DOR: 20.1001.1.2538161.1402.8.3.6.2
DOI : 10.22034/CSJ.2023.184641

پاییز ۱۴۰۲، دوره ۸، شماره ۳، پیاپی ۳۰

دریافت مقاله: ۱۴۰۲/۰۶/۲۱
پذیرش مقاله: ۱۴۰۲/۰۸/۳۰

نوع مقاله: پژوهشی

یک الگوریتم تقریبی برای حل مسئله تابع احاطه‌گر ایتالیایی روی گراف‌ها

ابوالفضل پورعیدی *
دانشکده علوم ریاضی- دانشگاه صنعتی شاهرود- شاهرود- ایران
پست الکترونیکی: a.poureidi@shahroodut.ac.ir

چکیده

گراف $G=(V,E)$ را در نظر بگیرید. تابع $f:V\rightarrow\{0,1,2\}$ را یک تابع احاطه‌گر ایتالیایی (احاطه‌گر $\{2\}$-رومن) گویند هر گاه هر راس $v\in V$ با $f(v)=0$ مجاور به حداقل یک راس $u\in V$ با $f(u)=2$ یا مجاور به حداقل دو راس $x,y\in V$ با $f(x)=f(y)=1$ باشد. وزن یک تابع احاطه‌گر ایتالیایی برای گراف G با کمترین مقدار را عدد احاطه‌گر ایتالیایی گراف G گوییم. مسئله تابع احاطه‌گر ایتالیایی برای گراف G به صورت یافتن یک تابع احاطه‌گر ایتالیایی با وزن برابر با عدد احاطه‌گر ایتالیایی برای گراف G تعریف می‌شود. ثابت شده است که مسئله تابع احاطه‌گر ایتالیایی NP-کامل است. در این مقاله ابتدا یک مدل برنامه‌ریزی خطی صحیح برای این مسئله پیشنهاد می‌کنیم و سپس با استفاده از این مدل یک الگوریتم تقریبی با ضریب $H(2\Delta(G)+2)$ برای حل مسئله ارائه می‌کنیم.

واژه‌های کلیدی: الگوریتم تقریبی، مدل برنامه‌ریزی عددی خطی صحیح، تابع احاطه‌گر ایتالیایی.

۱. مقدمه

فرض کنید $G=(V,E)$ یک گراف ساده و بدون جهت با مجموعه راس‌های V و مجموعه یال‌های E است. همسایگی باز یک راس $v\in V$ مجموعه $N_G(v)=\{u\in V: uv\in E\}$ و همسایگی بسته یک راس $v\in V$ مجموعه $N_G[v]=N_G(v)\cup\{v\}$ است. درجه یک راس v در گراف G برابر با $|N_G(v)|$ و آن را با $\deg_G(v)$ نشان می‌دهیم. بیشترین درجه در بین درجه راس‌های گراف G را با $\Delta(G)$ نمایش می‌دهیم. اگر از متن گراف G مشخص باشد فقط از نماد Δ استفاده خواهیم کرد.

تابع $f:V\rightarrow\{0,1,2\}$ را یک تابع احاطه‌گر رومن روی گراف G گوییم اگر هر راس $v\in V$ با $f(v)=0$ مجاور به حداقل یک راس $u\in V$ با $f(u)=2$ باشد. یک تابع احاطه‌گر رومن f را به صورت $f=(V_0,V_1,V_2)$ نیز نشان می‌دهیم که برای هر $i=0,1,2$ داریم $V_i=\{v\in V: f(v)=i\}$. وزن تابع احاطه‌گر رومن f را با $w(f)=\sum_{v\in V}f(v)$ تعریف می‌کنیم. مفهوم تابع احاطه‌گر رومن روی یک گراف توسط استوارت [۱] و رول و روسینگ [۲] مطالعه شد و سپس توسط کوکین و همکاران [۳] تعمیم داده شد. انگیزه اصلی مطالعه مفهوم تابع احاطه‌گر رومن راهبرد امپراتور کنستانتین برای دفاع از امپراتوری روم در حدود قرن چهارم بعد از میلاد بود. او دستور داد در هر شهر امپراتوری باید حداکثر دو هنگ مستقر شود. حال اگر یک شهر دارای هنگی برای محافظت نبود باید مجاور به شهری با دو هنگ باشد که اگر مورد حمله قرار گرفت از دو هنگ شهر مجاور یکی برای دفاع شهر بدون هنگ حرکت کند و هنگ دوم از خود شهر مجاور دفاع کند.

