

## یک روش ترکیبی مبتنی بر خوشه بندی برای حل مساله فروشنده دوره گرد با مقیاس بزرگ

محمد رضا میبیدی

دانشکده برق، مهندسی کامپیوتر و فناوری  
اطلاعات، دانشگاه صنعتی امیرکبیر، تهران، ایران  
MMeybodi@aut.ac.ir

کیوان اصغری

گروه کامپیوتر، دانشگاه آزاد اسلامی  
واحد خاмене، ایران  
Kayvan.Asghari@yahoo.com

باقر زارعی

دانشکده مهندسی کامپیوتر، دانشگاه آزاد اسلامی  
واحد شبستر، ایران  
Zarei\_Bager@yahoo.com

مساله فروشنده دوره گرد، تعمیم یافته مساله مشهور سیکل همیلتنی است. فرم کلی این مساله برای اولین بار در ۱۹۳۰ توسط Karl Menger مطرح شد و بعداً توسط Hassler Whitney و Flood ترویج داده شد. فرض کنید که یک گراف کامل داریم که هر یال  $(u, v) \in E$  یک هزینه صحیح نامنفی  $c(u, v)$  را دارد. فروشنده باید با شروع از یک مبدا، تمامی شهرها را دقیقاً یک بار ملاقات کرده و به شهر مبدا بازگردد طوری که هزینه کل تور حداقل گردد.

بعضی از مسائل وجود دارند که با افزایش بعد آنها، زمان حلشان به طور نمایی افزایش می یابد. این مسائل، مسائل بهینه سازی ترکیبی هستند، که زمان حل آنها به صورت تابعی غیر چند جمله ای است. مساله فروشنده دوره گرد یکی از آنها می باشد که حل مساله به معنای پیدا کردن بهترین تور، در مقایسه با تورهای شناخته شده قبلی نمی باشد بلکه همچنین باید ثابت کرد که توری با هزینه کمتر از تور پیدا شده نیز وجود ندارد.

آتوماتاهای یادگیر و الگوریتم های ژنتیکی، هر دو ابزار جستجوی عمومی می باشند که برای حل بسیاری از مسائل NP-Complete بکار برده شده است. در این مقاله یک الگوریتم ترکیبی (الگوریتم ژنتیک + آتوماتای یادگیر) مبتنی بر خوشه بندی برای حل مساله فروشنده دوره گرد با مقیاس بزرگ پیشنهاد شده است. این الگوریتم ابتدا با استفاده از تکنیک خوشه بندی، مساله اصلی را به چند زیر مساله با مقیاس کوچک افزایش داده و سپس از دو روش الگوریتم های ژنتیکی و آتوماتاهای یادگیر بطور همزمان برای جستجو در فضای حالت و حل هر زیر مساله استفاده می نماید. نشان داده شده است که با استفاده از آتوماتاهای یادگیر و الگوریتم ژنتیک در فرایند جستجو، سرعت رسیدن به جواب افزایش چشمگیری پیدا می کند و همچنین از بدام افتادن الگوریتم در حداقل های محلی جلوگیری می نماید. نتایج آزمایش ها، برتری الگوریتم ترکیبی را نسبت به الگوریتم ژنتیکی و آتوماتاهای یادگیر نشان می دهد و همچنین با استفاده از تکنیک خوشه بندی و اجرای الگوریتم ترکیبی بطور همزمان بر روی هر خوشه - با یک سیستم چند پردازنده ای - می توان زمان لازم برای حل مساله را به حداقل مقدار ممکن کاهش داد.

**چکیده:** یکی از مسائل بسیار مهم در تئوری گراف ها، مساله فروشنده دوره گرد می باشد که یک مساله NP-Complete است. اکثر مسائلی که می توان آنها را با مساله فروشنده دوره گرد مدل کرد، دارای مقیاس خیلی بزرگ هستند که الگوریتم های موجود قادر به حل آنها در یک زمان قابل قبول نیستند. آتوماتاهای یادگیر و الگوریتم های ژنتیکی هر دو از ابزارهای جستجو می باشند که برای حل بسیاری از مسائل NP-Complete بکار برده می شوند. در این مقاله یک الگوریتم ترکیبی (الگوریتم ژنتیک + آتوماتای یادگیر) مبتنی بر خوشه بندی برای حل مساله فروشنده دوره گرد با مقیاس بزرگ پیشنهاد شده است. این الگوریتم ابتدا با استفاده از تکنیک خوشه بندی، مساله اصلی را به چند زیر مساله با مقیاس کوچک افراز کرده و سپس از دو روش الگوریتم های ژنتیکی و آتوماتاهای یادگیر بطور همزمان برای جستجو در فضای حالت و حل هر زیر مساله استفاده می نماید. نشان داده شده است که با استفاده از آتوماتاهای یادگیر و الگوریتم ژنتیک در فرایند جستجو، سرعت رسیدن به جواب افزایش چشمگیری پیدا می کند و همچنین از بدام افتادن الگوریتم در حداقل های محلی جلوگیری می نماید. نتایج آزمایش ها، برتری الگوریتم ترکیبی را نسبت به الگوریتم ژنتیکی و آتوماتاهای یادگیر نشان می دهد و همچنین با استفاده از تکنیک خوشه بندی و اجرای الگوریتم ترکیبی بطور همزمان بر روی هر خوشه - با یک سیستم چند پردازنده ای - می توان زمان لازم برای حل مساله را به حداقل مقدار ممکن کاهش داد.

