אלגוריתמים מבוזרים - הרצאה 12, בעיית עץ פורש מינימלי

שרון מלטר, אתגר 17

18 בספטמבר 2024

תוכן עניינים

1 הקדמה

נתקלנו כבר בבעיית עץ פורש מינימלי בת.ב. היום נלמד באופן מקיף יותר על הבעיה.
ראינו כבר בעיות עם סיבוכיות $O(D),\ O(n)$ והיום נראה בעיה ב־ $\widetilde{O}(D+\sqrt{n})$.

2 עץ פורש מינימלי

2.1 תיאור הבעיה

נתון גרף עם משקלים על הקשתות.
המטרה: למצוא עץ פורש לגרף שסכום משקלי הקשתות שלו מינימלי.
הפלט: כל צומת יודע את הקשתות שסמוכות אליו בעץ.

בתחילת האלגוריתם כל צומת יידע את משקלי הקשתות הסמוכות אלי. נניח שהמשקלים פולינומים, וניתנים לייצוג ע"י $O(logn)$ ביטים (כלומר, ניתן לשלוח משקל בסיבוב אחד) כמו כן, נניח לשם פשטות שמהשקלים ייחודים (אם הם לא, ניתן לשבור סימטריה לפי המזהים)

זו בעיה גלובלית שמצריכה $\Omega(D)$ סיבובים (הוכחנו בת.ב) נתעניין באלגוריתמים ב־ $CONGEST$.

2.2 טענה 1:

טענה: אם נתון רכיב קשירות C בעץ הפורש המינימלי T, ונתסכל על הקשת המינימלית היוצאת מ־ C, שנסמנה e, אז e היא חלק מ־ T.

2.3 רעיון הוכחה לטענה 1

רעיון הוכחה: יהי u צומת מרכיב הקשירות C ו־ v צומת מחוץ לרכיב כך ש־ $u,\ v$ שכנים ונסמן $e=\{u,v\}$. נניח כי e היא הקשת בעלת משקל מינימלי בין צומת מ־ C לצומת מחוץ ל־ C.
נסתכל על המסלול בין u ל־ v ב־ T. אם e היא לא חלק מ־ T, אז חייבת להיות ב־ T קשת אחרת e' שנמצאת על המסלול בין u ל־ v וחוצה את החתך בין C לשאר העץ, כך ש־ $w(e)<w(e')$ (לפי איך שבחרנו את e)
אבל אז הגרף $(T\cup\{e\})/\{e'\}$ הוא עץ פורש בעל משקל קטן משל T, וקיבלנו סתירה.

2.4 אלגוריתם $Boruvka$ - אלג' סדרתי

להלן האלגוריתם;

1. נתחיל כאשר כל צומת הוא רכיב קשירות בפני עצמו.
2. נמצא את הקשת המינימלית שיוצאת מכל רכיב ונאחד רכיבים בהתאם.
3. נחזור על שלב 2 עם הרכיבים החדשים ונמשיך עד שיש רק רכיב אחד.

2.4.1 סיבוכיות האלגוריתם

במקרה הכי גרוע מספר רכיבי הקשירות קטן בחצי בכל איטרציה (מכיוון שכל רכיב מתחבר לרכיב אחד לפחות) אזי לאחר $O(logn)$ האלג' יסתיים.
כלומר, סיבוכיות הזמן של האלגוריתם היא $O(logn)$.

2.4.2 נכונות האלגוריתם

2.4.3 טענה 2

טענה: לאחר סיום האלגוריתם נקבל עץ פורש מינימלי.

2.4.4 הוכחת טענה 2

לפי טענה 1, כל הקשתות שהוספנו הן באמת חלק מעץ פורש מינימלי. נוכיח באינדוקציה שבסוף איטרציה i כל רכיב הוא מגודל לפחות 2^i ולכן אחרי $logn$ איטרציות קיבלנו רכיב מגודל n כך שקיבלנו עץ פורש וסיימנו.

- בסיס: עבור $i = 1$, נכון כי כל צומת מתחבר לעוד צומת אחד לפחות, לכן הרכיב שלו כולל לפחות $2^1 = 2$ צמתים.
- אינדוקציה: אם הטענה נכונה עבור i, אז בסוף איטרציה i הרכיבים הינם בגודל לפחות 2^i. באיטרציה הבאה כל רכיב מתחבר לפחות לרכיב אחד נוסף ולכן הרכיבים החדשים הינם מגודל לפחות $2 \cdot 2^i = 2^{i+1}$ כך שהטענה נכונה עבור $i + 1$.

