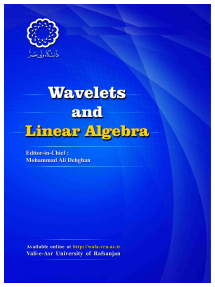
موجک‌ها و جبرخطی ۱۱(۳) (۱۴۰۳) ۹۱-۱۱۰



موجک‌ها و جبرخطی

http://wala.vru.ac.ir

دانشگاه ولیعصر (عج)

رفسنجان

الگوریتمی تقریبی برای حل مساله‌ی خوشه‌بندی همبستگی با استفاده از عملیات ماتریسی

*علی شکیبا[آ]

[آ] گروه علوم کامپیوتر، دانشکده علوم ریاضی، دانشگاه ولی‌عصر (عج) رفسنجان، کرمان، ایران

اطلاعات مقاله

تاریخچه مقاله:
دریافت شده:۲۶ بهمن ۱۴۰۱
پذیرفته شده:۲۷ خرداد ۱۴۰۲
دسترسی آنلاین: ۱۲ خرداد ۱۴۰۳

کلمات کلیدی:

خوشه‌بندی همبستگی،

جبرخطی، الگوریتم تقریبی.

چکیده

مساله‌ی خوشه‌بندی همبستگی، یکی از شهودی‌ترین مدل‌های افراز یک گراف مشابهت به اجتماعی از خوشه‌ها است. متاسفانه این مساله در رده‌ی مسائل $\mathbb{NP}$-سخت قرار دارد. به همین دلیل، ارائه‌ی یک الگوریتم کارآ از زمان چندجمله‌ای که یک افراز دقیق و بهینه را برای گراف‌های دلخواه ایجاد نماید، بعید به‌نظر می‌رسد. در این مقاله، ابتدا یک فرمول‌بندی جدید از این مساله با استفاده از رویکردی حریصانه به منظور محاسبه‌ی پاسخی تقریبی در زمان چندجمله‌ای ارائه خواهیم داد. سپس، با استفاده از عملیات پایه‌ی ماتریسی، فرمول‌بندی معادل از الگوریتم ارائه شده را ارائه خواهیم کرد که در هر زبان برنامه‌نویسی مبتنی بر عملیات ماتریسی، به سادگی قابل پیاده‌سازی است. علاوه بر این، فرمول‌بندی ارائه شده امکان استفاده از توازی موجود در عملیات ماتریسی پایه را فراهم می‌نماید.

*نویسنده مسئول
آدرس ایمیل: ali.shakiba@vru.ac.ir(علی شکیبا)

http://doi.org/10.22072/WALA.2023.1988384.1411

۱. مقدمه

یکی از مهمترین مسائل در یادگیری ماشین، «خوشه‌بندی» مجموعه‌ای از اشیاء به «خوشه»هایی است که اشیاء داخل خوشه بیشترین شباهت را به یکدیگر و بیشترین فاصله را از اشیاء خوشه‌های دیگر داشته باشند. فرمول‌بندی‌ها و الگوریتم‌های متعددی از مساله‌ی خوشه‌بندی ارائه شده است. خوشه‌بندی همبستگی[1] یکی از بدیهی‌ترین فرمول‌بندی‌ها برای مساله‌ی خوشه‌بندی است [۱]. این فرمول‌بندی دارای کاربردهای فراوانی در تشخیص اجتماعات در شبکه‌های اجتماعی [۵]، مسائل رفع‌ابهام [۸]، و اثرکلی[2] خوشه‌ها [۳] است. در ساده‌ترین حالت، ورودی یک مساله‌ی خوشه‌بندی همبستگی مجموعه‌ای از اشیاء مانند V و یک تابع مشابهت دودویی بین زوج اشیاء مانند $\mathrm{sim} : \binom{|V|}{۲} \to \{۰, ۱\}$ است. تنها شرط لازم برای تابع شباهت، تقارن است. به عبارت دیگر، برای هر $u, v \in V$ داشته باشیم $\mathrm{sim}(u, v) = \mathrm{sim}(v, u)$. دو شیء u و v مشابه هستند اگر $\mathrm{sim}(u, v) = ۱$ باشد، در غیر این صورت غیرمتشابهند. بر اساس این فرمول‌بندی، انتظار داریم تا اشیاء مشابه در یک خوشه قرار گیرند و اشیاء متفاوت در خوشه‌ای یکسان قرار نگیرند. هر افراز از مجموعه‌ی V مانند $V_۱, \ldots, V_K$ که در آن $V_i \neq \emptyset$ است، یک خوشه‌بندی گوییم. به منظور نمایش یک خوشه‌بندی، می‌توان از تابع برچسبی[3] مانند $\ell : V \to \mathbb{N}$ استفاده نمود. در این صورت، هر خوشه را می‌توان مجموعه‌ای ماکسیمال از اشیاء نامید که مقدار تابع برچسب آن‌ها یکسان است. به عبارت دقیق‌تر، هدف یافتن تابع برچسب‌گذاری مانند ℓ است به‌طوری‌که تابع هزینه‌ی

$$\mathrm{cost}(\ell) = \sum_{\substack{\{u,v\}\subseteq V \\ \ell(u)\neq\ell(v)}} \mathrm{sim}(u, v) + \sum_{\substack{\{u,v\}\subseteq V \\ \ell(u)=\ell(v)}} (۱ - \mathrm{sim}(u, v)) . \qquad (۱.۱)$$

کمینه شود. بخش اول تابع هزینه‌ی رابطه‌ی (۱.۱) مربوط به جفت داده‌هایی است که در خوشه‌های متفاوتی هستند. بخش دوم، مربوط به جفت داده‌هایی است که در یک خوشه قرار دارند. مسلم است تابعی که جفت داده‌های مشابه را در یک خوشه و جفت داده‌های غیرمشابه را در خوشه‌های متفاوتی قرار می‌دهد، جواب مطلوب و خوشه‌بندی ایده‌آل است. از آنجا که مساله‌ی خوشه‌بندی همبستگی یک مساله‌ی $\mathbb{NP}$ـسخت است [۱]، روش‌های مختلفی برای بدست آوردن یک جواب قابل قبول، و البته تقریبی، ارائه شده است. برای نمونه، الگوریتم ۲ـتقریب ارائه شده در [۱]، به یک الگوریتم $O(۱)$ـتقریب در [۶] توسعه داده شده است. علاوه بر این، در [۴] یک الگوریتم ۲/۰۶ـتقریب به منظور افزایش بیشترین مشابهت درون خوشه‌ای، فارغ از تعداد تفاوت‌ها ارائه شده است. یکی از اخیرترین الگوریتم‌های تقریبی برای حل این مساله به‌منظور کمینه‌سازی تعداد

[1] Correlation clustering
[2] Ensemble
[3] Labeling function

یال‌های ناموجود در خوشه‌ها، در [۶] ارائه شده است. همچنین الگوریتم‌های مختلفی برای حل تقریبی مساله‌ی خوشه‌بندی همبستگی در مدل‌های محاسباتی مختلف ارائه شده‌اند. یکی از این مدل‌ها، گراف‌های دینامیک است که در آن، راس‌های جدید می‌توانند به گراف اضافه شوند، یال‌های مشابهت حذف و یا ایجاد شوند و یا راس‌ها حذف شوند [۱۰].

