فصلنامه مهندسی حمل و نقل، دوره ۱۲، شماره ٤، تابستان ۱٤۰۰، ص. ۹۹۷-۹۷۳
مقاله علمی-پژوهشی

یافتن کوتاهترین دور هامیلتونی با استفاده از رویکرد ترکیبی هوش جمعی بر پایه شبکه‌های پیچیده

هادی محمدی، دانشجوی دکتری مهندسی کامپیوتر، واحد میبد، دانشگاه آزاد اسلامی، میبد، ایران
کمال میرزائی (مسئول مکاتبات)، عضو هیات علمی گروه مهندسی کامپیوتر، واحد میبد، دانشگاه آزاد اسلامی، میبد، ایران
Email: k.mirzaie@maybodiau.ac.ir
محمدرضا ملاخلیلی میبدی، عضو هیات علمی گروه مهندسی کامپیوتر، واحد میبد، دانشگاه آزاد اسلامی، میبد، ایران

دریافت: ۱۳۹۸/۱۲/۰٤ پذیرش: ۱۳۹۹/۰٦/۱۷

چکیده

در این مقاله از دور هامیلتونی در یک مسئله استاندارد و نظری بنام مسئله فروشنده دوره گرد و یک مسئله کاربردی بنام یافتن کوتاهترین مسیر هامیلتونی برای پیمودن تمام استان‌های ایران استفاده‌شده‌است. برای حل این گونه مسائل می‌توان از الگوریتم‌های هوش جمعی استفاده‌کرد که از عوامل طبیعی، زیست محیطی و اجتماعی نشأت‌گرفته‌اند. الگوریتم بهینه‌سازی ازدحام ذرات یکی از الگوریتم‌های هوش جمعی است. در روش پیشنهادی، به منظور بهبود نتایج هر ذره از جستجوی محلی در روند جستجو و برای افزایش تبادل اطلاعات بهتر میان ذرات و انتخاب موقعیت بعدی مناسب‌تر هر ذره، از شبکه پیچیده، استفاده‌می‌شود. در این شبکه گره‌ای که راه‌حلی بهتری در آن نگهداری‌می شود همواره درجه آن گره بزرگ تر می شود. در شبکه پیچیده از دو سنجه درجه و درجه همسایگی برای یافتن راه حل بهتر استفاده‌شده‌است. برای مقایسه نتایج از مسائل استاندارد TSPLib استفاده‌شده که نتایج حاکی از هزینه بهتر روش بهینه‌سازی ازدحام ذرات با جستجوی محلی شبکه‌ای پیچیده نسبت به بهینه‌سازی ازدحام ذرات با جستجوی محلی و ازدحام ذرات استاندارد است، همچنین، درصد خطا نسبت به بهترین جواب موجود در TSPLib به ترتیب در الگوریتم‌های بهینه‌سازی ازدحام ذرات با جستجوی محلی شبکه‌ای پیچیده و بهینه‌سازی ازدحام ذرات با جستجوی محلی نسبت به روش ازدحام ذرات استاندارد، کاهش‌داشته‌است. به طور نمونه، برای حل مسئله ST70 در الگوریتم های بهینه سازی ازدحام ذرات شبکه ای و پایه میانگین هزینه حل مسئله به ترتیب ۷۰۵ و ۷۹۷ می‌باشد.

واژه های کلیدی: کوتاهترین دور هامیلتونی، فروشنده دوره گرد، الگوریتم ازدحام ذرات، الگوریتم‌های هوش جمعی، شبکه پیچیده

۱. مقدمه

انسان در طبیعت و مسائل مهندسی همواره به دنبال بهترین جواب است که به طور خلاصه به آن بهینه‌سازی، گفته‌می‌شود. یکی از مسائل بهینه‌سازی کلاسیک، یافتن کوتاهترین دور هامیلتونی یا مسئله فروشنده دوره‌گرد است که تا کنون، توسط الگوریتم‌های پیشنهادی مختلف حل‌شده‌است. این مسئله یکی از پایه‌ترین مسائل مسیریابی و برنامه‌ریزی حمل و نقل محسوب‌می‌شود. مدار هامیلتونی، یک دور در گراف است که از تمام رئوس دقیقا یکبار عبورمی‌کند و به نقطه شروع برمی‌گردد. در گراف‌های وزن‌دار می‌توان از دور هامیلتونی برای یافتن کوتاهترین مسیر بسته گراف استفاده‌نمود. مسئله یافتن دور هامیلتونی در یک گراف علاوه بر اینکه جنبه نظری دارد، به صورت کاربردی برای مسائل مختلف از جمله زمانبندی کارها، جمع‌آوری زباله شهری، لوله‌کشی فاضلاب شهری، مسیریابی وسایل نقیله و غیره استفاده‌می‌شود. هدف از حل این مسأله، یافتن کوتاهترین مسیر از مجموعه‌ای از شهرهاست به طوری که هر شهر دقیقا یکبار ملاقات‌شود و در نهایت، به شهر آغازین برگردد. در واقع مسئله فروشنده دوره‌گرد یک دور هامیلتونی است [Letchford and Nasiri, 2015].

مسئله فروشنده دوره‌گرد و یافتن کوتاهترین دور هامیلتونی یکی از مسائل مهم و پیچیده در دنیای امروز می‌باشد که می‌تواند کمک شایانی در عرصه حمل و نقل و جابه‌جائی افراد و وسایل نقیله عمومی و خصوصی داشته باشد. به گونه‌ای که عدم مدیریت صحیح آن می‌تواند خسارات زیان‌باری در مصرف سوخت، هزینه و زمان انسان داشته باشد. در همین راستا یافتن کوتاهترین مسیر برای حرکت از یک نقطه و طی کردن مسیرهای مختلف و بازگشت به نقطه آغازین اهمیت بسزایی دارد. به طور نمونه در یک شرکتی که وظیفه پخش مواد لبنیاتی در سطح شهر میان عمده فروشان و خرده فروشان، هایپرما ها و غیره دارد اگر مسیر مناسبی برای پخش مواد لبنیاتی نداشته باشد هر روز هزینه های زیادی بابت نیروی انسانی، سود حاصل، استهلاک وسیله نقلیه و زمان پخش نامناسب متحمل خواهد شد. هدررفت هزینه هر روز در فرآیند پخش تکرار خواهد شد و سالیانه شرکت هزینه‌های هدررفت زیادی خواهد داشت. به عنوان نمونه دیگر یک شرکتی که در زمینه توریست و گردشگری فعالیت می‌کند در بخش گردشگری مراکز فرهنگی استان های مختلف کشورمان ایران اگر مسیر نامناسب برای طی کردن مراکز فرهنگی انتخاب نمایند روزانه هزینه های هنگفتی را به هدر خواهند داد. با این اوصاف پرداختن و یافتن کوتاهترین مسیر هامیلتونی برای بسیاری از مسائل کاربردی و ضروری می‌باشد.

در دهه اخیر از الگوریتم‌های ابتکاری و فراابتکاری مختلفی برای حل مسائل پیچیده مانند فروشنده دوره‌گرد، استفاده‌شده‌است. دلیل این امر این است که در روش‌های دقیق مرتبه زمانی حل مسئله نمائی می‌باشد، در حالی که با روش‌های فراابتکاری این مرتبه زمانی به چند جمله‌ای کاهش‌می‌یابد. به همین دلیل از الگوریتم فرابتکاری بهینه‌سازی ازدحام ذرات و روش‌های توسعه‌یافته آن برای حل مسئله فروشنده دوره‌گرد و یافتن کوتاهترین دور هامیلتونی استفاده شده است.

در این مقاله، به منظور یافتن کوتاهترین مسیر هامیلتونی از الگوریتم بهینه‌سازی ازدحام ذرات (PSO)و انواع بهبودیافته آن، استفاده‌شده‌است. الگوریتم بهینه‌سازی ازدحام ذرات در سال ۱۹۹۵ توسط کندی و ابرهارت ابداع‌شد [Kennedy and Eberhart, 1995] . این الگوریتم با الگوگرفتن از رفتار اجتماعی و زندگی گروهی جاندارانی مانند ماهی‌ها و پرندگان معرفی‌شده‌است. به طور مثال حرکت دسته جمعی ماهی‌ها که به دنبال غذا هستند با ارسال سیگنال بین یکدیگر مانع برخورد بین آنها می‌شود. در الگوریتم بهینه‌سازی ازدحام ذرات به هر یک از این ماهی‌ها یک ذره اطلاق‌می‌شود. این حرکت دسته جمعی از تجربه قبلی هر ذره و تجربه سایر ماهی‌ها برای یافتن غذا بهره‌می‌گیرد

[Heidarabadizadeh and Ghanizadeh, 2020]. در این مقاله، به منظور پیاده‌سازی این مفهوم یک فضا دو بعدی درنظرگرفته‌شده‌است که در آن ذرات به عنوان جمعیت اولیه پراکنده‌شده‌اند. طبیعتاً در این الگوریتم هر ذره در فضا با سرعتی در حال حرکت می‌باشد. در هر لحظه این حرکت منجر به تغییر موقعیت فعلی هر ذره در فضا می‌شود. برای نشان دادن سرعت و موقعیت بعدی هر ذره از رابطه (۱) استفاده‌می‌شود:

$$V_i^{t+1} = w.V_i^t + c_1.r_1^t(P_{best} - X_i^t) + c_2.r_2^t(G_{best} - X_i^t$$

$$X_i^{t+1} = X_i^t + V_i^{t+1} \qquad (۱)$$

که در آن w ضریب اینرسی است که نشان‌دهنده میزان تمایل به ماندن در وضعیت فعلی است، c_1,c_2 نیز معمولاً مجموع آنها ٤ در نظر گرفته می‌شود و r_1^t,r_2^t نیز اعداد تصادفی بین صفر تا یک می‌باشند. همچنین P_{best} نشان‌دهنده بهترین موقعیت محلی هر ذره است و G_{best} بهترین موقعیت سراسری را نشان‌می‌دهد. در بروزرسانی سرعت هر ذره ضرایب مربوط به بهترین موقعیت فعلی هر ذره و بهترین موقعیت سراسری ذرات تعادل بین مفاهیم اکتشاف و استخراج در جستجو را تنظیم‌می‌کنند. مفهوم اکتشاف با جستجوی سراسری به دنبال یافتن بهینه سراسری است در حالی که مفهوم استخراج با جستجوی محلی به دنبال یافتن بهینه محلی است [Auh and Menguc, 2005]. هر قدر ضرایب P_{best} و G_{best} بهتر تنظیم‌شوند تعادل بین مفاهیم استخراج و اکتشاف بیشتر برقرارخواهدبود و نتیجه بهتری حاصل‌می‌شود.

ساختار اصلی مقاله در چندین بخش تنظیم‌شده‌است. نخست، اشاره‌ای به کارهای مرتبط در بخش دوم آورده‌می‌شود. سپس در بخش سوم مدل ریاضی مسئله اصلی مقاله به‌نام مسئله فروشنده دوره‌گرد بیان‌شده‌است. در بخش چهارم نیز چهار رویکرد پیشنهادی ارائه‌می‌شود. بخش پنجم نیز شامل نتایج الگوریتم‌های مختلف مورد بررسی است. در پایان نتیجه‌گیری و کارهای آینده مطرح‌می‌شود.

۲. کارهای مرتبط

تاکنون الگوریتم‌های فراابتکاری مبتنی بر جمعیت متعددی مانند الگوریتم کلونی مورچگان، الگوریتم زنبور عسل، الگوریتم ژنتیک، الگوریتم بهینه‌سازی جغرافیای زیستی، الگوریتم جستجوی گرانشی، کرم شب تاب و غیره ارائه‌شده‌است[Dowlatshahi, Derhami and Nezamabadi-pour, 2019; Amouei, Mirzaie, 2019; Salehian, Tavakkoli-Moghaddam and Norouzi, 2019].

نوع دیگری از الگوریتم‌های فراابتکاری مانند الگوریتم شبیه‌سازی تبرید، الگوریتم تپه‌نوردی، الگوریتم جستجوی ممنوعه، مبتنی بر یک جواب تنها می‌باشند که بیشتر برای جستجوی محلی مورد استفاده قرارمی‌گیرند[Mafarja and Mirjalili, 2017; Corman, Andrea, Dario and Marco, 2010]. در ادامه به استفاده از الگوریتم‌های فراابتکاری متعددی برای حل مسئله فروشنده دوره‌گرد اشاره‌شده‌است.

در [Ouaarab, Ahiod and Yang, 2014] یک الگوریتم جستجوی فاخته با کلید تصادفی برای حل مسائل مشهور از فروشنده دوره‌گرد ارائه‌شده‌است. به منظور تبدیل فضای پیوسته به فضای ترکیبی از طرح رمزگذاری همرا با کلید تصادفی ساده، استفاده‌شده‌است. همچنین به منظور جابه‌جائی بین راه حل‌ها در هر دو فضا از پرواز لوی استفاده‌می‌شود. به منظور مقایسه نتایج از مسائل استاندارد TSPLib استفاده‌شده که نشان از برتری این الگوریتم نسبت به دیگر الگوریتم های فراابتکاری است. همچنین، پرواز لوی که از نام ریاضیدان فرانسوی بنام پویل لوی برگرفته‌شده‌است، یک کلاس از گام تصادفی است که برای اندازگیری‌های طول گام آماری مورد استفاده قرارمی‌گیرد [Brown, Liebovitch and Glendon, 2007].

در [Saji and Riffi, 2016] از نسخه توسعه یافته الگوریتم خفاشی که در سال ۲۰۱۰ توسط یانگ ارائه‌شد، برای حل مسئله

فروشنده دوره گرد استفاده‌شده‌است [Yang, 2010]. بدین منظور، برخی از عملگرهای الگوریتم خفاش پیوسته مانند موقعیت و سرعت متناسب با حل مسائل بهینه‌سازی گسسته تغییریافته است. علاوه بر مقایسه عملکرد روش پیشنهادی با الگوریتم‌های فراابتکاری دیگر، هدف اصلی از این تحقیق بررسی نسخه گسسته الگوریتم خفاش برای حل مسئله فروشنده دوره‌گرد بوده است.

