حل مسئله فروشنده دوره‌گرد با استفاده از الگوریتم ژنتیک

امین عباسی[۱]

سید امین حیدری[۲]

محمدحسین قاسمی[۳]

[۱] دانشجوی کارشناسی مهندسی نرم افزار، آموزشکده فنی و حرفه ای پسران شهرکرد، aminabbasi770 @gmail.com
[۲] کارشناسی ارشد مهندسی نرم مدیریت فناوری اطلاعات، آموزشکده فنی و حرفه ای پسران شهرکرد، am_saead1362 @yahoo.com
[۳] کارشناسی ارشد شبکه، آموزشکده فنی و حرفه ای پسران شهرکرد، m13t92 @gmail.com

چکیده: مساله فروشنده دوره‌گرد از نوع مسائل بهینه‌سازی ترکیبی می‌باشد. در تحقیق حاضر، مسئله‌ی فروشنده‌ی دوره‌گرد چندهدفه در نظر گرفته شده است. مسئله فروشنده دوره‌گرد چندهدفه تعمیم یافته‌ی مسئله فروشنده دوره‌گرد می‌باشد. در مسئله فروشنده چندهدفه، بیش از یک هدف مشخص می‌شود و برای این که به بهترین راه حل برسد، بهینه سازی می‌شود. مسئله‌ی فروشنده‌ی چند هدفه با استفاده از الگوریتم ژنتیک حل می‌شود. مسئله‌ی فروشنده‌ی چندهدفه امکان بهترین راه‌حل‌های بهینه‌ی پارتو را برای همه‌ی نمونه‌های داده‌ای که مورد آزمایش قرار گرفته، ارائه داد. الگوریتم ژنتیک، راه حل‌های نزدیک به بهینه را در دوره زمانی مناسبی تقریبا برای همه‌ی اهداف، تولید می‌کند. مسئله‌ی فروشنده‌ی دوره‌گرد، با نمونه‌های گوناگون از مجموعه داده‌های کتابخانه‌ای محک استاندارد که در برگیرنده‌ی تعداد مختلفی از شهرها می‌باشد، در این کار مورد آزمایش قرار گرفت. نتایج تجربی نشان داد که الگوریتم ژنتیک نزدیک‌ترین راه‌حل‌های بهینه به نمونه‌های مسئله را که از مجموعه مسئله‌ی فروشنده‌ی دوره‌گرد گرفته شده، را ارائه می‌دهد.

کلید واژه: *چندهدفه، مسئله‌ی فروشنده‌ی دوره‌گرد، الگوریتم ژنتیک، بهینه‌سازی .*

۱. مقدمه

مسئله‌ی فروشنده‌ی دوره‌گرد، یک مسئله چند جمله‌ای غیرقطعی و از نوع مسائل ان پی سخت[۱] می‌باشد. مسئله ان پی سخت از سخت‌ترین مسائل در حوزه بهینه سازی می‌باشد[۱]. مسئله‌ی فروشنده‌ی دوره‌گرد در حوزه‌های مختلفی مثل صنعت نیمه‌رسانا، حمل و نقل، بهینه‌سازی اسکن زنجیره، مسیریابی، حفاری، لجستیک و تست مدارهای یکپارچه مورد استفاده قرار می‌گیرند. همچنین در بسیاری از بخش‌های دیگر صنعتی و عملی استفاده می شوند. مسئله‌ی فرو شنده‌ی دوره‌گرد نمونه‌ای از بهینه سازی ترکیبی می‌باشد و در حوزه‌ی علوم کامپیوتری نظری و تحقیقات عملیاتی مورد استفاده و تحقیق قرار می‌گیرد[۲]. سرویلیام روان همیلتون، ریاضیدانی از ایرلند و توماس پنیتون کریک من، ریاضیدان بریتانیایی در قرن هجدهم مسئله‌ی فروشنده‌ی دوره‌گرد را معرفی کردند. کارل منگر و هاروارد فرم کلی مسئله‌ی فروشنده‌ی دوره‌گرد را طراحی کردند. سپس مریل، هسلر و ویتنی در پرینستون آن را ارتقا دادند[۳].

در حوزه‌ی تحقیقات عملیاتی، تعداد زیادی الگوریتم اکتشافی و دقیق برای حل مسئله‌ی فروشنده‌ی دوره‌گرد طراحی شد[۴]. مسئله‌ی فرو شنده‌ی دوره‌گرد مسئله‌ای است که فرو شندگان با آن مواجه هستند. این مسئله، دارای مجموعه‌ای از شهرها با فاصله‌های مربوط بین هر فروشنده می‌باشد. مسئله‌ی فروشنده‌ی دوره‌گرد از برخی از شهرها شروع می شود و وظیفه‌ی پیدا کردن

[۱] NP-hard

کوتاه‌ترین مسیر در بین تمامی‌شهرها را برعهده دارد. فروشنده باید تمامی‌شهرها را دقیقاً یک بار بررسی کند و دوباره به همان شهری که از آن شروع کرده است برگردد. پیدا کردن کوتاه‌ترین مسیر در یک تصویرکاملاً بهم پیوسته و کم کردن هزینه‌ی مسافرت در طول این سفر، هدف کاربردی مسئله‌ی فروشنده‌ی دوره‌گرد می‌باشد. تعداد مسیرهای ممکنی که همه‌ی شهر ها را بهم متصل می‌کند و یک راه‌حل قابل اجرا را ارائه می‌دهد.

مسئله فروشنده دوره‌گرد چندهدفه با استفاده از الگوریتم ژنتیک مطرح شده است. بهینه‌سازی چندهدفه برای ماکزیمم یا مینیمم کردن توابع چندگانه بر اساس این مسئله مورد استفاده قرار می‌گیرد. همچنین این بهینه‌سازی برای کاهش قیمت و فاصله مورد استفاده می‌گیرد. راه‌حلی که توسط بهینه پارتو ارائه می‌شود می‌تواند مغلوب و یا نامغلوب باشد. راه حل هایی که مغلوب نیستند، نامغلوب نامیده می شوند. مجموعه ای از راه‌حل‌هایی که در فضای تصمیم‌گیری عملی وجود دارد پارتو بهینه یا دسته مغلوب نامیده می‌شوند.

