

# الگوریتم بهینه‌یابی کلونی زنبورهای مصنوعی مبتنی بر مدل تابع تسهیم برای محیط‌های پویا

ناهید شایگان پور<sup>۱</sup> ، محمد رضا میبدی<sup>۲</sup>

<sup>۱</sup> دانشجوی کارشناسی ارشد کامپیوتر گرایش نرم افزار، دانشگاه آزاد اسلامی واحد قزوین، دانشکده برق و رایانه فناوری اطلاعات  
nahid\_shayeganpour@yahoo.com

<sup>۲</sup> عضو هیئت علمی دانشگاه امیر کبیر تهران، دانشکده کامپیوتر  
mmeybodi@aut.ac.ir

## چکیده

اخيراً بکارگيري هوش دسته جمعی در حل مسائل بهینه‌یابی پویا افزایش يافته است. هوش دسته جمعی، عامل‌ها يا دسته‌های برهم کنش را که قادر به خودسازماندهی هستند مدل‌سازی می‌کند. کلونی مورچه‌ها، دسته پرنده‌گان، گله حیوانات، قالب‌گیری باکتریها و گروه زنبورها مثال‌هایی از یک سیستم جمعی می‌باشند. عمکرد خوب بهینه‌یابی گروه زنبورها بروی یک سری از مسائل ایستا ثابت شده است ولیکن بیشتر مسائل دنیای واقعی پویا می‌باشند به این مفهوم که موقعیت و مقدار بهینه سراسری آنها در طول زمان تغییر می‌کنند. در این مقاله الگوریتم بهینه‌یابی کلونی زنبورهای مصنوعی مبتنی بر مدل تابع تسهیم برای حل مسائل پویا ارائه گردیده است. الگوریتم پیشنهاد شده برای بهینه‌یابی تابع محک قله‌های متحرک که رفتاری شبیه به مسائل پویا را در دنیای واقعی دارد، در فرکانس‌های تغییر و با تعداد قله‌های متفاوت آزمایش شده است. نتیجه‌های آزمایش‌های انجام شده با RPSO، mQSO، Adaptive mQSO، Cellular PSO Adaptive Cellular PSO تسهیم قابلیت اکتشاف را در کلونی زنبورهای مصنوعی با افزایش تنوع و تضمین بقای اهداف در محیط‌های پویا بهبود می‌بخشد.

## کلمات کلیدی

هوش دسته جمعی، کلونی زنبورهای مصنوعی، محیط پویا، مدل تابع تسهیم

الگوریتم‌های زنبور عسل در میان الگوریتم‌های هوش دسته جمعی،  
بخصوص الگوریتم خوارجوبی کلونی زنبورهای مصنوعی به عنوان دارا  
بودن ویژگی‌هایی همچون تقسیم کار مناسب، تعامل‌های گوناگون با  
یکدیگر و خودسازماندهی در صورتیکه قادر به حفظ تنوع در طول اجرا  
باشد می‌تواند کارایی بالایی داشته باشد.

در الگوریتم‌های ارائه شده قبلی مثل الگوریتم‌های تکاملی،  
الگوریتم‌های تکاملی مبتنی بر حافظه، چند جمعیتی [۱،۲]، الگوریتم‌  
های مختلف حرکت دسته جمعی ذرات [۳-۶] سعی شده با بهبود  
برخی از ویژگی‌های فوق نتایج بهتری حاصل شود. در این مقاله یک  
الگوریتم جدید پیشنهاد شده است که از مدل تابع تسهیم در الگوریتم  
خوارجوبی کلونی زنبورهای مصنوعی استفاده می‌کند. در مدل تابع  
تسهیم ایجاد نوع با کاهش برازش جواب‌های مشابه صورت می‌گیرد  
بدین ترتیب نوعی حفظ راه حل‌های قبلی و قابلیت پیش‌بینی راه

