فصلنامه مدیریت صنعتی دانشکده علوم انسانی، دانشگاه آزاد اسلامی
واحد سنندج – سال نهم / شماره ۲۸ / تابستان ۱۳۹۳

ترکیب الگوریتم‌های جستجوی ممنوع و جمعیت مورچگان برای مسئله مسیریابی وسیله نقلیه

نرگس محمودی دارانی
مربی، دانشگاه آزاد اسلامی، واحد رباط کریم، رباط کریم، ایران
اعظم دولت‌نژاد ثمرین
مربی، دانشگاه آزاد اسلامی، واحد تهران شمال، باشگاه پژوهشگران جوان و نخبگان، تهران، ایران
مجید یوسفی خوشبخت(نویسنده مسؤول)
مربی، دانشگاه آزاد اسلامی، واحد همدان، باشگاه پژوهشگران جوان و نخبگان، همدان، ایران
Email: khoshbakht@aut.ac.ir

چکیده

مسئله مسیریابی وسیله نقلیه (VRP) یکی از مهم‌ترین مسائل بهینه‌سازی ترکیباتی است که امروزه به علت کاربردهای وسیع که در مشکلات روزمره دارد بسیار مورد توجه قرار می‌گیرد. در این مسئله ناوگانی از وسایل نقلیه با ظرفیت Q از گره‌ای به نام انبار شروع به حرکت می‌کنند و بعد از سرویس‌دهی به مشتریان به آن باز می‌گردند به شرط آنکه هر کدام از مشتریان را فقط یک‌بار مورد ملاقات قرار دهند و در هیچ زمانی بیشتر از ظرفیت محدود Q بارگذاری نکنند. هدف در این مسئله کمینه‌کردن تعداد وسایل نقلیه به همراه مسیرهای پیموده شده توسط آن‌ها است. این مقاله نوعی روش ترکیبی جستجوی ممنوع را برای این مسئله پیشنهاد می‌کند. در این روش برای جستجوی همسایگی و حرکت از یک جواب به جواب دیگر از سه حرکت درج، جابجایی و الگوریتم جمعیت مورچگان استفاده می‌شود. برای آزمایش کارایی الگوریتم، چهارده مثال استاندارد کریستوفیدز در نظر گرفته شده و الگوریتم بر روی آن مورد اجرا قرار گرفته است. نتایج محاسباتی روی این مثال‌ها که دارای اندازه‌ای از ۵۰ تا ۱۹۹ می‌باشند نشان می‌دهد که الگوریتم پیشنهادی توانسته است که رقابت خوبی با الگوریتم‌های مشهور فراابتکاری از نظر کیفیت جواب‌ها داشته باشد. به علاوه جواب‌های نزدیک به بهترین جواب‌های تاکنون بدست آمده برای بیشتر مثال‌ها بدست آورده است به طوری که سه بهترین جواب توسط این الگوریتم به دست آمد.

کلمات کلیدی: جستجوی ممنوع، جمعیت مورچگان، مسیریابی وسیله نقلیه، ترکیب الگوریتم‌ها.

۱- مقدمه

مسئله مسیریابی وسیله نقلیه[1] (VRP) یکی از مهم‌ترین مسائل تحقیق در عملیات است که به علت سهم قابل توجه هزینه توزیع کالا از هزینه تولید، امروزه بسیار در لجستیک و زنجیره تأمین مورد توجه قرار می‌گیرد و محققان زیادی را به خود جذب کرده است. شاید بتوان گفت بسیاری از مسائل حوزه توزیع کالا را می‌توان به یکی از نسخه‌های این مسئله تبدیل کرد و سپس آن‌را مورد بررسی قرار داد. به علاوه این مسئله بیشتر در بخش‌های صنعت و خدمات کاربردهای گسترده‌ای دارد زیرا بهبود کارایی در حمل و نقل کالاها باعث صرفه‌جویی قابل توجه در قیمت تمام شده آن‌ها می‌شود و سبب می‌گردد شرکت‌های صنعتی و خدماتی قابلیت بیشتری برای رقابت در برابر رقبای خود داشته باشند و بتوانند با قیمت کمتری کالای خود را بدست خریداران رسانده و از این طریق بازار کالای خود را گسترش داده و سود بیشتری را بدست آورند.

اهمیت این مسئله در دنیای واقعی سبب شده است که این مسئله در زندگی روزمره بسیار مورد توجه قرار گرفته است (Yousefikhoshbakht et al., 2012). از کاربردهایی این مسئله در زندگی واقعی، می‌توان به کارهای انجام شده به وسیله هالسکو و لوکتانژن (۱۹۹۸) و پرایو و همکاران (۲۰۰۶) در صنعت آشامیدنی اشاره کرد. همچنین مین (۱۹۸۹) در ارتباط با سیستم توزیع کتابخانه عمومی این مسئله را بررسی کردند در حالی که آنیلی و موشیو (۱۹۹۴) کاربرد دیگری از آن در حمل و نقل کودکان را در برنامه اجتماعی ارائه دادند. کاربردهای دیگری که در ارسال محموله پستی است، توسط وازنر و زافل (۲۰۰۴) مورد توصیف قرار گرفته شد. همچنین این مسئله به طور طبیعی در جابجایی بطری‌های پر و خالی توسط دفلف (۲۰۰۱) و همچنین تانگ و گالوا (۲۰۰۶) به کار رفته است. از کاربردهای دیگر مسئله VRP می‌توان به سیستم اتوبوس‌رانی، توزیع‌کردن گل به گل‌فروشی‌ها، جمع‌آوری زباله‌های شهری، مسیریابی کشتی‌ها و هواپیماهای باری، توزیع روزنامه، جمع‌آوری پول از بانک‌ها و همچنین پشتیبانی ارتش در زمان جنگ اشاره کرد.

تعریف ساده این مسئله بدین صورت است که اگر $V=\{0,1,...,n\}$ و $A=\{(i,j)|\, i,j\in V \text{ and } i\neq j\}$ به ترتیب نشان دهنده مجموعه گره و مجموعه یال‌های گراف $G(V,A)$ باشد. گره 0 نمایش‌دهنده انبار کالا و بقیه گره‌های نشان دهنده مشتری‌ها هستند که هر کدام دارای مقدار تقاضای مشخص می‌باشند. همچنین تقاضاهای هر یک از مشتری‌ها فقط باید به وسیله یکی از سرویس دهنده مسئله برآورده شود. باید توجه کرد که در این مسئله همان‌طور که گفته شد نکات قابل توجه دیگری مانند اینکه همه وسیله نقلیه در ابتدای الگوریتم باید در گره انبار باشند، هیچ سرویس دهنده‌ای در هیچ زمان مجاز نیست که بیشتر از ظرفیت معین شده‌ای بارگذاری کند و تقاضای هر مشتری نباید بیشتر از ظرفیت مجاز باشد، وجود دارد. اگر ماتریس هزینه برای رفتن هر یک از سرویس دهنده‌ها از گره‌ای به گره دیگر با ماتریس متقارن C نمایش داده شود، هدف مسئله عبارت است از یافتن مسیرهایی با کمترین هزینه برای سرویس‌دهنده‌ها به طوری که بعد از سرویس‌دهی به انبار بازگردند. برای ارائه مدل، معرفی چندین متغیر و پارامتر به عنوان ورودی لازم است که به شرح زیر می‌باشند.

n: تعداد مشتریان که معمولاً به شکل اندیس‌های ۱ تا n همراه سایر پارامترها دیده می‌شود.

m: تعداد وسایل نقلیه مورد استفاده در هر مسئله که در ابتدا همگی در انبار مستقر شده‌اند.

Q: ظرفیت هر وسیله نقلیه که برای همه وسایل نقلیه یکسان در نظر گرفته می‌شود.

q_i: مقدار تقاضای مشتری iام است که فرض می‌شود برای هر i از مقدار Q بیشتر نباشد.

c_{ij}: هزینه سفر بین دو گره i و j است.

x_{ij}^k: اگر وسیله نقلیه k بعد از گره i گره j را ملاقات کند $x_{ij}^k=1$ و در غیر این صورت $x_{ij}^k=0$ می‌باشد $(\forall i,j=0,1,2,...,n,\forall k=1,...,m)$.