**کلمات کلیدی:** مساله فروشنده دوره گرد، آتوماتای یادگیر، الگوریتم ژنتیک، خوشه بندی

### ۱- مقدمه

گراف ها ابزارهای قدرتمندی هستند که به طور گسترده در کاربردهای متعددی مورد استفاده قرار می گیرند. یکی از مسائل بسیار مهم در تئوری گراف ها، مساله فروشنده دوره گرد می باشد. بسیاری از کاربردهای عمومی از جمله طراحی حلقه های شبکه های Sonet، کابل های برق، مسیر هواپیما ها، مسیریابی وسائط نقلیه و ... را می توان با مساله فروشنده دوره گرد مدل کرد.

## ۲- الگوریتم های ژنتیک

الگوریتم های ژنتیکی که بر مبنای ایده تکامل در طبیعت عمل می نماید، بر روی جمعیتی از راه حل های بالقوه به جستجوی راه حل نهایی می پردازند. در هر نسل، بهترینها ی آن نسل انتخاب می شوند، و پس از زاد و ولد، مجموعه جدیدی از فرزندان را تولید می کنند. در این فرایند افراد مناسبتر با احتمال بیشتری در نسل های بعدی باقی خواهند ماند.

در آغاز الگوریتم، تعدادی از افراد (جمعیت اولیه) به صورت تصادفی ساخته شده و تابع هدف برای تک تک آنها ارزیابی می شود. اگر شرط رسیدن به جواب برقرار نباشد (به جواب بهینه نرسیده باشیم)، نسل بعدی با انتخاب والدین بر اساس میزان برازندگی شان تولید می شود و فرزندان با احتمال ثابتی دچار جهش می شوند. سپس میزان برازندگی فرزندان جدید محاسبه شده و جمعیت جدید، از جایگزینی فرزندان با والدین ایجاد می شود و این فرایند تا برقرار شدن شرط خاتمه تکرار می شود.

## ۳- اتوماتاهای یادگیر

یادگیری در اتوماتاهای یادگیر، انتخاب یک اقدام بهینه از میان یک مجموعه از اقدام های مجاز اتوماتا می باشد. این اقدام روی یک محیط تصادفی اعمال می شود و محیط به این اقدام اتوماتا بوسیله یک پاسخ تصادفی از مجموعه پاسخ های مجاز جواب می دهد. پاسخ محیط بصورت آماری به اقدام اتوماتا وابسته است. واژه محیط شامل اجتماع تمام شرایط خارجی و تاثیرات آنها روی عملکرد اتوماتا می باشد.

برای یک گراف با اندازه  $n$ ،  $n!$  جایگشت مختلف از رئوس وجود دارد و در صورتیکه از اتوماتاهای یادگیر برای حل کردن مساله فروشنده دوره گرد استفاده شود، اتوماتا باید  $n!$  اقدام داشته باشد که تعداد زیاد اقدام ها سرعت همگرایی اتوماتا را کاهش می دهد. به همین جهت اتوماتای مهاجرت اشیاء توسط اومن و ما پیشنهاد شده است.

## ۴- روش بکار گرفته شده برای خوشه بندی گراف های بزرگ

ما در این مقاله جهت پیدا کردن تور فروشنده دوره گرد برای گراف هایی با مقیاس بزرگ (گراف هایی با تعداد گره های بسیار زیاد)، ابتدا گراف مورد نظر را توسط یکی از روش های خوشه بندی گراف بنام Kmean، به خوشه هایی با اندازه کوچک افراز کرده و سپس با استفاده از الگوریتم ترکیبی (الگوریتم ژنتیک + اتوماتای یادگیر) که در بخش بعدی توضیح داده شده است، برای هر خوشه تور فروشنده دوره گرد با کمترین هزینه را بطور جداگانه پیدا می کنیم. از هر خوشه نیز چندین گره را به عنوان نماینده خوشه، طوری انتخاب می کنیم که نسبت به خوشه های دیگر دارای کمترین فاصله باشند، سپس با استفاده از الگوریتم ترکیبی، بین خوشه ها (نمایندگان خوشه ها) تور فروشنده دوره گرد با کمترین هزینه را نیز بدست می آوریم. در نهایت هزینه تور برای گراف بزرگ برابر با، مجموع هزینه تورهای داخل هر کدام از خوشه ها بعلاوه هزینه تور بدست آمده بین نمایندگان خوشه ها خواهد

بود. با توجه به اینکه عملیات پیدا کردن تور فروشنده دوره گرد برای هر خوشه را می توان بصورت مستقل انجام داد، می توان برای هر خوشه تور فروشنده دوره گرد را توسط پردازنده های مجزا و بصورت موازی پیدا کرده و از این طریق کل زمان لازم برای پیدا کردن تور برای گراف های بزرگ را به حداقل مقدار ممکن کاهش داد.

