סיבוכיות / תרגיל בית 3#

אריאל סטולרמן

(1)

(*a*)

להלן אלגוריתם המקבל כקלט את הזוג $< G, k >$ ($G = (V, E)$ גרף לא מכוון) ומחזיר את תת הקבוצה $C \subseteq V$ המהווה קליק בגודל k בגרף G, או דוחה אם לא קיים כזה, המשתמש באלגוריתם הכרעת השפה *Clique* שנסמנו A. נסמן $V(G) = \{v_1, \dots, v_n\}$.

1. תחילה בדוק האם $A(< G, k >) = T$. אם לא, דחה מיד.
2. לכל $1 \leq i \leq n$ בצע:
   - הגדר $G^i = (V \backslash \{v_i\}, E \backslash \{\{v_i, v_j\} | \{v_i, v_j\} \in E\})$, ובדוק האם $A(< G^i, k >) = T$.
   - אם כן, זרוק את v_i.
   - אחרת החזר את v_i ל-V והמשך לאיטרציה הבאה.
3. עבור G^n מהאיטרציה האחרונה, החזר את $C = V(G^n)$.

נכונות:

תחילה, ברור כי אם אין ב-G קליק בגודל k, עלינו לדחות מיד. אם כן קיים קליק, אנו בודקים לכל קודקוד האם ניתן להסתדר בלעדיו. כלומר, האם ללא קודקוד זה עדיין יש לנו קליק בגודל k בגרף. יתכנו מספר קליקים בגרף, אך כל הקודקודים בהם נתקל לראשונה השייכים לקליקים, האלג' יעיף אותם, עד שיהיה מצב בו נותר קליק אחד בגודל k בגרף G^i עבור i כלשהו. מנקודה זו ואילך, ברור שהוצאת כל קודקוד מקליק זה תביא לתשובה F מ-$A(< G^j, k >)$ – וקודקוד זה ישאר לאיטרציה הבאה, וכל קודקוד שאינו בקליק יביא לתשובה T - ויועף. לבסוף נותר G^n, לאחר מעבר על כל הקודקודים, והוא בעל קבוצת קודקודים בגודל k בדיוק שהיא קליק כנדרש.

כיוון שלאורך הלולאה מצמצמים את G ולא מוסיפים לו קשתות או קודקודים חדשים, אז ברור שהקליק ב-G^n הוא גם קליק ב-G.

זמן ריצה:

תחת הנחה שזמן ביצוע אלג' ההכרעה הוא יחידת זמן אחת: השלב הראשון לוקח $O(1)$; השלב השני רץ על כל אחד מ-n הקודקודים ולכל אחד מבצע מספר הסרה כך שסה"כ ההסרות בכל n האיטרציות הוא $O(|V| + |E|)$; השלב השלישי גם כן $O(1)$. לפיכך זמן הריצה $O(n)$ - פולי' באורך הקלט כנדרש.

(*b*)

להלן אלגוריתם המקבל כקלט את זוג הגרפים $< G_1, G_2 >$ ומחזיר את האיזומורפיזם $f: V(G_1) \to V(G_2)$ או דוחה אם לא קיים כזה, המשתמש באלגוריתם הכרעת השפה *GraphIsomorphism* שנסמנו A:

1. תחילה בדוק האם $A(< G_1, G_2 >) = T$. אם לא, דחה מיד.
2. אם כן, יהיו $V(G_1) = \{v_1, \dots, v_n\}$ ו-$V(G_2) = \{u_1, \dots, u_n\}$. לכל $1 \leq i \leq n$:
   - הסר מ-$V(G_1)$ את v_i ומ-$E(G_1)$ את כל הקשתות בהם משתתף v_i.
   - לכל $j \in unmapped$ בדוק:
     - הסר מ-$V(G_2)$ את u_j ומ-$E(G_2)$ את כל הקשתות בהם משתתף u_j.
     - בדוק האם $A(< G_1, G_2 >) = T$.
     - אם כן: $f(v_i) \leftarrow u_j$, $unmapped(j) \leftarrow F$, ועבור לאיטרציה הבאה של ***i***.
     - אם לא, החזר את G_2 לקדמותו ועבור לאיטרציה הבאה של ***j***.
3. החזר את f.

נכונות:

תחילה, ברור כי אם אין איזו' בין G_1 ל-G_2, עלינו לדחות מיד. אם קיים איזו', נטען כי המיפוי המתבצע בכל שלב הוא נכון.