یکی از رویکردهای مرسوم در ابداع الگوریتم‌های تقریبی برای حل مسائل دشوار، رویکرد حریصانه [۱۱] به منظور کاهش تابع هزینه است. در این رویکرد تکراری، تلاش می‌شود تا در هر تکرار، مقدار تابع هزینه بیشترین کاهش ممکن در آن گام را پیدا کند. در حالت کلی، این رویکرد الزامی به رسیدن به بهینه‌ی عمومی ندارد، اما در صورت همگرایی، به یک بهینه‌ی محلی خواهیم رسید که می‌تواند بسته به شرایط مساله، قابل قبول باشد. در این مقاله، ابتدا رویکردی حریصانه برای حل مساله‌ی خوشه‌بندی چگالی ارائه خواهیم داد و به تحلیل آن خواهیم پرداخت. در این رویکرد، از فرمول‌بندی هم‌ارز با رابطه‌ی (۱.۱) و مبتنی بر گراف استفاده می‌کنیم. در عمل، از توازی برای کاهش زمان انتظار کاربر جهت محاسبه‌ی پاسخ استفاده می‌شود. لازم به ذکر است که هر الگوریتمی را نمی‌توان به صورت موازی اجرا نمود و قیود خاصی در این زمینه وجود دارد. علاوه بر این، استفاده از زیرساخت‌های موازی‌سازی موجود نظیر پردازنده‌های گرافیکی یا آرایه‌های پردازشی همگن و یا ناهمگن با معماری‌های مختلف نیازمند بهره‌گیری از کتابخانه‌های تخصصی و انحصاری هر زیرساخت است. خوشبختانه، عملیات‌های پایه‌ی جبرخطی[۴] [۲] مانند ضرب‌داخلی بردارها، ضرب ماتریس در بردار و ضرب ماتریس در ماتریس در بسیاری از این کتابخانه‌های تخصصی موازی‌سازی به صورت پیش‌فرض پیاده‌سازی شده است. استاندارد GraphBLAS[۷]، چارچوبی است که عملیات‌های پایه در الگوریتم‌های گراف را با استفاده از عملیات پایه‌ی جبرخطی و نمایش ماتریسی گراف‌ها در قالب ماتریس مجاورت یا ماتریس وقوع فراهم می‌کند. با الگوگیری از این استاندارد، روش پیشنهادی را با استفاده از عملیات پایه‌ی جبرخطی به صورت یک مساله‌ی ماتریسی فرمول‌بندی می‌کنیم. بدین صورت، امکان پیاده‌سازی روش پیشنهادی در هر زیرساخت موازی‌سازی که عملیات BLAS را فراهم کند، به سادگی امکان‌پذیر خواهد بود.

ادامه‌ی مقاله به شرح زیر سازمان‌یافته است: ابتدا در بخش ۲، فرمول‌بندی مبتنی بر نظریه‌ی گراف از مساله‌ی خوشه‌بندی همبستگی ارائه می‌شود. سپس، در بخش ۳، روش پیشنهادی شرح داده می‌شود. این روش به سادگی در زبان‌های برنامه‌نویسی به صورت سریال قابل پیاده‌سازی است. اما به منظور پیاده‌سازی آن با استفاده از BLAS، نیازمند فرمول‌بندی مبتنی بر جبرخطی از مساله‌ی خوشه‌بندی همبستگی هستیم که در بخش ۴ ارائه می‌شود. پایه‌ای‌ترین عملیات در روش پیشنهادی، انتقال راس از یک خوشه به خوشه‌ی دیگر است که در بخش ۵ به صورت دقیق مورد تحلیل قرار گرفته و نحوه‌ی محاسبه‌ی اثر آن با استفاده از عملیات ماتریسی مورد بحث قرار می‌گیرد. سپس، در بخش ۶، الگوریتم پیشنهادی در بخش ۳ با استفاده از عملیات ماتریسی پیاده‌سازی شده و پیچیدگی زمانی آن مورد بررسی قرار می‌گیرد. بخش ۷ به جمع‌بندی و نتیجه‌گیری مقاله

[4]Basic Linear Algebra Subprograms (BLAS)

خواهد پرداخت و برخی سوالات پژوهشی را مطرح خواهد کرد.

۲. پیش‌نیازها

به صورت معادل، می‌توان مساله‌ی خوشه‌بندی همبستگی را به صورت یک گراف بدون جهت ساده فرمول‌بندی نمود [۹]، که در آن، هر راس متناظر با یک شیء است. در این گراف، که به آن گراف مشابهت[۵] گفته می‌شود؛ بین دو راس $u, v \in V$ یال وجود دارد اگر و تنها اگر $\text{sim}(u, v) = ۱$ باشد. گراف مشابهت را به $G = (V, E)$ نمایش می‌دهیم که در آن $|V| = n$ و $|E| = m$ است. در این صورت، تابع هزینه به صورت

$$\text{cost}_{Orig} = \min_{\ell} \sum_{\substack{u,v \in V \\ \{u,v\} \in E \\ \ell(u) \neq \ell(v)}} ۱ + \sum_{\substack{u,v \in V \\ \{u,v\} \notin E \\ \ell(u) = \ell(v)}} ۱. \tag{۱.۲}$$

قابل بیان است. در این مقاله، به تابع هزینه‌ی رابطه‌ی (۱.۲)، تابع هزینه‌ی اصلی گوییم.

قضیه ۱.۲. *تابع برچسب‌گذاری ℓ مقدار تابع هزینه‌ی (۱.۱) را کمینه می‌کند اگر و تنها اگر مقدار تابع هزینه‌ی (۱.۲) کمینه شود.*

اثبات. تابع هزینه‌ی رابطه‌ی (۱.۱) از دو بخش تشکیل شده است:

$$A = \sum_{\substack{\{u,v\} \subseteq V \\ \ell(u) \neq \ell(v)}} \text{sim}(u, v),$$

و

$$B = \sum_{\substack{\{u,v\} \subseteq V \\ \ell(u) = \ell(v)}} (۱ - \text{sim}(u, v)).$$

با توجه به تعریف گراف مشابهت، می‌توان دید که $\text{sim}(u, v) = ۱$ است اگر و تنها اگر $\{u, v\} \in E$. بنابراین، A را

[۵] Similarity graph

می‌توان به صورت

$$\sum_{\substack{u,v\in V\\ \{u,v\}\in E\\ \ell(u)\neq\ell(v)}} ۱,$$

بازنویسی نمود. همچنین B را می‌توان به صورت

$$\sum_{\substack{\{u,v\}\subseteq V\\ \ell(u)=\ell(v)}} (۱ - \mathrm{sim}(u,v)),$$

نمایش داد، زیرا $۱ - \mathrm{sim}(u,v) = ۱$ است اگر و تنها اگر یالی بین رئوس u, v وجود نداشته باشد. □

