مهندسی صنایع و مدیریت شریف، (تابستان ۱۴۰۲)
دوره‌ی ۳۹، شماره‌ی ۱، صص. ۷۳-۸۴، (پژوهشی)

Original Article

ارائه‌ی دو مدل ریاضی و چهار الگوریتم ابتکاری برای مسئله‌ی مسیریابی وسایل نقلیه با در نظرگرفتن مکان ـ زمان‌های پیشنهادی مشتریان

علی آقاداودی جلفائی (دانشجوی کارشناسی ارشد)
مهدی علینقیان* (دانشیار)
دانشکده‌ی مهندسی صنایع و سیستم‌ها، دانشگاه صنعتی اصفهان

مسیریابی وسایل نقلیه، مسئله‌یی است که تاکنون توسط پژوهشگران متعددی مطالعه شده و توسعه یافته است. در سال‌های اخیر با توسعه‌ی فروش‌های اینترنتی مسئله‌ی مسیریابی وسائط نقلیه با در نظرگرفتن مکان ـ زمان‌های پیشنهادی مشتریان، که یکی از زیرشاخه‌های مسئله‌ی مسیریابی عمومی وسائط نقلیه است مورد توجه محققین قرارگرفته است. در این مقاله دو مدل ریاضی مبتنی برگره و مبتنی بر جریان برای مسئله ارائه شده است. نتایج حل مدل نشان می‌دهد که مدل ریاضی مبتنی بر جریان کارایی بالاتری نسبت به مدل مبتنی برگره دارد. در ادامه چهار الگوریتم ابتکاری شامل الگوریتم مبتنی بر صرفه‌جویی سری و موازی، الگوریتم مبتنی بر درج کردن و الگوریتم مبتنی بر نزدیک‌ترین مشتری بازدید نشده برای مسئله‌ی طراحی شده است. الگوریتم مبتنی بر درج کردن، در نمونه‌های کوچک نسبت به جواب بهینه، شش درصد خطا داشته است. در نمونه‌های بزرگ نیز، عملکرد مناسبی در مقایسه با سایر الگوریتم‌ها داشته است.

واژگان کلیدی: مسیریابی وسایل نقلیه، مسیریابی انتخابی وسایل نقلیه، مکان ـ زمان‌های پیشنهادی مشتریان، پنجره‌ی زمانی، الگوریتم ابتکاری.

a.aghadavoudi.jolfaei@in.iut.ac.ir
alinaghian@iut.ac.ir

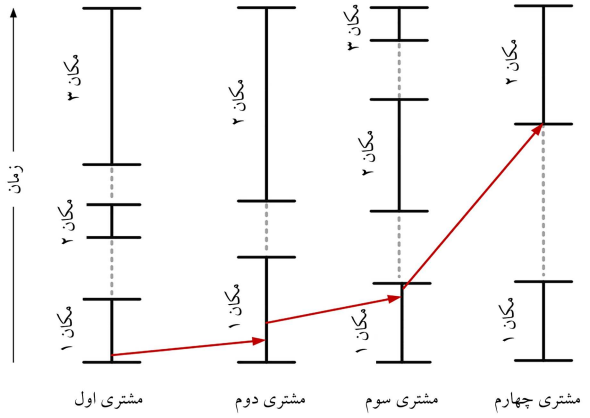
۱. مقدمه

«مسئله‌ی مسیریابی وسایل نقلیه» (VRP)،[۱] یکی از مسائل کاربردی و چالش برانگیز تحقیق در عملیات است. از زمان طرح این مسئله توسط دانتزیگ و رامسر[۱] پژوهشگران زیادی این مسئله را بررسی کردند. چالش مورد بررسی در این مسئله، نحوه‌ی تخصیص مشتریان به وسایل نقلیه و همچنین تعیین توالی بهینه، برای بازدید مشتریان است به گونه‌یی که تابع هدف مسئله بهینه شود.[۲] به طورکلی در مسئله‌ی مسیریابی وسایل نقلیه، الزامی بودن بازدید از تمام رئوس گراف، به عنوان یک فرض کلاسیک در نظرگرفته می‌شود. با این حال، در طیف گسترده‌ی VRP گونه‌ی جدیدی مطرح شده است که این فرض کلاسیک، به نحوی دیگر در نظرگرفته می‌شود. این مسئله تحت عنوان، مسیریابی عمومی وسایل نقلیه، در ادبیات موضوع مطرح شده است.[۳] در مسئله‌ی مسیریابی عمومی وسایل نقلیه، رئوس گراف ـ بنا به ماهیت یا شرایط ـ در دسته‌های مختلفی، خوشه‌بندی می‌شوند. بازدید هر مشتری از هر خوشه، الزام بازدید از سایر مشتریان در آن خوشه را برطرف می‌کند؛ به عبارت دیگر محدودیت بازدید مشتریان هر دسته، با بازدید فقط یکی از آن‌ها ارضا می‌شود.[۴]

همان‌طورکه مطرح شد، مسئله‌ی مسیریابی وسایل نقلیه، یک مسئله‌ی کاربردی در زمینه‌ی حمل‌ونقل کالاهاست. توزیع کارآمد کالا، یکی از چالش‌های اساسی برای شرکت‌ها و صنایع مختلف است که امروزه، تبدیل به یک رقابت اساسی شده است. کسب رضایت مشتریان درکنار مدیریت هزینه‌های توزیع کالا، عوامل مهم و اساسی در تعیین برندگان این رقابت، به حساب می‌آیند. درکنار شرایط رقابتی شرکت‌ها و سازمان‌های مختلف، در نظرگرفتن شرایط مشتریان نیز حائز اهمیت است. تغییر سبک زندگی در جوامع امروزی، منجر به افزایش فعالیت‌های اجتماعی افراد در محیط‌های مختلف وحضور در مکان‌های متعدد در طول روز شده است. از طرفی گسترش روزافزون شبکه‌های شهری، باعث پیچیده شدن فرایند حمل‌ونقل می‌شود. لذا با توجه به این دو مورد، با در نظرگرفتن مکان‌های مختلف افراد در طول روز، می‌توان گزینه‌های مکانی بیشتری را برای تحویل کالا در نظرگرفت. این مورد به معنای گسترش فضای حل است که با احتمال زیاد منجر به یافتن

* نویسنده مسئول
تاریخ: دریافت ۱۴۰۰/۸/۲۴، اصلاحیه ۱۴۰۱/۳/۲۱، پذیرش ۱۴۰۱/۵/۱۲.
DOI:10.24200/J65.2022.7185.2221

استناد به این مقاله:
آقاداودی جلفائی، علی و علینقیان، مهدی، ۱۴۰۲. ارائه‌ی دو مدل ریاضی و چهار الگوریتم ابتکاری برای مسئله‌ی مسیریابی وسایل نقلیه با در نظرگرفتن مکان ـ زمان‌های پیشنهادی مشتریان. مهندسی صنایع و مدیریت شریف، ۳۹(۱)، صص. ۷۳-۸۴.



شکل ۱. شمای کلی مسئله‌ی مسیریابی عمومی وسایل نقلیه همراه با پنجره‌ی زمانی.

جواب‌هایی بهتر خواهد شد. همان‌طورکه عنوان شد، در مسئله‌ی مسیریابی فراگیر وسایل نقلیه (GVRP)[۲] رئوس گراف در دسته‌های مختلف خوشه‌بندی می‌شوند. یکی از رویکردهای خوشه‌بندی رئوس، بر اساس کاربرد توزیع کالاها در زنجیره‌ی توزیع، مطرح شده است.[۵] در این کاربرد، مشتریان برای تحویل گرفتن کالای خود، می‌توانند چندین گزینه معرفی کنند. مکان این گزینه‌ها با توجه به زمان، متغیر در نظر گرفته می‌شود؛ به عنوان مثال یک مشتری در ساعات ابتدایی روز در محل کار، در ساعات میانی در فروشگاه و در ساعات پایانی در منزل حضور دارد. طبق این تعریف هر مشتری در هر بازه زمانی در یک مکان حضور دارد. در این مقاله از اصطلاح مکان ـ زمان برای گزینه‌های تحویل کالای مشتریان، استفاده می‌شود. در مسئله‌ی مطرح شده برای تحویل کالا به مشتری، علاوه بر پاسخگویی به سؤالات مسئله‌ی مسیریابی کلاسیک (مشتریان با چه ترتیبی و توسط کدام وسیله سرویس دهی شوند)، باید به دو سؤال اساسی دیگر به صورت همزمان پاسخ داد. اولاً مشخص کرد هر مشتری در چه زمانی باید بازدید شود و ثانیاً این بازدید در چه مکانی صورت گیرد. نکته قابل توجه، در نظرگرفتن پنجره‌ی زمانی سخت برای تحویل کالا به مشتریان است. این تحویل صرفاً در مکان‌های تعیین شده از طرف مشتریان صورت می‌گیرد. در شکل ۱، شمایی کلی از مواردی که عنوان شد، به صورت شماتیک نمایش داده شده است.

در این مقاله دو مدل ریاضی مبتنی برگره و مبتنی بر جریان برای مسئله‌ی مطرح، ارائه شده است. نتایج حل در ابعاد کوچک نشان‌دهنده کارایی بیشتر مدل مبتنی بر جریان نسبت به مدل مبتنی برگره است. در ادامه چهار دسته الگوریتم شامل الگوریتم مبتنی بر صرفه‌جویی سری و موازی، الگوریتم مبتنی بر درج کردن و الگوریتم مبتنی بر نزدیک‌ترین مشتری بازدید نشده برای مسئله‌ی طراحی و توسعه داده شده است.

۲. مروری بر پیشینه‌ی موضوع

مسئله‌ی مسیریابی فراگیر وسایل نقلیه (GVRP)، برای اولین بار توسط قیانی و ایمپرتا[۳] مطرح شد. در ادامه، پژوهشگران متعددی به بررسی GVRP پرداختند. از آن‌جا که ارائه‌ی یک مدل کارا برای مسائل اهمیت زیادی دارد، پژوهشگران زیادی سعی در ارائه و مقایسه‌ی مدل‌های متنوع داشته‌اند GVRP نیز از این قاعده مستثنی نبوده و پژوهشگران همواره سعی در ارائه‌ی مدل‌هایی کاراتر داشته‌اند. کارا و بکتاس،[۶] یک فرمول‌بندی چندجمله‌یی برای GVRP پیشنهاد کردند. پاپ و همکاران[۷] نیز یک مدل عدد صحیح برای مسئله‌ی مسیریابی عمومی وسایل نقلیه ارائه کردند. بکتاس و همکاران[۸] چهار مدل جدید با رویکردهای متفاوت ارائه و آنها را مقایسه کردند. نتایج حاصل، برتری رویکرد چهارم آنها راکه به‌صورت یک گراف بدون جهت و جریان محور ارائه شده بود، نشان می‌داد. پاپ و پاپ سیتار[۹] در پژوهش خود با ارائه‌ی یک مدل عدد صحیح برای GVRP روشی برای تبدیل آن به مسئله‌ی مسیریابی کلاسیک مطرح کردند. در ادامه‌ی مدل‌های ارائه شده، پاپ و همکاران[۴] دو مدل خطی عدد صحیح با دو رویکرد متفاوت ارائه دادند. مدل اول، با رویکرد مبتنی برگره ارائه شده و تا حدودی مشابه مدل ارائه شده توسط کارا و بکتاس[۶] است، اگرچه در ارائه‌ی حد پایین، قوی‌تر عمل کرده است. مدل دوم با رویکرد مبتنی بر جریان ارائه شده است. در تمام پژوهش‌های فوق، GVRP کلاسیک، بدون توجه به محدودیت‌های جانبی بررسی شده است.

مسئله‌ی مسیریابی فراگیر وسایل نقلیه (GVRP) در انواع حالت‌ها و با در نظرگرفتن محدودیت‌های مختلف بررسی شده است. از جمله این موارد، می‌توان به بررسی شکست تقاضا در GVRP،[۱۰] مسئله‌ی برداشت و تحویل،[۱۱] بررسی مسئله در حالت چندسطحی،[۱۲] بررسی مسئله با تقاضای احتمالی،[۱۳] در نظر گرفتن زمان‌های احتمالی سفر[۱۴] و بررسی مسئله با تعداد ناوگان منعطف[۱۵] اشاره کرد.

