הרצאה 13: סיבוכיות מקום

Based on previous iterations of this course, given by Nir Bitansky, Rotem Oshman, Iftach Haitner, and Omer Paneth.

מרצה: אורי שטמר

סיבוכיות מקום לא-דטרמיניסטית

הגדרה: תהי $S:\mathbb{N}\to\mathbb{N}$ ו- N מט"ד עם 3 סרטים: סרט קלט לקריאה בלבד, סרט עבודה, וסרט פלט לכתיבה חד פעמית. N רצה במקום $S(n)$ אם לכל $n\in\mathbb{N}$ ולכל קלט x באורך n, בכל ענף בעץ החישוב $T_{N,x}$, מתקיים ש- N משתמשת בלכל היותר $S(n)$ תאים בסרט העבודה בטרם עוצרת (בפרט תמיד עוצרת).

הגדרה: תהי $S:\mathbb{N}\to\mathbb{N}$.

$$NSpace(S(n)) \coloneqq \{L(M) : O(S(n)) \text{ מט"ד הרצה במקום } M\}$$

הגדרה:

$$NL = NSpace(\log(n))$$

מכילה בעיות מעניינות שאינן ידועות לפתרון ב- L (כפי שנראה בקרוב)

הגדרה: V מוודא במקום לוגריתמי עבור שפה A אם V מ"ט (דטרמיניסטית) 3-סרטית:

- סרט קלט לקריאה בלבד
- סרט עד **לקריאה חד פעמית**
- סרט עבודה

כך שלכל n וקלט x באורך n, מתקיים ש- V משתמש בלכל היותר $\boldsymbol{O(\log(n))}$ תאים בסרט העבודה ו- $x\in A$ אם"ם קיים עד w כך ש- $V(x,w)$ מקבל.

טענה: $A\in NL$ אם"ם קיים מוודא במקום לוגריתמי עבור A.

מאיפה מגיעה הדרישה שסרט העד הוא לקריאה חד פעמית? נניח של- A יש מוודא לוגריתמי ונרצה להראות שיש לה מט"ד במקום לוגריתמי. הדבר הטבעי לנסות פה הוא "לנחש" עד ולהריץ את V עם העד הזה. אבל העד הוא פולינומי ב |x| ולא נוכל "לנחש" אותו בבת אחת. אבל אם V קוראת את העד רק "במכה אחת" אז זה מאפשר לנו לנחש אותו "תו אחרי תו", כלומר כל פעם ננחש רק את התו הבא הנחוץ.
ומה לגבי הכיוון השני? נניח של- A יש מט"ד במקום לוגריתמי M. נוכל לבנות אלג' וידוא לוגריתמי אשר מקבל כעד את "סדרת הניחושים" לאורך החישוב של M(x). יספיק לנו לקרוא אותו "במכה אחת"...

תרגיל כיתה: הראו כי לכל $x\in A$ קיים עד באורך $poly(n)$.

פתרון: יהי w עד באורך קצר ביותר עבור x כך ש- $V(x,w)=1$, ונסמן $k=|w|$. נשים לב שלא יתכן שבמהלך הריצה של $V(x,w)$ יהיו שתי קונפ' שונות עם אותו תוכן בסרט העבודה, אותו מיקום של הראש הקורא בסרט הקלט, ואותו תו מתחת לראש הקורא מסרט העד. כלומר הראש הקורא של סרט העד הוא במיקום שונה בשתי הקונפ', אבל כל שאר הדברים זהים בין 2 הקונפ' (ראו ציור למטה). ולכן מס' הקונפ' ש $V(x,w)$ יכולה לעבור דרכן בריצה שלה חסום ע"י $n\times\Gamma^{s(n)}\times\Gamma=\text{poly}(n)$.

כעת נראה דוגמה לבעיה ב- NL שלא ידוע אם היא ב L:

הגדרה: $STCON = \{\langle G,s,t\rangle : t \text{ ל } s \text{ מסלול מ } G \text{ גרף מכוון וקיים}\}$
טענה: $STCON \in NL$

הוכחה: העד הוא רשימה סדורה של צמתים במסלול. ניתן לוודא במעבר חד-פעמי על המסלול תוך שימוש במקום לוגריתמי. (בכל שלב נחזיק בסרט העבודה את שני הקודקודים "הנוכחיים" ונבדוק בסרט הקלט האם יש קשת מהראשון לשני)

אנחנו לא יודעים האם $STCON \in L$. מאמינים שלא. איך נוכל "להשתכנע" באמונה כזאת?

