

## مسئله مسیریابی وسیله نقلیه چندانباری ظرفیت‌دار با در نظر گرفتن مسیر بین انبارها

مصطفی ستاک\*<sup>۱</sup>، سهیل جلیلی بوالحسنی<sup>۲</sup>، حسین کریمی<sup>۳</sup> و بهارک قربانی<sup>۴</sup>

<sup>۱</sup> استادیار رشته مهندسی صنایع- دانشکده مهندسی صنایع- دانشگاه صنعتی خواجه نصیرالدین طوسی  
<sup>۲</sup> دانش‌آموخته فوق لیسانس مهندسی صنایع- دانشکده مهندسی صنایع- دانشگاه صنعتی خواجه نصیرالدین طوسی  
<sup>۳</sup> دانشجوی دکتری مهندسی صنایع- دانشکده مهندسی صنایع- دانشگاه صنعتی خواجه نصیرالدین طوسی  
<sup>۴</sup> دانشجوی فوق لیسانس مهندسی صنایع- دانشکده مهندسی صنایع- دانشگاه صنعتی خواجه نصیرالدین طوسی

### چکیده

در این مقاله، مسئله مسیریابی وسیله نقلیه چندانباری، با در نظر گرفتن مسیر بین انبارها بررسی می‌شود که در آن وسایل نقلیه می‌توانند در دپوهای میانی، بارگیری مجدد انجام دهند. وسایل نقلیه با بار کامل، از دپوی مبدأ شروع به حرکت می‌کنند و مشتریان را تا پایان بار سرویس می‌دهند. آنها سپس می‌توانند برای بارگیری مجدد به دپوی میانی عزیمت کنند و سرانجام برای اتمام مسیر خود به دپوی مبدأ باز گردند. برای این مسئله، یک مدل ریاضی برنامه‌ریزی عدد صحیح مختلط معرفی می‌شود. هدف مسئله، یافتن مسیر برای وسایل نقلیه به گونه‌ای است که بدون نقض کردن محدودیت ظرفیت وسایل نقلیه، هزینه کل سفر و هزینه بارگیری‌های مجدد در دپوهای میانی کمینه شود. مسئله حاضر توسط حل‌کننده سیپلکس در نرم‌افزار گمز ۲۳،۵ و رویکردهای الگوریتم ژنتیک و جستجوی ممنوع حل می‌شود. نتایج محاسباتی به دست آمده، کارایی الگوریتم‌های پیشنهاد شده را از نظر زمان حل و کیفیت جواب نشان می‌دهند.

**واژه‌های کلیدی:** مسئله مسیریابی وسیله نقلیه چندانباری، مسیر بین انبارها، دپوی میانی، بارگیری مجدد، الگوریتم

ژنتیک، جستجوی ممنوع

### مقدمه

خود را قبل از بازگشت به دپوی مبدأ داشته باشد. این مقوله، نه تنها هزینه‌های سرمایه‌گذاری روی ناوگان حمل‌ونقل را کاهش می‌دهد، بلکه منجر به صرفه‌جویی در مصرف سوخت و آلودگی هوای کمتر می‌شود.

با توجه به اطلاعات ما، این مقاله اولین مطالعه‌ای است که یک مدل برنامه‌ریزی عدد صحیح مختلط مبتنی بر گره برای MDVRPIDR با معرفی نوع جدیدی از محدودیت‌های حذف زیرتور ارائه می‌دهد.

### مرور ادبیات

مسئله مسیریابی وسیله نقلیه چندانباری، اولین بار توسط سامیچراست و مارخام [۱] فرمول‌بندی شد. هر چند محققان دیگری نظیر تیلمن و کین [۲] و جیلت و جانسون [۳] قبل از آنها این مسئله را مطرح کرده و برای آن، روش‌های حل ابتکاری ارائه کرده بودند، بدون آنکه مدل ریاضی برای مسئله پیشنهاد کنند. گیوسا و همکاران [۴] مسئله مسیریابی چندانباری با پنجره زمانی را بررسی

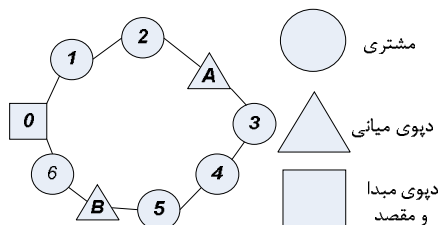
مسئله مسیریابی وسیله نقلیه چندانباری<sup>۱</sup> در دنیای واقعی، کاربردهای فراوانی دارد، زیرا اغلب در زنجیره‌های تأمین و یا شهرهای بزرگ برای ذخیره و توزیع کالاها از بیش از یک انبار استفاده می‌شود. هر مسئله مسیریابی وسیله نقلیه چندانباری را می‌توان به سه قسمت تجزیه کرد: قسمت اول مشخص می‌کند که هر مشتری توسط کدام انبار سرویس داده شود. قسمت دوم، تعیین مسیر سرویس‌دهی یا مسیریابی است. قسمت آخر نیز تعیین اولویت در سرویس‌دهی است که تقدم و تأخر ملاقات مشتریان در هر مسیر را مشخص می‌کند.

