

# الگوریتم کلونی مورچه سلولی

سحر کیان‌فر<sup>۱</sup>، محمدمرضا میبدی<sup>۲</sup>

<sup>۱</sup> دانشجوی کارشناسی ارشد، آزمایشگاه محاسبات نرم، دانشگاه صنعتی امیرکبیر، تهران  
kianfar@aut.ac.ir

<sup>۲</sup> گروه هوش مصنوعی، دانشکده مهندسی کامپیوتر، دانشگاه صنعتی امیرکبیر، تهران  
mmeybodi@aut.ac.ir

## چکیده

بسیاری از مسائل دنیای واقعی پویا نیاز به الگوریتمی داریم که علی رغم پیدا کردن بهینه در محیط، بتواند بهینه‌های در حال تغییر را دنبال کند. تاکنون الگوریتم‌های تکاملی مختلفی برای بهینه‌سازی در محیط‌های پویا پیشنهاد شده است. در یک محیط پویا پس از روی دادن تغییر در محیط، الگوریتم نیاز به تنوع کافی جهت جستجوی دوباره محیط دارد، در عین حال استفاده از اطلاعات جستجویی پیشین روند جستجو را سریع تر می‌کند. مشکل اصلی الگوریتم‌های تکاملی معمول در حل مسائل بهینه‌سازی پویا، همگرایی زودرس و کاهش تنوع جمعیتی در طول زمان است. بنابراین، در مواجه با مسائل بهینه‌سازی پویا نیاز به رویکردهایی است که تنوع را در طول زمان حفظ کنند.

در این مقاله یک الگوریتم کلونی مورچه سلولی برای حل مسائل بهینه سازی پیوسته و پویا ارائه می‌شود. ایده‌ی اصلی الگوریتم برای حفظ تنوع، تقسیم‌بندی فضای جستجو با استفاده از اتماتاتی سلولی است. جهت ارزیابی الگوریتم پیشنهادی، از تابع محک قله‌های متحرک استفاده می‌شود.

## کلمات کلیدی

الگوریتم کلونی مورچه، اتماتاتی سلولی، بهینه‌سازی، محیط پویا، تابع قله‌های متحرک

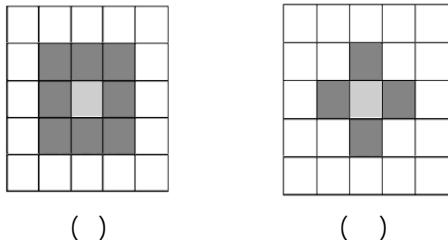
پویا است. با این وجود کارهای کمی برای الگوریتم کلونی مورچه در محیط‌های پویا انجام شده است [۴-۲].

## ۱- مقدمه

الگوریتم DHCIAAC اولین الگوریتم کلونی مورچه‌ی به کار رفته در بهینه سازی مسائل پیوسته و پویاست [۲]. این الگوریتم از دو کانال ارتباطی بین مورچه‌ها بهره می‌گیرد. الگوریتم دیگر CANDO [۳] و برای حفظ تنوع جمعیت یک شارژ الکتریکی دافع به هر مورچه نسبت می‌دهد. این شارژ از نزدیک شدن مورچه‌ها به یکدیگر تا حدفاصل معینی جلوگیری می‌کند بنابراین از همگرایی همه‌ی مورچه‌ها به یک نقطه ممانتع می‌شود و تنوع در جمعیت حفظ می‌شود.

در این مقاله یک الگوریتم کلونی مورچه مبتنی بر اتماتاتی سلولی به نام الگوریتم کلونی مورچه سلولی (CellularACO)<sup>۱</sup> ارائه می‌شود. در این الگوریتم، یک اتماتاتی سلولی فضای جستجو را به تعدادی پاریشن که هر کدام یک سلول از اتماتاتی سلولی است تقسیم‌بندی

در اواخر ۱۹۸۰ زمینه‌ی تحقیقاتی جدیدی در هوش مصنوعی توزع شده پدیدار شد که هوش جمعی نامیده می‌شود. در این زمینه از حرکت اجتماعی حشرات برای ایجاد الگوریتم‌های جدید الهام گرفته شد. در حقیقت توانایی تولید ساختارهای پیچیده و پیدا کردن راه حل برای مسائل غیربدهی (مرتب سازی، جستجوی بهینه، ...) با استفاده از عامل‌هایی که دید کلی از محیط‌شان ندارند، بدون هیچ کنترل مرکزی یا استراتژی کلی، محققان را جذب کرد. مفاهیم زیادی مثل خودسازماندهی و غیرمنتظره بودن بعدها مطرح شد. علم کامپیوتر از دو مفهوم خودسازماندهی و غیرمنتظره بودن برای تعریف هوش جمعی استفاده کرد. یکی از مزایای اصلی الگوریتم‌های الهام گرفته از طبیعت، مانند الگوریتم کلونی مورچه (ACO)<sup>۱</sup> اعطاف‌پذیری در محیط‌های



