

Solving Traveling Salesman Problem Using Combination Intelligent Techniques and Genetic Algorithm

حل مسئله فروشنده دوره‌گرد با استفاده از ترکیب تکنیک‌های هوشمند و الگوریتم ژنتیک

Behrooz Ameri Shahrabi, Samane Hosseinmardi, Mohammad Ebrahim Shiri Ahmad Abadi

Abstract — Various methods have been proposed to solve the traveling salesman problem such as a solution based on genetic algorithm. But in general, these algorithms and especially genetic algorithm are often faced with early convergence in solving combinatorial problem, in this paper we review and introduce new methods and survey to solve the problem. Our idea based on hybrid genetic algorithm for solving the traveling salesman problem. The implementation of our algorithm is that different methods at each step of the algorithm, including initial population by the nearest neighbor, greedy algorithm and combination of them with the genetic algorithm and also different methods for mutation operator has been surveyed. Combining untwist algorithm with genetic algorithm to improve the population and is expected to strengthen the global optimal solution. When the diversity of the population is reduced, artificial chromosomes with high diversity level are introduced to increase the diversity level in the system. Implementation has been shown the improvement result of hybrid algorithm than the pure genetic algorithm¹.

Keywords — hybrid genetic algorithm, artificial chromosomes, nearest neighbor, untwist, greedy, informed search, traveling salesman problem

الگوریتم ژنتیک معمولی را نشان می‌دهد.

1. چکیده

روش‌های مختلفی برای حل مسئله فروشنده دوره‌گرد ارائه شده است، از جمله راه حلی مبتنی بر الگوریتم ژنتیک، ولی در حالت کلی این الگوریتم‌ها و به خصوص الگوریتم ژنتیک در حل مسئله‌های ترکیبی غالباً با همگرایی زودرس مواجه هستند، که ما در این مقاله با بررسی و معرفی روش‌ها و عملگرهای جدید به رفع این مشکل پرداخته‌ایم. ایده‌ی ما مبنی بر الگوریتم ژنتیک ترکیبی برای حل مسئله‌ی فروشنده دوره‌گرد می‌باشد. روند اجرای الگوریتم ما چنین است که روش‌های مختلفی که در هر مرحله از الگوریتم، اعم از ایجاد جمعیت اولیه با الگوریتم‌های نزدیک‌ترین همسایگی، حریصانه و ترکیب آنها با الگوریتم ژنتیک و نیز روش‌های متفاوت برای عملگر جهش مورد بررسی قرار گرفته است. با افزودن و ترکیب الگوریتم جداساز به الگوریتم ژنتیک سعی در بهبود جامعه داشته و انتظار رسیدن به جواب بهینه‌ی سراسری تقویت بخشیده شده است. در زمان‌های کاهش گوناگونی جمعیت، کروموزوم‌های مصنوعی را با گوناگونی بالا برای افزایش سطح گوناگونی به سیستم وارد نموده ایم. پیاده‌سازی‌ها، نتیجه‌ی بهبود الگوریتم ترکیبی نسبت به

کلمات کلیدی

الگوریتم ژنتیک ترکیبی - کروموزوم‌های مصنوعی - نزدیک‌ترین همسایگی - جدا ساز - حریصانه - جستجوی آگاهانه - مسئله‌ی فروشنده دوره‌گرد

2. مقدمه

از آنجایی که برای حل مسائل NP، با اندازه بزرگ، استفاده از روش‌های جستجوی آگاهانه و ناآگاهانه ناکارآمد می‌باشد، روش‌های جستجوی بیان شده غیر ممکن به نظر می‌رسد و روش‌های جستجوی متاهوریستیک مطرح می‌گردد. چگونگی عملکرد این روش‌ها بدین صورت می‌باشد که به طور تصادفی در فضای مسئله حرکت می‌کند. لازم به ذکر است که این حرکت-های تصادفی با کنترل و هدایت الگوریتم انجام می‌گردد. مسائل بهینه‌سازی و بهینه‌سازی ترکیبی از جمله مسائلی هستند که برای حل آنها از روش-

¹ B. Ameri Shahrabi is with the College of Mathematics and Computer Sciences, Amir Kabir University, Tehran, Iran (e-mail: behrooz.ameri@aut.ac.ir).

S. Hosseinmardi is with the College of Mathematics and Computer Sciences, Amir Kabir University, Tehran, Iran (e-mail: samane.hosseinmardi@aut.ac.ir).

M.E. Shiri Ahmadabadi is with the College of Mathematics and Computer Sciences, Amir Kabir University, Tehran, Iran (e-mail: shiri@aut.ac.ir).