مسئله مسیریابی وسیله نقلیه چندانباری، با در نظر گرفتن مسیر بین انبارها<sup>۲</sup> مقوله یک شرکت حمل‌ونقل است که قصد استفاده از وسایل نقلیه کمتری (در حالت بهینه یک وسیله نقلیه) دارد که قادر به سفرهای طولانی‌مدت برای خدمت‌رسانی به همه مشتریان باشند. این ویژگی مسئله، وقتی اتفاق می‌افتد که وسیله نقلیه، اجازه بارگیری مجدد در دپوهای میانی در طول مسیر

حالت اول، به بررسی مسئله مسیریابی ظرفیت‌دار با در نظر گرفتن محدودیت‌های ظرفیت وسایل حمل و حداکثر طول مجاز هر مسیر مربوط می‌شود، اما در مسئله دوم علاوه بر محدودیت‌های موجود در حالت اول به واسطه قابلیت‌های سیستم‌های هوشمند حمل‌ونقل، هر وسیله نقلیه مجاز به بارگیری مجدد در دپوهای دیگر بوده و دپوی پایانی در هر مسیر، می‌تواند متفاوت از دپوی اولیه باشد.

### شرح و مدل‌سازی مسئله

مسئله مسیریابی وسیله نقلیه چندانباری با در نظر گرفتن مسیر بین انبارها حالت خاصی از MDVRP است که در آن وسیله نقلیه با بار پر، دپوی مبدأ را ترک می‌کند و در طول یک دوره زمانی، کالا را به مشتریان تحویل می‌دهد. به محض اتمام بار، وسیله نقلیه برای بارگیری مجدد، یکی از دپوها را ملاقات می‌کند. این فرآیند تا سرویس‌دهی همه مشتریان ادامه می‌یابد. سپس وسیله نقلیه به دپوی مبدأ باز می‌گردد. شکل (۱) مثالی از MDVRPIDR با سه دپو و شش مشتری را نشان می‌دهد. مسیر وسیله نقلیه در این مثال، به صورت  $B \ 6 \ 0 \ 1 \ 2 \ A \ 3 \ 4 \ 5 \ B$  اولین زیرمسیر از دپوی ۰ شروع می‌شود، به مشتریان ۱ و ۲ می‌رود و سپس برای بارگیری مجدد، به دپوی A می‌رود. پس از بارگیری، وسیله نقلیه به مشتریان ۳ و ۴ می‌رود و سپس برای بارگیری مجدد به دپوی B می‌رود. مشابه همین روند، زیرمسیر سوم به مشتری ۶ ادامه پیدا می‌کند و سپس وسیله نقلیه برای اتمام مسیر خود به دپوی ۰ باز می‌گردد.



شکل ۱: مثالی از MDVRPIDR با سه دپو و شش مشتری

MDVRPIDR به صورت زیر فرمول‌بندی می‌شود. فرض کنید  $G = (V, A)$  یک گراف جهت‌داری باشد که در

کردند. همچنین نجی و سالهی [۵] مسئله مسیریابی چندانباری با بارگیری و تخلیه همزمان را تحلیل کردند. محققان زیادی سعی در ارائه روش‌های حل ابتکاری و کاراً برای این دسته از مسائل داشته‌اند، طوری که بیش از ۸۵ درصد روش‌های حل ارائه‌شده برای مسئله MDVRP به صورت ابتکاری یا فراابتکاری است. تانگیا و سالهی [۶]، هو و همکاران [۷] و دوندو و سردا [۸] روش‌های ابتکاری مبتنی بر الگوریتم ژنتیک برای حل مسئله مزبور را توسعه دادند. در مطالعه نجی و سالهی [۵] نیز MDVRP با مراحل خدمت‌دهی، جمع‌آوری و تحویل ارائه شد و حل این مسئله با استفاده از یک الگوریتم ابتکاری بر مبنای برخی رویکردهای جستجوی همسایگی مانند 2-opt و معاوضه بررسی شد.

