

روش جدید برای حل مسئله مسیریابی وسایل نقلیه با استفاده از

الگوریتم تکاملی بهبود یافته

سپهر ابراهیمی مود^۱، محمد مسعود جاویدی^۲، محمدرضا خسروی^۳

تاریخ دریافت: ۱۳۹۸/۰۷/۳۰

تاریخ پذیرش: ۱۳۹۸/۱۰/۰۷

چکیده

امروزه مسئله مسیریابی وسایل نقلیه، یکی از موضوعات پرکاربرد در موضوعات صنعتی، نظامی و حتی امنیتی است که با هدف افزایش کارایی سیستم‌های حمل و نقل تعریف شده است. مسئله مسیریابی وسیله نقلیه با شرایط برداشت و تحویل هم‌زمان محموله از جمله این مسائل می‌باشد. این مسئله از نظر پیچیدگی محاسباتی در مجموعه مسائل سخت (NP-hard) قرار می‌گیرد؛ بنابراین یافتن بهترین پاسخ برای این مسائل در زمان اجرایی قابل قبول، امکان پذیر نیست. استفاده از الگوریتم‌های فرا ابتکاری از جمله الگوریتم جست و جوی گرانشی یکی از روش‌هایی است که به وسیله آن‌ها می‌توان جواب‌هایی مناسب و در زمان محاسباتی قابل قبول به دست آورد. در این مقاله، با استفاده از این الگوریتم، روشی برای حل مسئله مسیریابی وسایل نقلیه با شرایط برداشت و تحویل هم‌زمان ارائه شده است. با استفاده از این روش، علاوه بر محاسبه مسیرهای مناسب برای انجام خدمات، تعداد بهینه وسایل نقلیه برای فرایند خدماتی نیز تعیین می‌شود. برای ارزیابی کارایی روش پیشنهادی در این مقاله، روش پیشنهادی شبیه‌سازی شده و روی مجموعه داده‌ای استاندارد که برای این دسته از مسائل تعریف شده، اجرا شده است. نتایج تجربی و شبیه‌سازی نشان می‌دهد که این روش، با وجود سادگی در روش پیاده‌سازی و اجرا، دارای اثربخشی بهتری نسبت به الگوریتم‌ها و روش‌های موجود است.

کلمات کلیدی: الگوریتم جست و جوی گرانشی، الگوریتم‌های فرا ابتکاری، برداشت و تحویل هم‌زمان، تابع بولتزمن، مسیریابی وسایل نقلیه.

^۱ استاد دانشگاه شهید باهنر کرمان، دکتری علوم کامپیوتر/هوش مصنوعی و نویسنده مسئول، s.ebrahimi.mood@gmail.com

^۲ دانشیار دانشگاه شهید باهنر کرمان، دکتری علوم کامپیوتر، javidi@uk.ac.ir

^۳ دانشگاه و پژوهشگاه عالی دفاع ملی و تحقیقات راهبردی، پژوهشکده آآمد، فناوری‌های دفاعی و پدافند غیرعامل، تهران، ایران، morekhosravi@gmail.com

۱. مقدمه

مسیریابی وسیله نقلیه^۱ یک نام کلی برای دسته‌ای از مسائل است که در آن‌ها یک ایستگاه^۲ ثابت و با موقعیتی از پیش تعیین شده، شامل تعدادی وسیله نقلیه تعریف شده است. همچنین تعدادی کاربر^۳ در محیط مسئله تعریف شده است که وسایل نقلیه وظیفه سرویس‌دهی به کاربرها را دارند. در بسیاری از این مسائل، این خدمت شامل جابه‌جایی و رساندن محموله، از ایستگاه ابتدایی به کاربران است. هر وسیله نقلیه قابلیت جابه‌جایی و انتقال تعدادی محموله جهت انجام خدمت‌رسانی به یک یا چند کاربر را دارد. هدف از حل این مسئله، یافتن مسیرهایی برای هر کدام از این وسایل نقلیه است، به طوری که در کمترین زمان ممکن و با کمترین هزینه ممکن، بیشترین خدمات توسط وسایل نقلیه انجام شود. درعین حال تمام محدودیت‌ها، شرایط و قیود موجود در مسئله نیز رعایت شود.

مسئله مسیریابی وسایل نقلیه به صورت استاندارد (Christofides, 1976) توسط کریستفیدز تعریف شده است. در این مقاله، مسئله شامل یک ایستگاه مشخص و ثابت، مجموعه وسایل نقلیه ثابت و یکسان و مجموعه مشخصی کاربر که در انتظار دریافت محموله از ایستگاه هستند و در محیطی مشخص و از پیش تعریف شده قرار گرفته‌اند، تعریف شده است؛ اما با توجه به وضعیت‌های مختلف موجود در کاربردهای واقعی، می‌توان شرایط مختلف و قیود متفاوتی را برای این مسئله در نظر گرفت. توث و ویگو در (Toth & Vigo, 2002)، به صورت دقیق مسئله استاندارد مسیریابی وسایل نقلیه و همچنین انواع مختلف این مسئله با شرایط و قیود مختلف در کاربردهای واقعی را بررسی کرده‌اند. مسئله مسیریابی وسایل نقلیه از نظر پیچیدگی

¹ Vehicle Routing Problem (VRP)

² depot

³ customer

محاسبات از دسته مسائل NP-hard محسوب می‌شود؛ بنابراین حل این مسئله و یافتن بهترین جواب برای این مسئله با استفاده از روش‌های برنامه‌ریزی خطی و مدل‌های تئوری در عمل بسیار طولانی و غیرقابل استفاده است و کاربردی نمی‌باشد.

محققان برای یافتن جواب در مسائلی که روش‌های دقیق بسیار زمان‌بر هستند و عملاً یافتن جواب دقیق برای مسئله را در زمان معقول غیرممکن می‌سازد، با استفاده از الگوریتم‌های فرا ابتکاری^۱ تخمین مناسبی برای جواب‌ها پیدا می‌کنند. استفاده از الگوریتم‌های بهینه‌سازی برای حل مسئله مسیریابی وسایل نقلیه نیز بسیار رایج است و محققان برای حل این مسئله از روش‌های فرا ابتکاری استفاده کرده‌اند. جین و همکاران در (Ai & Kachitvichyanukul, 2009) با استفاده از الگوریتم بهینه‌سازی جمعیت ذرات^۲ روش جدیدی برای حل حالت خاصی از مسئله مسیریابی وسایل نقلیه، به نام مسئله مسیریابی وسایل نقلیه با شرایط برداشت و تحویل هم‌زمان^۳ محموله ارائه کردند. در این مقاله، ابتدا اهداف و شرایط موجود در مسئله به صورت تابع بهینه‌سازی فرموله شده‌اند. سپس، الگوریتم بهینه‌سازی جمعیت ذرات با توجه به شرایط موجود در تعریف مسئله، بهبود یافته و این الگوریتم برای یافتن مسیرهایی که در آن‌ها وسایل نقلیه علاوه بر ارسال محموله به کاربران، از آن‌ها اشیا و محموله‌ای را دریافت می‌کند و به ایستگاه اولیه باز می‌گردانند، استفاده می‌شود. در این مقاله، تعداد بهینه وسایل نقلیه مورد نیاز برای انجام خدمات مورد نظر کاربران محاسبه نشده است و تعداد وسایل نقلیه به صورت ثابت و از پیش تعیین شده در نظر گرفته شده است. همچنین، نتایج شبیه‌سازی موجود در این مقاله نشان می‌دهد که الگوریتم بهینه‌سازی جمعیت ذرات که در این مقاله استفاده شده است، کارایی مناسبی برای حل مسئله بهینه‌سازی نداشته و دچار همگرایی زودرس شده است.