2. محاسبه شایستگی⁶ برای هر کروموزم.
 3. انتخاب دو کروموزم بر اساس شایستگی.
 4. اعمال ترکیب⁷ و ایجاد کروموزم جدید.
 5. اعمال جهش⁸ با ضریب P.
 6. تغییر دادن جمعیت اولیه همراه با ورود نسل جدید.
 7. بررسی شرط پایان برنامه.
 8. رفتن به مرحله b.
- برای اطلاعات بیشتر در باره‌ی الگوریتم‌های ژنتیک می‌توان به [6] مراجعه کرد.

4. روش پیشنهادی ترکیبی برای مسئله‌ی فروشنده دوره‌گرد

در این بخش الگوریتم‌های ترکیبی را بررسی نموده و در انتهای بخش، الگوریتم پیشنهادی ارائه شده است.

ا. ایجاد جمعیت اولیه

الگوریتم نزدیک‌ترین همسایگی⁹ الگوریتم نزدیک‌ترین همسایگی [7]، یک الگوریتم هیوریستیک پیش رونده برای مسئله‌ی فروشنده دوره‌گرد است. این الگوریتم بر مبنای زیر می‌باشد:

1. انتقال همه شهرها به یک لیست.
 2. انتخاب اولین شهر به عنوان شهری که ملاقات شده و حذف آن از لیست.
 3. پیدا کردن نزدیک‌ترین شهر به آخرین شهر ملاقات شده در لیست، ملاقات آن و حذف شهر ملاقات شده از لیست.
 4. تکرار مرحله C تا لیست خالی شود.
 5. بازگشت به شهر آغازین.
- پیچیدگی این الگوریتم $O(n^2)$ است و کیفیت کروموزوم‌های ایجاد شده در الگوریتم نزدیک‌ترین همسایگی وابسته به انتخاب شهر آغازین است. استفاده از الگوریتم حریصانه¹⁰ برای ایجاد جمعیت به روش حریصانه، به روش زیر عمل می‌نماییم:
1. یک شهر را به عنوان شهر آغازین در نظر می‌گیریم.
 2. فاصله کلیه شهرها را از این شهر محاسبه نموده و به صورت مرتب شده در لیستی قرار می‌دهیم.
 3. شهری را که کوتاه‌ترین مسیر، نسبت به شهر اول را دارد در کروموزوم قرار می‌دهیم و آن را از لیست حذف می‌نماییم.
 4. تکرار مرحله C تا لیست خالی شود.
 5. بازگشت به شهر آغازین.

های متاهوریستیک استفاده می‌شود. مسائل بهینه سازی ترکیبی، از قبیل مسئله‌ی فروشنده دوره‌گرد² جزء مسائل **NP-Complete** است [1]. تعیین مسیر بهینه انتقال داده در استفاده از شبکه‌های کامپیوتری [2]، پردازش تصویر و تشخیص الگو [3] و تعیین مسیر بهینه حرکت مته برای سوراخ کردن صفحه‌های [4] PCB از جمله زمینه‌های هستند که حل مسئله‌ی فروشنده دوره‌گرد بسیار راه گشا می‌باشد. روش‌هایی که تاکنون برای حل این مسئله ارائه شده‌اند، معمولاً بر پایه جستجوگرها و آشکارسازهای ایستا یا احتمالی هستند که از آن جمله می‌توان به الگوریتم‌های کلاسیک جستجوی محلی، شبکه‌های عصبی مصنوعی، الگوریتم‌های ژنتیک [5] و سیستم کلونی مورچه‌ها³ اشاره کرد. در این مقاله سعی شده است که به حل مسئله‌ی فروشنده دوره‌گرد با استفاده از الگوریتم ژنتیک ترکیبی بپردازیم. یکی از معایب الگوریتم ژنتیک، گیر افتادن در بهینه محلی و همگرایی زودرس جمعیت می‌باشد. ایده‌ی بکار رفته در این مقاله، استفاده از روش‌های مختلف برای ایجاد جمعیت اولیه، عملگرهای جهش متفاوت، بکار بردن عملگر جداساز به منظور بالا بردن کیفیت جمعیت موجود و تزریق کروموزوم‌های مصنوعی برای جستجوی بیش‌تر در فضای مسئله می‌باشد. در بخش بعد به معرفی اجمالی الگوریتم ژنتیک پرداخته شده و در بخش سوم الگوریتم پیشنهادی و ترکیبی استفاده شده در این مقاله ارائه می‌شود و در نهایت در بخش‌های چهارم و پنجم به ترتیب پیاده‌سازی و نتیجه‌گیری ارائه شده است.

