

# استفاده از تبدیل فضا برای حل مسئله فروشنده دوره‌گرد با معیار فاصله جغرافیایی

رضا مرتضوی<sup>۱</sup>، استادیار

۱- دانشکده فنی و مهندسی - دانشگاه دامغان - دامغان - ایران - r\_mortazavi@du.ac.ir

**چکیده:** یکی از مسائل بهینه‌سازی مهم در حوزه الگوریتم‌های ترکیباتی، مسئله فروشنده دوره‌گرد است. با توجه به کاربردهای فراوان، حل این مسئله مورد توجه پژوهشگران است و به‌طور خاص به‌عنوان یک مسئله مهم تحلیل شبکه در فناوری‌های اطلاعات مکانی کاربرد دارد. علی‌رغم صورت ساده، حل کلی این مسئله از درجه پیچیدگی NP-سخت است. به‌همین جهت روش‌های ابتکاری زیادی در کاربردهای عملی پیشنهاد شده است. در این مقاله از روش تصویرسازی نقشه برای تبدیل فضا از نسخه‌ای از مسئله فروشنده دوره‌گرد با معیار فاصله جغرافیایی به نسخه ساده‌تر با معیار فاصله اقلیدسی استفاده شده است. در ادامه مقاله یکی از روش‌های ابتکاری حل این مسئله بهبود داده شده است. نتایج تجربی بر روی مجموعه داده‌های واقعی در مقایسه با روش‌های مشابه، نشان‌دهنده برتری روش پیشنهادی به‌لحاظ کیفیت پاسخ و زمان دستیابی به آن است.

**واژه‌های کلیدی:** مسئله فروشنده دوره‌گرد، تبدیل فضا، تصویرسازی نقشه، فاصله غیر اقلیدسی، بهینه‌سازی.

## Solving Geographic Travelling Salesman Problem Based on Space Transformation

R. Mortazavi, Assistant Professor<sup>1</sup>

1- School of Engineering, Damghan University, Damghan, Iran, r\_mortazavi@du.ac.ir

**Abstract:** One of the most important optimization problem in combinatorial algorithms is Travelling Salesman Problem. This is due to its practical usages, especially in geospatial information technology. It is proved that the problem complexity is NP-Hard, but many heuristic methods are presented for practical applications. In this paper, the space transformation method is used to transform a geographic instance of travelling salesman problem into a simpler Euclidean one, such that traditional heuristics can be applied more successfully. Additionally, a heuristic technique to solve the transformed instance is introduced. The results confirm the superiority of the proposed method in terms of both the tour length and running time of the method in comparison with similar techniques.

**Keywords:** Travelling Salesman Problem, space transformation, map projection, non-Euclidean distance, optimization.

تاریخ ارسال مقاله: ۱۳۹۵/۰۵/۱۲

تاریخ اصلاح مقاله: ۱۳۹۵/۰۷/۲۲

تاریخ پذیرش مقاله: ۱۳۹۵/۰۹/۱۲

نام نویسنده مسئول: رضا مرتضوی

نشانی نویسنده مسئول: ایران - دامغان - میدان دانشگاه - دانشگاه دامغان - دانشکده فنی و مهندسی - گروه مهندسی کامپیوتر

## ۱- مقدمه

مسئله فروشنده دوره گرد (TSP) یکی از مسائل معروف الگوریتم‌های ترکیباتی<sup>۱</sup> است. هدف این مسئله در بیانی تمثیلی بدین صورت است که فروشنده‌ای باید تعداد مشخصی شهر را با شروع از یکی از آن‌ها ملاقات کند و به نقطه شروع برگردد، به طوری که هر شهر دقیقاً یک بار مشاهده شود و طول مسیر طی شده کمینه باشد.

این مسئله اولین بار در ۱۹۳۰ به صورت ریاضی بیان شده است [۱] و در حوزه‌های مختلف مانند مسیریابی، شبکه‌های کامپیوتری، طراحی مدارات الکترونیکی و مدل‌سازی به کمک کامپیوتر [۲-۴] کاربرد دارد. مسئله TSP دارای قدمت و کاربردهای زیادی است. ولی در حالت کلی دارای راه حل دقیق و کارآمد در زمان چندجمله‌ای بر حسب تعداد شهرها ( $n$ ) نیست و در دسته مسائل NP- سخت تقسیم‌بندی می‌شود [۵]. بنابراین با توجه به اهمیت حل آن در کاربردهای عملی، روش‌های ابتکاری مختلفی برای حل تقریبی آن پیشنهاد شده است که سعی می‌کنند در زمان قابل قبولی جواب نزدیک به بهینه را بیابند [۶].

با استفاده از روش‌های بهینه‌سازی مبتنی بر تقسیم و حل با شروع از جواب‌های نزدیک به بهینه می‌توان راحت‌تر به جواب‌های بهینه رسید [۱]. یکی از معروف‌ترین ابزارها در این بخش Concorde<sup>۲</sup> است که برای دستیابی به حل بهینه از روش شاخه و برش استفاده می‌کند. این ابزار می‌تواند مسائلی به بزرگی ۸۵۹۰۰ شهر را در مدت ۱۳۶ پردازنده-سال حل کند [۷] و برای نسخه‌هایی با چند میلیون شهر نیز جواب آن در نزدیکی جواب بهینه است [۶]. دستیابی به جواب اولیه نزدیک به بهینه در زمان معقول، از اهمیت بالایی در حل این مسئله برخوردار است.

یکی از روش‌های موفق برای حل مسئله فروشنده دوره گرد با عنوان الگوریتم ابتکاری پس‌انداز شناخته می‌شود [۸]. اما نسخه اولیه از این الگوریتم برای مجموعه داده‌های بزرگ قابل استفاده نیست و با محدودیت حافظه و زمان اجرا مواجه است. بهبودهایی از این الگوریتم برای حل این مشکلات پیشنهاد شده است که مبتنی بر استفاده از ساختمان داده‌ای با نام kd-tree [۹] است [۱۰]. ولی این بهبودها برای نمونه‌هایی از مسئله با معیار فاصله اقلیدسی ارائه شده‌اند [۱۱-۱۲].

نسخه‌هایی از مسئله فروشنده دوره گرد در فضای غیر اقلیدسی تعریف می‌شوند. به بیان دیگر با توجه به کاربرد، از توابعی خاص برای محاسبه فاصله بین گره‌ها استفاده می‌شود [۱۳-۱۵]. برای مثال، Faigl و همکاران برای محاسبه فاصله بین دو گره به دلیل وجود موانع میانی، طولی بیش از مسافت مستقیم بین آن گره‌ها تعریف کرده‌اند و از نقشه‌های خود سازمانده برای حل مسئله استفاده کرده‌اند [۱۳]. به عنوان مثالی دیگر، در مسئله فروشنده دوره گرد با معیار فاصله جغرافیایی (GEOTSP)، با فرض قرار گرفتن شهرها بر روی کره زمین، فاصله جغرافیایی آن‌ها بر اساس

طول و عرض جغرافیایی آن‌ها محاسبه و منظور می‌شود. Curtin ضمن اشاره به کاربردهای گسترده GEOTSP در مسائل تحلیل شبکه، به استفاده از روش‌های برنامه‌ریزی عدد صحیح برای حل نسخه‌های با اندازه متوسط این مسئله (بر حسب تعداد شهرها) در سیستم‌های اطلاعات جغرافیایی<sup>۳</sup> اشاره کرده است [۱۴]. علی‌رغم اینکه نسخه‌های اقلیدسی TSP نیز برای حل مشکل هستند، اما راه‌حل‌های ابتکاری برای مواجهه با آن‌ها موفق‌تر هستند [۱۶]. با توجه به ویژگی‌های نامتعارف در فضاهای غیر اقلیدسی معمولاً نمی‌توان از روش‌های ابتکاری رایج در فضای اقلیدسی مانند الگوریتم ابتکاری پس‌انداز برای تسریع یا بهبود حل استفاده کرد. بنابراین حل این نسخه‌ها به دلیل نیاز به استفاده از حل‌کننده‌های عمومی‌تر نیاز به صرف منابع بیشتری دارد [۱۴]. برای مثال حل نسخه GEOTSP در سطح جهان با عنوان World TSP به روش تقریبی به ۲۵۶ روز زمان نیاز داشته است [۴].

