

دانشگاه صنعتی امیرکبیر
(پلی تکنیک تهران)

گروه مستقل مهندسی رباتیک

گزارش تحقیقاتی درس هوش محاسباتی – الگوریتم‌های تکاملی

**الگوریتم تکامل تفاضلی با تطبیق استراتژی برای بهینه‌سازی
سراسری عددی**

استاد درس:

دکتر نیک آبادی

نام دانشجو:

نوید خزاعی

۹۲۱۳۵۰۰۸

اردیبهشت ۱۳۹۳

چکیده

تکامل تفاضلی یکی از روش‌های قدرتمند بهینه‌سازی در فضای پیوسته است که در آن برداری به نام بردار آزمایشی تولید می‌شود که استراتژی‌های تولید متفاوتی دارد و هر استراتژی نیز پارامترهایی دارد. برای پیدا کردن استراتژی مناسب و پارامترهای مناسب آن، جست‌وجوی کورکورانه و آزمون و خطا بسیار پرهزینه است. همچنین استراتژی‌ها و پارامترها ممکن است در مراحل مختلف تکامل متفاوت باشند. در مقاله‌ی انتخابی ما روشی به نام تکامل تفاضلی خودتطبیقی ارائه شده است که این مشکلات را تا حدی حل می‌کند و استراتژی‌ها بر اساس احتمال تولید بردارهای امیدبخش برای ورود به نسل بعد، که در طول تکامل یادگرفته می‌شود، از استخر استراتژی‌ها انتخاب می‌شوند. آزمایش‌ها با ۲۶ تابع آزمون معروف در تکامل تفاضلی انجام شده‌اند و با ۸ روش معمول مقایسه انجام شده است. برای درک بهتر ادبیات مساله، کارهای مرتبط را فراتر از آنچه در مقاله آورده شده بود بیان کردیم و جدول‌هایی را برای این منظور طراحی نمودیم. تحلیل‌هایی نیز از نتایج گرفته‌شده ارائه خواهد شد.

فهرست مطالب

| | |
|----|--------------------------------|
| ۱ | ۱ مقدمه |
| ۴ | ۲ تکامل تفاضلی |
| ۴ | ۱.۲ جمعیت و افراد |
| ۴ | ۲.۲ عمل جهش |
| ۵ | ۳.۲ عمل بازترکیبی |
| ۶ | ۴.۲ عمل انتخاب |
| ۷ | ۳ کارهای مرتبط |
| ۱۲ | ۴ SaDE |
| ۱۲ | ۱.۴ تطبیق استراتژی تولید بردار |
| ۱۷ | ۲.۴ تطبیق پارامترها |
| ۲۰ | ۵ آزمایش‌های عددی و ارزیابی |
| ۲۰ | ۱.۵ توابع آزمون |
| ۲۶ | ۲.۵ الگوریتم‌های مقایسه‌شده |
| ۲۹ | ۳.۵ نتایج آزمایش و تحلیل‌ها |
| ۲۹ | ۱.۳.۵ مقایسه با روش‌های معمول |
| ۳۴ | ۲.۳.۵ مقایسه با روش‌های تطبیقی |
| ۳۵ | ۳.۳.۵ مقایسه‌ی کارایی کلی: |
| ۳۷ | ۶ نتیجه‌گیری |
| ۳۸ | مراجع |

۱ مقدمه

الگوریتم‌های تکاملی، الگوریتم‌هایی هستند که از تکامل طبیعی موجودات الهام گرفته شده‌اند و در حل مسایل بهینه‌سازی، در زمینه‌های بسیار متنوعی، موفق بوده‌اند. اما در عمل، برای رسیدن به یک پیاده‌سازی کارا در مساله‌ی واقعی، نیاز است استراتژی‌های مناسب برای انتخاب عملگرهای تکاملی انتخاب شوند و علاوه بر آن، باید مقادیر مناسب برای پارامترهای هر استراتژی نیز به دست آید. انجام این انتخاب‌ها با روش‌های آزمون و خطا، می‌تواند بسیار وقت‌گیر باشد. به همین دلیل، پژوهش‌های بسیاری بر روی تطبیق عملگرها و پارامترهای الگوریتم‌های تکاملی انجام شده‌است. برای بررسی بهتر روش‌های تطبیق، دسته‌بندی‌های گوناگونی ارایه شده‌است که از جمله‌ی آن‌ها می‌توان به موردی که در [۱] آورده شده‌است، اشاره کرد. در این دسته‌بندی، روش‌های تطبیق به دو دسته‌ی مطلق^۱ و تجربی^۲ تقسیم می‌شوند. روش‌های مطلق، چگونگی تغییرات پارامترها را از قبل مشخص می‌کنند. این در حالی است که روش‌های تجربی، قوانین تنظیم پارامترها را با توجه به اصل رقابت در الگوریتم‌های تکاملی، به‌روزرسانی می‌کنند. از طرفی، روش‌های تنظیم پارامتر در الگوریتم‌های تکاملی به سه گونه تقسیم می‌شوند [۲]:

- قطعی^۳: قوانین مطلق، که بر اساس مجموعه‌ای از استدلال‌های منطقی در زمینه‌ی حل مساله‌ی مورد نظر به دست آمده‌اند، چگونگی تغییرات پارامترها را مشخص می‌کنند و هیچ بازخوردی از پروسه‌ی جست‌وجو نمی‌گیرند.
- تطبیقی^۴: در این روش، پارامترها با توجه به بازخوردی که از پروسه‌ی جست‌وجو گرفته می‌شود، تنظیم می‌شوند.
- خود تطبیقی^۵:

^۱ Absolute

^۲ Empirical

^۳ Deterministic

^۴ Adaptive

^۵ Self-Adaptive

پارامترها در درون خود افراد جامعه قرار داده می‌شوند و روال تکامل با در نظر گرفتن پارامترها در جواب‌های مساله انجام می‌گیرد. به این ترتیب که پارامترهایی که جواب‌های بهتری تولید کنند، به بقا ادامه می‌دهند.

الگوریتم تکامل تفاضلی که در سال ۱۹۹۷ ارائه شد [۳]، روشی ساده و در عین حال قدرتمند برای جست‌وجو در فضای پیوسته است. این روش مبتنی بر جمعیت است و یک جست‌وجوی تصادفی به شمار می‌آید. این الگوریتم در زمینه‌های زیادی به کار گرفته شده و نتایج موفقیت آمیزی از خود نشان داده است. در این الگوریتم که در بخش آتی به بررسی جزئیات آن خواهیم پرداخت، استراتژی‌های زیادی برای تولید برداری به نام بردار آزمایشی^۱ وجود دارد که ممکن است تنها برخی از آن‌ها در حل یک مساله‌ی خاص، کارا باشند. از طرفی، سه پارامتر مهم در تکامل تفاضلی وجود دارند که مقادیر آن‌ها تاثیر چشم‌گیری بر کارایی الگوریتم دارد. این سه پارامتر عبارت‌اند از تعداد جمعیت NP ، ضریب مقیاس F ، و نرخ بازترکیبی CR .

انتخاب استراتژی مناسب برای تولید بردار آزمایشی و تنظیم پارامترهای یاد شده به روش سعی و خطا، هزینه‌بر و زمان‌گیر است. علاوه بر آن در حین اجرای الگوریتم، با حرکت در فضای جست‌وجو و جابه‌جا شدن میان نواحی مختلف، ممکن است استراتژی متفاوتی برای کارایی در آن ناحیه نیاز باشد و یا مقادیر بهینه‌ی پارامترها تغییر کند. به همین دلیل بهتر است استراتژی انتخاب بردار آزمایشی و مقادیر پارامترها در حین تکامل به صورت تطبیقی تنظیم شوند. در مقاله‌ی انتخاب شده، یک روش خودتطبیقی برای انتخاب بهترین استراتژی انتخاب بردار آزمایشی و مقادیر پارامترهای مناسب آن ارائه شده است. در روش ارائه شده، خودتطبیقی به وسیله‌ی یادگیری از تجربه‌ی قبلی در تولید جواب‌های امیدبخش انجام می‌شود.

در هر نسل، مجموعه‌ای از استراتژی‌های انتخاب به همراه مقادیر پارامترهای نسبت داده شده به آن‌ها، به صورت جداگانه به افراد مختلف جمعیت فعلی اعمال خواهد شد و این انتخاب، با توجه به احتمال‌های یاد گرفته شده در نسل‌های قبل انجام می‌شود. در ادامه‌ی این گزارش، در بخش ۲ به معرفی الگوریتم تفاضل تکاملی می‌پردازیم. در بخش ۳ مروری بر کارهای مرتبط آورده شده است. در بخش ۴ الگوریتم خودتطبیقی

^۱ Trial vector

تفاضل تکاملی^۱ ارایه شده است و در بخش ۵ ارزیابی آن آورده شده است. در پایان، در بخش ۶ به نتیجه گیری از مقاله ی بررسی شده خواهیم پرداخت.

^۱ SaDE

۲ تکامل تفاضلی

۱.۲ جمعیت و افراد

در تکامل تفاضلی، یک جمعیت NP تایی از بردار پارامترهای D بُعدی داریم که افراد جمعیت ما در نسل G هستند که راه حل به سمت بهینه‌ی سراسری را در خود دارند، مانند:

$$\mathbf{X}_{i,G} = \{x_{i,G}^1, \dots, x_{i,G}^D\}, i = 1, \dots, NP \quad (1)$$

بهرتر است جمعیت اولیه تا حد امکان تمام فضای جست و جو را به صورت یکنواخت پوشش دهد و این مقداردهی، به صورت تصادفی و با مشخص کردن مقادیر کمینه و بیشینه‌ای برای هر پارامتر، انجام شود که نمونه‌ی آن در فرمول شماره‌ی (۱) مقاله‌ی مورد نظر [۴]، آورده شده است.

۲.۲ عمل جهش

پس از تولید جمعیت اولیه، یک بردار جهش $V_{i,G}$ برای هر فرد $\mathbf{X}_{i,G}$ ، که بردار هدف نامیده می‌شود، تولید می‌شود. این بردار که به شکل $V_{i,G} = \{v_{i,G}^1, \dots, v_{i,G}^D\}$ است، با استراتژی‌های متفاوتی محاسبه می‌شود، برای نمونه می‌توان به پنج نمونه‌ی پیاده‌سازی شده^۱ از این استراتژی‌ها اشاره کرد:

۱. DE/rand/1 :

$$\mathbf{V}_{i,G} = \mathbf{X}_{r_1^i,G} + F \cdot (\mathbf{X}_{r_2^i,G} - \mathbf{X}_{r_3^i,G}) \quad (2)$$

۲. DE/best/1 :

$$\mathbf{V}_{i,G} = \mathbf{X}_{best,G} + F \cdot (\mathbf{X}_{r_1^i,G} - \mathbf{X}_{r_2^i,G}) \quad (3)$$

^۱ پیاده‌سازی این استراتژی‌ها در آدرس <http://www1.icsi.berkeley.edu/storn/code.html> برای استفاده‌ی عمومی در دسترس است.

۳. DE/rand-to-best/1 :

$$\mathbf{V}_{i,G} = \mathbf{X}_{i,G} + F \cdot (\mathbf{X}_{best,G} - \mathbf{X}_{i,G}) + F \cdot (\mathbf{X}_{r_1^i,G} - \mathbf{X}_{r_2^i,G}) \quad (4)$$

۴. DE/best/2 :

$$\mathbf{V}_{i,G} = \mathbf{X}_{best,G} + F \cdot (\mathbf{X}_{r_1^i,G} - \mathbf{X}_{r_2^i,G}) + F \cdot (\mathbf{X}_{r_3^i,G} - \mathbf{X}_{r_4^i,G}) \quad (5)$$

۵. DE/rand/2 :

$$\mathbf{V}_{i,G} = \mathbf{X}_{r_1^i,G} + F \cdot (\mathbf{X}_{r_2^i,G} - \mathbf{X}_{r_3^i,G}) + F \cdot (\mathbf{X}_{r_4^i,G} - \mathbf{X}_{r_5^i,G}) \quad (6)$$

اندیس‌های $r_1^i, r_2^i, r_3^i, r_4^i, r_5^i$ اعداد طبیعی منحصر به فردی هستند که به صورت تصادفی در بازه‌ی $[1, NP]$ تولید می‌شوند که از اندیس i نیز متفاوت هستند. این اندیس‌ها برای هر بردار جهش تنها یک بار تولید می‌شوند. ضریب مقیاس F ^۱ نیز یک پارامتر کنترلی مثبت است که تاثیر بردار تفاضل را کنترل می‌کند. بردار $\mathbf{X}_{best,G}$ نیز بهترین فرد موجود در نسل G است.

۳.۲ عمل بازترکیبی

پس از فاز جهش، عملگر بازترکیبی به هر جفت بردارهای هدف $\mathbf{X}_{i,G}$ و جهش $\mathbf{V}_{i,G}$ نظیر آن اعمال می‌شود تا بردار آموزشی $\mathbf{U}_{i,G} = (u_{i,G}^1, u_{i,G}^2, \dots, u_{i,G}^D)$ ساخته شود. در نسخه‌ی پایه‌ی تکامل تفاضلی، بازترکیبی دو جمله‌ای (یک‌نواخت) به این شکل تعریف می‌شود:

$$u_{i,G}^j = \begin{cases} v_{i,G}^j, & \text{if } (rand_j[\circ, 1] \leq CR) \text{ or } (j = j_{rand}) \\ x_{i,G}^j, & \text{otherwise} \end{cases}, j = 1, 2, \dots, D. \quad (7)$$

^۱ Scaling factor

در فرمول ۷ نرخ بازترکیبی CR توسط کاربر در بازه‌ی $(0, 1)$ تعریف می‌شود که کسری از پارامترهایی را که باید از بردار جهش کپی شوند، مشخص می‌کند. j_{rand} یک عدد تصادفی در بازه‌ی $[1, D]$ است. عملگر بازترکیبی دیگری نیز وجود دارد که در آن، پارامترهای بردار آموزشی از نقطه‌ای تصادفی با پارامترهای بردار جهش جایگزین می‌شوند، تا وقتی که برای اولین بار $CR > rand_j[0, 1)$ شود. پارامترهای باقی‌مانده نیز از بردار هدف به ارث برده می‌شوند. دقت کنید که شرط $j = j_{rand}$ نیز قرارداده شده‌است تا بردار آزمایشی تولید شده، دست کم در یک پارامتر با بردار هدف تفاوت داشته باشد. این عملگر شبیه عملگر بازترکیبی حلقوی دو نقطه‌ای است.