* نویسندهٔ مسئول

مفهوم تابع احاطه‌گر و انواع آن روی گراف‌ها در بیش از صدها مقاله مورد مطالعه و بررسی قرار گرفته‌اند: تابع احاطه‌گر رومن مضاعف [٤]، تابع احاطه‌گر $\{2\}$–رومن [٥]، تابع احاطه‌گر رومن تام [٦]، تابع احاطه‌گر رومن علامت‌دار [٧] و تابع احاطه‌گر عمومی [٨]. برای آشنایی بیشتر با انواع توابع احاطه‌گر رومن به مقاله [٩] مراجعه شود.

چلالی و همکاران [٥] مفهوم تابع احاطه‌گر $\{2\}$–رومن را تعریف کردند. تابع احاطه‌گر $\{2\}$–رومن را همچنین تابع احاطه‌گر ایتالیایی نیز می‌نامند [١٠]. یک تابع احاطه‌گر رومن $f: V \rightarrow \{0,1,2\}$ را یک تابع احاطه‌گر ایتالیایی روی گراف G گوییم اگر هر راس $v \in V$ با $f(v) = 0$ مجاور به حداقل یک راس $u \in V$ با $f(u) = 2$ یا مجاور به حداقل دو راس $x, y \in V$ با $f(x) = f(y) = 1$ باشد. وزن یک تابع احاطه‌گر ایتالیایی برای گراف G با کمترین مقدار را عدد احاطه‌گر ایتالیایی گراف G گوییم و با نماد $\gamma_{\{R2\}}(G)$ نشان می‌دهیم. مسئله تابع احاطه‌گر ایتالیایی برای گراف G به صورت یافتن یک تابع احاطه‌گر ایتالیایی با وزن برابر با عدد احاطه‌گر ایتالیایی برای گراف داده شده تعریف می‌شود. ثابت شده است که مسئله تابع احاطه‌گر ایتالیایی NP– سخت است حتی برای گراف‌های دوبخشی [٥] و گراف‌های مسطح [١١]. از آنجایی که مسئله تابع احاطه‌گر ایتالیایی روی گراف‌ها NP–سخت است، طراحی الگوریتم‌های تقریبی برای حل آن پیشنهاد می‌شود. در سال ۲۰۲۰ میلادی، پادامودان و همکاران [١٢] یک الگوریتم تقریبی برای مسئله تابع احاطه‌گر ایتالیایی روی گراف G با ضریب حداکثر $2(\ln(\Delta + 1) + 1)$ ارائه کردند. در این مقاله یک الگوریتم حریصانه برای مسئله تابع احاطه‌گر ایتالیایی روی گراف‌ها با ضریب $H(2\Delta(G) + 2)$ ارائه می‌دهیم به طوری که ضریب تقریب الگوریتم پیشنهادی در [١٢] را بهبود می‌دهد.

با توجه به این که بیشتر افراد معتقدند که P ≠ NP، آنگاه برای حل یک مسئله NP–سخت نباید دنبال یک الگوریتم چندجمله‌ای باشیم. به نظر می‌رسد الگوریتم‌های تقریبی یک راه‌حل موفق برای حل مسائل بهینه‌سازی سخت است. در طراحی الگوریتم‌های تقریبی برای حل یک مسئله سخت از پیچیدگی نمایی به پیچیدگی چندجمله‌ای عبور می‌کنیم و در عوض، جواب دقیق برای مسئله سخت را با جوابی جایگزین می‌کنیم که هزینه جواب تقریبی با جواب بهینه حداکثر ε % برای یک مقدار $\varepsilon > 0$ داده شده متفاوت است. این کارآیی برای مسایل سختی که کاربردهای مهمی دارند خیلی مورد توجه است. برای اطلاعات بیشتر در مورد الگوریتم‌های تقریبی به کتاب‌های [١٣–١٤] مراجعه کنید.

ساختار این مقاله به صورت زیر است. در بخش ۲، ابتدا یک مدل برنامه‌ریزی خطی صحیح برای مسئله تابع احاطه‌گر ایتالیایی روی گراف‌ها پیشنهاد می‌دهیم و سپس یک مدل برنامه‌ریزی خطی صحیح جایگزین برای مدل اولیه ارائه می‌کنیم. در بخش ۳ با استفاده از مدل برنامه‌ریزی خطی صحیح جایگزین پیشنهادی بخش ۲ یک الگوریتم حریصانه تقریبی برای حل مسئله تابع احاطه‌گر ایتالیایی روی گراف‌ها ارائه می‌کنیم.