یکی از مهم‌ترین محدودیت‌های دنیای واقعی، پنجره‌ی زمانی است. در طیف وسیعی از مسائل GVRP پنجره‌ی زمانی یکی از ارکان اصلی مسئله به شمار می‌رود. پنجره‌ی زمانی برای اولین بار توسط موشیا و همکاران در خصوص GVRP در نظر گرفته شد[۱۶] این پژوهشگران با در نظرگرفتن پنجره‌ی زمانی در GVRP، برنامه‌ریزی سرویس‌های پرسنل را مورد بررسی قرار دادند. با مطرح شدن پنجره‌ی زمانی در مسائل GVRP، ایده‌ی جدیدی در این حوزه شکل گرفت و پژوهشگران زیادی با در نظرگرفتن پنجره‌ی زمانی، گستره‌ی مسائل GVRP را توسعه دادند. این ایده بدین صورت بود که گزینه‌های تحویل کالا به مشتریان، نسبت به زمان متغیر در نظر گرفته می‌شد. ایده‌ی «متغیر بودن مکان نسبت به زمان»، برای اولین بار توسط رِیِس و همکاران[۵] معرفی شد. در این مسئله ماشین مشتری در هر بازه زمانی در یکی از مکان‌های برنامه‌ی سفر او پارک شده و کالای مشتری باید در یکی از این مکان‌ها، به وی تحویل داده شود. مکان‌های پارک ماشین را می‌توان یک خوشه در نظر گرفت که از هر خوشه با توجه به پنجره‌ی زمانی باید یکی از مکان‌های پارک انتخاب شود. با مطرح شدن این ایده، مسئله‌ی جدیدی در خانواده‌ی مسائل VRP تحت عنوان «مسیریابی وسایل نقلیه با در نظرگرفتن نقاط متغیری برای تحویل (VRPRDL)»،[۳] شکل گرفت و منشاء تحقیقات متعددی در این حوزه شد. هی و همکاران[۱۴] مسئله‌ی VRPRDL را در شبکه‌ی توزیع، با زمان‌های احتمالی سفر بررسی کردند. وضعیت بدی آب و هوا، ترافیک در جاده‌ها و غیره، منجر به زمان سفر نامشخص در سیستم توزیع کالا می‌شود. این مسئله به صورت یک مدل احتمالی دومرحله‌یی با هدف کمینه‌سازی هزینه‌ی سفر فرمول‌بندی شده است. اُزبایگین و همکاران[۱۷] مسئله‌ی VRPRDL را به صورت پویا در نظر گرفتند. این پژوهشگران با در نظرگرفتن تغییر و اختلال برنامه‌ی سفر مشتریان در طول افق برنامه‌ریزی، مسئله را با کمینه‌سازی مسافت حل کردند. اُرنستین و همکاران[۱۸] مسئله‌یی طراحی کردند که در آن هر مشتری برای تحویل گرفتن کالای خود، تعدادی نقطه‌ی خدمت[۴] مشخص می‌کند. نقاط خدمت می‌تواند نزدیک به منزل، نزدیک به محل کار، فروشگاه‌های خرید و غیره باشند. پژوهشگران یک مدل برنامه‌نویسی خطی عدد صحیح مختلط (MILP)[۵] با تابع هدف کمینه‌سازی هزینه‌های سفر برای مسئله فرموله کردند. دومز و همکاران[۱۹] ترجیحات مشتریان را برای تحویل کالا

در مکان‌های مختلف در پنجره‌های زمانی مختلف، در نظر گرفتند. در این مسئله مشتریان برای هرگزینه، ارجحیت مشخص می‌کردند. آنها یک مدل ریاضی MIP برای مسئله طراحی کردند. تابع هدف در نظر گرفته شده در این مقاله، به صورت کمینه‌سازی هزینه‌های پیمایش در نظر گرفته شده است.

تابع هدف در نظر گرفته شده در پژوهش‌های مذکور از جنس مسافت است. در حالی که به نظر میرسد این تابع هدف با در نظر گرفتن پنجره‌ی زمانی نتواند پاسخگوی شرایط دنیای واقعی باشد؛ چراکه ممکن است یک مسیر علیرغم کوتاه بودن از لحاظ مسافت، به دلیل وجود پنجره‌های زمانی با فواصل بسیار زیاد، از لحاظ زمانی مدت زمان بسیار زیادی داشته باشد. علاوه بر این، تابع هدف «مدت زمان پیمایش مسیرها» عمومی‌تر بوده و در مقایسه با تابع هدف مسافت کاربردی‌تر و در عین حال، محاسبه‌ی آن در الگوریتم‌ها چالش‌های بیشتری دارد؛ چرا که اگر مسیرها با انتظارهای بیش از حد مواجه باشند، می‌توان از طریق اصلاح زمان شروع به حرکت وسایل، از به وجود آمدن انتظارهای بیش از حد جلوگیری کرد؛ در صورتی که تابع هدف مسافت توجهی به این زمان‌ها نمی‌کند و صرفاً به رعایت پنجره‌های زمانی اکتفا می‌کند.

در این مقاله دو مدل ریاضی مبتنی بر گره و مبتنی بر جریان برای مسئله‌ی مطرح، ارائه شده است. مدل مبتنی بر گره با اقتباس از ادبیات موضوع و با تغییر تابع هدف از حالت «مسافت» به «مدت زمان پیمایش مسیرها»، توسعه داده شده است. در حالت دوم، مسئله با شیوه‌یی بدیع و با رویکرد مبتنی بر جریان، مدل‌سازی شده است. در مدل‌های توسعه داده شده، محدودیت زمان مسیرها نیز لحاظ شده است. نتایج حل در ابعاد کوچک نشان‌دهنده‌ی کارایی بیشتر مدل مبتنی بر جریان نسبت به مدل مبتنی برگره است. در ادامه چهار دسته الگوریتم، شامل الگوریتم مبتنی بر صرفه‌جویی سری و موازی، الگوریتم مبتنی بر درج کردن و الگوریتم مبتنی بر نزدیک‌ترین مشتری بازدید نشده برای مسئله طراحی شده است. همچنین در تمام مسائل این حوزه تابع هدف در نظر گرفته شده به صورت کمینه‌سازی مسافت یا هزینه‌های پیمایش یال عنوان شده است. در حالی که در این مقاله با نوآوری در نظر گرفته شده، تابع هدف جامع‌تر و به صورت کمینه‌سازی مدت زمان سفرها در نظر گرفته می‌شود. در این حالت علاوه بر کمینه‌سازی مسافت پیموده شده، هزینه‌های انتظار وسایل نیز مستقیماً در بهینه‌سازی لحاظ می‌شوند.

۳. تعریف مسئله

با توجه به موارد مطرح شده، مسئله‌ی مسیریابی عمومی با در نظر گرفتن پنجره زمان (GVRPTW)،[۶] بدین صورت تعریف می‌شود:گراف G به صورت $G = (N, A)$ مفروض است. در این گراف N، معرف رئوس گراف و A مشخص‌کننده یال‌هال گراف است. مجموعه‌ی N متشکل از دپو (اولین گره) و گره‌های مرتبط با مشتریان است. مجموعه‌ی A نیز همان‌طور که ذکر شد، مشخص‌کننده‌ی یال‌های ارتباطی میان رئوس گراف است. این مجموعه به صورت $\mathrm{A} = \{(\mathrm{i},\mathrm{j}) : \mathrm{i},\mathrm{j} \in \mathrm{N} | \mathrm{i} \neq \mathrm{j}\}$ تعریف می‌شود. هر یال $(i, j) \in A$ در زمان t_{ij} پیمایش می‌شود. سرویس‌دهی به مشتریان، شامل تحویل دادن/گرفتن کالا به/از مشتریان است. در چنین حالتی، هر مشتری به میزان مشخصی تقاضای قطعی دارد و هر وسیله ظرفیت مشخصی دارد. تعداد وسایل حمل بار مشتریان، ثابت و همگن در نظر گرفته می‌شود. وسایل نقلیه از دپو شروع به حرکت کرده و پس از سرویس‌دهی، به دپو باز می‌گردند. در این مسئله، هر مشتری ممکن است مکان‌های متفاوتی را برای سرویس گرفتن پیشنهاد کند. به عبارت دیگر، برای مشتری $c \in C$ در مکان ـ زمان‌های N_c امکان سرویس‌دهی فراهم است. $N_C \subset N$ در نظر گرفته می‌شود. هر نقطه‌ی i در گراف به اندازه d_i تقاضای قطعی دارد. این تقاضا در تمام مکان ـ زمان‌های معرفی شده از جانب مشتری c، قابل تحویل است. بازدید هر یک از مکان ـ زمان‌های مشتری، الزام بازدید را از سایر مکان ـ زمان‌های مشتری، برطرف می‌کند. مکان ـ زمان‌های مشتریان، با توجه به پنجره‌های زمانی مختلف در نظر گرفته می‌شود. با توجه به این که مکان مشتریان نسبت به زمان تغییر می‌کند، بنابراین امکان سرویس‌دهی خارج از پنجره‌های زمانی، فراهم نیست؛ پس پنجره‌های زمانی به صورت پنجره‌ی زمانی سخت در مسئله لحاظ می‌شود. هدف مسئله، سرویس‌دهی به مشتریان، در کم‌ترین مدت زمان ممکن است. این تابع هدف، علاوه بر در نظر گرفتن مدت زمان طی کردن مسافت بین مشتریان، پنجره‌های زمانی مشتریان را نیز مد نظر قرار داده و زمان‌های انتظار برای باز شدن پنجره‌های زمانی را نیز در بر می‌گیرد؛ بنابراین در نظر گرفتن مدت زمان سرویس‌دهی به مشتریان، حالت جامع‌تری برای مسائل پنجره‌ی زمانی است. در ادامه مسئله‌ی مطرح شده، با دو رویکرد فرمول‌بندی می‌شود. در رویکرد اول، یک مدل سه اندیسه برحسب گره ارائه شده و در رویکرد دوم یک مدل دو اندیسه ارائه می‌شود.

۱.۳. فرمول‌بندی مسئله برحسب گره

مدل اول $(P1)$ به صورت یک مدل سه اندیسه و با رویکرد مدل‌سازی برحسب گره است. در ادامه اجزای مختلف این مدل تشریح شده است.

نمادها

C: مجموعه‌ی مشتریان که با اندیس‌های c و c' مشخص می‌شود؛
N: مجموعه‌ی کل نقاط که با اندیس‌های i و j مشخص می‌شود. در این مجموعه $i = ۱$ نشان‌دهنده‌ی دپو است؛
N_c: مجموعه‌ی مکان ـ زمان‌های مشتری c، $N_c \subset N$؛
K: مجموعه‌ی وسایل نقلیه که با اندیس k نمایش داده می‌شود.

- **پارامترها**

t_{ij}: مدت زمان سفر بین دو نقطه‌ی i و j؛
d_i: تقاضای نقطه‌ی i؛
Q: ظرفیت هرکدام از وسایل نقلیه؛
a_i: کران پایین پنجره‌ی زمانی بازدید از نقطه‌ی i؛
b_i: کران بالای پنجره‌ی زمانی بازدید از نقطه‌ی i؛
St_i: مدت زمان سرویس‌دهی به نقطه‌ی i؛
T: بیشینه‌ی زمان در دسترس برای وسایل نقلیه؛
M: یک عدد بزرگ.

- **متغیرهای تصمیم**

x_{ij}^k: برابر ۱ است اگر مسیر بین دو گره i و j توسط وسیله‌ی k طی شود و در غیر این صورت برابر صفر خواهد بود؛
Δ_i^k: زمان رسیدن وسیله‌ی k به نقطه‌ی i؛
w_i: مدت زمان انتظار در نقطه‌ی i برای شروع سرویس‌دهی.