בעיות NL-שלמות

הגדרה:
- שפה A_0 היא NL-קשה (תחת רדוקציות מקום לוגריתמי) אם לכל $A \in NL$ מתקיים $A \leq_L A_0$
- שפה A_0 היא NL-שלמה אם היא NL-קשה ובנוסף $A_0 \in NL$

טענה: STCON היא NL-שלמה

מסקנה: $NL \subseteq P$

הוכחת המסקנה: תהי $A \in NL$ אז $A \leq_L STCON$ ולכן גם $A \leq_P STCON$.
מכיוון ש- $STCON \in P$ נקבל שגם $A \in P$.

הוכחת הטענה (סקיצה):
תהי $A \in NL$ ותהי N מט"ד עם מקום לוגריתמי $S(n) = O(\log n)$ המכריעה אותה.
אנחנו רוצים להראות רדוקציה אשר בהינתן קלט x עבור A מחזירה גרף וזוג קוד' שיהוו קלט ל STCON.

תיאור הרדוקציה:
בהינתן קלט x נגדיר את גרף הקונפיגורציות $G_{N,x}$:
- הקודקודים הם הקונפיגורציות: $[n] \times \Gamma^S \times Q \times [S]$
  ניתנים ליצוג ע"י מחרוזת באורך $O(S(n)) = O(\log n)$
- קשת מכוונת מ- c ל- c' אם"ם יש מעבר בניהן (נקבע ע"י פונק' המעברים δ והקלט x)
- נניח בה"כ קונפ' מקבלת יחידה (אחרת אפשר להוסיף קוד' חדש לגרף ולחבר קשת אליו מכל הקונפ' המקבלות)

הרדוקציה בהינתן x מוציאה $\langle G_{N,x}\ ,\ s{=}c_0\ ,\ t{=}c_{acc}\rangle$ כאשר c_0, c_{acc} הם הקונפ' ההתחלתית/מקבלת.
ניתן לחשב במקום לוגריתמי.

אינטואיציה: נתחזק שני "מונים" אשר בכל אחד מהם נעבור על כל המחרוזות המייצגות קונפיגורציה.
בכל שלב נבדוק האם מאחת משתי הקונפ' האלה אפשר לעבור לשניה ואם כן אז נוסיף קשת מתאימה בפלט.

כאמור, לא יודעים האם L=NL. בעצם מצב הידע שלנו די עגום בהקשר הזה:

L NL P NP PSPACE

אנחנו יודעים ש L מופרד מ PSPACE אבל מעבר לזה אנחנו לא יודעים להוכיח הפרדות בין השפות הנ"ל.
מאמינים שהכל מופרד. לגבי NL,P,NP,PSPACE זה נראה ממש לא הגיוני שלא יהיו הפרדות. לגבי L,NL זה אולי פחות ברור. הנה משהו שאולי מערער על האמונה ש L,NL מופרדים:

משפט (סאביץ'): $STCON \in DSpace(\log^2 n)$

מסקנה: $NL \subseteq DSpace(\log^2 n)$ (דומה להוכחת טענת 3 מהשיעור הקודם)

הכללה (לא נוכיח): לכל $S(n) \geq \log(n)$ מתקיים $NSpace(S(n)) \subseteq DSpace(S^2(n))$.

כלומר תמיד אפשר לבצע סימולציה למטל"ד עם מקום S בעזרת מ"ט עם מקום S^2

מסקנה: $PSPACE = NPSPACE$ (ולכן לא טורחים לדבר על $NPSPACE$)

רעיון הוכחת המשפט:

אבחנה: אם קיים מסלול מ s ל t באורך $\geq T$ אזי קיים קוד' w כך שיש מסלול מ s ל t באורך $\geq \frac{T}{2}$ ויש מסלול מ w ל t באורך $\geq \frac{T}{2}$.

