یک روش برای حل مسأله‌ی ممانعت ایمن ترین مسیر

جواد طیبی [*,1]، ابومسلم محمدی [2]

۱- استادیار، دانشگاه صنعتی بیرجند، گروه مهندسی صنایع

۲- هیات علمی دانشگاه امام علی (ع)، گروه ریاضی

چکیده

در این مقاله مسأله‌ی ممانعت ایمن‌ترین مسیر مورد مطالعه قرار می‌گیرد. این مسأله یک مسأله‌ی بهینه‌سازی شبکه است که کاربردهای نظامی فراوانی دارد. از طرف دیگر می‌توان این مسأله را جز مسایل بازی استکلبرگ به شمار آورد که بازی شامل دو بازیکن مهاجم و مدافع است. مهاجم برای نفوذ به یک نقطه می‌خواهد ایمن ترین مسیر ممکن را اختیار کند در حالی که هدف مدافع کم کردن ایمنی مسیرهاست به گونه‌ای که تا حد امکان جلوی نفوذ مهاجم را بگیرد. ما مسأله را از دیدگاه مدافع فرمول‌بندی کرده و روش‌هایی برای حل آن ارایه می‌کنیم.

کلمات کلیدی: مسایل ممانعت، مسأله‌ی امن‌ترین مسیر، کوتاهترین مسیر، روش نیوتن.

۱. مقدمه

در این مقاله به مطالعه‌ی یکی از مسایل کاربردی در حوزه‌ی نظامی خواهیم پرداخت که آن را مسأله‌ی ممانعت ایمن‌ترین مسیر نامیم. علی رغم این که گونه‌های مختلفی از مسایل ممانعت توسط پژوهشگران بررسی شده است ، این مسأله برای اولین بار در این مقاله بررسی می‌گردد [1, 2, 3]. اجازه دهید که ابتدا مسأله را از دیدگاه نظامی شرح دهیم:

فرض کنید یک گردان از دشمن که اکنون در مکان مشخصی که آن را نقطه‌ی شروع می‌نامیم، مستقر است. دشمن می‌خواهد به یک منطقه‌ی حساس (مانند یک شهر، یک پایگاه نظامی و ...) حمله کند که به آن منطقه، نقطه‌ی هدف گوییم. برای شناسایی نقطه هدف، یک گروه جاسوسی از افراد دشمن ملزم به نفوذ در این منطقه هستند. بین نقطه‌ی شروع و نقطه ی هدف مسیرهای متعددی قرار دارند که گروه جاسوسی با انتخاب هر یک از آن‌ها می‌تواند به نقطه‌ی هدف دست یابد. دشمن می‌خواهد از بین این مسیرها ایمن‌ترین مسیر ممکن را انتخاب کند تا از آن طریق به نقطه‌ی هدف رسیده و احتمال شناسایی نیروهای وی کاهش یابند. از طرفی نیروهای خودی می‌خواهند تا حد ممکن با استقرار نیروها در مسیرها جلوی نفوذ دشمن را به نقطه‌ی هدف بگیرند. به دلیل محدودیت تعداد نیروی انسانی (و همچنین مهمات)، نیروهای خودی باید تصمیم بگیرند که در کدام مسیرها و با چه تعداد مستقر شوند که بتوانند تا حد ممکن نیروهای دشمن را شناسایی و سرکوب کنند. پس به طور خلاصه، در این مسأله هدف دشمن یافتن ایمن‌ترین مسیر برای دستیابی به نقطه‌ی هدف است در حالی که هدف نیروهای خودی تا حد امکان کاهش دادن ایمنی مسیرهاست تا بدین وسیله از ورود دشمن به نقطه‌ی هدف

[*] **Corresponding author: Javad Tayyebi.**
Email: **javadtayyebi@birjandut.ac.ir**

جلوگیری شود. این مسأله را مسأله‌ی ممانعت ایمن‌ترین مسیر گوییم. در این فصل به مطالعه‌ی این مسأله و روش‌های حل آن خواهیم پرداخت.

در بخش اول مسأله‌ی ایمن‌ترین مسیر را معرفی کرده و در مورد روش‌های حل آن بحث می‌کنیم. این مسأله یک مقدمه برای وارد شدن به بحث اصلی است. در بخش دوم مسأله‌ی ممانعت ایمن‌ترین مسیر مورد توجه قرار گرفته و به طور کامل تشریح می‌گردد. هر دو مسأله‌ی مورد بحث از مسایل بهینه‌سازی شبکه هستند. در مسأله‌ی ایمن‌ترین مسیر شبکه‌ی زمینه ثابت بوده و به دنبال جواب بهینه (ایمن‌ترین مسیر) هستیم. در حالی که در مسأله‌ی ممانعت ایمن‌ترین مسیر شبکه ی زمینه متغیر است بدان معنا که ایمنی هر مسیر را می‌توان در صورت نیاز کاهش داد تا جلوی مطلوبیت جواب بهینه را بگیریم. به بیان روشن‌تر، مسأله‌ی ممانعت ایمن‌ترین مسیر یک بازی دو نفره استکلبرگ محسوب می‌شود که در آن یک بازیکن (نیروی دشمن) به دنبال یافتن ایمن‌ترین مسیر بوده و بازیکن دیگر (نیروی خودی) ایمنی مسیرها را تغییر می‌دهد تا بازیکن اول به جواب بهینه‌ی مطلوب دست پیدا نکند [4].

۲- مسأله‌ی ایمن‌ترین مسیر

فرض کنید یک گراف جهت‌دار $\boldsymbol{G(V,A)}$ داده شده است که در آن $\boldsymbol{V=\{1,2,\dots,n\}}$ مجموعه‌ی رئوس گراف بوده و $\boldsymbol{A}$ مجموعه‌ای شامل یال‌های گراف است. مجموعه‌ی $\boldsymbol{V}$ شامل دو رأس به خصوص $\boldsymbol{1}$ و $\boldsymbol{n}$ است که راس $\boldsymbol{1}$ همان مکان استقرار دشمن را نشان داده و راس $\boldsymbol{n}$ نشانگر منطقه‌ی حساس خودی است. منظور از یال‌های گراف مسیرهایی است که در امتداد این دو نقطه واقعند و دیگر رئوس گراف نشان دهنده‌ی تقاطع جاده‌ها می‌باشند. به هر یال $\boldsymbol{(i,j)\in A}$ از گراف یک عدد نامنفی $\boldsymbol{c_{ij}}$ را نسبت می‌دهیم. $\boldsymbol{c_{ij}}$ نشان دهنده‌ی ایمنی فعلی $\boldsymbol{(i,j)}$ است. منظور از یک مسیر $\boldsymbol{P}$ از رأس $\boldsymbol{1}$ به رأس $\boldsymbol{n}$ دنباله‌ای از یال‌ها مانند $\boldsymbol{(1,i_1)-(i_1,i_2)-\cdots-(i_k,n)}$ است. توجه کنید که در این دنباله یال اول از رأس $\boldsymbol{1}$ آغاز شده و یال انتهایی به رأس $\boldsymbol{n}$ ختم می‌گردد و همچنین هر دو یال متوالی در دنباله دارای رأس مشترک هستند. منظور از ایمنی هر مسیر $\boldsymbol{p}$ حداقل مقدار ایمنی یال‌های آن است. به بیان دیگر اگر با $\boldsymbol{c_p}$ ایمنی مسیر $\boldsymbol{p}$ را نشان دهیم، آن‌گاه $\boldsymbol{c_p=\min_{(i,j)\in P} c_{ij}}$. در مسأله‌ی ایمن‌ترین مسیر می‌خواهیم مسیری را با بیشترین ایمنی بیابیم. با توضیحات فوق می‌توان مسأله را به شکل زیر فرمول‌بندی کرد:

$$\max_{p\in P}\ \min_{(i,j)\in P} c_{ij},$$

)**Error! No text of specified style in document.**(۱-

که در این جا P مجموعه‌ی همه‌ی مسیرهایی در گراف G از راس 1 به راس n است. برای ارایه‌ی یک فرمول‌بندی دیگر از مسأله، نیاز به فرض زیر داریم:

فرض: همه‌ی مقادیر ایمنی مثبت هستند یعنی، به ازای هر $(i,j)\in A$، $c_{ij}>0$.

این فرض یک فرض محدود کننده نیست زیرا عملا مسیرهایی با ایمنی صفر هرگز انتخاب نخواهند شد، پس می‌توان یال‌هایی که ایمنی صفر دارند را از شبکه حذف کرد تا فرض فوق برقرار شود.