در [Ezugwu and Adewumi, 2017] با اعمال تغییراتی در الگوریتم جستجوی ارگانیسم همزیستی که در سال ۲۰۱٤ توسط چنگ و پرایگو ارائه شده است [Cheng and Prayogo, 2014]، به یافتن جواب نزدیک به بهینه برای حل مسئله فروشنده دوره‌گرد پرداخته‌شده‌است. در آن از سه عملگر جابه‌جائی، درجی و معکوس برای ساخت نسل‌ها استفاده‌شده‌است. این عملیات منجر به بهبود جستجوی محلی در مرحله جهش هر راه حل می‌گردد. به منظور بررسی کارائی الگوریتم پیشنهادی از نمونه‌های کتابخانه TSPLib استفاده‌شده‌است. نتایج حاکی از بهبود الگوریتم پیشنهادی نسبت به الگوریتم نسخه پایه می‌باشد.

در [Jaradat, 2018] از یک ساختار حافظه خارجی برای الگوریتم مورچگان نخبگی ترکیبی برای حل مسئله فروشنده دوره-گرد استفاده‌شده‌است. این حافظه معروف به استخر نخبگی است که شامل راه‌حل‌های متنوع و کیفیت بالا برای حفظ تعادل بین تنوع و کیفیت جستجو می‌باشد. به منظور تشدید و تمرکز در حول یک راه حل نخبه، جستجوی محلی تکراری با پیچیدگی زمانی n با سیستم مورچگان نخبگی ترکیب‌شده‌است. روش پیشنهادی باعث بالابردن کارائی و ثبات در تنوع جستجو و در حین حال کیفیت راه حل بالا در راه‌حل‌ها شده‌است.

در [Taillard and Helsgaun, 2019] یک روش فراابتکاری برای حل مسئله فروشنده دوره‌گرد با پیچیدگی کمتر از درجه دوم پیشنهادشده‌است. در مسئله فروشنده دوره‌گرد با نمونه‌های با اندازه شهر خیلی بزرگ فقط زیرمجموعه‌ای از لبه‌ها به شهرها متصل‌می‌باشد. بدین منظور در هر مرحله یک لیستی از لبه‌های کاندید خوب تهیه‌می‌شود. با این کار راه‌حل‌های با کیفیت بالا کاملاً به طور موثر یافت‌می‌شوند. این روش بر روی نمونه‌های بالای چندین میلیون شهر مسئله فروشنده دوره‌گرد تست‌شده‌است، نتایج عددی نشان از کاهش پیچیدگی زمانی به کمتر از درجه دوم می‌باشد.

در [Eremeev and Kovalenko 2020] یک الگوریتم جدید بازترکیبی بهینه برای حل مسئله فروشنده دوره‌گرد نامتقارن ارائه‌شده‌است. عملگرهای جدید تقاطع و جهش به ترتیب به کمک مسئله بازترکیب بهینه و پرش تصادفی به کمک 3-opt,4-opt ایجادمی‌شود. برای ارزیابی نتایج از نمونه‌های TSPLib استفاده‌شده‌است که در مقایسه الگوریتم پیشنهادی با دیگر الگوریتم-های مناسب دیگر بازدهی بهتر را نشان‌می‌دهد.

در [Adamo, Ghiani and Guerriero, 2020] یک روش کران پایین توسعه‌یافته برای مسئله فروشنده دوره‌گرد وابسته به زمان ارائه‌شده‌است. مسئله فروشنده دوره‌گرد وابسته به زمان نوع تعمیم-یافته مسئله فروشنده دوره‌گرد می‌باشد که وابسته به موقعیت گره‌ها در مسیر هامیلتونی است. یک تکنیک کران پایین قوی‌تر نسبت به دو مقاله مورد ارزیابی قرار گرفته ۲۰۱٤ و ۲۰۱۸ پیشنهادشده‌است. به طور ویژه، یک خانواده جدید از کران پایین پارامتری، ارائه‌شده‌است. نتایج محاسباتی نشان‌می‌دهد که استفاده از مکانیزم جدید در یک الگوریتم شاخه و حد ۷۲ نمونه بیشتر نسبت به روش-های قبلی را حل‌می‌کند و شکاف نیز کاهش‌یافته‌است.

در [Akhand et al. 2020] یک نوع موثری از بهینه‌سازی میمون عنکبوتی به‌نام بهینه‌سازی میمون عنکبوتی گسسته (DSMO) ارائه شده است. در DSMO هر میمون یک راه‌حل از فروشنده دوره‌گرد را نشان‌می‌دهد. در هر تکرار میمون عنکبوتی سعی بر بروزرسانی راه‌حل‌های رهبر محلی و سراسری را دارد. نتایج روی مسائل بزرگ فروشنده دوره‌گرد مورد آزمایش قرارگرفته‌است و با روش‌های خوب مقالات دیگر مانند روش بهینه‌سازی ازدحام ذرات سرعت

تجربی (VTPSO) مقایسه‌شده‌است که حاکی از میانگین هزینه مسیر مناسب‌تر روش پیشنهادی نسبت به دیگر روش‌ها مانند کلونی مورچگان و بهینه‌سازی ازدحام ذرات سرعت تجربی می‌باشد.

به منظور درک مناسب‌تر مطالب مطرح شده در مرور ادبیات در جدول ۱ خلاصه‌ای از مقالات لیست‌شده‌است.

جدول ۱. لیستی از کارهای مرتبط معرفی شده

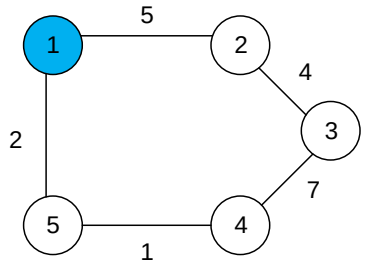
نویسندگان مقاله	الگوریتم پیشنهادی برای حل TSP	توضیحات مختصر روش
اویاراب و همکاران، ۲۰۱۴	جستجوی فاخته با کلید تصادفی	رمزگذاری همرابا کلید تصادفی ساده- جابه‌جائی بین راه‌حل‌ها با پروازلوی
ساجی و ریفی، ۲۰۱۶	الگوریتم خفاش توسعه یافته	تغییر عملگرهای موقعیت و سرعت الگوریتم خفاش برای مسائل بهینه‌سازی گسسته
ایزگوو و آدویومی، ۲۰۱۷	جستجوی ارگانیسم همزیستی بهبودیافته	سه عملگر جابه‌جائی، درجی و معکوس برای ساخت نسل‌ها در مرحله جهش
جرادات، ۲۰۱۸	الگوریتم مورچگان نخبگی ترکیبی	استفاده از استخر نخبگی برای حفظ تعادل بین تنوع و کیفیت جستجو
تائیلارد و هلسگون، ۲۰۱۹	الگوریتم با پیچیدگی کمتر از درجه دوم	لیستی از لبه‌های کاندید خوب برای یافت راه‌حل با کیفیت بالا استفاده می‌شود
اریومف و کووالنکو، ۲۰۲۰	الگوریتم جدید بازترکیبی بهینه	عملگرهای جدید تقاطع و جهش با مسئله بازترکیب بهینه و پرش تصادفی
آدامو و همکاران، ۲۰۲۰	کران‌پایین‌توسعه‌یافته وابسته‌به‌زمان	نوع تعمیم‌یافته TSP که وابسته به موقعیت گره‌ها در مسیر هامیلتونی است
آخند و همکاران، ۲۰۲۰	بهینه‌سازی میمون عنکبوتی‌گسسته	بروزرسانی راه‌حل‌ها با میمون عنکبوتی – مقایسه کارائی الگوریتم با VTPSO

مقاله [Akhand et al. 2020] نزدیکترین تحقیقی است که به الگوریتم‌های پیشنهادی این پژوهش می‌باشد. به گونه‌ای که در آن برای مقایسه الگوریتم پیشنهادی از یک الگوریتم مناسب بهینه سازی ازدحام ذرات سرعت تجربی برای بهبود الگوریتم بهینه‌سازی ازدحام ذرات پایه استفاده‌شده‌است. در الگوریتم VTPSO در عملگر سرعت تغییراتی ایجادشده‌است. در الگوریتم‌های پیشنهادی این پژوهش نیز در مرحله تعیین سرعت و موقعیت بعدی ذره در انتخاب پارامتر بهترین موقعیت فعلی ذره به‌جای اینکه بهترین موقعیت فعلی ذره در انتخاب راه حل بعدی ذره تاثیرگذار باشد، از شبکه پیچیده استفاده‌می‌شود به گونه‌ای که برای انتخاب بهترین موقعیت ذره آن گره‌ای که درجه بیشتری دارد راه‌حلی که در آن گره ذخیره‌شده‌است به عنوان بهترین موقعیت فعلی ذره در نظرگرفته‌می‌شود. در الگوریتم پیشنهادی مطرح شده دیگر به جای سنجه درجه، از سنجه درجه همسایگی استفاده‌شده‌است. تفاوت اصلی که الگوریتم VTPSO با الگوریتم‌های پیشنهادی دارد این است که در VTPSO با اینکه بهبودی در الگوریتم پایه PSO ایجادشده‌است ولی ارتباط بین ذرات در آن به صورت ضعیف دیده‌می‌شود فقط درحد اینکه هر ذره موقعیت فقط قبلی خود را ذخیره‌می‌کند و فقط با موقعیت قبلی خود ارتباط دارد، در صورتی که در الگوریتم بهینه‌سازی ازدحام ذرات شبکه‌ای پیشنهادی، یک شبکه‌ای از ارتباط بین ذرات وجود دارد که این امر باعث یافتن کاراتر بهینه سراسری می‌شود. در واقع، نوآوری الگوریتم‌های پیشنهادی در تشکیل شبکه‌ای از راه‌حل ها می‌باشد که در انتخاب موقعیت بعدی هر ذره تاثیر گذار است، به جای اینکه ذره فقط موقعیت بعدی را با بهترین موقعیت فعلی خودش (که از

مقایسه با یک مرحله قبل ذره بدست می‌آید) محاسبه کند. شبکه پیچیده‌ای که براساس مناسب‌ترین هزینه تشکیل و بروزرسانی‌می‌شود آن گره‌ای که سنجه درجه یا درجه همسایگی بالاتری دارد برای انتخاب موقعیت بعدی ذره، نقش ایفامی‌کند. با این کار بهبود در انتخاب موقعیت بعدی ذره حاصل‌می‌شود.

۳. مدل ریاضی مسئله

در شکل ۱ یک مسئله فروشنده دوره‌گرد با ۵ شهر همراه با فاصله میان برخی شهرها نشان‌داده‌شده‌است که در آن نقطه آغازین شهر۱-می‌باشد که باید از شهرهای دیگر دقیقاً یکبار عبور کند و به شهر ۱ بازگردد. مسیر 1-5-4-3-2-1 یک دورهامیلتونی برای این مسئله با ۵ شهر می‌باشد.



شکل ۱. دورهامیلتونی از مسئله فروشنده دوره‌گرد با ۵ شهر

مسئله فروشنده دوره‌گرد یا همان مدار هامیلتونی را می‌توان به صورت یک گراف کامل بدون جهت G=(V,E) در نظرگرفت که در آن V={0,1,…,n} نمایانگر مجموعه گره‌ها و $E = \{(i,j): i,j \in V, i \neq j\}$ مجموعه یال‌هاست. در صورتی که گراف کامل نباشد به جای هر یالی که در گراف وجود ندارد با یالی معادل مثبت بی نهایت جایگزین‌می‌شود. اگر در جواب پایانی مسئله یک یال مستقیمی بین گره i به گره j وجود داشته باشد $x_{ij} = 1$ و در غیر این صورت $x_{ij} = 0$ در نظرگرفته‌می‌شود بدین معنی که یال مستقیمی بین دو گره iوj وجود ندارد. اگر $C = [c_{ij}]$ ماتریس متقارن هزینه روی گراف G باشد آنگاه مدل ریاضی مسئله فروشنده-دوره‌گرد به صورت رابطه (۲) است:

$$min \sum_{i=0}^{n} \sum_{j=i}^{n} c_{ij} x_{ij} \qquad (۲)$$

$$\sum_{i=0}^{n} x_{ij} = 1 \qquad \forall\ j = 0,1,\dots,n; \qquad (۳)$$

$$\sum_{j=0}^{n} x_{ij} = 1 \qquad \forall\ i = 0,1,\dots,n;\ ; \qquad (٤)$$

$$x_{ij}\,(n-1) + y_i - y_j \leq n-2,\ \ \forall\ i = 0,1,\dots,n,$$
$$\forall\ j = 0,1,\dots,n, i \neq j; \qquad (٥)$$

$$x(ij) \in \{0,1\} \qquad \forall i,j = 0,1,\dots,n \qquad (٦)$$

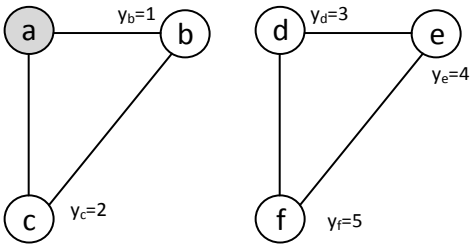
$$y_i \geq 1 \qquad \forall\ i = 1,2,\dots,n; \qquad (٧)$$

در این مدل، رابطه (۳) و (٤) به ترتیب محدویت ورود یک یال به هر گره مانند i و خروج یک یال از هر گره مانند گره j را نشان‌می-دهد. رابطه (٥) زیردورهای یافته‌شده که فاقد گره آغازین باشد را به عنوان جواب قابل قبول نمی پذیرد. که در آن y_i نشان‌دهنده مقدار موقعیت گره i در مسیر دور هامیلتونی می‌باشد که برابر با تعداد یال‌های پیموده شده از گره آغازین تا گره i می‌باشد. به طور مثال برای حالتی همانند شکل ۲ که یک گراف دوبخشی ایجادشده‌باشد با اینکه دارای دور بدون گره آغازین است در رابطه های (۳) و (٤) صدق‌می‌کند. مطابق محدویت رابطه (٥) گره d دو ورودی و خروجی دارد که از آن رابطه (۸) بدست‌می‌آید:

$$y_d \leq y_e - 1,\ y_d \geq y_f - 1; \qquad (۸)$$

چون $y_e = ٤$ و $y_f = ٥$ بنابراین رابطه (۸) دو نتیجه متناقض، ارائه‌می‌دهد به همین دلیل در رابطه (٥) صدق‌نمی‌کند و به عنوان دورهامیلتونی پذیرفته‌نمی‌شود.