۲. یک مدل برنامه‌ریزی خطی صحیح

در این بخش، یک مدل برنامه‌ریزی خطی صحیح (ILP) برای مسئله تابع احاطه‌گر ایتالیایی روی گراف‌ها ارائه می‌کنیم. سپس یک مدل ILP جایگزین برای مسئله تابع احاطه‌گر ایتالیایی روی گراف‌ها پیشنهاد می‌کنیم و نشان می‌دهیم که این دو فرمول‌بندی برای مسئله تابع احاطه‌گر ایتالیایی روی گراف‌ها معادل هستند. گراف $G = (V, E)$ را در نظر بگیرید و فرض کنید $f: V \rightarrow \{0,1,2\}$ یک تابع احاطه‌گر ایتالیایی روی G باشد. دو متغیر x_v و y_v را برای هر راس $v \in V$ به صورت زیر تعریف می‌کنیم.

$$x_v = \begin{cases} 1 & if \quad f(v) = 1 \\ 0 & if \quad f(v) \neq 1 \end{cases} \quad y_v = \begin{cases} 1 & if \quad f(v) = 2 \\ 0 & if \quad f(v) \neq 2 \end{cases}$$

فرمول‌بندی IDF1 برای مسئله تابع احاطه‌گر ایتالیایی روی گراف G را به صورت تعریف می‌کنیم.

$$\text{IDF1:}\ \min \sum_{v\in V} x_v + 2\sum_{v\in V} y_v \quad (۱)$$

$$s.t.\quad x_v + y_v + \frac{1}{2}\sum_{u\in N(v)} x_u + \sum_{u\in N(v)} y_u \geq 1 \quad \forall v\in V \quad (۲)$$

$$x_v + y_v \leq 1 \quad \forall v\in V \quad (۳)$$

$$x_v, y_v \in \{0,1\} \quad \forall v\in V \quad (۴)$$

مقدار تابع هدف با قید (۱) داده شـــده است. مقدار تابع هدف همان وزن گراف است که مجموع مقادیر نسبت داده شـــده به راس‌های گراف است. مقادیر نسبت داده شده به راس‌های گـــراف از مجموعه $\{0,1,2\}$ انتخاب می‌شـــوند. اگر مقـــدار ۱ را به یـــک راس مانند v نســـبت دهیم داریم $x_v = 1$ و اگر مقدار ۲ را به یک راس مانند v نســـبت دهیم داریـــم $y_v = 1$ و در غیر این صـــورت برای راس v داریم $x_v = y_v = 0$ در نتیجـــه مقـــدار تابع هـــدف به صورت $\sum_{v\in V} x_v + 2\sum_{v\in V} y_v$ تعریف می‌شود. برای هر راس $v\in V$، قید (۲) اطمینان می‌دهند اگر $f(v) = 0$، آنگاه حداقل یک راس u مجـــاور به v با $f(u) = 2$ یا دو راس x, y مجاور به v با $f(x) = f(y) = 1$ وجود دارد. واضح است برای هر راس $v\in V$ با $f(v) \geq 1$ قید (۲) ارضا می‌شود. برای هر راس $v\in V$، قید (۳) اطمینان می‌دهد $f(v)$ حداکثر یکی از دو مقـــدار مجموعه $\{1,2\}$ را انتخاب می‌کند. با توجه به قیود (٤)، برای هـــر راس $v\in V$، متغیرهای تصمیم‌گیری x_v و y_v مقدارشان را از مجموعه $\{0,1\}$ انتخاب می‌کنند.

مثـــال ۱. گـــراف $G = (V,E)$ را در نظـــر بگیرید که $V = \{1,2,\dots,5\}$ و $E = \{12,23,25,34,45\}$. فرمولبندی IDF1 برای گراف G به صورت زیر است.

$$\text{Min}\quad x_1 + \dots + x_5 + 2y_1 + \dots + 2y_5$$

$$s.t.\quad x_1 + y_1 + \frac{1}{2}x_2 + y_2 \geq 1$$

$$x_2 + y_2 + \frac{1}{2}(x_1 + x_3 + x_5) + (y_1 + y_3 + y_5) \geq 1$$

$$x_3 + y_3 + \frac{1}{2}(x_2 + x_4) + (y_2 + y_4) \geq 1$$

$$x_4 + y_4 + \frac{1}{2}(x_3 + x_5) + (y_3 + y_5) \geq 1$$

$$x_5 + y_5 + \frac{1}{2}(x_2 + x_4) + (y_2 + y_4) \geq 1$$

$$x_i + y_i \leq 1 \quad \forall i \in \{1,\dots,5\}$$

$$x_i, y_i \in \{0,1\} \quad \forall i \in \{1,\dots,5\}$$

در ادامـــه یک فرمـــول بندی ILP جایگزیـــن IDF2 برای مسئله تابع احاطه‌گر ایتالیایی روی گراف GG ارائه می‌کنیم.