## ۵- الگوریتم جستجوی ترکیبی برای پیدا کردن تور فروشنده

### دوره گرد در داخل هر خوشه

با ترکیب الگوریتم ژنتیک و اتوماتای یادگیر و تلفیق مفاهیم ژن، کروموزوم، اقدام و عمق، سابقه تاریخی تکامل راه حل مساله، به شکل کارا استخراج شده و در روند جستجو مورد استفاده قرار می گیرد. خاصیت مهم الگوریتم ترکیبی، مقاومت آن در مقابل تغییرات سطحی جواب هاست، به عبارتی دیگر تعادلی انعطاف پذیر بین کارایی الگوریتم ژنتیک و پایداری اتوماتای یادگیر در الگوریتم ترکیبی وجود دارد. خود ترمیمی، تولید مثل، جریمه و پاداش از ویژگی های الگوریتم ترکیبی است. در ادامه پارامترهای اصلی این الگوریتم توضیح داده شده است.

### ۵-۱- ژن و کروموزوم

در الگوریتم پیشنهادی برخلاف الگوریتم های ژنتیک کلاسیک، از کدگذاری دودویی برای کروموزوم ها استفاده نمی شود. هر کروموزوم توسط یک اتوماتای یادگیر از نوع مهاجرت اشیاء نشان داده می شود. بطوریکه هر کدام از ژن های کروموزوم به یکی از اقدام های اتوماتا نسبت داده می شود و در یک عمق مشخصی از آن اقدام قرار می گیرد.

در این اتوماتا  $\alpha = \{\alpha_1, \dots, \alpha_k\}$  مجموعه اقدام های مجاز برای اتوماتای یادگیر است. این اتوماتا  $k$  اقدام دارد (تعداد اقدام های این اتوماتا با تعداد راس های گراف برابر است). اگر راس  $u$  از گراف در اقدام  $m$  قرار گرفته باشد، در اینصورت راس  $u$  در ترتیب ملاقات کردن شهرها،  $m$ امین شهر می باشد.

$\underline{\phi} = \{\phi_1, \phi_2, \dots, \phi_{KN}\}$  مجموعه وضعیت ها و  $N$  عمق حافظه برای اتوماتا می باشد. مجموعه وضعیت های این اتوماتا به  $K$  زیر مجموعه  $\{\phi_1, \phi_2, \dots, \phi_N\}$  و  $\{\phi_{N+1}, \phi_{N+2}, \dots, \phi_{2N}\}$  و  $\{\phi_{(K-1)N+1}, \phi_{(K-1)N+2}, \dots, \phi_{KN}\}$  افراز می شود و راس های گراف بر اساس این که در کدام وضعیت قرار داشته باشند دسته بندی می گردند. اگر گره  $u$  از گراف در مجموعه وضعیت های  $\{\phi_{(j-1)N+1}, \phi_{(j-1)N+2}, \dots, \phi_{jN}\}$  قرار داشته باشد در اینصورت راس  $u$  در ترتیب ملاقات کردن شهرها،  $j$ امین شهر می باشد. در مجموعه وضعیت های اقدام  $j$ ، به وضعیت  $\phi_{(j-1)N+1}$  وضعیت داخلی و به وضعیت  $\phi_{jN}$  وضعیت مرزی گفته می شود.

به عنوان مثال گراف کامل شکل (۱) را که شامل ۶ راس می باشد در نظر بگیرد.

باشد. جمعیت اولیه حاصل از گراف شکل (۱) در شکل (۳) نشان داده شده است. در ابتدا هر گره در وضعیت مرزی اقدام خود قرار دارد.

### ۳-۵- تابع برازندگی

در الگوریتم های ژنتیک تابع برازندگی، شاخص زنده ماندن کروموزوم ها است. لذا برازندگی یک آتوماتا در مساله فروشنده دوره گرد به صورت  $f(LA_i) = 1/\text{Length of Specified Tour by } LA_i$  تعریف می شود.

### ۴-۵- عملگرها

از آنجاییکه در الگوریتم ترکیبی، هر کروموزوم به صورت یک آتوماتای یادگیر نمایش داده می شود، عملگرهای جایجایی و جهش مشابه عملگرهای سنتی ژنتیک نیستند.

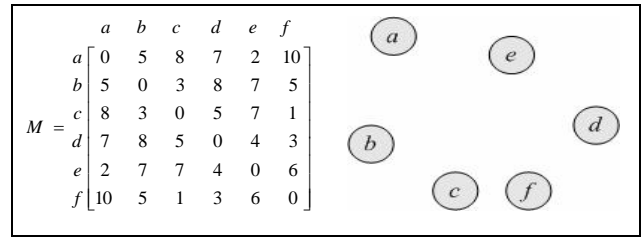
الف) عملگر انتخاب: برای انتخاب آتوماتاهای یادگیر (کروموزوم ها) برای عملگرهای جهش و ترکیب می توان از یکی از روشهای رتبه بندی، سازوکار چرخ رولت و یا Tournament استفاده کرد.

ب) عملگر ترکیب یا جایجایی: برای انجام دادن این عملگر می توان از یکی از روشهای Ordered, Partially Mapped Crossover, Crossover, Cycle Crossover و New Crossover که برای کار با جایگشت ها مناسب هستند استفاده کرد. در اینجا فقط روش پیشنهادی یعنی روش New Crossover توضیح داده می شود. در این روش دو کروموزوم والد انتخاب شده و به صورت تصادفی دو ژن  $i$  و  $j$  در یکی از دو کروموزوم والد انتخاب می شوند. سپس همین دو ژن در کروموزوم والد دیگر نیز انتخاب می شوند. مجموعه ژنهای با شماره های بین  $i$  و  $j$  را مجموعه جایجایی می نامیم. سپس ژن های هم شماره در دو مجموعه جایجایی با یکدیگر جایجا می شوند. با این عمل دو کروموزوم جدید حاصل می شوند که اصطلاحاً فرزندان دو آتوماتای والد خوانده می شوند.