با توجه به موارد ذکر شده، مدل ریاضی مسئله در روابط ۱ تا ۱۲ مشخص شده است.

$$P1: Minimize \quad Z = \sum_{k\in K}\sum_{j\in N}\sum_{i\in N} t_{ij}x_{ij}^{k} + \sum_{i\in N} w_i \qquad (1)$$

$S.t.$

$$\sum_{i\in N} x_{ij}^{k} = \sum_{i\in N} x_{ji}^{k} \qquad \forall j \in N\ ,\ k \in K \qquad (2)$$

$$\sum_{k\in K}\sum_{i\in N}\sum_{j\in N_c} x_{ij}^{k} = 1 \qquad \forall c \in C \qquad (3)$$

$$\sum_{k\in K}\sum_{j\in N} x_{ij}^{k} \le |K| \qquad \forall i = 1 \qquad (4)$$

$$\sum_{i\in N}\sum_{j\in N} d_i x_{ij}^{k} \le Q \qquad \forall k \in K \qquad (5)$$

$$\Delta_i^k + w_i + st_i + t_{ij} \le \Delta_j^k + M(1 - \sum_{k\in K} x_{ij}^{k})$$

$$\forall i \in N, j \in N/1\ ,\ k \in K \qquad (6)$$

$$\Delta_i^k + w_i + st_i + t_{ij} \ge \Delta_j^k - M(1 - \sum_{k\in K} x_{ij}^{k})$$

$$\forall i \in N, j \in N/1\ ,\ k \in K \qquad (7)$$

$$\Delta_i + w_i + st_i + t_{ij} \le T + M(1 - \sum_{k\in K} x_{ij}^{k})$$

$$\forall i \in N,\ \ j = 1 \qquad (8)$$

$$\Delta_i^k \le b_i \qquad \forall i \in N/1\ ,\ k \in K \qquad (9)$$

$$a_i \le \Delta_i^k + w_i \qquad \forall i \in N/1\ ,\ k \in K \qquad (10)$$

$$x_{ij}^{k} \in \{0, 1\} \qquad \forall i, j \in N\ ,\ k \in K \qquad (11)$$

$$w_i,\ \ \Delta_i \in \mathbb{R}^{+} \qquad \forall i \in N,\ k \in K \qquad (12)$$

معادله‌ی ۱ به عنوان تابع هدف مدل، مدت زمان کل مسیرها را کمینه می‌کند. این معادله از دو قسمت تشکیل شده است. قسمت اول کمینه‌سازی مدت زمان پیمایش مسیرها و قسمت دوم کمینه‌سازی مدت زمان انتظار وسایل نقلیه است. رابطه‌ی ۲ بیان می‌کند اگر وسیله به یک مشتری وارد شد، حتماً از آن خارج شود. رابطه‌ی ۳ بیان می‌کند که هر مشتری حتماً در یکی از مکان ـ زمان‌های مربوط بازدید شود. رابطه‌ی ۴ بیان‌گر محدودیت تعداد وسایل است. رابطه‌ی ۵ محدودیت ظرفیت وسایل نقلیه را درنظر می‌گیرد. روابط ۶ و ۷ زمان رسیدن به مشتریان متوالی را تنظیم می‌کند. رابطه‌ی ۸ بیشترین زمان مجاز برای وسایل نقلیه را در نظر می‌گیرد. روابط ۹ و ۱۰ تضمین می‌کند که هر مشتری تنها در پنجره‌ی زمانی مجاز خود بازدید شود. در نهایت روابط ۱۱ و ۱۲ وضعیت متغیرهای تصمیم مسئله را مشخص می‌کند.

۲.۳. فرمول‌بندی مسئله برحسب جریان

در قسمت قبل یک فرمول‌بندی بر اساس گره برای مسئله ارائه شد. در این قسمت با معرفی متغیرهای تصمیم جدید، یک فرمول‌بندی دو اندیسه برای مسئله ارائه می‌شود. پارامترها و مجموعه‌های مسئله‌ی مشابه حالت قبل است و تنها متغیرهای تصمیم جدیدی برای مسئله در نظر گرفته می‌شود.

- **متغیرهای تصمیم**

x_{ij}^{k}: اگر گره‌ی j بعد از گره‌ی i بازدید شود، برابر ۱ خواهد بود و در غیر این صورت مقدار صفر می‌گیرد؛

y_c: میزان بار باقیمانده در وسیله‌ی نقلیه، بعد از سرویس‌دهی به مشتری c؛

Δ_i: زمان رسیدن به نقطه‌ی i؛

θ_i: زمان خارج شدن از دپو و حرکت به سمت مشتری i.

بر این اساس، مدل ریاضی دوم ($P2$) در روابط ۱۳ تا ۲۶ مشخص شده است.

$$P2: Minimize \quad Z = \sum_{j\in N}\sum_{i\in N} t_{ij}x_{ij} + \sum_{i\in N} w_i \qquad (13)$$

$S.t.$

$$\sum_{i\in N} x_{ij} = \sum_{i\in N} x_{ji} \qquad \forall j \in N \qquad (14)$$

$$\sum_{j\in N_c}\sum_{i\in N} x_{ij} = 1 \qquad \forall c \in C \qquad (15)$$

$$\sum_{j\in N} x_{ij} \le |K| \qquad \forall\ \ i = 1 \qquad (16)$$

$$y_c \le Q - d_c \qquad \forall\ c \in C \qquad (17)$$

$$y_{c'} + Q\left(1 - \sum_{i\in N_c}\sum_{j\in N_{c'}} x_{ij}\right) \ge d_{c'} + y_{c'}$$

$$\forall\ c \in C \cup \{1\}\ \ , c' \in C\backslash c \qquad (18)$$

$$\theta_i + \Delta_i + w_i + st_i + t_{ij} \le \Delta_j + M(1 - x_{ij})$$

$$\forall i \in N\ \ j \in N/1 \qquad (19)$$

$$\theta_i + \Delta_i + w_i + st_i + t_{ij} \ge \Delta_j - M(1 - x_{ij})$$

$$\forall i \in N\ \ j \in N/1 \qquad (20)$$

$$\theta_i \le T x_{ji} \qquad \forall i \in N\ ,\ j = 1 \qquad (21)$$

$$\Delta_i + w_i + st_i + t_{ij} \le T + M(1 - x_{ij}) \qquad \forall i \in N,\ \ j = 1 \qquad (22)$$

$$\theta_i + \Delta_i \le b_i \qquad \forall i \in N/1 \qquad (23)$$

$$a_i \le\ \theta_i + \Delta_i + w_i \qquad \forall i \in N/1 \qquad (24)$$

$$x_{ij} \in \{0, 1\} \qquad \forall i, j \in N \qquad (25)$$

$$w_i,\ \ \Delta_i\ ,\ \theta_i \in \mathbb{R}^{+} \qquad \forall i \in N \qquad (26)$$

در مدل $P2$ روابط ۱۳ تا ۱۶ از لحاظ عملکردی مشابه با روابط ۱ تا ۴ در مدل $P1$ هستند. روابط ۱۷ و ۱۸ تضمین‌کننده‌ی محدودیت ظرفیت وسایل نقلیه‌اند. روابط ۱۹ و ۲۰ به منظور رعایت شرایط پنجره‌ی زمانی در نظر گرفته شده‌اند. رابطه‌ی ۲۱ تضمین می‌کند که متغیر مربوط به زمان خروج از دپو تنها برای نقطه‌ی اول تور مقدار بگیرد. رابطه‌ی ۲۲ تضمین‌کننده‌ی بیشترین زمان مجاز سفر است. روابط ۲۳ و ۲۴ نیز به‌عنوان حدود بالا و پایین پنجره‌ی زمانی برای نقطه لحاظ شده است. در نهایت روابط ۲۵ و ۲۶ مشخص‌کننده‌ی متغیرهای تصمیم مسئله هستند.

۴. رویکردهای حل مسئله

از آن‌جا که مسئله‌ی VRP یک مسئله‌ی NP-Hard است،[۲۰] بنابراین مسئله‌ی GVRPTW نیز یک مسئله‌ی NP-Hard خواهد بود؛ چرا که اگر در هر خوشه، تنها یک رأس وجود داشته باشد و پنجره‌های زمانیِ تمام رئوس مشابه فرض شود، مسئله‌ی GVRPTW به مسئله‌ی کلاسیک VRP تبدیل می‌شود. در چنین حالتی، توسعه‌ی الگوریتم‌های ابتکاری و فرااِبتکاری برای یافتن جواب مناسب برای مسئله‌ی GVRPTW، حائز اهمیت است. در این نوشتار برای حل مسئله‌ی GVRPTW چهار الگوریتم ابتکاری توسعه داده شده است. سولومن در سال ۱۹۸۷ با معرفی مفهوم «پنجره‌ی زمانی سخت»، چند الگوریتم ابتکاری برای حل مسئله‌ی مسیریابی، همراه با پنجره‌ی زمانی ارائه کرد. الگوریتم‌های موجود در این مقاله، با مبنا قرار دادن الگوریتم‌های ارائه شده توسط سولومن[۲۱] توسعه داده شده است. در این الگوریتم‌ها ایده‌ی مکان ـ زمان اضافه شده است. همچنین در جهت تکامل الگوریتم‌ها در بعضی از قسمت‌ها راهکارهای متعددی برای تنوع‌بخشی به عملکرد الگوریتم معرفی شده و به کار رفته است. در ادامه الگوریتم‌های ابتکاری طراحی شده برای حل مسئله‌ی مسیریابی انتخابی همراه با پنجره‌ی زمانی، تشریح می‌شود.

۱.۴. الگوریتم‌های ابتکاری با محوریت صرفه‌جویی

اولین رویکرد به کار رفته در این مقاله، استفاده از رویکرد صرفه‌جویی است. ایده‌ی الگوریتم صرفه‌جویی، برای اولین بار توسط کلارک و رایت[۲۲] مطرح شد. در این الگوریتم، فرض می‌شود که در ابتدا هر مشتری توسط یک وسیله‌ی نقلیه سرویس‌دهی می‌شود. هدف، ادغام مسیرها به نحوی است که بیشترین صرفه‌جویی حاصل شود. این الگوریتم در دو نسخه‌ی سری و موازی ارائه شده است. هریک از این الگوریتم‌ها در ادامه تشریح می‌شوند.

۱.۱.۴. الگوریتم صرفه‌جویی سری بر مبنای زمان

چنان که پیش‌تر نیز عنوان شد، در شروع الگوریتم باید اعداد مربوط به صرفه‌جویی‌ها محاسبه شود. بدین منظور، ابتدا فرض می‌شود مشتریان به صورت جداگانه سرویس‌دهی می‌شوند. در مرحله‌ی بعد، پس از ادغام مشتریان، صرفه‌جویی‌های حاصل از ادغام مسیرها محاسبه می‌شود. ابتدا فرض می‌شود که دو نقطه‌ی i و j به صورت جداگانه سرویس‌دهی شده‌اند. بنابراین زمان رفت و برگشت به نقطه‌ی i و زمان رفت و برگشت به نقطه‌ی j، در نظرگرفته می‌شود. در صورت ادغام مسیرها، هزینه‌ی رفتن به یکی از نقاط از دپو و هزینه‌ی بازگشت از یکی از نقاط به دپو از مجموع هزینه‌ها کم می‌شود، اما هزینه‌ی اتصال نقاط به مجموع هزینه‌ها اضافه می‌شود. اختلاف این دو هزینه، میزان صرفه‌جویی حاصل از ادغام دو نقطه‌ی i و j را مشخص می‌کند. رابطه‌ی ۲۷ میزان صرفه‌جویی حاصل از ادغام دو مشتری i و j را مشخص می‌کند.

$$Sav_{ij} = t_{io} + t_{oj} - t_{ij} \qquad (۲۷)$$

گام‌های الگوریتم صرفه‌جویی سری برمبنای زمان عبارت است از:

گام ۱. برای نقاط به صورت دو به دو، میزان صرفه‌جویی را طبق رابطه‌ی ۲۷ محاسبه کنید؛

گام ۲. بیشترین میزان صرفه‌جویی را انتخاب کنید؛

گام ۳. دو نقطه‌ی مربوط به صرفه‌جویی انتخاب شده را با توالی دلخواه در یک مسیر قرار داده و شرط شدنی بودن زمان را بررسی کنید؛

گام ۴. اگر شرط شدنی بودن زمان برقرار شد، مسیر مربوطه را برای وسیله در نظر بگیرید و مدت زمان مسیر و ظرفیت وسیله را به روزکنید و به گام ۶ بروید. در غیر این صورت جای دو نقطه را تعویض کنید و شرط شدنی بودن زمان را مجدداً بررسی کنید؛

گام ۵. در صورتی که شرط شدنی بودن زمان برقرار شد، آن را برای وسیله در نظر بگیرید. مدت زمان مسیر و ظرفیت وسیله را به روزکنید و به گام ۶ بروید. در غیر این صورت، بیشترین صرفه‌جویی بعدی را در نظر بگیرید و به گام ۳ بازگردید؛

گام ۶. اگر وسیله‌ی نقلیه ظرفیت دارد، یا در ماتریس صرفه‌جویی در سطر مربوط به نقطه‌ی آخر تور یا در ستون مربوط به نقطه‌ی ابتدای تور عدد مثبتی وجود دارد، به گام بعد بروید. در غیر این صورت به گام ۱۱ بروید؛

گام ۷. صرفه‌جویی‌های انتخاب نشده مربوط به نقطه‌ی ابتدای تور و نقطه‌ی انتهای تور را در نظر بگیرید؛

گام ۸. با توجه به صرفه‌جویی‌های انتخاب شده، مسیر اول را با نقطه‌ی ابتدای تور و مسیر دوم را با نقطه‌ی انتهای تور بسازید؛

گام ۹. شرط شدنی بودن زمان و ظرفیت را برای مسیر اول و مسیر دوم بررسی کنید.

