

# الگوریتم بهینه‌سازی ترکیبی با استفاده از الگوریتم ژنتیک و جستجوی ابرکروی مبتنی بر نظریه آشوب

سکینه سهرابی<sup>۱</sup> محمد کلانتری<sup>۲</sup> حمیدرضا رشیدی کنعان<sup>۳</sup>

۱- دانش آموخته کارشناسی ارشد- دانشکده مهندسی کامپیوتر - دانشگاه تربیت دبیر شهید رجایی - تهران - ایران -  
[s.sohrabi@sru.ac.ir](mailto:s.sohrabi@sru.ac.ir)

۲- استادیار- دانشکده مهندسی کامپیوتر - دانشگاه تربیت دبیر شهید رجایی - تهران - ایران  
[mkalantari@sru.ac.ir](mailto:mkalantari@sru.ac.ir)

۳- دانشکده مهندسی کامپیوتر - دانشگاه تربیت دبیر شهید رجایی - تهران - ایران  
[h.rashidykanan@sru.ac.ir](mailto:h.rashidykanan@sru.ac.ir)

**چکیده:** در این مقاله، یک الگوریتم بهینه‌سازی با استفاده از ترکیب الگوریتم ژنتیک و جستجوی ابرکروی مبتنی بر نظریه آشوب ارائه می‌شود که در آن به منظور افزایش کارایی در جستجوی جواب بهینه علاوه بر نظریه آشوب از عملگرهای ژنتیکی نیز استفاده شده است. این امر نه تنها منجر به قطعی بودن نتایج و کاهش انحراف معیار می‌شود، بلکه ضعف الگوریتم بهینه‌سازی جستجوی ابرکروی مبتنی بر نظریه آشوب یعنی سرعت همگرایی و عملکرد ضعیف در برخی از توابع محک را برطرف می‌نماید. نتایج حاصل از آزمایشات بیانگر همگرایی سریعتر و دقت بیشتر الگوریتم پیشنهادی در یافتن جواب بهینه نسبت به الگوریتم جستجوی ابرکروی استاندارد، الگوریتم جستجوی ابرکروی مبتنی بر نظریه آشوب و همچنین الگوریتم‌های بهینه‌سازی دیگر نظیر ژنتیک، ازدحام ذرات و جستجوی هارمونی است.

**کلمات کلیدی:** الگوریتم فراابتکاری مبتنی بر الگوریتم ژنتیک، الگوریتم جستجوی ابرکروی، الگوریتم ژنتیک، نظریه آشوب

تاریخ ارسال مقاله : ۱۳۹۶/۰۳/۲۱

تاریخ پذیرش مشروط مقاله: ۱۳۹۶/۰۸/۱۹

تاریخ پذیرش مقاله: ۱۳۹۶/۱۰/۱۶

نام نویسنده‌ی مسئول: دکتر محمد کلانتری

نشانی نویسنده‌ی مسئول: ایران - تهران - لویزان - خیابان شهید شعبانلو - دانشگاه تربیت دبیر شهید رجایی - دانشکده مهندسی کامپیوتر

## ۱- مقدمه

امروزه به کمک تکنیک‌های ابتکاری، در مواجهه با مسائل بهینه‌سازی می‌توان راه‌حل نزدیک به بهینه را در مدت زمان منطقی یافت. دسته‌ای از الگوریتم‌های تقریبی به نام الگوریتم‌های فراابتکاری با ترکیب اصول اولیه روش‌های ابتکاری به جای یافتن صرفاً جواب‌های بهینه یا نزدیک به بهینه، به دنبال روشی برای جستجوی مؤثر و کارای فضای جستجو هستند. این دسته از الگوریتم‌ها با استفاده از روش‌های گوناگون میان تنوع و افزایش فضای جستجو، تعادل ایجاد کرده [۱] و قادر به حل مشکلات بهینه‌سازی بزرگتر هستند [۳،۲]. سرعت بالای آنها سبب شده تا از عدم تضمین این روش‌ها در دستیابی به جواب‌های بهینه چشم‌پوشی شود [۴،۵]. برخی از الگوریتم‌های فراابتکاری عبارتند از الگوریتم جستجوی ابرکروی استاندارد [۶]، جستجوی هارمونی<sup>۱</sup> [۷]، تجمعی ذرات [۸] و الگوریتم ژنتیک [۹].

در این مقاله یک الگوریتم بهینه‌سازی با استفاده از ترکیب الگوریتم ژنتیک و جستجوی ابرکروی مبتنی بر نظریه آشوب ارائه شده است. بررسی نتایج بدست آمده بیانگر آن است که الگوریتم پیشنهادی ضمن آنکه ضعف الگوریتم بهینه‌سازی جستجوی ابرکروی مبتنی بر نظریه آشوب، مبنی بر سرعت همگرایی و عملکرد ضعیف در یافتن جواب بهینه در برخی از توابع محک را برطرف کرده است، همچنین از همگرایی سریع‌تر و دقت بیشتر در یافتن جواب بهینه نسبت به الگوریتم مذکور و همچنین الگوریتم‌های بهینه‌سازی دیگر نظیر ژنتیک، ازدحام ذرات، جستجوی هارمونی و الگوریتم جستجوی ابرکروی استاندارد برخوردار است.

ادامه مقاله به صورت زیر سازمان‌دهی شده است: در بخش دوم به توضیح الگوریتم جستجوی ابرکروی مبتنی بر نظریه آشوب پرداخته می‌شود که برای درک بهتر این الگوریتم، ابتدا الگوریتم جستجوی ابرکروی استاندارد به اختصار توضیح داده می‌شود. در بخش سوم الگوریتم پیشنهادی توضیح داده خواهد شد. در بخش چهارم نتایج حاصل از اجرای الگوریتم پیشنهادی بررسی می‌شود و نتیجه‌گیری در بخش پنجم ارائه می‌شود.

## ۲- الگوریتم جستجوی ابرکروی مبتنی بر نظریه آشوب

الگوریتم جستجوی ابرکروی مبتنی بر نظریه آشوب نسخه بهبود یافته‌ی الگوریتم جستجوی ابرکروی استاندارد است. الگوریتم جستجوی ابرکروی استاندارد اولین بار توسط کرمی و همکاران در سال ۲۰۱۴ ارائه شده است [۶] و در مسائل مهندسی از قبیل هماهنگی رله‌ها و دیسپاچینگ منابع انرژی بهینه در میکروگریدهای خانگی از آن استفاده شده است [۱۰،۱۱]. در ادامه به توضیح هر یک از الگوریتم‌های مذکور پرداخته می‌شود.

