

# یافتن کوتاه‌ترین تور همیلتونی ایران با استفاده از ترکیب الگوریتم سیستم اجتماع

## مورچه‌ها و جستجوی محلی

کیوان قصیری، استادیار، دانشکده مهندسی راه آهن، دانشگاه علم و صنعت ایران، تهران، ایران  
حسن سرحدی، دانشجوی کارشناسی ارشد، دانشکده مهندسی راه آهن، دانشگاه علم و صنعت ایران، تهران، ایران  
Email: ghoseiri@iust.ac.ir

### چکیده

واژه‌های کلیدی: مسأله فروشنده دوره‌گرد، کوتاه‌ترین تور همیلتونی ایران، بهینه‌سازی ترکیباتی، اجتماع مورچه‌ها، جستجوی محلی.

#### ۱. مقدمه

این مسئله می‌تواند به مرجع [۲] مراجعه کنند. مسأله فروشنده دوره‌گرد یکی از نخستین مسائلی است که NP-Complete بودن آن به اثبات رسیده است [۳] و بنابر این نمی‌توان الگوریتمی یافت که در زمان چندجمله‌ای آنرا به شکل قطعی حل کند. مسائل عمومی مسیریابی عموماً در ارتباط با حمل‌ونقل محصولات و یا سفر بر روی مسیرهایی در یک شبکه مفروض با روشی بهینه هستند. موارد زیر از جمله مسائل عمومی مسیریابی هستند:

- توزیع کالاها و خدمات از دیوها به محل مشتریان
  - برنامه‌ریزی تولید و توزیع محصولات
  - برنامه‌ریزی انبار و توزیع کالاها از دیوها به نمایندگی‌های فروش
  - زمان‌بندی خدمه
  - حمل و نقل کالاهای نیمه ساخته بین مکان‌های تولید
- چنین مسائلی همه روزه در مدیریت و برنامه‌ریزی عملیاتی و مسائل نقلیه و خدمه در شبکه رخ می‌دهند. مبداء و منشاء چنین

مسأله فروشنده دوره‌گرد<sup>۱</sup> یکی از معروف‌ترین و پرکاربردترین مسائل در بهینه‌سازی ترکیباتی<sup>۲</sup> است. بیان و شرح این مسأله بسیار ساده، اما حل کردن آن بسیار دشوار است. در این مسأله، فروشنده‌ای دوره‌گرد می‌خواهد کم هزینه‌ترین ترتیب بازدید مجموعه‌ای از شهرها را با شروع از یک شهر و بازگشت به همان شهر پیدا کند، به طوری که بقیه شهرها فقط یکبار بازدید شوند [۱]. مسأله فروشنده دوره‌گرد می‌تواند مقارن<sup>۳</sup> یا غیر مقارن<sup>۴</sup> باشد که در حالت مقارن فاصله دو شهر به جهت حرکت بستگی ندارد. علاوه بر فروشنده دوره‌گرد معمولی یا کلاسیک که در بالا تعریف شد، این مسأله دارای گونه‌های متعددی است که از آن جمله می‌توان به فروشنده دوره‌گرد با پنجره زمانی<sup>۵</sup>، فروشنده دوره‌گرد دوره‌ای<sup>۶</sup>، فروشنده دوره‌گرد سیاه و سفید<sup>۷</sup>، فروشنده دوره‌گرد انتخابی<sup>۸</sup>، فروشنده دوره‌گرد با هدف متحرک<sup>۹</sup> و ... اشاره کرد. خوانندگان علاقه‌مند برای آشنایی با گونه‌های

عملی به شکل یک مسأله فروشنده دوره‌گرد مدل‌سازی و حل شوند. این در حالی است که می‌توان این گونه مسائل را مستقیماً هم مدل‌سازی و حل کرد. اما گستردگی مطالعات در مورد مسأله فروشنده دوره‌گرد و در نتیجه وجود الگوریتم‌های موثر در حل آن باعث شده که محققین در حل بسیاری از مسائل عملی به این شکل از مسأله فروشنده دوره‌گرد استفاده کنند [5-7].

یافتن کوتاه‌ترین تور همیلتونی کشورها، یکی از کاربردهای مسأله فروشنده دوره‌گرد است که هم در زمینه نظری و هم در زمینه عملی بسیار حائز اهمیت است.

علاوه بر کاربردهای عملی که یافتن تور همیلتونی کشورها دارد (مثلاً یافتن کوتاه‌ترین تور همیلتونی بین ۵۳۲ سوئیس مخابراتی در کشور آمریکا برای شرکت AT&T [7])، دانشمندان تحقیق در عملیات و علوم رایانه ای عموماً دو هدف زیر را از یافتن تور همیلتونی کشورها دنبال می‌کنند:

- آزمایش ایده‌ها و الگوریتم‌های جدید در حل (قطعی یا تقریبی) این مسأله

- نشان دادن توانمندی در حل این مسأله و استفاده موثر از این مسأله در کاربردهای عملی آن.

این نحوه نگرش به مسأله فروشنده دوره‌گرد، از نخستین روزهای ظهور علم تحقیق در عملیات آغاز شده و تاکنون هم ادامه یافته و قطعاً در آینده هم ادامه خواهد داشت. جدول ۲ نشان دهنده تلاش دانشمندان و محققان در یافتن کوتاه‌ترین تور همیلتونی کشورهای مختلف در طی نیم قرن اخیر است.

در این جدول، مسأله Dantzig42 با ۴۹ شهر نخستین مسأله فروشنده دوره‌گرد بود که در سال ۱۹۵۴ توسط دانزیگ و همکارانش حل شد. مسائل D15112 و Sw24798 از جمله بزرگ‌ترین مسائل موجود در کتابخانه مسأله فروشنده دوره‌گرد<sup>۱</sup> هستند که به ترتیب شامل ۱۵۱۱۲ شهر در آلمان و ۲۴۷۹۸ شهر در سوئد می‌شوند. تور بهینه آنها به ترتیب در حدود ۶۶۰۰۰ و ۷۲۵۰۰ کیلومتر طول دارد. محاسبات این دو مسأله به ترتیب بر روی شبکه‌ای از ۱۱۰ پردازنده موازی در دانشگاه پرینستون (با صرف ۲۲/۶ سال زمان محاسباتی) و شبکه‌ای از ۹۶ پردازنده موازی Xeon 2.8 GHz در دانشکده مهندسی صنایع و سیستم‌های دانشگاه، جورجیا (با صرف ۸۴٫۴ سال زمان محاسباتی) انجام شدند. شکل ۱ تور بهینه این دو مسأله را نشان می‌دهد.

مسائلی، مسئله فروشنده دوره‌گرد کلاسیک است [۴]. از آنجا که هزینه‌های توزیع فیزیکی کالاها حجم قابل توجهی از هزینه‌های توزیع سالانه کالاها را در اقتصاد هر کشوری به خود اختصاص می‌دهند (به عنوان مثال در سال ۱۹۸۳، هزینه‌های توزیع کالاها در حدود ۴۰۰ میلیارد دلار در ایالات متحده آمریکا تخمین زده شده است، که حدود ۱۶٪ از آنها، هزینه‌های توزیع فیزیکی بوده است [۴])، هرگونه کاهش در این هزینه‌ها از طریق بهبود روش‌ها و الگوریتم‌های حل مسائل مسیریابی، منجر به صرفه جویی قابل توجهی شده و منافع زیادی را به مصرف‌کننده نهایی می‌رساند. در این مقاله برای نخستین بار، مسأله فروشنده دوره‌گرد متقارن (یافتن کوتاه‌ترین تور همیلتونی) برای ۳۶۰ نقطه ایران با دو روش (الگوریتم سیستم اجتماع مورچه‌ها و الگوریتم پیشنهادی ترکیب سیستم اجتماع مورچه‌ها و جستجوی محلی) حل شده و سپس نتایج حاصل از آنها مقایسه و در نهایت پیشنهاداتی برای بهبود کیفیت جواب‌ها ارائه می‌شود.