¹ Meta-heuristic algorithms

² Particle Swarm Optimization (PSO)

³ Vehicle Routing Problem with Simultaneous Pickup and Delivery (VPRSPD)

همچنین جانتینگ در (J. T. Zhang & Qiao, 2013) با استفاده از الگوریتم بهینه‌سازی جمعیت ذرات بهبودیافته، راه‌حلی برای مسئله مسیریابی وسایل نقلیه تعریف کرده است. او در این مقاله، برای بهبود عملکرد الگوریتم بهینه‌سازی جمعیت ذرات، عملگرهای خودیادگیر به این الگوریتم اضافه کرده است و سپس از این الگوریتم بهبودیافته برای یافتن بهترین جواب‌های مسئله استفاده شده است. در این مقاله، ویژگی‌های وسایل نقلیه و همین‌طور کاربران موجود در مدل، یکسان در نظر گرفته شده است؛ اما در مسائل کاربردی، تفاوتی بین وسایل نقلیه و همچنین نوع درخواست‌ها و فعالیت‌های کاربران وجود دارد که لزوماً شرایط تعریف آن‌ها یکسان نیست. همچنین او در (Yao, Yu, Hu, Gao, & Zhang, 2016) الگوریتمی برای حل نوع خاصی از مسئله مسیریابی وسایل نقلیه با مجموعه‌ای از ایستگاه‌ها با استفاده از الگوریتم‌های فرا ابتکاری بهبودیافته ارائه داده است. در این مقاله، مجموعه ایستگاه‌ها از یک تولیدکننده محموله را دریافت می‌کنند و سپس با استفاده از وسایل نقلیه، این محموله‌ها را به کاربران موجود در فضای تعریف شده، می‌رسانند. در این روش، کاربران تنها قابلیت دریافت محموله را دارند و امکان ارسال محموله برای آن‌ها وجود ندارد. در مقاله (Gupta & Saini, 2017)، الگوریتم‌های فرا ابتکاری و تأثیر این الگوریتم‌ها در حل مسئله مسیریابی وسایل نقلیه به صورت دقیق بررسی شده‌اند و نتایج آن‌ها با هم مقایسه شده‌اند.

الگوریتم جست‌وجوی گرانشی^۱ یکی از الگوریتم‌های نوین فرا ابتکاری است که از قوانین جاذبه و حرکت نیوتون الهام گرفته شده است. الگوریتم جست‌وجوی گرانشی که توسط راشدی در سال ۲۰۰۹ تعریف و ارائه شده است (Rashedi, Nezamabadi-Pour, & Saryazdi, 2009, 2010) شامل تعدادی عامل است که در محیط جواب‌های مسئله پراکنده

^۱ Gravitational Search Algorithm (GSA)

شدند. هر عامل، یک جواب شدنی برای مسئله است. عوامل دارای وزنی هستند که کیفیت جواب را نشان می‌دهد. عامل‌ها با جواب‌های بهتر، سنگین‌تر از عواملی هستند که جواب ضعیف‌تری را نشان می‌دهند. با توجه به قانون گرانش نیوتون، عوامل متناسب با جرم خود، به همدیگر نیرو وارد می‌کنند و قانون حرکت نیوتون نیز بیان می‌کند که این نیرو باعث ایجاد حرکت در عوامل می‌شود. عوامل سبک‌تر به سمت عوامل سنگین‌تر که نشان‌دهنده جواب‌های بهتر هستند، حرکت می‌کنند. الگوریتم جست‌وجوی گرانشی یکی از پرکاربردترین الگوریتم‌های فرا ابتکاری است و در بسیار از مسائل کاربردی و واقعی نظیر شبکه حسگر بیسیم (Rafsanjani & Dowlatshahi, 2012)، شبکه‌های عصبی (González, Melin, Valdez, & Prado-Arechiga, 2018)، پزشکی (Kato, Nakamoto, Ishimori, & Togashi, 2016)، اقتصاد (Mohanty, 2016) و رباتیک (Das, Behera, Jena, & Panigrahi, 2016) استفاده شده است. این الگوریتم ویژگی‌ها و قابلیت‌های مختلفی دارد که با توجه به کاربردهای مختلف از آن‌ها استفاده می‌شود.

کاوش^۱ و بهره‌وری^۲ از جمله قابلیت‌های الگوریتم‌های بهینه‌سازی هستند. کاوش به معنای توانایی الگوریتم بهینه‌سازی در پیدا کردن نقاط جدید و جست‌وجوی فضاهای جدید در فضای جواب‌های مسئله است. بهره‌وری نیز، توانایی جست‌وجوی الگوریتم پیرامون بهترین جواب، به‌منظور بهتر کردن جواب فعلی است. در ابتدای فرایند محاسباتی مربوط به الگوریتم بهینه‌سازی به دنبال افزایش قدرت کاوش هستیم تا فضای جواب‌های مسئله به‌خوبی جست‌وجو شده و بهترین جواب‌ها پیدا شوند؛ اما هر چه به انتهای اجرای الگوریتم نزدیک

¹ exploration

² exploitation

شویم، به دنبال بهتر کردن جواب مسئله با استفاده از جست‌وجوی محلی پیرامون بهترین جواب خواهیم بود.

تحقیقات زیادی برای بهبود عملکرد الگوریتم جست‌وجوی گرانشی انجام شده است. در مقاله (Ebrahimi, Rashedi, & Javidi, 2015)، توابع جدیدی برای محاسبه وزن عامل‌ها در الگوریتم جست‌وجوی گرانشی ارائه شده است تا به کمک آن‌ها قابلیت‌های کاوش و بهره‌وری الگوریتم، با توجه به روند پیشرفت محاسبات الگوریتم کنترل شود. در این مقاله از توابع جدیدی برای محاسبه جرم عوامل با استفاده از توابع مقیاس‌ساز سیگما^۱ و تابع بولتزمن^۲ استفاده شده است و نتایج شبیه‌سازی‌دهنده بهبود عملکرد این الگوریتم در مقایسه با الگوریتم جست‌وجوی گرانشی است. در (Kherabadi, Mood, & Javidi, 2017) عملگر جدیدی تحت عنوان جهش^۳ به الگوریتم جست‌وجوی گرانشی اضافه شده است. این عملگر باعث فرار از بهینه محلی در مسائلی که به صورت چندمدی^۴ هستند، می‌شود. پارامترهای موجود در این روش با استفاده از یک کنترل فازی تعیین می‌شود، بنابراین کاملاً مستقل از اطلاعات کاربر و با توجه به فرایند محاسباتی الگوریتم، از همگرایی زودرس الگوریتم در بهینه محلی^۵ جلوگیری می‌کند. همچنین در (Soleimanpour-Moghadam, Nezamabadi-Pour, & Farsangi, 2014)، مدل جدیدی از این الگوریتم با نام الگوریتم جست‌وجوی گرانشی کوانتومی تعریف شده که سرعت همگرایی بسیار بالایی دارد. علاوه بر این در (Hatamlou, 2013) عملگر سیاه‌چاله^۶ الهام گرفته از علم ستاره‌شناسی و نجوم به فرایند محاسباتی این الگوریتم اضافه شده است. این عملگر باعث بهبود قابلیت بهره‌وری الگوریتم می‌شود. همچنین در (Shams,

¹ Sigma scaling

² Boltzmann

³ mutation

⁴ Multi-modal

⁵ Local optimum

⁶ Black-hole

(Rashedi, & Hakimi, 2015) از ایده خوشه‌بندی در الگوریتم جست‌وجوی گرانشی استفاده شده است که نتیجه آن، کاهش میزان محاسبات مورد نیاز برای یافتن بهترین پاسخ مسئله است؛ بنابراین این الگوریتم می‌تواند در زمانی کوتاه، جواب‌های مناسب برای مسئله پیدا کند. برخی از پژوهش‌ها و ایده‌های مطرح شده برای بهبود عملکرد الگوریتم جست‌وجوی گرانشی در (Sabri, Puteh, & Mahmood, 2013) گردآوری شده است.

در این مقاله از الگوریتم جست‌وجوی گرانشی بهبود یافته، برای حل مسئله مسیریابی وسایل نقلیه با شرایط برداشت و تحویل هم‌زمان محموله استفاده شده است. ابتدا مسئله مفروض را با توجه به شرایط و حالات مختلفی که امکان رخ دادن آن در فضای واقعی ممکن است، تعریف شده و به صورت برنامه‌ریزی خطی فرموله شده است. سپس از الگوریتم جست‌وجوی گرانشی حقیقی برای یافتن جواب مناسب برای این مسئله استفاده شده است. همچنین، با استفاده از این الگوریتم، تعداد بهینه وسایل نقلیه برای انجام فرایندهای خدماتی مربوط به کاربران نیاز است، به دست خواهد آمد.