- עבור $i = 1$: ידוע כי קיים איזו' כנדרש, אזי קיים מיפוי $f(v_1) = u_j$ עבור j כלשהו, המתאים לקודקוד v_1 איזשהו קודקוד u_j המתאים לו איזומורפית, ולכן הסרת הקודקודים משני הגרפים והקשתות היוצאות מהן בהתאמה אמורות להשאיר את הגרפים הנותרים איזומורפים אחד לשני.
- נניח נכונות עד k, ונראה נכונות עד $k+1$: באופן דומה, ידוע שהגרפים שהתקבלו עד כה המכילים $n-k$ קודקודים הם איזומורפים, ולכן קיים מיפוי $f(v_{k+1}) = u_j$ עבור j כלשהו, כך שהסרת v_{k+1}, u_j והקשתות היוצאות מהן משני הגרפים בהתאמה תשאיר את שני הגרפים איזומורפיים. כמו כן, בכל מיפוי מסירים את הקודקוד שמופה ומסמנים אותו, כך שלא יתכן מיפוי נוסף אליו באיטרציות הבאות.

התהליך עובר על כל הקודקודים ולכן יגיע לשני הגרפים הריקים, ויתקבל האיזו' f.

זמן ריצה:

לפי הנחה השלב הראשון של בדיקת קיום איזו' לוקח $O(1)$. השלב השני לוקח $O(n^2)$ שכן רץ לכל קודקוד ב-$V(G_1)$ על (בערך) כל קודקוד ב-$V(G_2)$, והפעולות שמבצע בכל איטרציה מהירות. השלב השלישי מיידי. לפיכך זמן הריצה של האלג' הוא $O(n^2)$ ולכן פולי' כנדרש.

(2)

תהי $L \subseteq \{1\}^*$, ותהי $bin(L) = \{bin(n) | 1^n \in L\}$. להלן הוכחה ש-$L \in P \Leftrightarrow bin(L) \in E = \bigcup_{c\geq 1} DTIME(2^{cn})$:

תחילה נניח כי $L \in P$, אזי קיים פולינום כלשהו p ואלג' A הרץ בזמן $p(n)$ המכריע לכל x ($|x| = n$) האם $x \in L$ או לא. נבנה אלג' B המכריע את השפה $bin(L)$: בהינתן מספר בינארי כלשהו באורך k, נבנה ממנו את ייצוגו האונרי ע"י חישוב הייצוג העשרוני שהוא לכל היותר $2^k - 1$, ויצירת המספר האונרי ע"י שרשור 1'ים כמספר שקיבלנו (לכל היותר 1^{2^k-1}). נפעיל את אלג' A על המספר האונרי שחישבנו ונחזיר את התשובה של A.

נכונות האלג' ברורה, שכן פועל לפי הגדרת השפה – משחזר מתוך מספר בינארי את ייצוגו האונרי ובודק שייכות ל-L. זמן הריצה: לכל קלט בינארי באורך k, ייצור ייצוגו האונרי ייקח לכל היותר $2^k - 1$ צעדים. הפעלת אלג' A על מספר זה תקח $p(2^k - 1)$ זמן לכל היותר, וכיוון ש-p הוא פולינום, יתקבל סדר גודל של 2^{ck} צעדים בזמן הריצה (אם p הוא פולינום מסדר $c \geq 1$ כלשהו, אז $(x^c|_{2^k} = (2^k)^c = 2^{ck}$). לפיכך B רץ בזמן $O\left(2^k \cdot p(2^k)\right)$ ולכן קיים c כך ש-$bin(L) \in DTIME(2^{cn}) \subseteq E$, כנדרש.

כעת נניח כי $bin(L) \in \bigcup_{c\geq 1} DTIME(2^{cn})$, אזי קיים אלג' B המכריע את $bin(L)$ בזמן $O(2^{cn})$ לכל x (מספר בינארי כך ש-$|x| = n$). נבנה אלג' A המכריע את L בזמן פולינומיאלי: בהינתן קלט אונרי $x = 1^k$, נעביר אותו לייצוג בינארי ונקבל מספר בינארי באורך $O(\log k)$. נבדוק באמצעות B האם מספר זה שייך ל-$bin(L)$, ונחזיר את תוצאת B.

נכונות האלג' ברורה, שכן פועל לפי הגדרת השפה – מעביר מספר אונרי לייצוגו הבינארי ובודק שייכות ל-$bin(L)$. זמן הריצה: לכל קלט אונרי באורך k, ייצור הייצוג הבינארי ייקח $O(k)$ צעדים, ויתקבל מספר בינארי באורך $O(\log k)$. הפעלת אלג' B על מספר זה תקח $O(2^{c\cdot \log k})$ וזה כמובן שווה ל-$O(2^c \cdot k)$ כאשר 2^c הוא קבוע כלשהו. סה"כ קיבלנו כי זמן הריצה הוא $O(2^c \cdot k + k) = O(k)$ ולכן $L \in P$, כנדרש.