محدودیت رابطه (٦) نیز به شرایط دودوئی بودن x_{ij} اشاره‌می‌کند که فقط x_{ij} مقدار صفر یا یک را می‌پذیرد. رابطه (۷) نیز موقعیت گره i در دور هامیلتونی را نشان‌می‌دهد. این مقدار برابر با تعداد یال‌های پیموده‌شده از گره آغازین تا گره i است که مقدار آن برای اولین گره پیموده‌شده پس از گره آغازین یک می‌باشد.



شکل ۲. گراف دو بخشی با شش رأس

٤. رویکرد پیشنهادی

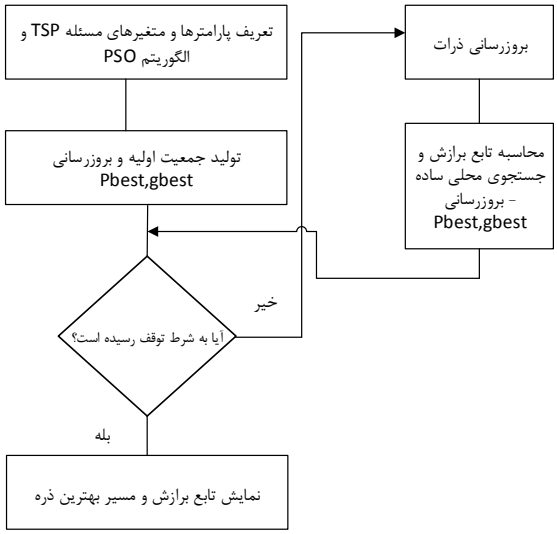
تاکنون الگوریتم‌های فراابتکاری متعددی توسط پژوهشگران علوم کامپیوتر، ابداع‌شده‌است. برخی از الگوریتم‌های فراابتکاری مبتنی بر جمعیت مانند ژنتیک و بهینه‌سازی ازدحام ذرات ارتباط ضعیفی بین کروموزوم‌ها و ذرات خود دارند، ولی در دنیای واقعی این ارتباط بین مولفه‌ها خیلی قوی‌تر و بیشتر است. به طور مثال در الگوریتم بهینه‌سازی ذراتی که در سال ۱۹۹۵ ابداع‌گردید فقط ارتباط موقعیت قبلی ذره ذخیره و نگهداری‌می‌شود، در حالی که در دنیای واقعی شبکه‌ای از ارتباط بین ذرات وجود دارد. بدین منظور در این مقاله به منظور تقویت این ارتباط بین ذرات و نزدیک‌شدن پیاده‌سازی به دنیای واقعی از شبکه پیچیده، استفاده‌شده‌است. در همین راستا در ادامه از میان الگوریتم‌های فراابتکاری متعدد به دلیل اینکه الگوریتم بهینه‌سازی ازدحام ذرات قابلیت شبکه‌کردن ذرات فضای‌جستجو وجود دارد، به بهبود الگوریتم بهینه‌سازی ازدحام ذرات پرداخته‌شده‌است. در این رویکرد پیشنهادی، الگوریتم‌های بهینه-سازی ازدحام ذرات مختلفی ارائه‌می‌شود. این الگوریتم‌ها براساس شبکه پیچیده یا بدون شبکه پیچیده برای یافتن کوتاهترین دور هامیلتونی ارائه‌شده‌اند. در ابتدا از الگوریتم بهینه‌سازی ازدحام ذرات استاندارد با کمی تغییر در نتیجه هر ذره، استفاده‌شده‌است. پس از آن روش پیشنهادی الگوریتم PSO با جستجوی محلی، بکاربرده‌می‌شود. سپس از دو روش شبکه پیچیده‌ای برای حل مسئله فروشنده دوره‌گرد استفاده‌شده‌است که در ادامه مطالب، هر روش به طور مجزا، بررسی‌می‌شود.

٤-۱ روش یکم: الگوریتم بهینه‌سازی ازدحام ذرات پایه به کمک جهش ساده

در الگوریتم PSO ذراتی که در فضا پراکنده هستند با یکدیگر ارتباط و همکاری دارند. این ارتباط بین ذرات در دو بخش از الگوریتم صورت‌می‌گیرد. یکی بهترین برازش شخصی هر ذره و دیگری بهترین برازش سراسری. بهترین برازش شخصی هر ذره بهترین برازش هر ذره در تکراری‌های قبلی را نشان‌می‌دهد. تمام ذراتی که در شروع الگوریتم به عنوان جمعیت اولیه تولیدمی‌شوند P_{best} آنها با بینهایت مقداردهی‌می‌شود. سپس برازش هر ذره محاسبه‌می‌شود و به عنوان بهترین برازش شخصی خود ذخیره‌می‌شود. به همین ترتیب در بروزرسانی موقعیت هر ذره این مقدار با برازش همان ذره در موقعیت جدید مقایسه‌می‌شود در صورت بهتربودن این برازش نسبت به موقعیت قبلی ذره، به عنوان بهترین برازش شخصی همان ذره ذخیره‌می‌شود. در غیر این صورت تغییری در بهترین برازش شخصی ذره رخ‌نمی‌دهد. اگر در بهترین برازش شخصی تغییری وجود داشته باشد، برازش این راه حل با بهترین برازش سراسری مورد مقایسه قرارمی‌گیرد. در صورت بهتر بودن راه حل بهترین برازش سراسری بروز رسانی‌می‌شود. فلوچارت این الگوریتم در شکل۳، نشان‌داده‌شده‌است.

همان‌طورکه از فلوچارت الگوریتم بهینه‌سازی ازدحام ذرات در شکل۲، مشخص‌است، ابتدا پارامترها و متغیرهای مسئله فروشنده دوره‌گرد تعریف‌می‌شود. سپس به تعداد ذرات که نشان‌دهنده جمعیت اولیه است، به طور تصادفی ذرات تولیدمی‌شود و موقعیت آنها مشخص‌می‌شود و در فضای الگوریتم بهینه‌سازی ازدحام ذرات قرارمی‌گیرند. در این حال سرعت هر ذره با صفر مقداردهی‌می‌شود. طبق سرعت هر ذره و موقعیتش براساس تابع هزینه، مسیر حل مسئله فروشنده دوره‌گرد به همراه تابع برازش آن ذره بدست‌می‌آید. حال

باید بهترین برازش شخصی هر ذره و بهترین برازش سراسری ذرات بروزرسانی‌شود. برای هر ذره نیز یک جهش ساده در راه‌حل‌ها به کمک یکی از جهش‌های جابه‌جائی، درجی و معکوس صورت‌می-گیرد. سپس برازش راه حل محاسبه‌می‌شود، در صورت بهبود، جایگزین بهترین برازش شخصی ذره‌می‌شود. در جدول ۲ پارامترهای بکاررفته به همراه خلاصه‌ای از آن پارامترها، آورده‌شده‌است. پارامتر MaxIt نشان‌دهنده تعداد تکرار مراحل الگوریتم است که متناسب با نوع الگوریتم می‌توان آن را تنظیم‌کرد. با بررسی چندین مرتبه تکرار مشخص‌گردید که تعداد تکرار مراحل الگوریتم برابر ۵۰۰۰ باشد، نتایج الگوریتم از نظر هزینه‌ای به ثبات و پایداری می‌رسد. nPop نیز جمعیت اولیه الگوریتم را نشان‌می‌دهد که به طور تجربی برابر ۱۵۰ مقداردهی‌شده‌است. کمتر از این مقدار به طور مثال نتایج مناسبی با تعداد تکرار ۵۰۰۰ ارائه‌نمی‌دهد و بیشتر از این مقدار به طور مثال nPop=500 زمان اجرای الگوریتم تقریباً با هزینه برابر با زمانی که nPop=150 است، خیلی بیتشر می‌باشد. بدین منظور npop با ۱۵۰ مقداردهی شده است. وزن اینرسی یا w میزان تاثیر سرعت فعلی برای بدست آوردن سرعت بعدی ذره را نشان‌می‌دهد که در ابتدای الگوریتم با یک مقداردهی‌شده‌است، ولی چون w_{damp} با ۰,۹۹ مقداردهی‌شده‌است در هر بار بروزرسانی مقدار w جدید در ۰,۹۹ مقدار قبلی خود ضرب‌می‌شود. در واقع با این کار میزان تاثیر سرعت فعلی ذره در هر تکرار برروی سرعت بعدی ذره، کاهش‌می‌یابد. ضریب اینرسی می‌تواند ثابت یا متغیر باشد [Umapathy, Venkataseshaiah and Arumugam, 2010]. در این مقاله به کمک پارامتر Wdamp در هر تکرار این ضریب کاهش‌می‌یابد. معمولاً مقدار wdamp نیز به طور تجربی بدست‌می‌آید. ضرایب c1,c2 به ترتیب میزان یادگیری شخصی و عمومی ذره را نشان‌می‌دهد که این مقادیر به طور تجربی تنظیم‌می-گردند. تعداد شهرها یا nVar هم متناسب است با تعداد گره‌های مسئله‌ای که حل‌می‌شود که بستگی به مسئله ورودی دارد. به عنوان مثال برای مسئله ST70 مقدار nVar برابر ۷۰ مقداردهی‌می‌شود.



شکل ۳. فلوچارت الگوریتم بهینه‌سازی ازدحام ذرات پایه به کمک جهش‌ساده

جدول ۲. پارامترهای مسئله TSP و الگوریتم PSO

نام پارامتر	مقدار	توضیحات
MaxtIt	۵۰۰۰	تعداد تکرار الگوریتم PSO
nPOP	۱۵۰	تعداد ذرات فضای الگوریتم PSO
W	۱	وزن اینرسی
Wdamp	۰,۹۹	ضریب کاهش وزن اینرسی
C1	یک عدد تصادفی بین صفر تا یک ضربدر ۳	ضریب یادگیری شخصی
C2	4-c1	ضریب یادگیری عمومی
nVar	متناسب با اندازه مسئله درنظرگرفته‌می‌شود	تعداد شهرها

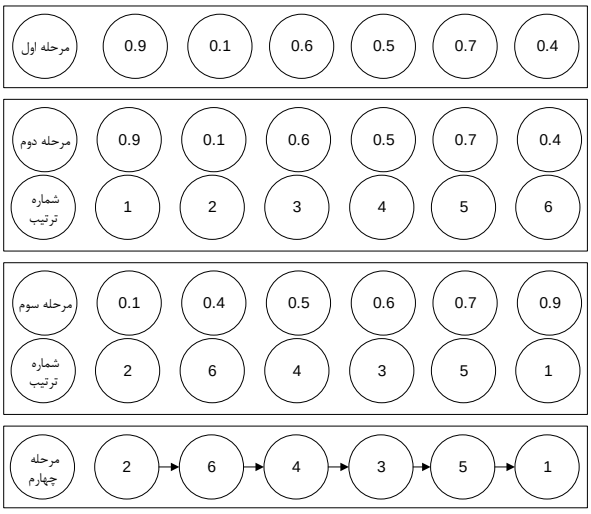
٤-٢ روش محاسبه موقعیت ذره در فضای بهینه‌سازی ازدحام ذرات

به دلیل اینکه مسئله فروشنده دوره‌گرد جزو مسائل گسسته است و از طرفی الگوریتم بهینه‌سازی ازدحام ذرات بیشتر برای محیط‌های پیوسته، استفاده‌می‌شود باید الگوریتم بهینه‌سازی ازدحام ذرات پیوسته به گسسته تبدیل‌شود [Strasser, Sheppard and Butcher, 2016]. بدین منظور به تعداد شهرها عدد تصادفی بین صفر تا یک تولیدمی‌شود. حال به هر کدام از اعداد تصادفی تولید شده به ترتیب از یک تا آخرین عدد تصادفی، شماره تخصیص‌داده‌می‌شود. سپس این اعداد تصادفی به همراه شماره ترتیب تولیدشان براساس اعداد تصادفی به صورت صعودی مرتب‌می‌شوند. در این مرتب‌سازی، جابه‌جائی عدد تصادفی به همراه شماره ترتیب‌شان می‌باشد. به منظور روشن‌شدن این روش نحوه تولید مسیری با شش شهر، با مثالی بیان می‌شود. همان طور که در شکل ٤، نشان‌داده‌شده‌است به تعداد شهرها (در این مثال شش شهر) عدد تصادفی بین صفر تا یک تولیدمی‌شود. در مرحله بعد به ترتیب اعداد یک تا شش به اعداد تصادفی داده‌می‌شود. حال براساس اعداد تصادفی لیست مرتب‌می‌شود. به طور مثال عدد تصادفی ۰,٦ شماره شهر ۳ را نشان‌می‌دهد و بعد از مرتب‌سازی به عنوان چهارمین شهری است که در این راه حل ملاقات می‌شود. در نهایت مسیر طی‌شدن شهرها مشخص و یک راه‌حل تولیدمی‌شود.

روش محاسبه تابع هزینه بدین صورت است که فاصله بین شهرها در یک ماتریسی n*n نگهداری‌می‌شود و سپس مسیر طی‌شده توسط فروشنده دورگرد به تابع هزینه داده‌می‌شود تا با توجه به مسیر طی‌شده میان شهرها طول مسیر محاسبه‌شود.

