

# الگوریتم بهینه‌سازی جستجوی ابرکروی مبتنی بر نظریه آشوب

محمد کلانتری، سکینه سهرابی، حمیدرضا رشیدی کنعان و حسین کرمی

بهینه‌سازی، روش‌های فراابتکاری بسیار سریع‌تر از روش‌های دقیق عمل می‌نمایند و این مزیت سبب شده تا از عدم تضمین این روش‌ها در دستیابی به جواب‌های بهینه چشم‌پوشی شود. ضمن آن که رسیدن به یک جواب تقریبی و نزدیک بهینه کلی، بهتر از بی‌جواب ماندن با روش‌های دقیق و غیر ابتکاری است [۴] تا [۵].

در یکی از دسته‌بندی‌های ارائه‌شده، الگوریتم‌های فراابتکاری را به دو دسته الگوریتم‌های مبتنی بر جمعیت<sup>۲</sup> و الگوریتم‌های مبتنی بر یک جواب<sup>۳</sup> تقسیم‌بندی می‌کنند. الگوریتم‌های مبتنی بر یک جواب در حین فرایند جستجو یک جواب را تغییر می‌دهند در حالی که در الگوریتم‌های مبتنی بر جمعیت در حین جستجو، یک جمعیت از جواب‌ها در نظر گرفته می‌شوند [۶]. الگوریتم پیشنهادی در دسته الگوریتم‌های مبتنی بر جمعیت قرار می‌گیرد. پیشرفت‌های اخیر که در نظریه‌ها و برنامه‌های کاربردی دینامیک غیر خطی از جمله نظریه آشوب<sup>۴</sup> صورت گرفته است منجر به توجه بیشتر به این نظریه در بسیاری از زمینه‌ها شده است [۷]. یکی از این زمینه‌ها استفاده کاربردی از نظریه آشوب در الگوریتم‌های بهینه‌سازی است [۸]. پیش از این از نظریه آشوب در الگوریتم‌های بهینه‌سازی فراابتکاری مانند جستجوی هارمونی<sup>۵</sup> (HSA) [۹]، تجمعی ذرات<sup>۶</sup> (PSO) [۱۰] و کلونی زنبور عسل<sup>۷</sup> [۱۱] استفاده شده است. مطالعات تجربی نشان می‌دهد که استفاده از نظریه آشوب در الگوریتم‌های فراابتکاری می‌تواند باعث بهبود عملکرد الگوریتم بهینه‌سازی شود [۱۲].

در این مقاله نسخه بهبودیافته الگوریتم جستجوی ابرکروی استاندارد ارائه می‌شود که در این راستا از نظریه آشوب استفاده شده و مقادیر حاصل از نگاشت‌های آشوب جایگزین مقادیر تصادفی موجود در الگوریتم جستجوی ابرکروی استاندارد می‌شود. نتایج آزمایشات بر روی توابع محک استاندارد نشان می‌دهد که الگوریتم پیشنهادی که از ترکیب الگوریتم جستجوی ابرکروی استاندارد و نظریه آشوب به دست می‌آید، ضمن برخورداری از همگرایی سریع‌تر، دقت بیشتری نیز در یافتن جواب بهینه نسبت به الگوریتم جستجوی ابرکروی استاندارد و همچنین الگوریتم‌های ژنتیک<sup>۸</sup>، بهینه‌سازی تجمعی ذرات و جستجوی هارمونی دارد.

ادامه مقاله به صورت زیر سازماندهی شده است: در بخش دوم الگوریتم جستجوی ابرکروی توضیح داده می‌شود. در بخش سوم نظریه آشوب شرح داده خواهد شد. در بخش چهارم به توضیح الگوریتم پیشنهادی پرداخته می‌شود. در بخش پنجم نتایج حاصل از اجرای الگوریتم شرح داده خواهد شد و نتیجه‌گیری در بخش ششم ارائه می‌شود.

چکیده: در این مقاله الگوریتم بهینه‌سازی جستجوی ابرکروی با استفاده از نظریه آشوب ارائه می‌شود که ضعف الگوریتم بهینه‌سازی جستجوی ابرکروی استاندارد، یعنی سرعت همگرایی و افزایش تعداد تکرار اجرای الگوریتم برای رسیدن به جواب بهینه را برطرف می‌نماید. در الگوریتم پیشنهادی، در گام ایجاد ذرات و گام جستجو، مقادیر حاصل از دو نگاشت چیشف و لیوویچ، جایگزین مقادیر تصادفی موجود در الگوریتم استاندارد می‌شود که این امر باعث بهبود نتایج حاصل از اجرای الگوریتم شده و انحراف معیار نتایج را کاهش می‌دهد. نتایج شبیه‌سازی بر روی توابع محک استاندارد نشان می‌دهد که الگوریتم پیشنهادی ضمن برخورداری از همگرایی سریع‌تر، دقت بیشتری نیز در یافتن جواب بهینه نسبت به الگوریتم جستجوی ابرکروی استاندارد و همچنین الگوریتم‌های بهینه‌سازی دیگر نظیر ژنتیک، ازدحام ذرات و الگوریتم جستجوی هارمونی دارد.

کلیدواژه: الگوریتم فراابتکاری مبتنی بر نظریه آشوب، الگوریتم جستجوی ابرکروی، مسئله بهینه‌سازی، نظریه آشوب.

## ۱- مقدمه

امروزه بهینه‌سازی نقش مهمی در حل مسایل ایفا می‌کند. بسیاری از مسایل بهینه‌سازی پرکاربرد به صورت غیر خطی می‌باشند که معمولاً با تعداد متعددی بهینه محلی همراه هستند و در نتیجه حل این دسته از مسایل بسیار چالش‌برانگیز است [۱]. تکنیک‌های بهینه‌سازی، به خصوص الگوریتم‌های تکاملی به دلیل انعطاف‌پذیری و سادگی به طور گسترده برای حل مشکلات علمی و مسایل بهینه‌سازی مختلف استفاده می‌شوند [۲]. تکنیک‌های فراابتکاری به عنوان روش‌های بهینه‌سازی سراسری شناخته شده، در حل بسیاری از مسایل دنیای واقعی و مسایل بهینه‌سازی پیچیده موفق بوده‌اند [۳]. روش‌های ابتکاری<sup>۱</sup>، تکنیک‌های طراحی شده برای حل سریع‌تر مسایلی است که روش‌های حل معمولی در آنها بسیار کند عمل کرده و یا نمی‌توانند به جواب برسند. هدف الگوریتم‌های ابتکاری تولید جوابی منطقی در مدت زمانی معقول است. این جواب الزاماً جواب بهینه کلی مسئله نمی‌باشد اما نزدیک به آن است. الگوریتم‌های فراابتکاری، یکی از انواع الگوریتم‌های بهینه‌سازی تقریبی هستند که دارای راهکارهایی جهت برون‌رفت از نقاط بهینه محلی هستند. در حوزه

این مقاله در تاریخ ۲۹ مهر ماه ۱۳۹۶ دریافت و در تاریخ ۱۸ اردیبهشت ماه ۱۳۹۷ بازنگری شد. این پژوهش با حمایت مالی دانشگاه تربیت دبیر شهید رجایی طبق قرارداد شماره ۲۶۲۳۶ مورخ ۱۳۹۵/۷/۱۰ انجام گردیده است.

محمد کلانتری، دانشکده مهندسی کامپیوتر، دانشگاه تربیت دبیر شهید رجایی، تهران، (email: mkalantari@sru.ac.ir).

سکینه سهرابی، دانشکده مهندسی کامپیوتر، دانشگاه تربیت دبیر شهید رجایی، تهران، (email: s.sohrabi@sru.ac.ir).

حمیدرضا رشیدی کنعان، دانشکده مهندسی کامپیوتر، دانشگاه تربیت دبیر شهید رجایی، تهران، (email: h.rashidykanan@sru.ac.ir).

حسین کرمی، دانشکده مهندسی برق، دانشگاه صنعتی امیرکبیر، تهران، (email: h.karami@aut.ac.ir).

2. Population Based Algorithm

3. Single Solution Algorithm

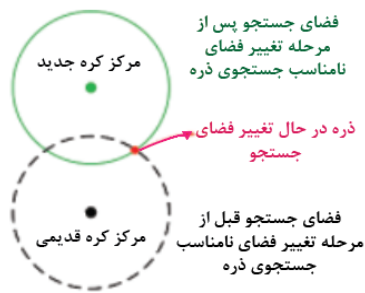
4. Chaos Theory

5. Harmony Search Algorithm

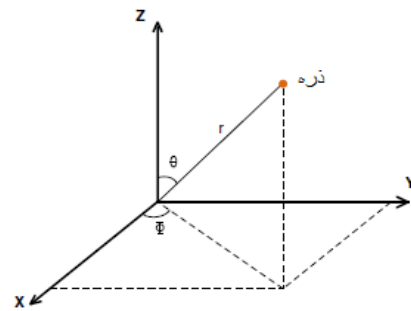
6. Particle Swarm Optimization Algorithm

7. Bee Colony

8. Genetic Algorithm



شکل ۲: تغییر فضای جستجوی نامناسب.



شکل ۱: زاویه‌های یک ذره در فضای سه‌بعدی.