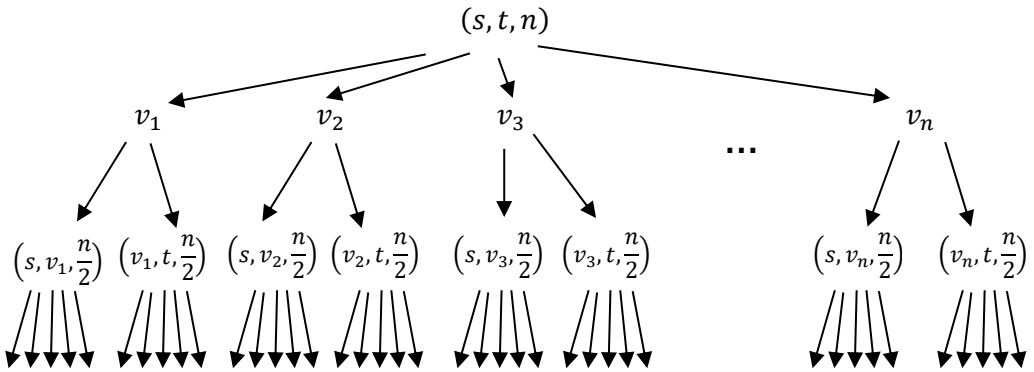
על סמך האבחנה הזאת נתכנן את האלג' הרקורסיבי הבא לבדיקה האם יש מסלול באורך $\geq T$ מ u ל v:

- עבור כל צומת w:
  - בדוק בצורה רקורסיבית אם יש מסלול באורך $\geq \frac{T}{2}$ מ u ל w
  - אם "כן" אז בדוק בצורה רקורסיבית (באותו זיכרון) אם יש מסלול מ w ל v באורך $\geq \frac{T}{2}$
  - אם שתי הבדיקות החזירו "כן" אז החזר "כן"
- החזר "לא"

בסיס: אם u=v או $(u,v) \in E$ אז החזר "כן". אחרת, אם $T \leq 1$, החזר "לא".

נכונות האלג' שתארנו ברורה. כמה מקום הוא צורך?

נחשוב על עץ הרקורסיה של הריצה:



- מס' רמות הרקורסיה הוא $\log n$
- בכל שלב ברקורסיה צריך לזכור:
  (1) צומת w - $O(\log n)$ ביטים
  (2) האם בקריאה הרקורסיבית הראשונה או השנייה - $O(1)$ ביטים
  (3) אורך מסלול T - $O(\log n)$ ביטים
- סה"כ בכל רמה $O(\log n)$
- סה"כ $O(\log^2 n)$.

הסבר נוסף: באלג' שלנו הקריאה הרקורסיבית השנייה קורת רק אם הראשונה החזירה "כן". לכן אם אני בקריאה השניה אני לא צריך לזכור כלום מהראשונה, כי אני יודע שהיא החזירה "כן".

באופן דומה, אם ברמה מסויימת אני כרגע בודק קודקוד v_k אז אני יודע שהבדיקות של $v_1, \dots v_{k-1}$ לא הצליחו (אחרת לא הייתי בודק בכלל את v_k). לכן אני לא צריך לזכור כלום מהקריאות שהיו עבור $v_1, \dots v_{k-1}$.

אז בכל מקרה, אני אף פעם לא צריך לזכור שום דבר מקריאות שהתרחשו במסלולים "שמאליים" / "קדומים" יותר.

שאלה: אנחנו יודעים שמחלקות דטרמניסטיות כמו L,P,PSPACE,EXP הן תמיד סגורות למשלים כי אנחנו יכולים פשוט להפוך את התוצאה. אבל במחלקות לא-דטר' זה פחות ברור, כמו למשל עם NP, coNP. האם NL סגורה למשלים?