به عنوان مثال فرض کنید که آتوماتاهای LA2 و LA5 از جمعیت تشکیل شده قبل به عنوان والد انتخاب شوند. با انتخاب تصادفی دو محل  $a_2$  و  $a_3$  مجموعه جایجایی  $\{\alpha_2, \alpha_3\}$  حاصل می شود و در نهایت مطابق شکل (۴) با جایجایی اقدام های متناظر در فاصله جایجایی، دو کروموزوم جدید حاصل می شود.

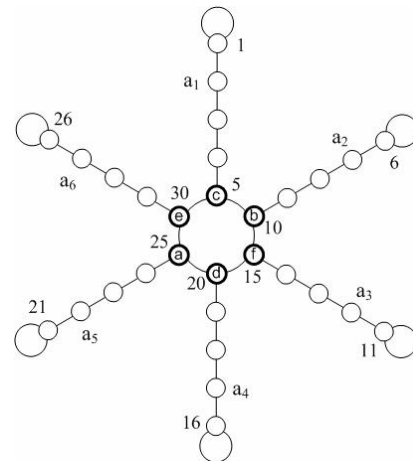
پ) عملگر جهش: برای انجام دادن این عملگر می توان از یکی از روشهای Inversion, Insertion Mutation, Swap Mutation, Scramble Mutation که برای کار با جایگشت ها مناسب هستند استفاده کرد. به عنوان مثال در روش Swap Mutation، دو اقدام (ژن) از یک آتوماتا (کروموزوم) به صورت تصادفی انتخاب شده و جایجا می شوند.

ت) عملگر جریمه و پاداش: از آنجاییکه هر کروموزوم به صورت یک آتوماتای یادگیر نشان داده شده است، در هر یک از آتوماتاها پس از بررسی میزان برازندگی یک ژن (راس یا اقدام) که به صورت تصادفی



شکل (۱)- گراف کامل با ۶ راس

جایگشت  $\langle c, b, f, d, a, e \rangle$  از گراف شکل (۱) را در نظر بگیرید. این جایگشت توسط یک آتوماتای یادگیر با اتصالات مشابه آتوماتای ستلین در شکل (۲) نشان داده شده است. این آتوماتا دارای ۶ اقدام  $\{\alpha_1, \alpha_2, \alpha_3, \alpha_4, \alpha_5, \alpha_6\}$  (به تعداد راسهای گراف) و عمق ۵ می باشد. مجموعه وضعیت های  $\{1, 6, 11, 16, 21, 26\}$  و مجموعه وضعیت های  $\{5, 10, 15, 20, 25, 30\}$  در ابتدا هر یک از راسهای گراف در وضعیت مرزی اقدام مربوطه قرار دارند.



شکل (۲)- نمایش جایگشت  $\langle c, b, f, d, a, e \rangle$  به وسیله آتوماتای یادگیر با اتصالات مشابه آتوماتای ستلین

### ۲-۵- جمعیت اولیه

با فرض اینکه تعداد اعضای جمعیت  $n$  باشد،  $n-1$  عضو جمعیت با ایجاد  $n-1$  جایگشت تصادفی تولید می شوند. برای تولید آخرین عضو جمعیت، از روش نزدیکترین همسایه ملاقات نشده استفاده می کنیم. به این جایگشت، جایگشت تقریبی می گوئیم. آخرین عضو اضافه شده به جمعیت بیشترین تشابه را با جواب نهایی دارد.

به عنوان مثال نحوه تشکیل جمعیت اولیه برای گراف شکل (۱) با فرض  $n=6$  در ادامه توضیح داده شده است. پنج عضو اول جمعیت به وسیله پنج جایگشت تصادفی  $\langle b, d, e, a, f, c \rangle$ ،  $\langle d, e, f, b, c, a \rangle$ ،  $\langle e, f, b, d, a, c \rangle$  و  $\langle c, f, b, e, d, a \rangle$  و  $\langle b, d, c, a, e, f \rangle$  ایجاد می شود. برای ایجاد جایگشت ششم از روش نزدیکترین همسایه ملاقات نشده استفاده می کنیم. با فرض اینکه راس شروع  $a$  باشد جایگشت ششم بصورت  $\langle a, e, d, f, c, b \rangle$  می

شماره ۱۶) قرار داشته باشد و پاداش بگیرد در همان وضعیت باقی می ماند. نحوه حرکت چنین راسی در شکل (۵) نشان داده شده است.

