus F) \times \Gamma \to Q \times \Gamma \times \{L, R, S\}$: פונקצית המעברים.

דוגמא: $\delta(q,\sigma) = (p,\tau,\lambda)$ פירושו: אם המכונה במצב q ורואה σ אז היא:

עוברת למצב הבקרה p , שמה τ על הסרט במקום σ והולכת עם הראש בכיוון λ .

קונפיגורציות:

האופן הנוח ביותר לתיאור החישוב שמבצעת מכונה הוא באמצעות קונפיגורציות, קונפיגורציה היא מעין צילום מסך של החישוב שכולל את כל האינפורמציה על אותו רגע.

הגדרה קונפיגורציה של מכונה M היא שלשה כך ש: $C = [q,i,w]$ כאשר:

$q \in Q$: מצב הבקרה הנוכחי של M . $i \in \mathbb{N}$: מיקום הראש. $w \in \Gamma^*$: תוכן הסרט.

קונפיגורציה ההתחלתית של M על הקלט x היא: $C = [q_0, 0, x]$.

קונפיגורציה סופית: קונפיגורציה שבה $q \in F$. במקרה זה פלט המכונה הוא: $w[0 \cdots i-1]$ (תוכן w מתא 0 ועד לתא $i-1$).

צעד חישוב: לכל קונפיגורציה נקבעת הקונפיגורציה העוקבת שלה באופן יחיד ע"פ פונקצית המעברים δ . אם הקונפיגורציה הנוכחית היא: (q,i,w) ואם $\delta(q,w_i) = (p,b,d)$ אז הקונפיגורציה הבאה מוגדרת להיות: (p, j, w') כאשר w' זהה ל- w פרט לכך ש- $w'[i] = b$.

- המצב q יוחלף במצב p .
- האות w_i תוחלף באות b .
- מיקום הראש ישתנה: אם $d = R$ אז: $i+1$.

אם $d = S$ אז: i .

אם $d = L$ אז: $\max\{1.i-1\}$.

מסקנה: לכל קונפיגורציה לא סופית C_1 קיימת קונפיגורציה עוקבת יחידה C_2 . סימון: $C_1 \vdash C_2$.

הערה: באותו אופן אומרים ש- $C_1 \vdash^* C_2$ אם יש סדרת צעדים שמובילה מ- C_1 ל- C_2 .

חישוב: החישוב של מ"ט M על קלט x הוא סדרת קונפיגורציות- $C_0, C_1, C_2, \cdots$ המקיימת: לכל $i > 0$, $C_{i-1} \vdash C_i$. אם הסדרה סופית את הקונפיגורציה האחרונה היא קונפיגורציה סופית.

הפונקציה של מ"ט: הפונקציה שמ"ט M מחשבת מסומנת: $f_M : \Sigma^* \to \Gamma^*$ ומוגדרת באופן הבא:

- אם החישוב של M על x מסתיים ו- $C = (q, i, w)$ היא הקונפיגורציה הסופית אזי:
  $f_M(x) = w_1 w_2 \cdots w_{i-1}$.
- אחרת (אם M לא עוצרת על x) $f_M(x)$ לא מוגדרת!

פונקציה מלאה: נאמר שפונקציה f היא מלאה אם היא מוגדרת לכל קלט. יכולות להיות פונקציות לא מלאות.

דוגמא: נבנה מ"ט שמחשבת את הפונקציה: $f(x) = 0x$ כאשר $x \in \{0,1\}^*$. יש יותר מדרך אחת לבנות את המכונה, נציע דרך שמבצעת את המשימה במעבר אחד- כשבכל פעם "נזכור" את התו הבא שצריך לרשום. נגדיר: $M = (Q, q_0, F, \Gamma, b, \delta)$ כאשר:

$$Q = \{q_0, q_1, q_2\},\ F = \{q_2\},\ \Sigma = \{0,1\},\ \Gamma = \{0,1,b\}$$

משמעות המצבים: q_0 מצב שזוכר 0 לכתיבה בצעד הבא.

q_1 מצב שזוכר 1 לכתיבה בצעד הבא.

פונקצית המעברים δ תתואר ע"י טבלה:

	0	1	b
q_0	$(q_0, 0, R)$	$(q_1, 0, R)$	$(q_2, 0, S)$
q_1	$(q_1, 1, R)$	$(q_1 1, R)$	$(q_2, 0, S)$

טענה: החישוב של M על x הוא סדרה של $n+2$ קונפיגורציות-

$$C_0 = (q_0, 1, x),\ C_i = (q x_i, i+1, 0 x_1 \cdots x_{i-1} x_{i+1} \cdots x_n),\ C_{n+1} = (q_2, n+2, 0x).$$

שקילות של מודלים

מודל חישוב: אוסף של אובייקטים כך שלכל אובייקט מתאימה הפונקציה שהוא מחשב.

מודלים שקולים: שני מודלים ייקראו שקולים אם אוסף הפונקציות המחושבות על-ידם זהה.

דוגמא- מ"ט זריזה: מוגדר בדומה למ"ט רגילה בתוספת שני מצבים של הראש: LL, RR שהם הליכת שני צעדים ימינה או שמאלה.

טענה: מודל מ"ט זריזה ≡ מ"ט רגילה.

הוכחה:

נוכיח שקילות ע"י הכלה דו-כיוונית:

כיוון-1: בהינתן מ"ט רגילה M המחשבת f_M, קיימת מ"ט זריזה המחשבת f_M - אותה M (מ"ט רגילה היא מקרה פרטי של מ"ט זריזה).

כיוון-2: בהינתן מ"ט זריזה M המחשבת f_M, נבנה מ"ט רגילה M' שמחשבת את אותה הפונקציה באופן הבא: $Q' = Q \cup Q_L \cup Q_R$ $Q_R = \{q_R \mid q \in Q\}$ $Q_L = \{q_L \mid q \in Q\}$, כלומר, הוספת מצבים שזוכרים ללכת צעד נוסף ימינה או צעד נוסף שמאלה לפני הגעה למצב.

$$\delta'(q_L, a) = (q, a, L) \quad \delta'(q_R, a) = (q, a, R)$$

לכל $q \in Q$, $a \in \Gamma$, אם $\delta(q,a) = (p,b,d)$ אז:

אם $d \in \{L, R, S\}$ אז: $\delta'(q,a) = (p,b,d)$

אם $d = LL$ אז: $\delta'(q,a) = (p_L, b, L)$

אם $d = RR$ אז: $\delta'(q,a) = (p_R, b, R)$.

דוגמא- מ"ט דו סרטית: מ"ט שיש לה גישה לשני סרטים, בכל סרט ראש קורא-כותב, כאשר מיקום ראש אחד על סרט אחד לא תלוי במיקום הראש השני על הסרט השני.

אתחול: סרט אחד מכיל את הקלט ואחריו- b, הסרט השני מכיל רק- b. שני הראשים במקום ה-1.

פלט: משאל לראש בסרט העליון, בזמן העצירה.

פונקציית המעברים: $\delta : (Q \setminus F) \times \Gamma^2 \to Q \times \Gamma^2 \times \{L, R, S\}^2$, היא מבוססת על שתי האותיות (משני הסרטים) ואומרת מה לכתוב בכל אחד מהם, לאיזה מצב לעבור ולאן להזיז כל אחד מהראשים.

טענה: מודל מ"ט k-סרטית ≡ מ"ט רגילה.

התזה של Church:

כל מודל כללי וסביר של חישוב שקול למ"ט.

כללי- מודל חזק לפחות כמו מ"ט, סביר- לכל אובייקט במודל יש תיאור סופי.

מ"ט אוניברסאלית: נרצה לבנות מ"ט שתוכל לבצע את המשימה של כל מ"ט אחרת, מ"ט כזו צריכה לקבל כקלט הן את המכונה עצמה- M והן את הקלט x, ולהחזר כפלט את: $f_M(x)$.

קידוד של מחרוזת: A - א"ב. בה"כ $A = \{1, 2, \cdots, |A|\}$. ואם $x = x_1 x_2 \cdots x_n$ מחרוזת מעל A^* הקידוד של x: $\langle x \rangle = 1^{x_1} 0 1^{x_2} 0 \cdots 1^{x_n} 0$.

קידוד של מ"ט: תהי $M=(Q,q_0,F,\Gamma,\Sigma,b,\delta)$, בה"כ: $Q=\{1,2,\cdots,|Q|\}$,

$q_0=1$, $F=\{2,3\}$, $\Gamma=\{1,2,\cdots,|\Gamma|\}$, $\Sigma=\{1,2,\cdots,|\Sigma|\}\subset\Gamma$. $(L,R,S)\equiv(1,2,3)$

קידוד פונקצית המעברים: אם $\delta(q,a)=(p,b,d)$ נקודד: $\langle\delta(q,a)\rangle=\langle qapbd\rangle=1^p01^b01^d0$

קידוד המכונה: $\langle M\rangle=00\langle\delta(1,1)\rangle0\langle\delta(1,2)\rangle0\cdots0\langle\delta(|Q|,|\Gamma|)\rangle00$

הערות:

- בהינתן קידוד של מחרוזת או של מ"ט, קל לבדוק תקינות ולשחזר את המחרוזת/מ"ט.
- מוסכמה: כל מחרוזת בינארית שאיננה "חוקית", נתייחס אליה כקידוד של מ"ט M_{STAM} שעוברת מייד למצב 3 ועוצרת: $\forall q,a:\delta(q,a)=(3,a,s)$.

ולכן, כל מחרוזת בינארית מתארת מ"ט.

קידוד של קונפיגורציה: תהי $C=(\alpha,q,i)$ ו- $\alpha=\alpha_1\alpha_2\cdots\alpha_m$, אז:

$$\langle C\rangle=\langle a_1a_2\cdots a_{i-1}\rangle01^q0\langle a_ia_{i+1}\cdots a_m\rangle00,\quad \langle\varepsilon\rangle=0$$

הערה: קידוד ה- α-ים מסתיים ב-0 ולכן ישנם שני אפסים לפני ה- 1^q ובאותו אופן יש 3 אפסים בסוף.

נגדיר פונקציה:

$$next(\langle M\rangle,\langle c\rangle)=\begin{cases}0 & \text{אם } c \text{ קונפיגורציה סופית / } \langle c\rangle \text{ / } M \text{ "לא מתאימה" ל- לא חוקית}\\ \langle M\rangle,\langle c'\rangle & \text{אחרת, תהי } c' \text{ הקונפיגורציה העוקבת ל- } c\end{cases}$$

טענה: הפונקציה $next$ ניתנת לחישוב, כלומר: קיימת M_{next} שמחשבת אותה.

הוכחה:

M_{next} על קלט $\langle M\rangle,\langle c\rangle$:

- בדוק תקינות $\langle M\rangle,\langle c\rangle$ (ע"פ הגדרת $next$). אם לא תקין, עצור עם פלט 0.
- אם תקין, מצא את q ואת האות הנוכחית a בתוך $\langle c\rangle$ (ע"י 00 בקידוד).
- מצא את $\delta(q,a)$ בתוך הקידוד $\langle M\rangle$ וחלץ מההמשך את p,b,d, שנה את c ל- c' בהתאם.
- אם $\langle M\rangle$ מתאים למ"ט M_{STAM}, עבור לקונפיגורציה הסופית של M_{STAM}.

הפונקציה האוניברסאלית:

$$U(\langle M\rangle,\langle x\rangle)=\begin{cases}\langle f_M(x)\rangle & \langle x\rangle \text{ חוקי ו- } x \text{ מתאים ל- } M \text{ וחישוב } M \text{ על } x \text{ מסתיים}\\ undefined & \text{אחרת}\end{cases}$$

טענה: הפונקציה האוניברסאלית U, ניתנת לחישוב.

הוכחה:

- אם $\langle x\rangle$ לא חוקי או לא מתאים ל- $\langle M\rangle$ - בצע לולאה אינסופית.
- כתוב על הסרט השני את $\langle M\rangle,\langle c\rangle$ כאשר $\langle c\rangle$ הקונפיגורציה ההתחלתית של M על x.

- כל עוד c לא קונפיגורציה סופית, חשב $\langle M \rangle, \langle c \rangle \leftarrow next(\langle M \rangle, \langle c \rangle)$.
- אם c קונפיגורציה סופית, הוצא את הפלט שלה- $f_M(x)$ (מה שמשאל לראש ב-$\langle c \rangle$).

<u>הערות:</u>

- כל מ"ט שמחשבת את U תיקרא מ"ט אוניברסאלית.
- ל- M_U יש איזשהו מספר מצבים ואיזשהו מספר אותיות ללא קשר למ"ט M שהיא מקבלת כקלט.
- אם M לא עוצרת על x, גם M_U לא עוצרת על $\langle M \rangle, \langle x \rangle$.

בעיות הכרעה

שפה: אוסף סופי או אינסופי של מחרוזות מתוך Σ^*.

הגדרה: מ"ט לזיהוי שפות היא מ"ט רגילה M שמקיימת $F = \{q_A, q_R\}$.

אומרים שמ"ט מקבלת את הקלט x אם M עוצרת על x במצב q_A.

אומרים שמ"ט דוחה קלט x אם M עוצרת על x במצב q_R.

ייתכן כמובן ש- M לא עוצרת על x.

הערה: נקודד- $q_A = 2,\ q_R = 3$.

הגדרה: השפה ש- M מקבלת מסומנת- $L(M) = \{x \mid M\ accepts\ x\}$.

אומרים ש- M מכריעה שפה L אם $L = L(M)$ ובנוסף היא עוצרת לכל קלט.

דוגמאות לשפות הניתנות להכרעה:

- Σ^* (מ"ט שמייד עוברת ל- q_A לכל קלט- $\delta(q,a) = (q_A, a, s)$).
- $\varnothing$ (באופן דומה- מ"ט שמייד עוברת ל- q_R לכל קלט).
- כל שפה סופית.
- שפת כל הגרפים הקשירים.

