

بهینه‌سازی مسیریابی وسایل نقلیه مبتنی بر ترکیب الگوریتم‌های کلونی مورچه و ازدحام ذرات با تابع ابتکاری کسینوس زوایا

حمیدرضا شفیعی^۱، کارشناسی ارشد، وحیدرافع^۲، دانشیار، مریم امیری^{۳*}، استادیار

^۱ مهندسی کامپیوتر- دانشگاه اراک - اراک - ایران - hr.shafiei1994@gmail.com

^۲ و ^۳ مهندسی کامپیوتر- دانشگاه اراک - اراک - ایران - v-rafe@araku.ac.ir، m-amiri@araku.ac.ir

چکیده: وضعیت شلوغی جاده‌ها عامل بسیار مهمی در ترافیک شهری است. برای حل این مشکل پژوهش‌های زیادی انجام شده است که عمده راهکارهای ارائه شده بر مبنای الگوریتم‌های فرا ابتکاری می‌باشند. در بسیاری از این الگوریتم‌ها در ابتدای کار مسیریابی به صورت تصادفی و در محدوده‌ای بزرگ انجام می‌شود. این موضوع موجب افزایش زمان جستجو می‌شود. همچنین در تحلیل‌های صورت گرفته بر مبنای این الگوریتم‌ها تنها فاصله‌ی فیزیکی بین خودروها در نظر گرفته شده است. با توجه به اینکه عوامل محیطی نظیر ترافیک، در مسیریابی بسیار مؤثر هستند، باید تأثیر این عوامل را در مسیریابی در نظر گرفت. بدین منظور برای حل مشکلات مطرح شده، در این مقاله یک روش برنامه‌ریزی مسیر پویا مبتنی بر ترکیب الگوریتم‌های کلونی مورچگان و ازدحام ذرات با اعمال تابع ابتکاری کسینوس زوایا پیشنهاد شده است. این روش عوامل مختلف وضعیت جاده‌ها از جمله طول جاده‌ی شهری و ترافیک ورودی و خروجی تقاطع‌ها را در مسیریابی وسایل نقلیه در نظر می‌گیرد و در مسیریابی برای نقاطی که هم‌راستای مسیر پیمایش به سمت مقصد نهایی باشد شانس بیشتری قائل می‌شود. نتایج حاصل از اعمال مدل پیشنهادی بر روی داده‌های کتابخانه‌ی معتبر TSPLIB که مبتنی بر فاصله فیزیکی بین خودروها است، نشان می‌دهد که زمان جستجوی روش پیشنهادی نسبت به سایر روش‌ها (ده ارزیابی انجام شده) بطور متوسط ۴۰٫۷۴ درصد کاهش داشته است. بیشترین و کمترین میزان کاهش به ترتیب ۹۸٫۱ و ۶٫۰۲ درصد می‌باشد. همچنین آزمایش برنامه‌ریزی مسیر پویا تحت ترافیک جاده‌ها بر روی برخی از تقاطع‌های شهر پکن نشان می‌دهد که روش پیشنهادی به صورت میانگین ۱٫۵۷ درصد تراکم و ازدحام ایجاد می‌کند.

واژه‌های کلیدی: شبکه‌های حمل‌ونقل، برنامه‌ریزی مسیر پویا، کلونی مورچگان، بهینه‌سازی ازدحام ذرات، تابع ابتکاری کسینوس زوایا.

Optimization of vehicle routing based on the combination of ant colony and particle swarm algorithms with the heuristic function of the cosine of angles

Hamidreza Shafiei¹, Vahid Rafe², Associate Professor, Maryam Amiri^{3*}, Assistant Professor

^{1,2,3} Department of Computer Engineering, Faculty of Engineering, Arak university, Arak, IRAN

hr.shafiei1994@gmail.com, v-rafe@araku.ac.ir, m-amiri@araku.ac.ir

Abstract: The congestion of roads are very important factors in urban traffic accidents, especially in crowded cities. A lot of research has tried to solve these problems using meta-heuristic algorithms. In these algorithms, firstly, routing is done randomly over large areas. This will increase the search time. In addition, these algorithms only consider the physical distance between the vehicles. Since environmental factors such as traffic are very effective in routing, these factors should be considered in routing. In this paper, to solve the problems, a dynamic path programming method based on the combination of the ant colony algorithm and particle swarm optimization, along with a cosine function of angle has been proposed. This method takes into account various factors of roads such as the length of the urban road and the incoming and outgoing traffic at intersections. In the method, the points that are aligned with the navigation path towards the final destination are given more chances. Therefore, the overall goal of this paper is to reduce the diversion rate and the search time in finding the best route under road traffic conditions. The results of applying the proposed model on the valid data of TSPLIB library, which is based on the physical distance between cars, show that the search time of the proposed method has decreased by 40.74% on average compared to the results of ten other methods used for evaluation. The highest and lowest rates of decrease are 98.01% and 6.02% respectively. The test of dynamic route planning under road traffic on some intersections of Beijing city also shows the proposed method only causes congestion of about 1.57%.

Keywords: Transport Networks, Dynamic Path Planning, Ant Colony optimization, Particle Swarm Optimization, heuristic Cosine Function of Angle

* Maryam Amiri, m-amiri@araku.ac.ir

۱. مقدمه

بر اساس مطالعات انجام شده در [۱]، در بسیاری از شهرها سیستم حمل و نقل تلفیقی از سیستم سنتی و مدرن است. سیستم سنتی شامل حمل و نقل ناوگانی (تاکسی و اتوبوس) و سیستم مدرن شامل حمل و نقل ریلی (مترو و قطار شهری) می باشد که این استخوان بندی اصلی حمل و نقل کشورها را تشکیل می دهد. مشکلات بسیاری در حمل و نقل کشورها وجود دارد و زیرساخت ها در بسیاری از نقاط جوابگوی حجم ترافیک نیست. در حمل و نقل شهری انتخاب مسیر بهینه برای رانندگان با توجه به پویایی مسیر و ازدحام شدید بسیار دشوار است و این خود می تواند باعث ایجاد سوانح رانندگی، اتلاف وقت و تشدید ترافیک شود. بنابراین در مواجهه با چنین زیرساختی پیدا کردن مسیر بهینه با بالاترین دقت و کمترین زمان جستجو از میان مسیرهای موجود که اکثراً شلوغ هستند، یک مسئله چالش برانگیز است. با توجه به افزایش روزافزون خودروها و مشکلات زیرساخت معابر، پیدا کردن مسیرهای بهینه با الگوریتم های فرا ابتکاری در [۲] یکی از استراتژی های مؤثر در کنترل ترافیک است. بیشتر تحقیق هایی که در این زمینه انجام شده تنها با در نظر گرفتن فاصله ی فیزیکی بین خودروها سعی در حل مسئله داشته اند. از آنجایی که فاصله ی فیزیکی تنها فاکتور مؤثر در پیدا کردن مسیر بهینه نیست، این پژوهش سعی دارد در یک برنامه ریزی پویا جریانات ترافیکی را لحاظ کند و با ارائه مسیر بهینه باعث یکدست شدن ترافیک معابر شود. الگوریتم های فرا

ابتکاری زیادی برای برنامه ریزی مسیر وجود دارد. بسیاری از این الگوریتم ها به دلیل پیچیدگی های مسائل مسیریابی و سنگین بودن تحلیل ها برای برنامه ریزی مسیر مناسب نیستند. اگرچه یکی از پرکاربردترین الگوریتم ها برای برنامه ریزی مسیر الگوریتم کلونی مورچگان^۱ ACO است، این الگوریتم در همگرایی به نقطه بهینه سراسری ضعف دارد و محدوده تنظیم پارامترهای آن بسیار وسیع است. بدین منظور برای رفع این محدودیت ها، در [۳] ترکیب الگوریتم ACO با سایر الگوریتم ها پیشنهاد شده است. با توجه به اینکه فاصله ی فیزیکی تنها فاکتور مؤثر در برنامه ریزی نیست و عوامل محیطی نظیر ترافیک تاثیر گذار است، در ترافیک شهری پیچیده این گونه برنامه ریزی ها مناسب نیستند. در این مقاله، با توجه به مشکلاتی که در حوزه تنظیم پارامترها و همگرایی الگوریتم ACO وجود دارد، الگوریتم ازدحام ذرات^۲ (PSO) که دارای بار محاسباتی پایین و تعداد کم پارامترها می باشد [۴]، با این الگوریتم ترکیب شده است. اختلاف پایه ای الگوریتم بهینه سازی ازدحام ذرات در مقایسه با سایر الگوریتم های ابتکاری در میزان سادگی پیاده سازی است. در صورتی که تابع هزینه بطور مناسب انتخاب شود، الگوریتم به نقطه اکسترمم همگرا می شود [۵]. همچنین برای افزایش دقت و کاهش زمان جستجوی الگوریتم ترکیبی، تابع ابتکاری کسینوس

¹ Ant colony optimization

² Particle Swarm Optimization

- کاهش زمان جستجو در پیدا کردن مسیرهای بهینه در جستجوهای جهانی با اعمال تابع ابتکاری: در ترکیب چند الگوریتم برای جلوگیری از همگرایی زودرس بصورت طبیعی ممکن است زمان جستجو برای پیدا کردن بهینه‌ترین مسیر افزایش یابد که در روش پیشنهادی با اعمال تابع ابتکاری کسینوس زوایا تاثیر این مسئله کاهش داده شده است.

- پیدا کردن مسیرهای بهینه با زمان جستجوی پایین: مطابق با نتایج ارزیابی‌ها، روش پیشنهادی قادر است بهینه‌ترین مسیر را در مقایسه با رایج‌ترین و جدیدترین روش‌های ارائه شده، با زمان جستجوی کمتری یافت نماید.

- پیدا کردن مسیرهایی با نرخ انحراف پایین: مطابق با نتایج ارزیابی‌ها، روش پیشنهادی قادر است بهینه‌ترین مسیرها را با نرخ انحراف پائین (دقت بالا) بیابد که در تکرارهای مداوم با ثبات کامل مسیر بهینه را تکرار می‌کند.

در ادامه این مقاله، در بخش دوم، روش‌های کلاسیک در مسیریابی، الگوریتم‌های مؤثر در مسیریابی و همچنین مطالعات انجام‌شده در حوزه مسیریابی مرور خواهند شد. همچنین در انتهای این بخش، به بررسی الگوریتم‌های پرکاربرد کلونی مورچه و الگوریتم ازدحام ذرات خواهیم پرداخت. در بخش سوم روش پیشنهادی به صورت گام به گام توضیح داده می‌شود. در بخش چهارم به ارزیابی و حل مسئله می‌پردازیم و نتایج به‌دست‌آمده را با نتایج حاصل از اعمال جدیدترین و پرکاربردترین الگوریتم‌های موجود مقایسه می‌کنیم. بخش پنجم به ارائه نتایج کلی مقاله، جمع‌بندی و ارائه پیشنهاداتی برای کارهای آتی می‌پردازد.

۲- مبانی نظری و مرور روش‌های پایه

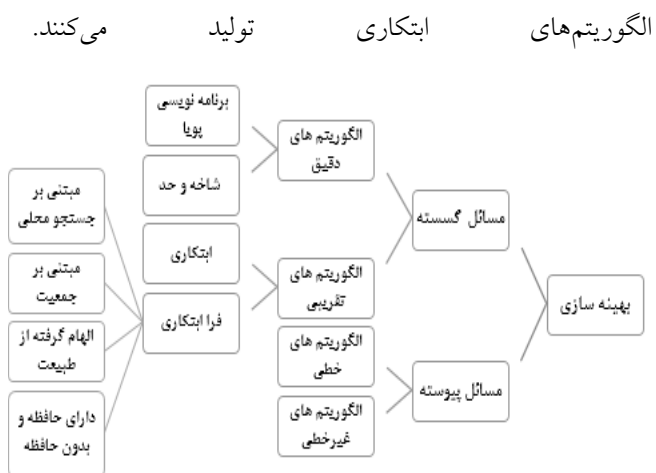
در این بخش در ابتدا به مرور کلی رویکردهای مؤثر در مسیریابی می‌پردازیم. در ادامه، در بخش ۲-۱ رایج‌ترین و

زوایا به آن اضافه شده است. این تابع، زاویه‌ی بین گره‌های مبدأ و مقصد را در نظر می‌گیرد و گره‌ی دارای زاویه‌ی کمتر نسبت به مقصد را در اولویت انتخاب قرار می‌دهد. در روش پیشنهادی برخلاف روش ارائه شده در [۶]، الگوریتم PSO در تمامی تکرارهای الگوریتم ACO استفاده می‌شود و بهینه‌های فردی و سراسری الگوریتم PSO در رابطه احتمال ACO به صورت مداوم تأثیر داده می‌شوند و در نهایت با اعمال تابع ابتکاری کسینوس زوایا دقت الگوریتم ACOPSO افزایش و زمان مسیریابی هم کاهش داده می‌شود. بصورت خلاصه می‌توان گفت ایده اصلی مقاله بررسی تاثیر ترافیک جاده‌ها در برنامه‌ریزی مسیر همچنین تاثیر الگوریتم PSO بر روی پارامترهای ACO، اعمال تابع ابتکاری کسینوس زوایا و یافتن ترکیب جدیدی برای بروزرسانی فرمون روی مسیرها است، که در نهایت موجب می‌شود زمان جستجو برای پیدا کردن بهینه‌ترین مسیر و همچنین نرخ انحراف در پیدا کردن مسیر بهینه کاهش پیدا کند که این خود موجب افزایش دقت نهایی روش اجرایی می‌شود. روش پیشنهاد شده در این مقاله ACOPSO نامگذاری می‌شود. بنابراین نوآوری این مقاله در موارد زیر خلاصه می‌شود:

- کاهش نرخ ازدحام و ترافیک مسیرها: انجام برنامه ریزی مسیر پویا با در نظر گرفتن عوامل محیطی (نظیر ترافیک جاده‌ها) به منظور یکدست کردن ترافیک مسیرها و کاهش نرخ ازدحام مسیرهای مختلف به عنوان یک هدف در روش پیشنهادی لحاظ شده است که در واقعیت و به صورت عملی منجر به کاهش هزینه و مصرف انرژی می‌شود.