فرض کنید p یک مسیر دلخواه از 1 به n است. به ازای هر یال $(i,j)\in A$، متغیر صفر و یک x_{ij} به صورت زیر تعریف شود:

$$x_{ij}=\begin{cases}1 & (i,j)\in p,\\ 0 & (i,j)\notin p.\end{cases}$$

با این تعریف، می‌توان مسأله‌ی ایمن‌ترین مسیر را به صورت زیر نیز فرمول‌بندی کرد:

$$\max \min\{c_{ij}+M(1-x_{ij})\}$$
$$s.t.\ \sum_{(i,j)\in A} x_{ij}-\sum_{(j,i)\in A} x_{ji}=\begin{cases}1 & i=1\\ 0 & i\neq 1,n\\ -1 & i=n\end{cases}\quad \forall i\in V,$$
$$x_{ij}\in\{0,1\}\qquad \forall (i,j)\in A.$$

در این فرمول‌بندی M بیانگر یک عدد ثابت بسیار بزرگ است. توجه کنید که به ازای هر مسیر p، اگر $(i,j)\in p$ آن‌گاه $c_{ij}+M(1-x_{ij})=c_{ij}$ و در غیر این صورت حاصل این مقدار عدد بسیار بزرگ است. بنابراین $\min c_{ij}+M(1-x_{ij})$ همان مقدار ایمنی مسیر p خواهد بود. محدودیت‌های مسأله همان محدودیت‌های مسأله‌ی کوتاهترین مسیر هستند که به محدودیت‌های بقای جرم نیز شهرت دارند [5]. این محدودیت‌ها بیان می‌کنند که یک واحد جریان بایستی از راس 1 آغاز شده و به راس n برسد. بقیه رئوس نقش رئوس واسطه را دارند یعنی تفاضل جریان ورودی از خروجی آن‌ها برابر صفر است.

با تعریف متغیر $c_p=\min c_{ij}+M(1-x_{ij})$ این مسأله قابل تبدیل به یک مسأله‌ی برنامه‌ریزی خطی عدد صحیح به شکل زیر است:

$$\max c_p$$
$$s.t.\ c_{ij}+M(1-x_{ij})\geq c_p\quad \forall (i,j)\in A,$$
$$\sum_{(i,j)\in A} x_{ij}-\sum_{(j,i)\in A} x_{ji}=\begin{cases}1 & i=1\\ 0 & i\neq 1,n\\ -1 & i=n\end{cases}\quad \forall i\in V,$$
$$x_{ij}\in\{0,1\}\qquad \forall (i,j)\in A.$$

)Error! No text of specified style in document.-2(

بنابراین می‌توان مسأله را با استفاده از تکنیک‌های حل مسایل برنامه ریزی خطی صفر و یک مانند روش جمعی بالاس حل کرد. اما از آن‌جا که پیچیدگی زمانی این‌گونه تکنیک‌ها نمایی است، در ادامه به تشریح یک الگوریتم کارا برای حل مسأله می‌پردازیم که بتوانند مسأله را در زمان چندجمله‌ای حل کنند.

یکی از مشهورترین الگوریتم‌هایی که برای حل مسأله‌ی کوتاهترین مسیر* به کار می‌رود، الگوریتمی منسوب به دیکسترا† است *[6]*. این الگوریتم را می‌توان برای یافتن کوتاهترین مسیر از یک راس مبدا s به سایر رئوس به کار برد. البته شایان ذکر است که تنها در صورتی الگوریتم قابل استفاده است که طول‌های یالی نامنفی باشند. در این بخش، الگوریتم دیکسترا را طوری تغییر می‌دهیم که بتواند مسأله‌ی (***2-****Error! No text of specified style in document.*) را حل کند.

این الگوریتم را الگوریتم تغییر یافته‌ی دیکسترا می‌نامیم. در این الگوریتم هر راس $\boldsymbol{i}$ دارای یک برچسب $(\boldsymbol{d_i},\boldsymbol{p_i})$ است که اولین مولفه‌ی این زوج مرتب نشان دهنده‌ی این است که تاکنون مسیری از راس $\mathbf{1}$ به راس $\boldsymbol{i}$ با ایمنی برابر $\boldsymbol{d_i}$ یافته‌ایم و $\boldsymbol{p_i}$ نشان دهنده‌ی آخرین راس قبل از $\boldsymbol{i}$ بر این مسیر است که می‌توان از آن پس از خاتمه‌ی الگوریتم برای یافتن مسیر بهینه استفاده کرد. در ابتدا بر چسب راس $\mathbf{1}$ را برابر $(\boldsymbol{d_1},\boldsymbol{p_1})=(+\infty,\mathbf{0})$ در نظر گرفته و بر چسب رئوس دیگر را برابر $(-\infty,\#)$ قرار می‌دهیم. بر چسب‌ها به دو دسته‌ی دایم و موقت تفکیک می‌شوند. در هر تکرار از الگوریتم یکی از برچسب‌های موقت دایم خواهد شد و الگوریتم تا زمانی ادامه می‌یابد که بر چسب راس $\boldsymbol{n}$ دایم شود. از بین همه‌ی رئوس بر چسب راسی دایم می‌شود که بیشترین مقدار $\boldsymbol{d(i)}$ را در بین رئوس با برچسب موقت داشته باشد. پس در اولین تکرار

---

* Shortest path problem
† Dijkstra

برچسب راس $\mathbf{1}$ دایم می‌شود. پس از دایم شدن برچسب راس $\boldsymbol{i}$، برچسب سایر رئوس مجاور با این راس مانند $\boldsymbol{j}$ به روز می‌شود. بدین صورت که اگر $\boldsymbol{d_j < min\{d_i, c_{ij}\}}$ آن‌گاه مقدار $\boldsymbol{d_j}$ به مقدار $\boldsymbol{(d_j, p_j) = (min\{d_i, c_{ij}\}, i)}$ تغییر می‌کند. در غیر این صورت بر چسب راس $\boldsymbol{j}$ تغییری نخواهد کرد. پس از به روز رسانی برچسب‌ها روند فوق ادامه می‌یابد تا زمانی‌که بر چسب راس $\boldsymbol{n}$ دایم شود. پس از خاتمه‌ی الگوریتم مقدار تابع هدف مسأله‌ی (*Error! No text of specified style in document.***2-**) برابر $\boldsymbol{d_t}$ است که می‌توان با استفاده از مولفه‌های دوم برچسب‌ها و پیمایش به عقب ایمن‌ترین مسیر را یافت. در زیر الگوریتم دیکسترای تغییر یافته آورده شده است.

الگوریتم ۱: الگوریتم دیکسترای تغییر یافته برای حل مسأله‌ی (*Error! No text of specified style in document.-2*).
گام ۱. قرار دهید $\boldsymbol{(d_1, p_1) = (+\infty, 0)}$ برای هر راس $\boldsymbol{i \in V \backslash \{1\}}$ قرار دهید $\boldsymbol{(d_i, p_i) = (-\infty, \#)}$.
گام ۲. از بین همه‌ی رئوس با بر چسب موقت، راسی را با بیشترین مقدار $\boldsymbol{d_i}$ دایم کنید. فرض می‌کنیم که راس $\boldsymbol{i}$ دایم شده است.
گام ۳. اگر راس $\boldsymbol{n}$ دایم شده است متوقف شوید، در غیر این صورت به گام بعدی بروید.
گام ٤. همه‌ی یال‌های $\boldsymbol{(i,j)}$ خروجی از راس $\boldsymbol{i}$ را بررسی کنید. اگر $\boldsymbol{d_j < min\{d_i, c_{ij}\}}$ قرار دهید $\boldsymbol{(d_j, p_j) = (min\{d_i, c_{ij}\}, i)}$. به گام ۲ بازگردید.

۳- مسأله‌ی ممانعت ایمن‌ترین مسیر

مسایل نظامی را معمولا می‌توان در رده به خصوصی از مسایل بهینه سازی به نام تئوری بازی ها جای داد. مسایل بازی‌ها دارای دیدگاه‌های دو طرفه هستند: دیدگاه بازیکن اول (نیروهای خودی) و دیدگاه بازیکن دوم (نیروهای دشمن). در این بخش به معرفی مسأله‌ی ممانعت ایمن‌ترین مسیر می‌پردازیم که یک بازی دو نفره شامل نیروهای خودی و دشمن است.

