# الگوریتم مورچه‌ای برای طراحی مسیر حرکت باربران خودکار در سیستم تک حلقه


کورش عشقی* و مرتضی کاظمی**

دانشکدهٔ مهندسی صنایع، دانشگاه صنعتی شریف



(دریافت مقاله: ۸۱/۲/۷ – دریافت نسخه نهایی: ۸۲/۸/۲۱)



**چکیده** – در این مقاله الگوریتمی ابتکاری برای مسئلهٔ طراحی مسیر حرکت باربران خودکار در سیستم تک حلقه معرفی می‌شود. در این مسئله هدف تعیین کوتاهترین حلقه برای یک باربر خودکار در چیدمان کارخانه به نحوی است که با هر دپارتمان لااقل یک ضلع مشترک داشته باشد. برای این منظور در ابتدا با استفاده از خواص مسئله آن را به مسئله‌ای معادل در نظریهٔ گراف تبدیل کرده و سپس با به‌کارگیری الگوریتم فرا ابتکاری مورچه‌ای مسئله را حل خواهیم کرد. نتایج آزمایشات کارایی مطلوب الگوریتم پیشنهادی را در مقایسه با سایر روشها در حل مسائل مسیریابی سیستم تک حلقه نشان می‌دهد.



**واژگان کلیدی:** طراحی مسیر حرکت باربران خودکار، الگوریتم مورچه‌ای، الگوریتمهای فرا ابتکاری


# Ant Colony Algorithm for the Single Loop Routing Problem


**K. Eshgee and M. Kazemi**

Associate Professor and PhD Student, respectively, Department of Industrial Engineering,
Sharif University of Technology



**Abstract**: ***In this paper, a new algorithm for solving the single loop routing problem is presented. The purpose of the single loop routing problem(SLRP) is to find the shortest loop for an automated guided vehicle covering at least one edge of each department of a block layout. First it shown that this problem can be represented as a graph model. Then a meta-heuristic algorithm based on and colony system is developed for ALRP by using the properties of the graph model. Computational results show the efficiency of the proposed algorithm in comparison with other techniques for solving SLRP.***



**Keywords**: ***Single loop routing problem, Ant colony optimizaion algorithm, Meta-heuristic algorithms***


---

*ـ دانشیار　　　　**ـ دانشجوی دکترا

## ۱– مقدمه

در سیستم حمل و نقل خودکار استفاده از باربران خودکار[۱] (AGV) از جمله سیستمهای جابه‌جایی محسوب می‌شود که استفاده از آن در سالهای اخیر روز به روز افزایش می‌یابد. دلیل این امر نیز انعطاف پذیری بالای این تجهیزات در سیستم جابه‌جایی مواد است. باربران خودکار خوروهایی بدون سرنشین‌اند که توسط رایانه کنترل شده و برای انتقال مواد از نقطه‌ای به نقطه دیگر در چیدمان کارخانه مورد استفاده قرار می‌گیرند. یکی از مهمترین عوامل در طراحی مسیر حرکت AGV آن است که کل فاصلهٔ طی شده توسط این وسایل کمینه شود. سیستمهای مختلفی برای مسیریابی AGV وجود دارد که می‌توان به سیستمهای سنتی، کوتاهترین مسیر دوطرفه و سیستم تک حلقه اشاره کرد[۲]. در این مقاله مسیریابی سیستم تک حلقه مورد بررسی قرار می‌گیرد. در سیستم تک حلقه، هدف طراحی یک حلقهٔ بسته و غیر متقاطع در کارخانه به نحوی است که همهٔ دپارتمانهای کارخانه حداقل به یکی از اضلاع این حلقه دسترسی داشته باشند. نام این مسئله را طراحی سیستم مسیریابی تک حلقه می‌نامیم و از این پس آن را با علامت اختصاری SLRP نشان خواهیم داد.

مسئلهٔ مسیریابی باربران خودکار در سیستم تک حلقه نخستین بار در سال ۱۹۹۰ توسط سینریچ[۸] معرفی شد. هدف وی معرفی سیستمی بود که علاوه بر داشتن کارایی سیستمهای مسیریابی که در آن زمان وجود داشت از نظر محاسباتی برای یافتن جواب به محاسبات کمتری نیاز داشته باشد. در سال ۱۹۹۲ تانچوکو و سینریچ[۹] به تعیین کوتاهترین حلقه و همچنین تعیین مکان ایستگاههای بارگیری و تخلیه در یک چیدمان پرداختند. در این مدل ابعاد مسئله قابل حل مشخص نشده است. این دو محقق یک سال بعد در سال ۱۹۹۳ [۱۰] مدل خود را بهبود داده و از روش شاخه و کران برای پیدا کردن جواب بهینه استفاده کردند. در سال ۱۹۹۶ لاپورته و سایرین، مسئلهٔ SLRP را به صورت یک مسئلهٔ فروشندهٔ دوره گرد تعمیم یافته مدل کردند[۷]. پیچیدگی محاسباتی مسئلهٔ مسیریابی توسط دازمن و دیگران[۳] بررسی شد. آنها نشان دادند که این مسئله به ردهٔ مسائل NP-hard تعلق دارد. در سال ۲۰۰۰ آصف وزیری و دیگران[۱] با استفاده از برنامه‌ریزی خطی این مسئله را مدل کرده و با استفاده از خواص هندسی مسئله به کاهش تعداد محدودیتهای مدل پرداختند و موفق شدند که مسئله را برای نمودهای با ابعاد متوسط در زمان مناسبی حل کنند.

در این مقاله، الگوریتمی ابتکاری برای حل مسئلهٔ SLRP با استفاده از تکنیک بهینه‌سازی بر مبنای رفتار مورچگان[۳] (ACO) که آن را الگوریتم مورچه‌ای می‌نامیم، ارائه خواهد شد. برای این منظور ابتدا مسئلهٔ SLRP را با استفاده از خواص آن به یک مسئلهٔ معادل در نظریهٔ گراف تبدیل کرده و سپس با استفاده از خواص این گراف، الگوریتم حل مسئله را طراحی خواهیم کرد.

## ۲– معرفی الگوریتم مورچه‌ای (ACO)

الگوریتم فرا ابتکاری بهینه‌سازی بر مبنای رفتار مورچگان (ACO) در اوایل دههٔ نود میلادی توسط دوریگو، مانیزو و کلرنی [٤ و ٦] معرفی شد. این الگوریتم از رفتار اجتماعی مورچه‌ها الهام گرفته شده است. مورچه‌ها با آنکه فاقد قدرت بینایی‌اند می‌توانند کوتاهترین مسیر از منبع تغذیه تا لانهٔ خویش را با استفاده از مواد شیمیایی که در هنگام حرکت از خود بر جای می‌گذارند، و به فرومون[۴] موسومند پیدا کنند[۶]. مورچه‌ها در هنگام حرکت با به جای گذاشتن فرومون، فرومونهای باقیمانده از بقیهٔ مورچه‌ها را (به صورت تصادفی) دنبال می‌نمایند. از نظر مورچه‌ها مسیری مطلوبتر است که مقدار فرومون بیشتری داشته باشد. طریقهٔ یافتن کوتاهترین مسیر با استفاده از فرومون در شکل (۱) نشان داده شده است.