به منظور بدست آوردن بهترین تجربه شخصی هر ذره در ابتدا اولین موقعیتی که ذره قرارمی‌گیرد، در آن مسیری به طور تصادفی تولیدمی‌شود که بر اساس این مسیر برازش این راه حل بدست‌می‌آید و به عنوان بهترین تجربه شخصی این ذره در نظرگرفته‌می‌شود. حال پس از بروز رسانی موقعیت این ذره مسیر جدیدی در این ذره قرارمی‌گیرد و طبق آن یک برازش جدید تولیدمی‌شود. حال این برازش با برازش قبلی ذره مقایسه‌می‌شود، اگر بهبود داشت به عنوان بهترین تجربه شخصی این ذره ذخیره‌می‌شود. در غیر این صورت همان برازش قبلی به عنوان بهترین تجربه این ذره تغییرنمی‌کند. همین روند در تکرارهای مختلف الگوریتم انجام‌می‌گیرد.

بهترین برازش سراسری ذرات، بهترین تجربه سراسری تمام ذرات در فضا براساس برازش آنها در هر تکرار است. به منظور بدست آوردن بهترین برازش سراسری ذرات، اگر بهترین برازش شخصی یک ذره تغییر پیداکند با مقدار بهترین برازش سراسری مقایسه‌می‌شود و در صورت بهبود به عنوان بهترین برازش سراسری جایگذاری‌می‌شود.



شکل ٤. تولید یک مسیر از شهرها به کمک موقعیت هر ذره

٤-٣ روش دوم: الگوریتم بهینه‌سازی ازدحام ذرات به همراه جستجوی محلی

به منظور بهبود راه‌حل‌های تولیدشده در هر تکرار از الگوریتم بهینه-سازی ازدحام ذرات می‌توان از جستجوی محلی استفاده‌کرد. همچنین، به منظور بدست‌آوردن موقعیت‌های مختلف یک ذره می‌توان به جای انتخاب تصادفی از حالت تئوری آشوب استفاده

نمود[Liu et al. 2005]. نحوه عملکرد این تئوری در رابطه (۹) آورده‌شده‌است:

$$X_{(j)} = M * X_{(j-1)} * (1 - X_{(j-1)}) \quad \forall j \le nPop \qquad (۹)$$

که در آن M یک عدد صحیح کوچک است. $X_{(j)}$ و $X_{(j-1)}$ به ترتیب موقعیت فعلی و پیشین ذره j را نشان‌می‌دهد. به منظور انتخاب جمعیت اولیه بر خلاف روش الگوریتم بهینه‌سازی ازدحام ذرات استاندارد که به طور تصادفی یکنواخت انتخاب‌می‌شوند، در این روش از تعیین موقعیت آشوبی ذره برای بدست‌آوردن جمعیت اولیه استفاده‌شده‌است. همچنین در ادامه الگوریتم به ازاء سه مورد از ذره‌های فضای الگوریتم بهینه‌سازی ازدحام از تئوری آشوب از رابطه (۱۰) در محاسبه سرعت ذره استفاده‌شده‌است:

$$V_i^{t+1} = w.V_i^t + cr_i.(P_{best} - X_i^t) + (1 - cr_i).(G_{best} - X_i^t) \quad \forall i \le nPop \qquad (۱۰)$$

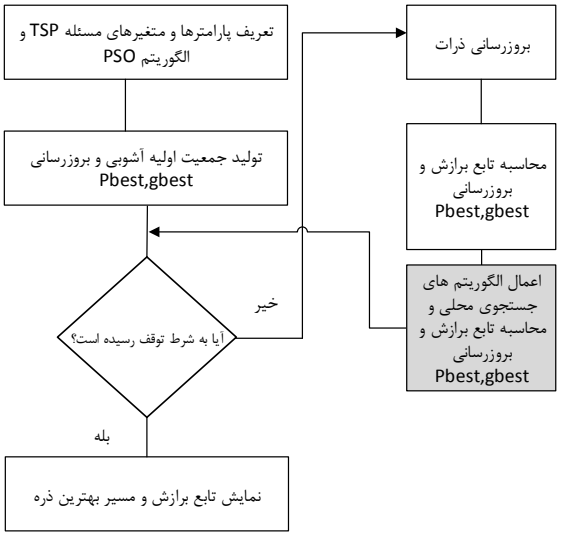
که در آن تاثیر تئوری آشوب برای بهترین موقعیت فعلی ذره روی P_{best} و برای بهترین موقعیت سراسری ذرات روی G_{best} اعمال‌می‌شود. به ازاء $nPop=150$ برای ذرات ۵۰،۱۰۰ و ۱۵۰ برای محاسبه سرعت بعدی از تئوری آشوب استفاده‌شده‌است. برای بقیه ذرات از رابطه (۱۱) برای بروزرسانی سرعت استفاده‌می‌شود:

$$V_i^{t+1} = w.V_i^t + L.c_1.(P_{best} - X_i^t) + G.c_2.(G_{best} - X_i^t) \quad \forall i \le nPop \qquad (۱۱)$$

که در آن مقادیر اولیه L,G با یک مقداردهی‌شده‌است و در هر تکرار مقدار L با ضرب در ۰,۹۹ خودش بروزرسانی‌می‌شود.

فلوچارت الگوریتم بهینه‌سازی ازدحام ذرات به کمک جستجوی محلی برای حل مسئله فروشنده دوره‌گرد در شکل ۵، آورده‌شده‌است. پس از تولید جمعیت اولیه آشوبی و بروزرسانی موقعیت و ارزیابی راه‌حل نوبت به بکارگیری جستجوی محلی در راه حل بدست‌آمده می‌باشد که در آن از سه نوع مختلف جستجوی محلی مختلف جابه‌جائی، معکوس و درجی روی یک راه‌حل استفاده‌شده‌است. هزینه بدست آمده از سوی هر جستجوی محلی نگهداری‌شده و در نهایت از میان سه راه حل، راه حلی که بهترین برازش را دارد، به عنوان راه حل بروز شده ذخیره‌می‌شود و سپس با راه حل اصلی در این ذره مقایسه‌می‌شود اگر بهبود داشت به عنوان راه حل این ذره جایگزین‌می‌شود. حال این راه حل با بهترین برازش شخصی خودش مقایسه‌می‌شود، اگر بهبود داشته باشد به عنوان بهترین برازش شخصی همان ذره جایگذاری‌می‌شود. در این حالت با بهترین برازش سراسری ذرات نیز مقایسه‌می‌شود، اگر بهبود داشته باشد به عناون بهترین برازش سراسری ذرات جایگذاری‌می‌شود. سپس همین عملیات جستجوی محلی مختلف جابه‌جائی، معکوس و درجی برای بهترین برازش سراسری ذرات نیز انجام‌می‌شود و اگر نتیجه بهتری حاصل‌شد به عنوان بهترین برازش سراسری ذرات جایگذاری‌می‌شود.

٤-٤ روش سوم: الگوریتم بهینه‌سازی ازدحام ذرات شبکه پیچیده‌ای براساس درجه ذرات هر نسل به همراه جستجوی محلی برای حل مسئله فروشنده دوره‌گرد



شکل ۵. فلوچارت الگوریتم بهینه‌سازی ازدحام ذرات به کمک جستجوی محلی برای حل مسئله فروشنده دوره‌گرد

در این روش تغییراتی در روش دوم اعمال‌شده‌است که در آن برای تولید جمعیت اولیه از اعداد تصادفی یکنواخت استفاده‌شده‌است. همچنین، نوآوری اصلی این روش استفاده از شبکه پیچیده در انتخاب راه‌حل‌ها برای یافتن موقعیت بعدی ذرات می‌باشد. در الگوریتم بهینه‌سازی ازدحام ذرات برای یافتن موقعیت بعدی ذرات از سه‌پارامتر سرعت فعلی ذره، بهترین موقعیت ذره که تاکنون تجربه‌کرده‌است و بهترین موقعیت سراسری تمام ذرات موجود در فضای جستجو استفاده‌می‌شود. برخلاف دنیای واقعی، ارتباط میان ذرات چندان در نظرگرفته‌نشده‌است برای اینکه ارتباط میان ذرات در مکانی نگهداری‌شود و هزینه مناسب‌تر و بهینه‌تر بدست‌آید نیاز به تشکیل شبکه پیچیده است تا به کمک آن راه حلی که در انتخاب موقعیت بعدی هر ذره نقش ایفامی‌کند هزینه مناسب‌تری داشته‌باشد. لازم به توضیح است که در PSO پایه برای انتخاب موقعیت بعدی ذره بهترین موقعیت فعلی ذره فقط با مقایسه هزینه فعلی ذره با هزینه قبلی ذره بدست‌می‌آید و این ارتباط ضعیف فقط شامل موقعیت قبلی ذره می‌باشد. ولی در الگوریتم بهینه‌سازی ازدحام ذرات شبکه‌ای سعی بر برقراری ارتباط با ذرات با هزینه بهتر از طریق سنجه‌درجه شبکه و انتخاب آن راه‌حل به عنوان بهترین بهینه موقعیت ذره است و تاثیر خود را در انتخاب موقعیت بعدی ذره نشان‌می‌دهد. در ادامه به نحوه اعمال و پیاده‌سازی مفهوم شبکه پیچیده پرداخته‌می‌شود.

معمولاً از گراف برای مدلسازی پدیده های واقعی مانند گراف شهرها و ارتباطات شهرها استفاده‌می‌شود. در دنیای واقعی گراف‌های متعددی دیده می‌شود که بر اساس ویژگی‌ها یا معیارهایی مختلفی مانند موقعیت، توزیع درجه، متوسط درجه و غیره تشکیل‌می‌شود. به این نوع شبکه‌ها، شبکه پیچیده گفته‌می‌شود. در دو دهه اخیر جنبش جدیدی بر مطالعه شبکه پیچیده مانند شبکه‌هایی که نامنظم، پیچیده و پویا هستند، ایجادشده‌است. تمرکز اصلی شبکه‌های پیچیده با حرکت از آنالیز و تجزیه و تحلیل شبکه‌های کوچک با سیستم‌هایی که دارای هزاران یا میلیون‌ها گره است آغازشده‌است. این فعالیت با دو مقدمه که در سال ۱۹۹۸توسط Watts و Strogoutz در مورد شبکه‌های جهان کوچک و یک سال بعد از آن توسط بارآباسی و آلبرت روی شبکه‌های مقیاس آزاد آغاز شده است [Watts and Strogatz, 1998; Baraba´si and Albert, 1999]. در شبکه‌های پیچیده اجزاء به صورت خودسازمانده عمل‌می‌کنند. می‌توان آن را با روش‌هایی مانند شبکه منظم، تصادفی، جهان کوچک و مقیاس آزاد مدل کرد. آنالیز تئوری شبکه در زمینه‌های مختلف مانند زیست‌شناسی، رفتار سازمانی، شبکه‌های اجتماعی و آنالیز داده‌های هوشمند مورد استفاده‌قرارمی‌گیرد. برای اندازه‌گیری شبکه از سنجه‌های مختلفی مانند متوسط درجه گره‌ها، ضریب خوشگی، تابع توزیع درجه، تشکیل مولفه بزرگ استفاده‌می‌شود [Wu et al. 2017; Mitchell, 2006].

یکی از جداولی که نیاز است تا اطلاعات شبکه پیچیده در آن ذخیره شود جدول nPop نام دارد که اندازه آن به تعداد جمعیت اولیه است، چون اطلاعات ذراتی که در جمعیت بروزمی‌شوند در آن نگهداری‌می‌شود. پارامترهای مورد استفاده در شبکه پیچیده در جدول ۳ به نمایش‌درآمده‌است:

جدول ۳. پارامترهای شبکه پیچیده در جدول nPop

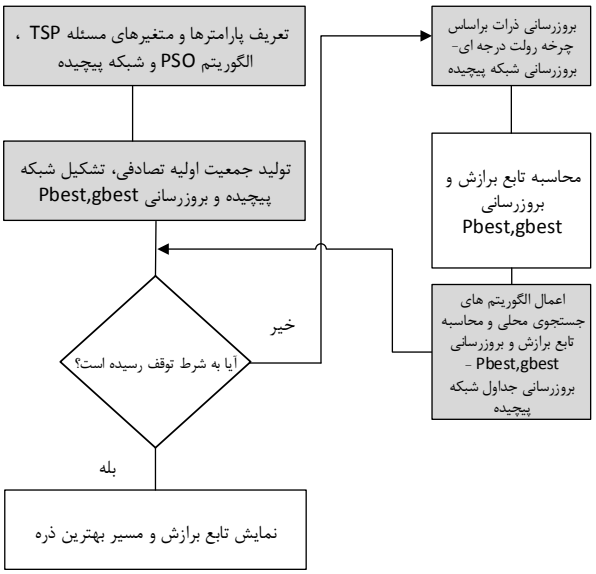
Number Solution	Position	Cost	Degree	Number Node

که در آن NumberSolution شماره راه حل، Position موقعیت ذره در فضای الگوریتم، Cost هزینه راه حل،Degree درجه راه‌حل و NumberNode شماره گره‌هایی که به راه حل NumberNode شماره i متصل‌می‌باشد را نشان‌می‌دهد. یکی دیگر از جداولی که در شبکه پیچیده برای نگهداری اطلاعات شبکه لازم است allPop نام دارد که اندازه آن به تعداد کل راه‌حل‌های تولیدشده در طی اجرای الگوریتم است و پارامترهای آن نیز همانند جدول nPop می‌باشد. به منظور تشکیل شبکه پیچیده در ابتدا باید اطلاعات جدول nPop برای هر ذره تکمیل‌شود. که در آن برای راه