۳. روش تکراری پیشنهادی

به منظور رسیدن به یک جواب تقریبی مناسب، از رویکردی تکراری استفاده خواهیم کرد که هر تکرار، از چند گام تشکیل شده‌است. عملیات پایه‌ی مورد استفاده در الگوریتم پیشنهادی، انتقال راس $v_i \in V$ از خوشه‌ی c که در آن قرار دارد، به خوشه‌ی c' است. این عملیات را به $c \xrightarrow{i} c'$ نمایش می‌دهیم. مراحل کلی در الگوریتم پیشنهادی بدین شرح است: ابتدا، تابع همانی را به عنوان تابع برچسب در نظر می‌گیریم. به عبارت دیگر، هر راس را در یک خوشه‌ی مجزا لحاظ می‌کنیم. سپس، سعی می‌کنیم با انجام عملیات انتقال راس تصادفی i از خوشه‌ی خودش به خوشه‌ی دیگری، مقدار تابع هزینه‌ی خوشه‌بندی را کاهش دهیم و تابع برچسب را به‌روزرسانی کنیم. بنابراین، تنها در صورتی عملیات انتقال یک راس از یک خوشه به خوشه دیگر مفید خواهد بود که هزینه‌ی خوشه‌بندی کاهش پیدا نماید. پس از بررسی تمام راس‌ها، یک تکرار به پایان می‌رسد. پس از آن، هر تکرار مانند تکرار اول خواهد بود، با این تفاوت که به جای تابع همانی، از تابع برچسب تکرار قبل فرایند را آغاز می‌کند. البته در آغاز هر تکرار، خوشه‌ها را مورد شماره گذاری مجدد قرار می‌دهیم تا خوشه‌های تهی حذف شوند. تکرارها زمانی به همگرایی می‌رسند که یکی از شرایط زیر برقرار شوند:

۱. به حداکثر تعداد تکرارها رسیده باشیم.

۲. در طی تکرار جاری، هیچ راسی انتقال پیدا نکند. به عبارت دیگر، تابع برچسب در پایان تکرار، همان تابع برچسب در ابتدای تکرار باشد.

۳. میزان تغییر در تابع هدف در طی تکرار، از یک مقدار آستانه بیشتر نباشد.

به منظور جلوگیری از ورود به حلقه‌ی بی‌نهایت و یا انتقال تکراری بین خوشه‌های مبدا و مقصد، در هر تکرار، هر راس تنها یک‌بار برای انتقال بررسی می‌شود. از آنجا که تابع هزینه (رابطه‌ی (۱.۱))، یک تابع نامنفی بوده و در هر انتقال، طبق تعریف عملیات انتقال، مقدار تابع هزینه کاهش پیدا می‌کند؛ بنابراین الگوریتم پیشنهادی برای کمینه‌کردن تابع هزینه، همگرا خواهد بود.

[الگوریتم ۱]

```
l = range(len(V))
while(True):
  unprocessed = V
  while(len(unprocessed) > 0):
    let v be random vertex in unprocessed
    c = l[v]
    let c_p be the cluster maximizes cost function
       decrease
    if c != c_p:
      l[v] = c_p
    unprocessed -= set((v,))
    renumber clusters in l
    if stop criteria is met:
      break
```

قضیه ۱.۳. *الگوریتم پیشنهادی برای حل تقریبی مساله‌ی خوشه‌بندی همبستگی، در هر تکرار، خطوط ۳ الی انتهای الگوریتم ۳، با $|V| = n$ راس و تابع برچسب $\ell : V \to [K]$ نیازمند $O(|V| \times K)$ محاسبه‌ی تابع هزینه (رابطه‌ی (۱.۲)) است.*

اثبات. در هر تکرار، هر راس یک بار پردازش می‌شود. در هر پردازش، ابتدا میزان کاهش بر اثر انتقال آن راس از خوشه‌ی جاری، c، به خوشه‌ی هدف محاسبه می‌شود. سپس، خوشه‌ی بهینه که منجر به بیشترین کاهش در تابع هزینه می‌گردد، به عنوان c_p انتخاب می‌شود. تعداد خوشه‌های مقصد ممکن، K خوشه است و تعداد راس‌هایی که در هر تکرار پردازش می‌شوند، $|V|$ راس است. بنابراین، تابع هزینه $O(|V| \times K)$ نوبت فراخوانی می‌شود. □

۴. فرمول‌بندی ماتریسی از مساله‌ی خوشه‌بندی همبستگی

به ازای هر تابع برچسب‌گذاری مانند $\ell : V \to [K]$ می‌توان ماتریس دودویی معادل با آن را به صورت $L = [\ell_{i,c}]_{|V|\times K}$ تعریف کرد. در این ماتریس، $\ell_{i,c} = ۱$ اگر راس i به خوشه‌ی c متعلق باشد. در غیراین‌صورت، $\ell_{i,c} = ۰$ منظور می‌شود. همچنین، فرض کنید ماتریس مجاورت گراف $G = (V, E)$ را به $A_G = [a_{i,j}]_{|V|\times|V|}$ نمایش دهیم که

در آن، $a_{i,j} = ۱$ است هرگاه بین رئوس i و j، یالی در گراف مشابهت وجود داشته باشد و در غیر این صورت، صفر منظور می‌شود.

منظور از بردار خوشه‌ی c برای $c = ۱, \dots, K$، برداری ستونی مانند ℓ_c است به‌طوری‌که درایه‌ی jـام آن برابر با یک است اگر $\ell(v_j) = c$ باشد، در غیر این صورت برابر با صفر خواهد بود. به عبارت دیگر، داریم

$$\ell_c = \sum_{\substack{v_j \in V \\ \ell(v_j)=c}} e_j, \tag{۱.۴}$$

که در آن v_j برداری ستونی با تنها یک درایه‌ی یک است که سایر درایه‌های آن به غیر از درایه‌ی jـام برابر با صفر هستند.

لم ۱.۴. *ماتریس $C = L^T L$ یک ماتریس قطری است که در آن، هر درایه‌ی قطری نشان‌دهنده‌ی تعداد رئوس حاضر در خوشه‌ی متناظر با آن درایه‌ی قطری است.*

اثبات. با توجه به اینکه ℓ یک تابع پوشا است، بنابراین هر راس تنها در یک خوشه قرار می‌گیرد. در نتیجه، برای هر $c \neq c'$ داریم $\ell_c^T \ell_{c'} = ۰$. بنابراین، درایه‌های غیرقطری ماتریس C برابر صفر هستند. حال، درایه‌ی قطری L_{cc} را در نظر بگیرید. از تعریف ضرب ماتریس‌ها و تعامد بردارهای ℓ_c، می‌دانیم این درایه برابر با $\ell_c^T \ell_c$ است که نشان‌دهنده‌ی تعداد رئوس خوشه‌ی c است. □

با استفاده از ماتریس C، ماتریس T را به صورت $T = C(C \ominus ۱)$ تعریف می‌کنیم که منظور از $C \ominus ۱$، تفریق نامنفی مقدار اسکالر ۱ از تمام درایه‌های ماتریس C است. به عبارت دقیق‌تر، داریم

$$c_{i,j} \ominus ۱ = \max\{۰, c_{i,j} - ۱\}. \tag{۲.۴}$$

به سادگی می‌توان دید که ماتریس T نیز طبق تعریف و لم ۱.۴، یک ماتریس قطری است.