حالت A را در شکل(۱) در نظر بگیرید. مورچه‌ها به یک دوراهی رسیده‌اند و مجبورند تصمیم بگیرند که به سمت بالا یا مستقیم حرکت کنند. در این لحظه هیچ پیش زمینه‌ای در مورد بهترین انتخاب وجود ندارد بنابراین مورچه‌ها مسیر حرکت خود را به صورت تصادفی انتخاب می‌کنند. می‌توان انتظار

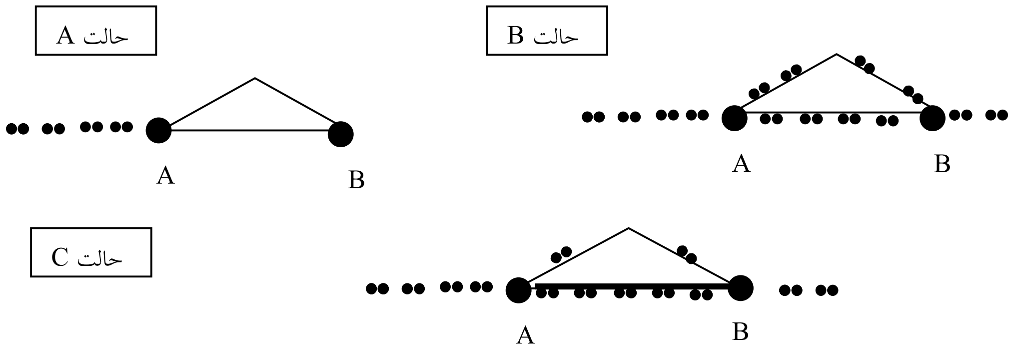
داشـت تـا بـه طـور متوسط نیمی از مورچه‌ها مسیر بالا و نیمی دیگر مسیر مستقیم را برای ادامهٔ حرکت خود انتخاب نمایند که در حالـت B نشـان داده شـده اسـت. از آنجـا که مسیر مستقیم پایینـی کوتاهتـر از مسـیر بالایـی بـوده بـا فـرض مساوی بودن سـرعت حـرکت مـورچه‌ها تعداد بیشتری مورچه می‌توانند این مسیر را در واحد زمان طی کنند که این امر موجب انباشته شدن سـریعتر فـرومون در ایـن مسـیر مـی‌شـود. بـه تـدریج اختلاف فـرومون دو مسیر زیاد می‌شود و پس از مدتی اختلاف فرومون در دو مسـیر بـه انـدازهٔ کافـی بـزرگ مـی‌شـود تـا بـر تصـمیم مـورچه‌هـای جدید در انتخاب مسیر تأثیرگذار باشد. این مطلب در قسـمت C شـکل (۱) نشـان داده شـده است. از حالا به بعد مـورچه‌ها به دلیل یافتن فرومون بیشتری در مسیر پایینی به طور احتمالی ترجیح می‌دهند تا این مسیر را انتخاب کنند. این فرایند بـا یـک بازخـور مثبت ادامه می‌یابد، یعنی اینکه افزایش انتخاب مسـیر مـوجب افـزایش فرومون در این مسیر و افزایش فرومون مـوجب افـزایش انـتخاب ایـن مسیر می‌شود و هر مورچه برای مدتـی مسـیر ثابتـی را انتخاب کرده به نحوی پس از مدتی تمام مـورچه‌هـا مسـیر کوتاهتـر را بـرای ادامـه حرکت خود انتخاب می‌کنند.

اولـین الگوریـتم بهینـه‌سـازی ACO براسـاس همـین رفتار مـورچه‌هـا ابداع شد. الگوریتم ACO نخستین بار در حل مسئله فروشندهٔ دوره گرد استفاده شد و پس از آن در حل دیگر مسائل بهینـه سـازی ترکیبی به‌کار رفت که می‌توان به مسائل تخصیص مربع، مسیریابی خودروها، رنگ آمیزی‌گرافها و ۰۰۰ اشاره کرد. صـورت کاملـی از ایـن مسـائل بـه همـراه مـراجع مـربوطه در [۵ و ۱۱] ذکـر شـده است. در روش ACO معمولا مسئله به یک مـدل در نظـریه گراف تبدیل شده و جواب مسئله نیز به‌صورت مسـیری بـرروی آن تعـریف خـواهد شـد [۴]. در این روش از تعـدادی مـورچهٔ مصنوعی برای حل مسئله استفاده می‌شود. این مـورچه‌هـا به جست‌وجو در گراف مسئله می‌پردازند و با مبادلهٔ اطلاعـات، از طـریق بـه جـای گذاشتن فرومون در طول یالهای گـراف در ساختن جواب به همکاری با یکدیگر می‌پردازند. هر چـه که فرومون یک یال بیشتر باشد آن یال از مطلوبیت بالاتری برای انتخاب برخوردار است. در نهایت جوابی تولید می‌شود که از مطلوبترین یالها استفاده شده و احتمالا نزدیک به جواب بهینه مسئله خواهد بود. در اکثر مسائلی که توسط الگوریتم مورچه‌ای حـل شـده‌انـد نـتایج حاصـل بیانگـر برتری این روش بر سایر روشهای فراابتکاری مخصوصاً در مسایل مسیریابی است.