شکل ۱) همسایگی در اتوماتای سلوی (الف) و نیومن (ب) مور

### ۳- الگوریتم کلونی مورچه پیوسته

برای تطبیق الگوریتم کلونی مورچه برای محیط پیوسته، توزیع احتمال گسسته اختصاص یافته به مولفه‌ها را که بیانگر مقدار فرومون است، با یک توزیع احتمال پیوسته مدل می‌کنیم. بدین منظور ما از توزیع گاوسی نرمال استفاده می‌کنیم. و با نمونه برداری از این توزیع راه حل جدید را می‌سازیم. توزیع دارای دو پارامتر میانگین و واریانس است. دو نوع فرومون یکی برای مقادیر میانگین و دیگری برای واریانس تعریف می‌شود. هر سلول  $c$  یک بردار میانگین<sup>۱۱</sup> و یک بردار واریانس<sup>۱۲</sup> دارد. مقادیر این بردارها مستقیماً مقدار فرومون را نشان می‌دهد. برای یک محیط  $d$  بعدی داریم:

$$\mu^i = \{\mu_1^i, \mu_2^i, \dots, \mu_d^i\}, \quad (1)$$

$$\sigma^i = \{\sigma_1^i, \sigma_2^i, \dots, \sigma_d^i\}. \quad (2)$$

در شروع الگوریتم مقدار میانگین هر سلول برابر با یک نقطه تصادفی درون سلول و مقدار واریانس با توجه به مشخصات فضای جستجو با استفاده محاسبه می‌شود.

برای هر مورچه، با نمونه بردار از توزیع نرمال به دست آمده، با استفاده از یک تولید کننده تصادفی نرمال مانند روش جعبه مولر<sup>۷</sup> با میانگین<sup>۱۳</sup>  $\mu$  و واریانس<sup>۱۴</sup>  $\sigma$  مقدار زامین متغیر مورچه  $i$  به دست می‌آید، فرآیند ساخت تا انتخاب  $d$  امین متغیر تکرار می‌شود. در انتهای مقدار تابع هدف محاسبه می‌شود. در انتهای فرآیند ساخت راه حل به تعداد مورچه‌ها ( $M$ ) راه حل جدید ساخته می‌شود. مرحله به روز رسانی فرومون شامل تبخیر و تقویت فرومون است. یعنی برای هر سلول در تکرار  $t$  مقداری از فرومون روی سلول با استفاده از یک ضریب ثابت فراموش می‌شود (روابط (۳) و (۴)). در این روابط  $\rho$  نرخ تبخیر فرومون است.

$$\vec{\mu}^i(t) = (1 - \rho) \vec{\mu}^i(t-1), \quad (3)$$

$$\vec{\sigma}^i(t) = (1 - \rho) \vec{\sigma}^i(t-1). \quad (4)$$

بهترین مورچه  $Ant_{best}$  برای تقویت مقادیر فرومون انتخاب می‌شود. مقادیر فرومون با استفاده از مورچه‌ی انتخابی مطابق روابط (۵) و (۶) تقویت می‌شود. میانگین و واریانس سلول به سمت بهترین مکان سلول حرکت می‌کنند.

می‌کند. این ایده برای اولین بار توسط هاشمی و میبدی<sup>[۴، ۵]</sup> ارائه شد. اتوماتای سلوی بطور ضمنی کلونی مورچه‌ها را به کلونی‌های کوچکتری تقسیم می‌کند. هریک از این کلونی‌ها وظیفه جستجو در سلول مقیم و سلول‌های همسایه را دارد. در ابتدای الگوریتم مورچه‌ها بطور تصادفی در فضای جستجو توزیع می‌شوند. مورچه‌ها در محیط حرکت می‌کنند تا به سلوی که احتمال وجود قله در آن بیشتر است، برسند. در این مقاله یک الگوریتم کلونی مورچه برای حرکت مورچه‌ها در محیط پیوسته پیشنهاد شده است. زمانی که تعداد مورچه‌های سلوی زیاد می‌شود، برخی از مورچه‌های سلوی به تصادف انتخاب شده و به سلوی‌های دیگری در اتوماتای سلوی انتقال می‌یابند. هنگامی که تغییری در محیط رخ می‌دهد، بهترین مورچه سلوی یک جستجوی محلی انجام می‌دهد.

