بررسی و مقایسه الگوریتم‌های کوتاهترین مسیر روی سطوح نامنظم مثلث‌بندی شده وزن‌دار

حوا علیزاده نوقابی[1]، فرزانه غیور باغبانی[2]

[1]گروه کامپیوتر، مجتمع آموزش عالی گناباد، alizadehn_h@yahoo.com

[2]گروه کامپیوتر، دانشگاه تهران، f.ghayour@ut.ac.ir

چکیده

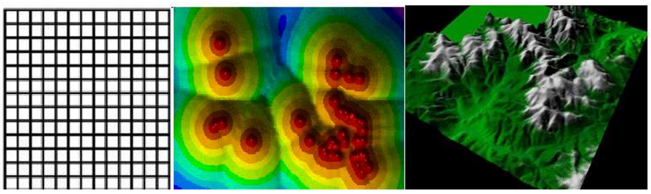
مساله کوتاهترین مسیر از مسایل مهم در نظریه گراف و هندسه محاسباتی می‌باشد. این مساله کاربردهای قابل توجهی در زمینه‌های گسترده‌ای همچون مسیریابی بسته‌ها در شبکه، تعیین مسیر حرکت روبات، طراحی نقشه‌ها و سیستم‌های اطلاعات جغرافیایی دارد. از این رو طی سالیان مختلف موردتوجه محققان قرار گرفته است. در این مقاله به مساله کوتاهترین مسیر در نظریه گراف و کارهای پیشین آن پرداخته شده و همچنین به طور مجزا مساله کوتاهترین مسیر در هندسه محاسباتی و در دو قسمت فضای دوبعدی و سه بعدی بررسی می‌شود. سپس الگوریتم‌های کوتاهترین مسیر روی سطوح نامنظم مثلث‌بندی شده وزن‌دار به طور جزئی معرفی و مقایسه می‌شوند.

واژه‌های کلیدی

مساله کوتاهترین مسیر در نظریه گراف، مساله کوتاهترین مسیر در هندسه محاسباتی، سطوح نامنظم مثلث‌بندی شده وزن‌دار، الگوریتم کوتاهترین مسیر روی سطوح نامنظم

۱- مقدمه

سطح نامنظم مثلث‌بندی شده، ساختاری در هندسه محاسباتی است که کاربرد بیشتر آن در سیستم‌های اطلاعات جغرافیایی و برای نگهداری تقریبی اطلاعات یک سرزمین[1] است. سرزمین سطحی دو و نیم بعدی است، به این معنی که برای هر نقطه (x,y) از صفحه دقیقا یک ارتفاع z در نظر گرفته می‌شود. شکل ۱-الف یک سرزمین را نشان می‌دهد. این دید، دیدی است که از ماهواره از سطح زمین بدست می‌آید. به علت پیوسته بودن سطح، نگهداری اطلاعات تمام نقاط ممکن نیست. از این رو برخی از این نقاط به عنوان نمونه انتخاب و نگهداری می‌شوند و دیگر نقاط با استفاده از نقاط نمونه برداری شده، تقریب زده می‌شوند.



الف　　　　ب　　　　پ

شکل ۱-الف-یک سرزمین، ب-مدل خطوط کرانه و پ- شبکه منظم‌مربعی

برای نمونه برداری از سطح زمین، اغلب از اشعه لیزر استفاده می‌شود. هواپیما با سرعت کنترل شده، سطح را پیمایش کرده و به فواصل مشخص، اشعه لیزر را ارسال می‌کند و با توجه به زمان رفت و برگشت، ارتفاع سطح را محاسبه می‌کند. برای تقریب سطح از روی نقاط نمونه برداری شده از مدلهای متنوعی همچون خطوط کرانه[2]، شبکه‌های منظم مربعی[3] و سطوح نامنظم مثلث‌بندی شده استفاده می‌شود. شکل ۱-ب مدل خطوط کرانه را نشان می‌دهد. در این مدل نقاط هم‌ارتفاع، به صورت منحنی‌های بسته نمایش داده می‌شوند. استفاده از این مدل پیچیدگی‌های خاص خود را دارد. به عنوان مثال منحنی بدست آمده برای ارتفاع‌ها خود نامنظم هستند و چگونگی ذخیره آنها خود مساله دیگری است. در مدل شبکه‌های منظم مربعی، نقاط نمونه‌برداری به صورت یکنواخت در صفحه قرار دارند. به این صورت که یک شبکه مربعی را تشکیل می‌دهند(شکل ۱-پ). این روش ساده به علت در نظر نگرفتن ناهمگونی‌های سطح چندان مناسب نیست.