اجازه دهید در ابتدا به بیان دقیق مسأله پرداخته و سپس آن را فرمول‌بندی کنیم. فرمول‌بندی ارایه شده، یک مسأله‌ی بهینه‌سازی دو سطحی* است.

فرض کنید یک گراف جهت‌دار $\boldsymbol{G(V, A)}$ داده شده است که در آن $\boldsymbol{V = \{1, 2, \dots, n\}}$ مجموعه‌ی رئوس گراف بوده و $\boldsymbol{A}$ مجموعه‌ای شامل یال‌های گراف است. مجموعه‌ی $\boldsymbol{V}$ شامل دو رأس به خصوص $\mathbf{1}$ و $\boldsymbol{n}$ است که راس $\mathbf{1}$ همان مکان استقرار دشمن (نقطه‌ی شروع یا مبدا) را نشان داده و راس $\boldsymbol{n}$ نشانگر منطقه‌ی حساس خودی (نقطه‌ی هدف یا مقصد) است. هر یال $\boldsymbol{(i,j) \in A}$ نشان دهنده‌ی مسیری است که نقطه‌ی $\boldsymbol{i}$ را به نقطه‌ی $\boldsymbol{j}$ متصل می‌کند. به هر یال $\boldsymbol{(i,j)}$ یک عدد نامنفی $\boldsymbol{c_{ij}}$ را نسبت می‌دهیم که همان مقدار کمی ایمنی مسیر است.

فرض کنید $\boldsymbol{d_{ij}}$ نشان دهنده‌ی متغیری نامنفی است که در صورت استقرار نیروهای خودی از ایمنی یال $\boldsymbol{(i,j)}$ کم می‌شود. پس اگر نیروهای خودی را بر این یال مستقر کنیم ایمنی یال از مقدار $\boldsymbol{c_{ij}}$ به مقدار $\boldsymbol{c_{ij} - d_{ij}}$ تغییر خواهد کرد. از این رو به ازای هر یال $\boldsymbol{(i,j) \in A}$ بایستی نامساوی $\boldsymbol{d_{ij} \le c_{ij}}$ برقرار باشد. این نامساوی بدان معناست که حداکثر میزان کاسته شدن ایمنی برابر مقدار ایمنی آن مسیر است. عدد $\boldsymbol{w_{ij}}$ مقدار هزینه‌ی واحدی است که باید در صورت استقرار یک واحد نیروهای خودی ($\boldsymbol{d_{ij} = 1}$) بر یال $\boldsymbol{(i,j)}$ بپردازیم. این هزینه می‌تواند شامل تعداد نیروی انسانی اعزام شده و همچنین مقدار تجهیزات مورد نیاز باشد. توجه کنید که به وضوح هر چه بخواهیم مقدار ایمنی یک مسیر را بیشتر کاهش

* Bilevel optimization problem

دهیم (یعنی d_{ij} را بزرگتر کنیم) نیاز است که هزینه‌ی بیشتری معادل $w_{ij}c_{ij}$ را برای یال (i,j) بپردازیم. در این مسأله فرض می‌کنیم که $W \geq 0$ حداکثر میزان بودجه‌ای است که می‌خواهیم برای ممانعت پیشروی دشمن بر یال‌ها هزینه کنیم. پس بایستی نامساوی

$$\sum_{(i,j)\in A} w_{ij}d_{ij} \leq W$$

)Error! No text of specified style in document.-3(

برقرار باشد. یعنی مجموع هزینه‌های صرف شده برای ممانعت، بیشتر از مقدار کل بودجه W نشود.

مسأله‌ی ممانعت ایمن‌ترین مسیر را می‌توان به صورت یک مسأله‌ی برنامه‌ریزی دو سطحی فرمول‌بندی کرد که در سطح پایین نیروهای دشمن ایمن‌ترین مسیر را انتخاب کرده و در سطح بالا نیروهای خودی با تخریب ایمنی، مانع از انتخاب ایمن‌ترین مسیرها می‌شوند. با این توضیحات و با استفاده از فرمول‌بندی (*Error! No text of specified style in document.*-**2**)، می‌توان سطح پایین مسأله را به شکل زیر فرمول‌بندی کرد:

$$\begin{aligned}
&z = \max c_p\\
&s.t.\ \ c_{ij} - d_{ij} + M(1-x_{ij}) \geq c_p \quad \forall (i,j)\in A,\\
&\sum_{(i,j)\in A} x_{ij} - \sum_{(j,i)\in A} x_{ji} = \begin{cases} 1 & i=1\\ 0 & i\neq 1,n\\ -1 & i=n\end{cases} \quad \forall i\in V,\\
&x_{ij}\in\{0,1\} \quad \forall (i,j)\in A.
\end{aligned}$$

(*Error! No text of specified style in document.*-۴)

این همان مسأله ایمن‌ترین مسیر (**2**-*Error! No text of specified style in document.*) است که در آن ایمنی هر یال از c_{ij} به $c_{ij} - d_{ij}$ تغییر کرده است. جواب بهینه‌ی این مسأله را در z ذخیره کرده‌ایم تا بتوانیم در مسأله سطح بالا از آن‌ها استفاده کنیم.در مسأله سطح بالا هدف مینیمم سازی مقدار تابع هدف مسأله

(**۴**-*Error! No text of specified style in document.*) یعنی همان c_p است. این سطح شامل محدودیت‌های بودجه $\sum_{(i,j)\in A} w_{ij}d_{ij} \leq W$و محدودیت‌های کران $0 \leq d_{ij} \leq c_{ij}$ به ازای هر یال $(i,j)\in A$ می‌باشد. بنابراین، مسأله‌ی ممانعت از ایمن‌ترین مسیر به صورت یک مسأله‌ی برنامه‌ریزی دو سطحی به شکل زیر فرمول‌بندی می‌شود:

$$\begin{aligned}
&\min z\\
&0 \leq d_{ij} \leq c_{ij} \quad \forall (i,j)\in A\\
&\sum_{(i,j)\in A} w_{ij}d_{ij} \leq W,\\
&z = \max c_p\\
&\qquad s.t.\ \ c_{ij} - d_{ij} + M(1-x_{ij}) \geq c_p \quad \forall (i,j)\in A,\\
&\qquad \sum x_{ij} - \sum x_{ji} = \begin{cases} 1 & i=1\\ 0 & i\neq 1,n \end{cases} \quad \forall i\in V,
\end{aligned}$$

(*Error! No text of specified style*

$$x_{ij} \in \{0,1\} \qquad \forall (i,j) \in A.$$ style in (۵-document.

این مسأله یک مسأله بهینه‌سازی خطی دو سطحی صفر و یک مختلط است که می‌توان آن را با نرم افزارهای بهینه‌سازی از جمله گمپس حل کرده و جواب بهینه را یافت. با این وجود، معمولا حل مسایل بهینه‌سازی خطی صفر و یک در ابعاد بزرگ حتی به وسیله‌ی ابرکامپیوترها ممکن است بسیار زمان‌بر باشد. از طرف دیگر، این مسأله یک مسأله‌ی تصمیم‌گیری استراتژیک است که زمان نقشی حیاتی را در آن ایفا می‌کند. از این رو بهتر است روش‌هایی برای حل این مسأله ارایه داد که زمان کمتری را صرف کنند. هدف ما در بخش‌های آتی رسیدن به این مقصود است.

۴- نتایج اولیه

اکنون آماده‌ایم که به بیان برخی نتایج اولیه در رابطه با مسأله‌ی (۱-۵) بپردازیم. با اسنفاده از این نتایج می‌توان الگوریتم‌هایی را برای حل این مسأله پیشنهاد کرد. در ابتدا به بیان فرض مثبت بودن داده‌ها می‌پردازیم:

فرض. از این پس فرض می‌کنیم که مقادیر ایمنی c_{ij} و هزینه‌ی w_{ij} برای هر یال $(i,j) \in A$ مثبت‌اند.

این فرض مسأله را محدود نمی‌کند زیرا اگر c_{ij} برابر صفر باشد، هیچ‌گاه دشمن مسیرهایی که از این یال می‌گذرند را انتخاب نخواهد کرد و بنابراین می‌توان یال‌هایی با ایمنی صفر را از شبکه حذف کرد. در حالتی که $w_{ij} = 0$ باشد، می‌توان در ابتدا ایمنی این گونه یال‌ها را (بدون هیچ هزینه‌ای) به مقدار صفر تغییر داده و آن‌ها را از شبکه حذف کرد.