## ۲. اهمیت مسأله و پیشینه مطالعات

اهمیت مسأله فروشنده دوره‌گرد در دو زمینه تجلی یافته است؛ کاربردهای عملی و کاربردهای نظری. در زمینه کاربردهای عملی، مسأله فروشنده دوره‌گرد و گونه‌های مختلف آن را می‌توان از پرکاربردترین مسائل در بهینه‌سازی ترکیباتی دانست. همچنان که از نام این مسأله انتظار داریم، نخستین کاربردهای آن در حمل و نقل بوده و در ادامه کاربردهایی در تولید صنعتی پیدا کرده است. به تدریج کاربردهایی از این مسأله در علوم جدیدتر هم به وجود آمده‌اند که از جمله می‌توان به کاربردهای آن در بیوانفورماتیک اشاره کرد. جدول ۱ برخی از کاربردهای عملی مسأله فروشنده دوره‌گرد را نشان می‌دهد.

اما در زمینه کاربردهای نظری، مسأله فروشنده دوره‌گرد به دلیل سادگی در بیان و مدل‌سازی و اینکه یکی از نخستین مسائلی بود که NP-Complete بودنش ثابت شد، به محلی برای آزمایش ایده‌های الگوریتمی جدید تبدیل شده است، به طوری که کارایی اکثر ایده‌های الگوریتمی جدید با اعمال آنها بر مسأله فروشنده دوره‌گرد تعیین می‌شود. نکته قابل توجه دیگر در مورد مسأله فروشنده دوره‌گرد این است که سادگی بیان، سادگی مدل‌سازی و وجود الگوریتم‌هایی که به طور موثری به حل این مسأله می‌پردازند در مجموع باعث شده است که بسیاری از مسائل

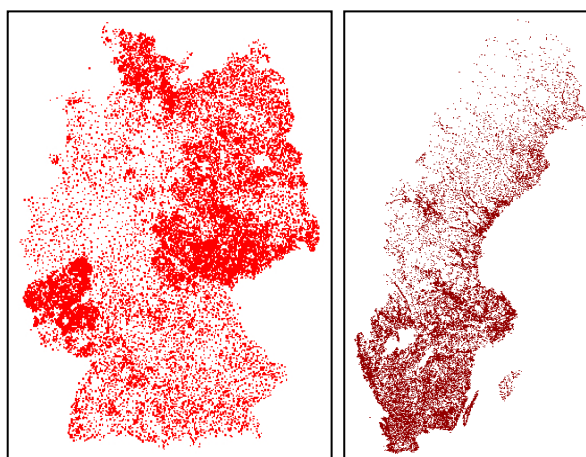
یافتن کوتاه‌ترین تور همیلتونی ایران با استفاده از ترکیب الگوریتم سیستم اجتماع مورچه‌ها و جستجوی محلی

جدول ۱. کاربردهای عملی مسأله فروشنده دوره‌گرد

نام مسأله	تعداد شهر	سال	مرجع
Dantzig42	49	1954	[۱۸و۶]
Gr120	120	1977	[۶]
Att532	532	1987	[۷و۶]
Usa13509	13509	1998	[۶]
D15112	15112	2001	[۶]
Sw24798	24798	2004	[۶]

جدول ۲. مهم‌ترین تورهای همیلتونی یافت شده کشورها

زمنه کاربردی	مثال	مرجع
حمل و نقل	توزیع درب به درب محصولات، مسیریابی وسائل نقلیه، زمان بندی خدمه، برنامه‌ریزی تخصیص لوکوموتیو، زمان‌بندی حرکت قطار، مسأله فروشنده دوره‌گرد راه آهن	[۴و۸و۹و۱۰و۱۱و۱۲]
تولید صنعتی	زمان‌بندی تولید کارگاهی <sup>۱۱</sup> ، زمان‌بندی کارگاه جریانی <sup>۱۲</sup> ، تولید مدارت مجتمع <sup>۱۳</sup>	[۵و۶و۸و۱۳]
بیو انفورماتیک	ایجاد دنباله‌های DNA، مسأله همترازی چندگانه دنباله‌ها (MSA)، ایجاد درخت‌های تکاملی	[۶و۴و۵و۱۶]
سایر زمین‌ها کاربردی	بهینه‌سازی ترتیب تصویر برداری از اشیای سماوی، مسیریابی خطوط انتقال نیرو، مسیریابی خطوط انتقال داده، کریستالوگرافی با اشعه ایکس	[۱و۶و۱۷]



شکل ۱. نمایش تور بهینه مسأله Sw24798 (سمت راست) و مسأله D15112 (سمت چپ) (برگرفته از [۶])

### ۳. مسأله فروشنده دوره گرد

#### ۱-۳ مدل سازی

#### ۱-۱-۳ مدل سازی عدد صحیح

مدل سازی مسأله فروشنده دوره گرد بر اساس برنامه ریزی عدد صحیح به این شکل است [۱۹]:

فرض کنیم  $n$  شهر به شکل رئوس گراف  $G = (V, E)$  وجود دارند و فاصله بین دو شهر  $i$  و  $j$  با  $d_{ij}$  نشان داده می شود. شهر آغازین با شماره ۱ و سایر شهرها به ترتیب با شماره های ۲ و ۳ ... شماره گذاری می شوند. متغیر دودویی  $x_{ij}$  را طوری تعریف می کنیم که  $x_{ij} = 1$  باشد. اگر فروشنده بلافاصله پس از شهر  $i$  به شهر  $j$  برود و در غیر این صورت صفر باشد. به علاوه، برای این که متغیر  $x_{ij}$  حتماً صفر فرض می کنیم که  $d_{ii}$  عدد مثبت بسیار بزرگی است. از آنجا که می خواهیم کل مسافت پیموده شده توسط فروشنده دوره گرد حداقل شود، تابع هدف را به شکل زیر تعریف می کنیم:

$$\text{Min} \sum_{i=1}^n \sum_{j=1}^n x_{ij} d_{ij} \quad (1)$$

برای اینکه از هر شهر دلخواه  $j$  فقط یک بار خارج شویم (شکل ۲) و فقط یکبار وارد شویم (شکل ۳)، روابط ۲ و ۳ را به این شکل تعریف می کنیم:

$$\sum_{i=1}^n x_{ji} = 1 \quad (2)$$

$$\sum_{i=1}^n x_{ij} = 1 \quad (3)$$

اگر چه در این دو رابطه  $x_{ij}$  و  $x_{ji}$  ظاهر می شوند، اما مطابق توضیحات داده شده مقدار آنها به دلیل حداقل بودن تابع هدف صفر خواهد شد. مسأله برنامه ریزی عدد صحیح که تا اینجا با روابط ۱، ۲، ۳ و توصیف شده است، همان مسأله تخصیص است که الگوریتم های مؤثری (مانند الگوریتم مجارستانی) برای حل آن وجود دارند. روابط ۲ و ۳ به تنهایی نمی توانند امکان ایجاد یک زیر تور یا حلقه را که در شکل ۴ دو نمونه از آن نشان داده شده است، از بین ببرند. یک زیر تور  $Q$  دارای این ویژگی است که هیچ یال  $(i, j)$  وجود ندارد که شهر  $i$  از  $Q$  را به شهر  $j$

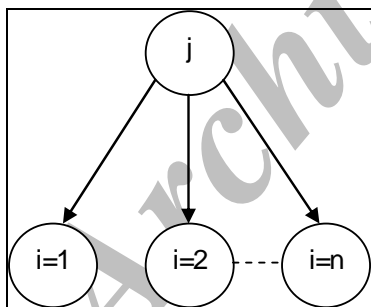
از  $Q = V - \bar{Q}$  متصل کند. برای از بین بردن امکان به وجود آمدن زیر تورها، محدودیت چهارم را به این شکل اضافه می کنیم که به ازای هر زیر مجموعه  $Q$  از  $V$ ، باید حداقل یک یال  $(i, j)$  وجود داشته باشد تا  $Q$  را به  $\bar{Q}$  متصل کند. به عبارت بهتر خواهیم داشت:

$$\bar{Q} = \{4,5,6\} \text{ و } Q = \{1,2,3\}$$

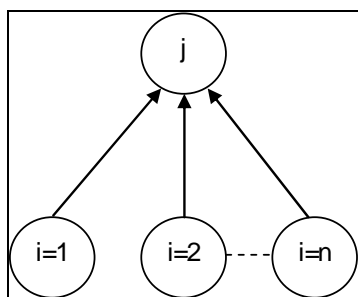
به عنوان مثال در شکل ۴ هیچ یالی بین این دو زیر مجموعه وجود ندارد. محدودیت دیگری که می توان آنرا به جای محدودیت چهارم بکار برد این است که در هر زیر مجموعه  $Q$  از  $V$  تعداد یالها باید کمتر از تعداد رئوس باشد تا امکان ایجاد حلقه در آن زیرمجموعه از بین برود. در شکل ۴ مشخص است که در هر دو زیرمجموعه تعداد رئوس و یالها مساوی هستند. بنابراین اگر  $|Q|$  تعداد رئوس یک زیرمجموعه از رئوس باشد، تعداد یالهای آن حداکثر می تواند  $|Q| - 1$  باشد. این محدودیت را می توان به این شکل نوشت:

$$\sum_{i \in Q} \sum_{j \in Q} x_{ij} \geq 1, \forall Q \subset V \quad (4)$$

$$\sum_{i \in Q} \sum_{j \in Q} x_{ij} \leq |Q| - 1 \quad (5)$$



شکل ۲. نمایش رابطه ۲



شکل ۳. نمایش رابطه ۳

نیاز دارد، الگوریتم از مرتبه  $O(n!) \times O(n) = O(n!)$  است که کاملاً بازدارنده و ناامیدکننده است.

با توجه به گستردگی فضای جواب این مسأله، نیاز به روش‌های آگاهانه شمارش برای احتراز از شمارش تمام فضای جواب ضروری است. این روش‌های آگاهانه بر دو دسته هستند. روش‌های قطعی و روش‌های تقریبی.

### ۳-۲-۱- روش‌های قطعی

الگوریتم‌های قطعی در حل این مسأله مبتنی بر مدل‌سازی برنامه‌ریزی خطی مخلوط، روش‌های انشعاب و تحدید<sup>۱۴</sup>، و انشعاب و برش<sup>۱۵</sup> هستند. این روش‌ها با بهره‌گیری از قدرت محاسباتی رایانه‌ها و صرف زمان و هزینه فراوان به یافتن جواب بهینه قطعی می‌پردازند. تمام مسائل بیان شده در جدول ۲، با این روش‌ها حل شده‌اند. الگوریتم‌های متعدد انشعاب و تحدید برای حل این مسأله پیشنهاد شده و تحقیقات در این زمینه ادامه دارد [۱ و ۲].

### ۳-۲-۲- روش‌های تقریبی

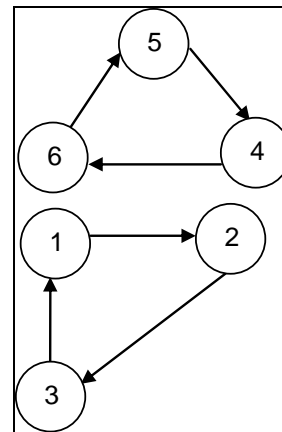
یکی از اشکالات الگوریتم‌های قطعی در حل مسأله فروشنده دوره‌گرد، زمان‌گیر و پرهزینه بودن آنها است (به زمان و هزینه صرف شده در حل مسائل D15112 و Sw24798 در بخش دو توجه کنید). این در حالی است که الگوریتم‌های ابتکاری<sup>۱۶</sup> یا تقریبی، جواب‌هایی نزدیک به بهینه را در مسائلی با ابعاد بزرگ در زمان قابل قبول ایجاد می‌کنند. الگوریتم‌های ابتکاری در حل این مسأله را به سه دسته کلی می‌توان تقسیم کرد [۱] و [۲]:

- ۱) الگوریتم‌های ابتکاری ایجادکننده تور: یک تور را به تدریج با اضافه کردن یک رأس در هر مرحله ایجاد می‌کنند.
- ۲) الگوریتم‌های ابتکاری بهبود دهنده تور: یک تور ایجاد شده را با تعویض جایگاه رئوس و کاهش هزینه، بهبود می‌بخشند.
- ۳) الگوریتم‌های ابتکاری ترکیبی: ابتدا یک تور را ایجاد کرده و سپس آنرا بهبود می‌دهند.

دسته‌ای دیگر از روش‌های تقریبی در حل این مسأله، الگوریتم‌های فوق‌ابتکاری<sup>۱۷</sup> هستند.

مهم‌ترین الگوریتم‌های فوق‌ابتکاری استفاده شده برای حل این مسأله عبارتند از:

- الگوریتم حریمان<sup>۱۸</sup> [۲ و ۲۱]



شکل ۴. زیرتورهای یک مسأله فروشنده دوره‌گرد

### ۳-۱-۲- مدل‌سازی در نظریه گراف

در این بخش با استفاده از اصطلاحات و مفاهیم نظریه گراف مسأله فروشنده دوره‌گرد مدل‌سازی می‌شود [۲].

فرض کنیم  $G = (V, A)$  یک گراف (جهت‌دار یا بدون جهت) است که  $V$  مجموعه رئوس و  $A$  مجموعه یال‌های آن است. برای هر یال  $e \in A$  یک هزینه (وزن)  $c_e$  وجود دارد. بدون دست دادن کلیت مدل‌سازی، می‌توانیم فرض کنیم گراف  $G$  یک گراف کامل است (هزینه یال‌هایی که برای تبدیل  $G$  به گراف کامل باید اضافه شوند را عدد مثبت بسیار بزرگی قرار می‌دهیم) آنگاه، مسأله فروشنده دوره‌گرد عبارت است از یافتن یک تور همیلتونی (توری که از هر رأس  $G$  یک بار عبور کرده و به رأس آغازین برگردد) در گراف  $G$  به طوری که مجموع هزینه یال‌های آن حداقل باشد.

در پایان، باید اشاره کرد که مدل‌سازی‌های متعددی از مسأله فروشنده دوره‌گرد ارائه شده که برخی از آنها از کاربرد این مسأله در یک زمینه خاص نتیجه شده است. برای آشنایی با سایر مدل‌سازی‌های این مسأله، خوانندگان علاقه‌مند را به مراجع [۲] و [۲۰] ارجاع می‌دهیم.

### ۳-۲-۳- روش‌های حل

مسأله فروشنده دوره‌گرد، مسأله‌ای شناخته شده در بهینه‌سازی ترکیباتی است که دارای مدل‌سازی ساده اما حلی دشوار است. یک رویکرد ناآگاهانه به این مسأله، شمارش تمام حالت‌های آن است. در یک مسأله با  $n$  رأس (شهر)، از آنجایی که  $(n-1)!$  حالت برای ایجاد تور وجود دارد و هر تور هم به زمان  $O(n)$

به سمت لانه برگردند. در نتیجه پس از گذشت مدتی، مقدار بیشتری از فرمون بر روی این مسیر باقی خواهد ماند. این موضوع باعث می‌شود که مورچه‌های بعدی هنگامی که به این محل می‌رسند تحت تأثیر مقدار بیشتر فرمون در مسیر پایینی قرار گرفته و مسیر پایینی را انتخاب کنند. این تمایل خود باعث افزایش بیش از پیش فرمون روی مسیر پایینی شده و سرانجام تمام مورچه‌ها در این محل از مسیر پایینی عبور خواهند کرد.