## ۲-۲ جستجو

در این گام یک ذره با جستجو در فضای محدود شده به وسیله یک کره که مرکز آن یکی از ذرات انتخاب شده در گام قبل می‌باشد، به دنبال یک پاسخ بهتر می‌گردد. فرایند جستجو با تغییر پارامترهای ذره در مختصات کروی یعنی  $r$  و  $\theta$  انجام می‌شود که در این حالت، مرکز مختصات در مرکز کره تنظیم شده است. لازم به ذکر است که در فضای  $N$  بعدی،  $N-1$  زاویه در مختصات کروی وجود دارد که تغییر هر یک منجر به جابجایی ذره در فضای جستجو می‌شود. برای مثال همان طور که در شکل ۱ نشان داده شده است، در یک فضای ۳ بعدی، یک نقطه دو زاویه با محور مختصات می‌سازد. در این مرحله، تمام  $N-1$  زاویه‌ای که یک ذره با مرکز کره (مرکز مختصات) می‌سازد، با احتمالی که در آغاز الگوریتم در نظر گرفته شده است، به اندازه  $\alpha$  رادین 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 (۳) محاسبه می‌شود

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

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

## ۳-۲ تغییر فضای جستجوی نامناسب

هر ذره که به عنوان مرکز کره انتخاب می‌شود و ذراتی که به آن تعلق می‌گیرند 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 و در نتیجه فضای تحت جستجوی آنها باید تغییر یابد. 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.

لازم به ذکر است بعد از جستجو احتمال دارد 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.

## ۴-۲ تولید ذرات جدید

برای جستجوی بهتر فضا و منعطف‌تر نمودن الگوریتم، تعداد مشخصی از بدترین ذرات، یعنی ذراتی که دارای مقدار تابع هدف بیشتری می‌باشند،

## ۲- الگوریتم فراابتکاری جستجوی ابرکروی استاندارد

الگوریتم جستجوی ابرکروی استاندارد (HSS)<sup>۱</sup>، اولین بار توسط کرمی و همکاران در سال ۲۰۱۴ ارائه شده است [۱۳]. این الگوریتم مانند سایر الگوریتم‌های فراابتکاری با تولید ذرات به صورت تصادفی شروع به کار می‌کند که هر ذره بیانگر یک جواب کاندید برای مسأله پیش رو می‌باشد. هر ذره دارای دو قسمت متغیرها و مقدار تابع هدف می‌باشد. متغیرهای یک ذره نشان‌دهنده 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 اولیه تولیدشده، بهترین ذرات به عنوان مراکز کره انتخاب می‌شوند و بقیه ذرات بر اساس معیاری که در گام اول الگوریتم توضیح داده خواهد شد، به کره‌ها تخصیص می‌یابند. الگوریتم مذکور از پنج گام به قرار زیر تشکیل می‌شود.

### ۱-۲ مقداردهی اولیه و تولید ذرات

در این گام پس از آن که پارامترهای اولیه الگوریتم مانند تعداد مراکز کره، تعداد ذرات جدیدی که در هر تکرار اضافه می‌شود و تعداد جمعیت اولیه 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 که با استفاده از (۱) به دست می‌آید در میان کره‌ها توزیع می‌شوند. هر کره و مرکز آن، نماینده یک ناحیه است. در واقع فضای جستجو از این طریق به نواحی مختلف تقسیم می‌شود

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

در (۱) مقدار  $OFD^2$  برای هر مرکز کره برابر است با اختلاف تابع هدف آن مرکز با بیشینه مقدار تابع هدف از بین همه مراکز کره‌ها که با استفاده از (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 کره مورد بررسی و  $SC_s$  بیانگر مجموعه مراکز کره‌ها می‌باشد.

1. Hyper Spherical Search
2. Objective Function Difference
3. Spherical Center Set

جدول ۱: نگاهش‌های آشوب.

نام	بازه	فرمول
نگاشت چبیشف (Chebyshev Map)	(۰, ۱)	$x_{k+1} = \cos(k \cdot \cos^{-1} x_k)$
نگاشت تکراری (Iterative Map)	(۰, ۱)	$x_{k+1} = \sin \frac{\alpha \pi}{x_k}, \alpha \in (-, ۱)$
نگاشت لجستیک (Logistic Map)	(۰, ۱)	$x_{k+1} = \alpha x_k (1 - x_k), x \in (-, ۱), \alpha = ۴$
نگاشت دایره (Circle Map)	(۰, ۱)	$x_{k+1} = x_k + b - \frac{\alpha}{\pi} \cdot \sin(\pi x_k) \bmod e(۱)$
نگاشت لیوویچ (Liebovitch Map)	(۰, ۱)	$x_{k+1} = \begin{cases} \alpha x_k & , 0 < x_k < p_1 \\ \frac{p_2 - x_k}{p_2 - p_1} & , p_1 < x_k < p_2 \\ 1 - \beta(1 - x_k) & , p_2 < x_k < 1 \end{cases}$

حذف می‌شوند و سپس 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.

## ۲-۵ 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. در این مقاله این مقدار را برابر  $10^{-7}$  در نظر گرفته‌ایم.

## ۳- نظریه آشوب

یکی از مفاهیم جدید و بنیادی علم نوین، مفهوم آشوب است که افق درک ما نسبت به هستی را بسیار گسترش می‌دهد. امروزه نظریه آشوب را نوعی نظم در بی‌نظمی تعریف می‌کنند. در واقع منظور از این تئوری، ناکارآمدی و یا به هم ریختگی نمی‌باشد. این نظریه را هلیز این گونه توصیف کرده است: "بی‌نظمی و آشوب، نوعی بی‌نظمی منظم یا نظم در بی‌نظمی است، بی‌نظم از آن رو که نتایج آن غیر قابل پیش‌بینی می‌باشد و نظم بدان جهت که از نوعی قطعیت برخوردار است" [۱۴].

آشوب در مفهوم دقیق و ریاضی آن پدیده‌ای به ظاهر تصادفی و پیچیده است که در باطن، دارای طبیعتی قطعی<sup>۱</sup> می‌باشد و به شکلی است که از یک معادله دیفرانسیل ساده می‌توان رفتارهای بسیار پیچیده‌ای را انتظار داشت [۱۵]. در اکثر الگوریتم‌های فراابتکاری با اجزای تصادفی، تابع توزیع احتمال استفاده شده برای تولید اعداد تصادفی، تابع توزیع احتمال گاوسی یا یکنواخت<sup>۲</sup> می‌باشد. با جایگزینی مقادیر حاصل از نگاهش آشوب به جای این مقادیر تصادفی نتایج بهتری حاصل شده و علی‌رغم تصادفی بودن اعداد می‌توان به قطعی بودن نتایج حاصل اطمینان داشت [۱۶]. چهار ویژگی مشترک در سیستم‌های آشوبناک مطرح شده‌اند که در ادامه به اختصار آورده شده است.

۱) جاذبه‌های عجیب: جاذبه‌های عجیب، الگوهایی هستند که از منظر یا منظرهای گوناگون بی‌نظم و آشفته، ولی از منظر یا منظرهای دیگر دارای نظم هستند. هرچه افق دید گسترده‌تر باشد، یافتن

جاذبه عجیب ممکن‌تر و قدرت پیش‌بینی بیشتر می‌باشد.

۲) اثر پروانه‌ای: سیستم‌های آشوبناک، سیستم‌های دینامیکی غیر خطی می‌باشند که نسبت به شرایط اولیه‌شان بسیار حساس هستند و تغییری اندک در شرایط اولیه چنین سیستم‌هایی، باعث تغییرات بسیار در عملکرد سیستم می‌شود. این پدیده به عنوان «اثر پروانه‌ای» مشهور است که توسط لورنز مطرح شده است [۱۷].

۳) سازگاری پویا: سیستم‌های بی‌نظم در ارتباط با محیطشان همانند موجودات زنده رفتار می‌کنند و نوعی تطابق و سازگاری پویا بین آنها و پیرامونشان برقرار است.

۴) خودشباهتی: در نظریه آشوب و معادلات آن، نوعی شباهت بین اجزا و کل، قابل تشخیص است. مثال معروف آن، یک صفحه هولوگرام است که توسط لیزر، تصویری بر آن ضبط شده باشد. این صفحه دارای خاصیتی است که اگر به قطعاتی تقسیم شود، هر قطعه آن تصویر کامل را نشان خواهد داد.

در جدول ۱ نگاهش‌های آشوب مورد استفاده در این مقاله آورده شده است. باید توجه داشت که نگاهش مناسب برای مسایل مختلف با توجه به فرکانس و دامنه موج آن نگاهش می‌تواند متفاوت باشد. در مسئله مد نظر دامنه و فرکانس نگاهش انتخابی باید به گونه‌ای باشد که تمام فضای جستجو مسئله را در تعداد تکرار کم پوشش دهد [۱۲] زیرا ممکن است الگوریتم با تعداد ذرات کمی شروع به کار کند و این مسئله به جستجوی فضای مسئله و همگراشدن به جواب بهینه کمک کند.

