## חלק 3: השמה בהגרלה

ישנן בעיות חלוקה רבות, שבהן חשוב להבטיח שכל שחקן יקבל לכל היותר חפץ אחד; בעיות כאלו נקראות בעיות השמה (assignment) או שידוך (matching). דוגמה אחת היא חלוקת חדרים בדירה שכורה בין דיירים. שם השגנו הוגנות על־ידי שינוי דמי־השכירות של החדרים, בהתאם להערכות של השחקנים. אולם ישנם מקרים שבהם אי־אפשר להשתמש בפתרון זה. לדוגמה, כשמחלקים מעונות לסטודנטים, דמי־השכירות בדרך־כלל קבועים מראש. לכן ננסה להשיג הוגנות על־ידי הגרלה.

דוגמה נוספת לבעיית השמה היא חלוקת תורנויות: יש n תורנויות שצריך לחלק בין n אנשים. כל התורנויות מתבצעות באותו זמן, ולכן בהכרח כל אדם חייב לבצע תורנות אחת בדיוק. האתגר המרכזי בבעיות השמה הוא: איך להשיג חלוקה הוגנת ויעילה לכתחילה, ויחד עם זה, להבטיח שכל שחקן יקבל חפץ אחד בדיוק בוודאות, לכל תוצאה אפשרית של ההגרלה?

נניח שאנחנו רוצים להשיג השמה אגליטרית־לכתחילה. אנחנו יכולים להשתמש באותה תוכנית ליניארית שהשתמשנו בה כדי להשיג חלוקה אגליטרית־לכתחילה, בתוספת אילוץ האומר, שכל שחקן מקבל חפץ אחד בדיוק:

Maximize z

such that  for all i and g;

 for all g;

 **for all i;**

 for all i.

שימו לב שיש כאן שני סכומים, שאנחנו דורשים שיהיו שווים ל־1: סכום ההסתברויות על כל חפץ צריך להיות 1 – כי כל חפץ נמסר לשחקן אחד בדיוק; וסכום ההסתברויות על כל שחקן צריך להיות 1 – כי כל שחקן מקבל חפץ אחד בדיוק. אם הערכות השחקנים מנורמלות, נקבל השמה פרופורציונלית־לכתחילה. אם נשתמש באלגוריתם הלקסימין, נקבל השמה שהיא גם יעילה־לכתחילה. האתגר כאן הוא איך לבצע את ההגרלות?

**דוגמה**. נניח שהפעלנו את האלגוריתם הלקסימין־אגליטרי על בעיית השמה של ארבעה סוגי מעונות (בקתה, צריף, וילה ומרתף) לארבעה סטודנטים, והפתרון שהתקבל הוא:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **בקתה** | **צריף** | **וילה** | **מרתף** |
| סטודנט א: | 0.7 | 0.3 | 0 | 0 |
| סטודנט ב: | 0 | 0 | 0.7 | 0.3 |
| סטודנט ג: | 0.3 | 0 | 0.3 | 0.4 |
| סטודנט ד: | 0 | 0.7 | 0 | 0.3 |

שימו לב שסכום הערכים בכל שורה ובכל עמודה הוא 1 – בהתאם לאילוצים שהגדרנו. אולם, אם נבצע הגרלה על כל חפץ בנפרד, בהתאם להסתברויות הכתובות בעמודה המתאימה, ייתכן שלא נקבל השמה חוקית. לדוגמה, ייתכן שסטודנט א יזכה גם בבקתה וגם בצריף.

כדי לוודא שתתקבל השמה חוקית, עלינו לבצע הגרלה על השמות שלמות, ולא על כל חפץ בנפרד. לשם כך, עלינו למצוא סדרה של השמות X1,...,Xk (עבור k שלם כלשהו), וסדרה מקבילה של הסתברויות, p1,...,pk, כך שסכום ההסתברויות הוא 1, ואם נבחר את השמה Xj בהסתברות pj, ההסתברות של כל אחד מהשחקנים לקבל כל אחד מהחפצים תהיה זהה להסתברויות הרשומות בטבלה.

בדוגמה למעלה, הגרלה אפשרית היא:

* בהסתברות 0.3: א–צריף, ב–מכולה, ג–בקתה, ד–מרתף.
* בהסתברות 0.4: א–בקתה, ב–מכולה, ג–מרתף, ד–צריף.
* בהסתברות 0.3: א–בקתה, ב–מרתף, ג–מכולה, ד–צריף.