## ۱- مقدمه

مسائل دنیای واقعی ماهیت پویا دارند یعنی در هدف بهینه‌یابی، نمونه مسئله یا در محدودیت یک مسئله بهینه‌یابی تغییراتی رخ می‌دهد، ممکن است بهینه آن مسئله تغییر کند به عبارت دیگر محیط تغییر کرده است. اگر این حالت رخ دهد و فق دادن راه حل با راه حل قدیمی ضروری است. بنابراین، بهتر است که الگوریتم بهینه‌یابی داشته باشیم که قادر باشد به طور پیوسته راه حل را با محیط در حال تغییر وفق دهد و از نتایج بدست آمده گذشته مجدد استفاده کند. مشکل اصلی الگوریتم‌ها، این است که سرانجام به یک بهینه همگرا می‌شوند و بنابراین تنوع لازم برای اکتشاف به صورت کارا و توانایی لازم را برای وفق‌پذیری با تغییرات محیط از دست می‌دهند. لذا باید الگوریتم‌هایی برای محیط‌های در حال تغییر بکار گرفته شود که قادر به تولید تنوع بعد از یک تغییر، نگهداری تنوع در طول اجرا، حفظ راه حل‌های قبلی و استفاده از شیوه‌های چند جمعیتی باشند.

مصنوعی کارگر و نیمه دوم شامل زنبورهای تماشگر می‌باشد و برای هر منبع غذایی، فقط یک زنبور کارگر وجود دارد. به عبارت دیگر، تعداد زنبورهای کارگر برابر با تعداد منابع غذایی می‌باشد. زنبورهای کارگر زمانیکه با یک منبع غذایی متروک مواجه می‌شوند به یک زنبور پیشاپنگ تبدیل می‌شوند. جستجو در الگوریتم کلونی زنبورهای مصنوعی، به صورت زیر انجام می‌شود [۸،۹]:

- زنبورهای کارگر یک منبع غذایی در مجاورت منبع غذایی حافظه خود (راه حل قبلی) جستجو می‌کنند به عبارت دیگر در پی یافتن راه حل بهتر، خود را طبق فرمول (۱) اصلاح می‌کنند.

$$v_{ij} = x_{ij} + \phi_{ij}(x_{ij} - x_{kj}) \quad (1)$$

$$i \neq k, k \in \{1, 2, \dots, SN\} \quad , j \in \{1, 2, \dots, D\}, \phi_{ij} \in [-1, 1]$$

در فرمول (۱)،  $\phi_{ij}$  یک عدد تصادفی در بازه‌ی  $[1, -1]$  است که تولید موقعیت منابع غذایی همسایه را در اطراف  $x_{ij}$  کنترل می‌کند. در این رابطه،  $SN$  تعداد زیورها می‌باشد. متغیر  $K$  نیز به صورت تصادفی تولید می‌شود و می‌باشد با  $1$  متفاوت باشد و هر چه تفاوت بین  $x_{i,j}$  و  $x_{i,k}$  کاهش یابد، انحراف از موقعیت  $x_{i,j}$  نیز کاهش خواهد یافت در این رابطه سعی می‌شود یک بعد از ابعاد یکی از موقعیت‌ها را انتخاب کرده و با توجه به میزان  $\phi$  به سمت آن و یا در خلاف جهت آن حرکت انجام شود.

- زنیورهای کارگر اطلاعات خود را در مورد موقعیت‌های منابع غذایی در اختیار تماشاگرها در سالن رقص قرار می‌دهند سپس تماشاگران یکی از آنها را بر طبق فرمول (۲) انتخاب می‌کنند.

$$P_i = \frac{\sum_{n=1}^N f_i^n}{\sum_{n=1}^N f_n^n} \quad (\text{¶})$$

در فرمول (۲)،  $fit$  میزان شایستگی منبع غذایی متناظر با زنبور آم می‌باشد که برای حفظ تنوع در این الگوریتم بجای استفاده از فرمول (۲) از مدل تابع تسهیم که در بخش بعد بیان شده استفاده می‌شود.

- زنبورهای تماساگر یک منبع غذایی در مجاورت منبع غذایی انتخاب شده مرحله قبلی را طبق فرمول (۱) جستجو می‌کنند.

