בס"ד

סיבוכיות מקום

סיבוכיות מקום באה למדוד את כמות הזיכרון הדרוש לחישוב, ולכן חשוב להבחין בין כמות הזיכרון הדרוש לחישוב לבין הזיכרון הדרוש לאחסון הקלט/הפלט. על מנת לבצע את אבחנה זו, המודל שבו נשתמש כולל מכונת טיורינג עם 3 סרטים:

1. סרט קלט – ~~כתיבה בלבד.~~
2. סרט פלט – כתיבה בלבד (חד כיוונית)
3. סרט עבודה – קריאה וכתיבה.

<u>הגדרה</u>: תהי M מכונת טיורינג דטרמיניסטית העוצרת לכל קלט. סיבוכיות המקום של M היא $s:N\rightarrow N$, כאשר $s(n)$ הוא האינדקס הגדול ביותר ש-M מגיעה אליו בסרט העבודה שלה לכל קלט מאורך n.

<u>הגדרה</u>: תהי $s:N\rightarrow N$ פונקציה. מחלקת הסיבוכיות $DSPACE(s(n))$ מכילה את כל השפות שניתן להכריע ע"י מ"ט דטרמיניסטית בסיבוכיות מקום $s(\cdot)$.

<u>הקשר בין סיבוכיות מקום וסיבוכיות זמן</u>

1. לכל פונקציה $t(\cdot)$ מתקיים: $DTIME(t(n))\subseteq DSPACE(t(n))$.
2. לכל פונקציה $s(\cdot)$ מתקיים: $DSPACE(s(n))\subseteq DTIME(n\cdot 2^{O(s(n))})$. בפרט, עבור $s(n)\geq \log n$ מתקיים: $DSAPCE(s(n))\subseteq DSPACE(2^{O(s(n))})$.

<u>חישוב לא דטרמיניסטי</u>

כאשר עסקנו בסיבוכיות זמן לא דטרמיניסטית הצגנו שני מודלים שקולים לאופן ביצוע החישוב:

1. מודל ה-online, כאשר המעברים של המכונה לא מוגדרים באופן דטרמיניסטי, אלא נקבעים ע"י בחירות לא דטרמיניסטיות בזמן ריצה. הקלט מתקבל אס"ם קיים עבורו מסלול חישוב מקבל של המכונה.
2. מודל ה-offline, במקרה זה המכונה היא דטרמיניסטית, אך מקבלת קלט עזר נוסף – עד. הקלט מתקבל אס"ם קיים עד שגורם למכונה לקבל.

בניגוד לסיבוכיות זמן בהקשר של סיבוכיות זיכרון מודלים אלה אינם שקולים. נחדד מעט את ההגדרות לפני שנראה את הקשר בינהן:

במודל ה-offline נגדיר את המכונה כבעלת סרט נוסף – סרט ניחוש - שהעד שלנו נקבע עליו באופן לא דטרמיניסטי בתחילת הריצה שלה. הסרט הנוסף הוא לקריאה בלבד, כך שלא ניתן לנצל אותו לביצוע החישוב בפועל. כמו כן, ניתן לנוע על הסרט הנוסף לשני הכיוונים – מדמה מצב של קלט שניתן לגשת שוב ושוב לחלקים שונים בו.

גם במודל ה-online ניתן להתייחס למכונה כמכונה דטרמיניסטית שמקבלת את הניחוש בסרט ניחוש מיוחד, אולם במצב זה סרט הניחוש יהיה ניתן לקריאה בלבד באופן חד כיווני.