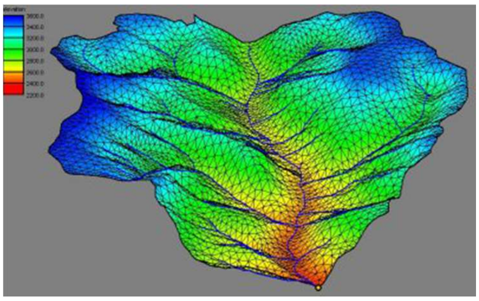
روشی که اغلب امروزه استفاده می‌شود، سطوح نامنظم مثلث‌بندی‌شده است. در این مدل، نقاط نمونه‌برداری، مثلث‌بندی می‌شوند و از این مثلث‌بندی برای تقریب سطح استفاده می‌شود. چگونگی مثلث‌بندی در

[1] terrain

[2] Countour Linesl

[3] Regular Square Grid

دقت تقریب سطح تاثیر مهمی دارد. مثلث‌بندی دلونی[4] با ماکزیمم کردن تمام مثلث‌ها، روشی متداول است. شکل ۲ یک سطح نامنظم مثلث‌بندی شده را نشان می‌دهد. هر یک از مثلث‌ها ممکن است پدیده طبیعی خاصی را نشان دهد، برخی دریاچه، برخی جنگل، منطقه شهری یا موارد دیگر. این تفاوت‌ها را می‌توان با تخصیص وزن به مثلث‌ها نشان داد.



شکل ۲- سطح نامنظم مثلث‌بندی شده

در ادامه این مقاله در بخش بعدی به مساله کوتاهترین مسیر در نظریه گراف‌ها و کارهای مرتبط پیشین پرداخته می‌شود. سپس در بخش دیگر مساله کوتاهترین مسیر در هندسه محاسباتی و کارهای مرتبط پیشین آن توضیح داده می‌شود که این بخش خود در دو قسمت مجزای کوتاهترین مسیر در فضای دوبعدی و کوتاهترین مسیر در فضای سه‌بعدی شرح داده می‌شود. و پس از آن در بخش بعد معرفی و مقایسه الگوریتم‌های کوتاهترین مسیر روی سطوح نامنظم مثلث‌بندی‌شده وزن‌دار آورده می‌شود.

۲- مساله کوتاهترین مسیر در نظریه گراف‌ها

ساده‌ترین حالت مساله کوتاهترین مسیر، پیدا کردن کوتاهترین مسیر در گراف‌هاست که انواع مختلفی دارد همچون کوتاهترین مسیر میان زوج راس مبدا و مقصد، کوتاهترین مسیر از مبدا واحد و کوتاهترین مسیر میان تمام زوج راس‌ها. الگوریتم دایکسترا، کوتاهترین مسیر از مبدا واحد را محاسبه می‌کند. [1] بسته به نحوه پیاده‌سازی، الگوریتم دایکسترا مرتبه زمانی $O(v^2)$ ، $O(e \log v)$ و $O(e + \log v)$ دارد که v تعداد رئوس گراف و e تعداد یال‌های گراف است. برای حالت خاص گراف‌های مسطح[5] با یال‌های نامنفی در سال ۱۹۹۷، Henzinger و دیگران الگوریتم خطی ارائه دادند که پایه این الگوریتم همان الگوریتم دایکسترا است.[2] این الگوریتم با تکرار ناحیه‌بندی به اندازه کافی با استفاده از جداکننده‌ها، به مرتبه زمانی خطی دست پیدا کرد. همچنین Throup در سال ۱۹۹۷ الگوریتمی از مرتبه $O(e)$ برای گراف‌های کلی با وزن یال صحیح ارائه کرده است که پایه این الگوریتم نیز مشاله دایکسترا است با این تفاوت که ترتیب ملاقات رئوس متفاوت است.[3]