حل جدید یک شماره راه‌حل (NumberSolution) اختصاص‌داده‌می‌شود. این کار با بعلاوه شدن مقدار قبلی شماره راه حل امکان پذیر خواهدشد. همچنین مسیر راه حل و هزینه ذره iام نیز به ترتیب در Position و Cost جدول nPop قرارداده‌می‌شود. در ابتدای کار نیز درجه راه‌حل جدید با یک مقداردهی می‌شود. هر تغییری که در راه حل i ام رخ‌می‌دهد در جدول allPop نیز به همان ترتیب اعمال‌می‌شود. در شکل ٦ فلوچارت الگوریتم بهینه‌سازی ازدحام ذرات شبکه پیچیده‌ای براساس درجات هر نسل به کمک جستجوی محلی برای حل مسئله فروشنده دوره‌گرد آورده‌شده‌است که در آن به منظور انتخاب راه‌حل‌ها از چرخه رولت درجه‌ای، استفاده‌شده‌است. در چرخه رولت درجه‌ای آن ذره‌ای که درجه بزرگ‌تری دارد شانس انتخاب بیشتری را دارد. در واقع هم اکنون ذره i در حال پردازش می‌باشد که با چرخه رولت درجه‌ای ذره دیگری به طور مثال i_1 انتخاب‌می‌شود. حال دو اتفاق در شبکه پیچیده خواهد افتاد: یکی در شبکه پیچیده درجه گره i یک واحد افزایش خواهدداشت چون قرار است گره i_1 به آن متصل‌شود. دوم این که تغییراتی در شبکه پیچیده به ترتیب زیر اعمال‌خواهد شد: با اضافه کردن یک واحدی به شماره راه حل ذره i_1 به عنوان راه حل جدید در جدول nPop ذخیره خواهد شد. همچنین، گره i_1 به لیست گره‌هایی که به ذره i ام متصل است، اضافه‌می‌شود. در ادامه همه تغییرات روی ذره i ام در جدول allPop اضافه‌خواهدشد. تغییراتی که در محاسبه سرعت ذره i ام ایجادخواهدشد به صورت رابطه (١٢) است:

$$V_i^{t+1} = w.V_i^t + L.c_1.r_1(P_{i1} - X_i^t) + G.c_2.r_2(G_{best} - X_i^t) \quad \forall\, i \le nPop \qquad (١٢)$$

که در آن منظور از P_{i1} موقعیت ذره i1 است که از طریق چرخه رولت درجه‌ای انتخاب‌شده بود. مقادیر r_1, r_2 نیز اعداد تصادفی بین صفر تا یک است. همچین مقادیر c_1, c_2 به گونه‌ای تنظیم‌شده‌است که حاصل جمع آنها برابر عدد ٤ باشد[Wu et al., 2017]. در ادامه نیز بر روی هر راه حل عملیات جستجوهای محلی جابه‌جائی، معکوس و درجی اعمال‌می‌شود و در نهایت همین عملیات جستجوی محلی بر روی بهترین برازش سراسری ذرات نیز برای بهبود هزینه اعمال‌می‌شود.

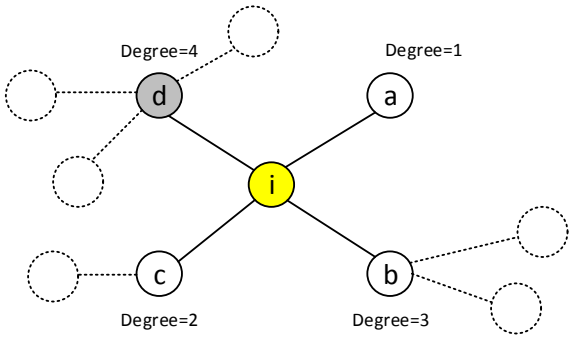


شکل ٦. فلوچارت الگوریتم بهینه‌سازی ازدحام ذرات شبکه پیچیده‌ای بر اساس درجه ذرات هر نسل به کمک جستجوی محلی برای حل مسئله فروشنده دوره‌گرد

٤-٥ روش چهارم: الگوریتم بهینه‌سازی ازدحام ذرات شبکه پیچیده‌ای براساس درجه همسایگی هر ذره به کمک جستجوی محلی برای حل مسئله فروشنده دوره‌گرد

در این روش برای حل مسئله فروشنده دوره‌گرد تغییراتی در روش سوم ایجادشده‌است. بدین صورت که برای انتخاب گره‌ای که می‌خواهد به گره i متصل‌شود یک فرآیند جدیدی درنظرگرفته‌شده‌است. در روش سوم چرخه رولت درجه‌ای از میان جمعیت فعلی آن که درجه بیشتری داشت یک گره انتخاب‌می‌شود. ایراد این روش این است که سنجه درجه برای کل جمعیت درنظرگرفته‌می‌شد. در روش فعلی از میان همسایگان گره i آنکه

درجه بیشتری دارد به عنوان گره اتصال انتخاب‌می‌شود. به منظور نمایش بهتر این روش شکل ۷ را درنظربگیرید:

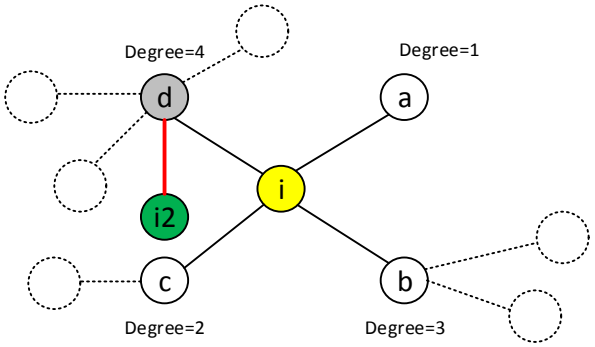


شکل ۷. بخشی از شبکه پیچیده

که در آن همسایگان گره i یعنی a,b,c,d هر کدام دارای درجه مشخصی هستند. آن گره‌ای که درجه بیشتر دارد یعنی گره d به عنوان گره اتصالی در شبکه پیچیده انتخاب‌می‌شود. همچنین گره d جایگزین بهترین موقعیت فعلی در رابطه بهینه‌سازی ازدحام ذرات استاندارد می‌باشد. به همین دلیل تاثیر انتخاب گره با بیشترین درجه در محاسبه موقعیت ذره i در رابطه (۱۳) نشان‌داده‌شده‌است:

$$V_i^{t+1} = w.V_i^t + L.c_1.r_1\left(P_{allPop(i1)} - X_i^t\right) + G.c_2.r_2(G_{best} - X_i^t) \quad \forall\, i \le nPop \qquad (۱۳)$$

که در آن منظور از $P_{allPop(i1)}$ همان موقعیت ذره i_1 در جدول allPop می‌باشد و i_1 نیز شماره گره‌ای است که در میان همسایگان گره i بیشترین درجه را داشته‌است. حال ذره جدیدی بنام i_2 تولیدمی‌شود که براساس تغییر مکان ذره i ایجادمی‌شود. تاثیر این عملیات در وضعیت شبکه این است که ابتدا i_2 به عنوان راه حل جدید تعریف‌می‌شود و درجه گره i_2 یکی اضافه‌می‌شود و سپس شماره گره i_1 به عنوان گره ای که به i_2 متصل است در لیست گره‌های متصل شده به i_2 اضافه‌می‌گردد. بنابراین شکل ۷ به شکل ۸ تغییر پیدامی‌کند که در آن گره جدید i_2 به گره d که قبلاً به واسطه بیشترین درجه از میان هفت همسایه گره i انتخاب‌شده‌بود، متصل‌می‌شود.



شکل ۸ بروزرسانی بخشی از شبکه پیچیده پس از اضافه شدن گره جدید i_2

۵. نتایج

به منظور مقایسه نتایج الگوریتم‌های پیشنهادی از دو گروه مسائل مختلف برای یافتن کوتاهترین دور هامیلتونی استفاده‌شده‌است. گروه اول مسائل استاندارد که در کتابخانه TSPLib می‌باشد. گروه دوم مقایسه نتایج استفاده از مسئله کاربردی یافتن کوتاهترین دور هامیلتونی برای استان‌های مختلف ایران است.

۵-۱ گروه اول: مسائل استاندارد کتابخانه TSPLib

در این مقاله برای روش اول یعنی الگوریتم بهینه‌سازی ازدحام ذرات استاندارد به دلیل مقداری کندی با ۵۰۰۰ تکرار مورد آزمایش قرار گرفته‌شده‌است. برای مقایسه مسائل مختلف موجود در مجموعه TSPLib با چهار روش موجود در این مقاله مورد مقایسه قرار گرفته است. به منظور مقایسه نتایج، هر روش ۳۰ بار متوالی اجراشده‌است. برای ارزیابی میزان خطای روش های پیشنهادی با بهترین نتیجه موجود در TSPLib از معیاری بنام Gap استفاده‌می‌شود. نحوه محاسبه آن در رابطه (۱٤) نشان‌داده‌شده‌است:

$$\text{Gap} = \frac{\text{ProposedSolution} - \text{BKS}}{\text{BKS}} * 100 \qquad (۱٤)$$

که در آن ProposedSolution میزان برازش راه حل پیشنهادی و BKS نشان‌دهنده بهترین راه حل موجود در پایگاه TSPLib می‌باشد. هرچه این مقدار به صفر میل‌کند، نشان از نتیجه بهتر

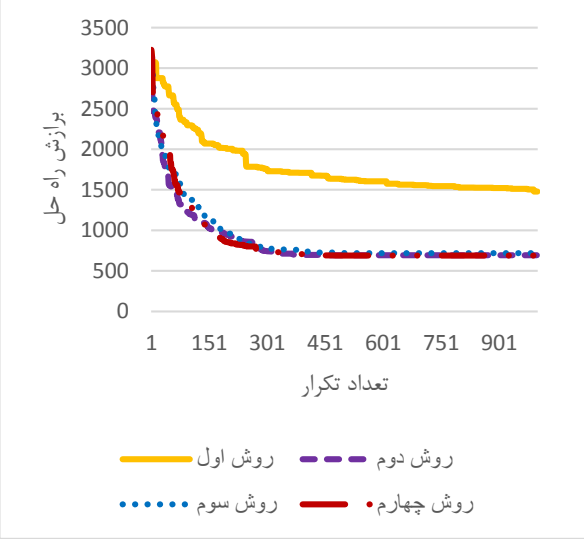
می‌باشد. در جدول٤ نتایج مقایسه روش‌های مختلف استاندارد، الگوریتم VTPSO ارائه‌شده توسط مرجع مقاله [Akhand et al. 2020] و ارائه‌شده در این مقاله، آورده‌شده‌است که در آن در بیشتر موارد روش پیشنهادی چهارم عملکرد بهتری نسبت به دیگر روش‌ها از خود نشان‌می‌دهد. به طور مثال برای حل مسئله ST70 میانگین هزینه راه حل به کمک الگوریتم پیشنهادی چهارم و الگوریتم VTPSO به ترتیب ۷۰۵ و ۷۱۱ می باشد که نشان از کارائی الگوریتم پیشنهادی چهارم نسبت به روش VTPSO دارد. همچنین، زمان اجرای الگوریتم‌های پیشنهادی شبکه‌ای نسبت به الگوریتم پایه بالاتر می‌باشد چون نیاز به محاسبات بیشتری دارد. به طور مثال الگوریتم پیشنهادی روش چهارم و الگوریتم پایه برای حل مسئله ST70 به ترتیب میانگین زمان اجرای ٥٤٤ و ۱۹۰ ثانیه دارد.

همچنین در نمودار شکل۹ برای مجموعه ST70 با چهار روش براساس تعداد تکرار و برازش بدست آمده، نشان‌داده‌شده‌است. همان‌طور که مشخص‌است برای این نمونه برازش روش چهارم نسبت به دیگر روش‌ها برای حل مسئله ST70 فروشنده دوره‌گرد مناسب‌تر می‌باشد.

یکی از ابزارها برای بررسی وضعیت متغیرهای مسئله، تحلیل حساسیت است که به کمک آن می‌توان بررسی‌نمود که کدام‌یک از پارامترهای الگوریتم مورد استفاده برای حل یک مسئله تاثیربیشتری بر روی نتایج تابع هدف دارد. به منظور بررسی میزان حساسیت پارامترهای الگوریتم‌های پیشنهادی برای حل مسئله TSP چندین پارامتر مهم c1, c2, w, wdamp, Iteration و nPop درنظرگرفته‌شده‌است. همان طور که در جداول ٤ و ٥ نشان‌داده‌شده، با مقداردهی مقادیر مختلف برای هر کدام از پارامترهای مذکور، نشان‌می‌دهد که الگوریتم‌های پیشنهادی برروی پارامترهای w,w_{damp} و nPop حساسیت چندانی ندارد، ولی تنظیم مناسب دو ضریب c1,c2 حساس می‌باشد، به گونه‌ای که هر چقدر مقدار c1 (میزان تاثیر بهترین موقعیت فعلی ذره) به سمت صفر میل‌کند، تقریباً میانگین هزینه نامناسب‌تری دارد. به این علت که تاثیر بهترین موقعیت فعلی ذره نسبت به بهترین موقعیت سراسری ذرات برای بدست‌آوردن موقعیت بعدی ذره خیلی کمتر می‌شود. لازم به توضیح است مقداردهی پارامترهای تاثرگذار c1,c2(c2=4-c1) با شش سطح از سطح اول تا سطح ششم به ترتیب برای مقادیر c1=0 تا c1=3*rand(i)) (rand(i) یک عدد تصادفی بین صفر و یک را نشان‌می‌دهد) نامگذاری‌شده‌است. در این مقاله، پارامترها مطابق سطح ششم مقداردهی‌شده‌است. همچنین، تعیین مقدار پارامتر Iteration نیز اهمیت دارد. به گونه‌ای که در تعداد تکرار پایین‌تر (قبل از ٥۰۰ تکرار) هر چقدر مقدار c1 کمتر باشد نتیجه نامطلوب‌تری حاصل‌می‌شود. در شکل ۷ نمودار بهترین هزینه سطوح مختلف (از سطح اول تا ششم) نسبت به تکرارهای مختلف برای الگوریتم پیشنهادی روش چهارم نشان‌داده‌شده‌است. روش سطح اول و دوم به دلیل پایین‌بودن مقدار پارامتر c1 آنها میانگین هزینه نامناسب‌تری دارند ولی سطح ششم به دلیل اینکه تعادل مناسب‌تری بین دو پارامتر c1,c2 ایجادمی‌کند، میانگین و بهترین هزینه را در تکرارهای پایین (قبل از ٥۰۰ تکرار) بدست‌می‌آورد. بنابراین در این الگوریتم‌های پیشنهادی سه پارامتر c1,c2 و تعداد تکرار حساسیت بیشتری نسبت به پارامترهای دیگر الگوریتم دارند.