## ۴- الگوریتم پیشنهادی بهینه‌سازی جستجوی ابرکروی مبتنی بر نظریه آشوب

در این بخش الگوریتم پیشنهادی ارائه می‌شود. همان طور که در بخش ۲ توضیح داده شد در الگوریتم جستجوی ابرکروی در دو مرحله تولید ذره (بخش ۲-۱ و ۲-۴) و جستجو (بخش ۲-۲) از اعداد تصادفی با تابع توزیع احتمال یکنواخت استفاده می‌شود. در الگوریتم پیشنهادی ابتدا از نگاهش‌های آورده شده در جدول ۱ برای هر مرحله به صورت جداگانه استفاده شده است. سپس با استفاده از نتایج حاصل از هر بخش، نگاهش‌هایی که باعث کاهش انحراف معیار و بهبود نتایج حاصل از اجرای الگوریتم در هر مرحله می‌شوند، انتخاب و به صورت هم‌زمان مورد استفاده قرار می‌گیرند تا جفت نگاهش مناسب انتخاب شوند. برای این منظور ابتدا گام جستجو مورد آزمایش قرار گرفته است. همان طور که در بخش ۲-۲ توضیح داده شد در قسمت جستجو یک ذره با جستجو در فضای محدود شده به وسیله یک کره که مرکز آن مختصات یکی از بهترین ذرات می‌باشد به دنبال یک پاسخ بهتر می‌گردد. فرایند جستجو با تغییر پارامترهای ذره در مختصات کروی یعنی  $r$  و  $\theta$  انجام می‌شود. در الگوریتم پیشنهادی برای تنظیم میزان تغییر شعاع و زاویه از اعداد تولید شده توسط نگاهش‌های آشوب به منظور بهبود الگوریتم جستجوی ابرکروی استفاده شده است. سپس قسمت تولید ذرات مورد بررسی قرار می‌گیرد که مستلزم بررسی گام اول و چهارم الگوریتم است. در گام اول برای تولید جمعیت اولیه و در گام چهارم برای تولید ذرات جدید از اعداد تصادفی در بازه تعیین شده برای هر متغیر استفاده می‌شود. در الگوریتم پیشنهادی مقادیر حاصل از نگاهش‌های آورده شده در جدول ۱ جایگزین

4. Butterfly Effect  
5. Dynamic Adaptation  
6. Self Similarity

1. Deterministic  
2. Uniform  
3. Strange Attraction

استفاده در جدول ۳ آمده است. با توجه به جدول ۲ تعداد تکرار اجرای الگوریتم برای رسیدن به جواب مطلوب در الگوریتم پیشنهادی برابر هفتاد و در دیگر الگوریتم‌های مورد بررسی برابر صد در نظر گرفته شده است. همچنین برای معتبر بودن نتایج مقایسه، فضای جستجو برای هر یک از توابع در همه الگوریتم‌ها یکسان در نظر گرفته شده است. این ۳ مرحله آزمایش به قرار زیر است:

### ۱-۵ انتخاب نگاشت مناسب برای قسمت جستجو

در آزمایش اول به دنبال یافتن نگاشت‌های مناسب جهت استفاده در قسمت جستجو هستیم. بررسی نتایج حاصل از آزمایش اول که در جدول ۴ آمده است بیانگر این نتیجه است که نگاشت‌های مناسب برای قسمت جستجو، نگاشت‌های لیوویچ و چیشف می‌باشند که نتایج بهتری را نسبت به دیگر نگاشت‌ها بر روی توابع محک دارا می‌باشند. برای مثال مقدار کمینه حاصل از نگاشت تکراری در تابع FA با مقدار کمینه مطلوب اختلاف قابل توجهی داشته و نگاشت تکراری نمی‌تواند گزینه مناسبی برای این تابع باشد. اما نگاشت لیوویچ جواب مناسب‌تری را برای این تابع دارا می‌باشد. همچنین در تابع F13 نگاشت چیشف نسبت به دیگر نگاشت‌ها موفق‌تر عمل کرده است.

### ۲-۵ انتخاب نگاشت مناسب برای قسمت تولید ذرات

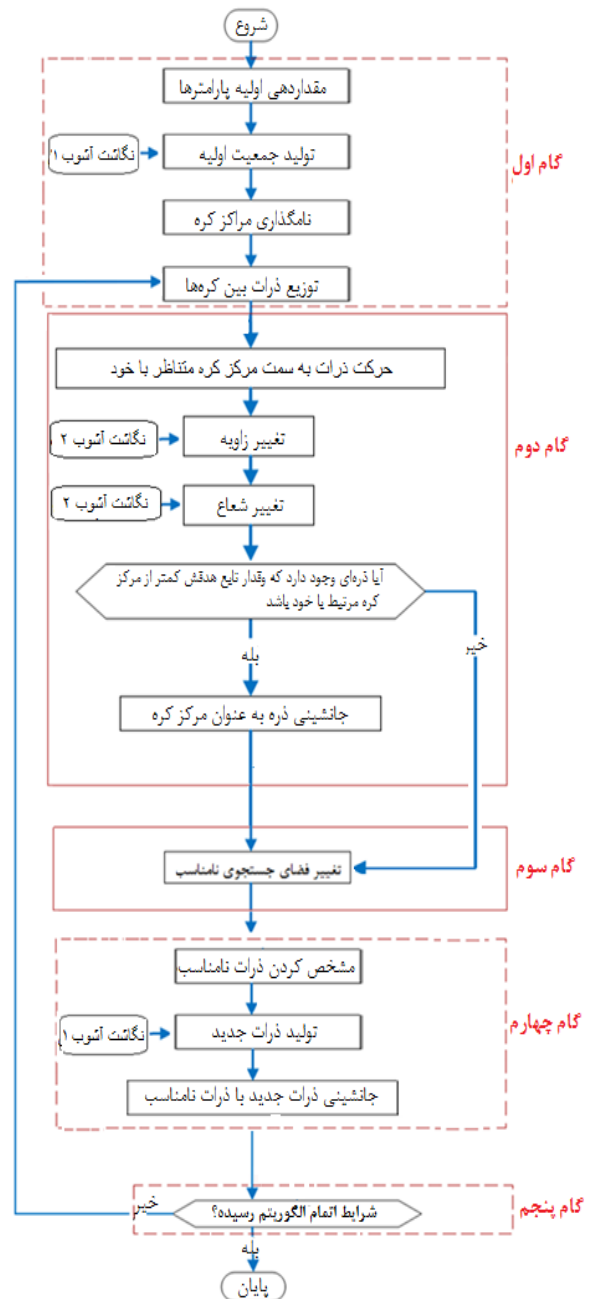
در آزمایش دوم به دنبال یافتن نگاشت‌های مناسب جهت استفاده در قسمت تولید ذرات هستیم. نتایج حاصل از آزمایش دوم در جدول ۵ آمده است. با توجه به مقادیر آورده شده در این جدول می‌توان نتیجه گرفت که نگاشت‌های مناسب برای قسمت تولید ذرات، نگاشت‌های لجیستیک و چیشف می‌باشند. برای مثال مقدار کمینه حاصل از نگاشت دایره در تابع FA با مقدار کمینه مطلوب اختلاف قابل توجهی داشته و این نگاشت نمی‌تواند گزینه مناسبی برای این تابع باشد. اما نگاشت چیشف جواب مناسب‌تری برای این تابع دارد.

### ۳-۵ انتخاب جفت نگاشت سازگار

در این مقاله به دنبال مناسب‌ترین نگاشت‌ها برای قسمت تولید ذرات و قسمت جستجو هستیم تا در صورت استفاده هم‌زمان بتوانند انحراف معیار و مقدار کمینه حاصل از الگوریتم را نسبت به دیگر الگوریتم‌های مورد بررسی بهبود دهند. باید توجه داشت که برای هر یک از توابع ممکن است چندین نگاشت نتایج مشابهی را داشته باشند اما در این مقاله به دنبال یک جفت نگاشت سازگار هستیم تا بتوانیم به نتیجه مطلوب بر روی تمامی و یا بیشتر توابع محک مورد بررسی برسیم.

پس از انتخاب نگاشت‌های مناسب در آزمایش اول و دوم، آزمایش سوم انجام می‌شود و هر یک از نگاشت‌های انتخاب شده در آزمایش اول به صورت هم‌زمان با هر یک از نگاشت‌های انتخاب شده در آزمایش دوم، مورد استفاده قرار می‌گیرد تا مناسب‌ترین جفت نگاشت انتخاب شوند و این آزمایش بیست مرتبه تکرار می‌شود.

در این مرحله علاوه بر اجرای الگوریتم پیشنهادی با استفاده از جفت نگاشت‌های انتخاب شده از آزمایش اول و دوم، آزمایش بر روی دیگر جفت نگاشت‌ها از بین نگاشت‌های آورده شده در جدول ۱ تکرار شده که نتایج برخی از جفت نگاشت‌ها که موفق‌تر بوده‌اند، در جدول ۶ آورده شده است. باید توجه داشت که برای هر یک از توابع ممکن است چندین نگاشت، نتایج مشابهی را داشته باشند اما در این مقاله به دنبال یک جفت نگاشت سازگار هستیم تا در صورت استفاده هم‌زمان بتوانند بر روی تمامی



شکل ۳: الگوریتم جستجوی ابرکروی پیشنهادی مبتنی بر نظریه آشوب.

اعداد تصادفی مورد استفاده در الگوریتم HSS می‌شوند. پس از بررسی نتایج حاصل از هر مرحله، نگاشت‌های مناسب برای هر قسمت تعیین و مجدداً به صورت هم‌زمان آزمایش می‌شوند. در این مقاله بر اساس نتایج به دست آمده به دنبال انتخاب مناسب‌ترین نگاشت برای هر بخش هستیم که منجر به کاهش انحراف معیار و بهبود نتایج حاصل از اجرای الگوریتم در تمامی توابع محک مورد بررسی شود. روند اجرای الگوریتم جستجوی ابرکروی پیشنهادی مبتنی بر نظریه آشوب در شکل ۳ آورده شده است.

### ۵- نتایج آزمایشات

در این بخش کارایی الگوریتم پیشنهادی را بر روی توابع محک استاندارد مورد بررسی قرار می‌دهیم. بدین منظور در ۳ مرحله آزمایش، نگاشت‌های مناسب برای هر قسمت را بررسی می‌نماییم. شرایط در نظر گرفته شده برای انجام هر آزمایش در جدول ۲ و توابع هدف [۱۸] مورد

جدول ۲: شرایط در نظر گرفته شده برای انجام آزمایش‌های انجام شده.

شماره آزمایش	اول	دوم	سوم
توضیح	انتخاب نگاشت مناسب برای قسمت جستجو	انتخاب نگاشت مناسب برای قسمت تولید ذرات	انتخاب جفت نگاشت سازگار
تعداد تکرار آزمایش	۲۰	۲۰	۲۰
حداکثر تعداد اجرای الگوریتم	۲	۲	۲
تعداد تکرار در هر بار اجرا	۱۰۰	۱۰۰	۱۰۰
تعداد جمعیت اولیه ذرات	۱۰۰۰	۱۰۰۰	۱۰۰۰
تعداد مرکز کره	۵۰	۵۰	۵۰

جدول ۳: توابع هدف.