2.5 אלג' במודל ה־ $CONGEST$

כמה סיבובים נצטרך על מנת לממש את $Boruvka$ ב־ $CONGEST$?
נצטרך לממש $O(logn)$ איטרציות ובכל איטרציה יש למצוא את הקשת המינימלית היוצאת מכל רכיב ולאחד רכיבים.
ניתן למצוא קשת מינימלית יוצאת מרכיב C ב־ $O(D_C)$ כאשר D_C הינו הקוטר של הרכיב C. זאת מכיוון שכך כל צומת יידע מהי הקשת המינימלית שיוצאת מכל צומת אחר מהרכיב.
נצטרך גם למצוא לכל רכיב ייצוג, או **שורש** ־ שיהיה הצומת בעל המזהה המינימלי שנמצא בו. לשם כך נצטרך $O(D_{C'})$ סיבובים כאשר C' הוא הרכיב החדש.
(הקוטר של עץ פורש יכול להיות גדול יותר, למשל עבור גרף קליקה שהעץ הפורש המינימלי שלו הוא גרף שרוך)
באופן כללי, קוטר הרכיב יכול להיות $O(n)$ ולכן סה"כ נצטרך $O(nlogn)$ זמן כדי לממש את האלגוריתם גם בגרפים בהם הקוטר של הגרף נמוך.

לא הכי חמוד. ניתן לטפל בבעיה באופן מהיר יותר?
רעיון: נפריד את רכיבי הקשירות לקטנים ולגדולים. רכיב קטן מכיל לכל היותר $\sqrt{n}$ צמתים וכל רכיב אחר הוא רכיב גדול.
ברכיב קטן נשתמש בתקשורת בתוך הרכיב כדי למצוא **קשת מינימלית יוצאת** בלכל היותר $O(\sqrt{n})$ (מכיוון שקוטר הרכיב לא יכול להיות גדול מ־ $O(\sqrt{n})$)
ברכיב גדול ניעזר בעץ BFS גלובלי בגרף כדי למצוא קשת מינימלית יוצאת. כלומר, אם C הוא רכיב גדול אז כל צומת ב־ C יחשב קשת מינימלית יוצאת ממנו ונשתמש בתקשורת על פני עץ BFS גלובלי כדי לדעת מה המינימום מביניהן. לוקח $O(D)$ זמן עבור רכיב אחד.
יש לכל היותר $O(\frac{n}{\sqrt{n}}) = \sqrt{n}$ רכיבים גדולים. אזי לחשב קשת מינימלית יוצאת מכולם ייקח $O(D+\sqrt{n})$ זמן (נוכיח כעת)

מטרה: חישוב $k = \sqrt{n}$ ערכי מינימום גלובלי, אחד לכל רכיב גדול.

ניתן לבצע את חישוב זה ב־ $O(D + k) = O(D + \sqrt{n})$ זמן ב־ $pipelining$. זהו מודל בו כמה אלגוריתמים רצים במקביל, אך אם צומת מקבל הודעה מכמה אלגוריתמים במקביל, הוא ישלח אותן הלאה לפי סדר המזהים.
ניתן להראות שהאלגוריתם הראשון יסתיים תוך D סיבובים (תרגיל). האלגוריתם השני יסתיים תוך $(D + 1)$ (ייתכן שהוא עוכב בסיבוב אחד ע"י האלגוריתם הראשון, אך לא מעבר לכך) האלגוריתם השלישי יסתיים תוך $(D+2)$ סיבובים וכן הלאה. בסך הכל ניתן להוכיח שכל האלגוריתמים יסתיימו לאחר $(D+k) = (D+\sqrt{n})$ סיבובים.

בעיה: איך נאחד רכיב?
למשל, נניח כי ישנם מספר רכיבים שהקשתות המינימליות שיוצאות מהם הן מאחד לשני, איך נאחד ביניהם? במקרה הגרוע, נצטרך זמן **לינארי** בשביל לעבור על צמתי כל הרכיבים ולעדכן את המזהה שלהם.

רעיון: נאחד רכיבים לפי **שידוך מקסימלי** (כן, כמו בתרגיל הבית)
נזכיר כעת מהו שידוך מקסימלי ונסביר איך הוא יסייע לנו.

נסתכל על הגרף שהצמתים בו הם רכיבי קשירות והקשתות הן קשתות יוצאות מינימליות מכל רכיב. נחשב שידוך מקסימלי בגרף זה, M, (אוסף של קשתות זרות כך שלכל קשת $\{u, v\}$ שלא בשידוך לפחות אחד מ־ $u,\ v$ נמצא בקשת שכן בשידוך)
נאחד רכיבים באופן הבא: תחילה אנחנו נחבר בין רכיבים שסמוכים לקשתות השידוך. עבור רכיב C שאינו בשידוך, אנו יודעים כי קיים שכן של C שכן בשידוך מכיוון שמדובר בשידוך מקסימלי והוא יצטרף לשכן כזה. כלומר, כעת, אם הקוטר המקורי של רכיב היה d, הקוטר החדש שלו יהיה $O(d)$ (הוא גדל לכל היותר פי קבוע ולא יותר מכך) זהו היתרון של שימוש בשידוך מקסימלי :)