همان طور که جدول ٤، نشان‌می‌دهد الگوریتم‌های پیشنهادی نسبت به تنظیم پارامتر c1,c2 به ویژه در تعداد تکرارهای پایین حساس هستند(به دلیل اینکه c2 برابر با 4-c1 می‌باشد مقدار آن در جدول آورده‌نشده‌است). به طور مثال برای تعداد تکرار ٥۰۰ زمانی که c1=0 (سطح اول) و c1=3*rand(i) (سطح ششم) تنظیم‌شده‌باشد به ترتیب مقدار بهترین هزینه برابر ۷۹۹٤ و ۷٥٤٤ می‌باشد که برای سطح اول با تعداد تکرار ٥۰۰ بهترین هزینه کل اجرا محقق‌نشده‌است. با افزایش تعداد تکرار به حدود ۱۰۰۰ بهترین هزینه برای سطح اول ۷٥٤٤ بدست‌می‌آید. از طرفی به طور مثال زمانی که پارامترها مطابق سطح اول و سطح ششم،

مقداردهی‌شده‌باشد، مقدار میانگین هزینه با تعداد تکرار ۵۰۰ به ترتیب برابر ۸۵۵۲ و ۷۹۳۷ می‌باشد که نشان‌دهنده این است که c1 های پایین‌تر میانگین هزینه بالاتری نسبت به مقدار c1 بالاتر دارد. در واقع، الگوریتم‌های پیشنهادی نسبت به تنظیم پارامتر c1 و تعداد تکرار، حساس‌می‌باشند. در شکل ۱۰ تاثیر تعداد تکرار و تنظیم پارامتر c1 نشان‌داده‌شده‌است. زمانی که از الگوریتم پیشنهادی چهارم برای حل مسئله‌فروشنده‌دوره‌گرد استفاده‌شده‌است، در سطح اول نسبت به سطوح دیگر (مانند سطح ششم) دیرتر بهترین هزینه بدست‌می‌آید.



شکل ۹. نمایش مقایسه نتایج روش‌های مختلف برای یافتن کوتاهترین مسیر هامیلتونی برای مسئله ST70

جدول ٤. تحلیل حساسیت پارامتر c1 الگوریتم‌های پیشنهادی برای حل مسئله TSP

		C1					
		۰	۰,۵	۱	۲	۳	3*rand(i)
		سطح اول	سطح دوم	سطح سوم	سطح چهارم	سطح پنجم	سطح ششم
⋮	میانگین هزینه	۸۵۵۲	۸۳۳٤	۸۰۲۷	۸۰۹۳	۸۰۰۲	۷۹۳۷

		C1					
		۰	۰,۵	۱	۲	۳	3*rand(i)
		سطح اول	سطح دوم	سطح سوم	سطح چهارم	سطح پنجم	سطح ششم
	بهترین هزینه	۷۹۹٤	۷۷٤۹	۷٦٦۱	۷۷۹۱	۷۸۱۵	۷۵٤٤
⋮	میانگین هزینه	۸۰۷۲	۸۱۵۱	۸۰۲۷	۸۰۹۳	۸۰۰۲	۸۰۸٦
	بهترین هزینه	۷۵٤٤	۷۷۷۹	۷٦٦۱	۷۷۹۱	۷۸۱۵	۷۵٤٤
⋮	میانگین هزینه	۸۰٤۳	۸۱۵۱	۸۰۲۷	۸۰۹۳	۸۰۰۲	۸۰۸٦
	بهترین هزینه	۷۵٤٤	۷۷٤۹	۷٦٦۱	۷۷۹۱	۷۸۱۵	۷۵٤٤

جدول ۵. تحلیل حساسیت پارامتر nPop الگوریتم‌های پیشنهادی برای حل مسئله TSP

		nPop ۲۵	۵۰	۱۰۰	۱۵۰	۲۰۰
		تعداد تکرار				
۵۰۰	میانگین هزینه	۸۱٦۳	۸۰۱۷	۸۰۹۳	۷۹۳۷	۸۱٤۱
	بهترین هزینه	۷۹۰۲	۷۵٤٤	۷۵٤٤	۷۵٤٤	۷۷٤۲
۱۰۰۰	میانگین هزینه	۷۹٦۷	۸۰۰۳	۸۰۹۳	۷۹۳۷	۸۱٤۱
	بهترین هزینه	۷۵٤٤	۷۵٤٤	۷۵٤٤	۷۵٤٤	۷۷٤۲
۱۵۰۰	میانگین هزینه	۷۹٦۷	۷۹۸۵	۸۰۹۳	۷۹۳۷	۸۱٤۱
	بهترین هزینه	۷۵٤٤	۷۵٤٤	۷۵٤٤	۷۵٤٤	۷۷٤۲

همان طور که در جدول ۵ نشان‌داده‌شده‌است، تعداد جمعیت اولیه در تعداد تکرارهای مختلف تابع هزینه را چندان تغییرنمی‌دهد، به طور مثال با تعداد تکرار ۵۰۰ زمانی که nPop=25,200 می‌باشد

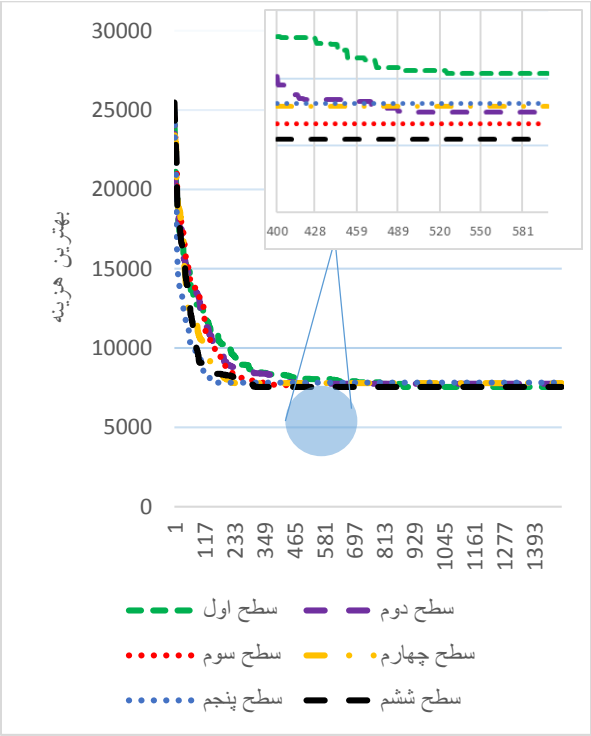
به ترتیب میانگین هزینه برابر ۸۱۶۳ و ۸۱۴۱ می‌باشد که با یکدیگر تفاوت چندانی ندارند.

شکل ۱۱ میزان پایداری تابع هدف بدست‌آمده برای ۳۰ بار اجرای متوالی الگوریتم پیشنهادی روش چهارم برای حل مسئله berlin52 نشان‌می‌دهد که در آن انحراف معیار نتایج ۲۵۶,۵۲ می‌باشد که نشان از پایداری الگوریتم پیشنهادی است.

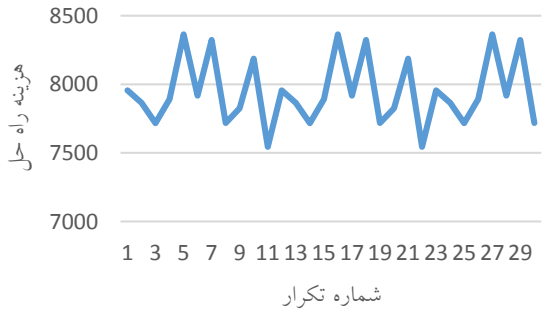
جدول ٦. مقایسه روش‌های مختلف حل مسئله فروشنده دوره‌گرد

نام مسئله	معیار	روش اول	روش دوم	روش سوم	روش چهارم	VTPSO
burma14	بهترین هزینه	۳۰,۸۷	۳۰,۸۷	۳۰,۸۷	۳۰,۸۷	۳۰,۸۷
	بدترین هزینه	۳۰,۸۷	۳۰,۸۷	۳۰,۸۷	۳۰,۸۷	---
	میانگین هزینه	۳۰,۸۷	۳۰,۸۷	۳۰,۸۷	۳۰,۸۷	۳۰,۸۷
	انحراف معیار	۰	۰	۰	۰	۰
	درصد خطا	۷,۰۸–	۷,۰۸–	۷,۰۸–	۷,۰۸–	---
	میانگین زمان اجرا(ثانیه)	۱۳۸	۲۸۰	۵۳۳	٤٩٤	---
ulysses22	بهترین هزینه	۷۵,۳	۷۵,۳	۷۵,۳	۷۵,۳	۷۵,۳۱
	بدترین هزینه	۷۷	۷۷	۷۷	۷٦	---
	میانگین هزینه	۷٦	۷٦	۷٦	۷۵,۳۳	۷۵,۳۳
	انحراف معیار	۰,۳۸	۰,٤٤	۰,٤٤	۰,۱۵	۰,۰٦
	درصد خطا	۸,۳۷	۸,۳۷	۸,۳۷	۷,٤۱	---
	میانگین زمان اجرا(ثانیه)	۱۳۹	۳۲۵	۵۳۱	۵۱۲	---
berlin52	بهترین هزینه	۷۸۱٦	۷۵٤٤	۷۵٤٤	۷۵٤٤	۷۵٤٤
	بدترین هزینه	۹٤۳۸	۸۵۰۵	۸۷۷۰	۸۳٦۳	---
	میانگین هزینه	۸٤۵٤	۸۰۵۲	۸۰۰۲	۷۹۳۷	۷۷۲۸
	انحراف معیار	۳۵۷,٤۸	۲٤۵,۲۳	۲۸۵,۷٤	۲٤٦,۵۲	۱٦۱,۸
	درصد خطا	۱۲,۰۹	٦,۷٦	٦,۱	۷,۲۱	---
	میانگین زمان اجرا(ثانیه)	۱٤٦	۳٤۰	۵۳۵	۵۲۰	---
st70	بهترین هزینه	٦۹۷	٦۸٦	٦۸٤	٦۸٦	٦۷۷
	بدترین هزینه	۹۱٤	۷۵۳	۷٦۱	۷٦۳	---
	میانگین هزینه	۷۹۷	۷۱۳	۷۱٤	۷۰۵	۷۱۱
	انحراف معیار	۵٦,۷۷	۱۹,۵۵	۲۳,٤٤	۱٤,۸۳	۱٤,۵۷
	درصد خطا	۱۸,۰۷	۵,٦۳	۵,۷۸	٤,٤٤	---

نام مسئله	معیار	روش اول	روش دوم	روش سوم	روش چهارم	VTPSO
	میانگین زمان اجرا(ثانیه)	۱۹۰	۳۵٦	٥٤٦	٥٤٤	---
lin105	بهترین هزینه	۱۸۰٤۰	۱٤٥۳٦	۱٤٥۳٤	۱٤٥۰۹	۱٤۳۸۳
	بدترین هزینه	۲٥٦٦۷	۱٦۰۳۳	۱٦٥٥۲	۱٦۸۲۲	---
	میانگین هزینه	۲۲۲٥۹	۱٥۳۷٤	۱٥۳۸۷	۱٥۲٥۹	۱٥٦۸٤
	انحراف معیار	۱۹٥۱,٦۲	٤۱۸,٥۷	٤۹٦,٤٦	٤۷٥,٥٥	٦۱۰,۳۸
	درصد خطا	٥٤,۸	٦,۹۲	۷,۰۱	٦,۱۲	---
	میانگین زمان اجرا(ثانیه)	۲۰۰	۳٦٤	٥۹۰	٥۸٦	---



شکل ۱۰. تحلیل حساسیت پارامترهای c1,c2 در تکرارهای مختلف برای حل مسئله TSP به کمک الگوریتم پیشنهادی چهارم

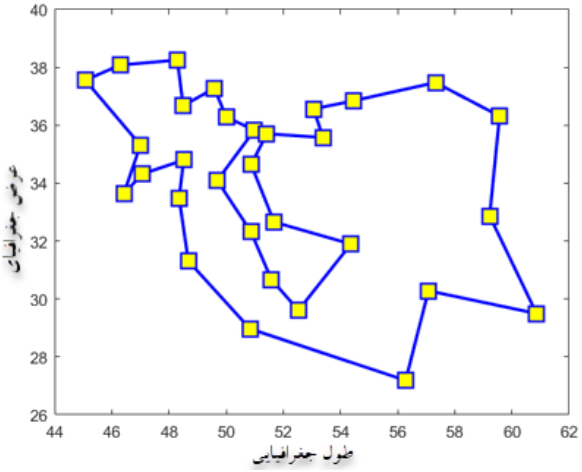


شکل ۱۱. میزان پایداری میانگین هزینه برای حل مسئله berlin52 به کمک الگوریتم پیشنهادی چهارم برای ۳۰ بار اجرای متوالی