مساله پیدا کردن کوتاهترین مسیر میان تمام زوج راس‌ها نیز موردتوجه بوده است. معروفترین الگوریتم برای این مساله الگوریتم Floyd-Warshall با زمان اجرای $O(v^3)$ می‌باشد. [4] الگوریتم بهینه‌تر، الگوریتم Johnson با مرتبه زمانی $O(v^2 \log v + ve)$ است.[5] Chan در سال ۲۰۰۶ الگوریتمی از مرتبه $O(ve)$ برای گراف‌های بدون جهت و بدون وزن ارائه داد.[6]

کلیه الگوریتم‌های کوتاهترین مسیر که تاکنون ارائه شده است را می‌توان در دو دسته به نام‌های انتساب برچسب[6] و تصحیح برچسب[7] قرار داد. در روش‌های انتساب برچسب مانند دایکسترا هر راس برچسبی دارد که در ابتدا مقدار بینهایت دارد و نشان‌دهنده حدبالایی از کوتاهترین مسیر از مبدا تا آن راس است. رئوس به سه دسته ملاقات‌نشده، در انتظار و محاسبه‌شده تقسیم می‌شوند. در این روش‌ها تنها از یالی که یک سر آن در مجموعه محاسبه شده باشد می‌توان به‌روزرسانی برچسب‌ها را انجام داد و در طی روند الگوریتم رئوس بتدریج و یکی پس از دیگری برچسب نهایی خود را دریافت می‌کنند. اما در روش‌های تصحیح برچسب مانند بلمنفورد[8] به‌روزرسانی برچسب هر راس تا انتهای الگوریتم نیز می‌تواند ادامه داشته باشد.

۳- مساله کوتاهترین مسیر در هندسه محاسباتی

۳-۱- کوتاهترین مسیر در فضای دوبعدی

ساده‌ترین حالت مساله پیدا کردن کوتاهترین مسیر در هندسه محاسباتی، یافتن کوتاهترین مسیر در دو بعد است. در حالت دوبعدی بی وزن، چند مانع[9] که اغلب چندضلعی هستند در صفحه وجود دارند و هدف یافتن کوتاهترین مسیر بین دو نقطه از سطح است به گونه ای که مسیر با موانع برخورد نداشته باشد. در حالت وزن‌دار صفحه به چندضلعی‌هایی تقسیم شده که هزینه عبور از واحد طول برای هر ناحیه به صورت وزن آن ناحیه بیان می‌شود. برای موانع وزن بینهایت در نظر گرفته می‌شود. برای تمام حالات دوبعدی مساله که نواحی (یا موانع) به صورت چندضلعی هستند، مساله به راحتی به مساله کوتاهترین مسیر در گراف‌ها کاهش پیدا می‌کند. به این ترتیب که اثبات می‌شود مسیر بهینه تنها در نقاط چندضلعی‌ها ممکن است تغییر جهت داشته باشد و با تشکیل گراف دید نقاط متناظر رئوس چندضلعی‌ها به عنوان رئوس گراف درنظر گرفته می‌شود و سپس میان رئوسی که پاره خط مستقیم واصل میان آنها به مانعی برخورد نکند، یال رسم می‌شود. الگوریتم‌های ساخت گراف دید متنوعی ارائه شده‌اند. ساده‌ترین آنها مرتبه زمانی $O(v^2 \log v)$ دارد. الگوریتم با مرتبه زمانی $O(v^2)$ توسط Welz ارائه شد.[7] الگوریتم‌های حساس به خروجی مانند الگوریتم Vetger و Pocchiola با مرتبه زمانی $O(v + v \operatorname{Log} v)$ و حافظه $O(v)$ نیز وجود دارند.[8]

[4] Delunay Triangulation

[5] planar

[6] label setting

[7] Label correcting

[8] Bellmanford

[9] obstacle

در روش گراف دید ممکن است طول کوتاهترین مسیر موردنظر بسیار کوچکتر از اندازه گراف دید ساخته شده باشد و ما هزینه اضافی بابت تشکیل گراف دید متحمل شویم. توجه کنید که اندازه گراف دید از مرتبه $O(v^2)$ و اندازه کوتاهترین مسیر از مرتبه $O(v)$ است. Suri و Hershberger الگوریتم دایکسترای پیوسته را با زمان اجرای $O(v\log v)$ در سال ۱۹۹۵ ارائه کردند.[9] در روش دایکسترای پیوسته، نقشه کوتاهترین مسیر به طور مستقیم ساخته می‌شود.