ادامه مقاله بصورت ذیل سازماندهی شده است. در بخش دوم اتوماتای سلوی معرفی می‌گردد. در بخش سوم یک الگوریتم کلونی مورچه برای حرکت مورچه‌ها در محیط پیوسته ارائه می‌شود. در بخش چهارم الگوریتم جستجوی محلی بکار رفته معرفی می‌گردد. در بخش پنجم الگوریتم پیشنهادی یعنی الگوریتم کلونی مورچه سلوی ارائه می‌شود. در بخش ششم طی انجام آزمایشاتی کارآبی الگوریتم مورد بررسی قرار می‌گیرد. در پایان نتیجه گیری ارائه می‌شود.

## ۲- اتوماتای سلوی

اتوماتای سلوی (CA)<sup>۳</sup> در اوخر دهه ۱۹۴۰ توسط ون نیومن<sup>۴</sup> مطرح و پس از او توسط ریاضیدانی بنام اولام<sup>۵</sup> به عنوان مدلی برای بررسی رفتار سیستم‌های پیچیده پیشنهاد شد. اتوماتای سلوی در حقیقت سیستم‌های دینامیکی گسته‌ای هستند که رفتارشان کاملاً بر ارتباط محلی استوار است<sup>[۶]</sup>.

در اتوماتای سلوی، فضا به صورت یک شبکه تعریف می‌گردد که به هر خانه آن یک سلول گفته می‌شود. زمان به صورت گسسته پیش می‌رود و قوانین آن به صورت سراسری است که از طریق آن در هر مرحله هر سلول، وضعیت جدید خود را با در نظر گرفتن همسایه‌های مجاور خود بدست می‌آورد. اتوماتای سلوی را می‌توان به عنوان سیستم‌های محاسباتی نیز در نظر گرفت که اطلاعات کد شده در خودشان را پردازش می‌کنند. قوانین اتوماتای سلوی، نحوه تأثیر پذیرفتن سلول از سلوی‌های همسایه خود را مشخص می‌کند. یک سلول را همسایه سلول دیگر گوئیم هرگاه بتواند آن را در یک مرحله و براساس قانون حاکم تحت تأثیر قرار دهد. همسایگی ون نیومن و مور<sup>۶</sup> با شعاع همسایگی ۱ در یک فضای دو بعدی در شکل ۱ نشان داده شده است.

اگر تعداد ابعاد محیط  $d$  باشد و هر بعد به  $N_p$  بخش مساوی تقسیم شود، نیاز به یک اتوماتای سلولی  $d$  بعدی  $CA = (Z^d, \varphi, N, F)$  با مجموعه سلول‌های  $\{cell_i | 1 \leq i \leq (N_p)^d\}$  داریم. برای هر سلول یک همسایگی تعریف می‌شود که دربرگیرنده سلول‌هایی است که با آن سلول همسایه هستند و تمامی اعمال تکاملی بر روی آن‌ها به صورت محلی انجام می‌پذیرد.

هر سلول در اتوماتای سلولی شامل اطلاعات حافظه‌ی سلول، مورچه‌های فعال در سلول و اطلاعات فرومون است که توسط مورچه‌ها در هر سلول برجای گذاشته می‌شود. جهت استفاده از نتایج جستجوهای پیشین برای هر سلول حافظه در نظر گرفته می‌شود. در این حافظه مکان و شایستگی بهترین موجود در آن سلول از زمان آخرین تغییر تاکنون حفظ می‌شود. این حافظه برای سلول  $i$  نامیده  $C_i^{Best}$  می‌شود.

مورچه‌ها بر حسب مکانشان به حالت سلول‌های متناظرشان اضافه می‌گردند و وظیفه جستجو در آن سلول و همسایگی آن را بر عهده دارند. هر مورچه می‌تواند فعال یا غیرفعال باشد. مورچه‌های غیرفعال در فرآیند جستجو شرکت نمی‌کنند و فقط در حالت سلول موجودند. در ابتدا تمام مورچه‌ها فعال هستند.

برای هر سلول پارامتر فرومون تعریف می‌شود. این مقدار با حرکت مورچه‌ها در محیط بروزرسانی می‌شود و مورچه‌های سلول آن را در فرآیند ساخت راه حل به کار می‌گیرند. مقدار فرومون می‌تواند میانگین یا واریانس توزیع نرمالی که برای ساخت راه حل توسط مورچه‌ها به کار می‌رود، یا هر دو مورد باشد. در ابتدا فرومون همه‌ی سلول‌ها برابر با یک مقدار اولیه است.

