2) נראה רדוקציה $AL_{TM} \le_m E_{TM}$:

<u>M_f:</u> על קלט $\langle M,w\rangle$, כאשר M מכונת טיורינג ו-w מילה:

1. בנה מכונת טיורינג M' באופן הבא:
   1.1 עבור קלט x:
   1.2 הרץ את M על w, אם דחתה קבל
   1.3 אחרת, דחה
2. החזר את $\langle M'\rangle$

הרדוקציה תמיד עוצרת, מכיוון שהיא בונה מכונה ומחזירה אותה.

<u>נכונות:</u>

1. אם $\langle M,w\rangle \in AL_{TM}$ $\Leftarrow$ M <u>לא</u> דוחה את w, לכן M' תמשיך לשלב 1.3 ותדחה עבור ל x $\Leftarrow$ $L(M')=\emptyset$ $\Leftarrow$ $\langle M'\rangle \in E_{TM}$
2. אם $\langle M,w\rangle \notin AL_{TM}$ $\Leftarrow$ M תדחה את w, לכן M' תקבל בשלב 1.2 ל x $\Leftarrow$ $L(M') \neq \emptyset$ $\Leftarrow$ $\langle M'\rangle \notin E_{TM}$

(3) נתון: $A,B\neq\emptyset$, $A\in NP$, $B\in P$, $A\cup B$ היא NP שלמה.

נוכיח כי גם A היא NP-שלמה: נתון כי $A\in NP$, לכן נותר

להראות רדוקציה אליה משפה NP-שלמה אחרת.

נראה $A\cup B\leq_p A$

מכיוון שהשפה B שייכת ל-P, לכן קיימת מ"ט דטרמיניסטית M_B

המכריעה אותה בזמן פולינומיאלי.

השפה A היא לא שפה ריקה, לכן בהכרח קיימת

מילה y כך ש- $y\in A$.

נראה $A\cup B\leq_p A$:

<u>M:</u> עבור קלט $\langle x\rangle$:

1. הרץ את M_B על $\langle x\rangle$

1.1 אם דחתה - החזר $\langle x\rangle$

1.2 אחרת - החזר $\langle y\rangle$

הרדוקציה תמיד עוצרת - היא מריצה את M_B שתמיד עוצרת

ומחזירה מילה בהתאם לתוצאת הריצה של M_B.

<u>נכונות:</u>

1. אם $x\in A\cup B$ - נפצל לשני מקרים:

1. $x\notin B$ - במקרה הזה בהכרח $x\in A$ לכן נחזיר x. $\Leftarrow x\in A$

2. $x\in B$ - הקלט שייך ל-$A\cup B$, לכן נרצה שהפלט יהיה

שייך ל-A. כדי להבטיח זאת נחזיר מילה אחרת שבוודאות

שייכת ל-A - נחזיר את y. $\Leftarrow y\in A$

בשני המקרים קיבלנו שאם הקלט שייך ל-$A\cup B$ הפלט

שייך ל-A כנדרש.

2. אם $x\notin A\cup B$ - M_B תדחה את x ונחזיר x.

מכיוון ש-$x\notin A\cup B$ נקבל $x\notin A$.

<u>זמן ריצה</u>

M מריצה את M_B הרצה בזמן פולינומיאלי ומחזירה פלט

בהתאם $\Leftarrow$ זמן ריצתה של M הוא פולינומיאלי בגודל הקלט.

$A\in NP$ והראינו רדוקציה פולי' משפה NP שלמה אחרת $\Leftarrow$ A

היא שפה NP שלמה כנדרש.

400288

לשימוש הבודק

(4) נוכיח כי השפה CLIQUE-AND-HC (או בקיצור CAH)
היא NP-שלמה - נראה כי השפה שייכת ל-NP וכי קיימת
רדוקציה אליה משפה NP-שלמה אחרת.

CLIQUE היא NP שלמה, לכן קיימת מ"ט M_C המכריעה אותה
בזמן פולינומיאלי. כמו כן HC היא NPC, לכן גם לה קיימת מ"ט
M_{HC} המכריעה אותה בזמן פולינומיאלי.

נבנה מ"ט M_{CAH} שיכריע את CAH בזמן פולינומיאלי:

M_{CAH}: על קלט $\langle G,K \rangle$ כאשר G גרף לא מכוון ו-K טבעי:

1. הרץ את M_C על $\langle G,K \rangle$

1.1 אם קיבלה - הרץ את M_{HC} על $\langle G \rangle$

1.1.1 אם קיבלה - קבל

2. דחה

M_{CAH} מריצה את M_C כדי לבדוק האם בגרף G יש קליקה
בגודל K. אם M_C קיבלה, M_{CAH} תריץ את M_{HC} כדי לבדוק האם
ב-G יש מעגל המילטון. אם 2 המכונות קיבלו, משמע ש-G
יש קליקה בגודל K ומעגל המילטוני, לכן M_{CAH} תקבל.
אם ב-G אין מעגל המילטוני או אין קליקה בגודל K, אחת
מהמכונות M_C או M_{HC} תדחה ולכן גם M_{CAH} תדחה.