$$\text{IDF2:}\ \min \sum_{v\in V} x_v + 2\sum_{v\in V} y_v \quad (۵)$$

$$s.t.\quad y_v + \frac{1}{2}\sum_{u\in N(v)} x_u + \sum_{u\in N(v)} y_u \geq 1 \quad \forall v\in V \quad (۶)$$

$$x_v, y_v \in \{0,1\} \quad \forall v\in V \quad (۷)$$

در ادامه معادل بودن دو فرمول بندی IDF1 و IDF2 را ثابت می‌کنیم.

لـــم ۱. بـــرای گـــراف داده شـــده $G = (V,E)$ بـــا $V = \{1,\dots,n\}$، فرض کنید $z^* = (x^*, y^*) \in \{0,1\}^{2n}$ که $x^* = (x_1^*, \cdots, x_n^*)$ و $y^* = (y_1^*, \dots, y_n^*)$ یـــک جـــواب بهینـــه بـــرای فرمول‌بنـــدی IDF2 اســـت. آن‌گاه برای هر $i \in \{1,\dots,n\}$ داریم $x_i^* + y_i^* \leq 1$.

اثبـــات. با برهـــان خلف فـــرض کنید برای یـــک مقدار $k \in \{1,\dots,n\}$داریم$x_k^* + y_k^* > 1$.ازآنجایی‌که$x_k^*, y_k^* \in \{0,1\}$ داریـــم $x_k^* = y_k^* = 1$. فـــرض کنیـــد $V' = V \setminus \{k\}$ و $z^{**} = (x^{**}, y^{**}) = (x_1^{**}, \dots, x_n^{**}, y_1^{**}, \dots, y_n^{**})$ به طوری که $x_i^{**} = x_i^*$ اگر $i \in V'$ و در غیر این صورت (اگر $i = k$) $x_i^{**} = 0$ و $y_i^{**} = y_i^*$ برای هر $i \in V$.

از آنجایـــی که $z^* = (x^*, y^*)$ یک جـــواب بهینه برای فرمـــول بندی IDF2 اســـت، پـــس یک جواب شـــدنی برای فرمو x_i^* $x_i^{**} + y_i^{**} + \frac{1}{2}\sum_{j\in N(i)} x_j^{**} + \sum_{j\in N(i)} y_j^{**}$ و x_i^{**} y_i^{**} $+ y_i^* + \frac{1}{2}\sum_{j\in N(i)} x_j^* + \sum_{j\in N(i)} y_j^* \geq 1$

$x_i^{**} = x_i^*$، $y_i^{**} = y_i^*$ و $x_i^{**}, y_i^{**} \in \{0,1\}$ بـــرای هر $i \in V'$ به طوری که k مجاور به راس i نباشـــد، به عبارت

دیگر $k \notin N(i)$. اگر $i \in V'$ و $k \in N(i)$، از آنجایی که $y_k^{**} = 1$ بدست می‌آوریم که $\sum_{j \in N(i)} y_j^{**} \geq 1$ و در نتیجه $x_i^{**} + y_i^{**} + \frac{1}{2}\sum_{j \in N(i)} x_j^{**} + \sum_{j \in N(i)} y_j^{**} \geq 1$، $x_i^{**} = x_i^*$ و $y_i^{**} = y_i^*$ و $x_i^{**}, y_i^{**} \in \{0,1\}$. همچنین،

$$x_k^{**} + y_k^{**} + \frac{1}{2}\sum_{j \in N(k)} x_j^{**} + \sum_{j \in N(k)} y_j^{**}$$
$$= 1 + \frac{1}{2}\sum_{j \in N(i)} x_j^* + \sum_{j \in N(i)} y_j^* \geq 1$$

$x_k^{**} = 0$، $y_k^{**} = 1$ و $x_k^{**}, y_k^{**} \in \{0,1\}$. بنابراین، z^{**} یک جواب شدنی برای IDF2 نیز هست.