در صورتیکه یک منبع پایان پذیرد زبورو کارگری منبع را ترک کرده به یک پیشاہنگ تبدیل می‌شود و شروع به جستجوی یک منبع غذایی به صورت تصادفی بر طبق فرمول (۳) می‌کند. بدین معنی که در یک بهینه محلی قرار داریم بنابراین آن نقطه حذف می‌شود و یک نقطه جدید به صورت تصادفی تولید می‌گردد.

$$x_i^j = x_{\min}^j + \text{rand}(0,1)(x_{\max}^j - x_{\min}^j) \quad (4)$$

هر چرخه از جستجو شامل ۳ گام می‌باشد: حرکت زبورهای کارگر و تماشگر به سوی منابع غذایی و محاسبه مقدار برآذش آنها (شهد آنها) و تعیین زبورهای پیشاهمگ و حرکت آنها به طور تصادفی

حل‌های احتمالی برای استفاده در شرایط جدید در الگوریتم ایجاد شده و باعث عملکرد بهتر الگوریتم می‌گردد. ادامه این مقاله بدین صورت سازماندهی شده است. بخش ۲ به معرفی شرح رفتار زنورها در طبیعت و معرفی الگوریتم پیشنهادی پرداخته شده است. در بخش ۳ معرفی تابع پویا و آزمایش‌های انجام شده بروی آن و مقایسه با روش‌های دیگر [۴-۶] انجام می‌شود و بخش پایانی نتیجه گیری است.

۲-کلونی زنبور در طبیعت

کلونی زنبورها در طبیعت [۷] شامل دو مولفه منابع غذایی و زنبورهای خوارک جو می‌باشد. زنبور خوارک جو (جمعيت) به دو دسته زنبور تقسيم می‌شود که عبارتند از: زنبور بیکار و زنبور کارگر. زنبور بیکار به دو نوع زنبور دیده و زنبور تازه نفس طبقه‌بندی می‌شود. زنبور دیده و عمل جستجو را بدون هیچ دانشی و به صورت تصادفی شروع می-کند و زنبور تازه نفس به سالن رقص رفته و جستجو خود را بواسیله استفاده از دانش رقص دم زنبورهای موجود در سالن شروع می‌کند.

گروه دوم زنبور کارگر می‌باشد، هرگاه زنبور تازه نفس منبعی غذایی را پیدا کند یک درجه ارتقا پیدا کرده و به زنبور کارگر تبدیل شده و جایگاه منبع غذایی را حفظ می‌کند و به سمت منبع غذایی پرواز کرده و یک بخش از شهد را برداشته و به کندو برمی‌گردد. در این صورت:

- اگر مقدار شهد به یک سطح کم کاهش پیدا کند زنبور کارگر منبع غذایی را ترک کرده و به یک زنبور بیکار تبدیل می‌شود.
  - اگر به مقدار کافی شهد در منبع غذایی وجود داشته باشد زنبور کارگر می‌تواند بدون تبادل اطلاعات به خوراک‌جویی خود ادامه دهد و به منبع غذایی برگردد.
  - اگر به مقدار کافی شهد در سالن رقص به تبادل اطلاعات بپردازد. کارگر می‌تواند در سالن رقص به کندو محل تبادل اطلاعات است که سالن مهمترین بخش یک کندو محل تبادل اطلاعات است که منابع رقص نام دارد، ارتباطات در بین زنبورها در رابطه با کیفیت منابع غذایی با رقص دم انجام می‌گیرد. انتخاب منبع توسط زنبورهای تازه نفس، به طور کامل مشخص نیست اما فرض بر این است که این انتخاب به صورت تابعی از کیفیت منبع غذایی با توجه به رقص دم می‌باشد. زنبورهای دیدهور نیز با احتمالی متناسب با کیفیت منبع غذایی اطلاعات خود را به اشتراک می‌گذارند.

در حقیقت به اشتراک گذاشتن این اطلاعات در هنگام رقص دم در طول زمان بیشتر خواهد شد، در نتیجه تعداد زبیرهایی که به هر منبع تخصیص داده می‌شود متناسب با کیفیت منابع غذایی خواهد بود.

#### ۱-۲- الگوریتم کلونی زنبورهای مصنوعی مبتنی بر

مدل تابع تسهیم

در این الگوریتم، کلونی شامل سه گروه از زنبورها، زنبورهای کارگر، تماشاگر و مشاهنگ می‌باشد. نمی‌توان کلونی شامل زنبورهای کارگر،