۲. بررسی ادبیات

پابلو و همکاران [۵] کاری برای حل کردن مسئله‌ی فروشنده‌ی دوره‌گرد با استفاده از روش دو هدفی جزیره‌ای برای حل کردن هزینه‌ی حمل و نقل زمینی و دریایی ارائه داده‌اند. هدف از هر دو تابع، به حداقل رساندن هزینه های حمل و نقل زمینی و دریایی می‌باشد. آنا اری گیلانو و همکاران [۶] بر روی مسئله‌ی فروشنده‌ی دوره‌گرد نامتقارن با جدول زمانی با استفاده از یک الگوریتم دقیق کار کردند. مسئله با استفاده از جدول زمانی مناسب و زمان‌های سفر حل شد و راه‌حل بهینه مستقل از زمان را نشان داد.

یورفت و همکاران [۷] کاری در زمینه حل کردن مسئله‌ی فروشنده‌ی دوره‌گرد با استفاده از الگوریتم تکراری ارائه دادند. این الگوریتم به منظور تخمین ترتیب مقادیر در لبه‌ها برای ایجاد راه‌حل‌های مختلف پیشنهاد شد. پاندیری و همکاران [۸] یک مسئله‌ی فروشنده‌ی دوره‌گرد از نوع K را با استفاده از الگوریتم جستجوی همسایگی متغیر ارائه و در این مقاله مورد بررسی قرار دادند. این الگوریتم متشکل از N تعداد شهر و یک مقدار ثابت که در این پیدا کردن حداقل شهر با بررسی کردن K از N تعداد شهر می‌باشد، مورد استفاده قرار گرفت. در این جا انتخاب و جایگزینی ترکیبی از مسئله‌ی فروشنده‌ی دوره‌گرد از نوع K می‌باشد . آرماندو و همکاران [۹] یک مسئله‌ی فروشنده‌ی دوره‌گرد برای برداشتن چندین محموله و تحویل آنها با استفاده از یک برنامه‌ی خطی عدد صحیح را معرفی کردند. هدف از این تحقیق، پیدا کردن مسیری برای وسیله نقلیه به طوری که همه درخواست‌ها را برآورده و مسیر سفر را کوتاه کند، می‌باشد.

لی سان و همکاران [۱۰] به منظور حل مسئله فروشنده دوره‌گرد دوره نامشخص روشی را ارائه کردند که در آن نودهای مشتری به تعداد نامحدود از دفعات قابل مشاهده است. رویکرد مشاهده در بلند مدت از راه حل بهینه که شامل تصمیم‌گیری برای مسیریابی و برنامه‌ریزی می‌باشد، تبعیت می‌کند. جوریس کیتابل و همکاران [۱۱] به منظور حل مسئله‌ی فروشنده‌ی دوره‌گرد با قیمت منصفانه روشی را ارائه داده‌اند که در آن مدلی ارائه شد که برنامه نویسی عدد صحیح صفر و یک نامیده شد و توانایی این فرمولاسیون‌ها، با هم مقایسه گردید.

کریستوف دفریان و همکاران[۱۲] روشی را در مورد حل مسئله فروشنده‌ی دوره‌گرد چندهدفه با همکاری‌های افقی ارائه دادند. این مقاله در مورد همکاری افقی سرویس‌های تدارکاتی که در آن بسیاری از شرکت‌ها گرد هم می‌آیند تا مسئله بهینه‌سازی لجستیکی را حل کنند، می‌باشد.

راکوئل و همکاران [۱۳] روشی برای حل مسئله‌ی فروشنده‌ی دوره‌گرد مجموعه‌ای بعنوان یک مسئله ان-پی سخت را ارائه دادند. هدف از این مقاله بررسی همه گره‌های مجموعه در کمترین قیمت بود. همچنین این مسئله از چندین مدل تراکم و غیرتراکم استفاده می‌کند. ماریو مستریا و همکاران [۱۴] مطالعه‌ای در مورد مسئله‌ی فروشنده‌ی دوره‌گرد خوشه‌ای با استفاده از الگوریتم ابتکاری هیبریدی، انجام داده‌اند. این مسئله از نوع مسائل ان-پی سخت بود که در آن رئوس به خوشه‌ها تقسیم‌بندی می‌شدند و هر خوشه به طور پیوسته بازبینی می‌شد. مارک و همکاران [۱۵] در حل مسئله فروشنده‌ی دوره‌گرد چندهدفه از الگوریتم تجزیه استفاده کردند. برای یافتن تقریبی از مرز پارتو این الگوریتم از چارچوب مدلی دو فازه استفاده می‌کند.

هزیم و همکاران [۱۶] برای حل مسئله فروشنده دوره گرد از الگوریتم بهینه‌سازی حشره میوه استفاده کردند. برای حل از یک الگوریتم بهبود یافته FOA استفاده گردید. این الگوریتم شامل بو کردن میوه از طریق بینی (فاز استخوانی) و سپس از طریق فاز بینایی به آن نزدیک می‌شوند. بیتر و همکاران [۱۷] روشی برای فروشنده دوره‌گرد اقلیدوسی با استفاده از سیستم شبکه عصبی موازی، ارائه دادند. روش تقسیم و حل در شبکه عصبی

خودسازمان‌ده برای حل فروشنده دوره‌گرد اقلیدوسی استفاده شد. ایراکلیس و همکاران [۱۸] برای حل مسئله‌ی فروشنده‌ی دوره‌گرد چندهدفه از الگوریتم تکاملی هیبریدی استفاده کردند. دوتا براساس الگوریتم تکامل تفاضلی و سومی‌براساس الگوریتم ژنتیک مرتب سازی نامغلوب می‌باشد. زوتانگ و همکاران [۱۹] گوناگونی مسئله‌ی فروشنده‌ی چندهدفی نامشخص با متغیرهای مبهم را بررسی کردند . این کار دلیل نامشخص بودن راه‌حل‌ها بیان کرده و روش جدیدی است که برای ارائه مسیر کارآمدی پارتو به کار می‌رود.

۳. اهداف تحقیق

- طراحی و توسعه ی الگوریتم ژنتیک بر پایه ی چارچوب بهینه سازی چند هدفه برای حل مسئله‌ی فروشنده‌ی دوره‌گرد.
- تست و آزمایش کارآمدی چارچوب الگوریتم ژنتیک ارائه شده با تعدادی از نمونه های مسئله فروشنده دوره گرد کتابخانه محک استاندارد.

۳,۱. بهینه سازی چند هدفه مسئله فروشنده دوره‌گرد

مسئله فروشنده دوره‌گرد مسئله‌ای است که در آن تعدادی شهر وجود دارد و هدف پیدا کردن کوتاه‌ترین مسیر از بین تمام مسیرهای ممکن است به طوری که فرد از تمـام شـهرها و از هـر یـک تنها یک بار گذشته باشد.