مقدار تابع هدف فرمول‌بندی IDF2 برای جواب شدنی z^{**} برابر با مقدار زیر خواهد بود.

$$\sum_{i \in V} x_i^{**} + 2\sum_{i \in V} y_i^{**} = \sum_{i \in V'} x_i^* + 2\sum_{i \in V'} y_i^* + 2$$

همچنین، مقدار تابع هدف فرمول‌بندی IDF2 برای z^* برابر با مقدار زیر خواهد بود.

$$\sum_{i \in V} x_i^{**} + 2\sum_{i \in V} y_i^{**} = \sum_{i \in V'} x_i^* + 2\sum_{i \in V'} y_i^* + 3$$

بنابراین، مقدار تابع هدف فرمول‌بندی IDF2 برای جواب شدنی z^{**} کمتر از مقدار تابع هدف فرمول‌بندی IDF2 برای جواب شدنی z^* است. این با فرض این که z^* یک جواب بهینه برای IDF2 است در تناقض است. این اثبات لم را کامل می‌کند.

قضیه ۲. برای یک گراف داده شده G، مقدار تابع هدف فرمول‌بندی (٤)–(١) برابر با مقدار تابع هدف فرمول‌بندی (۷)–(۵) است.

اثبات. فرض کنید $G = (V, E)$ یک گراف با $V = \{1, \ldots, n\}$ است. مقادیر توابع هدف فرمول‌بندی‌های IDF1 و IDF2 را به ترتیب با OPT1 و OPT2 نشان می‌دهیم. فرض کنید S_1 و S_2 به ترتیب مجموعه جواب‌های شدنی فرمول‌بندی‌های IDF1 و IDF2 باشند. واضح است که هر جواب شدنی برای IDF1 نیز یک جواب شدنی برای IDF2 است و در نتیجه داریم $S_1 \subseteq S_2$ این نتیجه می‌دهد که

$$\mathrm{OPT2} = \min_{S_2} \sum_{i \in V} (x_i + 2y_i) \leq \min_{S_1} \sum_{i \in V} (x_i + 2y_i) = \mathrm{OPT1}$$

برعکس، فرض کنید $z^* = (x^*, y^*) \in \{0,1\}^{2n}$ که $x^* = (x_1^*, \cdots, x_n^*)$ و $y^* = (y_1^*, \ldots, y_n^*)$ یک جواب بهینه برای فرمول بندی IDF2 است. در نتیجه $\mathrm{OPT2} = \sum_{i \in V} x_i^* + 2\sum_{i \in V} y_i^*$. با توجه به لم ۱، برای هر $i \in V$ داریم $x_i^* + y_i^* \leq 1$ و در نتیجه $z^* = (x^*, y^*)$ نیز یک جواب شدنی برای IDF1 است. بنابراین $\mathrm{OPT1} \leq \sum_{i \in V} (x_i^* + 2y_i^*) = \mathrm{OPT2}$ این اثبات قضیه را کامل می‌کند.

۳. یک الگوریتم تقریبی

در این بخش، یک الگوریتم حریصانه برای حل مسئله تابع احاطه‌گر ایتالیایی روی گراف G بر اساس فرمول‌بندی IDF2 پیشنهاد شده در بخش قبلی ارائه می‌کنیم. فرض کنید

$V = \{\underbrace{1, \ldots, n}_{n}\}$،

$\boldsymbol{b} = (\overbrace{2, \ldots, 2}^{n})$،

$\boldsymbol{c} = (\overbrace{1, \ldots, 1}^{n}, \overbrace{2, \ldots, 2}^{n})$،

$\boldsymbol{z} = (x_1, \ldots, x_n, y_1, \ldots, y_n)$ و

$A = [2I_n + A(G) \;\; 2I_n + 2A(G)]_{n \times 2n}$،

به طوری که I_n یک ماتریس همانی از اندازه n و $A(G)$ ماتریس مجاورت گراف G است، به عبارت دیگر یک ماتریس مربعی $n \times n$ به طوری که عنصر سطر i و ستون j مقدار یک است اگر $ij \in E$ و در غیر این صورت صفر است. می‌توانیم مدل برنامه‌ریزی خطی صحیح تعریف شده در (۷)–(۵) IDF2 را به صورت زیر بازنویسی کنیم.

$$\text{Min} \quad \boldsymbol{c} \cdot \boldsymbol{z}^t \qquad (۸)$$
$$s.t. \quad A\boldsymbol{z}^t \geq \boldsymbol{b}^t \qquad (۹)$$
$$\boldsymbol{z} \in \{0,1\}^{2n} \qquad (۱۰)$$

مثال ۲. برای گراف G داده شده در مثال ۱ فرمول‌بندی IDF2 در قالب شکل ماتریسی مدل برنامه‌ریزی خطی صحیح (۸)–(۱۰) به صورت زیر خواهد بود. توجه کنید که در (۸)–(۱۰) همه ضرایب باید صحیح باشند.