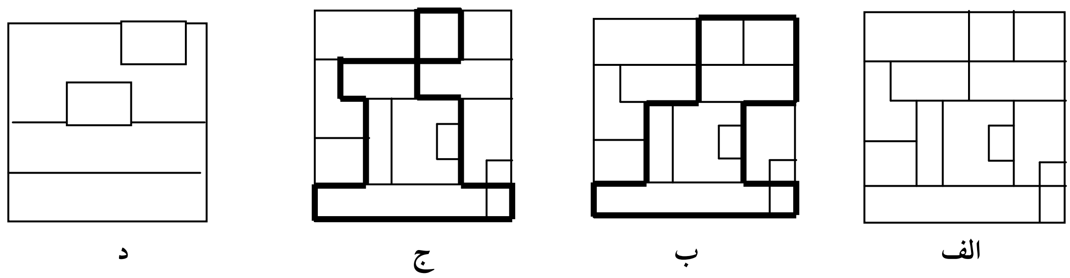
## ۳- تعاریف

در ایـن بخـش بـه معرفـی مسـئلهٔ مسیریابی باربران خودکار در سیستم تک‌حلقه و مفاهیم مربوط به آن می‌پردازیم. یک چیدمان کارخانـه را کـه از $n$ دپارتمـان بـا خطوط مستقیم تشکیل شده اسـت، در نظـر بگیـرید شـکل(۲- الـف) [۱]. چیدمان کارخانه یکپارچه است و فرض می‌شود که هیچ دپارتمانی را نتوان یافت کـه محـیط آن به‌طور کامل و بدون داشتن یال مشترک در درون دپارتمـان دیگر قرار بگیرد. دپارتمان ها لزوماً مستطیل نبوده اما فرض می‌شود که تمامی زوایای متعلق به دپارتمانها °۹۰، °۱۸۰ و °۲۷۰ اسـت. مسـئلهٔ مسـیریابی باربـران خودکار در سیستم تک حلقه یا SLRP عبارت است از یافتن یک حلقهٔ بسته با حداقل طول در امتداد اضلاع دپارتمانها به نحوی که اولاً این حلقه غیر مـتقاطع بوده و ثانیاً هر یک از دپارتمانها یک ضلع مشترک با آن داشـته باشد، شکل(۲- ب و ج). یک جواب موجه یا یک حلقهٔ مـوجه بـرای مسـئله حلقـه‌ای است که دارای شرایط اول و دوم باشـد امـا لـزوماً بـا حـداقل طول نباشد. البته هر مسئله SLRP لـزوما دارای جـواب مـوجه نیسـت و در حـالات خاصـی نظیر شـکل (۲- د) مسـئله دارای جـواب مـوجه نیست. در این مقاله فـرض می‌شود که روشهای ارائه شده در مسائلی به کار رود که حداقل یک جواب موجه دارند.

به ازای هر چیدمانی یک گراف نظیر G= (N,A) وجود دارد کـه آن را گـراف چـیدمان [۵] مـی‌نامـیم. در این گراف، منظور از مجمـوعه N مجمـوعه گـره‌ها و مجموعه A مجموعه یالهای این گـراف اسـت و به‌ازای هر دپارتمان موجود در چیدمان یک گره در نظـر می‌گیریم. علاوه برآن محوطهٔ بیرونی کل چیدمان را نیز




**شکل ۱- طریقهٔ یافتن کوتاهترین مسیر بین دو نقطه توسط مورچه‌ها**

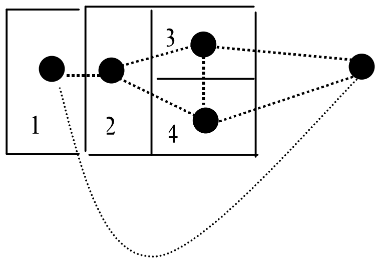



**شکل ۲ – الف- نمایی از چیدمان یک کارخانه، ب- مثالی از یک حلقهٔ موجه در طراحی مسیر حرکت، ج- یک مثال از یک حلقهٔ غیرموجه، د- مثالی از یک مسئله فاقد جواب موجه**

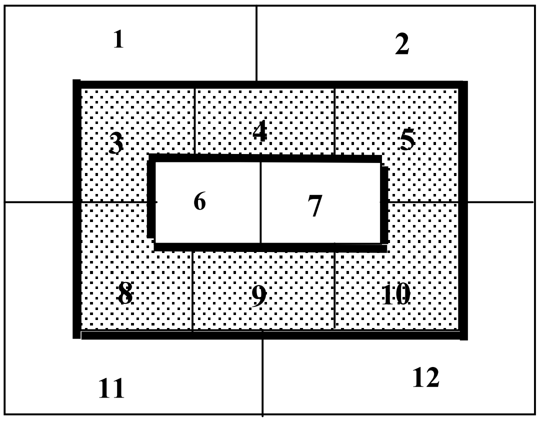
بـه عـنوان یک دپارتمان در نظر گرفته و گرهٔ متناظر با آن را در گـراف چـیدمان گرهٔ صفر می‌نامیم. در گراف چیدمان دو گره با یـک یـال به یکدیگر متصل‌اند ویا به عبارت دیگر مجاورند اگر وتـنها اگـر دپارتمانهـای نظیـر این دو گره دارای ضلع یا اضلاع مشترکی باشند. در نتیجه گرهٔ صفر و گره‌های نظیر دپارتمانهایی کـه در مـرز چیدمان قرار دارند نیزبا هم مجاور می‌شوند. گراف چـیدمان بـا فرضـیات ذکـر شـده یـک گـراف مسطح است. از خـواص یـک گـراف مسـطح[۶] بـا حداقل سه راس آن است که تعـداد یالهـای ایـن گـراف حد اکثر برابر با $3|N|-6$ است که $|N|$ تعـداد رئـوس ایـن گـراف است [۲]. در شکل (۳) گراف چـیدمان وابسـته بـه یـک چـیدمان متشکل از چهار دپارتمان با خطوط خط چین نشان داده شده است:

منظور از زیرگراف القایی[۷] نظیر مجموعه‌ای از دپارتمانها، یک زیرگراف از گراف چیدمان است که اولاً گره‌های آن وابسته به دپارتمانهای موجود در آن مجموعه باشند و ثانیاً یالهای آن مجموعه یالهایی باشد که در گراف چیدمان ما بین این گره‌ها موجود است. یک مجموعه از دپارتمانها را یکپارچه گوییم اگر زیرگراف القایی نظیر آن همبند باشد. برای مثال در شکل۳ زیرگراف القایی حاصل از دپارتمانهای ۲و۳و۴ تشکیل یک مثلث را داده و چون همبند است این دپارتمانها یکپارچه‌اند ولی زیرگراف القایی نظیر دپارتمانهای ۱و۳و۴ همبند نیست و در نتیجه این دپارتمانها یکپارچه نیستند.

اجـتماع مـرزی چـند دپارتمـان یکـپارچه عـبارت اسـت از اجـتماع یالهـای ایـن دپارتمانها با این شرط که یالهایی را که در مـرز مشـترک ایـن دپارتمانهـا واقـع شـده‌اند را حذف کنیم. در شـکل (٤) اجـتماع مـرزی دپارتمانهای۳ و ۴و۵ و ۸و ۹و۱۰ که بـه‌صـورت خطـو ط هاشـور نمایش داده شده است تشکیل دو حلقـه جدا از هم را می‌دهد. به عبارت دیگر اجتماع مرزی چند دپارتمـان مجمـوعه یالهایی است که بروی مرز حاصل از ادغام این دپارتمانها در یکدیگر به‌وجود می‌آید.