- الگوریتم‌های ژنتیک [۲۳ و ۲۴]
- جستجوی ممنوع [۲۴ و ۲۵]
- اجتماع مورچه‌ها [۲۵ و ۲۶ و ۲۷]
- شبیه‌سازی آنتیلینگ [۲۸]
- شبکه‌های عصبی مصنوعی [۲۹]

#### ۴. الگوریتم فوق ابتکاری اجتماع مورچه‌ها

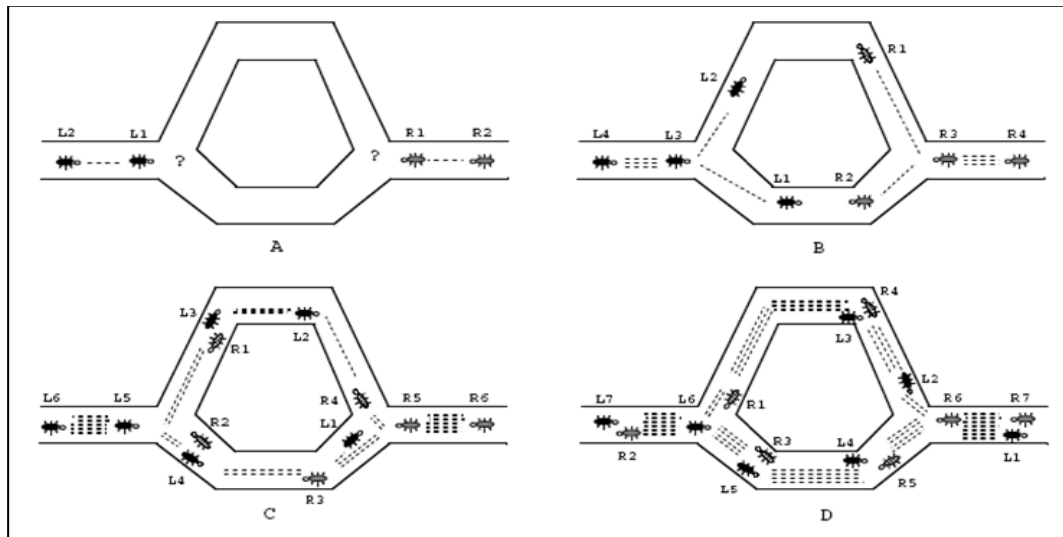
##### ۴-۱ سیستم مصنوعی اجتماع مورچه‌ها

یک سیستم مصنوعی اجتماع مورچه‌ها، سیستمی شبیه‌سازی شده از رفتار مورچه‌ها است که از سازوکار همکاری و یادگیری آنها استفاده می‌کند. در این سیستم، از مورچه‌های مصنوعی استفاده می‌شود که پیروی کننده رفتار مورچه‌های واقعی هستند. مورچه‌های مصنوعی از یک سو دارای ویژگی‌های مورچه‌های واقعی هستند و از سوی دیگر با خصوصیات اضافی که در مورچه‌های واقعی دیده نمی‌شوند، تقویت شده‌اند. مهم‌ترین تفاوت‌های مورچه‌های مصنوعی با مورچه‌های واقعی عبارتند از:

- مورچه‌های مصنوعی دارای حافظه هستند.
  - زمان برای آنها از اهمیت بالایی برخوردار است.
  - حرکات آنها بصورت گسسته و از حالتی به حالت دیگر صورت می‌گیرد.
  - مقدار فرمونی که برجای می‌گذارند می‌تواند متناسب با کیفیت جواب ساخته شده باشد.
  - برنامه زمانی برجایگذاری فرمون در آنها مانند مورچه‌های واقعی نیست.
  - برای کارایی بیشتر الگوریتم می‌توانند توانایی‌های بیشتری مانند بهینه‌سازی محلی داشته باشند.
- اما مهم‌ترین شباهت‌های مورچه‌های مصنوعی با مورچه‌های واقعی عبارتند از:
- مسیرهایی با فرمون بیشتر را ترجیح می‌دهند.
  - در مسیرهایی با طول (هزینه) کمتر، فرمون بیشتری برجای می‌گذارند.
  - از یک سیستم ارتباط غیر مستقیم به کمک فرمون استفاده می‌کنند.
  - عملیات صورت گرفته در اجتماع در نتیجه همکاری تمام مورچه‌ها است.

##### ۴-۲ رفتار مورچه‌ها

حشراتی که زندگی اجتماعی دارند مانند مورچه‌ها، مورچه‌ها و زنبورها وقتی به صورت یک گروه عمل می‌کنند با استفاده از قوانینی ساده مانند تماس، صدا و مواد شیمیایی قادر به برقراری ارتباط و حل مسائل پیچیده می‌شوند. چنین رفتارهایی که در گروه‌های خاصی از حشرات وجود دارد به هوش جمعی<sup>۲۴</sup> معروف شده است. در مورد مورچه‌ها، گونه‌های مختلفی از آنها می‌توانند ماده‌ای شیمیایی به نام فرمون<sup>۲۵</sup> ترشح کنند و با آن اثری را در مسیری که پیموده‌اند باقی گذارند. مورچه‌های بعدی با حس کردن این ماده و دنبال کردن اثر آن، از مسیری که مورچه‌های قبلی در یافتن غذا طی کرده‌اند استفاده می‌کنند. این موضوع با گذشت زمان باعث دستیابی به کوتاه‌ترین مسیر به سوی غذا خواهد شد [۲۷]. به عنوان مثال در شکل ۵ در قسمت A، مورچه‌ها به محلی می‌رسند که در آن باید مسیر حرکت خود را انتخاب کنند. از آنجا که در آغاز هیچ فرمونی وجود ندارد، آنها به صورت تصادفی یکی از دو مسیر بالایی یا پایینی را انتخاب می‌کنند. فرض می‌کنیم که نیمی از آنها مسیر بالا و نیمی دیگر هم مسیر پایین را انتخاب کنند. با فرض یکسان بودن سرعت تمام مورچه‌ها، قسمت‌های C و D نشان می‌دهند که در ادامه چه اتفاقی خواهد افتاد. از آنجا که مسیر پایینی کوتاه‌تر از مسیر بالایی است، در یک دوره زمانی مفروض تعداد مورچه بیشتری می‌تواند از آن عبور کرده، به غذا برسند و مجدداً



شکل ۵. چگونگی رفتار اجتماع مورچه‌ها در مسیریابی (برگرفته از [۲۷])

#### ۲-۴ چگونگی ساخت جواب توسط مورچه‌ها

مورچه‌ها روال‌های سازنده جواب هستند که با حرکت تصادفی بر روی گراف  $G(C, L)$  به ساخت جواب می‌پردازند و در حین حرکت خود باید سیاست‌ها یا محدودیت‌های مسئله را هم رعایت کنند. به هر رأس  $c_i \in C$  و هر یال  $l_{ij} \in L$  می‌توان مقدار فرمون  $\tau$  و یک عدد هیوریستیک دیگر مانند  $\eta$  نسبت داد که اگر به رأس  $c_i \in C$  نسبت داده شده باشند با  $\tau_i$  و  $\eta_i$  و اگر به یال  $l_{ij} \in L$  نسبت داده شده باشند با  $\tau_{ij}$  و  $\eta_{ij}$  نمایش داده می‌شوند.  $\tau$  یا اثر فرمون با گذشت زمان توسط مورچه‌ها به‌نگام می‌شود و نقش حافظه را بازی می‌کند و  $\eta$  حاوی اطلاعاتی است که توسط خود مسئله تعریف می‌شود یا در حین اجرای الگوریتم به دست می‌آید. در بسیاری از مواقع  $\eta$  هزینه یا تابعی از هزینه مثل معکوس هزینه است. در ادامه، روند ساخت جواب توسط مورچه‌ها یا چگونگی حرکت آنها را در گراف  $G(C, L)$  به دقت توضیح می‌دهیم.

هر مورچه دلخواه  $k$ ، دارای یک حالت شروع (مبداء حرکت)  $x_s^k$  و یک حالت پایانی  $e^k$  است. ارتباط مورچه‌ها هم به صورت غیر مستقیم از طریق اطلاعات ذخیره شده در فرمون‌ها برقرار می‌شود. مورچه  $k$  در فرآیند انتخاب رأس از حافظه خود یا  $M^k$  استفاده می‌کند. به علاوه از این حافظه برای ردیابی مسیر برگشت، برجایگذاری فرمون، ساخت جواب موجه و ارزیابی جواب‌های ایجاد شده هم استفاده می‌شود. به این ترتیب با جمع‌آوری اطلاعات لازم برای تصمیم‌گیری (از طریق فرمون و