זמן ריצה

M_{CAH} מריצה 2 מ"טים שרצות בזמן ריצה פולינומיאלי
ומחזירה תשובה בהתאם לתשובתן, לכן זמן ריצתה
של M_{CAH} הוא פולינומיאלי בגודל הקלט. בנוסף היא תמיד עוצרת כי 2 המ"טים תמיד עוצרות
$\Leftarrow CAH \in NP$ - הראינו מ"ט שמכריעה אותה בזמן
פולינומיאלי.

כעת נראה את הרדוקציה $HC \le_p CAH$!

<u>M</u>: על קלט $\langle G \rangle$ כך ש-G גרף לא מכוון: $G=(V,E)$

1. בנה גרף G' באופן הבא:

1.1 העתק את צמתי וקשתות G ל-G'

1.2 הוסף ל-G' 3 צמתים: v_1,v_2,v_3

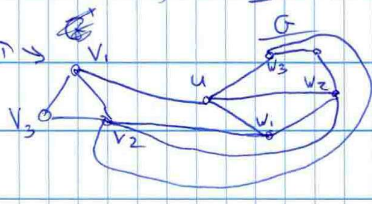
1.3 ~~הכ~~ הוסף את הקשתות: $(v_1,v_2), (v_1,v_3), (v_2,v_3)$

1.4 ~~[illegible]~~ עבור צומת $u \in V$ כלשהו הוסף את הקשת (v_1,u)

1.4.1 עבור כל צומת $w \in V$ כך שקיימת הקשת (u,w): הוסף את הקשת (v_2,w)

2. החזר $\langle G',3 \rangle$

הסבר - הרדוקציה לוקחת את הגרף G, מוסיפה לו קליקה בגודל 3, בוחרת צומת כלשהו ב-G (צומת אחד) ומחברת אותו לאחד מצמתי הקליקה. כדי להבטיח שאם ב-G היה מעגל המילטון יהיה גם ב-G', נחבר את כל השכנים של הצומת u לצומת אחר בקליקה. לדוגמה: ← הקליקה החדשה



ב-G היה מעגל המילטוני

וב-G' גם כן - נתחיל את המעגל ב-u, משם נמשיך ל-w_3, ונעבור דרך שאר הצמתים עד שנגיע ל-w_1, משם ל-$v_2 \to v_3 \to v_1$ וחזרה ל-u. חיברנו את כל השכנים של u ל-v_2 כדי למנוע מצב בו היינו בוחרים ב-w_2 וכך בגרף החדש לא היה מתקבל מעגל המילטוני.

<u>נכונות</u>

1. אם $G \in HC \Leftarrow$ ב-G קיים מעגל המילטוני. אותו מעגל יהיה גם ב-G'. הוספנו ל-G' קליקה וחיברנו אותה באמצעות מספר קשתות לגרף המקורי (מוסבר למעלה) לכן גם ב-G' יש מעגל המילטוני. בנוסף ב-G' יש קליקה בגודל 3 לכן $\langle G',3 \rangle \in CAH$

לשימוש הבודק

400288

4 המשך

2. אם $\langle G\rangle \notin HC$ ⇐ ב-G אין מעגל המילטוני ⇐
חסרה ב-G לפחות קשת אחת כדי שיהיה בו
מעגל המילטוני. מכיוון שהקשתות שהוספנו ב-G'
רק נועדו לחבר את הקליקה החדשה ל-G, ולא הוספנו
קשתות בין צמתי G המקוריים, נקבל שגם ב-G'
חסרה לפחות קשת אחת למעגל המילטוני ⇐ ב-G'
אין מעגל המילטוני ⇐ $\langle G,3\rangle \notin CAH$

<u>סיבוכיות זמן ריצה:</u>
הרדוקציה מעתיקה את הגרף G - פולינומיאלי בגודל הקלט,
מוסיפה לו קליקה בגודל 3 ומחברת אותה ל-G - מספר
הקשתות שנוספו חסום ע"י מספר הצמתים ב-G לכן
גם זה לוקח זמן פולינומיאלי בגודל הקלט ולבסוף מחזירה
את הגרף החדש ⇐ זמן ריצה של הרדוקציה הוא פולינומיאלי.

הראינו כי $CAH \in NP$ וכי קיימת רדוקציה פולי' $HC \le_p CAH$
⇐ CAH היא שפה NP שלמה, כנדרש.