## ۲-۱- الگوریتم جستجوی ابرکروی استاندارد

در الگوریتم جستجوی ابرکروی استاندارد همانند اکثر الگوریتم‌های فراابتکاری، الگوریتم با تولید ذرات اولیه به صورت تصادفی شروع بکار می‌کند که هر ذره بیانگر یک جواب کاندید برای مساله پیش‌رو می‌باشد. هر ذره دارای دو قسمت متغیرها و مقدار تابع هدف می‌باشد. متغیرهای یک ذره نشان‌دهنده‌ی مقادیر اختصاص داده شده به متغیرهای تابع محک مورد بررسی می‌باشند و مقدار تابع هدف یک ذره بیانگر نتیجه حاصل از اجرای الگوریتم برای تابع محک مورد نظر به ازای مقادیر موجود در قسمت متغیرها است. در این الگوریتم از بین ذرات اولیه‌ی تولید شده، بهترین ذرات به عنوان مراکز کره انتخاب می‌شوند و بقیه‌ی ذرات بر اساس معیاری که در گام اول الگوریتم توضیح داده خواهد شد، به کره‌ها تخصیص می‌یابند. فرآیند الگوریتم مذکور از پنج گام به قرار زیر تشکیل می‌شود.

## • گام اول: مقداردهی اولیه و تولید ذرات

در این گام پس از آنکه پارامترهای اولیه الگوریتم مقداردهی شدند، جمعیت اولیه به صورت تصادفی تولید می‌شود و به تعداد معین از بین بهترین ذرات (ذراتی با کمترین مقدار تابع هدف) مراکز کره انتخاب می‌شوند. سپس مابقی ذرات بر اساس معیار غلبه مراکز کره‌ها در میان کره‌ها توزیع می‌شوند. معیار غلبه مراکز کره‌ها با استفاده از رابطه (۱) بدست می‌آید.

$$D_{sc} = \left| \frac{OFD_{sc}}{\sum_{i=1}^{N_{sc}} OFD_i} \right| \quad (1)$$

در رابطه‌ی (۱) مقدار  $OFD^2$  برای هر مرکز کره برابر است با اختلاف تابع هدف آن مرکز با بیشینه مقدار تابع هدف از بین همه‌ی مراکز کره‌ها که با استفاده از رابطه (۲) محاسبه می‌شود.

$$OFD_{sc} = f_{sc} - \max_{j \in SC_s} \{f_j\} \quad (2)$$

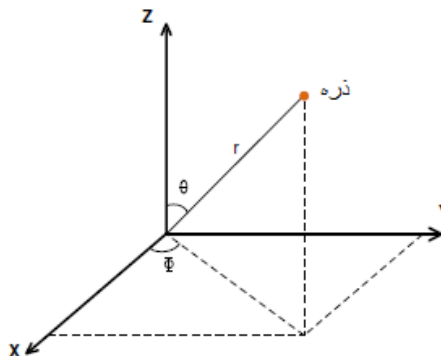
در این رابطه  $f_{sc}$  عبارت است از مقدار تابع هدف مرکز کره‌ی مورد بررسی و  $SC_s^2$  بیانگر مجموعه‌ی مراکز کره‌ها می‌باشد.

## • گام دوم: جستجو

در این گام یک ذره در فضای محدود شده به وسیله یک کره به دنبال یک پاسخ بهتر می‌گردد. فرآیند جستجو با تغییر پارامترهای ذره در مختصات کروی یعنی  $r$  و  $\theta$  انجام می‌شود. باید توجه داشت که در یک فضای  $N$  بعدی،  $N-1$  زاویه در مختصات کروی وجود دارد که تغییر هر یک منجر به جابه‌جایی ذره در فضای جستجو می‌شود. برای مثال همان‌طور که در شکل (۱) نشان داده شده است، در یک فضای ۳ بعدی، یک نقطه دو زاویه با محور مختصات می‌سازد. در این مرحله،  $N-1$  زاویه با احتمال معین، به اندازه  $\alpha$  رادیان تغییر می‌کند. مقدار  $\alpha$  به صورت تصادفی با یک توزیع یکنواخت بین  $(0, 2\pi)$  در هر تکرار تعیین می‌شود. در نهایت مقدار شعاع کره، فاصله بین ذرات و مرکز کره متناظر با آن‌ها، به صورت تصادفی در بازه  $[r_{min}, r_{max}]$  انتخاب می‌شود که  $r_{max}$  در یک کره  $N$  بعدی با استفاده از رابطه (۳) محاسبه می‌شود.

$$r_{max}^2 = \sum_{i=1}^N (p_{i,center} - p_{i,particle})^2 \quad (3)$$

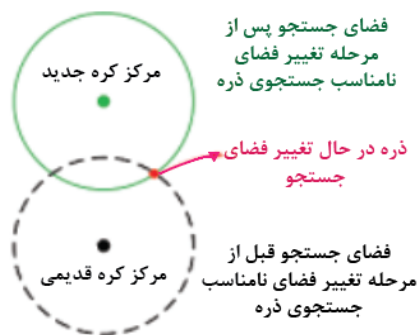
در رابطه‌ی (۳) متغیر  $N$  بی‌انگر تعداد متغی‌رهای تابع محک مورد بررسی بوده و مقدار متغی‌رهای  $P_{i,particle}$  و  $P_{i,center}$  به ترتی‌ب برابر مقدار متغی‌ر  $i$ ام در ذره و مرکز کره‌ی مورد بررسی می‌باشد.



شکل (۱): زاویه‌های یک ذره در فضای سه بعدی

#### • گام سوم: تغییر فضای جستجوی نامناسب

هر مرکز کره و ذراتی که به آن تعلق گرفته‌اند، یک مجموعه ذرات را به وجود می‌آورند. بعضی ذرات موجود در این مجموعه دارای مقدار تابع هدف نامناسبی هستند. بنابراین، انتظار نمی‌رود که آن‌ها بتوانند به مقدار مطلوب برای تابع هدف مورد بررسی برسند. این ذرات نامناسب شناسایی شده و به کره‌های دیگر اختصاص می‌یابند تا بتوانند فضای مناسب‌تری را جستجو نمایند. برای مثال می‌توانید شکل (۲) را مشاهده نمایید.