در این مقاله برای حل مشکل فاصله غیر اقلیدسی برای نمونه‌های با معیار فاصله جغرافیایی، از روش تصویرسازی نقشه<sup>۴</sup> استفاده شده است تا فضای مسئله را از جغرافیایی (روی کره) به اقلیدسی (در صفحه) تقریب بزنند. این روش برای حل نمونه‌هایی در حد یک کشور با حداکثر  $n = 7109$  شهر به کار رفته است<sup>۵</sup>. در حد اطلاع نویسنده مطالعه این روش در قالب چارچوب ارائه شده در این مقاله و در گستره جغرافیایی کل زمین صورت نگرفته است. در ادامه بهبودی در الگوریتم ابتکاری پس‌انداز [۸] همراه با پیش‌پردازشی اضافی نسبت به [۱۷] ارائه شده است که به بهبود کیفیت جواب نهایی منجر می‌شود. نهایتاً این روش بر روی نمونه‌هایی از مسئله GEOTSP با اندازه‌های متفاوت و نیز نمونه‌های اقلیدسی ارزیابی شده است و نتایج به لحاظ کیفیت جواب نهایی و زمان اجرا ارائه و مقایسه شده است.

ادامه این مقاله به صورت زیر بخش‌بندی شده است: در بخش ۲ مفاهیم پایه بیان می‌شوند. این مفاهیم شامل بیان رسمی GEOTSP است که در این مقاله بررسی شده است. همچنین روش‌های ابتکاری حل TSP که در بخش مقایسه به آن‌ها اشاره شده است، معرفی می‌شوند. در ادامه نسخه اولیه روش ابتکاری پس‌انداز تشریح می‌شود که برای حل TSP در فضای اقلیدسی در این مطالعه بهبود داده شده است. در بخش ۳ روش پیشنهادی ارائه شده است و در ادامه در بخش ۴ نتایج تجربی حاصل از ارزیابی روش ارائه می‌شود. بخش ۵ نیز به نتیجه‌گیری و پیشنهادهایی برای ادامه کار اختصاص دارد.

## ۲- مفاهیم پایه

در این بخش ابتدا مسئله فروشنده دوره گرد با معیار فاصله جغرافیایی به صورت رسمی بیان می‌شود. سپس روش‌های ابتکاری حل TSP به اجمال مرور می‌شود. در ادامه روش ابتکاری پس‌انداز توضیح داده می‌شود که در

برای بهبود آن انجام نمی‌شود و یال‌ها بدون تغییر می‌مانند. در روش‌های بهبود تدریجی، با شروع از یک جواب اولیه (مثلاً با استفاده از روش‌های ساختنی) به دفعات با تغییر یال‌ها سعی می‌شود دور نهایی بهبود یابد. به‌طور کلی روش‌های ساختنی خیلی سریع‌تر از روش‌های بهبود تدریجی هستند ولی معمولاً کیفیت جواب پایین‌تری نسبت به روش‌های بهبود تکراری دارند [۱۸].

روش‌های ساختنی شامل ابتکارات مبتنی بر نزدیک‌ترین همسایه (NN)، ابتکارهای درج یال [۷]، ابتکارهای مبتنی بر درخت‌های پوشا، ابتکار حریصانه (GH) [۱۹]، MTS [۱۲] و ابتکار پس‌انداز می‌شود. در روش NN، با شروع از یک گره، نزدیک‌ترین همسایه بعدی از گره جاری که هنوز مشاهده نشده به انتهای دور کنونی اضافه می‌شود. در روش GH، همواره کوتاه‌ترین یال به دور ناکامل جاری افزوده می‌شوند، به شرط اینکه این افزودن باعث ایجاد زیردور نشود و درجه هر دو گره این کوتاه‌ترین یال در دور ناکامل کم‌تر از ۲ باشد [۱۹]. بهبودهایی از این روش با استفاده از صف اولویت پیشنهاد شده است [۲۰]. روش برووکای سریع (QB) توسعه‌ای از روش GH و مبتنی بر درخت پوشاست که به جای استفاده از صف‌های اولویت، از مرتب‌سازی داده‌ها استفاده می‌کند [۲۰]. روش MTS [۱۲] مبتنی بر تطابق عمل می‌کند. در این روش ابتدا دورهای کوچک‌تر ساخته می‌شوند و با استفاده از نسخه مناسب‌سازی شده از مسئله تطابق، دور نهایی به‌دست می‌آید.

```

1. function tour = savings(V) {
2.   hub ← index of the most central city in V
3.   Vh ← {1, ..., n} - hub
4.   foreach i, j ∈ Vh, i < j do
5.     S[i, j] ← D(hub, i) + D(hub, j) - D(i, j)
6.     while |Vh| > 2 do
7.       add (vi, vj) with the best S[i, j] to partial tour provided that no
       subtour is created
8.       if degree of vi is 2, remove i from Vh
9.       if degree of vj is 2, remove j from Vh
10.      stitch remaining two cities in Vh through the hub node
11.      return tour
12. }
```

شکل ۲: روش ابتکاری پس‌انداز برای حل TSP - نسخه اولیه

روش‌های بهبود تدریجی شامل روش ۲-opt [۲۱] و روش‌های تکاملی مانند شبیه‌سازی ذوب فلزات [۲۲] و الگوریتم ژنتیک [۲۳] می‌شود. در روش ۲-opt به‌عنوان یک روش جستجوی محلی، جای گره‌های انتهایی دو یال از دور داده شده به صورت ضربدری تغییر می‌کند تا طول کلی دور کاهش یابد. توسعه‌هایی نیز از این روش برای در نظر گرفتن تعداد بیش‌تری یال ارائه شده است [۲۴]. در روش تکاملی ذوب فلزات تغییرات محدودی در دور داده شده پیشنهاد داده می‌شود. در صورتی که این تغییر منجر به بهبود شود، تغییر پذیرفته می‌شود و در صورتی که منجر به بهبود نشود، با احتمالی این تغییر پذیرفته می‌شود. این احتمال در طول اجرای الگوریتم به‌مرور کاهش می‌یابد. در روش تکاملی مبتنی بر الگوریتم ژنتیک

این مقاله از نسخه بهبودیافته آن برای حل TSP در فضای اقلیدسی استفاده کرده‌ایم. همان‌طور که در بخش مقدمه بیان شد، در این مقاله برای اینکه بتوان در حل GEOTSP از ابتکارات مختص فضای اقلیدسی استفاده کرد، از تصویرسازی نقشه استفاده می‌شود. در پاسخ به این سوال که کدام روش‌های تصویرسازی نقشه برای حل GEOTSP مناسب هستند، در این بخش توضیح داده می‌شود.