ניתן להראות שהגדרה זו שקולה מבחינת סיבוכיות המקום והזמן למודל הלא דטרמינסטי הקלאסי. מצד אחד, המכונה הלא דטרמינסטית יכולה להקליט את בחירותיה הלא דטרמינסטיות, וכך ניתן לסמלץ אותה ע"י מכונה דטרמינסטית המקבלת אותן בסרט הניחוש. מצד שני, ניתן במקום לקבל את הניחוש באופן לא דטרמינסטי מראש על סרט מיוחד, לנחש אותו תוך כדי ריצה בכל זמן שהוא נדרש. נקודה עדינה בהקשר זה היא האפשרות של הישארות במקום בסרט הניחוש מה שיכול ליצור תלויות בין החלטות לא דטרמינסטיות שונות. בעניין זה ניתן לטפל ע"י הכפלת המצבים בגודל האלפבית של סרט הניחוש בהגדרת המכונה הלא-דטרמינסטית. כך ניתן יהיה להשתמש במצב ע"מ ל"זכור" ניחוש לא דטרמינסטי שביצענו במידה ולא הייתה תזוזה בסרט הניחוש.

כלומר, לעשה, ההבדל בין מודל ה-online וה-offline מתמצה בשאלה האם ניתן לקרוא מסרט הניחוש באופן חד-כיווני או דו כיווני.

<u>משפט</u>: לכל פונקציה $s:N\rightarrow N$, כך ש-logs היא space-constructible ולפחות לוגריתמית ב-n, מתקיים:
$NSPACE_{online}(s(n))=NSPACE_{offline}(\Theta(\log(s(n))))$

(הערה: עבור ההכלה של מודל ה-online במודל ה-offline די שיתקיים $s(n)\geq\log n$).

<u>הוכחה (סקיצה)</u>:

($\Rightarrow$) לשם הוכחת כיוון זה נראה כיצד לסמלץ מכונה לא-דטרמינסטית המשתמשת ב-s(n) זיכרון ע"י מכונה דטרמינסטית במודל ה-offline – כלומר, מכונה לא דטרמינסטית המקבלת ניחוש בסרט מיוחד ויכולה לקרוא אותו באופן דו-כיווני. רעיון ההוכחה הוא שהניחוש על הסרט המיוחד במכונה במודל ה-offline יכיל את ההרצה של המכונה הלא-דטרמינסטית, והמכונה הדטרמינסטית תוודא האם הוא אמנם חישוב מקבל או לא.

הרצה של מכונה היא רצף של קונפיגורציות שכל אחת מתארת אותה ברגע נתון.

מה כוללת קונפיגורציה זו?

1. תכולת סרט העבודה.
2. מיקום ראש הקריאה בסרט העבודה.
3. המצב בו נמצאת המכונה.
4. מיקום ראש סרט הקריאה.

מה אורכה של קונפיגורציה זו עבור קלט באורך n?

תכולת סרט העבודה דורשת O(s(n)) תאי זיכרון. המיקום של הראש בסרט זה – O(log(s(n)). המצב דורש מיקום בעל גודל קבוע – O(1). מיקום הראש בסרט הקריאה דורש O(log(n)). מכיוון שנתון ש-s(n) היא לפחות לוגריתמית הרי שניתן לתאר קונפיגורציה ע"י O(s(n)) תאי זיכרון.

מה נדרש לבדוק ע"מ לוודא כי ההרצה שקיבלנו בניחוש היא הרצה של מקבלת של המכונה הלא דטרמינסטית?

1. הקונפיגורציה הראשונה המופיעה בסרט הניחוש היא הקונפיגורציה ההתחלתית של המכונה הלא דטרמינסטית – המצב הוא המצב ההתחלתי והסרט העבודה ריק.
2. עבור זוג קונפיגורציות סמוכות c_i ו-c_{i+1} ניתן להגיע מ-c_i ל-c_{i+1} ע"י פונקציית המעברים של המכונה.
3. הקונפיגורציה הסופית מתארת מצב שבו המכונה הלא דטרמינסטית מקבלת.

את הבדיקה בראשונה והשלישית ניתן לבצע ע״י שימוש בלכל היותר $O(1)$ זיכרון. הבדיקה השניה דורשת מעבר הלוך ושוב על סרט הניחוש לבדוק שאמנם תכולת סרט העבודה לא השתנתה אלא בנקודה שמתאימה למיקום הראש בסרט העבודה בהתאם לפונקציית המעברים. כמו כן, נדרש לבדוק תאימות של תזוזות ראשי הקריאה ומצב המכונה. לשם ביצוע כלל העבודה הזו נדרש מקום לוגריתמי בגודל קונפיגורציה בודדת, כדי שניתן יהיה לבצע את המעברים החוזרים בין זוג קונפיגורציות סמוכות. לכן נדרש מקום שהוא $O(\log(s(n)))$.