بنابراین مدل مسئله VRP عبارت است از:

[1] Vehicle Routing Problem (VRP)

(۱)

$$\min \sum_{k=1}^{m}\sum_{i=0}^{n}\sum_{j=0}^{n} c_{ij}x^{k}_{ij}$$

(۲)

$$\sum_{k=1}^{m}\sum_{i=0}^{n} x^{k}_{ij} = 1 \qquad \forall j = 1,...,n$$

(۳)

$$\sum_{k=1}^{m}\sum_{j=0}^{n} x^{k}_{ij} = 1 \qquad \forall i = 1,...,n$$

(۴)

$$\sum_{i=0}^{n} x^{k}_{iu} - \sum_{j=0}^{n} x^{k}_{uj} = 0 \qquad \forall u = 1,...,n, \forall k = 1,...,m$$

$$\sum_{i,j\in S} x^{k}_{ij} \le |S| - 1 \qquad (\forall S \subset \{1,...,n\} : 1 \le |S| \le n, \forall k = 1,...,m) \qquad (۵)$$

(۶)

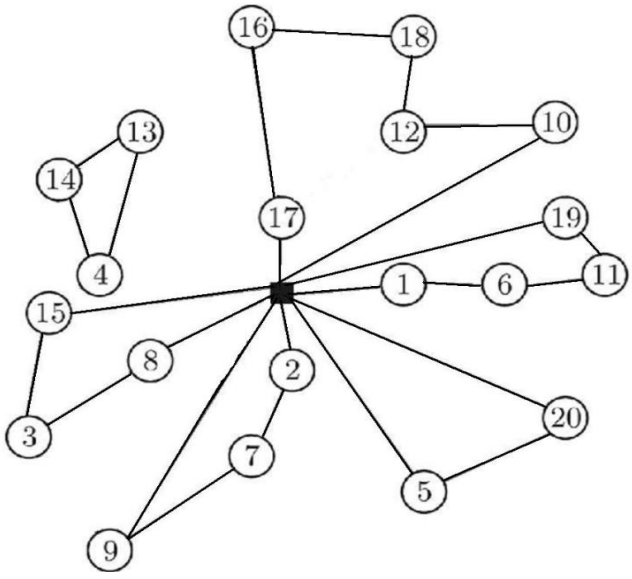
$$\sum_{j=1}^{n} q_j \left(\sum_{i=0}^{n} x^{k}_{ij} \right) \le Q \qquad \forall k = 1,...,m$$

(۷)

$$x^{k}_{ij} \in \{0,1\} \qquad \forall i = 0,...,n, \forall j = 0,...,n, \forall k = 1,...,m$$

در این مدل (۱) نشان دهنده تابع هدف مسئله است که باید تا حد امکان مینیمم شود در حالی که (۲) و (۳) نشان‌دهنده آن هستند که به هر گره بجز انبار فقط یک وسیله نقلیه وارد و خارج می‌شود. همچنین محدودیت‌های (۴) سبب می‌شوند پیوستگی هر وسیله نقلیه حفظ شود در حالی که محدودیت‌های (۵) باعث می‌شوند که الگوریتم جوابی را که دارای یک زیر دور فاقد انبار است، مورد قبول قرار ندهد. به علاوه محدودیت (۶) به این نکته اشاره می‌کند که هیچ کدام از وسائل نقلیه مجاز نخواهند بود که بیشتر از ظرفیت ثابت Q اقدام به بارگذاری نمایند. در نهایت محدودیت‌های (۷) به شرایط دودویی متغیرهای x^{k}_{ij} اشاره می‌کند.

توجه به این نکته ضروری است که محدودیت (۵) محدودیت بسیار مهمی محسوب می‌شود. این محدودیت سبب می‌شود که الگوریتم از ایجاد جوابی که همبند نباشد، جلوگیری کند. به عبارت دیگر همانطور که در شکل ۲ می‌بینید این مثال در شرایط (۱)، (۲) و (۳) صدق کرده و به هر گره به غیر از انبار یک یال وارد و یک یال خارج می‌گردد و همچنین این تعداد یال ورودی و خروجی برای انبار m است. در حالی که یک جواب غیر قابل قبول می‌باشد زیرا سه گره ۴، ۱۳ و ۱۴ به انبار کالا دسترسی ندارند. پس این محدودیت همراه دو محدودیت بالا سبب می‌شود که گره‌ها از نظر درجه مشکلی نداشته و دقیقا یکبار مورد ملاقات قرار گرفته و به انبار دسترسی داشته باشند. بنابراین این محدودیت از ایجاد زیر دور نامتصل به انبار جلوگیری می‌کند.



شکل شماره (۱): نمونه‌ای از یک حل غیر قابل قبول مسئله مسیریابی وسیله نقلیه

مسئله مسیریابی وسیله نقلیه که یکی از مهم‌ترین مسائل حمل و نقل است تقریباً از دهه ۵۰ میلادی مورد توجه قرار گرفته شده است و اولین بار به وسیله دانتزیک و همکاران در سال ۱۹۵۹ معرفی شد. در آن مقطع یک فرمول‌بندی مسئله مورد توجه قرار گرفت و الگوریتم ریاضی آن پیشنهاد شد (Toth & Vigo, 2002). بعلاوه در این مسئله با فرض اینکه ظرفیت وسایل نقلیه به طور یکسان و برابر با بی‌نهایت در نظر گرفته شده است، وسائل نقلیه با یکدیگر مشابه هستند و مدت زمان توقف در گره‌ها برابر با صفر در نظر گرفته می‌شود، هدف تعیین مسیری است که هر خودرو باید طی کند تا به دسته‌ای از مشتریان در یک گستره جغرافیایی خدمتی را ارائه نماید. لذا تعیین مجموعه مشتریان اختصاص یافته به هر خودرو و هم تعیین مسیر رفت و برگشت و تقدم و تأخر مراجعه به آن‌ها دو سؤالی است که باید در این مسئله به آن پاسخ داده شود. به عبارت دیگر، علاوه بر دسته‌بندی مشتریان و تخصیص دادن هر مشتری به وسیله‌ای خاص، باید مسیریابی خودروها هم انجام شود. بنابراین این مسئله ابتدایی همان مسئله چندین فروشنده دوره‌گرد است که در آن محدودیتی برای بازدید از یک مشتری وجود ندارد.

شاید ساده‌ترین نسخه از مسئله VRP، مسئله فروشنده دوره‌گرد[۲] (TSP) است زیرا در این مسئله فرض شده است که یک وسیله نقلیه وجود دارد که ظرفیت آن بی‌نهایت است. بنابراین در اینجا فاکتور انتخاب مشتری‌ها برای هر گره در نظر گرفته نمی‌شود و هدف فقط یافتن مسیری برای این وسیله نقلیه‌ای است که همه گره‌ها را باید پیمایش کرده و در انتها به محل شروع حرکت بازگردد. باید توجه کرد که مسئله TSP که یکی از مشهورترین مسائل بهینه‌سازی ترکیباتی محسوب می‌شود به وسیله دانتزیک (۱۹۵۴) ارائه شد. دانتزیک مدل برنامه‌ریزی صفر و یک را برای مسئله ارائه کرد. در دو دهه اول پس از طرح مسئله تلاش‌ها بر یافتن روش‌های دقیق برای مسئله متمرکز بود. در این روش‌ها سعی می‌شد که قسمت قابل توجهی از فضای شدنی مسئله مورد بررسی قرار گیرد و جواب بهینه مسئله بدست آید. باید توجه کرد که بعضی از مسائل، به علت افزایش بسیار زیاد جواب‌های شدنی در هنگام افزایش بعد مسئله، در این زمان به علت ابعاد بزرگ قابل حل نبودند. استفاده گسترده از روش‌های دقیق تا دهه ۷۰ ادامه داشت و به طور مثال الیون (۱۹۷۱) الگوریتم دقیقی که مبنای آن برنامه‌ریزی پویا و تابع هدف آن خطی بود، ارائه کرد، که زمان حل این الگوریتم تابعی به صورت نمایی بود. باید اضافه کرد که اگر چه الگوریتم آن‌ها دارای پیچیدگی کمی بود اما حداکثر مسئله‌ای که با این الگوریتم مورد حل قرار گرفت مسائلی با ۱۰ گره بود. استفاده از این روش‌ها ادامه داشت تا اینکه نظریه پیچیدگی محاسبات در این زمان مورد بحث قرار گرفت و مشخص شد که مسئله TSP به مجموعه مسائل NP–سخت تعلق دارد (Garey & Johnson, 1979). به دنبال آن تلاش‌هایی زیادی برای یافتن الگوریتم‌های ابتکاری برای حل این‌گونه از مسائل صورت گرفت تا بتواند مسئله را در یک زمان معقول مورد حل قرار دهد. به طور نمونه از این

[2] Traveling Salesman Problem

الگوریتم‌ها می‌توان به روش نزدیک‌ترین همسایه، الگوریتم‌های بهبود دهنده دوگانه، سه‌گانه و همچنین kگانه برای $k>3$ اشاره کرد.

اگر چه مسئله *TSP* فقط یک حالت بسیار ساده از مسائل مسیریابی وسائل نقلیه محسوب می‌شود، روند و نتایج تحقیقات بر روی این مسئله مستقیماً بر روی مسئله *VRP* تأثیر دارد به طوری که روش‌های حل آن، که تا قبل از طرح نظریه پیچیدگی محاسبات مبتنی بر روش‌های دقیق و استفاده از برنامه‌ریزی صفر و یک و پویا بودند، به طور کلی تغییر کرده و به سمت روش‌های ابتکاری حرکت کرده است. بنابراین با طرح روش‌های ابتکاری برای حل مسئله *TSP* راه برای توسعه روش‌های ابتکاری برای حل *مسائل VRP* نیز گشوده شده و می‌توان این‌گونه از الگوریتم‌ها را به آن مسائل گسترش داد. برای نمونه از این الگوریتم‌ها می‌توان به کارهای انجام شده توسط گلدن و اسد (۱۹۸۸) اشاره کرد. باید توجه کرد که مدل ارائه شده توسط کریستوفیدز نتایج خوبی را در بر داشته و توانست مسئله‌ای با ۲۵ گره را حل کند و به جواب بهینه دست یابد. همچنین در همین سال یک روش تخصیص عمومی برای حل مسئله *VRP* توسط فیشر و جی کومار (۱۹۷۸) ارائه شد که از کیفیت خوبی برخوردار بود و قادر بود مسائلی با ابعاد بزرگ را حل کند. این روش در سال ۱۹۹۰ توسط مارتلو و توث (۱۹۹۰) توسعه داده شد. در کنار حل مسائل *VRP* به شکل قطعی، می‌توان جنبه‌های دیگری نیز به مسئله اضافه کرد که باعث شود نسخه‌های دیگری از مسئله به وجود آیند که از این میان می‌توان به موارد احتمالی بودن تقاضای گره‌ها، احتمالی بودن زمان ارائه خدمات و وجود گره‌های احتمالی اشاره کرد. فرمول‌بندی این‌گونه از مسائل به دو صورت انجام می‌شود که یا به صورت برنامه‌ریزی احتمالی عدد صحیح و آمیخته است و یا اینکه از فرایندهای تصمیم‌گیری مارکوفی استفاده می‌شود. برای مثال از این‌گونه از روش‌ها می‌توان به کارهای انجام شده به وسیله استیوارت و گلدن (۱۹۸۳)، لاپورته و همکاران (۱۹۸۹) اشاره کرد.