הגדרה: $coNL = \{L : \overline{L} \in NL\}$

דיי ברור ש L נמצאת בחיתוך של NL ו- coNL. אבל היחס בין NL,coNL לא היה תמיד ברור. אנשים אפילו שיערו שקורה שם משהו אנלוגי ל NP,coNP. בשנת 1987 אימרמן הוכיח את התוצאה המפתיעה הבאה:

משפט (אימרמן): $\overline{STCON} \in NL$

מסקנה: $NL = coNL$
מדוע? לכל $A \in NL$ מתקיים $A \leq_L STCON$ ולכן גם $\overline{A} \leq_L \overline{STCON}$ (כי רדוקציות מיפוי אנחנו תמיד יכולים להפוך ככה) ולכן גם $\overline{A} \in NL$ ולכן גם $A \in coNL$. הכיוון השני מאוד דומה.

הכללה (לא נוכיח): לכל $S(n) \geq \log(n)$ מתקיים $NSpace(S(n)) = coNSpace(S(n))$.

הוכחת המשפט:
נתאר מוודא V במקום לוגריתמי עבור $\overline{STCON}$ (תזכורת: אנחנו מקבלים עד ואנחנו יכולים לקרוא אותו פעם אחת ומעבר לזה יש לנו מקום לוגריתמי)

סימון: עבור קלט $\langle G=(U,E), s, t\rangle$ נסמן $|U| = n$. לכל $0 \leq i \leq n$ נסמן ב- R_i את קבוצת הקודקודים שיש אליהם מסלול מ- s באורך i קשתות לכל היותר, ונסמן $r_i = |R_i|$.
מתקיים ש- $R_0 = \{s\}, r_0 = 1$.

בעזרת הסימונים האלה, המטרה שלנו היא להשתכנע ש $t \notin R_n$. זה אומר שאין מסלול מ- s ל- t בגרף.

העד לכך ש- $\langle G, s, t\rangle \notin STCON$, כלומר לכך ש- $t \notin R_n$, הוא קידוד:
$$\langle (r_1, w_1), (r_2, w_2), \ldots, (r_{n-1}, w_{n-1}), w_{t \notin R_n} \rangle$$
כאשר:
- w_i הוא עד לכך שאם r_{i-1} נכון אז גם r_i נכון.
- $w_{t \notin R_n}$ הוא עד לכך שאם r_{n-1} נכון אז $t \notin R_n$.

תיאור כללי של המוודא V:

V בהינתן קלט $\langle G, s, t\rangle$ ועד $\langle (r_1, w_1), (r_2, w_2), \ldots, (r_{n-1}, w_{n-1}), w_{t \notin R_n}\rangle$:

- כתוב $r_0 = 1$ על סרט העבודה
- לכל $1 \leq i \leq n-1$:
  - קרא את r_i מסרט העד וכתוב אותו לסרט העבודה
  - וודא את העד w_i עבור r_i בעזרת r_{i-1}. אם הווידוא נכשל אז תדחה.
  - מחק את r_{i-1} מסרט העבודה.
- וודא את העד $w_{t \notin R_n}$ בעזרת r_{n-1}. אם הווידוא נכשל אז תדחה.
- קבל.

למה המוודא הזה משתמש במקום לוגריתמי? בגל שלב צריך לשמור על סרט העבודה רק את האינדקס של הלולאה i ושני r-ים. אלו הם מספרים בני 1 ל n ולכן דורשים $logn$ ביטים. בנוסף כפי שנראה עכשיו, את תהליך הווידוא של כ"א מהעדים הנ"ל אפשר לבצע במקום לוגריתמי.

שימו לב שזאת צורה מאוד מוזרה לוודא שאין מסלול מ s ל t. אנחנו עושים את זה ככה כדי לחסוך במקום...

נותר לתאר את העדים w_i ואת $w_{t\notin R_n}$ ולהראות כיצד לוודא אותם במקום לוגריתמי.

ראשית נתאר את העד $w_{t\notin R_n}$ לכך ש- $t \notin R_n$ בהנחה ש- r_{n-1} נכון.

אינטואיטיבית: העד $w_{t\notin R_n}$ יספר לנו מיהם כל הקודקודים ב- R_{n-1} ואז נוכל להשתכנע ש $t \notin R_n$ כי נראה שאין קשת מאף אחד מהקודקודים הנ"ל ל- t. רק אנחנו צריכים לדאוג שנהיה מסוגלים לעשות את זה במעבר אחד על העד...