قضیه *Error! No text of specified style in document.*-**۱**. اگر مقدار بهینه‌ی مسأله‌ی ایمن‌ترین مسیر مخالف صفر باشد، آن‌گاه در جواب بهینه همواره محدودیت بودجه به شکل تساوی برقرار است.

اثبات. به برهان خلف فرض کنید که d^* جواب بهینه‌ی مسأله‌ی ممانعت ایمن‌ترین مسیر باشد اما داشته باشیم: $\sum_{(i,j)\in A} w_{ij} d^*_{ij} < W$. یک جواب d' را به صورت زیر تعریف می‌کنیم:

$$d'_{ij} = \begin{cases} d^*_{ij} + \epsilon & d^*_{ij} < c_{ij}, \\ d^*_{ij} & d^*_{ij} = c_{ij}, \end{cases} \qquad \forall (i,j) \in A.$$

که در این جواب $\epsilon > 0$ عددی بسیار کوچک انتخاب شده است به طوری که اولا $\sum_{(i,j)\in A} w_{ij} d'_{ij} < W$ و ثانیا اگر $d^*_{ij} < c_{ij}$، آن‌گاه $d'_{ij} < c_{ij}$. به راحتی می‌توان نشان داد که مقدار بهینه‌ی جواب d' حداقل به اندازه‌ی ϵ کمتر از جواب بهینه‌ی d^* است. پس d^* نمی‌تواند جواب بهینه‌ی مسأله باشد که به تناقض می‌رسیم. □

از این پس فرض می‌کنیم که مقدار بهینه‌ی مسأله‌ی ممانعت ایمن‌ترین مسیر (۱-۵) برابر z^* است. یعنی مطمئن هستیم نیروهای دشمن هر مسیری را که انتخاب کنند، مقدار ایمنی مسیر بیشتر از z^* نخواهد بود. در مسأله‌ی ممانعت ایمن‌ترین مسیر می‌خواهیم ایمنی هر مسیری از نقطه‌ی شروع به نقطه‌ی هدف را با کمترین بودجه کاهش دهیم. برای این کار، کافی است فقط برای یک یال از هر مسیر این کار را انجام دهیم. مجموعه‌ی این یال‌ها منجر به یک برش می‌شوند. این برش شبکه را به دو مولفه‌ی همبندی تقسیم می‌کند که نقطه‌ی شروع در یک مولفه و نقطه‌ی هدف در مولفه‌ای دیگر قرار دارند. در یک برش بهینه یال‌هایی که ایمنی آن‌ها کمتر یا مساوی z^* است نیاز به تغییر ندارند. تنها کافی است ایمنی یال‌هایی از برش را که مقدار ایمنی کنونی آن‌ها بیشتر از z^* است، کاهش یابند. توجه کنید که ملزم هستیم مقدار ایمنی هر این گونه یالی را به مقدار z^* کاهش می‌دهیم تا مطمئن شویم مسیرهای گذرنده از این یال‌ها دارای ایمنی بیشتر از z^* نیستند. اجازه دهید نتایج حاصل را در یک قضیه به طور رسمی بیان کنیم.

قضیه **Error! No text of specified style in document.**-۲: فرض کنید مقدار بهینه‌ی مسأله‌ی ممانعت ایمن‌ترین مسیر برابر z^* باشد. همواره یک جواب بهینه d^* از مسأله موجود است به طوری که

$$d_{ij}^* = \begin{cases} 0 & (i,j) \notin C, \\ 0 & (i,j) \in C \wedge c_{ij} \leq z^*, \\ c_{ij} - z^* & (i,j) \in C \wedge c_{ij} > z^*, \end{cases} \qquad \forall (i,j) \in A,$$

که در آن C نشان دهنده‌ی مجموعه یال‌های پیشروی یک برش است که نقاط شروع و هدف را از هم جدا می‌کند.

مثال ۱. مسأله‌ی ممانعت ایمن‌ترین مسیر ارایه شده در (۱-۵) را در نظر بگیرید. همین طور که آن مثال دیدیم، مقدار بهینه‌ی این مسأله برابر $z^* = 2.5$ است. یعنی هر مسیر از 1 به 4 نمی‌تواند ایمنی بیشتر از 2.5 داشته باشد. در این مسأله برش $C = (S, \bar{S})$ با $S = \{1,2\}$ و $\bar{S} = \{3,4\}$ متناظر با جواب بهینه‌ی معرفی شده است. توجه کنید که ایمنی مسیر $(1,4)$ کمتر از 2.5 بوده و تغییر نکرده است اما ایمنی دو یال دیگر از مقدار 3 به مقدار $z^* = 2.5$ کاهش یافته‌اند. آخرین نکته‌ای که در این مثال باید ذکر شود این است که ایمنی یال $(3,2)$ نیز تغییر نکرده است زیرا علی رغم این که این یال در برش مورد نظر قرار دارد، اما یک یال پسرو است.

نتیجه‌ی حاصل از قضیه **Error! No text of specified style in document.**-۲: فرض کنید مقدار بهینه‌ی مسأله‌ی ممانعت ایمن‌ترین مسیر برابر z^* باشد. همواره یک جواب بهینه d^* از مسأله موجود است به طوری که بسیار حائز اهمیت است. زیرا بیان می‌کند که می‌توان جواب بهینه را تنها در جواب‌های خاصی از مسأله که متناظر با یک برش هستند یافت. اما نقطه‌ی ضعف این قضیه در این است که برای ساخت جواب بهینه از روی برش نیاز است که ابتدا مقدار بهینه را بدانیم. در ادامه، برای این که این نقطه‌ی ضعف را بر طرف سازیم برای هر برش یک جواب منحصر به فرد ارایه می‌دهیم.

برش دلخواه $[S, \bar{S}]$ را در نظر بگیرید. با توجه به قضیه **Error! No text of specified style in document.**-۲: فرض کنید مقدار بهینه‌ی مسأله‌ی ممانعت ایمن‌ترین مسیر برابر z^* باشد. همواره یک جواب بهینه d^* از مسأله موجود است به طوری که تنها ایمنی یال‌های پیشرو برش تغییر خواهند کرد که مجموعه‌ی آن‌ها را با $C = (S, \bar{S})$ نشان می‌دهیم. فرض کنید ایمنی یال‌های متعلق به C را به طور غیر نزولی مرتب کرده‌ایم و $c_0(=0) \leq c_1 \leq c_2 \leq \cdots \leq c_s$ نشان دهنده‌ی این لیست مرتب شده باشد. به ازای اندیس ثابت $l \in \{1, 2, \dots, s\}$ بردار $d^l \in R^{|A|}$ را به صورت زیر تعریف می‌کنیم:

$$d_{ij}^l = \begin{cases} 0 & (i,j) \notin C, \\ 0 & (i,j) \in C \wedge c_{ij} \leq c_l, \\ c_{ij} - z & (i,j) \in C \wedge c_{ij} > c_l, \end{cases} \qquad \forall (i,j) \in A, \qquad (Error!\ No\ text\ of\ specified\ style\ in\ document.\text{-}۶)$$

که در آن z مقداری است که به ازای آن محدودیت بودجه به شکل تساوی برقرار باشد. یعنی،

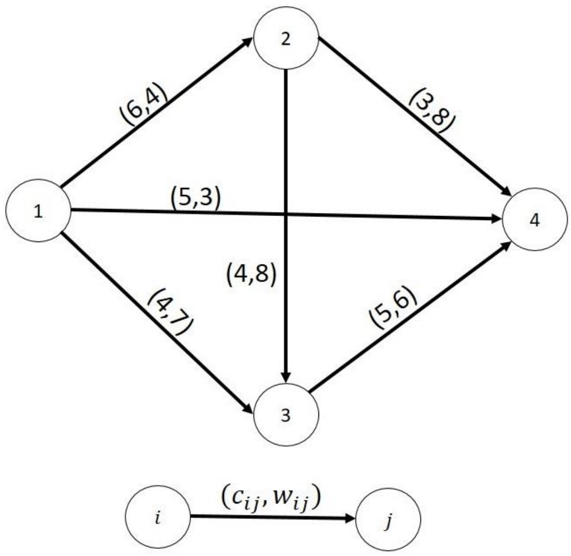
$$\sum_{(i,j) \in A} w_{ij} d_{ij}^l = W \implies$$

$$\sum_{(i,j)\in C:\, c_{ij}>c_l} w_{ij}\left(c_{ij}-z\right)=W \;\Longrightarrow$$

$$z=\frac{\sum_{(i,j)\in A: c_{ij}>c_l} c_{ij}w_{ij}-W}{\sum_{(i,j)\in A: c_{ij}>c_l} w_{ij}}. \qquad \text{(Error! No text of specified style in document.-۷)}$$

آخرین رابطه مقدار دقیق z را مشخص می‌کند. با توجه به قضیه *Error! No text of specified style in* z^* باشد. همواره یک جواب بهینه d^* از مسأله موجود است به طوری که، z را می‌توان یک برآورد از مقدار تابع هدف در نظر گرفت.