از طرف دیگر بنا بر توضیحات داده شده چون مسئله مسیریابی وسیله نقلیه جزء مسائل سخت دسته‌بندی می‌شود و الگوریتم‌هایی دقیق با پیچیدگی چندجمله‌ای برای مسئله وجود ندارد، تمایل به یافتن روش‌های ابتکاری و فراابتکاری در کنار روش‌های دقیق برای این مسئله گسترش یافت. به خصوص در دهه گذشته و با افزایش سرعت کامپیوترهای شخصی، استفاده از این روش‌ها بسیار فراگیر شده بطوری‌که روش‌های بسیار زیادی برای این مسئله پیشنهاد شدند که بعضی از مهم‌ترین آن‌ها جستجوی ممنوع (Zachariadis et al., 2009)، شبیه‌سازی آنیلی (Osman, 1993)، الگوریتم رقابت استعماری (Yousefikhoshbakht et al., 2012) ، الگوریتم مورچگان (Santos et al., 2010)، الگوریتم ممتیک (Prins, 2009)، جستجوی پراکنده (Tang et al., 2010)، ترکیبی نزدیک‌ترین همسایه با جستجوی ممنوع (Du & He, 2012)، الگوریتم شبه آهنربایی (Yurtkuran & Emel, 2010)، الگوریتم کاهنده متغیرهای همسایه (Chen et al., 2010)، جستجوی همسایگی بزرگ تطابقی (Hemmelmayr et al., 2010) و الگوریتم همسایه متغیر (Imran et al., 2009) می‌باشند.

کاربردهای ذکر شده باعث شده است که محققین الگوریتم‌هایی زیادی برای حل این مسئله ارائه کنند که می‌توان به طور کلی این الگوریتم‌ها را به دو دسته دقیق و تخمینی تقسیم‌بندی کرد. در دسته اول که برای مسائل با ابعاد کوچک مورد استفاده قرار می‌گیرد، جواب بهینه مسئله در یک زمان بسیار زیاد بدست می‌آید ولی در دسته دوم که تقریباً سه دهه است که بصورت جدی به آن پرداخته می‌شود جواب بهینه مسئله در بیشتر مواقع بدست نمی‌آید ولی الگوریتم می‌تواند به یک جواب نزدیک به بهینه در یک زمان قابل قبول دست پیدا کند. باید توجه کرد که الگوریتم‌های تخمینی خود نیز به دو دسته الگوریتم‌های ابتکاری و فراابتکاری تقسیم‌بندی می‌شوند. در روش‌های فراابتکاری که میزان اجرای الگوریتم، برخلاف روش‌های ابتکاری، به تصمیم کاربر وابسته است، جواب‌ها تقریباً در یک زمان بیشتر از الگوریتم‌های ابتکاری و کمتر از روش‌های دقیق بدست می‌آیند. اگر چه معمولاً جواب‌های این روش‌ها از جواب‌های روش‌های ابتکاری بهتر است و این‌گونه از الگوریتم‌ها از راهکارهایی استفاده می‌کنند که تا حد ممکن در بهینه‌های محلی گیر نیفتند، اما پارامترهای زیادی در این الگوریتم‌ها وجود دارند که باید توسط کاربر به صورت تجربی بدست آیند. این حجم زیاد پارامترها سبب می‌شوند که الگوریتم‌ها نتوانند در تکرارهای مشابه جواب‌های یکسانی بدست آورند. بنابراین می‌توان نتیجه گرفت که این‌گونه الگوریتم‌ها دارای یک روال ثابت برای رسیدن به جواب نبوده و پارامترهای تصادفی نقش زیادی را در این الگوریتم‌ها بازی می‌کنند. ترکیب روش‌های جستجوی ممنوع و مورچگان بدین علل

در این مقاله مورد بررسی قرار گرفته شده است که اولاً این دو روش از بهترین روش‌ها برای حل مسائل مسیریابی محسوب می‌شوند و ثانیا این ترکیب سبب می‌شود که تا حد ممکن از مزایای هر کدام از الگوریتم‌ها به نحو احسن استفاده شود. از طرف دیگر دو الگوریتم درج و جابجایی از قوی‌ترین روش‌ها برای جستجوی محلی هستند و بدین علت در این الگوریتم استفاده شده است. این استراتژی باعث شد که جواب‌های بسیار خوبی برای مسائل VRP بدست آید که در بخش‌های بعدی ارائه می‌گردند.

در این مقاله ابتدا در بخش ۲ به روش پیشنهادی با جزئیات بیشتری پرداخته می‌شود. در انتها نتایج محاسباتی که بر روی مثال‌های استاندارد اجرا شده است در بخش ۳ مورد تجزیه و تحلیل قرار می‌گیرد.

۲- مواد و روشها

اگر چه الگوریتم TS یکی از قدیمی‌ترین الگوریتم‌های فراابتکاری است اما هنوز هم در حل مسائل بهینه‌سازی بسیار مورد توجه قرار می‌گیرد و در بیشتر مسائل بهینه‌سازی مورد استفاده قرار می‌گیرد. این روش در سال‌های اخیر به وسیله دانشمندان به وسیله روش‌های ابداعی تکامل یافته و به روشی بسیار کارا تبدیل شده است به طوری‌که در بعضی از مسائل بهینه‌سازی ترکیباتی به بهترین جواب‌ها دست یافته است. از جمله مهم‌ترین اصلاحاتی که تاکنون بر روی این الگوریتم صورت گرفته است می‌توان به موارد زیر اشاره کرد:

- پیاده‌سازی الگوریتم به صورت موازی
- به‌کارگیری روش‌های دیگر برای ترکیب با این روش برای ارتقای بیشتر الگوریتم
- استفاده از روش‌های متفاوت و متنوع برای ایجاد همسایه
- استفاده از روش‌های مختلف تنوع[3] و تشدید[4] به منظور هدایت فرآیند جستجو در الگوریتم
- استفاده از جستجوی محلی برای افزایش قدرت الگوریتم در یافتن جواب‌های باکیفیت بهتر

استفاده از یکی از اصلاحات و یا تعدادی از آن سبب گردید که نسخه‌های کاراتر از این الگوریتم پدید آید که خواننده برای اطلاعات بیشتر می‌تواند به (Cordeau et al., 2001) مراجعه کند. بنابر تعاریف ذکر شده ایده اصلی الگوریتم پیشنهادی ارائه یک الگوریتم کارا برای ایجاد همسایه به وسیله استفاده از الگوریتم‌های درج[5]، جابجایی[6] و الگوریتم سیستم مورچگان است برای جستجوی بیشتر فضای نمونه و یافتن جواب‌های بهتر است. حال به گام‌های اساسی الگوریتم جستجوی ممنوع پیشنهادی پرداخته می‌شود.

ایجاد جواب اولیه: اگر چه الگوریتم‌های زیادی تاکنون برای ایجاد جواب اولیه در الگوریتم TS ارائه شده‌اند اما روش صرفه‌جو یکی از مشهورترین الگوریتم‌های ایجاد جواب اولیه است که به وسیله کلارک و رایت (۱۹۶۴) پیشنهاد شد و در مقایسه با دیگر الگوریتم‌های ابتکاری از کارایی بیشتری برخوردار است. به همین دلیل در این مقاله از این روش به عنوان ایجاد جواب اولیه برای الگوریتم TS از آن استفاده می‌شود.

این روش علاوه بر اینکه می‌تواند برای مسائلی که تعداد وسیله نقلیه آن یک متغیر تصمیم‌گیری است، بکار رود. می‌تواند به صورت موازی نیز مورد استفاده قرار گیرد. روند اجرای این الگوریتم به صورت زیر می‌باشد:

۱. به تعداد مشتری‌ها، n مسیر وسیله نقلیه $(0,i,0)$ برای $i=1,...,n$ بسازید.
۲. برای هر مشتری $i,j=1,...,n$ به طوری که $i\neq j$ مقدار صرفه‌جویی $s_{ij}=c_{i0}+c_{0j}-c_{ij}$ را محاسبه کنید.
۳. بزرگ‌ترین s_{ij} غیر تکراری را به طوری پیدا کنید که:
   أ. i و j در یک زیر تور یکسان نباشد.
   ب. هیچ‌کدام از گره‌های i و j نقاط داخلی زیر تور نباشند (از i و j به انبار راه مستقیم وجود داشته باشد.).
   ج. دو زیر تور را با استفاده از متصل‌کردن گره‌های i و j ، بهم وصل کرده و یک زیر تور بزرگ‌تر بسازید.