(3)

$UpToOneSat \in NP \Leftrightarrow NP = coNP$-להלן הוכחה ש-$UpToOneSat = UTOS = \{\varphi \in CNF |$ {ל $-\varphi$ לכל היותר השמה אחת מספקת}:

תחילה נניח כי $NP = coNP$. מתקיים כי $UTOS \in NP \Leftrightarrow \overline{UTOS} \in coNP$, ו-{ל $-\varphi$ לפחות שתי השמות מספקות} $\overline{UTOS} = \{\varphi \in CNF |$. כיוון ש-$NP = coNP$, אז מספיק להוכיח ש-$\overline{UTOS} \in NP$, ובעיה זו דומה לבעיה SAT: עבור קלט φ ניקח כעד שתי השמות מספקות ל-φ, והאלג' המוודא יבדוק שאכן שתי השמות אלה מספקות את φ. ברור כי העד בגודל פולי' לגודל הקלט, וכי האלג' רץ בזמן פולי' גם כן, ולכן $\overline{UTOS} \in NP$. לפיכך מתקיים כי $UTOS \in NP$, כנדרש.

כעת נניח כי $UTOS \in NP$. נראה רדוקציה $\overline{SAT} \leq_p UTOS$: בהינתן $\varphi \in CNF$, נחזיר את הנוסחה $\psi = \varphi \wedge (y \vee \neg y)$, כאשר y הוא משתנה חדש שלא מופיע ב-φ. ברור כי מספר ההשמות המספקות את ψ כפול מאלו המספקות את φ כיוון שלכל השמה מספקת $(x_1, \dots, x_n)$ של φ ניתן להציב ב-ψ את אותה השמה פעם אחת עם $y = t$ ופעם שניה עם $y = f$. כמו כן ברור כי בנית הרדוקציה פולי'. נניח כי $\varphi \in \overline{SAT}$, אזי ל-$\varphi$ אין הצבות מספקות, ולכן גם ל-ψ, ולפיכך $\psi \in UTOS$. נניח כי $\varphi \notin \overline{SAT}$, אזי ל-$\varphi$ לפחות הצבה מספקת אחת, ולכן ל-ψ לפחות 2 הצבות מספקות ולכן $\psi \notin UTOS$. לפיכך $\varphi \in \overline{SAT} \Leftrightarrow \psi \in UTOS$, ולכן $\overline{SAT} \leq_p UTOS$. כיוון שמצאנו רדוקציה משפה ב-$coNPC$ לשפה ב-NP, אז יש רדוקציה מכל שפה ב-$coNP$ לשפה ב-NP, ולכן $coNP \subseteq NP$. מכאן מתקיים גם $co(coNP) = NP \subseteq co(NP)$ ולכן $coNP = NP$, כנדרש.

(4)

להלן הוכחה לכל ש-$NICE = NP \cap coNP$:

תהי $L \in NICE$, אזי קיימת מכונת $nice$ לא דטרמיניסטית הרצה בזמן פולי' שנסמנה A המקבלת את L. לכל $x \in L$ מתקיים: קיים לפחות מסלול חישוב אחד במכונה A המחזיר $accept$ בריצה על x. כמו כן, כל מסלול שאינו מחזיר $quit$ בהכרח יחזיר $accept$, אחרת ישנה סתירה בפלט של A ו-A לא מקבלת את L בסתירה להנחה. לכל $x \notin L$ מתקיים באופן דומה שקיים לפחות מסלול חישוב אחד במכונה A המחזיר $reject$, וכל מסלול שאינו מחזיר $quit$ בהכרח יחזיר $reject$ מאותם שיקולים. כיוון ש-A פולי', אז לכל קלט x כך ש-$|x| = n$, A רצה בזמן $p(n)$ עבור p פולינום כלשהו. מהחלק הראשון (לכל $x \in L$) נובע ש-$L \in NTIME(p(n)) \subseteq NP$, ומהחלק השני (לכל $x \notin L$) נובע ש-$\bar{L} \in NTIME(p(n)) \subseteq NP$. לפיכך מתקיים ש-$L \in NP \cap coNP$, כנדרש.

תהי $L \in NP \cap coNP$. כיוון ש-$L \in NP$ אז קיימת מכונה A לא דטר' הרצה בזמן פולי' המקיימת: לכל $x \in \{0,1\}^*$ קיימת בחירה אי-דטר' הגורמת ל-A לקבל $x \in L \Leftrightarrow$. כיוון ש-$L \in coNP$ אז $\bar{L} \in NP$ ולכן קיימת מכונה B לא דטר' ופולי' המקיימת: לכל $x \in \{0,1\}^*$ קיימת בחירה אי דטר' הגורמת ל-B לקבל $x \in \bar{L} \Leftrightarrow x \notin L$. נשים לב כי בכל אחת מהמכונות יתכנו מספר מסלולים מקבלים עבור קלט x, וכי אם המכונה לא אמורה לקבל את הקלט, כל מסלול בה ידחה.