**شکل ۳- گراف چیدمان**




**شکل ۴- اجتماع مرزی دپارتمانها**

## ۴- یافتن جواب موجه مسئلهٔSLRP از روی گراف چیدمان

همان‌گونه که قبلاً دیدیم هر جواب موجه از یک مسئلهٔ SLRP را می‌توان به‌صورت یک حلقهٔ بسته در چیدمان نشان داد که آن را حلقهٔ موجه می‌نامیم.این حلقه در حقیقت از اجتماع مرزی دپارتمانهایی تشکیل می‌شود که در درون حلقه‌اند. زیرگراف القایی متناظر با دپارتمانهایی را که در درون این حلقه قرار می‌گیرند را نیز زیرگراف موجه نامیده و گره‌های موجود در آن را با برچسب مشخص می‌سازیم. در این‌صورت پس از یافتن حلقهٔ موجه هر گرهٔ موجود در گراف چیدمان با دارای برچسب خواهد بود و یا با یکی از گره‌های برچسب دار مجاور می‌شود. علاوه بر آن مشاهده می‌شود که در گراف چیدمان:

۱- زیر گراف القایی نظیر گره‌های برچسب دار همبند است.

۲- حداقل یکی از گره‌های مجاور یک گرهٔ برچسب دار است. زیرا در غیر این‌صورت گره برچسب داری نظیرv خواهیم داشت. که تمام گره‌های مجاورش نیز برچسب دار بوده و این بدین معنی است حلقهٔ ناشی از اجتماع مرزی دپارتمانv و دپارتمانهای مجاورش دارای ضلع مشترکی با دپارتمان نظیرv نخواهد داشت که این امر با فرض موجه بودن حلقه در تناقض است. برای مثال در شکل(٤) زیر گراف القایی ناشی از دپارتمانهای ۳، ٤، ٥، ٦، ٧، ٨، ۹ و ۱۰ زیر گراف موجه نیست چرا که در آن تمامی گره‌های مجاور گره‌های برچسب دار ٦ یا ٧ خود برچسب دارند. و یا به عبارت دیگر حلقه ناشی از اجتماع مرزی این دپارتمانها دارای ضلع مشترکی با دپارتمانهای ٦ یا ٧ نیست بنابراین نمی‌تواند موجه باشد.

۳- حداقل یکی از گره‌های مجاور یک گرهٔ غیر برچسب دار (غیر از گره صفر) برچسب دار خواهد بود چرا که در غیر این صورت اصولاً حلقه موجه نخواهد شد. به‌عبارت دیگر هر یک از دپارتمانهای خارج از حلقه می‌بایستی یال و یا یالهایی مشترک با حلقه بسته تشکیل شده داشته باشند. برای مثال در شکل(۴) اجتماع مرزی دپارتمانهای ۶ و۷ نمی‌تواند تشکیل یک

حلقـه موجه را بدهد زیرا که در  زیر گراف القایی وابسته به این دپارتمانهـا گـره هـای وابسـته به دپارتمانهای ۱و۲و۱۱و۱۲ غیر برچسـب دار بـوده و گـرههای مجاور آنها نیز غیر برچسب دار خواهد شد.

٤- اگـر از گراف چیدمان کلیه گرههای با برچسب دار و یالهای متصل به آن را حذف کنیم زیرگراف حاصل باید همچنان همبند باقـی بمانـد چـرا کـه در غیـر ایـن صـورت اجـتماع مـرزی دپارتمانهـای وابسته به این گرهها تشکیل دو حلقه جدا از هم را مـیدهـند. بـرای مـثال در شـکل (۴) زیر گراف القایی ناشی از دپارتمانهـای ۳و۴و۵و ۸و۹ و۱۰ زیر گراف موجه نیست چرا که در آن زیر گراف حاصل از حذف گرههای ۳و۴و۵ و ۸ و۹ و۱۰ از گراف چیدمان غیر همبند میشود.

در شرایط فوق در حقیقت وجود شرایط ۱و۳ بدین خاطر است که حلقه ناشی از اجتماع مرزی دپارتمانها یکپارچه بوده و با هر یک از دپارتمانهای موجود در چیدمان ضلع مشترک داشته باشد. واضح است که این مجموعه بایست یکپارچه باشد چرا که در غیر این صورت اصولاً حلقهای تعریف نمیشود. اگر زیرگراف القایی ناشی از دپارتمانها دارای شرایط ۱و ۳ باشد اما یکی از شروط ۲و یا ۴ را نقض کند اصطلاحا گفته می شود که دپارتمانهای وابسته به این زیر گراف دارای حفرهاند.

حلقه ناشی از اجتماع مرزی دپارتمانها در صورتی جواب مسئله است که دپارتمانهای وابسته به آن غیرحفرهدار نیز باشد زیرا اگر مجموعۀ دپارتمانها حفرهدار باشد آنگاه مرز این مجموعه یا تشکیل دو حلقۀ جدا از هم را میدهد نظیر هنگامی که در شکل (۴) اجتماع مرزی شامل دپارتمانهای ۹،۸،۵،٤،۳ و ۱۰ باشد و یا اینکه دپارتمانی از مجموعۀ دپارتمانهای حفرهدار وجود دارد که تمام دپارتمانهای مجاور با آن متعلق به جواباند و لذا نمیتواند یال مشترکی با حلقه داشته باشد مثلا در شکل (۴) وقتی که مجموعۀ ما شامل دپارتمانهای ۹،۸،۷،٦،۵،٤،۳ و ۱۰ باشد، در این حالت دپارتمان ٦ یا ۷ یال مشترکی با حلقۀ ایجاد شده ندارد.

شـرایط فـوق در حقـیقت شـرایط لازم بـرای وجود زیر گراف مـوجه بودند. اکنون به قضیه زیر که شرط کافی برای وجود یک زیر گراف موجه را بیان می کند توجه کنید:

قضیه: فرض کنید که یک زیر گراف القایی نظیر $G_U = (V_U, E_U)$ از یک گراف چیدمان داده شده است. اگر گراف$G_U$ در شرایط زیر صدق کند آنگاه $G_U$ یک زیر گراف موجه است:

۱. $G_U$همبند باشد.

۲. حداقل یکی ازگره های مجاور به یک گره$V_U$ عضوی از مجموعه $V- V_U$ باشد

۳. هر گره از مجموعه $V- V_U$ (غیر از گره صفر) دارای حداقل یک گره مجاوراز $V_U$ باشد.

۴. گراف $G - G_U$ همبند باشد.