3. الگوریتم ژنتیک⁴

الگوریتم ژنتیک روش جستجویی جهت یافتن جواب برتر و نه الزاماً بهینه می‌باشد. این الگوریتم ابزاری می‌باشد که توسط آن ماشین می‌تواند مکانیزم طبیعی را شبیه سازی نماید و از نوع الگوریتم‌های تکاملی می‌باشد که از روش‌های زیست‌شناسی مانند وراثت و جهش استفاده می‌کند. الگوریتم‌های ژنتیک معمولاً به عنوان یک شبیه‌ساز کامپیوتر، که در آن جمعیت (کروموزوم‌ها⁵) یک نمونه‌ی انتزاعی از گزینه‌های راه‌حل یک مسئله‌ی بهینه سازی برای دستیابی به راه حل بهتر، پیاده‌سازی می‌شوند. به طور سنتی راه‌حل‌ها به شکل رشته‌هایی از 0 و 1 بودند، اما امروزه به گونه‌های دیگری هم پیاده‌سازی می‌شوند. فرضیه با جمعیتی کاملاً تصادفی آغاز می‌شود و در نسل‌ها ادامه می‌یابد. در هر نسل تمام جمعیت ارزیابی می‌شوند، چندین کروموزوم در فرایندی تصادفی از نسل جاری انتخاب می‌شوند (براساس شایستگی‌ها) و برای شکل دادن نسل جدید، اصلاح می‌شوند (کسر یا دوباره ترکیب می‌شوند) و در تکرار بعدی الگوریتم به نسل جاری تبدیل می‌شود. روند اجرای الگوریتم ژنتیک به شرح زیر می‌باشد:

1. شروع با یک جمعیت متشکل از تعدادی کروموزوم تصادفی.

⁶Fitness

⁷Crossover

⁸Mutation

⁹Nearest Neighborhood Algorithm

¹⁰Greedy

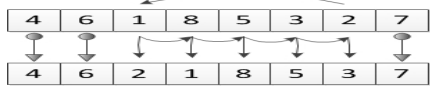
²Traveling Salesman Problem

³Ant Colony

⁴Genetic Algorithm

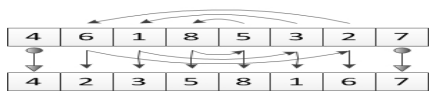
⁵Chromosomes

دو نقطه i, j ($i < j$) را به صورت تصادفی انتخاب می‌کنیم سپس مقادیر بین این دو نقطه را در کروموزوم به سمت راست یا چپ شیفت می‌دهیم. برای مثال کروموزومی با 8 ژن، i را 3 و j را 7 در نظر می‌گیریم، Temp(46185327) تبدیل می‌شود به Temp(46218537). (شکل 3).



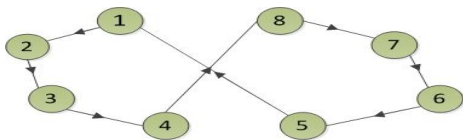
شکل 3: تصویری از کروموزوم اصلی و جهش یافته و نحوه تغییر ژن‌ها جهش پشتک¹⁴

دو نقطه i, j ($i < j$) را به صورت تصادفی انتخاب می‌کنیم، سپس چیدمان مقادیر بین این دو نقطه را در کروموزوم برعکس می‌کنیم. برای مثال کروموزومی با 8 ژن، i را 2 و j را 7 در نظر می‌گیریم، Temp(46185327) تبدیل می‌شود به Temp(42358167). (شکل 4)

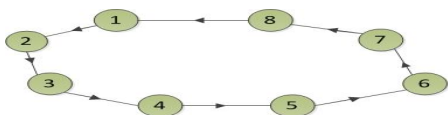


شکل 4: تصویری از کروموزوم اصلی و جهش یافته و نحوه تغییر ژن‌ها . عملگر جداساز¹⁵

در دسته‌ی جواب‌ها، مسیر شهرها غالباً در هم پیچیده می‌شوند مطابق شکل (5). برای مثال در مسئله‌ی فروشنده دوره‌گرد با 8 شهر جواب مسئله $D(4,8)+D(5,1) > D(4,5)+D(8,1)$ است، اما چون $D(4,8)+D(5,1) > D(4,5)+D(8,1)$ می‌باشد، طول مسیر افزایش پیدا کرده است و جواب خوبی نیست. برای بهینه کردن مسیرهای شبیه به شکل (5) عملیات جداساز [8] پیشنهاد می‌شود. این عملگر می‌تواند مسیر را جدا کند و به دنبال آن طول مسیر را نیز کوتاه‌تر نماید. شکل (6) مسیر بعد از جداسازی را نشان می‌دهد.