מחלקות של שפות:

ברירת מחדל: $\Sigma = \{0,1\}$ (מוסכמה).

{ קיימת מ"ט המכריעה את L $\quad R = \{L \subseteq \Sigma^* \mid$

{ קיימת מ"ט המקבלת את L $\quad RE = \{L \subseteq \Sigma^* \mid$

{ קיימת מ"ט ש: $x \in L$ - מקבלת/ לא עוצרת, $x \notin L$ -דוחה $\quad Co-RE = \{L \mid \overline{L} \in RE\} = \{L \mid$

תכונות:

- $R \subseteq RE$ (ב- R יש שפות ב- RE שקיימת להן מ"ט שגם עוצרת לכל קלט).
- R סגורה למשלים (אם $L \in R$ אז גם $\overline{L} \in R$).
- R סגורה לאיחוד (אם $L_1, L_2 \in R$ אז גם $L_1 \cup L_2 \in R$).
- RE סגורה לאיחוד.
- $RE \cap Co-RE = R$.

הוכחה:

כיוון-1: אם $L \in R$ אז היא בהכרח ב- RE וב- $Co-RE$ (נובע מההגדרה).

כיוון-2: תהא $L \in RE \cap Co-RE$.

$L \in RE \leftarrow$ קיימת מ"ט M_1 כך ש- $x \in L \leftarrow M_1$ עוצרת ומקבלת את x.

$x \notin L \leftarrow M_1$ עוצרת ודוחה או לא עוצרת על x.

$L \in C-RE \leftarrow$ קיימת מ"ט M_2 כך ש: $x \notin L \leftarrow M_2$ עוצרת ודוחה את x

$x \in L \leftarrow M_2$ עוצרת ומקבלת או לא עוצרת על x.

נבנה מ"ט M שמכריעה את L: בהינתן קלט x, נריץ במקביל את M_1 ו- M_2 על x. הראשונה שתעצור, נקבל או נדחה כמוה.

<u>הערות:</u>

- M עוצרת תמיד.
- M_1, M_2 כשהן עוצרות- התשובות שלהן נכונות.

מכל אלה נובע: $L = L(M)$.

<u>טענה:</u> $L, \overline{L} \in RE \leftarrow L \in R$.

<u>דוגמאות לשפות ב-RE:</u>

- כל שפה ב- R.
- שפת העצירה: $HP = \{\langle M \rangle, \langle x \rangle \mid$ M <u>עוצרת</u> על $x\}$
- השפה האוניברסאלית: $L_u = \{\langle M \rangle, \langle x \rangle \mid$ M <u>מקבלת</u> את $x\}$
- שפת האלכסון: $L_D = \{\langle M \rangle \mid$ M מקבלת את $\langle M \rangle\} = \{\langle M \rangle \mid \langle M \rangle \in L(M)\}$.

<u>טענה:</u> $HP, L_u, L_D \in RE$.

<u>הערה:</u> בהמשך הקורס נראה שכל השפות הללו גם לא נמצאות ב- R.

רדוקציה

רדוקציה היא פיתרון לבעיה אחת ע"י אלגוריתם שפותר בעיה אחרת.

הגדרה: תהיינה $L_1, L_2 \subseteq \Sigma^*$ שתי שפות. אומרים שפונקציה $f : \Sigma^* \to \Sigma^*$ היא רדוקציה מ- L_1 ל- L_2 אם:

1. f היא פונקציה מלאה (מוגדרת לכל קלט).
2. f ניתנת לחישוב (קיימת מ"ט M_f שמחשבת את f).
3. (תקפות) $x \in L_1 \Leftrightarrow f(x) \in L_2$.

סימון: $L_1 \le L_2$.

הערה: ברדוקציה אין "הרצה" של המכונות. יש ייצור של קוד אחד מתוך קוד אחר ("קומפילציה").

טענה: $L_D \le L_u$.

הוכחה: פונקציה f מתאימה: $f(\langle M \rangle) = (\langle M \rangle, \langle M \rangle)$. היא מלאה וניתנת לחישוב.

תקפות: $f(\langle M \rangle) \in L_u \leftrightarrow (\langle M \rangle, \langle M \rangle) \in L_u \leftrightarrow \langle M \rangle \in L(M) \leftrightarrow \langle M \rangle \in L_D$.

טענה: $L_u \le HP$.

הוכחה: פונקציה f מתאימה: $f(\langle M \rangle, \langle x \rangle) = (\langle A \rangle, \langle x \rangle)$. היא מלאה וניתנת לחישוב.

המכונה A זהה ל- M למעט: אם M עוברת ל- q_R אז A נכנסת ללולאה אינסופית.

תקפות: $\langle M \rangle, \langle x \rangle \in L_u \leftrightarrow M$ מקבלת את $x \leftrightarrow A$ עוצרת על $x \leftrightarrow \langle A \rangle, \langle x \rangle \in HP$.

$\leftrightarrow f(\langle M \rangle, \langle x \rangle) \in HP$.

תכונות של רדוקציות:

- לכל L מתקיים: $L \le L$.
- טרנזיטיביות: $L_1 \le L_2 \text{ and } L_2 \le L_3 \Rightarrow L_1 \le L_3$.

מסקנה: $L_D \le L_u \text{ and } L_u \le HP \Rightarrow L_D \le HP$.

טענה: $L_1 \le L_2 \Rightarrow \bar{L_1} \le \bar{L_2}$ (ע"י שימוש באותה f).

מסקנה: $\bar{L_D} \le \bar{L_u} \le \overline{HP}$.

משפט הרדוקציה:

נוסח-א: אם $L_1 \le L_2$ אזי:

1. $L_2 \in R \rightarrow L_1 \in R$.
2. $L_2 \in RE \rightarrow L_1 \in RE$.

3. $L_2 \in Co-RE \rightarrow L_1 \in Co-RE$.

נוסח-ב: אם $L_1 \leq L_2$ אזי:

1. $L_1 \notin R \rightarrow L_2 \notin R$.
2. $L_1 \notin RE \rightarrow L_2 \notin RE$.
3. $L_1 \notin Co-RE \rightarrow L_2 \notin Co-RE$.

טענה: קיימות שפות שאינן ב- RE.

הוכחה: נראה ע"י שיקולי ספירה ש- $|\{$ קבוצת כל המ"ט $\}| \leq |RE|$.

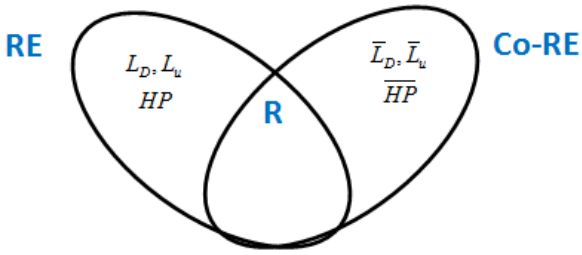
$|\{$ קבוצת כל השפות $\}| \geq |\{$ ממשיים בקטע $[0,1]\ \}| = |\square| \leq |\{0,1\}^*| \leq |RE|$.

הערה: $|\Sigma^*| = \aleph_0$, ומכיוון שכל מ"ט שקולה לקידוד בינארי מעל Σ, מספר המ"ט הוא בן-מנייה.
קבוצת כל השפות מעל Σ היא קבוצת החזקה של Σ^* בעוצמה: $2^{\aleph_0}$. ומכיוון שלכל מ"ט מתאימה שפה אחת, לא קיימות מספיק מ"ט לקבל את כל השפות הקיימות ולכן ישנן שפות שלא ב- RE.

טענה: $\overline{L}_D \notin R$. טענה מחוזקת: $\overline{L}_D \notin RE$.

הוכחה-1: נניח בשלילה ש- $\overline{L}_D \in RE$ אז קיימת מ"ט מתאימה M_D שמקיימת: $L(M_D) = \overline{L}_D$.
$\langle M_D \rangle \in \overline{L}_D \leftrightarrow \langle M_D \rangle \in L(M_D) \leftrightarrow M_D \in L_D \leftrightarrow M_D \notin \overline{L}_D$. הגענו לסתירה ולכן לא קיימת M_D כזו.

הוכחה-2: ע"י לכסון. כל שורה בטבלה היא שפה ב- RE וכל עמודה בטבלה היא מילה הניתנת כקלט למכונה. נשים-לב כי L_D זה האלכסון של הטבלה (כאשר הקלט למכונה שווה לקידוד של המכונה).
ולכן $\overline{L}_D$ = ה-NOT של האלכסון. ולכן אף שורה בטבלה לא מייצגת אותה.



מסקנה-1: $\overline{L}_D, \overline{L}_u, \overline{HP} \notin RE$.

מסקנה-2: $L_D, L_u, HP \in RE \setminus R$.

דוגמא: $L = \{\langle M \rangle |$ M עוצרת לכל קלט $\}$

טענה: $L \notin R$.

הוכחה: נראה רדוקציה $HP \leq L$, $f(\langle M \rangle, \langle x \rangle) = \langle M_x \rangle$. M_x תפעל על הקלט w באופן הבא:
תתעלם מ- w ותריץ את M על x. f מלאה וניתנת לחישוב. תקפות: $M \leftrightarrow \langle M \rangle, \langle x \rangle \in HP$
עוצרת על $x \leftrightarrow M_x$ עוצרת לכל קלט $w \leftrightarrow \langle M_x \rangle \in L \leftrightarrow f(\langle M \rangle, \langle x \rangle) \in L$.

דוגמא: $L_{\Sigma^*} = \{\langle M \rangle | L(M) = \Sigma^*\}$.

טענה: $L_{\Sigma^*} \notin R$.

הוכחה: אותה רדוקציה f מראה- $L_u \leq L_{\Sigma^*}$.

דוגמא: $L_{EQ} = \{\langle M_1 \rangle, \langle M_2 \rangle \mid L(M_1) = L(M_2)\}$.

טענה: $L_{EQ} \notin R$.

הוכחה: נראה רדוקציה $L_{\Sigma^*} \leq L_{EQ}$, $f(\langle M \rangle) = (\langle M \rangle, \langle M_0 \rangle)$. תקפות:

$$f(\langle M \rangle) \in L_{EQ} \leftrightarrow \langle M \rangle, \langle M_0 \rangle \in L_{EQ} \leftrightarrow L(M) = L(M_0) \leftrightarrow L(M) = \Sigma^* \leftrightarrow \langle M \rangle \in L_{\Sigma^*}$$

תכונות של שפות:

תכונה של שפות: זוהי תת-קבוצה של שפות- $S \subseteq RE$.

תכונה לא טריוויאלית: תכונה המקיימת: $\varnothing \neq S \neq RE$ (תכונה שמקיימות רק חלק מן השפות).

סימון: $L_S = \{\langle M \rangle \mid L(M) \in S\}$.

דוגמאות:

- $S_1 = \{L \in RE \mid \Sigma \in L\}$.
- $S_2 = \{L \mid L \ is \ \ finite\}$.
- $S_3 = \{\Sigma^*\}$ ($L_{S_3} = \{\Sigma^*\}$).
- $S_4 = \{\varnothing\}$ (תכונה שמכילה את השפה הריקה, זה שונה מ- $S_4 = \varnothing$).

אבחנה: אם S טריוויאלית אז $L_S \in R$.

$$L_S = \{\langle M \rangle \mid L(M) \in \varnothing\} = \varnothing \in R \quad \leftarrow S = \varnothing$$
$$L_S = \{\langle M \rangle \mid L(M) \in RE\} = \Sigma^* \in R \leftarrow S = RE$$

משפט Rice: לכל S לא טריוויאלית $L_S \notin R$.

שימושים במשפט:

1. $L_\varepsilon = \{\langle M \rangle \mid \varepsilon \in L(M)\}$. נראה ש- $L_\varepsilon \notin R$.

הוכחה: התכונה המתאימה- S_1, $L_{S_1} = L_\varepsilon$ ו- S_1 תכונה לא טריוויאלית ($\varnothing \notin S_1$, $\{\varepsilon\} \in S_1$) לכן ממשפט Rice $L_\varepsilon \notin R$.

2. $L_2 = \{\langle M \rangle \mid \varepsilon \in L(M)\}$. ע"י S_2 ניתן לראות כי $L_2 \notin R$.
3. $L_{\Sigma^*} = L_{S_3} \notin R$.
4. $L_\varnothing = \{\langle M \rangle \mid L(M) = \varnothing\}$ והתכונה המתאימה: $S = \{\varnothing\}$.

הוכחת טענות מהצורה $L \notin RE$:

- ישירות (למשל- $\overline{L_D}$).
- משפט הרדוקציה
- $L \notin RE \Leftarrow \begin{cases} L \notin R \\ \overline{L} \notin RE \end{cases}$
- משפטים כלליים (למשל- משפט רייס).

דוגמאות:

- $L_{\varnothing} = \{\langle M \rangle \mid L(M) = \varnothing\}$
- $L_{\Sigma^*} = \{\langle M \rangle \mid L(M) = \Sigma^*\}$
- $L_{EQ} = \{\langle M_1 \rangle, \langle M_2 \rangle \mid L(M_1) = L(M_2)\}$

טענה: $L_{\varnothing}, L_{\Sigma^*}, \overline{L}_{\Sigma^*}, L_{EQ}, \overline{L}_{EQ} \notin RE$.

משפט Rice לשפות RE: תהי S תכונה לא טריוויאלית של שפות המקיימת $\varnothing \in S$ אזי: $L_S \notin RE$.

(תזכורת- $L_S = \{\langle M \rangle \mid L(M) \in S\}$).