## ۲-۱- معرفی مسئله فروشنده دوره‌گرد با معیار فاصله جغرافیایی

فرض کنید شبکه‌ای از مکان‌های جغرافیایی (شهرها) به صورت گراف کامل  $G = (V, E)$  در اختیار است که  $V = \{v_1, v_2, \dots, v_n\}$  نشان‌دهنده شهرها و  $E = V \times V$  یال‌های بین آن‌هاست. وزن هر یال بین دو گره  $v_i$  و  $v_j$  به صورت  $c(v_i, v_j)$  نشان داده می‌شود. هدف مسئله فروشنده دوره‌گرد کمینه کردن دوری روی این گراف است که از همه گره‌ها دقیقاً یک‌بار می‌گذرد (و به مکان شروع برمی‌گردد). به صورت رسمی‌تر، هدف یافتن ترتیبی از شهرهاست که تابع هدف رابطه (۱) کمینه شود.

$$\min \sum_{i=2}^n c(v_{(i-1)}, v_i) + c(v_n, v_1) \quad (1)$$

نسخه‌ای از مسئله که در این مقاله به آن پرداخته شده است فواصل بین شهرها را به صورت متقارن تعریف می‌کند، یعنی  $c(v_i, v_j) = c(v_j, v_i)$ . برای نمونه‌هایی از TSP با معیار فاصله جغرافیایی و در گستره‌ای به وسعت کره زمین در مجموعه داده استاندارد TSPLib معیار فاصله جغرافیایی بین دو شهر  $v_i$  و  $v_j$  یعنی  $c(v_i, v_j)$  برحسب متر بر روی کره‌ای به شعاع  $R = 6378388$  متر به کمک تابع  $GEOM(i, j)$  محاسبه می‌شود. فرض کنید طول و عرض جغرافیایی شهر  $v_i$  در ارائه‌های متناظر به ترتیب به صورت  $long[i]$  و  $lat[i]$  و برحسب رادبان ذخیره شده‌اند. محاسبه فاصله در شکل ۱ نشان داده شده است.

```

1. GEOM(i, j) {
2.   R = 6378388
3.   q1 = cos(lat[j]) * sin(long[i] - long[j])
4.   q3 = sin(long[i] - long[j]) / 2.0
5.   q4 = cos(long[i] - long[j]) / 2.0
6.   q2 = sin(lat[i] + lat[j]) * q3 * q3 - sin(lat[i] - lat[j]) * q4 * q4
7.   q5 = cos(lat[i] - lat[j]) * q4 * q4 - cos(lat[i] + lat[j]) * q3 * q3
8.   return (int)(R * atan(sqrt(q1 * q1 + q2 * q2), q5) + 1.0)
9. }
```

شکل ۱: محاسبه فاصله جغرافیایی بین دو شهر  $v_i$  و  $v_j$  برحسب متر

در شکل ۱، توابع  $atan$ ،  $sqrt$  و  $int$  به ترتیب نشان‌دهنده معکوس تانژانت، جذر و جزء صحیح هستند.

## ۲-۲- روش‌های ابتکاری حل TSP

روش‌های ابتکاری برای حل TSP شامل روش‌های ساختنی و روش‌های بهبود تدریجی می‌شوند. در روش‌های ساختنی، بر اساس قاعده‌ای تعریف شده دور ساخته می‌شود و بعد از ساختن (بخشی از) آن، تلاشی

کمتر از ۲ برابر حداقل فاصله‌های آن گره‌ها تا گره قطب است. به بیان دیگر برای دو جفت گره با اندیس  $i$  و  $j$  همواره نامساوی  $S[i, j] \leq \min\{D(i, hub), D(j, hub)\}$  برقرار است [۱۰]. بنابراین نیازی نیست که همه جفت گره‌ها را در ضمن ساخت دور در نظر گرفت. به علاوه مرتضوی و جلیلی روشی برای بهبود این ابتکار ارائه کرده‌اند که از ساختمان داده kd-tree به همراه مفهومی بنام گره فعال استفاده می‌کند [۱۷]. در این روش برای محاسبه پس‌انداز به صورت تدریجی تنها نزدیک‌ترین همسایه فعال یک گره (یعنی همسایه‌ای با درجه کم‌تر از ۲ در دور ناکامل)، برای جفت شدن مورد بررسی قرار می‌گیرد، زیرا اصولاً احتمال جفت شدن دو گره نزدیک به هم در دور با طول کمینه بیش‌تر است. تمامی این بهبودها با فرض این قابل استفاده هستند که فاصله بین  $hub$  گره‌ها در فضای اقلیدسی محاسبه می‌شود.

#### ۲-۴- روش‌های تبدیل فضا در مختصات جغرافیایی

به روش‌های تبدیل فضا در مختصات جغرافیایی اصطلاحاً تصویر سازی نقشه می‌گویند. به بیان دقیق‌تر به روش قاعده‌مندی که برای تبدیل نقاطی بر روی سطح کره یا بیضی بر حسب طول و عرض جغرافیایی به نقاطی بر روی کاغذ استفاده می‌شود، تصویر سازی نقشه می‌گویند [۲۶]. همواره چنین تبدیلی همراه با نوعی اعوجاج در مساحت، شکل، جهت، فاصله یا مقیاس خواهد بود. همچنین نحوه مدل کردن زمین (به صورت کره، بیضی یا ...) نیز در ویژگی‌های نقشه نهایی تأثیر خواهد داشت.

به صورت طبیعی شباهت فواصل بین شهرها در تصویر ساخته شده در نقشه با اطلاعات اصلی قبل از تصویر سازی منجر به نتایج دقیق‌تری در حل GEOTSP می‌شود. به روش‌های تصویر سازی نقشه که (تا حد امکان) فواصل بین گره‌ها را پس از تصویر سازی حفظ می‌کنند، به اصطلاح هم‌فاصله<sup>۸</sup> می‌گویند. در ادامه این مقاله و با توجه به نحوه محاسبه فاصله در مجموعه داده‌های مورد نظر در مسئله TSP (بخش ۲-۱)، به دنبال روش‌های تصویر سازی نقشه هم‌فاصله و با فرض کروی بودن زمین هستیم. در این مقاله از روش‌های تصویر سازی نقشه هم‌فاصله Azimuthal equidistant، Equidistant Conic، و Cassini استفاده شده است. در روش Azimuthal equidistant که توسط ابوریحان بیرونی در سال ۱۰۰۰ میلادی ارائه شده است، از نقطه‌ای مرکزی استفاده می‌شود که فواصل تا این نقطه حفظ می‌شوند. روش Equidistant Conic در سال ۱۰۰۰ میلادی پیشنهاد شده است و فواصل را بر روی نصف‌النهارها (همان فاصله مدارهای استاندارد عرض‌های جغرافیایی)، حفظ می‌کند. روش Cassini که در سال ۱۷۴۵ توسط César-François Cassini de Thury ارائه شده است از نوع تصویر سازی نقشه استوانه‌ای است و مقیاس‌ها را در طول نصف‌النهار و خطوط موازی با آن حفظ می‌کند.