۴.۲ عمل انتخاب

اگر پارامترهای یک بردار آزمایشی از محدوده‌های کمینه و بیشینه‌ی نظیر خود خارج شوند، با توزیعی یکنواخت، مقداری تصادفی در بازه‌ای از پیش تعیین‌شده انتخاب شده و جاگذاری می‌شود. پس از آن، مقدار تابع ارزیابی برای همه‌ی بردارهای آموزشی محاسبه می‌شود و عملگر انتخاب اعمال می‌شود؛ به این ترتیب که مقدار تابع ارزیابی برای هر بردار آزمایشی $f(U_{i,G})$ با مقدار تابع ارزیابی برای بردار هدف نظیر آن $f(X_{i,G})$ مقایسه می‌شود، در صورتی که این مقدار برای بردار آزمایشی کمتر یا مساوی مقدار بردار هدف باشد، بردار هدف در جمعیت با بردار آزمایشی نظیرش جایگزین می‌شود و به نسل بعد می‌رود؛ در غیر این صورت خودش در جمعیت باقی می‌ماند و در نسل بعد نیز ظاهر می‌شود.

عملگر انتخاب را می‌توان به این شکل نشان داد:

$$X_{i,G+1} = \begin{cases} U_{i,G}, & \text{if } f(U_{i,G}) \leq f(X_{i,G}) \\ X_{i,G}, & \text{otherwise} \end{cases} \quad (8)$$

الگوریتم DE در جدول ۱ مقاله‌ی مورد نظر آورده شده‌است که در آن نسل اولیه تولید می‌شود و سپس تا رسیدن به یک شرط خاتمه، سه مرحله‌ی جهش، بازترکیبی و انتخاب تکرار می‌شود.

۳ کارهای مرتبط

کارایی الگوریتم DE معمول، به شدت به استراتژی تولید بردار آزمایشی و مقدار پارامترهای مورد استفاده بستگی دارد. انتخاب نامناسب استراتژی یا مقادیر پارامترها ممکن است منجر به همگرایی زودرس و یا سکون^۱ شود که اثبات آن به طور مفصل در [۵]، [۶]، [۷]، [۸] و [۹] آورده شده است. (متأسفانه مقاله‌ی مورد نظر در نوشتن این پاراگراف نه تنها ترتیب عددی ارجاعات را حفظ نکرده است بلکه آشکارا قالب ارجاع را نیز برهم زده است!).

از آن جا که مقاله‌ی مورد نظر از بررسی این اثبات‌ها عبور کرده است، نگاهی مختصر بر این مراجع داشتیم، که البته کتاب [۵] در دسترس نبود. در [۶] ابتدا مثالی از اجرای الگوریتم که به سکون منجر می‌شود آورده شده است، سپس به بررسی علل این سکون پرداخته است و با تعمیم نقش متغیرها در حالت کلی‌تر اجرای الگوریتم، و تاثیر آن‌ها در مشخص کردن فضای جست‌وجوی ممکن، تحلیل‌هایی ارائه شده است. همچنین اثبات شده است که در رابطه‌ی بازترکیبی خاص استفاده شده در آن مقاله، تنها تعداد مشخصی از بردارهای آزمایشی قابل تولید هستند که به نظر می‌رسد با روشی مشابه می‌توان هر رابطه‌ی بازترکیبی دیگر مانند ۷ را بررسی نمود. در نتیجه‌ی تولید بردارهای آزمایشی محدود، اگر هیچ کدام از آن‌ها نتوانند به جای بردار هدف متناظر در جمعیت جای‌گزین شوند، سکون رخ می‌دهد. همچنین مقادیری برای پارامترها در این مقاله پیشنهاد شده است.

در [۷] الگوریتم بر روی چند تابع تست تک‌قله‌ای و چند قله‌ای اجرا شده است و با دو نمونه از الگوریتم استراتژی تکاملی مقایسه شده است و بیشتر از آن که اثبات خاصی آورده شده باشد، از استدلال‌های شهودی و مشخص کمک گرفته شده است و در انتها، با مقایسه‌ی سه الگوریتم، به این نتیجه می‌رسد که حساسیت DE نسبت به تغییرات در پارامترها بسیار بیشتر است.

در [۸] آورده شده است که می‌توان با کنترل تنوع جمعیت، تعادلی را میان اکتشاف و استخراج ایجاد نمود و در نتیجه از سکون و همگرایی زودرس جلوگیری کرد. با توجه به نقش پارامترها در کارایی الگوریتم، یک روش تطبیقی برای DE ارائه شده است که

^۱ Stagnation

مبتنی بر ایده‌ی کنترل تنوع جمعیت است و روابط کنترلی در آن با استفاده از پارامترهای الگوریتم و پارامترهای معرفی شده استخراج شده است. با بدست آوردن واریانس مورد انتظار جمعیت پس از بازترکیبی، و نوشتن این رابطه برای دو نسل متوالی و بررسی نسبت آن‌ها، به یک پارامتر کنترلی رسیده است که تاثیر آن بر روی همگرایی و سکون را تحلیل کرده و مقادیری را برای آن پیشنهاد داده است. همچنین تحلیلی بر روی یک نسخه‌ی چند جمعیتی از روش تطبیقی ارایه شده نیز انجام شده است.

در [۹] تلاش شده است تا با تغییر ضریب مقیاس، بهبود ایجاد شود. دو حالت انتخاب تصادفی و انتخاب متناسب با زمان برای این پارامتر بررسی شده است و نتیجه با DE معمول، PSO [۱۰]، PSO-TVIW [۱۱] و PSO-RANDIW [۱۲] در هفت تابع آزمون پرکاربرد مقایسه شده است، این کار هیچ گونه تحلیل ریاضی قابل قبولی بر روی پیشرفت‌های حاصل شده انجام نداده است.

ابتدایی ترین کارها در زمینه‌ی انتخاب استراتژی و تعیین پارامترهای آن، که همگی راه کارهایی تجربی را به دست می دهند، در جدول ۱ آورده شده اند.

جدول ۱: روش‌های تجربی تنظیم استراتژی و پارامترها*

| مرجع | موارد بررسی شده | مقادیر پیشنهادی | توضیحات |
|------|--|---|--|
| [۱۳] | NP, CR, F مقایسه با ^۱ ASA و ^۲ ANM | $NP \in [5D, 10D]$ $CR_{init} = 0.5$ $F_{init} = 0.5$ $F_{eff} \in [0.4, 1]$ | $CR_{init} = 0.9$ برای تک‌قله‌ای و یا همگرایی زودرس مطلوب. افزایش F یا NP در صورت همگرایی زودرس نامطلوب. |
| [۱۴] | DE/current-to-rand/1 NP, F, K^3 | $NP = 20D$ $F = 0.8$ $K = 0.5$ | افزایش NP و F ، یا کاهش K در صورت همگرایی زودرس. کاهش NP یا F ، یا انتخاب $K \in [0, 1]$ به صورت تصادفی، در حالتی که سکون رخ دهد. در صورت عدم موفقیت باید از DE/rand/1/bin با CR کوچک استفاده شود. |
| [۷] | NP, CR, F مقایسه با ES و ^۴ CMA-ES | $NP \in [3D, 8D]$ $CR \in [0.3, 0.9]$ $F = 0.6$ | کاملاً تجربی است و برای سه تابع خاص آزمایش شده است. |
| [۱۵] | CR, F بررسی شهودی تاثیر پارامترها | $CR \in [0, 0.2]$ $F_{init} = 0.9$ $F_{eff} \in [0.4, 0.95]$ | اگر پارامترهای تابع وابسته باشند آن‌گاه: $CR \in [0.9, 1]$ ۲۵ تابع آزمون مختلف در حالات ۱۰ و ۳۰ بعدی مقایسه شده‌اند. |

* اندیس‌های $init$ و eff به ترتیب نشان‌گر مقدار اولیه و مقدار موثر است.

تنظیم دستی پارامترها در DE گاهی به نتایج متناقضی در پژوهش‌های موجود می‌انجامد که سبب سردرگمی پژوهشگران و دانشمندانی می‌شود که برای موارد کاربردی قصد استفاده از این الگوریتم را دارند، و این به دلیل آن است که اغلب این نتیجه‌گیری‌ها توجیه‌های لازم برای قابل قبول بودنشان در محدوده‌ی خاص مساله‌ی مورد بررسی را

^۱ Adaptive Simulated Annealing [۱۶]

^۲ Annealed Nelder and Mead [۱۷]

^۳ از پارامترهای استراتژی DE/current-to-rand/1 است.

^۴ Covariance Matrix Adaptation - Evolution Strategy [۱۸]

ندارند؛ حال آن که اعتبار این روش‌ها به خود مساله‌ی بررسی شده، استراتژی‌ها و مقادیر عددی مورد استفاده برای پارامترها وابسته است. به همین دلیل، پژوهشگران به سمت طراحی تکنیک‌هایی روی آوردند که از تنظیم دستی جلوگیری کنند و روش‌های تطبیقی ارائه شدند. خلاصه‌ای از کارهای موجود در این زمینه در جدول ۲ آورده شده است.

جدول ۲: روش‌های تطبیقی تنظیم استراتژی و پارامترها

| مرجع | ایده‌ی بررسی شده | توضیحات |
|------|--|--|
| [۹] | تغییر خطی F با زیاد شدن نسل | در روش اول (DERSF) ^۱ مقدار تصادفی $F \in [0.5, 1]$ در روش دوم (DETVSF) ^۲ ، F به صورت خطی بین مقادیر max و min تغییر می‌کند. |
| [۸] | کنترل تنوع جمعیت | این روش ADE ^۳ نام دارد و پیش‌تر توضیح داده شد. |
| [۱۹] | کنترل F و CR با یک کنترل‌کننده‌ی فازی | این روش FADE ^۴ نام دارد و با توجه به نسل‌های پیشین و مقادیر نسبی توابع هدف، پارامترها را برای جهش و بازترکیبی تطبیق می‌دهد. در ابعاد بالاتر بهتر از DE است. |
| [۲۰] | بررسی [۸] در مسایل چندهدفی | معادلات ADE را برای مسایل چندهدفی بازنویسی نموده و بهبود بخشیده است. |
| [۲۱] | انتخاب $F \in N(0, 1)$ و خودتطبیقی CR با قراردادن آن در هر فرد | برای مسایل چندهدفی نیز بررسی شده است. |

مقاله‌ی مورد بررسی، ابتدا به صورت اجمالی در [۲۲] ارائه شده است. در فاصله‌ی چاپ کار اولیه تا این مقاله، پژوهش‌های دیگری نیز بر روی تطبیق پارامترها و استراتژی‌ها انجام گرفته است. این پژوهش‌ها نیز در جدول ۳ آورده شده است.

^۱ DE with Random Scale Factor

^۲ DE with Time Varying Scale Factor

^۳ Addaptive DE

^۴ Fuzzy Addaptive DE

جدول ۳: روش‌های مطرح‌شده‌ی تطبیقی در فاصله‌ی انتشار [۲۲] تا [۴]

| مرجع | ایده‌ی بررسی‌شده | توضیحات |
|------|--|---|
| [۲۳] | خودتطبیقی F همانند روش [۲۱] | انتخاب $CR \in N(0.5, 0.15)$ برای هر فرد. نتایج بر روی ۴ تابع ارزیابی آزمایش شده‌است و از دیگر نسخه‌های DE بهتر بوده‌است. |
| [۲۴] | خودتطبیقی جمعیت خودتطبیقی F و CR | این روش DESAP ^۱ نام دارد و روش [۲۱] را خودتطبیقی نموده‌است. |
| [۲۵] | معرفی دو پارامتر جدید τ_1, τ_2 (jDE) | مقادیر F به افراد جامعه داده‌شده و عدد $rand \in [0, 1]$ تصادفی تولید می‌شود و اگر $rand \leq \tau_1$ آن‌گاه F در بازه‌ی $[0.1, 1]$ مقداردهی تصادفی می‌شود و در غیر این صورت بی‌تغییر می‌ماند. CR نیز همین‌گونه در بازه‌ی $[0, 1]$ تطبیق داده می‌شود. |

^۱ DE with Self-Adaptive Pareto

^۲ DE with Time Varying Scale Factor

^۳ Addaptive DE

^۴ Fuzzy Addaptive DE

ایده‌ی مطرح‌شده در این مقاله چندان پیچیده نیست: استراتژی‌ها و پارامترهای کنترلی مربوط به آن‌ها، با توجه به نقش هر کدام در تولید جواب‌های امیدبخش در نسل‌های قبل، خودتطبیقی می‌شوند. این روش شامل دو مرحله‌ی کلی است که به ترتیب تطبیق استراتژی و تطبیق پارامترها را در بر می‌گیرد. در ادامه به بررسی این مراحل می‌پردازیم:

۱.۴ تطبیق استراتژی تولید بردار

برای این منظور، یک استخر^۱ استراتژی‌های کاندید در نظر گرفته می‌شود. این استراتژی‌ها به دلخواه و با توجه به موثر بودن آن‌ها در حل مساله‌ی مورد نظر انتخاب می‌شوند، هرچند ممکن است ویژگی‌های یکسانی نداشته باشند. در طول تکامل، برای هر بردار هدف، یک استراتژی از این استخر انتخاب می‌شود و برای عمل جهش استفاده می‌شود. این انتخاب، بر اساس احتمال تولید جواب‌های امیدبخش توسط آن استراتژی که در نسل‌های قبل یادگرفته شده است، انجام می‌شود. هر آن قدر که یک استراتژی در یک نسل، جواب‌های امیدبخش‌تری تولید کند، احتمال انتخاب آن در نسل‌های بعد بیشتر خواهد شد. در ادامه برخی استراتژی‌های موثر در DE را بررسی نموده‌اند و برخی از آن‌ها را در ساخت استخر موجود در پژوهش انتخاب نموده‌اند.

