1 אלגוריתמי קירוב

1.1 בעיית חלוקת משימות בין מכונות ($Load\ Balancing$)

1.1.1 הצגת הבעיה:

בעיית חלוקת משימות בין מכונות

הקלט: k - מספר מכונות זהות. ו-n מספרים חיוביים $t_1, ..., t_n$ המייצגים זמני ריצה של המשימות.

הפלט: חלוקה מאוזנת כמה שאפשר של המשימות בין המכונות. כלומר חלוקה שעבורה זמן העבודה של המכונה העמוסה ביותר, הוא מינימלי.

1.1.2 דוגמא:

$k = 2$, $n = 4$. עם הזמנים $t_1 = 1, t_2 = \frac{1}{2}, t_3 = \frac{1}{2}, t_4 = 2$. אז החלוקה האופטימלית היא:

$1: t_4 \qquad sum = 2$

$2: t_1, t_2, t_3 \qquad sum = 2$

1.1.3 אלגוריתם קירוב חמדני:

אלגוריתם 1 קירוב חמדני לבעיית חלוקת משימות בין מכונות

- נעבור על המשימות לפי סדר הגעתן, בכל שלב נשלח את המשימה המגיעה למכונה הכי פחות עמוסה ברגע זה.

בדוגמא הקודמת זה ייתן את החלוקה:

$1: t_1, t_4 \qquad sum = 3$

$2: t_2, t_3 \qquad sum = 1$

ולכן זה רק קירוב.

1.1.4 הוכחת קירוב:

סימונים ופורמליסטיקה מרחב הפתרונות החוקיים לבעיה הוא:

$$\mathcal{S} = \{s : [n] \to [k]\}$$

לכל $1 \le j \le k$ נגדיר $T_j : \mathcal{S} \to \mathbb{R}_+$ זמן הריצה של המכונה ה-j בפתרון s באופן הבא:

$$T_j(s) = \sum_{i=1, s(i)=j}^{n} t_i$$

בנוסף נגדיר:

$$q(s) = \max_{1 \le j \le k} \{T_j(s)\}$$

המטרה שלנו היא למצוא מינימום של q על $\mathcal{S}$.

משפט האלגוריתם החמדן שהצגנו משיג $\left(2 - \frac{1}{k}\right)$ קירוב לפתרון האופטימלי. כלומר, יהי S^* הפתרון האופטימלי ו-G הפתרון החמדן. אז מתקיים:

$$\frac{q(G)}{q(S^*)} = 2 - \frac{1}{k}$$

למה 1 יהי t_{max} זמן הריצה של המשימה הארוכה ביותר אזי: $q(S^*) \ge t_{max}$

הוכחת למה 1 המשימה הארוכה ביותר נשלחת בפתרון S^* למכונה כלשהי, ומכונה זו עובדת לפחות t_{max} זמן.

למה 2 $q(S^*) \ge \frac{1}{k} \sum_{i=1}^{n} t_i$.

הוכחת למה 2

$$q(S^*) = \max_{1 \le j \le k} \{T_j(S^*)\} \ge \frac{1}{k} \sum_{j=1}^{k} T_j(S^*) = \frac{1}{k} \sum_{j=1}^{k} \sum_{i=1, s(i)=j}^{n} t_i = \frac{1}{k} \sum_{i=1}^{n} t_i$$

הוכחת המשפט יהי G הפתרון החמדן. יהי $1 \le j_0 \le k$ האינדקס של המכונה העמוסה ביותר בפתרון G. לכן $q(G) = T_{j_0}(G)$. יהי $1 \le l \le n$ האינדקס של המשימה האחרונה לפי סדר ההגעה שנשלחה למכונה j_0. עבור $1 \le j \le k$ נסמן ב-$F_j(G)$ את זמן העבודה הכולל של המכונה j על המשימות שהוקצו לה מתוך $l - 1$ המשימות הראשונות. כלומר:

$$F_j(G) = \sum_{i=1, s(i)=j}^{l-1} t_i$$

לפי דרך פעולתו של האלגוריתם החמדן, $F_{j_0}(0)=\min\limits_{1\le j\le k}\{F_j(G)\}$. לכן מתקיים