شناسه	نام تابع	فرمول	دامنه
F۱	EASOM	$f(x_1, x_2, \dots, x_n) = x_1 (\sin(x_1)x_2) + x_2 (\cos(x_2 - x_1))x_3$	$-5 < x_i < 5$
F۲	GHAEMI	$f(x_1, x_2) = x_1 \sin(4x_1) + 1.1x_2 \sin(2x_2)$	$0 < x_1, x_2 < 10$
F۳	DROP-WAVE	$f(x_1, x_2) = -\frac{1 + \cos(1.2\sqrt{x_1^2 + x_2^2})}{0.5(x_1^2 + x_2^2) + 2}$	$-6 < x_i < 6$ $i = 1, 2, \dots, 6$
F۴	SCHAFFER	$f(x_1, x_2) = \frac{1 + \cos(\sin^2(x_1^2 + x_2^2)) - 0.5}{1 + 0.1(x_1^2 + x_2^2)}$	$-10 < x_1, x_2 < 10$
F۵	EASOM	$f(x_1, x_2) = -(\cos x_1 \cos x_2) \cdot \exp(-(x_1 - \Pi)^2 - (x_2 - \Pi)^2)$	$-5 < x_i < 5$
F۶	EGGHOLDER	$f(x_1, x_2) = -(x_2 + 47) \cdot \sin\sqrt{ x_1 + \frac{x_2}{4} + 47 } - x_1 \sin\sqrt{ x_1 + (x_2 + 47) }$	$-512 < x_1, x_2 < 512$
F۷	GRAMACY & LEE	$f(x) = \frac{\sin(10\pi x)}{2x} + (x-1)^2$	$0.5 < x_1, x_2 < 2.5$
F۸	TRID	$f(x_1, x_2, x_3, x_4, x_5) = \sum_{i=1}^5 (x_i - 1)^2 - \sum_{i=1}^5 x_i x_{i-1}$	$-36 < x_i < 36$ $i = 1, 2, \dots, 6$
F۹	SHUBERT	$f(x_1, x_2) = (\sum_{i=1}^5 i \cos((i+1)x_1 + i)) \cdot (\sum_{i=1}^5 i \cos((i+1)x_2 + i))$	$-512 < x_1, x_2 < 512$
F۱۰	SIX-HUMP CAMEL	$f(x_1, x_2) = (4 - 2.1x_1^2 + \frac{x_1^4}{4})x_2^2 + x_1x_2 + (-4 + 4x_2^2)x_1^2$	$-3 < x_1 < 3$ $-2 < x_2 < 2$
F۱۱	BEALE	$f(x_1, x_2) = (1.5 - x_1 + x_1x_2)^2 + (2.25 - x_1 + x_1x_2^2)^2 + (2.625 - x_1 + x_1x_2^3)^2$	$-5 < x_1, x_2 < 5$
F۱۲	BRANIN	$f(x_1, x_2) = \frac{1}{5.1\pi} [(\frac{x_2}{4.71} - \frac{5.1x_1}{\pi} + 5)^2 + (10 - \frac{10x_2}{\pi}) \cos x_1 - 4.1\pi]$ where $\bar{x}_1 = 15x_1 - 5, \bar{x}_2 = 15x_2$	$0 < x_1, x_2 < 1$
F۱۳	MISHEA'S BIRD	$f(x_1, x_2) = \sin x_1 \cdot \exp(1 - \cos x_1) + \cos x_1 \cdot \exp(1 - \sin x_2) + (x_1 - x_2)^2$ subjected to: $(x_1 + 5)^2 + (x_2 + 5)^2 < 25$	$-10 < x_1 < 0$ $-6.5 < x_2 < 0$
F۱۴	CROSS-IN-TRAY	$f(x_1, x_2) = -0.7081 \left( \left  \sin x_1 \cdot \sin x_2 \cdot \exp\left  10 - \frac{\sqrt{ x_1^2 + x_2^2} }{\pi} \right  + 1 \right  \right)^{1.1}$	$-10 < x_1, x_2 < 10$

جدول ۴: نتایج حاصل از توابع محک استاندارد با استفاده از نگاشت‌های مختلف در قسمت جستجو.

تابع	معیار	نگاشت				
		دایره	تکراری	لیبویچ	لجیستیک	چیشف
F۱	Min	-۰.۹۹۹۸	-۰.۹۹۹۳	-۰.۹۹۸۶	-۰.۹۹۹۸	-۰.۹۹۹۹
	(SD)	(۳.۵۳E-۰۴)	(۵.۴۵E-۰۴)	(۳.۵۵E-۰۴)	(۸.۱۶E-۰۴)	(۱.۶۷E-۰۴)
F۲	Min	-۱۸.۵۴۳	-۱۸.۵۴۵	-۱۸.۵۵۰	۱۸.۵۵۲	-۱۸.۵۵۴
	(SD)	(۰.۰۰۶۷)	(۰.۰۲۳۷)	(۱.۵۷E-۰۳)	(۱.۵۵E-۰۳)	(۰.۰۰۴)
F۳	Min	-۰.۹۶۳۶	-۰.۹۹۸۰	-۰.۹۹۹۹	-۰.۹۷۳۲	-۰.۹۹۹۹
	(SD)	(۶.۶۱E-۴)	(۰.۰۰۱)	(۰.۰۰۱)	(۷.۸E-۴)	(۰.۰۰۲)
F۴	Min	۰.۲۹۵۱۴	۰.۲۹۵۱۴	۰.۲۹۵۱۴	۰.۲۹۵۱۴	۰.۲۹۵۱۴
	(SD)	(۸.۱۲E-۰۵)	(۵.۵۷E-۰۵)	(۳.۷۸E-۰۵)	(۱.۱۹E-۴)	(۲.۷۹E-۰۵)
F۵	Min	-۰.۹۹۹۸	-۰.۹۹۹۴	-۰.۹۹۹۹	-۰.۹۹۹۶	-۰.۹۹۹۳
	(SD)	(۰.۰۰۱۱)	(۸.۶۵E-۰۴)	(۱.۴۳۷E-۰۴)	(۹.۶۶۲E-۴)	(۷.۳۱E-۰۴)

F۶	Min (SD)	-۹۵۹,۶۴۱ (۳,۰۶E-۰۵)	-۹۵۹,۶۴۱ (۲,۶۸E-۰۴)	-۹۵۹,۶۴۱ (۱,۹۱E-۰۵)	-۹۵۹,۶۴۱ (۱,۴۸E-۰۴)	-۹۵۹,۶۴۱ (۲,۴۲E-۰۴)
F۷	Min (SD)	-۲,۸۷۳۹ (۱,۹۳E-۰۴)	-۲,۸۷۳۹ (۹,۲۱E-۰۵)	-۲,۸۷۳۹ (۹,۹۰۸E-۰۵)	-۲,۸۷۳۹ (۱,۰۰۴E-۰۴)	-۲,۸۷۳۹ (۶,۷۴E-۰۵)
F۸	Min (SD)	-۳۴,۳۵۸۵ (۴,۱۱۹۰)	-۳۲,۰۵۵۲ (۱,۳۰۲۰)	-۴۲,۰۱۴۵ (۳,۳۴۳)	-۳۱,۴۹ (۲,۸۶۸)	-۴۱,۲۷۳۵ (۱,۶۱۵۶)
F۹	Min (SD)	-۱۸۶,۲۹ (۰,۳۶)	-۱۸۶,۵۱ (۰,۶۸)	-۱۸۶,۷۱ (۰,۲۳۸)	-۱۸۶,۷۰ (۱,۷۲۰)	-۱۸۶,۶۸ (۱,۰۲۲)
F۱۰	Min (SD)	-۱,۰۳۱۶ (۰,۰۰۲۵)	-۱,۰۳۰۸ (۰,۰۰۳۴)	-۱,۰۳۱۶ (۰,۰۰۱۷)	-۱,۰۳۱۱ (۰,۰۰۲۵)	-۱,۰۳۱۲ (۰,۰۰۱۴)
F۱۱	Min (SD)	۹,۱۱۰E-۴ ,۰۰۰۱۳	۷,۰۷۹۳E-۴ ,۰۰۴۳۷E-۴	۲,۴۰۸E-۴ (۹,۱۶۳E-۴)	۳,۹۱۵E-۴ (۵,۰۲۷۵E-۴)	۵,۹۸۴E-۴ (۰,۰۰۱۶)
F۱۲	Min (SD)	-۱,۰۴۷۴ (۲,۰۵۵E-۵)	-۱,۰۴۷۴ (۲,۳۱۷E-۵)	-۱,۰۴۷۴ (۹,۰۱۴E-۶)	-۱,۰۴۷۴ (۳,۸۱۷E-۵)	-۱,۰۴۷۴ (۴,۰۸۲E-۵)
F۱۳	Min (SD)	-۱,۰۶۷۶ (۰,۰۰۷۹)	-۱,۰۶۷۶ (۰,۰۰۶۱)	-۱,۰۶۷۶ (۰,۰۰۳۳۳)	-۱,۰۶۷۶ (۰,۰۰۶۱۹)	-۱,۰۶۷۷ (۰,۰۰۶۶)
F۱۴	Min (SD)	-۲,۰۶۲ (۶,۵۴۹E-۴)	-۲,۰۶۲ (۴,۰۰۶E-۴)	-۲,۰۶۲ (۱,۸۲۵E-۴)	-۲,۰۶۲ (۴,۶۹۲E-۴)	-۲,۰۶۲ (۸,۴۶۰E-۴)

جدول ۵: نتایج حاصل از توابع محک استاندارد با استفاده از نگاشت‌های مختلف در قسمت تولید ذرات.