- استراتژی‌های متکی بر بهترین راه حل یافت‌شده مانند “DE/rand-to-best/1/bin”، “DE/best/1/bin” و “DE/best/2/bin” معمولاً همگرایی سریع دارند و برای حل مسایل تک‌قله‌ای کارا هستند. با این حال، در حل مسایل چندقله‌ای در بهینه‌ی محلی به دام می‌افتند.

- استراتژی “DE/rand/1/bin” به طور معمول همگرایی کندتری دارد و به اکتشاف بیشتری منجر می‌شود، به همین دلیل برای مسایل چندقله‌ای بهتر از استراتژی‌های متکی بر بهترین راه حل عمل می‌کنند.

^۱ Pool

• استراتژی‌های برپایه‌ی دو بردار تفاضل به تنوع بیشتری نسبت به استراتژی‌های تک بردار تفاضلی می‌انجامند. مسایل نظری و کاربردی این استراتژی‌ها در [۲۶] آورده شده‌است.

در کار قبلی همین پژوهشگران [۲۲] تنها از دو استراتژی استفاده کرده‌بودند که در پژوهش مورد بررسی آن را به ۴ استراتژی گسترش داده‌اند. استراتژی‌های قبلی “DE/rand/1/bin” و “DE/rand-to-best/2/bin” بوده‌اند که در مسایل DE محبوب بوده‌اند و چون تنوع ایجادشده توسط آن‌ها بیشتر شبیه توزیع گاسی است، اکتشاف بیشتری انجام می‌دهند، حال آن‌که دو استراتژی اضافه‌شده به استخر در پژوهش اخیر، “DE/rand/2/bin” و “DE/current-to-rand/1” هستند که در برابر چرخش مقاوم هستند. فرمول‌های این چهار استراتژی را در ۹ در انتهای صفحه مشاهده می‌کنید. برای بازترکیبی نیز از عملگر دوجمله‌ای که در معادله‌ی ۷ نشان داده‌شد استفاده شده‌است.

(۹)

DE/rand/1/bin:

$$u_{i,j} = \begin{cases} x_{r_1,j} + F \cdot (x_{r_2,j} - x_{r_3,j}), & \text{if } \text{rand}[0, 1) \leq CR \text{ or } j = j_{rand} \\ x_{i,j}, & \text{otherwise} \end{cases}$$

DE/rand-to-best/2/bin:

$$u_{i,j} = \begin{cases} x_{i,j} + F \cdot (x_{best,j} - x_{i,j}) + F \cdot (x_{r_1,j} - x_{r_2,j}) + F \cdot (x_{r_2,j} - x_{r_3,j}), & \text{if } \text{rand}[0, 1) \leq CR \text{ or } j = j_{rand} \\ x_{i,j}, & \text{otherwise} \end{cases}$$

DE/rand/2/bin:

$$u_{i,j} = \begin{cases} x_{r_1,j} + F \cdot (x_{r_2,j} - x_{r_3,j}) + F \cdot (x_{r_4,j} - x_{r_5,j}), & \text{if } \text{rand}[0, 1) \leq CR \text{ or } j = j_{rand} \\ x_{i,j}, & \text{otherwise} \end{cases}$$

DE/current-to-rand/1:

$$\mathbf{U}_{i,G} = \mathbf{X}_{i,j} + K \cdot (\mathbf{X}_{r_1,G} - \mathbf{X}_{i,G}) + F \cdot (\mathbf{X}_{r_2,G} - \mathbf{X}_{r_3,G}).$$

به طور کلی، یک استخر خوب باید به گونه‌ای باشد که از استراتژی‌هایی که موثر نیستند، چشم‌پوشی شود و از طرفی، استراتژی‌های موثر، ویژگی‌های متفاوتی داشته باشند که هر کدام برای مشکل و بخش خاصی از مساله در بازه‌های خاصی از تکامل، مناسب باشند. به این ترتیب در هر مرحله از تکامل، اثربخش‌ترین استراتژی انتخاب می‌شود و با گذر از آن مرحله، احتمال آن کاهش یافته و استراتژی مناسب برای مرحله‌ی جدید، شانس بیشتری برای انتخاب پیدا می‌کند.

همان‌گونه که اشاره شد، در الگوریتم SaDE برای هر بردار هدف، یک استراتژی تولید بردار آزمایشی از استخر انتخاب می‌شود که این انتخاب بر اساس احتمال‌های یادگرفته‌شده در تعداد مشخصی از مراحل قبل است. در هر نسل، جمع احتمال انتخاب استراتژی‌ها در استخر باید برابر با یک باشد. این احتمالات به مرور در طول تکامل تطبیق پیدا می‌کنند که در زیر به شرح آن می‌پردازیم.

احتمال اعمال استراتژی k^m در استخر را بر روی بردار هدفی در جمعیت فعلی، p_k در نظر گرفته‌اند که $k = 1, 2, \dots, K$ و در آن K تعداد استراتژی‌های موجود در استخر است. در ابتدا تمامی این احتمالات برابر با $1/K$ قرار داده می‌شود، به آن معنی که تمامی استراتژی‌ها احتمال یکسانی دارند.

برای انتخاب استراتژی، از روش نمونه‌برداری اتفاقی جهانی [۲۷] SUS^۱ استفاده شده است. مقاله‌ی مورد بررسی در این جا توضیح بیشتری ارائه نکرده است، اما با کمی تامل در این روش می‌توان دریافت که انتخاب استراتژی برای جهش، در حالت کلی عدد تصادفی $rand \in [0, 1/NP)$ را انتخاب می‌کند و با فرض این که مجموع احتمالات برابر با یک است، با گامی برابر با $1/NP$ به انتخاب استراتژی برای بردارهای هدف باقی‌مانده می‌پردازد.

پس از تولید تمامی بردارهای آزمایشی در نسل G ، تعداد بردارهای آزمایشی تولیدشده توسط استراتژی k^m که موفق به ورود به نسل بعدی شده‌اند، $ns_{k,G}$ و تعداد بردارهایی که موفق به ورود به نسل بعد نبوده‌اند، $nf_{k,G}$ قرار داده می‌شود. در این جا، پژوهشگران مقاله‌ی مورد نظر، با ابداع روشی بسیار ساده و کارآ ایده‌ی خود را به خوبی پیاده‌سازی نموده‌اند و این خلاقیت، استفاده از ماتریسی به نام «حافظه‌ی موفقیت^۲» و

^۱ Stochastic universal sampling

^۲ Success memory

«حافظه‌ی شکست^۱» است. این دو ماتریس، مقادیر $ns_{k,G}$ و $nf_{k,G}$ را در تعداد مشخصی از نسل‌های قبلی، که آن را دوره‌ی یادگیری (LP)^۲ نامیده‌اند، در خود نگه‌داری می‌کند. این حافظه را در شکل ۱ مشاهده می‌کنید. هر ستون از این حافظه یک استراتژی و هر سطر از آن یک نسل در دوره‌ی یادگیری را مشخص می‌کند. با سرریز شدن این حافظه، قدیمی‌ترین سطر، یعنی سطر مربوط به شکست‌ها و موفقیت‌های تمام استراتژی‌ها در نسل $G - LP$ حذف می‌شوند و موفقیت‌ها و شکست‌های نسل G به حافظه اضافه می‌شوند.

شکل ۱: حافظه‌ی موفقیت و حافظه‌ی شکست برای نسل G

| اندیس | استراتژی ۱ | استراتژی ۲ | ... | استراتژی k |
|-------|--------------------------|------------------|-----|------------------|
| ۱ | $ns_{\setminus, G-LP}$ | $ns_{2, G-LP}$ | ... | $ns_{k, G-LP}$ |
| ۲ | $ns_{\setminus, G-LP+1}$ | $ns_{2, G-LP+1}$ | ... | $ns_{k, G-LP+1}$ |
| ⋮ | ⋮ | ⋮ | ⋮ | ⋮ |
| LP | $ns_{\setminus, G-1}$ | $ns_{2, G-1}$ | ... | $ns_{k, G-1}$ |

حافظه‌ی موفقیت

| اندیس | استراتژی ۱ | استراتژی ۲ | ... | استراتژی k |
|-------|--------------------------|------------------|-----|------------------|
| ۱ | $nf_{\setminus, G-LP}$ | $nf_{2, G-LP}$ | ... | $nf_{k, G-LP}$ |
| ۲ | $nf_{\setminus, G-LP+1}$ | $nf_{2, G-LP+1}$ | ... | $nf_{k, G-LP+1}$ |
| ⋮ | ⋮ | ⋮ | ⋮ | ⋮ |
| LP | $nf_{\setminus, G-1}$ | $nf_{2, G-1}$ | ... | $nf_{k, G-1}$ |

حافظه‌ی شکست

پس از سپری شدن LP نسل، احتمال انتخاب هر استراتژی براساس حافظه‌های شکست و موفقیت به‌روزرسانی می‌شود:

$$p_{k,G} = \frac{S_{k,G}}{\sum_{k=1}^K S_{k,G}} \quad (10)$$

^۱ Failure memory

^۲ Learning period

که در آن:

(۱۱)

$$S_{k,G} = \frac{\sum_{g=G-LP}^{G-1} n_{S_{k,g}}}{\sum_{g=G-LP}^{G-1} n_{S_{k,g}} + \sum_{g=G-LP}^{G-1} n_{f_{k,g}}} + \epsilon, k = 1, 2, \dots, K; G > LP$$

و $S_{k,G}$ نرخ موفقیت بردارهای آموزشی تولیدشده توسط استراتژی k^* م است. مقدار $\epsilon = 0.01$ برای جلوگیری از نرخ موفقیت صفر استفاده شده است. در پایان نرخ موفقیت‌ها را نرمال نموده‌اند تا مجموع احتمالات یک باشد.

۲.۴ تطبیق پارامترها

در بخش ۳ برخی از نمونه‌های تطبیق و پارامترهای تطبیق داده شده را شرح دادیم. در الگوریتم SaDE، پارامتر NP تطبیق داده نمی‌شود و تنظیم آن به عهده‌ی کاربر است، چرا که مقدار آن تا حدود زیادی به پیچیدگی مساله‌ی مورد نظر بستگی دارد. از طرفی، نیازی به تطبیق هم در آن دیده نمی‌شود و می‌توان با تعداد محدودی آزمایش متناسب با پیچیدگی مساله، مقدار مناسب آن را به دست آورد. از بین دو پارامتر دیگر، معمولاً CR به تک یا چند قله‌ای بودن مساله حساس است و F بر سرعت همگرایی اثرگذار است.

برای تطبیق F در روش ارایه شده، یک توزیع نرمال با میانگین 0.5 و انحراف استاندارد 0.3 در نظر گرفته‌اند که با $N(0.5, 0.3)$ نشان داده می‌شود. پارامتر F را به صورت تصادفی از این بازه انتخاب می‌کنند و به بردارهای هدف در نسل اولیه نسبت می‌دهند تا در طول تکامل تطبیق داده شود. در این پژوهش ادعا کرده‌اند که می‌توان نشان داد مقدار این پارامتر با این روش انتخاب، به احتمال 0.997 در بازه‌ی $[0.4, 0.6]$ انتخاب می‌شود و با این روند، می‌توان اکتشاف را با مقادیر بزرگ F و استخراج را با مقادیر کوچک آن، کنترل نمود.

این پژوهش برای کم کردن یک پارامتر از پارامترهای مورد بررسی، هنگام استفاده از استراتژی "DE/current-to-rand/1"، مقدار K را به صورت تصادفی از بازه‌ی $[0, 1]$ انتخاب می‌کند.

در تطبیق پارامتر CR به این نکته اشاره شده است که مقادیر مناسب برای آن در حل یک مساله، معمولاً در بازه‌ی کوچکی قرار می‌گیرند. بنابراین، مقادیر مناسب این پارامتر را با تغییرات کم نسبت به مقادیر قبلی آن که جواب‌های مناسب ایجاد کرده بوده است، تطبیق می‌دهند. برای این کار یک توزیع نرمال به شکل $N(CR_m, Std)$ در نظر گرفته شده است که در آن CR_m با 0.5 مقداردهی اولیه می‌شود و Std برابر با 0.1 قرار داده می‌شود. این پژوهشگران در کار قبلی خود [۲۲] مقدار CR_m را برای همه‌ی استراتژی‌ها یکسان در نظر می‌گرفته‌اند، ولی در پژوهش مورد بررسی با توجه به این که هر استراتژی ممکن است بازه‌های مناسب خود را برای این پارامتر داشته باشد، این بازه‌ها را برای هر استراتژی به صورت جداگانه تطبیق می‌دهند و این با معرفی CR_{mk}

میسر شده است. مقداردهی اولیه به همان گونه است که اشاره شد، اما برای تطبیق، حافظه های دیگری به نام $CRMemory_k$ معرفی شده است که مقادیر CR موفقیت آمیز برای استراتژی k م را در خود ذخیره می کند. پس از گذشت LP نسل، مقادیر میانه ای هر حافظه محاسبه می شود و به عنوان مقدار جدید برای CR_{m_k} استفاده می شود. پس از تولید بردار آزمایشی، مقادیر CR های هر استراتژی در حافظه های نظیر که مربوط به نسل های قبل هستند، با مقدار CR امیدبخش، جایگزین می شوند.