[3] Diversification
[4] Intensification
[5] Insert Move
[6] Swap Move

د. اگر توری شامل همه مشتری‌ها ایجاد نشده است به ۳ بروید و در غیر این صورت به ۴ بروید.

۴. پایان

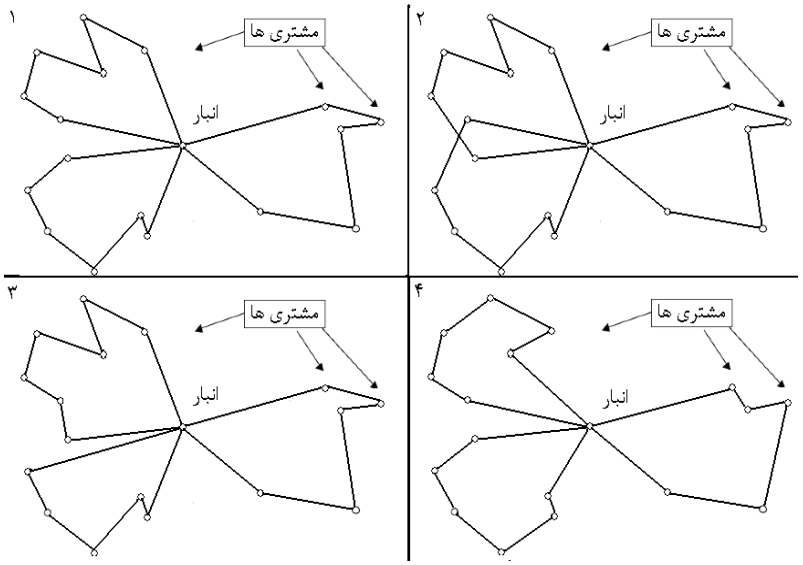
همسایگی: برای هر جواب s در الگوریتم‌های جستجوی همسایگی یک مجموعه‌ی همسایگی $N(s)$ تعریف می‌شود. برای به دست آوردن $N(s)$ می‌توان همه انتقال‌های ممکن را بر روی جواب فعلی s انجام داد و عناصر $N(s)$ را تعیین کرد. باید توجه کرد که در بسیاری از مسائل ممکن است تعداد این انتقال‌ها بسیار زیاد باشد و امکان محاسبه همه‌ی آن‌ها از نظر هزینه محاسبات وجود نداشته باشد. به همین علت به صورت تصادفی و یا با به کار بردن یک استراتژی کاهش‌دهنده زیر مجموعه‌ای از $N(s)$ در نظر گرفته می‌شود و به عنوان مجموعه همسایگی به کار می‌رود. حال مقادیر این مجموعه جواب مورد بررسی قرار می‌گیرد و از بین آن‌ها بهترین مقدار و جواب برای مقایسه با بهترین مقدار و جواب بدست آمده انتخاب می‌شود. بنابریان بر اساس شرایطی مانند لیست ممنوع، تنوع، تشدید و معیار رضایت که الگوریتم داراست، روند الگوریتم ادامه می‌یابد.

الگوریتم‌های زیادی تاکنون برای ایجاد همسایگی ارائه شده‌اند که برای مثال می‌توان به درج، جابجایی، جستجوی محلی دوگانه و سه‌گانه و تقاطع[۷] اشاره کرد. در الگوریتم پیشنهادی علاوه بر الگوریتم مورچگان، دو الگوریتم دیگر درج و جابجایی نیز بکار می‌رود. در الگوریتم درج یک مشتری از یک مسیر برداشته می‌شود و به مسیر دیگر ارسال می‌شود به شرط آنکه اولاً جواب جدید در محدودیت‌های مسئله صدق کند و ثانیاً جواب جدید دارای مقدار بهتری نسبت به قبل باشد. در حالی که در حرکت جابجایی حرکت با جابجا کردن یک مشتری در یک مسیر با یک مشتری دیگر در مسیر دیگر شکل می‌گیرد.

باید توجه کرد که این سه الگوریتم به این علت در الگوریتم پیشنهادی مورد استفاده قرار می‌گیرد که دارای ساختار مناسبی برای جستجوی فضای شدنی مسئله هستند. در الگوریتم درج و جابجایی همسایگی‌هایی در نظر گرفته می‌شوند که در یک مسیر یکسان نیستند و الگوریتم می‌تواند فضایی بیشتری را مورد بررسی قرار دهد در حالی که در زمانی که الگوریتم جمعیت مورچگان مورد استفاده قرار می‌گیرد n مشتری اختصاص یافته به m کامیون عوض نمی‌شوند بلکه در این هنگام که مسئله به m مسئله فروشنده دوره‌گرد تبدیل شده است، الگوریتم مورچگان مسیر هر کامیون را به علت کارایی بالای الگوریتم مورچگان به روز رسانی می‌کند. چون دو عمل درج و جابجایی تغییرات زیادی را در الگوریتم ایجاد می‌کنند و فضای جواب را با گستردگی بیشتری مورد بررسی قرار می‌گیرند، هر کدام با احتمال ۴۰ درصد در هر تکرار الگوریتم مورد استفاده قرار می‌گیرند. از طرف دیگر چون الگوریتم ACS مشتری‌های هر وسیله نقلیه را عوض نمی‌کنند بنابراین الگوریتم بیشتر به وسیله این الگوریتم جستجوی محلی را انجام می‌دهد و همسایگی‌های نزدیک جواب بدست آمده را جستجو می‌کند. به همین علت احتمال استفاده از این الگوریتم در روش پیشنهادی کمتر در نظر گرفته شده و ۲۰ درصد انتخاب می‌شود. باید توجه کرد که این مقادیر درصدها به صورت تجربی و انتخاب مقادیر گوناگون برای این سه احتمال بدست آمده که سبب شده الگوریتم بهترین جواب‌ها را کسب کند.

به طور مثال شکل ۲ را در نظر بگیرید. در قسمت ۱ یک جواب مسئله مسیریابی نشان داده شده است که از یک انبار و تعدادی مشتری با درخواستی معین تشکیل شده است که در آن فرض شده است که سه وسیله نقلیه وجود دارند و مجموع درخواست‌های هر کدام بیشتر از ظرفیت Q وسائل نقلیه نمی‌باشد. در ۲ یک جواب بدست آمده به وسیله الگوریتم جابجایی را نشان می‌دهد. در این جواب دو مشتری از دو مسیر انتخاب شده و به شرط برقراری محدودیت و ایجاد جواب بهتر با یکدیگر تعویض شده‌اند. بعلاوه در ۳ یک جواب ایجاد شده به وسیله الگوریتم درج نشان داده شده است. باید توجه کرد که در این الگوریتم یک مشتری از یک مسیر انتخاب و به یک مسیر دیگر می‌رود. در نهایت ۳ یک جواب ایجاد شده به وسیله الگوریتم ACS نشان داده شده است. همان‌طور که مشخص است مشتری‌های اختصاص یافته به هر کدام از وسائل نقلیه در این جواب عوض نشده‌اند بلکه برای هر کدام از تورها روش الگوریتم مورچگان استفاده شده است. توجه به این نکته ضروری است که در هر تکرار الگوریتم فقط یکی از الگوریتم‌های ۲، ۳ یا ۴ که در شکل ۲ نشان داده شده است، مورد استفاده قرار می‌گیرد.

[7] Cross



شکل شماره (۲): همسایگی‌ها در یک جواب شدنی مسئله *VRP*

لیست ممنوع و بهنگام‌رسانی جواب‌ها: اگر چه الگوریتم TS جواب‌هایی که در جهت عدم بهبود هستند را نیز می‌پذیرد. اما الگوریتم با این عمل ریسک مشاهده مجدد یک جواب و ایجاد دور را بالا می‌برد. در این‌جا برای جلوگیری از حرکاتی که ممکن است منجر به مشاهده جواب‌های تکراری شود، از یک حافظه کوتاه مدت به نام لیست ممنوع استفاده می‌شود و تمام جواب‌هایی که اخیراً مشاهده شده‌اند برای تعداد مشخصی تکرار در این لیست قرار می‌گیرند. به عبارت دیگر وقتی که مشتری i در مسیر k درج و یا جابجا شد آنگاه مقادیر (i,k) در لیست ممنوع قرار می‌گیرد. بنابراین جابجایی از جواب فعلی به جواب بعدی زمانی انجام می‌شود که جواب جدید در لیست ممنوع نباشد. در نتیجه بدین وسیله الگوریتم از بازگشت مجدد به جواب‌های اخیراً مشاهده شده و ایجاد دور جلوگیری می‌کند. باید توجه کرد که در الگوریتم پیشنهادی مانند روش کلاسیک بعد از هر حرکت (از یک جواب به جواب بعدی) لیست ممنوع به هنگام می‌شود. به این صورت که جواب جدید به لیست اضافه می‌شود و یکی از جواب‌های در لیست، از آن خارج می‌شود. ترتیب خروج از لیست آن است که جواب زودتر وارد شده به لیست ممنوع، زودتر از لیست خارج می‌شود (همانند صف).