کرویر و همکاران [۹] یک فرمول‌بندی مبتنی بر مسیر برای حالتی از مسئله مسیریابی وسیله نقلیه چندانباری معرفی کردند که در آن وسایل نقلیه همگن در دپوها مستقر هستند. وسایل نقلیه، دپوی مرکزی خود را ترک می‌کنند و تقاضای مجموعه‌ای از مشتریان را برآورد می‌کنند. سپس برای بارگیری مجدد یا به دپوی مرکزی خود برمی‌گردند یا یک دپوی میانی را ملاقات می‌کنند. آنها سپس این مسئله را توسط یک الگوریتم ابتکاری الحاق‌کننده جستجوی ممنوع و برنامه‌ریزی عدد صحیح حل کردند. جوردن و بارنز [۱۰] و جوردن [۱۱] حالت ساده‌تری از مسئله را بررسی می‌کنند که در آن، تقاضای مشتریان، همگی مساوی بوده و مسیرهای بین دو دپو به صورت رفت و برگشتی هستند. آنها این مسئله را توسط الگوریتم حریصانه حل کردند. تارانلیس و همکاران [۱۲] یک نام جایگزین برای مسئله معرفی‌شده توسط کرویر و همکاران [۹] به نام مسئله مسیریابی وسیله نقلیه با تسهیلات میانی بارگیری مجدد، پیشنهاد کردند تا هر دو مقوله، نقش بارگیری مجدد تسهیلات میانی و استفاده از یک دپوی مرکزی برای وسایل نقلیه را پررنگ‌تر کنند. آنها همچنین یک روش جستجوی همسایگی ترکیبی برای این مسئله ارائه کرده و آن را با رویکرد سه‌مرحله‌ای حل کردند. ایده آزادی در ملاقات دپوی ابتدایی و پایانی با عنوان "تخصیص انعطاف‌پذیر" توسط کک و همکاران [۱۳] مطرح شد. بر اساس این تحقیق، دو حالت جدید از مسئله مسیریابی ظرفیت‌دار مورد مطالعه قرار گرفت.

$$\sum_{i \in \{v_0, V_d\}} \sum_{j \in V_c} x_{ij} - \sum_{i \in \{V_d, v_{n+d+1}\}} \sum_{j \in V_c} x_{ji} = 0 \quad (7)$$

$$u_i \geq q_i \quad \forall i \in V_c \quad (8)$$

$$u_i \leq Q \quad \forall i \in V_c \quad (9)$$

$$u_i \leq Q + (q_i - Q)x_{ji} \quad \forall i \in V_c, \quad \forall j \in \{v_0, V_d\} \quad (10)$$

$$u_j \geq u_i + q_j - Q + Qx_{ij} + (Q - q_j - q_i)x_{ji} \quad \forall i \in V_c, \quad \forall j \in V_c \quad (11)$$

$$x_{v_{n+d+1}, j} = 0 \quad \forall j \in V \quad (12)$$

$$x_{i, v_0} = 0 \quad \forall i \in V_c \quad (13)$$

$$u_i = 0 \quad \forall i \in V \setminus V_c \quad (14)$$

$$x_{ij} \in \{0, 1\} \quad \forall i, j \in V, i \neq j \quad (15)$$

$$z_d = 0, 1, 2, \dots \quad \forall d \in V_d \quad (16)$$

$$\forall i \in V_c \quad u_i \geq 0 \quad (17)$$

رابطه (۱) تابع هدف مدل به معنی کمینه کردن هزینه خدمت‌دهی کلی مشتریان است که عبارت است از:

$$1. \text{ هزینه سفر } \sum_{i \in V} \sum_{\substack{j \in V \\ j \neq i}} x_{ij} t_{ij} c_{ij}$$

$$2. \text{ هزینه تعداد بارگیری‌های مجدد } \sum_{d \in V_d} CID z_d$$

معادله (۲) ایجاب می‌کند که هر مشتری فقط یک خروجی داشته باشد. محدودیت‌های (۳) و (۴) به ترتیب موجب می‌شوند وسیله نقلیه فقط یک بار دپوی مبدأ و دپوی مقصد را ملاقات کند. محدودیت‌های (۵) باعث می‌شوند وسیله نقلیه از هر دپوی میانی به تعداد دفعاتی که آن دپو برای بارگیری مجدد استفاده می‌شود، عبور کند. محدودیت‌های (۶) بیانگر حفظ جریان در گره‌ها هستند. محدودیت‌های (۷) ایجاب می‌کنند که حفظ جریان در دپوها از طریق وادار کردن وسیله نقلیه به بازگشت به دپوی مبدأ باشد. محدودیت‌های (۸) تا (۱۱) محدودیت‌های حذف زیرتور و ظرفیت هستند. مقدار  $u_i$  باید حداقل به اندازه تقاضای مشتری  $i$  (۸) و کمتر از ظرفیت وسیله نقلیه (۹) باشد. اگر مشتری  $i$  اولین مشتری یک تور باشد،  $u_i$  مساوی تقاضای این مشتری است. این مطلب از طریق سه محدودیت (۸)، (۹) و (۱۰)