شکل 5: تصویری از مسیر پیشنهادی الگوریتم



شکل 6: تصویری از مسیر پیشنهادی الگوریتم بعد از اجرای عملگر جداساز

عملگر جداساز برای یک کروموزوم استفاده می‌شود. در این فرایند دو موقعیت نیاز داریم که بر اساس روش زیرکار می‌کند.

1. یک کروموزوم از جمعیت را به طور تصادفی انتخاب می‌کنیم که با $X=(x_1...x_i...x_j...x_n)$ نشان می‌دهیم

این روش کاملاً وابسته به شهر آغازین می‌باشد.

ب. عملگر انتخاب¹¹

یکی از روش‌های انتخاب والدین قرار دادن کروموزوم‌ها در شرایط رقابتی می‌باشد، بدین صورت که در هر مرحله تعداد چهار کروموزوم را در نظر گرفته و از بین کروموزوم‌های فوق دو کروموزومی را که دارای بهترین شایستگی می‌باشند، به عنوان والدین انتخاب می‌گردند.

ج. عملگر ترکیب

عمل ترکیب به روش زیر بر روی کروموزوم‌ها انجام می‌گیرد. برای شرح این روش مسئله‌ی فروشنده دوره‌گرد با هشت شهر را در نظر می‌گیریم. برای مثال دو کروموزوم $P_1(46185327)$ و $P_2(32864715)$ را به عنوان والد در نظر می‌گیریم. دو عدد تصادفی i و j ($i < j$) را بین 1 تا 8 انتخاب می‌کنیم، برای مثال 2 و 5، دو فرزند $Child_1$ و $Child_2$ را داریم و مقادیر بین i و j از والدین P_1 و P_2 را به ترتیب در مکان‌های متناظر فرزندان $Child_1$ و $Child_2$ قرار می‌دهیم. در این مثال $Child_1(2864)$ و $Child_2(6185)$ سپس P_1 را پیمایش و مقادیری که در $Child_1$ وجود ندارد را در مکان‌های خالی $Child_1$ قرار می‌دهیم و همین کار را برای $Child_2$ انجام می‌دهیم بنابراین حاصل $Child_1(12864537)$ و $Child_2(36185247)$ می‌شود. مطابق شکل (1).

P1	4	6	1	8	5	3	2	7
P2	3	2	8	6	4	7	1	5
Tmp 1	6	1	8	5	Index1 = 2			
Tmp 2	2	8	6	4	Index2 = 5			
Initial Child 1	2	8	6	4				
Initial Child 2	6	1	8	5				
Final Child 1	1	2	8	6	4	5	3	7
Final Child 2	3	6	1	8	5	2	4	7

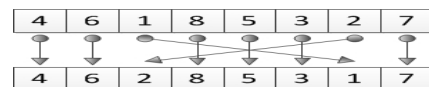
شکل 1: نحوه ایجاد کروموزوم‌های فرزند با توجه به والدین آنها

د. 3-4- عملگر جهش

فرایند جهش در الگوریتم ژنتیک بر روی یک کروموزوم اعمال می‌شود و ژن‌هایی از کروموزوم را به صورت تصادفی تغییر می‌دهد. روش‌های مختلف جهش به شرح زیر است:

جهش بیتی¹²

دو نقطه i, j را به صورت تصادفی انتخاب می‌کنیم سپس مقادیر این دو نقطه را جابه‌جا می‌کنیم. (شکل 2).



شکل 2: تصویری از کروموزوم اصلی و جهش یافته و نحوه تغییر ژن‌ها جهش شیفت چرخشی¹³

¹⁴Flip Mutation
¹⁵Untwist

¹¹Selection
¹²Bitwise Mutation
¹³Circular Shift Mutation

شکل 7: کروموزم‌های انتخاب شده از استخر به صورت تصادفی

با توجه به لیست کروموزوم‌های انتخاب شده، یک ماتریس غلبه تشکیل می‌دهیم.

شکل (8).