۳-۲- کوتاهترین مسیر در فضای سه بعدی

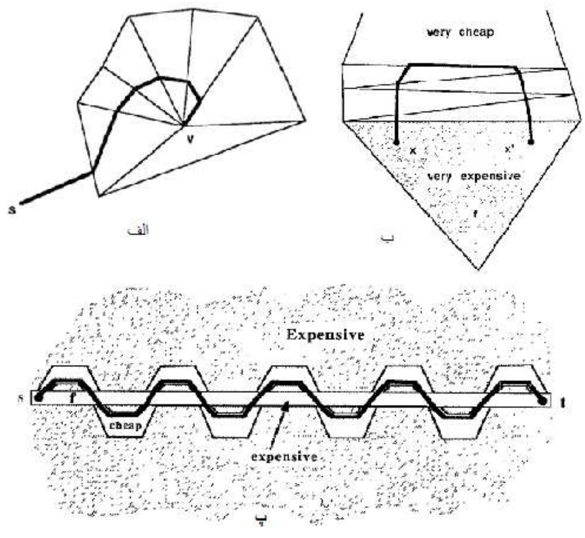
یافتن کوتاهترین مسیر در سه بعد و ابعاد بیشتر، بسیار مشکل‌تر از حالت دوبعدی مساله است. حتی حالت بی‌وزن مساله و با موانع چندضلعی، مساله NP-Hard است. برای درک شهود علت سختی مساله می‌توان این نکته را بیان کرد که برخلاف حالت دوبعدی نقشه‌های کوتاهترین مسیر بر روی هیچ گراف گسسته‌ای قرار نمی‌گیرند و با وجود اینکه نقاط شکستگی مسیرهای بهینه بر روی یال‌های موانع واقع می‌شود ولی تعیین دقیق مکان شکستگی بر روی یال ساده نیست. از این رو تمرکز بیشتر بر روی یافتن الگوریتم‌های تقریبی و حل حالت‌های خاص مساله است.

یک حالت خاص پرکاربرد این مساله یافتن کوتاهترین مسیر بر روی سطح یک چندوجهی میان دو نقطه است. اولین بار Schorr و Sharir برای حالتی که چندوجهی محدب باشد الگوریتمی از مرتبه زمانی $O(v^3 \log v)$ پیش پردازش و زمان پرس‌وجو $O(k+\log v)$ ارائه کردند که k تعداد یال‌های کوتاهترین مسیر است.[10] الگوریتمی مشابه دایکسترای پیوسته توسط Han و Chen در سال ۱۹۹۶ با مرتبه زمانی $O(v^2)$ ارائه شد که برای چندوجهی غیرمحدب نیز قابل استفاده است.[11] ایده این الگوریتم به این صورت است که با شروع حرکت از راس مبدا هر زمان به یالی برخورد کردیم صفحه مجاور را طوری دوران می‌دهیم که در راستای صفحه فعلی قرار بگیرد. با تکرار این روند و رسم کوتاهترین مسیر اقلیدسی میان مبدا و مقصد جواب بدست می‌آید. پس از باز شدن صفحات و تصویر کردن آنها در یک سطح دوبعدی از آنجایی که نواحی بدون وزن هستند با رسم پاره‌خط میان دو نقطه مبدا و مقصد کوتاهترین مسیر بدست می‌آید. این الگوریتم در واقع برای هر راس مبدا یک ساختار بدست می‌آورد که با استفاده از آن میتوان هر پرس‌وجو که مبدا آن ثابت و مقصد متفاوت است را در زمان بسیار کوتاه پاسخ دهد. نکته‌ای که در این الگوریتم وجود دارد ترتیب انتخاب یال‌ها برای دوران است.

اولین بررسی کلی برای حالتی از مساله که هدف یافتن کوتاهترین مسیر بر روی سطح چندوجهی است و سطوح چندوجهی وزن‌دار هستند در سال ۱۹۹۱ توسط Mitchell و Papadimitriou انجام شد و الگوریتمی با ضریب تقریب $1+\epsilon$ و مرتبه زمانی $O(n^4L)$ ارائه دادند که در آن $L=\log(vNW/w\epsilon)$ ضریب دقت مساله است. N بزرگترین مختصات صحیح نقاط سطح و W و w بیشترین و کمترین وزن صحیح ناحیه‌ها و ϵ درجه دقت موردنظر است.[12] در بخش بعد به صورت مفصل به معرفی الگوریتم‌های ارائه شده برای این مساله (کوتاهترین مسیر روی سطوح وزن‌دار) پرداخته می‌شود.

