13. חיפוש פתרונות כנגד אישוש נכונותם ("P" כנגד "NP")

כפי שראינו בפרק 7, חישוב פונקציות הוא מקרה פרטי של מציאת פתרונות ליחס דו-מקומי. נזכיר את הגדרת הבעיה: יהי R יחס דו-מקומי. **בעית החיפוש המתאימה** ל-R היא: "בהנתן מחרוזת x, מצא מחרוזת y המקיימת $(x,y)\in R$". בעיה בסיסית נוספת המתאימה ליחס הדו-מקומי R, היא זיהוי זוגות הנמצאים ביחס R. בעיה זו ראויה לשם **בעית אישוש פתרונות** (או **בעית זיהוי פתרונות**).

שתי מחלקות בעלות חשיבות מרכזית, הן:

(1) קבוצת היחסים הדו-מקומיים אשר **בעיות החיפוש** המתאימות להם ניתנות לפתרון בזמן פולינומיאלי. מחלקה זו מתאימה ל"משימות שניתן לפתרן ביעילות" (בעתיד, תזוהה מחלקה זו עם שפות המסומנות **P**).

(2) קבוצת היחסים הדו-מקומיים אשר **בעית הזיהוי** שלהם ניתנת לפתרון בזמן פולינומיאלי. מחלקה זו מתאימה ל"משימות שניתן לאשש את פתרונן ביעילות" (בעתיד תזוהה מחלקה זו עם מחלקת שפות המסומנת **NP**).

השאלה הבסיסית ביותר כיום, במדעי המחשב, היא שאלת הקשר בין שתי המחלקות הנ"ל. שאלה זו תהיה המוקד של המשך הקורס.

לשם פשטות נצמצם את הדיון שלנו ליחסים דו-מקומיים, בעלי פתרונות שארכם חסום ע"י פולינום באורך הקלט. יחסים שאינם מקיימים חסם כזה, ודאי שבעית החיפוש שלהן אינה ניתנת לפתרון בזמן פולינומיאלי (מכיוון שרק כתיבת הפתרון נמשכת זמן שאינו פולינומיאלי באורך הקלט).

<u>הגדרה 13.1:</u> יחס דו-מקומי, R, יקרא **חסום פולינומיאלית**, אם קיים פולינום, Q, כל שלכל $x,y\in\{0,1\}^*$, אם $(x,y)\in R$ אז $|y|\le Q(|x|)$.

הגיע הזמן להגדיר את שתי המחלקות שאוזכרו בתחילת הפרק.

<u>הגדרה 13.2:</u> **בעית החיפוש של יחס דו–מקומי, R, ניתנת לפתרון בזמן פולינומיאלי**, אם קיימת מכונת טיורינג פולינומיאלית, M, המקיימת, על כל קלט x בקבוצה $\{x:\exists y\ s.t.\ (x,y)\in R\}$, המכונה M עוצרת (במצב מקבל) עם פלט $M(x)$ המקיים $(x,M(x))\in R$ (שים לב כי ייתכן יותר מ-y אחד המקיים $(x,y)\in R$, אך מובטח כי המכונה M תמצא אחד כזה).

<u>הערה למתקדמים:</u> שים לב כי עבור קלט x אשר עבור כל y מתקיים $(x,y)\notin R$, אין כל דרישה לגבי פעולת M עליו, ובפרט היא אינה חייבת לדחות אותו. כלומר, M יכולה, בהנתן קלט x, לעצור במצב מקבל, לעצור במצב דוחה, להתקע, או לא לעצור כלל. ניתן לחזק את הגדרה 13.2, ע"י דרישה שהמכונה M תדחה קלטים שאין להם פתרון (כלומר, קלט x המקיים $\forall y\ (x,y)\notin R$). אנו לא נעשה זאת, מכיוון שבעתיד נצמצם ממילא את הדיון שלנו ליחסים

שבעית הזיהוי שלהם ניתנת לפתרון בזמן פולינומיאלי (ראה הגדרה 13.4) ואז שתי הגרסאות של הגדרה 13.2 תהיינה שקולות.

הגדרה 13.3: המחלקה $PolyFind$ (**PF**) היא קבוצת כל היחסים הדו-מקומיים, שבעית החיפוש שלהן ניתנת לפתרון בזמן פולינומיאלי (מחלקת יחסים זו, תקושר בעתיד אל מחלקת השפות **P**).

תרגיל 13.1: הוכח כי כל יחס $R \in PolyFind$, עבורו לכל x מתקיים $|\{y : (x,y) \in R\}| \leq 1$, הינו יחס חסום פולינומיאלית.