آن  $V = \{v_0, v_1, \dots, v_{n+d+1}\}$  مجموعه رئوس گراف است که  $v_0$  و  $v_{n+d+1}$  به ترتیب بیانگر دپوی مبدأ و دپوی مقصد هستند (مکان آن‌ها یکی است)،  $V_c = \{v_1, \dots, v_n\}$  مجموعه مشتریان است،  $V_d = \{v_{n+1}, v_{n+2}, \dots, v_{n+d}\}$  مجموعه  $d$  دپوی میانی است و  $A = \{(v_i, v_j) : v_i, v_j \in V, i \neq j\}$  مجموعه یال‌های گراف است. هر یال  $(i, j) \in V$  یک هزینه سفر به ازای هر واحد زمانی  $c_{ij}$  و زمان سفر  $t_{ij}$  دارد که نامعادله مثلثی قوی در آن برقرار است ( $t_{ij} + t_{jk} > t_{ik}$ ). دپوی مبدأ و همه دپوهای میانی، مقدار عرضه نامحدود کالا دارند. هر دپوی میانی  $d \in V_d$  هزینه بارگیری مجدد  $CID$  دارد. یک وسیله نقلیه با ظرفیت  $Q$  مستقر در دپوی مبدأ موجود است. هر مشتری  $i \in V_c$  تقاضای معلوم  $q_i$  دارد. در زیر، متغیرهای مورد استفاده در فرمول‌بندی بیان می‌شوند:

مقدار تقاضای تجمعی برآورد شده توسط وسیله نقلیه پس از ملاقات گره  $i \in V$

$$x_{ij} = \begin{cases} 1 & \text{اگر وسیله نقلیه از گره } i \text{ به گره } j \text{ برود،} \\ 0 & \text{در غیر این صورت،} \end{cases}$$

تعداد دفعاتی که دپوی میانی  $d$  برای بارگیری مجدد به کار می‌رود:

$$\text{با این فرضیات، مدل ریاضی به صورت زیر است:} \\ \text{Min} \sum_{i \in V} \sum_{\substack{j \in V \\ j \neq i}} x_{ij} t_{ij} c_{ij} + \sum_{d \in V_d} CID z_d \quad (1)$$

مقید به محدودیت‌های:

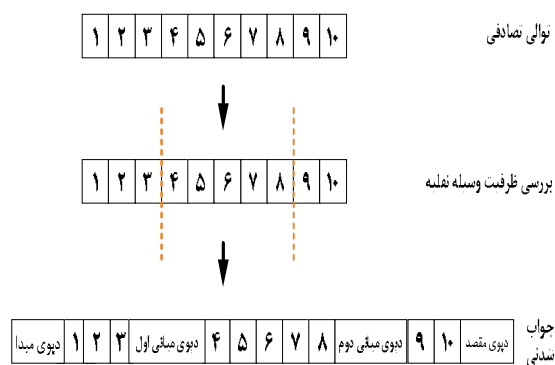
$$\sum_{\substack{j \in V \\ j \neq i}} x_{ij} = 1 \quad \forall i \in V_c \quad (2)$$

$$\sum_{j \in V_c} x_{v_0, j} = 1 \quad (3)$$

$$\sum_{i \in V_c} x_{i, v_{n+d+1}} = 1 \quad (4)$$

$$\sum_{j \in V_c} x_{dj} = z_d \quad \forall d \in V_d \quad (5)$$

$$\sum_{\substack{i \in V \\ i \neq j}} x_{ij} - \sum_{\substack{i \in V \\ i \neq j}} x_{ji} = 0 \quad \forall j \in V \quad (6)$$



شکل ۲: روند تولید جواب شذنی

### الگوریتم ژنتیک

الگوریتم ژنتیک، روش مبتنی بر جستجوی تصادفی فراگیر است و برگرفته از علم ژنتیک و نظریه تکامل داروین است. این الگوریتم، بر اساس بقای بهترین‌ها با انتخاب طبیعی استوار است. در یک الگوریتم ژنتیک یک جمعیت از افراد بر اساس میزان مطلوبیت در محیط باقی می‌مانند. افرادی با قابلیت‌های برتر، شانس ازدواج و تولید مثل بیشتری را خواهند یافت. بنابراین بعد از چند نسل، فرزندان با کارایی بهتر به وجود می‌آیند.

اولین گام در به‌کارگیری و پیاده‌سازی یک الگوریتم ژنتیک، نمایش جواب‌های مسئله به صورت یک کروموزوم است. کروموزوم مورد استفاده برای بررسی مسئله مورد بررسی، به صورت یک توالی از مشتریان در نظر گرفته می‌شود. شکل (۳) ساختار یک کروموزوم را برای مسئله نمونه با ۱۰ مشتری نشان می‌دهد.

۱	۵	۳	۹	۷	۶	۲	۸	۴	۱۰
---	---	---	---	---	---	---	---	---	----

شکل ۳: ساختار یک کروموزوم برای الگوریتم ژنتیک

#### پیشنهادی

مرحله بعد در الگوریتم ژنتیک، عمل تقاطع است. با اجرای عملگر تقاطع به شکل تصادفی، بخش‌هایی از کروموزوم‌ها با یکدیگر تعویض می‌شوند. این مسئله سبب می‌شود که فرزندان، ترکیبی از خصوصیات والدین خود را به همراه داشته باشند و به طور دقیق مشابه یکی از والدین نباشند. پیش از اجرای عملگر تقاطع، لازم است تا دو کروموزوم از کروموزوم‌های جمعیت کنونی به عنوان والدین انتخاب شوند. برای انتخاب والدین، از فرآیند انتخاب چرخ رولت استفاده می‌شود. عملگر تقاطع مورد