- جلوگیری از همگرایی زودرس الگوریتم ACO: در این مقاله، مشکل بزرگ همگرایی زودرس الگوریتم ACO با حذف استقلال انتخاب گره‌ها، ترکیب این الگوریتم با PSO و در نظر گرفتن بهینه‌های فردی و سراسری حل می‌شود. این امر، سبب جستجو در سطح جهانی خواهد شد.

جدیدترین روش‌های مسیریابی و در بخش ۲-۲ الگوریتم‌های پایه روش ACOPSOCA بررسی می‌شوند.



شکل (۱) - طبقه‌بندی الگوریتم‌ها در حوزه مسیریابی

برای حل مسائل مسیریابی وسایل نقلیه در طول سالیان روش‌های متفاوتی ارائه شده که بسیاری از این روش‌ها امروزه منسوخ شده‌اند و کاربرد مناسبی در حل مسائل ندارند. مسائل بهینه‌سازی مطابق شکل ۱ به دو دسته مسائل گسسته و پیوسته تقسیم می‌شوند. مسائل گسسته در دو دسته‌ی دقیق و تقریبی تقسیم‌بندی می‌شوند. الگوریتم‌های دقیق توانایی یافتن جواب بهینه در مسائل بسیار کوچک را به صورت دقیق دارند در الگوریتم‌های تقریبی تضمینی برای پیدا کردن جواب دقیق وجود ندارد. در این الگوریتم‌ها در هر تکرار با توجه به جواب‌های تکرار قبلی، جواب‌های تکرارهای بعدی مشخص می‌شوند و در نهایت جوابی نزدیک به واقعیت به دست می‌آید. الگوریتم‌های تقریبی به دو دسته ابتکاری و فرا ابتکاری تقسیم‌بندی می‌شوند. الگوریتم‌های ابتکاری در مسائلی که پیچیدگی زیادی ندارند، می‌توانند جواب بهینه را بیابند. الگوریتم‌های فرا ابتکاری برای حل مشکلات الگوریتم‌های ابتکاری الگوهایی ارائه کرده‌اند. یکی از ویژگی‌های مهم برای این‌گونه الگوریتم‌ها این است که برای یک محدوده وسیع که جواب تحلیلی ندارند، نزدیکترین جواب به حل مسئله را خواهند یافت [۷]. در واقع این نوع الگوریتم‌ها راهکارهای برون رفت از نقاط بهینه محلی و قابلیت کاربرد در طیف گسترده‌ای از مسائل را دارند. از آنجایی که الگوریتم‌های فرا ابتکاری بر جستجوی عمیق در نقاط مختلف از فضای جواب تأکید دارند، جواب‌های با کیفیت‌تری نسبت به

- بررسی روش‌های پیشین

مسئله‌ی مسیریابی در سال ۱۹۵۹ توسط دانتریک و رامسر مطرح شد [۸]. این مقاله مبنای بسیاری از تحقیقات بعدی قرار گرفت و پس از آن، مسیریابی وسایل نقلیه به کمک الگوریتم‌های فرا ابتکاری بررسی شدند: دوریگوو نسخه اولیه‌ی الگوریتم ACO را برای مسئله‌ی فروشنده دوره‌گرد در سال ۱۹۹۱ منتشر کرد [۹]. از سال ۲۰۰۲ به بعد مسئله‌ی مسیریابی وسایل نقلیه با ترکیب الگوریتم‌های فرا ابتکاری مورد بررسی قرار گرفت. در [۱۰]، از PSO برای بهینه‌سازی پارامترهای الگوریتم ACO استفاده شد. در [۱۱] از ترکیب ACOPSO در مسئله برنامه‌ریزی مسیر استفاده شد. در [۱۲] روش هیبرید را با ACOPSO بهبود بخشیدند. در [۱۳] یک الگوریتم بهبودیافته ACOPSO برای حل مشکل فروشنده دوره‌گرد در مقیاس بزرگ پیشنهاد شده است. در [۱۴] برای مسئله زمانبندی مسیریابی استراتژی حریص تطبیقی، در [۱۵] استراتژی خوشه بندی و در [۱۶] استراتژی تولید سلسله مراتبی با الگوریتم PSO ارائه شده است. در [۱۷] با استفاده از منطق فازی و در [۱۸] با استفاده از

مرجع	سال	روش پیشنهادی	ابزار شبیه سازی - دیتاست	معیارهای کارایی (مزایا و معایب)
[۶]	۲۰۱۹	برنامه ریزی مسیری بویا ACOPSO	MATLAB TSPLIB	این الگوریتم از PSO برای جلوگیری از همگرایی زودرس ACO به این شکل که جواب اولیه را پیدا میکند و در اختیار ACO قرار می دهد. مزیت جستجوی جهانی نسبتا قوی و اختصاص حفظه مجزا به هر مورچه برای نگه داشتن جواب بهینه های فردی عیب روش استفاده از PSO فقط برای جواب اولیه که با گذر زمان از جواب های بهینه کمتری فاصله می گیرد همچنین پیچیدگی تا حدودی بالا می رود.
[۱۴]	۲۰۲۲	استراتژی حریص تطبیقی	MATLAB TSPLIB	این الگوریتم از استراتژی تطبیقی استفاده می کند مزیت روش جستجوی در سطح جهانی است ولی عیب روش استفاده از روش حریصانه در بروزسانی مسیرها است که در نهایت در پیدا کردن مسیرهای بهینه تاثیر منفی دارد.
[۱۵]	۲۰۲۲	استراتژی خوشه بندی	MATLAB TSPLIB	این الگوریتم از سه مفهوم خوشه بندی، تبخیر فرمون تطبیقی و شرایط پایان جدید بر اساس تنوع راه حل در جمعیت برای عملکرد کلی، کاهش زمان اجرا استفاده می کند. مزیت روش تنوع روش ها در سطح جهانی است. معایب میتوان به تعداد تکرار بالا در پیدا کردن مسیر بهینه اشاره کرد.
[۱۶]	۲۰۲۲	استراتژی تولید سلسله مراتبی با الگوریتم PSO	MATLAB TSPLIB	با الگوریتم PSO از همگرایی زودرس جلوگیری می کند و برای بروزسانی مسیرها یک استراتژی سلسله رانینی دولا به مبتنی بر یادگیری حریصانه استفاده می - شود. مزیت عدم همگرایی اولیه است و معایب آن با استفاده از روش حریصانه در نهایت نرخ انحراف جوابها بالا می رود.
[۱۷]	۲۰۲۲	استفاده از منطق فازی	MATLAB JAVA TSPLIB-	با منطق فازی سعی در حل مشکل همگرا شدن زودرس ACO هست. مزایای روش این است تا حدودی توانسته با منطق فازی و استفاده از کنترلر FACS از همگرا شدن روش ACO جلوگیری کند. معایب این روش پیچیدگی بیش از حد است که در هر تکرار برای تابع ابتکاری باید ورودی و خروجی های تعیین کرد که این کار خود موجب افزایش تعداد تکرار برای پیدا شدن مسیر بهینه می شود.
[۱۸]	۲۰۲۲- ۲۰۲۳	الگوریتم زنبور عسل	MATLAB TSPLIB	از تابع ابتکاری MULTI-INSERT استفاده می کند و بهترین همسایه را در اولویت قرار می دهد. مزایای روش تعداد پارامترهای بسیار کم می باشد. معایب این روش تکیه بر جواب های بهینه ابتدایی و نادیده گرفتن جستجوی جهانی در هر تکرار می باشد

۲-۲- روش های پایه

الگوریتم زنبور عسل با تابع Multi insert مسئله مسیریابی حل شده است. در جدول ۱ خلاصه فعالیت های انجام شده در زمینه مسیریابی وسایل نقلیه شامل الگوریتم های حل کلاسیک و الگوریتم های فرا ابتکاری و ترکیبی اخیر لیست شده است. لازم به ذکر است با توجه به محدودیت فضا امکان بررسی جامع کارهای پیشین وجود ندارد علاقمندان می توانند با مراجعه به [۱۹] توضیحات کاملتری را دریافت کنند.

جدول (۱) - خلاصه پژوهش های انجام شده در حوزه مسیریابی

مرجع	سال	روش پیشنهادی	ابزار شبیه سازی - دیتاست	معیارهای کارایی (مزایا و معایب)
[۱۲]	۲۰۰۲	شبیه سازی تبرید	-	از الگوریتم شبیه سازی تبرید با انتخاب گروهی از جواب ها شروع میشود و در هر تکرار با جواب های همسایه مقایسه میشود. عیب روش همگرا شدن سریع و عدم توانایی در پیدا کردن بهینه ترین جواب را دارد
[۱۳]	۲۰۰۵	الگوریتم زنتیک	-	از الگوریتم زنتیک استاندارد استفاده شده که بشدت میل به همگرایی زودرس دارد. عیب این الگوریتم جواب های غیر بهینه با نرخ انحراف بالا است.
[۲۰]	۲۰۰۹	روش ابتکاری دوفازی برای vtp	-	در این مقاله برای بهبود پارامترهای ACO از زنتیک استفاده شده است نتایج بدست آمده نشان می دهد موفقیت آنچنانی ندارد و نرخ انحراف جواب ها بالا است.
[۲۱]	۲۰۱۰	بهینه سازی پارامترهای ACO بر اساس PSO	MATLAB TSPLIB	با الگوریتم PSO از همگرایی زودرس جلوگیری می کند و جواب اولیه را در اختیار ACO قرار می دهد. مزیت عدم همگرایی زودرس اولیه است و معایب آن تعداد تکرار بالا برای پیدا کردن مسیری بهینه و بالا بودن نرخ انحراف جواب ها بهینه است.

به روزرسانی فرامون بر اساس رابطه ۳ محاسبه می شود.

(۳)

در (۷) رابطه ρ نرخ تبخیر فرامون است که از تجمع فرامون جلوگیری می کند. این امر سبب می شود تصمیمات نامناسب قبلی در نظر گرفته نشود و در نتیجه از همگرایی سریع جلوگیری می کند. مقدار فرامونی است که توسط مورچه در یال (i, j) به جا گذاشته می شود و طبق رابطه ۴ محاسبه می شود. در این رابطه m تعداد مورچه ها را نشان می دهد و $\Delta\tau_{ij}^k$ مقدار فرامونی است که قرار است توسط مورچه k در یال (i, j) ریخته شود و مقدار آن توسط رابطه ۵ محاسبه می شود که مقدار Q معمولاً ۱ در نظر گرفته می شود و L_k طول دور کامل

$$P_{ij}^k = \begin{cases} \frac{\tau_{ij}^\alpha \eta_{ij}^\beta}{\sum_{w \in N^k} \tau_{ij}^\alpha \eta_{ij}^\beta} & \text{مورچه } k \text{ از یال } (i, j) \text{ می گذرد} \\ 0 & \text{تغییر می شود} \\ & j \notin J_k(i) \end{cases} \quad (1)$$

(۴)

$$\Delta\tau_{ij}^k = \begin{cases} \frac{Q}{L_k} & \text{اگر یال } (i, j) \text{ توسط مورچه } k \text{ طی شود} \\ 0 & \text{در غیر این صورت} \end{cases} \quad (5)$$

۲-۲-۲-۲- بهینه سازی ازدحام ذرات PSO

در الگوریتم ازدحام ذرات [۲۱] هر ذره دو ویژگی سرعت V_t و موقعیت X_t دارد که بر اساس تجربیات شخصی و گروهی تغییر می کنند. تغییر مکان ذره تحت تأثیر همسایگانش قرار می گیرد و ذرات به سمت نواحی موفق حرکت می کنند. علت آن این است که ذرات از همدیگر می آموزند و با دانش کسب شده به سمت بهترین همسایگان حرکت می کنند و در ادامه در حین جستجو مقادیر مناسب بهینه فردی انتخاب می شود و ذره آن را به خاطر می سپارد. همین روال ادامه می یابد تا بهینه سراسری انتخاب شود. سرعت با توجه به رابطه ۶ و موقعیت با توجه به رابطه ۷ به روزرسانی می شود.