הגדרה 13.4: **בעית הזיהוי של יחס דו–מקומי, R, ניתנת לפתרון בזמן פולינומיאלי**, אם קיימת מכונת טיורינג פולינומיאלית, M, כך שהמכונה M מקבלת את $x\#y$ אם ורק אם $(x,y) \in R$ (שים לב כי המכונה M פולינומיאלית ב-$x\#y$ ולא ב-x בלבד).

הגדרה 13.5: המחלקה $PolyVerify$ (**PV**) היא מחלקת היחסים הדו-מקומיים החסומים פולינומיאלית, שבעית הזיהוי שלהן ניתנת לפתרון בזמן פולינומיאלי (מחלקה זו תקושר בעתיד עם מחלקת השפות **NP**).

תרגיל 13.2: יהי Q פולינום ותהי f פונקציה מלאה, כך שלכל x מתקיים $|f(x)| \leq Q(x)$.

נגדיר: $R_f \triangleq \{(x, f(x)) : x \in \{0,1\}^*\}$

הוכח:

(1) $f \in \mathbf{POLY}$ אם ורק אם $R_f \in PolyFind$.

(2) אם $R_f \in PolyFind$ אז $R_f \in PolyVerify$.

השאלה הבסיסית של מדעי המחשב (ניסוח לא מקובל אך טבעי יותר): "האם $PolyVerify \subseteq PolyFind$?" במילים: האם כל יחס דו-מקומי (חסום פולינומיאלית) שבעית הזיהוי שלו ניתנת לפתרון בזמן פולינומיאלי, גם בעית החיפוש שלו ניתנת לפתרון בזמן פולינומיאלי. כלומר:

האם כל בעיה שניתן לאשש את פתרונה ביעילות, ניתן גם לפותרה ביעילות.

התשובה, כפי שנראה בהמשך, אינה ידועה עדיין.

הערה: שים לב לחשיבות הצמצום של $PolyVerify$ ליחסים חסומים פולינומיאלית. יחס שאינו חסום פולינומיאלית עלול שלא להיות ב-$PolyFind$. בפרט, אפילו יחס טריביאלי כמו $\{(x, 1^{2^{|x|}}) : x \in \{0,1\}^*\}$ איננו ב-$PolyFind$ ("הפתרון" הינו ארוך מכדי שניתן יהיה לכתוב אותו בזמן פולינומיאלי באורך הקלט). אולם, הענין שלנו כמובן אינו

ביחסים כאלו, אלא ביחסים ש"הפתרון" שלהם הינו באורך סביר (כלומר, יחסים חסומים פולינומיאלית).

שים לב כי יש ערך מעשי מועט מאד לבעיות שלא ניתן לאשש את נכונות הפתרון שלהן באופן יעיל, מכיוון שאפילו אם ינתן לנו פתרון לבעיה כזו, לא נוכל "להרגיש" בכך.

דוגמאות לבעיות חיפוש שפתרונן ניתן לאישוש בזמן פולינומיאלי:

(1) <u>מציאת מסלול המילטוני</u> (מסלול המילטוני, הינו מסלול פשוט העובר דרך כל אחד מצמתי הגרף בדיוק פעם אחת). הבעיה הינה: בהנתן גרף, מצא בו מסלול המילטוני. היחס הדו-מקומי שזו בעית החיפוש שלו, הוא, $R_{HP} \triangleq \{(G,p) : G$ **הינו מסלול המילטוני בגרף** $p\}$. הראה כי $R_{HP} \in PolyVerify$. בהמשך נראה כי מאד סביר כי $R_{HP} \notin PolyFind$ (<u>למתקדמים:</u> נראה כי הבעיה R_{HP} הינה שלמה ב-$PolyVerify$, במובן שאם $R_{HP} \in PolyFind$, אז $PolyVerify \subseteq PolyFind$).

(2) <u>מציאת מסלול אוילרי</u> (מסלול אוילרי, הינו מסלול העובר דרך כל אחת מקשתות הגרף בדיוק פעם אחת). הבעיה הינה: בהנתן גרף, מצא בו מסלול אוילרי. היחס הדו-מקומי, שזו בעית החיפוש שלו, הוא, $R_{EP} \triangleq \{(G,p) : G$ **הינו מסלול אוילרי בגרף** $p\}$. הראה כי $R_{EP} \in PolyVerify$. זכור כי בקורס באלגוריתמים בתורת הגרפים הראו כי $R_{EP} \in PolyFind$ (כלומר הראו אלגוריתם פולינומיאלי למציאת מסלול אוילר בגרף).