در ادامه این مقاله، بخش‌های مختلف آن به صورت زیر سازماندهی شده است. در بخش دوم، تعاریف مقدماتی مسئله مسیریابی وسایل نقلیه با شرایط ارسال و دریافت هم‌زمان محموله بیان شده است. این مسئله و تمام شرایط و قیود مربوطه تعریف و به صورت مدل ریاضی، با استفاده از برنامه‌ریزی خطی فرموله شده‌اند. در بخش ۳، تعاریف و مباحث مربوط به الگوریتم جست‌وجوی گرانشی حقیقی بهبود یافته که در این مقاله برای حل مسئله از آن استفاده شده است، به صورت دقیق شرح داده شده است. در ادامه و در بخش چهارم، جزئیات مربوط به شبیه‌سازی این الگوریتم روی مجموعه داده‌های استاندارد مورد بررسی قرار گرفته است. همچنین نتایج تجربی حاصل از اجرای این الگوریتم و سایر الگوریتم‌های معتبر مورد تحلیل و

بحث قرار گرفته‌اند. در نهایت بخش پنجم این مقاله شامل نتیجه‌گیری در مورد روش پیشنهادی این مقاله و آنالیز نتایج به دست آمده از آن می‌باشد.

۲. روش شناسی

در این بخش، مسئله مسیریابی وسایل نقلیه با شرایط برداشت و تحویل هم‌زمان به صورت دقیق تعریف شده و روابط و قیود مورد استفاده تعریف می‌شود. سپس راه‌حل پیشنهادی برای حل این مسئله و همچنین روش ارزیابی الگوریتم پیشنهادی مورد بررسی قرار گرفته است.

مسئله مسیریابی وسایل نقلیه با شرایط برداشت و تحویل هم‌زمان، از جمله مسائل NP-Hard است؛ بنابراین به دست آوردن جواب دقیق برای این مسئله در زمان اجرایی قابل قبول، امکان‌پذیر نیست. برای حل اینگونه از مسائل، از الگوریتم‌های ابتکاری استفاده می‌شود. در این دسته از الگوریتم‌ها، تخمین مناسبی از جواب مسئله، در زمان اجرایی قابل قبولی محاسبه می‌شود. در این مقاله، از روش جست‌وجوی گرانشی بهبودیافته که از جمله الگوریتم‌های فرا ابتکاری است که قابلیت کنترل کاوش و بهره‌وری را دارد، برای حل مسئله مسیریابی وسایل نقلیه استفاده شده است. برای ارزیابی کارایی روش پیشنهادی، ابتدا کارایی الگوریتم تکاملی بهبودیافته با سایر الگوریتم‌های فرا ابتکاری مقایسه خواهد شد. سپس، الگوریتم پیشنهادی برای حل مسئله مسیریابی وسایل نقلیه شبیه‌سازی شده و بر روی دو دسته از توابع استاندارد از پیش تعریف شده برای این مسئله، استفاده شده است. نتایج به دست آمده از شبیه‌سازی الگوریتم پیشنهادی، با نتایج سایر روش‌های ارائه شده برای این موضوع مقایسه شده است. در ادامه تعریف دقیق مدل محاسباتی این مسئله تعریف شده است.

مدل دقیق مسئله مسیریابی وسایل نقلیه با شرایط برداشت و تحویل هم‌زمان به صورت زیر تعریف می‌شود (Ai & Kachitvichyanukul, 2009). $G = (V, A)$ تعریفی برای یک گراف است که در آن $V = \{v_0, v_1, \dots, v_n\}$ مجموعه رئوس گراف و $A = \{(v_i, v_j) | v_i, v_j \in V, i \neq j\}$ مجموعه یال‌های گراف در نظر گرفته می‌شود. متناظر با هر یک از اعضای مجموعه یال‌های A ، یک ماتریس فاصله (d_{ij}) و یک ماتریس زمان سفر (t_{ij}) تعریف می‌شود. راس v بیانگر مکان ایستگاه در فضای مسئله است که در آن m وسیله نقلیه متناظر استقرار دارند. سایر رئوس گراف متناظر با n کاربر موجود در مسئله هستند. هر کدام از این کاربرها محموله‌ای با مقدار p_i واحد برای برداشت و محموله‌ای با q_i واحد برای تحویل و همچنین s_i واحد زمانی برای پردازش و انجام سرویس در اختیار دارد. هر کدام از وسایل نقلیه دارای یک هزینه ثابت f ، یک هزینه متغیر g برای هر واحد مسافت، ظرفیت Q و محدودیت زمانی خدمات سرویس D است. پاسخ مسئله مسیریابی وسایل نقلیه با شرایط برداشت و تحویل هم‌زمان محموله شامل یافتن حداکثر m مسیر مختلف است به طوری که شرایط زیر در آن صدق کند:

- ۱- پایان و شروع هر مسیر، ایستگاه v باشد.
- ۲- هر کاربر دقیقاً یک بار و توسط دقیقاً یک وسیله نقلیه ملاقات می‌شود.
- ۳- میزان محموله‌ای که به هر وسیله نقلیه تخصیص داده شده، از ظرفیت در نظر گرفته شده برای آن وسیله نقلیه تجاوز نکند.
- ۴- مجموع طول بازه زمانی سرویس خدمات، از محدودیت تعریف شده برای هر وسیله نقلیه تجاوز نکند.
- ۵- مجموع هزینه مربوط به خدمات و مسیرها کمینه شود.

فرمول‌بندی ریاضی مسئله مسیریابی وسایل نقلیه با شرایط برداشت و تحویل همزمان که از نوع مسائل جریان در شبکه^۱ است، با توجه به تعاریف و شرایط بیان‌شده در بالا و با استفاده از برنامه‌ریزی خطی در ادامه ارائه شده است. متغیرهای تصمیم‌گیری موجود در این مسئله به شرح زیر است:

x_{ijk} : متغیر دودویی برای مشخص کردن تردد وسیله نقلیه k ام در مسیری از رأس i به رأس j . اگر وسیله نقلیه مذکور در این مسیر باشد، مقدار این متغیر ۱، در این صورت ۰ خواهد بود.

y_{ijk} : مقدار محموله موجود در وسیله نقلیه k ام، در مسیر بین رئوس i و j .

δ_{ik} : زمان شروع خدمات سرویس مشتری i ام توسط وسیله نقلیه k ام.

با توجه به تعاریف فوق برای متغیرهای تصمیم‌گیری، تابع هدف^۲ مناسب برای این مسئله به همراه تمام قیود موجود به صورت زیر تعریف می‌شود.

$$\min Z = f \sum_{k=1}^m \sum_{j=1}^n x_{.jk} + g \sum_{i=1}^n \sum_{j=1}^{n+1} \sum_{k=1}^m d_{ij} x_{ijk} \quad (۱)$$

پاسخ مناسب برای این مسئله، جوابی است که تابع فوق را کمینه کرده و همچنین شرایط و قیود زیر در آن صدق کند.

$$\sum_{i=1}^n \sum_{k=1}^m x_{ijk} = 1, \quad \text{for } 1 \leq j \leq n \quad (۲)$$

$$\sum_{j=1}^n x_{jik} = \sum_{j=1}^{n+1} x_{ijk}, \quad \text{for } 1 \leq i \leq n, 1 \leq k \leq m \quad (۳)$$

$$\sum_{j=1}^n x_{.jk} \leq 1, \quad \text{for } 1 \leq k \leq m \quad (۴)$$

^۱ Network flow-based formulation

^۲ Objective function

$$\delta_{ik} + s_i + t_{ij} - \delta_{jk} \leq (1 - x_{ijk})M, \quad \text{for } 0 \leq i \leq n, 1 \leq j \leq n+1, 1 \leq k \leq m \quad (5)$$

$$\delta_{n+1,k} - \delta_{0,k} \leq D, \quad \text{for } 1 \leq k \leq m \quad (6)$$

$$y_{ijk} \leq x_{ijk}Q, \quad \text{for } 0 \leq i \leq n, 1 \leq j \leq n+1, 1 \leq k \leq m \quad (7)$$

$$\sum_{j=1}^n y_{.jk} = \sum_{j=1}^n q_j \sum_{i=0}^n x_{ijk}, \quad \text{for } 1 \leq k \leq m \quad (8)$$

$$\sum_{i=0}^n y_{ijk} + (p_j - q_j) \sum_{i=0}^n x_{ijk} = \sum_{j=1}^{n+1} y_{jik}, \quad \text{for } 1 \leq j \leq n, 1 \leq k \leq m \quad (9)$$

$$x_{ijk} \in \{0, 1\}, \quad \text{for } 0 \leq i \leq n, 1 \leq j \leq n+1, 1 \leq k \leq m \quad (10)$$

$$y_{ijk} \geq 0, \quad \text{for } 0 \leq i \leq n, 1 \leq j \leq n+1, 1 \leq k \leq m \quad (11)$$

$$\delta_{ik} \geq 0, \quad \text{for } 0 \leq i \leq n+1, 1 \leq k \leq m \quad (12)$$

تابع هدف (۱) نشان می‌دهد که مدل فوق با هدف کمینه کردن هزینه وسایل نقلیه در مسیرها با توجه به هزینه‌های ثابت و متغیر از پیش تعیین شده، تعریف شده است. روابط و فرمول‌های (۲) تا (۱۲) قیود و شرایط مسئله را تعیین می‌کند. قید دوم بررسی می‌کند که خدمات مربوط به هر کاربر دقیقاً توسط یک وسیله نقلیه انجام شود. قید شماره (۳) تضمین می‌کند که هر وسیله نقلیه که به موقعیت یک کاربر وارد شود، از آن خارج شده باشد. همچنین هر وسیله برای سرویس‌دهی در حداکثر یک مسیر انتخاب شود که این مسیر می‌تواند شامل خدمات مربوط به تعدادی کاربر باشد (۴).