شکل (۲): تغییر فضای جستجوی نامناسب

باید توجه داشت که بعد از اتمام فرآیند جستجو در هر تکرار از اجرای الگوریتم، اگر یک ذره به موقعیتی با مقدار تابع هدف کم‌تر از مرکز کره متناظر با خود برسد، در این حالت ذره مورد نظر به عنوان مرکز کره شناخته شده و مرکز کره قبلی به عنوان ذره، حول مرکز کره جدید، فضا را جستجو می‌کند.

#### • گام چهارم: تولید ذرات جدید

به منظور منعطف‌تر نمودن الگوریتم و برای جستجوی بهتر فضا در هر تکرار از اجرای الگوریتم، تعداد مشخصی از بدترین ذرات، ذراتی که دارای مقدار تابع هدف بیشتری می‌باشند، حذف می‌شوند و به همان تعداد، ذرات جدید تولید و جایگزین می‌شوند. لازم به ذکر است این تعداد نباید زیاد باشد. برای تولید ذرات جدید مانند گام اول عمل می‌شود.

#### • گام پنجم: آزمون همگرایی

الگوریتم با رسیدن به یکی از شرایط زیر خاتمه می‌یابد: الف) تعداد تکرار اجرای الگوریتم به یک تعداد معین برسد. ب) اختلاف بین بهترین کره از بین کره‌های موجود در دو تکرار متوالی کمتر از مقدار آستانه تعیین شده شود.

### ۲-۲- الگوریتم جستجوی ابرکروی مبتنی بر نظریه

#### آشوب

الگوریتم جستجوی ابرکروی مبتنی بر نظریه آشوب همانند الگوریتم جستجوی ابرکروی استاندارد از پنج گام که در بخش ۲-۱ به اختصار توضیح داده شد، تشکیل می‌شود. سیستم‌های آشوب در ظاهر پدیده‌ای تصادفی بوده اما در باطن، دارای طبیعتی قطعی هستند [۱۲]، به همین جهت استفاده از نظریه آشوب منجر به بهبود نتایج شده و علی‌رغم تصادفی بودن می‌توان به قطعی بودن نتایج حاصل اطمینان داشت [۷، ۱۳]. برخلاف الگوریتم جستجوی ابرکروی استاندارد که در آن دو گام تولید ذره (گام اول و چهارم) و جستجو (گام دوم)، از اعداد تصادفی با توزیع یکنواخت استفاده می‌شود، در گام‌های مذکور از الگوریتم جستجوی ابرکروی مبتنی بر نظریه آشوب، مقادیر حاصل از نگاشت‌های آشوب جایگزین این مقادیر می‌گردد. از این رو در گام اول و چهارم برای تولید ذره از نگاشت چبیشف [۱۳] که در رابطه (۴) آورده شده است، استفاده می‌شود.

$$x_{k+1} = \cos(k \cdot \cos^{-1}(x_k)) \quad (4)$$

و در گام جستجو از نگاشت لیوویچ [۱۳] که در رابطه (۵) آورده شده است، برای تعیین مقدار  $\alpha$  استفاده می‌شود.

$$x_{k+1} = \begin{cases} \gamma x_k, & 0 < x_k < p_1 \\ \frac{p_2 - x_k}{p_2 - p_1}, & p_1 < x_k < p_2 \\ 1 - \beta(1 - x_k), & p_2 < x_k < 1 \end{cases} \quad (5)$$

### ۳- الگوریتم پیشنهادی

الگوریتم پیشنهادی نیز مانند الگوریتم جستجوی ابرکروی مبتنی بر نظریه آشوب از پنج گام تشکیل می‌شود با این تفاوت که در آن برای منعطف‌تر نمودن و افزایش سرعت و دقت، ضمن استفاده از نظریه آشوب از الگوریتم ژنتیک نیز استفاده شده است.

#### ۳-۱- الگوریتم ژنتیک

این الگوریتم یکی از روش‌های بهینه‌سازی قدرتمند است که براساس فرآیند تکامل طبیعی می‌باشد و یک روش جستجو است که از اصول انتخاب طبیعی داروین و تکنیک‌های زیست‌شناسی استفاده می‌کند [۱۴]. برای درک جزئیات الگوریتم ژنتیک به [۱۵] مراجعه نمایید. لازم به ذکر است که الگوریتم ژنتیک یک ابزار سودمند برای بازشناسی الگو، انتخاب ویژگی و یادگیری ماشینی است [۱۶]. از دیگر کاربردهای

الگوریتم مذکور می‌توان به طراحی بهینه یک سیستم قدرت ترکیبی چند منبعی [۱۷] و تعیین فاصله خطا در خطوط انتقال چند پایانه‌ای مبتنی بر امواج سیار [۱۸] اشاره کرد. به طور کلی اجزاء این الگوریتم عبارت‌اند از کروموزوم<sup>۴</sup>، جمعیت، تابع برازندگی<sup>۵</sup> و عملگرهای ژنتیکی که از این عملگرها برای تولیدمثل<sup>۶</sup> استفاده می‌شود. به این صورت که با تاثیر این عملگرها بر روی یک جمعیت، نسل<sup>۷</sup> بعدی آن جمعیت تولید می‌شود. عملگرهای انتخاب<sup>۸</sup>، آمیزش<sup>۹</sup> و جهش<sup>۱۰</sup>، معمولاً بیشترین کاربرد را در الگوریتم‌های ژنتیکی دارند.

### ۳-۲- الگوریتم ترکیبی پیشنهادی

در این بخش الگوریتم پیشنهادی ارائه می‌شود که در آن برای منعطف‌تر نمودن و افزایش سرعت و دقت، در دو گام جستجو و تولید ذرات جدید (گام دوم و چهارم) ضمن استفاده از نظریه آشوب از الگوریتم ژنتیک نیز استفاده می‌شود و گام‌های اول، سوم و پنجم بدون تغییر باقی می‌مانند. روند اجرای الگوریتم پیشنهادی در شکل (۴) آورده شده است. در ادامه گام‌ها دوم و چهارم توضیح داده می‌شود.