این رابطه، پدیده جالبی را نشان می‌دهد. اگر تابع برازش فرضی  $f'$  (که تابع برازش تسهیم نامیده می‌شود) به صورت زیر تعریف شود:

$$f' = \frac{f_i}{m_i} \quad (6)$$

آنگاه همه جواب‌های بهینه (چه نسبی چه مطلق) یک مقدار برازش تسهیم فرضی خواهند داشت. حال اگر انتخاب مناسب با مقادیر برازش تسهیم صورت پذیرد، همه جواب‌های بهینه به میزان مورد انتظار یکسانی کپی می‌شوند و در نتیجه تاکید یکسانی بر جواب‌های بهینه صورت می‌پذیرد. از آنجا که این فرایند در هر نسل اجرا می‌شود امکان حفظ جواب‌های بهینه چندگانه مقدور می‌گردد. در این مدل جهت تشخیص تعداد جواب‌هایی که به هر جواب بهینه تعلق دارند از یک تابع تخمین طبق فرمول (7) استفاده شده است.

$$sh(d) = \begin{cases} 1 - \frac{d}{\sigma_{share}} & \text{if } d \leq \sigma_{share}; \\ 0 & \text{otherwise} \end{cases} \quad (7)$$

پارامتر  $d$ ، فاصله بین هر دو جواب در جمعیت است. تابع فوق، با

توجه به مقادیر  $d$  و  $\sigma_{share}$  مقداری در بازه  $[0, 1]$  محاسبه می‌گردد. در یک جمعیت، یک جواب ممکن است تحت تأثیر تسهیم هیچ جواب دیگری نباشد یا از تعدادی از جواب‌ها اثر تسهیم جزئی دریافت کند و یا تحت تأثیر کامل خود باشد، اگر مقادیر تابع تسهیم که با توجه به کلیه اعضای جمعیت (شامل خود آنها) محاسبه می‌شود با یکدیگر جمع شوند، شماره موقعیت  $i$  برای آمین جواب طبق فرمول (8) محاسبه می‌گردد:

$$nc_i = \sum_{j=1}^N sh(d_{ij}) \quad (8)$$

آنگاه این شماره موقعیت، تخمینی از میزان تجمع اطراف یک جواب خواهد بود. در اینجا  $d_{ij}$  فاصله بین آمین و  $j$ -امین جواب است. در نهایت باید مقدار برازش تسهیم، یعنی  $f'_i = f_i / nc_i$  محاسبه گردد. از آنجا که کلیه جواب‌های بهینه فراتر از نماینده، مقدار  $nc_i$  بزرگتری دارند، برازش کلیه جواب‌های نماینده آنها باید به میزان زیادی کاهش یابد. کلیه جواب‌های بهینه پایین تر از نماینده، مقدار  $nc_i$  کوچکتری دارند. بنابراین نباید از برازش جواب‌های نماینده آنها به میزان زیادی کاست و در نتیجه بر جواب‌های بهینه پایین تر از نماینده تاکید بیشتری صورت می‌پذیرد.

### ۳-آزمایش‌ها

جهت پدید آوردن یک محیط پویا از یک تابع محک مناسب، تابع چند قله‌ای متحرک [۱، ۲] در آزمایش‌ها استفاده شده است که شامل یک فضای ۵ بعدی، چندین قله با ارتفاع و پهنای متفاوت است، که ارتفاع، پهنا و موقعیت هر قله در طول زمان در حال تغییر است.

به سوی منابع غذایی ممکن. یک منبع غذایی یک راه حل ممکن برای مسئله بوده و مقدار شهد منبع غذایی مربوط به کیفیت راه حل ارائه شده می‌باشد. در این روش تماساگرها بوسیله یک مکانیزم انتخاب، منابع غذایی مناسب‌تر را انتخاب می‌کنند و در انتهای هر چرخه زنبورهای پیشاپنگ برای جستجوی تصادفی و بدون هیچ دانشی انتخاب می‌شود هر چند که این زنبورها دارای کیفیت بالایی نیستند و هزینه زیادی نیز صرف نمی‌کنند ولی گاهی اوقات می‌توانند موقعیت‌های غنی منابع غذایی که ناشناخته می‌باشند را پیدا کنند. البته ممکن است در هر چرخه زنبور پیشاپنگ نداشته باشیم زیرا می‌بایست منابع غذایی (کارگرها) که در چندین چرخه در الگوریتم بهبود نیافرته‌اند را تعیین کرده و بوسیله تولید منابع غذایی جدید تصادفی با توزیع یکنواخت جایگزین کنیم. تعداد چرخه‌ای که راه حل در آن بهبود نیافرته است بوسیله متغیر limit مشخص می‌شود. این مقدار بوسیله رابطه (4) محاسبه می‌شود.