## ۱-۵- روند اجرای الگوریتم

بعد از مقداردهی اولیه که شامل توزیع مورچه‌ها در اتوماتای سلولی است، الگوریتم در یک حلقه قوانین بروزرسانی اتوماتای سلولی را تکرار می‌کند. اتوماتای سلولی شامل ۶ قانون است که آن‌ها را بصورت موازی بر روی تک‌تک سلول‌ها اجرا می‌کند. این قوانین عبارتند از: ۱) نظرارت بر تغییرات محیط، ۲) جابجایی مورچه‌های غیرفعال (۳) به روزرسانی حافظه سلول‌ها، ۴) بهنگام سازی مکان مورچه‌ها (با استفاده از الگوریتم کلونی مورچه) (۵) به روزرسانی فرومون سلول‌ها، ۶) بررسی تراکم مورچه‌های درون سلول. در ادامه جزئیات قوانین آورده شده است.

**نظرارت بر تغییرات محیط:** برای شناسایی تغییر در محیط شایستگی بهترین مکان سلول  $C_i^{Best}$  محاسبه می‌شود، اگر تغییر کرده باشد، تغییر در محیط رخ داده است. بمحض شناسایی تغییر در محیط، شایستگی مورچه‌های داخل سلول مجدداً محاسبه شده و حافظه سلول فراموش می‌شود.

**جابجایی مورچه‌های غیرفعال:** همگام جابجایی مورچه با استفاده از الگوریتم کلونی مورچه ممکن است مکان جدید مورچه در سلول

$$\vec{\mu}(t) = \vec{\rho} \cdot \vec{x}_{best}, \quad (5)$$

$$\vec{\sigma}(t) = \rho * \left| \vec{x}_{best} - \vec{\mu} \right|, \quad (6)$$

→ مکان بهترین راه حل ساخته شده در سلول است.

## ۴- جستجوی محلی HJ

HJ یک روش جستجوی الگو<sup>۸</sup> است که در سال ۱۹۶۱ توسط هوک و جیوز<sup>۹</sup> معرفی شد [۸]. علی رغم گذشت زمان طولانی از ارائه الگوریتم، هنوزهم در بسیاری از مسائل جستجوی محلی قطعی به عنوان انتخاب اول بکار می‌رود. HJ راه حل به دست آمده از جستجوی سراسری را به عنوان نقطه پایه انتخاب می‌کند و در دو فاز اکتشاف و حرکت الگو آن را تقویت می‌کند.

الگوریتم با یک حرکت اکتشافی آغاز می‌شود که در آن همه‌ی متغیرهای نقطه پایه به نوبت با یک گام از پیش تعریف شده (۵) حرکت می‌کند. سپس حرکت الگو همه‌ی تغییرات موفق فاز اکتشاف را همزمان تکرار می‌کند. نتیجه حرکت ارزیابی می‌شود و در صورت موفق بودن حرکت نقطه جدید به عنوان نقطه پایه به کار می‌رود. این پروسه تا زمانی که بهبودی در هیچ یک از ابعاد حاصل نشود، تکرار می‌شود. اندازه گام‌های بکار رفته در طول زمان کاهش می‌یابد. شبه کد جستجوی محلی در شکل ۲ نشان داده شده است.

### Hooke-Jeeves Pattern Search Algorithm

1. Obtain initial base point  $x^t$ . Determine set of step lengths
2. Move the base point along every one of the  $d$  dimensional axes at a time and evaluate the result. Adopt each new point if improvement on the previous point. This takes at least  $d$ , at most  $2d$  evaluations.  
If any of the moves was successful, go to 3. If none was successful, go to 4.
3. Repeat the successful moves in a combined pattern move. If the new point has a better fitness, assume it as the new base point. Return to 2 whichever the outcome.
4. Adjust step length to next smaller step. If there is a smaller step, continue from 2. If not, terminate.

شکل ۲) شبه کد الگوریتم جستجوی محلی HJ [۸]

## ۵- الگوریتم کلونی مورچه سلولی

در الگوریتم کلونی مورچه سلولی، یک اتوماتای سلولی فضای جستجو را به تعدادی پارتبیشن افزای می‌کند، که هریک از این پارتبیشن‌ها یک سلول در اتوماتای سلولی است. اتوماتای سلولی بطور ضمنی کلونی مورچه‌ها را به تعدادی کلونی کوچکتر تقسیم می‌کند که هر کلونی ساکن یک سلول است. هر کلونی مقیم سلول مسئول پیدا کردن بهینه در آن سلول و سلول‌های همسایه‌ی آن سلول است.