	1	2	3	4	5
1	2	3	1	2	2
2	2	1	3	2	2
3	2	2	1	3	2
4	1	3	1	2	3
5	3	1	4	1	1

شکل 8: ماتریس غلبه

بر اساس بیش‌ترین تکرار هر ژن در یک موقعیت، موقعیت ژن را مشخص می‌نماییم. بر اساس مرحله‌ی سوم، تمام ژن‌ها را در جای خود قرار می‌دهیم تا کروموزوم مصنوعی تکمیل گردد. (شکل (9)).

5	1	2	3	4
---	---	---	---	---

شکل 9: کروموزوم مصنوعی ایجاد شده با توجه به ماتریس غلبه

این امر موجب می‌شود که همواره گوناگونی کروموزوم‌های مصنوعی حفظ شود. با کنترل پویای گوناگون سازی جمعیت، مانع همگرایی ناپهنگام الگوریتم ژنتیک ترکیبی می‌شویم که موجب رسیدن به جواب بهتر می‌شود. بر اساس مشاهدات روند همگرایی، زمانی که شایستگی به بهینه‌ی محلی می‌رسد، جمعیت بسیار همگن خواهد بود، بنابراین عامل ژنتیک نمی‌تواند کروموزوم‌های بهتری برای خروج از بهینه‌ی محلی را تولید کند و این نشان دهنده‌ی این امر است که گوناگونی کروموزوم‌ها در این لحظه بسیار پایین است. با تولید کروموزوم‌های مصنوعی گوناگون و تزریق آن به داخل جمعیت موجب می‌شود که بهره‌وری فرایند همگرایی دوباره شروع شود.

ز. الگوریتم پیشنهادی ترکیبی

روند اجرای الگوریتم ژنتیک ترکیبی برای مسئله‌ای دارای N شهر، به شرح ذیل می‌باشد:

1. ایجاد N کروموزوم با استفاده از الگوریتم نزدیک‌ترین همسایگی.

2. ایجاد N کروموزوم با استفاده از الگوریتم حریرانه.

3. ایجاد $(2 * N) - 500$ کروموزوم به صورت تصادفی.

4. محاسبه شایستگی برای هر کروموزوم.

5. انتخاب دو کروموزوم بر اساس شایستگی.

6. اعمال ترکیب و ایجاد کروموزوم جدید.

7. اعمال عملگرهای جهش با ضریب P .

8. اعمال عملگر جداساز بر روی کروموزوم‌ها.

9. تغییر دادن جمعیت اولیه همراه با ورود نسل جدید.

10. افزایش نرخ جهش، در صورتی که در 250 تکرار گذشته هیچ بهبودی در بهترین کروموزوم جامعه دیده نشد.

11. ساخت و تزریق کروموزوم‌های مصنوعی به جامعه، در صورتی که در 600 تکرار

گذشته هیچ بهبودی در بهترین کروموزوم جامعه دیده نشد.

12. بررسی شرط پایان برنامه.

13. رفتن به مرحله d .

2. موقعیت دو شهر را به صورت تصادفی انتخاب می‌کنیم $i < j$.

3. اگر موقعیت شهر i و j به فرمت زیر باشد

$$D(X_i, X_{i-1}) + D(X_{j+1}, X_j) > D(X_i, X_{i-1}) + D(X_{j+1}, X_j)$$

کروموزوم جدید Y به شکل زیر تعریف می‌شود در غیر این صورت کروموزوم

جدیدی نخواهیم داشت. فرمول (1):

$$Y_m = \begin{cases} X_m & , m \geq 1 \text{ and } m < i \\ X_{j-m+i} & , m \geq i \text{ and } m \leq j \\ X_m & , m \geq j \text{ and } m \leq n \end{cases} \quad (1)$$

و. کروموزوم‌های مصنوعی 16

از آنجایی که گوناگون سازی جمعیت، کلید مشکل همگرایی الگوریتم ژنتیک می‌باشد، کنترل مناسب روی گوناگون سازی جمعیت در طول فرایند همگرایی مسئله برای تعادل اکتشاف و بهره‌وری جستجو نیاز است. و احتمال رسیدن به بهینه سراسری با گوناگون سازی کروموزوم‌ها افزایش می‌یابد. بالا بردن گوناگونی در روند اجرای الگوریتم برای اجتناب از همگرایی ناپهنگام و فرار از بهینه محلی حائز اهمیت می‌باشد.