۴- معرفی و مقایسه الگوریتم‌های کوتاهترین مسیر روی سطوح نامنظم مثلث‌بندی شده وزن‌دار

اولین بار در سال ۱۹۹۱، Mitchell و Papadimitriou مساله پیدا کردن کوتاهترین مسیر در سطوح وزن‌دار را بررسی و خصوصیات مسیرهای بهینه را به دقت بیان کردند. همچنین همانطور که گفته شد الگوریتمی بر مبنای دایکسترای پیوسته با ضریب تقریب $1+\epsilon$ و مرتبه زمانی $O(n^4L)$ ارائه دادند. آن‌ها ویژگی‌های مسیرهای بهینه محلی (مسیری که با تغییرات محلی طول آن کوتاهتر نشود) را بیان کردند و تفاوت‌های آن را با حالت بدون وزن برشمردند. تفاوت ۱: در مساله نواحی بدون وزن، کوتاهترین مسیر به یک راس تنها از یک ناحیه مجاور آن می‌تواند عبور کند اما در حالت وزن‌دار این ویژگی صادق نیست.(شکل ۳- الف) تفاوت ۲: در مساله نواحی بدون وزن، کوتاهترین مسیر میان دو نقطه در یک ناحیه، پاره خط مستقیم میان آنهاست. ولی در نواحی وزن‌دار ممکن است از نواحی دیگر نیز عبور کند.(شکب ۳-ب) تفاوت ۳: در مساله نواحی بی وزن، مسیر بهینه از هر ناحیه حداکثر یکبار می‌تواند عبور کند، اما در مساله نواحی وزن‌دار حداکثر $O(n)$ بار است.(شکل ۳-پ)

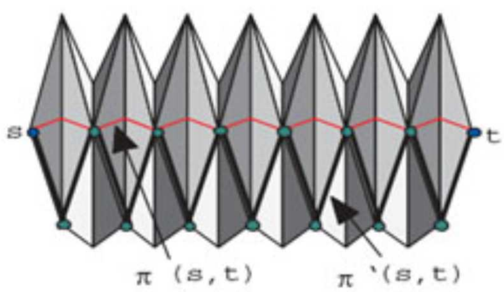


شکل ۳- الف- عبور کوتاهترین مسیر به یک راس از چندین ناحیه مجاور آن، ب- عبور کوتاهترین مسیر میان دو نقطه از یک ناحیه از نواحی دیگر، پ- $O(n)$ بار عبور مسیر بهینه از یک ناحیه

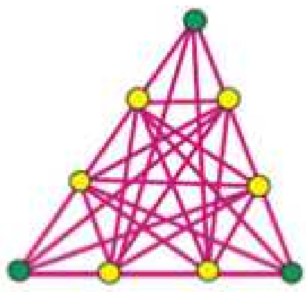
Aleksandrov و دیگران در سال ۱۹۹۷ ایده گراف تقریب تین را بیان کردند.[13] برای تشکیل گراف تقریب ابتدا به ازای هر راس تین یک راس در گراف قرار داده و رئوس را که در تین به هم یال دارند در گراف نیز به هم وصل می‌کنیم. وزن هر یال مینیمم وزن نواحی دو طرف یال تعریف می‌شود. سپس با استفاده از الگوریتم‌های کوتاهترین مسیر در گراف‌ها کوتاهترین مسیر بدست می‌آید. اگر از این گراف تقریب استفاده کنیم، ضریب تقریب برابر است با $2/\sin\Theta_{min}$ که در آن Θ_{min} اندازه کوچکترین

زاویه در مثلث‌بندی تین است. این ضریب تقریب به شکل مثلث‌بندی وابسته می‌شود و می‌تواند بسیار بد باشد. در شکل ۴ یک نمونه از عملکرد بد این الگوریتم دیده می‌شود. مسیر بهینه $\pi(s, t)$ است در حالیکه الگوریتم مسیر $\pi'(s, t)$ را پیدا می‌کند.