رخ می‌دهد، تعداد قله‌ها، شکل قله‌ها و مکان قله‌ها تغییر می‌کند. میزان شدت تغییرات و فرکانس تغییرات قابل تنظیم است. از معیار کارآیی خطای آفلاین<sup>۱۱</sup> استفاده می‌شود. خطای آفلاین برابر با میانگین بهترین مقدار شایستگی محاسبه شده در هر گام از الگوریتم است. یعنی داریم:

$$\text{offline\_error} = \frac{1}{\text{Max\_FE}} \sum_{t=1}^{\text{Max\_FE}} (f(\text{Ant}_{\text{best}}(t))) \quad (7)$$

در صورتی که در تمام موارد ارزیابی شایستگی، مقدار کارآیی آفلاین برابر صفر باشد، مقدار خطای آفلاین صفر خواهد بود. پارامترهای تابع محک قله‌های متوجه در جدول ۱ ذکر شده است.

در الگوریتم کلونی مورچه نرخ تبخر فرومون برابر با  $0.3$  و تعداد مورچه‌ها برابر  $100$  در نظرگرفته می‌شود. برای اتوamatatی سلوی تعداد تقسیم بندی و ظرفیت سلوول برابر  $10$  و همسایگی نوع مور با شاعع  $2$  در نظر می‌گیریم. برای جستجو محلی اندازه گام و لیه  $5$ ، شتاب کاهشی برابر  $0.2$ ، تعداد دفعات کاهش هر بعد برابر  $4$  است.

نتایج بدست آمده با نتایج الگوریتم‌های Cellular Pso[۴] و mQSO[۱۱] مقایسه شده است. الگوریتم CellularPSO از جهت شباهت ساختاری به الگوریتم پیشنهادی انتخاب شده است. الگوریتم mQSO به عنوان یک الگوریتم استاندارد برای مقایسه در محیط‌های پویا به کار می‌رود. نتایج بدست آمده در مقایسه با الگوریتم‌های دیگر برای فرکانس تغییر  $2500$  و  $5000$  به ترتیب در جدول ۲ و  $3$  درج شده است. همانطور که نتایج نشان می‌دهد الگوریتم دارای خطای کمتری نسبت به الگوریتم‌های دیگر است. در محیط پویا نیاز به حفظ تنوع امری ضروری است و تقسیم‌بندی فضای جستجو این تنوع را در طول اجرای الگوریتم حفظ می‌کند. مزیت دیگر الگوریتم کلونی مورچه سلوی استفاده از یک جستجوی محلی قوی است.

جدول ۱) پارامترهای تابع محک قله‌های متوجه

مقدار	پارامتر
$10-200$	تعداد قله‌ها
$5$	تعداد ابعاد
$[30-70]$	محدوده ارتفاع قله‌ها
$50$	ارتفاع استاندارد قله‌ها
$[1-12]$	محدوده عرض قله‌ها
$[0-100]$	محدوده فضای جستجو
$1000-5000$	فرکانس تغییر
مخروطی	شکل قله‌ها
ثابت	گام‌های تغییر

دیگری واقع شود. اگر سلوول جدید یکی از سلوول‌های همسایه باشد، مورچه از حالت سلوول فعلی حذف و به حالت سلوول جدید اضافه می‌شود. اما اگر سلوول در مکان دورتری باشد، با توجه به محدودیت محلی بودن قوانین اتوماتاتی سلوولی، انتقال یک مرتبه و ناگهانی مورچه از یک سلوول به سلوول دیگر امکان‌پذیر نیست. برای رساندن مورچه به مقصد آن را بین سلوول‌های میانی جابجا می‌کنیم. مورچه هنگام جابجایی غیرفعال می‌شود و در تکامل شرکت نمی‌کند.

به روزرسانی حافظه سلوول‌ها: بهترین مکان یافته شده در سلوول از زمان آخرین تغییر محیط با استفاده از مورچه‌های موجود در سلوول بهنگام می‌شود. در صورتی که شایستگی مورچه‌ای بیشتر از بهترین مکان سلوول باشد، جایگزین آن می‌شود.