تابع	معیار	نگاشت				
		دایره	تکراری	لیبویچ	لجیستیک	چیشف
F۱	Min (SD)	-۰,۹۸۵۷ (۴,۴۵E-۰۴)	-۰,۵۹۳۳ (۲,۱۴E-۰۴)	-۰,۹۹۵۶ (۸,۶۲E-۰۷)	-۰,۹۹۹۷ (۲,۱۹E-۰۴)	-۰,۹۹۹۶ (۵,۳۰E-۰۵)
F۲	Min (SD)	-۱۸,۴۶۴۱ (۰,۰۰۳۷)	-۱۸,۵۴۸۵ (۰,۰۰۲۳)	-۱۸,۳۰۵۷ (۲,۳۳E-۰۵)	-۱۸,۰۵۹۸ (۱,۸۰E-۰۴)	-۱۸,۵۵۳۵ (۰,۰۰۲۱۰)
F۳	Min (SD)	-۰,۹۳۶۲ (۲,۲۹E-۰۴)	-۰,۹۹۹ (۰,۰۰۲۰)	-۰,۹۹۰۹ (۰,۰۰۳۰)	-۰,۹۹۹۹ (۲,۴۳E-۰۷)	-۰,۹۹۰۹ (۶,۰۴E-۰۴)
F۴	Min (SD)	۰,۲۹۵۱۴۴ (۸,۵۷E-۰۶)	۰,۲۹۵۱۴۷ (۲,۷۲E-۰۴)	۰,۲۹۵۱۴۸ (۱,۱۰E-۰۴)	۰,۲۹۵۱۴۴ (۵,۲۶E-۰۵)	۰,۲۹۵۱۴۳ (۲,۵۲E-۰۴)
F۵	Min (SD)	-۰,۹۸۵۸ (۱,۴۲E-۰۴)	-۰,۹۶۳۶۲ (۰,۰۱۹۵)	-۰,۸۸۵۳۵ (۱,۴۱E-۰۶)	-۰,۹۹۹۷۸ (۳,۰۹E-۰۵)	-۰,۹۹۹۴۵ (۷,۶۰E-۰۴)
F۶	Min (SD)	-۹۵۹,۶۴۱ (۰,۰۰۱۱)	-۹۵۹,۶۴۰ (۱,۲۷E-۰۴)	-۹۵۹,۶۴۱ (۳,۶۸E-۰۴)	-۹۵۹,۶۴۱ (۱,۹۰E-۰۴)	-۹۵۹,۶۴۱ (۸,۳۰E-۰۵)
F۷	Min (SD)	-۲,۸۷۳۹ (۴,۲۶E-۰۵)	-۲,۸۷۳۹ (۵,۸۹E-۰۵)	-۲,۸۷۳۹ (۴,۸۷۷E-۰۴)	-۲,۸۷۳۹ (۵,۴۲۲E-۰۵)	-۲,۸۷۳۹ (۳,۰۲۶E-۰۵)
F۸	Min (SD)	۲۴۴,۷۲۰۴ (۴,۷۳۸۲)	-۱۸,۱۳۸۱ (۱,۷۰۵۱)	۱,۸۱۴۱ (۰,۵۲۵۷)	۳,۷۵۹۸ (۰,۰۰۶۰۹)	-۴۲,۶۹۱۳ (۱,۷۰۱)
F۹	Min (SD)	-۷۷,۵۲ (۰,۰۰۰۸)	-۱۸۶,۰۵ (۰,۴۶۶)	-۱۲,۰۷۲ (۶,۰۹۸)	-۱۸۱,۵۷ (۱,۰۰۲۵)	-۱۸۶,۳۴ (۰,۰۰۰۶)
F۱۰	Min (SD)	-۱,۰۳۱۱ (۰,۰۰۱۱)	-۱,۰۱۶۹ (۲,۵۲E-۰۴)	-۰,۸۶۳۹ (۰,۰۰۳۵)	-۱,۰۲۲۷ (۰,۰۰۸۲)	-۱,۰۳۱۴ (۰,۰۰۱۴)
F۱۱	Min (SD)	۰,۰۰۱۱ (۰,۰۰۰۴۱)	۰,۰۰۱۱ (۰,۰۰۰۱۸)	۰,۰۰۲۳ (۰,۰۰۰۴۷)	۰,۰۰۲۳ (۰,۰۰۰۴۹)	۷,۸۱۷E-۰۴ (۰,۰۰۰۸۰)
F۱۲	Min (SD)	-۱,۰۴۷۴ (۱,۱۳۴۹E-۰۴)	-۱,۰۴۷۴ (۱,۴۲۷E-۰۵)	-۱,۰۴۷۴ (۱,۴۲۷E-۰۵)	-۱,۰۴۷۴ (۱,۰۰۲۶E-۰۵)	-۱,۰۴۷۴ (۴,۸۸۷E-۰۶)
F۱۳	Min (SD)	-۱,۰۶۷۲ (۰,۰۱۷۷)	-۱,۰۶۷۶ (۰,۰۰۵۸)	-۱,۰۶۷۲ (۰,۰۱۷۷)	-۱,۰۶۷۶ (۰,۰۰۶۲۳)	-۱,۰۶۷۸ (۰,۱۶۷۹)
F۱۴	Min (SD)	-۲,۰۶۲ (۵,۷۲۶E-۰۴)	-۲,۰۶۲ (۴,۰۰۷۷E-۰۴)	-۲,۰۶۲ (۲,۳۵۸E-۰۴)	-۲,۰۶۲ (۳,۰۱۱E-۰۴)	-۲,۰۶۲ (۱,۶۶۵E-۰۴)



جدول ۶: نتایج برخی از جفت نگاشت‌های استفاده شده در الگوریتم پیشنهادی بر روی توابع محک استاندارد.

