סיבוכיות – תרגול 8

כל העולם הוא חישוב אחד גדול

12.4.11

חישוב הסתברותי

וידוא כפל מטריצות
קלט: שלוש מטריצות A, B, C בגודל $n \times n$ שאיבריהן ב- $\{0,1\}$.
שאלה: האן $A \cdot B = C$? (כאשר פעולות החיבור והכפל הן מודולו 2).
הפתרון הנאיבי: לחשב את המכפלה AB ולבדוק האם שווה ל- C. כל איבר של AB ניתן לחשב בזמן $O(n)$ ולכן זמן הריצה הדרוש כאן $O(n^3)$. נדגיש ש- n אינו מסמן כאן את גודל הקלט. עבור קלט באורך k אלגוריתם זה מצריך זמן $O(k^{3/2})$.

הערה. יש אלגוריתמים יעילים יותר לכפל מטריצות: [Strassen,1969] $O(n^{\log_2 7}) = O(n^{2.81})$ (מומלץ בחום להכנס לויקיפדיה ולראות את האלגוריתם). היעילות הטובה ביותר הידועה היא $O(n^{2.37})$.

רעיון: נסתכל על אלגוריתם שמגריל איבר במטריצה C ובודק אותו תוך שימוש בשורה המתאימה ב- A ובעמודה המתאימה ב- B. במקרה ש- $AB \neq C$ הסיכוי לתפוס זאת עשוי להיות $\frac{1}{n^2}$ וכדי לקבל סיכוי הצלחה קבוע יידרשו $O(n^2)$ חזרות ואז נקבל זמן ריצה $O(n^3)$ שאין לו עדיפות על פני האלגרותים הנאיבי.
אלגוריתם הסתברותי:

1. נגריל וקטור $x \in \{0,1\}^n$ מתוך התפלגות אחידה.
2. נחשב Cx וכן את ABx. אם הוקטורים שווים נקבל, ואחרת נדחה.

ניתוח זמן הריצה: חישוב מכפלת מטריצה בוקטור מצריך זמן $O(n^2)$ כי כדי לחשב כל קורדינטה מהוקטור המתקבל דרוש זמן $O(n)$. נחשב את Cx. כדי לחשב את ABx קודם נחשב את Bx ואז נכפיל את A בוקטור המתקבל. סה"כ $O(n^2)$.
ניתוח האלגוריתם:
כאשר $AB = C$ מתקיים לכל x, $ABx = Cx$ ולכן האלגוריתם מקבל בהסתברות 1.
נניח כעת $AB \neq C$. השאלה היא, מהיא ההסתברות בבחירת x אקראי ש- $ABx \neq Cx$ או באופן שקול $(AB - C)x \neq 0$. נסמן $D = AB - C$ ($D \neq 0$).

טענה. תהא D מטריצה $n \times n$, $D \neq 0$. אזי $\mathbb{P}_x[Dx \neq 0] \geq \frac{1}{2}$ (כאשר x מתפלג אחיד).

הוכחה. תהא $D \neq 0$. נניח בה"כ $d_1 \neq 0$.

$$Dx = \begin{pmatrix} | & | & & | \\ d_1 & d_2 & \dots & d_n \\ | & | & & | \end{pmatrix} \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix} = \begin{matrix} | & | & & | \\ x_1 \cdot d_1 & + x_2 \cdot d_2 & + \dots + & x_n d_n \\ | & | & & | \end{matrix}$$

נשים לב שלכל בחירה של $x_2,\ldots,x_n \in \{0,1\}$ הוקטורים הבאים שונים: $D\begin{pmatrix}1\\x_2\\\vdots\\x_n\end{pmatrix}$, $D\begin{pmatrix}0\\x_2\\\vdots\\x_n\end{pmatrix}$ כי ההפרש ביניהם $d_1 \neq 0$. לכן, לפחות אחד מהם אינו וקטור האפס. קיבלנו שמספר ה- x עוברם $Dx \neq 0$ הוא לפחות 2^{n-1}. לכן,

$$\mathbb{P}_x[Dx \neq 0] = \frac{2^{n-1}}{2^n} = \frac{1}{2}$$

□

כדי לשפר את סיכוי ההצלחה ניתן להריץ את האלגוריתם k פעמים ולקבל אם ורק אם כל k ההרצות קיבלו. עדיין כאשר $AB = C$ סיכוי ההצלחה הוא 1 וכאשר $AB \neq C$ האלגוריתם טועה בהסתברות $\geq \frac{1}{2^k}$. זמן הריצה $O(kn^2)$.

תרגיל. הוכיחו שאם $SAT \in BPP$ אז $SAT \in RP$.

פתרון.

$$BPP = BPP\left(\frac{1}{3},\frac{2}{3}\right) = BPP(2^{-n}, 1-2^{-n})$$

$$RP = BPP\left(0,\frac{1}{2}\right)$$

רעיון ההוכחה:

כזכור, בהינתן אלגוריתם לבעית ההכרעה SAT עם זמן ריצה פולינומיאלי, ניתן למצוא השמה מספקת (אם ישנה) לנסוחא בזמן פולינומיאלי (תוך שימוש ב- n קריאות לאלגוריתם ההכרע, n - מספר משתני הנוסחא). נניח $SAT \in BPP$ אזי בעזרת אמפליפיקציה יש אלגוריתם פולינמיאלי שמכריע את SAT באופן הסתברותי עם סיכוי כישלון $\geq 2^{-n}$. נריץ אלגוריתם למציאת השמה מספקת שמשתמש באלגוריתם ההכרעה הנ"ל. בסוף התהליך נבדוק אם ההשמה מספקת. אם כן נקבל ואחרת נדחה.
$\varphi \notin SAT \Rightarrow$ האלגוריתם משיב ש- φ ספיקה בהסתברות 0.
כאשר $\varphi \in SAT$ האלגוריתם יענה נכון במקרה שכל הקריאות לבעית ההכרעה נתנו תשובות נכונות. הסיכוי שלפחות אחת מהקריאות נתנה תשובה שגויה הוא $\geq n \cdot 2^{-n}$: לכן $\frac{1}{2} > 2$ עבור φ ספיקה תוחזר תשובה נכונה בסיכוי $\leq \frac{1}{2}$ כנדרש.