$$\begin{aligned}q(G)=T_{j_0}(G)&=t_l+F_{j_0}(G)=t_l+\min_{1\le j\le k}\{F_j(G)\}\\&\le t_l+\frac{1}{k}\sum_{j=1}^{k}F_j(G)=t_l+\frac{1}{k}\sum_{j=1}^{k}\sum_{i=1,s(i)=j}^{l-1}t_i=t_l+\frac{1}{k}\sum_{i=1}^{l-1}t_i=(1-\frac{1}{k})t_l+\frac{1}{k}\sum_{i=1}^{l}t_i\\&\le(1-\frac{1}{k})t_l+\frac{1}{k}\sum_{i=1}^{n}t_i\le(1-\frac{1}{k})t_{max}+\frac{1}{k}\sum_{i=1}^{n}t_i\le(1-\frac{1}{k})q(S^*)+q(S^*)=(2-\frac{1}{k})q(S^*)\end{aligned}$$

1.2 בעיית כיסוי ע"י קבוצות $(Set\ Cover)$

1.2.1 הצגת הבעיה:

בעיית כיסוי ע"י קבוצות

סיפור מסגרת: רוצים ללמד רשימת נושאים במדעי המחשב, ויש קורסים שכל אחד מכסה חלק מהנושאים, ורוצים ללמוד כמה שפחות קורסים ולכסות את כל הנושאים.

הקלט: מספר טבעי n, ואוסף של r תת-קבוצות $A_1,A_2,...,A_r$ של $[n]$ כך ש: $\bigcup\limits_{i=1}^{r}A_i=[n]$

הפלט: תת-קבוצה $S\subseteq[r]$ כך ש$\bigcup\limits_{i\in S}A_i=[n]$ וכך ש-$|S|$ מינימלי.

זו בעיה NP קשה, אז נחפש פתרון מקורב.

1.2.2 דוגמא:

$n=10$, $r=5$. $A_1=\{1,3,5,7,9\}$, $A_2=\{2,4,6,8,10\}$, $A_3=\{1,3,5,6,8,10\}$, $A_4=\{1,7,9\}$, $A_5=\{2,4\}$.

אז המשפחה הכי קטנה של קבוצות שתכסה את כל n היא 2 (כלומר $|S^*|=2$). לדוגמא $S^*=\{1,2\}$.

1.2.3 אלגוריתם קירוב חמדני:

אלגוריתם 2 קירוב חמדני לבעיית כיסוי ע"י קבוצות

1. אתחול: $G=\emptyset$, $X=[n]$. (G - הפתרון החמדן. X - מה שנותר לכסות)
2. איטרציה: יהי $1\le i^*\le r$ האינדקס המקיים $|X\cap A_{i^*}|=\max\limits_{1\le i\le r}|X\cap A_i|$. נעדכן $G:=G\cup\{i^*\}$, $X:=X\setminus A_{i^*}$.
3. סיום: כאשר $X=\emptyset$ נעצור ונחזיר את G.

אם האלגוריתם היה מופעל על הדוגמא הקודמת היינו מקבלים את הפתרון $G=\{3,1,5\}$. (שהוא כמובן לא אופטמלי).

1.2.4 הוכחת קירוב:

משפט יהי S^* פתרון אופטימלי לבעיה, ויהי G הפתרון החמדן. אז:

$$(1 \leq) \frac{|G|}{|S^*|} \leq \lceil ln(n) \rceil$$

במילים אחרות, הפתרון החמדן הוא $\lceil ln(n) \rceil$ קירוב לפתרון האופטימלי.

אינטואיציה להוכחת המשפט נראה כי אם S^* אז מספר האיטרציות של האלגוריתם עד העצירה קטן יחסית (שזה גודל הפתרון החמדן כי בכל איטרציה האלגוריתם מוסיף בדיוק איבר אחד). נראה זאת על ידי כך שנראה כי $|X|$ (מה שנותר לכסות) קטן מהר.