با توجه به اینکه روش ارائه شده در این مقاله مبتنی بر الگوریتم های PSO و ACO می باشد در ادامه به شرح آنها می پردازیم.

۲-۲-۱- الگوریتم ACO

در [۲۰]، فرآیند جستجوی مورچه دو مدل احتمال انتقال و مدل به روزرسانی فرامون دارد. در این الگوریتم، مورچه ها در شبکه $n * m$ قرار می گیرند که n تعداد گره ها و m تعداد یال ها می باشد. مورچه ها یک رأس را از میان جواب های کاندید برای حرکت روبه جلو انتخاب می کنند. در رابطه ۱ که به رابطه احتمال معروف است داریم:

که P_{ij}^k احتمال حرکت مورچه k از نقطه i به نقطه j است. $J_k(i)$ یعنی شهرهایی که هنوز مورچه k انتخاب نکرده و J_{ij} شدت غلظت فرامون یال (i, j) را مشخص می کند. همچنین، پارامترهای α و β به ترتیب اهمیت نسبی شدت غلظت فرامون مسیر و اطلاعات هیوریستیکی را اندازه گیری می کنند. هرچه میزان غلظت فرامون مسیر بیشتر باشد احتمال انتخاب آن بیشتر می شود و تصادفی بودن جستجو کم می شود. پارامتر β هرچه مقدار بیشتری داشته باشد، عملکرد مورچه ها بر اساس اطلاعات هیوریستیکی بیشتر خواهد بود. η_{ij} که بر اساس رابطه ۲ محاسبه می شود، اطلاعات هیوریستیکی مسئله است که می تواند هر نوع داده ای اضافی مانند ترافیک جاده ها باشد به طوری که اولویت شهرها را برای حرکت انتخاب می کنند. در این رابطه، d_{ij} فاصله بین شهرهای i و j است که η_{ij} رابطه معکوسی با d_{ij} (تابع هزینه) دارد.

$$\eta_{ij}(t) = \frac{1}{d_{ij}} \quad (2)$$

ویژگی‌های $(x_i(t))$ و $(v_i(t))$ پارامترهای جستجو و بهینه‌سازی و

همگرایی سریع است بدین شکل استفاده شده است که PSO

فقط جواب اولیه مناسب $v_i(t)$ می $x_i(t)$ قرار

می‌دهد. ولی در این مقاله PSO به عنوان بخشی ثابت در تمامی

تکرارها استفاده می‌شود. مورچه‌های ACO مثل ذرات PSO از

تجربیات همدیگر استفاده می‌کنند. استقلال انتخاب مقصد توسط

مورچه‌ها وجود ندارد و مورچه‌ها ملزم به استفاده از تجربیات

فردی همدیگر هستند. برای هر مورچه یک حافظه اختصاص

داده می‌شود که بهترین تجربه شخصی خود را در آن ذخیره کند

و حالا علاوه بر اثر فرومون مسیر، تجربیات بقیه مورچه‌ها (بهینه

فردی و سراسری) در به‌روزرسانی فرومون تأثیر داده می‌شود.

اعمال PSO برای بهینه‌سازی پارامترهای ACO می‌تواند تأثیر

محدودیت‌های الگوریتم ACO را تا حد زیادی کاهش دهد و در

نتایج بسیار مؤثر باشد [۲۲]. برای توضیح بیشتر روش پیشنهادی

گرافی با n ایستگاه با مختصات مختلف در نظر می‌گیریم. مطابق

شکل ۲ قبل از اقدام به مسیریابی پارامترهای ورودی مسئله

تعیین می‌شوند. ابتدا تعداد مورچه‌ها (خودروها) و تعداد

تکرارهای الگوریتم تعیین و با توجه به توضیحات ذکر شده در

بخش ۲-۲-۱، پارامترهای الگوریتم کلونی مورچگان شامل

میزان بزرگنمایی فرومون (α)، میزان بزرگنمایی هیورستیک (β)

و شدت تبخیر فرومون را ابتدا باید مقداردهی کرد. همچنین یک

ماتریس فرومون تشکیل و در هر مرحله از تکرار، میزان فرومون

هر مسیر تعیین می‌شود. با توجه به تأثیر دادن بهینه‌های فردی و

بهینه سراسری الگوریتم PSO در الگوریتم ACO این مقادیر

به‌عنوان دو پارامتر اساسی در پیدا کردن مسیر بهینه باید دارای

مقادیر اولیه مناسب باشند که با تأثیر دادن آن‌ها و اثر ترافیک

جاده‌ها به عنوان پارامترهای مؤثر در رابطه ۱ می‌توان اقدام به

تولید جواب‌های مناسب کرد.

$$c_2 r_2 (gbest - x_i(t))$$

در رابطه‌ی ۶، w وزنی مثبت برای تنظیم محدوده جستجو

سراسری است. c_1 فاکتور یادگیری مدل ادراکی و نشان‌دهنده

میزان یادگیری فردی در هنگام جستجو در فضا است و c_2

فاکتور یادگیری مدل جمعی و میزان یادگیری گروهی و به

اشتراک‌گذاری اطلاعات اعضا در فضا را نشان می‌دهد. r_1 و r_2

دو پارامتر با مقادیر تصادفی در بازه $[0, 1]$ هستند که از همگرایی

پارامترها ممانعت می‌کنند. $pbest_i$ بهترین موقعیت ذره i و

$gbest$ بهترین موقعیت سراسری را نمایش می‌دهد. در رابطه‌ی

۷، موقعیت جدید ذرات را با توجه به موقعیت فعلی $x_i(t)$ و

رابطه‌ی سرعت $v_i(t)$ به‌روزرسانی می‌کند.

۳- روش پیشنهادی

الگوریتم‌های فرا ابتکاری در حل مسئله‌ی مسیریابی نقش

بسیاری دارند. با توجه به خاصیت جستجوی هیورستیک در این

دسته از مسائل، الگوریتمی باید انتخاب شود که در شرایط

پیچیده و غیرقطعی که نیازمند همکاری و هوشمندی اعضاست،

بتواند پاسخگوی موضوع باشد. برای این منظور از ترکیب

الگوریتم کلونی مورچگان و الگوریتم ازدحام ذرات استفاده

می‌شود. در ادامه برای افزایش دقت الگوریتم‌های ذکر شده باید

به دنبال تابع ابتکاری بود که تعداد تکرارها را برای پیدا کردن

مسیر بهینه کاهش دهد. برای این هدف تابع ابتکاری کسینوس

زوایا به الگوریتم ترکیبی ACOPSO اضافه شده است. محدوده

تنظیم پارامتر الگوریتم کلونی مورچگان شامل سه پارامتر کلیدی

α ، β و ρ است. از آنجایی که دامنه این پارامترها گسترده است،

تنظیم آن‌ها چالش برانگیز است و انتخاب نامناسب پارامترها

منجر به افزایش زمان جستجو و کاهش دقت در پیدا کردن مسیر

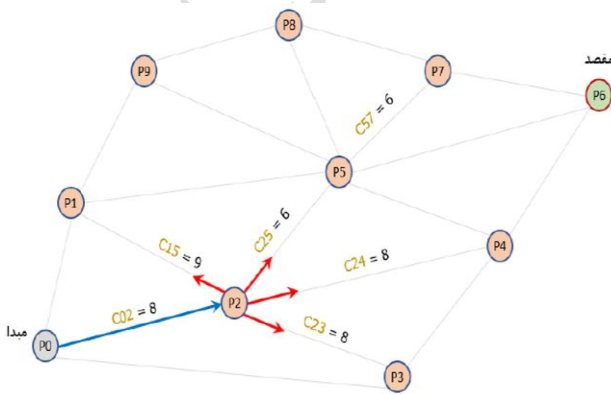
بهینه می‌شود. در [۶] از ترکیب ACO با PSO که دارای

شامل مینیمم امتیاز (score) مورچه‌ها در هر تکرار (k) است تعریف می‌شود.

و به ازای کل مورچه‌ها یک متغیر بهینه سراسری (gbest) مطابق با رابطه ۹ که بهترین بهینه فردی را در خود نگه می‌دارد تعریف می‌شود.

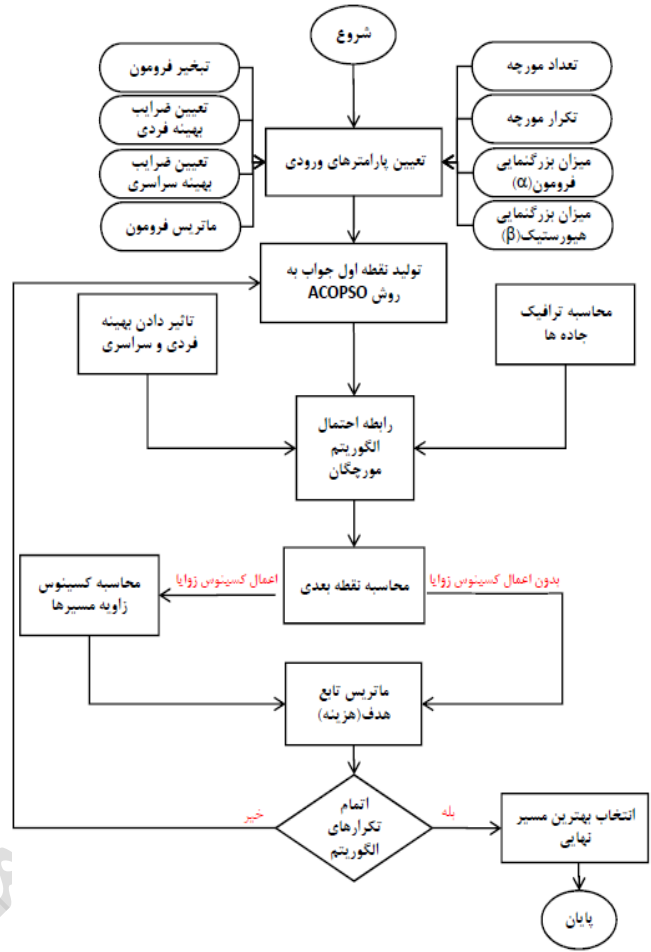
$$gbest = \min_{i \in \{1, \dots, m\}} p_b \quad (9)$$

برای به‌روزرسانی ترافیک، فرض کنید وضعیت ترافیک مطابق شکل ۳ است. که از نقطه مبدا p_0 بین مسیریابی که به $\{p_1, p_2, p_3\}$ وجود دارد نقطه p_2 که دارای ترافیک (C_{02}) ۸ است، به‌عنوان مقصد بعدی انتخاب شود. حال در نقطه p_2 از بین مسیرهای $\{p_3, p_4, p_5\}$ فرضاً مسیر p_5 برای ادامه مسیر انتخاب شود. بدین ترتیب مقدار ترافیک مسیر p_0 به p_2 یک واحد کم ($C_{02} = 7$) و مقدار ترافیک مسیر p_2 به p_5 یک واحد افزایش ($C_{25} = 7$) پیدا می‌کند. بعد از به‌روزرسانی ماتریس ترافیک، مقادیر آن را در ماتریس اصلی فرمون تأثیر داده می‌شود.



شکل (۳) - وضعیت ترافیکی فرضی یک منطقه شهری

در فرآیند مسیریابی که انجام می‌دهیم، برای به‌روزرسانی ماتریس فرمون $T[i, j]$ مطابق با رابطه ۱۰ علاوه بر فرمون روی



شکل (۲) - مراحل الگوریتم ACOPSO

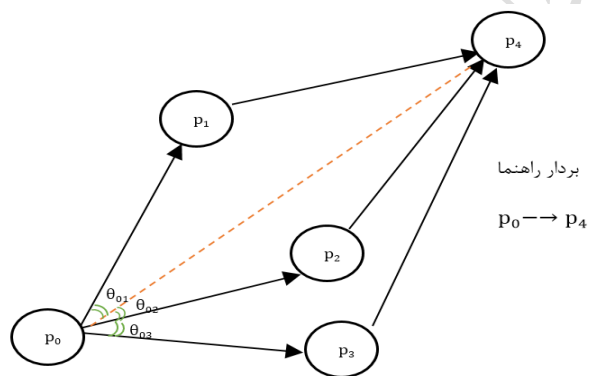
فرض کنید تعداد m مورچه روی مسیر هستند و الگوریتم‌ها K مرتبه تکرار می‌شوند. برای تولید مسیرهای تصادفی اولیه توسط مورچه‌ها به ازای فرمون و ترافیک اولیه ثابت ماتریسی به ابعاد $n * m$ ایجاد می‌شود. هر سطر نماینده یک مسیر مورچه و هر ستون نماینده یک ایستگاه است. مورچه‌ها اولین ایستگاه (مبدأ) را انتخاب می‌کنند. سپس به ازای هر تکرار به تعداد ایستگاه‌ها تکرار فرعی دارند. در این تکرارها مورچه‌ها قدم به قدم به سمت مقصد، ایستگاه‌ها را انتخاب می‌کنند. در تکرارهای فرعی مورچه اول تا مورچه m -ام یک ایستگاه را بین گره‌های $\{0, 1, \dots, m\}$ انتخاب می‌کنند. همچنین به ازای هر مورچه i یک متغیر بهینه فردی (pb_{est_i}) مطابق با رابطه ۸ که

مسیر، بهینه‌های فردی و سراسری مورچه‌ها نیز تأثیر داده می‌شوند.