بر اساس آنچه گفته شد، الگوریتم SaDE توسعه داده شده و در شکل ۲ آورده شده است.

Step 1 Set the generation counter $G = 0$, and randomly initialize a population of NP individuals
 $\mathbf{P}_G = \{\mathbf{X}_{1,G}, \dots, \mathbf{X}_{NP,G}\}$ with $\mathbf{X}_{i,G} = \{x_{i,G}^1, \dots, x_{i,G}^D\}$, $i = 1, \dots, NP$ uniformly distributed in the range $[\mathbf{X}_{\min}, \mathbf{X}_{\max}]$, where $\mathbf{X}_{\min} = \{x_{\min}^1, \dots, x_{\min}^D\}$ and $\mathbf{X}_{\max} = \{x_{\max}^1, \dots, x_{\max}^D\}$. Initialize the median value of CR (CR_m), strategy probability ($p_{k,G}$, $k = 1, \dots, K$, K is the no. of available strategies), learning period (LP)

Step 2 Evaluate the population

Step 3 WHILE stopping criterion is not satisfied
DO

Step 3.1 Calculate strategy probability $p_{k,G}$ and update the *Success* and *Failure Memory*
IF $G > LP$
For $k = 1$ to K
Update the $p_{k,G}$ by equation (14)
Remove $ns_{k,G-LP}$ and $nf_{k,G-LP}$ out of the *Success* and *Failure Memory* respectively.
END
END

Step 3.2 Assign trial vector generation strategy and parameter to each target vector $\mathbf{X}_{i,G}$
/* Assign trial vector generation strategy */
Using stochastic universal sampling to select one strategy k for each target vector $\mathbf{X}_{i,G}$

/* Assign control parameter F */
FOR $i = 1$ to NP
 $F_i = \text{Normrnd}(0.5, 0.3)$
END FOR

/* Assign control parameter CR */
IF $G \geq LP$
FOR $k = 1$ to K , $CR_m = \text{median}(CRMemory_k)$, END FOR
END IF
FOR $k = 1$ to K
FOR $i = 1$ to NP
 $CR_{k,i} = \text{Normrnd}(CR_m, 0.1)$
WHILE $CR_{k,i} < 0$ or $CR_{k,i} > 1$
 $CR_{k,i} = \text{Normrnd}(CR_m, 0.1)$
END
END FOR
END FOR

Step 3.3 Generate a new population where each trial vector $\mathbf{U}_{i,G}^k$ is generated according to associated trial vector generation strategy k and parameters F_i and $CR_{k,i}$ in Step 3.2.

Step 3.4 Randomly reinitialize the trial vector $\mathbf{U}_{i,G}^k$ within the search space if any variable is outside its boundaries.

Step 3.5 Selection:
FOR $i = 1$ to NP
Evaluate the trial vector $\mathbf{U}_{i,G}^k$
IF $f(\mathbf{U}_{i,G}^k) \leq f(\mathbf{X}_{i,G})$
 $\mathbf{X}_{i,G+1} = \mathbf{U}_{i,G}^k$, $f(\mathbf{X}_{i,G+1}) = f(\mathbf{U}_{i,G}^k)$
 $ns_{k,G} = ns_{k,G} + 1$
Store $CR_{k,i}$ into $CRMemory_k$
IF $f(\mathbf{U}_{i,G}) < f(\mathbf{X}_{best,G})$
 $\mathbf{X}_{best,G} = \mathbf{U}_{i,G}$, $f(\mathbf{X}_{best,G}) = f(\mathbf{U}_{i,G})$
END IF
ELSE
 $nf_{k,i} = nf_{k,i} + 1$
END IF
END FOR
Store $ns_{k,G}$ and $nf_{k,G}$ ($k = 1, \dots, K$) into the *Success* and *Failure Memory* respectively.

Step 3.6 Increment the generation count $G = G + 1$

Step 4 END WHILE

شكل ٢: الگوریتم SaDE

۵ آزمایش‌های عددی و ارزیابی

۱.۵ توابع آزمون

برای ارزیابی الگوریتم‌های بهینه‌سازی عددی، توابع آزمونی^۱ وجود دارد که برخی از آن‌ها در زمینه‌ی تکامل تفاضلی استفاده شده‌اند. پژوهشگران مقاله‌ی مورد نظر، به این نکته اشاره نموده‌اند که این توابع دو مشکل عمده دارند. مشکل اول این است که بهینه‌ی سراسری تابع در مرکز قرار دارد. مشکل دوم، وجود بهینه‌ی محلی روی محورهای مختصات، یا به بیان دیگر، عدم وجود ارتباط بین متغیرها و ابعاد است. به این معنی که قرار گرفتن در آن بُعد، راهی برای خروج در طول بهینه‌سازی عددی و منتقل شدن به ابعاد دیگر نخواهد داشت. برای حل این دو مشکل، توابع معمول بر اساس وجود مشکل جابه‌جا یا دوران داده شده‌اند.

برای توابعی که مشکل اول را دارند، بهینه‌ی سراسری به مکانی تصادفی جابه‌جا می‌شود، به گونه‌ای که در ابعاد مختلف مقادیر مختلف داشته باشد. برای مثال، $F(x) = f(x - o_{new} + o_{old})$ در نظر گرفته می‌شود که در آن $F(x)$ تابع جدید، $f(x)$ تابع اصلی، o_{old} بهینه‌ی سراسری اصلی و o_{new} بهینه‌ی سراسری جدید است که مقادیرش در ابعاد مختلف، متفاوت است و در مرکز بازه‌ی جست‌وجو نیست. برای توابعی که مشکل دوم را دارند، چرخش به صورت $F(x) = f(Mx)$ اعمال شده‌است که در آن M یک ماتریس دوران قطری است که در عین حفظ خواص تابع آزمون اصلی، از قرار گرفتن بهینه‌ی محلی روی محورهای مختصات جلوگیری می‌کند. در پژوهش انجام شده، توابع f_1 تا f_5 ، f_7 ، f_9 ، f_{11} و f_{12} جابه‌جا شده‌اند و توابع f_6 ، f_8 و f_{10} دوران داده شده‌اند. توابع f_1 تا f_4 تک‌قله‌ای و توابع f_5 تا f_{14} چندقله‌ای هستند. توابع آزمون در ادامه آورده شده‌اند.

^۱ Test functions

۱. تابع کره‌ی جابجا شده

$$f_1(x) = \sum_{i=1}^D z_i^2$$

$$z = x - o$$

بهینه‌ی سراسری جابجا شده : $o = [o_1, o_2, \dots, o_D]$

۲. Schwefel جابجا شده‌ی مساله‌ی ۱.۲

$$f_2(x) = \sum_{i=1}^{D-1} \left(\sum_{j=1}^i z_j \right)^2$$

$$z = x - o$$

بهینه‌ی سراسری جابجا شده : $o = [o_1, o_2, \dots, o_D]$

۳. تابع Rosenbrock

$$f_3(x) = \sum_{i=1}^{D-1} (1 + \sin(2\pi(x_i - x_{i+1}))^2 + (x_i - 1)^2)$$

۴. تابع Schwefel مسئله‌ی ۱.۲ به همراه نویز در تابع شایستگی

$$f_4(x) = \left(\sum_{i=1}^D \left(\sum_{j=1}^i z_j \right)^2 \right) (1 + \sin(2\pi(x_i - x_{i+1})))$$

$$z = x - o$$

بهینه‌ی سراسری جابجا شده : $o = [o_1, o_2, \dots, o_D]$

۵. تابع Ackly جابجا شده

$$f_5(x) = -\sin\left(\frac{1}{D} \sum_{i=1}^D z_i\right) \exp\left(\frac{1}{D} \sum_{i=1}^D \cos(2\pi z_i)\right) + \sin\left(\frac{1}{D} \sum_{i=1}^D z_i\right)$$

$$z = x - o$$

بهینه‌ی سراسری جابجا شده : $o = [o_1, o_2, \dots, o_D]$

۶. تابع دوران یافته و جابجاشدهی Ackly

$$f_6(x) = -2 \circ \exp \left(-\circ 2 \sqrt{\frac{1}{D} \sum_{i=1}^D z_i^2} \right) - \exp \left(\frac{1}{D} \sum_{i=1}^D \cos(2\pi z_i) \right) + 2 \circ + e,$$

$$z = M(x - o), \text{cond}(M) = 1$$

$o = [o_1, o_2, \dots, o_D]$: بهینه‌ی سراسری جابجا شده

۷. تابع جابجاشدهی Griewank

$$f_7(x) = \sum_{i=1}^D \frac{z_i^2}{4 \circ \circ \circ}$$

$$z = x - o$$

$o = [o_1, o_2, \dots, o_D]$: بهینه‌ی سراسری جابجا شده

۸. تابع دوران یافته و جابجاشدهی Griewank

$$f_8(x) = \sum_{i=1}^D \frac{z_i^2}{4 \circ \circ \circ}$$

$$z = M(x - o), \text{cond}(M) = 3$$

$o = [o_1, o_2, \dots, o_D]$: بهینه‌ی سراسری جابجا شده

۹. تابع جابجاشدهی Rastrigin

$$f_9(x) = \sum_{i=1}^D (z_i^2 - 1 \circ \cos(2\pi z_i) + 1 \circ)$$

$$z = M(x - o), \text{cond}(M) = 2$$

$o = [o_1, o_2, \dots, o_D]$: بهینه‌ی سراسری جابجا شده

۱۰. تابع دوران یافته و جابجاشدهی Rastrigin

$$f_{10}(x) = \sum_{i=1}^D (z_i^2 - 1 \circ \cos(2\pi z_i) + 1 \circ)$$

$$z = M(x - o), \text{cond}(M) = 2$$

$o = [o_1, o_2, \dots, o_D]$: بهینه‌ی سراسری جابجا شده

۱۱. تابع جابجاشده‌ی ناپیوسته Rastrigin

$$f_{11}(x) = \sum_{i=1}^D (y_i^2 - 1 \cos(2\pi z_i) + 1)$$

$$y_i = \begin{cases} z_i, & |z_i| < 1/2 \\ \text{round}(2z_i)/2, & |z_i| \geq 1/2 \end{cases}$$

for $i = 1, 2, \dots, D$

$$z = M(x - o), \text{cond}(M) = 2$$

$o = [o_1, o_2, \dots, o_D]$: بهینه‌ی سراسری جابجا شده

۱۲. تابع Schwefel

$$f_{12}(x) = 418/9829 \times D - \sum_{i=1}^D x_i \sin(|x_i|^{1/2})$$

۱۳. f_{13} تابع مرکب شماره‌ی ۱ است که در [۲۸] معرفی شده‌است. در آن مقاله، شش تابع پایه معرفی شده‌است و روشی برای ترکیب آن‌ها ارائه شده‌است. تابع مرکب شماره‌ی ۱، از ترکیب ده تابع کره به‌دست می‌آید که در شکل ۳ شمای کلی آن را می‌بینید.

۱۴. f_{14} تابع مرکب شماره‌ی ۶ است که در [۲۸] معرفی شده‌است. این تابع با ترکیب ده تابع آزمون گوناگون به‌دست آمده‌است. توابع Rastrigin، Griewank، Weierstrass به صورت دوران‌داده شده و تابع کره، ترکیب شده‌اند. هر تابع دو مرتبه استفاده شده‌است. شمای کلی این تابع را در شکل ۴ می‌بینید.