گام ۱۰. سه حالت ممکن است به وجود آید:

- اگر تنها یکی از مسیرها شدنی هستند، آن مسیر را برای وسیله درنظر بگیرید، مدت زمان مسیر و ظرفیت وسیله را به‌روزکنید. همچنین در ماتریس صرفه‌جویی، عدد مربوط به صرفه‌جویی آن مشتریان را صفرکنید و به گام بعد بروید؛
- اگر هر دو مسیر شدنی هستند، مسیری را انتخاب کنید که هزینه‌ی کم‌تری دارد. مدت زمان مسیرو ظرفیت وسیله را به روزکنید. همچنین در ماتریس صرفه‌جویی، عدد مربوط به صرفه‌جویی آن مشتریان را صفرکنید و به گام بعد بروید؛
- اگر هیچ‌کدام از مسیرهای اول و دوم شدنی نیستند، بیشترین مقدار صرفه‌جویی‌های انتخاب نشده را درنظر بگیرید و به گام ۸ برگردید.

گام ۱۱. وسیله‌ی نقلیه‌ی فعلی امکان بازدید نقطه‌ی جدید ندارد، در صورتی که در ماتریس صرفه‌جویی هنوز درایه‌ی مثبتی وجود دارد، مسیر جدید را برای وسیله جدید بازکنید و به گام ۲ بازگردید. در غیر این صورت، مشتریان بازدید شده‌اند. الگوریتم به انتها رسیده است.

در رابطه با الگوریتم فوق لازم است به چند نکته توجه شود. نکته اول در رابطه با استفاده از واژه‌های «مشتری» و «نقطه» است. «مشتری» به تمام مکان ـ زمان‌هایی که یک مشتری برای سرویس‌دهی معرفی کرده است، اطلاق می‌شود؛ در حالی که «نقطه» تنها مشخص‌کننده‌ی یکی از مکان ـ زمان‌های مشتری است. نکته‌ی بعدی در رابطه با به روزرسانی ماتریس صرفه‌جویی است. وقتی که یک مشتری در تور قرار می‌گیرد، ماتریس صرفه‌جویی باید به روزرسانی شود. بدین‌منظور، باید تمام صرفه‌جویی‌های مربوط به سایر مکان ـ زمان‌های مربوط به آن مشتری صفر شود تا از بازدید یک مشتری در چند مکان ـ زمان جلوگیری شود.

۲.۱.۴. الگوریتم صرفه‌جویی موازی بر مبنای زمان

گونه‌ی دوم الگوریتم‌های صرفه‌جویی، الگوریتم موازی است. در الگوریتم سری، مشتریان به صورت متوالی در تور قرار می‌گرفتند، در حالی که در الگوریتم موازی وضعیت چندین وسیله به صورت هم‌زمان بررسی می‌شود. روند کلی این الگوریتم در ادامه تشریح شده است.

گام ۱. برای نقاط به صورت دو به دو، میزان صرفه‌جویی را طبق رابطه‌ی ۲۷ محاسبه کنید؛

گام ۲. بیشترین میزان صرفه‌جویی را انتخاب کنید؛

گام ۳. دو نقطه‌ی مربوط به صرفه‌جویی انتخاب شده را با توالی دلخواه در یک مسیر جدید قرار داده و شرط شدنی بودن زمان را بررسی کنید؛

گام ۴. اگر شرط شدنی بودن زمان برقرار شد، مسیر مربوطه را برای وسیله در نظر بگیرید و مدت زمان مسیر و ظرفیت وسیله را به روزکنید و به گام ۶ بروید. در غیر این صورت جای دو نقطه را تعویض کنید و شرط شدنی بودن زمان را مجدداً بررسی کنید؛

گام ۵. در صورتی که شرط شدنی بودن زمان برقرار شد، آن را برای وسیله در نظر بگیرید. مدت زمان مسیر و ظرفیت وسیله را به روزکنید و به گام ۶ بروید. در غیر این صورت، بیشترین صرفه‌جویی بعدی را در نظر بگیرید و به گام ۳ بازگردید؛

گام ۶. بیشترین صرفه‌جویی انتخاب نشده را در نظر بگیرید؛

گام ۷. نقاط مربوط به این صرفه‌جویی را در نظرگرفته و اشتراک آن‌ها را با مسیرهای فعلی بررسی کنید؛

گام ۸. با توجه به وضعیت اشتراک دو حالت به وجود می‌آید:

- اگر صرفه‌جویی انتخاب شده با مسیرهای موجود نقطه‌ی مشترک دارد، شدنی بودن ادغام نقطه‌ی غیرمشترک را با آن مسیر بررسی کنید. در صورتی که ادغام شدنی است، مسیر جدید را تشکیل دهید. ماتریس صرفه‌جویی و ظرفیت وسیله نقلیه را به روزکنید؛
- اگر صرفه‌جویی انتخاب شده با مسیرهای موجود نقطه‌ی اشتراکی ندارد، مشتریان انتخاب شده را در یک وسیله‌ی جدید قرار دهید. ماتریس صرفه‌جویی و ظرفیت وسیله نقلیه را به روزکنید.

گام ۹. اگر در ماتریس صرفه‌جویی درایه‌ی مثبتی وجود دارد، به گام ۶ بازگردید. در غیر این صورت، اگر تمام درایه‌های ماتریس صرفه‌جویی صفر هستند، یعنی تمام مشتریان بازدید شده‌اند و الگوریتم به پایان رسیده است.

۲.۴. الگوریتم‌های ابتکاری با محوریت درج کردن

رویکرد دومی که برای الگوریتم‌های ابتکاری توسعه داده شده است، استفاده از رویکرد درج مشتریان است. در این رویکرد، مشتریان به صورت متوالی در تورها قرار می‌گیرند. بر این اساس، تمام گزینه‌های درج مشتریان در تور، بررسی شده و بهترین مکان برای درج مشتری انتخاب می‌شود. برای درج مشتری اول در تور و شروع کردن الگوریتم، راهکارهای متفاوتی متصور است. بر این اساس، الگوریتم‌های متفاوتی را می‌توان توسعه داد. سولومن[۲۱] سه راهکار برای شروع تور معرفی کرده است. راهکار اول، انتخاب دورترین نقطه نسبت به دپو؛ راهکار دوم، انتخاب نقطه با نزدیک‌ترین حد بالای پنجره‌ی زمانی (b)؛ و راهکار سوم، ترکیبی راهکار اول و دوم است. در این مقاله پنج راهکار برای انتخاب اولین نقطه‌ی تور معرفی می‌شود. راهکارهای شروع تور، در ادامه عنوان می‌شود.

I۱: انتخاب نقطه با بیشترین فاصله نسبت به دپو؛

I۲: انتخاب نقطه با نزدیک‌ترین حد پایین پنجره‌ی زمانی (a)؛

I۳: انتخاب نقطه با نزدیک‌ترین حد بالای پنجره‌ی زمانی (b)؛

I۴: انتخاب نقطه با بحرانی‌ترین پنجره‌ی زمانی (کم‌ترین مقدار $a-b$)؛

I۵: انتخاب نقطه به صورت تصادفی.

برای توضیح راهکارهای ذکر شده، یکی از این راهکارها تشریح می‌شود. به عنوان مثال، در راهکار انتخاب نقطه با بحرانی‌ترین پنجره‌ی زمانی، مقدار $a-b$ برای تمام نقاط محاسبه می‌شود. هرچه $a-b$ مقدار کم‌تری داشته باشد، پنجره‌ی‌زمانی آن بحرانی‌تر محسوب می‌شود؛ یعنی مشتری در یک بازه محدود باید بازدید شود؛

پس از شروع الگوریتم، مشتریان طبق یک توالی مشخص در تور قرار می‌گیرند. برای توالی اضافه کردن مشتریان در تور نیز می‌توان راهکارهای متفاوتی را در نظر گرفت. در این مقاله پنج راهکار برای تعیین توالی اضافه شدن مشتریان در تور معرفی می‌شود. این راهکارها در ادامه عنوان شده است.

S۱: توالی مشتریان بر اساس بیشترین تقاضا؛

S۲: توالی مشتریان بر اساس کم‌ترین حد پایین پنجره‌ی زمانی (a)؛

S۳: توالی مشتریان بر اساس کم‌ترین حد بالای پنجره‌ی زمانی (b)؛

S۴: توالی مشتریان بر اساس بحرانی‌ترین پنجره‌ی زمانی $(b-a)$؛

S۵: توالی تصادفی مشتریان.

به عنوان توضیح یکی از راهکارهای تعیین توالی تشریح می‌شود. بدین منظور، راهکار S۳ تشریح می‌شود. در این راهکار، بیشترین حد بالای پنجره‌ی زمانی (b) برای تمام مکان ـ زمان‌های آن در نظرگرفته می‌شود. در مرحله‌ی بعد، مشتریان با توجه به این مقادیر مرتب می‌شوند. در واقع با انجام این کار، مشتریان بر اساس زودترین گزینه‌ی تحویلی ممکن مرتب می‌شوند.

با در نظرگرفتن راهکارهای معرفی شده برای انتخاب مشتری اول و راهکارهای معرفی شده برای تعیین توالی مشتریان، در مجموع ۵ × ۵ الگوریتم برای رویکرد درج مشتریان می‌توان در نظر گرفت. در ادامه، روند کلی الگوریتم درج مشتریان تشریح می‌شود.

گام ۱. مشتریان را بر اساس یکی از راهکارهای توالی، منظم کنید؛

گام ۲. طبق یکی از راهکارهای شروع تور، یکی از مشتریان انتخاب نشده را برای قرارگرفتن در تور انتخاب کنید؛

گام ۳. اگر وسیله‌ی فعلی ظرفیت دارد، به گام بعد بروید. در غیر این صورت، یک وسیله‌ی جدید ایجاد کنید و به گام ۲ بازگردید؛

گام ۴. تمام مکان ـ زمان‌های مشتری انتخاب شده را در تمام جایگاه‌ها موجود در تور بررسی کنید. یک لیست ایجاد کنید و هزینه‌ی جایگذاری مکان ـ زمان‌های مختلف را در صورت شدنی بودن، در لیست در نظر بگیرید؛

گام ۵. در لیست ایجاد شده کم‌ترین هزینه را انتخاب کرده و مکان ـ زمان مربوط به مشتری فعلی را در جایگاهی با کم‌ترین هزینه، قرار دهید. زمانِ مسیر و ظرفیت وسیله را به روزکرده و مشتری فعلی را به عنوان مشتری بازدید شده، علامت‌گذاری کنید. در صورتی که در لیست جایگاه شدنی برای هیچ یک از مکان ـ زمان‌های مشتری وجود ندارد، مشتری را به عنوان مشتری بازدید نشده، علامت بزنید؛

گام ۶. در صورتی که ترتیب مشتریان به انتها رسیده، الگوریتم متوقف می‌شود؛ در غیر این صورت، مشتری بعدی را انتخاب کنید و به گام ۳ بازگردید.

در الگوریتم این مقاله، برای راهکارهای آغاز تور و راهکار توالی ورود مشتریان به تور، از رویکرد ترکیبی استفاده می‌شود. بدین منظور از راهکارهای I۱ و I۳ برای شروع تور و از راهکارهای $S3$ و $S4$ به عنوان راهکارهای تعیین توالی استفاده شده است.

۳.۴. الگوریتم ابتکاری با محوریت نزدیک‌ترین مشتری دیدار نشده

الگوریتم نزدیک‌ترین مشتری دیدار نشده، یکی از الگوریتم‌های معروف برای حل مسئله‌ی فروشنده دوره‌گرد است. این الگوریتم به صورت حریصانه عمل می‌کند. الگوریتم در هر مرحله نزدیک‌ترین مشتری دیدار نشده را نسبت به نقطه‌ی فعلی پیدا کرده و به مسیر اضافه می‌کند. سولومن[۲۱] در الگوریتمی که برای مسئله‌ی مسیریابی همراه با درنظرگرفتن پنجره‌ی زمانی توسعه داده است، از سه معیار برای فاصله، مدت زمان و معیاری برای بحرانی بودن مشتریان، استفاده می‌کند تا نزدیک‌ترین مشتری را پیدا کند. در این مقاله، با توجه به تابع هدف در نظرگرفته شده که از جنس زمان است، از دو معیار استفاده می‌شود. معیار اول، مدت زمان فاصله‌ی میان دو نقطه (رابطه‌ی ۲۸) است. معیار دوم، به عنوان معیار بحرانی بودن نقاط (رابطه‌ی ۲۹) در نظرگرفته می‌شود.

$$T_{ij} = \Delta_j - (\Delta_i + St_i) \quad (۲۸)$$

$$V_{ij} = b_j - (\Delta_i + St_i + t_{ij}) \quad (۲۹)$$

در نهایت معیار نزدیکی مشتریان، به صورت جمع وزنی دو معیار فوق در نظرگرفته می‌شود. معیار نزدیکی برای مشتریان در رابطه‌ی ۳۰ مشخص شده است.

$$C_{ij} = \alpha T_{ij} + \beta V_{ij} \quad (۳۰)$$

با در نظرگرفتن توضیحات فوق، روند کلی الگوریتم نزدیک‌ترین مشتری دیدار نشده، در ادامه تشریح می‌شود.