הוכחת המשפט (התחלה) נסמן ב-S^* פתרון אופטימלי, נסמן ב-G את הפתרון החמדן. נסמן $|S^*| = k$, $|G| = t$. (נשים לב לכך ש-t זה מספר האיטרציות של האלגוריתם החמדן עד העצירה). לכל $0 \leq j \leq t$, נסמן ב-X_j את העולם שנותר לכסות אחרי j האיטרציות הראשונות של האלגוריתם ($X_0 = [n]$, $X_t = \emptyset$). אנו רוצים להראות כי $\frac{|G|}{|S^*|} \leq \lceil ln(n) \rceil$, או באופן שקול: $t \leq k \lceil ln(n) \rceil$.

למה מרכזית יהי $1 \leq j \leq t$ מספר איטרציה. יהי i^* האינדקס שהאלגוריתם בוחר באיטרציה ה-j של האלגוריתם. אזי $|X_{j-1} \cap A_{i^*}| \geq \frac{|X_{j-1}|}{k}$ (כלומר בכל איטרציה מוסיפים לפחות $\frac{1}{k}$ ממה שנותר להוסיף).

הוכחת הלמה מתקיים $\bigcup_{i \in S^*} A_i = [n]$ לכן:

$$X_{j-1} = X_{j-1} \cap [n] = X_{j-1} \cap \left(\bigcup_{i \in S^*} A_i \right) = \bigcup_{i \in S^*} (X_{j-1} \cap A_i)$$

ולכן קיים $i_0 \in S^*$ כך ש-$|X_{j-1} \cap A_{i_0}| \geq \frac{|X_{j-1}|}{k}$.

על פי דרך פעולתו של האלגוריתם החמדן, מתקיים:

$$|X_{j-1} \cap A_{i^*}| = \underset{1 \leq i \leq r}{max} |X_{j-1} \cap A_i| \geq |X_{j-1} \cap A_{i_0}| \geq \frac{|X_{j-1}|}{k}$$

למת עזר לכל $x \neq 0$, מתקיים $1 - x < e^{-x}$. ולכן, לכל $0 < x \leq 1$ ולכל $y > 0$ מתקיים $(1-x)^y < e^{-xy}$.

הוכחת המשפט (המשך) ההוכחה תהיה בשלילה. נסמן $u = k \lceil ln(n) \rceil$. נניח בשלילה כי $t > u$. בהנחה זו, האלגוריתם לא עוצר אחרי u איטרציות, ולכן $|X_u| \geq 1$. נגיע מכך לסתירה.

מהלמה המרכזית, מתקיים כי לכל $1 \leq j \leq t$, כי $|X_j| \leq (1 - \frac{1}{k})|X_{j-1}|$.

אם $t > u$ אז $X_u \neq \emptyset$ ולכן $|X_u| \geq 1$. אז מתקיים:

$$1 \leq |X_u| \leq (1-\frac{1}{k})|X_{u-1}| \leq (1-\frac{1}{k})^2|X_{u-2}| \leq ... \leq (1-\frac{1}{k})^u|X_0|$$
$$= (1-\frac{1}{k})^{k\lceil ln(n)\rceil} \cdot n < e^{-\frac{1}{k}\cdot k\lceil ln(n)\rceil} \cdot n = e^{-\lceil ln(n)\rceil} \cdot n \leq e^{-ln(n)} \cdot n = 1$$

סתירה כי קיבלנו $1 < 1$.

1.3 בעיית כיסוי ע"י קודקודים $(Vertex\ Cover)$

1.3.1 הצגת הבעיה:

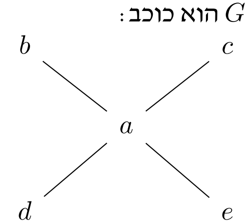
בעיית כיסוי צלעות ע"י קודקודים

<u>הקלט</u>: גרף לא מכוון $G = (V, E)$.

<u>הפלט</u>: תת-קבוצה $S \subseteq V$ כך שלכל $e \in E$ = $\{x, y\}$ מתקיים $x \in S$ או $y \in S$ (או שניהם), וכך שהגודל של S מינימלי בתנאי זה.