نقش اساسی لیست ممنوع جلوگیری از دوباره ملاقات کردن جواب‌ها است مگر شرایط رضایت (آزادسازی)[8] برقرار باشد. بعلاوه طول لیست ممنوع فرآیند جستجو را کنترل می‌کند به طوری که لیست کوچک سبب می‌شود حالت ممنوعیت سریع‌تر از جواب‌ها برداشته شده و جستجو روی ناحیه کوچکی از جواب‌ها متمرکز می‌شود. اما زمانی که لیست بزرگ باشد، جواب‌های بیشتری در لیست قرار می‌گیرند و محدوده جستجو گسترش می‌یابد. الگوریتم پیشنهادی از حالت واکنشی[9] استفاده می‌کند که در این حالت طول لیست بین صفر (مقدار ابتدایی) تا ۷ است. در فرایند اجرا طول این لیست به تدریج به مقدار ۷ افزایش می‌یابد.

شرایط رضایت: اگر چه لیست ممنوع سبب می‌شود که حرکت‌های (i,k) برای مدتی ممنوع گردد اما این عمل علاوه بر اینکه از دور جلوگیری می‌کند ممکن است سبب شود که جواب‌های باکیفیتی نیز الگوریتم از دست بدهد. به عبارت دیگر این عمل سبب می‌شود که مدیریت لیست آسان‌تر شود علیرغم اینکه جستجو محدودتر از حالتی است که تمام جواب ذخیره شود. زیرا ممکن است جواب‌های دیگری که شامل این ویژگی‌ها هستند و قبلاً مشاهده نشده‌اند نیز، ممنوع گردند. بنابراین برای هموار کردن این مشکل شرایط رضایت تعریف می‌شود که سبب می‌گردد برخی از جواب‌ها که در لیست ممنوع قرار دارند، زودتر از موعد مقرر از لیست خارج شوند. دو نمونه از معیارهای رضایت که در الگوریتم پیشنهادی از آن استفاده شده است به شرح زیر می‌باشد:

۱. اگر حرکت به عضوی، منجر به جوابی با مقدار هزینه بهتر از بهترین جواب تاکنون بدست آمده باشد، ان گاه ان عضو از لیست ممنوع خارج می‌شود

[8] Aspiration condition
[9] Reactive tabu search

۲. اگر ساختار روش و لیست‌های ممنوع در مرحله‌ای از الگوریتم به گونه‌ای باشد که امکان هیچ گونه حرکت به عضوی وجود نداشته باشد. در این صورت عضوی که از همه به خروج نزدیک‌تر است از لیست خارج می‌شود.

مکانیزم تشدید[10]: برای یافتن جواب‌های با کیفیت بالاتر در روش‌های فراابتکاری، فرآیند جستجو باید قادر باشد به صورت هوشمندانه، بخش‌هایی از فضای جواب که شامل جواب‌های با کیفیت خوب است را خیلی عمیق‌تر مورد جستجو قرار دهد. این مفهوم در الگوریتم جستجوی ممنوع، با مکانیزم تشدید پیاده می‌شود. تشدید در جستجوی ممنوع، به مکانیزمی اطلاق می‌شود که بر اساس آن جواب‌ها و انتقال‌هایی که منجر به جواب‌های خوب شده‌اند، تقویت می‌شوند. به عبارتی دیگر، این استراتژی به بازگشت به جواب‌های نخبه و جستجوی بیشتر در محدوده آن‌ها، اشاره دارد. بنابراین به ابزاری برای مشخص کردن مجموعه جواب‌های نخبه نیاز دارد تا پایه‌ای برای ترکیب خصوصیت‌های خوب و ساختن جواب‌های جدید خوب در دست داشته باشد.

در الگوریتم پیشنهادی این عمل به وسیله الگوریتم جستجوی محلی دوگانه صورت می‌گیرد و هنگامی فعال می‌شود که بهترین جواب یدست آمده ارتقا یابد. این روش بر اساس حذف کردن دو یال غیر همسایه از تور و دوباره متصل‌کردن آن دو یال از طریقی دیگر، کار می‌کند. باید توجه کرد که برای متصل‌کردن مسیر و ایجادکردن دوباره تور چندین راه وجود دارد اما فقط حالتی پذیرفته می‌شود که در محدودیت‌های مسئله، که در بالا به آن اشاره شد، صدق کند و تور جدید مقدار بهتری را برای مسئله نسبت به جواب قبلی بدست آورد. توجه به این نکته ضروری است که عمل حذف دو کمان و متصل‌کردن دوباره آن‌ها به طور متوالی تا جایی ادامه پیدا می‌کند که دیگر هیچ حرکت بهبوددهنده جدیدی برای الگوریتم یافت نشود.

شرایط توقف الگوریتم: در الگوریتم پیشنهادی شرط پایانی اجرای الگوریتم به تعداد دو برابر تعداد گره‌های هر مثال در نظر گرفته شده است. باید توجه داشت که شرایط دیگری برای اتمام الگوریتم‌های فراابتکاری وجود دارند که در اینجا در نظر گرفته نشدند. بعضی از این شرایط می‌تواند تکرار بهترین جواب به تعدادی معین یا اجرای الگوریتم در زمانی معین باشد.

۳- نتایج و بحث

در این بخش، نتایج محاسباتی الگوریتم پیشنهادی با جزئیات بیشتری ارائه می‌گردد و نتایج آن با سایر روش‌های مشهور فراابتکاری مورد مقایسه قرار می‌گیرد. توجه به این نکته ضروری است که الگوریتم مربوطه به وسیله زبان *Matlab* نوشته شده و روی کامپیوتری با ۲ گیگا بایت حافظه و Mhz ۲۴۰۰ قدرت پردازشگر اجرا می‌گردد. برای مقایسه کامل‌تر الگوریتم پیشنهادی، کارایی آن را روی ۱۴ مثال استاندارد تولید شده توسط کریستوفیدز اجرا می‌گردد. این مسائل در آدرس اینترنتی زیر وجود دارند.

URL:http:// mscmga.ms.ic.ac.uk/jeb/orlib/vrpinfo.html

باید توجه کرد که این مثال‌ها بدین جهت مورد بررسی قرار گرفته‌اند که اولاً دارای یک ترکیب مناسب از مسائل هستند که بازه‌ای بین ۵۰ تا ۱۹۹ گره را، بدون انبار کالا، در بر می‌گیرند و ثانیاً الگوریتم‌های زیادی بر روی این مثال‌ها مورد آزمایش قرار گرفته‌اند و بدین جهت می‌توان مقایسه مناسبی بین الگوریتم پیشنهادی و الگوریتم‌های دیگر انجام داد تا بدین وسیله کارایی الگوریتم با دقت بیشتری مورد بررسی قرار گیرد. همچنین باید اضافه کرد که در بین این ۱۴ مثال، ۱۰ مثال ابتدایی شامل مشتری‌هایی هستند که به طور تصادفی در اطراف انبار توزیع شده‌اند، در حالی که در ۴ مثال باقیمانده مشتری‌ها در کلاس‌ترهایی که در آن انبار کالا در مرکز واقع نشده است، قرار گرفته شده‌اند. از طرف دیگر همه مثال‌های در نظر گرفته شده دارای محدودیت ظرفیت بوده ولی مثال‌های *C6-C10, C13, C14* علاوه بر داشتن محدودیت ظرفیت، دارای محدودیت‌های طول مسیر و زمان سرویس‌دهی نیز هستند. خصوصیات این مثال‌ها در جدول ۱ با جزئیات کامل نشان داده شده است. لازم به ذکر است که در این جدول ستون‌های هفتم و هشتم به ترتیب تعداد وسیله‌های نقلیه مورد استفاده توسط الگوریتم پیشنهادی (PA) و جواب‌های بدست آمده را نشان می‌دهد. برای اطلاع از جزئیات جواب‌های بدست آمده می‌توان به پیوست مراجعه کرد.