حافظه)، هر مورچه با یک تصمیم‌گیری احتمالی رأس بعدی خود را انتخاب می‌کند. یک مورچه زمانی که در حالت  $X_r = \{x_{r-1}, i\}$  (حالتی که دارای  $r$  رأس است و رأس فعلی آن  $C_i$  است) قرار گرفت به یک رأس  $j$  از همسایگی موجه  $N_i^k$  حرکت می‌کند تا حالت  $\{x_r, j\}$  ایجاد شود. در این الگوریتم ابتکاری و مشتقات آن به جایابی از یک رأس به رأس دیگری در همسایگی آن، براساس معیاری مشخص اصطلاحاً تغییر حالت<sup>۲۳</sup> و به آن معیار، قانون تغییر حالت<sup>۲۷</sup> گفته می‌شود. پس از اینکه گره  $C_r$  به گره فعلی تبدیل شد، مورچه می‌تواند به به‌نگام سازی اثر فرمون برجای مانده بپردازد. به این تغییر در مقدار فرمون که پس از اضافه کردن یک گره رخ می‌دهد، اصطلاحاً به‌نگام سازی گام به گام فرمون<sup>۲۸</sup> گویند. در نهایت، مورچه  $k$  زمانی متوقف می‌شود که یکی از حالت‌های پایانی  $e^k$  آن رخ دهد. به این ترتیب یک اجتماع از مورچه‌ها به صورت غیرهمزمان بر روی گراف مسئله حرکت می‌کنند. این حرکت همراه با تصمیم‌گیری‌های احتمالی است. زمانی که یک جواب ساخته شد، مورچه آن را ارزیابی کرده و مقدار فرمون را در مسیری که استفاده کرده است، به‌نگام می‌کند. این فرمون در مراحل بعد اطلاعاتی را در اختیار مورچه‌های دیگر قرار می‌دهد. گذشته از این فعالیت‌ها، دو دسته دیگر از فعالیت‌ها هم در اجتماع مورچه‌ها رخ می‌دهند: تبخیر فرمون<sup>۲۹</sup> و فعالیت‌های خودکار<sup>۳۰</sup>. فرآیند تبخیر بدان معنا است که با گذشت زمان مقدار

در رابطه ۶،  $\eta_{ij} = \frac{1}{d_{ij}}$  و  $\alpha, \beta$  پارامترهایی از مسأله هستند که اهمیت نسبی مقدار فاصله و فرمون دو شهر  $i, j$  را نشان می‌دهند.  $\tau_{ij}(t)$  میزان فرمون در یال اتصال دهنده دو رأس  $i, j$  در تکرار یا مرحله  $t$  ام و  $N_i^k$  هم همسایگی‌هایی بازدید نشده توسط مورچه  $k$  است زمانی که این مورچه در رأس  $i$  قرار دارد. با قرار دادن  $\alpha = 0$ ، احتمال انتخاب رأس بعدی تنها به فاصله آن از رأس فعلی منوط می‌شود و الگوریتم کاملاً به یک الگوریتم حریصانه تبدیل می‌شود، در حالی که با انتخاب  $\beta = 0$  مقدار فرمون در انتخاب رأس بعدی تنها عامل تعیین کننده خواهد بود و این موضوع باعث همگرایی سریع به یک بهینه محلی و توقف جستجو می‌شود.  $S$  متغیری تصادفی تولید شده از روی توزیع احتمال فرمول ۷ است. به قانون تغییر حالت به دست آمده از این دو فرمول اصطلاحاً انتخاب شبه تصادفی<sup>۳۲</sup> گفته می‌شود. پس از اینکه یک مورچه رأس بعدی خود را انتخاب کرد، نوبت به مورچه بعدی می‌رسد تا آن هم رأس بعدی خود را انتخاب کند. این عمل آنقدر ادامه می‌یابد تا تمام مورچه‌ها رأس بعدی خود را انتخاب کنند. سپس، نوبت به بهنگام سازی محلی فرمون می‌رسد که بر اساس فرمول ۸ انجام می‌شود و در آن  $\tau_0$  و  $\zeta$  مقادیری ثابت هستند.

$$s = \begin{cases} \text{Max}(\tau_{ij})^\alpha \times (\eta_{ij})^\beta & q \leq q_0 \\ S & q > q_0 \end{cases} \quad (6)$$

$$P_{ij}^k = \frac{(\tau_{ij}(t))^\alpha \times (\eta_{ij})^\beta}{\sum_{l \in N_{ki}} (\tau_{il}(t)) \times (\eta_{il})^\beta} \quad (7)$$

$$\tau_{ijk}(t) = (1 - \zeta) \times \tau_{ijk}(t-1) + \zeta \times \tau_0 \quad (8)$$

در این فرمول  $i, j$  رئوسی هستند که به ترتیب توسط مورچه  $k$  بازدید شده‌اند یعنی نخست رأس  $i$  و سپس (بر اساس انتخاب شبه تصادفی) رأس  $j$ . به این ترتیب مقدار فرمون در یال‌هایی که توسط مورچه  $k$  پیموده شده است کاهش می‌یابد. این امر به طور نسبی باعث افزایش احتمال انتخاب یال‌هایی می‌شود که تاکنون بازدید نشده‌اند. پس از اینکه مورچه  $k$  در یالی که به این ترتیب بین دو رأس  $i, j$  ساخت، مقدار فرمون را بهنگام کرد، نوبت به سایر مورچه‌ها می‌رسد که در یال‌هایی که به این ترتیب

فرمون در رئوس  $(\tau_i)$  و یا یال‌ها  $(\tau_{ij})$  کاهش پیدا می‌کند. تبخیر فرمون برای جلوگیری از همگرایی سریع به بهینه محلی طراحی شده است به طوری که با ایجاد یک فراموشی باعث ادامه جستجو در نواحی جدیدی از فضای جواب می‌شود. فعالیت‌های خودکار برای اجرای فعالیت‌های متمرکز بکار می‌روند که یک مورچه به تنهایی قادر به انجام آن نیست. یک نمونه از این فعالیت‌ها این است که مورچه در حین جستجو قبل از انتخاب مسیر بداند که مسیری که قصد انتخاب آن را دارد توسط مورچه‌ای که بهترین جواب را ساخته است انتخاب شده است یا خیر. بهنگام‌سازی‌هایی که توسط فعالیت‌های خودکار در مقدار فرمون صورت می‌گیرند بهنگام‌سازی منفصل نامیده می‌شوند. بر اساس چگونگی بر جای گذاری فرمون، چگونگی تبخیر آن، چگونگی انتخاب رأس بعدی بر اساس رئوس قبلی، الگوریتم‌های متعدد اجتماع مورچه‌ها ایجاد شده است که از جمله می‌توان به الگوریتم سیستم مورچه [۲۵]<sup>۳۱</sup>، الگوریتم سیستم اجتماع مورچه‌ها [۲۷]<sup>۳۲</sup>، الگوریتم سیستم ماکزیمم-مینیمم مورچه [۳۰]<sup>۳۳</sup> و ..... اشاره داشت که از بین آنها به توضیح الگوریتم سیستم اجتماع مورچه‌ها خواهیم پرداخت.

#### ۳-۴ الگوریتم سیستم اجتماع مورچه‌ها

یکی از الگوریتم‌های مشتق شده از اجتماع مورچه‌ها است که در سال ۱۹۹۶ توسط دریگو و گامباردلا ارائه شد و نسبت به الگوریتم‌های قبل از خود نتایج بهتری ایجاد کرده است. مدل‌سازی مسأله فروشنده دوره‌گرد با این الگوریتم به شکل زیر است [۲۷]:

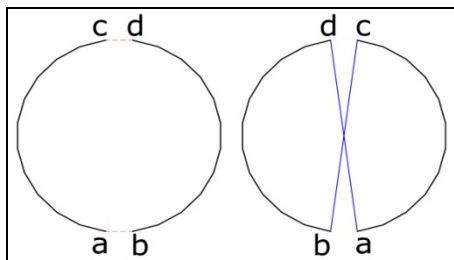
در حل مسأله فروشنده دوره‌گرد، در ابتدا  $m$  مورچه به تصادف بر روی تعدادی رأس قرار می‌گیرند (یک مورچه روی یک رأس). مورچه‌ها به صورت موازی با هم اقدام به ساخت تور می‌کنند. زمانی که یک مورچه بر روی رأسی قرار گرفت، بر اساس قانون تغییر حالت به انتخاب رأس بعد و در نهایت ایجاد تور می‌پردازد. به عنوان مثال زمانی که مورچه  $k$  ام در رأس  $i$  قرار دارد، بر اساس فرمول ۶ ابتدا عدد تصادفی  $q$  در فاصله  $[0,1]$  تولید می‌شود و با  $q_0$  (یکی از پارامترهای الگوریتم) مقایسه می‌شود.