لم ۲.۴. *درایه‌های قطری ماتریس $L^T A_G L$ بیانگر دو برابر تعداد راس‌های حاضر در خوشه‌ی متناظر با آن درایه‌ی قطری هستند.*

اثبات. اگر قرار دهیم $B = A_G L$، آنگاه $b_{i,c}$ برابر با تعداد همسایگان راس i است که در خوشه‌ی c قرار دارند. در این صورت، درایه‌ی غیرقطری ماتریس $D = L^T B$ مانند $d_{i,j}$ برابر با تعداد یال‌هایی از گراف مشابهت است که یک سر آن‌ها

در خوشه‌ی iـام و سر دیگر در خوشه‌ی jـام قرار دارد. درایه‌های قطری ماتریس D مانند $d_{i,i}$، نشان‌دهنده‌ی دو برابر تعداد یال‌های گراف مشابهت است که در خوشه‌ی iـام قرار دارند. □

لم ۳.۴. *تعداد زوج راسی که در یک خوشه قرار دارند، اما در گراف مشابهت یالی بین آن‌ها وجود ندارد برابر است با:*

$$\frac{۱}{۲}\operatorname{tr}\left(T - L^T A_G L\right). \tag{۳.۴}$$

اثبات. از لم ۱.۴ می‌دانیم درایه‌های قطری ماتریس C بیانگر تعداد راس‌های حاضر در خوشه‌ی متناظر با آن درایه هستند. از طرف دیگر، تعداد یال‌های ممکن بین n راس برابر با $\frac{n(n-۱)}{۲}$ است. در نتیجه، گراف کامل حاصل از درنظر گرفتن C_{cc} راس در خوشه‌ی cـام، دارای $\frac{C_{cc}(C_{cc}\ominus ۱)}{۲} = \frac{۱}{۲}T_{cc}$ یال خواهد بود. از طرف دیگر، طبق لم ۲.۴، هر درایه‌ی قطری ماتریس $L^T A_G L$، دو برابر تعداد راس‌های حاضر در خوشه‌ی متناظر با آن درایه است. در نتیجه، درایه‌های قطری ماتریس $T - L^T A_G L$ برابر با دو برابر تعداد زوج رئوسی در خوشه‌های متناظر است که در آن خوشه قرار دارند، اما در گراف مشابهت مجاور نیستند. با استفاده از تعریف عملگر tr و تقسیم حاصل بر دو، نتیجه‌ی دلخواه حاصل می‌شود. □

لم ۴.۴. *تعداد زوج راس‌هایی که در گراف مشابهت بین آن‌ها یال وجود دارد، اما در خوشه‌های متفاوتی قرار گرفته‌اند، از رابطه‌ی*

$$m - \frac{۱}{۲}\operatorname{tr}\left(L^T A_G L\right), \tag{۴.۴}$$

بدست می‌آید.

اثبات. از تعریف گراف مشابهت می‌دانیم که این گراف دارای m یال است. همچنین از لم ۲.۴ می‌دانیم تعداد یال‌هایی از گراف مشابهت که در یک خوشه قرار می‌گیرند از رابطه‌ی $\frac{۱}{۲}\operatorname{tr}\left(L^T A_G L\right)$ بدست می‌آید. بنابراین، تعداد یال‌هایی که بین خوشه‌های مختلف وجود دارند از تفاضل این دو کمیت بدست می‌آید. □

در قضیه ۵.۴ نشان می‌دهیم که چگونه می‌توان رابطه‌ی (۱.۲) را با استفاده از عملیات ماتریسی بازتعریف کرد.

قضیه ۵.۴. *تابع هزینه‌ی ماتریسی*

$$\operatorname{cost}_{mat} = \min_{\ell}\left[m + \frac{۱}{۲}\operatorname{tr}(T) - \operatorname{tr}\left(L^T A_G L\right)\right], \tag{۵.۴}$$

هم‌ارز با تابع هزینه‌ی رابطه‌ی (۱.۲) *است.*

اثبات. با استفاده از لم‌های ۳.۴ و ۴.۴، می‌توان نوشت

$$\text{cost}_{mat} = \min_{\ell} \left[m - \frac{1}{2} \operatorname{tr}\left(L^T A_G L\right) + \frac{1}{2} \operatorname{tr}\left(T - L^T A_G L\right) \right].$$

با استفاده از ویژگی خطی‌بودن عملگر tr نسبت به جمع ماتریس‌ها، نتیجه‌ی موردنظر حاصل می‌شود. □

نتیجه ۶.۴. *هر تابع برچسب خوشه‌ای مانند ℓ که جواب بهینه‌ای برای مساله‌ی رابطه‌ی* (۵.۴) *باشد، جواب بهینه‌ای برای تابع هزینه‌ی ماتریسی کاهش‌یافته زیر نیز خواهد بود:*

$$\text{cost}_{red\_mat} = \min_{\ell} \left[\operatorname{tr}(T) - \operatorname{tr}\left(L^T A_G L\right) \right]. \tag{۶.۴}$$

اثبات. از مقایسه‌ی روابط (۵.۴) و (۶.۴)، می‌توان مشاهده نمود که $\text{cost}_{mat} = \text{cost}_{red\_mat} + m$، که m یک مقدار ثابت و برابر با تعداد یال‌های گراف مشابهت است. بنابراین، نتیجه به سادگی حاصل می‌شود. □

در [۱] نشان داده‌شده است که مساله‌ی خوشه‌بندی همبستگی، یک مساله‌ی $\mathbb{NP}$ـسخت است. بنابراین یافتن پاسخی بهینه برای توابع هزینه‌ی روابط (۱.۲)، (۵.۴) و (۶.۴) در زمان چندجمله‌ای بسیار بعید به‌نظر می‌رسد. در نتیجه لازم است تا از رویکردهای دیگری به منظور محاسبه‌ی یک جواب قابل قبول استفاده نمود. الگوریتم پیشنهادی در بخش ۳، الگوریتمی تقریبی است که در ادامه، با استفاده از بیان ماتریسی، می‌تواند دارای درجه‌ی بالایی از توازی در عمل برخوردار شود.

۵. عملیات انتقال راس بین خوشه‌ها

یک پیاده‌سازی بدیهی از الگوریتم پیشنهادی (بخش ۳) فاقد کارآیی زمانی خواهد بود، زیرا در هر تکرار نیازمند محاسبه‌ی تابع هزینه‌ی کاهش یافته برای هر انتقال ممکن بین تمام خوشه‌ها است. کلید محاسبه‌ی بهینه‌ی مقصد یک راس به منظور انجام عملیات انتقال، در رابطه‌ی (۱۷.۵) نهفته است. از یک سو، هدف الگوریتم پیشنهادی کاهش مقدار تابع هزینه‌ی ماتریسی کاهش یافته است. بنابراین، محاسبه‌ی میزان تغییرات در تابع هزینه قبل و بعد از انتقال و بدون نیاز به محاسبه‌ی کل مقدار تابع هزینه با استفاده از این رابطه قابل انجام خواهد بود. از سوی دیگر، با بررسی این رابطه، می‌توان دریافت که با آغاز از تابع همانی به عنوان تابع برچسب، تابع هزینه تنها در صورتی کاهش پیدا می‌کند که مقصد انتقال برای راس

iـام، یکی از خوشه‌های همسایه‌های راس iـام در گراف G باشد. به عبارت دیگر، می‌توان در این رابطه، مقصد را تنها به خوشه‌های همسایه‌های راس iـام در گراف G محدود نمود.