הוכחה: בהוכחת משפט Rice מקרה ב'- $\overline{HP} \leq L_S \Leftarrow L_S \notin RE$.

הערות:

- התנאי הנוסף על S הוא הכרחי: למשל- $\overline{L}_{\varnothing} \in RE$ (למרות ש- S לא טריוויאלית).
- המשפט איננו אפיון לשפות המקיימות $L_S \notin RE$, למשל: $L_{\Sigma^*} \notin RE$ (למרות שהמשפט לא חל עליו כי התכונה לא מכילה את $\varnothing$).
- קיים אפיון מלא לשפות L_S המקיימות $L_S \in RE$ (לא עוברים על זה בהרצאה).

הערה מתרגיל-בית:

$$L_S = \{\langle M \rangle \mid L(M) \in coRE\} = \{\langle M \rangle \mid \overline{L(M)} \in RE\}$$

סוגים של בעיות חישוב

- בעיות הכרעה (שפות).
- פונקציות.
- בעיות חיפוש (יחסים).

עד כה דיברנו על בעיות הכרעה או שפות, כעת נעבור לדבר על פונקציות.

פונקציות:

סימון: $L_f \triangleq \{(x,y) | f(x)=y\}$

משפט: תהי $f: \Sigma^* \to \Sigma^*$, f ניתנת לחישוב $\leftrightarrow$ $L_f \in RE$.

הערות:

- אם בנוסף f מלאה אז: f ניתנת לחישוב $\leftrightarrow$ $L_f \in R$.
- קיימת f לא מלאה כך ש- $L_f \in R$. למשל: f לא מוגדרת לשום קלט $\leftarrow$ $L_f = \varnothing \in R$.

הוכחה:

כיוון-א: נניח ש- f ניתנת לחישוב ע"י מ"ט M_f ונבנה מ"ט M המקבלת את L_f.

M על הקלט (x,y):

- הרץ את M_f על x.
- נסמן ב- $f(x)$ את הפלט של הצעד הקודם, ואם $f(x)=y$, קבל.
- אחרת, דחה.

כיוון-ב: $L_f \in RE$ אז קיימת מ"ט המקבלת את f. נבנה M_f המחשבת את f.

M_f על קלט x: (הרצה מבוקרת)

- עבור $i = 1,2,3,\cdots$
- הרץ את M על הזוגות $(x,w_1),(x,w_2),\cdots,(x,w_i)$ כל אחד במשך i צעדים. אם M קיבלה (x,w_j) כלשהו, עצור עם פלט w_j, אחרת- המשך.

דוגמא:
$$f(\langle M \rangle) = \begin{cases} |L(M)| & L(M) < \infty \\ undefined & otherwise \end{cases}$$

טענה: f לא ניתנת לחישוב.

הוכחה לפי המשפט: $L_f = \{\langle M \rangle, k \mid |L(M)| = k,\ k \in \mathbb{N}\}$. מהמשפט, מספיק להראות:

$L_f \notin RE$. מראים זאת ע"י רדוקציה: $L_\varnothing \le L_f$ $g(\langle M \rangle) = \langle M \rangle, 0$.

דוגמאות נוספות:

$f_1(\langle M\rangle,\langle x\rangle)=\begin{cases}1 & M\ stops\ om\ x\\ 0 & otherwise\end{cases}$ - לא ניתנת לחישוב, כי אם כן אז: $HP\in R$.

$f_2(\langle M\rangle,\langle x\rangle)=\begin{cases}1 & M\ stops\ on\ x\\ undefined & otherwise\end{cases}$ - ניתנת לחישוב (ע"י מכונה שמראה $HP\in RE$).

$f_3(\langle M\rangle,\langle x\rangle)=\begin{cases}undefined & M\ stops\ on\ x\\ 0 & otherwise\end{cases}$ - לא ניתנת לחישוב, כי אם כן אז: $\overline{HP}\in RE$.

בעיות חיפוש:

דוגמא: $L\subseteq\Sigma^*\times\Sigma^*$, $L=\{(G,T)|G-graph,\ T-spanning\ tree\ in\ G\}$.

בעית זיהוי: נתונים- (G,T), האם הם מקיימים את היחס.

בעיית חיפוש: נתון גרף G, מצא עץ פורש T, אם קיים אחד.

הגדרה: יהא $S\subseteq\Sigma^*\times\Sigma^*$ יחס דו-מקומי.

- אומרים שבעיית הזיהוי של S ניתנת לפיתרון אם $S\in RE$.
- אומרים שבעיית החיפוש של S ניתנת לפיתרון אם קיימת מ"ט M שלכל x -
  - אם קיים y כך ש- $(x,y)\in S$, M עוצרת עם y כזה (כלשהו) כפלט.
  - אם לא קיים y כזה, M לא עוצרת.

הגדרה: בהינתן פונקציה f נגדיר: $S_f=\{(x,y)|\ f(x)=y\}$. בעיית החיפוש של S_f שקולה לבעית החישוב של f.

דוגמא- סיבוכיות קולמגורוב

אקראיות היא ההפך מחוקיות, לסדרה עם חוקיות קיים אלגוריתם (=מ"ט) קצר שמייצר את המחרוזת. ולכן הסדרה- 001001001001 נראית פחות אקראית מהסדרה-011101100010.

הגדרה: $\Gamma=\{0,1,b\}$ $\Sigma=\{0,1\}$. עבור מחרוזת $x\in\Sigma^*$ סיבוכיות קולמגורוב היא מספר המצבים הקטן ביותר של מ"ט שעל קלט ε פולטת את x. סימון: $k(x)$.

משפט: הפונקציה k לא ניתנת לחישוב.

אבחנות:

- k היא פונקציה מלאה, ולכל x $k(x)\le|x|+1$ (מצב לכתיבת כל אות ב- x ובסוף צעד ימינה).
- k היא פונקציה לא חסומה, כלומר לכל t קיים x כך ש- $k(x)>t$.

הוכחה:

נניח בשלילה ש- k ניתנת לחישוב ו- M_k מ"ט המחשבת אותה ונגיע לסתירה ב-2 שלבים:

1. נבנה מ"ט M_1 שעל קלט t מוצאת x כך ש- $k(x)>t$ (קיים כזה כי הפונקציה לא חסומה).

נעבור על כל ה- x-ים לפי סדר לקסיקוגרפי, לכל x נחשב $k(x)$ ע"י M_k (שעוצרת תמיד) עד שנגיע ל- x המבוקש (קיים כזה) ונפלוט אותו.

נסמן ב- m את מספר המצבים של M_1.

יהי n מספר מספיק גדול כך ש- $2^n - n \geq m + 3$.

2. נבנה M_2 שעל קלט ε עובדת באופן הבא:
   - כותבת 2^n על הסרט שלה באופן הבא: $1000\cdots 0$ ומחזירה את הראש להתחלה.
   - מריצה את M_1 על 2^n. נסמן ב- x את הפלט.

מספר המצבים של M_2: $n+2$ מצבים לכתיבת 2^n, 1 להחזרת הראש, m להפעלת M_1.
בסה"כ: $n+m+3$ מצבים.

לפי אופן בחירת n $\Leftarrow$ $n + m + 3 \leq 2^n$.

נסתכל על x:

- M_2 על ε פולטת x במספר מצבים הקטן מ- 2^n. ולכן: $k(x) < 2^n$
- M_1 על קלט 2^n פולטת x, ולכן: $k(x) > 2^n$.

סתירה להנחה שקיימת M_k, ולכן הפונקציה k אינה ניתנת לחישוב.

משפט: קיים פיתרון לבעית הזיהוי $\leftarrow$ קיים פיתרון לבעית החיפוש.

הוכחה: בהינתן קלט x נמצא בהרצה מבוקרת זוג (x, y) כך ש- $(x, y) \in S$ (אם קיים).

טענה: קיים פיתרון לבעית החיפוש $\not\leftarrow$ קיים פיתרון לבעית הזיהוי.

דוגמא: L_{EQ}, בעיית הזיהוי קשה ($L_{EQ} \notin RE$) אבל בעית החיפוש ניתנת לפיתרון- $(\langle M \rangle, \langle M \rangle)$.

דוגמאות:

- $S_1 = L_u = \{\langle M \rangle, \langle x \rangle | \ M \ \ accepts \ x\}$. בעיות הזיהוי והחיפוש ניתנות לפיתרון.
- $S_2 = \overline{L}_u = \{\langle M \rangle, \langle x \rangle | M \ \ does \ not \ accept \ x\}$. לא ניתנת לזיהוי ($\overline{L}_u \notin RE$). גם לא ניתנת לחיפוש כי אם קיים לבעית החיפוש פיתרון אז גם ל- $\overline{L}_{\Sigma^*}$ בסתירה לכך ש- $\overline{L}_{\Sigma^*} \notin RE$.

מבוא לחלק II

איזה מן הבעיות הניתנות לפיתרון (לפי חלק I) ניתנות לפיתרון יעיל? (במונחים של זמן או זיכרון).

מפתה אולי למדוד יעילות לפי הזמן שלוקח לנו להריץ תוכנה, אבל זה לא מדד טוב כי הוא תלוי מימוש, תלוי חומרה שעליה אנחנו מריצים וכו'. לא ניתן להשתמש במדד כזה עבור הגדרה תיאורטית. אנחנו נשתמש במדד אחר- כמה צעדים דרושים למ"ט על-מנת לבצע את החישוב.

סיבוכיות הזמן: סיבוכיות הזמן של מ"ט M היא פונקציה חלקית $t_M : \Sigma^* \to \mathbb{N}$ שמתאימה לכל $x \in \Sigma^*$ את מספר הצעדים של M על x אם M עוצרת עליו, ואחרת היא לא מוגדת.

חסם סיבוכיות: אומרים שפונקציה $T : \mathbb{N} \to \mathbb{N}$ היא חסם סיבוכיות עבור מ"ט M אם לכל x $t_M(x) \le T(|x|)$ (מספר הצעדים שהמכונה עושה חסום ע"י פונקציה של אורך הקלט).

נתעניין בשאלה- איזה חסם סיבוכיות ייחשב כיעיל?

- לכל מכונה לא טריוויאלית, $|x| = n$, $T(n) \ge n$ (כי צריך לקרוא את הקלט).
- מצד שני $O(2^n), O(2^{2^n})$ הם ערכים גדולים מאוד ל- n.
- התאמה למציאות.
- "נוחות מתמטית".

חישוב יעיל

הגדרה: מ"ט M תיקרא יעילה/פולינומית אם קיים פולינום $p(n)=O(n^c)$ המהווה חסם סיבוכיות עבורה.

יתרונות ההגדרה:

- כמעט כל אלגוריתם שנמצא בשימוש נופל להגדרה.
- עמידות: מושג הפולינומיות לא רגיש למודל (הוכחות השקילות בין מ"א משמרות פולינומיות).
- פולינומיות מקיימת תכונות סגור.

דוגמא: f,g ניתנות לחישוב יעיל אז גם $h(x)=f(g(x))$ ניתנת לחישוב יעיל.

הוכחה: f ניתנת לחישוב יעיל $\leftarrow$ קיימת M_f המחשבת אותה בזמן $O(n^c)$.

g ניתנת לחישוב יעיל $\leftarrow$ קיימת M_g המחשבת אותה בזמן $O(n^d)$.

נבנה M_h שמחשבת את h על קלט x.

1. נריץ את M_f על x לקבלת $y=f(x)$.
2. נריץ את M_g על y לקבלת $h(x)=z=g(y)$ ונפלוט אותו.

סיבוכיות: בשלב 1 M_f רצה זמן $O(n^c)$ ומייצרת y שאורכו $|y|=O(n^c)$. בשלב-2 M_g רצה זמן $O(|y|^d)=O(n^{cd})$. סה"כ: $O(n^{cd})$.

חסרונות ההגדרה:

- ההתאמה למציאות היא לא מושלמת
  - במציאות נעדיף $n^{\log\log n}$ על-פני n^{1000} למרות שהראשון לא פולינומי והשני כן.
  - סימפלקס- אלגוריתם שממומש בהרבה תוכנות במציאות ובמקרים מסויימים רץ בזמן אקספוננציאלי. אבל בפועל נתקלים כמעט אך ורק בקלטים עליהם האלגוריתם יעיל.

מחלקות של חישוב יעיל:

$P=\{L\subseteq\Sigma^* \mid L(M)=L$ כך ש- M קיימת מ"ט פולי $\}$.

$POLY=\{f:\Sigma^*\to\Gamma^* \mid f_M=f$ כך ש- M קיימת מ"ט פולי $\}$.

תכונות: $P\subseteq R$, $f\in POLY\leftarrow f$ מלאה. כל שפה סופית היא ב- P.

הגדרה: f חסומה פולינומיאלית אם קיים פולינום $p(n)$ כך ש- $|f(x)|\le p(|x|)$.

קשר בין פונקציות לשפות:

תזכורת: בהינתן פונקציה f מלאה הגדרנו $L_f=\{(x,y)\mid y=f(x)\}$ והוכחנו כי: f ניתנת לחישוב אם"ם $L_f\in R$. נרצה לדעת האם קיים קשר דומה מבחינת חישוב יעיל.

הטענה: f ניתנת לחישוב יעיל אם"ם $L_f\in P$ אינה נכונה.

דוגמא: $L_f \in P$ - $f(x)=1^{2^{|x|}}$ כי בהינתן (x,y) ניתן לחשב $f(x)$ ולהשוות ל- y (יעיל) אבל $f \notin POLY$ כי אין מספיק זמן כדי לכתוב את הפלט ביעילות, הקלט הוא רק x וחיפוש של y מתאים עבורו לא יכול להתבצע בזמן פולינומי בקלט.

הערה: הטענה בעייתית גם עבור פונקציות שהן חסומות פולינומיאלית.