شهر ها از ۱تا n شماره گذاری می‌شوند، فاصله‌ی بین دو شهر i و j به صورت فاصله $(i \neq j)$ [i][j] ، و هزینه ی بین دو شهر به صورت هزینه $(i \neq j)$ [i][j] می‌باشند. مدل ریاضی مسئله‌ی فروشنده‌ی چندهدفه را می‌توان به صورت رابطه ی (۱) نشان داد:

$$minimize\ f(x)\begin{cases} f1(x) = c_1^{x(n),x(1)} + \sum_{i=1}^{n-1} c_1^{x(i),x(i+1)} \\ f1(x) = c_1^{x(n),x(1)} + \sum_{i=1}^{n-1} c_1^{x(i),x(i+1)} \\ f1(x) = c_1^{x(n),x(1)} + \sum_{i=1}^{n-1} c_1^{x(i),x(i+1)} \end{cases} \quad (۱)$$

که در آن N تعداد شهر، Cj ماتریس هزینه با توجه به هدف j و X جایگشت N شهر می‌باشند. مسئله‌ی فروشنده‌ی دوره‌گرد برای تخمین توالی غیرتکراری از ۱ و ۲ و .... N می‌باشد، به طوری که در آن شهرها به ترتیب از ۱ تا N شماره‌گذاری شده‌اند . در مسئله فروشنده دوره‌گرد متقارن، Cij برابر با Cji می‌باشد $(c_{ij} = c_{ji})$. مجموع جایگشت‌ها نشان می‌دهد که شهری که بازدید می‌شود ، فواصل جستجو را به حداقل رسانده و شامل تمام جایگشت‌های ممکن برای شهرهای متوالی از ۱ تا N می‌باشد.

همه‌ی راه‌حل‌ها برای فروشنده دوره‌گرد متقارن همراه با جایگشت چرخشی معکوس با استفاده از رابطه‌ی (۲) ایجاد می‌شوند.

$$\frac{(n-1)!}{2} \quad (۲)$$

فاصله‌ی اقلیدسی برای ایجاد ماتریس هزینه و فاصله استفاده می‌شود. فاصله‌ی اقلیدسی یک خط صاف بین دو گره می‌باشد. معیار اندازه‌گیری می‌تواند به شکل یک‌بعدی، دوبعدی و سه‌بعدی باشد. در این تحقیق، فاصله‌های اقلیدسی دو بعدی مورد استفاده قرار گرفته است. فاصله‌ی اقلیدسی برای اندازه‌ی دوبعدی بین هر دو شهر با مختصات (x_1, y_1) و (x_2, y_2)بااستفاده از رابطه‌ی (۳) محاسبه می‌شود:

$$Euclidean\ distances = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \quad (۳)$$

۳,۲. الگوریتم ژنتیک

الگوریتم ژنتیک، روشی برای حل مسائل بهینه‌سازی قیددار و بدون قید است که بر مبنای نظریه انتخاب طبیعی (فرآیندی که تکامل زیست شناسی را پیش می‌برد) عمل می‌کند.

از الگوریتم ژنتیک برای حل مسائل مختلف بهینه‌سازی که الگوریتم های استاندارد بهینه سازی برای حل آنها مناسب نیست، استفاده می‌شود. به عنوان نمونه‌ای از این دست مسائل می‌توان به مسائلی اشاره کرد که در آن تابع هدف ناپیوسته، مشتق ناپذیر، تصادفی و یا غیرخطی از مرتبه بالا می‌باشد. الگوریتم ژنتیک همچنین می‌تواند در حل مسائل برنامه ریزی عدد صحیح مختلط، که در آن برخی از اجزا محدود به مقادیر صحیح هستند نیز استفاده شود. در زیر پارامترهای الگوریتم ژنتیک ارائه می‌گردد.

اندازه‌ی جمعیت: جمعیت فرایندی می‌باشد که با دسته‌ای از افراد شروع می‌شود و هر فرد راه‌حلی برای هر مسئله‌ی مشخص ارائه می‌دهد . اندازه ی جمعیت تعداد کروموزوم‌ها در جمعیت را نشان می‌دهد و بستگی به نوع رمز گشایی و نوع مسئله دارد.

انتخاب: انتخاب اولین عملگری است که برروی جمعیت مورد استفاده قرار می‌گیرد. عامل انتخاب بهترین راه‌حل برای ایجاد جمعیت جدید از جمعیت فعلی می‌باشد. این عملگر، رشته‌ها یا کروموزوم‌های خوب جمعیت را انتخاب کرده و با تکثیر آنها جمعیت جدید شکل می‌گیرد.

تابع برازندگی: تابعی است که ورودی را به عنوان کاندید راه‌حل انتخاب کرده و خروجی با توجه به مسئله فروشنده دوره‌گرد تولید می‌شود. به طور کلی، یک تابع برازندگی، در ابتدا با استفاده از تابع هدف فرموله می‌شود و در عملیات ژنتیکی متوالی در نسل‌های

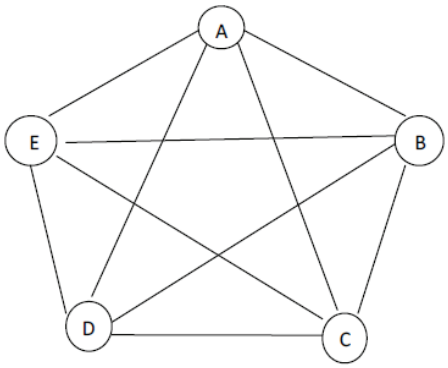
الگوریتم ژنتیک مورد استفاده قرار می‌گیرد. این تابع در واقع میزان خوب بودن جواب مسئله را نشان می‌دهد.

عملگر ترکیب یا آمیزش: ترکیب یا آمیزش فاز حیاتی و مهم در الگوریتم ژنتیک می‌باشد. این عملگر همراهی بین افراد را نشان می‌دهد. ترکیب فرایند گرفتن دو کروموزوم و ایجاد یک تولید مثل می‌باشد.

جهش: در فرایند جهش، ژن‌هایی که برازندگی کمتری دارند جهش می‌یابند. عملگر جهش مکانیزم هوشمندانه‌ای برای جستجوی محلی در فضای جستجوی جواب‌های مسئله به شمار می‌آید. با استفاده از عملگر جهش در الگوریتم ژنتیک، رشته‌ها یا کروموزوم‌های جدیدی در همسایگی رشته یا کروموزوم فعلی ایجاد می‌شوند. در نتیجه، عملگر جهش سبب ایجاد مکانیزم جستجوی محلی اطراف کروموزوم‌های فعلی می‌شود. همچنین، عملگر جهش سبب حفظ تنوع در جمعیت کروموزوم‌ها یا جواب‌های کاندید می‌شود.