فرض کنید قصد داریم تا راس $v_i \in V$ را از خوشه‌ی c که در آن قرار دارد، به خوشه‌ی c' انتقال دهیم، به عبارت دیگر $c \xrightarrow{i} c'$. از تعریف tr، اثر ماتریس $L^T A_G L$ را می‌توان به‌صورت زیر نوشت:

$$\operatorname{tr}\left(L^T A_G L\right) = \sum_{c=1}^{C} \ell_c^T A_G \ell_c. \tag{۱.۵}$$

فرض کنید راس v_i متعلق به خوشه‌ی c باشد. در این صورت، بردار خوشه‌ی ℓ_c را می‌توان به صورت حاصل‌جمع

$$\ell_c = \ell_{c\setminus\{i\}} + e_i, \tag{۲.۵}$$

نوشت که e_i برداری از اندازه‌ی متناسب با سایر عملوندها، در اینجا $|V|$ درایه، است که تمام درایه‌های آن غیر از درایه‌ی iـام، صفر بوده و تنها درایه‌ی iـام برابر با یک است. همچنین، منظور از $\ell_{c\setminus\{i\}}$، بردار برچسب متناظر با خوشه‌ی $c \setminus \{i\}$ است. به عبارت دیگر، بردار $\ell_{c\setminus\{i\}}$، همان بردار ℓ_c است که در آن درایه‌ی iـام صفر شده است. در نتیجه داریم

$$\begin{aligned}
\ell_c^T A_G \ell_c &= \left(\ell_{c\setminus\{i\}} + e_i\right)^T A_G \left(\ell_{c\setminus\{i\}} + e_i\right) \\
&= \left(\ell_c - e_i + e_i\right)^T A_G \left(\ell_c - e_i + e_i\right) \\
&= \left(\ell_c - e_i\right)^T A_G \left(\ell_c - e_i\right) \\
&\quad + \underbrace{e_i^T A_G \left(\ell_c - e_i\right) + \left(\ell_c - e_i\right)^T A_G e_i + e_i^T A_G e_i}_{\text{سهم راس } i\text{ـام در خوشه‌ی } c}.
\end{aligned} \tag{۳.۵}$$

از رابطه‌ی (۳.۵) می‌توان مشاهده کرد که سطر چهارم بیانگر سهم راس iـام در خوشه‌ی c است. حال فرض کنید راس iـام به خوشه‌ی c' اضافه شود. در این صورت، داریم:

$$\ell_{c'\cup\{i\}} = \ell_{c'} + e_i, \tag{۴.۵}$$

که در آن، $\ell_{c'\cup\{i\}}$ برابر با بردار متناظر با خوشه‌ی c' است که در آن درایه‌ی iـام دارای مقدار یک است. به عبارت دیگر،

$\ell_{c'\cup\{i\}}$ متناظر با بردار خوشه‌ی $c' \cup \{i\}$ است. با طی رویکردی مشابه با رابطه‌ی (۳.۵)، خواهیم داشت:

$$\begin{aligned}\ell_{c'\cup\{i\}}^T A_G \ell_{c'\cup\{i\}} &= (\ell_{c'} + e_i)^T A_G (\ell_{c'} + e_i) \\ &= \ell_{c'}^T A_G \ell_{c'} \\ &\quad + \underbrace{e_i^T A_G \ell_{c'} + \ell_{c'}^T A_G e_i + e_i^T A_G e_i}_{\text{سهم راس } i\text{ـام در خوشه‌ی } c'} .\end{aligned} \tag{۵.۵}$$

بنابراین، عملیات $c \xrightarrow{i} c'$ منجر به کسر سهم راس iـام از خوشه‌ی c (سطر چهارم در رابطه‌ی (۳.۵)) و اضافه‌شدن سهم راس iـام به خوشه‌ی c' (سطر سوم در رابطه‌ی (۵.۵)) خواهد شد.

لم ۱.۵. *میزان تغییرات در تابع هزینه‌ی ماتریسی کاهش‌یافته، رابطه‌ی (۶.۴)، به ازای تغییر سهم راس iـام از خوشه‌ی c به خوشه‌ی c' عبارت است از:*

$$-e_i^T \left(A_G + A_G^T\right)(\ell_{c'} - \ell_c + e_i) . \tag{۶.۵}$$

اثبات. از تجمیع روابط (۳.۵) و (۵.۵)، داریم

$$\begin{aligned}&+ e_i^T A_G \ell_{c'} + \ell_{c'}^T A_G e_i + e_i^T A_G e_i \\ &- e_i^T A_G (\ell_c - e_i) - (\ell_c - e_i)^T A_G e_i - e_i^T A_G e_i \\ &= e_i^T A_G (\ell_{c'} - \ell_c + e_i) + (\ell_{c'} - \ell_c + e_i)^T A_G e_i \\ &= e_i^T A_G (\ell_{c'} - \ell_c + e_i) + e_i^T A_G^T (\ell_{c'} - \ell_c + e_i) \\ &= e_i^T \left(A_G + A_G^T\right)(\ell_{c'} - \ell_c + e_i) .\end{aligned}$$

دلیل وجود علامت قرینه در رابطه‌ی (۶.۵)، وجود عملگر تفریق در رابطه‌ی (۶.۴) قبل از $\operatorname{tr}\left(L^T A_G L\right)$ است. □

نتیجه‌ی بعد مربوط به محاسبه‌ی رابطه‌ی (۶.۵) به ازای تمام خوشه‌های ممکن با استفاده از عملیات ماتریسی است. نکته‌ی قابل ذکر این که در این محاسبات، تنها به ستون‌ها، درایه‌ها و یا سطرهایی توجه می‌کنیم که برای انتقال مجاز هستند. به عبارت دیگر، درایه‌های متناظر با خوشه‌ی جاری راس iـام یا خوشه‌هایی که راس iـام در آن‌ها همسایه‌ای ندارد، مورد نظر

نیستند.

نتیجه ۲.۵. *میزان تغییرات در تابع هزینه‌ی ماتریسی کاهش‌یافته، رابطه‌ی (۶.۴)، به ازای تغییر سهم راس iـام از خوشه‌ی c به تمام خوشه‌های مجاز مانند c' در دراید‌های متناظر با آن‌ها آمده است:*

$$-e_i^T \left(A_G + A_G^T\right) \hat{L}, \tag{۷.۵}$$

که ماتریس $\hat{L}$ برابر است با

$$\hat{L} = L - (\ell_c + e_i)\, \mathbb{1}_K^T. \tag{۸.۵}$$

بدیهی است منظور از $\mathbb{1}_K$ برداری ستونی شامل K درایه با مقدار یک است.