2.6 תיאור מלא של האלגוריתם

1. בהתחלה כל צומת הוא רכיב קשירות בפני עצמו.
2. כל עוד הגודל של רכיבים הוא לכל היותר $\sqrt{n}$ נפעל כך;
   (א) נחשב קשתות מינימליות יוצאות מכל רכיב ע"י תקשורת בתוך הרכיב ($O(\sqrt{n})$ זמן)
   (ב) נחשב שידוך מקסימלי M על הקשתות הנ"ל ונאחד רכיבים לפי השידוך כפי שהוסבר מקודם.
3. כאשר הגודל של **כל הרכיבים** הוא יותר מ־ $\sqrt{n}$ נפעל באופן הבא;
   (א) נחשב קשתות מינימליות יוצאות מכל הרכיבים ע"י תקשורת על עץ BFS ($O(D + \sqrt{n})$ זמן כי יש $O(\sqrt{n})$ רכיבים גדולים)
   (ב) נאסוף את $O(\sqrt{n})$ הקשתות שחושבו עץ ה־ $r\ BFS$ שיודע לקבוע איך לאחד את הרכיבים ויודיע לכל רכיב את המזהה החדש לו. בסך הכל מדובר ב־ $O(\sqrt{n})$ הודעות אזי $O(D + \sqrt{n})$ זמן.

כפי שראינו קודם, יש $O(logn)$ איטרציות סה"כ כי בכל שלב כל רכיב מתאחד לפחות עם עוד רכיב אחד.

2.7 המשך סיבוכיות האלגוריתם

נרצה להראות שניתן לחשב כל איטרציה ב־ $\widetilde{O}(D + \sqrt{n})$ זמן.

<u>תרגיל:</u> בגרף בעל דרגת יציאה 1 ניתן לחשב שידוך מקסימלי ב־ $O(log^{*}n)$ זמן ב־ $CONGEST$.

אנחנו פועלים בגרף הרכיבים ולכן כדי לסמלץ סיבוב של האלגוריתם לחישוב מקסימלי, נצטרך זמן פרופורציונאלי לקוטר הרכיב (תלוי באלג') סה"כ ניתן לחשב שידוך בגרף הרכיבים ב־ $O(\sqrt{n}log^{*}n)$ זמן.

אחרי שחישבנו שידוך אנחנו יכולים לאחד את הרכיבים לרכיב חדש ב־ $O(\sqrt{n})$ זמן כי קוטר הרכיב החדש הוא $O(\sqrt{n})$.
כאשר הרכיבים גדולים, יש לכל היותר $\sqrt{n}$ רכיבים ולכן ניתן לחשב את כל הקשתות היוצאות ולאחד ע"י תקשורת על עץ ה־ BFS, גם אם יש שרשראות ארוכות. השורש של עץ ה־ BFS יראה איזה רכיבים מתחברים לרכיב חדש; יקבע להם מזהה חדש ויודיע לכל רכיב מקורי את המזהה החדש שלו. מדובר בתהליך שלוקח $O(D + \sqrt{n})$ זמן מכיוון שיש $O(\sqrt{n})$ רכיבים בזמן הזה.

מתי נעבור מהחישוב הלוקלי של האלגוריתם לשלב 'הגלובלי' של כל האלגוריתם?
כל רכיב שגודלו לכל היותר $\sqrt{n}$ מריץ את האלגוריתם הלוקלי. רכיב שגדול מ־ $\sqrt{n}$ מחכה שכל הרכיבים האחרים יהיו גדולים ורק אז מריצים את החלק הגלובלי.

<u>דקות:</u> רכיב קטן עשוי להתחבר לרכיב גדול בשלב זה, רק צריך ללמוד את המזהה של הרכיב הגדול ב־ $O(\sqrt{n})$ זמן.

2.7.1 טענה 3

<u>טענה:</u> אחרי $O(logn)$ איטרציות גודל כל רכיב הוא לכל הפחות $\sqrt{n}$.

2.7.2 טענה 4:

בסוף איטרציה i, עבור $1 \leqslant i \leqslant log\sqrt{n}$ כל הרכיבים בגרף הם בגודל של לפחות 2^i.
מכאן נובע שכל רכיב שגודלו קטן מ־ $\sqrt{n}$ מתחבר לרכיב אחר ולכן גודלו מוכפל.

2.8 סיכום

האלגוריתם לוקח $\widetilde{O}(D+\sqrt{n})$ זמן כפי שרצינו :)

2.9 חסם תחתון

<u>הערה:</u> ישנו חסם תחתון לבעיה, שהוא באופן מפתיע $\widetilde{\Omega}(D+\sqrt{n})$.

3 בפרקים הבאים

נתחיל לתרגל שאלות למבחן :)
מוזמנים להביא שאלות / חומר בשביל שנעבור עליו.