כלומר, קיבלנו כי ניתן לסמלץ הרצה מלאה של מכונה לא דטרמינסטית ע״י בדיקה האם רצף קונפיגורציות הרצה הוא תקין ומסתיים במצב מקבל. כלל הסימולציה דורשת לכל היותר $O(\log(s(n)))$ זיכרון כנדרש.

($\Leftarrow$) בכיוון זה אנו נדרשים לסמלץ מכונת טיורינג שיש לה ניחוש לא דטרמינסטי המתקבל בתחילת ריצתה וניתן לנוע עליו הלוך ושוב (נסמנה ב-M_{off}) באמצעות מכונה שיכולה לסרוק את הניחוש על הסרט שלה באופן חד-כיווני (נסמנה ב-M_{on}). נקרא לקונפיגורציה של המכונה M_{off} ללא התייחסות לסרט הניחוש כ-CWG (configuration without guess). קונפיגורציה זו כוללת את ארבעת המרכיבים שציינו קודם: תכולת סרט העבודה, מיקום הראש בסרט העבודה ובסרט הקלט, ומצב המכונה.

נסמן את מס׳ ב-CWG האפשריים בריצת M על קלט x ע״י - #conf(M,x).

מה ערכו של מס׳ זה?

$|\Gamma|^{s(|x|)}s(|x|)|S_{Moff}||x|=2^{O(s(|x|)}$ (כאשר Γ הוא גודל האלפבית של סרט העבודה, ו-S_{Moff} הוא מס׳ המצבים שלה. נשים לב, ששני גדלים אלה קבועים ואינם תלויים באורך הקלט)

כמה פעמים לכל היותר המכונה M_{off} יכולה לבקר בתא מסויים בסרט הניחוש עבור ריצה מקבלת?

לכל היותר $\#conf(M,x)=2^{O(s)}$ פעמים. מדוע? אם המספר היה גדול יותר, הרי שאז M_{off} הייתה מבקרת בתא מסויים בסרט הניחוש עם אותה קונפיגורציה לפחות פעמיים. מצב זה יגרור בהכרח כניסה של M_{off} ללולאה אינסופית מה שלא יאפשר לה לסיים במצב מקבל.

מה החסם על אורכו של הניחוש בריצה מקבלת?

אם אורך הניחוש לא ארוך מדי, ניתן יהיה להעתיק אותו לסרט העבודה, וכך לבצע את הסימולציה בפשטות.

ניתן להראות כי אם קיים ניחוש y עבורו המכונה M_{off} מקבלת את x, המכונה תוכל לקבל את x גם בהינתן y' שאורכו לכל היותר $|\Gamma|^{2^{2^{O(s)}}}$.

נסמן את התאים של סרט הניחוש ב-$c_0,\dots,c_{|y|}$ ואת תכולתם ב-$y=g_0,\dots,g_{|y|}$. במהלך הריצה של M_{off} על הקלט x המכונה יכולה לבקר בתא c_i כמה פעמים, אך לא יותר מ-$\#conf(M_{off},x)$ ביקורים בתא זה. נכנה את סדרת ה-CWG-ים המציינים את מצב המכונה בזמן הביקור ב-c_i כ״רצף הביקורים״ בתא c_i.

נשים לב, שאם לשני תאים c_k ו-c_l המכילים אותו תו ניחוש יש בדיוק אותו רצף ביקורים, אז ניתן למחוק את כלל התווים ביניהם ממחרוזת הניחוש כולל אחד מהם.