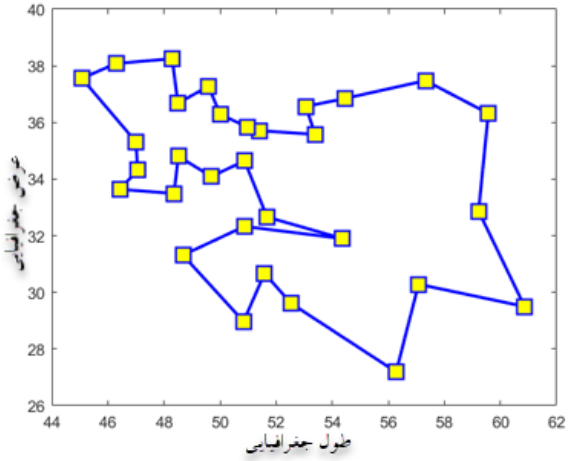
۵-۲ گروه دوم: یافتن کوتاهترین مسیر هامیلتونی استان‌های ایران

پس از بکارگیری و ارزیابی الگوریتم‌های مختلف پیشنهادی برای یافتن کوتاهترین دور هامیلتونی بر روی مسائل استاندارد کتابخانه TSPLib ، نوبت به حل مسئله کاربردی یافتن کوتاهترین دور هامیلتونی استان‌های ایران است. در مرجع [yaghini, Momenei and Sarmadi, 2010] برای یافتن کوتاهترین مسیر هامیلتونی ٤۲۳ شهر ایران از الگوریتم‌های جستجوی ممنوعه و ممتیک استفاده‌شده‌است. در همین راستا در این مقاله برای یافتن کوتاهترین

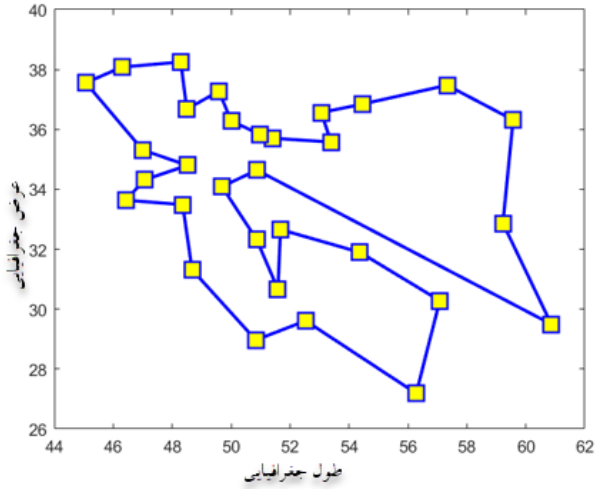
مسیر هامیلتونی تمام ۳۱ مرکز استان‌های کشور ایران، درنظرگرفته‌شده‌است که در آن به کمک سایت Google Maps فاصله تک تک مراکز استان‌های ایران در قالب یک ماتریس ۳۱ در ۳۱ با عنوان ماتریس فاصله در نظرگرفته‌شده‌است. به طور مثال در سایت Google Map فاصله میان مرکز استان تهران و مرکز استان مرکزی ۲۷۹ کیلومتر نشان‌داده‌شده‌است. بقیه استان‌ها نیز به همین طریق بدست‌آمده تا ماتریس ۳۱ در ۳۱ فاصله میان مراکز استانی کامل‌گردد. حال با داشتن ماتریس فاصله می‌توان کوتاهترین مسیر هامیلتونی برای پیمودن از مرکز استان مبداء و پس از آن برگشت به مرکز استان مبداء به کمک چهار الگوریتم پیشنهادی، بدست‌آید. از طرفی طول و عرض جغرافیایی هر مرکز استان نیز از روی سایت Google Map بدست‌آمده و به ترتیب به عنوان طول و عرض نقاط در مختصات دو بعدی درنظرگرفته‌می‌شود. به طور مثال طول و عرض جغرافیایی مرکز استان تهران به ترتیب برابر ۵۱,۲۶ و ۳۵,٤ می‌باشد که در پیاده‌سازی به عنوان نقطه (۵۱,۲۶،۳۵,٤) در مختصات دو بعدی درنظرگرفته‌می‌شود. با اعمال چهار الگوریتم برای یافتن کوتاهترین مسیر هامیلتونی مراکز استانی ایران، نقاط مسیر طی‌شده بین استان‌های ایران متناسب با طول و عرض جغرافیایی به ترتیب شکل ۱۲،۱۳،۱٤ و ۱۵ بدست‌آمده‌است. به گونه‌ای اگر نقطه شروع استان تهران در نظرگرفته‌شود در شکل‌ها طول و عرض جغرافیایی (51.26,35.4) نقطه آغازین برای طی‌شدن تمام مراکز استانی ایران خواهدبود. طبق چهار رویکرد مسیر هامیلتونی مراکز استانی طی‌می‌شود و در نهایت به استان تهران برمی‌گردد. همچنین نمودار براساس تعداد تکرار (۱۰۰۰ بار) و بهترین هزینه بدست‌آمده برای پیمودن تمام مراکز استان‌های ایران به کمک چهار الگوریتم به ترتیب در شکل‌های ۱۶،۱۷،۱۸ ۱۹ نشان‌داده‌شده‌است. همان‌گونه که در شکل‌های ۱۲ تا ۱۵ مشخص‌است، روش سوم و چهارم پایداری بیشتری در یافتن مسیر بهینه دارند.در جدول ۷ نتایج الگوریتم‌های مختلف پیشنهادی برای مسیر هامیلتونی استان‌های ایران به نمایش درآمده‌است که نتایج، نشان از عملکرد بهتر روش سوم نسبت به روش چهارم است و روش چهارم پیشنهادی نیز نسبت به دو روش اول و دوم عملکرد بهتری برای یافتن کوتاهترین مسیر هامیلتونی استان های کشور دارد.



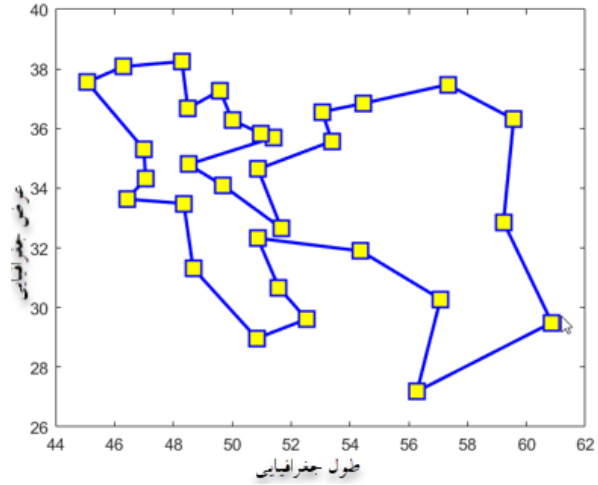
شکل ۱۲. دورهامیلتونی مراکز استان‌های ایران به کمک روش یکم



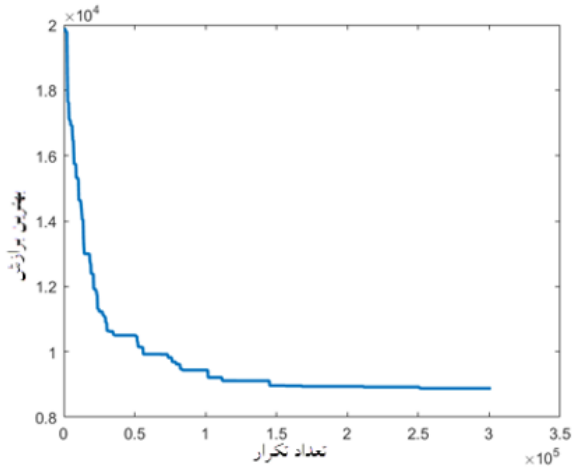
شکل ۱۳. دور هامیلتونی مراکز استان‌های ایران روش دوم



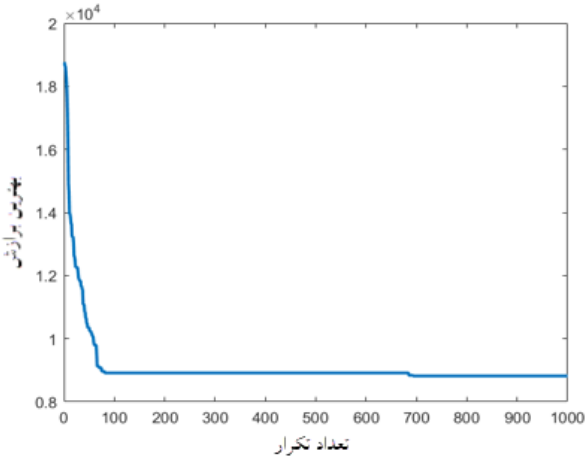
شکل ۱۴. دور هامیلتونی مراکز استان‌های ایران روش سوم



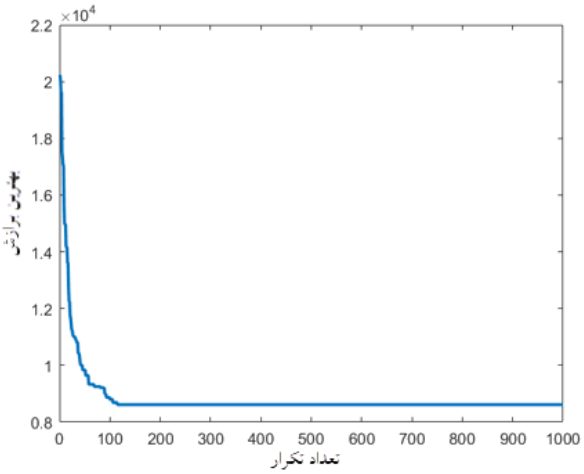
شکل ۱۵. دور هامیلتونی مراکز استان‌های ایران روش چهارم



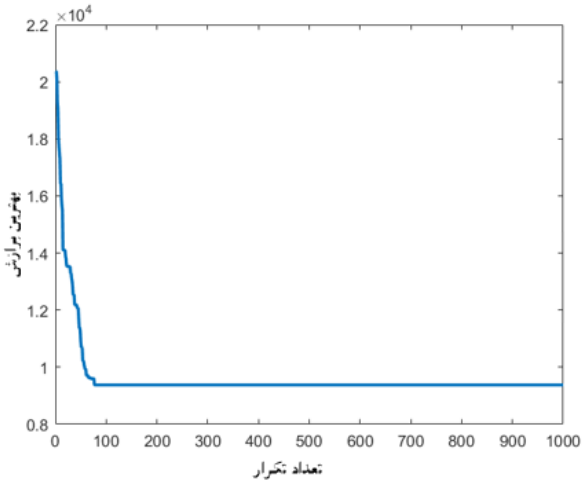
شکل ۱۶. نمودار بهترین هزینه به تعداد تکرار الگوریتم روش اول برای یافتن کوتاهترین مسیر هامیلتونی ۳۱ مرکز استانی ایران



شکل ۱۷. نمودار بهترین هزینه به تعداد تکرار الگوریتم روش دوم برای یافتن کوتاهترین مسیر هامیلتونی ۳۱ مرکز استانی ایران



شکل ۱۸. نمودار بهترین هزینه به تعداد تکرار الگوریتم روش سوم برای یافتن کوتاهترین مسیر هامیلتونی ۳۱ مرکز استانی ایران



شکل ۱۹. نمودار بهترین هزینه به تعداد تکرار الگوریتم روش چهارم برای یافتن کوتاهترین مسیر هامیلتونی ۳۱ مرکز استانی ایران

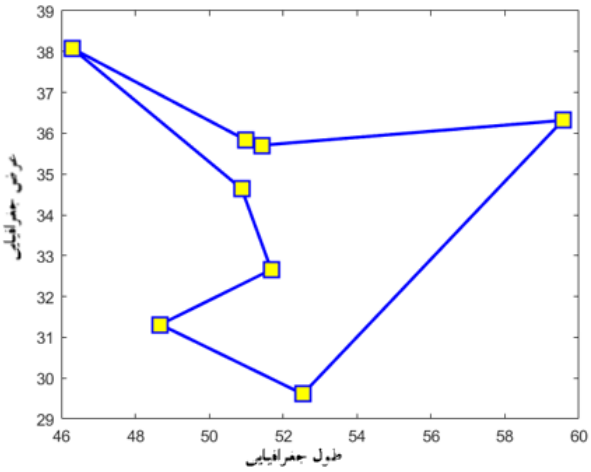
جدول ۷. یافتن کوتاهترین دور هامیلتونی مراکز استان‌های مختلف ایران به کمک الگوریتم‌های مختلف

نام روش	میانگین طول مسیر(km)	بدترین مسیر(km)	بهترین مسیر(km)
روش یکم	۸۹۱۷	۸۹۵۷	۸۸۱۱
روش دوم	۸۸۵۵	۸۹۳۰	۸۸۱۱
روش سوم	۸٦٦٤	۸۸۲۲	۸۵۷٦
روش چهارم	۸٦۸۵	۸۸۰۲	۸۵۷٦

مدل ریاضی یافتن کوتاهترین دور هامیلتونی مراکز استانی همانند مدل ریاضی مسئله فروشنده دوره گرد می‌باشد (رابطه ۲ از بخش ۳) که در آن n نشان دهنده تعداد مراکز استانی است. همچنین به منظور مقایسه نتایج الگوریتم‌های پیشنهادی با روش حل دقیق برای مسئله یافتن کوتاهترین دور هامیلتونی مراکز استانی ایران برای ابعاد پایین با ۸ مرکز استانی درنظرگرفته‌شده‌است. در جدول ۸ ، هزینه و زمان اجرا برای هشت مرکز استانی نشان‌داده‌شده‌است که هر چهار رویکرد، بهینه‌ترین هزینه را بدست‌آورند، ولی زمان اجرای روش دقیق در مقایسه با روش پیشنهادی چهارم به ترتیب ۹٤۹ و ۱۳۸ ثانیه است که نشان از اختلاف زمانی حدود هفت برابری روش دقیق نسبت به روش پیشنهادی می‌باشد. در شکل ۲۰ مسیر کوتاهترین دور هامیلتونی هشت مرکز استانی ایران، نشان‌داده‌شده‌است.