قضیه *Error! No text of specified style in document.*-۳. بردار d^l یک جواب شدنی از مسأله ممانعت ایمن‌ترین مسیر با مقدار تابع هدف z باشد، اگر و تنها اگر $c_l \le z \le c_{l+1}$.



شکل ۱: نمونه‌ای از مسأله‌ی ممانعت ایمن ترین مسیر با بودجه کل $W=20$.

اثبات. ابتدا قسمت لزوم را ثابت می‌کنیم. فرض کنید (i_l, j_l) یالی است که $c_{i_l j_l} = c_l$ و همچنین، (i_{l+1}, j_{l+1}) یالی است که $c_{i_{l+1} j_{l+1}} = c_{l+1}$.

به برهان خلف، اگر $z > c_{l+1}$ ، آن‌گاه $d^l_{i_{l+1} j_{l+1}}$ مقداری منفی است که نتیجه می‌دهد d^l یک جواب شدنی نیست. اما اگر $z < c_l$، آن‌گاه ایمنی همه‌ی یال‌های تغییر یافته از ایمنی یال (i_l, j_l) کمتر خواهد شد. پس هنوز هم مسیرهایی از نقطه شروع به نقطه هدف که از یال (i_l, j_l) می‌گذرند لزوما دارای مقدار ایمنی کمتر از z نیستند. بنابراین مقدار تابع هدف d^l برابر z نیست.

برای اثبات قسمت کفایت، توجه کنید که نامساوی $z \le c_{l+1}$، نامنفی بودن بردار d^l تضمین شده است. از طرف دیگر، رابطه‌ی

(۷-*Error! No text of specified style in document.*) برقراری قید بودجه را نیز تضمین می‌کند. پس d^l یک جواب شدنی از مسأله خواهد بود. در نهایت چون نامساوی $c_l \leq z$ برقرار است، پس ایمنی هر یال از برش حداکثر برابر z می‌باشد که نشان می‌دهد مقدار تابع هدف d^l برابر z است. $\square$

با توجه به این قضیه، کافی است برای هر برش C اندیسی مانند l را بیابیم که به ازای آن، نامساوی

$$c_l \leq \frac{\sum_{(i,j)\in A: c_{ij} > c_l} c_{ij} w_{ij} - W}{\sum_{(i,j)\in A: c_{ij} > c_l} w_{ij}} \leq c_{l+1}$$

برقرار باشد. در این صورت می‌توانیم جواب شدنی d^l را به دست آوریم.

مثال ۲. یک نمونه از مسأله‌ی ممانعت ایمن‌ترین مسیر در شکل فوق آورده شده است. در این مثال می‌خواهیم، جواب شدنی متناظر با برش $C = (S, \bar{S})$ با $S = \{1,2\}$ را به دست آوریم. مجموعه یال‌های پیشرو این برش برابر $C = \{(1,3),(1,4),(2,3),(2,4)\}$ هستند. مجموعه ایمنی این یال ها را در لیست زیر مرتب کرده‌ایم:

$$0(= c_0) \leq 3(= c_1) \leq 4(= c_2) \leq 4(= c_3) \leq 5(= c_4).$$

به ازای مقادیر مختلف $l = 0,1,2,3$ مقدار z را در جدول زیر به دست آورده‌ایم:

جدول ۱. مقادیر حاصل به ازای l های مختلف

l	z	$[c_l, c_{l+1}]$
0	$\frac{3\times 8 + 4\times 7 + 4\times 8 + 5\times 3 - 20}{8+7+8+3} = 3.81$	$[0,3]$
1	$\frac{4\times 7 + 4\times 8 + 5\times 3 - 20}{7+8+3} = \frac{55}{18} = 3.06$	$[3,4]$
2	$\frac{4\times 8 + 5\times 3 - 20}{8+3} = 2.45$	$[4,4]$
3	$\frac{5\times 3 - 20}{3} = -1.67$	$[4,5]$

همین طور که از مقادیر جدول فوق مشهود است تنها به ازای $l = 1$ شرط $c_l \leq z \leq c_{l+1}$ برقرار است. بنابراین جواب شدنی مربوط به این برش به صورت

$$d_{12}^0 = d_{34}^0 = d_{32}^0 = d_{24}^0 = 0,$$
$$d_{13}^0 = \frac{17}{18} = 0.944\ , d_{23}^0 = \frac{17}{18} = 0.944\ \ , d_{14}^0 = \frac{35}{18} = 1.944,$$

با مقدار $z^* = \frac{55}{18} = 3.06$ است. بنابراین این جواب ایمنی سه یال $(1,3), (2,3), (1,4)$ را به 3.06 تغییر می‌دهد و ایمنی بقیه یال‌ها بدون تغییر باقی می‌ماند. قابل ذکر است که یال $(2,4)$ با این که روی برش قرار دارد مقدار ایمنی آن تغییر نخواهد کرد. زیرا این یال در شرط $c_{24} > c_l$ صدق نمی‌کند.

۵- الگوریتم کارا

در این بخش به بیان یک الگوریتم زمان چندجمله‌ای برای حل مسأله‌ی ممانعت ایمن‌ترین مسیر می‌پردازیم. این الگوریتم شامل سه مرحله (سه فاز) است. الگوریتم در فاز اول یک بازه را جستجو می‌کند که مقدار تابع هدف بهینه در آن واقع شده است. در فاز دوم با استفاده از روش نیوتن مقدار بهینه‌ی دقیق را می‌یابد. در نهایت در سومین فاز، جواب بهینه را از روی مقدار بهینه‌ی حاصل از فاز دوم به دست می‌آورد. در زیر شمای کلی الگوریتم پیشنهادی به اختصار آورده شده است.

شمای کلی الگوریتم پیشنهادی:

- ✓ فاز اول: یافتن یک بازه که شامل مقدار بهینه‌ی تابع هدف است.
- ✓ فاز دوم: استفاده از روش نیوتن برای به دست آوردن مقدار بهینه.
- ✓ فاز سوم: محاسبه‌ی جواب بهینه از روی مقدار بهینه.

برای تشریح فاز اول، ابتدا نیاز به برخی نتایج در مورد جواب‌های شدنی مسأله داریم. فرض کنید z_{max} مقدار بهینه‌ی مسأله‌ی

$$\min_{C \in \mathcal{C}} \max_{(i,j) \in C} c_{ij}$$

(Error! No text of specified style in document.-۸)

باشد که $\mathcal{C}$ مجموعه‌ی همه‌ی برش‌هایی در G است که نقطه شروع را از نقطه هدف جدا می‌کند. مسأله‌ی (Error! No text of specified style in document.-۱۲) را مسأله‌ی برشی با کمترین تنگنا* نامیده و الگوریتم‌های زمان چندجمله‌ای برای حل آن موجود است. توجه کنید که z_{max} مقدار تابع هدف جواب شدنی $d = 0$ (یعنی بدترین جواب شدنی ممکن) است. بنابراین، مقدار تابع هدف هر جواب شدنی بایستی در بازه‌ی $[0, z_{max}]$ قرار گیرد.

فرض کنید عناصر مجموعه‌ی $\left(\{c_{ij}: (i,j) \in A\} \cup \{0\}\right) \cap [0, z_{max}]$ را به طور غیر نزولی مرتب کرده‌ایم و فرض کنید

$$(0 =)c_0 \leq c_1 \leq c_2 \leq c_3 \leq \cdots \leq c_k (= z_{max})$$

نشان دهنده‌ی این لیست مرتب شده است. از این پس این لیست را با بردار $L = [c_0, \dots, c_k]$ نمایش می‌دهیم. قضایای زیر در مورد وجود جواب‌های شدنی مسأله‌ی ممانعت ایمن‌ترین مسیر است.