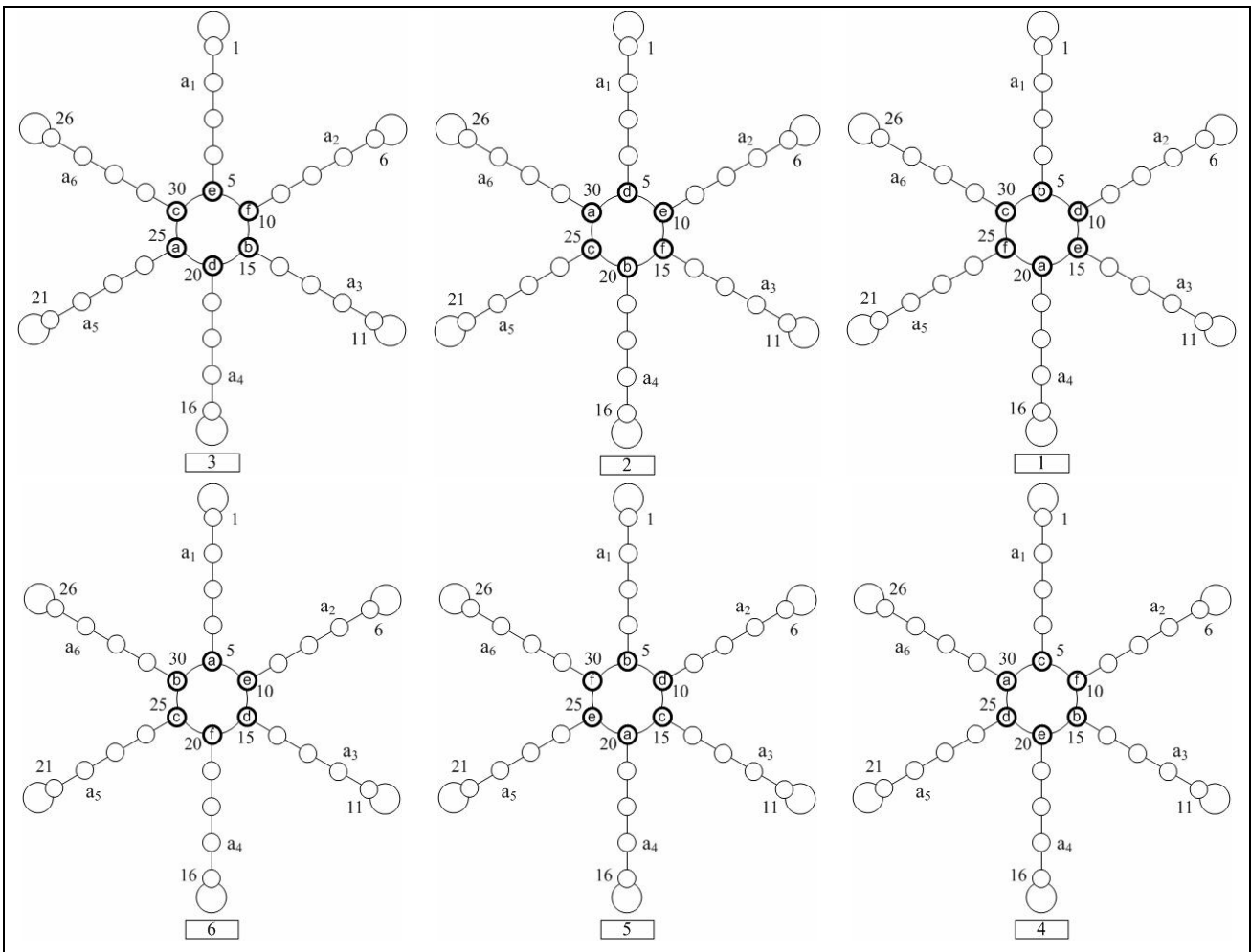
اگر میزان برانزنگی یک راس از مقدار آستانه بزرگتر باشد در اینصورت تور برقرار شده مناسب نبوده و این راس جریمه می شود. نحوه حرکت چنین راسی برای دو حالت مختلف در زیر آمده است.

الف) راس در وضعیتی غیر از وضعیت مرزی قرار داشته باشد: جریمه نمودن این راس سبب کم اهمیت شدن این راس شده و راس به سمت وضعیت های مرزی حرکت می کند.

ب) راس در وضعیت مرزی قرار داشته باشد: در این حالت راسی از گراف را پیدا می کنیم بطوریکه اگر در جایگشت مربوطه جای دو راس عوض شوند بیشترین کاهش در هزینه تور حاصل گردد. در اینصورت اگر راس پیدا شده در وضعیت مرزی قرار داشته باشد جای دو راس عوض می شود و در غیر اینصورت ابتدا راس مشخص شده به وضعیت مرزی اقدام خود منتقل و سپس جایجایی صورت می پذیرد. نحوه حرکت چنین راسی در شکل (۶) نشان داده شده است.

انتخاب می شود، آن ژن پاداش یا جریمه می شود. در اثر پاداش دادن یا جریمه کردن یک ژن، وضعیت ژن در مجموعه وضعیت های اقدام مربوطه، تغییر می کند. اگر ژنی در وضعیت مرزی یک اقدام قرار داشته باشد، جریمه شدن آن باعث تغییر اقدام آن و در نتیجه باعث ایجاد جایگشت جدیدی می شود. نرخ این عملگر باید پایین باشد زیرا این عملگر، یک عملگر جستجوی تصادفی است و اگر با نرخ بالا اعمال شود باعث کاهش در کارایی الگوریتم می شود. عملگر جریمه و پاداش با توجه به نوع آتوماتای یادگیر متفاوت می باشد.