מדוע? בכל פעם שמתבצעת תזוזה מ-c_k ל-c_{k+1} עם קונפיגורציה מסויימת, לאחר מכן תתבצע תזוזה של c_l ל-c_{l+1} עם אותה קונפיגורציה (או בסדר הפוך) – לכן ניתן למעשה לדלג על כל שלבי החישוב שמתבצעים ביניהן. באותו אופן ניתן להמשיך ולדלג על כל שלבי החישוב בין שני תאים אלה גם עבור הביקורים הבאים בהם, כיוון

שיש להם בדיוק אותה סדרת ביקורים. מכאן שלמעשה אין כלל צורך בתכולת התאים שבין שני מקומים אלה, שכן תמיד ניתן לדלג על הביקורים בהם בהתאם למה שתארנו.

בשל האבחנה כי ניתן ליצור ניחוש בו לכל תא סדרת ביקורים שונה, הרי שהחסם על אורך ניחוש שכזה הוא כמות סדרות הביקורים השונות. מכיוון שהראנו קודם שסדרת ביקורים אורכה לכל היותר $\#conf(M_{off},x)$, הרי שהחסם שנקבל הוא:

$$\sum_{i=1}^{\#conf(M_{off},x)} \#conf(M_{off},x)^i \leq \#conf(M_{off},x) \cdot \#conf(M_{off},x)^{\cdot \#conf(Moff,x)} = 2^{2^{O(s(|x|))}}$$

אם נכפיל זאת בגודל האלפבית של סרט הניחוש נקבל את החסם המבוקש על גודל 'y – הניחוש המקוצר למסלול חישוב מקבל.

בסיכומו של דבר, ראינו כיצד נוכל לבצע סימולציה של M_{off} ע״י M_{on}, אך סימולציה זו תדרוש $2^{2^{O(s)}}$ מקום ואנו מעוניינים בביצועה באופן חסכוני יותר.

אשר על כן, נייצג ריצה של המכונה M_{off} באופן מתוחכם יותר ע״י הרצף של רצפים הבאים:

לכל תא c_i נגדיר את *רצף הביקורים המכוון* שלו המכיל את הנתונים הבאים:

- תכולת התא c_i
- רצף הביקורים בתא c_i
- לכל קונפיגורציה ברצף מהסעיף הקודם – את הכיוון ממנה הגיע ראש הקריאה בסרט הניחוש.

את הרצף של רצפי הביקורים המכוונים המכונה M_{on} תקבל על סרט הניחוש שלה (לפי סדר התאים בסרט הניחוש של M_{off}) וכל שעליה לבצע הוא לסמלץ את ריצת המכונה M_{off} ע״י בדיקת תקינות הרצף שבסרט הניחוש. הבדיקה תבצע את הצעדים הבאים:

1. בדיקה שהרצף המכוון הראשון מתחיל עם הקונפיגורציה התחלתית של M_{off}.
2. בדיקה שרצף מסויים הוא תקין מבחינת הכיוונים של הגעת ראש הקריאה אליו – אם הגענו לביקור משמאל בפעם הבאה ברצף נגיע לביקור מימין וכו'.
3. בדיקה שכל זוג עוקב של רצפי ביקורים מכוונים הוא קונסיסטנטי. כלומר, יש התאמה בין כל זוג קונפיגורציות ברצף, כשהשינויים מתבצעים רק בהתאם לפונקציית המעברים של M_{off}.
4. בדיקה שיש רצף שבו הקונפיגורציה האחרונה מתארת קונפיגורציה מקבלת.
5. בדיקה שברצף האחרון אין קונפיגורציה שמשתמע ממנה שהגענו לתא האחרון בסרט הניחוש מימין.

כדי לבצע את כלל הבדיקות המכונה M_{on} לא נדרשת להעתיק לסרט העבודה יותר משני רצפי ביקורים מכוונים ועוד מקום שהוא $O(1)$. כבר הראנו שרצף ביקורים בתא מסוים יכול להכיל לכל היותר $2^{O(s(n))}$ קונפיגורציות עבור קלט באורך n, לכן ניתן להסתפק בכמות שהיא סדר גודל של מס' זה ע״מ לבצע סימולציה מלאה של M_{off} באמצעות המכונה M_{on}.