برای بالا بردن گوناگونی کروموزوم‌ها در طول اجرای الگوریتم از تزریق کروموزوم‌های مصنوعی [9] استفاده می‌شود. با تولید و تزریق کروموزوم‌های مصنوعی بر حسب شرایط موجود در مسئله، گوناگونی جمعیت را به صورت پویا کنترل می‌نماییم. کروموزوم‌های مصنوعی همواره از بهترین‌های جمعیت بر اساس مقدار تابع شایستگی تولید می‌شوند. در این جا ما استخری با 3 برابر جمعیت اولیه در نظر گرفته‌ایم، در هر بار تکرار الگوریتم 10 عدد از بهترین‌های جمعیت را در درون استخر قرار می‌دهیم تا زمانی که استخر پر شود. از آنجایی که کیفیت و تنوع داخل استخر در مکانیزم تولید کروموزوم‌های مصنوعی برای گسترش فضای جستجو در الگوریتم ژنتیک ترکیبی بسیار مهم می‌باشد، همواره شرایط ذیل را بررسی می‌نماییم. در این شرایط مکانیزم پویای تولید کروموزوم‌های مصنوعی را تضمین می‌نماییم.

1. اگر در تکرارهای بعدی کروموزومی وجود داشته باشد که مقدار شایستگی

آن از کم‌ترین مقدار شایستگی موجود در استخر کروموزوم‌های مصنوعی

بهتر باشد، این کروموزوم را با کروموزوم موجود در استخر جابه‌جا می-

کنیم.

2. هنگام انتخاب کروموزوم برای ورود به استخر بررسی می‌کنیم که

کروموزومی با این شایستگی در استخر موجود نباشد.

3. در مرحله ساخت کروموزوم مصنوعی از کروموزوم‌های داخل استخر

به روش زیر عمل می‌نماییم:

طول کروموزوم را در عدد 2 ضرب کرده و به صورت تصادفی به تعداد عدد

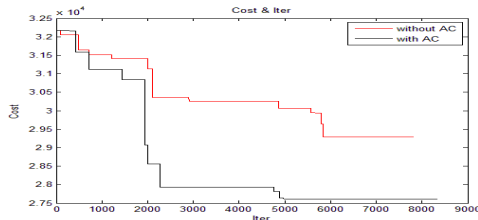
حاصل شده از استخر، کروموزوم انتخاب می‌کنیم. (شکل (7)).

Chromosome 1	1	2	3	5	4
Chromosome 2	3	5	4	1	2
Chromosome 3	2	4	5	3	1
Chromosome 4	5	1	2	3	4
Chromosome 5	4	1	2	3	5
Chromosome 6	2	4	5	1	3
Chromosome 7	5	3	1	2	4
Chromosome 8	5	1	2	4	3
Chromosome 9	3	4	5	2	1
Chromosome 10	1	3	5	4	2

پذیرد. در صورتی که برنامه 600 تکرار را بدون هیچ‌گونه تغییری در بهترین جواب حاصل شده داشته باشد، با تزریق کروموزم‌های مصنوعی، مجدد جمعیت را پراکنده نموده. در هر بار عمل تزریق، ابتدا بر روی جمعیت حال حاضر عمل مرتب‌سازی انجام می‌گیرد و 20% از بهترین‌های جمعیت فعلی نگهداری می‌شوند و سپس 80% دیگر جمعیت با عمل تزریق جایگزین می‌گردند. در شکل (12) تأثیر تزریق کروموزم مصنوعی در روند اجرا و تکامل الگوریتم، در شکل (13) نمایی از پراکندگی جمعیت قبل از تزریق کروموزم-های مصنوعی و در شکل (14) نمایی از پراکندگی جمعیت بعد از تزریق کروموزم‌های مصنوعی قابل مشاهده می‌باشد.

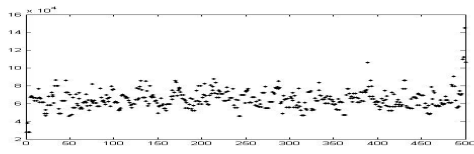
د. عملگر جداساز

در هر بار اجرای برنامه بعد از عملگرهای ترکیب و جهش، عملگر جداساز اجرا می‌گردد. عملگر جداساز بر روی روند اجرای برنامه تأثیر گذار بوده و در رسیدن به بهینه سراسری نقش شایانی را ایفا می‌کند، که در شکل (15) قابل مشاهده می‌باشد.