נקבע סדר על הקודקודים ונסמן $R_{n-1} = \{u_1, \dots, u_{r_{n-1}}\}$ כאשר $u_1 < \cdots < u_{r_{n-1}}$.
העד $w_{t\notin R_n}$ הוא קידוד:

$$w_{t\notin R_n} = \langle (u_1, w_{u_1\in R_{n-1}}),\ (u_2, w_{u_2\in R_{n-1}}), \dots,\ (u_{r_{n-1}}, w_{u_{r_{n-1}}\in R_{n-1}}) \rangle$$

כאשר $w_{u_i\in R_{n-1}}$ הוא קידוד של מסלול מ- s ל- t באורך לכל היותר $n-1$.

מוודא עבור העד $w_{t\notin R_n}$ לכך ש- $t \notin R_n$ בהנחה ש- r_{n-1} נכון:

בהינתן r_{n-1} והעד $w_{t\notin R_n} = \langle (u_1, w_{u_1\in R_{n-1}}), (u_2, w_{u_2\in R_{n-1}}), \dots, (u_{r_{n-1}}, w_{u_{r_{n-1}}\in R_{n-1}}) \rangle$:

- לכל $1 \le i \le r_{n-1}$:
  - קרא את u_i לסרט העבודה.
  - בדוק ש- $w_{u_i\in R_{n-1}}$ הוא מסלול מ- s ל- u_i באורך לכל היותר $n-1$. אחרת דחה.
  - אם $i > 1$ אז בדוק ש- $u_{i-1} < u_i$. אחרת דחה.
  - בדוק שאין קשת מ- u_i ל- t. אחרת דחה.
  - מחק את u_{i-1} מסרט העבודה.
- קבל.

שאלה: למה רצינו שהקודקודים u_i יופיעו בעד $w_{t\notin R_n}$ בסדר ממויין?
תשובה: אנחנו כרגע מוודאים את $w_{t\notin R_n}$ בהנחה שאנחנו יודעים את r_{n-1}. לצורך כך אנחנו רוצים לקבל מהעד $w_{t\notin R_n}$ רשימה של r_{n-1} קודקודים שונים ב- R_{n-1}. אבל מכיוון שאנחנו קוראים את העד רק פעם אחת, אנחנו רוצים שהקודקודים יופיעו בו בסדר ממויין כדי שנוכל להיות בטוחים שקודקוד לא מופיע יותר מפעם אחת בסדרה הזאת.

טענה א: אם r_{n-1} נכון אז קיים עד $w_{t\notin R_n}$ שמתקבל אם"ם $t \notin R_n$.

נותר לתאר את העד w_i לכך ש- r_i נכון בהנחה ש- r_{i-1} נכון.

אינטואיטיבית: העד w_i יספר לנו מיהם לכל קוד' בגרף אם הוא ב R_i או לא.

נסמן $U = \{u_1, \dots, u_n\}$.

העד w_i הוא קידוד:

$$w_i = \langle w_{i,1}\ ,\ w_{i,2}\ , \dots,\ w_{i,n} \rangle \quad , \quad \text{where} \quad w_{i,j} = \begin{cases} w_{u_j \in R_i} & , \quad u_j \in R_i \\ w_{u_j \notin R_i} & , \quad u_j \notin R_i \end{cases}$$

כאשר:

- $w_{u_j \in R_i}$ הוא קידוד של מסלול מ- s ל- u_j באורך לכל היותר i
- $w_{u_j \notin R_i}$ הוא עד לכך ש- $u_j \notin R_i$ בהינתן ש- r_{i-1} נכון, בדומה לעד $w_{t \notin R_n}$. כלומר $w_{u_j \notin R_i}$ יכיל רשימה ממויינת של r_{i-1} זוגות של קוד' ב- R_{i-1} פלוס מסלול באורך $\geq i-1$ אליו מ s כדי להוכיח שהוא אכן ב R_{i-1}.