اثـبات: حلقـه ناشـی از اجـتماع مـرزی دپارتمانهـای متناظـر با گـرههـای $V_U$ را در نظر بگیرید. شرط ۱ بیانگر آن ست که اولا دپارتمانهـا یکـپارچهانـد و تشکیل یک حلقه پیوسته را میدهند. شـرط ۳ تضـمین میکند که حلقه مذکور با هر دپارتمان حداقل دارای یـک ضـلع مشـترک اسـت و شرایط ۲و ۴ از وقوع حفره جلوگیری میکنند.

## ۵- الگوریتم یافتن "زیر گراف موجه"

بـا تـوجه بـه قضـیه بخـش قبل میتوان نتیجه گرفت که مسئله تعیـین یـک حلقه موجه در SLRP (در صورتیکه جواب موجه بـرای آن موجود باشد) با یافتن یک زیرگراف از گراف چیدمان مسـئله که واجد چهار شرط ذکر شده باشد معادل است. حال به ارائـۀ الگوریتمی خواهیم پرداخت که قادر است یک زیر گراف مـوجه از گـراف چیدمان را بیابد بنحوی که با توجه به آن بتوان یک جواب موجه را برای مسئله SLRP به دست آورد:

۱- یـک گـره غیـر از گـرۀ صـفر را به دلخواه از گراف چیدمان برگـزینید و آن را برچسـب دار کنـید. اگـر ایـن گـره بـه تمـام گـرههـای دیگـر گـراف (غیر از گره صفر) متصل باشد متوقف شوید. در این حالت حلقه موجه بر روی مرز این دپارتمان قرار دارد.

۲- از میان گره‌های غیر برچسب‌دار (به‌جز گره صفر) متصل به گره‌های برچسب‌دار، گر ه‌ای را انتخاب کنید که در شرایط زیر بـرای زیـر گـراف القایی حاصل از گره‌های برچسب دار و این گره صدق کند:

الـف: هـر گـره از ایـن زیـر گراف حداقل با یک گره برچسب نخورده مجاور باشد.

ب: اگـر از گـراف چیدمان این زیر گراف القایی را حذف کنیم همچنان همبند باقی بماند.

اگر چنین گره‌ای پیدا نشد مجددا از گام ۱ شروع کنید.

۳. گـره را بـه مجمـوعه گـره‌های برچسب‌دار اضافه کنید و زیر گـراف القایـی حاصل از گره‌های برچسب دار را با توجه به این گـره تعمـیم دهـید. اگـر هر گره غیر برچسب‌دار مجاور یکی از گـره‌هـای این زیر گراف باشد متوقف شوید. در این‌صورت زیر گـراف مـوجه بـه‌دست آمده است. در غیر این‌صورت به گام ۲ برگردید.

زیرگـراف به‌دست آمده از این الگوریتم از آنجا که در گام۲ گـرهٔ جدیـد را از مـیان گـره‌هـای مجـاور بـا یکـی از گره‌های زیرگـراف سـاخته شده برمی‌گزیند، در نهایت همبند خواهد بود و بـه‌دلـیل بندهای الف و ب گام ۲ شرایط ۲ و۴ موجود در یک جـواب مـوجه مسـئله را ارضـا کـرد و بـا توجه به این‌که شرط توقـف الگوریـتم ارضـای شرط ۳ موجه بودن یک جواب است در نهایت منجر به یک زیرگراف موجه خواهد شد.

## ۶- حل مسئلهٔ SLRP به کمک الگوریتم ACO

در بخـش(۴) جـواب مـوجه بـرای یـک مسـئلهٔ SLRP را بـه‌صـورت یـک زیرگـراف از گـراف چیدمان بیان کردیم و در بخـش (۵) الگوریتمی بر این پایه طراحی نمودیم. اینک می‌توان از ایـده موجـود در الگوریتمهای ACO در الگوریتم بخش ۵ به شرح زیر استفاده کرد:

در ابتدا $m$ مورچه را به‌صورت تصادفی در $m$ گرهٔ گراف چـیدمان قـرار مـی‌دهیم. سپس هر یک از مورچه‌های مصنوعی، طـبق الگوریـتم ارائـه شـده در بخش(۵)، شروع به ساختن یک جواب موجه برای مسئله می‌کنند. مورچه‌ها ساختن جواب خود را زمانـی آغـاز مـی‌کنـند که مورچهٔ قبلی جواب خود را تکمیل کـرده باشـد. در ایـن مسـئله جـواب نهایـی با یک زیرگراف از گـراف چیدمان مشخص می‌شود. از آنجا که چنین زیرگرافی در حقـیقت یـک زیـر گـراف القایی از گراف چیدمان است پس با مشـخص شـدن مجمـوعه گـره‌هـایش می‌توان آن را به صورت منحصـر به فردی مشخص ساخت. در هر مرحله از ساختن یک جواب مسئله توسط هر یک از مورچه‌ها نظیر مورچه $k$ توسط گام ۲ الگوریتم "یافتن زیر گراف موجه"، گرهٔ جدید $i$ با احتمال $S_i^k$ کـه از معادلـهٔ زیـر بـه‌دسـت می‌آید برای انتخاب استفاده می‌شود:

$$S_i^k = \begin{cases} \dfrac{\tau_i.[\eta_i]^\alpha}{\sum_{j\in J_k} \tau_j.[\eta_j]^\alpha} & \text{if } i \in J_k \\ 0 & \text{otherwise} \end{cases} \qquad (۱)$$

در ایـن معادلـه $\eta_i$ تابعـی اسـت کـه بـه هر گره تخصیص داده مـی‌شـود و مقـدار آن برابـر است با $W_i / P_i$ که $P_i$ بیانگر محیط دپارتمـان نظیرگرهٔ $i$ و $W_i$ نیز درجهٔ این گره در گراف است. در حقـیقت ایـن نسـبت بیانگـر ایـن نکـته است که هر چقدر گره جدیـد بـا گـره‌هـای بیشتری مجاور باشد بهتر بوده و هر چقدر محـیط بیشـتری داشته باشد به‌دلیل آن‌که می‌تواند بر طول حلقه حاصل بیفزاید از مطلوبیت کمتری برخوردار خواهد بود.

$\tau_i$ نیـز مقـدار فرومون گرهٔ $i$ است. $\alpha$ پارامتری است مثبت کـه بیانگـر اهمـیت $\eta_i$ در مقابـل فـرومون گـره مـی‌باشـد. $J_k$ مجمـوعه گره‌هایی است که حداقل با یکی از گره‌های زیرگراف سـاخته شده مورچهٔ $k$ ام مجاور است و طبق بندهای الف و ب گـام ۲ الگوریـتم انتخاب آنها برای اضافه کردنشان به زیرگراف مجـاز است. با توجه به این معادلهٔ می‌توان گفت گره‌هایی برای انتخاب مناسبترند که مقدار فرومون بیشتری داشته، با تعداد گرهٔ بیشـتری مجـاور باشـند و در عین حال محیط کمتری نیز داشته باشـد. همان‌گونه که دیدیم طبق گام ۲ اگر گره‌ای برای انتخاب وجود نداشت فرایند ساختن جواب توسط این مورچه مجدداً از نو آغاز می‌شود.