اثبات. اثبات، نتیجه‌ی مستقیم از تعریف ماتریس $\hat{L}$ و لم ۱.۵ است. □

از طرف دیگر، حذف‌شدن راس iـام از خوشه‌ی c، منجر به کاهش تعداد یال‌های مورد انتظار در خوشه‌ی c خواهد شد. همچنین اضافه‌شدن این راس به خوشه‌ی c' منجر به افزایش تعداد یال‌های مورد انتظار خوشه‌ی c' می‌شود. با استفاده از تجزیه‌ی رابطه‌ی (۲.۵) و متعامدبودن بردارهای ℓ_c، می‌توان مشارکت راس iـام در خوشه‌ی c را به صورت رابطه‌ی

$$\begin{aligned} \ell_{c\setminus\{i\}}^T \ell_{c\setminus\{i\}} &= (\ell_c - e_i)^T (\ell_c - e_i) \\ &= \ell_c^T \ell_c \underbrace{-2\ell_c^T e_i + e_i^T e_i}_{\text{مشارکت راس } i\text{ـام}}, \end{aligned} \tag{۹.۵}$$

و میزان مشارکت راس iـام در خوشه‌ی c' را به صورت

$$\begin{aligned} \ell_{c'\cup\{i\}}^T \ell_{c'\cup\{i\}} &= (\ell_{c'} + e_i)^T (\ell_{c'} + e_i) \\ &= \ell_{c'}^T \ell_{c'} \underbrace{+2\ell_{c'}^T e_i + e_i^T e_i}_{\text{مشارکت راس } i\text{ـام}}, \end{aligned} \tag{۱۰.۵}$$

تفکیک نمود. با استفاده از تعریف $T = C(C \ominus \mathbb{1})$، قطری‌بودن ماتریس T و عملگر tr، می‌توان میزان مشارکت راس

iـام در تعداد یال‌های مورد انتظار خوشه‌ی c را به صورت عمیات ماتریسی تعریف نمود. اگر ماتریس T قبل و بعد از عمل انتقال را به ترتیب با T^{old} و T^{new} نمایش دهیم، آنگاه داریم:

$$T^{\text{new}}_{c,c} = T^{\text{old}}_{c,c} - ۴\ell_c^T \ell_c \ell_c^T e_i - ۴\left(\ell_c^T e_i\right)^۲ - ۲\ell_c^T e_i + ۲\ell_c^T \ell_c, \tag{۱۱.۵}$$

که می‌تواند با توجه به عضویت راس iـام در خوشه‌ی c و $\ell_c^T e_i = ۱$ به صورت زیر ساده شود:

$$T^{\text{new}}_{c,c} = T^{\text{old}}_{c,c} - ۲\ell_c^T \ell_c - ۶. \tag{۱۲.۵}$$

همچنین، برای خوشه‌ی c' داریم:

$$T^{\text{new}}_{c',c'} = T^{\text{old}}_{c',c'} + ۴\ell_{c'}^T \ell_{c'} \ell_{c'}^T e_i + ۴\left(\ell_{c'}^T e_i\right)^۲ + ۲\ell_{c'}^T e_i + ۲\ell_{c'}^T \ell_{c'}. \tag{۱۳.۵}$$

با توجه به تعریف ماتریس L و عدم عضویت راس iـام به خوشه‌ی c'، داریم $\ell_{c'}^T e_i = ۰$. بنابراین، رابطه‌ی (۱۳.۵) را می‌توان به صورت زیر ساده کرد:

$$T^{\text{new}}_{c',c'} = T^{\text{old}}_{c',c'} + ۲\ell_{c'}^T \ell_{c'}. \tag{۱۴.۵}$$

لم ۳.۵. *میزان تغییرات در* $\operatorname{tr}(T)$ *به ازای عمل انتقال راس* iـام *از خوشه‌ی* c *به خوشه‌ی* c' *برابر است با*

$$۲\left(\ell_{c'}^T \ell_{c'} - \ell_c^T \ell_c - ۳\right). \tag{۱۵.۵}$$

اثبات. از جمع روابط (۱۲.۵) و (۱۴.۵)، نتیجه‌ی مورد نظر حاصل می‌شود. □

نتیجه ۴.۵. *میزان تغییرات در* $\operatorname{tr}(T)$ *به ازای عمل انتقال راس* iـام *از خوشه‌ی* c *به تمام خوشه‌های مجاز در قطر اصلی ماتریس زیر قرار دارد:*

$$۲\left(L^T L - \left(\ell_c^T \ell_c + ۳\right)\mathbf{I}_{K\times K}\right), \tag{۱۶.۵}$$

که منظور از $\mathbf{I}_{K\times K}$ *ماتریس همانی از ابعاد* $K \times K$ *است.*

اثبات. با استفاده از لم ۳.۵ و قطری‌بودن ماتریس C (لم ۱.۴)، نتیجه حاصل می‌شود. □

قضیه ۵.۵. *میزان تغییرات در تابع هزینه‌ی ماتریسی کاهش یافته‌ی بر اثر عمل انتقال* $c \xrightarrow{i} c'$ *عبارت است از*

$$\delta_{c\xrightarrow{i}c'}(\ell) = ۲\left(\ell_{c'}^T\ell_{c'} - \ell_c^T\ell_c - ۳\right) - e_i^T\left(A_G + A_G^T\right)(\ell_{c'} - \ell_c + e_i)\,. \qquad (۱۷.۵)$$

اثبات. اثبات از جایگذاری روابط (۶.۵) (در لم ۱.۵) و (۱۵.۵) (در لم ۳.۵) به‌ترتیب به‌جای $\mathrm{tr}(T)$ و $\mathrm{tr}\left(L^T A_G L\right)$ در تابع هزینه‌ی کاهش‌یافته، رابطه‌ی (۱۷.۵) حاصل می‌شود. □

نتیجه ۶.۵. *میزان تغییرات در تابع هزینه‌ی ماتریسی کاهش یافته‌ی بر اثر عمل انتقال راس* i*ـام از خوشه‌ی* c *به تمام خوشه‌های مجاز در درایه‌های متناظر از بردار زیر آمده است:*

$$\delta_i(\ell) = diag\left(۲\left(L^TL - \left(\ell_c^T\ell_c + ۳\right)\mathbf{I}_{C\times C}\right)\right) - e_i^T\left(A_G + A_G^T\right)\hat{L}, \qquad (۱۸.۵)$$

که ماتریس $\hat{L}$ *در رابطه‌ی* (۸.۵) *تعریف شده است و منظور از* $diag(A)$*، برداری است که حاوی درایه‌های قطری ماتریس* A *است.*

اثبات. اثبات از جایگذاری روابط (۷.۵) (نتیجه‌ی ۲.۵) و (۱۶.۵) (نتیجه‌ی ۴.۵) به‌ترتیب به‌جای $\mathrm{tr}(T)$ و $\mathrm{tr}\left(L^T A_G L\right)$ در تابع هزینه‌ی کاهش‌یافته، رابطه‌ی (۱۷.۵) حاصل می‌شود. □

برای اینکه بتوان عملیات انتقال را انجام داد، لازم است تا $\delta_{c\xrightarrow{i}c'}(\ell) < ۰$ باشد. از سوی دیگر، تمایل داریم تا هر چه سریع‌تر به همگرایی و پاسخ کمینه برسیم. بدین منظور، راسی را برای انتقال انتخاب می‌کنیم که بیشترین کاهش را در تابع هزینه ایجاد نماید. به عبارت دیگر، راس iـام از خوشه‌ی جاری $\ell(v_i)$ به خوشه‌ی c' انتقال داده می‌شود که $\min_{v_i\in V, c'\neq\ell(v_i)} \delta_{\ell(v_i)\xrightarrow{i}c'}(\ell)$ باشد.

بردار $\delta_i(\ell)$ را با C درایه به صورت

$$\delta_i^T(\ell) = \left(\delta_{c\xrightarrow{i}۱}(\ell), \ldots, \delta_{c\xrightarrow{i}C}(\ell)\right), \qquad (۱۹.۵)$$

تعریف می‌کنیم.