خط‌چین نارنجی به عنوان بردار راهنما در نظر گرفته می‌شود و زاویه‌ی بین هر شهر و بردار راهنما مطابق رابطه $\cos(\theta) = \frac{\mathbf{u} \cdot \mathbf{v}}{|\mathbf{u}| |\mathbf{v}|}$ محاسبه می‌شود که \mathbf{u} و \mathbf{v} به ترتیب گرهِ مبدأ و گرهِ

مقصد هستند. به این ترتیب در مرحله اول مسیریابی زاویه بین $T[i, j] = T[i, j] + c_1 * \sum_{l=1}^m pbest_l + c_2 * (gbest)$

بردار راهنما و مسیرهای $(\theta_{01}) p_0 p_1$ ، $(\theta_{02}) p_0 p_2$ و $(\theta_{03}) p_0 p_3$ محاسبه می‌شود و مسیر دارای زاویه کمتر شانس بیشتری برای انتخاب می‌گیرد. این رابطه با توجه به ویژگی‌های جغرافیایی مسئله عمل می‌کند و می‌تواند تصمیم‌گیری‌هایی متناسب با شرایط را انجام دهد. در واقع در هر مرحله کسینوس زاویه‌ی بین گرهِ مقصد و همه‌ی گرهِها محاسبه و در ماتریس احتمال الگوریتم کلونی مورچگان تأثیر داده می‌شود. با این کار مسیرها و گرهِهای نامربوط بیشتری از دایره‌ی محاسبات خارج می‌شوند و احتمال انتخاب مسیر دارای کمترین زاویه با مقصد، افزایش می‌یابد. در ادامه نحوه‌ی عملکرد تابع ابتکاری کسینوس زوایا را در قالب یک مثال بررسی خواهیم کرد.



شکل (۴) - محاسبه‌ی کسینوس زوایا بین نقاط مبدأ و مقصد

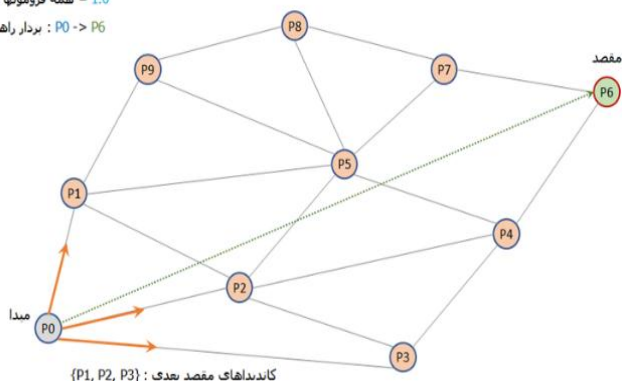
مطابق شکل ۵ ده گرهِ فرضی با مختصات متفاوت را از یک منطقه شهری در نظر می‌گیریم. بین تمامی گرهِها مسیریابی با طول متفاوت وجود دارد. در ابتدا باید مقدار فرمون روی مسیر مشخص شود. برای این منظور، مقدار همه فرمون‌های روی مسیر را برابر عدد یک قرار می‌دهیم. مسیریابی موردنظر از یک مبدأ به یک مقصد مشخص صورت می‌گیرد. مطابق شکل ۵ ما

در ادامه برای افزایش دقت و کاهش زمان مسیریابی، تابع ابتکاری کسینوس زوایا اعمال می‌شود. در مسیریابی که از یک مبدأ به یک مقصد مشخص صورت می‌گیرد، همیشه مقصد نهایی معلوم است و جهت بین مبدأ و مقصد به عنوان یک بردار راهنما در نظر گرفته می‌شود. اگر مقصد نهایی مشخص نباشد، محاسبه کسینوس زوایا امتیازی محسوب نمی‌شود. ACO مراحل اولیه را کاملاً تصادفی و بدون نظم خاصی جستجو می‌کند و هرچه جلوتر می‌رود و تعداد تکرارهای بیشتری انجام می‌شود، انتخاب مسیرها از حالت تصادفی خارج شده و هدفمند می‌شود. چون مراحل اولیه کاملاً تصادفی است، می‌توان از خاصیت کسینوس زوایا که فقط روی نقاطی تأکید می‌کند که هم راستای مسیر پیمایش به سمت مقصد نهایی است، استفاده کرد تا دقت الگوریتم ACOPSO برای انتخاب مسیرها در ابتدای مسیریابی بالا برود و همچنین تعداد تکرارها برای پیدا کردن مسیر بهینه کاهش یابد. هرچه زاویه بین شهر موردنظر و بردار راهنما بزرگ‌تر باشد، مقدار کسینوس به سمت صفر میل می‌کند و احتمال انتخاب آن کمتر می‌شود. اگر زاویه کمتر باشد، مقدار کسینوس به سمت یک میل می‌کند و احتمال انتخاب آن بالا می‌رود. در مسیریابی بر اساس الگوریتم ACO، هنگام حرکت از مبدأ به مقصد، با افزایش طول مسیر و بعد از گذشت تکرارهای طولانی تفاوت بین مسیر کوتاه و مسیر بهینه کاهش می‌یابد. ولی این تفاوت ممکن است بعد از گذشت زمان زیادی به صفر برسد. بدین منظور برای افزایش دقت محاسبات در مراحل ابتدایی و همچنین برای کاهش زمان جستجو در مسیریابی، تابع ابتکاری کسینوس زوایا را با الگوریتم تلفیقی ACOPSO ترکیب می‌کنیم. در تابع ابتکاری ACOPSOca، مطابق شکل ۴،

شهر p_2 و p_5 و بردار راهنما نسبت به سایر مسیرها کمتر است و کسینوس زاویه‌ی آن دارای مقدار بزرگ‌تری است، احتمال انتخاب شهر p_5 به‌عنوان مقصد بعدی نسبت به سایر شهرها بیشتر است. فرض می‌کنیم شهر p_5 به‌عنوان مقصد بعدی انتخاب شود. در ادامه کاندیدای انتخاب مقصد بعدی $\{p_4, p_7, p_8, p_9\}$ است که با توجه به محاسبات کسینوس زاویه، احتمال انتخاب شهر p_7 نسبت به سایر شهرها بیشتر است. فرض می‌کنیم شهر p_7 به‌عنوان مقصد بعدی انتخاب شود. در نهایت در شهر p_7 کاندیدای مقصد بعدی شهرهای $\{p_8, p_6\}$ است که با توجه به محاسبات کسینوس زاویه، احتمال انتخاب شهر p_6 به‌عنوان مقصد نهایی بیشتر از سایر شهرها است.

در ادامه، مراحل الگوریتم ACOPSOca مطابق شکل ۲ طی می‌شود. سرانجام، بعد از اتمام تعداد تکرارهای تعیین شده الگوریتم، بر اساس ماتریس تابع هدف (هزینه) که در هر گام بروزرسانی شده، بهترین مسیر نهایی انتخاب شده و فرآیند مسیریابی به پایان می‌رسد.

$t_0 = 1.0$ همه فرمونها
 $P_0 \rightarrow P_6$: بردار راهنما

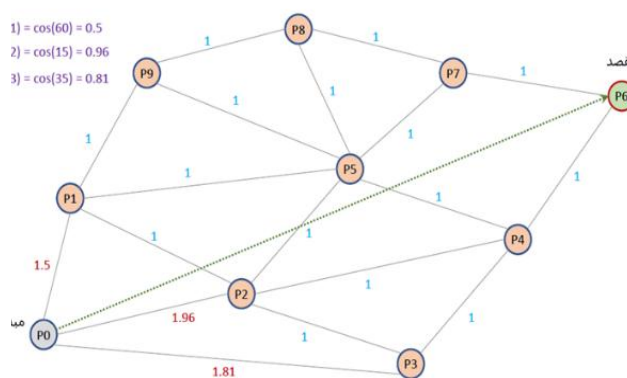


شکل (۵) - کاندیدای مقصد بعدی هر مورچه به کمک کسینوس زوایا

می‌توانیم از این مورد استفاده کنیم و یک بردار راهنما داشته باشیم و در مسیریابی که در هر مرحله انجام می‌دهیم بردار راهنما را مدنظر قرار دهیم. در ادامه فرض می‌کنیم p_0 شهر مبدأیی است که در آن قرار داریم و قرار است به شهر p_6 به‌عنوان شهر مقصد برویم. کاندیدای مقصد بعدی p_0 شهرهای $\{p_1, p_2, p_3\}$ است که برای انتخاب شهر بعدی کسینوس زاویه بین شهرهای مقصد و شهرهای کاندید محاسبه می‌شود. فرض کنید کسینوس زاویه بین بردار راهنما و مسیر p_0, p_1 60° درجه که $\cos 60 = 0.5$ ، کسینوس زاویه بین بردار راهنما و مسیر p_0, p_2 15° درجه که $\cos 15 = 0.96$ و کسینوس زاویه بین بردار راهنما و مسیر p_0, p_3 35° درجه است که $\cos 35 = 0.81$ می‌باشد.

بعد از مشخص شدن مسیرهای ارتباطی بین شهر p_0 و سایر شهرها، مطابق شکل ۶، مقدار کسینوس به دست آمده از هر مسیر با فرمون روی مسیر جمع می‌شود و در ماتریس تأثیر داده می‌شود. حال مورچه (خودرو روی مسیر) بر اساس ماتریس محاسبه شده که شامل تأثیر فرمون مسیر، حجم ترافیک مسیر و مقدار کسینوس زوایا است، عمل می‌کند و شهری را به‌عنوان شهر بعدی انتخاب می‌کند که دارای مقدار نهایی بزرگ‌تری است. بنابراین، مسیر بین شهر p_0 به شهر p_2 با مقدار کسینوس 1.96 ، احتمال بیشتری برای انتخاب به‌عنوان مقصد بعدی دارد ($p_2 = 1.96 > p_3 = 1.81 > p_1 = 1.5$). نکته حائز اهمیت این است که لزوماً مقدار کسینوس بزرگ‌تر به معنای انتخاب همان مسیر نیست بلکه مقدار کسینوس بزرگ‌تر یعنی شانس بیشتری برای انتخاب مسیر متناظر وجود دارد. حال فرض می‌کنیم مقدار کسینوس زاویه‌ی مسیر شهر p_2 در ماتریس احتمالات تأثیر داده شده و به‌عنوان شهر بعدی انتخاب شده است. بعد از رسیدن به شهر p_2 کاندیدای انتخاب مقصد بعدی شهرهای $\{p_1, p_3, p_4, p_5\}$ است. با توجه به اینکه زاویه‌ی بین

این منظور دوازده ارزیابی مبتنی بر فاصله‌ی فیزیکی طراحی شده که در هر ارزیابی روش پیشنهادی با نتایج روش‌های ACOPSO [۱]، ICMPACO [۲۴]، ACSGA [۲۵]، DSOS [۲۶]، DPC – ACO – KOPT [۲۷]، EHO [۲۸]، SOS – SA [۲۹]، DA [۳۰]، GSACO [۱۴]، BA [۱۸] مقایسه شده و نتایج حاصل با توجه به داده‌های موجود در دو فاز اصلی ارزیابی می‌شوند.



شکل (۶) - اضافه کردن مقادیر کسینوس زاویه به فرمون روی مسیر

۱-۳- پیچیدگی محاسباتی روش پیشنهادی

در فاز اول، نرخ انحراف (دقت الگوریتم) و در فاز دوم میانگین تعداد تکرار (زمان محاسبات)، برای رسیدن به بهترین جواب بررسی می‌گردد. در ارزیابی یازدهم، بر روی نقاط شلوغ و دارای ترافیک سنگین شهر پکن ارزیابی دیگری انجام خواهیم داد [۱]. در ارزیابی دوازدهم، برای بررسی تأثیر تابع کسینوس زوایا بر روی الگوریتم ACOPSO، بر اساس [۶] مسیرهایی از گراف شبکه‌ای شهر پکن انتخاب می‌شود و هزینه زمان سفر بر روی هر مسیر محاسبه می‌شود. در این آزمایش‌ها چهار معیار مهم برای ارزیابی وجود دارد که Mean iter میانگین تعداد تکرار برای به دست آوردن بهترین نتیجه‌ی نهایی، Mean best میانگین بهترین نتیجه‌ی راه‌حل، best بهترین نتیجه از میان نتایج

در این مقاله مقرون به صرفه بودن الگوریتم ACO با تعداد مورچه‌ها، تقاطع‌ها و زمان‌های تکرار رابطه خطی دارد. وقتی زمان تکرار به مقدار معین از پیش تعیین شده برسد و پیشرفتی در جواب‌های بهینه حاصل نشود الگوریتم از اجرا صرف‌نظر می‌کند. بنابراین پیچیدگی محاسبات را می‌توان به صورت $O(n * (n - 1)^2 * m * \frac{T}{2})$ بیان کرد که n تعداد تقاطع‌ها و m تعداد مورچه‌ها را نشان می‌دهد و T نشان‌دهنده زمان تکرار است. در آزمایش صورت گرفته زمان اجرا به زمان مصرف‌شده از نقطه شروع تا نقطه پایان یا زمان موردنیاز برای ایجاد بهترین مسیر در تعداد تکرارها اشاره دارد. از بیان پیچیدگی محاسبات به راحتی می‌توان مشاهده کرد که وقتی تعداد مورچه‌ها و تعداد تقاطع‌ها ثابت است، پیچیدگی محاسبه و زمان اجرا با زمان تکرار رابطه مثبت دارد. به عبارت دیگر، هرچه تعداد تکرارها بیشتر باشد، پیچیدگی محاسباتی بیشتر و به زمان بیشتری نیاز است.