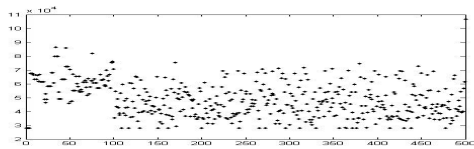


شکل 12: تفاوت در جواب نهایی با توجه به تزریق و یا عدم تزریق

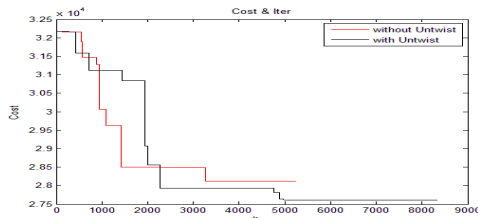
کروموزم مصنوعی



شکل 13: جمعیت قبل از تزریق کروموزم مصنوعی



شکل 14: جمعیت بعد از تزریق کروموزم مصنوعی



شکل 15: تأثیر عملگر جداساز در بهبود کروموزم‌ها و نزدیک شدن

هرچه بیش‌تر به بهینه‌ی سراسری

6. ارزیابی

آزمایش‌های انجام شده بر روی 2 نمونه از سایت TSPLIB صورت گرفته

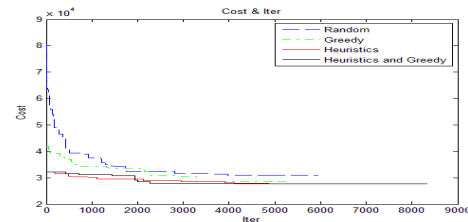
5. پیاده‌سازی و ارزیابی

در الگوریتم ژنتیک ترکیبی¹⁷ برای بهبود نرخ همگرایی و پیدا کردن بهینه‌ی بهتر، از الگوریتم ژنتیک ترکیبی استفاده شده است. در این الگوریتم ترکیبی برای هر یک از عملگرهای استاندارد در الگوریتم ژنتیک سعی شده است که روش‌های متعددی آزمایش شود و بهترین گزینه‌ها در کنار یکدیگر قرار گیرند به علاوه این که از عملگرهای خاص دیگری نیز استفاده شود. در اجرای این برنامه دو شرط پایان در نظر گرفته شده است.

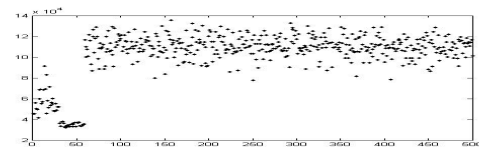
- تعداد تکرار برنامه به عدد 10000 برسد.
- برنامه 2000 بار بدون تغییر در آخرین جواب، اجرا گردد.

آ. جمعیت اولیه

برای تولید جمعیت اولیه به تعداد نودهای موجود در مسئله جمعیتی را با روش هیوریستیک نزدیکترین همسایگی و به همین تعداد جمعیتی را به روش حرصانه و مابقی را به صورت تصادفی تولید می‌کنیم. در این الگوریتم جمعیت اولیه معادل با 500 کروموزوم در نظر گرفته شده است. در شکل (10) به چهار روش مختلف جمعیت اولیه ایجاد گردیده و الگوریتم اجرا شده است، تفاوت جواب نهایی با روش‌های مختلف در شکل مشخص می‌باشد و در شکل (11) نمایی از پراکندگی جمعیت اولیه قابل مشاهده می‌باشد.



شکل 10: بررسی جواب نهایی با توجه به تفاوت در ایجاد جمعیت اولیه



شکل 11: پراکندگی جمعیت اولیه

ب. عملگر جهش

از هر سه روش ذکر شده در بخش 4-4 استفاده می‌گردد. نرخ جهش در ابتدای شروع برنامه 6% می‌باشد، اما این مقدار در طول روند اجرای برنامه تغییر می‌کند. در صورتی که برنامه 250 تکرار را بدون هیچ‌گونه تغییری در بهترین جواب حاصل شده داشته باشد، ضریب جهش در عدد 1,1 ضرب می‌گردد.

ج. تزریق کروموزوم مصنوعی

در هر بار تکرار الگوریتم، کروموزوم‌های موجود در استخر مورد بررسی قرار گرفته و ممکن است که عمل بروز رسانی بر روی تعدادی از آنها صورت

مسافت	تعداد تکرار	
11771	10000	بهترین جواب
13580	6053	بدترین جواب

8. قدردانی

از آقایان سید محمد نکویی و فرهاد ملکی بابت راهنمایی‌های مفیدشان تشکر و قدردانی می‌نماییم.