בהגרלה זו k=3. כדי לוודא שההגרלה מתאימה לטבלה, ניתן לחשב, עבור כל סטודנט וכל חדר, מה ההסתברות שהסטודנט יקבל את החדר. לדוגמה, סטודנט א מקבל את הבקתה בהסתברות 0.4+0.3=0.7, בהתאם לרשום בטבלה.

האם לכל מטריצת השמה אקראית קיימת הגרלה על השמות שלמות? התשובה היא כן! כדי להסביר תשובה זו, נשתמש בכלים מתורת הגרפים.

**הגדרה**. נתונה מטריצת־הסתברויות המייצגת חלוקה של הסתברויות על חפצים בין שחקנים. **גרף הצריכה** של המטריצה הוא גרף דו־צדדי לא־מכוון, שבו:

\* הקודקודים בצד אחד הם n השחקנים;

\* הקודקודים בצד השני הם m החפצים;

\* יש צלע בין שחקן i לבין חפץ g אם שחקן i מקבל את חפץ g בהסתברות חיובית ממש;

\* משקל הצלע בין i ל־g שווה להסתברות ששחקן i מקבל את חפץ g.

הנה גרף הצריכה המתאים למטריצה בError! Reference source not found.:

0.7

0.3

0.7

0.3

0.3

0.3

0.4

0.7

0.3

כל השמה אפשרית של שחקנים לחפצים מתאימה לשידוך מושלם בגרף הצריכה. לדוגמה, ההשמה א–צריף, ב–מכולה, ג–בקתה, ד–מרתף מתאימה לשידוך הבא:

אנחנו רוצים "לפרק" את גרף־הצריכה לשידוכים מושלמים. הפירוק נקרא על־שם המתמטיקאי גארט בירקהוף (Birkhoff), שתיאר את הפירוק לראשונה.

**הגדרה**. נתון גרף G עם משקלים חיוביים על הקשתות. פירוק בירקהוף (Birkhoff decomposition) של G הוא סדרה של שידוכים מושלמים X1,...,Xk (עבור k שלם כלשהו), וסדרה מקבילה של משקלים, p1,...,pk, כך שהמשקל של כל קשת e בגרף G שווה לסכום המשקלים של השידוכים המכילים את e.

האם לכל גרף יש פירוק בירקהוף? התשובה היא לא: הרי יש גרפים שאין בהם אפילו שידוך מושלם אחד. אבל אנחנו מתעניינים בגרפים מיוחדים – גרפי־צריכה של מטריצת השמה אקראית. הגרפים האלה מתאפיינים בתכונה הבאה:

**הגדרה**. גרף G עם משקלים חיוביים על הקשתות נקרא מאוזן אם לכל קודקוד v בגרף, סכום המשקלים של הקשתות המחוברות ל-v הוא מספר קבוע.

גרף־הצריכה של כל מטריצת השמה אקראית הוא מאוזן, כי לכל קודקוד, סכום המשקלים של הקשתות המחוברות אליו הוא 1. הדבר נובע מההגדרה של השמה אקראית: סכום הסתברויות ההשמה של כל חפץ הוא 1, וסכום ההסתברויות המשוייכות לכל שחקן הוא 1.

**משפט**. בכל גרף דו־צדדי מאוזן ולא ריק יש שידוך מושלם.

**הוכחה:** נוכיח את המשפט בעזרת משפט ידוע בתורת הגרפים – משפט הנישואין של הול (Hall's marriage theoren). המשפט קובע, שבגרף דו־צדדי יש שידוך מושלם אם־ורק־אם, לכל קבוצה של k קודקודים באחד הצדדים, יש ביחד לפחות k שכנים בצד השני.

נוכיח כעת, שהתנאי של הול מתקיים בכל גרף דו־צדדי מאוזן. נסמן באות w את סכום משקלי הקשתות המחוברות לקודקוד כלשהו (המשקל זהה לכל הקודקודים, לפי הגדרת גרף מאוזן). ניקח קבוצה כלשהי של k קודקודים באחד הצדדים, ונסתכל על כל הצלעות היוצאות מהן. סכום המשקלים על כל הצלעות הללו הוא . כל קודקוד בצד השני "תורם" לסכום הזה לכל היותר את סכום המשקלים בקשתות המחוברות אליו, שהוא w. לכן, כדי להגיע לסכום של , חייבים להיות בצד השני לפחות k שכנים. \*\*\*

כעת נוכיח, שלכל גרף דו־צדדי מאוזן יש פירוק בירקהוף. ההוכחה היא על־ידי אלגוריתם המוצא פירוק כזה.