۳،۳. مثالی از مسئله فروشنده دوره گرد با الگوریتم ژنتیک

P=(C1,C2,C3,…..CN) را در نظر بگیرید. یعنی فروشندگان از C1 به C2 به C3 به Cn منتقل می‌شوند. این شهرها A،B،C،D،E هستند. سفر از A شروع می‌شود و در A نیز پایان می‌یابد. فاصله



بین شهرها مطابق شکل ۱ در جدول ۱ نشان داده است.

شکل ۱ - پیوند از TSP

جدول ۱ - فاصله‌ی بین هر جفت شهر در مثال فوق

۱	A	B	۷
۲	A	C	۵
۳	A	D	۹
۴	A	E	۹
۵	B	C	۷
۶	B	D	۲
۷	B	E	۸
۸	C	D	۴
۹	C	E	۳
۱۰	D	E	۶

در این مثال هدف پیدا کردن کمترین مقدار بهینه سراسری می‌باشد. در زیر توضیحی در مورد محاسبات الگوریتم ژنتیک ارائه شده است جدول ۲ شش کروموزوم تصادفی را برای مثال ارائه می‌دهد.

جدول ۲ - جمعیت اولیه با شش کروموزوم تصادفی

۱	B	D	E	C
۲	D	B	E	C
۳	C	B	D	E
۴	E	B	C	D
۵	E	C	B	D
۶	C	D	E	B

برای هر کروموزوم از جمعیت بر اساس طول مسیر برازندگی محاسبه شده است جدول۳ محاسبه برازندگی را انجام می‌دهد.

جدول ۳ - محاسبه برازندگی

	مسیر					برازندگی
۱	AB	BD	DE	EC	CA	
	۷	۲	۶	۳	۵	۲۳
۲	AD	DB	BE	EC	CA	
	۹	۲	۸	۳	۵	۲۷
۳	AC	CB	BD	DE	EA	
	۵	۷	۲	۶	۹	۲۹
۴	AE	EB	BC	CD	DA	
	۹	۸	۷	۴	۹	۳۷
۵	AE	EC	CB	BD	DA	
	۹	۳	۷	۲	۹	۳۰
۶	AC	CD	DE	EB	BA	
	۵	۴	۶	۸	۷	۳۰

انتخاب کروموزوم به دلیل مسئله فروشنده دوره گرد به طور مطلوب انجام می‌شود که کروموزوم‌هایی که از برازندگی کمتری برخوردار هستند احتمال بیشتری برای تولید مثل دارند. جدول۴ احتمال انتخاب هر کروموزوم را محاسبه میکند.

جدول ۴ - احتمال انتخاب هر کروموزوم

	برازندگی	نرمال سازی	احتمال
۱	۲۳	۰,۰۴۳۴۷۸	۰,۲۰۸۳۳۷
۲	۲۷	۰,۰۳۷۰۳۷	۰,۱۷۷۴۷۲
۳	۲۹	۰,۰۳۴۴۸۳	۰,۱۶۵۲۳۳
۴	۳۷	۰,۰۲۷۰۲۷	۰,۱۲۹۵۰۷
۵	۳۰	۰,۰۳۳۳۳۳	۰,۱۵۹۷۲۵
۶	۳۰	۰,۰۳۳۳۳۳	۰,۱۵۹۷۲۵

با توجه به انتخاب کروموزوم‌هایی با برازندگی بیشتر انتظار می‌رود که با اعمال عملگر ترکیب کروموزوم‌های حاصل از ارزش برازندگی بیشتری برخوردار باشند. در اینجا مقدار احتمال ترکیب برابر است با ۰,۲۵ است. جدول ۵ مثالی از عملگر ترکیب را نشان می‌دهد.

جدول ۵- اعمال عملگر ترکیب

قدیم	جدید	کروموزوم			
۱	۲	D	B	E	C
۲	۱	B	D	E	C
۳	۳	C	B	D	E
۴	۶	C	D	E	B
۵	۵	E	C	B	D
۶	۴	E	B	C	D

در این بخش از فرآیند جهش انجام می‌شود. این جهش برای تبادل ژن با ژن‌های دیگر کروموزوم‌ها کار می‌کند .نتایج ثابت شده ارزش برازندگی را افزایش می‌دهد .اگر یک ژن در انتهای کروموزوم رد و بدل شود ، این ژن با ژن اول مبادله می‌شود. یک پارامتر وجود دارد که تعیین می‌کند چند ژن جهش می‌یابند. نرخ جهش ۰/۲ است. جدول ۶ اعمال عملگر جهش را نشان می‌دهد.

در نسل اول دیده شده است که کوچک‌ترین مقدار برازندگی وجود دارد که تغییر نمی‌کند. اگر محاسبه به نسبت NTH ادامه یابد، فرض می‌شود که کمترین مقدار برازندگی بدون تغییر باقی می‌ماند. اگرچه محاسبه به اندازه کافی برای نسل اول تعریف شده است، یک راه حل نزدیک بهینه از فرآیند الگوریتم ژنتیک بالا، نتیجه نهایی مسیر با کوتاه‌ترین فاصله بهینه A, B, D, E, C, A است.

۳,۴. چارچوب ارائه شده برای مسئله فروشنده دوره‌گرد

الگوریتم ژنتیک یک روش هوش محاسباتی می‌باشد که در حوزه‌ی علوم کامپیوتر، برای تشخیص دادن راه‌حل‌های مقایسه‌ای، برای بهینه‌سازی مسائل مورد استفاده قرار می‌گیرد. مسئله‌ی فروشنده‌ی دوره‌گرد چندهدفه، در این کار مد نظر قرار گرفته است. مسئله‌ی بهینه‌سازی چندهدفه یکی از بهترین مسائل بهینه‌سازی می‌باشد. فاصله‌ی اقلیدسی برای محاسبه‌ی فاصله‌ی بین نقاط مورد استفاده قرار می‌گیرد. پارامترهای الگوریتم ژنتیک اعمال می‌شوند و کوتاه‌ترین مسیر در بین فواصل آن پیدا خواهند شد. بیشترین روش‌های بکارگرفته شده در مسائل چند هدفه، شامل بکارگیری کلیه‌ی اهداف در تابع و تضمین تولید جواب‌های بهینه پارتو می‌باشند. از نرم افزار متلب در این تحقیق استفاده شده است.