#### • گام دوم: جستجو

در این گام یک ذره به دو روش که احتمال انتخاب آن‌ها برابر است، فضای مساله را جستجو می‌کند:

- مانند گام دوم الگوریتم جستجوی ابرکروی مبتنی بر نظریه آشوب عمل کرده و جستجو با تغییر پارامترهای ذره در مختصات کروی یعنی  $r$  و  $\theta$  انجام می‌شود. میزان تغییر هر یک از پارامترها براساس مقادیر حاصل از نگاشت لیپوویچ تعیین می‌شود.
- از عملگر جهش الگوریتم ژنتیک استفاده می‌کنیم، به این صورت که برای جستجوی هر ذره (کروموزوم) یک ژن از کروموزوم مورد بررسی و یک کروموزوم از بین بهترین کروموزوم‌ها به صورت تصادفی انتخاب می‌شود و محتوای ژن در کروموزوم مدنظر به محتوای ژن متناظر در کروموزوم انتخابی، تغییر می‌یابد.

#### • گام چهارم: تولید ذرات جدید

در این گام ذرات جدید به سه روش زیر که احتمال انتخاب هر روش برابر است، تولید می‌شوند. این امر منجر به انعطاف بیشتر الگوریتم می‌شود.

- مانند گام چهارم الگوریتم جستجوی ابرکروی مبتنی بر نظریه آشوب عمل کرده و ذرات به کمک مقادیر حاصل از نگاشت چبیشف تولید می‌شوند.
- در این روش ذرات تولید شده در گام اول الگوریتم، به عنوان جمعیت اولیه به الگوریتم ژنتیک داده می‌شوند. در الگوریتم ژنتیک هر ذره نقش یک کروموزوم را خواهد داشت به این صورت که مقدار متغیرهای موجود در قسمت متغیرها، بیانگر مقادیر ژن-ها در کروموزوم متناظر و مقدار تابع هدف ذره مشخص‌کننده مقدار تابع برازندگی کروموزوم مورد بررسی می‌باشد. در ادامه الگوریتم ژنتیک به تعداد تکراری برابر  $m$  ( $m$  تعداد ذرات جدید

است) اجرا می‌شود و پس از آنکه ذرات حاصل براساس مقدار تابع هدف‌شان (تابع برازندگی) مرتب شدند، در نهایت  $m$  ذره، از بین بهترین ذرات انتخاب شده و به عنوان ذرات جدید به الگوریتم داده می‌شوند.

۳. در این روش عملگری الهام گرفته شده از عملگر آمیزش الگوریتم ژنتیک به تعداد  $m$  مرتبه اجرا می‌شود. همان‌طور که در شکل (۳) مشاهده می‌کنید این عملگر به جای عمل بر روی زوج کروموزوم، بر روی بهترین کروموزوم‌ها (ذراتی را که به عنوان مراکز کره تعیین شده‌اند) عمل می‌کند و در هر تکرار به جای تولید دو کروموزوم، یک کروموزوم تولید می‌کند. در هر بار اجرای این عملگر، به ازاء هر ژن (هر متغیر) یک کروموزوم به صورت تصادفی انتخاب شده و محتوای ژن مدنظر در کروموزوم جدید برابر محتوای ژن متناظر در کروموزوم انتخابی می‌شود.



شکل (۳): روند اجرای عملگر پیشنهادی

### ۴- نتایج آزمایشات

در این قسمت با اجرای الگوریتم پیشنهادی بر روی توابع محک استاندارد، کارایی الگوریتم پیشنهادی را بررسی می‌کنیم. این آزمایش بیست مرتبه تکرار می‌شود و کمینه نتایج حاصل از اجرای الگوریتم-های مورد بررسی بر روی توابع محک و انحراف معیار نتایج مورد بررسی قرار می‌گیرد. برای معتبر بودن نتایج مقایسه، فضای جستجو برای هر یک از توابع در همه‌ی الگوریتم‌ها یکسان در نظر گرفته شده است. شرایط در نظر گرفته شده برای انجام آزمایش در جدول (۱) آورده شده است که باید توجه داشت تعداد تکرار اجرای الگوریتم پیشنهادی در هر بار اجرا برابر ۴۰ در نظر گرفته شده است و این تعداد نسبت به الگوریتم‌های جستجوی ابرکروی استاندارد و الگوریتم جستجوی ابرکروی مبتنی بر نظریه آشوب کمتر است. مدت زمان اجرای الگوریتم‌های مورد بررسی در جدول (۲)، توابع محک [۱۹] مورد استفاده در این مقاله در جدول (۳)، و نتایج حاصل از آزمایش در جدول (۴) آورده شده است. مقدار آستانه برای اختلاف بین بهترین کره از بین کره‌های موجود در دو تکرار متوالی را برابر  $10^{-3}$  در نظر گرفته‌ایم.

در الگوریتم ژنتیک، احتمال جهش را برابر ۰,۰۹ در الگوریتم ازدحام ذرات ضریب انقباض، پارامترهای شناختی و اجتماعی را برابر ۲ و در الگوریتم جستجوی هارمونی اندازه حافظه هارمونی، نرخ در نظر گرفتن حافظه هارمونی<sup>۱۱</sup> و نرخ تطبیق گام<sup>۱۲</sup> را به ترتیب برابر ۰,۶، ۰,۳ و ۰,۳ در نظر گرفته‌ایم.

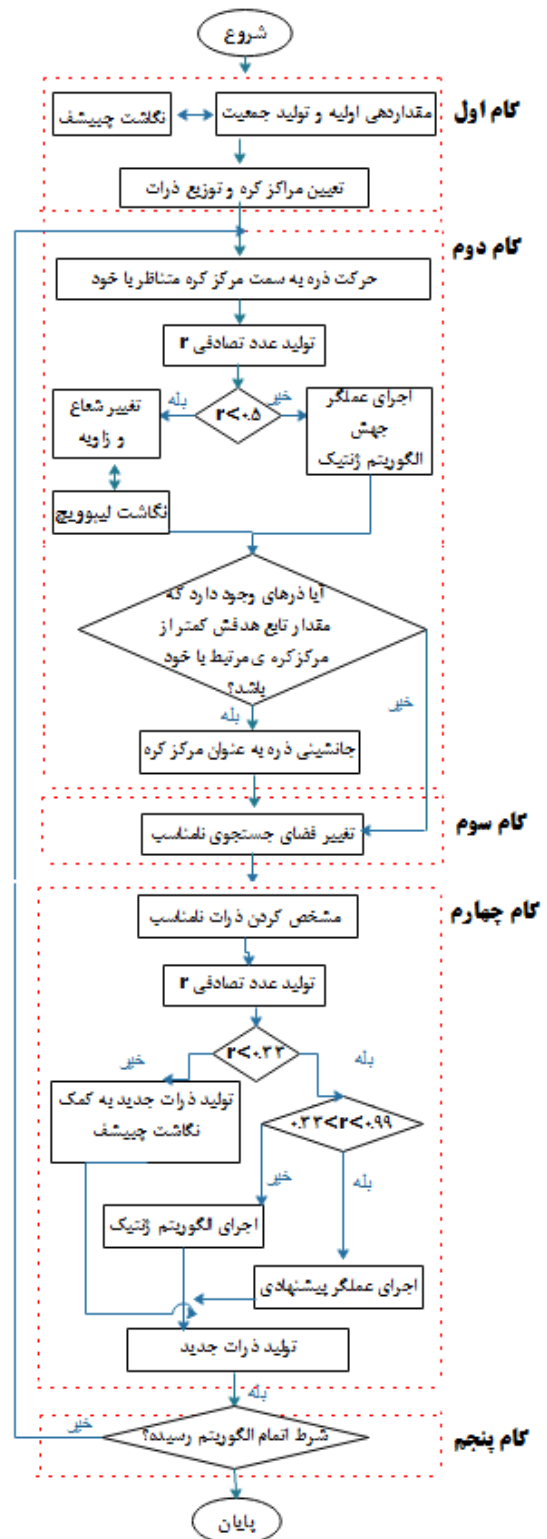
جدول (۲): مدت زمان (بر حسب میلی ثانیه) اجرای الگوریتم‌های مورد بررسی برای برخی از توابع محک استاندارد