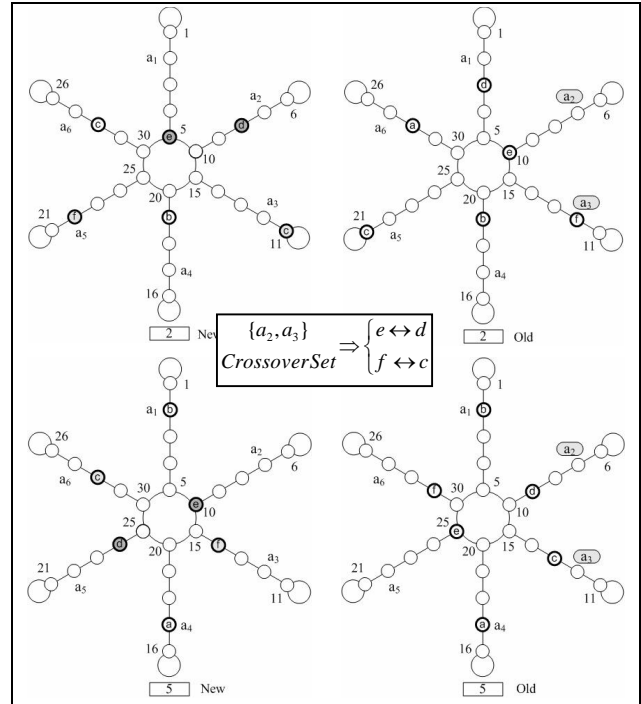
به عنوان مثال در آتوماتای با اتصالات مشابه آتوماتای ستلین، اگر راس  $b$  در مجموعه وضعیت های  $\{16,17,18,19,20\}$  قرار داشته باشد و میانگین هزینه یالهای ورودی و خروجی به راس  $b$  (هزینه یال ورودی به راس  $b$  + هزینه یال خروجی از راس  $b$  تقسیم بر ۲) از مقدار آستانه (مقدار آستانه بصورت تطبیقی مشخص می گردد و مقدار آن در هر لحظه برابر است با نسبت هزینه کل تور به تعداد راسها) کوچکتر باشد به این راس پاداش داده می شود و به سمت وضعیت های داخلی تر این اقدام حرکت می کند. اگر راس  $b$  در داخلی ترین وضعیت (وضعیت



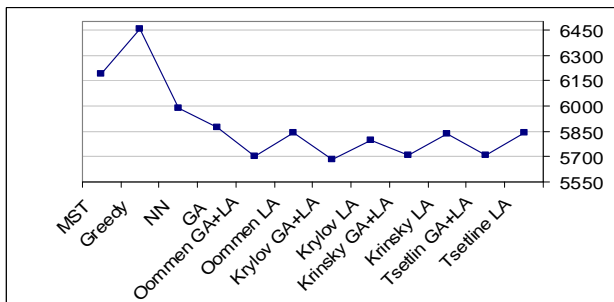
شکل (۳) - جمعیت اولیه برای گراف شکل (۱)

روشهای مبتنی بر اتوماتای یادگیر و الگوریتم ژنتیک نشان می دهد. در آزمایش های انجام گرفته اندازه گراف ها (گراف ها از TSPLIB انتخاب شده اند) از ۲۲ تا ۲۸۰ راس و تعداد تکرارها از ۵۰ تا ۵۰۰ تکرار در نظر گرفته شده است. در الگوریتم ترکیبی و اتوماتای یادگیر عمق های ۱، ۴، ۷، ۱۰ و ۱۵ آزمایش شده اند. در الگوریتم ترکیبی و الگوریتم ژنتیکی روش Swap Mutation با نرخ ۲۵٪ و روش رتبه بندی برای انتخاب کروموزمها استفاده شده است و همچنین سایز جمعیت برابر با تعداد نودهای گراف در نظر گرفته شده است.

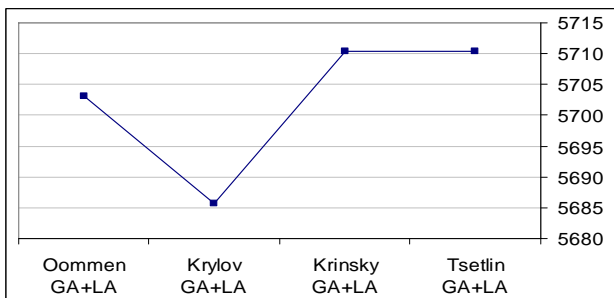
در نمودار ۱ مقایسه الگوریتم ترکیبی با سایر الگوریتم های حل مساله فروشنده دوره گرد، و در نمودارهای ۲ و ۳ نتایج بدست آمده از الگوریتم ترکیبی مبتنی بر اتوماتاهای مختلف، آورده شده است. همانطور که از نتایج معلوم است، الگوریتم ترکیبی مبتنی بر اتوماتای کرایلو در مقایسه با سایر الگوریتم ها، هم از لحاظ زمان اجرا و هم از لحاظ طول تور بدست آمده عملکرد مطلوبی برای حل مساله فروشنده دوره گرد با مقیاس کوچک از خود نشان می دهد. در جدول (۱) نتایج بدست آمده برای یک نمونه از مساله فروشنده دوره گرد با مقیاس بزرگ با استفاده از روش ترکیبی مبتنی بر خوشه بندی آورده شده است.



