

# استفاده از کولونی مورچه ها به صورت معکوس به منظور افزایش قدرت کاوش و جستجوی عامل ها در سیستم شبیه ساز امداد

مصطفی اصغری<sup>۱</sup>، بهروز معصومی<sup>۲</sup>، محمدرضا میبدی<sup>۳</sup>

## چکیده

یکی از بسترهای مناسب برای تست و ارزیابی ایده ها و تکنیک های مربوط به سیستم های چند عامله، سیستم شبیه ساز امداد می باشد. در سیستم شبیه ساز امداد، وقوع یک حادثه مخرب شبیه سازی شده و عامل های نرم افزاری برای مقابله با اثرات حادثه در محیط مصنوعی فعالیت می کنند. وظیفه عامل های پلیس، پاکسازی و باز کردن مسیرهای مسدود شده است تا عامل های دیگر بتوانند به امداد و نجات مصدومین بپردازند. در این مقاله الگوریتمی ارائه شده است که از کولونی مورچه ها به صورت معکوس استفاده نموده و قدرت کاوش بیشتری را به عامل ها می دهد. با استفاده از الگوریتم ارائه شده، عامل های پلیس می توانند در زمان کمتری فضای بیشتری را جستجو کنند. آزمایشات و شبیه سازی های انجام شده نشان می دهند که الگوریتم ارائه شده از نظر جستجو و کاوش در محیط هایی که اطلاعات آنها در دسترس نیست، عملکرد بهتری نسبت به الگوریتم های قبلی دارد.

**کلمات کلیدی:** جستجو و کاوش، کولونی مورچه ها، شبیه ساز امداد

## A New Repelling Ant Colony Based Algorithm for Improving the Search And Exploration Power of Rescue Agents In Rescue Simulation System

Mostafa Asghari, Behrouz Masoumi, Mohammad Reza Meybodi

### Abstract

RoboCup Rescue Simulation System is a suitable test-bed for test and evaluation of multiagent system's related ideas and techniques. In this simulation system, with the start of simulation, police agents should search and explore the earthquake area and open the blocked roads so that the ambulance and fire brigades be able to rescue the injured civilians and to extinguish the burning buildings. In this paper, a new repelling ant colony based algorithm was proposed which gives more exploration power to rescue agents. Results of experiments shows that the proposed algorithm can explore large area at the same time in comparison with other algorithms.

### ۱. مقدمه

یکی از چالش هایی که انسان امروزه با آن مواجه است افزودن قابلیت یادگیری و هوش مصنوعی به ماشین ها می باشد. هدف نهایی، پرورش روباتهای هوشمند و مستقلی است که بتوانند در کنار انسان و یا در بعضی جاها به جای انسان به ایفای نقش بپردازند. پرورش روباتها از دو جنبه مورد نظر است: جنبه فیزیکی و ظاهری و دیگری از نظر هوشی و ذهنی. در این راستا همه ساله مسابقات روباتیک برگزار شده و محققین نتایج تحقیقات خود را در قالب تیم های مختلف ارائه نموده و ارزیابی می کنند [۹]. هدف از پروژه شبیه ساز امداد به عنوان یکی از زیرمجموعه های روباتیک، پرورش روباتها از نظر ذهنی و هوشی می باشد. در سیستم شبیه ساز امداد، یک حادثه مخرب مانند زلزله شبیه سازی شده و عامل های نرم افزاری امداد و نجات برای مقابله با اثرات آن برنامه ریزی می شوند [۱۰]. در این سیستم سه نوع عامل وجود دارند که در محیط حادثه به امداد و نجات می پردازند: عامل های پلیس، عامل های آمبولانس و عامل های آتش نشان. وظیفه عامل های پلیس جستجو در محیط

۱- دانشگاه آزاد اسلامی واحد میاندوآب، m\_asghary86@yahoo.com

۲- دانشکده مهندسی برق، کامپیوتر و فناوری اطلاعات، دانشگاه آزاد اسلامی قزوین، قزوین، bmasoumi@Qazviniau.ir

۳- دانشکده مهندسی کامپیوتر و فناوری اطلاعات، دانشگاه صنعتی امیرکبیر، تهران، ایران، mmeybodi@aut.ac.ir