تابع	معیار	جفت نگاشت‌های مورد استفاده در قسمت تولید ذرات / جستجو							
		لیبویویچ	لیبویویچ	دایره	تکراری	لجستیک	لیبویویچ	چیبیشف	
		لیبویویچ	چیبیشف	تکراری	لیبویویچ	تکراری	لیبویویچ	چیبیشف	
F1	Min (SD)	-۰.۹۹۹۹ (۰.۰۰۳)	-۰.۹۹۹۹ (۹.۵E-۰۴)	-۰.۹۸۵۷ (۲.۶۶E-۰۴)	-۰.۶۶۹۸ (۰.۰۰۲۶)	-۰.۹۹۹۷ (۴.۳۴E-۰۴)	-۰.۹۹۷۲ (۱.۰۱E-۰۶)	-۰.۹۹۹۷ (۰.۰۰۱)	-۰.۹۹۹۶ (۲.۵۸E-۰۴)
F2	Min (SD)	-۱۸.۵۴ (۰.۰۰۲۶)	-۱۸.۵۵ (۰.۰۱۶۹)	-۱۸.۲۸ (۱.۳۱۶E-۰۴)	-۱۶.۰۸ (۲.۶۹۹E-۰۴)	-۱۸.۰۵ (۳.۷۲۸E-۰۴)	-۱۸.۵۵ (۰.۰۰۳۰)	-۱۸.۵۳ (۰.۰۰۲۳۶)	-۱۸.۵۵ (۰.۰۰۱۵)
F3	Min (SD)	-۰.۹۹۰ (۰.۰۰۱۴)	-۰.۹۸۵ (۲.۲۲۵E-۰۴)	-۰.۹۳۶ (۲.۶۲E-۰۶)	-۰.۹۹۰ (۰.۰۰۱۱)	-۱ (۴.۹۲E-۰۸)	-۰.۹۹۹ (۰.۰۱۰)	-۰.۹۵۰ (۲.۴۵۷E-۰۴)	-۰.۹۹۹ (۴.۸۳۸E-۰۵)
F4	Min (SD)	۰.۲۹۵۱ (۱.۸۷E-۰۵)	۰.۲۹۵۱ (۱.۴۷۵E-۰۴)	۰.۲۹۵۱ (۷.۷۵۶E-۰۵)	۰.۲۹۵۱ (۱.۲۵۰E-۰۴)	۰.۲۹۵۱ (۱.۴۹۱E-۰۴)	۰.۲۹۵۱ (۱.۷۱۸E-۰۴)	۰.۲۹۵۱ (۳.۴۸۶E-۰۴)	۰.۲۹۵۱ (۱.۴۷E-۰۷)
F5	Min (SD)	-۰.۹۹۹۵۸ (۰.۰۰۲۳)	-۰.۹۹۹۹۱ (۴.۳۱۵E-۰۴)	-۰.۹۹۹۵۸ (۰.۰۰۲۵)	-۰.۹۹۹۴۳ (۷.۸۹۴E-۰۴)	-۰.۹۹۹۵۸ (۰.۰۰۱۰)	-۰.۹۹۸۷ (۱.۴۹E-۰۵)	-۰.۹۹۹۸۷ (۱.۸۵۹E-۰۴)	-۰.۹۹۹۹۵ (۱.۴۶E-۰۵)
F6	Min (SD)	-۹۵۹.۶۴۰ (۴.۳۲۷E-۰۴)	-۹۵۹.۶۴۰ (۱.۳۱۴E-۰۴)	-۹۵۹.۶۴۰ (۱.۶۴۳E-۰۵)	-۹۵۹.۶۴۰ (۹.۷۳۸E-۰۵)	-۹۵۹.۶۴۰ (۹.۰۵E-۰۵)	-۹۵۹.۶۴۰ (۶.۹۷۸E-۰۵)	-۹۵۹.۶۴۰ (۱.۳۶۴E-۰۵)	-۹۵۹.۶۴۰ (۸.۳۷E-۰۸)
F7	Min (SD)	-۲.۸۷۳۸ (۲.۶۳۰E-۰۴)	-۲.۸۷۳۸ (۳.۵۲۸E-۰۴)	-۲.۸۷۳۸ (۹.۱۲E-۰۵)	-۲.۸۷۳۸ (۲.۱۱E-۰۴)	-۲.۸۷۳۸ (۱.۲۳۸E-۰۵)	-۲.۸۷۳۸ (۲.۴۲E-۰۴)	-۲.۸۷۳۸ (۳.۰۸۵E-۰۴)	-۲.۸۷۳۸ (۲.۴۰E-۰۶)
F8	Min (SD)	-۱۱.۴۰۵ (۰.۲۲۸۷)	-۲۱.۹۵ (۰.۳۲۰۵)	۹۹.۹۱ (۲.۷۲۳۰)	-۲۹.۹۹ (۰.۱۰۵۲)	۳.۷۵۹ (۰.۰۰۷۵)	-۳۵.۹۰ (۰.۳۰۵)	-۳۴.۷۱ (۲.۷۰۷۹)	-۴۲.۶۹ (۲.۲۰۴۵)
F9	Min (SD)	-۱۸۶.۶۹ (۰.۳۰۶۰)	-۱۸۶.۷۱ (۰.۰۶۷۸)	-۷۷.۴۷۵ (۰.۰۷۶۵)	-۱۸۶.۰۵۶ (۴.۹۷۴۴)	-۱۸۱.۵۸۳ (۰.۴۹۰۱)	-۱۸۶.۶۸ (۰.۴۶۹)	-۱۸۶.۶۹ (۰.۱۰۲۳)	-۱۸۶.۷۲ (۱.۰۵۲)
F10	Min (SD)	-۱.۰۳۰ (۰.۰۰۳۴)	-۱.۰۳۱ (۰.۰۰۲۵)	-۱.۰۳۱ (۰.۰۰۳۴)	-۱.۰۰۹ (۸.۵۰۱E-۰۴)	-۰.۹۳۸۰ (۰.۰۰۲۸۶)	-۱.۰۳۱ (۰.۰۰۲۲)	-۱.۰۳۱ (۰.۰۰۴۵)	-۱.۰۳۱ (۰.۰۰۲۱)
F11	Min (SD)	۰.۰۰۲۳ (۰.۰۰۴۸)	۴.۸۰۹E-۰۴ (۰.۰۰۴۳)	۰.۰۰۲۳ (۰.۰۰۵۲)	۶.۱۷۲E-۰۴ (۰.۰۰۲۲)	۰.۰۰۲۳ (۰.۰۰۵۰)	۰.۰۰۲۷ (۰.۰۰۶۷)	۴.۵۴۵E-۰۴ (۰.۰۰۱۵)	۴.۱۲E-۰۴ (۰.۰۰۲۹)
F12	Min (SD)	-۱.۰۴۷۴ (۹.۷۴۷E-۰۵)	-۱.۰۴۷۴ (۵.۱۴۶E-۰۵)	-۱.۰۴۷۴ (۷.۷۰۷E-۰۵)	-۱.۰۴۷۴ (۹.۷۴۷E-۰۵)	-۱.۰۴۷۴ (۹.۵۶۸E-۰۶)	-۱.۰۴۷۴ (۱.۰۹E-۰۴)	-۱.۰۴۷۴ (۶.۷۰۴E-۰۶)	-۱.۰۴۷۴ (۴.۸۸E-۰۶)
F13	Min (SD)	-۱.۰۶۷۶ (۰.۰۶۱۹)	-۱.۰۶۷۷ (۰.۰۱۰۳)	-۱.۰۶۷۸ (۰.۰۶۰۵)	-۱.۰۶۷۸ (۰.۰۶۰۵)	-۱.۰۶۷۸ (۰.۰۶۰۵)	-۱.۰۶۷۳ (۰.۱۱۲۵)	-۱.۰۶۶۴ (۰.۰۵۵۳)	-۱.۰۶۷۸ (۰.۰۲۶۵)
F14	Min (SD)	-۲.۰۶۲ (۴.۴۳۲E-۰۴)	-۲.۰۶۲ (۵.۵۰۴E-۰۵)	-۲.۰۶۲ (۱.۳۸۳E-۰۴)	-۲.۰۶۲ (۲.۹۱۱E-۰۴)	-۲.۰۶۲ (۲.۳۳۲E-۰۵)	-۲.۰۶۲ (۶.۰۱۶E-۰۴)	-۲.۰۶۲ (۲.۲۳۴E-۰۴)	-۲.۰۶۲ (۱.۲۸E-۰۵)

۱، نگاشت چیبیشف و نگاشت آشوب ۲، نگاشت لیبویویچ است. استفاده هم‌زمان از این دو نگاشت منجر به بهبود انحراف معیار و مقدار کمینه حاصل از اجرای الگوریتم پیشنهادی در مقایسه با الگوریتم‌های ژنتیک، ازدحام ذرات، جستجوی هارمونی و جستجوی ابرکروی استاندارد در تعداد تکرار کمتر می‌شود. همچنین الگوریتم پیشنهادی از سرعت همگرایی مناسبی برخوردار است زیرا همان‌طور که پیشتر آورده شده است، تعداد تکرار اجرای الگوریتم برای رسیدن به جواب مطلوب در سایر الگوریتم‌های مورد بررسی برابر صد اما در الگوریتم پیشنهادی برابر هفتاد در نظر گرفته شده است.

در الگوریتم ژنتیک، احتمال جهش را برابر ۰.۸، 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، نرخ تطبیق گام و فاصله پهنای باند را به ترتیب برابر ۱۰، ۰.۶، ۰.۳ و ۰.۱ در نظر گرفته‌ایم.

و یا بیشتر توابع محک مورد بررسی به نتیجه مطلوب برسند. پس از انتخاب جفت نگاشت سازگار، انحراف معیار و مقدار کمینه حاصل از الگوریتم پیشنهادی با نتایج حاصل از الگوریتم‌های ژنتیک، ازدحام ذرات، جستجوی هارمونی و جستجوی ابرکروی استاندارد مقایسه می‌شود که نتایج حاصل در جدول ۷ آورده شده است.

زمان اجرای الگوریتم‌های مورد بررسی برای هر یک از توابع محک استاندارد در جدول ۸ آمده و مقدار کمینه مدت زمان اجرا که به صورت پرنرنگ نشان داده شده است، متعلق به الگوریتم جستجوی هارمونی می‌باشد. با توجه به نتایج آورده شده در جدول ۸ می‌توان مشاهده کرد که مدت زمان اجرای الگوریتم پیشنهادی برای برخی از توابع محک نسبت به سایر الگوریتم‌های مورد بررسی بیشتر است و این در حالی است که با توجه به نتایج آورده شده در جدول ۷ می‌توان مشاهده کرد که در تمامی توابع مورد بررسی الگوریتم پیشنهادی از سایر الگوریتم‌های مورد بررسی موفق‌تر عمل کرده است. با توجه به نتایج آورده شده در جدول ۶ و ۷ که کمینه نتایج به صورت پرنرنگ نمایش داده شده است می‌توان نتیجه گرفت که مناسب‌ترین و سازگارترین جفت نگاشت، نگاشت چیبیشف برای قسمت تولید ذرات (گام اول و چهارم) و نگاشت لیبویویچ برای قسمت جستجو می‌باشد. پس می‌توان نتیجه گرفت که در شکل ۳، نگاشت آشوب

1. Harmony Memory Consideration Rate
2. Pitch Adjusting Rate
3. Distance Bound Wide

جدول ۷: نتایج حاصل از توابع محک استاندارد با استفاده از الگوریتم‌های مورد بررسی.