$$\text{Min} \quad (1,1,1,1,1,2,2,2,2,2)\,(x_1, \ldots, x_5, y_1, \ldots, y_5)^t$$

$$s.t.\quad \begin{bmatrix} 2&1&0&0&0&2&2&0&0&0\\ 1&2&1&0&1&2&2&2&0&2\\ 0&1&2&1&0&0&2&2&2&0\\ 0&0&1&2&1&0&0&2&2&2\\ 0&1&0&1&2&0&2&0&2&2 \end{bmatrix} \begin{bmatrix} x_1\\ \vdots\\ x_5\\ y_1\\ \vdots\\ y_5 \end{bmatrix} \geq \begin{bmatrix} 2\\2\\2\\2\\2 \end{bmatrix}$$

$$(x_1, \dots, x_5, y_1, \dots, y_5) \in \{0,1\}^{10}$$

قضیه ۳. فرض کنید OPT جواب بهینه مدل برنامه‌ریزی خطی صحیح تعریف شده در (۱۰)–(۸) و $\boldsymbol{x}$ خروجی الگوریتم ۳٫۱ است. آنگاه $\frac{c\cdot x^t}{\mathrm{OPT}} \leq H(\max_{j\in\{1,\dots,2n\}} \sum_{i=1}^{n} a_{ij})$ که $H(d) = \sum_{i=1}^{n} 1/i$.

اثبات. به [۱۵] مراجعه شود.

توجه کنید که بر اساس مدل برنامه‌ریزی خطی صحیح (۸)–(۱۰) الگوریتم ۳٫۱ پیشنهادی مقاله [۱۵] کار می‌کند. بنابراین برای استفاده از این الگوریتم حریصانه برای حل یک مسئله سخت به صورت تقریبی باید یک مدل برنامه‌ریزی خطی صحیح برای مسئله سخت به شکل (۸)–(۱۰) ارائه کنیم. در فرمول‌بندی IDF1 در قید (۳) برای هر راس $v \in V$ داریم $x_v + y_v \leq 1$ که این قید را نمی‌توان در قالب مدل برنامه‌ریزی خطی صحیح (۸)–(۱۰) بیان کرد. در قضیه ۲ ثابت کردیم که مقدار تابع هدف فرمول‌بندی‌های IDF1 و IDF2 برابر هستند. در فرمول‌بندی IDF2 قید (۳) فرمول‌بندی IDF1 حذف شده است. بنابراین از فرمول‌بندی IDF2 برای طراحی الگوریتم استفاده می‌کنیم که می‌توان آن را به شکل مدل برنامه‌ریزی خطی صحیح (۸)–(۱۰) بازنویسی کرد.

در مورد الگوریتم ۳٫۱ توضیحات زیر قابل توجه است. الگوریتم، حریصانه یکی از ستون‌های j ماتریس A را انتخاب می‌کند که مقدار $\frac{c_j}{\sum_{i=1}^{n} a_{ij}}$ را کمینه می‌کند. این انتخاب تابع هدف را مقدار کمی افزایش می‌دهد و بیشتر شرطها (ستون‌های ماتریس A) را تا حدودی ارضا می‌کند. وقتی که داریم $a_{ij} > b_i$ با انتخاب $x_j = 1$ ممکن است که شرط i-ام ارضا شود و مقدار بزرگ a_{ij} باعث شود که نسبت $\frac{c_j}{\sum_{i=1}^{n} a_{ij}}$ کوچک به نظر آید. بنابراین لازم است مقادیر بزرگ a_{ij} ماتریس A را در پایان هر دور اجرای حلقه تکرار الگوریتم حذف کنیم. این کار با جایگزین کردن مقادیر بزرگ a_{ij} با مقدار $\min\{a_{ij}, b_i\}$ انجام می‌شود.

قضیه ۴. برای یک گراف داده شده $G = (V, E)$ با $V = \{1, \dots, n\}$، یک الگوریتم تقریبی با ضریب $H(2\Delta(G) + 2)$ برای مسئله تابع احاطه‌گر ایتالیایی روی گراف G وجود دارد.