<u>תרגיל 13.3:</u> הראה יחס דו-מקומי, אשר בעית החיפוש שלו הינה ב-$PolyFind$, אך בעית הזיהוי שלו אינה כריעה כלל (רמז: חשוב על יחס שהינו איחוד של שני יחסים. האחד יחס מלא ופשוט והשני יחס שזיהויו יתן פתרון לבעיה שאינה כריעה).

<u>תרגיל 13.4:</u> הראה כי לכל יחס דו-מקומי, R, המקיים $R \in PolyFind$ וגם $R \in PolyVerify$, קיימת מכונת טיורינג פולינומיאלית, M, המקיימת לכל x:

(1) אם קיים y כך ש- $(x,y) \in R$ אז M מקבלת את x ומחזירה פלט $M(x)$ המקיים, $(x,M(x)) \in R$.

(2) אם לכל y מתקיים $(x,y) \notin R$ אז המכונה M <u>דוחה</u> את x.

(רמז: זכור כי בהגדרת $PolyFind$, נדרש רק תנאי (1). לצורך קיום תנאי (2) יש להעזר בעובדה $R \in PolyVerify$).

<u>תרגיל 13.5</u> (הכללה של תרגיל 13.4): הראה כי לכל מכונת טיורינג פולינומיאלית, M, ולכל יחס דו-מקומי, R, במחלקה $PolyVerify$, קיימת מכונת טיורינג פולינומיאלית, M', המקיימת:

(1) לכל x עבורו $(x,M(x)) \in R$, המכונה M' מקבלת את x ומתקיים $M'(x) = M(x)$.

(2) אם המכונה M' מקבלת את x, מתקיים $(x,M'(x))\in R$.

הוכח את תרגיל 13.4, תוך שימוש מיידי בתרגיל זה.

מטרתינו בשארית פרק זה, היא להראות שהדיון בשאלה הבסיסית, של היחס בין מחלקות הבעיות $PolyFind$ ו-$PolyVerify$, ניתן למיצוי ע"י דיון בשאלת היחס בין מחלקות השפות הידועות בסימונים **P** ו-**NP**.

<u>הגדרה 13.6:</u> מחלקת השפות **P** (מסמל $Polynomial-Time$) הינה קבוצת השפות הניתנות להכרעה באמצעות מכונות טיורינג פולינומיאליות.

$$\mathbf{P} \triangleq \{L : L \text{ קיימת מכונת טיורינג פולינומיאלית, אשר מכריעה את השפה}\}$$

כלומר, $L\in\mathbf{P}$ פירושו כי קיימת מכונת טיורינג פולינומיאלית, M, העוצרת תמיד ומקבלת קלט x אם ורק אם $x\in L$ (אם $x\notin L$ אז M דוחה את x).

<u>תרגיל 13.6:</u> הראה כי הגדרת **P** הייתה זהה לו הוגדרה כקבוצת השפות "המתקבלות (לא מוכרעות) באמצעות מכונה פולינומיאלית". כלומר הראה כי בהנתן מכונה פולינומיאלית, M, המקבלת את L (כלומר, עוצרת ומקבלת כל $x\in L$) ניתן לבנות מכונת טיורינג פולינומיאלית המכריעה את L (כלומר עוצרת על כל קלט, מקבלת כל $x\in L$ ודוחה כל $x\notin L$).

<u>הגדרה 13.7:</u> מחלקת השפות **NP**, היא קבוצת השפות אשר מאופינות ע"י התחום של יחסים דו-מקומיים הנמצאים במחלקה $PolyVerify$. כלומר, $L\in\mathbf{NP}$ אם קיים יחס דו-מקומי, R_L, השייך למחלקה $PolyVerify$ ומתקיים: $L=\{x : \exists y\ (x,y)\in R_L\}$.