برای حل TSP چندین جواب کاندید در قالب یک نسل در نظر گرفته می‌شود و علاوه بر تغییرات مختصر بر روی اعضای یک نسل برای بهبود آن‌ها، از ترکیب جواب‌های خوب نیز برای تولید جواب جدید استفاده می‌شود. به طور کلی استفاده از روش‌های تکاملی بر روی نمونه‌های بزرگ TSP به علت سرعت پایین آن‌ها، به عنوان اولویت اول مطرح نمی‌شود.

#### ۲-۳- روش ابتکاری پس‌انداز و بهبودهای آن

یکی از روش‌های ابتکاری موفق برای حل TSP ابتکار پس‌انداز است [۸]. این روش در اصل برای حل مسائلی در مسیریابی پیشنهاد شده و نسخه‌های متنوعی دارد. شکل ۲ نسخه اولیه این روش را نشان می‌دهد. در شکل ۲، در خط ۲ مرکزی‌ترین شهر از مجموعه  $V$  با عنوان گره قطب انتخاب می‌شود و اندیس آن در متغیر ذخیره می‌شود. اندیس همه شهرها به جز  $hub$  در خط ۳ در  $V_h$  ذخیره می‌شود. در خطوط ۴ و ۵، برای همه جفت شهرها در  $V_h$  مقداری به عنوان پس‌انداز محاسبه و در متغیر متناظر یعنی  $S[i, j]$  ذخیره می‌شود. در خط ۵، تابع  $D(i, j)$  نشان‌دهنده فاصله بین  $v_i$  و  $v_j$  است. در حلقه اصلی این روش، یعنی خطوط ۶ تا ۹ تا زمانی که تعداد عناصر  $V_h$  بیش‌تر از دو است، یال متصل‌کننده شهرهای  $i$ -ام و  $j$ -ام یعنی  $(v_i, v_j)$  که مقدار پس‌انداز مربوط به آن از همه بیش‌تر است، در صورتی که به ایجاد دوری منجر نشود، به مجموعه یال‌های دور ناکامل اضافه می‌شود (خط ۷). در خط ۸ و ۹، اندیس مربوط به شهرهای  $v_i$  و  $v_j$  در صورتی که به طور کامل از دو طرف در tour متصل شده باشند، از  $V_h$  حذف می‌شود. در پایان و در خط ۱۰، گره قطب با اندیس  $hub$  به دو گره باقی‌مانده که اندیس‌هایشان در  $V_h$  قرار دارد، متصل شده و دور را تکمیل می‌کند. دور تکمیل شده در خط ۱۱ برگشت داده می‌شود.

پیاده سازی این الگوریتم اولیه آسان است. ولی مرتبه زمان اجرای آن  $O(n^2 \log n)$  است و همچنین فضای مورد نیاز برای ذخیره کردن مقادیر پس‌انداز از مرتبه  $O(n^2)$  است. این بدان معناست که اجرای این الگوریتم برای مقادیر بزرگ  $n$  میسر نیست. بهبودهایی برای رفع این مشکلات پیشنهاد شده است. Golden و همکاران نشان داده‌اند که کافی است فقط برخی جفت گره‌های خاص را در ضمن ساخت دور در نظر بگیریم و بر این اساس روالی برای مرتب سازی پس‌اندازها را با استفاده از ساختمان داده کپه<sup>۹</sup> پیشنهاد کرده‌اند [۲۵].

به علاوه، Paessens روشی با عنوان «یافتن تدریجی بیش‌ترین مقدار پس‌انداز» ارائه کرده است [۱۰]. در این روش ابتدا همه رکوردها بر اساس فاصله تا گره قطب مرتب می‌شوند و تنها آن دسته از جفت گره‌ها در محاسبه پس‌انداز وارد می‌شوند که فاصله هر دو گره آن‌ها تا گره قطب حداقل به اندازه نصف بیش‌ترین پس‌انداز موجود باشد، زیرا در فضای اقلیدسی حداکثر مقدار پس‌انداز برای یک جفت گره همواره مساوی یا

## ۳- روش پیشنهادی

پس از انجام این مقدمات، فاز پیش پردازش شروع می شود که تفاوت اصلی روش پیشنهادی با روش ارائه شده در FDM [۱۷] محسوب می شود. این فاز پیش پردازش بر روی گره‌ها به ترتیب عکس فاصله آن‌ها تا گره قطب و در خطوط ۷ و ۸ انجام می شود. در این فاز، سعی می شود حتی الامکان بهترین انتخاب برای گره‌ها به صورت یک انتخاب محلی صورت بگیرد: اگر  $knn(i,1) = j$  و  $knn(j,1) = i$ ، در این صورت یال  $(v_i, v_j)$  قبل از شروع حلقه اصلی به یکدیگر متصل می شوند. همچنین در مورد گره‌های  $v_i$  و  $v_j$  که  $j > i$ ، چنانچه  $knn(i,1) = j$  و  $knn(j,2) = i$  باشد و در مرحله قبلی گره  $j$  به گره دیگری متصل نشده باشد، در این صورت یال  $(v_i, v_j)$  به عنوان جزئی از دور نهایی افزوده می شود. در واقع این مراحل به نوعی تضمین می کنند که بهترین انتخاب برای گره  $v_i$  انجام شده است، زیرا به طور طبیعی اتصال به نزدیکترین همسایه  $v_i$  به عنوان تصمیمی محلی بهترین انتخاب حریصانه برای  $v_i$  محسوب می شود. استفاده از این پیش پردازش باعث کاهش بار محاسباتی در مراحل بعدی الگوریتم می شود، زیرا زودتر همسایه های نهایی گره  $v_i$  در دور نهایی تعیین می شوند. در خط ۹ شماره گر گره جاری برای محاسبه پس انداز به آخرین گره در با اندیس  $n$  تنظیم می شود. حلقه اصلی برنامه در خطوط ۱۰ تا ۲۴ به اضافه کردن یال‌ها تا زمانی که اندازه  $V_h$  بیش تر از ۲ هست، می پردازد. ابتدا در حلقه ای داخلی تا زمانی که امکان بهبود بهترین پس انداز باشد (خط ۱۱)، اطلاعات پس اندازها در کپه به روز می شود. دیس نزدیکترین همسایه قابل اتصال به گره جاری با اندیس  $idx$  در متغیر NN ذخیره می شود (خط ۱۲) و مقدار پس انداز مربوط به یال  $(v_{idx}, v_{NN})$  در  $S$  ذخیره می شود (خط ۱۳). این اطلاعات در کپه ذخیره می شوند (خط ۱۴) و مقدار  $idx$  برای اشاره به گره بعدی به روز می شود (خط ۱۵). در خط ۱۶ بهترین یال کاندید برای اضافه شدن به دور ناکامل در  $(i, j)$  ذخیره می شود و اطلاعات آن از بالای کپه حذف می شود. اگر  $v_i$  کاملاً متصل شده باشد و اندیس آن در  $V_h$  نباشد، ادامه حلقه نادیده گرفته می شود و مورد بعدی بررسی می شود (خط ۱۷). در صورتی که  $v_j$  به طور کامل متصل نشده باشد و افزودن یال  $(i, j)$  باعث ایجاد دور نشود، این یال به دور ناکامل اضافه می شود (خط ۱۹). در خط ۲۰ و ۲۱ اطلاعات  $V_h$  در صورت لزوم به روز می شود. در خطوط ۲۲ تا ۲۴ نیز اطلاعات مربوط به همسایگی گره  $v_i$  محاسبه و کپه به روز می شود. پس از پایان حلقه اصلی، خط ۲۵ وظیفه کامل کردن دور ناکامل را با افزودن یال‌هایی بین گره قطب و گره‌های با اندیس ذکر شده در  $V_h$  به عهده دارد. نهایتاً خط ۲۶ دور کامل شده را برمی گرداند.