اثبات. فرض کنید OPT مقدار بهینه تابع هدف IDF2 است. بنا به قضیه ۲، OPT مقدار بهینه تابع هدف IDF1 نیز هست. فرض کنید $\boldsymbol{x} = (z_1, \dots, z_{2n})$ خروجی الگوریتم ۳٫۱ با ورودی گراف G است. از آنجایی که $\boldsymbol{x}$ خروجی الگوریتم ۳٫۱ با ورودی گراف G است $\boldsymbol{x}$ یک جواب شدنی برای فرمول‌بندی IDF2 است و ممکن است برای یک مقدار $i \in \{1, \dots, n\}$ داشته باشیم $z_i = z_{2i} = 1$؛ یا به عبارت دیگر برای یک راس مانند $i \in V$ داریم $x_i = y_i = 1$. چنین جوابی نمی‌تواند یک جواب شدنی برای فرمول‌بندی IDF1 باشد چرا که قید (۳) ارضا نمی‌شود. بنابراین، بردار جدید $\boldsymbol{y} = (x_1, \dots, x_n, y_1, \dots, y_n)$ را به صورت $x_i = 0$ اگر $z_i = z_{i+n}$ و در غیر این صورت (اگر $z_i \neq z_{i+n}$) $x_i = 1$ و $y_i = z_{i+n}$ برای هر $i \in V$ محاسبه می‌کنیم. با توجه به این که $\boldsymbol{y}$ علاوه بر قیدهای IDF2 قید $x_i + y_i \leq 1$ را برای هر $i \in V$ ارضا می‌کند یک جواب شدنی برای IDF1 است به طوری که $\boldsymbol{c}\cdot\boldsymbol{y}^t \leq \boldsymbol{c}\cdot\boldsymbol{x}^t$. (توجه کنید که تابع هدف هر دو فرمول‌بندی IDF1 و IDF2 یکسان هستند.) با توجه به قضیه ۳ بدست می‌آوریم که

$$\frac{c\cdot y^t}{\mathrm{OPT}} \leq \frac{c\cdot x^t}{\mathrm{OPT}} \leq H\left(\max_{j\in\{1,\dots,2n\}} \sum_{i=1}^{n} a_{ij}\right) = H(2\Delta(G) + 2)$$

این اثبات قضیه را کامل می‌کند.

با توجه به این که $H(d) \leq \ln d + 1$، در نتیجه ضریب تقریب قضیه ٤ می‌تواند حداکثر $\ln(2\Delta + 2) + 1$ باشد $\ln(2\Delta + 2) + 1 \leq \ln(\Delta + 1) + \ln 2 + 1 < \ln(\Delta + 1) + 2 < 2\ln(\Delta + 1) + 2$ که نشان می‌دهد ضریب تقریب الگوریتم پیشنهادی در [۱۲] را بهبود می‌بخشد.

(G) Algorithm IDFP

$G = (V, E)$ Input: A graph

Output: A feasible solution $\boldsymbol{x}$ to ILP defined in (8)-(10).

$\boldsymbol{x} = (0, \dots, 0)_{1\times 2n}$

$S = \{1, \dots, 2n\}$

while $\boldsymbol{b} \neq \boldsymbol{0}$ do

$k = \arg\min_{j \in S}\{\frac{c_j}{\sum_{i=1}^{n} a_{ij}}\}$

$x_k = 1$

$S = S \setminus \{k\}$

$b_i = b_i - a_{ik}$ for all $i \in \{1, \dots, n\}$

$a_{ij} = \min\{a_{ij}, b_i\}$ for all $i \in \{1, \dots, n\}, j \in \{1, \dots, 2n\}$

return $\boldsymbol{x}$

مثال ۳. در اینجا الگوریتم ۳٫۱ را روی گراف G داده شده در مثال ۱ دنبال می‌کنیم. توجه کنید که ماتریس‌های $\boldsymbol{b}, \boldsymbol{z}, \boldsymbol{c}, A$ در مدل برنامه‌ریزی خطی صحیح (۸)-(۱۰) برای گراف G در مثال ٤ داده شده‌اند. قبل از اجرای حلقه تکرار while داریم:

$x = (0, \dots, 0)_{1\times 10}$ و $S = \{1, \dots, 10\}$.

بعد از اولین اجرای حلقه تکرار while داریم:

$k = 2$ و $x = (0,1,0, \dots, 0)_{1\times 10}$ و $S = \{1,3, \dots, 10\}$ و

$\boldsymbol{b} = (1,0,1,2,1)^t$ و

$$A = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 2 & 1 & 0 & 0 & 2 & 2 & 2 \\ 0 & 1 & 0 & 1 & 1 & 0 & 1 & 0 & 1 & 1 \end{bmatrix}$$

بعد از دومین اجرای حلقه تکرار while داریم:

$k = 4$ و $x = (0,1,0,1,0, \dots, 0)_{1\times 10}$ و $S = \{1,3,5, \dots, 10\}$

و $\boldsymbol{b} = (1,0,0,0,0)^t$ و

$$A = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

بعد از سومین اجرای حلقه تکرار while داریم:

$k = 1$ و $x = (1,1,0,1,0, \dots, 0)_{1\times 10}$ و $S = \{3,5, \dots, 10\}$

و $\boldsymbol{b} = (0,0,0,0,0)^t$ و

$$A = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

حال چون $\boldsymbol{b} = (0,0,0,0,0)^t$ شرط حلقه تکرار while غلط است و حلقه تکرار while خاتمه می‌یابد و الگوریتم $x = (1,1,0,1,0, \dots, 0)_{1\times 10}$ به عنوان خروجی برمی‌گرداند که می‌گوید به راس‌های ۱، ۲ و ٤ مقدار ۱ و به بقیه راس‌ها مقدار صفر را نسبت دهیم که از اتفاق یک جواب بهینه برای مسئله تابع احاطه‌گر ایتالیایی روی گراف G است.

نتیجه‌گیری

در این مقاله مسئله یافتن یک تابع احاطه‌گر ایتالیایی روی گراف‌ها را بررسی کردیم. با توجه به اینکه این مسئله NP-سخت است، ارائه یک الگوریتم تقریبی یک پیشنهاد خوب برای حل این مسئله است. در این مقاله ابتدا یک مدل برنامه‌ریزی خطی صحیح برای مسئله تابع احاطه‌گر ایتالیایی روی گراف‌ها پیشنهاد دادیم و سپس یک مدل برنامه‌ریزی خطی صحیح جایگزین برای مدل اولیه ارائه کردیم. در انتها با استفاده از مدل برنامه‌ریزی خطی صحیح جایگزین پیشنهادی، یک الگوریتم حریصانه تقریبی برای حل مسئله تابع احاطه‌گر ایتالیایی روی گراف‌ها ارائه کردیم.

مراجع

[1] Stewart, I., "Defend the Roman empire!", Sci. Amer. 281 (1999) 136–139.
[2] ReVelle, C.S., Rosing, K.E., "Defendens imperium romanum: a classical problem in military strategy", Amer. Math. Monthly 107 (2000) 585–594.
[3] Cockayne, E.J., Dreyer Sr., P.M., Hedetniemi, S.M., Hedetniemi, S.T., "Roman domination in graphs", Discrete Math. 278 (2004) 11–22.
[4] Beeler, R.A., Haynes, T.W., Hedetniemi, S.T., "Double Roman domination", Discrete Appl. Math. 211 (2016) 23–29
[5] Chellali, M., Haynes, T.W., Hedetniemi, S.T., MacRae, A., "Roman {2}-domination", Discrete Appl. Math. 204 (2016) 22–28.
[6] Abdollahzadeh Ahangar, H., Henning, M.A., Samodivkin, V., Yero, I.G., "Total Roman domination in graphs", Applicable Analysis and Discrete Mathematics, 10 (2016) 501-517.
[7] Abdollahzadeh Ahangar, H., Henning, M.A., Löwenstein, C., Zhao, Y., Samodivkin, V., "Signed Roman domination in graphs", Journal of Combinatorial Optimization, 27 (2014) 241-255.
[8] Roushini Leely Pushpam, P., Padmapriea, S., "Global Roman domination in graphs", Discrete Appl. Math. 200 (2016) 176–185.
[9] Chellali, M., Jafari Rad, N., Sheikholeslami, S.M., Volkmann, L., "Varieties of Roman domination II", AKCE International Journal of Graphs and Combinatorics, 17 (2020) 966-984.
[10] Henning, M.A., Klostermeyer, W.F., "Italian domination in trees", Discrete Appl. Math. 217 (2017) 557–564.
[11] Poureidi, A., Jafari Rad, N., "On the algorithmic complexity of Roman {2}-domination (Italian domination)", Iran J. Sci. Technol. Trans. Sci. 44 (2020) 791–799.
[12] Padamutham, C., Palagiri , V. S. R., "Complexity of Roman {2}-domination and the double Roman domination in graphs", AKCE Int. J. Graphs and Combin., 17 (2020) 1081–1086.
[13] Vazirani., V., "Approximation Algorithms" Springer-Verlag, Berlin, Germany, 2001.
[14] Hromkovic, J., "Algorithms for Hard Problems" Springer-Verlag Berlin Heidlberg, 2001.
[15] Dobson, G., "Worst-case analysis of greedy heuristics for integer programming with nonnegative data", Math. Oper. Res 7 (1982) 515–531.