הרצאה 9

תכנון דינאמי

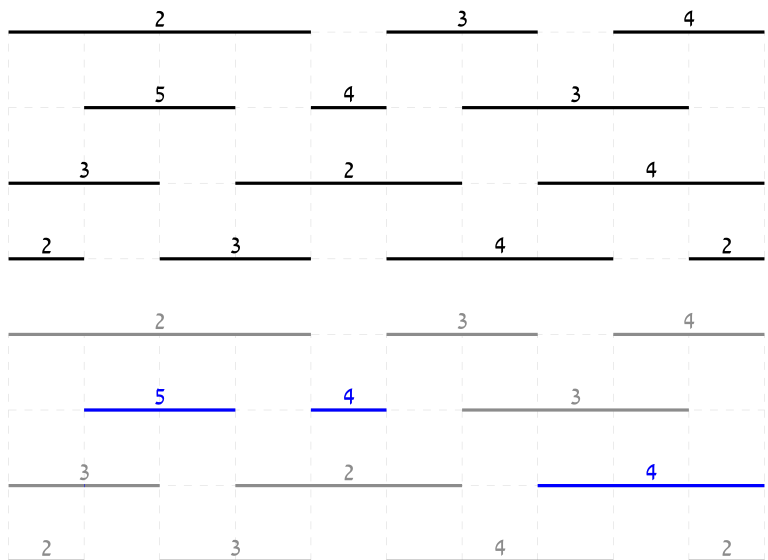
שיבוץ אינטרוולים, מסלולים קלים ביותר

קבוצה בלתי תלויה של אינטרוולים עם משקל מקסימלי

נתונים n אינטרוולים (נניח שכבר אחרי מיון לפי זמן סיום) $A=(a_1,\ldots,a_n)$ לכל אינטרוול זמן התחלה $s(a_i)$, זמן סיום $e(a_i)>s(a_i)$ ומשקל $w(a_i)$. תת קבוצה $I\subseteq A$ של אינטרוולים נקראת בלתי תלויה אם לכל $a_i,a_j\in I$ אחד מהשניים מתקיים:

1. $s(a_j)>e(a_i)$
2. $s(a_i)>e(a_j)$

רוצים למצוא קבוצה בלתי תלויה של אינטרוולים עם משקל מקסימלי.
דוגמה: קלט לבעיה וקבוצה בלתי תלויה במשקל 13.



נסמן $A_i=(a_1,\ldots,a_i)$. נגדיר

$$p(i)=\max\begin{cases}\max\{j:e(a_j)<s(a_i)\}\\0\end{cases}$$

כלומר $j=p(i)$ הוא האינדקס המקסימלי כך ש-a_j מסתיים לפני ש-a_i מתחיל או 0 אם לא קיים כזה.
נגדיר את $\alpha(i)$ להיות משקל פתרון אופטימלי עבור A_i אז $\alpha(n)$ הוא הערך אותו אנחנו מחפשים.

טענה 1.

$$\alpha(i)=\max\begin{cases}w(i)+\alpha(p(i))\\0+\alpha(i-1)\end{cases}$$

כמו כן מתקיים ש:

$$\alpha(0)=0$$

הוכחה. באינדוקציה על i.
בסיס: עבור $i=0$ טריוויאלי.
עבור i כלשהו נקבע פתרון אופטימלי OPT ונסמן $OPT_i=OPT\cap A_i$.
אם $a_{i+1}\in OPT$ אז OPT לא יכול להכיל אף אינטרוול $a_{p(i)+1},\ldots,a_{i+1}$ לפי הנחת האינדוקציה

$$\alpha(p(i)+1)\geq w(OPT_{p(i)})$$

ולכן הטענה מתקיימת כי

$$\alpha(i)\geq\alpha(p(i))+w(a_i)\geq w(OPT_{p(i)})+w(a_i)$$

מצד שני, אם $a_{i+1}\notin OPT$ אז לפי ההנחה

$$\alpha(i-1)\geq OPT_{i-1}$$

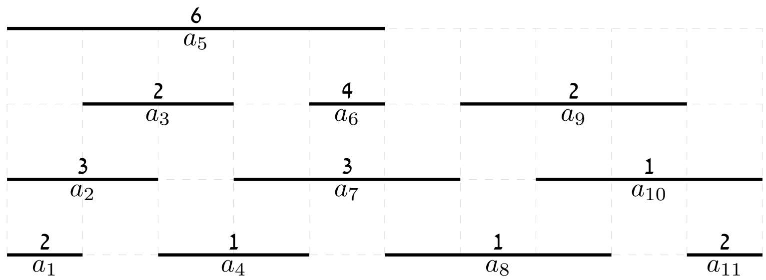
והטענה מתקיימת. $\square$

חישוב יעיל של O

כיצד נחשב את O ביעילות ? נשים לב שאם מחשבים את ערכי O מ-1 עד n ושומרים את הערכים (למשל במערך) אז חישוב של כל ערך לוקח $O(1)$ זמן.
זמן הריצה של האלגוריתם:

1. מיון - $O(n \log n)$
2. חישוב p - $O(n \log n)$ (חיפוש בינארי לכל i)
3. חישוב O - $O(n)$

סך הכל $O(n \log n)$
דוגמת הרצה:



נחשב:

i	0	1	2	3	4	5	6	7	8	9	10	11
p	0	0	0	1	2	0	4	3	6	7	7	9
α	0	2	3	4	4	6	8	8	9	10	10	12

נמצא את הקבוצה עצמה:

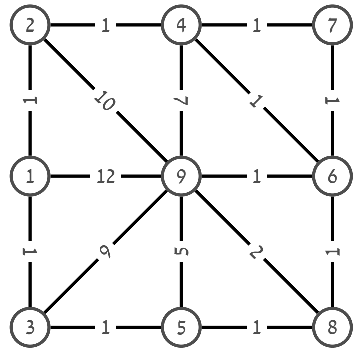
i	0	1	2	3	4	5	6	7	8	9	10	11
p	0	0	0	1	2	0	4	3	6	7	7	9
α	0	2	3	4	4	6	8	8	9	10	10	12

נקודות חשובות:

- מה יקרה אם במקום לחשב את ערכי α בסדר עולה ושמירת הערכים במערך נחשב את ערכי α באופן רקורסיבי על המחסנית ?
- מה יקרה אם לכל תא במערך נזכור גם את הקבוצה שמתאימה לערך התא ?

מסלולים קלים ביותר בין כל הזוגות

בהינתן גרף (מכוון או לא) עם n צמתים נרצה להדפיס טבלה בגודל $n \times n$ שבכניסה ה-ij שלה נמצא ערך מסלול קל ביותר מצומת i לצומת j.
ניתן, כמובן, לעשות זאת על ידי n הרצות של אלגוריתם בלמן פורד או דייקסטרה ולמצוא את התשובה בסיבוכיות זמן של $O(n^2m)$ ו- $O(nm \log n)$ בהתאמה. נראה שאפשר גם יותר טוב.
בהינתן גרף שצמתיו ממוספרים מ-1 עד n נגדיר את d_{ij}^k להיות משקל מסלול קל ביותר מצומת i לצומת j שיכול לעבור רק בצמתי ביניים עם אינדקסים ב-$[k]$.
דוגמה:



למה שווים הערכים הבאים d_{19}^0, d_{19}^1, d_{19}^2, d_{19}^6, d_{19}^7, d_{19}^9?

אבחנה 1. משקל מסלול קל ביותר מצומת i לצומת j שווה ל-d_{ij}^n.

אבחנה 2. $d_{ij}^0 = w(ij)$

נניח עכשיו שלכל i, j ו-k אנחנו מקבעים מסלול קל ביותר עבור d_{ij}^k.

אבחנה 3. אם הצומת k לא שייך למסלול שמתאים ל-d_{ij}^k אז $d_{ij}^k = d_{ij}^{k-1}$

אבחנה 4. אם הצומת k שייך למסלול שמתאים ל-d_{ij}^k אז $d_{ij}^k = d_{ik}^{k-1} + d_{kj}^{k-1}$.

מסקנה 1.

$$d_{ij}^k = \begin{cases} w(ij) & k = 0 \\ \min\{d_{ij}^{k-1}, d_{ik}^{k-1} + d_{kj}^{k-1}\} & k > 0 \end{cases}$$

חישוב: נגדיר $n+1$ מטריצות בגודל $n \times n$, $A^0, \ldots, A^n$, כאשר $A_{ij}^k = d_{ij}^k$. נמלא את ערכי המטריצות מ-A^0 ועד A^n כאשר נשים לב כי בשביל למלא את מטריצה A^k אנו צריכים לדעת אך ורק את ערכי המטריצה A^{k-1}.
סיבוכיות: אנחנו מחשבים n^3 ערכים וחישוב של ערך בודד לוקח $O(1)$ פעולות.

מסלולים קלים ביותר

כעת נפתור את בעיית המסלול הקל ביותר.
תזכורת: בהינתן גרף (מכוון או לא) $G = (E, V)$, פונקציית משקל $w : E \to \mathcal{R}$, צומת מקור $s \in V$, וצומת יעד $t \in V$ רוצים למצוא מסלול מ-s ל-t במשקל מינימלי.

ניסיון ראשון

נגדיר את $a(v)$ להיות המסלול הקל ביותר מ-s ל-v, אז מתקיים ש:

$$a(v) = \min_{uv \in E} a(u) + w(uv))$$

מה הבעיה ?

ניסיון שני

נגדיר את $a(v, U)$ להיות המסלול הקל ביותר מ-s ל-v בגרף $G[U]$, אז מתקיים ש:

$$a(v, U) = \min_{uv \in E} a(u, U \setminus \{v\}) + w(uv)$$

מה הבעיה ?