1.3.2 דוגמאות:

1. G הוא כוכב:



במקרה זה $|S^*| = 1$.

2. $G = k_n$ (הגרף השלם על n קודקודים). במקרה זה $|S^*| = n - 1$.

3. $G = k_{a,b}$ (גרף דו צדדי שלם על a קודקודים בצד אחד ב-b קודקודים בצד השני). במקרה זה $|S^*| = min\{a, b\}$.

1.3.3 בעיה שקולה:

<u>הגדרה</u> תת קבוצה $I \subseteq V$ נקראת <u>קבוצה בלתי תלויה</u> אם לכל $x, y \in I$ מתקיים כי $\{x, y\} \notin E$.

<u>למה</u> $S \subseteq V$ היא כיסוי קודקודים אם ורק אם $V \setminus S$ היא קבוצה בלתי תלויה.

הוכחה S היא כיסוי קודקודים אם ורק אם לכל צלע $\{x, y\} \in E$ מתקיים $x \in S$ או $y \in S$. שזה שקול לכך שלכל זוג קודקודים $u, v \in V \setminus S$, מתקיים כי $\{u, v\}$ אינו שייך ל-E, שזה שקול לכך ש-$V \setminus S$ היא קבוצת קודקודים בלתי תלויה.

1.3.4 אלגוריתם קירוב חמדני:

ניתן לתאר אלגוריתם חמדני איטרטיבי שבכל שלב מכסה הכי הרבה צלעות ממה שנותר לכסות. זה מקרה פרטי של בעיית הכיסוי ע״י קבוצות. זה ייתן לנו $\lceil ln(n) \rceil$ קירוב.

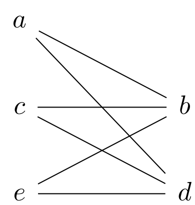
1.3.5 אלגוריתם קירוב לא חמדני:

אלגוריתם 3 אלגוריתם לפתרון מקורב של בעיית כיסוי ע״י קודוקדים

---

1. אתחול: $R = \emptyset$, $X = E$ (R זה הפתרון שנחזיר, X - קבוצת הצלעות שנותר לכסות)
2. איטרציה: נבחר (שרירותית) צלע $\{x, y\} = e \in X$. נעדכן $R = R \cup \{x\} \cup \{y\}$. נעדכן את X ע״י מחיקת כל הצלעות העוברות דרך x או דרך y.
3. סיום: כאשר $x = \emptyset$ נעצור ונחזיר את R.

דוגמאות להפעלת האלגוריתם: $G = k_{3,2}$.



- אם באיטרציה הראשונה האלגוריתם בחר את הצלע $\{a, b\}$ ובשנייה את $\{c, d\}$, אז הוא יוסיף ל-R את a, b ואז את c, d ואז יעצור. לכן $|R| = 4$. אבל הכיסוי האופטימלי הוא בגודל 2 ($\{b, d\}$).

1.3.6 הוכחת קירוב:

טענה האלגוריתם מחזיר פתרון 2 מקרב לבעיית כיסוי ע״י קודקודים. כלומר, יהי S^* הפתרון האופטימלי, ויהי R הפתרון המוחזר ע״י האלגוריתם. אזי:

$$\frac{|R|}{|S^*|} \leq 2$$

הגדרה זיווג בגרף הוא קבוצת צלעות ללא קודקודים משותפים.

למה אם הגרף G מכיל זיווג בגודל t צלעות, אזי כיסוי קודקודים מינימלי ב-G מכיל לפחות t קודקודים. (כלומר $|S^*| \geq t$)

הוכחת הלמה יהי S^* כיסוי קודקודים בגודל מינימלי ב-G. S^* מכיל לפחות קודקוד אחד על כל אחת מ-t צלעות הזיווג. מכיוון שלצלעות האלה אין קודקודים משותפים, כל הקודקודים האלה שונים זה מזה ולכן $|S^*| \geq t$.