قید شماره (۵) در مورد زمان اجرای خدمات مربوط به کاربران است. اگر وسیله نقلیه k ام، خدمات مربوط به کاربر j را بعد از انجام سرویس خدمات کاربر i انجام دهد ($x_{ijk} = 1$)، در این صورت زمان شروع خدمات مربوط به کاربر j باید بیشتر از مجموع زمان شروع سرویس مربوط به کاربر i ام، مدت زمان اجرای خدمات مربوط به سرویس این کاربر و همچنین زمان مربوط به جابه‌جایی بین مکان دو کاربر باشد ($\delta_{ik} + s_i + t_{ij} \leq \delta_{jk}$). برای حالتی که خدمات مربوط به کاربر j بعد از انجام خدمات مربوط به کاربر i انجام نشود، هیچ رابطه‌ای بین زمان شروع سرویس‌ها و مدت زمان فرایندهای خدماتی برقرار نیست. شرط شماره (۶) تضمین می‌کند که مدت زمان اجرای فرایندهای خدماتی کاربران، توسط هر یک از وسایل نقلیه از محدودیت زمانی تعریف شده برای آن وسیله تجاوز نکند.

قیدهای شماره (۷) تا (۹) هم شرایط مربوط به وزن محموله‌ها و محدودیت وزنی برای وسایل نقلیه را تشریح می‌کنند. شرط شماره (۷) تضمین می‌کند که وسایل نقلیه، سرویس خدمات مربوط به یک کاربر را انجام می‌دهند در صورتی که وزن محموله مورد نظر بیشتر از حد از پیش تعیین شده برای وسیله نقلیه نباشد. قید شماره (۹) برای تأیید برابری میزان محموله‌های خارج شده از ایستگاه توسط یک وسیله نقلیه، با مجموع میزان محموله‌های رسیده به همه کاربران و خدمات انجام شده توسط همان وسیله نقلیه، استفاده می‌شود. قید شماره (۹) برای بررسی توازن بار محموله‌های موجود در یک وسیله نقلیه، پیش و پس از انجام سرویس خدمات مربوط به یک کاربر است. همچنین قیدهای شماره (۱۰) تا (۱۲) برای تأیید وجود مقدار متغیرهای تصمیم‌گیری در دامنه‌های از پیش تعیین شده استفاده می‌شود.

با توجه به تابع هدف تعریف شده در رابطه (۱) و قیود موجود در این تابع هدف، مسئله مسیریابی وسایل نقلیه با شرایط برداشت و تحویل هم‌زمان از جمله مسائل سخت است که

امکان به دست آوردن پاسخ دقیق برای آن در زمان اجرایی معقول، امکان پذیر نیست. در این مقاله، از الگوریتم جست و جوی گرانشی بهبود یافته برای به دست آوردن تخمین مناسبی برای جواب این مسئله بهینه سازی استفاده شده است. در این روش، کنترل قابلیت های کاوش و بهره وری باعث افزایش کارایی الگوریتم در محاسبه مقادیر بهینه و جواب های مسئله شده است. در نتیجه، با احتمال کمتری در بهینه های محلی گرفتار خواهد شد؛ بنابراین، برای استفاده در مسائل پیچیده و چندمدی همانند مسئله فوق، بسیار مناسب است. در ادامه مقاله، روش پیشنهادی برای حل این مسئله شرح داده شده است و کارایی این روش، با سایر الگوریتم های ارائه شده برای حل این مسئله مقایسه شده است.

۳. روش ارائه شده برای مسئله مسیریابی وسایل نقلیه با شرایط برداشت و تحویل هم زمان

در این بخش از مقاله، ابتدا عملگرهای مربوط به الگوریتم جست و جوی گرانشی بهبود یافته تعریف شده و نحوه انجام فرایند محاسباتی این الگوریتم به صورت دقیق بررسی می شود. در ادامه، نحوه استفاده از این الگوریتم در مسئله مسیریابی وسایل نقلیه با شرایط برداشت و تحویل هم زمان بیان می شود. نحوه نمایش پاسخ های مربوط به این مسئله در الگوریتم جست و جوی گرانشی بهبود یافته و همچنین فرایندهای مربوط به رمزگشایی پاسخ های الگوریتم بهینه سازی، در این بخش به تفصیل مورد بحث قرار می گیرد. همچنین، نحوه به دست آوردن تعداد بهینه وسایل نقلیه، برای انجام خدمات مربوط به کاربران نیز در این بخش بیان می شود.

۳-۱- الگوریتم جست و جوی گرانشی بهبود یافته

الگوریتم جست و جوی گرانشی، یکی از جدیدترین الگوریتم‌های فرا ابتکاری ارائه شده که از قوانین حرکت و جاذبه نیوتن الهام گرفته شده است. نتایج شبیه‌سازی‌ها نشان می‌دهد که این الگوریتم، با وجود ساده بودن فرایند محاسباتی و اجرایی، کارایی بسیار خوبی در حل مسائل بهینه‌سازی دارد (Rashedi, 2010 et al., 2009). در این الگوریتم، عوامل^۱ ذراتی هستند که دارای جرمی مشخص و متناسب با تابع هدف موجود در مسئله می‌باشند. هر چه این ذرات جواب‌های بهتری در فضای شدنی^۲ مسئله نشان دهند، وزن بیشتری خواهند داشت. با توجه به قانون جاذبه نیوتون، همه عوامل که دارای جرم هستند، به یکدیگر نیروی گرانشی وارد می‌کنند و اگر این عامل سنگین‌تر باشد (بیانگر جواب بهتری برای مسئله باشد) با نیروی بیشتری سایر عوامل را به سمت خود جذب خواهد کرد.

در الگوریتم جست و جوی گرانشی با N عامل، مکان عامل i ام در فضای جست و جوی مسئله، به صورت زیر نشان داده می‌شود.

$$X_i = (x_i^1, \dots, x_i^d, \dots, x_i^m), i = 1, 2, \dots, N \quad (13)$$

که در آن مکان عامل i ام در بعد d ام مسئله است و m نیز نشان‌دهنده بعد فضای جست و جوی شدنی مسئله است. مقدار نیروی گرانشی وارد شده به عامل j در بعد d و در زمان t ، با توجه به قانون گرانش نیوتون با توجه به رابطه (۱۴) محاسبه می‌شود.

$$F_j^d(t) = \sum_{j \in Kbest, j \neq i} rand \cdot G(t) \frac{(M_j \times M_i)}{(R_{ij}(t) rPower + \epsilon)} (x_j^d(t) - x_i^d(t)) \quad (14)$$

¹ Agents

² Feasible area

در این رابطه، $rand$ یک عدد تصادفی تولیدشده در بازه (۰ و ۱)، $Kbest$ مجموعه‌ای از k بهترین جواب، M_i و M_j معادل با جرم عوامل i و j است. $G(t)$ ثابت گرانش است که به صورت تابعی نزولی از زمان تعریف می‌شود (Rashedi et al., 2009). $R_{ij}(t)$ فاصله اقلیدسی عوامل مذکور در زمان t است. ثابت $rPower$ در این مقاله مشابه با (Rashedi et al., 2009) برابر با یک در نظر گرفته می‌شود. همچنین ε یک عدد بسیار کوچک و ثابت فرض می‌شود.