הקלט: גרף דו־צדדי מאוזן ולא ריק, G.

1. איתחול: j=1.

2. מצא שידוך מושלם ב־G; סמן אותו ב Xj.

3. מצא את המשקל הקטן ביותר של קשת כלשהי בשידוך Xj; סמן אותו ב־pj.

4. לכל קשת e בשידוך Xj: הפחת pj מהמשקל של e. אם המשקל החדש 0, מחק את e.

5. אם הגרף ריק, סיים; אחרת, הגדל את j ב־1 וחזור לצעד 2.

**משפט**. האלגוריתם הנ"ל מוצא לכל גרף דו־צדדי מאוזן G, פירוק בירקהוף של G בזמן פולינומיאלי במספר הקודקודים.

**הוכחה:** נוכיח תחילה טענת־עזר: בכל פעם שהאלגוריתם מגיע לצעד 2, הגרף G מאוזן, ולכן לפי המשפט שהוכחנו קודם יש בו שידוך מושלם. הוכחה: בתחילת האלגוריתם, הטענה נכונה לפי ההנחה. נסמן באות w את סכום המשקלים בקשתות המחוברות לקודקוד כלשהו. לאחר שהאלגוריתם מוצא שידוך מושלם, הוא מפחית מספר קבוע pj מהמשקל של כל קשת בשידוך. כיוון שהמספר pj הוא המשקל הקטן ביותר של קשת כלשהי בשידוך, משקלי כל הקשתות נשארים לפחות 0. כיוון שהשידוך מושלם, כל קודקוד מופיע בשידוך בדיוק פעם אחת. לכן, לכל קודקוד, סכום משקלי הקשתות המחוברות לקודקוד משתנה ל: w–pj. סכום המשקלים עדיין קבוע לכל הקודקודים, ולכן הגרף עדיין מאוזן. סוף הוכחת הטענה.

בכל פעם שהאלגוריתם מגיע לצעד 4, לפחות קשת אחת נמחקת – הקשת שמשקלה היה בדיוק pj. מספר הקשתות בגרף הוא לכל היותר n2, כאשר n הוא מספר הקודקודים בכל צד (כי יש לכל היותר קשת אחת בין כל קודקוד בצד אחד לכל קודקוד בצד השני). לכן האלגוריתם יסתיים תוך n2 סיבובים לכל היותר.[[1]](#footnote-1) בכל סיבוב צריך לחשב שידוך מושלם; הדבר ניתן לביצוע ע"י אלגוריתמים ידועים הרצים בזמן פולינומיאלי. לכן האלגוריתם כולו מסתיים בזמן פולינומיאלי. לפי הגדרת המשקלים pj, סדרת השידוכים והמשקלים שהאלגוריתם מוצא היא פירוק בירקהוף של G. \*\*\*

נדגים את האלגוריתם על הגרף בדוגמה למעלה. נניח שבסיבוב 1 נמצא השידוך המושלם א–צריף, ב–מכולה, ג–בקתה, ד–מרתף (זה יהיה השידוך X1). המשקל הקטן ביותר של קשת בשידוך הוא 0.3, ולכן p1=0.3. אחרי שמפחיתים 0.3 מארבעת הקשתות בשידוך, ומוחקים קשתות שמשקליהן הפכו לאפס, נשאר הגרף הבא:

0.7

0.4

0.3

0.3

0.4

0.7

שימו לב שהגרף עדיין מאוזן – לכל קודקוד, סכום משקלי הקשתות המחוברות לקודקוד הוא 0.7.

בסיבוב 2, האלגוריתם עשוי למצוא את השידוך המושלם א–בקתה, ב–מכולה, ג–מרתף, ד–צריף (זה יהיה השידוך X2). המשקל הקטן ביותר של קשת בשידוך הוא 0.4, ולכן p2=0.4. אחרי שמפחיתים 0.4 מארבעת הקשתות בשידוך, ומוחקים קשתות במשקל אפס, נשאר הגרף הבא:

0.3

0.3

0.3

0.3

שימו לב שהגרף עדיין מאוזן – לכל קודקוד, סכום משקלי הקשתות המחוברות אליו הוא 0.3.