بیان می‌شود. فرض کنید که  $i$  مشتری اول تور نیست. آنگاه  $u_i$  باید برابر با جمع تقاضاهای برآورد شده بین دپو (میانی یا مبدأ) و مشتری  $i$  باشد، یعنی اگر مشتری  $j$  بلافاصله پس از مشتری  $i$  سرویس‌دهی شود،  $u_j$  باید مساوی تقاضای برآورد شده روی تور از دپو به  $i$  به اضافه مقدار سفارش داده شده توسط  $j$  باشد. این موضوع در محدودیت (۱۱) نهفته است. حتی وقتی  $j$  بلافاصله پس از  $i$  نباشد، محدودیت (۱۱) معتبر است. محدودیت‌های (۱۲) و (۱۳) به ترتیب جریان کالای وسیله نقلیه را در دپوی مقصد و مبدأ ختم و شروع می‌کنند. محدودیت‌های (۱۴) متغیر  $u_i$  را در دپوها برابر صفر قرار می‌دهند. محدودیت‌های (۱۵)، (۱۶) و (۱۷) نشان می‌دهند که  $x_{ij}$  متغیر صفر و یک،  $z_d$  متغیر عدد صحیح و  $u_i$  متغیر غیر صفر هستند.

### الگوریتم‌های حل فرا ابتکاری

از آنجا که مسئله مسیریابی وسیله نقلیه، یک مسئله NP-hard است، با افزایش ابعاد مسئله زمان حل آن به شکل نمایی افزایش می‌یابد. برای کاهش زمان محاسباتی در مسائلی با ابعاد بزرگ‌تر، دو رویکرد فراابتکاری مبتنی بر الگوریتم‌های ژنتیک<sup>۳</sup> و جستجوی ممنوع<sup>۴</sup> برای حل این مسئله پیشنهاد می‌شود. ابتدا باید ساختاری به منظور دستیابی به جواب‌های اولیه شذنی ایجاد کرد. بدین منظور، ابتدا یک توالی از مشتریان تولید می‌شود. توالی مورد نظر، وارد فرآیندی می‌شود که طبق آن برای جلوگیری از تجاوز از حداکثر ظرفیت وسیله نقلیه، مشتریان دسته‌بندی می‌شوند. برای این کار، با شروع از مشتری اول در توالی، تقاضاها با هم جمع می‌شوند. این کار تا زمانی که جمع تقاضاها از ظرفیت وسیله نقلیه تجاوز نکند، ادامه می‌یابد. به محض اینکه جمع تقاضاها از ظرفیت وسیله نقلیه بیشتر شد، وسیله نقلیه برای بارگیری مجدد به یک دپوی میانی می‌رود. سپس این روند بار دیگر تکرار می‌شود و تا سرویس‌دهی به همه مشتریان ادامه می‌یابد. در نهایت یک جواب شذنی به صورت شکل (۲) نمایش داده می‌شود.

می‌گیرند که  $N$  تعداد مشتریان است. در الگوریتم پیشنهادی برای ایجاد پراکندگی در جواب و جستجوی بهتر همسایگی‌ها، دو استراتژی برای بررسی همسایگی در نظر گرفته شده است که در هر مرحله به طور تصادفی، یکی از آن‌ها مورد استفاده قرار می‌گیرد. این دو استراتژی عبارتند از استراتژی تعویض و معکوس‌سازی. در استراتژی تعویض فقط گره‌های انتخاب شده با یکدیگر جابه‌جا می‌شوند، ولی در استراتژی معکوس‌سازی همه گره‌های مابین نیز جابه‌جا می‌شوند. سرانجام الگوریتم پیشنهادی پس از تعداد معینی تکرار پایان می‌یابد.

### ارائه مسائل نمونه، حل آنها و نتایج عددی

در این بخش، ۷ مسئله با ۵ تا ۱۲ مشتری برای مسئله مورد بررسی طراحی می‌شود. نتایج حاصل از حل این نمونه مسائل با استفاده از حل‌کننده سیپلکس در نرم‌افزار گمز نسخه ۲۳،۵ و الگوریتم‌های ژنتیک و جستجوی ممنوع پیشنهادی که روی یک کامپیوتر با مشخصات CPU AMD Phenom II Triple-Core (2.1 GHz) و RAM 4G اجرا شده، در جداول (۱) و (۲) ارائه شده است.

جدول (۱) نشان می‌دهد که فاصله بهینگی خیلی کوچکی بین جواب‌های سیپلکس و GA وجود دارد، طوری که در خیلی از مسائل نمونه، این فاصله برابر صفر است. حتی در مسائل نمونه ۴ و ۵، فاصله موجود قابل توجه نیست. این به این دلیل است که هزینه‌های مسیریابی، نقش بسزایی در افزایش مقدار تابع هدف ندارند. به علاوه، همگرایی شدید جواب‌های GA به مقادیر بهینه به دلیل تعداد بارگیری‌های مجدد یکسان است. این مطلب دال بر این است که الگوریتم پیشنهاد شده، در برآورده کردن هدف مسئله از طریق کمینه‌کردن هزینه‌های مسیریابی و تعداد بهینه بارگیری مجدد، بسیار کارآ است، به طوری که در همه مسائل نمونه مشهود است.