[10] Intensification

جدول شماره (۱): خصوصیات مثال‌های مورد آزمایش

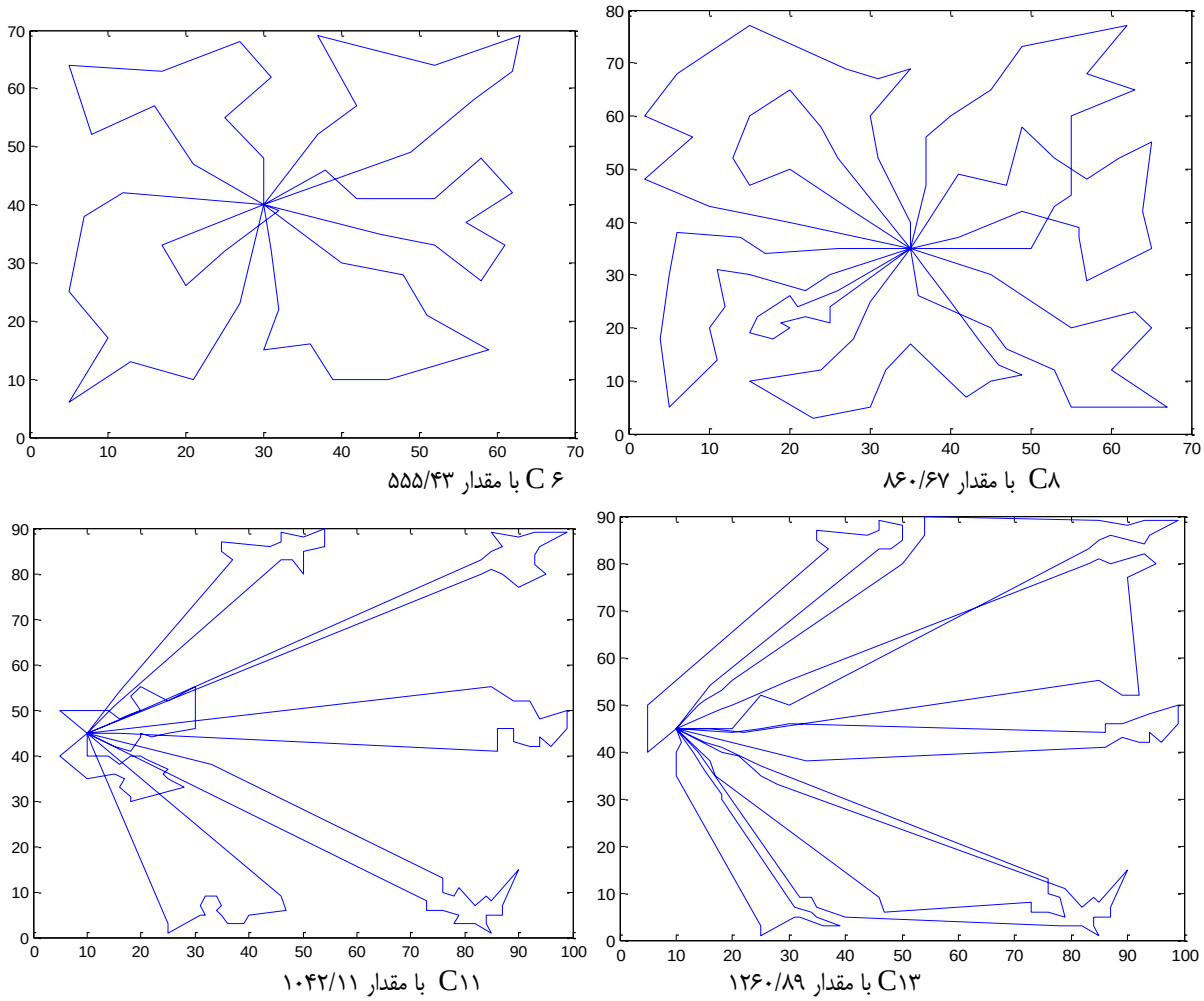
مثال	تعداد گره	تعداد وسیله نقلیه	ظرفیت	ماکزیمم طول هر تور	زمان سرویس‌دهی	m-PA	PA
C۱	۵۰	۵	۱۶۰	∞	۰	۵	۵۲۴/۶۱
C۲	۷۵	۱۰	۱۴۰	∞	۰	۱۰	۸۳۵/۲۶
C۳	۱۰۰	۸	۲۰۰	∞	۰	۸	۸۲۶/۱۴
C۴	۱۵۰	۱۲	۲۰۰	∞	۰	۱۲	۱۰۶۶/۸۹
C۵	۱۹۹	۱۷	۲۰۰	∞	۰	۱۷	۱۳۱۱/۳۵
C۶	۵۰	۶	۱۶۰	۲۰۰	۱۰	۶	۵۵۵/۴۳
C۷	۷۵	۱۱	۱۴۰	۱۶۰	۱۰	۱۱	۹۰۹/۶۸
C۸	۱۰۰	۹	۲۰۰	۲۳۰	۱۰	۹	۸۶۰/۶۷
C۹	۱۵۰	۱۴	۲۰۰	۲۰۰	۱۰	۱۲	۱۰۸۰/۳۰
C۱۰	۱۹۹	۱۸	۲۰۰	۲۰۰	۱۰	۱۷	۱۴۰۱/۶۵
C۱۱	۱۲۰	۷	۲۰۰	∞	۰	۷	۱۰۴۲/۱۱
C۱۲	۱۰۰	۱۰	۲۰۰	∞	۰	۱۰	۸۱۹/۵۶
C۱۳	۱۲۰	۱۱	۲۰۰	۷۲۰	۵۰	۷	۱۲۶۰/۸۹
C۱۴	۱۰۰	۱۱	۲۰۰	۱۰۴۰	۹۰	۱۱	۸۶۶/۳۷

در جدول ۲ الگوریتم پیشنهادی با تعدادی از بهترین روش‌های موجود برای مسئله *VRP* مورد مقایسه قرار گرفته شده است. در اینجا *SA* و *TS* نشان دهنده روش‌های شبیه‌سازی آنیلی و جستجوی ممنوع عثمان (۱۹۹۳)، *GA* روش الگوریتم ژنتیک ارائه شده توسط بکر و ایچو (۲۰۰۳)، *ACO* و *SS_ACO* به ترتیب نشان دهنده روش الگوریتم مورچگان بدون داشتن جستجوی محلی و روش ترکیبی الگوریتم مورچگان به همراه روش جستجوی محلی پراکنده به وسیله زانگ و تانک (۲۰۰۹)، *PSO* روش الگوریتم پرندگان به وسیله آی و کاچیتویش یانوکال (۲۰۰۹)، *GAP* روش ترکیبی الگوریتم پرندگان و ژنتیک به وسیله مارینا‌کیس و مارینا‌کی (۲۰۱۰) و در نهایت *PA* نشان دهنده الگوریتم پیشنهادی می‌باشد. همچنین برای مقایسه بهتر این الگوریتم‌ها، بهترین جواب‌هایی که تاکنون برای هر کدام از مثال‌ها بدست آمده است، با نماد *BA* نشان داده شده است.

جدول شماره (۲): مقایسه الگوریتم پیشنهادی با بهترین الگوریتم‌های فراابتکاری

مثال	SA	TS	GA	SS_ACO	PSO	ACO	PGA	PA	BA
C۱	۵۲۸	۵۲۴	۵۲۴/۸۱	۵۲۴/۸۱	۵۲۴/۸۱	۵۳۳/۱۵	۵۲۴/۸۱	۵۲۴/۶۱	۵۲۴/۶۱
C۲	۸۳۸	۸۴۴	۸۴۹/۷۷	۸۳۵/۲۶	۸۴۴/۴۲	۸۶۰/۲۶	۸۳۵/۲۶	۸۳۵/۲۶	۸۳۵/۲۶
C۳	۸۲۹	۸۳۵	۸۴۰/۷۲	۸۳۰/۱۴	۸۲۹/۴۰	۸۴۰/۵۱	۸۲۶/۱۴	۸۲۶/۱۴	۸۲۶/۱۴
C۴	۱۰۵۸	۱۰۵۲	۱۰۵۵/۸۵	۱۰۳۸/۲۰	۱۰۴۸/۸۹	۱۰۷۳/۰۰	۱۰۲۸/۴۲	۱۰۶۶/۸۹	۱۰۲۸/۴۲
C۵	۱۳۷۶	۱۳۵۴	۱۳۷۸/۷۳	۱۳۰۷/۱۸	۱۳۲۳/۸۹	۱۳۵۰/۲۴	۱۲۹۴/۲۱	۱۳۱۱/۳۵	۱۲۹۱/۴۵
C۶	۵۵۵	۵۵۵	۵۶۰/۲۹	۵۵۹/۱۲	۵۵۵/۴۳	۵۶۶/۰۰	۵۵۵/۴۳	۵۵۵/۴۳	۵۵۵/۴۳
C۷	۹۰۹	۹۱۳	۹۱۴/۱۳	۹۱۲/۶۸	۹۱۷/۶۸	۹۲۷/۰۰	۹۰۹/۶۸	۹۰۹/۶۸	۹۰۹/۶۸
C۸	۸۶۶	۸۶۶	۸۷۲/۸۲	۸۶۹/۳۴	۸۶۷/۰۱	۸۸۹/۴۵	۸۶۵/۹۴	۸۶۰/۶۷	۸۶۵/۹۴
C۹	۱۱۶۴	۱۱۸۸	۱۱۹۳/۰۵	۱۱۷۹/۴	۱۱۸۱/۱۴	۱۱۹۸/۱۲	۱۱۶۳/۴۱	۱۰۸۰/۳۰	۱۱۶۲/۵۵
C۱۰	۱۴۱۸	۱۴۲۲	۱۴۸۳/۰۶	۱۴۱۰/۲۶	۱۴۲۸/۴۶	۱۴۵۱/۰۰	۱۳۹۷/۵۱	۱۴۰۱/۶۵	۱۳۹۵/۸۵
C۱۱	۱۱۷۶	۱۰۴۲	۱۰۶۰/۲۴	۱۰۴۴/۱۲	۱۰۵۱/۸۷	۱۰۷۳/۰۰	۱۰۴۲/۱۱	۱۰۴۲/۱۱	۱۰۴۲/۱۱
C۱۲	۸۲۶	۸۱۹	۸۷۷/۸	۸۲۴/۳۱	۸۱۹/۵۶	۸۴۶/۰۰	۸۱۹/۵۶	۸۱۹/۵۶	۸۱۹/۵۶
C۱۳	۱۵۴۵	۱۵۴۷	۱۵۶۲/۲۵	۱۵۵۶/۵۲	۱۵۴۶/۲۰	۱۶۱۳/۰۰	۱۵۴۴/۵۷	۱۲۶۰/۸۹	۱۵۴۱/۱۴
C۱۴	۸۹۰	۸۶۶	۸۷۲/۳۴	۸۷۰/۲۶	۸۶۶/۳۷	۸۹۰/۰۰	۸۶۶/۳۷	۸۶۶/۳۷	۸۶۶/۳۷