زمانی که تمامی مورچه‌ها جواب خود را تکمیل کردند مطلوبیت آن توسط استفاده از قاعدهٔ به هنگام کردن نهایی[۸] اندکی کاهش می‌یابد. این قاعده باعث می‌شود که مطلوبیت گره‌ها به‌صورتی پویا در حال تغییر باشد و از همگرا شدن جوابها در اطراف یک بهینه محلی جلوگیری شود. این قاعده از این اصل طبیعی ناشی می‌شود که همواره مقداری از فرومونی که مورچه‌ها بروی یک مسیر باقی می‌گذارند به‌دلیل تبخیر از بین می‌رود. برای به هنگام کردن فرومون گره‌های موجود در گراف چیدمان از معادلهٔ زیر استفاده می‌کنیم:

$$\tau_i \leftarrow (1-\rho).\tau_i + \rho.\Delta\tau_i \qquad (۲)$$

در این معادله $\Delta\tau_i = \begin{cases} (L_{gb})^{-1} & \text{if } i \in \text{global best Loop} \\ 0 & \text{otherwise} \end{cases}$ است که منظور از $L_{gb}$ طول بهترین حلقه به‌دست آمده تا تکرار کنونی الگوریتم است و $\rho$ نیز به نام پارامتر تبخیر است. هدف از معادلهٔ فوق آن است که اگر گره‌هایی متعلق به بهترین حلقه موجه تا تکرار کنونی نباشد مقدار فرومونش کمترخواهد شد و اگر گره‌هایی متعلق به بهترین حلقه موجه بود اگر چه به دلیل تبخیر مقدار از فرومون آن کم می‌شود اما این مقدار به اندازه سایر گره‌ها نخواهد بود.

از ترکیب الگوریتم یافتن ″زیر گراف موجه″ با قواعد ذکر شده در بالا الگوریتم مورچه‌ای برای حل مسئله SLRP به‌صورت زیر به‌دست می‌آید:

گام ۰: شمارندهٔ تکرار الگوریتم یعنی I و شمارنده تعداد مورچه‌ها یعنی k را برابر ۱ قرار دهید. حداکثر مقدار هریک از آنها را نیز به ترتیب $I_{max}$ و m فرض کنید.

گام ۱: یک گره غیر از گرهٔ صفر را به‌صورت تصادفی از گراف چیدمان برگزینید و آن را برچسب دار کنیدو مورچه k ام را بروی آن قرار دهید. اگر این گره به تمام گره‌های دیگر گراف (غیر از گره صفر) متصل باشد حلقه موجه بروی مرز این دپارتمان قرار دارد و به گام ۴ بروید..

گام ۲: از میان گره‌های غیر برچسب دارنظیر i، گره‌ای را برای حرکت بعدی مورچه k ام انتخاب کنید که در شرایط زیر برای زیر گراف القایی حاصل از گره‌های برچسب دار و این گره صدق کند:

الف : هر گره از این زیر گراف حداقل با یک گره برچسب نخورده مجاور باشد.

ب: اگر از گراف چیدمان این زیر گراف القایی را حذف کنیم همچنان همبند باقی بماند

اگر چنین گره‌ای پیدا نشد مجددا از گام ۱ شروع کنید.

ج: دارای احتمال انتخاب $S_i^k$ با توجه به معادلهٔ ۱ باشد.

گام ۳: گره را به مجموعه گره‌های برچسب دار اضافه کنید و مورچه k ام را بر روی آن قرار دهید و زیر گراف القایی حاصل از گره‌های برچسب دار را با توجه به این گره تعمیم دهید. اگر هر گره غیر برچسب دار مجاور یکی از گره‌های این زیر گراف باشد زیر گراف موجه به‌دست آمده است و در نتیجه حلقه موجه توسط مورچه k ام تکمیل شده است در این‌حالت به گام بعدی بروید. در غیر این‌صورت به گام ۲ برگردید.

گام ۴ :k= k+1 و اگر $k \le m$ شد به گام۱ برگردید. در غیر این‌صورت به گام ۵ بروید.

گام ۵: طول حلقه موجه حاصل در این تکرار را با بهترین حلقه به‌دست آمد تا این تکرار مقایسه کرده و مقدار بهتر را در $L_{gb}$ قرار دهید. فرومون هر گره را با توجه به معادله ۲ بهنگام کرده و شمارنده تعداد تکرارها یعنیi را یک واحد افزایش دهید در صورتی که این مقدار کمتریا مساوی $I_{max}$ بود k =1 قرار داده به گام۱ برگردید. در غیر این‌صورت متوقف شوید بهترین جواب را در بین حلقه‌های موجه تعیین کنید.

الگوریتم فوق را می‌توان در قالب زیر خلاصه کرد:

*Initialize*
*For t=1 to number of cycles do*
*For k=1 to ant-no. do*
*Repeat until ant k make feasible solution*
*Select item i from* $J_k$ *with probability* $P_i^k$ *given by Eq.(1)*
*if item i is not feasible then* $J_k \leftarrow J_k - \{i\}$

*End.*

*Calculate the objective function of the generated solution*

*End.*

*Find the best solution.*

*Make local – optimum.*

*Update the pheromone for each node by Eq. (2)*

*End.*

## ۷- نتایج محاسباتی

برای بررسی کارایی الگوریتم نرم افزاری به زبان برنامه‌نویسی دلفی تهیه شد و بر روی ۳۲ مسئله نمونه آزمایش شد. مسائل نمونه برای انجام این آزمایش مسائلی بودند که در مرجع [۱] نیز مورد استفاده قرار گرفته‌اند به این دلیل که بتوان کارایی الگوریتم را با الگوریتم دیگر حل SLRP سنجید. در جداول زیر نتایج حاصل از اجرای الگوریتم بروی مسائل نمونه و با استفاده از رایانه شخصی Pentium 3 با پردازنده 800 MHZ و 128 مگابایت حافظه اصلی آورده شده است.