با توجه به جدول (۲)، با همان مسائل نمونه جدول (۱)، فاصله بهینگی قابل توجهی بین جواب‌های سیپلکس و الگوریتم TS وجود ندارد. مشابه GA، همگرایی جواب‌های حاصل از الگوریتم TS به مقادیر بهینه به دلیل کمینه‌کردن هزینه‌های مسیریابی و تعداد بارگیری مجدد

استفاده در الگوریتم پیشنهادی، عملگر تقاطع تک‌نقطه‌ای است. در تقاطع تک‌نقطه‌ای، ابتدا نقطه‌ای تصادفی در کروموزوم انتخاب می‌شود، سپس توالی مشتریان در سمت چپ (راست) نقطه در والد برای فرزند متناظر آن حفظ شده و ترتیب مشتریان در سمت چپ (راست) نقطه بر اساس موقعیت آن‌ها در کروموزوم والد دیگر مرتب می‌شوند.

مرحله بعد، اجرای عمل جهش است. نحوه عملگر عملگر جهش، به این ترتیب است که یک یا چند ژن از کروموزوم را به طور تصادفی انتخاب می‌کند و سپس محتوای آن ژن‌ها را تغییر می‌دهد. عملگر جهش انواع مختلفی دارد. در این الگوریتم، در هر مرتبه از اجرای عملگر جهش به طور تصادفی، از یکی از سه عملگر جهش تعویض، جاگذاری و معکوس‌سازی استفاده شده است. این مسئله به جستجوی بهتر فضای جواب کمک می‌کند. با اجرای عملگر تعویض، مقادیر موجود در دو ژن انتخاب شده از کروموزوم با یکدیگر جابه‌جا می‌شود. عملگر جاگذاری نیز به این شکل است که مقادیر یکی از دو ژن انتخاب شده، به بعد از ژن انتخاب شده دیگر منتقل می‌شود و مقادیر ژن‌های بعدی به همان ترتیب اولیه قرار می‌گیرند. در عملگر معکوس‌سازی نیز همه مقادیر بین دو ژن انتخاب شده، به صورت عکس توالی اولیه خود قرار می‌گیرند. سرانجام الگوریتم ژنتیک پس از تعداد معینی تکرار پایان می‌یابد.

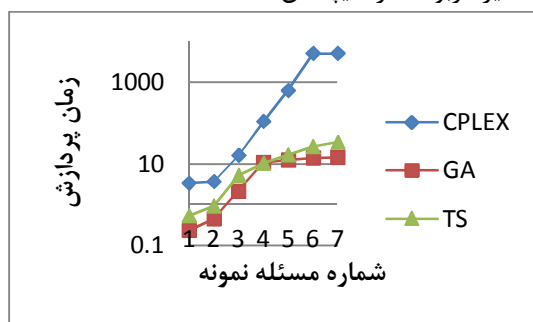
### الگوریتم جستجوی ممنوع

جستجوی ممنوع در سال ۱۹۸۶ توسط فرد گلوور ارائه شد. ایده اساسی یک الگوریتم TS این است که به طور تکراری از جواب جاری به جواب دیگری در همسایگی جواب فعلی حرکت کند. حرکت از جواب فعلی به دیگری، توسط تغییرات کوچکی در جواب فعلی با توجه به مجموعه قوانین موجود در ساختار همسایگی صورت می‌گیرد. رویکرد کلی اجتناب از تشکیل حلقه و گیر کردن الگوریتم در بهینه، موضعی است که از طریق ممنوع کردن یا جریمه کردن حرکت‌هایی که جواب را به فضای جوابی که قبلاً رفته شده (یعنی لیست ممنوع) می‌برند، انجام می‌گیرد.

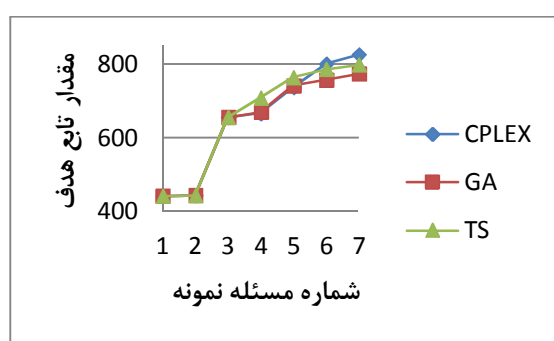
در هر مرحله از اجرای الگوریتم پیشنهادی، به تعداد مرتبه همسایگی‌ها مورد بررسی قرار

$$\frac{N(N-1)}{2}$$

مقادیر تابع هدف در GA و TS به طور طبیعی کمتر از مقادیر مربوطه در سیپلکس هستند.