و باز کردن خیابانهای مسدود شده است تا عامل های دیگر در امداد رسانی به شهروندان با مانع روبرو نشوند. وظیفه عامل های آمبولانس حمل مصدومین به پناهگاه می باشد. عامل های آتش نشان وظیفه خاموش کردن آتش را بر عهده دارند. عامل های آمبولانس و آتش نشان زمانی کارایی مناسبی خواهند داشت که راههای مورد نیاز آنها در کوتاهترین زمان توسط عامل های پلیس پاکسازی شوند. هر چه عامل های پلیس بتوانند فضای بیشتری را جستجو و پاکسازی کنند به همان اندازه آمبولانس ها و آتش نشانها در انجام وظایف خودشان موفق تر خواهند بود. بنابراین یکی از مواردی که باید در مورد عامل های پلیس در نظر گرفته شود این است که بتوانند در زمان کمتری فضای بیشتری را جستجو کنند [۷] و [۸]. با توجه به اینکه در سیستم شبیه ساز امداد برای برقراری ارتباط بین عامل ها، از نظر پهنای باند محدودیت وجود دارد مناسب است از روش هایی استفاده شود که ارتباطات بین عامل ها را به حداقل برسانند. در این مقاله برای جستجو و کاوش عامل های پلیس در محیط حادثه، الگوریتمی ارائه شده است که از کولونی مورچه ها به صورت معکوس استفاده نموده و این قابلیت را به عامل های پلیس می دهد که بتوانند با حداقل ارتباطات، در زمان کمتری فضای بیشتری را جستجو و کاوش کنند.

بخش بندی بقیه مطالب این مقاله به صورت زیر است: ابتدا در بخش ۲ کولونی مورچه ها به طور مختصر و خلاصه معرفی می شود. بخش ۳ به معرفی الگوریتم ارائه شده می پردازد. در بخش ۴ الگوریتم ارائه شده ارزیابی شده است. بخش ۵ مقایسه و نتیجه گیری می باشد.

## ۲. بهینه سازی با کولونی مورچه ها

کولونی مورچه ها الهام گرفتن از حل مسائل موجود در طبیعت توسط حس غریزی حشرات (مورچه ها) می باشد که ابتدا توسط Marko Dorigo مطرح گردید [۳] و [۴]. کولونی مورچه ها یکی از جالبترین سیستمهای مورد مطالعه تاکنون بوده که کاربردهای فراوانی در علوم و مهندسی یافته است. یافتن کوتاهترین مسیر یک مسئله بهینه سازیست (Optimization) که گاه حل آن بسیار دشوار و گاه نیز بسیار زمانبر است. مورچه های واقعی قادر به یافتن کوتاهترین مسیر از منبع غذا تا لانه هایشان هستند که این کار بدون استفاده از قوه بینایی انجام می گیرد [۱]. راه حل مورچه ها برای پیدا کردن کوتاهترین مسیر بین لانه و منبع غذا به صورت زیر است: مورچه ها هنگام راه رفتن بر روی زمین، از خود ردی از فرمون (pheromone) به جا می گذارند. فرمون با گذشت زمان تبخیر می شود اما مدتی به عنوان رد مورچه بر سطح زمین باقی می ماند. یک رفتار بسیار ساده پایه ای در مورچه ها وجود دارد: آنها یک مسیر را از بین چندین مسیر بر اساس احتمال انتخاب می کنند. مسیری که فرمون بیشتری داشته باشد یا به عبارت دیگر مورچه های بیشتری قبلاً از آن عبور کرده باشند احتمال بیشتری برای انتخاب شدن خواهد داشت. در شروع کار انتخاب مسیر کاملاً تصادفی است اما بعد از اینکه در هر یک از مسیرها ردی از فرمون ایجاد شد مورچه ها به احتمال بیشتر مسیر با فرمون بیشتر را انتخاب می کنند (هر چه فرمون بیشتر باشد احتمال انتخاب مسیر بیشتر است). به همین دلیل پس از مدتی همه مورچه ها (یا تقریباً همه مورچه ها) مسیر کوتاه تر را طی خواهند کرد.

### - مکانیزم انتخاب مسیر در الگوریتم های کولونی مورچه ها

مورچه ها برای انتخاب گره بعدی از مقادیر فرمون ها و فاصله بین گره ها استفاده می کنند. هر چه فاصله بین گره جاری تا یک گره کمتر و مقدار فرمون بین آنها بیشتر باشد، احتمال انتخاب گره بیشتر می شود. به عبارت دیگر فرمول انتخاب گره بعدی رابطه مستقیمی با مقدار فرمون روی یال و عکس فاصله بین گره ها دارد. عکس فاصله بین دو گره را دید می نامیم و آنرا  $\eta_{ij} = 1/d_{ij}$  تعریف می کنیم. احتمال عبور از گره  $i$  به گره  $j$  طبق رابطه (۱) تعریف می شود. پارامترهای  $\alpha$  و  $\beta$  به کاربر اجازه می دهند بر روی نسبت اهمیت مقدار دنباله فرمون به دید، کنترل

$$p_{ij}(t) = \frac{[\tau_{ij}(t)]^\alpha \cdot [\eta_{ij}]^\beta}{\sum_{j_k=1}^n [\tau_{ij_k}(t)]^\alpha \cdot [\eta_{ij_k}]^\beta} \quad \text{فرمول (۱)} \quad \text{داشته باشد.}$$