$$\text{limit} = SN * D \quad (4)$$

تعداد زنبورها و  $D$  ابعاد مسئله می‌باشد. بدین ترتیب شبه کد الگوریتم پیشنهادی در شکل (1) نشان داده شده است.

```

1. Initialize population of solution  $x_{ij}$   $i=1\dots SN$ ,  $j=1\dots D$ 
2. Evaluate the population
3. cycle=1
Repeat
4. Produce new solutions  $v_{i,j}$  for employ bees from the solution  $x_{i,j}$  and evaluated them
5. Apply the greedy selection process
6. use sharing function model and proportionate selection
7. Produce new solutions  $v_{i,j}$  for the onlooker from the solution  $x_{i,j}$  selected depending on sharing function model and proportionate selection and evaluate them
8. Apply the greedy selection process
9. Determine the abandoned solution for scot, if exists replace it with a new randomly produced solution  $x_{i,j}$  by (3)
11. Memorize the best solution achieved so far
12. cycle=cycle+1;
Until cycle= maximum cycle number

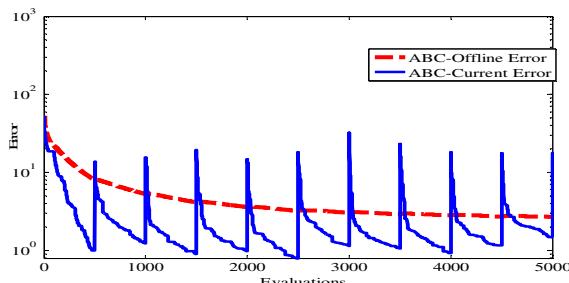
```

شکل (1): شبیه کد الگوریتم پیشنهادی

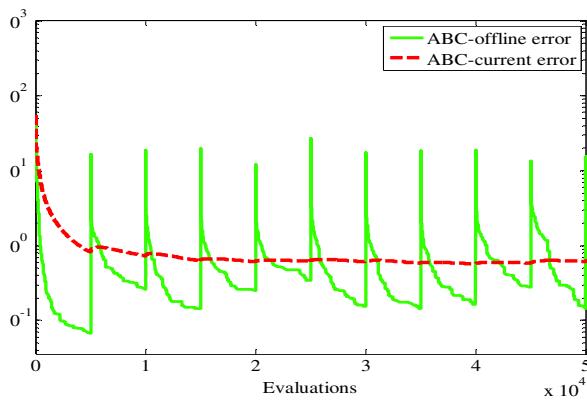
### ۲-۱-۱-۲- مدل تابع تسهیم

در این مدل [۱۰] ایجاد تنوع، مرکز بر کاهش برازش جواب‌های مشابه دارد. اگر در هر نسلی جواب بهینه بیش از حد مورد انتظار، نماینده داشته باشد به ضرر بهینه دیگری است. بدین ترتیب، برازش هر کدام از این جواب‌های نماینده باید کاهش پیدا کند، بطوریکه در یک رقابت کلی، عملگر انتخاب، همه این جواب‌های نماینده را انتخاب نکنند. از سوی دیگر جواب‌های بهینه‌ای که نماینده کمتری دارند باید بوسیله عملگر انتخاب مورد تاکید بیشتری قرار گیرند. جمعیتی با بازده  $f$  با تعداد جواب‌های ( $m$ ) مناسب است. اگر  $q$  جواب بهینه داشته باشیم این مفهوم نشان می‌دهد که نسبت بازده  $f$  (مقدار تابع هدف) به  $m$  (تعداد زیر جمعیت‌ها) به ازای هر جواب بهینه باید یکسان باشد:

$$\frac{f_1}{m_1} = \frac{f_2}{m_2} = \dots = \frac{f_q}{m_q} \quad (5)$$

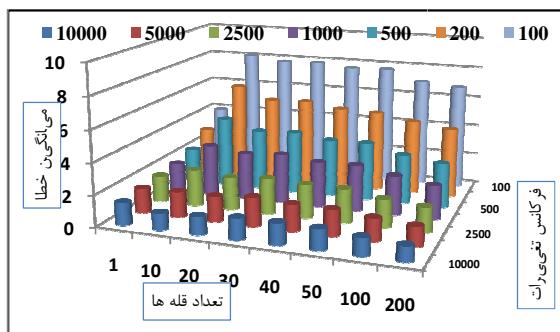


شکل (۲) : میانگین خطای و خطای لحظه‌ای در ۱۰ تغییر اولیه فرکانس sharing ABC ارزیابی با ۱۰ قله در روش پیشنهادی ۵۰۰۰