הגדרה: $L_{f'}=\{(x,y)|\ y$ היא הרישא של $f(x)\}$.

משפט: $f \in POLY$ אם"ם f חסומה פולינומיאלית וגם $L_{f'} \in P$.

הוכחה:

כיוון-1: $f \in POLY$ אז קיימת מ"ט M_f לחישוב $f \leftarrow f$ חסומה פולינומיאלית (כי הפונקציה של M_f שהיא f היא פולינומאלית). מ"ט פולי עבור L_f על קלט (x,y) תחשב את $f(x)$ ע"י M_f ותשווה ל- y.

כיוון-2: f חסומה פולינומיאלית $\leftarrow$ קיים פולינום מתאים p. $L_{f'} \in P \leftarrow$ קיימת מ"ט M' עבור השפה אשר רצה זמן q. נבנה M_f המחשבת את f על קלט x:

- אתחול: $y=\varepsilon$.
- איטרציה: עבור כל $a \in \Gamma$ בדוק האם $(x,ya) \in L_{f'}$ (ע"י M').
  - אם כן, $y \leftarrow ya$ ומתחילים איטרציה חדשה.
  - אם לא, מנסים את האות הבאה.
  - אם ניסינו את כל האותיות, עוצרים עם פלט y (גם רישא וגם הפיתרון).

סיבוכיות: מספר האיטרציות- פולינומי (כי f חסומה פולינמיאלית). בכל איטרציה החישוב הוא פולינומי (כי הוא נעשה ע"י M' מ"ט פולינומית). סה"כ: M_f רצה בזמן פולינומיאלי.

בעיות חיפוש:

הגדרה: יהי $S \subseteq \Sigma^* \times \Sigma^*$ יחס דו-מקומי.

- אומרים שבעית הזיהוי של S ניתנת לפיתרון יעיל אם $S \in P$.
- אומרים שבעית החיפוש של S ניתנת לפיתרון יעיל אם קיימת מ"ט פולינומיאלית M כך שלכל $x \in \Sigma^*$: אם קיים y כך ש- $(x,y) \in S$ אז M עוצרת ב- q_A עם פלט y כזה, אחרת M עוצרת ב- q_R (ואין חשיבות לפלט).

אבחנה: חיפוש יעיל $\not\Leftarrow$ זיהוי יעיל.

הגדרה: יחס S הוא חסום פולינומיאלית אם קיים פולינום p כך שלכל $(x,y) \in S$ מתקיים: $|y| \le p(|x|)$.

השאלה הפתוחה המרכזית של מדעי המחשב (נוסח-1):

עבור S חסומה פולינומיאלית, האם זיהוי יעיל $\Leftarrow$ חיפוש יעיל?

הגדרה: שפה L שייכת למחלקה NP אם קיים יחס דו-מקומי R_L המקיים:

1. R_L חסום פולינומיאלית.
2. R_L ניתן לזיהוי יעיל.
3. $L=\{x \mid \exists y:(x,y)\in R_L\}$ - ביחס כל ה- x והפיתרונות שלהם לבעית החיפוש.

אבחנה-1: $P\subseteq NP$.

הוכחה: בהינתן $L\in P$ נראה יחס R_L מתאים- $R_L=\{(x,x)\mid x\in L\}$ (הוא ניתן לזיהוי כי ניתן לבדוק אם $y=x$ וגם ניתן לבדוק אם $x\in L$ כי $L\in P$).

אבחנה-2: $NP\subseteq R$.

הוכחה: תהי $L\in NP$ ויהי R_L היחס המתאים. בפרט R_L חסומה פולינומיאלית ע"י פולינום p, וגם R_L ניתן לזיהוי יעיל ע"י מ"ט פולינומיאלית M. נתאר מ"ט M' שמזהה את L ועוצרת תמיד: בהינתן x נעבור על כל הy-ים שאורכם קטן או שווה ל- $p(|x|)$. לכל אחד מהם נבדוק (ע"י M) האם $(x,y)\in R_L$. אם מצאנו y כזה $x\in L\Leftarrow$, ונקבל. אם עברנו על כל הy-ים ולא מצאנו $x\notin L\Leftarrow$ ולכן נדחה. הנכונות נובעת מדרישה (3) של R_L שהיא מכילה (x,y) אם $x\in L$.

השאלה הפתוחה המרכזית במדעי המחשב (נוסח-2):

האם $P=NP$?

משפט: שני הנוסחים של הבעיה הפתוחה המרכזית הם שקולים, כלומר:

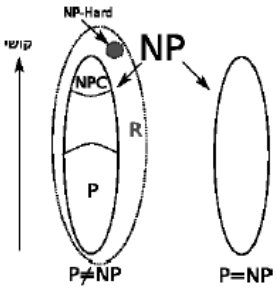
$P=NP \Leftrightarrow$ לכל יחס פולינומיאלי, זיהוי יעיל גורר חיפוש יעיל.

הוכחה:

כיוון-1: מספיק להוכיח ש- $NP\subseteq P$ (כי ההכלה ההפוכה קיימת). תהי $L\in NP$ שפה כלשהי, אז קיים R_L כמובטח מההגדרה. בפרט R_L חסום פולינומיאלית וניתן לזיהוי יעיל ומההנחה נובע ש- R_L ניתן לחיפוש יעיל. כלומר, קיימת מ"ט פולינומיאלית M כך שלכל x, אם קיים y כך ש- $(x,y)\in R_L$, היא עוצרת ב- q_A (עם פלט y כזה) ואם לא קיים היא עוצרת ב- q_R. ולכן: $L\in P \Leftarrow L(M)=L$.

כיוון-2: יהי S יחס חסום פולינומיאלית הניתן לזיהוי יעיל ונרצה להראות שהוא ניתן לחיפוש יעיל. נגדיר: $S'=\{((x,z),w)\mid (x,zw)\in S\}$. עבורו מתקיים (בגלל תכונות S): S' חסום פולינומיאלית וגם S' ניתן לזיהוי יעיל. $L_{S'}=\{(x,z)\mid \exists w:((x,z),w)\in S'\}$. מהגדרת NP: $L_{S'}\in NP$. מההנחה נובע ש- $L_{S'}\in P$ ולכן קיימת M' פולינומיאלית עבור $L_{S'}$. נתאר מ"ט M שפותרת את בעית החיפוש של S: בהינתן קלט x נבדוק האם $(x,\varepsilon)\in L_{S'}$ (כלומר, האם יש פיתרון שמרחיב את המחרוזת הריקה- כל פיתרון הוא כזה, אם קיים). אם לא (אין פיתרון), עוצרים ב- q_R. אם כן, $y\leftarrow\varepsilon$ מתחילים באיטרציות: לכל $a\in\Gamma$ בדוק האם $(x,ya)\in L_{S'}$. אם כן, $y\leftarrow ya$, אם לא- בדוק את האות הבאה. כשסיימנו לעבור על כל האותיות, עוצרים ב- q_A עם פלט y.

NPC- בעיות קשות



NPC: הבעיות הקשות ביותר ב-NP.

הערה: לא ידוע אם יש משהו ב-NP שלא נמצא ב-P.

משפט הרדוקציה המקורי: $L_2 \in R \Leftarrow \begin{cases} L_1 \le L_2 \\ L_2 \in R \end{cases}$

האם מתקיים גם: $L_2 \in P \Leftarrow \begin{cases} L_1 \le L_2 \\ L_2 \in P \end{cases}$?

לא מתקיים- ההוכחה המקורית לא עובדת וגם הטענה אינה נכונה.

הגדרה: פונקציה f היא רדוקציה פולינומית מ- L_1 ל- L_2 ומסמנים: $L_1 \le_p L_2$ אם:

1. $f \in POLY$ (ובפרט מלאה).
2. תקפות: $x \in L_1 \Leftrightarrow f(x) \in L_2$.

תכונות בסיסיות:

- משפט הרדוקציה: $L_1 \in P \Leftarrow \begin{cases} L_1 \le_p L_2 \\ L_2 \in P \end{cases}$

הסבר: אותה הבניה ואותה הוכחה של משפט הרודקציה. הסיבוכיות פולינומית כי $M_f + M_2$ מ"ט פולינומיות.

- טרנזיטיביות: $L_1 \le_p L_3 \Leftarrow \begin{cases} L_1 \le_p L_2 \\ L_2 \le_p L_3 \end{cases}$

הגדרה: שפה L תיקרא **NP-שלמה** אם:

1. $L \in NP$.
2. לכל $L' \in NP$ מתקיים: $L' \le_p L$ (שפה שמקיימת רק את 2 נקראת: **NP-קשה**).

קבוצת השפות הנ"ל תסומן: NPC.

משפט: תהא $L \in NPC$ אז מתקיים: $L \in P \Leftrightarrow P = NP$.

הוכחה:

כיוון $\Rightarrow$: $L \in NPC$ אז מההגדרה- $L \in NP$ אז מההנחה- $L \in P$.

כיוון $\Leftarrow$: מספיק להוכיח $NP \subseteq P$. תהי $L' \in NP$ שפה כלשהי. אז $L' \le_p L$ (כי L היא NP-שלמה) וגם $L \in P$ אז לפי משפט הרדוקציה $L' \in P$.

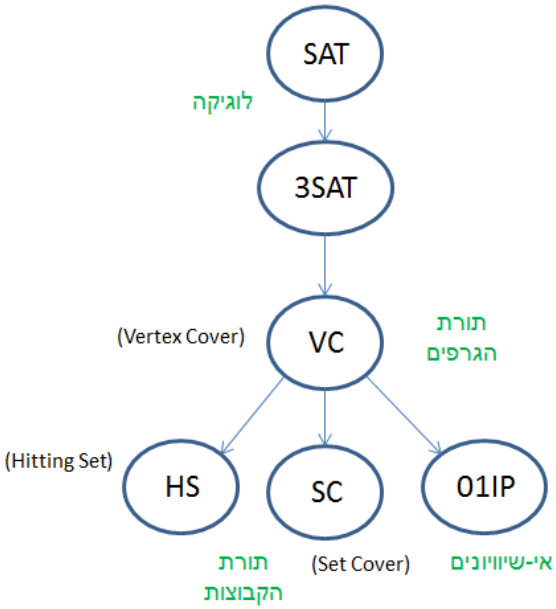
<u>דרכים להוכיח ששפה היא ב-NPC:</u>

1. ישירות ע"פ ההגדרה.
2. הוכחה עקיפה על-סמך:

<u>טענה:</u> אם $L \in NP$ וגם $L_1 \in NPC$ וגם מתקיים: $L_1 \leq_p L$ אז: $L \in NPC$.

<u>הוכחה:</u> מטרנזיטיביות: לכל $L' \in NP$ מתקיים: $L' \leq_p L_1 \leq_p L \Leftarrow L' \leq_p L$.

<u>דוגמאות לשפות NP-שלמות:</u>



<u>בעית הכיסוי בצמתים (Vertex Cover):</u>

<u>הגדרה:</u> יהי $G = (V, E)$ גרף. קבוצה $B \subseteq V$ נקראת כיסוי צמתים אם לכל קשת $(a, b) \in E$ מתקיים: $a \in B$ או $b \in B$ (או שניהם).

<u>הערות:</u>

- בגרף כוכב קיים כיסוי צמתים בגודל-1.
- כל גרף ניתן לכסות ע"י $|V|-1$ צמתים.
- בגרף המלא K_n כל כיסוי גודלו לפחות $|V|-1$.

$VC = \{(G,k) \mid$ קיימת קבוצה B בגודל k המהווה כיסוי בצמתים של $G\}$

$S_{VC} = \{((G,k), B) \mid$ B הוא כיסוי בצמתים בגודל k עבור $G\}$.

<u>תכונות:</u>

- S_{VC} חסום פולינומיאלית.
- S ניתן לזיהוי יעיל (ניתן בזמן פולינומיאלי לוודא ש- $|B| = k$ ושו- B מכסה כל קשת ב- G).
- לא ידוע אם בעית החיפוש ניתנת לפיתרון יעיל (קיים פיתרון אקספוננציאלי).

<u>טענה:</u> $VC \in NPC$.

($NC \in NP$ ע"פ היחס S_{VC}. נראה יותר מאוחר- $3SAT \leq_p VC$).

בעית הקבוצה המייצגת (Hitting Set):

קלט: טבעיים n,k וקבוצות $A_1,\cdots,A_m \subseteq [n]$ ($[n]=0,1,\cdots,n$).

פלט: האם קיימת קבוצה R בגודל k כך שלכל i, $A_i \cap R \neq \varnothing$?

$HS=\{n,k,A_1,\cdots,A_m \mid$ ... המקיימת R $\}$

טענה: $HS \in NPC$.

הוכחה: נחלק אותה ל-3 חלקים:

1. $HS \in NP$ - ע"י היחס מתאים:

$R_{HS}=\{((n,k,A_1,\cdots,A_m),R) \mid$ R קבוצה בגודל k שחותכת כל A_i $\}$

2. $VC \in NPC$

3. $VC \leq_p HS$ - נראה רדוקציה מתאימה: $f(G,k)=(n,k,A_1,\cdots,A_m)$, כך ש: $n=|V|$, $m=|E|$ ולכל $e_i=(a,b)$ נתאים קבוצה- $A_i=\{a,b\}$.

נכונות הרדוקציה: $f \in POLY$. תקפות: $(G,K) \in VC$ $\Leftrightarrow$ קיימת $B=\{V_1,\cdots,V_k\}$ המהווה כיסוי צמתים ל- G $\overset{\text{מהבנה}}{\Leftrightarrow}$ קיימת קבוצה $R=\{i_1,\cdots,i_k\}$ שחותכת כל A_i $\Leftrightarrow f(G,k) \in HS$.

בעית כיסוי הקבוצות (Set Cover):

קלט: טבעיים n,k וקבוצות $C_1,\cdots,C_t \subseteq [n]$.