### - بهنگام سازی فرمون در کولونی مورچه ها

بعد از اینکه هر مورچه یک دور تولید کرد مقدار دنباله فرمون روی یال ها بهنگام می شود. در کولونی مورچه ها این عمل ابتدا با کاهش مقدار فرمون با یک فاکتور ثابت (ضریب تبخیر) و سپس قرار دادن فرمون توسط هر مورچه روی یال هایی که در دور خود پیموده انجام می شود. ضریب تبخیر که مقداری بین صفر و یک دارد برای اجتناب از انباشتن بی حد دنباله فرمون است و الگوریتم را قادر می سازد تا تصمیم های بدی را که قبلاً گرفته شده است فراموش کند. روی یال هایی که توسط مورچه ها انتخاب نشده اند شدت فرمون با تعداد تکرارها بصورت نمایی کاهش می یابد. الگوریتم به مرور مورچه های بیشتری را در طول مسیرهای کوتاه تر هدایت می کند و در نتیجه مقدار فرمون قرار داده شده روی این مسیرها بالا خواهد رفت که این باعث افزایش احتمال انتخاب مسیرهای کوتاه تر در آینده خواهد شد.  $\tau_{ij}(t+1)$  را شدت دنباله روی یال بین گره های  $i$  و  $j$  در لحظه  $t+1$  می نامیم که با فرمول (۲) محاسبه می شود.

$$\tau_{ij}(t+1) = \rho \cdot \tau_{ij}(t) + \Delta \tau_{ij}(t, t+1) \quad \text{فرمول (۲)}$$

در این فرمول  $\rho$  ضریب تبخیر و  $\sum_{k=1}^m \Delta \tau_{ij}(t, t+1)$  که  $\Delta \tau_{ij}^k(t, t+1)$  مقدار دنباله فرمون است که روی یال بین گره‌های  $i$  و  $j$  توسط مورچه  $k$  در زمان بین  $t$  و  $t+1$  قرار داده می‌شود. مقدار دنباله فرمون در زمان  $t$ ،  $\tau_{ij}(0)$ ، می‌تواند مقداری تصادفی باشد (عموماً مقداری کوچک). انتخاب‌های مختلفی در رابطه با چگونگی محاسبه  $\Delta \tau_{ij}^k(t, t+1)$  و زمان بهنگام سازی  $\tau_{ij}(t)$  وجود دارد که منجر به نسخه‌های مختلفی از الگوریتم کولونی مورچه‌ای می‌گردد.

### ۳. جستجو و کاوش با استفاده از کولونی مورچه‌ها

الگوریتم ارائه شده برای جستجو و کاوش در سیستم شبیه ساز امداد، از کولونی مورچه‌ها به صورت معکوس استفاده می‌کند، بدین صورت که ابتدا بر روی تمام خیابانها فرمون اولیه قرار داده می‌شود. عامل پلیس از هر خیابان که عبور می‌کند فرمون آن را به صورت ضربی از پارامتر  $\square$  کاهش می‌دهد. با گذشت زمان، بر مقدار فرمون روی خیابانها افزوده می‌شود. برای جلوگیری از انباشته شدن بیش از حد فرمون و یا کاهش بیش از حد آن در خیابانها، برای مقدار فرمون یک خیابان، مقدار حداقل و حداکثر در نظر گرفته شده است. عامل پلیس در گرهی که بیش از یک یال مجاور دارد یکی از یال‌های مجاور را بر اساس مکانیزم انتخاب مسیر کولونی مورچه‌ها، فرمول شماره (۱)، انتخاب می‌کند. در انتخاب مسیر براساس کولونی مورچه‌ها، دو عامل در احتمال انتخاب یک یال دخالت دارد: عکس طول یال که بنام دید شناخته می‌شود و مقدار فرمون انباشته شده بر روی یال. در این الگوریتم برای جلوگیری از افزایش بیش از حد اختلاف با کوتاهترین مسیر از عکس فاصله (دید) و برای جلوگیری از حلقه و تکرار از مقدار فرمون استفاده می‌شود و سعی می‌شود با تنظیم مناسب ضرایب دو پارامتر فوق علاوه بر جستجو و کاوش بیشتر در محیط، فاصله با کوتاهترین مسیر نیز حداقل گردد. هر چه طول یک یال کمتر باشد عکس طول یا دید آن بیشتر شده و احتمال انتخاب یال را بالا می‌برد. یال‌های با طول بیشتر نیز احتمال انتخاب کمتری دارند. از آنجا که احتمال انتخاب یک یال با مقدار فرمون آن یال رابطه مستقیمی دارد هر چه مقدار فرمون یک یال بیشتر باشد احتمال انتخاب آن یال بیشتر است. عامل پلیس با عبور از هر یال و کاهش فرمون آن، احتمال انتخاب آن را در آینده کاهش می‌دهد. انتخاب یک یال از بین دو یا چندین یال بر اساس فرمول (۱) انجام می‌شود. عامل پلیس، هر یالی را که مطابق فرمول شماره (۱) انتخاب می‌کند فرمون آن را مطابق فرمول (۳) کاهش می‌دهد:

$$t_{ij}(t+1) = t_{ij}(t) - \rho t_{ij}(t) = (1 - \rho) t_{ij}(t) \quad \text{فرمول (۳)}$$

با توجه به اینکه احتمال انتخاب یک یال، رابطه مستقیمی با مقدار فرمون روی آن دارد هر چه فرمون روی یال کمتر شود به همان نسبت نیز احتمال انتخاب یال کمتر می‌شود. یال‌هایی که انتخاب نشده اند نسبت به یال‌های انتخاب شده دارای فرمون بیشتری هستند از اینرو احتمال انتخاب آنها در آینده نسبت به یال‌های انتخاب شده بیشتر خواهد بود. چنانکه در فرمول‌ها نیز مشخص است میزان فرمون دارای پارامتری بنام  $\square$  و عکس فاصله دارای پارامتری بنام  $\square$  و تبخیر فرمون با گذشت زمان دارای پارامتری بنام  $\square$  می‌باشد. دو پارامتر اول برای بیان اهمیت عکس فاصله و مقدار فرمون نسبت به یکدیگر کاربرد دارند. اگر در تصمیم‌گیری، میزان فرمون مسیر مهم‌تر از طول مسیر باشد می‌توان با افزایش مقدار  $\square$  آن را پیاده‌سازی نمود ولی اگر طول مسیر مهم‌تر از مقدار فرمون باشد می‌توان با افزایش مقدار  $\square$  به آن دست یافت. با توجه به اینکه در این الگوریتم با عبور از هر یال، از مقدار فرمون آن کاسته می‌شود مقدار پارامتر  $\square$  باید در فاصله  $[0, \infty]$  باشد. مقادیر بزرگتر از ۱ باعث منفی شدن مقدار فرمون و مقادیر کوچکتر از ۰ باعث افزایش مقدار فرمون می‌شوند. آنچه که مهم است تعیین مناسب مقادیر پارامترهای  $\square$  و  $\square$  می‌باشد که در این الگوریتم برای تعیین مقادیر آنها از روش آزمایش و خطا استفاده شده است. شبه کد انتخاب گره بعدی ( $V_{i+1}$ ) برای  $V_i$  در این الگوریتم به صورت زیر است:

```

Initialize the Road's Pheromone;
Initialize The Parameters  $\alpha, \beta, \rho, \tau$ ;
For each Edge  $E_j$  in Vertex  $i$ 
    Compute the probability value according to (۱);
Select one of the edges  $E_j$  in vertex  $V_i$  randomly;
For each edge  $E_j$  in vertex  $V_i$ 
    If edge  $E_j$  has been selected
        Decrease  $E_j$ 's pheromone according to (۳);
Increase all edges's pheromone over the time.

```

شکل ۱- الگوریتم انتخاب گره بعدی در الگوریتم ارائه شده برای فاز جستجوی عامل پلیس

#### ۴. ارزیابی الگوریتم ارائه شده

چنانکه در بخش قبل نیز بیان شد در چنین روش هایی، مقادیر پارامترها را معمولاً با روش آزمایش و خطا تنظیم می کنند. بدین صورت که مسئله را با مقادیر متفاوت پارامتر حل نموده و مقداری را برای پارامتر در نظر می گیرند که دارای مناسب ترین جواب است. چگونگی تنظیم مقادیر پارامترهای کولونی مورچه ها در این مسئله در بخش بعد بیان شده است.