قبل از اجـرای الگوریتم ضروری بود که مقادیر پارامترهای موجـود در الگوریـتم تخمـین زده شـود. بـرای این منظور برای هـریک از پارامتـرهای موجـود در الگوریـتم دامـنه تغییـرات بـه‌صـورت $\rho$=[0.05,2.5], m=[5,5],$\alpha$ = [1.0, 3.0 , $I_{max}$=10 در نظـر گـرفته شـد و الگوریـتم بـر روی مسایل نمونه به ازای ترکیبهای مختلفی از مقادیر برای پارامترهای مذکور اجرا شد که در نتـیجه بهتـرین نـتایج بـا توجه به تجزیه و تحلیل حساسیت بروی پارامترهای فوق در هنگام اجرای الگوریتم به‌صورت زیر $\rho$ = 0.1 m =10,, $\alpha$ = 2.0, $I_{max}$ = 10 به‌دست آمد و الگوریتم بـرای هـر نمونه مسئله ۳۰ بار اجرا شده است. در ستونهای این جـدول بـه ترتیب از سمت چپ شماره مسئله (.NO) نام مسئله در مـرجع [۱] (Name) تعـداد دپارتمـان موجـود در مسـئله (.Dept#) زمان اجرای الگوریتم بروی هر مسئله نمونه برحسب ثانـیه(Time) بهتـرین جـواب بـه‌دسـت آمـده در ۳۰ بـار اجـرا (Best Solution) بدتـرین جـواب به‌دست آمده در ۳۰ بار اجرا (Worst Solution) متوسط جواب به‌دست آمده در ۳۰ بار اجرا (Average Solution) و تعـداد اجـراهـایـی که در آن جـواب به‌دست آمده بهینه است (BI) درج شده است.

بـا تـوجه به این نتایج در می‌یابیم که در تمام ۳۲ مسئله نمونه، بهتـرین جـواب حاصـل از الگوریـتم مـورچه‌ای در واقع همان جواب بهینه مسئله است. نکته جالب آن است که به‌طور متوسط در ۲۱ اجرا از ۳۰ اجرای الگوریتم برای هر مسئله نمونه جواب بهیـنه بـه‌دسـت آمـده اسـت و این بدان معنی است که می‌توان انـتظار داشـت کـه جـواب حاصل از یک‌بار اجرای الگوریتم به احـتمال ۷۰٪ جواب بهینه را تولید می‌کند. از طرف دیگر در ۱۳ مسـئله در تمامـی ۳۰ بـار اجرای الگوریتم جواب به‌دست آمده بهینه بوده است .با توجه به این نکته که در مورد مسائل با ۱۰ و ۲۰ دپارتمـان در ۹۴٪ مواردبیش از ۸۰٪ اجراها جواب بهینه حاصل شده است می‌توان نتیجه گرفت که در این نوع از مسائل می‌توان از تعداد اجراهای الگوریتم کاست. بیشترین میزان خطا در جواب حاصل از یک تکرار الگوریتم در مسئله ۳۱ که دارای ۱۸٪ خطـا اسـت اتفـاق افتاده است. اما با توجه به این نکته که میانگـین انحـراف بدتـرین جواب مربوط به هر مسئله نسبت به جواب بهینه از ۴٫۷٪ بیشتر نیست می‌توان نتیجه گرفت که حتی بدتـرین جوابهای حاصل از الگوریتم نیز نزدیک به جواب بهینه اسـت. همچنین متوسط جواب حاصل از اجرای الگوریتم بروی تمـام مسـائل نمـونه در بـازه [۰٪، ۸٪] نسـبت بـه جواب بهینه مسـائل قـرار دارد. متوسـط زمـان اجرای الگوریتم بر روی تمام مسـائل نمـونه برابر ۰٫۶۷۲ ثانیه و به تفکیک هر گروه از مسائل برحسـب دپارتمـان در جـدول ۲ داده شـده است. بدترین زمان اجـرای الگوریتم در کل۹۶۰ اجرای متعلق به۳۲ مسئله در مسئله شـماره ۳۱ بـا ۴۰ دپارتمان اتفاق افتاده است که ۱٫۸۳ ثانیه بوده اسـت کـه بـا تـوجه به ابعاد مسئله زمان بسیار سریعی محسوب مـی‌شود. برای مقایسهٔ کارایی الگوریتم با سایر روشهای موجود بـرای حـل مسئله، الگوریتم مرجع [۱] که حل بهینه مسئله را با اسـتفاده از روشـهای بهیـنه‌سازی انجام می‌دهد انتخاب شد. در جـدول (۲) مقایسـه ایـی مابـین نتایج حاصل از این الگوریتم و الگوریـتم موجـود در مـرجع [۱] داده شـده اسـت. لازم به ذکر اسـت کـه زمـان ارائـه شـده در ایـن جـدول بـرای الگوریـتم مرجـع [۱] مسـتقیما از روی نتایج گزارش شده در این مرجع ذکر

**جدول ۱- نتایج محاسباتی حاصل از اجرای الگوریتم بر روی ۳۲ مسئله نمونه**

| BI | Average | Worst Solution | Best | Time (sec) | #Dept. | Name | NO. |
|---|---|---|---|---|---|---|---|
| 30 | 180 | 180 | 180 | 0.073 | 10 | A1 | 1 |
| 30 | 320 | 320 | 320 | 0.113 | 10 | A2 | 2 |
| 30 | 150 | 150 | 150 | 0.047 | 10 | T1 | 3 |
| 30 | 234 | 234 | 234 | 0.065 | 10 | T2 | 4 |
| 30 | 340 | 240 | 240 | 0.056 | 10 | F1 | 5 |
| 30 | 224 | 224 | 224 | 0.075 | 10 | F2 | 6 |
| 30 | 224 | 224 | 224 | 0.062 | 10 | F3 | 7 |
| 30 | 188 | 188 | 188 | 0.085 | 10 | F4 | 8 |
| 20 | 425.5 | 452 | 422 | 0.325 | 20 | A10T20 | 9 |
| 30 | 490 | 490 | 490 | 0.401 | 20 | A20T10 | 10 |
| 30 | 373 | 373 | 373 | 0.267 | 20 | F10F20 | 11 |
| 25 | 396.5 | 404 | 396 | 0.295 | 20 | F30F40 | 12 |
| 3 | 416.8 | 426 | 402 | 0.326 | 20 | A10F20 | 13 |
| 30 | 504 | 504 | 504 | 0.433 | 20 | A20F40 | 14 |
| 30 | 430 | 430 | 430 | 0.274 | 20 | T10F10 | 15 |
| 29 | 400.6 | 434 | 398 | 0.329 | 20 | T20F30 | 16 |
| 26 | 600.7 | 658 | 592 | 0.850 | 30 | A10T20F30 | 17 |
| 30 | 580 | 580 | 580 | 0.871 | 30 | F40A20T10 | 18 |
| 12 | 592.8 | 612 | 580 | 0.727 | 30 | F10A10T20 | 19 |
| 4 | 598.3 | 610 | 580 | 0.895 | 30 | F20A20T10 | 20 |
| 27 | 530 | 554 | 526 | 0.717 | 30 | F30A10T20 | 21 |
| 29 | 602 | 752 | 598 | 0.649 | 30 | F40T10A20 | 22 |
| 26 | 588 | 598 | 584 | 0.875 | 30 | F10F20F30 | 23 |
| 7 | 536 | 538 | 534 | 0.700 | 30 | F20F30F41 | 24 |
| 24 | 675.4 | 750 | 664 | 1.451 | 40 | A10F20T10T20 | 25 |
| 3 | 747 | 784 | 710 | 1.445 | 40 | F10F20 F30F40 | 26 |
| 1 | 701 | 725 | 680 | 1.379 | 40 | F20F31 F42F13 | 27 |
| 17 | 698 | 732 | 680 | 1.549 | 40 | F30F41 F12F23 | 28 |
| 2 | 757 | 786 | 728 | 1.435 | 40 | F40F11 F22F33 | 29 |
| 15 | 655.6 | 684 | 642 | 1.470 | 40 | A10T10 F10F30 | 30 |
| 10 | 694.3 | 756 | 640 | 1.682 | 40 | T20A10 F20F40 | 31 |
| 4 | 833.4 | 892 | 770 | 1.626 | 40 | A10F10A20F20 | 32 |