جدول ۶ - اعمال عملگر جهش

	مسیر					برازندگی
۱	AD	DB	BC	CE	EA	
	۹	۲	۷	۳	۹	۳۰
۲	AB	BD	DE	EC	CA	
	۷	۲	۶	۳	۵	۲۳
۳	AC	CE	ED	EB	BA	
	۵	۳	۶	۲	۷	۲۳
۴	AE	EC	CB	BD	DA	
	۹	۳	۷	۲	۹	۳۰
۵	AD	DB	BC	CE	EA	
	۹	۲	۷	۳	۹	۳۰
۶	AE	ED	DB	BC	CA	
	۹	۶	۲	۷	۵	۲۹

جدول ۷ - پارامترهای به کار رفته در الگوریتم ژنتیک

شماره	پارامترهای الگوریتم	مقادیر
۱	جمعیت	بر مبنای تعداد شهرها
۲	نسل های تولیدی	N فرزندان، تولید می‌شوند زمانی که ساز جمعیت n باشد.
۳	عملگر انتخاب(تولید مثل)	انتخاب مسابقه‌ای
۴	عملگر ترکیب	ترکیب تک نقطه‌ای
۵	عملگر جهش	جهش جابجایی
۶	رضایت‌بخشی خاتمه یافتن	پایان تکرار

شبه کد الگوریتم ژنتیک در شکل ۲ آورده شده است [ایجاد الگوریتم ژنتیک].

شروع

۱- آغاز به کار : ایجاد جمعیت تصادفی از راه‌حل‌های P کروموزوم‌ها

۲- برای هر فرد i ϵ p

i) محاسبه برازندگی

۳- تکرار

۴- برای L = i به تعداد نسل‌ها ، (i) انتخاب عملیات تصادفی (ترکیب یا جهش)

۵- اگر ترکیب انتخاب شد

i)دو والد ia و ib به صورت تصادفی انتخاب شود

ii) در اثر ترکیب فرزند ایجاد شود (ترکیب ib و ia)=ic

۶- در غیر این صورت اگر جهش انتخاب شد

i) به صورت تصادفی یک کروموزوم i را انتخاب کنید

ii) تولید مثل ic را ایجاد کنید i جهش یافته = ic

۷- خاتمه اگر

۸- جستجوی محلی (ic)

i)برازندگی تولید مثل ic محاسبه شود

ii) اگر ic بهتر باشد ، سپس بدترین کروموزوم را با ic جایگزین کنید.

iii) i جدید را وارد کنید.

۹- انتها را چک کنید.

۱۰- پایان

۳،۴،۱. مراحل الگوریتم ژنتیک در حل مسئله فروشنده دوره‌گرد

۱- داده‌های ورودی برای نمونه‌های چندهدفه را بخوانید (کرولاک و اقلیدس).

۲- فاصله را برای نمونه‌های اقلیدس با استفاده از فرمول اقلیدس پیدا کنید.

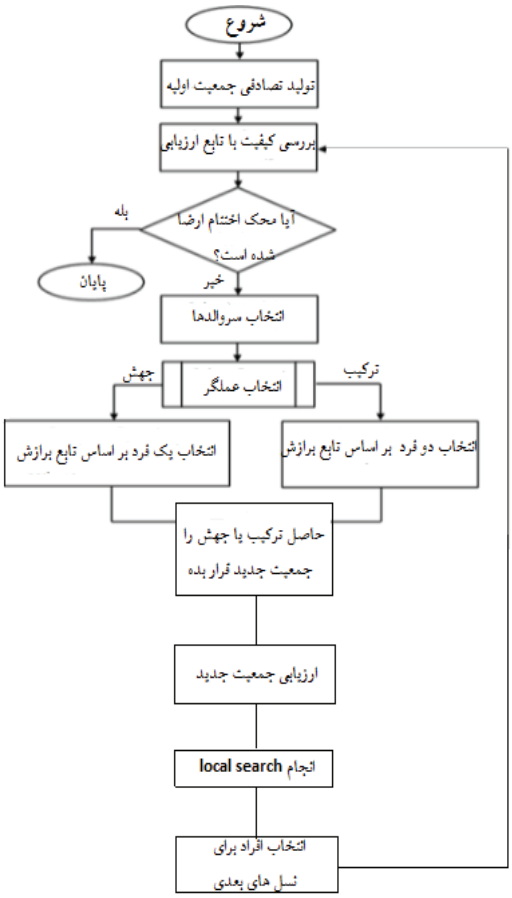
۳-ماتریس با توجه به شهر ارائه شده تشکیل شود.

۴-الگوریتم ژنتیک برای ایجاد کردن ماتریس بکار می‌رود.

۴-۱- جمعیت را به صورت تصادفی از ماتریس شروع کنید.

۴-۲- برازندگی جمعیت را از طریق ماتریس تعیین کنید.

۴-۳- ترکیب و جهش بر اساس افرادی که انتخاب شده‌اند انجام می‌پذیرد.



شکل ۲ - مراحل اجرای الگوریتم ژنتیک در مسئله حاضر

۴-۴-برازندگی برای شهر تازه توسعه‌یافته از طریق ماتریس محاسبه می‌شود.

۴-۵- بهترین مسیر همراه با فاصله و نازلترین هزینه مربوط به آن را چاپ کنید.

۴.بحث در مورد نتایج

نمونه‌های داده، برای این کار تحقیقی از مجموعه داده‌های کتابخانه‌ای محک استاندارد جمع آوری شده‌اند. نمونه‌های مسئله فروشنده دوره‌گرد برای تست از نمونه‌های کورلاک با ۱۰۰ شهر، ۱۵۰ شهر، ۲۰۰ شهر و همچنین از اقلیدس با ۱۰۰ شهر، ۳۰۰ شهر، ۵۰۰ شهر از جدول کتابخانه‌ای [رینلت ، ۱۹۹۵] انتخاب شده‌اند. نمونه‌های کرولاک AB ۱۰۰ به عنوان اهداف هزینه و فاصله برای جفت اول در نظر گرفته شد ، سپس کرولاک AC ۱۰۰ ، کرولاک AD ۱۰۰ و.... در نظر گرفته شدند.

جدول ۸ نتایج بدست آمده از بهینه‌سازی چندهدفه از نمونه‌های کورلاک با ۱۰۰ ، ۱۵۰ و ۳۰۰ شهر را نشان می‌دهد.

جدول ۹ نتایج بدست آمده برای نمونه‌های اقلیدسی ۱۰۰ ABتا EF۱۰۰ ، AB۳۰۰ تا EF۳۰۰ و AB۵۰۰ تا EF۵۰۰ با استفاده از الگوریتم ژنتیک برای مسئله‌ی فروشنده‌ی دوره‌گرد و بهترین راه‌حل بهینه بر اساس TSPLIB را نشان می‌دهد.