شکل (۳) : میانگین خطای و خطای لحظه‌ای در ۱۰ تغییر اولیه فرکانس sharing ABC ارزیابی با ۱۰ قله در روش پیشنهادی ۵۰۰۰

به منظور بررسی دقیقتر رفتار الگوریتم، میانگین خطای فرکانس‌های تغییر مختلف با قله‌های متفاوت محاسبه و در شکل (۴) نمایش داده شده است. با توجه به آزمایش‌ها مشخص می‌گردد در توابع چند قله‌ای با کاهش فرکانس تغییر، مقدار میانگین خطای کاهش می‌یابد به دلیل اینکه زمان کافی برای یافتن قله‌ها و همگرایی وجود دارد همچنین با افزایش تعداد قله‌ها در هر فرکانس میزان خطای افزایش می‌یابد اما این مسئله تا ۴۰ قله دیده می‌شود و با عبور از ۴۰ قله و افزایش تا ۲۰۰ کاهش در خطای افزایش می‌شود که ناشی از پوشش قله‌ها بر روی یکدیگر می‌باشد.



شکل (۴) : طیف میانگین خطای با توجه به فرکانس تغییرات و تعداد قله‌ها در الگوریتم پیشنهادی sharing ABC

تابع چند قله‌ای متحرک در نظر گرفته شده شامل  $m$  قله در  $n$  بعد با پارامترهای حقیقی است و هدف، یافتن بیشینه در هر زمان در میان

قله تا تغییر بعدی می‌باشد که به صورت فرمول (۹) تعریف می‌شود:

$$F(\vec{x}, t) = \text{Max}(B(\vec{x}), \text{Max}_{i=1..m} P(\vec{X}_i, H_i(t), W_i(t), \vec{p}(t))) \quad (9)$$

در این تابع  $B(\vec{x})$  یک شمای پایه ثابت در زمان و  $p$  تابع چند قله‌ای است که در هر  $\Delta t$  ارزیابی، ارتفاع و پهنای قله‌ها بوسیله افزودن یک عدد تصادفی گویند با میانگین صفر و واریانس  $\sigma$  و مکان هر قله بوسیله افزودن بردار  $v$  با طول ثابت  $s$  (سختی) طبق رابطه (۱۰) تغییر می‌کند. در این تابع می‌توان پیچیدگی را با افزایش و کاهش تغییرات در زمان، تغییر داد.

$$\begin{cases} \sigma \in N(0,1) \\ h_i(t) = h_i(t-1) + \text{height\_severity} \cdot \sigma \\ w_i(t) = w_i(t-1) + \text{width\_severity} \cdot \sigma \\ \vec{p}_i(t) = \vec{p}_i(t-1) + \vec{v}_i(t) \end{cases} \quad (10)$$

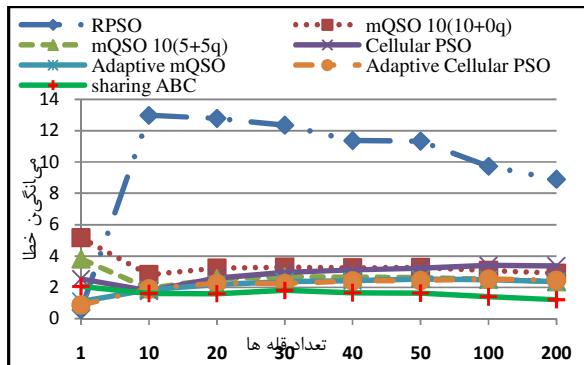
مقادیر پارامترهای برای انجام آزمایش‌ها طبق جدول (۱) در نظر گرفته شده است.