**جدول ۲- مقایسهٔ نتایج الگوریتم مورچه‌ای و الگوریتم مرجع [۱]**

| | | | | |
|---|---|---|---|---|
| تعداد دپارتمانهای موجود در مسئله | ۱۰ | ۲۰ | ۳۰ | ۴۰ |
| متوسط زمان حل در الگوریتم مرجع [۱] (ثانیه) | ۱ | ۳ | ۱۲ | ۳۳ |
| متوسط زمان حل توسط الگوریتم مورچه‌ای (ثانیه) | ۰٫۰۷ | ۰٫۳۳۱ | ۰٫۷۸۵ | ۱٫۵۰۴ |

**جدول ۳- نتایج حاصل از اجرای الگوریتم مورچه‌ای بر روی مسائل بزرگ**

| | | | | |
|---|---|---|---|---|
| تعداد دپارتمانهای موجود در مسئله | ۸۰ | ۱۲۰ | ۱۶۰ | ۲۰۰ |
| تعداد مسئله نمونه حل شده | ۴ | ۳ | ۲ | ۱ |
| متوسط زمان حل توسط الگوریتم مورچه‌ای (ثانیه) | ۵۰ | ۳۱۴ | ۹۳۴ | ۱۴۲۰ |

شـده اسـت و حاصـل از اجرای الگوریتم مذکور بر روی شبکه پیشـرفته SUN SPARC STATION 20, 150 MHz, 64 MB است که همان‌گونه که مشخص است با افزایش تعداد دپارتمانها هـم بـر زمـان اجـرای الگوریتم مرجع [۱] افزوده می‌شود و هم بـرای مسائل با بیش از ۴۰ دپارتمان به‌دلیل افزایش ابعاد مسئله، الگوریـتم مذکـور قـادر نیسـت جـواب بهیـنه را تعیین کند. اما اشـکال زمـان حـل در الگوریـتم مورچه‌ای بسیار کمتر به چشم مـی‌خـورد و علیـرغم اجـرای الگوریتم بر روی رایانه شخصی، زمـان اجـرای آن سـریعتر از الگوریـتم مرجع است. همچنین با تـوجه بـه بررسیهای به‌عمل آمده الگوریتم مورچه‌ای قادر است بـرای مسـائل بـزرگ نیز در زمان قابل قبول حل تقریبی را ارائه کـند کـه این موضوع با اجرای الگوریتم بروی ۱۰ مسئله بزرگ بـا ابعـاد مخـتلف که به‌صورت تصادفی تولید شدند بررسی شد که نتایج آن در جدول (۳) آمده است.

لازم به ذکر است که برای مسائل عملی حداکثر دپارتمان مورد بررسی معمولا کمتر از ۱۰۰ دپارتمان است که با توجه به آن می‌توان گفت که الگوریتم مورچه‌ای قادر است جواب نزدیک به بهینه را برای چنین مسائلی در زمان کوتاهی تولید کند.

## واژه‌نامه

1. automated guided vehicles
2. single loop routing problem
3. ant colony optimization
4. pheromone
5. layout graph
6. planar graph
7. induced subgraph
8. global updating rule

## مراجع


1. Asef-Vaziri, A., Laporte, G., and Sriskandarajah, C., "The Block Layout Shortest Loop Design Problem," *IIE Transactions on Computers and Intractability.*, Vol. 32, pp. 727-734, 2000.
2. Askin, R. G., and Standridge, C. R., *Modeling and Analysis of Manufacturing Systems,* pp. 232-235, Wiley, 1993.
3. De Duzman, M. C., Prabhu, N., and Tanchoco, J. M. A., "Complexity of the AGV Shortest Path and Single Loop Guide Path Layout Problem," *International Journal of Production Research* , Vol. 8, pp. 2083-2091,1997.
4. Dorigo, M., and Di Caro, G., "Ant Colony Optimization: A New Meta-Heuristic," *Proceedings of the 1999 Congress on Evolutionary Computation,* 1999.
5. Dorigo, M., and Gambardella, L. M., "Ant Colony System: A Cooperative Learning Approach to the Traveling Salesman Problem," *IEEE Transactions on Evolutionary Computation,* Vol. 1, pp. 53-66, 1997.
6. Dorigo, M., Maniezzo, V., and Colorni, A., "Positive Feedback as a Search Strategy," Technical Report 91-016, University of Milan, Italy,1991.
7. Laporte, G., Asef-Vaziri, A., and Sriskandarajah, C., "Some Applications of the Generalized Traveling Salesman Problem," *Journal of the Operational Research Society,* Vol. 47, pp. 1461-1467, 1996.
8. Sinrich, D., "Design and Evaluation of Optimal Single Loop Guide Paths for Automated Guided Vehicle Systems," M.S.I.E Thesis, Purdue University, Indiana, 1990.
9. Sinriech, D., and Tanchoco, J.M.A., "Impact of Empty Vehicle Flow on Performance of Single-Loop AGV Systems," *International Journal of Production Research* , Vol. 30, pp. 2237-2252, 1992.
10. Sinriech, D., and Tanchoco, J.M.A., "Solution Methods for the Mathematical Models of Single-Loop AGV Systems," *International Journal of Production Research* , Vol. 31, pp.705-723, 1993.
11. Stützle, T., and Hoos, H. H., "Max-Min Ant System," *Future Generation Computer Systems*, Vol. 16, pp. 889-914, 2000.