شکل ۴: نمایش زمان‌های پردازش به دست آمده از حل دقیق و ابتکاری مسائل نمونه



شکل ۵: نمایش جواب‌های به دست آمده از حل دقیق و ابتکاری مسائل نمونه

یکسان در همه مسائل نمونه است. این دلالت بر این دارد که الگوریتم TS ارائه شده کارآ است.

با توجه به شکل (۴)، از نظر زمان پردازش، الگوریتم TS و GA در همه مسائل نمونه بهتر از رویکرد بهینگی عمل می‌کنند. حتی در مسائل نمونه بزرگ ۶ و ۷، زمان‌های پردازش در GA و TS کمتر از ۵۰ ثانیه است، در حالی که در سیپلکس، زمان حل این نمونه‌ها خیلی بالا است و به همین دلیل برای به دست آوردن یک حد بالا برای تابع هدف در این نمونه‌ها، فرآیند حل در ۵۰۰۰ ثانیه متوقف شده است. علاوه بر این، دیده می‌شود که با افزایش تعداد مشتریان، زمان حل در سیپلکس به صورت نمایی افزایش می‌یابد. این بدین معنی است که سیپلکس در عمل قادر به حل مسائل نمونه بزرگ نیست.

با توجه به شکل (۵)، GA در همگرایی به مقادیر بهینه در همه مسائل نمونه بهتر از TS عمل می‌کند. به علاوه، افزایش قابل توجهی در تابع هدف در حرکت از مسئله نمونه ۲ به ۳ در همه رویکردهای حل مشاهده می‌شود. این به دلیل افزایش در اندازه مسئله و به طور مشخص به دلیل نیاز به یک بارگیری مجدد بیشتر در مسئله نمونه ۳ هست. در مسائل نمونه بزرگ ۶ و ۷،

جدول ۱: نتایج و زمان‌های محاسباتی حل دقیق و الگوریتم ژنتیک پیشنهادی روی مسائل نمونه

شماره نمونه	تعداد مشتری‌ها	سیپلکس			GA			
		تابع هدف	زمان (ثانیه)	تعداد دفعات بارگیری مجدد	بهترین مقدار	زمان (ثانیه)	تعداد دفعات بارگیری مجدد	فاصله بهینگی (%)
۱	۵	۴۴۱,۱۸۴*	۳,۳۹۱	۱	۴۴۱,۱۸۴*	۰,۲۳۶	۱	۰
۲	۶	۴۴۲,۹۷۴*	۳,۶۹۶	۱	۴۴۲,۹۷۴*	۰,۴۵۲	۱	۰
۳	۸	۶۵۵,۸۰۷*	۱۶,۱۵۵	۲	۶۵۵,۸۰۷*	۲,۱۷۰	۲	۰
۴	۹	۶۶۶,۴۷۱*	۱۰۸,۲۳۷	۲	۶۶۹,۸۸۳	۱۰,۸۹۰	۲	-۵۱
۵	۱۰	۷۳۶,۹۹۳*	۶۱۹,۳۲۲	۲	۷۴۲,۲۲۴	۱۲,۲۶۰	۲	-۰,۷
۶	۱۱	۸۰۱,۳۸۵**	۵۰۰۰	۲	۷۵۷,۸۰۱	۱۴,۰۷۲	۲	-
۷	۱۲	۸۲۶,۳۳۹**	۵۰۰۰	۲	۷۷۳,۸۹۴	۱۴,۵۵۳	۲	-

\* جواب بهینه؛ \*\* حد بالا

جدول ۲: نتایج و زمان‌های محاسباتی حل دقیق و الگوریتم جستجوی ممنوع پیشنهادی روی مسائل نمونه

شماره نمونه	تعداد مشتری‌ها	سیپلکس			TS			
		تابع هدف	زمان (ثانیه)	تعداد دفعات بارگیری مجدد	بهترین مقدار	زمان (ثانیه)	تعداد دفعات بارگیری مجدد	فاصله بهینگی (%)
۱	۵	۴۴۱,۱۸۴*	۳,۳۹۱	۱	۴۴۱,۱۸۴*	۰,۵۴۸	۱	۰
۲	۶	۴۴۲,۹۷۴*	۳,۶۹۶	۱	۴۴۲,۹۷۴*	۰,۹۳۸	۱	۰
۳	۸	۶۵۵,۸۰۷*	۱۶,۱۵۵	۲	۶۵۵,۸۰۷*	۵,۲۳۱	۲	۰
۴	۹	۶۶۶,۴۷۱*	۱۰۸,۲۳۷	۲	۷۰۸,۵۹۳	۱۰,۳۸۴	۲	۵,۹۴
۵	۱۰	۷۳۶,۹۹۳*	۶۱۹,۳۲۲	۲	۷۶۴,۴۷۲	۱۶,۴۳۰	۲	۳,۵۹
۶	۱۱	۸۰۱,۳۸۵**	۵۰۰۰	۲	۷۸۶,۳۲۷	۲۶,۲۳۵	۲	-
۷	۱۲	۸۲۶,۳۳۹**	۵۰۰۰	۲	۷۹۸,۵۲۸	۲۳,۵۵۲	۲	-