اگر  $q \leq q_0$  باشد، رأسی که حداکثر کننده  $(\tau_{ij})^\alpha (\eta_{ij})^\beta$  باشد از مجموعه همسایگی‌های رأس  $i$  انتخاب می‌شود. در غیر این صورت رأس بعد با احتمال  $S$  انتخاب می‌شود.



## یافتن کوتاه‌ترین تور همیلتونی ایران با استفاده از ترکیب الگوریتم سیستم اجتماع مورچه‌ها و جستجوی محلی

شکستن دو یال دیگر از این تور می‌رویم. این کار آنقدر تکرار می‌شود تا با شکستن هیچ دو یالی دیگر نتوان تابع هدف را کاهش داد. در این حالت به کمینه محلی رسیده‌ایم [۲۱]. شکل ۷ چگونگی عملکرد الگوریتم ابتکاری 2-OPT را نشان می‌دهد. شبه کد ترکیب الگوریتم سیستم اجتماع مورچه‌ها و جستجوی محلی در شکل ۸ نشان داده شده است.



شکل ۷. الگوریتم جستجوی محلی 2-OPT

در الگوریتم پیشنهادی (ترکیب الگوریتم سیستم اجتماع مورچه‌ها و جستجوی محلی)، برای نخستین بار در ادبیات موضوع، پس از اینکه تمام مورچه‌ها جواب‌ها (تورهای) خود را ساختند (زمانی که یک تکرار به پایان می‌رسد) از الگوریتم جستجوی محلی 2-OPT برای تبدیل آنها به بهینه محلی استفاده می‌شود. در ادامه، از جواب‌های بهینه محلی برای بهنگام سازی عمومی فرمون استفاده می‌شود. استفاده از جواب‌های بهینه محلی برای بهنگام سازی عمومی فرمون باعث می‌شود که فرمون بیشتری بر روی یال‌های جواب‌های بهینه محلی باقی بماند و جستجو به سوی چنین جواب‌هایی هدایت شود. اگر چه این امر می‌تواند باعث همگرایی زود هنگام مورچه‌ها به سوی جواب‌های بهینه محلی شود، اما به دلیل ماهیت احتمالی انتخاب رأس بعد (که بر اساس فرمول‌های ۶ و ۷ انجام می‌شود) عملاً خطر همگرایی زود هنگام به جواب‌های بهینه محلی از بین می‌رود. شکل ۸ شبه کد الگوریتم پیشنهادی را نشان می‌دهد.

```

Procedure Ant colony system (one iteration)
Set pheromone trails to small constant
While termination condition does not meet do
Place each ant on initial node
For i=1 to n do (#nodes)
For k=1 to m do (#ants)
Apply state transition rule
End (ants)
Apply local update
End for (one Iteration)
Local Search (for all tours made by ants)
Global update
End while
    
```

شکل ۸. الگوریتم ترکیبی ACS و جستجوی محلی

ساخته‌اند مقدار فرمون را بهنگام کنند. این کار ادامه می‌یابد تا تمام مورچه‌ها به این ترتیب تورهای خود را بسازند (به این ترتیب یک تکرار به پایان می‌رسد). در این هنگام بهنگام سازی عمومی فرمون بر اساس فرمول ۹ انجام می‌شود. مقدار  $\Delta\tau_{ij}$  بر اساس فرمول ۱۰ تعیین می‌شود که در آن  $\cos t_{gb}$  طول (هزینه) بهترین توری است که تاکنون ساخته شده است. شبه کد الگوریتم ACS هم در شکل ۶ نشان داده شده است.

```

Procedure Ant colony system (one iteration)
Set pheromone trails to small constant
While termination condition does not meet do
Place each ant on initial node randomly
For i=1 to n do (#nodes)
For k=1 to m do (#ants)
Apply state transition rule
End (ants)
Apply local update
End for (one Iteration)
Global update
End (while)
    
```

شکل ۶. شبه کد الگوریتم ACS

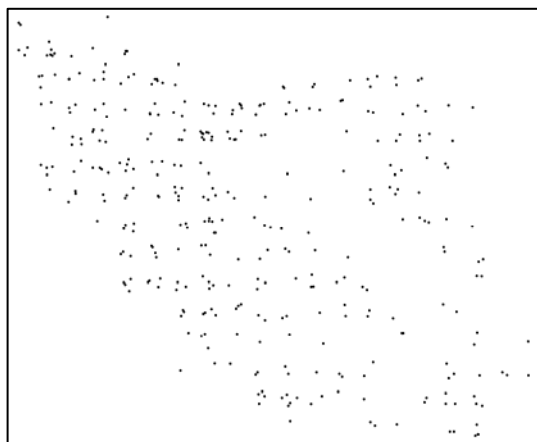
$$\tau_{ij}(t) = (1 - \rho) \times \tau_{ij}(t-1) + \rho \times \Delta\tau_{ij} \quad (9)$$

$$\Delta\tau_{ij} = \begin{cases} \frac{1}{\cos t_{gb}} & (i, j) \in \psi^{gb} \\ 0 & \text{Otherwise} \end{cases} \quad (10)$$

## ۵. الگوریتم پیشنهادی: ترکیب الگوریتم سیستم اجتماع مورچه‌ها و جستجوی محلی (ACS + 2-OPT)

ترکیب الگوریتم‌های فوق ابتکاری سیستم اجتماع مورچه‌ها و الگوریتم جستجوی محلی<sup>۳۰</sup> رویکردی موفقیت آمیز در حل مسائل دشوار بهینه‌سازی ترکیباتی و از جمله مسأله فروشنده دوره‌گرد است. این رویکرد با استفاده از قابلیت‌های هر دو روش به موفقیت‌های فراوانی در حل مسائل معیار فروشنده دوره‌گرد نائل آمده است [۲۲ و ۲۳]. الگوریتم جستجوی محلی بکار رفته در این ترکیب، الگوریتم 2-OPT است. در این الگوریتم ابتدا، تور در محل دو یال شکسته می‌شود. لبه‌های این دو یال به صورت ضربدری به هم متصل می‌شوند. اگر این تغییر منجر به بهبود (کاهش) در تابع هدف شود، به تور اعمال می‌شود. در غیر این صورت، این تغییر در نظر گرفته نمی‌شود و به دنبال

در جدول ۴، ماتریس فاصله ۵ در ۵ برای آن شهر نشان داده شده است. در شکل ۹، ۳۶۰ نقطه منتخب ایران در مختصات دو بعدی نمایش داده شده‌اند.



شکل ۹. نمایش ۳۶۰ نقطه منتخب ایران در مختصات دو بعدی

### ۷. حل مسأله

برای حل این مسأله، از دو روش استفاده شده است. در روش اول از الگوریتم فوق ابتکاری سیستم اجتماع مورچه‌ها [۲۷] و در روش دوم از الگوریتم پیشنهادی (ترکیب الگوریتم فوق ابتکاری سیستم اجتماع مورچه‌ها و جستجوی محلی) استفاده شده است.

### ۸. نتایج حاصل از حل مسأله

برای حل مسأله، الگوریتم سیستم اجتماع مورچه‌ها و الگوریتم پیشنهادی در زبان برنامه‌نویسی MATLAB پیاده‌سازی شده‌اند. سپس پارامترهای الگوریتم فوق ابتکاری سیستم اجتماع مورچه‌ها براساس نتایج [۲۷] به شرح زیر تنظیم شده است:

$$\alpha = 1 \quad \beta = 2$$

$$q_0 = 0.9 \quad \xi = 0.1$$

$$\tau_0 = \frac{1}{n \times L_{mn}} \quad m = 20 \quad \rho = 0.1$$

( $L_{mn}$  طول تور ایجاد شده با یک الگوریتم ابتکاری ایجاد کننده تور به نام نزدیکترین همسایه [۲۱] است). نتایج حاصل از حل مسأله فروشنده دوره‌گرد متقارن در قالب میانگین طول تور (برحسب کیلومتر)، کوتاه‌ترین طول تور (برحسب کیلومتر) و همچنین میانگین زمان حل (برحسب ثانیه) برای ۳۶۰ شهر ایران با هر یک از دو روش فوق پس از ۲۰ بار اجرا در جدول ۵ نشان داده شده است. تور حاصل از این دو روش در شکل‌های ۱۰ و