## ۴- نتایج تجربی

در این بخش نتایج اجرای نسخه بهبود یافته ابتکار پس انداز بر روی نمونه‌های TSP با معیار فاصله جغرافیایی ارائه می شود. این ۱۰ نمونه از

ایده اصلی این مقاله برای حل TSP با معیار فاصله جغرافیایی بر پایه تبدیل فضا هست. بدین منظور از روش‌های ترسیم نقشه که به نوعی فاصله‌ها را حفظ می کنند، برای تبدیل مختصات جغرافیایی شهرها به مختصات کارتریز در دو بعد استفاده می شود. برای حل نسخه دوبعدی، نسخه بهبود یافته‌ای از روش ابتکاری پس انداز استفاده می شود که می تواند با سرعت و کیفیت بالا نسخه‌هایی تا چند میلیون شهر از مسئله TSP را حل کند. در پایان می توان جواب نمونه‌های دوبعدی را به نسخه اولیه مسئله در فضای اصلی اعمال کرد تا طول مسیر جواب پیشنهادی به دست آید.

همان طور که در بخش ۲-۳ بیان شد، نسخه اولیه روش ابتکاری پس انداز برای حل TSP دارای مشکلاتی هست. به همین دلیل در این مقاله برای بهبود زمان و حافظه مصرفی این الگوریتم، بهبودهایی در آن داده شده است. فرض کنید  $knn(i, k) \in \{1, \dots, n\}$  نشان دهنده اندیس  $k$ -امین نزدیکترین همسایه قابل اتصال گره  $v_i$  در  $V$  (البته به جز خود  $v_i$ ) باشد. شکل ۳ در قالب تابع savings2 روش پیشنهادی را با عنوان savings2 ارائه می دهد.

```

1. function tour=savings2(V) {
2.   hub ← index of the most central city in V
3.   V ← sort(V, hub)
4.   KD ← kd-tree on V
5.   H ← empty Max Heap
6.   Vh ← {2, ..., n}
7.   if knn(i, 1) = j  ∀ i ∈ Vh // decreasing loop on i
8.     if knn(j, 1) = i or (i < j and degree of vj is 0 and
       knn(j, 2) = i), add (i, j) to partial tour
9.   idx ← n
10.  while |Vh| > 2 do
11.    while H is empty or D(idx, hub) ≥ H.top/2 and idx > 1 do
12.      NN ← knn(idx, 1)
13.      S ← D(hub, idx) + D(hub, NN) - D(idx, NN)
14.      H.push(S)
15.      idx ← idx - 1
16.      (i, j) ← H.pop
17.      if i ∉ Vh continue
18.      if j ∈ Vh and adding(i, j) to partial tour does not create a
        subtour
19.        add(i, j) to partial tour
20.        if degree of vj is 2, remove j from Vh
21.        if degree of vi is 2, remove i from Vh, and continue
22.      NN ← knn(i, 1)
23.      S ← D(hub, i) + D(hub, NN) - D(i, NN)
24.      H.push(S)
25.      stitch remaining two cities in Vh through the hub node
26.    return tour
27. }
```

شکل ۳: نسخه بهبود یافته و کامل ابتکار پس انداز برای حل TSP

در خط ۱ مختصات گره‌ها به تابع تحویل می شود. در خط ۲ مرکزی ترین گره بنام قطب انتخاب می شود. در خط ۳، همه گره‌ها در مجموعه  $V$  بر حسب فاصله از گره قطب به صورت صعودی مرتب می شوند به نحوی که گره قطب با اندیس ۱ در  $V$  شناخته می شود. در خطوط ۴ و ۵ به ترتیب ساختمان داده‌های kd-tree و کپه ساخته می شوند. در خط ۶ اندیس همه گره‌ها به جز گره قطب (با اندیس ۱) در  $V_h$  ذخیره می شود.

شده در مرجع [۱۷]، نتایج طول دور بدون در نظر گرفتن پیش‌پردازش در الگوریتم پیشنهادی (خطوط ۷ و ۸ از شکل ۳) با عنوان نسخه ساده و با در نظر گرفتن پیش‌پردازش با عنوان نسخه کامل نشان داده شده است. این طول دورها با نتایج طول دور مربوط به روش‌های GH، NN و QB بر روی نمونه اصلی (بدون تصویرسازی نقشه)، مقایسه شده‌اند. پیاده‌سازی این روش‌ها در برنامه‌ای با عنوان linkern<sup>۱۱</sup> ارائه شده است. زمان اجرای روش‌های فوق (بدون در نظر گرفتن زمان کار با دیسک) نیز در جدول ۲ ارائه شده است. زمان اجرای savings2 به‌ازای هر نمونه با اندازه ثابت، برای نسخه‌های ساده و کامل بسیار کم و تقریباً یکسان است. بنابراین تنها یک زمان اعمال تصویرسازی نقشه ( $t_{projection}$ ) همراه با زمان اجرای ابتکار پس‌انداز بهبودیافته ( $t_{savings}$ ) در جدول ۲ ارائه شده است.

۱۰۰۰۰ تا ۱۰۰۰۰۰ موقعیت جغرافیایی شهرهای کره زمین را دربر دارند و از ابتدای نمونه TSP جغرافیایی با عنوان World TSP با ۱۹۰۴۷۱۱ شهر انتخاب شده‌اند. اطلاعات این شهرها در قالب طول و عرض جغرافیایی ارائه شده است. کلیه محاسبات بر روی کامپیوتری شخصی با پردازنده Core i7 - 1.60 GHz، سیستم‌عامل ویندوز ۱۰ و ۸ گیگابایت حافظه اصلی انجام شده است. برای اعمال تصویرسازی نقشه از نرم‌افزار ArcMap 10.3.1 و جعبه‌ابزار تصویرسازی نقشه نرم‌افزار MATLAB استفاده شده است. در این مورد از مقادیر پیش‌فرض پارامترهای این نرم‌افزارها برای تصویرسازی نقشه استفاده شده است.

نتایج ارائه شده در این بخش مربوط به سه نوع تصویرسازی نقشه معرفی شده در بخش ۲-۴ یعنی Conic و Cassini، Azimuthal است. نتایج این روش‌ها به ترتیب با AZ، CA و CO در جدول ۱ ارائه شده است. برای مشاهده اثر پیش‌پردازش پیشنهادی و مقایسه دقیق‌تر با روش ارائه

جدول ۱: طول دور روش savings2 مبتنی بر تصویرسازی نقشه (نسخه ساده بدون پیش‌پردازش و نسخه کامل با پیش‌پردازش) در مقایسه با سایر روش‌های ابتکاری بدون استفاده از تصویرسازی نقشه (همه فاصله‌ها بر حسب متر هستند.)