یک راهکار برای بهبود ضریب تقریب اضافه کردن نقاط کمکی به گراف تقریب است. به این صورت که بر روی هر یال تین تعدادی نقطه قرار داده و بین هر دو نقطه ای که بر روی محیط یک مثلث هستند اما بر روی یک ضلع قرار ندارند، یالی در گراف قرار می‌دهیم. شکل ۵ چگونگی اینکار را نشان می‌دهد. شیوه‌های مختلفی برای درج این نقاط کمکی وجود دارد. یک روش استفاده از نقاط کمکی ثابت است. به این صورت که بر روی هر یال m نقطه قرار می‌دهیم که m عددی ثابت است. اگر $m=n^2$ نقطه کمکی برای هر یال استفاده شود، حافظه موردنیاز $O(n^3)$ و مرتبه زمانی الگوریتم $O(n^5)$ است و طول مسیر بدست آمده حداکثر به اندازه $|W|L$ از مسیر بهینه بیشتر است. این روش اولین بار توسط Lanthier و دیگران پیشنهاد شد.[14]



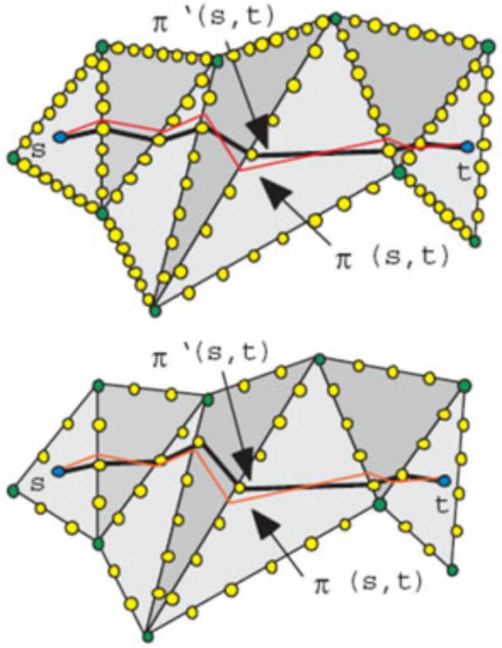
شکل ۴- نمونه عملکرد بد الگوریتم [14]



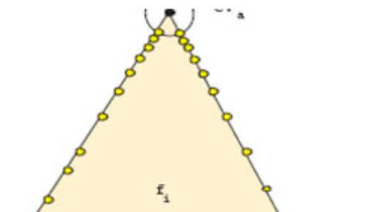
شکل ۵- افزودن نقاط کمکی

روش دیگر قرار دادن نقاط کمکی، بازه‌های ثابت است. یعنی فاصله نقاط بر روی یال‌ها ثابت در نظر گرفته می‌شود. به این ترتیب دقت مسیر بدست آمده مانند روش نقاط ثابت است و زمان اجرا و حافظه موردنیاز در بدترین حالت معادل روش نقاط ثابت است. ولی در عمل و در حالت میانگین حافظه و زمان اجرای کمتری مصرف می‌کند. علت این است که روش نقاط ثابت، میان یال‌های با طول کم و زیاد تفاوتی در تعداد نقاط کمکی نداشت و امکان داشت که در یال‌های کوتاه نقاط بسیار نزدیک به هم قرار بگیرند. در شکل ۶ این تفاوت را می‌توان ملاحظه کرد.

Aleksandrov با استفاده از ایده گراف تقریب و نقاط کمکی الگوریتمی ارائه داد.[13] شیوه او برای قرار دادن نقاط کمکی، روش بازه‌های نمایی است. در این روش با شروع از هر راس تین با پیش رفتن بر روی یال، فاصله هر نقطه تا نقطه قبلی دو برابر فاصله قبلی است. شکل ۷ این روش را نشان می‌دهد.



شکل ۶- مقایسه روش نقاط کمکی ثابت و بازه‌های ثابت[14]



شکل ۷- روش بازه‌های نمایی[13]

در جدول ۱ مقایسه چند الگوریتم کوتاهترین مسیر بر روی سطوح وزن‌دار آورده شده است. در جدول برای ساده‌نویسی از پارامتر $K=O(N^2W/\epsilon w)$ استفاده شده است که در آن N بزرگترین مختصات صحیح نقاط سطح، W و w به ترتیب بیشترین و کمترین وزن صحیح ناحیه‌ها و ϵ دقت الگوریتم است. همانطور که مشاهده می‌شود الگوریتم Aleksandrov از نظر مرتبه زمانی و دقت بر دیگر روش‌ها برتری دارد.