جدول ۳. طول و عرض جغرافیایی ۵ نقطه منتخب

	ACS	ACS + 2-OPT
میانگین طول تور (کیلومتر)	۲۶۵۳۸,۷۳۲,۰۳	۱۹۰۸۷,۴۳۴,۴۵
کوتاه‌ترین طول تور (کیلومتر)	۲۴۸۶۸,۷۹۸,۵	۱۸۹۳۲,۶۱۱,۳
زمان حل (ثانیه)	۴۵۶۲	۵۵۲۳

جدول ۴. ماتریس فاصله ۵ نقطه منتخب (کیلومتر)

شماره	۱	۲	۳	۴	۵
۱	-	۷,۱۳۶۱	۴۹۰,۸۳	۸۴۳,۹۲	۶۱۱,۰۹
۲	۷,۱۳۶۱	-	۴۸۳,۸۱	۸۵۰,۸۹	۶۱۸,۰۳
۳	۴۹۰,۸۳	۴۸۳,۸۱	-	۱۳۱۰,۳	۱۰۷۸,۷
۴	۸۴۳,۹۲	۸۵۰,۸۹	۱۳۱۰,۳	-	۲۳۳,۱۳
۵	۶۱۱,۰۹	۶۱۸,۰۳	۱۰۷۸,۷	۲۳۳,۱۳	-

### ۶. مسأله ۳۶۰ شهر ایران

با توجه به مباحث مطرح شده در قسمت ۲، مسأله یافتن کوتاه‌ترین تور همیلتونی در بین ۳۶۰ شهر ایران به عنوان نمادی از توانایی در حل این مسأله و برای آزمودن الگوریتم پیشنهادی در مقابل الگوریتم شناخته شده سیستم اجتماع مورچه‌ها ارائه و حل شده است. بدیهی است که انتخاب ۳۶۰ نقطه برای محاسبه تور همیلتونی ایران بر اساس محدودیت‌های زمان و به ویژه منابع محاسباتی در دسترس صورت گرفته است و در صورت فراهم بودن منابع محاسباتی بیشتر و به ویژه امکان استفاده از پردازش‌های موازی، می‌توان تورهای همیلتونی با تعداد بیشتری شهر را برای ایران حل کرد. در این مسأله، ابتدا ۳۶۰ نقطه در ایران بر اساس پارامترهای اقتصادی و اجتماعی (موقعیت جغرافیایی، جمعیت، وضعیت اقتصادی و صنعتی، موقعیت سیاسی) انتخاب شده‌اند. سپس طول و عرض جغرافیایی این نقاط استخراج شده [۳۱] و در نهایت فاصله جغرافیایی<sup>۳۶</sup> - فاصله بر روی کره‌ای به شعاع ۶۳۷۸,۳۸۸ کیلومتر - این نقاط با هم (برحسب کیلومتر) در قالب یک ماتریس ۳۶۰ در ۳۶۰ محاسبه شده است. جدول ۳ طول و عرض جغرافیایی یک نمونه ۵ تایی از ۳۶۰ شهر ایران را نشان می‌دهد.

## ۹. نتیجه‌گیری

مسئله فروشنده دوره‌گرد از مهم‌ترین و پرکاربردترین مسائل در حوزه بهینه‌سازی ترکیباتی است که روش‌های متعدد برای حل آن پیشنهاد شده است. از آنجا که موفقیت در حل این مسئله نشانه توانمندی در استفاده از آن در حوزه‌های مختلف علوم و مهندسی (از جمله حمل‌ونقل) است، در این مقاله برای نخستین بار به حل مسئله فروشنده دوره‌گرد متقارن برای ۳۶۰ نقطه منتخب ایران با ترکیب الگوریتم سیستم اجتماع مورچه‌ها و جستجوی محلی و مقایسه نتایج حاصل با نتایج الگوریتم سیستم اجتماع مورچه‌ها پرداخته شد. این مقایسه نشان دهنده برتری قابل ملاحظه کیفیت نتایج ترکیب الگوریتم سیستم اجتماع مورچه‌ها و جستجوی محلی بر نتایج الگوریتم سیستم اجتماع مورچه‌ها بود. بر اساس نتایج [۲۷]، الگوریتم سیستم اجتماع مورچه‌ها به تنهایی در حل مسائلی با ابعاد کوچک در مقایسه با سایر الگوریتم‌های تقریبی بسیار موفق بوده است. اما در مورد مسائلی با ابعاد بزرگ‌تر (مانند مسئله ۳۶۰ شهر ایران) حتماً این الگوریتم باید به کمک سایر ایده‌ها و روش‌های حل تقویت شود. برتری قابل ملاحظه الگوریتم پیشنهادی بر الگوریتم سیستم اجتماع مورچه‌ها مویده موفق بودن رویکرد ترکیبی اتخاذ شده در حل این مسئله دشوار بهینه‌سازی ترکیباتی است. دشواری این مسئله (علی‌الخصوص مسائلی با ابعاد بزرگ) استفاده از نقاط قوت روش‌های متعدد (در قالب یک الگوریتم ترکیبی) در حل آن را ضروری می‌سازد.

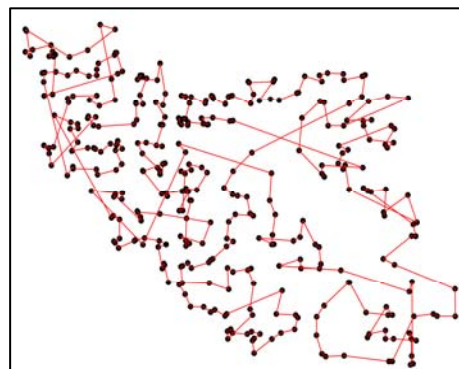
در انجام تحقیقات آتی پیشنهادات زیر ارائه می‌شوند:

۱. استفاده از زبان‌های برنامه‌نویسی سطح پایین‌تر مانند ++C برای دستیابی به سرعت بیشتر در حل مسئله.
۲. استفاده از الگوریتم‌های جستجوی محلی قوی‌تر مانند لین-کرنیگان [۳۲] در ترکیب الگوریتم سیستم اجتماع مورچه‌ها و جستجوی محلی برای دستیابی به کیفیت بالاتر جواب.
۳. آزمایش سایر روش‌های ارتباط و تبادل اطلاعات میان مورچه‌ها (مانند روش‌های توالی سری و یا ترکیب سری و موازی بین مورچه‌ها).
۴. استفاده از سایر روش‌ها در تنظیم پارامترهای الگوریتم سیستم اجتماع مورچه‌ها.
۵. استفاده از سایر روش‌ها-قطعی، ابتکاری و فوق ابتکاری- در حل مسئله ۳۶۰ شهر ایران و مقایسه نتایج حاصل از آنها با نتایج حاصل در این مقاله.

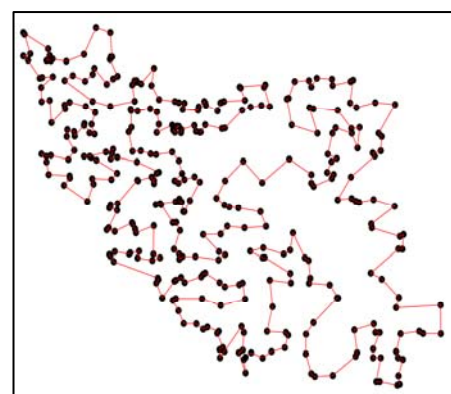
۱۱ نشان داده شده است. همان طور که در این دو شکل مشخص است، در تور حاصل از الگوریتم سیستم اجتماع مورچه‌ها یال‌های متقاطع زیادی وجود دارد. وجود هریک از این تقاطع‌ها نشان دهنده این است که امکان بهبود این تور وجود دارد. هر اندازه یک تور به تور بهینه نزدیک‌تر باشد تعداد چنین تقاطع‌هایی کمتر شده به طوری که در تور بهینه هیچ تقاطع‌هایی وجود ندارد. در واقع برای تشخیص بصری یک تور با کیفیت از یک تور بی کیفیت می‌توان از این روش استفاده کرد. تور حاصل از الگوریتم پیشنهادی (ترکیب الگوریتم سیستم اجتماع مورچه‌ها و جستجوی محلی) دارای تقاطع‌های بسیار کمتری است و به تبع (همان طور که در جدول ۵ مشخص است) دارای کیفیت بهتری هم است.