الگوریتم					
تابع	CHSS_GA	CHSS	HSSA	GA	PSO
F1	5.063	2.545	2.437	0.2178	0.3425
F2	0.704	0.483	0.486	0.2134	0.3297
F3	1.827	0.437	0.593	0.2099	0.2605
F4	0.860	0.469	0.584	0.2104	0.2743
F5	1.266	0.683	0.714	0.2214	0.2934

با توجه به نتایج آورده شده در جدول (۴) که شامل مقدار انحراف معیار ( $SD^{13}$ ) و مقدار کمینه (Min) می‌شود، می‌توان مشاهده کرد الگوریتم جستجوی ابرکروی مبتنی بر نظریه آشوب در تابع F5 در رسیدن به مقدار کمینه نسبت به الگوریتم ازدحام ذرات کمی ضعیف‌تر عمل کرده و الگوریتم پیشنهادی ضعف این الگوریتم را برطرف کرده است. با توجه به تابع F4 می‌توان مشاهده کرد که در صورت برابری مقدار کمینه، انحراف معیار الگوریتم پیشنهادی از سایر الگوریتم‌ها کمتر می‌باشد. در جدول (۴) کمینه نتایج به صورت پرننگ نمایش داده شده است. نمودار میانگین مراکز کره، میانگین ذرات و مقدار کمینه نتایج حاصل از اجرای الگوریتم پیشنهادی در شکل (۵) آورده شده است. در تمامی توابع مورد بررسی، با افزایش تعداد تکرار، مقدار کمینه بهبود یافته و همچنین مقادیر میانگین ذرات و میانگین مراکز کره به مقدار کمینه نزدیک می‌شود.

#### ۴- نتیجه‌گیری

الگوریتم پیشنهادی در راستای برطرف کردن ضعف الگوریتم جستجوی ابرکروی مبتنی بر نظریه آشوب در یافتن جواب بهینه در برخی توابع محک استاندارد ارائه شده است که ضمن برخورداری از همگرایی مناسب‌تر، باعث کاهش انحراف معیار نتایج نیز می‌شود. در الگوریتم مذکور به منظور افزایش کارایی علاوه بر نظریه آشوب به عملگرهای ژنتیکی نیز استفاده شده است که این امر نه تنها منجر به قطعی بودن نتایج حاصل و کاهش انحراف معیار می‌شود، بلکه ضعف الگوریتم بهینه‌سازی جستجوی ابرکروی مبتنی بر نظریه آشوب، مبنی بر سرعت همگرایی و عملکرد نسبتاً ضعیف در برخی از توابع محک را نیز برطرف می‌نماید. بررسی نتایج بیانگر همگرایی مناسب و توانایی بالای الگوریتم پیشنهادی در یافتن جواب بهینه نسبت به الگوریتم‌های جستجوی ابرکروی استاندارد، ژنتیک، ازدحام ذرات، جستجوی هارمونی و الگوریتم جستجوی ابرکروی مبتنی بر نظریه آشوب است.



شکل (۴): روند اجرای الگوریتم پیشنهادی

جدول (۱): شرایط در نظر گرفته شده برای انجام آزمایش

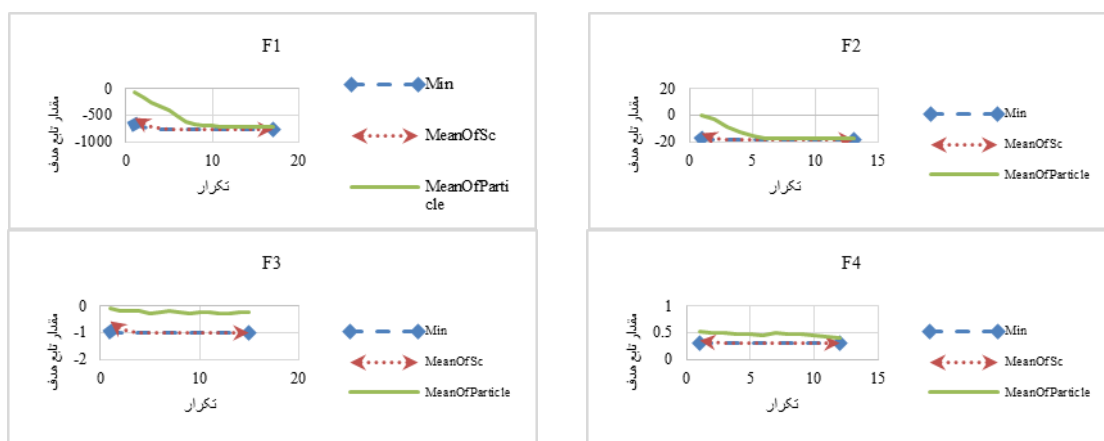
الگوریتم	HSS	CHSS	CHSS_GA
تعداد تکرار آزمایش	20	20	20
تعداد تکرار در هر بار اجرا	100	70	40
تعداد جمعیت اولیه ذرات	1000	1000	1000
تعداد مرکز کره	50	50	50