הוכחת הטענה נסמן ב-R את הכיסוי המוחזר ע"י האלגוריתם. נסמן ב-S^* פתרון אופטימלי. נסמן ב-t את מספר האיטרציות של האלגוריתם עד העצירה, ונשים לב כי $|R| = 2t$ (כי בכל איטרציה מוסיפים שני קודקודים ל-R). עבור $1 \leq j \leq t$ נסמן ב-e_j את הצלע שהאלגוריתם בוחר באיטרציה ה-j. הבחנה מרכזית: מדרך פעולתו של האלגוריתם, לצלעות $e_1, ..., e_t$ אין קודקודים משותפים, ולכן הן מהוות זיווג עם t צלעות בגרף. לפי הלמה שהוכחנו, מתקיים $|S^*| \geq t$ ולכן $\frac{|R|}{|S^*|} = \frac{2t}{|S^*|} \leq \frac{2t}{t} = 2$.

1.4 בעיית כיסוי ע"י קודקודים ממושקלים $(Weighted\ Vertex\ Cover)$

1.4.1 הצגת הבעיה:

בעיית כיסוי צלעות ע"י קודקודים ממושקלים

<u>הקלט</u>: גרף לא מכוון $G = (V, E)$, ופונקציית משקל $w : V \to \mathbb{R}_+$.

<u>הפלט</u>: תת-קבוצה $S \subseteq V$ כך שלכל $e \in E = \{x, y\}$ מתקיים $x \in S$ או $y \in S$ (או שניהם), וכך שהמשקל של S מינימלי בתנאי זה. $w(S) = \sum\limits_{v \in S} w(v)$.

זו בעיה NP קשה, נחפש אלגוריתם קירוב.

1.4.2 אלגוריתם קירוב:

נציג אלגוריתם קירוב המשתמש בתכנון ליניארי, כלומר ברילקסציה לינארית של בעיית תכנון ליניארי בשלמים (ILP) השקולה לבעייה המקורית. שלבים:

1. <u>נתרגם את הבעיה לבעיית ILP</u>: (מציאת מינימום של פונקציה ליניארית על קבוצת וקטורים המקיימים אילוצים ליניארים ואילוץ של ערכים שלמים).

   נסמן $n = |V|$ לכל תת קבוצה $S \subseteq V$. נתאים וקטור ממשי x באורך n באופן הבא:

$$X(v) = \begin{cases} 1 & v \in S \\ 0 & v \notin S \end{cases}$$

נרצה להגדיר אילוצים ליניארים על x כך שהקבוצה S המתאימה ל-x תהיה כיסוי קודקודים. לכל צלע $\{a, b\} = e \in E$ נדרוש כי $x(a) + x(b) \geq 1$. נרצה לתרגם את משקל הכיסוי של הערך שאנו רוצים למזער לערך של פונקציה ליניארית על וקטורים ממשים באורך n. נשים לב לכך ש-

$$w(S) = \sum_{w \in S} w(v) = \sum_{v \in V} w(v) \cdot x(v)$$

כלומר תכנית ה-ILP היא:

$$ILP: \begin{cases} min \sum_{v \in V} w(V) \cdot x(V) & \\ x(v) \in \{0,1\} & \forall v \in V \\ x(a)+x(b) \geq 1 & \forall \{a,b\} \in E \end{cases}$$

2. <u>בניית רילקסציה ליניארית</u>: זו תכנית LP המחליפה אילוצים של ערכים בשלמים באילוצים ליניאריים עם הדרישה שכל ווקטור המקיים את אילוצי ה-ILP מקיים גם את אילוצי ה-LP.

$$LP: \begin{cases} min \sum_{v \in V} w(v) \cdot x(v) & \\ 0 \leq x(v) \leq 1 & \forall v \in V \\ x(a)+x(b) \geq 1 & \forall \{a,b\} \in E \end{cases}$$

3. נפתור את התכנית הליניארית ונקבל פתרון אופטימלי x^*_{LP}.