פלט: האם קיימות k קבוצות $C_{i_1},\cdots,C_{i_k}$ שמכסות את $[n]$?

$SC=\{n,k,C_1,\cdots,C_t \mid$ קיים כיסוי בקבוצות בגודל k מתוך t הקבוצות $\}$.

טענה: $SC \in NPC$.

הוכחה: (בדרך עקיפה)

- $SC \in NP$ - ע"י היחס המתאים: $R_{SC}=\{((n,k,C_1,\cdots,C_t),(i_1,\cdots,i_k))|\cdots\}$
- $VC \in NPC$
- $VC \leq_p SC$ - נראה רדוקציה מתאימה: $f(G,k)=(n,k,C_1,\cdots,C_t)$, $n=|E|$, $t=|V|$, לכל צומת v_i נתאים קבוצה- $\{$ הצומת v_i מכסה את e_j $\mid j\}=C_i$.

נכונות הרדוקציה: $f \in POLY$. תקפות: $(G,K) \in VC$ $\Leftrightarrow$ קיים כיסוי $B=\{i_1,\cdots,i_k\}$ עבור G $\overset{\text{מהבנה}}{\Leftrightarrow}$ הקבוצות $C_{i_1},\cdots,C_{i_k}$ מכסות את $[n]$ $\Leftrightarrow f(G,K) \in SC$.

בעית התכנות בשלמים (01IP):

קלט: מטריצה $A \in Z^{m \times n}$, ווקטור $b \in Z^m$.

פלט: האם יש פיתרון למערכת האי-שוויונים $Ax \geq b$ שבו $x \in \{0,1\}^n$?

באופן שקול: נתונה מערכת אי-שוויונים: $\begin{matrix} a_{11}x_1 + \cdots + a_{1n}x_n \geq b_1 \\ \vdots \\ a_{m1}x_1 + \cdots + a_{mn}x_n \geq b_m \end{matrix}$, האם קיים פיתרון שבו $x_i \in \{0,1\}$ לכל i?

$01IP = \{ A,b \mid \exists x... \}$

<u>טענה:</u> $01IP \in NPC$.

<u>הוכחה: (בדרך העקיפה).</u>

- $01IP \in NP$ - ע"י היחס המתאים: $R_{01IP} = \{ ((A,b),x) \mid Ax \geq b,\ x \in \{0,1\}^n \}$.
- $VC \in NPC$.
- $VC \leq_p 01IP$ - נראה רדוקציה מתאימה: $f(G,k) = (A,b)$, הרעיון: לכל צומת v_i בגרף נתאים משתנה x_i כך ש- $x_i = 1$ או ש- v_i בכיסוי ו- $v_i = 0$ או ש- v_i אינו בכיסוי.

  $|V| = n$, מריצה A מסדר: $(|E|+1) \times |V|$ מוגדרת באופן הבא:

  - לכל קשת $e_j = (u,v)$ נגדיר אי-שוויון: $x_u + x_v \geq 1$, כלומר: $A_{ju} = A_{jv} = 1$ ו- $A_{jk} = 0$ לכל $k \neq u,v$.
  - את השורה האחרונה במטריצה נגדיר ע"י אי-השוויון: $x_1 + x_2 + \cdots + x_{|V|} \leq k$, כלומר $A_{(|E|+1)i} = -1$ לכל i

  נגדיר את b , לפי הבנייה לעיל- $b_i = 1,\ \forall 1 \leq i \leq |E|$ ו- $b_{(|E|+1)} = -k$.

  כעת בהינתן כיסוי מתאים B , ההשמה המספקת היא $x_i = 1 \Leftrightarrow i \in B$.

  מצד שני אם קיימת השמה x המקיימת את כל האי-שוויונים נגדיר את B ע"י: $B = \{ i \mid x_i = 1 \}$.
  וקל לראות ש- $|B| \leq k$, ואכן B כיסוי בצמתים (הקשת $e_j = (u,v)$ בהכרח מכוסה כי אי-השוויון ה- j מתקיים אם"ם $x_u + x_v \geq 1$, כלומר: לפחות אחד מהצמתים u,v נמצא ב- v).

<u>בעית החיפוש החסום (Bounded search):</u>

<u>קלט:</u> מ"ט M , איבר x ווקטור של 1-ים באורך t

<u>פלט:</u> האם קיים y באורך לכל היותר t ש- M מקבלת את x ומוציאה כפלט את y תוך t צעדים?

$BS = \{ \langle M \rangle, \langle x \rangle, 1^t \mid ...t$ { קיים y באורך קטן שווה ל-

<u>טענה:</u> $BS \in NPC$.

<u>הוכחה: (ישירה).</u>

$BS \in NP$ לפי היחס המתאים- הוא חסום פולינומיאלית $|y| \leq t$ וניתן לזיהוי יעיל (עושים סימולציה של t צעדים של המכונה) והיחס מגדיר את השפה.

צ"ל: לכל $L \in NP$ מתקיים: $L \leq_p BS$. מהגדרת NP קיים R_L , כלומר- חסום פולינומית, ניתן לזיהוי יעיל ע"י מ"ט M_L ופולינום P_L עבורה, שרצה זמן $q(|x|+|y|)$.

נגדיר רדוקציה: $f_L(x)=\left(\langle M_L\rangle,\langle x\rangle.1^{p_L(|\langle x\rangle|)}\right)$. $f_L\in POLY$.

תקפות: $x\in L$ קיים y כך ש- $(x,y)\in R_L$ קיים y באורך $p(|x|)\geq$ כך ש- $(x,y)\in R_L$ קיים y באורך $p(|x|)\geq$ מקבלת את x תוך $q(|x|+|y|)\leq t$ צעדים.

3SAT:

שפת כל פסוקים הספיקים מהצורה- 3CNF.

$\varphi=C_1\wedge C_2\wedge\cdots\wedge C_m$ כאשר כל C_i הוא פסוקית בת 3 ליטרלים: $C_i=l_{i_1}\vee i_{i_2}\vee l_{i_3}$ כאשר ליטרל : $l_{i_j}\in\{x_1,\overline{x}_1,x_2,\overline{x}_2,\cdots,x_n,\overline{x}_n\}$.

טענה: $3SAT\leq_p VC$.

הוכחה:

נראה רדוקציה מתאימה- $f(\varphi)=(G,k)$.

בניית G: נבנה את הגרף ב-3 שלבים:

א. $2n$ צמתי ליטרלים המסומנים: $x_1,\overline{x}_1,x_2,\overline{x}_2,\cdots,x_n,\overline{x}_n$ וקשת בין כל x_i ל- $\overline{x}_i$.

ב. $3m$ צמתי פסוקית מסודרים במשולשים כאשר צמתי המשולש המתאים לפסוקית C_i יסמנו: l_{i_1},l_{i_2},l_{i_3} (שמות הצמתים הם כשמות הליטרלים).

ג. לכל צומת פסוקית (מתוך משולש) המסומן ע"י ליטרל, נוסיף קשת בינו ובין צומת הליטרל מקבוצת צמתי להיטרלים משלב-א.

הגדרת k: $k=n+2m$.

מתקיים- f ניתנת לחישוב בזמן פולינומי.

תקפות:

כיוון-1: נניח $\varphi\in 3SAT$ ונראה- $f(\varphi)=(G,k)\in VC$.

$\varphi\in 3SAT$ ולכן קיימת השמה מספקת α עבורו. נשתמש בה כדי לבנות כיסוי B בגודל k ל- G .

נוסיף ל- B את כל צמתי הליטרלים המקיימים: $\alpha(l)=T$. α מספקת כל פסוקית ולכן בכל C_i יש ליטרל שמקבל T, את שני האחרים נוסיף לכיסוי B .

מתקיים: $|B|=n+2m$ כי ישנם n ליטרלים המקבלים T (כל השמה תיתן T ל- n מתוך n ליטרלים ושלילותיהם) וישנם m פסוקיות ונלקחו שני צמתים מכל משולש שהוא פסוקית.

- בכל זוג $x_i,\overline{x}_i$, α נותנת T לבדיוק אחד מהם ולכן הקשת ביניהם מכוסה.
- בכל משולש, כל 2 צמתים מהווים כיסוי ולכן B מכסה קשתות פנימיות של המשולשים.
- לכל קשת שהיא בין קודקוד משולש לצומת ליטרל, או שצומת המשולש ב- B ולכן הקשת מכוסה, אחרת נובע מהבנייה שערך האמת של הליטרל הוא T ולכן צומת הליטרל של הקשת נמצא בכיסוי.

כיוון-2: נניח ש- $(G,k)\in VC$ (קיים כיסוי $|B|=k$) ונוכיח: $\varphi\in 3SAT$.

כל כיסוי משתמש לפחות בצומת ליטרל אחד מבין $x_i,\overline{x}_i$ (כי יש ביניהם קשת), שהם לפחות n ולפחות ב-2 צמתי פסוקית מכל משולש, שהם לפחות $2m$. ומכיוון ש- $k=n+2m$, כל כיסוי משתמש בדיוק ב- n צמתי ליטרלים וב- $2m$ צמתי פסוקית (2 מכל משולש).

הגדרת השמת האמת α: הליטרל מבין $x_i, \bar{x}_i$ שבכיסוי- יקבל ערך T.

נתבונן בצומת הפסוקית C_i שאיננו בכיסוי. יהי l הליטרל המסמן אותה. ונתבונן בקשת e שהיא בין צומת הליטרל לצומת הפסוקית הזה. כיוון ש- B הוא כיסוי, הוא בפרט מכסה את e הנ"ל. מכיוון שצומת הפסוקית לא ב- B אז צומת הליטרל l ב- B ולפי הגדרת ההשמה- $\alpha(l)=T$ ולכן C_i מסופק. זה מתקיים לכל C_i ולכן φ מסופק $\leftarrow \varphi \in 3SAT$.

Subset Sum (SS):

קלט: $x_1, x_2, \cdots, x_s, k$ - טבעיים.

פלט: האם קיימת קבוצה $I \subseteq [s]$ כך ש- $\sum_{i \in I} x_i = k$.

טענה: $SS \in NPC$.

הוכחה (בדרך עקיפה):

1. $SS \in NP$, ע"י: היחס המתאים: $R_{SS} = \left\{ (x_1, \cdots, x_s, k), I \mid I \subseteq [s] \text{ and } \sum_{i \in I} x_i = k \right\}$.

2. $VC \leq_p SS$

נתאר רדוקציה מתאימה: $f(G,k) = b_1, \cdots, b_m, a_1, \cdots, a_n, k'$ כאשר: $n = |V|,\ m = |E|$ ונגדיר:

- לכל $1 \leq j \leq m$: $b_j = 10^{j-1}$
- לכל $1 \leq i \leq n$: $a_i = 10^m + \sum_{j : v_i \in e_j} 10^{j-1}$ (האינדקס בסכימה- כל הקבוצות j ש- v_i שייך אליהן).
- $k' = k \cdot 10^m + \sum_{j=1}^{m} 2 \cdot 10^{j-1}$.

נסדר בטבלה- m השורות הראשונות הן אברי b ו- n השורות אחריהן הן אברי a, והשור האחרונה היא k'.

$b_1 = 0 \cdots \cdots \cdots \cdots 1$
$b_2 = 0 \cdots \cdots \cdots \cdots 10$
$b_3 = 0 \cdots \cdots \cdots \cdots 100$
$\vdots$
$b_m = \overbrace{10 \cdots \cdots \cdots \cdots 0}^{m-1}$
$a_i = \overbrace{10..0110..010..0}^{m}$
$\vdots$
$k' = k \overbrace{22 \cdots \cdots \cdots \cdots 2}^{m}$

נשים לב שלכל a_i יש 1 בספרה השמאלית ביותר וגם 1 ב-3 מקומות נוספים: המקומות j בהם הצומת i מופיע בקשת j.

$f \in POLY$. תקפות:

כיוון-1: ננית ש- $(G,k) \in VC$ ויהי B כיסוי מתאים. אז נבנה פיתרון לבעיה: $(b_1, \cdots, b_m, a_1, \cdots, a_n, k')$. ונבנה פיתרון לבעיה- I:

- אם $v_i \in B$, הוסף את a_i לסכום. ונקבל מספר מהצורה $k2212112\cdots2$ ואז לכל מקום שעבורו קיבלנו 1 נוסיף b_j מתאים-
- אם הקשת e_i מכוסה רק פעם אחת אז הוסף את b_j.

ואז קיבלנו את הסכום k' כנדרש.

כיוון-2: בהינתן פתרון לבעית ה- SS נבנה כיסוי חוקי B: $B = \{ v_i \mid a_i \in I \}$. ומתקיים:

- $|B| = k$ כיוון שה"ספרה" המובילה ב- k' היא k. אין נשא (carry בחיבור) ולכן רק a-ים מכילים 1 בעמודה השמאלית ובהכרח יש בדיוק k צמתים בכיסוי.

- B הוא כיסוי. לכל e_j, ה- b_j המתאים תורם לכל היותר 1 לספרה ה- j של k' ולכן לפחות אחד מה-a-ין בכיסוי תורם 1 לעמודה זו- וזה קורה אם"ם v_i הוא חלק מהקשת של j. ולכן כל הקשתות מכוסות.

בעית החלוקה (PART):

קלט: $x_1, x_2, \cdots, x_s$ טבעיים.

פלט: האם קיימת קבוצה $I \subseteq S$ כך ש- $\sum_{i \in I} x_i = \sum_{i \notin I} x_i$.

טענה: $PART \in NPC$.

הוכחה:

נראה- $SS \leq_p PART$. רדוקציה מתאימה: $f(x_1, \cdots, x_s, k) = x_1, \cdots, x_s, B, C$

ונגדיר: $A = \sum_{i-1}^{k} x_i, \quad B = 2A - k, \quad C = A + k$.