تابع	معیار	الگوریتم				
		CHSS	HSS	HSA	PSO	GA
F1	Min (SD)	-۰.۹۹۹۶۱ (۱.۴۶E-۰۶)	-۰.۹۹۹۵ (۲.۸۹E-۰۴)	-۰.۹۹۵۵ (۰.۲۰۰۰)	-۰.۹۹۸۵ (۰.۳۳۶)	-۱ (۴.۴۲E-۰۶)
F2	Min (SD)	-۱۸.۵۵ (۰.۰۰۱۵)	-۱۸.۵۵ (۰.۰۲۷۴)	-۱۸.۵۴ (۱.۴۳۶۴)	-۱۸.۵۴ (۰.۶۵۰۵)	-۱۸.۵۱ (۰.۶۵۵۳)
F3	Min (SD)	-۰.۹۹۹ (۴.۸۳۸E-۰۵)	-۰.۹۷۲۶ (۰.۰۰۱)	-۰.۹۵۵۰ (۰.۱۲۰۰)	-۱ (۰.۲۶۹۵)	-۱ (۰.۰۰۱۷)
F4	Min (SD)	۰.۲۹۵۱ (۱.۴۷۸E-۰۷)	۰.۲۹۵۱ (۱.۰۴E-۰۴)	۰.۲۹۵۵ (۰.۰۲۵۶)	۰.۲۹۵۱ (۰.۰۰۲۴)	۰.۲۹۵۱ (۹.۴۳۶E-۰۶)
F5	Min (SD)	-۰.۹۹۹۹ (۱.۴۶E-۰۵)	-۰.۹۹۹۵ (۲.۸۹E-۰۴)	-۰.۹۹۵۵ (۰.۲۰۰۰)	-۰.۹۹۸۵ (۰.۳۳۶)	-۱ (۴.۴۲E-۰۶)
F6	Min (SD)	-۹۵۹.۶۴۰ (۸.۳۷۲E-۰۸)	-۹۵۹.۶۴۰ (۱.۵۵E-۰۴)	-۹۳۰.۴۴۷ (۱۰۴.۳۵)	-۹۴۶.۵۱۷ (۳۴۰.۸۷۲)	-۸۸۵.۷۵۷ (۷۴.۶۱۲۳)
F7	Min (SD)	-۲.۸۷۳ (۲.۴۰۱E-۰۶)	-۲.۸۷۳ (۳.۲۰E-۰۵)	-۰.۸۶۹ (۰.۰۹۹۶)	-۰.۸۶۹ (۷.۰۹E-۰۶)	-۲.۸۷۳ (۰.۱۱۲۴)
F8	Min (SD)	-۴۲.۶۹ (۲.۲۰۴۵)	-۳۶.۵۰۲ (۶.۰۲۰)	-۲۳.۶۱۶۱ (۱۴۵.۱۴)	-۵ (۳.۴۲۸۱)	-۴۰.۱۶ (۴.۹۴۳)
F9	Min (SD)	-۱۸۶.۷۲ (۱.۰۵۲)	-۱۸۶.۶۵ (۰.۴۷۶۶)	-۱۸۵.۵۱ (۰.۴۴۶۹)	-۱۸۶.۴۰ (۱۸.۸۱)	-۱۸۶.۱۸۵ (۳.۶۸۷۶)
F10	Min (SD)	-۱.۰۳۱ (۰.۰۰۲۱)	-۱.۰۳۰ (۰.۰۰۲)	-۱.۰۲۶ (۰.۲۰۸۰)	-۱.۰۳۰ (۰.۰۱۳۲)	-۱.۰۳۱ (۰.۰۰۲۱)
F11	Min (SD)	۴.۱۲۹E-۰۴ (۰.۰۰۲۹)	۶.۰۳E-۰۴ (۳.۲۷E-۰۴)	۷.۷۳E-۰۴ (۱.۱۸۷۲)	۰.۰۰۱ (۰.۰۲۱۱)	۰.۰۱۴۸ (۳.۹۰E-۰۲)
F12	Min (SD)	-۱.۰۴۷۴ (۴.۸۸۷E-۰۶)	-۱.۰۴۷۳ (۲.۱۴E-۰۵)	-۱.۰۴۷۲ (۰.۰۱۵۴)	-۱.۰۴۷۴ (۴.۸۶E-۰۴)	-۱.۰۴۷۴ (۲.۸۴E-۰۴)
F13	Min (SD)	-۱.۰۶۷۸ (۰.۰۲۶۵)	-۱.۰۶۷۵ (۰.۰۵۳)	-۱.۰۶۷۳ (۱۵.۸۸۶)	-۱.۰۶۷۷ (۰.۹۳۶۵)	-۱.۰۶۶۵ (۰.۷۳۳)
F14	Min (SD)	-۲.۰۶۲۶ (۱.۲۸۱E-۵)	-۲.۰۶۲۶ (۰.۰۰۱)	-۲.۰۶۲۶ (۰.۰۰۶)	-۲.۰۶۲۶ (۱.۱۱E-۰۴)	-۲.۰۶۲۶ (۰.۰۰۲)

جدول ۸: زمان اجرای الگوریتم‌های مورد بررسی برای برخی از توابع محک استاندارد.

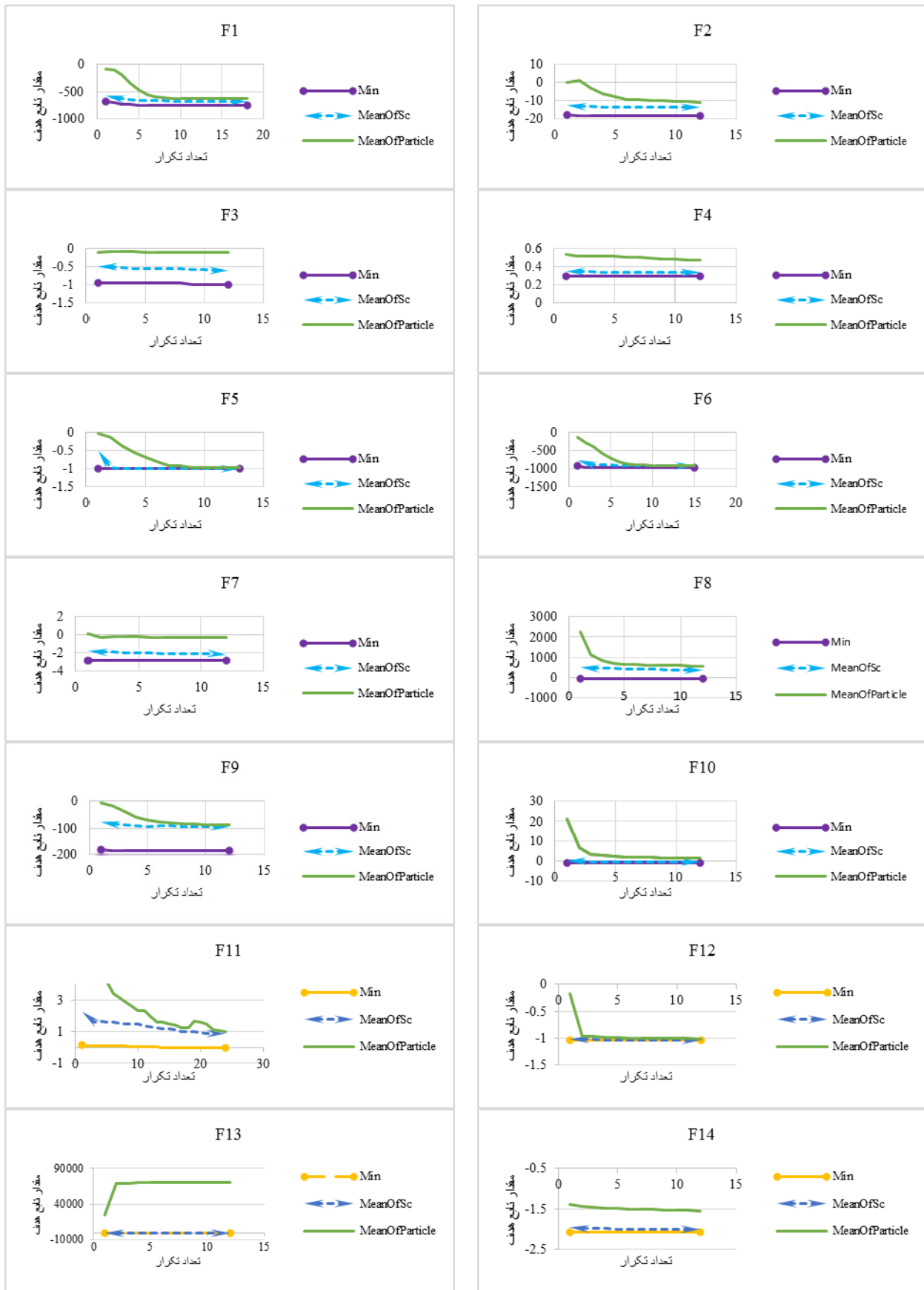
تابع	الگوریتم				
	CHSS	HSS	HSA	GA	PSO
F1	۰.۶۸۳	۰.۷۱۴	۰.۲۲۱	۲.۲۱۴	۰.۲۹۳
F2	۰.۴۸۳	۰.۴۸۶	۰.۲۱۳	۳.۵۰۰	۰.۳۲۹
F3	۰.۴۳۷	۰.۵۹۳	۰.۲۰۹	۲.۶۶۰	۰.۲۶۰
F4	۰.۴۶۹	۰.۵۸۴	۰.۲۱۰	۲.۷۳۷	۰.۲۷۴
F5	۰.۶۸۳	۰.۷۱۴	۰.۲۲۱	۲.۷۱۴	۰.۲۹۳
F6	۰.۹۳۸	۱.۰۴۷	۰.۲۱۰	۲.۸۱۷	۰.۲۹۴
F7	۰.۶۳۳	۰.۶۳۶	۰.۲۱۶	۲.۷۲۹	۰.۲۷۴
F8	۱.۶۲۵	۱.۸۷۸	۰.۲۱۵	۲.۸۱۵	۰.۳۷۴
F9	۰.۷۷۵	۰.۸۴	۰.۲۰۷	۲.۷۹۳	۰.۲۴۳
F10	۰.۸۲۸	۰.۷۷	۰.۲۲۰	۲.۸۴۷	۰.۲۵۴

با توجه به نتایج آورده شده در جدول ۷ که شامل مقدار انحراف معیار (SD) و مقدار کمینه (Min) می‌شود می‌توان مشاهده کرد که مقدار کمینه حاصل از اجرای الگوریتم بر روی تابع F8 نسبت به دیگر الگوریتم‌های مورد بررسی به مراتب بهتر می‌باشد. به عبارت دیگر نه تنها به مقدار کمینه رسیده است بلکه انحراف معیار نیز کمتر است.

بر اساس نتایج آورده شده برای تابع F4، F7، F12 و F14 می‌توان مشاهده کرد مقدار کمینه یکسان است اما انحراف معیار الگوریتم پیشنهادی از سایر الگوریتم‌ها کمتر می‌باشد. زیرا استفاده از نظریه آشوب در الگوریتم‌های فراابتکاری منجر به کاهش انحراف معیار در نتایج حاصل از الگوریتم و همچنین بهبود کارایی الگوریتم می‌شود.

بررسی نتایج حاصل روی توابع F3 و F5 بیانگر آن است که اگر الگوریتم پیشنهادی در رسیدن به مقدار کمینه نسبت به دیگر الگوریتم‌ها کمی ضعیف عمل کند، اختلاف نتایج به دست آمده بسیار کم بوده و انحراف معیار الگوریتم پیشنهادی از اکثر الگوریتم‌ها بسیار کمتر می‌باشد. به عبارت دیگر، ثبات بیشتری در ارائه نتایج دارد.





شکل ۴: نمودار مقدار کمینه، میانگین مراکز کره و میانگین ذرات حاصل از توابع محک با استفاده از الگوریتم پیشنهادی.

تمامی توابع مورد بررسی با افزایش تعداد تکرار، مقدار کمینه و میانگین مراکز کره بهبود یافته و همچنین در اکثر توابع مورد بررسی با افزایش تعداد تکرار مقادیر میانگین ذرات به مقدار کمینه نزدیک می‌شود.