##### - تنظیم پارامتر $\alpha$

برای پیدا کردن مقدار مناسب برای پارامترهای کولونی مورچه ها در این مسئله، آزمایش هایی انجام شده است که نتایج آنها در جداول زیر ارائه شده است. برای ارزیابی عملکرد الگوریتم در فواصل مختلف زمانی، کل زمان شبیه سازی (۳۰۰ ثانیه) به چندین بازه زمانی تقسیم شده و تعداد گره هایی که عامل پلیس توانسته است به ازای مقادیر مختلف پارامترها در هر بازه زمانی ملاقات کند یادداشت شده است. جدول ۱ نتایج شبیه سازی با مقادیر مختلف برای پارامتر  $\alpha$  و تعداد گره های ملاقات شده توسط عامل پلیس را نشان می دهد.

جدول ۱- مقادیر مختلف پارامتر  $\alpha$  و تاثیر آن در تعداد گرههای ملاقات شده توسط عامل پلیس در زمان های مختلف

برای کم کردن فاصله زمانی بین انتخاب هدف توسط عامل پلیس و رسیدن به آن، از میانگین زمان انتظار اهداف استفاده شده است که برابر میانگین زمان هایی است که از لحظه انتخاب هدف توسط عامل پلیس تا رسیدن به آن هدف سپری می شود. این مقدار برای تمام اهدافی که توسط عامل پلیس انتخاب شده اند یادداشت شده و میانگین آنها محاسبه شده است. جدول ۲ تاثیر مقادیر مختلف پارامتر  $\beta$  بر میانگین زمان انتظار اهداف را نشان می دهد:

جدول ۲- مقادیر مختلف پارامتر  $\beta$  و تاثیر آن در میانگین زمان انتظار اهداف انتخاب شده توسط عامل پلیس

$\beta = 0.8$	$\beta = 0.7$	$\beta = 0.6$	$\beta = 0.5$	$\beta = 0.4$	$\beta = 0.3$	$\beta = 0.2$	$\beta = 0.1$	
۲۵۰	۱۴۶	۶۵	۲۵۰	۹۵	۳۶	۶۶	۹۱	میانگین زمان انتظار اهداف ( بر حسب ثانیه )

با ملاحظه داده های جدول های فوق می توان نتیجه گرفت مناسب ترین مقداری که می توان برای پارامتر  $\beta$  در نظر گرفت به طوری که درجه کاوش مناسبی داشته باشد و میانگین زمان انتظار اهداف را نیز کم کند  $\beta = 0.3$  است.

##### - تنظیم پارامتر $\beta$

برای تنظیم پارامتر  $\beta$  مانند پارامتر  $\alpha$  عمل شده و تاثیر آن در عملکرد عامل پلیس در زمانهای مختلف یادداشت شده است. جدول ۳ تاثیر مقادیر مختلف  $\beta$  در عملکرد الگوریتم در فاصله های زمانی مختلف را نشان می دهد:

جدول ۳- مقادیر مختلف پارامتر  $\alpha$  و تاثیر آن در تعداد گرههای ملاقات شده توسط عامل پلیس

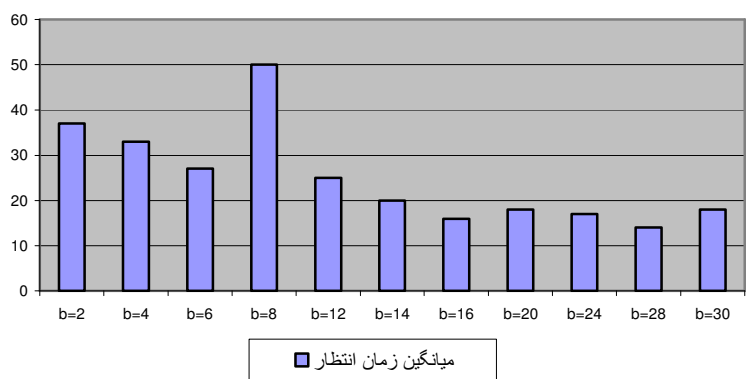
زمان	$\alpha=2$	$\alpha=4$	$\alpha=6$	$\alpha=8$	$\alpha=12$	$\alpha=14$	$\alpha=16$	$\alpha=20$	$\alpha=24$	$\alpha=28$	$\alpha=30$
۰ - ۱۵	۱۸۶	۱۸۹	۱۲۰	۱۸۲	۱۱۸	۳۱	۱۲۰	۸۵	۱۰۳	۲۱۳	۳۱
۱۵ - ۳۰	۲۰۵	۲۲۱	۱۷۹	۲۴۱	۲۱۹	۱۲۹	۱۲۳	۱۰۳	۱۵۴	۲۴۹	۱۲۹
۳۰ - ۴۵	۲۰۷	۲۲۲	۱۸۳	۲۵۶	۲۲۸	۱۷۴	۱۲۹	۲۲۸	۱۶۴	۲۵۷	۱۷۴
۴۵ - ۶۰	۲۳۳	۲۲۲	۱۹۱	۲۵۷	۲۳۸	۱۸۱	۲۶۰	۲۳۶	۱۹۷	۲۵۸	۱۸۱
۶۰ - ۱۰۰	۲۶۴	۲۳۱	۲۱۰	۲۵۷	۲۷۹	۱۹۹	۲۸۷	۲۶۷	۲۳۵	۲۷۱	۱۹۹
۱۰۰ - ۱۵۰	۲۹۰	۲۳۵	۲۵۹	۲۵۷	۲۹۱	۲۱۴	۲۸۸	۲۷۵	۲۵۵	۲۷۱	۲۱۴
۱۵۰ - ۲۰۰	۲۹۱	۲۷۱	۲۵۹	۲۶۸	۲۹۱	۲۸۸	۲۸۹	۲۷۵	۲۷۰	۲۷۲	۲۸۸
۲۰۰ - ۲۵۰	۲۹۱	۲۸۳	۲۵۹	۲۷۶	۲۹۱	۲۸۸	۲۸۹	۲۸۰	۲۷۱	۲۸۴	۲۸۸
۲۵۰ - ۳۰۰	۲۹۱	۲۸۳	۲۷۷	۲۹۱	۲۹۱	۲۹۱	۲۸۹	۲۸۸	۲۸۸	۲۸۵	۲۹۱