جدول (۳): توابع هدف

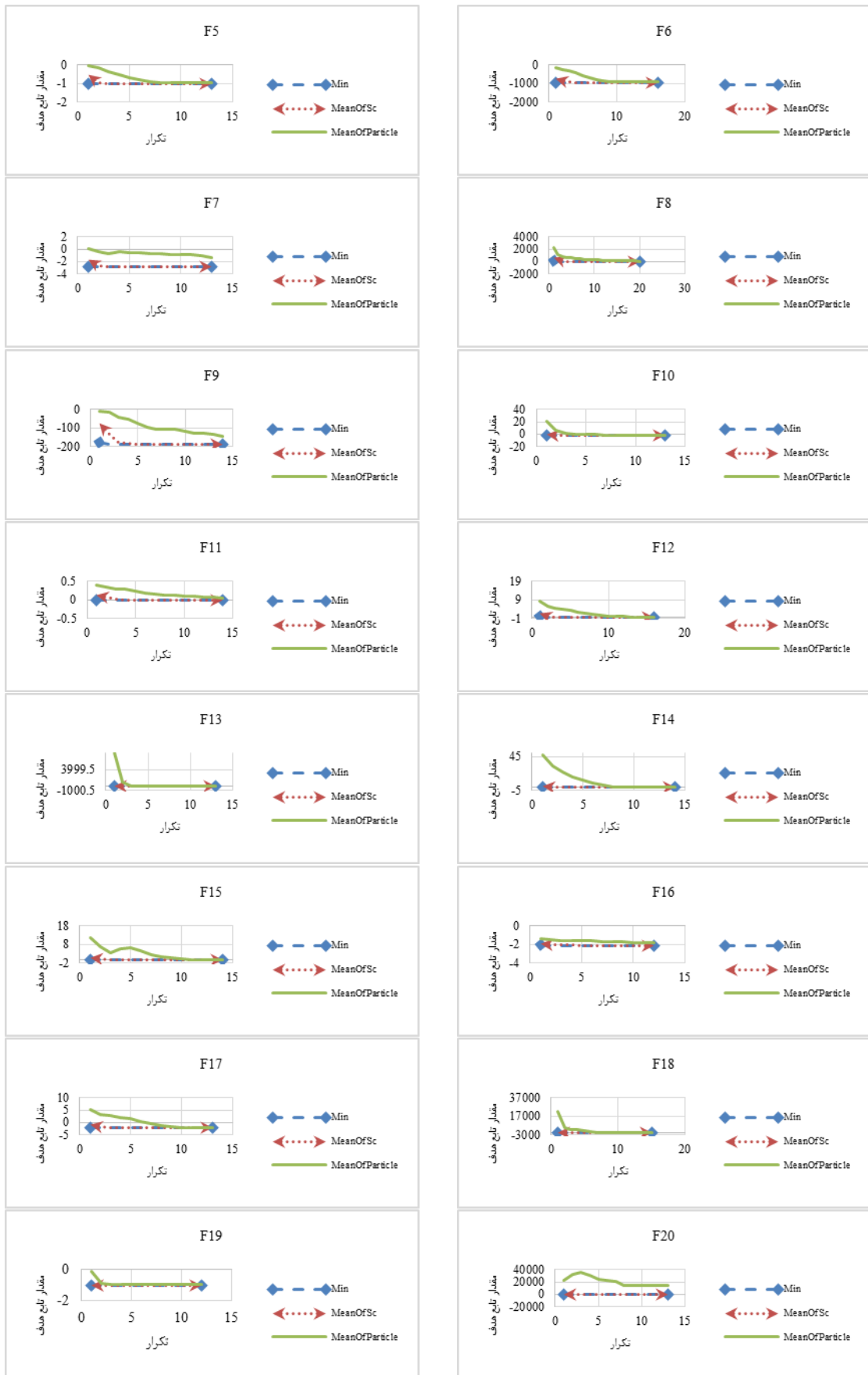
دامنه	فرمول	شناسه
$-20 < x_1 < 15$ $5 < x_2 < 12$ $-12 < x_3 < 7$ $15 < x_4 < 25$ $18 < x_5 < 26$	$f(x_1, x_2, x_3, x_4, x_5) = x_2 (\sin(x_1)x_3) + x_4 (\cos(x_2 - x_4)).x_5$	F1
$-20 < x_1, x_2 < 20$	$f(x_1, x_2) = x_1 (\sin(4x_1)) + 1.1x_2 (\sin(2x_2))$	F2
$-20 < x_1, x_2 < 20$	$f(x_1, x_2) = -\frac{1 + \cos(12\sqrt{x_1^2 + x_2^2})}{0.5(x_1^2 + x_2^2) + 2}$	F3
$-10 < x_1, x_2 < 10$	$f(x_1, x_2) = \frac{1 + \cos^2(\sin^2(x_1^2 + x_2^2)) - 0.5}{1 + (0.01(x_1^2 + x_2^2))}$	F4
$-5 < x_1, x_2 < 5$	$f(x_1, x_2) = -(\cos(x_1) \cdot \cos(x_2)) \cdot \exp(-(x_1 - \Pi)^2 - (x_2 - \Pi)^2)$	F5
$-512 < x_1, x_2 < 512$	$f(x_1, x_2) = -(x_2 + 47) \cdot \sin\left(\sqrt{x_2 + \frac{x_1}{4} + 47}\right) - x_1 \sin\left(\sqrt{x_1 + (x_2 + 47)}\right)$	F6
$0.5 < x < 10$	$f(x) = \frac{\sin(10\Gamma x)}{2x} + (x-1)^4$	F7
$-36 < x_i < 36$ $i = 1, 2, \dots, 6$	$f(x_1, x_2, x_3, x_4, x_5) = \sum_{i=1}^6 (x_i - 1)^2 - \sum_{i=1}^6 x_i x_{i-1}$	F8
$-5.12 < x_1, x_2 < 5.12$	$f(x_1, x_2) = (\sum_{i=1}^6 i \cdot \cos((i+1)x_1 + i)) \cdot (\sum_{i=1}^6 i \cdot \cos((i+1)x_2 + i))$	F9
$-3 < x_1 < 3$ $-2 < x_2 < 2$	$f(x_1, x_2) = (4 - 2.1x_1^2 + \frac{x_1^4}{3})x_1^2 + x_1x_2 + (-4 + 4x_2^2)x_2^2$	F10
$-10 < x_1, x_2 < 10$	$f(x_1, x_2) = 0.5 + \frac{\sin^2(x_1^2 + x_2^2) - 0.5}{1 + 0.01(x_1^2 + x_2^2)}$	F11
$-100 < x_1, x_2 < 100$	$f(x_1, x_2) = 0.1 \times (10 \times 2 + (x_1^2 - 10 \cos(2\Gamma x_1)) + (x_2^2 - 10 \cos(2\Gamma x_2)))$	F12
$-20 < x_1, x_2 < 20$	$f(x_1, x_2) = (1.5 - x_1 + x_1x_2)^2 + (2.25 - x_1 + x_1x_2^2)^2 + (2.625 - x_1 + x_1x_2^3)^2$	F13
$-5 < x_1, x_2 < 5$	$f(x_1, x_2) = (x_1 + 2x_2 - 7)^2 + (2x_1 + x_2 - 5)^2$	F14
$-5 < x_1, x_2 < 5$	$f(x_1, x_2) = \sin^2(3\Gamma x_1) + (x_1 - 1)^2 [1 + \sin(3\Gamma x_2^2)] + (x_2 - 1)^2 [1 + \sin(2\Gamma x_1^2)]$	F15
$-20 < x_1, x_2 < 20$	$f(x_1, x_2) = -0.0001 \left( \left  \sin(x_1) \sin(x_2) \exp\left(100 - \frac{\sqrt{x_1^2 + x_2^2}}{\Pi}\right) \right  + 1 \right)^{0.1}$	F16
$-1.5 < x_1 < 4$ $-3 < x_2 < 4$	$f(x_1, x_2) = \sin(x_1 + x_2) + (x_1 - x_2)^2 - 1.5x_1 + 2.5x_2 + 1$	F17
$-20 < x_1, x_2 < 20$	$f(x_1, x_2) = \left[ 1 + (x_1 + x_2 + 1)^2 \cdot (19 - 14x_1 + 3x_1^2 - 14x_2 + 6x_1x_2 + 3x_2^2) \right] \times$ $\left[ 30 + (2x_1 - 3x_2)^2 \cdot (18 - 32x_1 + 12x_1^2 + 48x_2^2 - 36x_1x_2 + 27x_2^2) \right]$	F18
$-20 < x_1, x_2 < 20$	$f(x_1, x_2) = \frac{1}{51.95} \left[ \left( \frac{\bar{x}_2 - 5.1\bar{x}_1}{4\pi^2} + \frac{5\bar{x}_1}{\pi} - 6 \right)^2 + \left( 10 - \frac{10}{8\pi} \right) \cos(\bar{x}_1) - 44.81 \right]$ where $\bar{x}_1 = 15x_1 - 5$ , $\bar{x}_2 = 15x_2$	F19
$-20 < x_1, x_2 < 20$	$f(x_1, x_2) = \sin(x_2) \cdot \exp(1 - \cos(x_1))^2 + \cos(x_1) \cdot \exp(1 - \sin(x_2))^2 + (x_1 - x_2)^2$ subjected to: $(x_1 + 5)^2 + (x_2 + 5)^2 < 25$	F20