۴- ارزیابی

برای ارزیابی رویکرد پیشنهادی، نتایج حاصل از اجرای الگوریتم ACOPSO با جدیدترین و رایج‌ترین روش‌ها بر روی داده‌های کتابخانه معروف TSPLIB [۲۳] مقایسه می‌شود. برای

³ multi-population strategy

⁴ Ant colony system & Genetic

⁵ Discrete Symbiotic Organisms Search

⁶ Density Peaks Clustering & Ant colony optimization

^۷ Elephant herding optimization

⁸ Symbiotic Organisms Search & Simulated Annealing

⁹ Dragonfly algorithm

¹⁰ Best Algorithm

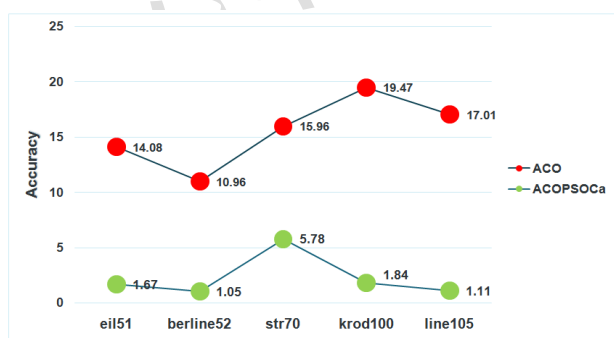
¹¹ Ant colony optimization algorithm based on adaptive strategy (۱۱)

ρ, β, α قبل از تکرار تعیین می‌شود و در طول فرایند تکرار تغییر نمی‌کند. در این آزمایش‌ها، برای انتخاب پارامترها طبق نتیجه‌گیری به دست آمده در [۳۱]، محدوده‌ی پارامترها ρ, β, α طبق جدول ۲ تنظیم می‌شود.

جدول (۲) - محدوده پارامترهای ارزیابی اول

آزمایش	تعداد دفعات اجرا	حداکثر تعداد تکرارها	عدم تغییر راه حل مطلوب	α	β	ρ
آزمایش اول	100	1000	200	0.1-0.3	3-6	0.7-0.9
آزمایش دوم	100	1000	200	0.5-1.5	1-5	0.5-1
آزمایش سوم	100	1000	200	0.5-1.5	1-5	0.5-1

نرخ انحراف در دیتاست‌های مختلف به‌طور متوسط در آزمایش اول مطابق شکل ۷، ۸۵٫۲۸٪، در آزمایش دوم مطابق شکل ۹، ۸۱٫۵۴٪ و در آزمایش سوم مطابق شکل ۱۱، ۸۹٫۸۹٪ است که دقت روش ACOPSOCa به نسبت روش ACO و ACOPSO بیشتر بوده است. روش ACOPSOCa مطابق شکل ۸ در کاهش تعداد تکرارها در آزمایش اول نسبت به روش ACO متوسط ۱۱٫۵۸٪، در آزمایش دوم مطابق شکل ۱۰، ۱۰٫۸۹٪ و در آزمایش سوم مطابق شکل ۱۲، ۸٫۲۱٪ بهبودی داشته است.



شکل (۷) - مقایسه‌ی نرخ انحراف (دقت) الگوریتم‌ها در آزمایش اول

Mean best و dev_{rate} یک نرخ انحراف برای مقایسه دقت الگوریتم‌ها است که مطابق رابطه ۱۱ محاسبه می‌شود.

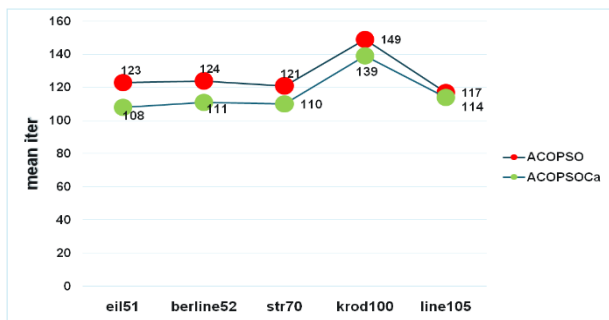
در این رابطه، $result_{avr}$ میانگین حاصل از بهترین نتایج به دست آمده و $result_{best}$ بهترین نتیجه از میان نتایج به دست آمده است. در این رابطه هرچه مقدار نرخ انحراف پایین‌تر باشد نشان‌دهنده‌ی عملکرد صحیح و هدفمند الگوریتم است و هرچه مقدار آن بالاتر باشد، عملکرد نامنظم الگوریتم را نشان می‌دهد.

*توجه: در ارزیابی‌های صورت گرفته در هر نمودار، به ازای هر دیتاست روش پیشنهادی مقاله با روش‌های سایر مقالات در دو فاز نرخ انحراف Accuracy و تعداد تکرارها Mean iter مقایسه شده است. در توضیحات هر نمودار، میانگین تمام حالت‌ها بصورت درصد بیان شده است.

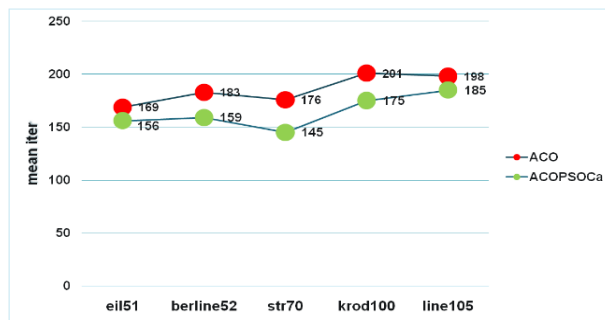
*توجه: علاقمندان می‌توانند به دیتاست‌های استفاده شده و کدهای شبیه‌سازی مقاله در [۱۹] دسترسی داشته باشند.

۴-۱- ارزیابی اول

در [۴]، آزمایشی برای بهینه‌سازی پارامترهای ACO بر اساس PSO طراحی شده است. خود این ارزیابی شامل سه نوع آزمایش متفاوت است که دو آزمایش برای انتخاب تصادفی پارامترهای ACO و یک آزمایش برای بهینه‌سازی پارامترها ACO در نظر گرفته شده است. در آزمایش انتخاب تصادفی پارامترهای ACO، مقادیر ρ, β, α به‌طور تصادفی در محدوده‌ای تنظیم می‌شوند و مقادیر سه پارامتر با هر تکرار تغییر می‌کند. در آزمایش بهینه‌سازی پارامترهای ACO، مقادیر سه پارامتر



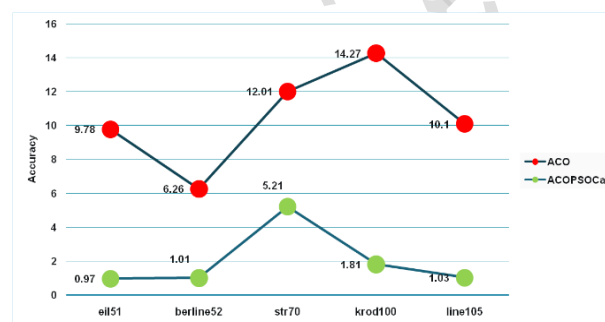
شکل (۱۲) - مقایسه‌ی میانگین تعداد تکرار در آزمایش سوم



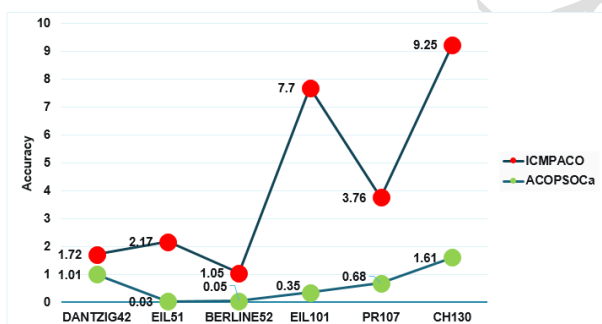
شکل (۸) - مقایسه‌ی میانگین تعداد تکرار در آزمایش اول

۲-۴- ارزیابی دوم

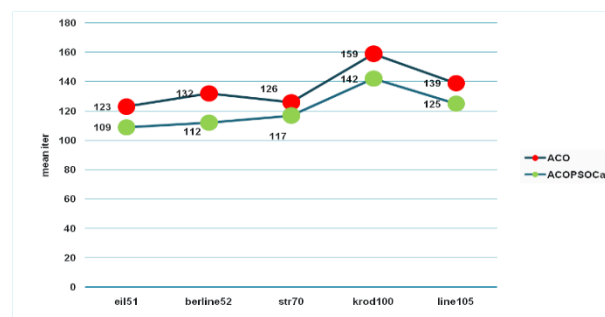
در [۲۴]، یک الگوریتم بهبودیافته‌ی کلونی مورچگان بر اساس استراتژی چند جمعیتی برای بهبود عملکرد مسیریابی پیشنهاد شده است. مقایسه نرخ انحراف با توجه به شکل ۱۳ نشان می‌دهد در دیتاست‌های مختلف به جز ch130 به طور متوسط ۶۲،۱۶٪ دقت روش ACOPSOca به نسبت روش ICMPACO بیشتر بوده است.



شکل (۹) - مقایسه‌ی نرخ انحراف (دقت) الگوریتم‌ها در آزمایش دوم



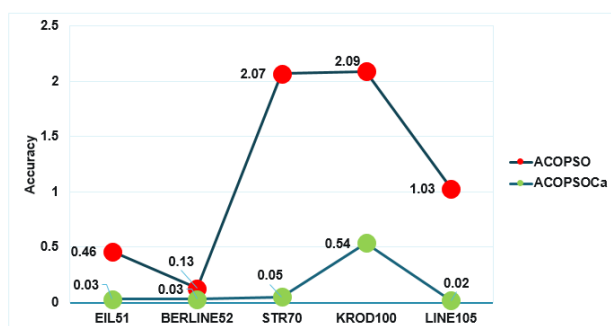
شکل (۱۳) - مقایسه‌ی نرخ انحراف الگوریتم‌ها در ارزیابی دوم



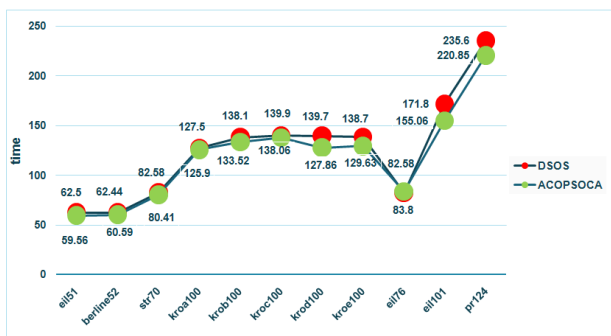
شکل (۱۰) - مقایسه‌ی میانگین تعداد تکرار در آزمایش دوم

۳-۴- ارزیابی سوم

در [۲۵] ترکیب الگوریتم کلونی مورچگان و الگوریتم ژنتیک برای بهبود عملکرد مسیریابی پیشنهاد شده است. مقایسه نرخ انحراف با توجه به شکل ۱۴ نشان می‌دهد در دیتاست‌های مختلف دقت روش ACOPSOca به نسبت روش ACSGA به طور متوسط ۲۷،۶۶٪ بیشتر بوده است.