<u>הערה:</u> כפי שנראה בהמשך, זו אינה ההגדרה המקובלת ל-**NP** (שהיא פחות מסורבלת). מקור הסימון **NP** הוא בניסוח המקובל יותר. המחלקה **NP** מאופיינת כאן, כמחלקת שפות המכילות קלטים שיש להם פתרון שניתן לאישוש יעיל. זוית ראיה מעניינת נוספת, הנובעת מניסוח זה היא ששפה ב-**NP** היא קבוצת מילים אשר ניתן לודא ביעילות את שייכותן לשפה בעזרת "עדות" קצרה. כלומר, לכל $x\in L\in\mathbf{NP}$, ישנה "עדות" קצרה, y, כך שניתן לודא בזמן פולינומיאלי כי $(x,y)\in R_L$. אם נדייק יותר, לכל $L\in\mathbf{NP}$, קיים יחס דו-מקומי (חסום פולינומיאלית) R_L הניתן לזיהוי בזמן פולינומיאלי, כך שלכל $x\in L$ קיימת "עדות" קצרה, y, המקיימת $(x,y)\in R_L$ ואילו עבור $x\notin L$ אין עדות כזו. שים לב כי לא מובטח שקל למצוא את ה"עדויות" הללו, עבור $x\in L$, רק מובטח כי בהנתן x ומועמד ל"עדות" (y) עבור x, קל לודא האם אמנם מועמד זה מהווה "עדות" עבור x. טבעי לקרוא ל-y כזה, עדות לשייכות של x לשפה L.

<u>הערה:</u> המחלקה **NP** כנראה אינה סימטרית. כלומר, $L\in\mathbf{NP}$ לא בהכרח גורר כי $\bar{L}\in\mathbf{NP}$.

דוגמא: קבוצת הגרפים ההמילטונים, היא שפה ב-**NP**. ה"עדות" להיות גרף נתון המילטוני, הינה מסלול המילטוני בגרף (שים לב כי "קל" לודא כי מסלול הוא המילטוני, בניגוד למציאת מסלול כזה שהיא כנראה בעיה "קשה") אולם לא ידועות עדויות קצרות לכך שגרף אינו מכיל מסלול המילטוני.

תרגיל 13.7: הוכח כי $\mathbf{P} \subseteq \mathbf{NP}$ (רמז: עבור $L \in \mathbf{P}$ הגדר למשל, $R_L \triangleq \{(x,x) : x \in L\}$).

תרגיל 13.8: יהי R יחס דו-מקומי חסום פולינומיאלית, אשר ניתן לזיהוי בזמן פולינומיאלי. הראה כי קיימת מכונת טיורינג פולינומיאלית, M ופולינום, P, כך שהמכונה M מזהה את R ומספר הצעדים של M על כל זוג (x,y) הינו לכל היותר $P(|x|)$. (רמז: מכיוון ש-R חסום פולינומיאלית, קיים פולינום P_0, כך ש- $(x,y) \in R$ גורר $|y| \leq P_0(|x|)$. המכונה M תשתמש בפולינום P_0. אם ארכו של y הנתון הינו ארוך מדי (ביחס ל-x) המכונה תדחה את הזוג (x,y). אחרת, המכונה M תפעיל על הקלט את המכונה הפולינומיאלית לזיהוי R. שים לב כי הפולינום החוסם את זמן הריצה, הינו הרכבה של שני פולינומים).

תרגיל 13.9: יהי R יחס דו-מקומי, כך שקיימת מכונת טיורינג, M, המזהה את R, ופולינום, P, כך שזמן הריצה של M על כל זוג (x,y) חסום ע"י $P(|x|)$. הראה כי קיים יחס דו-מקומי חסום פולינומיאלית, R', כך ש-R' ניתן לזיהוי בזמן פולינומיאלי וכן מתקיים:

$$\{x : \exists y \ (x,y) \in R\} = \{x : \exists y (x,y) \in R'\}$$

הדרכה: המכונה M, בהכרח סורקת רק רישא באורך לכל היותר $P(|x|)$ של y, שתסומן y' ולכן היא מקבלת את (x,y) אמם היא מקבלת את (x,y').

תרגיל למתקדמים: בהתבסס על שני התרגילים הקודמים, הראה כי המחלקה **NP**, זהה למחלקת השפות המכילה שפה, L, אם קיים יחס דו-מקומי, R ומכונה, M, המזהה, האם $(x,y) \in R$, בזמן פולינומיאלי ב-$|x|$ ומתקיים, $L = \{x : \exists y \ (x,y) \in R\}$ (שים לב כי **NP** הוגדרה כמחלקת השפות, המאופיינת ע"י יחסים חסומים פולינומיאלית אשר ניתנים לזיהוי בזמן פולינומיאלי [באורך הזוג]).

לסיום ברצוננו להראות כי השאלה הבסיסית, "האם $PolyVerify$ הינה חלקית ל-$PolyFind$?", ניתנת למיצוי בשאלה, "האם $\mathbf{P} = \mathbf{NP}$?".

משפט 13.1: $PolyVerify \subseteq PolyFind$ אם ורק אם $\mathbf{NP} = \mathbf{P}$.