با توجه به قانون حرکت نیوتون، نیروی وارد شده به هر عامل، موجب ایجاد شتاب و حرکت در آن عامل خواهد شد. در الگوریتم جست‌وجوی گرانشی، شتاب و سرعت تولید شده در هر عامل، همچنین مکان جدید عامل در بعد d ام، با توجه به روابط زیر محاسبه می‌شود.

$$a_i^d(t) = \frac{F_i^d(t)}{M_i(t)} \quad (15)$$

$$V_i^d(t+1) = rand_i \cdot V_i^d(t) + a_i^d(t) \quad (16)$$

$$X_i^d(t+1) = X_i^d(t) + V_i^d(t+1) \quad (17)$$

با توجه به رابطه (۱۴) هر چه جرم عوامل بیشتر باشد، نیروی جاذبه وارد شده توسط این عامل، به سایر ذرات بیشتر خواهد بود. علاوه بر این با توجه به روابط (۱۵) تا (۱۷)، هر چه جرم عامل بیشتر باشد، میزان شتاب تولید شده برای ایجاد حرکت در آن عامل کمتر خواهد بود. در نتیجه عوامل سنگین‌تر یا معادلاً عوامل با پاسخ‌های بهتر، جابه‌جایی کمتری خواهند داشت؛ بنابراین، نحوه محاسبه جرم‌ها در الگوریتم جست‌وجوی گرانشی، تأثیر زیادی در فرایند پیشرفت الگوریتم خواهد داشت و با استفاده از توابع و تعاریف مناسب برای محاسبه جرم‌ها می‌توان قابلیت‌های الگوریتم نظیر قدرت کاوش و بهره‌وری را کنترل نمود. در این

مقاله، از تابع بولتزمن که در (Ebrahimi et al., 2015) تعریف شده است، برای محاسبه جرم‌ها به صورت زیر استفاده می‌شود.

$$NFit_i(t) = \frac{fit_i(t) - worst(t)}{\sum_{j=1}^N fit_j(t) - worst(t)} \quad (18)$$

$$M_i(t) = \frac{\exp\left(\frac{Nfit_i(t)}{T}\right)}{\langle \exp\left(\frac{Nfit_i(t)}{T}\right) \rangle_t} \quad (19)$$

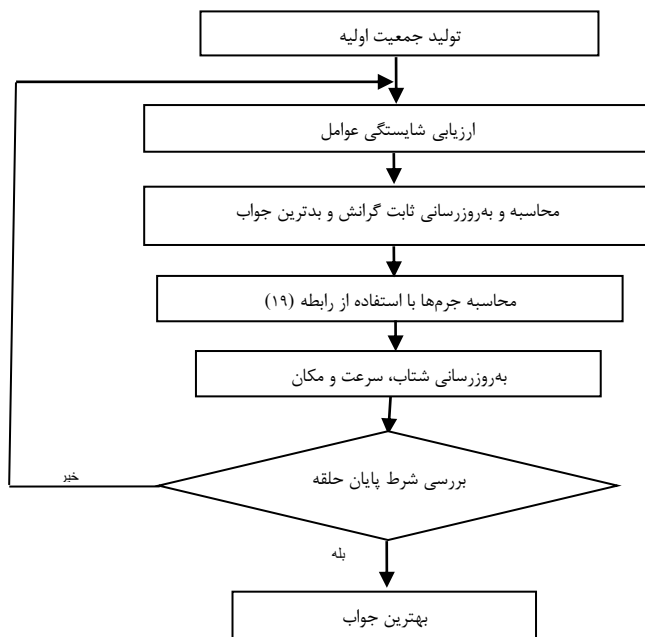
در رابطه (۱۸)، شایستگی^۱ نرمال شده عوامل محاسبه می‌شود. $fit_i(t)$ شایستگی عامل i ام در زمان t را بیان می‌کند. $worst$ هم عامل با ضعیف‌ترین شایستگی ممکن است. عملگر $\langle \rangle_t$ بیانگر مفهوم میانگین‌گیری بین همه عوامل در زمان t است. T نیز تابع دما در الگوریتم ذوب شبیه‌سازی شده^۲ است که به صورت زیر تعریف می‌شود.

$$T = \frac{0.2}{\ln(t)} \quad (20)$$

همان‌طور که در مقاله (Ebrahimi et al., 2015) بیان شده است، این تعاریف برای جرم‌ها، موجب کنترل قدرت کاوش و بهره‌وری و در مجموع بهبود عملکرد و کارایی در الگوریتم جست‌وجوی گرانشی خواهد شد. در ابتدای الگوریتم، جرم عوامل نزدیک به هم بوده و موجب افزایش قدرت کاوش الگوریتم خواهد شد. در نتیجه، در ابتدای الگوریتم فضای شدنی مسئله به خوبی مورد کاوش قرار می‌گیرد. اما هر چه فرایند محاسباتی الگوریتم جلو می‌رود، فاصله جرم عوامل بیشتر شده و این خود موجب افزایش قابلیت بهره‌وری الگوریتم می‌شود. در نتیجه الگوریتم به بهترین جواب همگرا خواهد شد. نمودار جریان مربوط به فرایند محاسباتی الگوریتم جست‌وجوی گرانشی در شکل شماره ۱ آورده شده است.

¹ Fitness

² Simulated annealing



شکل شماره ۱. نمودار جریان الگوریتم جست‌وجوی گرانشی بهبودیافته

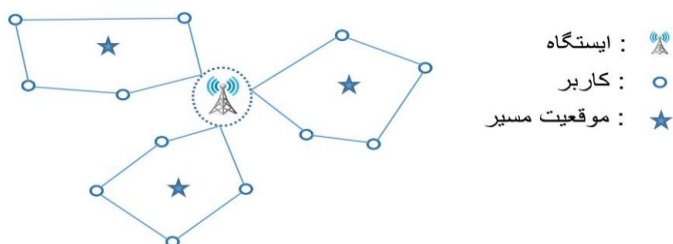
۳-۲- حل مسئله با توجه به الگوریتم جست‌وجوی گرانشی بهبودیافته

در این بخش، از الگوریتم جست‌وجوی گرانشی بهبود یافته که در بخش ۱-۳ معرفی شد، برای حل مسئله مسیریابی وسایل نقلیه با شرایط برداشت و تحویل هم‌زمان و کمینه کردن رابطه (۱) با توجه به شرایط و قيود (۲) تا (۱۲) استفاده شده است. ابتدا نحوه نمایش پاسخ‌های مسئله در الگوریتم بهینه‌سازی که یکی از مهمترین قسمت‌های مؤثر پیاده‌سازی به‌منظور بهبود کارایی روش پیشنهادی است، بررسی خواهد شد. سپس پاسخ‌های به‌دست‌آمده از الگوریتم و نحوه رمزگشایی این پاسخ‌ها به‌صورت دقیق بیان می‌شود.

هر پاسخ الگوریتم جست‌وجوی گرانشی بهبودیافته، شامل دو قسمت است. بخشی از پاسخ مربوط به اطلاعات کاربران و بخشی دیگر مربوط به اطلاعات وسایل نقلیه است. در بخشی از

پاسخ، اولویت مربوط به کاربران مشخص می‌شود. n بعد برای این بخش از پاسخ تخصیص داده می‌شود و در هر بعد، شماره مربوط به یکی از کاربران قرار می‌گیرد. در این صورت، کاربرانی که اندیس متناظر با آن‌ها عدد کوچکتری دارد، دارای اولویت بیشتری برای انجام فرایند خدماتی هستند.

بخش دوم اطلاعات موجود به پاسخ‌ها، با توجه به مفهوم موقعیت مسیر^۱ ارائه شده‌اند. موقعیت مسیر برای هر وسیله نقلیه، مکانی در محدوده فضایی تعریف شده و مشخص شده برای کاربران است که وسیله نقلیه با احتمال بیشتری سرویس خدمات خود را در آن ناحیه انجام می‌دهد. در نتیجه مسیر یک وسیله نقلیه طوری تعیین می‌شود که خدمات مربوط به کاربران پیرامون موقعیت مسیر آن وسیله را سرویس‌دهی کند. شکل شماره ۲ رابطه شهودی بین موقعیت مسیر و مسیر تعیین شده برای هر وسیله نقلیه را نشان می‌دهد. مکان موقعیت مسیر برای هر وسیله نقلیه، در فضای دو بعدی x و y مشخص می‌شود؛ بنابراین فضایی به اندازه $2m$ داده برای مشخص کردن موقعیت مسیر مربوط به m وسیله نقلیه نیاز است.