* Minimum bottleneck cut problem

لم ‎.‎Error! No text of specified style in document.‎-۱. به ازای اندیس ثابت $l = 0, \dots, k$، اگر مسأله‌ی ممانعت ایمن‌ترین مسیر فاقد جواب شدنی با مقدار تابع هدف $z_l \in (c_{l-1}, c_l]$ باشد، آن‌گاه فاقد جواب شدنی با مقدار تابع هدف $z' \in [c_{l-1}, z_l)$ است.

اثبات. به برهان خلف فرض می‌کنیم مسأله‌ی ممانعت ایمن‌ترین مسیر دارای جواب شدنی d' با مقدار تابع هدف $z' \in [c_{l-1}, z_l]$ است. همچنین فرض می‌کنیم C' نشان دهنده‌ی مجموعه یال‌هایی است که مقدار ایمنی آن‌ها در این جواب تغییر کرده است. بدون از دست دادن کلیت می‌توان دو فرض اضافی بر جواب d' و مجموعه‌ی C' تحمیل کرد. اول این که فرض می‌کنیم فقط یال‌هایی متعلق به C' هستند که مقدار ایمنی c_{ij} آن‌ها بیشتر از z' است زیرا تغییر ایمنی سایر یال‌ها نه تنها مقدار تابع هدف را تغییر نمی‌دهد بلکه باعث هزینه‌ی اضافی نیز می‌گردد. دومین فرض این است که برای هر یال $(i,j) \in C'$ نامساوی $c_{ij} - d'_{ij} \leq z'$ برقرار است. این فرض نیز محدود کننده نیست زیرا اگر این فرض برای یک یال برقرار نباشد می‌توان با عدم تغییر ایمنی آن یال باز هم به جوابی شدنی با همین مقدار تابع هدف دست یافت. پس به ازای هر یال $(i,j) \in C'$، نامساوی‌های زیر را داریم:

الف. $d'_{ij} > 0$.

ب. $z' < c_{ij}$.

پ. $c_{ij} - d'_{ij} \leq z'$.

حال جواب شدنی d^l را به صورت زیر تعریف می‌کنیم:

$$d^l_{ij} = \begin{cases} 0 & (i,j) \notin C', \\ d'_{ij} - \epsilon & (i,j) \in C', \end{cases} \qquad \text{(Error! No text of specified style in document.-۹)}$$

که $\epsilon = z_l - z' > 0$.

با توجه به این که به ازای هر یال $(i,j) \in A$ نامساوی $d^l_{ij} \leq d'_{ij}$ برقرار است و d' یک جواب شدنی است، پس لزوما به ازای هر یال $(i,j) \in A$ نامساوی $d^l_{ij} \leq c_{ij}$ نیز برقرار بوده و حتی d^l در قید بودجه نیز صدق می‌کند. پس اگر نشان دهیم که به ازای هر یال $(i,j) \in A$ ، $d^l_{ij} \geq 0$، آن‌گاه d^l یک جواب شدنی از مسأله‌ی ممانعت ایمن‌ترین مسیر خواهد بود. نامنفی بودن d^l_{ij} به ازای $(i,j) \notin C'$ واضح است. برای هر $(i,j) \in C'$ داریم:

$$d^l_{ij} = d'_{ij} - \epsilon = d'_{ij} + z' - z_l \geq c_{ij} - z_l \geq 0,$$

که نامساوی اول همان نامساوی (پ) است و سومین نامساوی از (ب) و $c_{l-1} \leq z' < z \leq c_l$ نتیجه می‌شود. بنابراین نتیجه می‌گیریم که d^l یک جواب شدنی است.

برای رسیدن به تناقض، کافی است نشان دهیم مقدار تابع هدف جواب شدنی d^l برابر z_l است. از آن‌جا که مقدار تابع هدف d' برابر z' است داریم:

$$c_{ij} - d'_{ij} \leq z', \quad \forall P \in \boldsymbol{P}, \exists (i,j) \in P,$$ (Error! No text of specified style in document.-۱۰)

$$c_{ij} - d'_{ij} = z', \quad \exists P_0 \in \boldsymbol{P}, \exists (i,j) \in P_0 \cap C',$$ (Error! No text of specified style in document.-۱۱)

که در این‌جا $\boldsymbol{P}$ مجموعه‌ی همه‌ی مسیرهای جهت‌دار در G از نقطه شروع به نقطه هدف است. توجه کنید که در تساوی (*Error! No text of specified style in document.*-۱۱) فرض اضافی $(i,j) \in C'$ را تحمیل کرده‌ایم. طبق نامساوی (پ) اگر تساوی (*Error! No text of specified style in document.*-۱۱) برای هیچ یالی از C' رخ ندهد، به ازای هر $(i,j) \in C'$ خواهیم داشت: $c_{ij} - d'_{ij} < z'$. پس می‌توان با کاهش مقادیر d'_{ij} ها به اندازه‌ی یک مقدار مثبت کوچک به یک جواب شدنی رسید که تساوی (*Error! No text of specified style in document.*-۱۱) حداقل برای یک یال رخ دهد. با جمع کردن طرفین نامساوی (*Error! No text of specified style in document.*-۱۰) و (*Error! No text of specified style in document.*-۱۱) با عدد ϵ داریم:

$$c_{ij} - d^l_{ij} \leq c_{ij} - (d'_{ij} - \epsilon) \leq z' + \epsilon = z_l, \quad \forall P \in \boldsymbol{P}, \exists (i,j) \in P,$$

$$c_{ij} - d^l_{ij} = c_{ij} - \left(d'_{ij} - \epsilon\right) = z' + \epsilon = z_l, \quad \exists P_0 \in \boldsymbol{P}, \exists (i,j) \in P_0 \cap C',$$

بنابراین، مقدار تابع هدف جواب شدنی d^l برابر z_l است که به تناقض می‌رسیم. $\square$

قضیه *Error! No text of specified style in document.*-۴. اگر مسأله‌ی ممانعت ایمن‌ترین مسیر فاقد جواب شدنی با مقدار تابع هدف $z \in [0, z_{max}]$ باشد، آن‌گاه فاقد جواب شدنی با مقدار تابع هدف کمتر از z است.

اثبات. اگر $z \in (c_{l-1}, c_l]$ آن‌گاه طبق لم قبل مسأله فاقد جواب شدنی در بازه‌ی $[c_{l-1}, z]$ است. حال قرار می‌دهیم: $z_{l-1} = c_{l-1}$. پس باز هم طبق لم قبل چون مسأله فاقد جواب شدنی با مقدار هدف z_{l-1} است پس همچنین، فاقد جواب شدنی با مقدار هدف $z \in (c_{l-2}, z_{l-1}]$ خواهد بود. با تکرار این روند حکم ثابت می‌شود. $\square$

نتیجه *Error! No text of specified style in document.*-۱. اگر مسأله‌ی ممانعت ایمن‌ترین مسیر فاقد جواب شدنی با مقدار تابع هدف z_{max} باشد، آن‌گاه مسأله نشدنی است.

اثبات. از آن‌جا که هر مقدار تابع هدف متعلق به بازه‌ی $[0, z_{max}]$ است، این نتیجه یک حکم فوری از قضیه *Error! No text of specified style in document.*-۴. اگر مسأله‌ی ممانعت ایمن‌ترین **جواب فاقد مسیر** شدنی با مقدار تابع هدف $z \in [0, z_{max}]$ باشد، **آن‌گاه** فاقد جواب شدنی با مقدار تابع هدف کمتر از z است. است. $\square$

فرض کنید $z \in [0, z_{max}]$ یک عدد حقیقی نامنفی دلخواه باشد. می‌خواهیم ببینیم که آیا می‌شود یک جواب شدنی d را به دست آورد که مقدار تابع هدفش برابر z باشد (از دیدگاهی دیگر، ایمنی هر مسیر حداکثر برابر z باشد). در این خصوص، برای هر یال $(i,j) \in A$، یک وزن u_{ij}^{z} معرفی می‌کنیم که هزینه‌ی تغییر ایمنی این یال را نشان می‌دهد. اگر $z \le c_{ij}$ باشد، نیازی به تغییر ایمنی یال نیست. پس برای این یال‌ها هزینه $u_{ij}^{z} = 0$ را در نظر می‌گیریم. اما اگر $c_{ij} > z$ باشد، باید ایمنی به اندازه‌ی $d_{ij} = c_{ij} - z$ کاهش یابد تا به مقدار z برسد. این تغییر، هزینه‌ی $u_{ij}^{z} = w_{ij}(c_{ij} - z)$ را در بر خواهد داشت. فرض کنید C کمترین برش شبکه‌ی $G(V, A, u^{z})$ باشد. اکنون اگر ظرفیت کمترین برش C از W کمتر باشد، جواب شدنی d^{z} که به صورت

$$d_{ij}^{z} = \begin{cases} 0 & (i,j) \notin C, \\ 0 & (i,j) \in C \wedge u_{ij}^{z} = 0, \\ c_{ij} - z & (i,j) \in C \wedge u_{ij}^{z} > 0, \end{cases} \qquad \forall (i,j) \in A,$$

(Error! No text of specified style in document.-۱۲)

تعریف می‌شود، حاصل می‌گردد. اما در غیر این صورت قید بودجه برقرار نبوده، پس نمی‌تواند d^{z} یک جوابی شدنی باشد. نتیجه‌ی حاصل در قضیه زیر آمده است.