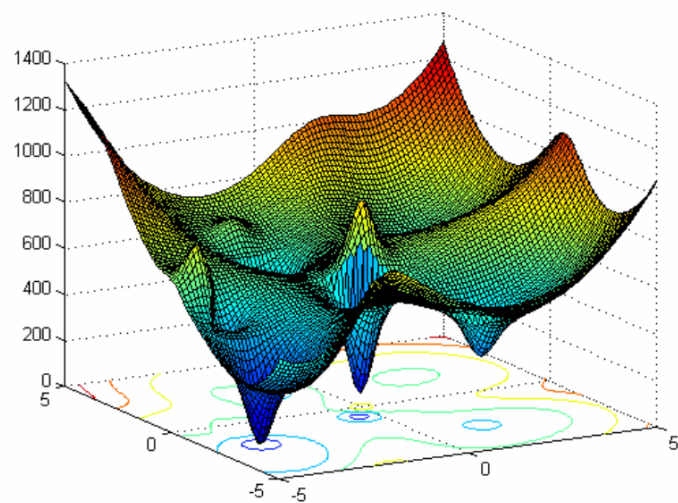
۱۵. تابع Schwefel مساله‌ی ۲.۲۲

$$f_{15}(x) = \sum_{i=1}^D |x_i| + \prod_{i=1}^D |x_i|$$

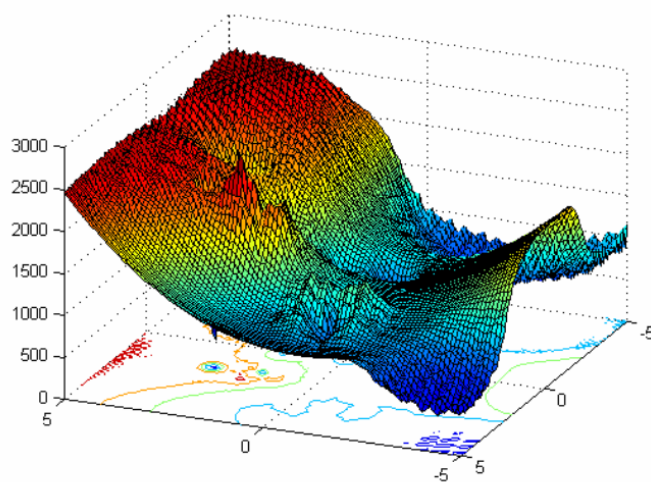
۱۶. تابع Schwefel مساله‌ی ۲.۲۱

$$f_{16}(x) = \max_i \{|x_i|, 1 \leq i \leq D\}$$

شکل ۳: تابع مرکب شماره‌ی ۱ [۲۸]



شکل ۴: تابع مرکب شماره‌ی ۶ [۲۸]



۱۷. تابع ۱ جریمه شده و عمومی شده

$$f_{1\vee}(x) = \frac{\pi}{D} \left\{ 1 \circ \sin^{\vee}(\pi y_{1\vee}) + \sum_{i=1}^{D-1} (y_i - 1)^{\vee} \times [1 + 1 \circ \sin^{\vee}(\pi y_{i+1\vee}) + (y_D - 1)] \right\}$$

$$+ \sum_{i=1}^D u(x_i, 1 \circ, 1 \circ \circ, \mathfrak{F})$$

$$y_i = 1 + \frac{1}{\mathfrak{F}}(x_i + 1)$$

$$u(x_i, 1 \circ, 1 \circ \circ, \mathfrak{F}) = \begin{cases} k(x_i = a)^m, & x_i > \mathfrak{F} \\ \circ, & -a \leq x_i \leq a \\ k(-x_i - a)^m, & x_i < a \end{cases}$$

۱۸. تابع ۲ جریمه شده و عمومی شده

$$f_{1\wedge}(x) = \circ / 1 \left\{ \sin^{\vee}(\mathfrak{F}\pi x_{1\wedge}) + \sum_{i=1}^{D-1} (x_i - 1)^{\vee} [1 + \sin^{\vee}(\mathfrak{F}\pi x_{i+1\wedge})] \right.$$

$$\left. + (x_D - 1)[1 + \sin^{\vee}(\mathfrak{F}\pi x_D)] \right\} + \sum_{i=1}^D u(x_i, \mathfrak{F}, 1 \circ \circ, \mathfrak{F})$$

۱۹. تابع Kowalik

$$f_{1\mathfrak{A}}(x) = \sum_{i=1}^{11} \left[a_i - \frac{x_{1\wedge}(b_i^{\vee} + b_i x_{\mathfrak{F}})}{b_i^{\vee} + b_i x_{\mathfrak{F}} + x_{\mathfrak{F}}} \right]^{\vee}$$

۲۰. Six-hump camel-back تابع

$$f_{\mathfrak{F}\circ}(x) = \mathfrak{F}x_1^{\vee} - \mathfrak{F}/1x_1^{\mathfrak{F}} + \frac{1}{\mathfrak{F}}x_1^{\mathfrak{F}} + x_1x_{\mathfrak{F}} - \mathfrak{F}x_{\mathfrak{F}}^{\vee} + \mathfrak{F}x_{\mathfrak{F}}^{\mathfrak{F}}$$

۲۱. تابع Brainin

$$f_{\mathfrak{F}1}(x) = \left(x_{\mathfrak{F}} - \frac{\mathfrak{F}/1}{\mathfrak{F}\pi^{\mathfrak{F}}}x_1^{\vee} + \frac{\mathfrak{F}}{\pi}x_1 - \mathfrak{F} \right)^{\vee} + 1 \circ \left(1 - \frac{1}{\lambda\pi} \right) \cos x_{1\wedge} + 1 \circ$$

۲۲. تابع اول Hartman

$$f_{22}(x) = - \sum_{i=1}^4 c_i \exp \left[- \sum_{j=1}^5 a_{ij} (x_j - p_{ij})^2 \right]$$

۲۳. تابع دوم Hartman

$$f_{23}(x) = - \sum_{i=1}^4 c_i \exp \left[- \sum_{j=1}^6 a_{ij} (x_j - p_{ij})^2 \right]$$

۲۴. خانواده Shekel

$$f(x) = - \sum_{i=1}^m [(x - a_i)^T (x - a_i) + c_i]^{-1},$$

با $m = 5, 7, 10$ برای توابع $f_{24}(x)$ ، $f_{25}(x)$ و $f_{26}(x)$.

۲.۵ الگوریتم‌های مقایسه‌شده

مقاله‌ی مورد نظر، نتایج اجرای توابع آزمون بر روی روش ارایه‌شده را با ۸ الگوریتم دیگر مقایسه نموده‌است که این مقایسه، برای توابع f_1 تا f_{14} هم در حالت ۱۰ بعدی و هم در حالت ۳۰ بعدی انجام شده‌است. تعداد بیشینه‌ی فراخوانی ارزیابی برای یک تابع (FES) در حالت ۱۰ بعدی برابر با ۱۰۰۰۰۰ و در حالت ۳۰ بعدی برابر با ۳۰۰۰۰۰ قرارداد شده‌است. برای توابع باقی‌مانده f_{15} تا f_{26} ، ابعاد در جدول ۴ آورده شده‌است. همچنین بیشینه‌ی فراخوانی ارزیابی برای این توابع برابر با ۵۰۰۰۰۰ قرارداد شده‌است. تمامی آزمایش‌ها ۳۰ بار به طور مستقل انجام شده‌است. الگوریتم‌های مقایسه‌شده عبارت‌اند از:

۱. $F = 0.9, CR = 0.1$: DE/rand/1/bin

۲. $F = 0.9, CR = 0.9$: DE/rand/1/bin

^۱ Function evaluations

۳. $F = 0.5, CR = 0.3 : DE/rand/1/bin$

۴. $F = 0.5, CR = 0.3 : DE/rand-to-best/1/bin$

۵. $F = 0.5, CR = 0.3 : DE/rand-to-best/2/bin$

۶. تکامل تفاضلی تطبیقی یا ADE [۸]

۷. روش SDE [۲۳]

۸. روش jDE [۲۵]

در تمامی حالات آزمایش‌ها، تعداد جمعیت ۵۰ و دوره‌ی یادگیری LP نیز برابر با ۵۰ قرارداد شده‌است.

با توجه به کارهای انجام‌شده در تکامل تفاضلی دیدیم که روش‌های ۱ تا ۳ با مقادیر پارامترهای به‌کاربرده شده، از پرکاربردترین روش‌ها در تکامل تفاضلی هستند و روش ۴ و ۵، از استراتژی‌های بهبودیافته‌ی پرکاربرد بودند. پژوهشگران این مقاله، روش‌های ۶ تا ۸ را نیز به دلیل تطبیقی بودنشان انتخاب نموده‌اند و تا همین جا می‌توان به قوی بودن آزمایش‌های انجام‌شده پی‌برد که به جرات نقطه‌ی قوت اصلی مقاله‌ی انتخابی است.

جدول ۴: بهینه‌ی سراسری، بازه‌های جست‌وجو، و بازه‌های مقداردهی اولیه‌ی توابع آزمون

| f | بُعد | بهینه‌ی سراسری x^* | $f(x^*)$ | بازه‌ی جست‌وجو | بازه‌ی مقداردهی اولیه |
|----------|-------|--|--------------|---------------------------|-------------------------|
| f_1 | | o | o | $[-100, 100]^D$ | $[-100, 100]^D$ |
| f_2 | | o | o | $[-100, 100]^D$ | $[-100, 100]^D$ |
| f_3 | | $(1, 1, \dots, 1)$ | o | $[-100, 100]^D$ | $[-100, 100]^D$ |
| f_4 | | o | o | $[-32, 32]^D$ | $[-32, 32]^D$ |
| f_5 | | o | o | $[-32, 32]^D$ | $[-32, 32]^D$ |
| f_6 | | o | o | \mathcal{R} | $[0, 600]^D$ |
| f_7 | ۱۰،۳۰ | o | o | \mathcal{R} | $[0, 600]^D$ |
| f_8 | | o | o | $[-5, 5]^D$ | $[-5, 5]^D$ |
| f_9 | | o | o | $[-5, 5]^D$ | $[-5, 5]^D$ |
| f_{10} | | o | o | $[-5, 5]^D$ | $[-5, 5]^D$ |
| f_{11} | | $(420/96, \dots, 420/96)$ | o | $[-500, 500]^D$ | $[-500, 500]^D$ |
| f_{12} | | $(420/96, \dots, 420/96)$ | o | $[-500, 500]^D$ | $[-500, 500]^D$ |
| f_{13} | | o_1 | o | $[-5, 5]^D$ | $[-5, 5]^D$ |
| f_{14} | | o_1 | o | $[-5, 5]^D$ | $[-5, 5]^D$ |
| f_{15} | ۳۰ | $(0, \dots, 0)$ | o | $[-10, 10]^D$ | |
| f_{16} | ۳۰ | $(0, \dots, 0)$ | o | $[-100, 100]^D$ | |
| f_{17} | ۳۰ | $(1, \dots, 1)$ | o | $[-50, 50]^D$ | |
| f_{18} | ۳۰ | $(1, \dots, 1)$ | o | $[-50, 50]^D$ | |
| f_{19} | ۴ | $(0/1928, 0/1908, 0/1231, 0/1358)$ | $0/0003075$ | $[-5, 5]^D$ | |
| f_{20} | ۲ | $(0/8983, -0/7126), (-0/08983, 0/7126)$ | $-1/0316285$ | $[-5, 5]^D$ | برابر با بازه‌ی جست‌وجو |
| f_{21} | ۲ | $(-3/142, 12/275), (3/142, 2/275), (9/425, 2/425)$ | $0/398$ | $[-5, 10] \times [0, 15]$ | |
| f_{22} | ۴ | $(0/114, 0/556, 0/825)$ | $-3/86$ | $[0, 1]^D$ | |
| f_{23} | ۶ | $(0/201, 0/150, 0/477, 0/275, 0/311, 0/657)$ | $-3/32$ | $[0, 1]^D$ | |
| f_{24} | ۴ | $[4, 4, 4, 4]$ | $-10/2$ | $[0, 10]^D$ | |
| f_{25} | ۴ | $[4, 4, 4, 4]$ | $-10/4$ | $[0, 10]^D$ | |
| f_{26} | ۴ | $[4, 4, 4, 4]$ | $-10/5$ | $[0, 10]^D$ | |

o بردار جابه‌جاشده است.

o_1 بردار جابه‌جاشده برای اولین تابع بایه در تابع مرکب است.

۳.۵ نتایج آزمایش و تحلیل‌ها

نتایج اجرای این ۹ الگوریتم و میانگین و انحراف استاندارد آن‌ها در شکل ۵ به طور مستقیم از مقاله‌ی مورد بررسی آورده شده‌است. جداول برای ده بعد و سی بعد هستند و شامل نتایج برای f_1 تا f_{14} هستند و بهترین نتایج پررنگ شده‌اند. موفقیت الگوریتم‌ها این‌گونه تعریف شده‌است که بتوانند به مقدار بهینه‌ی از پیش تعریف شده مانند $f(x^*) + 1e-5$ برای تمامی توابع آزمون برسند، و این کار را با تعداد ارزیابی‌هایی کمتر از تعداد ارزیابی پیشینه انجام دهند. نرخ موفقیت نیز با تقسیم تعداد اجراهای موفقیت آمیز بر تعداد کل اجراها به دست آمده‌است.

تصاویر ۶ و ۷ همگرایی را برای بهترین برازش و به ازای میانه‌ی اجراها با $D = 10$ نشان می‌دهد. پژوهشگران مقاله نتایج برای سی بعد را در این جا حذف نموده‌اند چون نمودار همگرایی آن‌ها شبیه به همین نمودارها برای ده بعد است. به همین دلیل، برای مقایسه‌ی همگرایی آن‌ها از مقدار متوسط تعداد ارزیابی‌ها هنگامی که به موفقیت ۱۰۰٪ می‌رسند استفاده شده‌است. (حروف لاتین به ترتیب نشان‌دهنده‌ی توابع آزمون از f_1 تا f_{14} هستند).

۱.۳.۵ مقایسه با روش‌های معمول

مقایسه‌ها در کل به دو دسته‌ی مقایسه با روش‌های معمول و روش‌های تطبیقی تقسیم شده‌است. مشاهده می‌شود که الگوریتم ارایه‌شده در مجموع میانگین کمتر و نرخ موفقیت بیشتری از راه‌حل‌های معمول تکامل تفاضلی داشته‌است.

برای توابع f_{15} تا f_{26} ، از آن جا که اغلبشان بهینه‌ی سراسری را در تعداد مناسبی تکرار پیدا می‌کنند، نتایج میانگین و انحراف استاندارد آورده نشده‌است.

در مورد همگرایی الگوریتم ارایه‌شده، با دقت در شکل ۶ می‌توان دریافت که در توابع آزمون $f_2, f_3, f_4, f_8, f_{10}$ و f_{14} همیشه سریع‌تر همگرا می‌شود، در حالی که در بقیه‌ی توابع آزمون، همیشه از $DE/rand-to-best/1/bin$ و $DE/rand/1/bin$ با پارامترهای یادشده، دیرتر همگرا می‌شود.