جدول (۴): نتایج حاصل از توابع محک استاندارد با استفاده از الگوریتم‌های مورد بررسی

تابع	معیار	الگوریتم					
		CHSS_GA	CHSS	HSS	HSA	PSO	GA
F1	Min (SD)	<b>-761.1508</b> (0.4184)	-755.58 (0.2005)	-749.21 (0.5421)	-670.135 (55.789)	-700.371 (28.3438)	-674.498 (26.2315)
F2	Min (SD)	<b>-39.75</b> (4.23E-5)	-39.75 (4.31e-4)	-39 (0.0077)	-38.29 (2.92)	-38.88 (3.81)	-38.17 (2.30)
F3	Min (SD)	-0.936 (0.023)	-0.924 (0.008)	-0.9924 (0.0255)	-0.926 (0.22)	-1 (0.096)	<b>-1</b> (0.0014)
F4	Min (SD)	<b>0.2951</b> (4.34E-12)	0.2951 (6.92E-6)	0.2951 (1.04E-04)	0.2955 (0.0256)	0.2951 (0.0024)	0.2951 (9.436E-06)
F5	Min (SD)	<b>-1</b> (3.02E-06)	-0.99961 (1.46E-6)	-0.9995 (2.89E-04)	-0.9955 (0.2000)	-0.9985 (0.0336)	-1 (4.42E-06)
F6	Min (SD)	<b>-959.641</b> (7.65E-06)	-959.641 (3.988E-05)	-959.641 (1.55E-04)	-930.4477 (104.35)	-946.517 (34.0872)	-885.757 (74.6123)
F7	Min (SD)	<b>-2.8739</b> (0.003)	-2.8739 (0.006)	-2.8739 (0.001)	-0.81 (0.17)	-0.86 (0.21)	-2.8739 (0.37)
F8	Min (SD)	<b>-49.99</b> (4.35E-01)	-42.69 (2.2045)	-36.502 (6.020)	-23.6161 (145.14)	-5 (3.4281)	-40.16 (4.943)
F9	Min (SD)	<b>-186.73</b> (5.78E-04)	-186.72 (0.0389)	-186.65 (0.4766)	-185.51 (0.4469)	-186.401 (18.81)	-186.185 (3.6876)
F10	Min (SD)	<b>-1.031</b> (5.11E-09)	-1.031 (1.48E-04)	-1.030 (0.002)	-1.026 (0.2080)	-1.030 (0.0132)	-1.031 (0.0021)
F11	Min (SD)	<b>0</b> (2.04E-08)	4.19E-05 (3.94E-04)	5.69E-06 (0.00222)	0.0155 (0.033)	8.94E-08 (1.30E-03)	0 (0.0083)
F12	Min (SD)	0.0013 (0.140)	1.185 (0.231)	0.535 (0.277)	1.07 (5.88)	0 (0.382)	<b>0</b> (0.001)
F13	Min (SD)	<b>2.27E-6</b> (0.016)	0.001 (0.007)	0.003 (0.007)	0.42 (23.51)	0.32 (28.27)	3.76E-5 (0.002)
F14	Min (SD)	<b>1.69E-18</b> (1.78E-07)	2.30E-04 (0.0022)	0.001422 (0.00101)	0.0078 (12.465)	1.03E-08 (2.91E-04)	0.0135 (0.2301)
F15	Min (SD)	<b>1.35E-31</b> (9.50E-05)	7.76E-04 (2.21E-03)	1.37E-04 (4.39E-04)	0.0142 (0.1494)	4.48E-09 (4.52E-05)	1.35E-31 (0.012)
F16	Min (SD)	-2.062 (1.19E-4)	-2.0626 (0.84)	-2.062 (2.39)	-2.0626 (0.05)	-2.0626 (0.0122)	<b>-2.0626</b> (4.15E-5)
F17	Min (SD)	<b>-1.91322</b> (8.28E-07)	-1.91117 (5.49E-04)	-1.91210 (8.33E-04)	-1.9113 (0.1206)	-1.9132 (0.0022)	-1.9127 (0.0111)
F18	Min (SD)	<b>3.68</b> (1.40E-05)	7.43 (2.44E-01)	17.74 (0.120489)	82.43 (7.15E+05)	9.12 (255.78)	3.008 (0.2562)
F19	Min (SD)	-1.0474 (0.001)	-1.0473 (0.008)	-1.0473 (0.002)	-1.029 (1.40)	-1.0472 (0.0900)	<b>-1.0474</b> (2.84E-04)
F20	Min (SD)	<b>-106.78</b> (2.01)	-106.18 (0.84)	-105.02 (2.39)	-103.87 (13.74)	-104.30 (19.91)	-106.78 (2.16)