قضیه *Error! No text of specified style in document.*-۵. به ازای هر $z \in [0, z_{max}]$، بردار d^{z} یک جواب شدنی از مسأله‌ی ممانعت ایمن‌ترین مسیر است اگر و تنها اگر ظرفیت کمترین برش شبکه‌ی $G(V, A, u^{z})$ کمتر یا مساوی W باشد.

اثبات. اثبات واضح است. $\square$

قضیه *Error! No text of specified style in document.*-۶. اگر مسأله دارای جواب شدنی d' با مقدار تابع هدف z باشد، آن‌گاه بردار d^{z} تعریف شده به وسیله‌ی (*Error! No text of specified style in document.*-۱۲) نیز یک جواب شدنی است.

اثبات. مجموعه‌ی $S = \{(i,j) \in A : c_{ij} - d'_{ij} \le z\}$ را در نظر بگیرید. این مجموعه از یال‌ها حداقل شامل یال‌های یک برش مانند C می‌شوند، زیرا در غیر این صورت می‌توان مسیری یافت که ایمنی‌اش بیشتر از z باشد که نتیجه می‌دهد مقدار تابع هدف d' از z بیشتر است. اکنون جواب d^{z} تعریف شده به وسیله‌ی (*Error! No text of specified style in document.*-۱۲)متناظر با این برش C را در نظر بگیرید. به سادگی می‌توان دید که به ازای هر $(i,j) \in A$ نامساوی $d_{ij}^{z} \le d'_{ij}$ برقرار است. پس با استفاده از این نامساوی d^{z} در قید بودجه و در محدودیت $d^{z} \le c$ صدق می‌کند. بنابراین d^{z} یک جواب شدنی از مسأله ممانعت ایمن‌ترین مسیر است. $\square$

یک نتیجه‌ی مورد توجه از قضیه *Error! No text of specified style* in *document.*-۶. اگر مسأله **دارای** جواب شدنی d' مقدار با تابع هدف z باشد، آن‌گاه بردار dz تعریف شده به وسیله‌ی (*Error! No text of specified style in document.*-۱۲) نیز یک جواب شدنی است.، این است که برای یافتن مقدار بهینه‌ی مسأله‌ی ممانعت ایمن‌ترین مسیر می‌توان خود را محدود به جواب‌های d^{z} تعریف شده به وسیله‌ی (*Error! No text of specified style in document.*-۱۲) کرد و کوچکترین مقدار $z \in [0, z_{max}]$ را یافت به طوری که d^{z} شدنی باشد و یا معادلا ظرفیت کمترین برش شبکه‌ی $G(V, A, u^{z})$ کمتر یا مساوی W باشد. اکنون آماده‌ایم تا فاز اول الگوریتم را توضیح دهیم. در فاز

اول در جستجوی بازه‌ای مانند $[c_{l-1}, c_l)$ هستیم که شامل مقدار بهینه‌ی z^* مسأله شود. بدین منظور اندیسی مانند $l \in \{1, \dots, k\}$ را جستجو می‌کنیم که به ازای آن، مسأله‌ی ممانعت ایمن‌ترین مسیر دارای جوابی شدنی با مقدار تابع هدف $z = c_l$ بوده اما فاقد جواب شدنی با مقدار تابع هدف $z = c_{l-1}$ باشد. بنابراین، در فاز اول اندیسی مانند $l \in \{1, \dots, k\}$ را خواهیم یافت به طوری که ظرفیت کمترین برش در شبکه‌ی $G(V, A, u^{c_l})$ کمتر یا مساوی W بوده در حالی که در شبکه‌ی $G(V, A, u^{c_{l-1}})$ بیشتر از W باشد. برای یافتن اندیس l با این خاصیت کافی است از تکنیک جستجوی دودویی استفاده کنیم.

اکنون آماده‌ایم که در مورد روند فاز دوم بحث کنیم. در این فاز، ابتدا نشان می‌دهیم مسأله به یک مسأله‌ی کمترین نسبت برش کاهش می‌یابد. سپس مسأله‌ی حاصل را به روش نیوتن حل می‌کنیم.

در فاز اول بازه‌ی $[c_{l-1}, c_l)$ را به دست آوردیم که مقدار بهینه z^* متعلق به آن است. با توجه به قضیه *Error! No text of specified style in document.*-۶ اگر مسأله **دارای** جواب شدنی d^z مقدار با تابع هدف z باشد، آن‌گاه بردار dz تعریف شده به وسیله‌ی (*Error! No text of specified style in document.*-۱۲) نیز یک جواب شدنی است.، می‌دانیم که مسأله‌ی ایمن‌ترین مسیر دارای یک جواب بهینه d^{z^*} تعریف شده به وسیله‌ی (*Error! No text of specified style in document.*-۱۲) به ازای $z = z^* \in (c_{l-1}, c_l]$ است. پس کافی است جستجوی خود را برای یافتن جواب بهینه محدود به جواب‌های تعریف شده به صورت (*Error! No text of specified style in document.*-۱۲) کنیم که مقدار تابع هدفشان در بازه $[c_{l-1}, c_l)$ قرار گیرد. مجموعه‌ی

$$L = \{(i, j) \in A: c_{ij} \geq c_l\}$$

را در نظر بگیرید. این مجموعه شامل همه‌ی یال‌هایی می‌شود که می‌توان ایمنی آن‌ها را تنها برای به دست آوردن یک جواب شدنی با مقدار تابع هدف $z \in (c_{l-1}, c_l]$ تغییر داد. پس با توجه به فاز اول، مجموعه‌ی L به طور منحصر به فرد مشخص شده و در اجرای فاز دوم ثابت می‌ماند. یک جواب دلخواه d^z متناظر با برش C را در نظر بگیرید. به وسیله‌ی تعریف مجموعه‌ی Lمی‌توان این جواب را به شکل زیر بازنویسی کرد:

$$d_{ij}^z = \begin{cases} 0 & (i, j) \notin C, \\ 0 & (i, j) \in C \backslash L, \\ c_{ij} - z & (i, j) \in C \cap L, \end{cases} \qquad \forall (i, j) \in A, \tag{*Error! No text of specified style in document.*-۱۳}$$

که در آن z مقداری است که به ازای آن محدودیت بودجه به شکل تساوی برقرار باشد. پس مقدار z به طور یکتا به شکل زیر محاسبه می‌شود:

$$\sum_{(i,j) \in A} w_{ij} d_{ij}^z = W \Longrightarrow$$
$$\sum_{(i,j) \in C \cap L} w_{ij} (c_{ij} - z) = W \Longrightarrow$$
$$z = \frac{\sum_{(i,j) \in C \cap L} w_{ij} c_{ij} - W}{\sum_{(i,j) \in C \cap L} w_{ij}}.$$

با توجه به این که هدف مینیمم سازی مقدار z است، می‌توان مسأله‌ی ممانعت ایمن‌ترین‌توان مسیر را به شکل زیر کاهش داد:

$$min_{c\in \mathbb{C}}\ z = \frac{\sum_{(i,j)\in C\cap L} w_{ij}c_{ij} - W}{\sum_{(i,j)\in C\cap\ L} w_{ij}},$$ (Error! No text of specified style in document.-۱۴)

که مشابه با قبل $\mathbb{C}$ نشان دهنده‌ی مجموعه‌ی برش‌هایی از شبکه‌ی G است که نقاط شروع و هدف را از هم جدا می‌کند. برای حذف مجموعه‌ی L در فرمول‌بندی مسأله، شبکه‌ی $G^L(V, A, c^L, w^L)$ را معرفی می‌کنیم که در آن

$$c_{ij}^L = \begin{cases} c_{ij} & (i,j)\in L, \\ 0 & (i,j)\notin L, \end{cases} \quad \forall (i,j)\in A,$$

$$w_{ij}^L = \begin{cases} w_{ij} & (i,j)\in L, \\ 0 & (i,j)\notin L, \end{cases} \quad \forall (i,j)\in A.$$