מוודא עבור העד w_i לכך ש- r_i נכון בהנחה ש- r_{i-1} נכון:

בהינתן r_{i-1} והעד $w_i = \langle w_{i,1}, w_{i,2}, \dots, w_{i,n} \rangle$:

- אתחל משתנה $r = 0$ שסופר את מספר הקודקודים ב- R_i
- לכל $1 \leq j \leq n$:
  - וודא את העד $w_{i,j}$. אם הווידוא נכשל אז דחה.
  - אם $w_{i,j} = w_{u_j \in R_i}$ אז הגדל את המונה r ב- 1.
- אם $r = r_i$ אז קבל. אחרת דחה.

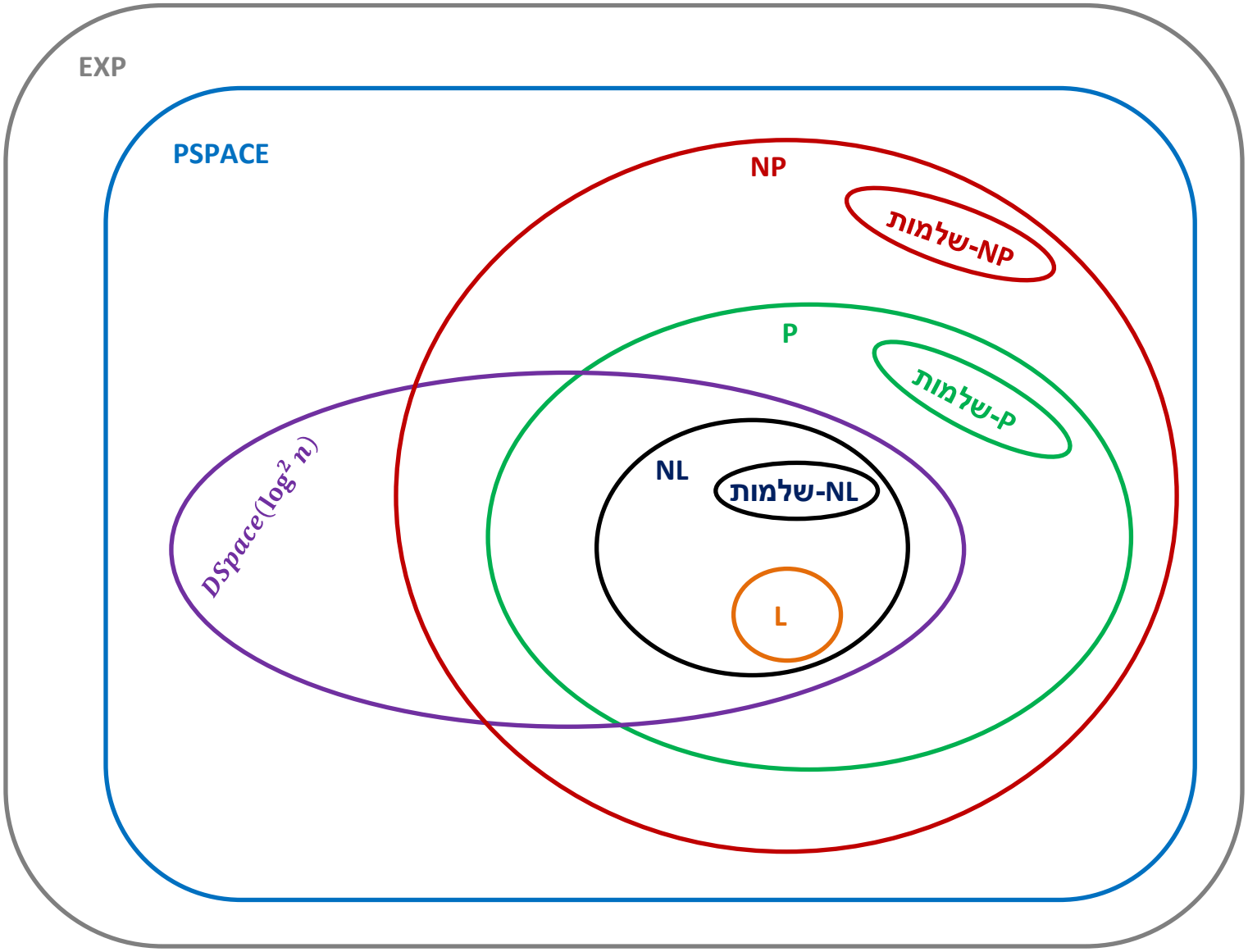
שאלה: אז בעצם אולי היינו יכולים לוותר על הכללת ה-r-ים כחלק מהעד, כי בעצם ה-w-ים מאפשרים לנו לחשב אותם בעצמנו אחד אחד?
תשובה: נכון. זה סתם נח לנו לכלול את ה-r-ים בשביל ההצגה של ההוכחה.