شکل ۳ جواب‌های بهینه پارتو که از نمونه‌های کورلاک AB۱۰۰ بدست آمده است را نشان می‌دهد.

شکل ۴ جواب‌های بهینه پارتو که نمونه‌های کورلاک AB۱۵۰ بدست آمده را نشان می‌دهد.

شکل ۵ جواب‌های بهینه پارتو که نمونه‌های کورلاک AB۲۰۰ به دست آمده را نشان می‌دهد.

شکل ۶ جواب‌های بهینه پارتو که از اقلیدس CD۱۰۰ بدست آمده را نشان می‌دهد.

شکل ۷ جواب‌های بهینه پارتو که برای نمونه‌های اقلیدس AB۳۰۰ بدست آمده را نشان می‌دهد.

شکل ۸ جواب‌های بهینه پارتو که برای نمونه‌های اقلیدس AB۵۰۰ بدست آمده را نشان می‌دهد.

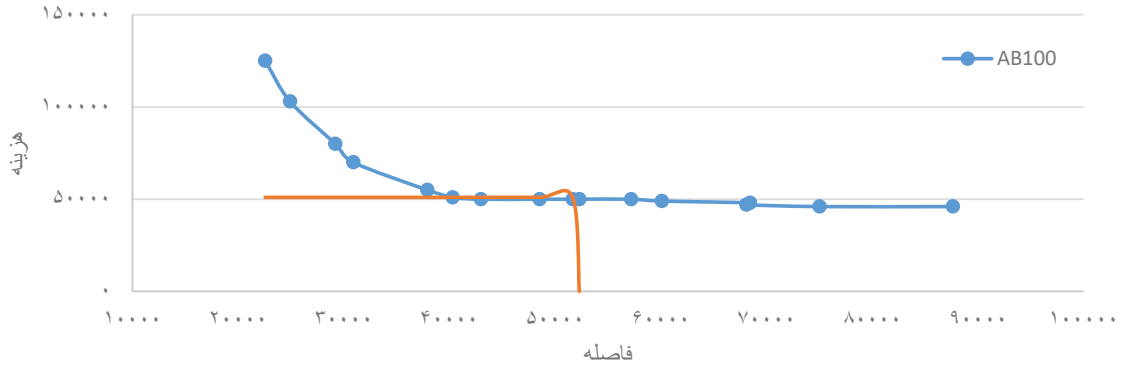
جدول ۸ - نتایج اجرا برای فروشنده دوره‌گرد چندهدفه (نمونه‌های کرولاک)

شماره	مورد	سایز	بهترین تا کنون		الگوریتم ژنتیک بهینه	
			فاصله	هزینه	فاصله	هزینه
۱	کرولاک ۱۰۰ AB	۱۰۰	۵۰,۵۲۳	۵۱,۴۴۷	۵۰۶۵۴,۴	۴۹,۷۷۲
۲	کرولاک ۱۰۰ AC	۱۰۰	۵۰,۸۴۰	۵۱,۸۳۳	۵۰,۳۸۳	۵۱۸۳۳,۴۸
۳	کرولاک ۱۰۰ AD	۱۰۰	۵۷,۸۰۳	۴۴,۱۵۹	۵۷۵۳۵,۷	۴۳۹۱۷,۴
۴	کرولاک ۱۰۰ AE	۱۰۰	۴۸,۸۸۶	۴۸,۸۹۱	۴۸۴۰۷,۰۷	۴۹۴۶۲,۳۵
۵	کرولاک ۱۰۰ BC	۱۰۰	۴۸,۴۷۳	۵۰,۴۸۲	۴۸۹۸۹,۳۳	۵۰۰۷۰,۸۱
۶	کرولاک ۱۰۰ BD	۱۰۰	۵۲,۹۵۶	۴۷,۸۷۲	۵۱۲۸۴,۵۵	۴۹۱۲۹۶,۰۶
۷	کرولاک ۱۰۰ BE	۱۰۰	۵۰,۳۸۸	۵۰,۶۴۸	۵۱۶۷۹,۹۸	۴۹۷۳۳,۵۶
۸	کرولاک ۱۰۰ CD	۱۰۰	۴۸,۷۱۱	۴۶,۵۹۴	۴۷۷۷۶,۸۱	۴۷۶۹۲,۲۴
۹	کرولاک ۱۰۰ CE	۱۰۰	۴۹,۰۳۲	۵۰,۶۹۹	۵۰۲۹۸,۷۷	۴۹۲۸۵,۶۶
۱۰	کرولاک ۱۰۰ DE	۱۰۰	۵۰,۵۹۴	۵۰,۳۰۰	۵۰۰۸۹,۳۲	۵۰۸۸۷,۶۸
۱۱	کرولاک ۱۵۰ AB	۱۵۰	۴۹,۱۰۸	۴۹,۵۶۹	۴۹۲۰۸,۹۸	۴۹۴۳۱,۰۳
۱۲	کرولاک ۲۰۰ AB	۲۰۰	۸۲۳۴۱	۸۲,۰۷۶	۸۳۴۱۳,۸۷	۸۱۲۳۴,۲۳

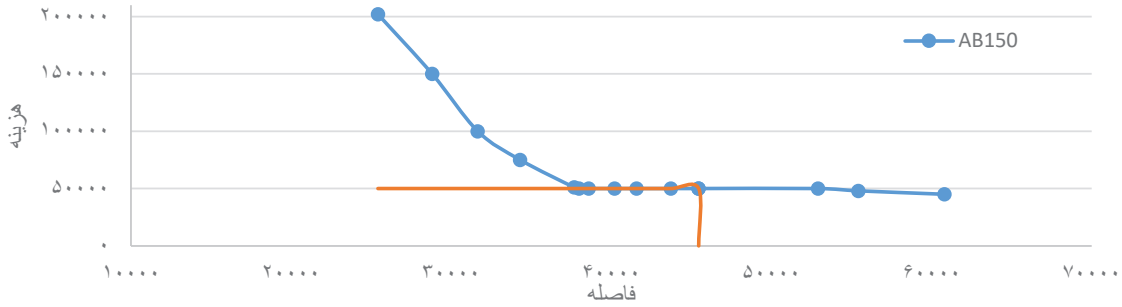
جدول ۹ - نتایج اجرا برای فروشنده دوره‌گرد چندهدفه (نمونه‌های اقلیدسی)