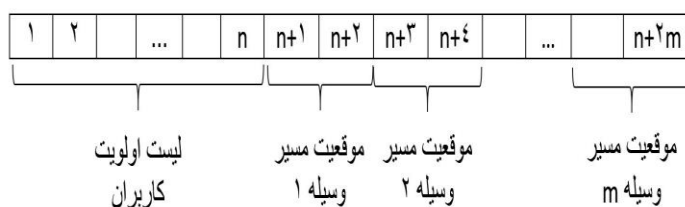
شکل شماره ۲. رابطه بین مسیرهای وسایل نقلیه و همچنین موقعیت مسیر مربوط به وسایل نقلیه

با استفاده از موقعیت مسیر وسیله نقلیه، می‌توان مسیرهای مربوط به هر وسیله نقلیه را مشخص کرد. بعد از مشخص شدن موقعیت مسیر تمام وسایل نقلیه، نزدیکی بین سرویس

¹ Route orientation

کاربر موردنظر از نظر فاصله اقلیدسی، با موقعیت‌های مسیر موجود، می‌تواند معیاری برای تعیین وسیله نقلیه باشد؛ بنابراین، اگر خدمات کاربران، باعث نقض شرایط موجود برای وسیله نقلیه مذکور نشود، به نزدیک‌ترین موقعیت مسیر اختصاص می‌یابد. این روش تخصیص خدمات و تعیین مسیر، باعث کاهش هزینه‌های انجام‌شده برای خدمات و افزایش کارایی وسایل نقلیه خواهد شد.

بنابراین، هر پاسخ الگوریتم جست‌وجوی گرانشی بهبودیافته برای مسئله مسیریابی وسایل نقلیه با شرایط برداشت و تحویل هم‌زمان که در آن n کاربر و m وسیله نقلیه موجود است، $n + 2m$ واحد حافظه نیاز دارد که هر واحد حافظه یک عدد حقیقی است. در هر پاسخ، اطلاعات موجود برای اولویت کاربران برای پاسخگویی و همچنین موقعیت مسیر مربوط به تمام وسایل نقلیه موجود است. شکل شماره ۳ اطلاعات موجود در یکی از عامل‌های الگوریتم جست‌وجوی گرانشی را که معادل با یک پاسخ برای این مسئله است، نمایش می‌دهد.



شکل شماره ۳. اطلاعات موجود در یک عامل الگوریتم جست‌وجوی گرانشی بهبودیافته

رمزگشایی اطلاعات موجود در عوامل الگوریتم و پاسخ‌های ارائه‌شده برای این مسئله، در سه مرحله انجام می‌پذیرد. در مرحله نخست، اطلاعات موجود در n واحد اول حافظه استخراج شده و اولویت مربوط به کاربران تعیین می‌شود. سپس، با توجه به اطلاعات موجود در $2m$ واحد حافظه باقیمانده، مکان دقیق موقعیت مسیر تمام وسایل نقلیه تعیین شده و به وسیله آن

ماتریسی برای اولویت وسایل نقلیه ایجاد می‌شود. ماتریس اولویت وسایل نقلیه، با توجه به فاصله اقلیدسی خدمات کاربران موجود، با مکان موقعیت مسیر وسیله نقلیه مذکور محاسبه می‌شود. اولویت انجام سرویس خدمات مربوط به هر مشتری، با وسیله نقلیه‌ای است که نزدیک‌ترین موقعیت مسیر را به این کاربر داشته باشد. هر سطر در این ماتریس، اولویت وسایل نقلیه را برای یک کاربر مشخص، نشان می‌دهد. در مرحله آخر، یک مسیر با توجه به لیست اولویت کاربران و ماتریس اولویت وسایل نقلیه، تعیین می‌شود. در این مرحله، هر کاربر با توجه به لیست اولویت کاربران انتخاب شده و به یک وسیله نقلیه با توجه به ماتریس اولویت وسایل نقلیه تخصیص داده می‌شود؛ به طوری که تمام قیود و شرایط موجود برای وسیله نقلیه و تعریف مسئله را رعایت کند. با توجه به این روند رمزنگاری، احتمال وجود پاسخی که در آن، یکی از کاربران به هیچ وسیله نقلیه‌ای اختصاص پیدا نکند، وجود دارد. برای حل این موضوع عبارت رگیولاریسینون^۱ به تابع هدف اضافه شده است. بدین معنا که در صورت وجود چنین حالتی، عبارتی با جریمه بسیار بزرگ به شایستگی این عامل اضافه می‌شود. در این صورت، جواب مربوط به این عامل شایستگی بسیار کمی خواهد داشت و الگوریتم از تولید چنین پاسخ‌هایی در مراحل بعدی، جلوگیری به عمل می‌آورد.

در مسائل مسیریابی وسایل نقلیه، تعداد وسایل نقلیه ثابت و از پیش تعیین شده است. در راه‌حل‌های مختلفی که برای این مسئله ارائه شده است، از همه وسایل نقلیه برای انجام خدمات کاربران استفاده می‌شود؛ اما چون هر کدام از این وسایل نقلیه هزینه ثابتی دارند، برای کاهش هزینه‌ها بهتر است تعداد بهینه وسایل نقلیه محاسبه شده و از همان تعداد وسیله در فرایند خدماتی کاربران استفاده شود. یکی از قابلیت‌های روش پیشنهادی و نحوه نمایش

¹ Regularization

جواب‌ها، به دست آوردن تعداد بهینه وسایل نقلیه برای انجام خدمات مورد نیاز کاربران است. در این روش و در مرحله مقداردهی اولیه عوامل، پاسخ‌ها با توجه به تعداد وسایل نقلیه از پیش تعیین شده، m تولید می‌شوند. در ادامه و پس از رمزگشایی پاسخ‌ها و محاسبه مسیرها و ارائه خدمات به همه کاربران، یکی از وسایل نقلیه که کمترین تعداد سرویس را به مشتریان انجام داده، حذف می‌شود؛ به عبارت دیگر، دو واحد از حافظه اختصاص داده شده به هر عامل که معادل با موقعیت مسیر یک وسیله نقلیه بوده است، حذف می‌شود. حال، مسئله رمزگشایی جواب الگوریتم با $m - 1$ وسیله نقلیه انجام می‌شود. در ادامه، وسیله نقلیه دیگری از مجموعه وسایل نقلیه موجود حذف می‌شود و شدنی بودن پاسخ برای حل این مسئله مورد ارزیابی قرار می‌گیرد. این فرایند تا زمانی ادامه می‌یابد که تعداد بهینه وسایل نقلیه محاسبه شود.

۴. نتایج تجربی و محاسباتی

در این بخش، ابتدا برای بررسی کارایی الگوریتم جست‌وجوی گرانشی بهبودیافته، این الگوریتم شبیه‌سازی شده و نتایج آن با نتایج شبیه‌سازی سایر الگوریتم‌های معتبر، روی مجموعه داده‌های استاندارد CEC13^۱ که برای مسائل بهینه‌سازی تعریف شده‌اند (Tang et al., 2007)، مقایسه می‌شود. در ادامه، از این الگوریتم برای حل مسئله مسیریابی وسایل نقلیه با شرایط برداشت و تحویل هم‌زمان استفاده شده است و نتایج شبیه‌سازی آن، با نتایج الگوریتم‌ها و روش‌های معتبر برای حل این نوع مسئله مقایسه شده است. برای این مقایسه، از مجموعه داده‌های استاندارد دل-آمیکو^۲ (Dell'Amico, Righini, & Salani, 2006) و دتلاف^۲ (Dethloff, 2001) که برای این دسته از مسائل تعریف شده‌اند، استفاده شده است. این

^۱ Dell'Amico

^۲ Dethloff

شبیه‌سازی‌ها با استفاده از نرم‌افزارهای MATLAB_R2017b و Visual Studio 2013 روی سیستمی با مشخصات Intel Core i7-6700HQ و 12 GB RAM پیاده‌سازی شده است.

در ابتدا برای بررسی کارایی الگوریتم جست‌وجوی گرانشی بهبودیافته (MO-GSA)، این الگوریتم شبیه‌سازی شده و روی مجموعه داده‌های استاندارد CEC اجرا شده و نتایج این شبیه‌سازی به نتایج سایر الگوریتم‌ها نظیر الگوریتم جست‌وجوی گرانشی استاندارد (GSA) (Rashedi et al., 2009)، الگوریتم جست‌وجوی گرانشی خوشه‌بندی شده (CGSA) (Shams et al., 2015)، GPSO (Noel, 2012) و JADEEP (J. Zhang & Sanderson, 2009) مقایسه شده است. الگوریتم‌ها با 10^5 ارزیابی شایستگی^۱، روی این توابع اجرا شده‌اند. نتایج مقدار خطای محاسبه‌شده توسط الگوریتم‌های مذکور، روی ۵۱ تکرار مختلف و مستقل محاسبه‌شده و میانگین این خطا روی این توابع تست، در جدول شماره ۱ گزارش شده است. در این جدول، بهترین پاسخ برای هر یک از توابع تست به صورت طرح سیاه^۲ نشان داده شده است.