برای بررسی تاثیر مقادیر مختلف پارامتر  $\alpha$  در میانگین زمان انتظار اهداف به منظور انتخاب مقدار مناسب برای این پارامتر، شبیه سازی های مختلفی انجام شده و در آنها میانگین زمان انتظار اهداف به ازای مقادیر مختلف  $\alpha$  یادداشت شده است. این نتایج در جدول ۴ نشان داده شده اند:

جدول ۴- مقادیر مختلف پارامتر  $\alpha$  و تاثیر آن در میانگین زمان انتظار اهداف انتخاب شده توسط عامل پلیس

	$\alpha=2$	$\alpha=4$	$\alpha=6$	$\alpha=8$	$\alpha=12$	$\alpha=14$	$\alpha=16$	$\alpha=20$	$\alpha=24$	$\alpha=28$	$\alpha=30$
میانگین زمان انتظار	۳۷	۳۳	۲۷	۵۰	۲۵	۲۰	۱۴	۱۸	۱۷	۱۶	۱۸

برای مقایسه بهتر تاثیر مقادیر مختلف پارامتر  $\alpha$  بر میانگین زمان انتظار اهداف، داده های جدول ۴ به صورت نمودار شکل ۲ نشان داده شده است:



شکل ۲- تاثیر مقادیر مختلف پارامتر  $\alpha$  بر میانگین زمان انتظار اهداف انتخاب شده توسط عامل پلیس

چنانکه قبلا نیز بیان شد هدف از جداول و نمودارهای بالا، نمایش تاثیر مقادیر مختلف پارامترها بر عملکرد الگوریتم کولونی مورچه ها در این مسئله و انتخاب مقادیر مناسب برای پارامترها می باشد که تجزیه و تحلیل آنها مقادیر زیر را به عنوان مقادیر مناسب پارامترها در این مسئله نتیجه می دهد:

$$\alpha = 16$$

$$\beta = 1$$

$$\rho = 0.3$$

$$\text{نرخ تبخیر} = 0.1$$

شکل ۳- مقادیر مناسب برای پارامترهای الگوریتم کولونی مورچه ها در مسئله جستجو و کاوش در سیستم شبیه ساز امداد

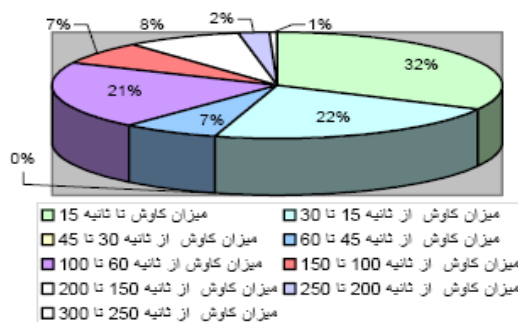
## ارزیابی الگوریتم ارائه شده با مقادیر تنظیم شده پارامترها