گام ۱. یک وسیله‌ی جدید در نظر بگیرید. اندیس i را برابر با اندیس دپو در نظر بگیرید؛

گام ۲. معیارهای T_{ij} و V_{ij} را برای تمام مکان ـ زمان‌های مربوط به مشتریان بازدید نشده، محاسبه کنید و C_{ij} را به دست آورید؛

گام ۳. کم‌ترین C_{ij} را انتخاب کنید. اگر تقاضای مربوط به این نقطه از ظرفیت وسیله کم‌تر است، آن را در وسیله‌ی فعلی قرار دهید. مشتری مربوط به این مکان ـ زمان را به عنوان مشتری بازدید شده، علامت بزنید. ظرفیت وسیله نقلیه و مدت زمان آن را به روز کنید.

۴.۴. الگوریتم بهبود جواب اولیه

این الگوریتم به نوعی مکمل الگوریتم‌های قبلی محسوب می‌شود. هدف این الگوریتم، بهبود جواب‌های حاصل شده از الگوریتم‌های ابتکاری ارائه شده قبلی است. روند کلی الگوریتم به این صورت است که تمام نقاط فعلی حاضر در تورها، از جایگاه خود حذف شده و مجدداً بهترین جایگاه برای مکان ـ زمان‌های مشتری حذف شده، بررسی می‌شود. در صورتی که یکی از مکان ـ زمان‌ها هزینه‌ی کم‌تری داشت، آن نقطه در مسیر مربوطه قرار گرفته و اطلاعات مربوط به مسیرها به روزرسانی می‌شود. در غیر این صورت، همان جایگاه قبلی برای آن نقطه حفظ می‌شود.

۵. نتایج عددی

در این قسمت نتایج عددی الگوریتم‌های توسعه داده شده مقایسه می‌شوند. این الگوریتم‌ها در نرم‌افزار متلب ۲۰۱۹ پیاده‌سازی شده و با یک رایانه با پردازشگر core i۳ – ۸۱۰۰ و حافظه‌ی رم ۸ گیگابایت اجرا شدند. در ادامه، ابتدا توضیحاتی در رابطه با داده‌های مورد استفاده ارائه می‌شود، سپس نتایج عددی حاصل، مورد بحث و بررسی قرار می‌گیرند.

۱.۵. داده‌های استفاده شده

با توجه به این که مسئله‌ی این مقاله، با مسئله‌ی VRPTW [۷] مشابهت دارد، برای مقایسه‌ی عملکرد الگوریتم‌های طراحی شده، از داده‌های R۱، C۱ و RC۱ معرفی شده در مقاله‌ی سولومن[۲۱] استفاده می‌شود؛ اما با توجه به این که نقاط گراف باید خوشه‌بندی شوند، این نقاط به صورت تصادفی با یکدیگر مرتبط شده‌اند. بدین منظور، یک عدد تصادفی در بازه [۱, ۳] برای هر مشتری در نظرگرفته می‌شود. این عدد مشخص‌کننده‌ی تعداد مکان ـ زمان‌های آن مشتری است. سپس با توجه به اعداد تولید شده، نقاط گراف به مشتریان تخصیص می‌یابند. لازم به ذکر است که داده‌های مقاله، بنا به درخواست خواننده در قالب فایل اکسل قابل ارسال است.

سولومن،[۲۱] داده‌های خود را با سه رویکرد ساخته است. رویکرد اول (R)، در نظرگرفتن مکان مشتریان به صورت تصادفی است. رویکرد دوم (C)، نقاط را به صورت خوشه‌های مکانی متعدد در نظر می‌گیرد و در نهایت رویکرد سوم (RC)، نقاط را به صورت ترکیبی از نقاط تصادفی و نقاط خوشه‌بندی شده در نظر می‌گیرد. متفاوت بودن این رویکردها برای تولید مثال‌های مختلف، می‌تواند عملکرد الگوریتم‌های طراحی شده را به چالش بکشد.

۲.۵. مقایسه‌ی مدل‌های ریاضی

به منظور صحت‌سنجی الگوریتم‌های طراحی شده و همچنین مقایسه مدل‌های ریاضی فرموله شده، از داده‌های سایز پایین (سایز ۲۵ نقطه) سولومن[۲۱] استفاده می‌شود. نتایج این مقایسه در جدول ۱ درج شده است. در این جدول، میزان تابع هدف، بهترین حد پایین گزارش شده از نرم‌افزار گمز، خطای نسبی میان تابع هدف و حد پایین (Optcr) و همچنین زمان حل مدل گزارش شده است. اگر مقدار تابع هدف با حد پائین یکسان شود، به معنای رسیدن به جواب بهینه است. چنان که مشهود است، در دسته‌ی C مدل $P۲$ در تمام مسائل به جواب بهینه رسیده است، در حالی که مدل $P۱$ برای سه مسئله، در مدت زمان ۳۶۰۰ ثانیه، از منطبق کردن حد پایین بر حد بالا ناتوان بوده است. این شرایط در دسته‌ی R برای یک مسئله و در دسته‌ی RC، برای پنج مسئله تکرار شده است. علاوه بر حد پایین، زمان حل مدل‌ها نیز اختلاف قابل توجهی دارد. مدل $P۲$ به صورت میانگین در حدود ۲۲ مرتبه سریع‌تر از مدل $P۱$ به جواب بهینه رسیده است.

۳.۵. نتایج عددی الگوریتم‌های ابتکاری در ابعاد کوچک

در قسمت قبل مدل‌های ریاضی $P۱$ و $P۲$ با یکدیگر مقایسه شدند. نتایج حاصل نشان‌دهنده برتری مدل $P۲$ نسبت به مدل $P۱$ بود. در این قسمت عملکرد الگوریتم‌های ابتکاری توسعه داده شده، در سایز پایین (۲۵ نقطه) با مدل $P۲$ مقایسه می‌شوند. نتایج حاصل در جدول ۲ نشان داده شده است. در این جدول مقدار تابع هدف حاصل از روش‌های مختلف، با جواب بهینه مسئله مقایسه شده است.

نتایج موجود در جدول ۲ نشان می‌دهد که الگوریتم ابتکاری درج کردن (Ins)،

جدول ۱. نتایج عددی مقایسه مدل‌های ریاضی P۱ و P۲.

دسته	شماره مسئله	P۱				P۲			
		تابع هدف	حد پایین	Optcr	زمان (s)	تابع هدف	حد پایین	Optcr	زمان (s)
#C	C101	۱۲۲/۹	۱۲۲/۹	۰/۰	۱/۱	۱۲۲/۹	۱۲۲/۹	۰/۰	۰/۸
	C102	۱۳۸/۱	۱۳۸/۱	۰/۰	۶۶/۳	۱۳۸/۱	۱۳۸/۱	۰/۰	۶/۸
	C103	۱۳۷/۷	۸۵/۶	۰/۳۸	۳۶۰۰/۴	۱۰۹/۹	۱۰۹/۹	۰/۰	۳۴/۸
	C104	۱۸۰/۶	۸۵/۲	۰/۵۳	۳۶۰۰/۳	۱۰۸/۸	۱۰۸/۸	۰/۰	۱۶۹۶/۲
	C105	۱۴۴/۶	۱۴۴/۶	۰/۰	۳۳۰/۳	۱۴۴/۶	۱۴۴/۶	۰/۰	۴۵/۳
	C106	۱۲۸/۵	۱۲۸/۵	۰/۰	۰/۸	۱۲۸/۵	۱۲۸/۵	۰/۰	۰/۵
	C107	۱۲۲/۹	۱۲۲/۹	۰/۰	۱/۲	۱۲۲/۹	۱۲۲/۹	۰/۰	۰/۷
	C108	۱۲۶/۹	۱۲۶/۹	۰/۰	۹۴/۸	۱۲۶/۹	۱۲۶/۹	۰/۰	۴/۹
	C109	۱۲۷/۲	۱۱۸/۴	۰/۰۷	۳۶۰۰/۴	۱۲۷/۲	۱۲۷/۲	۰/۰	۳۸/۶
	میانگین	–	–	۰/۱۱	۱۲۵۵/۱	–	–	۰/۰	۲۰۳/۲
#R	R101	۳۲۲/۸	۳۲۲/۸	۰/۰	۰/۹	۳۲۲/۸	۳۲۲/۸	۰/۰	۰/۵
	R102	۲۳۷/۶	۲۳۷/۶	۰/۰	۷۶/۴	۲۳۷/۶	۲۳۷/۶	۰/۰	۵/۷
	R103	۲۳۷/۷	۲۳۷/۷	۰/۰	۱۰۲/۹	۲۳۷/۷	۲۳۷/۷	۰/۰	۵/۶
	R104	۱۰۲/۵	۱۰۲/۵	۰/۰	۳۱۰۳/۲	۱۰۲/۵	۱۰۲/۵	۰/۰	۱۴۲/۷
	R105	۲۷۴/۹	۲۷۴/۹	۰/۰	۵/۸	۲۷۴/۹	۲۷۴/۹	۰/۰	۰/۸
	R106	۱۹۵/۷	۱۹۵/۷	۰/۰	۵۱۷/۳	۱۹۵/۷	۱۹۵/۷	۰/۰	۴۰/۴
	R107	۱۹۵/۸	۱۹۵/۸	۰/۰	۶۵۹/۱	۱۹۵/۸	۱۹۵/۸	۰/۰	۲۷/۱
	R108	۲۰۴/۳	۱۷۶/۰	۰/۱۴	۳۶۰۰/۴	۲۰۴/۱	۲۰۴/۱	۰/۰	۱۱۶/۵
	R109	۲۵۳/۱	۲۵۳/۱	۰/۰	۲۱۶/۳	۲۵۳/۱	۲۵۳/۱	۰/۰	۱۳/۲
	R110	۲۱۵/۲	۲۱۵/۲	۰/۰	۴۷۹/۹	۲۱۵/۲	۲۱۵/۲	۰/۰	۱۷/۸
	R111	۲۳۷/۷	۲۳۷/۷	۰/۰	۱۱۹۳/۴	۲۳۷/۷	۲۳۷/۷	۰/۰	۳۰/۴
	R112	۱۸۷/۰	۱۸۷/۰	۰/۰	۲۲۶۸/۶	۱۸۷/۰	۱۸۷/۰	۰/۰	۶۹/۹
	میانگین	–	–	۰/۰۱	۱۰۱۸/۷	–	–	۰/۰	۳۹/۲
#RC	RC101	۲۷۹/۹	۲۷۹/۹	۰/۰	۵۸/۳	۲۷۹/۹	۲۷۹/۹	۰/۰	۱/۳
	RC102	۲۴۲/۵	۲۴۲/۵	۰/۰	۲۷۹/۹	۲۴۲/۵	۲۴۲/۵	۰/۰	۱۱۲/۱
	RC103	۲۲۶/۴	۱۳۷/۹	۰/۳۹	۳۶۰۰/۳	۲۲۶/۴	۲۲۶/۴	۰/۰	۸۹۸/۳
	RC104	۲۲۱/۲	۹۰/۲	۰/۵۹	۳۶۰۰/۲	۲۱۵/۲	۱۴۳/۸	۰/۳۶	۳۶۰۰/۴
	RC105	۲۶۳/۷	۱۹۸/۱	۰/۲۵	۳۶۰۰/۴	۲۶۳/۷	۲۶۳/۷	۰/۰	۲۶۳/۳
	RC106	۲۱۶/۹	۲۱۶/۹	۰/۰	۱۲۷۱/۶	۲۱۶/۹	۲۱۶/۹	۰/۰	۲۵/۵
	RC107	۲۴۸/۷	۲۱۵/۱	۰/۱۴	۳۶۰۰/۳	۲۴۸/۷	۲۴۸/۷	۰/۰	۱۰۰/۴
	RC108	۲۲۲/۹	۱۱۰/۷	۰/۵۰	۳۶۰۰/۲	۲۲۲/۹	۱۳۵/۶	۰/۳۹	۳۶۰۰/۶
	میانگین	–	–	۰/۲۳	۲۴۵۱/۴	–	–	۰/۰۹	۱۰۷۵/۲

نسبت به سایر الگوریتم‌ها در هر سه دسته مسائل عملکرد بهتری داشته است. مقدار خطای این الگوریتم در دسته‌ی C برابر ۰/۰۷، در دسته‌ی R برابر با ۰/۰۸ و در دسته‌ی RC میزان خطا برابر ۰/۰۵ است. این الگوریتم در بین تمام نمونه‌ها، به صورت میانگین با حدود ۰/۰۶ خطا به جواب نهایی رسیده است. نکته‌ی قابل توجه این‌که عملکرد الگوریتم مبتنی بر درج در نمود مسئله‌ی RC۱۰۸، از عملکرد گمز در مدت زمان مجاز (۳۶۰۰ ثانیه)، بهتر بوده است. همچنین مشخص است که الگوریتم ابتکاری نزدیک‌ترین مشتری دیدار نشده، عملکرد ضعیفی ارائه کرده است. با در نظرگرفتن رویکرد حریصانه‌ی آن و عدم توجه به ابعاد مختلف، عملکرد ضعیف آن قابل توجیه است. نکته‌ی دیگر در رابطه با عملکرد نه چندان کارای الگوریتم صرفه‌جویی در حالت موازی است. این الگوریتم یکی از الگوریتم‌های نسبتاً قوی برای حل مسائل کلاسیک مسیریابی محسوب می‌شود، اما برای مسئله‌ی مطرح در این مقاله عملکرد مناسبی نداشته است. دلیل این پیشامد را شاید بتوان تعداد زیاد گزینه‌های انتخابی دانست. تعدد مکان ـ زمان‌ها برای هر مشتری، ممکن است منجر به ساخت مسیرهای متفاوت شده و به بدتر شدن جواب مسئله بینجامد.