הערה: העובדה שהניסוח "$\mathbf{NP} \stackrel{?}{=} \mathbf{P}$" הינו מפורסם יותר, נובעת בעיקר מכך שהוא פחות מסורבל. השאלה

"$PolyFind \overset{?}{\subseteq} PolyVerify$" נראת לנו בסיסית יותר.

<u>הוכחת המשפט:</u> זכור כי $\mathbf{NP} \subseteq \mathbf{P}$.

<u>כיוון I:</u> נראה תחילה כי $\mathbf{P} \neq \mathbf{NP}$ גורר ש-$PolyVerify$ אינו חלקי ל-$PolyFind$. נניח כי קיימת שפה, $L \in \mathbf{NP}$ אשר אינה ב-$\mathbf{P}$. יהי R_L היחס הדו-מקומי המובטח מהיות $L \in \mathbf{NP}$. כמובן מתקיים $R_L \in PolyVerify$. נראה כעת כי לו היה מתקיים $R_L \in PolyFind$ היה נובע מכך קיום מכונה פולינומיאלית, M, לזיהוי השפה L. המכונה M, על קלט x, היתה מפעילה חיפוש עבור y המקיים $(x,y) \in R_L$ (באמצעות המכונה המובטחת מהיות $R_L \in PolyFind$) ומקבלת את x אם"ם אמנם נמצא y כך ש-$(x,y) \in L$. ודא כי הינך מבין את פעולת M וכי M אמנם מקבלת את L. קיבלנו סתירה להנחתנו כי $L \notin \mathbf{P}$. לפיכך נובע כי $R_L \notin PolyFind$ ולפיכך $PolyVerify$ (המכילה את R_L) אינה חלקית ל-$PolyFind$.

<u>כיוון II:</u> נניח כעת כי $\mathbf{P} = \mathbf{NP}$ ונראה כי $PolyVerify \subseteq PolyFind$. יהי R יחס דו-מקומי כלשהו במחלקה $PolyVerify$. נבנה את השפה הבאה:

$$L_R \triangleq \{(x,y') : \exists y''\ (x,y'y'') \in R\}$$

ודא כי $L_R \in \mathbf{NP}$ (רמז: הגדר R_{L_R} כך ש-$((x,y'),y'') \in R_{L_R}$ אם"ם $(x,y'y'') \in R$). מההנחה שלנו ($\mathbf{P} = \mathbf{NP}$) נובע $L_R \in \mathbf{P}$. תהא M מכונה פולינומיאלית המקבלת את L_R. נבנה מכונה M', אשר על קלט x, מוצאת y כך ש-$(x,y) \in R$ (כלומר, פותרת את בעית החיפוש של R). בהנתן קלט x, המכונה M' (בשלב מספר 0) בודקת, בעזרת המכונה M, האם $(x,\lambda) \in L_R$ (כאשר λ הוא הקלט הריק). אם לא, M' עוצרת ודוחה את x. אחרת ($(x,\lambda) \in L_R$) המכונה M', בשלב ה-i, כאשר יש לה רישא, y_{i-1}, באורך $i-1$, בודקת האם $(x,y_{i-1}0) \in L_R$. אם כן $y_i := y_{i-1}0$ והמכונה ממשיכה לשלב הבא (השלב ה-$i+1$). אחרת, המכונה M' בודקת האם $(x,y_{i-1}1) \in L_R$. אם כן $y_i := y_{i-1}1$ והמכונה ממשיכה לשלב הבא (השלב ה-$i+1$). אחרת, המכונה M' עוצרת ומחזירה את y_{i-1}. ברור כי $(x,y_{i-1}) \in R$ כי מהיות $(x,y_{i-1}) \in L_R$ נובע כי קיים y'' המקיים $(x,y_{i-1}y'') \in R$ וכן y'' אינו מתחיל ב-0 או ב-1 לכן ברור כי $y'' = \lambda$. שים לב לכך שמכיוון שהמכונה M פולינומיאלית, גם M' פולינומיאלית, מכיוון שמספר השלבים הוא פולינומיאלי (כי מתקיים $R \in PolyVerify$ ולכן R הינו חסום פולינומיאלית). לכן קיבלנו כי $R \in PolyFind$. □

<u>השאלה הבסיסית במדעי המחשב</u> (ניסוח מקובל): האם $\mathbf{P} = \mathbf{NP}$? כלומר האם כל שפה, שניתן להכריע חברות בה "באופן יעיל באמצעות עדויות קצרות", ניתן גם להכריע חברות בה ביעילות (ללא עזרת עדויות)?