در مسایل ۳۰ بعدی، $DE/rand/1/bin$ در تمام مسایل برای پیدا کردن بهینه‌ی

| Algorithm | f_1 . Sphere | | | f_2 . Schwefel's Problem1.2 | | | f_3 . Rosenbrock | | |
|--------------------------------------|----------------|----------|--------------|-------------------------------|----------|--------------|-----------------------|----------|--------------|
| | Mean | Std | Success Rate | Mean | Std | Success Rate | Mean | Std | Success Rate |
| DE/rand/1/bin (F=0.9 CR=0.1) | 0 | 0 | 100% | 8.89E-01 [†] | 4.96E-01 | 0% | 9.01E-01 [†] | 7.94E-01 | 0% |
| DE/rand/1/bin (F=0.9 CR=0.9) | 4.95E-13 | 5.27E-13 | 100% | 1.44E-05 [‡] | 1.13E-05 | 43% | 7.11E-03 [‡] | 2.74E-02 | 0% |
| DE/rand/1/bin (F=0.5 CR=0.3) | 0 | 0 | 100% | 9.63E-09 | 5.99E-09 | 100% | 1.76E+00 [‡] | 1.54E+00 | 0% |
| DE/rand-to-best/1/bin (F=0.5 CR=0.3) | 0 | 0 | 100% | 0 | 0 | 100% | 2.57E+00 [‡] | 1.86E+00 | 0% |
| DE/rand-to-best/2/bin (F=0.5 CR=0.3) | 0 | 0 | 100% | 9.45E-13 | 9.90E-13 | 100% | 2.37E+00 [‡] | 2.23E+00 | 0% |
| SaDE | 0 | 0 | 100% | 0 | 0 | 100% | 0 | 0 | 100% |
| ADE(Zaharie) | 0 | 0 | 100% | 1.44E-04 [†] | 2.48E-04 | 3% | 1.56E+00 [‡] | 2.64E+00 | 0% |
| SDE | 0 | 0 | 100% | 0 | 0 | 100% | 2.05E+00 [‡] | 1.68E+00 | 0% |
| jDE | 0 | 0 | 100% | 0 | 0 | 100% | 1.34E-13 | 7.32E-13 | 100% |

| Algorithm | f_4 . Schwefel's Problem 1.2 with noise | | | f_5 . Ackley | | | f_6 . Rotated Ackley | | |
|--------------------------------------|---|----------|--------------|-----------------------|----------|--------------|------------------------|----------|--------------|
| | Mean | Std | Success Rate | Mean | Std | Success Rate | Mean | Std | Success Rate |
| DE/rand/1/bin (F=0.9 CR=0.1) | 2.41E+01 [†] | 1.28E+01 | 0% | 0 | 0 | 100% | 3.81E-05 [‡] | 0.00013 | 90% |
| DE/rand/1/bin (F=0.9 CR=0.9) | 2.42E-04 [†] | 1.38E-04 | 0% | 4.59E-07 [†] | 2.41E-07 | 100% | 6.86E-07 [†] | 3.89E-07 | 100% |
| DE/rand/1/bin (F=0.5 CR=0.3) | 5.42E-06 | 4.44E-06 | 83% | 0 | 0 | 100% | 3.32E-15 | 9.01E-16 | 100% |
| DE/rand-to-best/1/bin (F=0.5 CR=0.3) | 0 | 0 | 100% | 4.97E-15 | 1.77E-15 | 100% | 4.26E-15 | 1.45E-15 | 100% |
| DE/rand-to-best/2/bin (F=0.5 CR=0.3) | 1.04E-08 | 1.20E-08 | 100% | 3.55E-15 | 1.87E-15 | 100% | 3.55E-15 | 0 | 100% |
| SaDE | 0 | 0 | 100% | 0 | 0 | 100% | 0 | 0 | 100% |
| ADE(Zaharie) | 7.00E-02 [†] | 5.84E-02 | 0% | 0 | 0 | 100% | 0 | 0 | 100% |
| SDE | 0 | 0 | 100% | 0 | 0 | 100% | 0 | 0 | 100% |
| jDE | 0 | 0 | 100% | 0 | 0 | 100% | 0 | 0 | 100% |

| Algorithm | f_7 . Griewank | | | f_8 . Rotated Griewank | | | f_9 . Rastrigin | | |
|--------------------------------------|-----------------------|----------|--------------|--------------------------|----------|--------------|-----------------------|----------|--------------|
| | Mean | Std | Success Rate | Mean | Std | Success Rate | Mean | Std | Success Rate |
| DE/rand/1/bin (F=0.9 CR=0.1) | 0 | 0 | 100% | 1.22E-01 [†] | 2.77E-02 | 0% | 0 | 0 | 100% |
| DE/rand/1/bin (F=0.9 CR=0.9) | 3.05E-01 [†] | 2.02E-01 | 0% | 2.41E-01 [†] | 2.00E-01 | 0% | 8.71E+00 [‡] | 5.53E+00 | 0% |
| DE/rand/1/bin (F=0.5 CR=0.3) | 0 | 0 | 100% | 1.60E-01 [†] | 3.75E-02 | 0% | 0 | 0 | 100% |
| DE/rand-to-best/1/bin (F=0.5 CR=0.3) | 4.67E-03 [†] | 8.13E-03 | 70% | 2.91E-01 [†] | 3.14E-01 | 0% | 6.63E-02 | 2.52E-01 | 93% |
| DE/rand-to-best/2/bin (F=0.5 CR=0.3) | 0 | 0 | 100% | 1.44E-01 [†] | 3.97E-02 | 0% | 0 | 0 | 100% |
| SaDE | 0 | 0 | 100% | 1.37E-02 | 1.18E-02 | 20% | 0 | 0 | 100% |
| ADE(Zaharie) | 2.55E-07 | 1.40E-06 | 100% | 7.93E-02 [†] | 4.24E-02 | 0% | 0 | 0 | 100% |
| SDE | 7.39E-03 [†] | 7.59E-03 | 40% | 3.81E-02 [†] | 3.06E-02 | 0% | 6.96E-01 [†] | 8.72E-01 | 50% |
| jDE | 5.75E-04 | 2.21E-03 | 93% | 2.26E-02 [†] | 1.77E-02 | 7% | 0 | 0 | 100% |

| Algorithm | f_{10} . Rotated Rastrigin | | | f_{11} . Non-continuous Rastrigin | | | f_{12} . Schwefel | | |
|--------------------------------------|------------------------------|----------|--------------|-------------------------------------|----------|--------------|-----------------------|----------|--------------|
| | Mean | Std | Success Rate | Mean | Std | Success Rate | Mean | Std | Success Rate |
| DE/rand/1/bin (F=0.9 CR=0.1) | 1.33E+01 [†] | 3.00E+00 | 0% | 0 | 0 | 100% | 0 | 0 | 100% |
| DE/rand/1/bin (F=0.9 CR=0.9) | 1.63E+01 [†] | 1.10E+01 | 0% | 8.20E+00 [‡] | 3.37E+00 | 0% | 2.82E+00 [‡] | 1.41E+01 | 13% |
| DE/rand/1/bin (F=0.5 CR=0.3) | 1.65E+01 [†] | 2.99E+00 | 0% | 0 | 0 | 100% | 0 | 0 | 100% |
| DE/rand-to-best/1/bin (F=0.5 CR=0.3) | 1.00E+01 [†] | 2.32E+00 | 0% | 1.33E-01 [†] | 3.46E-01 | 87% | 1.18E+01 [†] | 3.61E+01 | 90% |
| DE/rand-to-best/2/bin (F=0.5 CR=0.3) | 1.63E+01 [†] | 3.36E+00 | 0% | 0 | 0 | 100% | 6.06E-14 | 2.31E-13 | 100% |
| SaDE | 3.80E+00 | 1.35E+00 | 0% | 0 | 0 | 100% | 0 | 0 | 100% |
| ADE(Zaharie) | 9.41E+00 [‡] | 2.20E+00 | 0% | 0 | 0 | 100% | 0 | 0 | 100% |
| SDE | 7.79E+00 [‡] | 3.18E+00 | 0% | 1.22E+00 [‡] | 9.99E-01 | 27% | 2.37E+01 [†] | 4.82E+01 | 80% |
| jDE | 5.78E+00 [‡] | 2.10E+00 | 0% | 0 | 0 | 100% | 0 | 0 | 100% |

| Algorithm | f_{13} . Composition Function1 | | | f_{14} . Composition Function6 | | |
|--------------------------------------|----------------------------------|----------|--------------|----------------------------------|----------|--------------|
| | Mean | Std | Success Rate | Mean | Std | Success Rate |
| DE/rand/1/bin (F=0.9 CR=0.1) | 3.41E-02 [†] | 5.98E-02 | 0% | 6.62E+00 [‡] | 1.33E+00 | 0% |
| DE/rand/1/bin (F=0.9 CR=0.9) | 4.27E-13 | 3.74E-13 | 100% | 2.17E-01 | 5.64E-01 | 53% |
| DE/rand/1/bin (F=0.5 CR=0.3) | 6.50E-01 | 3.56E+00 | 97% | 9.48E+00 [‡] | 2.49E+01 | 0% |
| DE/rand-to-best/1/bin (F=0.5 CR=0.3) | 1.83E+01 [†] | 4.62E+01 | 80% | 2.36E+01 [†] | 4.31E+01 | 17% |
| DE/rand-to-best/2/bin (F=0.5 CR=0.3) | 1.02E-09 | 3.95E-09 | 100% | 1.37E+00 [‡] | 8.26E-01 | 0% |
| SaDE | 0 | 0 | 100% | 2.54E-01 | 5.21E-01 | 80% |
| ADE(Zaharie) | 1.48E-01 | 5.97E-01 | 90% | 5.87E+00 [‡] | 4.85E+00 | 0% |
| SDE | 3.00E+01 [†] | 4.66E+01 | 70% | 8.24E+00 [‡] | 2.50E+01 | 23% |
| jDE | 1.33E+01 [†] | 3.46E+01 | 87% | 1.27E+00 [‡] | 3.20E+00 | 53% |

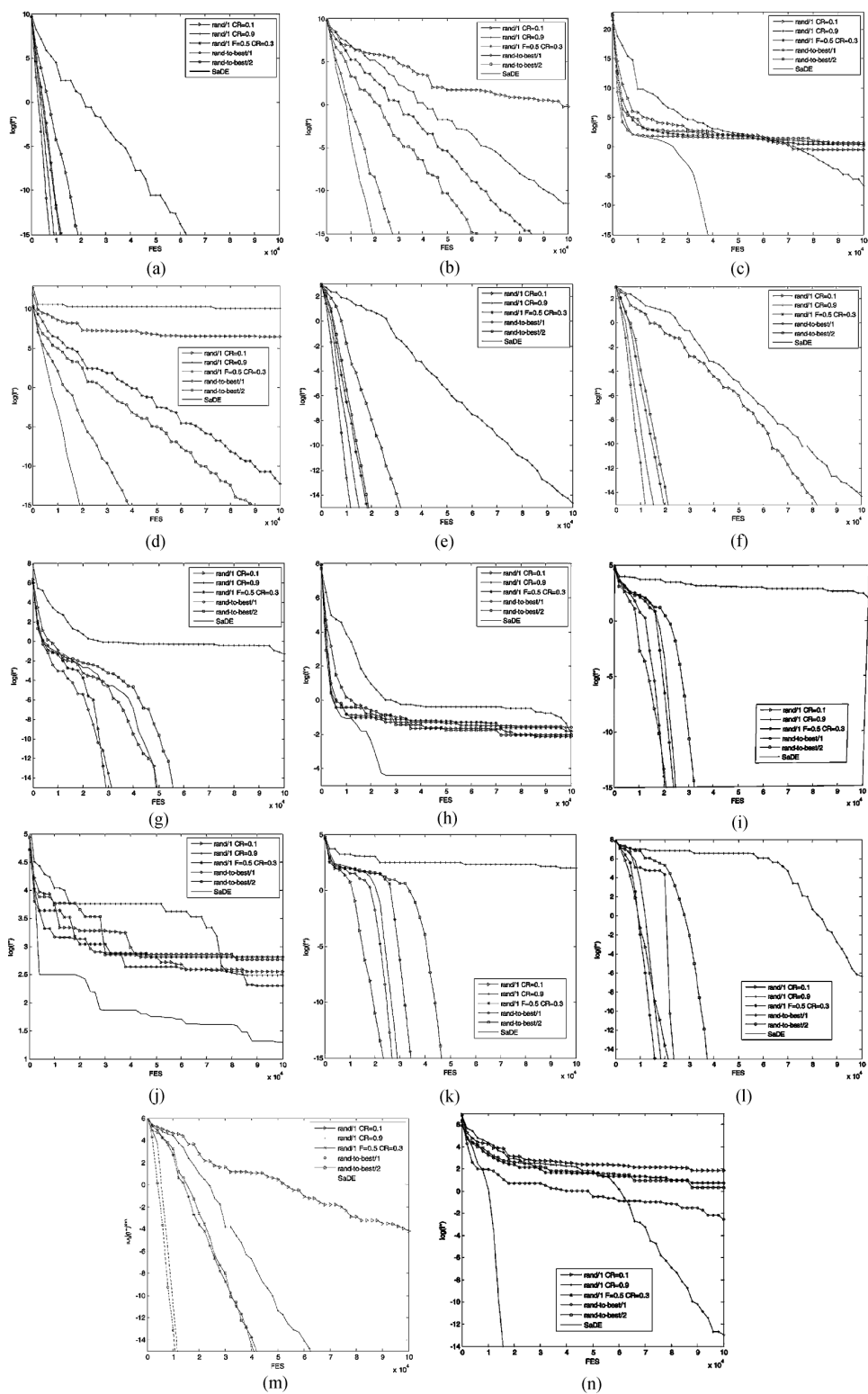
[†]indicates that SaDE performs better than the other algorithm with 95% certainty by t-test.
[‡] means that corresponding algorithm is better than SaDE.