4. "נעגל" את x^*_{LP} לפתרון x_R של התכנית בשלמים:

$$x_R(v) = \begin{cases} 1 & x^*_{LP}(v) \geq \frac{1}{2} \\ 0 & x^*_{LP}(v) < \frac{1}{2} \end{cases}$$

1.4.3 דוגמא לאלגוריתם:

דוגמא: $G = k_4$. $w(1) = ... = w(4) = 1$.

$$ILP: \begin{cases} min \sum_{v \in V} w(V) \cdot x(V) & \\ x(v) \in \{0,1\} & \forall v \in V \\ x(1)+x(2) \geq 1 & \\ x(1)+x(3) \geq 1 & \\ \vdots & \\ x(3)+x(4) \geq 1 & \end{cases}$$

$$LP: \begin{cases} min\sum_{v\in V} w(V)\cdot x(V) \\ 0\le x(v)\le 1 & \forall v\in V \\ x(1)+x(2)\ge 1 \\ x(1)+x(3)\ge 1 \\ \vdots \\ x(3)+x(4)\ge 1 \end{cases}$$

1.4.4 הוכחת חוקיות וקירוב:

נסמן ב-$\mathcal{S}_{ILP}\in\mathbb{R}^n$ את אוסף הפתרונות החוקיים לתכנית ה-ILP וב-$\mathcal{S}_{LP}\in\mathbb{R}^n$ את אוסף הפתרונות החוקיים לתכנית ה-LP. נסמן ב-x^* פתרון אופטימלי לתכנית ה-ILP וב-x^*_{LP} פתרון אופטימלי לתכנית הליניארית. נסמן ב-x_R את הפתרון המעוגל שאנו מחזירים.

<u>למה 1</u>

$$\sum_{v\in V} w(v)\cdot x^*_{LP}(v)\le\sum_{v\in V} w(v)\cdot x^*(v)$$

הוכחת למה 1 מתקיים כי $\mathcal{S}_{ILP}\subseteq\mathcal{S}_{LP}$ כי כל ווקטור המקיים את האילוצים של התכנית בשלמים מקיים גם את האילוצים של התכנית הליניארית הרגילה. לכן, $x^*\in\mathcal{S}_{ILP}\subseteq\mathcal{S}_{LP}$ ולכן:

$$\sum_{v\in V} w(v)\cdot x^*_{LP}(v)\le \min_{x\in\mathcal{S}}\sum_{v\in V} w(v)\cdot x(v)\le\sum_{v\in V} w(v)\cdot x^*(v)$$

<u>למה 2</u> לכל $v\in V$ מתקיים $x_R(v)\le 2x^*_{LP}(v)$.

הוכחת למה 2 כי אנחנו מגדילים לכל היותר פי 2 כשאנחנו מעגלים.

<u>טענה</u>

1. x_R הוא פתרון חוקי של התכנית בשלמים.
2. $\sum\limits_{v\in V} w(v)\cdot x_R(v)\le 2\sum\limits_{v\in V} w(v)\cdot x^*(v)$ (כלומר זה 2 קירוב).

הוכחת הטענה

1. לכל $v \in V$ מתקיים $x_R(v) \in \{0,1\}$ לפי הגדרת x_R.

תהי $\{a,b\} \in E$. מכיוון ש-x^*_{LP} הוא פתרון חוקי של התכנית הליניארית הרגילה, מתקיים $x^*_{LP}(a)+x^*_{LP}(b) \geq 1$. לכן, $x^*_{LP}(a) \geq \frac{1}{2}$ או $x^*_{LP}(b) \geq \frac{1}{2}$. לכן $x_R(a) = 1$ או $x_R(b) = 1$. ולכן, $x_R(a) + x_R(b) \geq 1$.

2. מתקיים:

$$\sum_{v\in V} w(v) \cdot x_R(v) \underset{*}{\leq} \sum_{v\in V} w(v) \cdot 2x^*_{LP}(v) = 2\sum_{v\in V} w(v) \cdot x^*_{LP}(v) \underset{**}{\leq} 2\sum_{v\in V} w(v) \cdot x^*(v)$$

כאשר $*$ נובע מלמה 2, ו-$**$ נובע מלמה 1.