۴.۵. مقایسه الگوریتم‌های ابتکاری در ابعاد بزرگ

در قسمت قبلی کارایی الگوریتم‌ها با داده‌هایی با سایز ۲۵ نقطه و ۱۲ مشتری بررسی شد. در این قسمت به منظور بررسی قدرت الگوریتم‌های طراحی شده،

جدول ۲. نتایج عددی حاصل از مقایسه مدل ریاضی P۲ الگوریتم‌های ابتکاری طراحی شده در ابعاد کوچک.

دسته	شماره مسئله	P۲		Ins			Nearest_N			Par_Saving			Seq_Saving		
		تابع هدف	زمان (s)	تابع هدف	Gap	زمان (s)	تابع هدف	Gap	زمان (s)	تابع هدف	Gap	زمان (s)	تابع هدف	Gap	زمان (s)
#C	C101	۱۲۲/۹	۰/۸	۱۲۶/۴	۰/۰۳	۰/۰۲۲	۱۴۶/۰	۰/۱۶	۰/۰۱۲	۲۵۵/۴	۰/۵۲	۰/۰۱۴	۲۵۶/۵	۰/۵۲	۰/۰۱۴
	C102	۱۳۸/۱	۶/۸	۱۵۶/۱	۰/۱۳	۰/۰۲۵	۲۴۵/۱	۰/۴۴	۰/۰۱۷	۲۲۴/۵	۰/۳۸	۰/۰۱۹	۱۷۸/۶	۰/۲۳	۰/۰۱۲
	C103	۱۰۹/۹	۳۴/۸	۱۱۸/۲	۰/۰۸	۰/۰۷۴	۱۵۶/۴	۰/۳۰	۰/۰۱۰	۲۵۷/۶	۰/۵۷	۰/۰۲۲	۱۹۷/۲	۰/۴۴	۰/۰۲۵
	C104	۱۰۸/۸	۱۶۹۶/۲	۱۱۵/۱	۰/۰۶	۰/۰۱۹	۲۳۷/۴	۰/۵۴	۰/۰۰۹	۱۹۳/۸	۰/۴۴	۰/۰۲۱	۱۹۴/۶	۰/۴۴	۰/۰۱۳
	C105	۱۴۴/۶	۴۵/۳	۱۶۲/۰	۰/۱۲	۰/۰۲۵	۲۸۹/۵	۰/۵۰	۰/۰۱۷	۲۳۴/۳	۰/۳۸	۰/۰۱۹	۱۷۰/۶	۰/۱۵	۰/۰۱۱
	C106	۱۲۸/۵	۰/۵	۱۲۸/۵	۰/۰۰	۰/۰۲۲	۱۳۷/۴	۰/۰۶	۰/۰۱۲	۳۷۰/۸	۰/۶۵	۰/۰۱۵	۲۵۶/۵	۰/۵۰	۰/۰۱۳
	C107	۱۲۲/۹	۰/۷	۱۲۶/۴	۰/۰۳	۰/۰۲۱	۱۴۶/۰	۰/۱۶	۰/۰۱۲	۲۵۵/۴	۰/۵۲	۰/۰۱۴	۲۵۶/۵	۰/۵۲	۰/۰۱۷
	C108	۱۲۶/۹	۴/۹	۱۴۴/۷	۰/۱۴	۰/۰۲۵	۲۱۷/۲	۰/۴۲	۰/۰۰۸	۲۳۴/۳	۰/۴۶	۰/۰۱۴	۱۷۲/۶	۰/۲۶	۰/۰۱۴
	C109	۱۲۷/۲	۳۸/۶	۱۳۷/۰	۰/۰۸	۰/۰۲۲	۲۱۰/۱	۰/۳۹	۰/۰۱۱	۱۵۸/۴	۰/۲۰	۰/۰۱۵	۱۵۴/۱	۰/۱۷	۰/۰۱۵
	میانگین	–	۲۰۳/۲	–	۰/۰۷	۰/۰۲۸	–	۰/۳۳	۰/۰۱۱	–	۰/۴۶	۰/۰۱۷	–	۰/۳۶	۰/۰۱۴
#R	R101	۳۲۲/۸	۰/۵	۳۴۶/۷	۰/۰۷	۰/۰۲۶	۴۲۳/۳	۰/۲۴	۰/۰۱۸	۳۶۰/۱	۰/۱۰	۰/۰۲۲	۴۲۶/۷	۰/۲۴	۰/۰۲۰
	R102	۲۳۷/۶	۵/۷	۲۴۳/۶	۰/۰۳	۰/۰۲۸	۳۷۰/۱	۰/۳۶	۰/۰۱۲	۳۴۲/۸	۰/۳۱	۰/۰۲۴	۲۴۰/۶	۰/۰۱	۰/۰۱۵
	R103	۲۳۷/۷	۵/۶	۲۴۳/۷	۰/۰۳	۰/۰۳۰	۳۷۰/۲	۰/۳۶	۰/۰۱۷	۳۴۴/۹	۰/۳۱	۰/۰۲۴	۲۴۰/۷	۰/۰۱	۰/۰۱۶
	R104	۱۰۲/۵	۱۴۲/۷	۱۰۹/۳	۰/۰۷	۰/۰۴۷	۱۴۷/۰	۰/۳۰	۰/۰۱۶	۱۶۰/۶	۰/۳۶	۰/۰۱۹	۱۲۸/۴	۰/۲۰	۰/۰۴۱
	R105	۲۷۴/۹	۰/۸	۳۰۰/۷	۰/۰۹	۰/۰۳۰	۳۷۲/۵	۰/۲۶	۰/۰۱۰	۳۵۱/۳	۰/۲۲	۰/۰۲۳	۳۵۴/۱	۰/۲۲	۰/۰۱۴
	R106	۱۹۵/۷	۴۰/۴	۲۰۳/۹	۰/۰۴	۰/۰۳۲	۲۷۰/۵	۰/۲۸	۰/۰۱۱	۲۸۷/۵	۰/۳۲	۰/۰۲۱	۲۶۴/۶	۰/۲۶	۰/۰۱۷
	R107	۱۹۵/۸	۲۷/۱	۲۱۶/۵	۰/۱۱	۰/۰۲۹	۲۵۲/۵	۰/۲۲	۰/۰۰۹	۲۷۴/۶	۰/۲۹	۰/۰۱۶	۲۵۰/۲	۰/۲۲	۰/۰۱۴
	R108	۲۰۴/۱	۱۱۶/۵	۲۲۱/۵	۰/۰۹	۰/۰۲۰	۲۹۴/۱	۰/۳۱	۰/۰۰۸	۲۳۶/۷	۰/۱۴	۰/۰۲۰	۲۴۱/۹	۰/۱۶	۰/۰۱۲
	R109	۲۵۳/۱	۱۳/۲	۲۸۰/۷	۰/۱۱	۰/۰۱۹	۳۲۰/۱	۰/۲۱	۰/۰۰۸	۲۹۳/۲	۰/۱۴	۰/۰۱۷	۲۶۹/۲	۰/۰۶	۰/۰۱۲
	R110	۲۱۵/۲	۱۷/۸	۲۳۵/۷	۰/۱۰	۰/۰۲۵	۳۷۰/۰	۰/۴۲	۰/۰۰۷	۳۰۹/۷	۰/۳۱	۰/۰۱۷	۲۰۸/۷	۰/۰۰	۰/۰۱۰
	R111	۲۳۷/۷	۳۰/۴	۲۵۷/۱	۰/۱۰	۰/۰۳۰	۳۱۱/۶	۰/۲۵	۰/۰۱۰	۳۴۳/۳	۰/۳۲	۰/۰۲۳	۳۵۲/۷	۰/۳۴	۰/۰۱۵
	R112	۱۸۷/۰	۶۹/۹	۲۰۸/۲	۰/۱۱	۰/۰۲۱	۲۷۰/۹	۰/۳۱	۰/۰۱۰	۲۹۵/۳	۰/۳۷	۰/۰۱۶	۲۱۲/۱	۰/۱۲	۰/۰۰۹
	میانگین	–	۳۹/۲	–	۰/۰۸	۰/۰۲۸	–	۰/۲۹	۰/۰۱۱	–	۰/۲۶	۰/۰۲۰	–	۰/۱۵	۰/۰۱۶
#RC	RC101	۲۷۹/۹	۱/۳	۲۸۸/۳	۰/۰۳	۰/۰۳۴	۳۶۶/۸	۰/۲۴	۰/۰۱۶	۳۴۸/۶	۰/۲۰	۰/۰۳۳	۳۰۸/۲	۰/۰۹	۰/۰۳۷
	RC102	۲۴۲/۵	۱۱۲/۱	۲۴۲/۷	۰/۰۰۱	۰/۰۳۳	۴۰۸/۹	۰/۴۱	۰/۰۱۴	۲۹۲/۰	۰/۱۷	۰/۰۱۷	۲۸۰/۶	۰/۱۴	۰/۰۱۵
	RC103	۲۲۶/۴	۸۹۸/۳	۲۴۹/۴	۰/۱۰	۰/۰۲۴	۳۵۹/۱	۰/۳۷	۰/۰۱۱	۲۷۳/۹	۰/۱۷	۰/۰۲۵	۲۸۷/۳	۰/۲۱	۰/۰۲۸
	RC104	۲۱۵/۲	۳۶۰۰/۴	۲۴۳/۹	۰/۱۳	۰/۰۲۶	۴۶۸/۵	۰/۵۴	۰/۰۱۶	۲۶۹/۴	۰/۲۰	۰/۰۱۷	۲۶۷/۷	۰/۲۰	۰/۰۱۵
	RC105	۲۶۳/۷	۲۶۳/۳	۲۶۵/۷	۰/۰۱	۰/۰۲۵	۳۵۷/۶	۰/۲۶	۰/۰۱۲	۲۶۶/۶	۰/۰۱	۰/۰۱۵	۲۹۸/۶	۰/۱۲	۰/۰۱۶
	RC106	۲۱۶/۹	۲۵/۵	۲۴۲/۹	۰/۱۲	۰/۰۳۱	۴۲۰/۶	۰/۴۸	۰/۰۰۹	۲۷۱/۰	۰/۲۰	۰/۰۱۷	۲۹۳/۳	۰/۲۶	۰/۰۱۷
	RC107	۲۴۸/۷	۱۰۰/۴	۲۶۰/۳	۰/۰۵	۰/۰۲۲	۳۹۵/۰	۰/۳۷	۰/۰۱۰	۲۶۹/۲	۰/۰۸	۰/۰۱۷	۲۶۲/۸	۰/۰۵	۰/۰۲۰
	RC108	۲۲۲/۹	۳۶۰۰/۶	۲۲۲/۴	−۰/۰۰۲	۰/۰۲۹	۵۱۳/۶	۰/۵۷	۰/۰۰۹	۲۶۰/۲	۰/۱۴	۰/۰۱۳	۳۰۲/۵	۰/۲۶	۰/۰۱۵
	میانگین	–	۱۰۷۵/۲	–	۰/۰۵	۰/۰۲۸	–	۰/۴۰	۰/۰۱۲	–	۰/۱۵	۰/۰۱۹	–	۰/۱۷	۰/۰۲۰

از داده‌هایی با سایز ۱۰۰ نقطه و ۵۰ مشتری استفاده می‌شود. نتایج حاصل از بررسی این الگوریتم‌ها و مقایسه قدرت و کارایی آنها با یکدیگر، در جدول ۳ درج شده است. چنان که مشاهده می‌شود، الگوریتم درج کردن (Ins) در ابعاد بزرگ نیز با میانگین خطای ۰/۰۲۱ در دسته C، میانگین خطای ۰/۰۰۷ در دسته‌ی R و میانگین خطای ۰/۰۲۱ در دسته‌ی RC و در مجموع خطای ۰/۰۱۶ عملکرد مناسبی نسبت به سایر الگوریتم‌ها داشته است. پس از الگوریتم درج کردن، به ترتیب الگوریتم‌های صرفه‌جویی سری، صرفه‌جویی موازی و در نهایت الگوریتم مبتنی بر نزدیک‌ترین مشتری دیدار نشده قرار می‌گیرند.