در الگوریتم جست‌وجوی گرانشی بهبود یافته، مقدار α برابر با ۲۰ و مقدار ثابت گرانش اولیه (G_0) برابر با ۱۰۰ در نظر گرفته شده است. همچنین اندازه مجموعه نخه (K) در ابتدا برابر با کل تعداد اعضای جمعیت (N) در نظر گرفته شده و به صورت نزولی کاهش پیدا کرده است تا در مرحله پایانی فرایند محاسباتی الگوریتم به مقدار ۱ برسد. در الگوریتم GPSO، مقدار متغیر ω از ۰.۹ تا ۰.۴ به صورت نزولی مقداردهی شده است. همچنین ضرایب شتاب c_1 و c_2 برابر با ۲ در نظر گرفته شده است. متغیرهای الگوریتم JADEEP مشابه با (J. Zhang

¹ Fitness evaluations

² Bold face

(Sanderson, 2009) تعیین شده و پارامترهای انواع مختلف الگوریتم‌های جست‌وجوی گرانشی مشابه با (Rashedi et al., 2009) مقداردهی شده است.

جدول شماره ۱. نتایج بهینه‌سازی الگوریتم‌ها روی توابع CEC.

	<i>GPSO</i>	<i>JADEP</i>	<i>GSA</i>	<i>C-GSA</i>	<i>MO-GSA</i>
F1	5.29E+03	0.00E+00	0.00E+00	1.42E-12	0.00E+00
F2	8.34E+07	5.43E+03	1.88E+06	2.09E+06	2.74E+06
F3	9.81E+10	6.19E+06	1.27E+08	1.37E+08	1.48E+08
F4	1.58E+04	5.00E+03	1.75E+04	1.88E+04	6.68E+04
F5	1.59E+03	0.00E+00	5.74E-05	5.97E-05	4.32E-05
F6	4.88E+02	9.09E-01	4.97E+01	5.97E-05	3.83E-01
F7	1.63E+02	4.67E+00	2.17E+01	1.83E+01	3.15E+00
F8	2.12E+01	2.09E+01	2.04E+01	2.10E+01	2.02E+01
F9	4.43E+01	2.69E+01	4.14E+00	4.16E+00	2.66E+00
F10	1.45E+03	3.75 E-02	1.03E-02	6.62E-03	4.18E-03
F11	1.56E+02	0.00E+00	2.53E+01	2.60E+01	1.84E+01
F12	3.73E+02	2.06E+01	2.40E+01	2.30E+01	1.22E+01
F13	5.87E+02	4.16E+01	4.46E+01	4.70E+01	3.12E+01
F14	2.59E+03	4.39E-02	8.99E+02	8.82E+02	5.12E+02
F15	7.76E+03	3.20E+03	4.91E+02	4.89E+02	4.68E+02
F16	2.09E+00	1.75E+00	1.56E-02	1.49E-02	7.16E-04
F17	3.46E+02	3.04E+01	1.28E+01	1.30E+01	1.14E+01
F18	3.45E+02	7.31E+01	1.28E+01	1.35E+01	1.26E+01
F19	4.36E+04	1.43E+00	1.22E+00	1.14E+00	1.14E+00
F20	2.22E+01	1.01E+01	4.08E+00	4.70E+00	4.86E+00
F21	9.33E+02	2.98E+02	4.00E+02	4.00E+02	4.00E+02
F22	4.25E+03	1.93E+03	1.93E+03	1.98E+03	1.82E+03
F23	1.06E+04	3.25E+03	1.28E+03	1.33E+03	1.26E+03
F24	3.37E+02	2.10E+02	2.20E+02	2.24E+02	1.32E+02
F25	4.81E+02	2.63E+02	2.15E+02	2.15E+02	1.64E+02
F26	4.17E+02	2.09E+02	3.25E+02	3.82E+02	3.68E+02
F27	1.68E+03	5.34E+02	4.00E+02	4.00E+02	4.64E+02

F28	4.33E+03	3.00E+02	6.46E+02	6.38E+02	2.26E+02
-----	----------	----------	----------	----------	----------

همان‌طور که داده‌ها و اطلاعات موجود در جدول شماره ۱ نشان می‌دهد، الگوریتم جست‌وجوی گرانشی بهبود یافته، دارای عملکرد مناسب‌تری در مقایسه با سایر الگوریتم‌ها، خصوصاً در مسائل چندمدی ساده و ترکیبی است؛ بنابراین استفاده از این الگوریتم در مسائل پیچیده و کاربردی بهینه‌سازی، نظیر مسئله مسیریابی وسایل نقلیه با شرایط برداشت و تحویل هم‌زمان، موجب دست یافتن به جواب‌های بهتری خواهد شد. دلیل این بهبود در کارایی الگوریتم، کنترل قابلیت‌های کاوش و بهره‌وری الگوریتم در این روش است که در بخش دوم مقاله توضیح داده شد.

در ادامه، الگوریتم جست‌وجوی گرانشی بهبودیافته برای یافتن پاسخی در مسئله مسیریابی وسایل نقلیه با شرایط برداشت و تحویل هم‌زمان استفاده شده و نتایج شبیه‌سازی شده این روش، با سایر روش‌های ارائه شده برای حل این مسئله، مقایسه شده است. داده‌های مربوط به دل-آمیگو (Dell'Amico et al., 2006) که شامل پنج نمونه کلاس برای مسئله مسیریابی وسایل نقلیه با شرایط برداشت و تحویل هم‌زمان، اولین مجموعه داده‌های استاندارد است که در این مقاله استفاده شده است و روش پیشنهادی روی آن شبیه‌سازی شده است. هر کلاس در این مجموعه داده، شامل نمونه‌هایی با ۲۰ و ۴۰ کاربر است. کلاس ۱ شامل ۱۲ نمونه و کلاس‌های 2s, 2c, 3s و 3c شامل ۱۸ نمونه است. پارامترها و متغیرهای این مسئله به صورت زیر مقداردهی شده است. $f = 0$, $g = 1$, $D = \infty$ و تعداد وسایل نقلیه، برابر با تعداد وسایل نقلیه استفاده شده در بهترین پاسخ محاسبه شده است.

جدول شماره ۲، نتایج مربوطه بهترین جواب محاسبه شده در ۱۰ تکرار از الگوریتم‌های PSO (Ai & Kachitvichyanukul, 2009) و MO-GSA را با بهترین پاسخ محاسبه شده توسط دل-آمیگو مقایسه کرده است. در این جدول میانگین مربوط به هزینه همه نمونه‌ها در هر کلاس ارائه شده است. نتایج این جدول نشان می‌دهد که الگوریتم ارائه شده در این مقاله، کارایی بهتری نسبت به الگوریتم PSO برای حل مسئله مسیریابی وسایل نقلیه با شرایط برداشت و تحویل هم‌زمان، دارد و جواب‌هایی نزدیک به جواب‌های موجود برای دل آمیگو محاسبه کرده است.

جدول شماره ۲. مقایسه نتیجه الگوریتم‌ها روی داده‌های دل-آمیگو

Instance class	Average total cost		
	Dell' Amico et al. (Dell'Amico et al., 2006)	PSO(Ai & Kachitvichyanukul, 2009)	MO-GSA
Class 1	522.5	524.7	523.9
Class 2s	236822.7	236826.4	236828.2
Class 2c	341481.2	341994.3	34156.8
Class 3s	12082.3	11912.3	12210.7
Class 3c	15979.6	15984.6	15980.4

داده‌های دتلاف (Dethloff, 2001)، دومین مجموعه داده برای ارزیابی الگوریتم ارائه شده در این مقاله روی مسائل مسیریابی وسایل نقلیه با شرایط برداشت و تحویل هم‌زمان است. این مجموعه شامل ۴ مجموعه داده با نام‌های SCA3، SCA8، CON3 و CON8 است که هر مجموعه داده از ۱۰ نمونه با ۵۰ کاربر تشکیل شده است. در مجموعه داده‌های SCA همه کاربران به صورت تصادفی و یکنواخت در فضای تعریف شده، قرار گرفته‌اند. در حالی که، در مجموعه داده‌های CON، نیمی از کاربران به صورت تصادفی و یکنواخت در فضای تعریف شده برای مسئله قرار دارند و نیمه دیگر کاربران در نقاط خاصی از محیط مسئله جایگذاری

شده‌اند. پارامترهای موجود در این مسئله نیز مشابه با مقادیر تعریف شده در شبیه‌سازی موجود در مجموعه داده‌های دل-آمیکو، مقداردهی شده است.