جدول ۵. نتایج حل مسئله ۳۶۰ شهر ایران با دو روش

شماره	نام	طول جغرافیایی	عرض جغرافیایی
۱	بابل	۵۲,۵'	۳۶,۴'
۲	بابلسر	۵۲,۴۵'	۳۶,۴۵'
۳	بيله سوار	۴۸,۲۰'	۳۹,۲۲'
۴	بردسیر	۵۶,۳۵'	۲۹,۵۳'
۵	بافق	۵۵,۲۵'	۳۱,۴'



شکل ۱۰. نمایش بهترین تور حاصل از ACS



شکل ۱۱. نمایش بهترین تور حاصل از ACS + 2-OPT

3. Garey, M. R., Johnson, D. S. (1979) "Computers and intractability : A guide to the theory of NP-Completeness ", Freeman

4. Chong, Y.N. (2001) "Heuristic algorithms for routing problems ", Ph.D Thesis, School of Mathematics and Statistics, Curtin University of Technology, Australia

5. Tapan, P. B, Gupta , J. N.D, Kandarajah, C. S (2006) "A review of TSP based approaches for flowshop scheduling", European Journal of Operational Research, 169, pp. 816-854.

6. www.tsp.gatech.edu, visited in 2005-6

7. Padberg, M., Rinaldi, G (1987) "Optimization of a 532-city symmetric traveling salesman problem by branch and cut", Operations Research Letters, 6, pp. 1-7

8. Lenstra, J. K., Rinnooy Kan, A. H. G. (1975) " Some simple applications of the traveling salesman problem", Operational Research Quarterly, Vol. 126, No.4, Part 1, pp. 717-733

9. Chi-Kong Lee, Chao-Hui, Chen (2003), "Scheduling of train driver for Taiwan Railway Administration", Journal of the Eastern Asia Society of Transportation Studies, Vol. 5, pp.292-306.

10. Kocjan, W. (2001) "Heuristic methods for routing and scheduling "SICS Technical Report T2001-17

11. Ghoseiri, K., Morshedsolouk, F. (2006) "ACS-TS: train scheduling using ant colony system," Journal of Applied Mathematics and Decision Sciences.

12. Hadjicharalambous, G., Pop, P., Pyrga, E. Tsaggouris, G. and Zaroliagis, C. (2007) "The railway traveling salesman problem", Algorithmic methods for railway optimization ,Proc. ATOMS, 2004, Lecture Notes in Computer Science ,Springer Verlag, Vol. 4359.

13. Picard, J. C, Queyranne, M. (1978) "The time-dependant traveling salesman problem and its application to the tardiness problem in One- machine Scheduling", Operations Research, Vol. 26, No.1, Scheduling, pp. 86-110

14. Agrawala, R., Applegate ,D. L., Maglott, D., Schuler, G. D. and Schaffer, A.A. (2000) "A fast and scalable radiation hybrid map construction and integration strategy", Genome Research, Vo.10, Issue 3, pp. 350-364.

15. Korostensky, C. and Gonnet, G. (1999) "Near optimal multiple sequence alignments using a traveling salesman problem approach", Proceedings of the String Processing and Information Retrieval Symposium & International Workshop on Groupware.

## ۱۰. تشکر و قدردانی

در پایان، جا دارد از داوران محترم ناشناس که نظراتشان باعث بهبود و پربارتر شدن این مقاله گردید قدردانی کنیم.

## ۱۱. پانویس‌ها

- 1- Traveling Salesman Problem
- 2- Combinatorial Optimization
- 3- Symmetric
- 4- Asymmetric
- 5- Traveling Salesman Problem with Time Windows
- 6- Period Traveling Salesman Problem
- 7- Black and White Traveling Salesman Problem
- 8- Selective Traveling Salesman Problem
- 9- Moving Target Traveling Salesman Problem
- 10- TSPLIB
- 11- Job Shop Scheduling
- 12- Flow Shop Scheduling
- 13- Integrated Circuits
- 14- Branch and Bound
- 15- Branch and Cut
- 16- Heuristics
- 17- Metaheuristics
- 18- Greedy Algorithm
- 19- Genetic Algorithms
- 20- Tabu Search
- 21- Ant Colonies
- 22- Simulated Annealing
- 23- Artificial Neural Networks
- 24- Stigmergy
- 25- Pheromone
- 26- State Transition
- 27- State Transition Rule
- 28- Step by Step Pheromone Update
- 29- Pheromone Evaporation
- 30- Daemon
- 31- Ant System(AS)
- 32- Ant Colony System(ACS)
- 33- MAX-MIN Ant System(MMAS)
- 34- Pseudo Random Proportional Rule
- 35- Genetic Local Search
- 36- Geometric Distance

## ۱۲. مراجع

1. Paralos P.M, Resende M.G.C. (2002) "Handbook of applied optimization" , Oxford University Press, pp.616-624
2. Gutin, G. and Punnen, A. P. (2002) "The traveling salesman problem and its variations", Combinatorial Optimization, Vol. 12, Kluwer Academic.

24. Fiechter, C. N. (1994) "A parallel tabu search algorithm for large traveling salesman problem," *Discrete Applied Mathematics and Combinatorial Operations Research and Computer Science*, 51, pp. 243-267
25. Gambardella, L. M. and Dorigo, M. (1996) "Solving symmetric and asymmetric TSPs by ant colonies", in *Proc. IEEE Conf. On Evolutionary Computation, ICEC96*, pp. 622-627, IEEE Press
26. Dorigo, M. and Di Caro, G. (1999) "Ant algorithms for discrete optimization", *Artificial Life* 5, no.2, pp. 73-81
27. Dorigo, M. and Gambardella, L. M. (1997) "Ant colony system: A cooperative learning approach to the traveling salesman problem" *IEEE Transactions on Evolutionary Computation*, Vol. 1, pp. 53-66
28. Kirkpatrick, S. and Gelatt, C. D. and Vecchi, M. P. (1983) "Optimization by simulated annealing", *Science*, Vol. 220, pp.671-680.
29. Aarts, E.H.L. and Stehouwer, H.P. (1993) "Neural networks and the traveling salesman problem", in *Proc. Int. Conf. on Artificial Neural Networks*, pp. 950-955, Springer-Verlag
30. Stutzle, T., Hoos, H. H. (2000) "MAX-MIN Ant System", *Future Gen. Comput. Systems* 16, pp. 889-914.
۳۱. "اطلس راه‌های ایران" (۱۳۸۵)، انتشارات گیتاشناسی.
32. Lin, S. , Kernighan, B. (1973) " An effective heuristic algorithm for the traveling salesman problem", *Operation Research*, Vol. 21, pp. 498-516.
16. Korostensky, C. and Gonnet, G. H. (2000) "Using traveling salesman problem algorithms for evolutionary tree construction", *Bioinformatics*, Vol.16, no.7, pp. 619-627
17. Bailey, C.A., Maclain, T.W. and Beard , R.W. (2000) "Fuel saving strategies for separated spacecraft Interferometry", *AIAA Guidance, Navigation and Control Conference*.
18. Dantzig, G., Fulkerson, R., Johnston, S. (1954) "Solution of a large-scale traveling salesman problem", *Operations Research*, 2, pp. 393-410
19. Salkin, M., Mathur, K. and Has, R. (1989) "Foundations of integer programming", Prentice Hall, pp. 91-104
20. Orman, A.J. and William, H.P. (2007) "A survey of different integer programming formulations of the traveling salesman problem", *Advances in Computational Management Science*, Springer, Berlin.
21. Johnson, D. S. and McGeoch, L. A. (1997) "The traveling salesman problem: A case study in local optimization", *Local Search in Combinatorial Optimization*.
22. Freisleben, B., Merz P. (1996) "A genetic local search algorithm for solving symmetric and asymmetric traveling salesman problems", *International Conference on Evolutionary Computation*, pp. 616-62
23. Freisleben B., Merz P. (1996) "New genetic local search operators for the traveling salesman problem," *Conference on Parallel Problem Solving from Nature*, pp. 890-899