جدول (۱) : پارامترهای پیش فرض در آزمایش تابع قله‌های متحرک

پارامتر	مقدار پیش فرض
سختی ارتفاع	۷
سختی پهنا	۱
مشخوص	مشخوص
طول تغییر	۱
H مقدار مینیمم و ماکزیمم ارتفاع	[۳۰ و ۷۰]
W مقدار مینیمم و ماکزیمم پهنا	[۱ و ۱۲]
مقدار اولیه ارتفاع	۵۰
A محدوده فضای جستجو	[۰ و ۱۰۰]

در آزمایش‌های انجام شده هدف بررسی و مقایسه الگوریتم‌ها در محیط‌های پویا به کمک تابع قله‌های متحرک می‌باشد از آنجا که در محیط‌های پویا بهترین راه حل بدست آمده معیار مفیدی برای ارزیابی و گزارش نمی‌باشد، از Offline\_error و Current\_error جهت مقایسه و نتیجه‌گیری استفاده شده است.

در شکل‌های (۲) و (۳) میانگین خطای و خطای لحظه‌ای روش پیشنهادی در دو فرکانس تغییر مختلف ۱۰۰۰ و ۵۰۰۰ ارزیابی با ۱۰ قله در ۱۰ تغییر ابتدایی نمایش داده شده است. با مقایسه این دو شکل مشخص می‌گردد که الگوریتم در فرکانس تغییر پایین نسبت به فرکانس بالا بهتر عمل می‌کند به این دلیل که میانگین خطای در فرکانس تغییر پایین کمتر از فرکانس تغییر بالا می‌باشد ضمناً با توجه به نحوه کاهش خطای لحظه‌ای از هنگام ایجاد تغییر تا تغییر بعدی مشخص می‌گردد که الگوریتم پس از تغییر به خوبی وفق می‌یابد.



شکل (۸) : نمودار میانگین خطای با تعداد قله های مختلف و فرکانس تغییر

روش پیشنهادی با توجه به نتیجه های شکل (۷) و (۸) در تعداد قله های مختلف با فرکانس تغییر بالا و پایین با دیگر روش ها مقایسه شده است و خطای کمتری نسبت به دیگر روش ها بخصوص در فرکانس تغییرات بالا دارد.

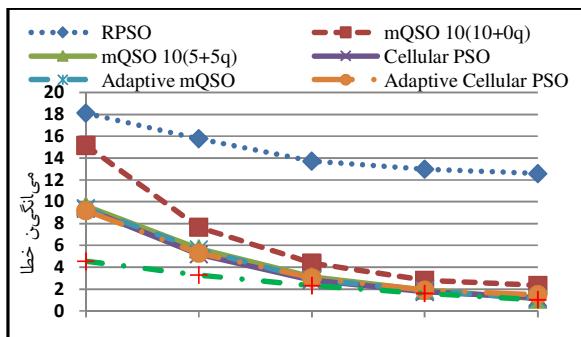
#### ۴-نتیجه گیری

الگوریتم پیشنهادی با معیار میانگین خطای با الگوریتم های موجود برای حل مسائل پویا مقایسه شده و نتایج نشان می دهد این الگوریتم نسبت به آنها میانگین خطای کمتری داشته و نتایج را بهبود می بخشد. در روش پیشنهاد شده استراتژی مدل تابع تسهیم به الگوریتم کلونی زنبورهای مصنوعی اجازه می دهد تا چند بهینه (محالی) را از تابع چند مدلی تخمین بزند. در این ایده افراد در یک ناحیه یکسان از محیط، برازش هایشان را به اشتراک می گذارند که در این صورت افراد در ناحیه کم جمعیت تر با افراد در ناحیه پر جمعیت تر به رقابت می پردازند و روی قله های مختلف پراکنده می شوند در این حالت یک ت نوع مناسبی با افرادی با شایستگی نسبتاً خوب پدید می آید که لازمه موفقیت در محیط های پویا می باشد این الگوریتم قادر به یافتن بهینه های محلی علاوه بر بهینه سراسری می باشد و می توان از آن در حل مسائل زمان بندی، مسیر یابی، کنترل ترافیک و غیره استفاده نمود و پیشنهاد می شود این الگوریتم با یک حافظه صریح در حل مسائل تکرار پذیر ترکیب و به کار گرفته شود.