در جدول شماره ۳، نتایج مربوط به پیاده‌سازی و اجرای الگوریتم‌های مختلف روی مجموعه داده‌های دتلاف قرار داده شده است. داده‌های موجود در این جدول مربوط به میانگین ۱۰ نمونه موجود در هر کلاس است. در این جدول، بهترین پاسخ مربوط به ۱۰ تکرار الگوریتم‌های PSO و MO-GSA، با نتایج به‌دست‌آمده از الگوریتم‌های دتلاف (Dethloff, 2001)، تنگ^۱ (Montané & Galvao, 2006) و بیانچسی^۲ (Bianchessi & Righini, 2007) مقایسه شده است.

جدول شماره ۳. مقایسه نتیجه الگوریتم‌ها روی داده‌های دتلاف

set	Average total cost				
	Dethloff (Dethloff, 2001)	Tang(Montané & Galvao, 2006)	Bianchessi (Bianchessi & Righini)	PSO (Ai & Kachitvichyanukul)	MO-GSA
SCA3	۷۴۶.۶	۶۷۴.۲	۶۸۴.۶	۶۷۵.۸	۶۷۲.۶
SCA8	۱۱۶۶.۴	۱۰۴۴.۴	۱۰۳۵.۷	۱۰۴۱.۸	۱۰۴۲.۲
CON3	۵۹۷.۳	۵۶۴.۲	۵۶۸.۵	۵۶۹.۶	۵۷۲.۴
CON8	۸۶۰.۶	۷۷۴.۳	۷۷۶.۴	۷۹۸.۳	۷۸۴.۶

۵. نتیجه‌گیری

نتایج موجود در بخش ارزیابی، نشان‌دهنده کارایی بالای روش ارائه شده در این مقاله، در مقایسه با سایر روش‌ها و الگوریتم‌های موجود برای مسئله مسیریابی و وسایل نقلیه با شرایط برداشت و تحویل هم‌زمان است. دلیل این بهبود کارایی و عمل‌کرد، استفاده از الگوریتم

¹ Tang

² Bianchessi

بهینه‌سازی جست‌وجوی گرانشی بهبود یافته است. در این الگوریتم با استفاده از تابع بولتزمن قابلیت‌های کاوش و بهره‌وری الگوریتم به‌خوبی کنترل شده است. در ابتدای روند محاسباتی برنامه، عوامل الگوریتم، به‌خوبی فضای شدنی مسئله را جست‌وجو می‌کنند و قدرت کاوش برنامه بسیار بالاست. این قابلیت باعث می‌شود تا الگوریتم در بهینه محلی همگرا نشود. هر چه از زمان محاسباتی مسئله می‌گذرد و برنامه به انتهای خود نزدیک می‌شود، قدرت بهره‌وری الگوریتم افزایش می‌یابد. بدین معنا که عوامل جست‌وجوی خود را پیرا هون بهترین جواب انجام می‌دهند و این باعث همگرایی بالای الگوریتم می‌شود؛ بنابراین استفاده از این الگوریتم بهینه‌سازی در مسائل کاربردی و پیچیده مانند مسیریابی و سایل نقلیه با شرایط برداشت و تحویل همزمان نتایج خوبی را تولید کند.

منابع

- Ai, T. J., & Kachitvichyanukul, V. (2009). A particle swarm optimization for the vehicle routing problem with simultaneous pickup and delivery. *Computers & Operations Research*, 36(5), 1693-1702.
- Bianchessi, N., & Righini, G. (2007). Heuristic algorithms for the vehicle routing problem with simultaneous pick-up and delivery. *Computers & Operations Research*, 34(2), 578-594.
- Christofides, N. (1976). The vehicle routing problem. *Revue française d'automatique d'informatique et de recherche opérationnelle. Recherche opérationnelle*, 10(1), 55-70.
- Das, P., Behera, H., Jena, P., & Panigrahi, B. (2016). Multi-robot path planning in a dynamic environment using improved gravitational search algorithm. *Journal of Electrical Systems and Information Technology*, 3(2), 295-313.
- Dell'Amico, M., Righini, G., & Salani, M. (2006). A branch-and-price approach to the vehicle routing problem with simultaneous distribution and collection. *Transportation science*, 40(2), 235-247.
- Dethloff, J. (2001). Vehicle routing and reverse logistics: the vehicle routing problem with simultaneous delivery and pick-up. *OR-Spektrum*, 23(1), 79-96.
- EBRAHIMI, M. S., RASHEDI, E., & JAVIDI, M. M. (2015). New functions for mass calculation in gravitational search algorithm.

- González, B., Melin, P., Valdez, F., & Prado-Arechiga, G. (2018). Ensemble neural network optimization using a gravitational search algorithm with interval type-1 and type-2 fuzzy parameter adaptation in pattern recognition applications. In *Fuzzy Logic Augmentation of Neural and Optimization Algorithms: Theoretical Aspects and Real Applications* (pp. 17-27): Springer.
- Gupta, A., & Saini, S. (2017). On solutions to vehicle routing problems using swarm optimization techniques: a review. In *Advances in Computer and Computational Sciences* (pp. 345-354): Springer.
- Hatamlou, A. (2013). Black hole: A new heuristic optimization approach for data clustering. *Information sciences*, 222, 175-184.
- Kato, A., Nakamoto, Y., Ishimori, T., & Togashi, K. (2016). Predictability of Posthepatectomy Liver Failure (PHLF) using 99mTc-GSA scintigraphy. *Journal of Nuclear Medicine*, 57(supplement 2), 644-644.
- Kherabadi, H. A., Mood, S. E., & Javidi, M. M. (2017). Mutation: a new operator in gravitational search algorithm using fuzzy controller. *Cybernetics and Information Technologies*, 17(1), 72-86.
- Mohanty, D. K. (2016). Gravitational search algorithm for economic optimization design of a shell and tube heat exchanger. *Applied Thermal Engineering*, 107, 184-193.
- Montané, F. A. T., & Galvao, R. D. (2006). A tabu search algorithm for the vehicle routing problem with simultaneous pick-up and delivery service. *Computers & Operations Research*, 33(3), 595-619.
- Noel, M. M. (2012). A new gradient based particle swarm optimization algorithm for accurate computation of global minimum. *Applied Soft Computing*, 12(1), 353-359.
- Rafsanjani, M. K., & Dowlatsahi, M. B. (2012). Using gravitational search algorithm for finding near-optimal base station location in two-tiered WSNs. *International Journal of Machine Learning and Computing*, 2(4), 377.
- Rashedi, E., Nezamabadi-Pour, H., & Saryazdi, S. (2009). GSA: a gravitational search algorithm. *Information sciences*, 179(13), 2232-2248.
- Rashedi, E., Nezamabadi-Pour, H., & Saryazdi, S. (2010). BGSA: binary gravitational search algorithm. *Natural Computing*, 9(3), 727-745.
- Sabri, N. M., Puteh, M., & Mahmood, M. R. (2013). A review of gravitational search algorithm. *Int. J. Advance. Soft Comput. Appl*, 5(3), 1-39.

- Shams, M., Rashedi, E., & Hakimi, A. (2015). Clustered-gravitational search algorithm and its application in parameter optimization of a low noise amplifier. *Applied Mathematics and Computation*, 258, 436-453.
- Soleimanpour-Moghadam, M., Nezamabadi-Pour, H., & Farsangi, M. M. (2014). A quantum inspired gravitational search algorithm for numerical function optimization. *Information sciences*, 267, 83-100.
- Tang, K., Yáo, X., Suganthan, P. N., MacNish, C., Chen, Y.-P., Chen, C.-M., & Yang, Z. (2007). Benchmark functions for the CEC'2008 special session and competition on large scale global optimization. *Nature Inspired Computation and Applications Laboratory, USTC, China*, 24.
- Toth, P., & Vigo, D. (2002). *The vehicle routing problem*: SIAM.
- Yao, B., Yu, B., Hu, P., Gao, J., & Zhang, M. (2016). An improved particle swarm optimization for carton heterogeneous vehicle routing problem with a collection depot. *Annals of Operations Research*, 242(2), 303-320.
- Zhang, J., & Sanderson, A. C. (2009). JADE: adaptive differential evolution with optional external archive. *IEEE Transactions on evolutionary computation*, 13(5), 945-958.
- Zhang, J. T., & Qiao, L. X. (2013). Optimization Mechanism Control Strategy of Vehicle Routing Problem Based on Improved PSO. Paper presented at the *Advanced Materials Research*.