REFERENCES

- 1 Crescenzi P, Kann V. A Compendium of NP Optimization Problems. Technical Report SI/RR-95/01, DSI, University of Rome, 1995.
- 2 Mehme Ali MK, Kamoun F. Neural Networks for Shortest Path Computation and Routing in Computer Networks. IEEE Trans. Neural Network, 1993; 4(6).
- 3 Banaszak D, Dale GA, Watkins AN, Jordan JD. An Optical Technique for Detecting Fatigue Cracks in Aerospace Structures. 18th International Congress on Instrumentation in Aerospace Simulation Facilities (ICIASF), 1999; 1-7.
- 4 Fujimura K, Fujiwaki S, Kwaw OC, Tokutaka H. Optimization of Electronic Chip-Mounting Machine Using SOM-TSP Method with 5 Dimensional Data. International Conference on Info-tech and Info-net (ICII), 2001; 4:26-31.
- 5 Jiao L, Wang L. A Novel Genetic Algorithm based on Immunity. IEEE Trans. On System, Man, and Cybernetics-Part A: System and Humans, 2000; 30(5).
- 6 Goldberg DE. Genetic Algorithms in Search Optimization and Machine Learning. Reading, MA, Addition-Wesley, 1989.
- 7 Kaur D, Murugappan MM. Performance Enhancement in solving Traveling Salesman Problem using Hybrid Genetic Algorithm. IEEE, 2008.
- 8 Wang LY, Zhang J, Huali I. An Improved Genetic Algorithm For TSP. Proceedings of the Sixth International Conference on Machine Learning and Cybernetics, Hong Kong, 2007.
- 9 Chang PC, Huang WH, Ting CJ. Dynamic Diversity Control in Genetic Algorithm for Mining Unsearched Solution Space in TSP Problems. Expert System with Application, 2010; 37:1863-1878.

است که عبارتند از WI29 و DJ38، برای هر 2 نمونه الگوریتم ژنتیک ترکیبی و الگوریتم ژنتیک معمولی بررسی گردیده است که نتایج آن به شرح زیر می‌باشد.

بهترین مسیر ممکن برای نمونه‌ی WI29 در سایت TSPLIB مسافتی معادل 27603 کیلومتر و برای DJ38 معادل 6656 کیلومتر حاصل شده است. نتایج زیر حاصل 10 بار اجرای هر 2 الگوریتم بر روی هر 2 نمونه می‌باشد.

نتایج حاصل از اجرای هر دو الگوریتم و بررسی آنها، بهبود 40 درصدی عملکرد الگوریتم ژنتیک ترکیبی را نسبت به الگوریتم ژنتیک نشان می‌دهد.

7. نتیجه گیری

نتایج حاصل شده به شرح ذیل می‌باشد:

- با افزایش تعداد جمعیت، ممکن است در تعداد تکرارهای پایین‌تر به بهینه‌ی سراسری به توان نزدیک شد، اما این عمل الگوریتم ژنتیک را تبدیل به یک جستجوی تصادفی¹⁸ می‌نماید و با افزایش تعداد جمعیت سرعت اجرای برنامه کاهش می‌یابد.

- ایجاد جمعیت اولیه به صورت هیوریستیک و حریصانه موجب نزدیک شدن به بهینه‌ی سراسری می‌گردد و باعث می‌شود در تعداد تکرار کم‌تر بتوان به بهینه‌ی سراسری نزدیک گردید.

- تزریق کروموزوم مصنوعی در واگرا شدن جمعیت و پیمایش بیش‌تری از فضای جستجو و فرار از بهینه‌ی محلی تأثیر بسزایی دارد.

- افزایش نرخ جهش در روند اجرای برنامه و در زمانی که جمعیت به سمت همگرایی و بهینه‌ی محلی پیش می‌رود کمک می‌نماید بتوان از این شرایط خارج گردید.

- با توجه به پارامترهای ذکر شده و با ترکیب عملگرهای فوق و اجرای چندین بار برنامه (با توجه به تقریبی بودن الگوریتم ژنتیک) امکان دست یابی به بهینه‌ی سراسری مهیا می‌گردد.

جدول 1: نتایج اجرای الگوریتم ژنتیک ترکیبی بر روی نمونه‌ی W29

مسافت	تعداد تکرار	
27603	9131	بهترین جواب
29286	7013	بدترین جواب

جدول 2: نتایج اجرای الگوریتم ژنتیک بر روی نمونه‌ی W29

مسافت	تعداد تکرار	
34548	10000	بهترین جواب
46630	4487	بدترین جواب

جدول 3: نتایج اجرای الگوریتم ژنتیک ترکیبی بر روی نمونه‌ی DJ38

مسافت	تعداد تکرار	
6656	5571	بهترین جواب
6712	3815	بدترین جواب

جدول 4: نتایج اجرای الگوریتم ژنتیک بر روی نمونه‌ی DJ38

¹⁸Random Search