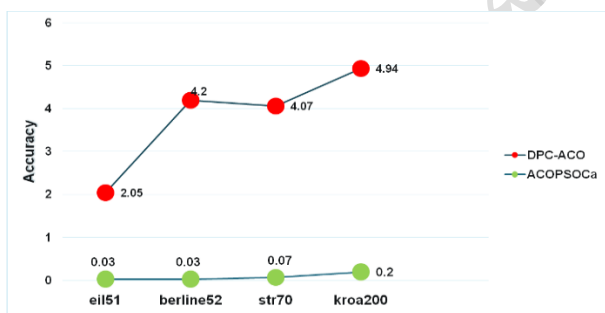
شکل (۱۱) - مقایسه‌ی نرخ انحراف (دقت) الگوریتم‌ها در آزمایش سوم



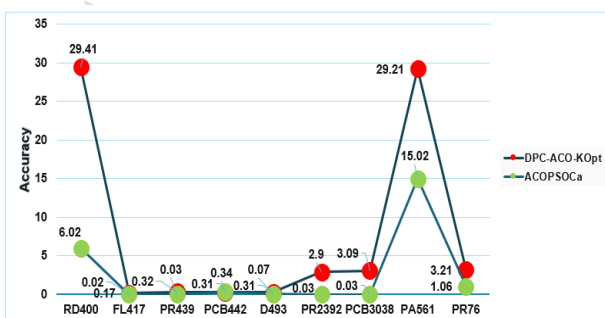
شکل (۱۶)- مقایسه زمان محاسبات الگوریتم‌ها در ارزیابی چهارم

۴-۵- ارزیابی پنجم

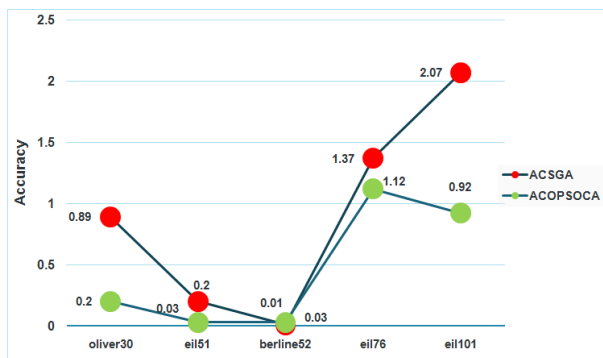
یک الگوریتم ترکیبی با توجه به ایده‌ی تقسیم و غلبه برای مسیریابی در [۲۷] پیشنهاد شده است. مقایسه‌ی نرخ انحراف نشان می‌دهد در دیتاست‌های مختلف در آزمایش اول با توجه به شکل ۱۷ به‌طور متوسط ۹۸,۰۱٪ و در آزمایش دوم مطابق با شکل ۱۸، با درصد بسیار بالا دقت روش ACOPSOCA به نسبت روش DPC - ACO - KOPT در انواع مقیاس پارامترها بیشتر بوده است.



شکل (۱۷)- مقایسه‌ی نرخ انحراف الگوریتم‌ها در ارزیابی پنجم



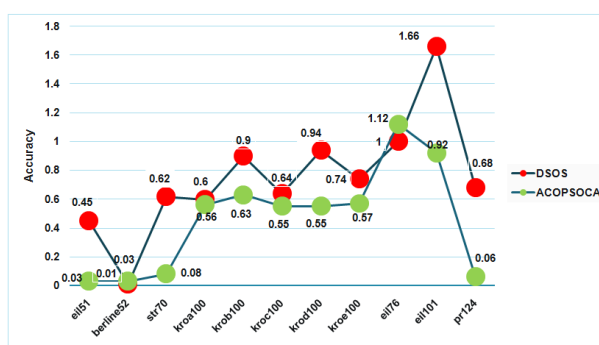
شکل (۱۸)- مقایسه‌ی نرخ انحراف الگوریتم‌ها در ارزیابی پنجم



شکل (۱۴)- مقایسه نرخ انحراف الگوریتم‌ها در ارزیابی سوم

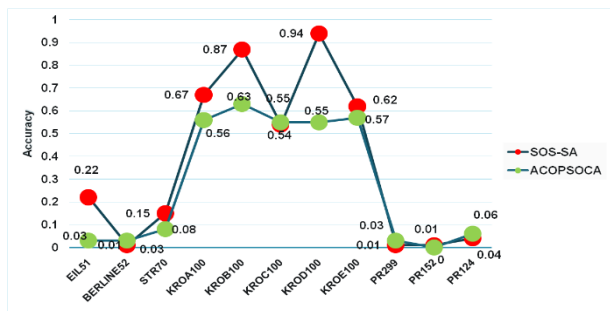
۴-۴- ارزیابی چهارم

در [۲۶]، یک الگوریتم جستجوی موجودات همزیست گسترده DSOS برای یافتن راه‌حل مناسب برای مسائل مسیریابی پیشنهاد شده است. الگوریتم DSOS یک الگوریتم فرا هیوریستیک طبق اکوسیستم بقا و تکثیر است. مقایسه نرخ انحراف با توجه به شکل ۱۵ نشان می‌دهد در دیتاست‌های مختلف طور متوسط ۴۶,۶٪ دقت روش ACOPSOCA به نسبت روش DSOS بیشتر بوده است. در نمودار ۱۶ مقایسه زمان محاسبات برای به دست آوردن مسیر بهینه را نشان می‌دهد. روش ACOPSOCA در کاهش زمان محاسبات نسبت به روش DSOS به‌طور متوسط ۴,۱۶٪ بهبودی داشته است.



شکل (۱۵)- مقایسه نرخ انحراف الگوریتم‌ها در ارزیابی چهارم

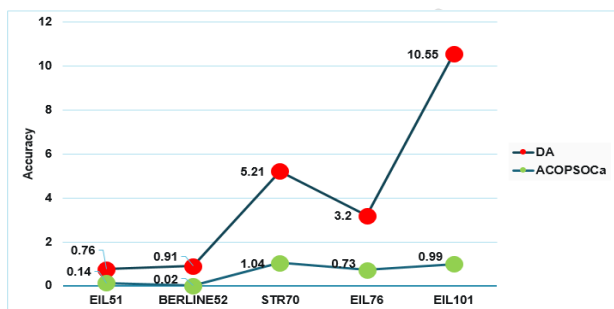
۴-۶- ارزیابی ششم



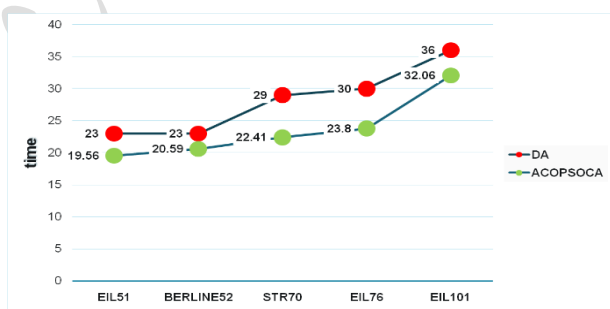
شکل (۲۰) - مقایسه نرخ انحراف الگوریتم‌ها در ارزیابی هفتم

۴-۸- ارزیابی هشتم

در [۳۰]، برای بهینه‌سازی مسیر از الگوریتم فرا اکتشافی سنجاک استفاده شده است. مقایسه نرخ انحراف با توجه به نتایج شکل ۲۱ نشان می‌دهد در دیتاست‌های مختلف به‌طور متوسط ۶۳،۳۴٪ دقت روش ACOPSOCA به نسبت روش DA بیشتر بوده است. شکل ۲۲ زمان محاسبات برای به دست آوردن مسیر بهینه را نشان می‌دهد. روش ACOPSOCA در کاهش زمان محاسبات نسبت به روش DA به طور متوسط ۱۵،۹۵٪ بهبودی داشته است.

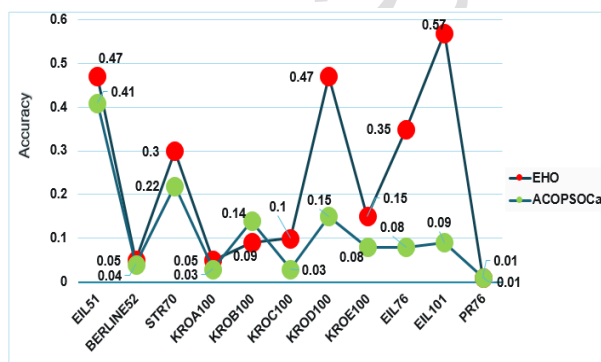


شکل (۲۱) - مقایسه‌ی نرخ انحراف در ارزیابی هشتم



شکل (۲۲) - مقایسه‌ی زمان محاسبات در ارزیابی هشتم

در [۲۸]، برای بهینه‌سازی مسیر از الگوریتم گله‌ی فیل که از رفتار طبیعی فیل‌ها الهام گرفته شده، استفاده می‌شود. مقایسه نرخ انحراف با توجه به نمودار ۱۹ نشان می‌دهد روش اجرایی ACOPSOCA عملکرد بهتری داشته است و در مجموع به‌طور متوسط ۱۱،۶۳٪ دقت روش ACOPSOCA به نسبت روش Improved Discrete EHO بیشتر بوده است.

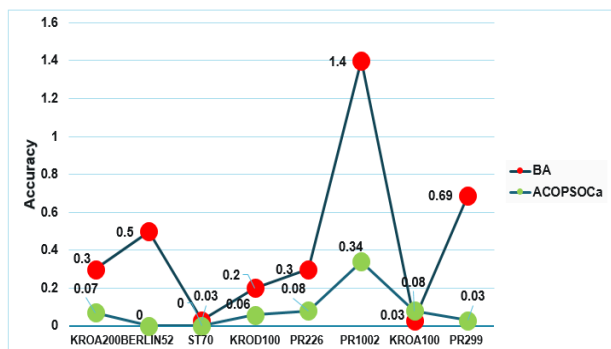


شکل (۱۹) - مقایسه نرخ انحراف الگوریتم‌ها در ارزیابی ششم

۴-۷- ارزیابی هفتم

در [۲۹]، الگوریتم جست‌وجوی موجودات همزیست با شبیه‌سازی تبرید ترکیب شده و روش جدیدی با عنوان SOS - SA معرفی شده است. مقایسه نرخ انحراف با توجه به شکل ۲۰ نشان می‌دهد در دیتاست‌های berline52، kroc100، Pr124، Pr299 روش SOS - SA نسبت به روش اجرایی ACOPSOCA عملکرد بهتری داشته است ولی در مجموع به‌طور متوسط ۷،۵۹٪ دقت روش ACOPSOCA به نسبت روش SOS - SA بیشتر بوده است.

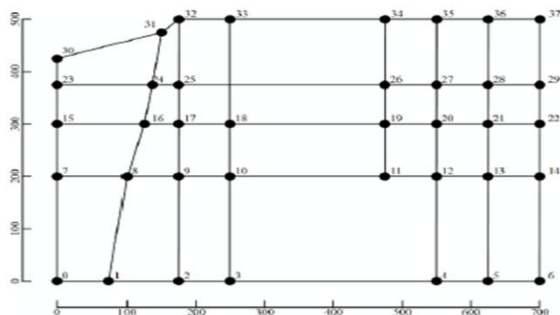
۴-۹- ارزیابی نهم



شکل (۲۴) - مقایسه‌ی نرخ انحراف در ارزیابی هفتم

۴-۷- ارزیابی یازدهم

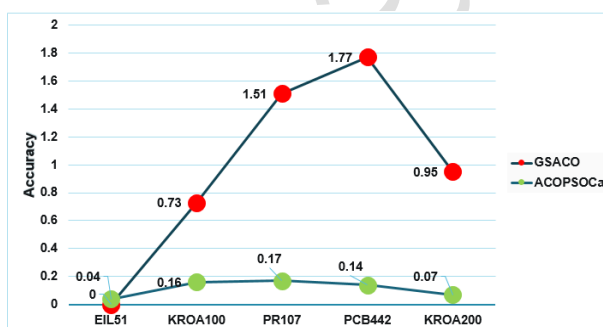
به منظور بررسی اثربخشی روش برنامه‌ریزی مسیر پویا بر اساس الگوریتم ACOPSOCa از شبکه‌های ترافیکی انتخاب شده از یک منطقه شهری برای شبیه‌سازی استفاده می‌شود. در بین شهرهای بزرگ چین، شبکه جاده‌ای پکن بالاترین درصد ازدحام را دارد. بنابراین مطابق شکل ۲۵ برخی از تقاطع‌های شهر پکن به‌عنوان ناحیه‌ای برای آزمایش برنامه‌ریزی مسیر پویا بر اساس الگوریتم بهبود یافته‌ی کلونی مورچگان انتخاب می‌شود [۳۲].



شکل (۲۵) - شبکه‌ی شهری پکن

در این آزمایش، ابتدا سه پارامتر الگوریتم ACO، به صورت $\alpha = 1$ (۱۶) و $\rho = 0.8$ تنظیم می‌شوند. همچنین نقاط شروع و پایان به‌طور تصادفی انتخاب می‌شوند. در آزمایش برنامه‌ریزی مس (۱۷) بر اساس الگوریتم ACOPSOCa در شرایط ترافیک جاده‌ای ابتدا مسیر اولیه $Tour_{int}$ ، مسیر ایستا $Tour_{static}$ ، مسیر پویای $Tour_{Aco}$ و مسیر پویا $Tour_{opt}$ را تعریف

در [۱۴]، برای بهینه‌سازی مسیر از الگوریتم ترکیبی بهینه‌سازی کلونی مورچه بر اساس استراتژی حریص تطبیقی (GSACO) استفاده شده است. مقایسه نرخ انحراف (دقت) با توجه به نتایج شکل ۲۳ نشان می‌دهد در دیتاست‌های مختلف به‌طور متوسط GSACO به نسبت روش ACOPSOCa به نسبت روش GSACO بیشتر بوده است.