اندازه (۱۰۰۰)	نسخه savings2	مبتنی بر تصویرسازی نقشه			بدون تصویرسازی نقشه		
		AZ	CA	CO	GH	NN	QB
۱۰	ساده	۱۵۸۴۰۱۵۲۴	۱۵۴۶۰۳۶۳۵	۱۵۳۰۵۳۵۳۲	۱۶۷۷۹۸۹۳۹	۱۶۰۶۰۲۰۵۰	۱۷۰۸۰۷۷۱۷
	کامل	۱۵۸۲۸۱۴۹۱	۱۵۴۲۱۵۵۴۰	۱۵۴۳۲۲۵۴۹			
۲۰	ساده	۲۴۷۵۹۴۶۲۱	۲۵۱۵۶۴۴۴۵	۲۴۷۵۱۱۴۶۶	۲۵۸۰۸۸۵۹۲	۲۴۹۷۷۹۸۸۵	۲۷۵۲۸۶۲۳۸
	کامل	۲۴۳۴۲۴۹۸۹	۲۴۸۳۳۲۶۲۱	۲۵۱۸۰۷۷۶۲			
۳۰	ساده	۳۴۹۱۷۱۴۲۸	۳۵۲۴۳۶۲۰۳	۳۴۶۴۹۲۰۲۳	۳۶۶۵۱۵۹۵۲	۳۷۰۲۲۸۴۱۶	۳۷۲۱۹۵۷۷۲
	کامل	۳۴۵۹۰۶۸۳۶	۳۴۶۶۹۱۳۷۹	۳۴۳۴۰۸۱۵۳			
۴۰	ساده	۴۱۶۸۱۵۵۰۴	۴۱۹۳۵۴۰۸۹	۴۱۴۸۷۶۷۷۲	۴۶۰۳۶۶۲۹۳	۴۴۷۷۶۴۵۶۳	۴۵۶۴۹۶۹۲۱
	کامل	۴۱۲۱۹۲۲۲۹	۴۱۵۸۸۶۱۸۷	۴۱۴۸۷۳۳۳۹			
۵۰	ساده	۴۸۰۹۲۷۴۰۷	۴۸۱۵۰۵۶۲۱	۴۷۶۳۲۹۵۹۶	۵۴۷۰۰۹۸۰۳	۵۱۷۱۶۴۹۷۵	۵۱۲۹۶۱۹۴۱
	کامل	۴۷۵۷۶۱۲۵۰	۴۷۶۳۳۹۸۱۲	۴۷۹۰۵۱۷۴۳			
۶۰	ساده	۵۴۳۸۱۵۹۹۷	۵۴۵۴۳۹۸۰۸	۵۴۰۹۰۴۶۵۲	۶۱۶۴۲۰۸۲۰	۵۹۳۷۴۵۹۰۵	۶۰۸۵۴۹۶۳۲
	کامل	۵۳۸۴۸۳۰۴	۵۴۰۱۷۱۳۷۰	۵۳۷۸۰۴۲۸۸			
۷۰	ساده	۶۰۵۰۴۹۰۸۴	۶۰۶۲۹۷۵۶۳	۶۰۳۶۷۹۳۵	۶۷۵۰۲۱۸۸۰	۶۷۵۲۹۵۷۲۴	۶۵۸۹۴۱۸۴۳
	کامل	۵۹۹۲۶۰۹۶۴	۶۰۰۳۳۷۵۷۹	۵۹۷۶۴۷۸۸۰			
۸۰	ساده	۶۷۳۷۷۶۷۸۸	۶۷۴۲۷۹۹۳۸	۶۷۴۹۰۱۱۰۸	۷۳۰۴۰۶۸۸۷	۷۴۵۳۱۵۱۹۰	۷۳۹۵۶۸۹۷۰
	کامل	۶۶۵۴۴۵۶۵۸	۶۶۷۹۸۷۱۱۶	۶۶۸۶۱۳۳۲۸			
۹۰	ساده	۷۳۴۷۶۰۳۵۳	۷۳۴۷۲۲۸۳۳	۷۳۷۶۴۹۳۴۳	۸۱۴۴۱۵۳۲۱	۸۰۶۹۰۱۲۱۷	۸۰۷۲۵۴۶۰۴
	کامل	۷۳۰۱۰۶۶۴۰	۷۲۸۳۶۷۲۲۵	۷۳۱۴۵۴۹۳۶			
۱۰۰	ساده	۷۹۳۳۲۲۸۲۰	۷۸۹۳۴۷۲۹۸	۷۹۱۷۸۳۹۳۷	۸۸۶۳۷۲۴۱۸	۸۷۲۹۹۳۱۴۰	۸۸۵۴۲۶۵۴۰
	کامل	۷۸۶۲۲۵۴۷۹	۷۸۹۷۷۲۲۵۲	۷۸۷۲۸۱۳۵۲			

**جدول ۲: زمان اجرای روش مبتنی بر تصویرسازی نقشه در مقایسه با سایر روش‌های ابتکاری بدون استفاده از تصویرسازی نقشه (همه زمان‌ها بر حسب ثانیه هستند).**

اندازه	مبتنی بر سیستم تصویر		بدون سیستم تصویر	
(۱۰۰۰)	$t_{projection}$	$t_{savings2}$	$t_{GH}$	$t_{NN}$
۱۰	۰/۰۱	۰/۱۴	۰/۲۷	۳۳/۸۹
۲۰	۰/۰۱	۰/۱۸	۰/۸۷	۱۲۸/۷۶
۳۰	۰/۰۱	۰/۲۳	۱/۸۵	۱۸۴/۲۴
۴۰	۰/۰۲	۰/۲۵	۳/۲۱	۳۹۷/۷۵
۵۰	۰/۰۲	۰/۳۳	۴/۹۱	۵۹۲/۲۵
۶۰	۰/۰۲	۰/۳۷	۶/۷۶	۹۲۸/۷۹
۷۰	۰/۰۲	۰/۴۱	۹/۰۴	۱۲۵۷/۴۵
۸۰	۰/۰۲	۰/۴۳	۱۱/۶۱	۱۶۴۷/۵۶
۹۰	۰/۰۳	۰/۵۲	۱۴/۹۵	۱۶۴۹/۰۶
۱۰۰	۰/۰۳	۰/۵۸	۱۸/۲۷	۲۳۶۱/۲۱

**جدول ۳: بهترین نتایج طول دور مربوط به QB، MTS و savings2 در مورد**

مجموعه داده‌های جغرافیایی مبتنی بر تصویرسازی نقشه			
اندازه (۱۰۰۰)	QB	MTS	savings2
۱۰	۲۰۰۵۹۷۳۲۲	۲۹۹۴۶۳۳۹۲	۱۵۳۰۵۳۵۳۲
۲۰	۳۳۴۸۳۳۸۶۵	۵۲۹۵۲۲۲۰۸	۲۴۴۴۲۴۹۸۹
۳۰	۵۱۵۱۹۸۹۸۶	۷۸۵۰۹۴۹۷۷	۳۴۳۴۰۸۱۵۳
۴۰	۵۸۳۲۲۳۷۳۰	Error	۴۱۲۱۹۲۲۲۹
۵۰	۶۹۵۵۶۸۱۰۴	۱۲۳۱۹۹۹۹۹۹	۴۷۵۷۶۱۲۵۰
۶۰	۸۴۱۴۴۷۸۶۸	۱۴۶۵۲۹۵۴۳۱	۵۳۷۸۰۴۲۸۸
۷۰	۹۳۴۷۶۱۴۷۸	-	۵۹۷۶۴۷۸۸۰
۸۰	۹۷۸۵۱۸۲۲۲	-	۶۶۵۴۴۵۶۵۸
۹۰	۱۰۷۰۸۳۸۳۳۰	-	۷۲۸۳۶۷۲۲۵
۱۰۰	۱۱۸۸۸۳۹۹۲۲	-	۷۸۶۲۲۵۴۷۹

برای مثال در آزمایش‌های انجام‌شده در این جدول، نتایج savings2 همواره کم‌تر از نصف طول دور به‌دست آمده از MTS است. همچنین طول دور روش QB همواره حداقل ۳۰٪ بیش‌تر از طول دور روش savings2 است. نکته قابل توجه دیگر روند تقریبی بهبود نسبی طول دور روش savings2 به طول دور روش QB است که به‌صورت صعودی می‌باشد و روش پیشنهادی را به‌عنوان گزینه مناسبی برای نمونه‌های بزرگ TSP معرفی می‌کند.