אבחנה: סכום כל האיברים הוא $4A$. בשלה החלוקה B, C לא יימצאו ביחד (כי סכומם ביחד הוא $3A$). הקבוצה I שסכומה הוא k, ביחד עם B, תהיה בסכום של $2A$ ולכן זוהי חלוקה.

הבניה מבטיחה שאם לקלט הרדוקציה יש פתרון ב- SS, אז לפלט יש פיתרון ב- $PART$.

בצד השני- בהינתן פיתרון ל- $PART$, נתבונן ב"צד" שמכיל את B, נבחר בתור פיתרון ל- SS את כל ה- x_i-ים מאותו צד. מאותו שיקול ברור כי זהו פיתרון מתאים ל- SS (סכומם הוא k).

בעית האריזה בתאים (Bin Packaging):

קלט: k תאים בגודל B כל אחד ועצמים שגדליהם: $x_1, \cdots, x_s$.

פלט: האם ניתן להכניס את העצמים ל- k התאים כך שבכל תא סכום העצמים אינו גדול מ- B.

טענה: $BP \in NPC$.

הוכחה:

נראה רדוקציה- $PART \leq_p BP$. $f(x_1, \cdots, x_s) = \left(x_1, \cdots, x_s, k = 2, B = \frac{1}{2} \sum_{i=1}^{s} x_i \right)$,

מחלקים לשני תאים כאשר בכל תא יש בדיוק חצי מהסכום.

מציאת מסלול בגרף מכוון עם אורך ומחיר מוגבלים (Lpath):

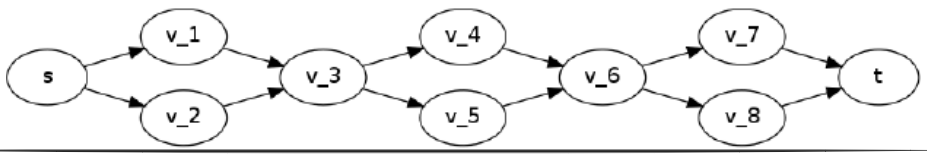
קלט: גרף מכוון G כך שלכל קשת נתון אורך $l(e)$ ומחיר $w(e)$, שני צמתים בגרף s, t ושני מספרים W, L.

פלט: האם קיים מסלול מ- s ל- t שאורכו קטן או שווה ל- L ומחירו קטן או שווה ל- W?

טענה: $Lpath \in NPC$

הוכחה:

נראה רדוקציה $PART \leq_p Lpath$. בהינתן $x_1, \cdots, x_n$ קלט ל-partition, נבנה גרף מתאים מהצורה:



הגרף מורכב מסדרה של מעויינים, כאשר במעויין ה- i, המסלול העליון מתאים לבחירת x_i, והמסלול התחתון לכך שלא נבחר את x_i. הדבר יתבצע ע"י הגדרת המשקלים על הקשתות העליונות כ- x_i והאורכים של הקשתות התחתונות כ- x_i, וכך מעבר במעויין מלמעלה יוסיף לאורך ואילו מעבר במעויין מלמטה יוסיף למשקל. נגדיר: $W = L = \frac{1}{2}\sum_{i=1}^{n} x_i$, ואפשר לראות שמסלולים המוגבלים באורך ובמשקל שלהם ע"י ערך זה, מתאימים לפתרון של בעית החלוקה.

SAT:

שפת כל פסוקי ה-CNF הספיקים (פסוק CNF הוא and בין פסוקיות המורכבות מ-or בין ליטרלים).

טענה: $VC \le_p SAT$

הוכחה:

נראה רדוקציה f - $f(G,k)=\varphi$ המקיימת: $(G,k)\in VC \leftrightarrow \varphi \in SAT$.

המשתנים של φ: x_{ij}, $i\in[n]$, $j\in[k]$ (המשמעות: $x_{ij}=T \leftrightarrow$ הקודקוד ה- i הוא המספר ה- j בכיסוי).

הפסוקיות של φ:

א. לכל $j\in[k]$, פסוקית- $\left(\bigvee_{i\in[n]} x_{ij}\right)$ (כלומר, יש לפחות קודקוד אחד במקום ה- j בכיסוי).

ב. לכל $i\neq i'$, פסוקית- $\left(\overline{x}_{ij}\vee\overline{x}_{i'j}\right)$ (כלומר, רק אחד מביניהם יקבל T בו-זמנית, ישנו רק קודקוד אחד במקום ה- j בכיסוי).

ג. לכל קשת $e=(a,b)$, פסוקית- $\left(\left(\bigvee_{j\in[k]} x_{aj}\right)\vee\left(\bigvee_{j\in[k]} x_{bj}\right)\right)$ (כלומר, זהו אכן כיסוי צמתים, הפסוקית תקבל T אם לפחות אחד מהצמתים a,b נמצא בכיסוי).

נכונות: $f\in POLY$.

תקפות:

כיוון-1: $(G,k)\in VC \leftarrow$ קיים כיסוי B, $B=\{t_1,t_2,\cdots,t_k\}$ ונשתמש ב- B על-מנת לבנות השמה מספקת ל- φ: $x_{ij}=T$ אם"ם במקום ה- j בכיסוי מופיע הצומת- i. מהבנייה נובע שכל הפסוקיות מסופקות ולכן: $\varphi\in SAT$.

כיוון-2: $f(G,k)=\varphi\in SAT \leftarrow$ יש השמת אמת α המספקת את φ, בפרט, α מספקת את הפסוקיות מסוג א ו-ב ולכן לכל j יש i יחיד כך ש- $x_{ij}=T$. אז ניקח את כל ה- i הנ"ל לכיסוי B $\leftarrow |B|=k$ (כי ישנם k j-ים). B מהווה כיסוי כי α מספקת גם את הפסוקיות מסוג ג ולכן לכל קשת אחד מהמשתנים x_{aj},x_{bj} מקבל T וזה קורה אם הצומת a או b בכיסוי.

טענה (ללא הוכחה): לכל פונקציה בוליאנית $f:\{0,1\}^k\to\{0,1\}$, קיים CNF שנסמן φ, המחשב את f. יתר-על כן ב- φ יש לכל היותר 2^k פסוקיות $\leftarrow$ אם $k=O(1)$ אז גם $|\varphi|=O(1)$.

משפט Cook:

$SAT \in NPC$.

הוכחה:

תהי $L \in NP$. צ"ל: $L \leq_p SAT$. מהגדרת NP נובע ש- קיים ל- L יחס דו-מקומי R_L המקיים:

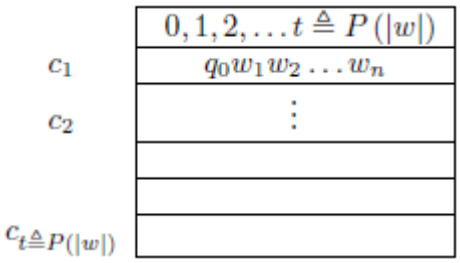
1. R_L חסום פולינומית ע"י $p(n)$.
2. ניתן לזיהוי יעיל ע"י מ"ט M הרצה זמן פולינומי- $q(|w|+|y|)$ (פולינומי בקלט (w,y)).
3. R_L מגדירה את L $L = \{w \mid \exists y : (w,y) \in R_L\}$.

נראה רדוקציה מתאימה- $h = h_L$: $h(w) = \varphi$ המקיימת:

$h(w) = \varphi \in SAT \leftrightarrow w \in L$, כלומר, קיים y באורך $p(|w|)$ כך שמ"ט M מקבלת את (w,y) בזמן $\underbrace{q(|w|+p(|w|))}_{t}$ אם"ם קיימת השמת אמת המספקת את φ.

אם $w \in L$ אז קיים y וקיימת סדרת קונפיגורציות $C_0, \cdots, C_t$ המהווה חישוב מקבל של M על w,y.

נתבונן על טבלת חישוב:



כניסות הטבלה הן מתוך $\Gamma \cup Q$.

בכל שורה נרשום $C_i = \alpha_1 \alpha_2 \cdots \alpha_j q \alpha_{j+1} \cdots \alpha_t$.

- אם קונפיגורציה היא קצרה מ- t, נוסיף $ בסופה.
- אם החישוב קצר מידי (הסתיים לפני "זמן" t = מצב C_t), נשכפל את הקונפיגורציה האחרונה לכל השורות התחתונות.

משתני φ: x_{ija}, $0 \leq i,j \leq t$, $a \in \Gamma \cup Q$ (משמעות: $x_{ija} = T \leftrightarrow$ בתא ה- (i,j) של הטבלת חישוב מופיע a).

בניית φ: $\varphi = \varphi_0 \wedge \varphi_A \wedge \varphi_{move} \wedge \varphi_{cell}$.

- $\varphi_A = \bigvee_{0 \leq j \leq t} x_{t\,j\,q_A}$, מקבל T אם"ם השורה האחרונה בטבלה היא קונפיגורציה מקבלת, המכילה q_A. המשמעות היא: באיזשהו מקום j בשורה האחרונה- t, רשום q_A.
- $\varphi_{cell} = \bigwedge_{0 \leq i,j \leq t} \varphi_{cell}^{i,j}$, כאשר: $\varphi_{cell}^{i,j} = \left(\bigvee_{a \in \Gamma \cup Q} x_{i\,j\,a}\right) \wedge \left(\bigwedge_{a \neq a'} \left(\overline{x}_{i\,j\,a} \vee \overline{x}_{i\,j\,a'}\right)\right)$, כלומר, בכל תא i,j בטבלה מופיע בדיוק ערך אחד (משמעות השמאלי: לפחות-1, משמעות הימני: לכל היותר 1).
- $\varphi_0 = x_{00q_0} \wedge x_{01w_1} \wedge \cdots \wedge x_{0nw_n} \wedge x_{0n+1b} \wedge \cdots x_{0tb}$, מקבל T אם"ם השורה הראשונה בטבלה מתארת קונפיגורציה תחילית של M על w,y.
- לכל $i \geq 1$, השורה ה- i מתקבלת מהשורה ה $i-1$ ע"י צעד אחד (חוקי) של M.
  - 2 קונפיגורציות עוקבות נבדלות בלכל היותר 3 מקומות.
  - התא (i,j) בטבלה מוגדר באופן יחיד מתוך 4 התאים: $\{j-1, j, j+1, j+2\}$ בשורה שלפניו- $i-1$.
  - תבנית של 5 תאים כזו תיקרא: חוקית אם תהיה קונסיסטנטית עם δ_M.

- S_M - אוסף התבניות הקונסיסטנטיות הוא קבוע (תלוי ב- M ולא ב- w). קל לחשב אותו מתוך $\langle M \rangle$ וגודלו $O(1)$.
- טבלת החישוב היא חוקית $\leftrightarrow$ כל תבנית במקום (i,j) כלשהו היא חוקית.

$\varphi_{move} = \bigwedge_{i,j} \varphi_{move}^{i,j}$, משמעות $\varphi_{move}^{i,j}$ - התבנית במקום ה- (i,j) היא חוקית.

לפי הטענה, ל- $\varphi_{move}^{i,j}$ יש CNF בגודל $O(1)$ שתלוי רק ב- M ולא ב- w שמתאר אותו. נשתמש בו בתור $\varphi_{move}^{i,j}$.

<u>תקפות הרדוקציה:</u>

כיוון-1: $w \in L \leftarrow$ קיים חישוב מקבל של M על w עבור איזשהו $y \in \{0,1\}^{p(n)}$. נתבונן בטבלת החישוב המתאימה ונגדיר: $x_{ija} = T \leftrightarrow$ בטבלה הנ"ל במקום (i,j) מופיע a. מהבנייה נובע: כל תתי הנוסחאות מסופקות. ולכן $\varphi \in SAT$.

כיוון-2: $h(w) = \varphi \in SAT \leftarrow$ קיימת השמה מספקת α. בפרט α מספקת את $\varphi_{cell}^{i,j}$ ולכן לכל (i,j) יש a יחיד כך ש- $x_{ija} = T \leftarrow \alpha$ משרה טבלת חישוב. מכיוון ש- α מספקת את $\varphi_0, \varphi_A, \varphi_{move} \leftarrow$ הטבלה הנ"ל מתחילה בקונפיגורציה תחילית של M על w, y עבור $y \in \{0,1\}^{p(n)}$, מסתיימת בקונפיגורציה מקבלת ומהווה חישוב חוקי $\leftarrow M$ מקבלת את $(w,y) \leftarrow w \in L$.

<u>טענה:</u> $SAT \leq_p 3SAT$.

<u>הוכחה:</u>

מספיק להראות איך לתרגם פסוקית אחת $C = (u_1 \vee u_2 \vee \cdots u_n)$ לפסוק 3CNF. עבור פסוק כללי אפשר להפעילת את התהליך על כל פסוקית לחוד.

- אם $C = (u_1)$ אז מתרגמים אותה לפסוקית $(u_1 \vee u_1 \vee u_1)$.
- אם $C = (u_1 \vee u_2)$ אז מתרגמים אותה לפסוקית $(u_1 \vee u_1 \vee u_2)$.
- אם $C = (u_1 \vee u_2 \vee u_3)$ אז משאירים אותה כמו שהיא.
- עבור $C = (u_1 \vee u_2 \vee \cdots \vee u_n)$ כך ש- $n \geq 4$ הפתרון מורכב יותר ודורש שימוש במשתני עזר שנסמן ב- $y_1, y_2, \cdots, y_{n-2}$. נחליף את הפסוק בסדרת הפסוקיות הבאה:
  $(u_1 \vee u_2 \vee y_1) \wedge (\overline{y}_1 \vee u_3 \vee y_1) \wedge \cdots \wedge (\overline{y}_{n-2} \vee u_{n-1} \vee u_n)$.