נבנה את המכונה C (לא דטר') הפועלת באופן הבא על קלט $x \in \{0,1\}^*$: מריצה עליו את A ו-B במקביל (באופן לא דטר'), כאשר A מוגדרת להחזיר $accept$ אם מקבל ו-$quit$ אחרת, ו-B מוגדרת להחזיר $reject$ אם מקבלת ו-$quit$ אחרת. נסמן את זמני הריצה (של כל מסלול חישוב בגודל n) של A ב-$p(n)$ ושל B ב-$q(n)$, כאשר p,q פולינומים כלשהם. מתקיים: אם $x \in L$, כל מסלולי החישוב של B יוציאו $quit$, ויהיה לפחות מסלול חישוב אחד של A שיוציא $accept$ (שאר המסלולים יוציאו $accept$ אם מקבלים או $quit$ אם לא). באופן דומה אם $x \notin L$, כל מסלולי החישוב של A יוציאו $quit$, ויהיה לפחות מסלול חישוב אחד של B שיוציא $reject$. זמן הריצה של כל מסלול חישוב ב-C הוא $O(\max\{p(n), q(n)\})$, וזה הרי פולי'. לפיכך C היא מכונה מסוג $nice$ המקבלת את L ולכן $L \in NICE$ כנדרש.

(5)

להלן הוכחה לכך ש-$SAT - UNSAT = \{< \varphi_1, \varphi_2 > \mid$ ספיקה ו φ_1 – φ_2 לא ספיקה$\}$ היא DPC (*DP*-שלמה):

תחילה נוכיח כי $SAT - UNSAT \in DP$: נגדיר $SAT - CNF = \{< \varphi_1, \varphi_2 > | \varphi_1 \in SAT, \varphi_2 \in CNF\}$. ברור כי שפה זו היא NPC: בהינתן $< \varphi_1, \varphi_2 >$ ניקח כעד השמה מספקת ל-φ_1 והאלג' המוודא יבדוק ש-φ_1 אכן מסתפקת מהשמה זו (וש-φ_2 היא נוסחת CNF). ברור כי הבדיקה ואורך העד פולי', ולכן השפה ב-NP. כמו כן ישנה רדוקציה מיידית $SAT - CNF \leq_p SAT$: לכל קלט φ נחזיר את הקלט $< \varphi, \varphi >$ - ברור שרדוקציה זו פולי' ושמתקיים: $\varphi \in SAT$ אמ"מ קיימת ל-φ הצבה מספקת ו-φ היא נוסחת CNF אמ"מ $< \varphi, \varphi > \in SAT - CNF$. לפיכך הוכחנו כי $SAT - CNF \in NPC$.

באופן דומה נגדיר $CNF - UNSAT = \{< \varphi_1, \varphi_2 > | \varphi_1 \in CNF, \varphi_2 \in \overline{SAT}\}$, ובאופן דומה ניתן להוכיח ש-$CNF - UNSAT \in coNPC$. ברור כי $SAT - UNSAT = SAT - CNF \cap CNF - UNSAT$, ולכן $SAT - UNSAT \in DP$.

תהי $L \in DP$, כלומר קיימות $L_1 \in NP, L_2 \in coNP$ כך ש-$L = L_1 \cap L_2$. נראה שקיימת רדוקציה $L \leq_p SAT - UNSAT$: כיוון ש-$L_1 \in NP$ אזי $L_1 \leq_p SAT - CNF$, כי $SAT - CNF \in NPC$ כפי שהוכח לעיל. כמו כן כיוון ש-$L_2 \in coNP$ אזי $L_2 \leq_p CNF - UNSAT$, כי $CNF - UNSAT \in coNPC$. יהיו f, g רדוקציות אלו בהתאמה, אזי לכל $x \in L = L_1 \cap L_2$ ניתן להחזיר את $h(x) =< \pi_1(f(x)), \pi_2(g(x)) >$, וברור כי $x \in L \Leftrightarrow h(x) \in SAT - UNSAT$, מהגדרת *SAT-UNSAT* כחיתוך *SAT-CNF* עם *CNF-UNSAT*. כמו כן זמן הריצה של הרדוקציה הוא חיבור זמני הריצה של f ו-g שהן פולי' כמוכח לעיל. לכן לכל $L \in DP$ קיימת רדוקציה פולי' ל-*SAT-UNSAT*, ולכן $SAT - UNSAT \in DP$, כנדרש.