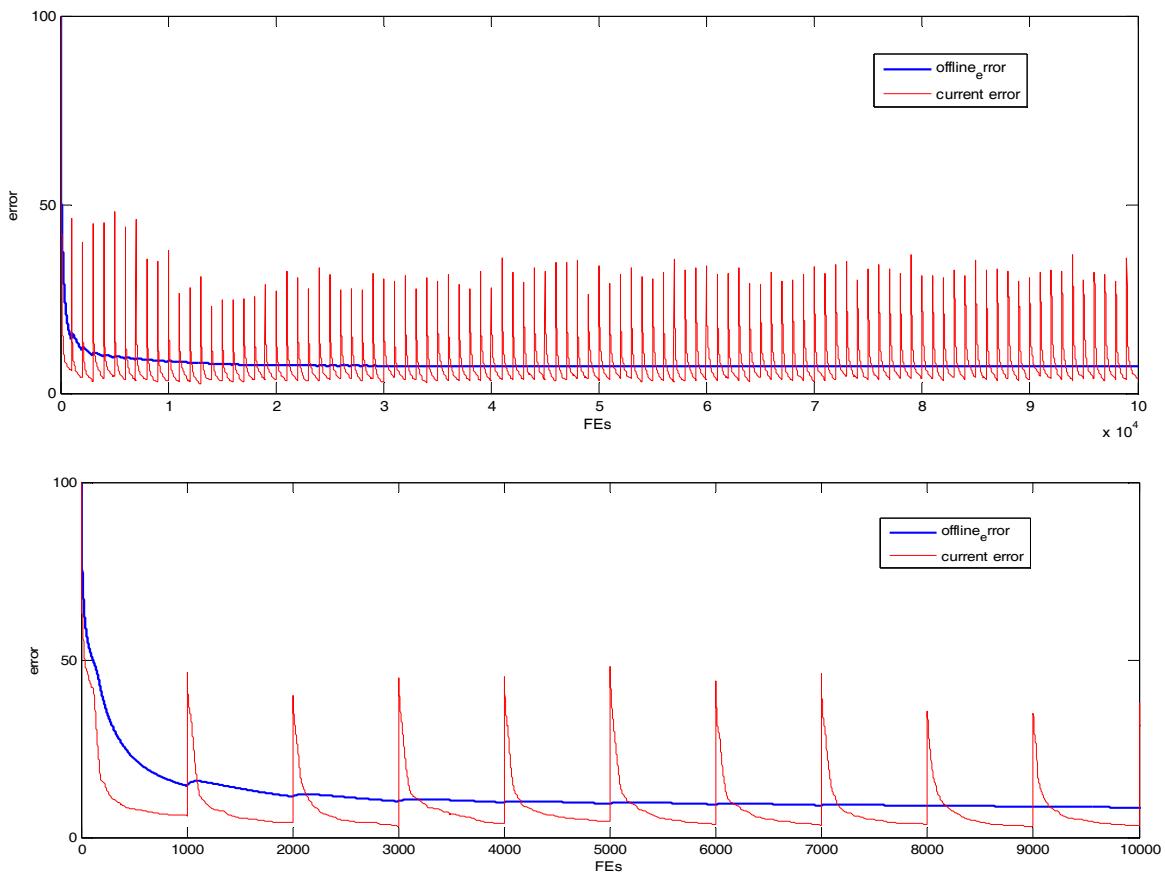
بهنگام سازی مکان مورچه‌ها: مورچه‌ها از الگوریتم کلونی مورچه پیشنهادی در بخش ۳ برای جابجایی در سلوول‌ها و پیدا کردن بهینه استفاده می‌کنند. در هر تکرار از اتوamatatی اگر تغییری در محیط رخ داده باشد، مورچه‌های سلوول در یک تکرار بعد از تغییر با استفاده از جستجوی محلی مکان خود را بهنگام می‌کند. مورچه‌ها برای جستجوی محلی الگوریتم HJ را به کار می‌گیرند. پس از هر تغییر در محیط امکان تغییر مکان بهینه هست، بنابراین برای دنبال کردن بهینه‌ها از یک جستجوی محلی پس از هر تغییر استفاده می‌شود.

بروزرسانی فرومون سلوول‌ها: بعد از ساخت را حل در هر سلوول، فرومون آن سلوول با استفاده از مکان و شایستگی بهترین مورچه سلوول بهنگام می‌شود. همانطور که در بخش ۳ اشاره شد، فرومون با استفاده از بهترین مورچه سلوول بهنگام می‌شود.