<u>טענה ב:</u> אם r_{i-1} נכון אז קיים עד w_i שמתקבל אם"ם r_i נכון.

מ.ש.ל.

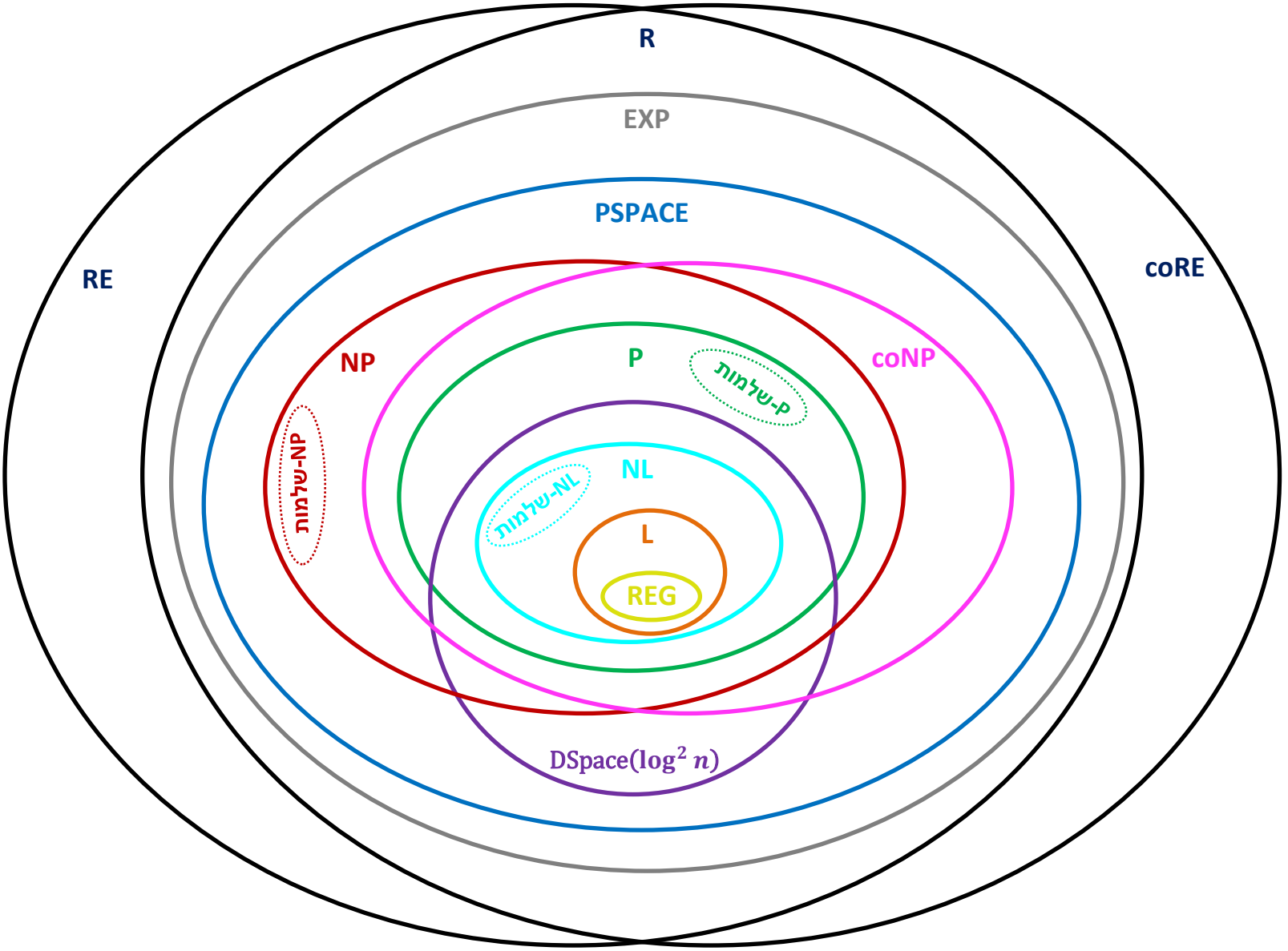
ההוכחה הזאת מראה שאם כותבים את העדים שלנו בצורה מאוד "חזרתית" (אותו מידע מופיע שוב ושוב לאורך העד) אז אפשר לוודא את העד הזה במעבר אחד ע"י זה ששומרים מספר קבוע של אינדקסים.