شکل (۲۳) - مقایسه‌ی نرخ انحراف در ارزیابی نهم

۴-۱۰- ارزیابی دهم

در [۱۸]، برای بهینه‌سازی مسیر نسخه جدیدی از الگوریتم بهینه‌سازی زنبور عسل با جایگزینی تابع Multi-Insert بجای Swap ارائه شد. مقایسه نرخ انحراف (دقت) با توجه به نتایج شکل ۲۴ نشان می‌دهد در دیتاست KroA100 عملکرد الگوریتم BA بهتر بوده است و در سایر دیتاست عملکرد ACOPSOCa برتری دارد. در مجموع به‌طور متوسط 6.0% دقت روش ACOPSOCa به نسبت روش BA بیشتر بوده است.

جدول (۳) - نتایج مسیر پویای شهر پکن با روش ACOPSO

n	Tour _{int}	Tour _{static}	Tour _{aco}	Tour _{opt}	CR _{STATIC}	CR _{ACO}	CR _{OPT}
100	1699	1950	2024	1727	14.8	19.12	1.64
200	1742	1998	2070	1800	14.7	18.8	3.32
500	1753	2037	2039	1791	16.2	16.31	2.16
1000	1758	2042	2080	1784	16.2	18.3	1.47
1406	1723	2003	2041	1754	16.3	18.5	1.79
Avg	1735	2006	2050	1771	15.61	18.2	2.08

می‌کنیم. مسیر اولیه Tour_{int} نشان‌دهنده‌ی شرایط جاده بدون ترافیک است که توسط الگوریتم ACOPSO [۶] محاسبه می‌شود. مسیر ایستا Tour_{static} نشان‌دهنده‌ی مجموع عوامل شرایط جاده که توسط الگوریتم ACO به دست می‌آید. مسیر پویا Tour_{aco} نشان‌دهنده‌ی مجموع عوامل شرایط جاده است که توسط الگوریتم ACOPSO ذکر شده در مقاله به دست می‌آید. مسیر پویا Tour_{opt} نشان‌دهنده‌ی مجموع عوامل شرایط جاده است که با روش برنامه‌ریزی مسیر پویا بر اساس الگوریتم ACOPSOca به دست می‌آید. مجموع عوامل وضعیت جاده در مسیر اولیه Tour_{int} ثابت و در Tour_{static}، Tour_{aco}، Tour_{opt} با توجه به روابط ۱۲ تا ۱۵ به دست می‌آید.

$$\text{Tour}_{int} = \frac{\sum_{i=0}^n \text{Tour}_{int}(i)}{n} \quad (13)$$

از نتایج تجربی جدول ۴ می‌توان به راحتی دریافت که وقتی ضریب تراکم ترافیک از وضعیت ترافیکی شکل ۲۵ تبعیت می‌کند، شاخص ارزیابی CR_{STATIC}، به‌طور متوسط منجر به نرخ (۱۲) - - ۱۵,۲۴، شاخص ارزیابی CR_{ACO}، منجر به نرخ

ازدحام ۱۷,۴۲ و شاخص ارزیابی CR_{OPT} منجر به نرخ ازدحام

۱,۵۷ می‌شود. حال آنکه با توجه به جدول ۳ روش برنامه‌ریزی

با روش ACOPSOca و روش ACOPSO [۶]، را در نمودار

۲۶ می‌توان مشاهده کرد.

جدول (۴) - نتایج مسیر پویای شهر پکن با روش ACOPSOca

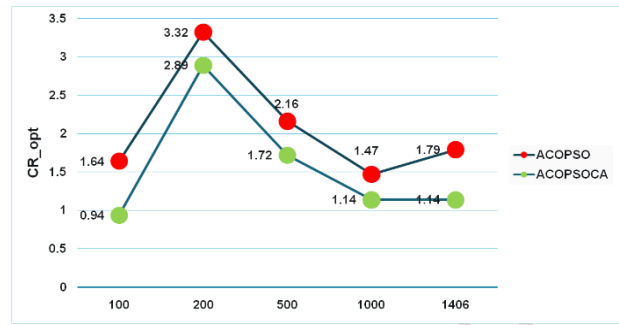
n	Tour _{int}	Tour _{static}	Tour _{aco}	Tour _{opt}	CR _{STATIC}	CR _{ACO}	CR _{OPT}
100	1704	1921	2001	1720	12.7	17.4	0.94
200	1730	1973	2043	1780	14.0	18.1	2.89
500	1743	2022	2026	1773	16.0	16.2	1.72
1000	1749	2040	2071	1769	16.6	18.4	1.14
1406	1753	2046	2050	1773	16.7	16.9	1.14
Avg	1735	2000	2038	1763	15.24	17.42	1.57

$$CR_{ACO} = \frac{\text{Tour}_{aco} - \text{Tour}_{int}}{\text{Tour}_{int}}$$

$$CR_{OPT} = \frac{|\text{Tour}_{opt} - \text{Tour}_{int}|}{\text{Tour}_{int}}$$

شاخص‌های ارزیابی آزمایش‌های برنامه‌ریزی مسیر پویا در شرایط ترافیک جاده‌ها CR_{STATIC}، CR_{ACO} و CR_{OPT} است. CR_{STATIC} میزان تراکم ایجاد شده توسط الگوریتم ACO، CR_{ACO} میزان تراکم ایجاد شده توسط الگوریتم ACOPSO و CR_{OPT} میزان ازدحام ایجاد شده توسط الگوریتم ACOPSOca است. نحوه محاسبات شاخص‌ها بر اساس روابط ۱۶ تا ۱۸ نشان داده شده است.

ACOPSOCa	ACOPSO	ACO	Dest	Start
2.46	2.85	5.46	14	24
2.15	2.36	4.32	6	16
2.51	2.76	4.7	32	12
2.62	3.18	5.99	AVE	



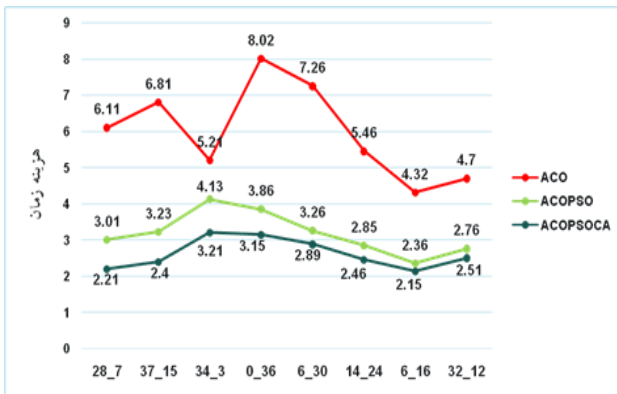
شکل (۲۶) - مقایسه نرخ ازدحام ACOPSO و ACOPSOCa

۴-۸- ارزیابی دوازدهم

در این قسمت به منظور بررسی تأثیر اعمال کسینوس زوایا در الگوریتم ACOPSO، بر روی شبکه شهری شکل ۲۷ آزمایشی به صورت مجزا بر روی یال‌های مشخص با پارامترهای $\alpha = 1$ ، $\beta = 3$ و $\rho = 0.1$ و تعداد تکرار ۵۰ به ازای الگوریتم ACOPSO، ACOPSOCa، ACOPSO انجام داده‌ایم. نتایج حاصل از این آزمایش در جدول ۵ نمایش داده شده است. با بهره‌گیری از مقایسه هزینه‌ای (هزینه‌ی زمان) با توجه به نمودار ۲۷ می‌توان مشاهده کرد الگوریتم ترکیبی ACOPSOCa به‌طور میانگین بر روی مسیرهای مختلف ۲،۶۲ ثانیه زمان مصرف می‌کند درحالی‌که الگوریتم ACOPSO، ۳،۱۸ و الگوریتم ACO، ۵،۹۹ ثانیه زمان می‌برد تا مسیرهای مختلف را طی کند. از بررسی نتایج زمان طی شده بر روی مسیرهای مختلف می‌توان به این نتیجه رسید با وجود ترافیک روی مسیرها، دقت الگوریتم ACOPSOCa نسبت به سایر روش‌ها بهتر است.

جدول (۵) - هزینه‌های سفر بر روی مسیرهای متفاوت شهر پکن

ACOPSOCa	ACOPSO	ACO	Dest	(۱۹)
2.21	3.01	6.11	28	7
2.4	3.23	6.81	37	15
3.21	4.13	5.21	30	-
3.15	3.86	8.02	0	30
2.89	3.26	7.26	6	30



شکل (۲۷) - مقایسه زمان محاسبات مسیرهای متفاوت شهر پکن

۴-۹- آزمون‌های آماری

در آزمون‌های آماری هدف بررسی یک حدس یا فرضیه بر اساس داده‌های نمونه است. برای مقایسه نرخ انحراف و میانگین تعداد تکرارها با توجه به جواب‌های بدست آمده در بخش‌های قبلی آزمون آماری را مطابق رابطه ۱۹ انجام داده‌ایم [۳۳]. که در این رابطه منظور از dev_{base} نرخ انحراف الگوریتم‌های پایه است که در بخش‌های قبلی با روش اجرایی مقاله بررسی کردیم. همچنین منظور از $dev_{ACOPSOCa}$ میزان نرخ انحراف روش اجرایی مقاله است.

یا بطور معادل مطابق رابطه ۲۰ عمل می‌کنیم:

$$\begin{cases} H_0: dev_{base} = dev_{ACOPSOCa} \\ H_1: dev_{base} \neq dev_{ACOPSOCa} \end{cases} \quad (19)$$

$$\begin{cases} H_0: dev_{base} - dev_{ACOPSOCa} = 0 \\ H_1: dev_{base} - dev_{ACOPSOCa} \neq 0 \end{cases} \quad (20)$$

ارزیابی-روش	Lower	Upper	t	df	Sig. (2-tailed)
مقایسه نرخ انحراف					
ارزیابی ششم-مقایسه نرخ انحراف	0.01	0.23	2.45	10	0.034
ارزیابی هفتم-مقایسه نرخ انحراف	0.00	0.18	2.24	10	0.049
ارزیابی هشتم-مقایسه نرخ انحراف	-0.99	8.08	2.17	4	0.096
ارزیابی هشتم-مقایسه تعداد تکرارها	2.00	6.98	5.00	4	0.007
ارزیابی نهم-مقایسه نرخ انحراف	0.06	1.69	2.99	4	0.040
ارزیابی دهم-مقایسه نرخ انحراف	0.04	0.66	2.66	7	0.032
ارزیابی یازدهم-مقایسه نرخ انحراف	0.31	0.71	7.23	4	0.002
ارزیابی دوازدهم-مقایسه نرخ انحراف	0.32	0.80	5.57	7	0.001

مطابق با نتایج جدول ۶، روش ACOPSOCA از نظر آماری تفاوت معناداری در مقایسه با سایر روش‌ها ایجاد کرده و به طور متوسط نرخ انحراف و میانگین تعداد تکرارها را کاهش داده است. بنابراین، ترکیب الگوریتم‌های یادگیری ماشین [۳۷-۳۴] می‌توانند به خوبی برای برنامه‌ریزی مسیر استفاده گردند.

۵- نتایج

رشد بی‌رویه‌ی تولید خودرو از یک سو و عدم گسترش صحیح زیرساخت‌های جاده‌ای از سوی دیگر باعث ایجاد تراکم شدید، تأخیرهای طولانی و سوانح زیاد رانندگی به‌خصوص در شهرهای شلوغ شده است. برای حل مشکلات فوق پژوهش‌های زیادی انجام شده که عمده این روش‌ها بر رویکردهای فرا ابتکاری استوار هستند. در بسیاری از این الگوریتم‌ها در ابتدای کار مسیریابی بصورت تصادفی و در محدوده‌ای بزرگ انجام می‌شود. این موضوع موجب افزایش زمان جستجو می‌گردد. همچنین در تحلیل‌های صورت گرفته بر مبنای این الگوریتم‌ها تنها فاصله‌ی فیزیکی بین خودروها در نظر گرفته شده است. با توجه به اینکه عوامل محیطی نظیر ترافیک، در مسیریابی بسیار

از آزمون T زوجی نتایج جدول ۶ حاصل شده است. در این جدول بر روی تمامی ارزیابی‌های صورت گرفته در بخش‌های قبلی تست‌های آماری انجام گرفته است. ستون‌های LOWER و UPPER را می‌توان نوعی برآورد فاصله‌ای در نظر گرفت که آن را فاصله اطمینان گویند. ستون t آماره‌ی آزمون و df درجه آزادی (Degree of Freedom) است که بیانگر تعداد مقادیری است که در یک محاسبه مرتبط با شاخص می‌تواند آزادانه تغییر کنند. ستون آخر نیز، مقدار Sig یا مقدار احتمال (p - Value) برای آزمون است. در صورتی که مقدار Sig کمتر از احتمال خطا (در سطح اطمینان ۹۵٪ مقدار $\alpha = 0.05$ و در سطح اطمینان ۹۰٪ مقدار $\alpha = 0.1$ است) باشد، فرض صفر را رد خواهیم کرد و میتوان گفت روش پیشنهادی اختلاف معناداری ایجاد کرده است. به‌عنوان نمونه در سطر اول جدول ۶ که بر روی آزمایش اول از ارزیابی اول مقایسه انجام شده است، فرض صفر رد شده و روش پیشنهادی توانسته است اختلاف معناداری را ایجاد نماید.