شکل (۵): نمودار مقدار کمینه، میانگین مراکز کره و میانگین ذرات حاصل از توابع محک با استفاده از الگوریتم پیشنهادی



شکل (۵): ادامه



## تشکر و قدردانی

این پژوهش با حمایت مالی دانشگاه تربیت دبیر شهید رجایی طبق قرارداد شماره ۳۳۴۸۴ مورخ ۹۴/۱۲/۱۵ انجام گردیده است.

## مراجع

- Algorithm." International Arab Journal of Information Technology (IAJIT) 12.2 (2015).
- [۱۷] شهیری نیا امیرحسین، و دیگران. "طراحی بهینه سیستم قدرت ترکیبی چند منبعی مستقل از شبکه با استفاده از الگوریتم ژنتیک." نشریه مهندسی برق و الکترونیک ایران ۲۸ (۲۰۱۲).
- [۱۸] مهدی قاضی‌زاده احسائی. "روشی دقیق برای تعیین فاصله خطا در خطوط انتقال چند پایانه‌ای مبتنی بر امواج سیار و الگوریتم ژنتیک." نشریه مهندسی برق و الکترونیک ایران ۱۴،۳ (۲۰۱۸).
- [19] Mallipeddi, Rammohan, and Ponnuthurai Nagaratnam Suganthan. "Problem definitions and evaluation criteria for the CEC 2010 competition on constrained real-parameter optimization." Nanyang Technological University, Singapore 24 (2010).

## زیر نویس‌ها

- <sup>1</sup> Harmony Search Algorithm
- <sup>2</sup> Objective Function Difference
- <sup>3</sup> Spherical Center
- <sup>4</sup> Chromosome
- <sup>5</sup> Fitness Function
- <sup>6</sup> Reproduction
- <sup>7</sup> Generation
- <sup>8</sup> Selection
- <sup>9</sup> Crossover
- <sup>10</sup> Mutation
- <sup>11</sup> Harmony Memory Consideration Rate
- <sup>12</sup> Pitch Adjusting Rate
- <sup>13</sup> Standard Deviation

- [1] Yang, Xin-She. Nature-inspired metaheuristic algorithms. Luniver press, 2010.
- [2] Yang, Xin-She. Engineering optimization: an introduction with metaheuristic applications. John Wiley & Sons, 2010.
- [3] Sahab, Mohammed Ghasem, Vassili V. Toropov, and Amir Hossein Gandomi. "A review on traditional and modern structural optimization: problems and techniques." Metaheuristic Applications in Structures and Infrastructures(2013): 25-47.
- [4] Nallakumar, R., N. Sengottaiyan, and KS Sruthi Priya. "A Survey on Scheduling and the Attributes of Task Scheduling in the Cloud." Int. J. Adv. Res. Comput. Commun. Eng 3 (2014): 8167-8171.
- [5] Tsai, Chun-Wei, and Joel JPC Rodrigues. "Metaheuristic scheduling for cloud: A survey." IEEE Systems Journal 8.1 (2014): 279-291.
- [6] Karami, Hossein, Mohammad Javad Sanjari, and Gevorg B. Gharehpetian. "Hyper-Spherical Search (HSS) algorithm: a novel meta-heuristic algorithm to optimize nonlinear functions." Neural Computing and Applications 25.6 (2014): 1455-1465.
- [7] Alatas, Bilal. "Chaotic harmony search algorithms." Applied Mathematics and Computation 216.9 (2010): 2687-2699.
- [8] Gandomi, Amir Hossein, et al. "Chaos-enhanced accelerated particle swarm optimization." Communications in Nonlinear Science and Numerical Simulation 18.2 (2013): 327-340.
- [9] Coello, Carlos A. Coello, and Efrén Mezura Montes. "Constraint-handling in genetic algorithms through the use of dominance-based tournament selection." Advanced Engineering Informatics 16.3 (2002): 193-203.
- [10] Ahmadi, S. A., et al. "Application of hyper-spherical search algorithm for optimal coordination of overcurrent relays considering different relay characteristics." International Journal of Electrical Power & Energy Systems 83 (2016): 443-449.
- [11] Sanjari, Mohammad Javad, et al. "Application of Hyper-Spherical Search algorithm for optimal energy resources dispatch in residential microgrids." Applied Soft Computing 37 (2015): 15-23.
- [12] Kellert, Stephen H., and Lawrence Sklar. "In the wake of chaos: Unpredictable order in dynamical systems." Philosophy of Science 64.1 (1997): 181.
- [13] Gandomi, Amir H., and Xin-She Yang. "Chaotic bat algorithm." Journal of Computational Science 5.2 (2014): 224-232.
- [14] Yan-hua, Zhang, Feng Lei, and Yang Zhi. "Optimization of cloud database route scheduling based on combination of genetic algorithm and ant colony algorithm." Procedia Engineering 15 (2011): 3341-3345.
- [15] McCall, John. "Genetic algorithms for modelling and optimisation." Journal of Computational and Applied Mathematics 184.1 (2005): 205-222.
- [16] Javidi, Mohammad, and Roghiyeh Hosseinpoufard. "Chaos Genetic Algorithm Instead Genetic