פתרון

נגדיר את $a(v, k)$ להיות מסלול קל ביותר מ-s ל-v עם k קשתות לכל היותר, ונחשב:

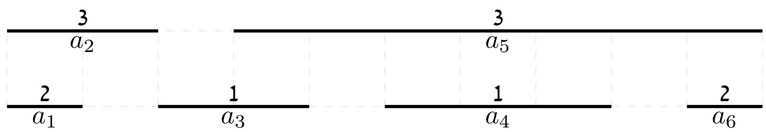
$$\begin{aligned} &\forall\, v \neq s, 1 \leq k \leq n-1 && a(v,k) = \min_{uv \in E} a(u, k-1) + w(uv) \\ &\forall\, u \neq s && a(u, 0) = \infty \\ & && a(s, 0) = 0 \end{aligned}$$

טענה 2. אם ב-G אין מעגלים שליליים אז לכל v, $a(v,n-1)$ הוא משקל המסלול הקל ביותר מ-s ל-v.

הוכחת נכונות: כתרגיל.

גרף החישוב

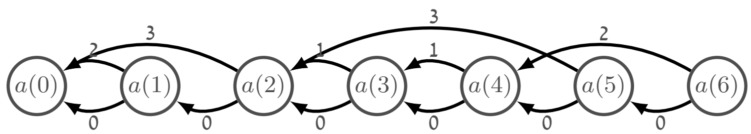
בהינתן נוסחת נסיגה, f, נסתכל על גרף החישוב שלה, G_f זהו גרף מכוון שבו כל צומת מתאימה למצב (ערך פרמטרים מסוים לנוסחה) וקיימת קשת ממצב s_i למצב s_j אמ"מ לצורך חישוב מצב s_i יש צורך לחשב את מצב s_j.
למשל עבור הקלט הבא לבעיית האינטרוולים:



ונוסחת הנסיגה

$$\alpha(i) = \max \begin{cases} w(i) + \alpha(p(i)) \\ 0 + \alpha(i-1) \end{cases}$$

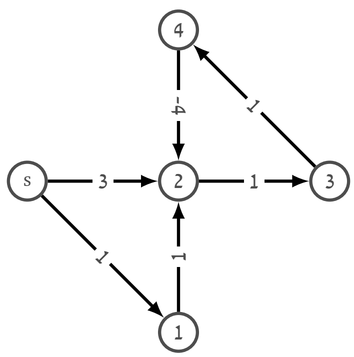
גרף החישוב יראה כך (ניתן אף לשים משקלים מתאימים על הקשתות):

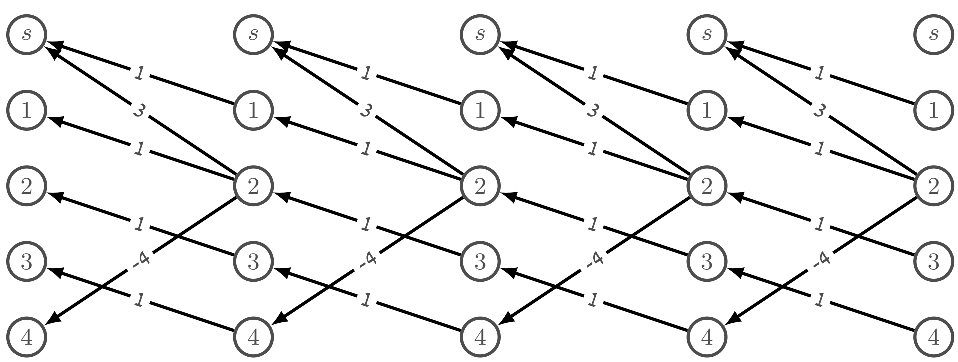


מה נדרוש מגרף החישוב ?

1. חסר מעגלים
2. לא גדול מדי
3. ניתן לחשב את הערכים של הבורות

דוגמה נוספת, כיצד יראה גרף החישוב עבור נוסחת הנסיגה של מסלולים קלים ביותר והקלט הבא:





1 מימוש

נסתכל על שלוש פונקציות שמחשבות את מספר פיבונצ'י ה-i. מה הסיבוכיות של כל פונקציה?

```
function fib(i: number): number {
    if (i <= 1) return 1
    return fib(i - 1) + fib(i - 2)
}

function dp(i: number): number {
    const fib = [1, 1]
    for (let j = 2; j <= i; j++)
        fib[j] = fib[j - 1] + fib[j - 2]
    return fib[i]
}

const cache = [1, 1]
function dp2(i: number): number {
    if (cache[i]) return cache[i]
    const n = dp2(i - 1) + dp2(i - 2)
    cache[i] = n
    return n
}

console.time('no dp')
console.log(fib(40))
console.timeEnd('no dp')
// no dp: 1501.243ms

console.time('dp')
console.log(dp(40))
console.timeEnd('dp')
// dp: 0.129ms

console.time('dp2')
console.log(dp2(40))
console.timeEnd('dp2')
// dp2: 0.112ms

```