نکته ۷.۵. لازم به ذکر است که بردار تعریف شده در رابطه‌ی (۱۹.۵)، معادل درایه‌های قطری ماتریس رابطه‌ی (۱۸.۵) است.

همانگونه که در ابتدای بخش ۵ مطرح شد، می‌توان مقصد را تنها به خوشه‌های همسایه‌های راس iـام در گراف G محدود نمود. بنابراین، می‌توان از یک بردار نقاب دودویی برای محدودکردن محاسبات استفاده نمود:

$$\gamma_i(\ell) = \left(e_i^T A_G L \neq 0\right). \tag{۲۰.۵}$$

بنابراین، قرار می‌دهیم:

$$\lambda_i(\ell) = \gamma_i(\ell) \circ \delta_i(\ell), \tag{۲۱.۵}$$

که منظور از $\circ$، عملیات ضرب درایه‌های نظیر به نظیر بین دو بردار است. بنابراین، خوشه‌ی مقصد برای راس i برابر با خوشه‌ی متناظر با کوچکترین مقدار در $Q_i(\ell)$ است. به عبارت دیگر، اگر قرار دهیم: $x = \min \lambda_i(\ell)$، آنگاه می‌توان خوشه‌ی مقصد برای راس iـام را به صورت بردار دودویی $\sigma_i = (\lambda_i(\ell) = x)$ نشان داد.

نکته ۸.۵. لازم به ذکر است که از σ_i علاوه بر اشاره به بردار دودویی خوشه‌ی مقصد، به منظور اشاره به اندیس خوشه‌ی مقصد با کمترین مقدار تغییر در تابع هزینه نیز استفاده می‌کنیم.

پس از انجام انتقال، لازم است تا نگاشت برچسب و ماتریس متناظر با آن نیز به‌روزرسانی شوند:

$$L^{\text{new}} = L^{\text{old}} + \sigma_i \sigma_i^T - e_i e_i^T L^{\text{old}}. \tag{۲۲.۵}$$

فرایند انتقال راس‌ها تا جایی انجام می‌شود که انتقال دیگری ممکن نباشد. به عبارت دیگر، برای هر راسی مانند i و برای هر خوشه‌ی مقصدی مانند c'، داشته باشیم:

$$\delta_{\ell(v_i) \xrightarrow{i} c'}(\ell) \geq 0. \tag{۲۳.۵}$$

به‌صورت معادل، می‌توان شرط توقف را برابر با

$$\forall v_i \in V, \delta_{\ell(v_i) \xrightarrow{i} \sigma_i}(\ell) \geq 0, \tag{۲۴.۵}$$

نیز قرار داد.

۶. پیاده‌سازی ماتریسی از روش پیشنهادی

الگوریتم پیشنهادی، الگوریتم ۳ را می‌توان با استفاده از سازوکار ارائه شده در بخش ۵ با استفاده از عملیات ماتریسی پیاده‌سازی نمود. این پیاده‌سازی در الگوریتم ۶ آمده است.

قضیه ۱.۶. *الگوریتم ۶ با الگوریتم ۳ معادل است.*

اثبات. خط چهارم در الگوریتم ۲، معادل با خط اول در الگوریتم ۱ است که نگاشت همانی به عنوان تابع برچسب اولیه لحاظ می‌شود. خطوط ۳ الی ۵ در الگوریتم ۱، معادل با خط ۱۲ هستند. در این خط، هر راس به صورت تصادفی و برای یک بار مورد پردازش قرار می‌گیرد. شرط توقف در خط ۹ الگوریتم ۲ آمده است. این شرط معادل با شرط خروج از حلقه در الگوریتم ۱ در خطوط ۱۲ الی ۱۳ است. عملیات یافتن خوشه‌ی مقصد برای راس iـام که دارای بیشترین کاهش در تابع هزینه باشد، خط ۷ در الگوریتم ۱، در خطوط ۱۳ الی ۲۹ الگوریتم ۲ آمده است. ابتدا، با استفاده از رابطه‌ی (۱۸.۵)، مقدار کاهش در تابع هزینه محاسبه می‌شود. سپس، در خطوط ۲۴ الی ۲۹، خوشه‌های مجاز برای انتقال پالایش شده و از بین آن‌ها، خوشه‌ای با بیشترین کاهش در تابع هزینه انتخاب می‌شود. عملیات انتقال در صورتی قابل انجام است که میزان کاهش یک عدد منفی بوده و خوشه‌ی مقصد با خوشه‌ی مبدا متفاوت باشند. این شرط در الگوریتم ۱ در خط ۸ و در الگوریتم ۲ در خط ۳۱ آمده است. در صورتی که عملیات انتقال انجام شود، نیاز است تا ماتریس برچسب‌گذاری به روزرسانی شود. این عملیات در خط ۳۲ الگوریتم ۲، معادل با خطوط ۹ و ۱۱ در الگوریتم ۱ انجام می‌شود. سایر خطوط الگوریتم ۲، مربوط به آماده‌سازی برخی بردارها و ماتریس‌های ضروری جهت انجام بهینه‌ی محاسبات است. □

قضیه ۲.۶. *پیچیدگی زمانی هر تکرار از الگوریتم ۶، بدنه‌ی حلقه‌ی تکرار در خط ۹، با فرض وجود K خوشه برای N راس از مرتبه‌ی $O\left(N\left(N^{۲}\left(K^{۲}+K\right)+N\left(K+۱\right)+K^{۲}\right)\right)$ است.*

اثبات. در خط ۲۱، جهت محاسبه‌ی رابطه‌ی (۸.۵) نیازمند انجام یک ضرب خارجی بین دو بردار از اندازه‌های $K \times ۱$ و $N \times ۱$، یک عملیات جمع بردارهای $N \times ۱$ و محاسبه‌ی تفاضل بین دو ماتریس $N \times K$ هستیم. سپس، در خط ۲۲، جهت محاسبه‌ی رابطه‌ی (۷.۵) نیازمند یک ضرب ماتریس در ماتریس از ابعاد $N \times N$ و $N \times K$ و سپس یک ضرب ماتریس در بردار از ابعاد $N \times K$ و $N \times ۱$ هستیم. لازم به ذکر است که ماتریس $A_G + A_G^T$ در خط ۵ برای یک بار محاسبه شده و دیگر نیازی به عملیات جمع ماتریسی برای محاسبه‌ی آن نیست. رابطه‌ی (۱۶.۵) در خط ۲۳ با استفاده از یک ضرب ماتریس در ماتریس $K \times N$ و $N \times K$ و تفاضل بین دو ماتریس $K \times K$ انجام می‌پذیرد. خط ۲۵ به