جدول (۶)- هزینه‌های سفر بر روی مسیرهای متفاوت شهر پکن

ارزیابی-روش	Lower	Upper	t	df	Sig. (2-tailed)
ارزیابی اول-آزمایش اول نرخ انحراف	8.93	17.48	8.57	4	0.001
ارزیابی اول-آزمایش اول تعداد تکرارها	11.37	31.43	5.92	4	0.004
ارزیابی اول-آزمایش دوم نرخ انحراف	5.10	11.85	6.97	4	0.002
ارزیابی اول-آزمایش دوم تعداد تکرارها	9.73	19.87	8.10	4	0.001
ارزیابی اول-آزمایش سوم نرخ انحراف	0.05	2.00	2.90	4	0.044
ارزیابی اول-آزمایش سوم تعداد تکرارها	4.74	16.06	5.10	4	0.007
ارزیابی دوم-مقایسه نرخ انحراف	0.41	6.90	2.89	5	0.034
ارزیابی سوم-مقایسه نرخ انحراف	-0.14	1.03	2.13	4	0.100
ارزیابی چهارم-مقایسه نرخ انحراف	0.10	0.47	3.39	10	0.007
ارزیابی چهارم-مقایسه تعداد تکرارها	1.94	10.09	3.29	10	0.008
ارزیابی پنجم-آزمایش اول مقایسه نرخ انحراف	1.85	5.62	6.30	3	0.008
ارزیابی پنجم-آزمایش دوم	-1.13	11.42	1.89	8	0.095

مراجع

[1] Yang, Jun, Avralt-Od Purevjav, and Shanjun Li. "The marginal cost of traffic congestion and road pricing: evidence from a natural experiment in Beijing." *American Economic Journal: Economic Policy* 12.1 (2020): 418-53. doi:10.1257/pol.20170195

[2] Dorigo, Marco, Mauro Birattari, and Thomas Stutzle. "Ant colony optimization." *IEEE computational intelligence magazine* 1.4 (2006): 28-39. doi:10.1109/ci-m.2006.248054

[3] Karaoglan, Ismail, et al. "The location-routing problem with simultaneous pickup and delivery: Formulations and a heuristic approach." *Omega* 40.4 (2012): 465-477. doi:10.1016/j.omega.2011.09.002

[4] Doostali, Saeed, and Mohammad Khalily-Dermany. "A multi-hop PSO based localization algorithm for wireless sensor networks." *Soft Computing Journal* 8.1 (2021): 58-69. doi:10.22052/8.1.58

[5] Ghasemi, Reza, Hamid Reza Mohammadi, and Seyed Abbas Taher. "Frequency Control of an Islanded Microgrid based on Intelligent Control of Demand Response using Fuzzy Logic and Particle Swarm Optimization (PSO) Algorithm." *Soft Computing Journal* 6.2 (2021): 18-31.

[6] Wu, Chunjiang, Shijie Zhou, and Licai Xiao. "Dynamic path planning based on improved ant colony algorithm in traffic congestion." *IEEE Access* 8 (2020): 180773-180783. doi:10.1109/access.2020.3028467

[7] Mohammadi, Hamid Reza, and Ali Akhavan. "Performance Comparison of HSA, ICA and PSO Algorithms for Selective Harmonic Elimination in Cascaded Multilevel Inverter with Variable DC Sources." *Soft Computing Journal* 3.2 (2021): 20-30. doi:10.1109/isie.2014.6864834

[8] Dantzig, Fulderson, et al. "Solution of a large-scale traveling-salesman problem." *50 Years of Integer Programming 1958-2008: From the Early Years to the State-of-the-Art* (2010): 393-410. doi:10.1007/978-3-540-68279-0_1

[9] Colorni, Alberto, Marco Dorigo, and Vittorio Maniezzo. "Distributed optimization by ant colonies." *Proceedings of the first European conference on artificial life*. Vol. 142. 1991. doi:10.7717/peerjcs.729/fig-4

[10] Wu, Min, and Licai Zhang. "A hybrid algorithm of ant colony and particle swarm optimization on solving TSP problem." *Jilin Univ* (2006): 402-405. doi:10.1109/chicc.2006.280708

[11] Shi, Chun-xue, et al. "Solving path planning problem by an aco-pso hybrid algorithm." *International Conference on Intelligent Systems and Knowledge Engineering 2007*. Atlantis Press, 2007. doi:10.2991/iske.2007.91

مؤثر هستند، باید تأثیر این عوامل را در مسیریابی در نظر گرفت. بدین منظور برای حل مشکلات مطرح شده یک روش برنامه‌ریزی مسیر پویا مبتنی بر ترکیب الگوریتم‌های کلونی مورچگان و ازدحام ذرات با اعمال تابع ابتکاری کسینوس زوایا در این مقاله پیشنهاد شد. روش پیشنهادی عوامل مختلف وضعیت جاده‌ها از جمله طول جاده‌ی شهری و ترافیک ورودی و خروجی تقاطع‌ها را در مسیریابی وسایل نقلیه در نظر می‌گیرد و در مسیریابی برای نقاطی که هم راستای مسیر پیمایش به سمت مقصد نهایی باشد شانس بیشتری قائل می‌شود. برای انجام بهینه‌سازی مسیریابی از الگوریتم کلونی مورچگان استفاده شد. با توجه به ضعف این الگوریتم در همگرایی و جستجوی تصادفی در مراحل اولیه و همچنین با توجه به زمان‌بر بودن تنظیم محدوده پارامترهای این الگوریتم، الگوریتم کلونی مورچگان با الگوریتم ازدحام ذرات ترکیب و برای بهبود دقت و کاهش زمان مسیریابی، تابع ابتکاری کسینوس زوایا به آن افزوده شده است. نتایج آزمایش‌های صورت گرفته بر اساس کتابخانه TSPLIB و در آزمایش برنامه‌ریزی مسیر پویا بر اساس داده‌های ترافیکی واقعی شهر پکن نشان می‌دهد روش پیشنهادی مقاله دقت محاسباتی و همچنین زمان اجرا محاسبات را نسبت به روش‌های پیشین بهبود بخشیده است. در کارهای آتی می‌توان برای اجرای واقعی روش پیشنهادی مقاله در محیط حقیقی که نیازمند سخت‌افزار و نرم‌افزار لازم است اقدام کرد. و می‌توان علاوه بر تاثیر و وضعیت جاده‌ها، تاثیر وضعیت آب و هوایی را به‌عنوان پارامترهای تاثیرگذار در الگوریتم پیشنهادی مقاله تاثیر داد. همچنین با توجه به عملکرد نسبتاً خوب الگوریتم‌های SOS – SA و Discrete – EHO می‌توان از این دو در ترکیب با الگوریتم کلونی مورچگان یا سایر الگوریتم‌ها برای افزایش دقت و کاهش زمان مسیریابی در برنامه‌ریزی مسیر پویا استفاده کرد.

- [25] Elsayed, Eman K., Asmaa Hekal Omar, and K. E. Elsayed. "Smart solution for STSP semantic traveling salesman problem via hybrid ant colony system with genetic algorithm." *International Journal of Intelligent Engineering and Systems* 13.5 (2020): 476-489. doi:10.22266/ijies2020.1031.42
- [26] Ezugwu, Absalom El-Shamir, and Aderemi Oluyinka Adewumi. "Discrete symbiotic organisms search algorithm for travelling salesman problem." *Expert Systems with Applications* 87 (2017): 70-78. doi:10.1016/j.eswa.2017.06.007
- [27] Liao, Erchong, and Changan Liu. "A hierarchical algorithm based on density peaks clustering and ant colony optimization for traveling salesman problem." *Ieee Access* 6 (2018): 38921-38933. doi:10.1109/access.2018.2853129
- [28] Hossam, Anass, Abdelhamid Bouzidi, and Mohammed Essaid Riffi. "Elephants herding optimization for solving the travelling salesman problem." *Advanced Intelligent Systems for Sustainable Development (AI2SD'2018) Vol 2: Advanced Intelligent Systems Applied to Energy*. Springer International Publishing, 2019. doi:10.1007/978-3-030-12065-8_12
- [29] Ezugwu, Absalom El-Shamir, Aderemi Oluyinka Adewumi, and Marc Eduard Frincu. "Simulated annealing based symbiotic organisms search optimization algorithm for traveling salesman problem." *Expert Systems with Applications* 77 (2017): 189-210. doi:10.1016/j.eswa.2017.01.053
- [30] Hammouri, Abdelaziz I., et al. "A dragonfly algorithm for solving traveling salesman problem." *2018 8th IEEE international conference on control system, computing and engineering (ICCSCE)*. IEEE, 2018. doi:10.1109/iccsce.2018.8684963
- [31] Jiang, Y., J. Zhang, and S. Zhong. "Parameter analysis of ant colony algorithm." *J. Comput. Eng. Appl.* 20.3 (2007): 31-36. doi:10.12783/dtetr/icca2016/6052
- [32] <https://report.amap.com/congest.do>
- [33] Neter, John, William Wasserman, and George A. Whitmore. "Applied statistics". No. 519.5 N469. Allyn and Bacon, 1978. doi:10.1177/001316448004000148
- [34] Amiri, Maryam, Leyli Mohammad-Khanli, and Raffaella Mirandola. "A new efficient approach for extracting the closed episodes for workload prediction in cloud." *Computing* 102 (2020): 141-200. doi:10.1007/s00607-019-00734-3
- [35] Amiri, Maryam, and Hesam Askari. "Illegal Miner Detection based on Pattern Mining: A Practical Approach." *Journal of Computing and Security* 9.2 (2022): 1-10. doi: 10.22108/jcs.2022.133306.1096
- [36] Farnaghi-Zadeh, Fatemeh, Mohsen Rahmani, and Maryam Amiri. "Feature Selection Using Neighborhood based Entropy." *Journal of Universal Computer Science* 28.11 (2022): 1169-1192. doi:10.3897/jucs.79905
- [37] Jaryani, Farhang, and Maryam Amiri. "A Pre-Trained Ensemble Model for Breast Cancer Grade Detection Based on
- [12] Pal, Dheeraj, et al. "Improved optimization technique using hybrid ACO-PSO." *2016 2nd International Conference on Next Generation Computing Technologies (NGCT)*. IEEE, 2016. doi:10.1109/ngct.2016.7877428
- [13] Ouyang, Ai Jia, and Yong Quan Zhou. "An improved PSO-ACO algorithm for solving large-scale TSP." *Advanced Materials Research*. Vol. 143. Trans Tech Publications Ltd, 2011. doi:10.4028/www.scientific.net/amr.143-144.1154
- [14] Li, Wei, et al. "An ant colony optimization algorithm with adaptive greedy strategy to optimize path problems." *Journal of Ambient Intelligence and Humanized Computing* (2022): 1-15. doi:10.1007/s12652-021-03120-0
- [15] Stodola, Petr, Pavel Otrisal, and Kamila Hasilová. "Adaptive ant colony optimization with node clustering applied to the travelling salesman problem." *Elsevier, Swarm and Evolutionary Computation* 70 (2022): 101056. doi:10.1016/j.swevo.2022.101056
- [16] Zheng, Rui-zhao, Yong Zhang, and Kang Yang. "A transfer learning-based particle swarm optimization algorithm for travelling salesman problem." *Journal of Computational Design and Engineering* 9.3 (2022): 933-948. doi:10.1093/jcde/qwac039
- [17] Sawsan, Agamy, Mutaher, Alwi, and Mohsen, "A Fuzzy Logic for Parameter Adaptation in Ant Colony Optimization Approach", *International Journal of Innovative Science and Research Technolog* (2022). doi:10.2991/ifs-a-eusflat-15.2015.108
- [18] Sahin, Murat. "Solving TSP by using combinatorial Bees algorithm with nearest neighbor method." *Neural Computing and Applications* 35.2 (2023): 1863-1879. doi:10.1007/s00521-022-07816-y
- [19] <https://github.com/article2023-uni/code-dataset-Description>
- [20] Zhangqi, Wang, Zhu Xiaoguang, and Han Qingyao. "Mobile robot path planning based on parameter optimization ant colony algorithm." *Procedia Engineering* 15 (2011): 2738-2741. doi:10.1016/j.proeng.2011.08.515
- [21] Ahmed, A. K. M., and Ji Ung Sun. "An improved particle swarm optimization algorithm for the travelling salesman problem." *Advanced Science Letters* 22.11 (2016): 3318-3322. doi:10.1166/asl.2016.7864
- [22] Xie, Xin, and Peng Wu. "Research on the optimal combination of ACO parameters based on PSO." *2010 International Conference on Networking and Digital Society*. Vol. 1. IEEE, 2010. doi:10.1109/icnds.2010.5479311
- [23] <http://comopt.ifi.uni-heidelberg.de/software/TSPLIB95/>.
- [24] Deng, Wu, Junjie Xu, and Huimin Zhao. "An improved ant colony optimization algorithm based on hybrid strategies for scheduling problem." *IEEE access* 7 (2019): 20281-20292. doi:10.1109/access.2019.2897580

پذیرفته شده در مجله محاسبات نرم