הרצאה 11 – יום רביעי 27.8

מכונות טיורינג הסתברותיות – *PTM*

מ"ט הסתברותית M היא סוג של NTM שבה כל צעד לא-דטרמיניסטי נקרא צעד **הטלת מטבע** שיש בו שני צעדים אפשריים. כל פעם שה-TM מבצע צעד הטלת מטבע, הוא נכנס לענף לפי הבחירה האקראית הזו – בהסתברות $1/2$ לכל ענף.

ההסתברות של ענף b היא: $\mathbb{P}[b]=(1/2)^k$, כאשר k הוא מספר צעדי ההטלה שיש בענף b.

$$\mathbb{P}[M \text{ accepts } x]=\sum_{b:b\ accepts\ x}\mathbb{P}[b]$$

מחלקת *PP*

נאמר ש-M PTM מכריע שפה L עם טעות $0\le\varepsilon\le 1/2$ אם מתקיים:

$$x\in L\Longrightarrow \mathbb{P}[M(x)=1]\ge 1-\varepsilon,\qquad x\notin L\Longrightarrow \mathbb{P}[M(x)=0]\ge 1-\varepsilon$$

מחלקת *Probabilistic Polynomial Time (PP)* היא מחלקת השפות שמוכרעות ע"י PTM בזמן פולינומי עם טעות לכל היותר $1/2$.

מחלקת *BPP*

כמו PP, אבל הטעות היא עד $1/3$.

הערה: זו ההגדרה בקורס. הגדרה שקולה היא שהטעות היא $0<\varepsilon<1/2$, כי אפשר פשוט להריץ את המכונה כמה פעמים שנרצה עד שנקבל את הסתברות שאנחנו רוצים. זה המשפט הבא:

Amplification lemma

יהי קבוע $0<\varepsilon<1/2$. אזי לכל פולינום $p(n)$, לכל M PTM עם טעות ε יש M' PTM שקול, עם טעות $2^{-p(n)}$.

כלומר: אם יש מכונה עם טעות קטנה ממש מחצי, אפשר להריץ אותה כמה פעמים שצריך עד שהטעות תהיה קטנה ככל שנרצה. (ממש אותו רעיון שיש באלגוריתמים הסתברותיים בהסתברות 2). ובגלל שמספר הפעמים שנצטרך להריץ הוא פולינום שתלוי ב-n, אז גם M' הוא פולינומי.

<u>הוכחה</u>: בהינתן M, נבנה את M'. בהינתן קלט x, נבצע:

1. נחשב את $k:=-p(n)\ /\ \big(\log\big(4\varepsilon(1-\varepsilon)\big)\big)$. (נראה בהמשך למה המספר הזה).
2. נריץ $2k$ סימולציות בלתי-תלויות של $M(x)$.
3. אם לפחות $k+1$ מההרצות קיבלו, נקבל. אחרת, נדחה.

נשים לב שהחישוב של k הוא פולינום חלקי קבוע, שזה פולינום. ו- $0\le\varepsilon(1-\varepsilon)\le 1/2$ אז הלוג יהיה שלילי אז k יהיה חיובי.

ההסתברות לטעות: נסמן c את מספר התוצאות הנכונות של M, ו-w את מספר התוצאות השגויות. מתקיים $2k=c+w$.

סדרה "רעה" S היא סדרה שבה $c\le w$, וההסתברות לסדרה כזו היא:

$$\mathbb{P}[S]=\varepsilon^w(1-\varepsilon)^c\le\varepsilon^k(1-\varepsilon)^k$$

כאשר האי-שוויון נובע מכך ש- $0<\varepsilon<1/2$, אז כשנקטין את החזקה של ε, ושנקטין את החזקה של $(1-\varepsilon)$, נקבל משהו גדול יותר.

ההסתברות ש- $M'(x)$ טועה:

$$\sum_{bad\ S}\mathbb{P}[S]\le\varepsilon^k(1-\varepsilon)^k\sum_{bad\ S}1\le 2^{2k}\varepsilon^k(1-\varepsilon)^k=4^k\varepsilon^k(1-\varepsilon)^k=\big(4\varepsilon(1-\varepsilon)\big)^k$$

אז בשביל הדרישה $\big(4\varepsilon(1-\varepsilon)\big)^k\le 2^{-p(n)}$, נדרוש: $k\ge -p(n)\ /\ \big(\log\big(4\varepsilon(1-\varepsilon)\big)\big)$.

הערה: באופן כללי, ההגדרות של PP ו-BPP קצת שונות. בדרך כלל PP מוגדרת ע"י הסתברות לטעות קטנה ממש מחצי, אבל לא מוגדר עד כמה קטן מחצי – זה יכול להיות גם משהו שלא קבוע, תלוי ב-n. ואז אי אפשר להשתמש בחסם צ'רנוף כמו באלגוריתמים הסתברותיים (או שזה דורש יותר ממספר פולינומי של חזרות). ההבדל העיקרי הוא ש-BPP דורש חסם קבוע של טעות, וזה מאפשר את זה שנשתמש במספר קבוע של חזרות כדי להקטין את ההסתברות.

דוגמה 1 – מבחן פרמה *Fermat's Test*

המשפט הקטן של פרמה: לכל מספר ראשוני p ולכל מספר שלם $a \in \{1, \dots, p-1\}$, מתקיים: $a^{p-1} \equiv 1(mod\ p)$.