محاسبه‌ی رابطه‌ی (۲۰.۵) با استفاده از یک عملیات ضرب ماتریس در ماتریس از ابعاد $N \times N$ و $N \times K$ و یک ضرب ماتریس در بردار $N \times K$ و $N \times ۱$ اختصاص دارد. در خط ۲۶، عملیات ضرب هادامارد بین دو بردار از ابعاد $K \times ۱$ انجام می‌پذیرد تا رابطه‌ی (۲۱.۵) محاسبه شود. در خط ۲۷، کوچکترین مقدار تفاضل محاسبه می‌شود که در زمان $O(K)$ قابل انجام است. در نهایت، در خط ۳۲، عملیات به‌روزرسانی تابع برچسب با استفاده از دو عملیات ضرب خارجی بردارهای از اندازه‌ی $K \times ۱$ و سپس، دو عملیات جمع ماتریسی از اندازه‌های $K \times K$ انجام می‌شود. در مجموع، هر تکرار شامل یک گام به ازای هر راس در گراف اصلی است و در هر گام،

$$O\left(N^{۲}\left(K^{۲}+K\right)+N(K+۱)+K^{۲}\right), \tag{۱.۶}$$

تعداد عملیات انجام می‌شود. بنابراین، از ضرب رابطه‌ی (۱.۶) در تعداد رئوس N، نتیجه‌ی دلخواه حاصل می‌شود. □

```
MAX_ITER = 10 # maximum number of iterations
CHG_THR = -0.5 # change threshold
N = |V|
L = np.eye(N) # NxN identity matrix
A_p_AT = A + A.T
v_changed = True
iteration = 0
iter_change = 0
while v_changed and iteration < MAX_ITER and iter_change
    < CHG_THR:
  v_changed = False
  iter_change = 0
  for i in np.random.permutation(range(N)):
    # initial setup
    K = L.shape[1] # number of clusters
    c = np.nonzero(L[i, :])[0] # c = l(i)
    l_c = L[:, c]
    e_i = np.zeros_like(l_c)
    e_i[i] = 1
    ones = np.ones_like(l_c)
    # Changes in the reduced cost function
    L_hat = L - (l_c - e_i) @ ones.T # Eq. (5.8)
    delta_i_l = - e_i.T @ A_p_AT @ L_hat # Eq. (5.7)
    delta_i_l += 2 * np.diag(L.T @ L - (l_c.T @ l_c - 3)
        * np.eye(K)) # Eq. (5.16)
    # Finding the destination cluster
    gamma_i_l = (e_i.T @ A @ L) != 0 # Eq. (6.2)
    delta_i_l *= gamma_i_l # Eq. (6.3)
    sigma_i = np.argmin(delta_i_l) # Remark 6.2
    sigma_i_vect = np.zeros_like(delta_i_l, dtype=int)
    sigma_i_vect[sigma_i] = 1 # Remark 6.2
    # Can the transportation be done?
    if sigma_i != c and delta_i_l[sigma_i] < 0:
      L += (sigma_i_vect @ sigma_i_vect.T - e_i @ e_i.T)
          # Eq. (6.4)
      iter_change += delta_i_l[sigma_i]
      v_changed = True
  Remove all zero columns of L
  iteration += 1

# L is the output label
```

[الگوریتم ۲]

در مقام مقایسه، مشاهده می‌شود که پیچیدگی زمانی الگوریتم ۶ نسبت به پیچیدگی زمانی الگوریتم ۳، از نظر مجانبی به اندازه‌ی $O\left(N^{۲}\right)$ بیشتر است. دلیل این تفاوت، استفاده از عملیات پایه‌ی ماتریسی، ضرب ماتریس در بردار، در پیاده‌سازی الگوریتم است. مابه‌ازای این افزایش در پیچیدگی زمانی، بیان الگوریتم بر پایه‌ی عملیات ماتریسی، امکان استفاده از توازی محاسباتی در هر کلاسترمحاسباتی پشتیبانی کننده از عملیات‌های پایه ماتریسی را فراهم می‌نماید [۷].

۷. نتیجه‌گیری

در این مقاله، ابتدا یک الگوریتم تقریبی با استفاده از رویکردی حریصانه به منظور حل مساله‌ی خوشه‌بندی همبستگی ارائه شد. سپس، با وارسی دقیق‌تر عملیات انتقال راس از یک خوشه به خوشه‌ی دیگر، فرمول‌بندی معادلی بر پایه‌ی عملیات ماتریسی برای الگوریتم پیشنهادی ارائه شد. با استفاده از این فرمول‌بندی، الگوریتم پیشنهادی در قالب عملیات‌های پایه‌ی جبرخطی پیاده‌سازی و پیچیدگی محاسباتی آن مورد تحلیل و بررسی قرار گرفت. یکی از مهم‌ترین مزیت‌های این فرمول‌بندی، امکان پیاده‌سازی الگوریتم پیشنهادی در قالب هر زبان و چارچوب برنامه‌نویسی است که از عملیات‌های پایه‌ی جبرخطی پشتیبانی می‌کند. در کارهای آتی می‌توان به بهبود زمان اجرای الگوریتم پیشنهادی با استفاده از الگوریتم‌های ضرب ماتریسی سریع‌تر پرداخت. همچنین، بسط فرمول‌بندی ماتریسی به الگوریتم‌های موجود، امکان بهره‌گیری از توازی عملیات‌های پایه‌ی ماتریسی را فراهم می‌نماید.

مراجع

[1] Nikhil Bansal, Avrim Blum, and Shuchi Chawla. Correlation clustering. *Machine learning*,**56**(1)(2004),89–113.

[2] L Susan Blackford, Antoine Petitet, Roldan Pozo, Karin Remington, R Clint Whaley, James Demmel, Jack Dongarra, Iain Duff, Sven Hammarling, Greg Henry, et al. An updated set of basic linear algebra subprograms (BLAS). *ACM Transactions on Mathematical Software***28**(2)(2002),135–151.

[3] Francesco Bonchi, Aristides Gionis, and Antti Ukkonen. Overlapping correlation clustering. *Knowledge and Information Systems*, **35**(1)(2013),1–32.

[4] Shuchi Chawla, Konstantin Makarychev, Tselil Schramm, and Grigory Yaroslavtsev. Near optimal LP rounding algorithm for correlationclustering on complete and complete k-partite graphs. In *Proceedings of the forty-seventh annual ACM symposium on Theory of computing*, (2019) 219–228.

[5] Yudong Chen, Sujay Sanghavi, and Huan Xu. Clustering sparse graphs. *Advances in neural information processing systems*, **25**(2012).

[6] Vincent Cohen-Addad, Silvio Lattanzi, Slobodan Mitrović, Ashkan Norouzi-Fard, Nikos Parotsidis, and Jakub Tarnawski. Correlation clustering in constant many parallel rounds. In *International Conference on Machine Learning*, 2069–2078. PMLR, 2021.

[7] Timothy A Davis. Algorithm 1000: SuiteSparse: GraphBLAS: Graph algorithms in the language of sparse linear algebra. *ACM Transactions on Mathematical Software (TOMS)*,**45**(4)(2019)1–25.

[8] Dmitri V Kalashnikov, Zhaoqi Chen, Sharad Mehrotra, and Rabia Nuray-Turan. Web people search via connection analysis. *IEEE Transactions on Knowledge and Data Engineering*, **20**(11)(2008),1550–1565.

[9] Ali Shakiba. Online correlation clustering for dynamic complete signed graphs. *arXiv preprint arXiv:2211.07000*, 2022.

[10] Ali Shakiba. Correlation clustering algorithm for dynamic complete signed graphs: An index-based approach. *arXiv preprint arXiv:2301.00384*, 2023.

[11] Vijay V Vazirani. *Approximation algorithms*, volume 1. Springer, 2001.