جدول ۸ . مقایسه هزینه و زمان اجرا برای یافتن کوتاهترین مسیر هامیلتونی هشت مرکز استانی ایران

نام روش	هزینه راه حل	زمان اجرا (ثانیه)
روش دقیق	٤۸٤٦,۲	۹٤۹
روش یکم	٤۸٤٦,۲	٤۲
روش دوم	٤۸٤٦,۲	۸۹
روش سوم	٤۸٤٦,۲	۱٤۰
روش چهارم	٤۸٤٦,۲	۱۳۸



شکل ۲۰. مسیر هامیلتونی هشت مرکز استانی ایران به کمک روش دقیق

۳-۵ تحلیل‌های مدیریتی حاصل از پژوهش

در هر پژوهش بررسی تحلیل‌های مدیریتی جایگاه ویژه‌ای دارد. در این مقاله، به دلیل اینکه مسئله بهینه‌سازی یافتن کوتاه‌ترین دور-هامیلتونی و به‌خصوص یافتن کوتاه‌ترین دور هامیلتونی مراکز استانی به کمک الگوریتم‌های پیشنهادی مبتنی بر شبکه پیچیده حل‌شده‌است می‌توان از جهت زوایای مختلف باعث صرفه‌جویی و بهینگی گردد. به طور مثال، با مدیریت در یافتن کوتاه‌ترین مسیر هامیلتونی باعث کاهش هزینه‌ها، افزایش درآمد فرد یا سازمان بکارگیرنده این روش حل مسئله (فروشنده دوره‌گرد یا مدیر گردشگری تور ملی و بین المللی و غیره)، کاهش مصرف سوخت و انرژی (در دراز مدت باعث کمک به اقتصاد خرد و کلان کشور می‌شود)، افزایش رضایتمندی مشریان (به‌طور مثال تور مراکز استان‌های مختلف سریعتر به اتمام‌می‌رسد) می‌گردد. مزایای ذکرشده به دلیل بکارگیری مفهوم شبکه پیچیده به همراه الگوریتم بهینه‌سازی ازدحام ذرات همراه با جستجوی محلی می‌باشد که باعث افزایش کارائی حل مسائل بهینه-سازی مانند فروشنده دوره‌گرد گردیده‌است.

۶. نتیجه‌گیری و کارهای آینده

مسائل بهینه‌سازی مانند یافتن کوتاهترین دور هامیلتونی را می‌توان به کمک روش‌های تقریبی مانند الگوریتم ابتکاری و فراابتکاری، حل‌کرد. برای یافتن کوتاهترین دور هامیلتونی از الگوریتم بهینه‌سازی ازدحام ذرات استفاده‌شده‌است. سپس این الگوریتم به کمک جستجوی محلی پیشنهادی تبدیل به الگوریتم ممتیک شده‌است. در این مقاله برای ایجاد ارتباط بین ذرات از مفهومی بنام شبکه پیچیده استفاده‌شده‌است. در این شبکه به منظور تشکیل شبکه از ارتباط میان ذرات استفاده‌می‌شود. به منظور ارزیابی شبکه نیز سنجه درجه گره، بکارگیری‌شده‌است. در هر مرحله با ارزیابی شبکه پیچیده برای انتخاب یک راه حل، تصمیم‌گیری‌می‌شود. آن گره‌ای که درجه بیشتری دارد شانس انتخاب بیشتری نیز برای انتخاب‌شدن دارد. برای بررسی تاثیر شبکه پیچیده برای یافتن کوتاهترین دور هامیلتونی دو گروه مسائل استاندارد کتابخانه TSPLib و یافتن کوتاهترین دور هامیلتونی استان‌های ایران درنظرگرفته‌شده‌است. نتایج، بدست‌آوردن برازش مناسب‌تر به کمک روش شبکه پیچیده و الگوریتم بهینه‌سازی ازدحام ذرات ممتیکی نسبت به الگوریتم پایه و الگوریتم بهینه‌سازی ازدحام ذرات ممتیکی را نشان‌می‌دهد. به منظور بهبود عملکرد الگوریتم‌های ممتیکی پیشنهادی می‌توان از سنجه‌های دیگری مانند ضریب خوشگی، مرکزیت نزدیکی و مرکزیت درجه در شبکه پیچیده استفاده‌نمود. همچنین برای اعمال تاثیر شبکه پیچیده برای انتخاب راه‌حل مناسب‌تر در الگوریتم بهینه‌سازی ازدحام ذرات، می‌توان در معادله محاسبه موقعیت ذره در الگوریتم PSO در قسمت بهترین موقعیت سراسری به جای بهترین موقعیت فعلی ذره، از شبکه پیچیده استفاده‌نمود یا اینکه از روش‌های ممتیکی دیگر مانند جستجوی ممنوعه و شبیه‌سازی تبرید به جای جستجوی محلی پیشنهادی (عملیات درج، معکوس و جابه‌جائی) برای تقویت راه‌حل

موجود استفاده‌کرد. همچنین در الگوریتم شبکه پیچیده‌ای نیز می‌توان معیارهای دیگر را برای بررسی وضعیت شبکه مد نظر قرار داد.

۷. پی‌نوشت‌ها

1- Particle Swarm Optimization
2- Exploration
3- Exploitation
4- Random-Key Cuckoo Search Algorithm
5- Lévy Flights
6- Bat Algorithm
7- Symbiotic Organisms Search Algorithm
8- Hybrid elitist-ant Algorithm
9- Intensification
10- Iterated Local Search
11- Gap
12- Spider Monkey Optimization(SMO)
13- Discrete SMO
14- Local Leader
15- Global Leader
16- Velocity Tentative PSO
17- Fitness
18- Chaos Theory

۸. منابع

- حیدرآبادی زاده، نسرین، غنی زاده، علیرضا، (۱۳۹۹) "ارزیابی توابع جریمه مختلف در بهینه‌سازی خط پروژه راه با استفاده از الگوریتم‌های بهینه‌سازی ازدحام ذرات شتابدار (APSO) و برخورد اجسام(CBO)"، فصلنامه مهندسی حمل و نقل، دوره ۱۱، شماره ۳، صفحه ۶۹۷-۷۱۷.

- صالحیان، فرهاد، توکلی مقدم، رضا، و نوروزی، نرگس (۱۳۹۸) "حل مساله مسیریابی وسایط نقلیه با در نظر گرفتن رضایت‌مندی مشتریان و کاهش انرژی مصرفی با الگوریتم زنبور عسل"، فصلنامه مهندسی حمل و نقل، دوره ۱۱، شماره ۲، صفحه ۲۹۹-۳۱۱.

- یقینی، مسعود، مومنی، محسن، سرمدی، محمدرضا، (۱۳۸۹) "یافتن کوتاهترین مسیر همیلتونی برای شهرهای ایران با استفاده از الگوریتمهای جستجوی ممنوعه و ممتیک"، فصلنامه مهندسی حمل و نقل، دوره ۲، شماره ۲، صفحه ۱۸۱-۱۹۶.

- Adamo T., Ghiani G. and Guerriero E. (2020) "An enhanced lower bound for the time-dependent travelling salesman problem", Computers and Operations Research, Vol.113, No.104795, pp. 1-9.

- Akhand M.A.H., Ayon S.I., Shahriyar S.A. and Siddique N. (2020) "Discrete Spider Monkey Optimization for Traveling Salesman Problem", Applied Soft Computing Journal, Vol.86, No. 105887.

- Amouei S., Mirzaie K. (2019) "Vector quantization using a modified firefly algorithm for image compression", Tabriz Journal of Electrical Engineering, Vol.49, No.2, pp. 693-707.

- Auh S. and Menguc B. (2005) "Balancing exploration and exploitation: the moderating role of competitive intensity", Journal of Business Research, Vol.58, No.12, pp. 1652-1661.

- Baraba´si A.L. and Albert R. (1999) "Emergence of scaling in random networks", SCIENCE, Vol.286, No.1, pp.509-512.

- Brown C.T., Liebovitch L.S. and Glendon R. (2007) "Lévy flights in dobe ju/'hoansi foraging patterns", Human Ecology, Vol.35, No.1, pp. 129-138.

- Cheng M.Y. and Prayogo D. (2014) "Symbiotic organisms search: a new metaheuristic optimization algorithm", Computers & Structures, Vol.139, No.1, pp. 98-112.

- Corman F., Andrea D., Dario P. and Marco P. (2010) "A tabu search algorithm for rerouting trains during rail operations", Transportation Research Part B: Methodological, Vol.44, No.1, pp. 175-192.

- Dowlatshahi M.B., Derhami V. and Nezamabadi-pour H. (2019) "Gravitational search algorithm with nearest-better neighborhood for multimodal optimization problems", Babol Noshirvani University of Technology, Vol.8, No.3, PP. 1-9.

- Eremeev A.V. and Kovalenko Y.V. (2020) "Amemetic algorithm with optimal recombination for the asymmetric travelling salesman problem", Memetic Computing, Vol.12, No. 1, pp. 23-36.

- Ezugwu A.E.S and Adewumi A.O. (2017) "Discrete symbiotic organisms search algorithm for travelling salesman problem", Expert Systems with Applications, Vol.87, No.1, pp. 70-78.

- Jaradat G.M. (2018) "Hybrid elitist-ant system for a symmetric traveling salesman problem: case of jordan", Neural Computing and Application, Vol.29, No.2, pp. 565-578.

- Kennedy J. and Eberhart R. (1995) "Particle swarm optimization", Proceedings of the IEEE International Conference on Neural Networks, pp. 1942-1948.

- Letchford A. and Nasiri S.D. (2015) "The steiner travelling salesman problem with correlated costs", European Journal of Operational Research, Vol.245, No.1, pp. 62-69.

- Liu B., Wang L., Jin Y.H., Tang F. and Huang D.-X. (2005) "Improved particle swarm optimization combined with chaos", Chaos, Solitons & Fractals, Vol.25, No.5, pp. 1261-1271.

- Mafarja M.M. and Mirjalili S.A. (2017) "Hybrid Whale Optimization Algorithm with Simulated Annealing for Feature Selection", Neuro computing, Vol.260, No.1, pp. 302-312, 2017.

- Mitchell M. (2006) "Complex systems: network thinking", Artificial Intelligence, Vol.170, No.2006, pp. 1194-1212.

- Ouaarab A., Ahiod B. and Yang X.S. (2014) "Random-key cuckoo search for the travelling salesman problem", Soft Computing, Vol.19, No.4, pp. 1099-1106.

- Saji Y. and Riffi M.E. (2016) "A novel discrete bat algorithm for solving the travelling salesman problem", Neural Computing and Applications, Vol.27, No.7, pp. 1853-1866.

- Strasser S., Sheppard J. and Butcher S. (2016) "A new discrete particle swarm optimization algorithm," GECCO '16 Proceedings of the Genetic and Evolutionary Computation Conference 2016, pp. 53-60.

- Taillard E.D. and Helsgaun K. (2019) "POPMUSIC for the travelling salesman problem", European Journal of Operational Research, Vol.272, No.2, pp. 420-429.

- UmapathyP. Venkataseshaiah C.and Arumugam M.S. (2010) "Particle swarm optimization with various inertia weight variants for optimal power flow solution", Discrete Dynamics in Nature and Society, Vol.2010, No.1, pp. 1-15.

- Watts D.J. and Strogatz S.H. (1998) "Collective dynamics of 'small-world' networks", Nature, Vol.393, No.1, pp.440-442.

- Wu D., Jiang N., Du W., Tang K., and Cao X. (2017) “Particle swarm optimization with moving particles on scale-free networks”, Journal Of Latex Class Files, Vol.14, No.8, pp. 1-10.
- Yang X.S. (2010) “A new metaheuristic bat-inspired algorithm”, Springer.

هادی محمدی، درجه کارشناسی در رشته مهندسی کامپیوتر را در سال ۱۳۸۵ از دانشگاه آزاد اسلامی واحد اراک و درجه کارشناسی ارشد در رشته مهندسی کامپیوتر در سال ۱۳۸۹ را از دانشگاه آزاد اسلامی واحد اراک اخذ نمود. در حال حاضر، دانشجوی دکتری رشته مهندسی کامپیوتر در دانشگاه آزاد اسلامی واحد میبد است. زمینه‌های پژوهشی مورد علاقه ایشان، الگوریتم‌های تکاملی، الگوریتم‌های هوش‌جمعی و شبکه‌های پیچیده پویا است.

کمال میرزائی، درجه کارشناسی در رشته مهندسی کامپیوتر را در سال ۱۳۸۰ از دانشگاه علم و صنعت تهران و درجه کارشناسی ارشد در رشته مهندسی کامپیوتر در سال ۱۳۸۳ را از دانشگاه اصفهان اخذ نمود. در سال ۱۳۹۰ موفق به کسب درجه دکتری در رشته مهندسی کامپیوتر از دانشگاه علوم و تحقیقات تهران گردید. زمینه‌های پژوهشی مورد علاقه ایشان، علوم شناختی، الگوریتم‌های تکاملی، شبکه‌های پیچیده پویا، محاسبات نرم، داده کاوی پزشکی، پردازش تصویر و شناسایی الگو بوده و در حال حاضر عضو هیات علمی با مرتبه استادیار در دانشگاه آزاد اسلامی واحد میبد است.

محمدرضا ملاخلیلی میبدی، درجه کارشناسی در رشته مهندسی کامپیوتر را در سال ۱۳۸۰ از دانشگاه شهید بهشتی تهران و درجه کارشناسی ارشد در رشته مهندسی کامپیوتر در سال ۱۳۸۲ را از دانشگاه صنعتی امیرکبیر اخذ نمود. در سال ۱۳۹۳ موفق به کسب درجه دکتری در رشته مهندسی کامپیوتر از دانشگاه آزاد اسلامی واحد علوم و تحقیقات تهران گردید. زمینه‌های پژوهشی مورد علاقه ایشان، ارتباط شبکه‌ای، سیستم‌های یادگیری، الگوریتم‌های تصادفی، بهینه‌سازی تصادفی و محاسبات نرم بوده و در حال حاضر عضو هیات علمی با مرتبه استادیار در دانشگاه آزاد اسلامی واحد میبد است.