جدول ۱- مقایسه الگوریتم‌های کوتاهترین مسیر بر روی سطوح وزن‌دار

الگوریتم	مرتبه زمانی
Papadimitriou and Mitchell	$O(n^{۸}log(\frac{nK}{N})$
Lanthier	$O(n^{۵})$
Aleksandrov	$O(n(K \log K)log(nK \log K))$

۵- جمع بندی

با توجه به کاربردهای گسترده مساله کوتاهترین مسیر، در این مقاله به این موضوع پرداخته شد. الگوریتم‌های موجود برای این مساله در نظریه گراف معرفی شدند و همچنین الگوریتم‌های این مساله در هندسه محاسباتی با در نظر گرفتن فضا به صورت دو بعدی و سه بعدی مورد بررسی قرار گرفت. پس از آن به طور خاص الگوریتم‌های کوتاهترین مسیر روی سطوح نامنظم مثلث‌بندی شده وزن‌دار معرفی شدند و در نهایت به مقایسه آنها پرداخته شد.

مراجع

[1] E. W. Dijkstra, “A Note on Two Problems in Connection with Graphs”, Numerische Mathematik, no 1, pp. 269-271, 1959.

[2] M. R. Henzinger, P. Klein, S. Rao, S. Subramanian, “Faster Shortest-Path Algorithms for Planar Graphs”, Journal of Computer and System Sciences 55, no 1, pp. 3-23, 1997.

[3] M. Thorup, “Undirected single-source shortest paths with positive integer weights in linear time”, Journal of the ACM (JACM), Vol 46, Issue 3, pp. 362-394, 1999.

[4] T. H. Cormen, C. E. Leiserson, R. L. Rivest, C. Stein. “Introduction to Algorithms” ,3rd ed., MIT Press, 2009.

[5] D. B. Johnson, “Efficient Algorithms for Shortest Paths in Sparse Networks”, Journal of the ACM (JACM), Vol 24, Issue 1, pp. 1-13, 1977.

[6] T. M. Chan, “All-pairs shortest paths for unweighted undirected graphs in o(mn) time”, ACM Transactions on Algorithms (TALG) 8, no. 4, 2012.

[7] E. Welzl, “Constructing the visibility Graph for n Line Segments in $O(n^2)$ Time”, Information Processing Letters 20, no. 4, pp. 167-171, 1985.

[8] M. Pocchiola, G. Vegter, “Computing the Visibility Graph via Pseudotriangulations”, In Proceesing of 11th Annual ACM Symposim of Computational Geometry, pp.248-257, 1995.

[9] J. Hershberger, S. Suri, “An optimal Algorithm for Euclidean Shortest Paths in the Plane”, SIAM Journal on Computing 28, no. 6, pp. 2215-2256, 1999.

[10] M. Sharir, A. Schorr, “On Shortest Path in Polyhedral Spaces”, SIAM Journal of Computing 15, no. 1, pp. 193-215, 1986.

[11] J. Chen, Y. Han, “Shortest paths on a polyhedron, Part I: Computing shortest paths." International Journal of Computational Geometry & Applications 6, no. 2, pp. 127-144, 1996.

[12] J. S. B. Mitchell, C. H. Papadimitriou, “The Weighted Region Problem: Finding Shortest Path Trough a Weighted Planar Subdivision”, Journal of the ACM 38, no. 1, pp. 18-73, 1991.

[13] L. Aleksandrov, M. Lanthier, A. Maheshvari, J. R. Sack, “An ε approximation Algorithm for Weighted shortest Paths on Polyhedral Surfaces”, The 6th Scandinavian Workshop on Algorithm Theoty, pp. 11-22, 1998.

[14] M. Lanthier, A. Maheshwari, J.S.B. Sack, “Approximating Weighted Shortest Paths on Polyhedral Surfaces”, In Proceedings of the thirteenth annual symposium on Computational geometry, pp. 274-283, 1997.