بررسی تراکم مورچه‌های سلوول: برای کنترل تراکم مورچه‌ها در الگوریتم پیشنهادی در هر بار بروزرسانی، هر سلوول تعداد مورچه‌های فعال خود را با ظرفیت سلوول مقایسه می‌کند، در صورتی که تعداد مورچه‌ها بیشتر باشد، از بین بدترین مورچه‌های فعال به تعداد ظرفیت اضافه سلوول مورچه به تصادف انتخاب شده و برای حفظ تنوع کلونی آن‌ها را به سلوول‌های که بصورت تصادفی انتخاب شده اند منتقل می‌شود. در صورتی که سلوول جدید در مکانی غیر از همسایگی سلوول فعلی باشد، مورچه تا رسیدن به سلوول مقصد غیرفعال می‌شود. با حرکت تصادفی مورچه‌ها، قدرت اکتشاف الگوریتم جستجو افزایش می‌باید.

## ۶- آزمایشات

برای ارزیابی کارایی‌در محیط‌های پویا توابع محک متفاوتی ارائه شده است. تابع محک قله‌های متوجه<sup>۱۰</sup> که رایج‌ترین و کامل‌ترین آن‌هاست مورد استقبال فراوانی قرار گرفته است<sup>۹، ۱۰</sup>. این تابع شامل یک محیط چند بعدی با چندین قله است. هنگامی که تغییری در محیط



شکل ۳) خطای جاری برای فرکانس تغییر ۱۰۰۰۰۰ ارزیابی، ب) برای ۱۰۰۰۰ ارزیابی تابع

جدول (۳) خطای آفلاین به همراه میانگین خطای استاندارد الگوریتم‌های مختلف به ازای تعداد قله‌ای مختلف برای فرکانس تغییر ۵۰۰۰

نام الگوریتم			
Cellular ACO	Cellular PSO	mQSO	تعداد قله‌ها
۰.۳۳±۰.۰۳	۳.۴۶±۰.۲۲	۰.۵۳±۰.۰۱	۱
۱.۳۶±۰.۰۲	۱.۷۹±۰.۱۲	۱.۰۵±۰.۰۶	۵
۱.۳۲±۰.۰۲	۱.۸۴±۰.۰۸	۱.۳۱±۰.۰۳	۱۰
۱.۸۷±۰.۰۱	۲.۶۳±۰.۱۱	۱.۶۹±۰.۰۵	۲۰
۲.۱۶±۰.۰۲	۲.۹۱±۰.۱۰	۱.۷۸±۰.۰۲	۳۰
۲.۲۴±۰.۰۱	۳.۱۶±۰.۱۱	۱.۸۶±۰.۰۲	۴۰
۲.۲۸±۰.۰۲	۳.۲۲±۰.۱۱	۱.۹۰±۰.۰۲	۵۰
۲.۲۴±۰.۰۱	۳.۴۳±۰.۱۰	۱.۹۰±۰.۰۱	۱۰۰
۲.۰۵±۰.۰۰	۳.۳۸±۰.۰۹	۱.۹۰±۰.۰۱	۲۰۰

جدول (۲) خطای آفلاین به همراه میانگین خطای استاندارد الگوریتم‌های مختلف به ازای تعداد قله‌ای مختلف برای فرکانس تغییر ۲۵۰۰

نام الگوریتم			
Cellular ACO	Cellular PSO	mQSO	تعداد قله‌ها
۰.۳۸±۰.۰۱	۴.۹۱±۰.۲۸	۷.۷۹±۰.۷۲	۱
۳.۵۳±۰.۳۶	۲.۹۵±۰.۲۰	۳.۵۳±۰.۱۸	۵
۴.۵۱±۰.۳۷	۲.۹۷±۰.۱۵	۳.۲۰±۰.۱۴	۱۰
۴.۱۴±۰.۱۲	۳.۵۱±۰.۱۴	۳.۸۳±۰.۱۲	۲۰
۴.۰۵±۰.۱۲	۳.۸۷±۰.۱۲	۴.۰۲±۰.۱۲	۳۰
۳.۶۷±۰.۰۶	۳.۸۹±۰.۱۰	۳.۹۰±۰.۱۱	۴۰
۳.۴۹±۰.۰۵	۴.۱۶±۰.۱۵	۳.۹۰±۰.۱۰	۵۰
۳.۱۰±۰.۰۲	۴.۱۸±۰.۱۱	۳.۸۱±۰.۱۰	۱۰۰
۲.۵۸±۰.۰۱	۴.۰۴±۰.۰۹	۳.۶۶±۰.۰۷	۲۰۰

- salesman problem,"* IEEE transactions on evolutionary computation, vol. 1, pp. 53-66, 1997.
- [<sup>8</sup>] R. Hooke and T. A. Jeeves, "Direct Search Solution of Numerical and Statistical Problems" J. ACM, vol. 8, pp. 212--229, 1961.
- [<sup>9</sup>] J. Branke, "Memory enhanced evolutionary algorithms for changing optimization problems," in Evolutionary Computation, 1999. CEC 99. Proceedings of the 1999 Congress on, p. 1882 Vol. 3 , 1999.
- [<sup>10</sup>] J. Branke, "evolutionary algorithms for dynamic optimization problems-a survey," Insitute AIFB 1999.
- [<sup>11</sup>] T. Blackwell and J. Branke, "Multiswarms, exclusion, and anti-convergence in dynamic environments," Evolutionary Computation, IEEE Transactions on, vol. 10, pp. 459-472, 2006.