شماره	مورد	سایز	بهترین تا کنون		الگوریتم ژنتیک بهینه	
			فاصله	هزینه	فاصله	هزینه
۱	اقلیدس ۱۰۰ AB	۱۰۰	۵۴,۱۷۴	۵۵,۲۵۰	۵۳,۶۴۳	۵۵,۷۷۷
۲	اقلیدس ۱۰۰ CD	۱۰۰	۵۶,۶۲۶	۵۶,۷۳۱	۵۷,۹۳۴	۵۶,۴۹۳
۳	اقلیدس ۱۰۰ EF	۱۰۰	۵۴,۲۵۸	۵۳,۵۳۶	۵۴,۲۱۳	۵۳,۵۸۱
۴	اقلیدس ۳۰۰ AB	۳۰۰	۱۲۰,۱۴۹	۱۲۰,۲۰۸	۱۲۰,۳۶۸	۱۱۹,۹۴۹
۵	اقلیدس ۳۰۰ CD	۳۰۰	۱۱۹,۴۵۳	۱۱۹,۴۵۷	۱۱۹,۸۰۵	۱۱۹,۱۶۵
۶	اقلیدس ۳۰۰ EF	۳۰۰	۱۱۹,۷۲۶	۱۱۹,۷۳۵	۱۲۰,۰۰۲	۱۱۹,۴۹۰
۷	اقلیدس ۵۰۰ AB	۳۰۰	۱۷۲,۳۲۶	۱۷۲,۳۴۴	۱۷۲,۶۲۸	۱۷۲,۰۵۹

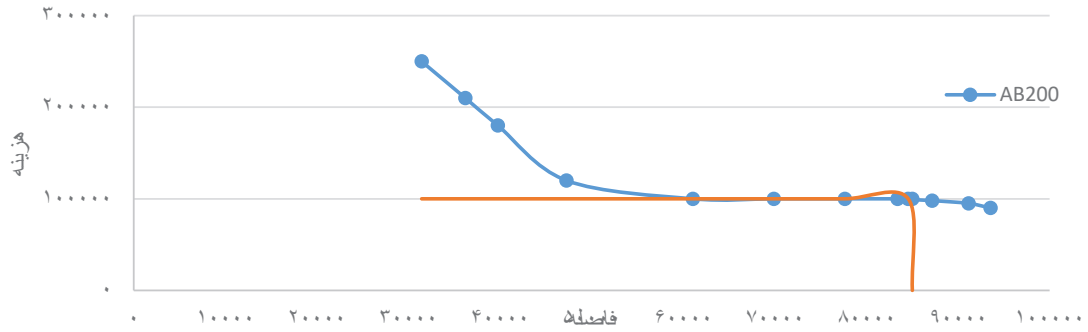
۸	اقلیدس CD ۵۰۰	۳۰۰	۱۷۴,۵۴۳	۱۷۴,۵۴۶	۱۷۵,۱۴۱	۱۷۳,۹۰۳
۹	اقلیدس EF ۵۰۰	۳۰۰	۱۷۵,۱۴۴	۱۷۵,۲۱۸	۱۷۶,۲۸۰	۱۷۴,۹۸۷



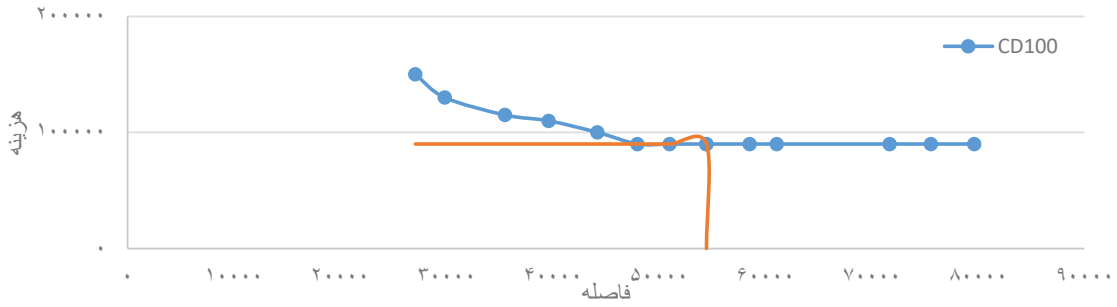
شکل ۳ – جواب‌های بهینه پارتو نمونه‌های کورلاک AB۱۰۰



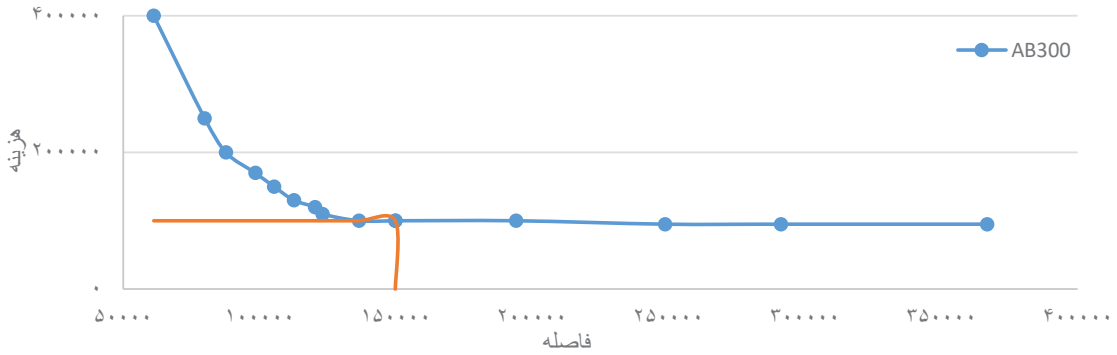
شکل ۴ – جواب‌های بهینه پارتو نمونه‌های کورلاک AB۱۵۰



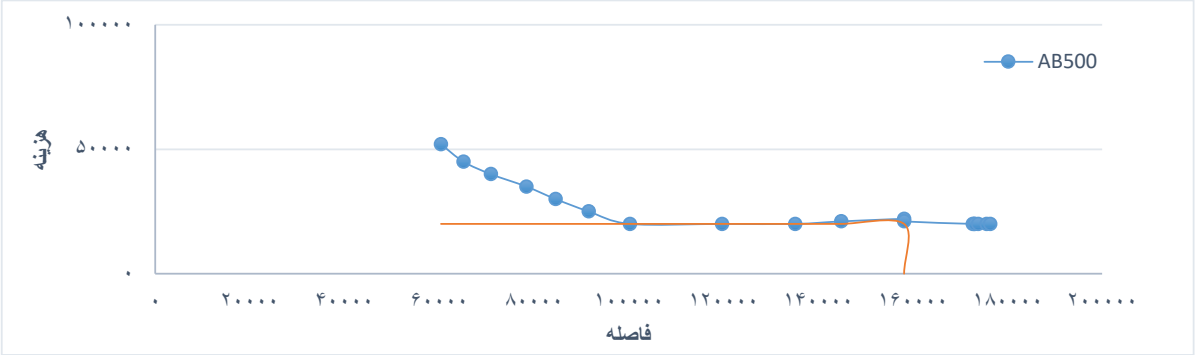
شکل ۵ – جواب‌های بهینه پارتو نمونه‌های کورلاک AB۲۰۰