به‌منظور نشان دادن توانایی روش پیشنهادی در مقایسه با سایر ابتکارهای ساختنی حل TSP، الگوریتم پیشنهادی روی ۵ نمونه از TSP مربوط به نقاط جغرافیایی کشورها در فضای اقلیدسی دوبعدی اعمال شد. این نمونه‌ها از مجموعه داده TSP LIB<sup>۶</sup> به‌عنوان مجموعه استاندارد گرفته شده است و در مراجع جدید مانند [۱۲] به آن‌ها اشاره شده است. نتایج طول دور savings2 در مقایسه با QB در جدول ۴ نشان داده شده است. زمان اجرا برای savings2 همواره کم‌تر از QB بود. چون این زمان‌ها

نتایج جدول ۱ نشان می‌دهد که همواره بهترین جواب روش‌های مبتنی بر تصویرسازی نقشه از بهترین جواب سایر روش‌ها بهتر بوده است. این بهبود به‌طور میانگین بیش از ۹٪ بوده است. برای مثال طول دور نهایی روش مبتنی بر تصویرسازی نقشه Azimuthal به‌ازای  $n=10000$  نسبت به ابتکار QB بدون تصویرسازی نقشه بهبود ۱۳٪ را نشان می‌دهد. در بین ابتکارهای QB، NN و GH اکثراً کوتاه‌ترین دورها توسط روش NN به دست آمده‌اند. البته با توجه به زمان اجرای ثبت‌شده برای این روش در جدول ۲ در مقایسه با زمان روش‌های QB و GH، این روش ابتکاری کندترین روش است. در هر صورت برتری مطلق به‌لحاظ زمان اجرا نیز با روش‌های مبتنی بر تصویرسازی نقشه است. درواقع این زمان اجرا (شامل تصویرسازی نقشه و اعمال روش ابتکاری پس‌انداز بهبودیافته) همواره کم‌تر از ۱ ثانیه بوده که برتری مطلق روش پیشنهادی را نشان می‌دهد.

برای انجام مقایسه‌ای عادلانه و کامل، نتایج روش ابتکاری ساختنی QB<sup>۱۱</sup> و MTS [۱۲] بر روی نسخه تصویرسازی‌شده از مجموعه داده پیشنهادی اعمال شده است. برای این منظور مانند آزمایش قبلی ابتدا انواع تصویرسازی نقشه انجام شده و روش ابتکاری بر روی آن اعمال شده است. سپس نتایج مربوط به دور نهایی بر روی مجموعه داده اصلی (قبل از تبدیل فضا) اعمال شده است. بهترین نتایج مربوط به انواع روش‌های تصویرسازی نقشه در این آزمایش در جدول ۳ آمده است. در این آزمون از تنظیمات پیش‌فرض MTS که مبتنی بر ساخت درخت پوشای کمینه<sup>۱۲</sup> هست، استفاده شده است. از آنجاکه روش MTS برای نمونه‌های با تعداد شهر بزرگ‌تر از ۶۰۰۰۰ بسیار کند (بیش از ۱۰ دقیقه) عمل می‌کند، در جدول ۲ فقط بهترین نتایج طول دور برای نمونه‌های کوچک برای MTS گزارش شده است<sup>۱۳</sup>. نتایج جدول ۳ مؤید برتری مطلق روش پیشنهادی به‌لحاظ طول دور نهایی است.

[3] Z. Lin *et al.*, "Tool path generation for multi-axis freeform surface finishing with the LKH TSP solver," *Computer-Aided Design*, vol. 69, pp. 51-61, 2015.

[4] G. Laporte, "A concise guide to the traveling salesman problem," *Journal of the Operational Research Society*, vol. 61, no. 1, pp. 35-40, 2010.

[5] R. M. Karp, "Reducibility among combinatorial problems," *Complexity of Computer Computations: Proceedings of a symposium on the Complexity of Computer Computations*, pp. 85-103: Springer, 1972.

[6] C. Rego *et al.*, "Traveling salesman problem heuristics: leading methods, implementations and latest advances," *European Journal of Operational Research*, vol. 211, no. 3, pp. 427-441, 2011.

[7] D. L. Applegate *et al.*, *The Traveling Salesman Problem: A Computational Study: A Computational Study*: Princeton university press, 2011.

[8] G. Clarke, and J. W. Wright, "Scheduling of vehicles from a central depot to a number of delivery points," *Operations research*, vol. 12, no. 4, pp. 568-581, 1964.

[9] J. L. Bentley, "Multidimensional binary search trees used for associative searching," *Communications of the ACM*, vol. 18, no. 9, pp. 509-517, 1975.

[10] H. Paessens, "The savings algorithm for the vehicle routing problem," *European Journal of Operational Research*, vol. 34, no. 3, pp. 336-344, 1988.

[11] A. Czumaj, "Euclidean Traveling Salesman Problem," *Encyclopedia of Algorithms*, M.-Y. Kao, ed., pp. 1-6, Berlin, Heidelberg: Springer Berlin Heidelberg, 2014.

[12] A. B. Kahng, and S. Reda, "Match twice and stitch: a new TSP tour construction heuristic," *Operations Research Letters*, vol. 32, no. 6, pp. 499-509, 2004.

[13] J. Faigl *et al.*, "An application of the self-organizing map in the non-Euclidean Traveling Salesman Problem," *Neurocomputing*, vol. 74, no. 5, pp. 671-679, 2011.

[14] K. M. Curtin, "Network analysis in geographic information science: Review, assessment, and projections," *Cartography and Geographic Information Science*, vol. 34, no. 2, pp. 103-111, 2007.

[15] A. L. Walwyn, and D. J. Navarro, "Minimal paths in the city block: Human performance on Euclidean and non-Euclidean traveling salesperson problems," *The Journal of Problem Solving*, vol. 3, no. 1, pp. 5, 2010.

[16] T. Moore, *Implementing the Held-Karp Lower Bound Algorithm in Python*, 2015.

[17] R. Mortazavi, and S. Jalili, "Fast data-oriented microaggregation algorithm for large numerical datasets," *Knowledge-Based Systems*, vol. 67, pp. 195-205, 2014.

[18] H. Okano, S. Misono, and K. Iwano, "New TSP construction heuristics and their relationships to the 2-Opt," *Journal of Heuristics*, vol. 5, no. 1, pp. 71-88, 1999.

[19] J. L. Bentley, "Experiments on traveling salesman heuristics," *Proceedings of the first annual ACM-SIAM symposium on Discrete algorithms*, pp. 91-99, San Francisco, California, USA: Society for Industrial and Applied Mathematics, 1990.

[20] D. S. Johnson, and L. A. McGeoch, "Experimental analysis of heuristics for the STSP," *The traveling salesman problem and its variations*, pp. 369-443: Springer, 2007.