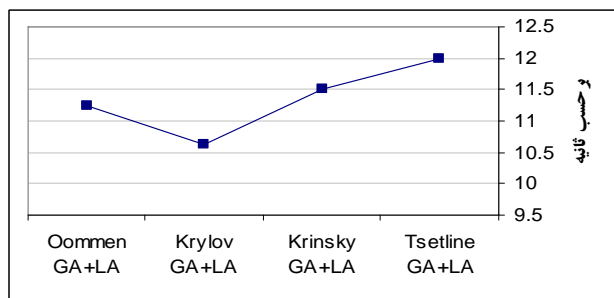
شکل (۴) - نحوه انجام عملگر جابجایی (New Crossover)



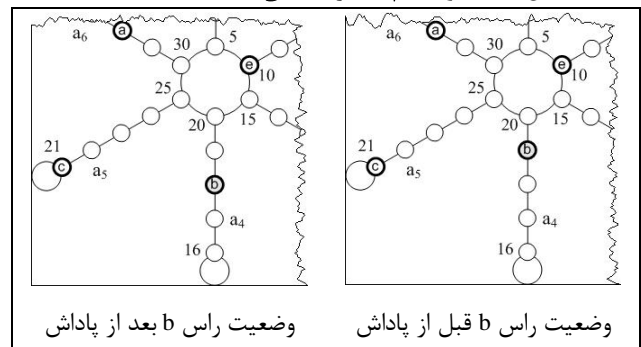
نمودار ۱- میانگین طول (هزینه) تور بدست آمده از الگوریتم های مختلف



نمودار ۲- میانگین طول (هزینه) تور بدست آمده از الگوریتم ترکیبی

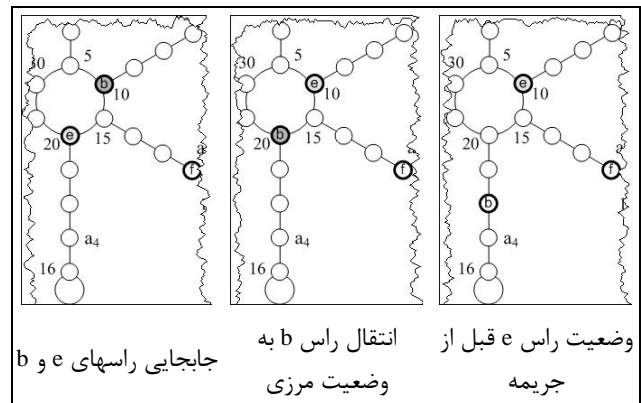


نمودار ۳- میانگین زمان لازم برای الگوریتم ترکیبی



وضعیت راس b قبل از پاداش      وضعیت راس b بعد از پاداش

شکل (۵) - نحوه پاداش دادن به راس b



وضعیت راس e قبل از جابجایی راسهای e و b      انتقال راس b به وضعیت مرزی

شکل (۶) - نحوه جریمه کردن راسی که در وضعیت مرزی قرار دارد

## ۶- نتایج آزمایش ها

در این بخش نتایج آزمایشی الگوریتم های حل مساله فروشنده دوره گرد که براساس اتوماتای یادگیر، الگوریتم ژنتیک و الگوریتم ترکیبی (الگوریتم ژنتیک + اتوماتای یادگیر) پیاده سازی شده اند، نشان داده شده است. این نتایج بهبود قابل توجه الگوریتم ترکیبی را نسبت به

Editors, Kluwer Academic Publishers, Boston, pp. 445-487, 2002.

- [6] H. Beigy, and M. R. Meybodi, *Optimization of Topology of Neural Networks Using Learning Automata*, Proc. of 3th Annual Int. Computer Society of Iran Computer Conference CSICC-98, Tehran, Iran, pp. 417-428, 1999.
- [7] B. Freisleben, and P. Merz, *A Genetic Local Search Algorithm for Solving Symmetric and Asymmetric Traveling Salesman Problems*, appeared in Proc. of the 1996 IEEE Int. Conference on Evolutionary Computation, Nagoya, Japan, pp. 616-621, 1996.
- [8] B. Freisleben, and P. Merz, *New Genetic Local Search Operators for the Traveling Salesman Problem*, in Proc. of the 4th Conference on Parallel Problem Solving from Nature - PPSN IV, (H.-M. Voigt, W. Ebeling, I. Rechenberg, H.-P. Schwefel, eds.), Vol. 1141 of Lecture Notes in Computer Science, pp. 890-899, 1996.
- [9] M. Jünger, G. Reinelt, and G. Rinaldi, *The Traveling Salesman Problem*, in Handbooks in Operations Research and Management Science, Vol. 7 (M.O. Ball, T. Magnanti, C.L. Monma, and G. Nemhauser, eds), Elsevier Science B.V., pp. 225-330, 1995.
- [10] P. Moscato, and M.G. Norman, *An Analysis of the Performance of Traveling Salesman Heuristics on Infinite-Size Fractal Instances in the Euclidean Plane*, Oct. 1994.
- [11] M. Grötschel, and O. Holland, *Solution of Large-Scale Symmetric Traveling Salesman Problems*, Mathematical Programming 51, pp. 141-202, 1991.
- [12] M. Padberg, and G. Rinaldi, *A Branch-and-Cut Algorithm for the Resolution of Large-Scale Symmetric Traveling Salesman Problems*, SIAM Review 33, pp. 60-100, 1991.
- [13] B. J. Oommen, R. S. Valiveti, and J. R. Zgierski, *An Adaptive Learning Solution to the Keyboard Optimization Problem*, IEEE Transaction On Systems, Man, And Cybernetics, Vol. 21, No. 6, pp. 1608-1618, 1991.
- [14] D. E. Goldberg, *Genetic Algorithms in Search, Optimization and Machine Learning*, Reading, MA, Addition-Wesley, 1989.
- [15] K. S. Narendra, and M. A. L. Thathachar, *Learning Automata: An Introduction*, Prentice-hall, Englewood cliffs, 1989.
- [16] B. J. Oommen, and D. C. Y. Ma, *Deterministic Learning Automata Solution to the Keyboard Optimization Problem*, IEEE Transaction on Computers, Vol. 37, No. 1, pp. 2-3, 1988.
- [17] A. A. Hashim, S. Amir, and P. Mars, *Application of Learning Automata to Data Compression*, in Adaptive and Learning Systems, K. S. Narendra, Editor, New York, Plenum Press, pp. 229-234, 1986.
- [18] Tapas Kanungo, David M. Mount, Nathan S. Netanyahu, Christine D. Piatko, Ruth Silverman, and Angela Y. Wu, *An Efficient K-Means Clustering Algorithm: Analysis and Implementation*, IEEE Transaction on Pattern Analysis and Machine Intelligence, Vol. 24, No. 7, July 2002.
- [19] Tapas Kanungo, David M. Mount, Nathan S. Netanyahu, Christine D. Piatko, Ruth Silverman, and Angela Y. Wu, *The Analysis of a Simple K-Means Clustering Algorithm*, Proc. of the Sixteenth Annual Symposium on Computational Geometry, pp. 162, June 2000.
- [20] Leng Mingwei, Tang Haitao, and Chen Xiaoyun, *An Efficient K-means Clustering Algorithm Based on Influence Factors*, Eighth ACIS Int. Conference on Software Engineering, Artificial Intelligence, Networking, and Parallel/Distributed Computing, pp. 815-820, July 2007.