\* جواب بهینه؛ \*\* حد بالا

## نتیجه و جمع بندی

در این مقاله، یک مدل برنامه ریزی ریاضی عدد صحیح مختلط با نوع جدیدی از محدودیت‌های حذف زیرتور برای مسئله مسیریابی وسیله نقلیه چندانباری، با در نظر گرفتن مسیر بین انبارها ارائه شده است که تخصیص بهینه مشتریان به دپوها و تعداد بهینه مسیرها را بین دپوها و مشتریان به طور همزمان مشخص می‌کند. VRP یک مسئله NP-hard است و حل بهینه آن در زمان معقول مشکل است. بنابراین، در این مقاله یک الگوریتم ژنتیک و یک الگوریتم جستجوی ممنوع برای به دست آوردن

جواب‌های نزدیک بهینه در مسائل بزرگ ارائه شده و سپس کارآیی آنها از نظر زمان حل و کیفیت جواب، بررسی می‌شود. نتایج محاسباتی نشان می‌دهند که الگوریتم‌های پیشنهاد شده کارآ هستند.

به عنوان توصیه برای پژوهش‌های بعدی، فرمول بندی مسئله می‌تواند شامل دپوهای میانی با هزینه‌های موجودی و مقدار عرضه محدود باشد. همچنین، در نظر گرفتن یک یا چند وسیله نقلیه در هر دپو به جنبه واقعی تر بودن مسئله می‌افزاید.

## مراجع

- 1- Sumichrast, R.T., and Markham, I.S. (1995). "A heuristic and lower bound for a multi-depot routing problem." *Computers & Operation Research*, Vol. 22, PP. 1047-1056.
- 2- Tillman, F.A., and Cain, T.M. (1972). "An Upper Bound Algorithm for the Single and Multiple Terminal Delivery Problem." *Management Science*, Vol. 18, No. 11, PP. 664-682.
- 3- Gillett, B.E., and Johnson, J.G. (1976). "Multi-terminal vehicle dispatch algorithm." *Omega*, Vol. 4, No. 6, PP. 711-718.
- 4- Giosa, I.D., Tansini, I.L., and Viera, I.O. (2002). "New Assignment Algorithms for the Multi-Depot Vehicle Routing Problem." *Journal of the Operational Research Society*, Vol. 53, No. 9, PP. 977-987.
- 5- Nagy, G., and Salhi, S. (2005). "Heuristic Algorithms for Single and Multiple Depot Vehicle Routing Problems with Pickups and Deliveries." *European Journal of Operational Research*, Vol. 162, No. 1, PP. 126-141.
- 6- Thangiah, S.R., and Salhi, S. (2001). "Genetic Clustering: an Adaptive Heuristic for the Multi-Depot Vehicle Routing Problem." *Applied Artificial Intelligence*, Vol. 15, No. 4, PP. 361-383.
- 7- Ho, W., Ho, G.T.S., Ji, P., and Lau, H.C.W. (2008). "A Hybrid Genetic Algorithm for the Multi-Depot Vehicle Routing Problem." *Engineering Application of Artificial Intelligence*, Vol. 21, No. 4, PP. 548-557.
- 8- Dondo, R., and Cerda, J. (2009). "A Hybrid Local Improvement Algorithm for Large-Scale Multi-Depot Vehicle Routing Problems with Time Windows." *Computer and Chemical Engineering*, Vol. 33, No. 2, PP. 513-530.
- 9- Crevier, B., Cordeau, J.-F. and Laporte, G. (2007). "The multi-depot vehicle routing problem with inter-depot routes." *European Journal of Operational Research*, Vol. 176, PP. 756-773.
- 10- Jordan, W. C. and Burns, L. D. (1984). "Truck backhauling on two terminal networks." *Transportation Research Part B: Methodological*, Vol. 18, PP. 487-503
- 11- Jordan, W. C. (1987). "Truck backhauling on networks with many terminals." *Transportation Research Part B: Methodological*, Vol. 21, PP. 183-193.
- 12- Tarantilis, C. D., Zachariadis, E. E. and Kiranoudis, C. T. (2008). "A hybrid guided local search for the vehicle-routing problem with intermediate replenishment facilities." *INFORMS Journal on Computing*, Vol. 20, PP. 154-168.

- 13- Kek, A. G., Cheu, R. L. and Meng, Q. (2008). "Distance-constrained capacitated vehicle routing problems with flexible assignment of start and end depots." *Mathematical and Computer Modelling*, Vol. 47, PP. 140-152.

### واژه‌های انگلیسی به ترتیب استفاده در متن

- 1 - Multi-Depot Vehicle Routing Problem (MDVRP)
  - 2 - Multi-Depot Vehicle Routing Problem with Inter-Depot Routes (MDVRPIDR)
  - 3 - Genetic Algorithm (GA)
  - 4 - Tabu Search (TS)
-