[21] G. A. Croes, "A method for solving traveling-salesman problems," *Operations research*, vol. 6, no. 6, pp. 791-812, 1958.

[22] S. Rees, and R. C. Ball, "Criteria for an optimum simulated annealing schedule for problems of the travelling salesman type," *Journal of Physics A: Mathematical and General*, vol. 20, no. 5, pp. 1239-1239, 1987.

[23] J. Knox, and F. Glover, *Comparative Testing of Traveling Salesman Heuristics Derived from Tabu Search, Genetic Algorithms, and Simulated Annealing*, 1989.

[24] K. Helsgaun, "General k-opt submoves for the Lin-Kernighan TSP heuristic," *Mathematical Programming Computation*, vol. 1, no. 2-3, pp. 119-163, 2009.

همواره حدود ۱ ثانیه بوده است، مقدار آن‌ها ذکر نشده است. مجدداً مشاهده می‌شود روش پیشنهادی نسبت به یکی از بهترین روش‌های ابتکار ساختنی موجود یعنی QB که در ابزار linkern استفاده شده است، به‌لحاظ طول دور نهایی برتری مطلق دارد.

جدول ۴: مقایسه روش‌های savings2 و QB برای نمونه‌هایی از کشورها

مجموعه داده	نام کشور	اندازه	savings2	QB
usa13509	آمریکا	۱۳۵۰۹	۲۱۶۵۴۹۰۰	۲۳۴۱۸۴۷۵
brd14051	آلمان	۱۴۰۵۱	۵۰۲۹۶۳	۵۴۵۳۸۰
d15112	آلمان	۱۵۱۱۲	۱۶۸۲۳۳۰	۱۷۹۴۱۸۸
d18512	آلمان	۱۸۵۱۲	۶۹۰۲۷۳	۷۳۵۳۶۵
ch71009	چین	۷۱۰۰۹	۴۹۱۴۵۴۰	۵۲۹۵۶۱۸

## ۵- نتایج و پیشنهادها

در این مقاله روشی برای حل مسئله فروشنده دوره‌گرد با معیار فاصله جغرافیایی بر روی کره زمین ارائه شد. این روش در ابتدا داده‌های مکانی را با استفاده از تصویرسازی نقشه از فضای جغرافیایی به فضای اقلیدسی منتقل می‌کند و سپس نسخه بهبودیافته‌ای از روش ابتکاری پسانداز را بر روی این نمونه جدید اعمال می‌کند. نتایج تجربی نشان‌دهنده برتری نسبی ۱۰٪ مربوط به روش پیشنهادی به‌لحاظ کیفیت جواب نهایی برحسب طول دور نسبت به روش QB است. همچنین روش پیشنهادی در مقایسه با سایر روش‌های ابتکاری ساختنی بسیار سریع‌تر هست و برای کاربردهای نزدیک به بلادرنگ و پویا [۲۷، ۲۸] مناسب به‌نظر می‌رسد. نتیجه اینکه این مطالعه نشان داد استفاده از روش تبدیل فضا برای حل GEOTSP مبتنی بر تصویرسازی نقشه نسبت به تلاش برای حل آن در فضای اولیه در صورت انتخاب مناسب روش تصویرسازی مورد استفاده موفقیت قابل توجهی به‌لحاظ کیفیت و زمان دستیابی به پاسخ خواهد داشت. استفاده از مقیاس‌گذاری چندبعدی (MDS) برای تبدیل فضای مسئله فروشنده دوره‌گرد با معیار فاصله جغرافیایی یکی از روش‌های توسعه ایده این مقاله است که در کارهای آتی به آن توجه خواهد شد. ایده دیگر استفاده از روش پیشنهادی برای کاربردهای دنیای واقعی است. نمونه‌ای از این کاربرد ها در [۲۹] در مورد برنامه‌ریزی مسیر برای هواپیمای بدون سرنشین ارائه شده است. ترکیب روش پیشنهادی با یکی از الگوریتم‌های تکاملی جدید مانند الگوریتم بهینه‌سازی فاخته [۳۰] نیز از جمله ایده‌های ادامه کار جاری محسوب می‌شود.

## مراجع

[1] E. L. Lawler, *The Travelling Salesman Problem: A Guided Tour of Combinatorial Optimization*: John Wiley & Sons, 1985.

[2] R. Dewil, P. Vansteenwegen, and D. Cattrysse, "A review of cutting path algorithms for laser cutters," *The International Journal of Advanced Manufacturing Technology*, vol. 87, no. 5, pp. 1865-1884, 2016.



- [۲۹] هوشنگ جعفری و حامد خراطی، «طراحی و توسعه سیستم برنامه‌ریزی مسیر سه‌بعدی برای هواپیمای بدون سرنشین»، مجله مهندسی برق دانشگاه تبریز، جلد ۴۶، شماره ۳، صفحه ۹۴-۸۳، ۱۳۹۵.
- [۳۰] زینب صادقی چوبنلی و سیدمحمدحسین معطر، «زمان‌بندی سیستم‌های تولید کارگاهی انعطاف‌پذیر با استفاده از الگوریتم جستجوی فاخته بهبودیافته با خوشه‌بندی مارکوف و پرواز لوی»، مجله مهندسی برق دانشگاه تبریز، جلد ۴۶، شماره ۳، صفحه ۱۹۴-۱۸۵، ۱۳۹۵.
- [25] B. L. Golden, T. L. Magnanti, and H. Q. Nguyen, "Implementing vehicle routing algorithms," *Networks*, vol. 7, no. 2, pp. 113-148, 1977.
- [26] J. P. Snyder, "Album of Map Projections, United States Geological Survey Professional Paper," *United States Government Printing Office*, vol. 1453, 1989.
- [27] H. Zhang *et al.*, "The Steiner traveling salesman problem with online advanced edge blockages," *Computers & Operations Research*, vol. 70, pp. 26-38, 2016.
- [28] T. Cheong, and C. C. White, "Dynamic traveling salesman problem: Value of real-time traffic information," *IEEE Transactions on Intelligent Transportation Systems*, vol. 2, no. 13, pp. 619-630, 2012.

## زیر نویس‌ها

<sup>۹</sup> به این مجموعه داده در TSPLIB به‌عنوان معیار ارزیابی در مسابقات اشاره شده است.

<sup>۱۰</sup><http://www.math.uwaterloo.ca/tsp/concorde/downloads/codes/cygwin/linkern.exe.gz>

<sup>۱۱</sup> در بین روش‌های ابتکاری روش QB جدیدتر است و به مصالحه بهتری بین زمان اجرا و کیفیت جواب نهایی برحسب طول دور دست می‌یابد. بنابراین در ادامه این بخش این روش برای سایر مقایسه‌ها انتخاب شد.

<sup>۱۲</sup> Minimum Spanning Tree (MST)

<sup>۱۳</sup> برای اندازه ۴۰۰۰۰ برنامه MTS به‌دلیل نامعلومی با خطا مواجه شد.

<sup>۱</sup> Combinatorial algorithms

<sup>۲</sup> <http://www.math.uwaterloo.ca/tsp/concorde/downloads/codes/cygwin/linkern.exe.gz>

<sup>۳</sup> Geographic Information System (GIS)

<sup>۴</sup> Map projection

<sup>۵</sup> <http://www.math.uwaterloo.ca/tsp/world/countries.html>

<sup>۶</sup> Quick Boruvka (QB)

<sup>۷</sup> Heap

<sup>۸</sup> Equidistant