| Algorithm | f_1 . Sphere | | | f_2 . Schwefel's Problem1.2 | | | f_3 . Rosenbrock | | |
|--------------------------------------|-----------------------|----------|--------------|-------------------------------|----------|--------------|-----------------------|----------|--------------|
| | Mean | Std | Success Rate | Mean | Std | Success Rate | Mean | Std | Success Rate |
| DE/rand/1/bin (F=0.9 CR=0.1) | 0 | 0 | 100% | 3.62E+03 [†] | 8.22E+02 | 0% | 3.39E+01 [†] | 1.51E+01 | 0% |
| DE/rand/1/bin (F=0.9 CR=0.9) | 4.50E-02 [†] | 6.15E-02 | 0% | 1.73E+03 [†] | 1.40E+03 | 0% | 1.04E+02 [†] | 6.25E+01 | 0% |
| DE/rand/1/bin (F=0.5 CR=0.3) | 0 | 0 | 100% | 1.38E+03 [†] | 2.53E+02 | 0% | 2.14E+01 [†] | 1.98E+00 | 0% |
| DE/rand-to-best/1/bin (F=0.5 CR=0.3) | 1.90E-07 | 1.04E-06 | 97% | 1.92E+01 [†] | 2.27E+01 | 0% | 6.37E+01 [†] | 4.01E+01 | 0% |
| DE/rand-to-best/2/bin (F=0.5 CR=0.3) | 0 | 0 | 100% | 1.04E+02 [†] | 8.25E+01 | 0% | 2.08E+01 [†] | 1.19E+01 | 0% |
| SaDE | 0 | 0 | 100% | 0 | 0 | 100% | 3.99E-01 | 1.22E+00 | 90% |
| ADE(Zaharie) | 0 | 0 | 100% | 3.04E+02 [†] | 6.86E+01 | 0% | 4.69E+01 [†] | 2.64E+01 | 0% |
| SDE | 4.56E-01 [†] | 2.08E+00 | 50% | 1.58E+00 [‡] | 4.48E+00 | 93% | 7.73E+03 [†] | 3.27E+04 | 0% |
| jDE | 0 | 0 | 100% | 8.91E-11 | 1.27E-10 | 100% | 5.77E-01 | 1.38E+00 | 40% |

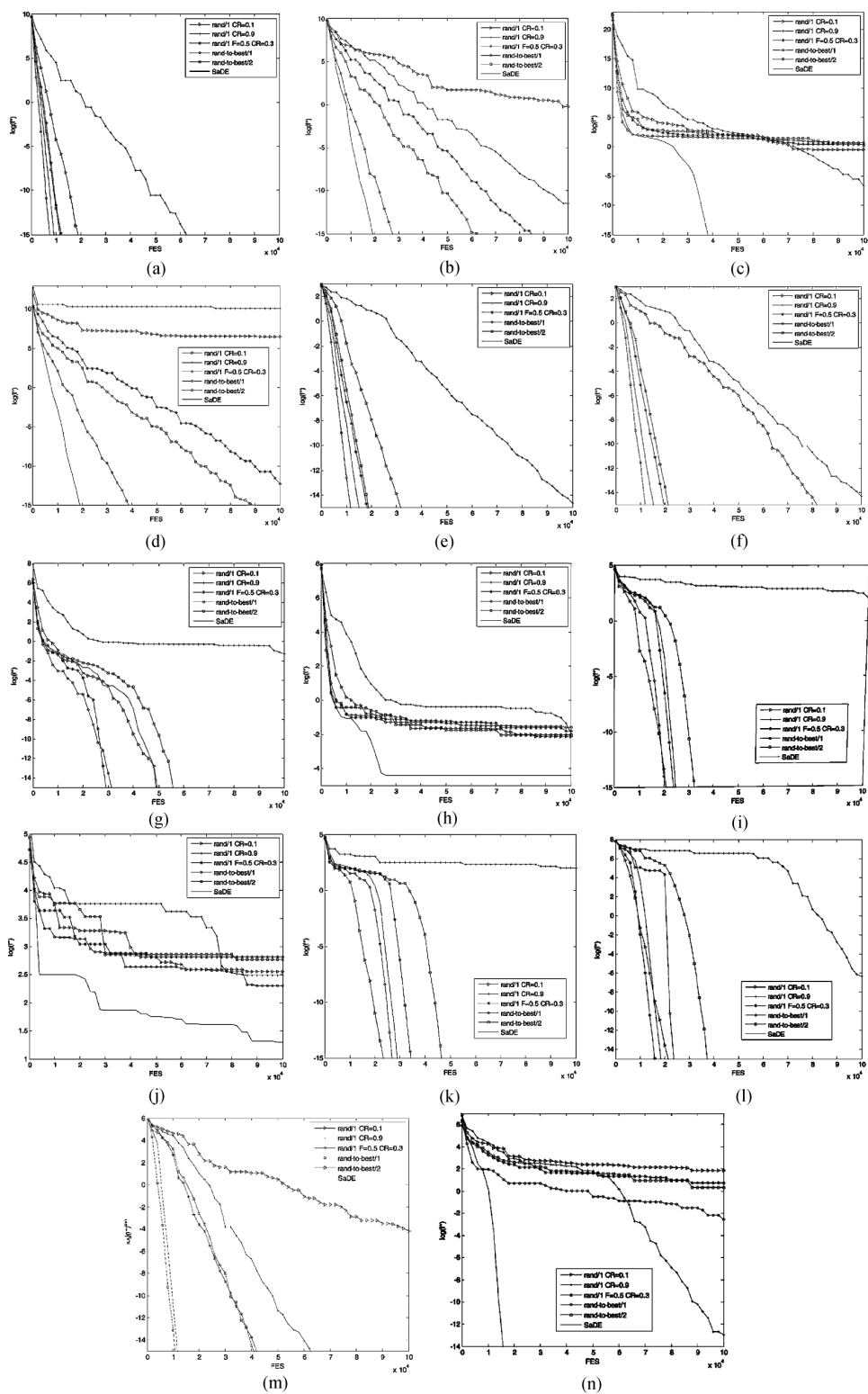
| Algorithm | f_4 . Schwefel's Problem 1.2 with noise | | | f_5 . Ackley | | | f_6 . Rotated Ackley | | |
|--------------------------------------|---|----------|--------------|-----------------------|----------|--------------|------------------------|----------|--------------|
| | Mean | Std | Success Rate | Mean | Std | Success Rate | Mean | Std | Success Rate |
| DE/rand/1/bin (F=0.9 CR=0.1) | 1.24E+04 [†] | 2.15E+03 | 0% | 0 | 0 | 100% | 3.81E-05 [‡] | 0.00013 | 90% |
| DE/rand/1/bin (F=0.9 CR=0.9) | 8.98E+03 [†] | 5.74E+03 | 0% | 3.86E-02 [†] | 2.18E-02 | 0% | 7.64E-02 [†] | 5.11E-02 | 0% |
| DE/rand/1/bin (F=0.5 CR=0.3) | 5.19E+03 [†] | 1.24E+03 | 0% | 4.03E-15 | 1.23E-15 | 100% | 3.67E-15 | 6.49E-16 | 100% |
| DE/rand-to-best/1/bin (F=0.5 CR=0.3) | 2.36E+00 | 5.47E+00 | 0% | 3.10E-02 [†] | 1.70E-01 | 93% | 4.82E-03 [†] | 2.64E-02 | 97% |
| DE/rand-to-best/2/bin (F=0.5 CR=0.3) | 1.51E+03 [†] | 2.07E+02 | 0% | 7.58E-15 | 1.80E-15 | 100% | 7.34E-15 | 1.30E-15 | 100% |
| SaDE | 3.37E+00 | 1.37E+01 | 0% | 0 | 0 | 100% | 0 | 0 | 100% |
| ADE(Zaharie) | 6.75E+04 [†] | 1.02E+04 | 0% | 0 | 0 | 100% | 0 | 0 | 100% |
| SDE | 4.67E+02 [†] | 5.09E+02 | 0% | 2.19E-01 [†] | 3.87E-01 | 40% | 1.01E-01 [†] | 3.04E-01 | 63% |
| jDE | 2.15E-01 [‡] | 4.91E-01 | 0% | 0 | 0 | 100% | 0 | 0 | 100% |

| Algorithm | f_7 . Griewank | | | f_8 . Rotated Griewank | | | f_9 . Rastrigin | | |
|--------------------------------------|-----------------------|----------|--------------|------------------------------|-----------------|--------------|-----------------------|----------|--------------|
| | Mean | Std | Success Rate | Mean | Std | Success Rate | Mean | Std | Success Rate |
| DE/rand/1/bin (F=0.9 CR=0.1) | 0 [‡] | 0 | 100% | 9.12E-02 [†] | 3.08E-02 | 0% | 0 | 0 | 100% |
| DE/rand/1/bin (F=0.9 CR=0.9) | 1.52E-01 [†] | 1.15E-01 | 0% | 9.01E-01 [†] | 1.40E-01 | 0% | 8.54E+01 [†] | 3.30E+01 | 0% |
| DE/rand/1/bin (F=0.5 CR=0.3) | 0 [‡] | 0 | 100% | 2.24E-05 [‡] | 1.19E-04 | 93% | 3.10E+01 [†] | 3.24E+00 | 0% |
| DE/rand-to-best/1/bin (F=0.5 CR=0.3) | 1.08E+01 | 1.00E+01 | 0% | 1.82E+02 [†] | 5.47E+01 | 0% | 9.58E+00 [‡] | 3.88E+00 | 0% |
| DE/rand-to-best/2/bin (F=0.5 CR=0.3) | 0 [‡] | 0 | 100% | 3.97E-03 | 1.85E-02 | 37% | 4.03E+01 [†] | 3.73E+00 | 0% |
| SaDE | 2.38E-03 | 5.03E-03 | 80% | 8.54E-03 | 9.09E-03 | 40% | 0 | 0 | 100% |
| ADE(Zaharie) | 0 [‡] | 0 | 100% | 2.93E-03 | 6.65E-03 | 10% | 2.32E-01 [†] | 5.01E-01 | 80% |
| SDE | 1.59E+00 [‡] | 2.23E+00 | 13% | 1.39E+00 [‡] | 4.24E+00 | 13% | 1.09E+01 [†] | 4.23E+00 | 0% |
| jDE | 0 [‡] | 0 | 100% | 5.17E-03 | 6.64E-03 | 57% | 0 | 0 | 100% |

| Algorithm | f_{10} . Rotated Rastrigin | | | f_{11} . Non-continuous Rastrigin | | | f_{12} . Schwefel | | |
|--------------------------------------|------------------------------|-----------------|--------------|-------------------------------------|----------|--------------|-----------------------|----------|--------------|
| | Mean | Std | Success Rate | Mean | Std | Success Rate | Mean | Std | Success Rate |
| DE/rand/1/bin (F=0.9 CR=0.1) | 1.68E+02 [†] | 1.43E+01 | 0% | 0 | 0 | 100% | 0 | 0 | 100% |
| DE/rand/1/bin (F=0.9 CR=0.9) | 2.45E+02 [†] | 2.20E+01 | 0% | 6.93E+01 [†] | 2.37E+01 | 0% | 4.98E+03 [†] | 1.75E+03 | 0% |
| DE/rand/1/bin (F=0.5 CR=0.3) | 1.87E+02 [†] | 1.09E+01 | 0% | 2.88E+01 [†] | 1.94E+00 | 0% | 0 | 0 | 100% |
| DE/rand-to-best/1/bin (F=0.5 CR=0.3) | 1.44E+02 [†] | 1.09E+01 | 0% | 1.18E+01 [†] | 2.70E+00 | 0% | 4.18E+02 [†] | 1.75E+02 | 3% |
| DE/rand-to-best/2/bin (F=0.5 CR=0.3) | 1.88E+02 [†] | 7.15E+00 | 0% | 3.17E+01 [†] | 2.25E+00 | 0% | 2.48E+03 [†] | 4.01E+02 | 0% |
| SaDE | 1.67E+01 | 5.26E+00 | 0% | 0 | 0 | 100% | 0 | 0 | 100% |
| ADE(Zaharie) | 1.21E+02 [†] | 1.28E+01 | 0% | 0 | 0 | 100% | 3.95E+00 | 2.16E+01 | 97% |
| SDE | 3.63E+01 [†] | 6.78E+00 | 0% | 1.56E+01 [†] | 3.52E+00 | 0% | 4.68E+02 [†] | 2.08E+02 | 0% |
| jDE | 3.65E+01 [†] | 8.29E+00 | 0% | 6.67E-02 | 2.54E-01 | 93% | 6.32E+01 [†] | 1.07E+02 | 63% |