۵.۵. بررسی عملکرد الگوریتم ابتکاری بهبود جواب

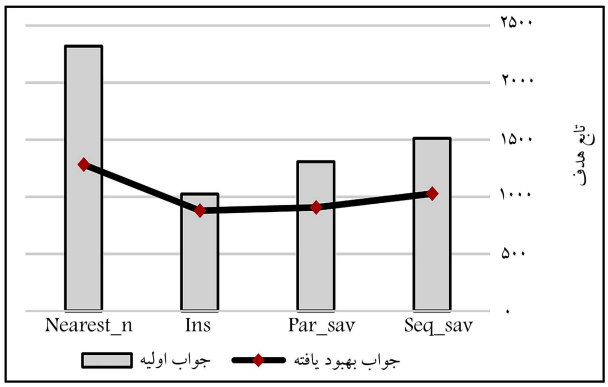
چنان که در قسمت‌های قبل عنوان شد، روش ابتکاری استفاده شده در این مقاله،

جدول ۳. نتایج عددی حاصل از مقایسه الگوریتم‌های ابتکاری در ابعاد بزرگ.

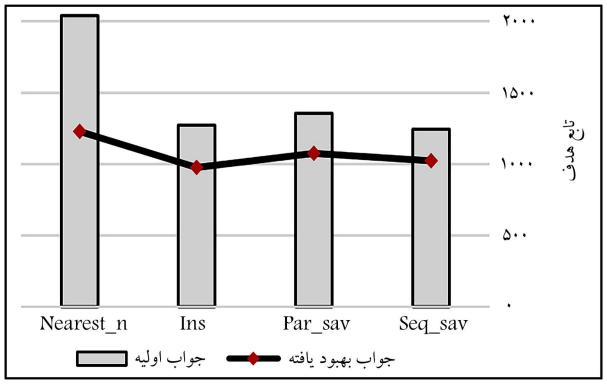
دسته	شماره مسئله	Ins			Nearest_N			Par_Saving			Seq_Saving		
		تابع هدف	Gap	زمان (s)	تابع هدف	Gap	زمان (s)	تابع هدف	Gap	زمان (s)	تابع هدف	Gap	زمان (s)
#C	C101	۹۹۹/۸	۰/۰۰۰	۰/۲۶۴	۱۳۹۴/۹	۰/۳۹۵	۰/۰۹۵	۱۱۰۲/۲	۰/۱۰۳	۰/۳۲۹	۱۰۶۴/۰	۰/۰۶۴	۰/۱۶۱
	C102	۶۷۰/۸	۰/۰۰۰	۰/۲۲۶	۱۲۱۰/۲	۰/۸۰۴	۰/۱۳۶	۷۱۵/۸	۰/۰۶۷	۰/۲۹۵	۸۵۴/۵	۰/۲۷۴	۰/۲۳۷
	C103	۶۵۸/۷	۰/۰۰۰	۰/۲۰۴	۱۱۱۷/۵	۰/۶۳۰	۰/۰۹۴	۶۹۷/۱	۰/۰۱۷	۰/۲۵۲	۹۵۵/۷	۰/۳۹۴	۰/۱۴۸
	C104	۷۰۷/۱	۰/۰۱۴	۰/۲۱۷	۱۰۶۹/۸	۰/۵۳۵	۰/۰۹۰	۶۹۷/۲	۰/۰۰۰	۰/۲۵۸	۸۳۸/۸	۰/۲۰۳	۰/۱۵۰
	C105	۱۲۶۳/۸	۰/۰۷۳	۰/۱۸۲	۱۵۸۶/۱	۰/۳۴۷	۰/۰۸۵	۱۴۲۱/۳	۰/۲۰۷	۰/۳۲۳	۱۱۷۷/۷	۰/۰۰۰	۰/۱۶۳
	C106	۹۴۶/۹	۰/۰۶۷	۰/۱۶۲	۱۳۵۵/۵	۰/۵۲۸	۰/۰۷۸	۸۸۷/۲	۰/۰۰۰	۰/۲۰۹	۱۱۵۴/۶	۰/۳۰۱	۰/۱۵۷
	C107	۹۳۴/۱	۰/۰۱۷	۰/۱۷۰	۱۱۹۱/۲	۰/۲۹۶	۰/۰۷۹	۹۱۸/۹	۰/۰۰۰	۰/۲۰۲	۱۱۵۹/۷	۰/۲۶۲	۰/۱۶۱
	C108	۹۳۷/۰	۰/۰۱۸	۰/۲۰۸	۱۳۰۴/۷	۰/۴۱۸	۰/۰۹۸	۹۲۰/۳	۰/۰۰۰	۰/۲۲۸	۱۲۱۵/۸	۰/۳۲۱	۰/۱۶۸
	C109	۷۵۲/۷	۰/۰۰۰	۰/۱۷۸	۱۳۲۳/۶	۰/۷۵۸	۰/۰۷۵	۸۰۷/۸	۰/۰۷۳	۰/۲۴۰	۸۲۹/۲	۰/۱۰۲	۰/۱۵۶
	میانگین	–	۰/۰۲۱	۰/۲۰۱	–	۰/۵۲۳	۰/۰۹۲	–	۰/۰۵۲	۰/۲۶۰	–	۰/۲۱۳	۰/۱۶۷
#R	R101	۱۰۴۵/۰	۰/۰۲۰	۰/۲۲۵	۱۲۸۲/۸	۰/۲۵۲	۰/۰۹۵	۱۱۲۷/۵	۰/۱۰۰	۰/۴۳۷	۱۰۲۴/۶	۰/۰۰۰	۰/۱۸۱
	R102	۱۰۴۸/۰	۰/۰۲۳	۰/۲۰۸	۱۲۲۱/۱	۰/۱۹۲	۰/۱۱۲	۱۱۵۵/۴	۰/۱۲۸	۰/۴۵۸	۱۰۲۴/۵	۰/۰۰۰	۰/۲۴۰
	R103	۹۴۴/۷	۰/۰۰۰	۰/۲۲۰	۱۲۴۲/۸	۰/۳۱۵	۰/۱۳۴	۱۰۸۸/۷	۰/۱۵۲	۰/۳۵۶	۹۶۶/۲	۰/۰۲۳	۰/۲۲۴
	R104	۹۷۲/۳	۰/۰۰۶	۰/۲۳۵	۱۲۸۷/۷	۰/۳۳۳	۰/۲۴۶	۱۰۵۷/۵	۰/۰۹۴	۰/۳۶۰	۹۶۵/۰	۰/۰۰۰	۰/۲۰۶
	R105	۱۰۵۸/۳	۰/۰۱۴	۰/۲۱۷	۱۲۹۵/۰	۰/۲۴۱	۰/۱۰۶	۱۱۳۷/۷	۰/۰۹۰	۰/۶۵۴	۱۰۴۳/۶	۰/۰۰۰	۰/۲۶۷
	R106	۸۷۲/۶	۰/۰۰۰	۰/۱۹۰	۱۱۷۸/۵	۰/۳۴۵	۰/۰۹۴	۹۳۴/۷	۰/۰۶۷	۰/۲۶۱	۹۴۲/۴	۰/۰۷۶	۰/۱۷۳
	R107	۱۰۹۲/۵	۰/۰۰۰	۰/۲۱۹	۱۲۸۷/۰	۰/۱۷۸	۰/۱۰۷	۱۱۹۵/۵	۰/۰۹۴	۰/۴۴۰	۱۱۱۴/۳	۰/۰۲۰	۰/۱۸۲
	R108	۱۰۲۶/۴	۰/۰۱۹	۰/۱۹۴	۱۱۸۴/۰	۰/۱۷۶	۰/۰۸۷	۱۰۰۶/۸	۰/۰۰۰	۰/۲۷۳	۱۰۶۹/۳	۰/۰۶۲	۰/۱۷۵
	R109	۹۶۸/۲	۰/۰۰۰	۰/۲۱۸	۱۲۷۹/۰	۰/۳۲۱	۰/۱۱۸	۱۲۱۳/۳	۰/۲۵۳	۰/۵۱۶	۱۰۶۲/۸	۰/۰۹۸	۰/۲۳۷
	R110	۸۳۹/۴	۰/۰۰۰	۰/۲۰۵	۱۱۶۹/۲	۰/۳۹۳	۰/۰۹۵	۱۰۰۵/۳	۰/۱۹۸	۰/۲۵۳	۱۰۲۳/۰	۰/۲۱۹	۰/۱۶۵
	R111	۸۳۵/۳	۰/۰۰۰	۰/۲۰۴	۱۱۳۹/۴	۰/۳۶۴	۰/۱۰۲	۹۶۳/۲	۰/۱۵۳	۰/۲۵۲	۱۰۱۰/۵	۰/۲۱۰	۰/۱۸۰
	R112	۱۰۱۳/۸	۰/۰۰۰	۰/۲۹۳	۱۱۷۲/۲	۰/۱۵۶	۰/۱۳۰	۱۰۲۴/۰	۰/۰۱۰	۰/۲۹۵	۱۰۳۸/۰	۰/۰۲۴	۰/۱۶۰
	میانگین	–	۰/۰۰۷	۰/۲۱۹	–	۰/۲۷۲	۰/۱۱۹	–	۰/۱۱۲	۰/۳۸۰	–	۰/۰۶۱	۰/۱۹۹
#RC	RC101	۱۰۸۳/۵	۰/۰۰۰	۰/۲۰۷	۱۶۱۱/۸	۰/۴۸۸	۰/۰۹۰	۱۴۱۹/۹	۰/۳۱۰	۰/۳۵۳	۱۲۴۹/۲	۰/۱۵۳	۰/۱۵۲
	RC102	۱۰۵۲/۹	۰/۰۵۳	۰/۲۲۱	۱۴۰۴/۸	۰/۴۰۵	۰/۱۲۸	۱۱۵۱/۵	۰/۱۵۲	۰/۳۱۲	۹۹۹/۷	۰/۰۰۰	۰/۱۵۳
	RC103	۹۹۳/۷	۰/۰۰۰	۰/۲۳۸	۱۶۷۵/۳	۰/۶۸۶	۰/۱۰۲	۱۱۶۱/۱	۰/۱۶۸	۰/۲۸۷	۱۱۳۳/۶	۰/۱۴۱	۰/۱۴۹
	RC104	۱۰۸۳/۵	۰/۰۸۱	۰/۲۱۵	۱۴۰۴/۴	۰/۴۰۱	۰/۱۱۳	۱۱۵۷/۵	۰/۱۵۴	۰/۳۱۷	۱۰۰۲/۷	۰/۰۰۰	۰/۱۴۵
	RC105	۱۱۲۳/۴	۰/۰۰۰	۰/۲۸۵	۱۶۰۴/۴	۰/۴۴۲	۰/۰۹۹	۱۴۵۶/۸	۰/۳۱۰	۰/۳۴۵	۱۲۵۷/۹	۰/۱۳۱	۰/۱۴۸
	RC106	۹۸۹/۹	۰/۰۳۲	۰/۲۱۱	۱۴۷۷/۷	۰/۵۴۰	۰/۱۰۵	۱۱۸۳/۵	۰/۲۳۴	۰/۲۶۳	۹۵۹/۳	۰/۰۰۰	۰/۱۴۵
	RC107	۱۰۹۲/۷	۰/۰۰۰	۰/۲۷۰	۱۵۹۸/۰	۰/۴۶۲	۰/۰۹۷	۱۴۵۶/۸	۰/۳۳۳	۰/۳۴۴	۱۲۵۷/۸	۰/۱۵۱	۰/۱۴۶
	RC108	۱۱۱۴/۱	۰/۰۰۰	۰/۴۰۸	۱۵۶۸/۵	۰/۴۰۸	۰/۰۹۴	۱۴۱۹/۵	۰/۲۷۴	۰/۳۴۸	۱۲۵۱/۰	۰/۱۲۳	۰/۱۵۰
	میانگین	–	۰/۰۲۱	۰/۲۵۷	–	۰/۴۷۹	۰/۱۰۳	–	۰/۲۴۲	۰/۳۲۱	–	۰/۰۸۷	۰/۱۴۸

یک روش ابتکاری دوفازی است. در فاز اول توسط الگوریتم‌های تشریح شده یک جواب اولیه ایجاد می‌شود و در فاز دوم، جواب ایجاد شده توسط یک الگوریتم ابتکاری دیگر، بهبود می‌یابد. کارایی این الگوریتم با توجه به این که جواب اولیه توسط چه الگوریتمی ایجاد شده است، متفاوت خواهد بود. در این قسمت، کارایی این الگوریتم با توجه به الگوریتم‌های اولیه متفاوت بررسی می‌شود. بدین منظور، میانگین جواب مسائل هر دسته برای جواب‌های اولیه و همچنین میانگین جواب‌های به دست آمده پس از به کارگیری الگوریتم ابتکاری بهبود جواب، با یکدیگر مقایسه شده است. نتایج حاصل در شکل‌های ۲، ۳ و ۴ مشخص شده است.