با این تعریف می‌توان مسأله‌ی را به صورت زیر بازنویسی کرد:

$$min_{c\in \mathbb{C}^L}\ z = \frac{\sum_{(i,j)\in C} w_{ij}c_{ij} - W}{\sum_{(i,j)\in C} w_{ij}},$$ (Error! No text of specified style in document.-۱۵)

که $\mathbb{C}^L$ نشان دهنده‌ی مجموعه‌ی همه‌ی برش‌هایی از شبکه‌ی G^L است که نقاط شروع و هدف را از هم جدا می‌کند. در این مسأله هدف یافتن برشی از شبکه G^L است که مقدار کسر $\frac{\sum_{(i,j)\in C} w_{ij}c_{ij}-W}{\sum_{(i,j)\in C} w_{ij}}$ را مینیمم کند. این مسأله یک نوع خاص از مسایل بهینه‌سازی ترکیبیاتی کسری* است که آن را مسأله‌ی مینیمم برش کسری† نامند. ما برای حل این مسأله از روش نیوتن که توسط رادزیک‡ [7] برای حل مسأله‌ی مینیمم برش کسری پیشنهاد شد، استفاده می‌کنیم.

مسأله‌ی بهینه‌سازی پارامتری $h(z) = min\{\sum_{(i,j)\in C} w_{ij}(c_{ij} - z):\ C\in \mathbb{C}^L\}$ را در نظر بگیرید. این مسأله ارتباط نزدیکی با مسأله‌ی (*Error! No text of specified style in document.*-۱۵) دارد که در قضیه زیر آن را بیان می‌کنیم.

قضیه *Error! No text of specified style in document.*-۷. فرض کنید C_{z^*} جواب بهینه‌ی مسأله‌ی پارامتری به ازای پارامتر z^* باشد. اگر $h(z^*) = W$، آن‌گاه z^* مقدار بهینه‌ی مسأله‌ی (*Error! No text of specified style in document.*-۱۵) و C_{z^*} جواب بهینه‌ی آن است.

اثبات. با توجه به بهینگی C_{z^*}، به ازای هر $C\in \mathbb{C}^L$ داریم:

* Fractional combinatorial optimization problem
† Fractional minimum cut problem
‡ Radzik

$$\sum_{(i,j)\in C} w_{ij}(c_{ij}-z^*) \geq W = \sum_{(i,j)\in C_{z^*}} w_{ij}(c_{ij}-z^*) \Rightarrow$$
$$\sum_{(i,j)\in C} w_{ij}(c_{ij}-z^*) - W \geq 0 = \sum_{(i,j)\in C_{z^*}} w_{ij}(c_{ij}-z^*) - W \Rightarrow$$
$$\frac{\sum_{(i,j)\in C} w_{ij}c_{ij} - W}{\sum_{(i,j)\in C} w_{ij}} \geq z^* = \frac{\sum_{(i,j)\in C_{z^*}} w_{ij}c_{ij} - W}{\sum_{(i,j)\in C_{z^*}} w_{ij}}$$

که نشان می‌دهد C_{z^*} جواب بهینه‌ی مسأله‌ی (*Error! No text of specified style in document.*-۱۵) با مقدار تابع هدف z^* است.□

از قضیه فوق می‌توان به جای حل مسأله‌ی (*Error! No text of specified style in document.*-۱۵) معادله‌ی $h(z) = W$ را حل کرد که برای این کار از روش نیوتن استفاده می‌کنیم.

فرض کنید z_0 یک تقریب اولیه از مقدار تابع هدف z^* از مسأله‌ی (*Error! No text of specified style in document.*-۱۵) باشد به طوری که $z^* \leq z_0$. با توجه به این که $z^* \in (c_{l-1}, c_l]$ می‌توان $z_0 = c_l$ را اختیار کرد. الگوریتم با شروع از نقطه‌ی z_0 یک دنباله اعداد مانند $z_0 > z_1 > z_2 > \cdots$ را می‌یابد که همگرا به z^* است. چنانچه z_k محاسبه شده باشد و $h(z_k) = W$ آن‌گاه جواب بهینه را یافته‌ایم. در غیر این صورت لزوما داریم:

$$h(z_k) < W \Rightarrow$$
$$\sum_{(i,j)\in C_k} w_{ij}(c_{ij}-z_k) < W \Rightarrow$$
$$\frac{\sum_{(i,j)\in C_k} w_{ij}c_{ij} - W}{\sum_{(i,j)\in C_k} w_{ij}} < z_k$$

که عبارت سمت چپ نامساوی اخیر را برابر z_{k+1} در نظر گرفته و $h(z_{k+1})$ را محاسبه می‌کنیم. مراحل را تا رسیدن به بهینگی ادامه می‌دهیم. واضح است که $z_{k+1} < z_k$. توجه کنید حالتی که $h(z_k) > W$ هیچ‌گاه رخ نمی‌دهد.

رادزیک ثابت کرد که در حداکثر n تکرار از روش نیوتن، مسأله کمترین برش کسری به جواب بهینه می‌رسد [7]. پس فاز دوم حداکثر نیاز به $O(n)$ تکرار دارد که در هر تکرار بایستی یک مسأله کمترین برش $h(z_k)$ را حل کرد. به جای حل مسأله‌ی کمترین برش می‌توان مسأله‌ی دوگان آن را یعنی مسأله‌ی ماکزیمم جریان را حل کرد که در زمان $O(n^2m)$ قابل حل است. بنابراین فاز دوم الگوریتم در زمان $O(n^3m)$ اجرا می‌شود. یادآوری می‌کنیم که فاز اول الگوریتم از جستجوی دودویی استفاده می‌کند و در هر مرحله باز هم نیاز به حل یک مسأله‌ی کمترین برش دارد. بنابراین پیچیدگی فاز اول برابر $O(n^2m\ log\ m) = O(n^2mlog\ n)$ است.

در پایان فاز دوم یک جواب بهینه‌ی C^* با مقدار بهینه‌ی z^* از حل مسأله‌ی کمترین برش کسری حاصل می‌شود. پس از آن، در سومین فاز بایستی جواب بهینه‌ی مسأله‌ی ممانعت ایمن‌ترین مسیر را محاسبه کنیم. جواب بهینه‌ی مسأله همان d^{z^*} که به وسیله‌ی (*Error! No text of specified style in document.*-۸) به ازای $z = z^*$ و $C = C^*$ به دست می‌آید. بنابراین فاز سوم فقط نیاز به $O(m)$ زمان دارد. پس توانستیم، قضیه زیر را ثابت کنیم.

قضیه *Error! No text of specified style in document.*-۸. مسأله‌ی ممانعت ایمن‌ترین مسیر در زمان $O(n^3m)$ قابل حل است.

قدردانی

این کار بخشی از پروژه‌ی جایگزین خدمت جواد طیبی با همکاری مهندس ابومسلم محمدی در دانشگاه افسری امام علی (ع) است. نویسندگان لازم می‌دانند که در این‌جا از زحمات و همکاری معاونت محترم پژوهشی دانشگاه و همکاران دیگر این حوزه تشکر و قدردانی نمایند.

مراجع:

[1] J. E. Ramirez-Marquez, "A bi-objective approach for shortest-path network interdiction," *Computers & Industrial Engineering,* vol. 59, no. 2, pp. 232-240, 2010.

[2] S. Sadeghi, A. Seifi and E. Azizi, "Trilevel shortest path network interdiction with partial fortification," *Computers & Industrial Engineering,* pp. 400-411, 2017.

[3] J. A. Sefair and C. Smith, "Dynamic shortest-path interdiction," *Networks,* vol. 68, no. 4, pp. 315-330, 2016.

[4] H. Stackelberg, The theory of the market economy, New York: Oxford University Press, 1952.

[5] R. K. Ahuja, T. L. Magnanti and J. B. Orlin, Network Flows, Englewood Cliffs: rentice Hall, 1993.

[6] E. W. Dijkstra, "A note on two problems in connexion with graphs," *Numerische mathematik,* vol. 1, no. 1, pp. 269-271, 1959.

[7] Radzik, "Fractional combinatorial optimization," in *Handbook of combinatorial optimization*, New York, Springer, 2013, pp. 1311-1355.