אם בהשמה כלשהי C מסתפק אז קיים u_k שמקבל T. נגדיר השמה שמספקת את סדרת הפסוקיות החדשה: למשתנים המקוריים יושמו אותם ערכים כמו בהשמה המקורית. ואילו ל- y_i-ים יושמו ערכי אמת באופן הבא: $y_i = T$ לכל $i \leq k-2$ ו- $y_i = F$ לאחרים. השמה זו מספקת את כל הספוקיות למעט $(\overline{y}_{k-2} \vee u_k \vee y_{k-1})$ ופסוקית זו מסתפקת שכן u_k מקבל ערך T.

בכיוון השני אם ישנה השמה שמספקת את הפסוק החדש אז אותה השמה כשהיא מצומצמת מספקת גם את הפסוק המקורי, כי אם לא, בהכרח כל ה- u_k-ים מקבלים F ולכן (יש להוכיח באינדוקציה) כל ה- y_i ים מקבלים T ולכן הפסוקית: $(\overline{y}_{n-2} \vee u_{n-1} \vee u_n)$ אינה מסתפקת- סתירה.

<u>2SAT:</u> שפה השייכת ל-P. בהינתן פסוק 2CNF ניתן להכריע אם הוא ספיק או לא.

מכונת טיורינג אי-דטרמיניסטית:

מוגדרת כמו מ"ט רגילה לקבלת שפות למעט: $\delta:(Q\setminus F)\times\Gamma\to\left(Q\times\Gamma\times\{L,R,S\}\right)^2$, כלומר: ישנה אפשרות ל-2 מצבים לכל היותר (אם ישנם 0 מצבים, המסלול נקטע וזה שקול למעבר ל- q_R).

משמעות: אם $\delta(q,a)=\{(p_0,b_0,d_0),(p_1,b_1,d_1)\}$ אז אחד מהשניים מתבצע ולכן: במקום מסלול חישוב יש עץ חישוב.

הגדרה: אומרים שמ"ט א"ד M מקבלת קלט x אם בעץ החישוב של M על x קיים מסלול המסתיים ב- q_A.

דוגמא: $\overline{L_\varnothing}=\{\langle M\rangle \mid L(M)\neq\varnothing\}\in RE$.

נתאר מ"ט א"ד $M_\varnothing$ עבור $\overline{L_\varnothing}$ על קלט $\langle M\rangle$:

- המכונה "תנחש" מחרוזת $x\in\{0,1\}^*$.
- המכונה תבדוק (בצורה דטרמיניסטית) האם M מקבלת את x ותקבל אם"ם התשובה חיובית.
- מימוש הניחוש: ע"י מ"א א"ד דו-סרטית עם דרגת פיצול 3, כלומר: 3 אפשרויות ליצירת מחרוזת- הוספת 1, הוספת 0 וסיום. בכל מסלול חישוב נוצרת מחרוזת ולכל מחרוזת יש מסלול חישוב שבו היא נוצרת.

נכונות: $\langle M\rangle\in\overline{L_\varnothing}\leftrightarrow$ קיים $x\in L(M)\leftrightarrow$ קיים מסלול בו $M_\varnothing$ מנחשת את x ובמסלול הזה מגיעה ל- $q_A\leftrightarrow\langle M\rangle\in L(M_\varnothing)$. ולכן: $L(M_\varnothing)=\overline{L_\varnothing}$.

משפט: אוסף השפות הניתנות לקבלה ע"י מ"ט א"ד
= אוסף השפות הניתנות לקבלה ע"י מ"ט דטרמיניסטית
= RE.

רעיון ההוכחה:

כיוון-1: כל מ"ט דטרמיניסטית היא בפרט מ"ט א"ד.

כיוון-2: בהינתן מ"ט א"ד M נראה שניתן לחשב את הפרדיקט הבא: $M(x,w)=1$ (אם"ם M מקבלת את x במסלול w). אופן חישוב הפרדיקט: נעבור על כל ה- w בסדר לקסיקוגרפי עד שנמצא כזה שעבורו: $M(x,w)=1$ ואז נקבל.

הגדרה: מ"ט א"ד M תיקרא פולינומית (או יעילה) אם קיים פולנים $p(n)$ כך ש- M עוצרת על x תוך $p(|x|)$ צעדים בכל מסלולי החישוב שלה.

הגדרה נוספת ל-NP: אוסף השפות L שקיימת עבורן מ"ט א"ד פולינומית.

הערה: האם P=NP זו שאלה על כוחו של אי-דטרמיניזם (כלומר: האם לכל שפה שיש לה מ"ט א"ד יעילה יש לה גם מ"ט דטרמיניסטית יעילה?)

דוגמא: $SAT\in NP$. מ"ט א"ד M_{SAT} על קלט φ "תנחש" השמה α באורך n=מספר המשתנים. ואז היא תבדוק באופן דטרמיניסטי האם α מספקת את φ ומקבלת אם כן. $M_{SAT}\in POLY$.

משפט: שתי ההגדרות של NP הן שקולות.

הוכחה:

כיוון-1: נניח ש- $L \in NP$ ע"פ הגדרה-1. כלומר: קיים יחס R_L מתאים. בפרט- חסום פולינומית ע"י פולינום $p(n)$ וניתן לזיהוי פולינומי ע"י מ"ט פולינומית M שרצה $q(|x|+|y|)$. נראה ש- $L \in NP$ ע"פ הגדרה 2: ע"י בניית M' א"ד פולינומית עבור L:

- היא תנחש מחרוזת y באורך $\geq p(|x|)$.
- תבדוק האם $(x,y) \in R_L$ ע"י M המובטחת ותקבל אם"ם התשובה חיובית.

נכונות: $x \in L \leftrightarrow$ קיים y באורך $p(|x|)$ כך ש- $(x,y) \in R_L \leftrightarrow$ קיים מסלול בו מנחשים y כזה $x \in L(M') \leftrightarrow$.

סיבוכיות: צעד הניחוש- $O(p(|x|))$, בדיקה- $q(|x|+p(|x|)) \geq q(|x|+|y|)$ -פולינומי ב- x.

כיוון-2: נניח ש- $L \in NP$ ע"פ הגדרה-2 ו- M מ"ט א"ד פולינומית (עם פולינום $p(n)$) מתאימה.

נראה ש- $L \in NP$ ע"פ הגדרה-1: $R_L = \{(x,w) \mid |w| \leq p(|x|), M(x,w)=1\}$ ומתקיים:

- R_L חסום פולינומית – מההגדרה.
- R_L ניתן לזיהוי יעיל- המימוש של M שראינו עולה $|w|$ צעדים.

$L = \{x \mid x$ מקבלת את M שבו $p(|x|)$ באורך לכל היותר w קיים מסלול $\}$={x| $\exists w : (x,y) \in R_L$}.

NP כמערכת הוכחה:

שלמות: ניתן להוכיח כל טענה נכונה.

נאותות: שום הוכחה לא משכנעת בטענה לא נכונה.

טענה: $NP \equiv$ אוסף השפות L שיש עבורן מערכת הוכחה כנ"ל עם מוודא יעיל.

רעיון ההוכחה: $L \in NP$ אז קיים יחס R_L כך שההוכחה y מקיימת: $(x,y) \in R_L$ וגם שלמות ונאותות ולכן R_L ניתן לזיהוי יעיל ולכן לוידוא פולינומי.

צד שני: נניח של- L קיימת מערכת הוכחה כנ"ל ונוכיח ש- $L \in NP$. $R_L = \{x,y \mid V(x,y)=1\}$.

מהשלמות והנאותות והיעילות של המוודא מקבלים את כל תכונות R_L.

התמודדות עם בעיות קשות

אלגוריתמי קירוב:

רוב עיסוק שלנו בקורס היה בבעיות הכרעה, אך ניתן לדבר באופן כללי יותר על בעיות חישוב פונקציות. בדרך-כלל חישוב הפטנקציות גם מאפשר לפתור את בעיית ההכרעה ולכן הוא קשה לפחות כמו פיתרון בעיית ההכרעה. מצד שני, כאשר התשובה היא מספרית (ולא רק "כן/לא") ניתן לקוות שאפשר יהיה לקרב אותה- להחזיר תשובה שאינה האופטימאלית, אבל היא "לא רעה".

הגדרה: תהא $f:\Sigma^* \to \square$ פונקציה.

- מכונה M לחישוב פונקציות היא קירוב d-חיבורי של f $(d>0)$ אם לכל $x\in\Sigma^*$ מתקיים: $|M(x)-f(x)|\le d$, כלומר: $f(x)-d\le M(x)\le f(x)+d$.
- מכונה M לחישוב פונקציות היא קירוב α-כפלי של f $(\alpha>1)$ אם לכל $x\in\Sigma^*$ מתקיים: $\frac{1}{\alpha}f(x)\le M(x)\le \alpha f(x)$.

<u>הערה:</u> הקירובים הללו הם דו-צדדיים, אך ברוב המקרים דורשים שהקירוב יהי חד-צדדי, כלומר: אם $f(x)$ מייצג בעיית מינימיזציה לרוב דורשים כי יתקיים: $f(x)\le M(x)$ ואם $f(x)$ מייצג בעיית מקסימיזציה אז דורשים כי $M(x)\le f(x)$.

הערה: במקרים רבים לא קיימים אפילו קירובים טובים בזמן פולניומי

דוגמא: מספר הצמתים בכיסוי הצמתים המינימאלי של G $f_{VC}(G)=G$.

<u>אבחנה:</u> $VC\in P \leftrightarrow f_{VC}\in POLY$.

טענה: קיים אלגוריתם 2-קירוב עבור f_{VC}. כלומר, קיים אלגוריתם A פולינומי שלכל G פולט כיסוי צמתים המקיים: $f_{VC}(G)\le A(G)\le 2f_{VC}(G)$.

שידוך מקסימאלי: שידוך שלא ניתן להרחבה (הגדרה שונה משידוך מקסימום).

<u>אלגוריתם יעיל למציאת שידוך מקסימאלי בגרף:</u>

<u>אתחול:</u> $M=\varnothing$. עבור על כל הקשתות $e\in E$ לפי סדר כלשהו. אם e זרה לכל הקשתות ב- M, הוסף אותה ועדכן: $M\leftarrow M\cup\{e\}$.

<u>נכונות:</u>

- האלגוריתם פולינומי.
- (באינדוקציה) M הוא שידוך לכל אורך האלגוריתם (ולכן גם בפלט).
- הפלט M הוא מקסימאלי כי אם בסוף יכולנו להוסיף קשת, הרי שגם כשעברנו עליה יכולנו להוסיפה לקבוצה- M קטנה יותר.

<u>אלגוריתם A:</u>

- מצא שידוך מקסימאלי M בגרף G.
- פלוט ככיסוי צמתים B את כל הצמתים שמשתתפים בקשתות M.

<u>נכונות:</u>

- A פולינומי.
- B הוא כיסוי צמתים- נבדוק שכל $e\in E$ מכוסה: אם $e\in M$ היא מכוסה. אם $e\notin M$ וגם לא מכוסה, אז ניתן להוסיפה אז ניתן להוסיפה לשידוך בסתירה למקסימאליות.

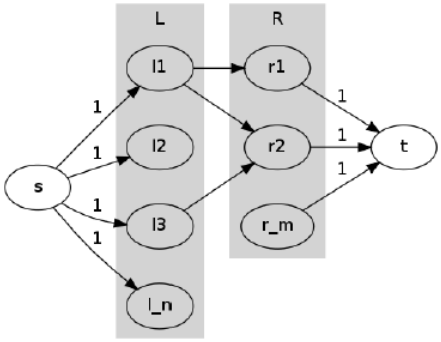
- נסמן ב- B^* כיסוי צמתים בגודל מינימאלי. מתקיים: $|B^*| \le |B|$.
- B^* כיסוי ולכן מכיל צומת מכל קשת ב- M. הקשתות הנ"ל זרות ולכן- $|B^*| > |M|$ ומצד שני: $|B| = 2|M| \le 2|B^*|$.

צמצום מרחב הקלט:

אפשר לחשוב על מרחבי קירוב בתור "הרחבת מרחב הפלט"- במקום לאפשר לכל קלט רק פלט מסויים, מרחיב את מרחב הפלטים שאנחנו מוכנים לקבל (לפלטים "קרובים") ובכך נהפוך את הבעיה לקלה יותר. באופן דומה- נוכל לצמצם את מספר הקלטים שאנחנו מוכנים לקבל, בשאיפה- נוציא מהבעיה את החלק ה"קשה" ונוכל לפתור את הבעיה המצומצמת בקלות.

דוגמא: מציאת כיסוי צמתים בגרף דו-צדדי:

<u>אלגוריתם פולינומי לבעיה (רדוקציה לזרימה):</u>



בהינתן גרף דו-צדדי $G = (L, R, E)$, נבנה רשת זרימה N:

נוסיף לגרף מקור s שיחובר לכל צמתי L בקשתות שקיבולן 1, בור t שכל צמתי R יחוברו אליו עם קיבול 1. ואת כל קשתות הגרף המקורי נגדיר לקיבול ∞.

נחשב זרימת מקסימום ב- N ונוציא אותה כפלט.

min-cut-max-flow: ערך חתך מינימאלי=ערך זרימה מקסימאלית.

<u>טענה:</u>

1. אם קיים כיסוי בצמתים ב- G, $B = (B_L, B_R)$ אז קיים חתך $(T, \bar{T})$ ב- N שקיבולו $|B|$.
2. אם קיים חתך $(T, \bar{T})$ ב- N שקיבולו $k < \infty$ אז קיים כיסוי צמתים ב- G שגודלו k.