1. נבחר $a_1, \dots a_k$ אקראיים מתוך $\{1, \dots, p-1\}$.
2. נחשב את $a_i^{p-1}(mod\ p)$ לכל i. העלאה בחזקה מודולו דורשת זמן פולינומי (מופיע במטלה 2).
3. אם כולם שווים 1, נקבל. אחרת, נדחה.

נשים לב: המשפט אומר שאם p ראשוני, אז התנאי מתקיים. האלגוריתם בודק את התנאי ומניח שאם התנאי מתקיים עבור מספיק מספרים, אז p ראשוני. אבל יש מספרים שעבורם התנאי מתקיים ($a^{n-1} \equiv 1(mod\ n)$) אבל הם לא ראשוניים. מספרים כאלה נקראים מספרי קרמייקל (*Carmichael*).

עבור k קבוע, ההסתברות לטעות היא לכל היותר 2^{-k} (חוץ ממספרי קרמייקל, שעבורם הטעות היא 1). נאמר שמבחן פרמה מכריע את השפה:

$$PSEUDOPRIMES \coloneqq \{p : p \text{ is prime or Carmichael}\}$$

דוגמה 2 – מבחן מילר-רבין *Miller-Rabin*

נשפר את מבחן פרמה.

משפט "שורש ריבועי של 1": לכל מספר ראשוני p ולכל מספר שלם $a \in \{2, \dots, p-2\}$, מתקיים: $a^2 \neq 1(mod\ p)$.

ננצל גם את העובדה שלכל מספר n יש פירוק ייחודי: $n = s \cdot 2^t$ כך ש-s אי-זוגי.

מבחן מילר-רבין (*Miller-Rabin*):

1. אם p זוגי: אם $p = 2$ נקבל, אחרת נדחה.
2. נבחר $a_1, \dots a_k$ אקראיים מתוך $\{2, \dots, p-2\}$.
3. נמצא את ה-s האי-זוגי שמקיים $p - 1 = s \cdot 2^t$.
4. לכל $i \in \{1, \dots k\}$:
   a. אם $a_i^{p-1}(mod\ p) \neq 1$, נדחה.
   b. נחשב את הסדרה: $b_0 = a_i^s\ (mod\ p),\ \ b_{j+1} = b_j^2\ (mod\ p)$ עבור $j \in \{0, \dots, t-1\}$.
   c. אם אין את $(p-1)$ לפני ה-1 הראשון בסדרה, נדחה.
5. נקבל.

כמו שאמרנו מקודם, חזקה במודולו דורשת רק זמן פולינומי. עבור k קבוע, ההסתברות לטעות היא לכל היותר 4^{-k}. אז המבחן מכריע את השפה:

$$PRIMES \coloneqq \{p : p \text{ is prime}\}$$

אז $PRIMES \in BPP$. (מאז 2002 ידוע ש- $PRIMES \in P$, אבל שיהיה).

מחלקת *RP – Randomized Polynomial Time*

כל השפות שמוכרעות ע"י PPT שעבורן:

$$x \in L \Longrightarrow \mathbb{P}[M(x) = 1] \geq 1/2\,, \qquad x \notin L \Longrightarrow \mathbb{P}[M(x) = 0] = 1$$

כלומר אם המילה לא בשפה, תמיד נתפוס את זה. אם המילה לא בשפה, אולי נפספס אותה. (אלגוריתם מונטה קרלו עם טעות חד צדדית, הסתברות 2).

אז לדוגמה, השפה $COMPOSITES \coloneqq \{m : m \text{ is composite } (non-prime)\}$ היא ב-RP, כי מבחן מילר-רבין תמיד תופס את הראשוניים.

מאותה סיבה, $PRIMES$ היא ב- $coRP$.

מחלקת *ZPP – Zero-error Probabilistic Polynomial Time*

$$ZPP \coloneqq RP \cap coRP$$

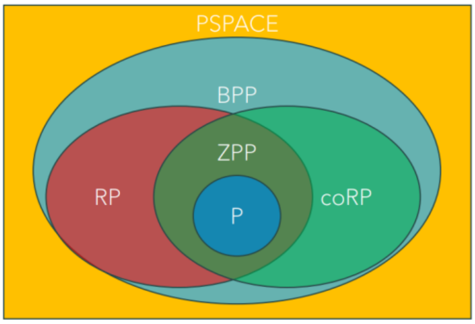
שפות שמוכרעות ע"י PTM שתמיד צודק, וזמן הריצה הוא פולינומי **בתוחלת** (אלגוריתם לאס וגאס).

לחלופין, אפשר להגדיר ע"י שפות שמוכרעות ע"י PTM שתמיד רץ בזמן פולינומי, ותמיד מחזיר תשובה: כן, לא, לא יודע. אם התשובה היא כן או לא, היא בוודאות נכונה. ההסתברות ל"לא יודע" היא לכל היותר $1/2$ לכל קלט.

מתקיים: $ZPP = coZPP$. שאלה פתוחה: האם $P =_? coRP =_? RP$. לא ידוע, ההשערה היא שכן.

כל המחלקות ההסתברותיות הפולינומיות מוכלות ב- $PSPACE$, כי אפשר להריץ אותם במקום פולינומי. גם אם צריך לעשות הרבה הרצות, בין הרצות צריך לשמור רק את מספר ההצלחות / כישלונות, וזה דורש רק מקום לוגריתמי.

לסיכום:



.