پس از به دست آوردن مقادیر مناسب برای پارامترها، شبیه سازی با مقادیر جدید پارامترها انجام شده است. برای شبیه سازی از دو نقشه Kobe و VC استفاده شده است. نتایج شبیه سازی در جدول ۵ بیان شده است:

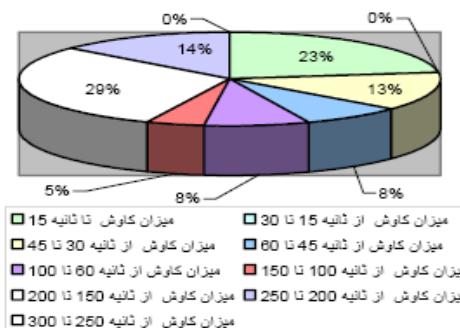
جدول ۵- تعداد گره های ملاقات شده توسط عامل پلیس در الگوریتم کولونی مورچه ها در زمانهای مختلف

زمان	۰-۱۵	۱۵-۳۰	۳۰-۴۵	۴۵-۶۰	۶۰-۱۰۰	۱۰۰-۱۵۰	۱۵۰-۲۰۰	۲۰۰-۲۵۰	۲۵۰-۳۰۰
Kobe	۹۴	۶۴	۰	۲۰	۶۰	۲۰	۲۴	۶	۲
VC	۳۰۴	۰	۱۲۰	۷۰	۷۲	۴۱	۲۶۹	۱۲۲	۰

نمودارهای شکل های ۴ و ۵ که از داده های جدول ۵ بدست آمده اند، میزان کاوش الگوریتم را به ترتیب در نقشه های Kobe و VC در زمان های مختلف نشان می دهند:



شکل ۴- میزان کاوش عامل پلیس در الگوریتم ارائه شده - نقشه Kobe



نمودار ۵- میزان کاوش عامل پلیس در الگوریتم ارائه شده - نقشه VC

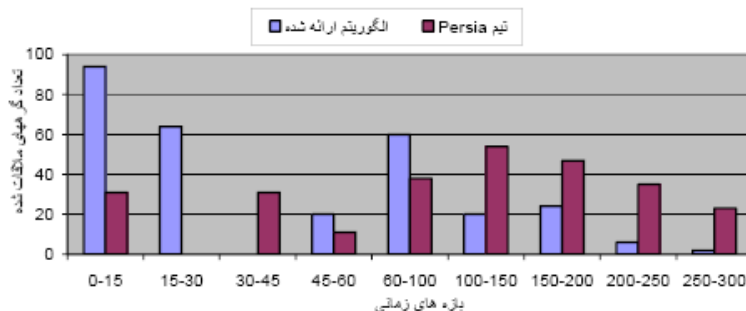
برای مقایسه عملکرد الگوریتم ارائه شده از نظر امتیاز نهایی کسب شده، شبیه سازی با الگوریتم ارائه شده و الگوریتم تیم Persia به تعداد ۴ بار انجام شده است. نتایج شبیه سازی ها در جدول ۶ نشان داده شده است:

جدول ۶- امتیاز کسب شده توسط الگوریتم ترکیبی و تیم Persia در ۴ بار اجرا

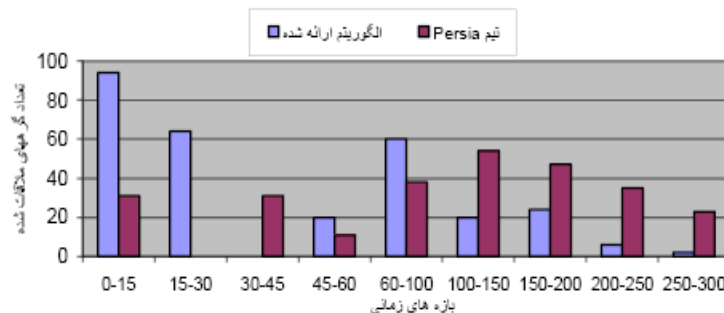
ردیف	۱	۲	۳	۴	میانگین
الگوریتم ارائه شده	۲۸.۳۷	۲۸.۴۵	۲۹.۲۲	۲۹.۸۴	۲۸.۹۷
الگوریتم تیم Persia (کوته ترین مسیر)	۲۷.۹۶	۳۰.۰۱	۲۷.۶۳	۲۹.۲۹	۲۸.۷۲