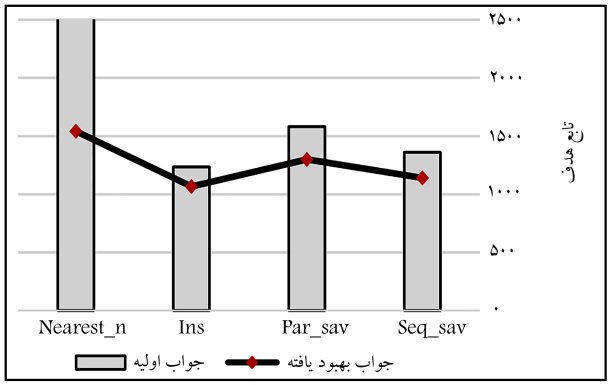
مشخص است که الگوریتم ابتکاری بهبود جواب، در تمام دسته مسائل تأثیر قابل توجهی در بهبود جواب‌های اولیه ایجاد شده داشته است. عملکرد این الگوریتم برای جواب‌های اولیه‌یی که توسط الگوریتم نزدیک‌ترین مشتری دیدار نشده ایجاد شده بودند، نسبت به سایر الگوریتم‌ها بیشتر است. از طرفی این تأثیر برای الگوریتم



شکل ۲. بررسی تأثیر الگوریتم بهبود جواب در دسته مسائل C.



شکل ۳. بررسی تأثیر الگوریتم بهبود جواب در دسته مسائل R.



شکل ۴. بررسی تأثیر الگوریتم بهبود جواب در دسته مسائل R.

اولیه مبتنی بر درج کردن، مقدار زیادی نیست. دلیل این امر را می‌توان به استفاده الگوریتم ابتکاری بهبود، از رویکردی مشابه با الگوریتم ابتکاری درج کردن نسبت داد. بنابراین در مرحله‌ی بهبود جواب، تأثیر بهبود نسبت به سایر الگوریتم‌ها کم‌تر است.

۶. نتیجه‌گیری

مسئله‌ی مسیریابی وسایل نقلیه یکی از مسائل قدیمی بهینه‌سازی ترکیبی است. اگرچه زمان زیادی از طرح این مسئله گذشته است، همچنان یکی از مسائل پرکاربرد و پراهمیت است. در این مقاله گونه‌یی خاص از مسئله‌ی مسیریابی وسایل نقلیه مورد بررسی قرار گرفت. این مسئله با عنوان مسئله‌ی مسیریابی با در نظر گرفتن مکان ـ زمان‌های پیشنهادی مشتریان، مطرح شده است. در این مقاله دو مدل ریاضی خطی برای مسئله ارائه شد؛ مدل اول به صورت سه اندیسه‌ی مبتنی بر گره و مدل دوم با رویکرد مبتنی بر جریان و به صورت دو اندیسه ارائه شد. تابع هدف این مدل‌ها به صورت کمینه‌سازی مدت زمان پیمایش مسیرها در نظر گرفته شده است. این تابع هدف در مسائل پنجره‌ی زمانی سخت، تابع هدفی جامع بوده و علاوه بر مسافت پیموده شده، مدت زمان انتظار را نیز دربر می‌گیرد. نتایج حاصل نشان می‌دهد که مدل مبتنی بر جریان در حدود ۲۲ مرتبه سریع‌تر از مدل مبتنی بر گره بوده و در ارائه‌ی حدود پایین عملکرد مناسبی داشته است.

در قسمت بعد، با توجه به این نکته که مسئله‌ی مورد بررسی یک مسئله‌ی NP-Hard است، با رویکردهای متفاوت، چهار الگوریتم ابتکاری برای مسئله ارائه شد. این الگوریتم‌ها شامل الگوریتم نزدیک‌ترین مشتری دیدار نشده، الگوریتم درج کردن و الگوریتم‌های ابتکاری با محوریت صرفه‌جویی ـ شامل دو الگوریتم در حالت سری و موازی ـ بودند. پس از تولید جواب اولیه توسط این الگوریتم‌ها، از یک الگوریتم ابتکاری به منظور تقویت و بهبود جواب استفاده شد. در نهایت عملکرد الگوریتم‌های طراحی شده بر روی سه دسته داده که شاکله‌ی ساختاری متفاوتی داشتند، آزمایش شدند. نتایج نشان می‌دهد که الگوریتم درج کردن عملکردی خوب در زمان مناسبی ارائه می‌دهد و می‌توان از آن به عنوان یک جواب اولیه برای الگوریتم‌های فرابتکاری بهره برد.

۷. پیشنهادها

در راستای توسعه‌ی موضوع، موارد متعددی را می‌توان پیشنهاد کرد. با بررسی مقاله‌های GVRP، می‌توان به این نتیجه رسید که توجه زیادی به توابع هدف نشده است. به عنوان مثال، در نظر گرفتن بالانس وسائط نقلیه از نقطه نظرهای متفاوتی چون بار یا مسافت طی شده می‌تواند ایده‌یی باشد برای در نظر گرفتن تابع هدف مسئله. پیشنهاد دیگر برای تحقیقات بعدی، لحاظ کردن ارتباط راننده‌ها با مشتریان است؛ بدین معنی که اگر مشتری توسط یک راننده‌ی خاص سرویس‌دهی شود، رضایت بالاتری به دست می‌آورد.

علاوه بر این موارد، ایده‌ی جالب دیگری که می‌تواند به عنوان یک ایده کاربردی مطرح شود، اولویت‌بندی نقاط هر خوشه است؛ به عنوان مثال رضایت مشتری برای تحویل کالا، در نقاط مختلف، در نظر گرفته شود و مسئله به صورت یک مسئله‌ی چندهدفه توسعه یابد. نکته دیگر، در نظر گرفتن چند تابع هدف برای مسئله‌ی GVRP است. این مسئله با توجه به وجود گزینه‌های مختلف و ماهیت چندگزینه‌یی داشتن، پتانسیل زیادی برای توسعه در جهت مسائل چندهدفه دارد، اما علی‌رغم این پتانسیل توجه چندانی به مسائل چندهدفه نشده است. علاوه بر موارد یاد شده، با توجه به این که یکی از کاربردهای مسئله‌ی مطرح شده می‌تواند در مسائل امدادرسانی در شرایط بحران مطرح باشد، می‌توان حالت‌های مختلفی از عدم قطعیت را برای توسعه مسائل GVRP پیشنهاد کرد. در این راستا، در نظر گرفتن مشتریان احتمالی و احتمال خرابی وسایل نقلیه و مواردی همچون کمبود اطلاعات و وجود داده‌های فازی در پارامترهای مسئله، ارائه مدل رباست مسئله و در نظر گرفتن اختلال در مسئله، می‌تواند پیشنهاداتی برای مطالعات آتی باشد.

پانوشت‌ها

1. Vehicle Routing Problem
2. Generalized Vehicle Routing Problem
3. Vehicle Routing Problem with Roaming Delivery Locations
4. Service Point
5. Mixed Integer Linear Programming
6. Generalized Vehicle Routing Problem with Time Windows
7. Vehicle Routing Problem with Time Windows

منابع (References)

1. Dantzig, G.B. and Ramser, J.H. "The truck dispatching problem", *Management Science*, **6**, pp. 80-91 (1959).
2. Toth, P. and Vigo, D. *Vehicle routing: Problems, methods, and applications*, SIAM (2014).
3. Ghiani, G. and Improta, G. "An efficient transformation of the generalized vehicle routing problem", *European Journal of Operational Research*, **122**, pp. 11-17 (2000).
4. Pop, P.C., Kara, I. and Marc, A.H. "New mathematical models of the generalized vehicle routing problem and extensions", *Applied Mathematical Modelling*, **36**, pp. 97-107 (2012).
5. Reyes, D., Savelsbergh, M. and Toriello, A. "Vehicle routing with roaming delivery locations", *Transportation Research Part C: Emerging Technologies*, **80**, pp. 71-91 (2017).
6. Kara, I. and Bektas, T. "Integer linear programming formulation of the generalized vehicle routing problem", *Proceeding of EURO/INFORMS Joint International Meeting*, Istanbul, July pp. 06-10 (2003).
7. Pop, P.C., Matei, O., Sitar, C.P. and et al. "A genetic algorithm for solving the generalized vehicle routing problem", *Proceeding of International Conference on Hybrid Artificial Intelligence Systems*, Springer, pp. 119-126 (2010).
8. Bektaş, T., Erdoğan, G. and Røopke, S. "Formulations and branch-and-cut algorithms for the generalized vehicle routing problem", *Transportation Science*, **45**, pp. 299-316 (2011).
9. Pop, P. and Pop-Sitar, C. "A new efficient transformation of the generalized vehicle routing problem into the classical vehicle routing problem", *Yugoslav Journal of Operations Research*, **21**, pp. 187-198 (2011).
10. Moumou, M., Rhofir, K. and Allaoui, R. "Multi-split delivery for the vehicle routing problem with clustering: Mathematical formulation and algorithm", *Proceeding of 2019 International Colloquium on Logistics and Supply Chain Management (LOGISTIQUA)*, IEEE, pp. 1-4 (2019).
11. Los, J., Spaan, M.T. and Negenborn, R.R. "Fleet Management for Pickup and Delivery Problems with Multiple Locations and Preferences", *Proceeding of International Conference on Dynamics in Logistics*, Springer, pp. 86-94 (2018).
12. Zhou, L., Baldacci, R., Vigo, D. and et al. "A multi-depot two-echelon vehicle routing problem with delivery options arising in the last mile distribution", *European Journal of Operational Research*, **265**, pp. 765-778 (2018).
13. Biesinger, B., Hu, B. and Raidl, G.R. "A variable neighborhood search for the generalized vehicle routing problem with stochastic demands", *Proceeding of European Conference on Evolutionary Computation in Combinatorial Optimization*, Springer, pp. 48-60 (2015).
14. He, Y., Qi, M., Zhou, F. and et al. "An effective metaheuristic for the last mile delivery with roaming delivery locations and stochastic travel times", *Computers & Industrial Engineering*, **Vol**, p. 106513 (2020).
15. Hà, M.H., Bostel, N., Langevin, A. and et al. "An exact algorithm and a metaheuristic for the generalized vehicle routing problem with flexible fleet size", *Computers & Operations Research*, **43**, pp. 9-19 (2014).
16. Moccia, L., Cordeau, J.-F. and Laporte, G. "An incremental tabu search heuristic for the generalized vehicle routing problem with time windows", *Journal of the Operational Research Society*, **63**, pp. 232-244 (2012).
17. Ozbaygin, G. and Savelsbergh, M. "An iterative re-optimization framework for the dynamic vehicle routing problem with roaming delivery locations", *Transportation Research Part B: Methodological*, **128**, pp. 207-235 (2019).
18. Orenstein, I., Raviv, T. and Sadan, E. "Flexible parcel delivery to automated parcel lockers: Models, solution methods and analysis", *EURO Journal on Transportation and Logistics*, **8**, pp. 683-711 (2019).
19. Dumez, D., Lehuédé, F. and Péton, O. "A large neighborhood search approach to the vehicle routing problem with delivery options", *Transportation Research Part B: Methodological*, **144**, pp. 103-132 (2021).
20. Lenstra, J.K. and Kan, A.R. "Complexity of vehicle routing and scheduling problems", *Networks*, **11**, pp. 221-227 (1981).
21. Solomon, M.M. "Algorithms for the vehicle routing and scheduling problems with time window constraints", *Operations Research*, **35**, pp. 254-265 (1987).
22. Clarke, G. and Wright, J.W.J.O.r. "Scheduling of vehicles from a central depot to a number of delivery points", **12**, pp. 568-581 (1964).