جدول ۱- طول تور بدست آمده برای مساله fn14461 با ۴۴۶۱ گره

شماره خوشه	تعداد گره در خوشه	طول تور بهینه
۱	۴۴۱	۲۴۱۴۱.۱۴
۲	۵۵۵	۲۷۳۰۲.۴۹
۳	۵۵۳	۲۹۰۸۲.۸۶
۴	۶۲۶	۳۱۹۹۳.۵۱
۵	۳۹۱	۲۴۲۰۵.۰۳
۶	۶۷۲	۳۲۵۵۶.۲۲
۷	۶۸۲	۳۲۷۶۷.۵۳
۸	۵۴۱	۲۹۴۱۴.۴۷
بی‌ن خوشه‌ها	۴۳	۱۵۱۳۴.۷۵
مجموع	۴۴۶۱+۴۳	۲۴۶۵۹۸

#### ۷- نتیجه گیری و پیشنهادها

گراف‌ها، بویژه گراف‌های برجسب دار، ابزارهای قدرتمند و پراستفاده‌ای هستند که به طور گسترده در کاربردهای کامپیوتر مورد استفاده قرار می‌گیرند. یکی از مسائل بسیار مهم در تئوری گراف‌ها، پیدا کردن تور فروشنده دوره گرد می‌باشد. محققان بیش از دو دهه بر روی این مساله کار کرده‌اند، ولی با توجه به این حقیقت که هنوز الگوریتمی از درجه چند جمله‌ای برای حل این مساله وجود ندارد، پژوهش‌ها در این زمینه همچنان ادامه دارد. با استفاده از روشهای جستجوی مناسب و ترکیب آنها، می‌توان الگوریتم‌های بهینه برای این مساله پیدا نمود. همچنین با خوشه بندی گره‌های گراف و اجرای الگوریتم ترکیبی بطور همزمان بر روی هر خوشه، می‌توان به نتایج بهتری رسید و در صورت استفاده از یک سیستم چند پردازنده‌ای می‌توان زمان لازم برای حل مساله اصلی را چندین برابر (به تعداد خوشه‌ها) کاهش داد.

#### ۸- مراجع

- [۱] میبیدی، محمد رضا و بیگی، حمید. *حل مساله تناظر گراف توسط اتوماتاهای یادگیر*. دانشکده مهندسی کامپیوتر. دانشگاه صنعتی امیرکبیر. تهران. ایران. ۱۳۷۹.
- [۲] میبیدی، محمد رضا و رضایور میرصالح، مهدی. *یک روش ترکیبی (GA+LA) برای حل مساله تناظر گراف*. دانشکده مهندسی کامپیوتر. دانشگاه صنعتی امیرکبیر. تهران. ایران. ۱۳۸۲.
- [3] Bager Zarei, M. R. Meybodi, and Mortaza Abbaszadeh, *A Hybrid Method for Solving Traveling Salesman Problem*, Proceedings of the 6th IEEE/ACIS International Conference on Computer and Information Science (ICIS 2007), IEEE Computer Society, Melbourne, Australia, pp. 394-399, 11-13 July 2007.
- [4] D. S. Johnson, and L. A. McGeoch, *Experimental Analysis of Heuristics for the STSP*, in the Traveling Salesman Problem and its Variations, G. Gutin and A. Punnen, Editors, Kluwer Academic Publishers, Boston, pp. 369-443, 2002.
- [5] D. S. Johnson, G. Gutin, L. A. McGeoch, A. Yeo, W. Zhang, and A. Zverovich, *Experimental Analysis of Heuristics for the ATSP*, in the Traveling Salesman Problem and its Variations, G. Gutin and A. Punnen,