<u>מסקנה:</u> גודל הכיסוי המינימאלי ב- G=גודל החתך המינימאלי ב- N =זרימה מקסימאלית ב- N.

<u>הוכחה:</u>

<u>כיוון-1:</u> בהינתן כיסוי $B = (B_L, B_R)$ נבנה חתך- $T = \{s\} \cup \bar{B}_L \cup B_R$ ונחשב את קיבול החתך:

- קיבול הקשתות מ- s ל- B_L הוא: $|B_L|$.
- קיבול הקשתות מ- $\bar{B}_L$ ל- $\bar{B}_R$ הוא: 0 (אין קשתות כאלה, כי קשת כזו לא מכוסה ע"י הכיסוי).
- קיבול הקשתות מ- B_R ל- t הוא: $|B_R|$.

ולכן קיבול החתך הוא: $|B|$.

<u>כיוון-2:</u> בהינתן חתך $(T, \bar{T})$ שקיבולו k (סופי), נגדיר כיסוי: $B_L = \bar{T} \cap L$, $B_R = T \cap R$.

כיוון שקיבול החתך הוא סופי, אין קשת מ- $\bar{B}_L$ ל- $\bar{B}_R$ ולכן כל הקשתות מכוסות בכיסוי. ומכיוון שקיבול החתך הוא k, ניתן לחזור על החשבון מהסעיף הקודם ונקבל כי $|B| = k$.

<u>הערה:</u> אנחנו מכירים אלגוריתמים פולינומיאליים למציאת זרימת מקסימום ולכן יכולים לפתור את הבעיה בזמן פולינומי.

גישות הסתברותיות:

סיבוכיות ממוצעת: מניחים שקיים פילוג הסתברות D על הקלטים לבעיה, ואז מראים אלגוריתם ש**בממוצע** עובד בזמן פולינומיאלי (ביחס לפילוג הזה).

הגישה הזו בעייתית, כיוון שאנחנו חייבים לדעת במדוייק מה ההסתברות לקבל כל אחד מהקלטים האפשריים לבעיה. למעשה- לקבל את D בעצמו זו שאיפה כמעט לא פרקטית. שימוש בפילוג לא נכון, גם אם קרוב לפילוג האמיתי, יכול להשפיע בצורה קיצונית על האנליזה. וגם אם ידוע לנו מהו D- החישוב המדוייק של התוחלת הוא (ברוב המוחלט של המקרים) קשה מאוד.

אלגוריתמים הסתברותיים: אין פילוג על הקלטים, אנחנו מניחים את המקרה הרע ביותר (הקלט הקשה ביותר). אך במקרה זה האלגוריתם עצמו הוא הסתברותי.

הרעיון המרכזי הוא שעל אותו קלט, ריצות שונות יעבדו באופן שונה, כאשר צעדים מסויימים באלגוריתם יבוססו על החלטות רנדומליות. ונשאל- מה ההסתברות שהאלגוריתם יגיע לתשובה הנכונה? ודורשים: לכל קלט x, האלגוריתם מחשב נכון בהסתברות $0.5 < P < p(x)$.

הגברה: עבור בעיות הכרעה, אם חוזרים על האלגוריתם $O(k)$ פעמים, ומחליטים ע"פ הרוב, השגיאה קטנה ל- $\frac{1}{2^k}$.

דוגמא- Max-Cut:

נתון: גרף (לא מכוון) G עם קיבולת $c(e) \geq 0$ על כל $e \in E$

צריך למצוא: חתך בגרף עם קיבולת מקסימאלית.

נראה אלגוריתם הסתברותי יעיל A, שלכל קלט G פולט חתך ב- G כך שגודל החתך X ש-A פולט מקיים: $E(X) \geq \frac{1}{2} opt$ (חישוב תוחלת) כאשר opt הוא גודל החתך המקסימאלי.

אלגוריתם A:

לכל צומת $v \in V$, בהסתברות $\frac{1}{2}$ הוסף את v לקבוצה S. אחרת, הוסף את v ל- $\bar{S}$.

החזר את החתך: $(S, \bar{S})$.

אבחנות:

- $opt \leq \sum_{e \in E} c(e)$.
- עבור $e = (a,b)$ קשת כלשהי, ההסתברות שהיא קשת בחתך היא: $\frac{1}{2}$.
- לכל e נגדיר מ"מ: $x_e = 1$ אם"ם e בחתך. נקבל כי: $E(x_e) = \frac{1}{2}$ and $P(x_e = 1) = \frac{1}{2}$.
- נגדיר: $X = \sum_e c(e) \cdot x_e$ - קיבול החתך שבחרנו.
- ולכן: $E(X) = \frac{1}{2} \sum_e c(e) \geq \frac{1}{2} opt$.

לפי א"ש מרקוב: $P\left(x \geq \frac{1}{4} \sum_e c(e) \right) \geq \frac{1}{3}$.

ואם נפעיל הגברה: נריץ את האלגוריתם k פעמים ונפלוט את החתך הגדול ביותר שפגשנו. יתקיים:

$$P\left(X \geq \frac{1}{4} \sum_e c(e) \right) \geq 1 - \left(\frac{2}{3}\right)^k$$

פונקציות "קשות לקירוב":

דוגמא: בהינתן פסוק CNF φ, הפונקציה מחזירה את מספר ההשמות שמספקות אותו- $\#SAT(\varphi)$.

אבחנה: לכל קבוע $d \geq 1$ אם קיים d-קירוב עבור $\#SAT$, כלומר: אלגוריתם A יעיל מספיק המקיים לכל φ: $\frac{\#SAT(\varphi)}{d} \leq A(\varphi) \leq d \cdot \#SAT(\varphi)$ אז $P = NP$.

הוכחה: אם $\#SAT(\varphi) > 0$ אז בשני אגפי אי-השוויון נקבל מספר גדול מ-0 ולכן בהכרח גם $A(\varphi)$ ייתן מספר גדול מ-0. אחרת, $\#SAT(\varphi) = 0$ ואז שני אגפי אי-השוויון הם 0 ולכן גם $A(\varphi) = 0$.

בפרט- $A(\varphi)$ מספק לנו פתרון פלינומיאלי ל- SAT, וכיוון ש- $SAT \in NPC$ זה מוכיח כי: $P = NP$.

כלומר: קירוב כפלי לא מועיל כי הבעיה היא להבדיל בין מצב שבו $\#SAT$ שווה לאפס למצב שבו הוא שונה מאפס, וכפל בקבוע לא מועיל כאן.

טענה: לכל קבוע $d \geq 0$ אם קיים d-קירוב חיבורי עבור $\#SAT$, כלומר קיים אלגוריתם A יעיל המקיים לכל φ: $\#SAT(\varphi) - d \leq A(\varphi) \leq \#SAT(\varphi) + d$ אז $P = NP$.

הוכחה:

נראה אלגוריתם פולינומי B עבור SAT (על קלט φ):

- נגדיר: $k = \lceil \log d \rceil + 2$.
- נגדיר: $\varphi' = \varphi \wedge (y_1 \vee \overline{y}_1) \wedge \cdots \wedge (\overline{y}_k \vee y_k)$ כאשר $y_1, \cdots, y_k$ משתנים חדשים.
- נריץ את $A(\varphi')$. אם הפלט > 2d- נקבל. אחרת- נדחה.

נכונות:

- האלגוריתם פולינומי (כי A כזה).
- $\#SAT(\varphi') = 2^k \#SAT(\varphi)$.

$\varphi \notin SAT \rightarrow \#SAT(\varphi) = 0 \rightarrow \#SAT(\varphi') = 0 \rightarrow A(\varphi') \leq d$ דוחים.

$\varphi \in SAT \rightarrow \#SAT(\varphi) \geq 1 \rightarrow \#SAT(\varphi) \geq 2^k \geq 4d \rightarrow A(\varphi') \geq 3d \geq 2d$ מקבלים.

עולם הבעיות הנוכחי

משפט Lander: אם $P \neq NP$ אז קיימות שפות L שאינן ב- P ואינן ב- NPC.

סיבוכיות זיכרון: סיבוכיות זיכרון של M על x זהו אינדקס התא הימני ביותר אליו M מגיעה בריצתה על x.

PSPACE: אוסף השפות L שיש עבורן מ"ט בעלת סיבוכיות זיכרון פולינומית.

טענה: $P \subseteq PSPACE$ - סיבוכיות זמן הריצה של מכונה תמיד גדולה יותר מסיבוכיות הזיכרון שלה.

הבדל עקרוני בין זמן ריצה לזיכרון: זיכרון אפשר למחזר, זמן אי-אפשר.

שאלות פתוחות:

- האם $P = PSPACE$?
- האם $NP = PSPACE$?

CO-NP: $Co - NP = \{L \mid \overline{L} \in NP\}$.

דוגמא: $\overline{SAT}$, אוסף כל פסוקי ה-CNF שאינם ספיקים. $\overline{SAT} \in Co - NP$.

אבחנות:

- $P \subseteq Co - NP$ (כי P סגורה למשלים ולכן: $L \in P \to \overline{L} \in P \to \overline{L} \in NP$).
- **Co-NP שלמות:** שפה L שנמצאת ב- NP המקיימת: לכל $L' \in Co - NP$, $L' \leq_p L$.

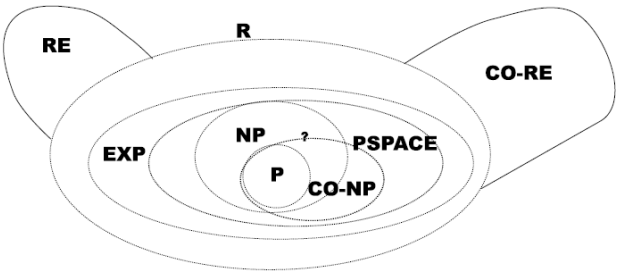
סימון: Co-NPC.

מסקנה: $\overline{SAT} \in Co - NPC$.

טענה: $Co - NP \subseteq PSPACE$.

שאלות פתוחות:

- האם $P = Co - NP$?
- האם $NP = Co - NP$?



EXP: $EXP = \{L \mid L \in DTIME(2^{n^c}),\ for\ a\ constat\ c\}$, אוסף השפות הניתנות לזיהוי ע"י מ"ט בעלת זיכרון אקספוננציאלי באורך הקלט.

שאלות פתוחות:

- האם $EXP = PSPACE$?
- האם $EXP = NP$?

$$\boxed{P \subseteq NP \subseteq PSPACE \subseteq EXP \subseteq R}$$

כן ידוע: $P \neq EXP$

טענה: קיימת שפה L כך ש- $L \in R \setminus P$.

הוכחה:

נבנה L כזו ע"י תיאור של מ"ט U כך ש- $L = L(U)$ היא כמובטח.

U על קלט x:

- מפרשת את x כ- $x = 1^k 0 \langle M \rangle$.

- מסמלצת את M על x למשך 2^k צעדים.
- אם M עצרה במהלך הסימולציה, U מקבלת/וחה ההפך ממנה.
- אם M לא עצרה במהלך הסימולציה- U עוצרת ודוחה.

מתקיים: $L \in R$ (מהבנייה U עוצרת תמיד).

$L \notin P$: נניח בשלילה שקיימת מ"ט M הרצה זמן $p(|x|)$ עבור השפה L.

נבחר k מספיק גדול כך שמתקיים: $2^k \geq p(k+1+|\langle M \rangle|)$ (קיים כזה כי הפונקציה באגף שמאל אקספוננציאלית ובאגף ימין פולינומיאלית ב- k ו- $|\langle M \rangle|+1$ הוא מספר קבוע).

נתבונן בריצת U על הקלט: $1^k 0 \langle M \rangle$. על-פי האופן שבו בחרנו את k, בהכרח הסימולציה של M על x תסתיים בתוך פחות מ- 2^k צעדים. (מספר הצעדים הדרושים ל- M עד לעצירה: $p(|x|) = p(k+1+|\langle M \rangle|) \leq 2^k$). ולכן U ו- M יחזירו תשובות הפוכות על x ובפרט: $L(U) \neq L(M)$, סתירה. ולכן לא קיימת מ"ט פולינומית המקבלת את L ומתקיים: $P \neq R$.

<u>מסקנה-1:</u> קיימת שפה $L \in EXP \setminus P$ (השפה מהבנייה הקודמת).

<u>מסקנה-2:</u> קיימת שפה $L \in R \setminus EXP$ (אותה הבנייה בדיוק רק עבור 2^{2^k} צעדים).

שפות PSPACE שלמות:

<u>PSPACE שלמות:</u> שפה אי PSPACE שלמה אם"ם היא מקיימת:

1. $L \in PSPACE$.
2. לכל $L' \in PSPACE$, $L' \leq_p L$.

<u>מסקנה:</u> עבור L שהיא PSPACE-שלמה:

$$P = PSPACE \leftrightarrow L \in P$$

$$NP = PSPACE \leftrightarrow L \in NP$$

<u>דוגמאות:</u>

$$SAT = \{\psi \mid \psi = \exists x_1 \exists x_2 \cdots \exists x_n \varphi(x_1, x_2, \cdots, x_n)\ \varphi \text{ is } CNF, \psi \text{ is } TRUE\}$$

$$\overline{SAT} = \{\psi \mid \psi = \forall x_1 \forall x_2 \cdots \forall x_n \neg \varphi(x_1, x_2, \cdots, x_n)\ \varphi \text{ is } CNF, \psi \text{ is } TRUE\}$$

$$TQBF = \{\psi \mid \psi = Q_1 x_1 Q_2 x_2 \cdots Q_n x_n \varphi(x_1, x_2, \cdots, x_n)\ \varphi \text{ is } CNF, \psi \text{ is } TRUE\}$$

כאשר: $Q_i \in \{\forall, \exists\}$.

<u>משפט סביץ':</u> $PSPACE = NPSPACE$.