א. ~~[illegible]~~

תהי שפה מזוהה טיורינג B <= קיימת מ"ט M_B המזהה אותה.
נבנה מ"ט טיורינג M שמזהה אותה לפי ההגדרה הדמוקרטית.
המכונה M תהיה זהה ל-M_B למעט השינויים הבאים:

- נוסיף את המצב q_B
- σ מעבר אל המצב המקבל ב-M_B יוחלף במעבר ל-q_B: $\delta(q,\sigma)=(q_B,\sigma',L\backslash R) \leftarrow \delta(q,\sigma)=(q_{acc},\sigma',L\backslash R)$
- נוסיף מעברים מ-q_B ל-q_{acc} ! מ-q_B לעצמו

נכונות:

עבור $w \in B$, קיים מסלול חישוב שיגיע איתו ל-q_B. משם, יש אינסוף אפשרויות להגיע למצב המקבל - ניתן ישר לעבור למצב המקבל או להישאר בלולאה מספר כלשהו של פעמים, כאשר ניתן להישאר גם מספר גדול מאוד, כזה שיהיה גדול יותר ממספר המסלולים החישוב הלא מקבלים ב-M (שקיימים כאלו, גם M_B היא מטל"ד).

אולי יש אינסוף מסלולי חישוב לא מקבלים?

עבור $w \notin B$, לא קיים ב-M_B מסלול מקבל => ב-M לא קיים מסלול שיגיע ל-q_B ולכן ~~[illegible]~~ יש 0 מסלולים מקבלים עבור w ב-M. לפי ההגדרה, גם אם מספר המסלולים המקבלים והדוחים שווה M תדחה, לכן גם אם אין מסלול דוחה ב-M עבור w M לא תקבל את w.

במקרה זה חייב להיות מסלול לא מקבל.

-1

שימוש הבודק

1

ב. נבנה מט"ד שיכריע את $\overline{A_{TM}}$ באופן דמוקרטי:

M': עבור קלט $\langle M,w \rangle$, M מ"ט ו-w מילה:

1. נחש מספר x
2. הרץ את M על w x צעדים לכל היותר
   2.1 אם M הספיקה לקבל - דחה
   2.2 אחרת - קבל

נכונות:

המכונה מנחשת מספר ומריצה את M על w כמספר הצעדים שנוחשו. אם M מקבלת את w, היא עושה זאת תוך מספר סופי של צעדים, נסמנו ב-n. לכן, עבור כל $n \le x$ M תקבל את w ו-M' תדחה את הקלט. מכיוון ש-x אינו מוגבל בערכו, נקבל שיש אינסוף אפשרויות דחייה ומספר סופי של אפשרויות קבלה ($n-1$ אפשרויות) ולכן לפי ההגדרה הדמוקרטית M' לא תקבל, כנדרש.

אם M לא מקבלת את w, לא קיים x עבורו M תקבל את w, לכן M' תקבל בשלב 2.2 עבור כל x ולעולם לא תדחה $\Rightarrow$ ישנן יותר אפשרויות קבלה $\Rightarrow$ M' תקבל את $\langle M,w \rangle$ לפי ההגדרה הדמוקרטית, כנדרש.

לשימוש הבודק

6

SAT $= \{\langle \phi \rangle \mid \phi$ is a satisfiable Boolean formula$\}$

כדי שנוסחה ϕ תהיה שייכת לשפה SAT, נדרש שיש לה לפחות השמה אחת מספקת.

נסמן ב-x את מספר המשתנים ב-ϕ, ל-ϕ ישנן 2^x השמות אפשריות.

הסיכוי של-ϕ יש לפחות השמה אחת מספקת הוא $\frac{1}{2^x}$.

בשאלה, n מייצג את אורך מילת הקלט. מתקיים $n \ge x$ שכן n כולל גם את הסימנים של הנוסחה ($\vee, \wedge, \dots$) ואת שלילת המשתנים. $\Rightarrow 2^n \ge 2^x \Rightarrow \frac{1}{2^n} \le \frac{1}{2^x}$

נבנה מ"ט הסתברותית עבור SAT:

M: על קלט $\langle \phi \rangle$, ϕ נוסחה בוליאנית:

1. קבע השמה לכל משתני הנוסחה
2. החזר את ערך השמת של ϕ תחת ההשמה שנקבעה.

ניתוח הסתברותי

M קובעת השמה ל-ϕ. אם ϕ לא ספיקה, לא קיימת אף השמה שמספקת אותה, לכן M דוחה את ϕ בהסתברות 1.

אם ϕ ספיקה, הסיכוי ש-M תבחר השמה מספקת ל-ϕ הוא $\frac{1}{2^x}$ אם יש השמה אחת כזו או גדול מכך אם ישנה יותר מהשמה אחת. לכן, הסיכוי ש-M תקבל את ϕ אם ϕ ספיקה הוא לפחות $\frac{1}{2^x}$.

כפי שהוסבר למעלה $\frac{1}{2^n} \le \frac{1}{2^x}$ לכן M עומדת בדרישות - דוחה בהסתברות 1 ומקבלת בהסתברות לפחות 2^{-n}.

$\Rightarrow$ SAT $\in$ LPP.

M רצה בזמן פולינומיאלי באורך הקלט, שכן קביעת ההשמה לוקחת זמן פולי' ובדיקה האם ההשמה מספקת לוקחת גם היא זמן פולי'.