בסיבוב השלישי נשאר רק שידוך מושלם אחד: א–בקתה, ב–מרתף, ג–מכולה, ד–צריף. זהו השידוך X3. משקל כל הקשתות בו הוא 0.3, ולכן p3=0.3. אחרי שמפחיתים 0.3 מארבעת הקשתות בשידוך, ומוחקים קשתות במשקל אפס, נשאר גרף ריק, ולכן האלגוריתם מסתיים.

### הגרלה הוגנת ויעילה לכתחילה.

בעזרת הכלים שלמדנו בסעיף הקודם, נתאר אלגוריתם המוצא הגרלה בין השמות, שהיא גם הוגנת־לכתחילה וגם יעילה־לכתחילה.

1. מצא חלוקה שברית לקסימין־אגליטרית, לפי האלגוריתם בפרק 3, עם אילוץ נוסף הקובע, שסכום השברים המשוייך לכל שחקן שווה בדיוק ל־1.

2. בנה את גרף־הצריכה של החלוקה.

3. מצא פירוק־בירקהוף של גרף־הצריכה בעזרת אלגוריתם בירקהוף. פירוק זה מגדיר הגרלה בין השמות.

ההגרלה המוחזרת היא לקסימין־אגליטרית, ולכן היא יעילה־לכתחילה. אם ערכי השחקנים מנורמלים, אז ההגרלה היא גם פרופורציונלית־לכתחילה.

מה אם רוצים שההגרלה תהיה לא רק פרופורציונלית־לכתחילה אלא גם ללא־קנאה־לכתחילה? כאן התשובה מורכבת יותר. למדנו, שחלוקה הממקסמת את מכפלת ערכי השחקנים (או את סכום הלוגריתמים של ערכי השחקנים) היא יעילה־פארטו וללא־קנאה. היה אפשר לחשוב, שאם נוסיף את האילוץ, שסכום השברים המשוייך לכל שחקן שווה 1, נקבל מטריצת השמה אקראית, שהיא ללא־קנאה־לכתחילה. אבל זה לא נכון אפילו כשיש שני שחקנים.

**דוגמה**. נתונים שני חפצים ושני שחקנים עם הערכים הבאים:

|  |  |  |
| --- | --- | --- |
|  | **בקתה** | **צריף** |
| **עמי:** | 6 | 12 |
| **תמי:** | 12 | 18 |

ללא האילוץ שסכום השברים של כל שחקן צריך להיות 1, החלוקה הממקסמת את מכפלת הערכים נותנת לעמי 5/6 מהצריף, ואת כל השאר לתמי (מכפלת הערכים היא ). ניתן לוודא שאף אחד מהם אינו מקנא בשני. עם זאת, כיוון שגרף־הצריכה של ההשמה אינו מאוזן, אי־אפשר לפרק אותו להגרלה על השמות.

אם מוסיפים את האילוץ שסכום השברים של כל שחקן שווה 1, החלוקה הממקסמת את מכפלת הערכים נותנת לעמי את הצריף בהסתברות 1, ולתמי את הבקתה בהסתברות 1 (מכפלת הערכים היא ). זוהי כבר השמה לא־אקראית, ואין צורך לפרק אותה. אבל יש בה קנאה – תמי מקנאת בעמי.

אאנונד היילנאנד (Hylland) וריצ'ארד זקהאוסר (Zeckhauser) פיתחו אלגוריתם אחר, המוצא חלוקה יעילה־לכתחילה וללא־קנאה־לכתחילה עם האילוץ שסכום השברים של כל שחקן שווה 1, כך שאפשר לפרק את המטריצה להגרלה על השמות. האלגוריתם שלהם הוא מסובך למדי, זמן הריצה שלו אינו פולינומיאלי במספר השחקנים, והסיבוכיות המדוייקת שלו היא עדיין שאלה פתוחה. למיטב ידיעתנו, עדיין לא נמצא אלגוריתם פולינומיאלי לפתרון הבעיה.

סיכם: אראל סגל-הלוי.

1. למעשה, האלגוריתם יסתיים תוך n2–n+1 סיבובים לכל היותר, כי בסיבוב האחרון נשארות תמיד n קשתות שמשקליהן הופכים ל־0 בו־זמנית. [↑](#footnote-ref-1)