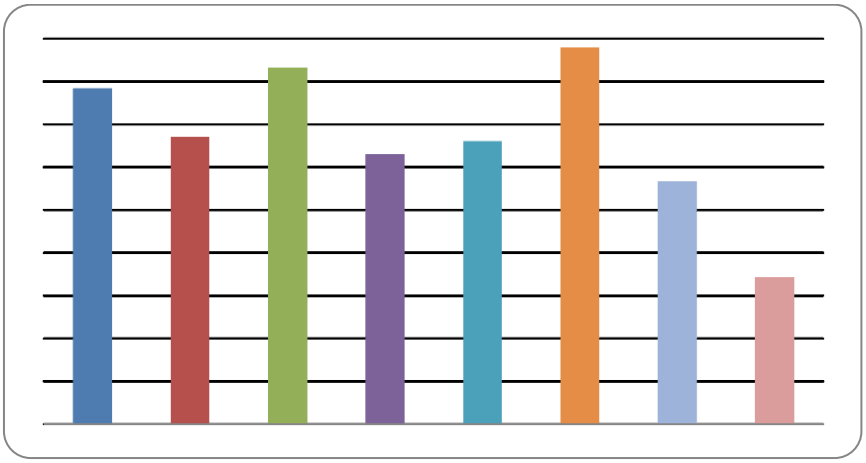
با مقایسه نتایج بدست آمده می‌توان نتیجه گرفت که الگوریتم PA یک روش کاملاً رقابتی برای حل این مسئله محسوب می‌شود که توانسته است به جواب‌های باکیفیتی دست پیدا کند به طوری که از بین ۱۴ مثال آزمایش شده به ۱۱ تا از بهترین جواب‌های تاکنون بدست آمده، دست یافته است. به طور نمونه چهار جواب نهایی که به وسیله الگوریتم پیشنهادی بدست آمده است در شکل ۳ نشان داده شده است. الگوریتم PA در مقایسه با الگوریتم ژنتیک، مورچگان بدون جستجوی محلی و شبیه‌سازی آنیلی توانسته به جواب‌های بسیار بهتری دست پیدا کند بطوری‌که مقدار متوسط الگوریتم پیشنهادی برای این ۱۴ مثال برابر با ۹۵۴/۳۵ است که در مقایسه با مقادیر متوسط ۱۰۰۳.۲۸ برای الگوریتم ژنتیک، ۹۹۸/۴۳ برای الگوریتم شبیه‌سازی آنیلی و ۱۰۰۷/۹۱ برای الگوریتم مورچگان بدون جستجوی محلی مقدار بسیار خوبی است و اختلاف بسیار زیادی با این روش‌ها دارد (شکل ۴). از طرف دیگر این الگوریتم در مقایسه با روش‌های پرندگان، ترکیبی الگوریتم پرندگان و ژنتیک ، ترکیبی مورچگان و جستجوی پراکنده و جستجوی ممنوع نیز دارای رقابت خوبی است و علیرغم اینکه نتوانسته است که اختلاف قابل توجهی با آن‌ها داشته باشد (کمتر از سه درصد) اما توانسته جواب‌های بهتری نسبت به مقادیر متوسط این چهار الگوریتم که به ترتیب ۹۸۶/۰۸، ۹۷۶/۶۷، ۹۸۲/۹۷ و ۹۸۷/۶۴ است، بدست آورد. بعلاوه این روش در بیشتر مثال‌ها جواب‌های بهتری نسبت به این چهار الگوریتم ارائه کرده است. بنابراین الگوریتم پیشنهادی در مقایسه با سایر روش‌ها توانسته است که جواب‌های بسیار باکیفیتی برای این دسته از مثال‌ها بدست آورده به نحوی که بهترین الگوریتم در بین این الگوریتم‌ها است.



C۶ با مقدار ۵۵۵/۴۳ C۸ با مقدار ۸۶۰/۶۷

C۱۱ با مقدار ۱۰۴۲/۱۱ C۱۳ با مقدار ۱۲۶۰/۸۹

شکل شماره (۳): چندین جواب بدست آمده بوسیله الگوریتم پیشنهادی



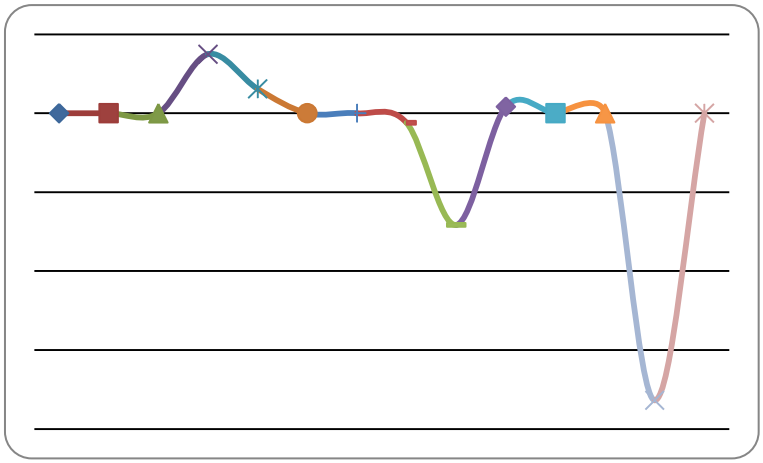
شکل شماره (۴): مقایسه میانگین جواب‌های بدست آمده توسط الگوریتم پیشنهادی با دیگر الگوریتم‌ها

بعلاوه برای اینکه تحلیل حساسیت الگوریتم نسبت به افزایش اندازه مسئله معلوم شود، در شکل ۵، درصد انحراف جواب‌های بدست آمده الگوریتم پیشنهادی نسبت به بهترین جواب‌های بدست آمده، آورده شده است. درصد انحراف نسبت به بهترین جواب شناخته‌شده از فرمول (۱۲) بدست می‌آید.

(۵)

$$100\times[c(s^{*})-c(s^{**})]/c(s^{**})$$

در این فرمول s^{*} بهترین جواب پیدا شده به وسیله الگوریتم مورد نظر برای مثال داده شده و s^{**} جواب BA برای همان مثال است. از طرف دیگر باید توجه کرد که اگر این مقدار منفی بدست آید بدین معنی است که الگوریتم پیشنهادی توانسته بهترین جواب تاکنون بدست آمده را ارتقا دهد. در این شکل محور افقی نشان دهنده ۱۴ مثال مورد آزمایش است در حالی که محور عمودی نشان‌دهنده درصد انحراف الگوریتم PA نسبت به بهترین جواب تاکنون بدست آمده است. باید توجه کرد که مثال‌های نشان داده شده در این شکل را می‌توان از نظر تعداد مشتری به سه دسته کمتر از ۱۰۰ (C۱ و C۶ با ۵۰ مشتری، C۲ و C۷ با ۷۵ مشتری)، مابین ۱۰۰ تا ۱۵۰ (C۳، C۸، C۱۲ و C۱۴ با ۱۰۰ مشتری و C۱۱ و C۱۳ با ۱۲۰ مشتری) و بالاتر از ۱۵۰ مشتری (C۴ و C۹ با ۱۵۰ مشتری و C۵ و C۱۰ با ۱۹۹ مشتری) تقسیم‌بندی نمود. بنابراین می‌توان نتایج الگوریتم را روی این سه دسته به طور مجزا مورد بررسی قرار داد.



شکل شماره (۵): تحلیل حساسیت الگوریتم پیشنهادی نسبت به افزایش اندازه مسئله

در دسته اول الگوریتم توانسته است به جواب‌های بسیار خوبی دست یابد و دارای موفقیت تقریباً کامل است زیرا توانسته است که در هر چهار مثال موجود به بهترین جواب‌های تاکنون یافته شده، دست پیدا کند. بنابراین می‌توان نتیجه گرفت که الگوریتم در مثال‌های کوچک از کارایی بسیار عالی برخوردار است. در دسته دوم همان طور که ذکر شد، شش مثال موجود است که در چهار مثال الگوریتم توانسته است که به بهترین جواب‌ها دست پیدا کند. به علاوه الگوریتم در $C8$ و $C13$ توانسته که بهترین جواب

تاکنون بدست آمده را ارتقا دهد. بنابراین می‌توان گفت که الگوریتم مانند دسته اول کارایی بسیار عالی برای حل مثال‌های متوسط دارد. نهایتاً در دسته سوم چهار مثال وجود دارد که دارای اندازه ۱۵۰ و ۱۹۹ هستند. در این مثال‌ها الگوریتم از کارایی نسبتاً خوبی برخوردار است علی‌رغم آنکه نسبت به دو دسته قبلی کیفیت کمتری دارد و در چهار مثال موجود توانسته است که در یکی از مثال‌ها به بهترین جواب‌هایی تاکنون بدست آمده، دست یابد. بنابراین می‌توان نتیجه گرفت که اگر چه الگوریتم در این دسته از مثال‌ها نسبت به دسته اول و دوم نتوانسته است که به بهترین جواب‌ها دست پیدا کند اما همان‌طور که گفته شد، توانسته است که این جواب‌ها را تقریباً با کمتر از چهار درصد انحراف تخمین بزند.