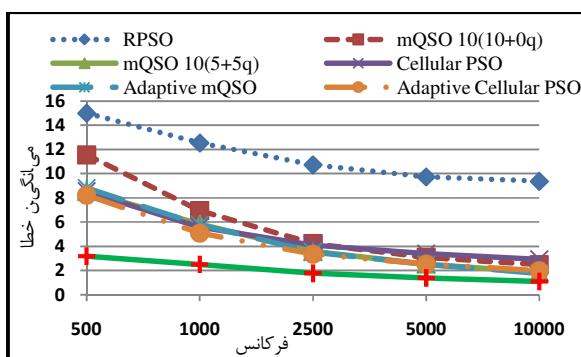
#### مراجع

- [1] Branke, J., *Evolutionary optimization in dynamic environments*, kluwer academic publishers, 2000.
- [2] Branke, J., "Memory Enhanced Evolutionary Algorithms for Changing Optimization Problems", in 1999 Congress on Evolutionary Computation, Washington D. C., USA, pp. 1875-1882, 1999.
- [3] Blackwell, T., Branke, J., "Multiswarms, Exclusion, and Anti-Convergence in Dynamic Environments", IEEE Transactions on Evolutionary Computation, vol. 10, no. 4, pp. 459-472, 2006.

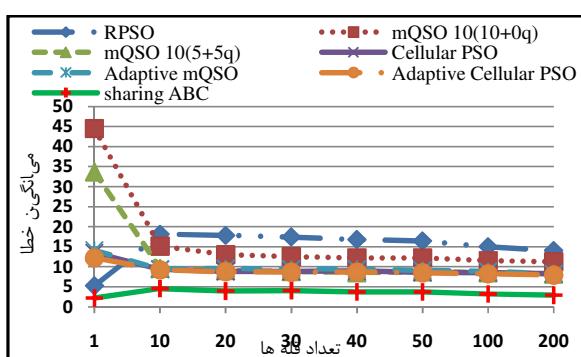
به منظور مقایسه رفتار الگوریتم ها تغییرات در فرکانس های مختلف با تعداد قله های ۱۰ و ۱۰۰ و ۵۰۰ نمایش داده شده است نمودارها مشخص می نمایند که با کاهش فرکانس میانگین خطای همه روش ها کاهش می یابد و با افزایش تعداد قله ها میانگین خطای همه روش ها کاهش می یابد و در روش پیشنهادی در اکثر فرکانس های بخصوص فرکانس های بالا مثل (۵۰۰۰) ارزیابی) میزان خطای بسیار پایین تر از روش های دیگر است.



شکل (۵) : نمودار میانگین خطای در فرکانس های تغییر مختلف با تعداد قله



شکل (۶) : نمودار میانگین خطای در فرکانس های تغییر مختلف با تعداد قله



شکل (۷) : نمودار میانگین خطای با تعداد قله های مختلف و فرکانس تغییر

- [4] Blackwell, T., Branke, J., Li, x., "Particle Swarms for Dynamic Optimization Problems", in Swarm Intelligence, Natural Computing Series, vol. Part II, pp. 193-217, 2008.
- [5] Hashemi, A. B., Meybodi, M. R., "Cellular Pso: A Pso for Dynamic Environment", in to Be Appear in the 4<sup>th</sup> International Symposium on Intelligence Computation and Applications (ISICA 2009), Huangshi, China, 2006.
- [6] Hu, X., Eberhart, R. C., "Adaptive Particle Swarm Optimization: Detection and Response to Dynamic Systems," in IEEE Congress on Evolutionary Computation, Honolulu, HI, USA, pp. 1666-1670, 2002.
- [7] Karaboga, D., "An Idea Based On Honey bee swarm numerical Optimization", Department of Computer Engineering, University of Erciyes, Tech. Rep. TR06, 2005.
- [8] Karaboga, D., Basturk, B., "A powerful and Efficient Algorithm for Numerical Function Optimization: Artificial Bee Colony (ABC) Algorithm", Journal of Global Optimization, vol. 39, pp. 459-471, 2007.
- [9] Karaboga, D., Basturk, B., "Artificial Bee Colony (ABC) Optimization Algorithm for Solving constrained Optimization Problems", Department of Computer Engineering, University of Erciyes, Tech. Rep. No. 2007-02, 2007.
- [10] Goldberg, D. E., Richardson, J., "Genetic Algorithm with Sharing for Multimodal Function Optimization", in Proceeding of the first International Conference on Genetic Algorithm and their Application, pp. 41-49, 1987.