תמונת עולם מעודכנת:



- לא ידוע אם $L \neq NL$. מאמינים ששונה. בפרט מאמינים ש- $STCON \in NL \setminus L$.
- ממשפט היררכיית המקום אנחנו יודעים שיש שפה ב- $DSpace(\log^2 n) \setminus L$ ושיש שפה ב $\text{PSPACE} \setminus DSpace(\log^2 n)$.
- ממשפט היררכיית הזמן אנחנו יודעים שיש שפה ב $EXP \setminus P$.
- לגבי NP מול $DSpace(\log^2 n)$ מאמינים שאין הכלה באף כיוון:
  - תחת ההנחה ש- $NP \neq coNP$ אז NP לא סגורה למשלים אבל $DSpace(\log^2 n)$ כן סגורה ולכן $NP \neq DSpace(\log^2 n)$.
  - אם NP מוכל ב $DSpace(\log^2 n)$ אז ז"א שכל שפה ב NP אפשר לפתור בזמן $2^{O(\log^2 n)}$ שזה נראה לא סביר.
  - מצד שני, את השפה $\{\langle M, 1^n\rangle : M(1^n) \text{ halts using at most } \log^2|x| \text{ space}\}$ ניתן להכריע עם זיכרון $O(\log^2 n)$ אבל נראה שהיא לא ב NP כי לא ברור איך אפשר לייצר עדים קצרים עבורה. עוד דוגמה: שפת כל הגרפים שיש בהם <u>בדיוק</u> קליק <u>אחד</u> בגודל $\log n$. עם זיכרון $\log^2 n$ ניתן לעבור על כל האפשרויות לקליקים בגודל הזה, אבל נראה שהשפה לא ב NP כי לא ברור איך לשכנע שיש <u>בדיוק</u> קליק <u>אחד</u> כזה.
- גם לגבי P מול $DSpace(\log^2 n)$ מאמינים שאין הכלה באף כיוון:
  - למשל מאמינים ש $CVAL \in P \setminus DSpace(\log^2 n)$.
  - שתי הדוגמאות הנ"ל (לגבי $DSpace(\log^2 n)$ מול NP) הן בפרט דוגמאות לשפות ב $DSpace(\log^2 n)$ שכנראה לא ב P.
  - אם הייתה ידועה שפה ב $DSpace(\log^2 n)$ שאיננה ב P אז זה בפרט היה מפריד בין P לבין PSPACE.

איך R, RE, coRE, coNP, REG נכנסות לתמונה הזאת?



לגבי REG (אוסף השפות הרגולריות), מתקיים $REG = DSspace(O(1))$ ולכן $REG \subseteq L$. בנוסף מתקיים $REG \subsetneq L$ כי למשל השפה 0^n1^n היא ב L למרות שהיא לא רגולרית.

האינטואיציה לכך ש- $REG = DSspace(O(1))$ היא שמ"ט עם זיכרון קבוע לא יכולה לעשות יותר מאוטומט. ההוכחה הפורמלית יותר מסובכת כי מ"ט עם זיכרון קבוע יכולה לנוע על סרט הקלט גם ימינה וגם שמאלה (בניגוד לאוטומט), אבל אפשר להראות שזה לא מוסיף כוח כשהזיכרון הוא קבוע.