در شکل ۴ نمودار میانگین مراکز کره، میانگین ذرات و مقدار کمینه نتایج حاصل از اجرای الگوریتم پیشنهادی بر روی توابع محک مورد بررسی آورده شده است. همان طور که در شکل ۴ مشاهده می‌کنید در

in *Nonlinear Science and Numerical Simulation*, vol. 18, no. 2, pp. 327-340, Feb. 2013.

- [11] B. Alatas, "Chaotic bee colony algorithms for global numerical optimization," *Expert Systems with Applications*, vol. 37, no. 8, pp. 5682-5687, Aug. 2010.
- [12] A. H. Gandomi and X. S. Yang, "Chaotic bat algorithm," *J. of Computational Science*, vol. 5, no. 2, pp. 224-232, Mar. 2014.
- [13] H. Karami, M. J. Sanjari, and G. B. Gharehpetian, "Hyper-spherical search (HSS) algorithm: a novel meta-heuristic algorithm to optimize nonlinear functions," *Neural Computing and Applications*, vol. 25, no. 6, pp. 1455-1465, Nov. 2014.
- [14] P. Arena, R. Caponetto, L. Fortuna, A. Rizzo, and M. La Rosa, "Self-organization in nonrecurrent complex systems," *International J. of Bifurcation and Chaos*, vol. 10, no. 5, pp. 1115-1125, May 2000.
- [15] B. Alatas, E. Akin, and A. B. Ozer, "Chaos embedded particle swarm optimization algorithms," *Chaos, Solitons & Fractals*, vol. 40, no. 4, pp. 1715-1734, May 2009.
- [16] E. N. Lorenz, "Deterministic nonperiodic flow," *J. of the Atmospheric Sciences*, vol. 20, no. 2, pp. 130-141, Mar. 1963.
- [17] L. Dos Santos Coelho and V. C. Mariani, "Use of chaotic sequences in a biologically inspired algorithm for engineering design optimization," *Expert Systems with Applications*, vol. 34, no. 3, pp. 1905-1913, Apr. 2008.
- [18] S. Das and P. N. Suganthan, *Problem Definitions and Evaluation Criteria for CEC 2011 Competition on Testing Evolutionary Algorithms on Real World Optimization Problems*, Technical Report, Dec. 2010.

**محمد کلانتری** در سال ۱۳۷۷ مقطع کارشناسی در مهندسی کامپیوتر را در دانشگاه علم و صنعت ایران به پایان رساند و مدارک کارشناسی ارشد و دکترای مهندسی کامپیوتر خود را به ترتیب در سال‌های ۱۳۸۰ و ۱۳۸۷ از دانشگاه صنعتی امیرکبیر (پلی تکنیک تهران) دریافت نمود و هم‌اکنون استادیار دانشکده مهندسی کامپیوتر دانشگاه تربیت دبیر شهید رجایی تهران می‌باشد. زمینه‌های تحقیقاتی مورد علاقه ایشان عبارتند از: پردازش سیگنال آماری، بهبود سیگنال گفتار، آرایه میکروفونی کروی، فرآیندهای تصادفی.

**سکینه سهرابی** در سال ۱۳۹۲ مدرک کارشناسی مهندسی نرم‌افزار کامپیوتر خود را از دانشکده شریعی تهران و در سال ۱۳۹۶ مدرک کارشناسی ارشد مهندسی نرم‌افزار کامپیوتر خود را از دانشگاه تربیت دبیر شهید رجایی دریافت نمود. از سال ۱۳۹۰ نام‌برده به برنامه‌نویسی و طراحی کاربردهای تحت وب مشغول بوده و از سال ۱۳۹۳ به استخدام آموزش و پرورش درآمده است. زمینه‌های تحقیقاتی مورد علاقه ایشان عبارتند از: بهینه‌سازی، الگوریتم‌های تکاملی، محاسبات نرم.

**حمیدرضا رشیدی کنعان** در سال ۱۳۸۷ مدرک دکترای برق (گرایش الکترونیک) خود را از دانشگاه صنعتی امیرکبیر (پلی تکنیک تهران) دریافت نمود و هم‌اکنون استادیار دانشکده مهندسی کامپیوتر دانشگاه شهید رجایی می‌باشد. نام‌برده قبل از پیوستن به دانشگاه شهید رجایی، از سال ۱۳۸۸ تا ۱۳۹۴ استادیار دانشگاه بوعلی سینا بوده است. زمینه‌های تحقیقاتی مورد علاقه ایشان عبارتند از پردازش تصویر و ویدئو، بینایی ماشین، بازشناسی الگو، بیومتریک، الگوریتم‌های بهینه‌سازی و یادگیری عمیق.

**حسین کرمی** تحصیلات خود را در مقاطع کارشناسی، کارشناسی ارشد و دکترای مهندسی برق - قدرت به ترتیب در سال‌های ۱۳۸۹، ۱۳۹۱ و ۱۳۹۶ از دانشگاه صنعتی امیرکبیر به پایان رسانده است و هم‌اکنون محقق پسادکتر در دانشکده مهندسی برق - قدرت دانشگاه صنعتی امیرکبیر می‌باشد. نام‌برده دارای بیش از ۱۷ مقاله در ژورنال‌های نمایه‌شده در ISI و ۳۰ مقاله در کنفرانس‌های ملی و بین‌المللی است. همچنین از ایشان سه اختراع داخلی و یک اختراع بین‌المللی به ثبت رسیده است. زمینه‌های تحقیقاتی مورد علاقه ایشان عبارتند از: بهینه‌سازی در سیستم قدرت، سیستم‌های هوشمند خانگی، برنامه‌ریزی سیستم‌های انرژی، پایش برخط ترانسفورماتورهای قدرت، عیب تخلیه جزئی در ترانسفورماتورهای قدرت.

## ۶- نتیجه گیری

در این مقاله نسخه جدیدی از الگوریتم جستجوی ابرکروی با استفاده از نظریه آشوب ارائه شده که ضعف جستجوی ابرکروی استاندارد را برطرف کرده است. الگوریتم جستجوی ابرکروی استاندارد دارای ضعف همگرایی و افزایش تعداد تکرار اجرای الگوریتم برای رسیدن به جواب کمینه در برخی از توابع محک استاندارد می‌باشد. در اکثر الگوریتم‌های فراابتکاری با اجزای تصادفی، تابع توزیع احتمال استفاده شده تابع توزیع احتمال گاوسی یا یکنواخت می‌باشد اما در روش ارائه شده در این مقاله، پس از انجام آزمایشات مختلف به این نتیجه رسیده‌ایم که از نگاشت چیشف برای قسمت تولید ذرات و نگاشت لیوویچ برای قسمت جستجو استفاده نماییم. اعداد تولیدشده توسط نگاشت‌های مذکور جایگزین اعداد تصادفی با تابع توزیع احتمال یکنواخت در الگوریتم جستجوی ابرکروی استاندارد شده است. نتایج حاصل بیانگر آن است که الگوریتم پیشنهادی ضمن برخورداری از همگرایی مناسب، توانایی بالایی در یافتن جواب کمینه نسبت به الگوریتم‌های ژنتیک، ازدحام ذرات، جستجوی هارمونی و همچنین الگوریتم جستجوی ابرکروی استاندارد دارد. استفاده از نظریه آشوب منجر به بهبود و کاهش انحراف معیار نتایج حاصل از اجرای الگوریتم شده است.

## ۷- تشکر و قدردانی

این پژوهش با حمایت مالی دانشگاه تربیت دبیر شهید رجایی طبق قرارداد شماره ۲۶۲۳۶ مورخ ۱۳۹۷/۱۰/۹۵ انجام گردیده است.

## مراجع

- [1] X. S. Yang, *Nature-Inspired Metaheuristic Algorithms*, Luniver Press, 2010.
- [2] S. A. Ahmadi, "Human behavior-based optimization: a novel metaheuristic approach to solve complex optimization problems," *Neural Computing and Applications*, vol. 28, no. 1, pp. 233-244, Dec. 2017.
- [3] A. H. Gandomi, A. H. Alavi, and S. Talatahari, "Structural optimization using krill herd algorithm," *Swarm Intelligence and Bio-Inspired Computation: Theory and Applications*, pp. 335-349, 2013.
- [4] R. Nallakumar, N. Sengottaiyan, and K. S. Priya, "A survey on scheduling and the attributes of task scheduling in the cloud," *Int. J. Adv. Res. Comput. Commun. Eng.*, vol. 3, no. 10, pp. 8167-8171, Oct. 2014.
- [5] C. W. Tsai and J. J. Rodrigues, "Metaheuristic scheduling for cloud: a survey," *IEEE Systems J.*, vol. 8, no. 1, pp. 279-291, Mar. 2014.
- [6] F. Dong, J. Luo, L. Gao, and L. Ge, "A grid task scheduling algorithm based on QoS priority grouping," in *Proc. 5th Int. Conf on Grid and Cooperative Computing, GCC'06*, pp. 58-61, Hunan, China, 21-23 Oct. 2006.
- [7] U. Parlitz and L. Junge, "Synchronization of chaotic systems," in *Proc. European Control Conf., ECC'99*, pp. 4637-4642, Karlsruhe, Germany, 31 Aug.-3 Sept. 1999.
- [8] D. Yang, G. Li, and G. Cheng, "On the efficiency of chaos optimization algorithms for global optimization," *Chaos, Solitons & Fractals*, vol. 34, no. 4, pp. 1366-1375, Nov. 2007.
- [9] B. Alatas, "Chaotic harmony search algorithms," *Applied Mathematics and Computation*, vol. 216, no. 9, pp. 2687-2699, Jul. 2010.
- [10] A. H. Gandomi, G. J. Yun, X. S. Yang, and S. Talatahari, "Chaos-enhanced accelerated particle swarm optimization," *Communications*