## ۵. مقایسه و نتیجه گیری

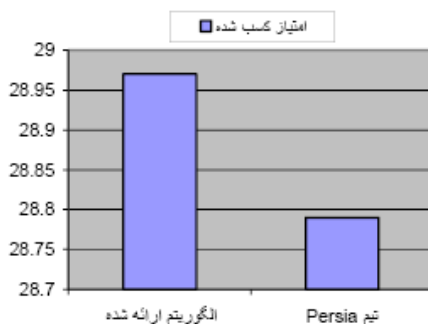
برای پیاده سازی عامل های پلیس، از کد تیم Persia استفاده شده است [۵] و [۶]. این تیم (Persia) نمونه ای از تیم هایی است که در آن عامل پلیس برای حرکت به سوی اهداف خود و همچنین در فاز جستجو از الگوریتم کوتاهترین مسیر (دایجسترا) استفاده می کند. مقایسه عملکرد عامل های پلیس که برای کاوش و جستجو در محیط شبیه ساز امداد از الگوریتم کولونی مورچه ها استفاده می کنند با عملکرد عامل های پلیس تیم Persia می تواند معیار مناسبی برای ارزیابی الگوریتم ارائه شده باشد. نمودارهای نشان داده شده در شکل های ۶ و ۷ الگوریتم ارائه شده را از نظر میزان کاوش و جستجو با تیم Persia به ترتیب در نقشه های Kobe و VC مقایسه می کند. نمودار شکل ۸ نیز مقایسه الگوریتم ارائه شده با الگوریتم تیم Persia از نظر امتیاز نهایی می باشد.



شکل ۶- مقایسه تعداد گره های ملاقات شده توسط الگوریتم ارائه شده و الگوریتم تیم Persia در زمان های مختلف- نقشه Kobe



شکل ۷- مقایسه تعداد گره های ملاقات شده توسط الگوریتم ارائه شده و الگوریتم تیم Persia در زمان های مختلف - نقشه VC



شکل ۸- مقایسه الگوریتم ارائه شده با الگوریتم تیم Persia از نظر امتیاز کسب شده

با ملاحظه جداول و نمودارها می توان دریافت فضایی که الگوریتم ارائه شده در بازه های زمانی اول شبیه سازی کاوش می کند تقریباً یک و نیم برابر فضایی است که الگوریتم تیم Persia (الگوریتمهای کوتاهترین مسیر) در همان زمان کاوش می کنند. نتیجه دیگر این است که نقطه اوج کاوش برای الگوریتم ارائه شده در شروع شبیه سازی است که برای یک الگوریتم کاوش و جستجو امتیاز مهمی محسوب می شود در حالیکه الگوریتم های کوتاهترین مسیر به این صورت عمل نمی کنند و اگر هم عمل می کنند به سرعت و قدرت الگوریتم ارائه شده نیستند.

## مراجع

- [۱] A. Colorni; M. Dorigo and V. Maniezzo; “Distributed optimization by ant colonies”, Proceedings of ECAL’۹۱, European Conference on Artificial Life, Elsevier Publishing, Amsterdam, ۱۹۹۱.
- [۲] Bowling M.; “Robocup Rescue : Agent Development Kit”, Version ۰.۴.
- [۳] Dorigo M.; Di Caro G.; and Gambardella M. L.; “Ant algorithms for discrete optimization”, Artif. Life, vol. ۵, no. ۲, pp. ۱۳۷-۱۷۲, ۱۹۹۹.
- [۴] Dorigo M.; and Stützle T.; “Ant Colony Optimization”, Cambridge, MA: The MIT Press. ۲۰۰۲.

- [Δ] Khojasteh M. R.; Kazimi A.; and Ghaseminik Z.; “*Persia ۲۰۰۶, Towards a Full Learning Automata-Based Cooperative Team*”, Team Description Paper, ۲۰۰۶.
- [ϫ] Khojasteh M. R. and Heidari H.; “*Persia ۲۰۰Δ Team Description*”, Team Description Paper, ۲۰۰Δ.
- [Υ] Kleiner A. ; Brenner M. ; Bräuer T. ; Dornhege C. ; Göbelbecker M. ; Luber M. Prediger j.; and Stuckler J. Nebel B.; “*Successful Search and Rescue in Simulated Disaster Areas*”, Institut für Informatik, University at Freiburg, ۲۰۱۰ Freiburg, Germany.
- [Λ] Morimoto T.; “*How to Develop a RoboCupRescue Agent for RoboCup Rescue Simulation System*”, version ۰, ۱st edition.
- [ϣ] RoboCup Rescue Official Site, <http://robomec.cs.kobe-u.ac.jp/robocup-rescue/>.
- [۱۰]The Robocup Rescue Technical Committee, “*Robocup-Rescue Simulator Manual*”, Web page : <http://robomec.cs.kobe-u.ac.jp/robocup-rescue>.