شکل ۶ – جواب‌های بهینه پارتو نمونه‌های اقلیدس CD۱۰۰



شکل ۷ – جواب‌های بهینه پارتو نمونه‌های اقلیدس AB۳۰۰



شکل ۸ – جواب‌های بهینه پارتو نمونه‌های اقلیدس AB۵۰۰

۵. نتیجه‌گیری

مسئله فروشنده دوره‌گرد یک مسئله کاری مهم در زندگی روزمره می‌باشد و بهینه‌سازی فاصله و هزینه در این مسئله منجر به عملیات و کارهای کارآمد، قابل اطمینان و اقتصادی می‌شود. این کار تحقیقی، عمدتاً بر روی مسئله‌ی فروشنده‌ی چندهدفه تمرکز دارد و با استفاده از الگوریتم ژنتیک بهینه شده است. مسئله‌ی فروشنده‌ی دوره‌گرد چندهدفه همه‌ی توابع هدف را بهینه می‌کند و بهترین راه‌حل‌های مطلوب را پیدا می‌کند. مسئله‌ی فروشنده‌ی چندهدفه جواب‌های تقریبی بهینه پارتو را برای همه‌ی نمونه‌های داده موردآزمایش ارائه می‌دهد . الگوریتم ژنتیک نزدیک‌ترین راه‌حل بهینه را در طی یک دوره زمانی منطقی با بهینه‌سازی همه‌ی اهداف ارائه می‌دهد. نمونه‌های آزمایش برای این مسئله‌ی چندهدفه فروشنده‌ی دوره‌گرد از مجموعه داده‌های کتابخانه‌ای محک استاندارد جمع آوری شده است. نتایج مسئله‌ی چندهدفه فروشنده‌ی دوره‌گرد با استفاده از الگوریتم ژنتیک، نزدیک‌ترین راه‌حل‌های مطلوب به نتایجی که در مجموعه ی TSP ارائه شده است را بیان می‌کند.

۶. مراجع

[1] Gupta, S., & Panwar, P. (2013). Solving travelling salesman problem using genetic algorithm. International Journal of Advanced Research in Computer Science and Software Engineering, 3(6).

[2] Pandey, S., & Kumar, S. (2013). Enhanced artificial bee colony algorithm and it's application to travelling salesman problem. HCTL Open International Journal of Technology Innovations and Research, 2.

[3] Matai, R., Singh, S. P., & Mittal, M. L. (2010). Traveling salesman problem: An overview of applications, formulations and solution approaches. November 2010.

[4] Plante, R. D., Lowe, T. J.,&Chandrasekaran, R. (1987). The productmatrix traveling salesman

problem: An application and solution heuristics. Operations Research, 35, 772–783.
[5] Miranda, P. A., Blazquez, C. A., Obreque. C., Maturana-Ross, J.,&Gutierrez-Jarpa, G. (2018). The bi-objective insular traveling salesman problem with maritime and ground transportation costs. European Journal of Operational Research.
[6] Arigliano, A., Ghiani, G., Grieco, A., Guerriero, E., & Plana, I. (2017). Time-dependent asymmetric traveling salesman problem with time windows: Properties and an exact algorithm. Discrete Applied Mathematics.
[7] Nuriyev, U.,Ugurlu,O.,&Nuriyeva, F. (2018). Self-organizing iterative algorithm for travelling salesman problem. IFAC PapersOnLine 51(30), 268–270.
[8] Venkatesh, P., Srivastava, G., & Singh, A. (2018). A general variable neighborhood search algorithm for the k-traveling salesman problem. In 8th International Conference on Advances in Computing and Communication ICACC, Procedia Computer Science, vol. 143 (pp. 189–196).
[9] Pereira, A. H., & Urrutia, S. (2018). Formulations and algorithms for the pickup and delivery traveling salesman problem with multiple stacks. Computers and Operations Research.
[10] Sun, L.,Karwan, M. K.,&Diaby, M. (2018). The indefinite period traveling salesman problem. European Journal of Operational Research.
[11] Kinable, J., Smeulders, B., Delcour, E., & Spieksma, F. C. R. (2017). Exact algorithms for the equitable traveling salesman problem. European Journal of Operational Research.
[12] Defryn, C., & Sörensen, K. (2017). Multi-objective optimisation models for the travelling
[13] salesman problem with horizontal cooperation. European Journal of Operational Research.
[14] Bernardino, R., & Paias, A. (2017). Solving the family traveling salesman problem. European Journal of Operational Research.
[15] Mestria, M. (2017). New hybrid heuristic algorithm for the clustered traveling salesman problem. Computers & Industrial Engineering.
[16] Cornu, M., Cazenave, T., & Vanderpooten, D. (2017). Perturbed decomposition algorithm applied to the multi-objective Traveling Salesman Problem. Computers and Operation Research.
[17] Iscan, H.,&Gunduz,M. (2017). An application of fruit fly optimization algorithm for traveling salesman problem. In 8th International Conference on Advances in Information Technology, IAIT2016, Procedia Computer Science, vol. 111, (pp. 58–63).
[18] Avs, B., & Aliabadi, D. E. (2015). Parallelized neural network system for solving Euclidean traveling salesman problem. Applied Soft Computing 34, 862–873.
[19] Psychas, I.-D., Delimpasi, E., & Marinakis, M. (2015). Hybrid evolutionary algorithms for the multiobjective traveling salesman problem. Expert Systems with Applications.
[20] Wang, Z.,Guo, J., Zheng, M.,&Wanga,Y. (2015).Uncertain multiobjective traveling salesman problem. European Journal of Operational Research 241, 478–489.
[21] Kumar, S., Jain, S., & Sharma, H. (2018). Genetic Algorithms. In Advances in Swarm Intelligence for Optimizing Problems in Computer Science (pp. 27–52). Chapman and Hall/CRC.
[22] elib.zib.de/pub/mp-testdata/tsp/tsplib/stsp-sol.html.visited on 7/16/2021.