در این مقاله یک روش ترکیبی جستجوی ممنوع برای حل مسئله *VRP* ارائه گردید. همان‌طور که نشان داده شد این الگوریتم دارای رقابت بسیار خوبی با روش‌های مشهور فراابتکاری بوده و در مقایسه با این روش‌ها توانسته است که جواب‌های باکیفیتی تولید کند. به نظر می‌رسد که استفاده از روش‌های دیگر برای بدست آوردن جواب اولیه شدنی الگوریتم مورچگان می‌تواند باعث ارتقای جواب‌های بدست آمده شود. بعلاوه استفاده از الگوریتم‌های قوی‌تر محلی برای بدست آوردن جواب‌های بهتر می‌تواند راهکار مناسب دیگری برای افزایش کارایی الگوریتم مربوطه باشد. استفاده از این الگوریتم‌ها و بکارگیری این روش به گسترش‌های دیگر این مسئله مانند مسیریابی وسیله نقلیه باز و مسیریابی وسیله نقلیه همراه با دریافت و تحویل کالا به مقاله‌های بعدی موکول می‌شود.

سپاسگزاری

این مقاله نتیجه اجرای طرح پژوهشی تحت عنوان "ترکیب الگوریتم‌های جستجوی ممنوع و جمعیت مورچگان برای مسئله مسیریابی وسیله نقلیه" است که توسط دانشگاه آزاد اسلامی واحد رباط‌کریم مورد حمایت مادی و معنوی قرار گرفته است.

٤- منابع

1- Ai, T. J. & Kachitvichyanukul, V. (2009). A Particle Swarm Optimization for the Heterogeneous Fleet Vehicle Routing Problem. International Journal of Logistics and SCM Systems, 3(1), 32-39.

2- Anily S. & Mosheiov G. (1994). The traveling salesman problem with delivery and backhauls. Operations Research Letters, 16, 11-18.

3- Berger, J. & Barkaoui, M. (2003). A hybrid genetic algorithm for the capacitated vehicle routing problem. in: Proceedings of the International Genetic and Evolutionary Computation Conference – GECCO03, LNCS 2723, 646–656.

4- Chen, P., Huang, H. K. & Dong, X. Y. (2010). Iterated variable neighborhood descent algorithm for the capacitated vehicle routing problem. Expert Systems with Applications, 37(2), 1620-1627.

5- Christofides, N. (1985). Vehicle routing, in: E.L. Lawler, J.K. Lenstra, A.H.G. Rinnooy Kan and D.B. Shmoys, (eds.), The Traveling Salesman Problem. A Guided Tour of Combinatorial Optimization, Wiley, Chichester, 431-448.

6- Clarke, G. & Wright, J. W. (1964). Scheduling of vehicles from a central depot to a number of delivery points. Operations Research, 12, 568-581.

7- Cordeau, J. F., Laporte, G., & Mercier, A. (2001). A unified tabu search heuristic for vehicle routing problems with time windows. Journal of the Operational Research Society, 52, 928–936.

8- Dantzig, G. B., Fulketson, D.R. & Johnson, S.M. (1954). Solution of a large-scale traveling-salesman problem. Operations Research, 2, 393-410.

9- Dethloff, J. (2001). Vehicle routing & reverse logistics: the vehicle routing problem with simultaneous delivery and pick-up. Operations Research Spectrum, 23, 79-96.
10- Dorigo, M. & Ganbardella, L. (1997) .Ant colony system: a cooperative learning approach to the traveling salesman problem. IEEE trans. Evolutionary Computing, 53-66.
11- Du, L. & He, R. (2012). Combining Nearest Neighbor Search with Tabu Search for Large-Scale Vehicle Routing Problem. Physics Procedia, 25, 1536-1546.
12- Eilon, S., Watson-Gandy, C. D. T. & Christofides, N. (1971). Distribution Management: Mathematical Modelling & Practical Analysis, Griffin, London.
13- Fisher, M.L., & Jaikumar, R. (1978). A decomposition algorithm for large-scale vehicle routing. Working Paper 78-11-05, Department of Decision Sciences, University of Pennsylvania.
14- Garey, M. R. & Johnson, D. S. (1979). Computers and Intractability: A Guide to the Theory of NP completeness. W.H. Freeman & Co., New York.
15- Glover, F. (1986). Future Paths for Integer Programming and Links to Artificial Intelligence. Computers and Operations Research, 13, 533-549.
16- Golden, B. L., & Assad, A. A. (1988). Vehicle Routing: Methods and Studies. North-Holland, Amsterdam.
17- Halskau, K. & Lokketangen, A. (1998). Analyse av distribusjonsopplegget ved Sylte Mineralvannfabrikk A/S, Research Report M9812. More Research, Molde (in Norwegian).
18- Hansen, P. (1986). The Steepest Ascent Mildest Descent Heuristic for Combinatorial Programming, capri, Italy: Presented at the Congress on Numerical Methods in Combinatorial Optimization.
19- Hemmelmayr, V. C., Cordeau, J. F. & Crainic, T. G. (2012.) An adaptive large neighborhood search heuristic for Two-Echelon Vehicle Routing Problems arising in city logistics. Computers & Operations Research, 39(12), 3215-3228.
20- Imran, A., Salhi, S. & Wassan, N. A. (2009). A variable neighborhood-based heuristic for the heterogeneous fleet vehicle routing problem, European Journal of Operational Research. 197(2), 509-518.
21- Jaszkiewicz, A. & Kominek, P. (2003). Genetic local search with distance preserving recombination operator for a vehicle routing problem. European Journal of Operational Research, 151, 352-364.
22- Laporte, G., Louveaux, F. & Mercure, H. (1992). The vehicle routing problem with stochastic travel times. Transportation Science, 26(3), 161-170.
23- Marinakis, Y. & Marinaki, M. (2010). A hybrid genetic – Particle Swarm Optimization Algorithm for the vehicle routing problem. Expert Systems with Applications, 37(2), 1446-1455.
24- Martello, S. & Toth, P. (1990). Generalized assignment problem, in: S. Martello and P. Toth (eds.), Knapsack Problems. Algorithms and Computer Implementations, Wiley, Chichester, . 189-220.
25- Min, H. (1989). The multiple vehicle routing problem with simultaneous delivery and pick-up points. Transportation Research A 23A, . 377-386.
26- Osman, L. H. (1993). Metastrategy simulated annealing and tabu search algorithms for the vehicle routing problem. Annal Operations Research, 41, 421–451.
27- Prins, C. (2009). Two memetic algorithms for heterogeneous fleet vehicle routing problems. Engineering Applications of Artificial Intelligence, 22, 916–928.
28- Prive, J., Renaud, J., Boctor, F.F. & Laporte, G. (2006). Solving a vehicle routing problem arising in soft drink distribution. Journal of the Operational Research Society, 57(9), 1045-1052.

29- Santos, L., Coutinho-Rodrigues, J. & Current, J. R. (2010). An improved ant colony optimization based algorithm for the capacitated arc routing problem. Transportation Research Part B: Methodological, 44(2), 246-266.
30- Stewart, W.R. & Golden, B.L. (1983). Stochastic vehicle routing: A comprehensive approach. European Journal of Operational Research, 14, 371-385.
31- Tang, F.A. & Galvao, R. D. (2006). A tabu search algorithm for the vehicle routing problem with simultaneous pick-up and delivery service. Computers & Operations Research, 33, 595-619.
32- Tang, J., Zhang, J. & Pan, Z. (2010). A scatter search algorithm for solving vehicle routing problem with loading cost. Expert Systems with Applications, 37(6), 4073–4083.
33- Toth, P. & Vigo, D. (2002). The Vehicle Routing Problem, SIAM monographs on discrete mathematics and applications.
34- Wasner, M. & Zaphel, G. (2004). An integrated multi-depot hub-location vehicle routing model for network planning of parcel service. International Journal of Production Economics, 90, 403-419.
35- Werra, D. & Hertz, A. (1989). Tabu Search Techniques: A tutorial and an application to neural networks. OR Spektrum, 131-141.
36- Yousefikhoshbakht, M., Didehvar, F., Rahmati, R. & Saadati Eskandari, Z. (2012). A hybrid ant colony system for the heterogeneous fixed fleet vehicle routing problem. Transportation Research Journal, 9(2), 191-207.
37- Yousefikhoshbakht, M., Didehvar, F., Rahmati, R. & Sedighpour, M. (2012). An effective imperialist competitive algorithm for solving the open vehicle routing problem. Transportation Research Journal, 9(1), 83-95
38- Yurtkuran, A. & Emel, E. (2010). A new Hybrid Electromagnetism-like Algorithm for capacitated vehicle routing problems. Expert Systems with Applications, 37(4), 3427-3433.
39- Zachariadis, E. E., Tarantilis, C. D. & Kiranoudis, C. T. (2009). A Guided Tabu Search for the Vehicle Routing Problem with two-dimensional loading constraints. European Journal of Operational Research, 195(3), 729-743.
40- Zhang, X. & Tang, L. (2009). A new hybrid ant colony optimization algorithm for the vehicle routing problem. Pattern Recognition Letters, 30(9), 848-855.