شکل ۳ خطای جاری را برای فرکانس تغییر ۱۰۰۰، در محیط قله‌ای نشان می‌دهد. محور افقی تعداد ارزیابی‌هایتابع شایستگی و محور عمودی بیانگر خط است. خط فرمز کمترین خط در هر ارزیابی و خط آبی میانگین خط یا همان خطی آفلاین است. نمودار الف خط را تا برقراری شرط توقف و نمودار ب ۱۰ تغییر اول را نشان می‌دهد. مشاهده می‌شود که بعد از روی دادن تغییر در محیط الگوریتم کلونی مورچه سلوی به خوبی آن را دنبال می‌کند. علت این امر اعمال جستجوی محلی بعد از شناسایی تغییر در محیط است.

## - نتیجه

### تقدیر و تشکر

قسمتی از این کار تحقیقاتی با حمایت مالی مرکز تحقیقات مخابرات ایران انجام گرفته است.

### زیرنویس‌ها

- 
- <sup>1</sup> Ant Colony Optimization  
<sup>2</sup> Cellular Ant Colony Optimization  
<sup>3</sup> Cellular Automata  
<sup>4</sup> Von Neumann  
<sup>5</sup> Ulam  
<sup>6</sup> moor  
<sup>7</sup> Box-Muller  
<sup>8</sup> Pattern Search  
<sup>9</sup> Hook-Jeevs  
<sup>10</sup> Moving peak benchmark  
<sup>11</sup> Offline error

در این مقاله یک الگوریتم کلونی مورچه مبتنی بر اتوماتای سلوی به نام الگوریتم کلونی مورچه سلوی برای حل مسائل بهینه‌سازی پویا با متغیرهای پیوسته ارائه شد. الگوریتم پیشنهادی برای حفظ تنوع و استفاده از نتایج پیشین جستجو، تقسیم‌بندی فضای جستجو با استفاده از اتوماتای سلوی، را به کار می‌گیرد. جستجو در هر سلوی و همسایگانش به صورت موازی و جداگانه، توسط الگوریتم کلونی مورچه‌ی پیوسته انجام می‌شود. استفاده از اتوماتای سلوی موجب حفظ تنوع جمعیت مورچه‌ها می‌شود.

پس از شرح کامل روند پیشنهادی، آزمایشاتی در محیط پویای قله‌های متحرک جهت ارزیابی و مقایسه کارایی مدل پیشنهادی با سایر روش‌های تکاملی انجام شد. در این آزمایشات نشان داده شد که روش پیشنهادی از کارایی نسبتاً خوبی نسبت به سایر روش‌ها برخوردار است.

## مراجع

- [۱] M. Dorigo, V. Maniezzo, and A. Colorni, "The Ant System: Optimization by a colony of cooperating agents" IEEE Transaction on Systems, man, and cybernetics-part b, vol. 26, pp. 29-41, 1996.
- [۲] J. Dreо and P. Siarry, "An ant colony algorithm aimed at dynamic continuous optimization," Applied Mathematics and Computation, vol. 181, pp. 457-467, 2006.
- [۳] W. Tfaili and P. Siarry, "A new charged ant colony algorithm for continuous dynamic optimization," Applied Mathematics and Computation, vol. 197, pp. 604-613, 2008.
- [۴] A. Hashemi, M. Meybodi, Z. Cai, Z. Li, Z. Kang, and Y. Liu, "Cellular PSO: A PSO for Dynamic Environments Advances in Computation and Intelligence." vol. 5821: Springer Berlin / Heidelberg, pp. 422-433, 2009.
- [۵] A. B. Hashemi and M. R .Meybodi, "A multi-role cellular PSO for dynamic environments," in Computer Conference, 2009. CSICC 2009. 14th International CSI, pp. 412-417, 2009.
- [۶] A. W. B. John Von Neumann, Theory of Self-Reproducing Automata: University of Illinois Press 1966.
- [۷] M. Dorigo and L. M. Gambardella, "Ant Colony System: A cooperative learning approach to the traveling