شکل ۶: نتایج همگرایی الگوریتم‌های معمول برای f_1 تا f_{14}



شکل ۷: نتایج همگرایی الگوریتم‌های تطبیقی برای f_1 تا f_{14}

| | rand/1/bin (F=0.9 CR=0.1) | | rand/1/bin (F=0.9 CR=0.9) | | rand/1/bin (F=0.5 CR=0.3) | | rand-to-best/1/bin (F=0.5 CR=0.3) | | rand-to-best/2/bin (F=0.5 CR=0.3) | | SaDE | |
|----------|------------------------------|------|------------------------------|------|------------------------------|------|--------------------------------------|------|--------------------------------------|------|---------------|------|
| | NFE | SR | NFE | SR | NFE | SR | NFE | SR | NFE | SR | NFE | SR |
| 10D | | | | | | | | | | | | |
| f_1 | 16770 | 100% | 53298 | 100% | 10291 | 100% | 6318 | 100% | 10058 | 100% | 8375 | 100% |
| f_2 | - | 0% | - | 43% | 72436 | 100% | 23383 | 100% | 53658 | 100% | 14867 | 100% |
| f_3 | - | 0% | - | 0% | - | 0% | - | 0% | - | 0% | 42446 | 100% |
| f_4 | - | 0% | - | 0% | - | 83% | 30925 | 100% | 71278 | 100% | 15754 | 100% |
| f_5 | 25335 | 100% | 82919 | 100% | 15157 | 100% | 9436 | 100% | 15045 | 100% | 12123 | 100% |
| f_6 | - | 90% | 85272 | 100% | 16682 | 100% | 9923 | 100% | 16980 | 100% | 12244 | 100% |
| f_7 | 41247 | 100% | - | 0% | 29961 | 100% | - | 70% | 59205 | 100% | 35393 | 100% |
| f_9 | 19200 | 100% | - | 0% | 23155 | 100% | - | 93% | 30621 | 100% | 23799 | 100% |
| f_{11} | 20898 | 100% | - | 0% | 29559 | 100% | - | 87% | 46592 | 100% | 26945 | 100% |
| f_{12} | 19094 | 100% | - | 13% | 14698 | 100% | - | 90% | 33091 | 100% | 16663 | 100% |
| f_{13} | - | 0% | 53131 | 100% | - | 97% | - | 80% | 38290 | 100% | 9740 | 100% |
| 30D | | | | | | | | | | | | |
| f_1 | 66339 | 100% | - | 0% | 34687 | 100% | - | 97% | 31470 | 100% | 20184 | 100% |
| f_2 | - | 0% | - | 0% | - | 0% | - | 0% | - | 0% | 118743 | 100% |
| f_3 | 92421 | 100% | - | 0% | 49822 | 100% | - | 93% | 45948 | 100% | 26953 | 100% |
| f_6 | - | 0% | - | 0% | 55108 | 100% | - | 97% | 49961 | 100% | 33014 | 100% |
| f_7 | 80741 | 100% | - | 0% | 39436 | 100% | - | 0% | 41314 | 100% | - | 80% |
| f_9 | 90391 | 100% | - | 0% | - | 0% | - | 0% | - | 0% | 58723 | 100% |
| f_{11} | 108406 | 100% | - | 0% | - | 0% | - | 0% | - | 0% | 77920 | 100% |
| f_{12} | 78056 | 100% | - | 0% | 73756 | 100% | - | 3% | - | 0% | 44283 | 100% |
| f_{13} | - | 47% | - | 0% | 34012 | 100% | - | 53% | 38386 | 100% | 19031 | 100% |
| f_{15} | 80944 | 100% | - | 0% | 39460 | 100% | 41729 | 100% | 18617 | 100% | 25137 | 100% |
| f_{16} | - | 0% | - | 0% | 149511 | 100% | 112445 | 100% | - | 0% | 88934 | 100% |
| f_{17} | 62766 | 100% | - | 90% | 32420 | 100% | 32442 | 100% | 14289 | 100% | 18742 | 100% |
| f_{18} | 62388 | 100% | - | 87% | 31346 | 100% | 28690 | 100% | - | 93% | 19390 | 100% |
| f_{19} | - | 90% | 7782 | 100% | 31121 | 100% | 33900 | 100% | - | 93% | 6426 | 100% |
| f_{20} | 2744 | 100% | 2628 | 100% | 2178 | 100% | 2086 | 100% | 1416 | 100% | 2076 | 100% |
| f_{21} | 3659 | 100% | 2713 | 100% | 2246 | 100% | - | 97% | 1593 | 100% | 2614 | 100% |
| f_{22} | 1314 | 100% | 996 | 100% | 926 | 100% | 672 | 100% | 1004 | 100% | 802 | 100% |
| f_{23} | 5656 | 100% | - | 60% | 3970 | 100% | - | 67% | 4759 | 100% | 3080 | 100% |
| f_{24} | 31615 | 100% | 7720 | 100% | 10468 | 100% | - | 90% | 12381 | 100% | 4947 | 100% |
| f_{25} | 35503 | 100% | 6596 | 100% | 8653 | 100% | 5085 | 100% | 12921 | 100% | 4173 | 100% |
| f_{26} | 33905 | 100% | 6974 | 100% | 8742 | 100% | 4853 | 100% | 14933 | 100% | 4267 | 100% |

شکل ۸: مقایسه‌ی سرعت روش پیشنهادی با روش‌های معمول تکامل تفاضلی

سراسری با مشکل روبه‌رو شده‌است. در حالی که SaDE بیشتر توابع آزمون را با نرخ ۱۰۰٪ موفقیت بهینه کرده‌است و تنها برای f_3 به موفقیت ۹۰٪ رسیده‌است که دیگر الگوریتم‌های معمول در آن‌جا کاملاً شکست خورده‌اند. جالب آن‌جاست که در حالت ۳۰ بعدی، f_4 ، f_{10} و f_{14} به قدری سخت شده‌اند که هیچ‌کدام از روش‌های موجود قادر به پیدا کردن بهینه بدون تجاوز از تعداد مجاز ارزیابی نبوده‌اند!

برای مقایسه‌ی سرعت همگرایی‌ها، شکل ۸ تعداد ارزیابی‌ها و نرخ موفقیت‌های ۱۰۰٪ را مقایسه نموده‌است. مشاهده می‌شود که تنها در f_7 ، الگوریتم ارایه‌شده از دیگر الگوریتم‌ها کندتر بوده‌است و در باقی آزمون‌ها موفق‌تر عمل کرده‌است.

۲.۳.۵ مقایسه با روش‌های تطبیقی

در مقایسه‌ی SaDE با دیگر الگوریتم‌های تطبیقی، که jDE، SDE و ADE هستند، بهترین پاسخ‌ها برای ۱۰ بعد و توابع تک‌قله‌ای f_1 تا f_4 توسط SaDE و jDE به‌دست آمده‌است که با توجه به سرعت همگرایی شکل ۷، الگوریتم ارایه‌شده کمی برتری دارد. با توجه به شکل ۵ نیز می‌توان دریافت که در توابع f_7 و f_9 تا قبل از چرخش، همه‌ی روش‌ها پاسخ موفق بوده‌اند اما پس از چرخش، یعنی توابع f_8 و f_{10} تنها SaDE و jDE پاسخ‌گو بوده‌اند و آن‌هم در مورد f_{10} هیچ‌کدام موفق نبوده‌اند و به نظر می‌رسد این‌که پژوهشگران مقاله‌ی مورد نظر در این‌جا برتری را برای روش خود دانسته‌اند، تنها معیار ممکن برای چنین تحلیلی میانگین نزدیک‌تر به بهینه‌ی سراسری و انحراف استاندارد بوده‌است که توسط روش حاصل شده‌است.

در شکل ۵ می‌توان دید که روش ارایه‌شده در سی بعد برای f_4 ، f_{10} و f_{14} نسبت به دیگر روش‌های تطبیقی اصلاً موفق نبوده‌است، هرچند نسبت به دیگر روش‌ها میانگین و انحراف استاندارد بهتری داشته‌است، به جز برای f_{14} که تنها jDE از روش ارایه‌شده مقادیر بهتری را به‌دست داده‌است.

۳.۳.۵ مقایسه‌ی کارایی کلی:

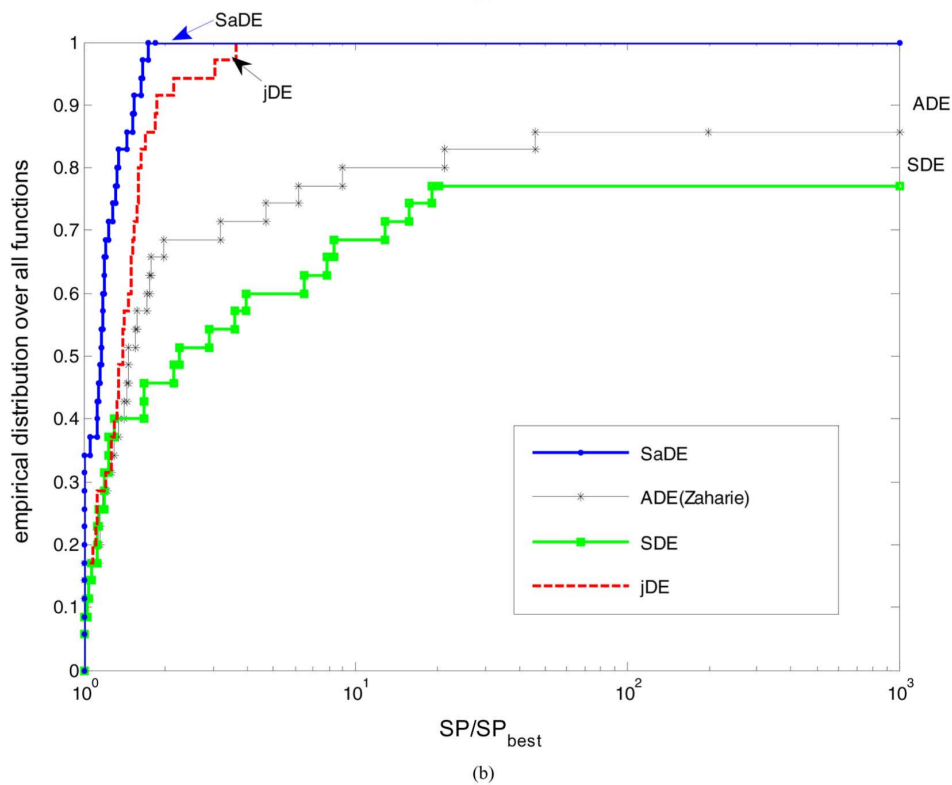
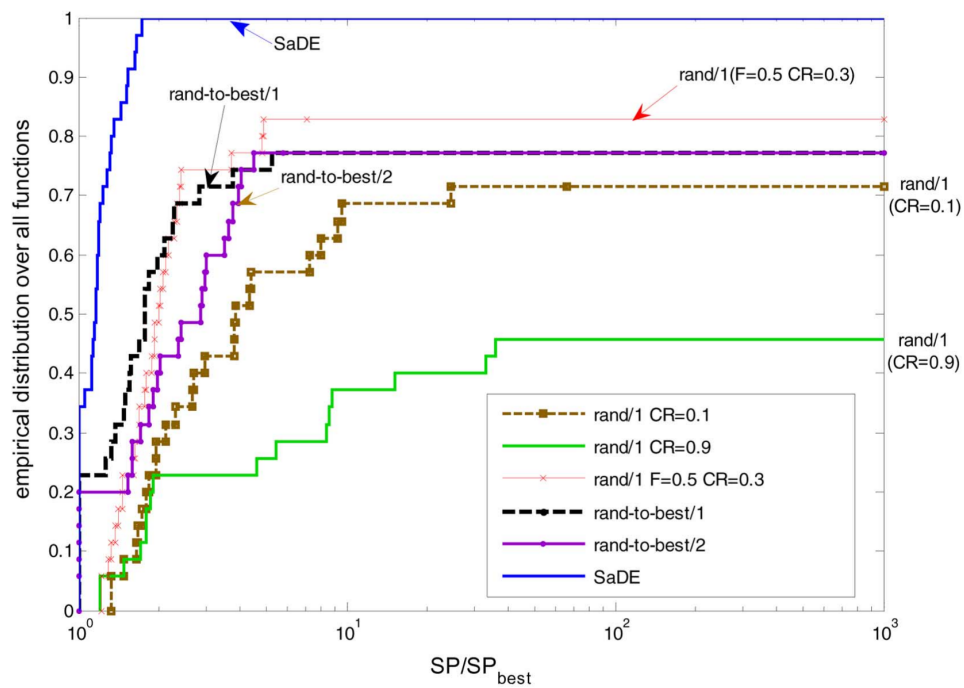
برای مقایسه‌ی کارایی ۹ الگوریتم مورد بررسی، پژوهشگران مقاله از روش رسم توزیع نمونه‌ای^۱ کارایی موفقیت نرمال شده استفاده کرده‌اند که در [۲۹] معرفی شده‌است. این روش کارایی موفقیت (SP) را به صورت زیر تعریف نموده‌است:

$$SP = \text{mean}(FEs) \times \frac{\#totalruns}{\#successfulruns} \quad (۱۲)$$

همان‌گونه که مشاهده می‌شود، این معیار از حاصل ضرب میانه‌ی تعداد ارزیابی‌ها در نرخ موفقیت است. با کمی تامل در این فرمول، می‌توانیم به این نکته پی ببریم که اگر نرخ موفقیت الگوریتمی کم باشد اما میانه‌ی تعداد ارزیابی‌های آن نیز کم باشد، با توجه به این که معکوس نرخ موفقیت در این معیار استفاده شده‌است، پس می‌تواند کارایی موفقیت بیشتری نسبت به الگوریتمی داشته باشد که نرخ موفقیتش بیشتر است اما برای رسیدن به این نرخ موفقیت، ارزیابی‌های بیشتری انجام داده‌است و این به معنی کند بودن آن است، به گونه‌ای که ممکن است از الگوریتمی که مثال زده شد، کارایی موفقیت کمتری را به دست دهد. پژوهشگران این مقاله نرمال کردن را نیز با تقسیم هر کارایی موفقیت بر بهترین کارایی موفقیت برای آن تابع انجام داده‌اند. این نمودار توزیع نمونه‌ای (توزیع تک تک مشاهدات در یک نمونه) در شکل ۹ آورده شده‌است که در آن هر قدر نرمال کارایی موفقیت کمتر و توزیع نمونه‌ای در آن جا بیشتر باشد، بهتر است.

در آخر، نتایج این روش با FADE نیز مقایسه شده‌است که دلیل آن، آسان بودن حل ابعاد پایین برای هر دو الگوریتم است و نتایج برای مسایل ۵۰ بعدی به صورت خاص برای این دو روش انجام شده‌است. این بررسی در خود مقاله‌ی انتخابی نیز به طور مختصر آورده شده‌است که برتری الگوریتم ارایه شده را مشخص می‌کند، اما متأسفانه بررسی الگوریتم FADE در مقاله‌ی مورد نظر انجام نگرفته است و از حوصله‌ی این گزارش نیز خارج است.

^۱ Empirical distribution



شکل ۹: توزیع نمونه‌ای کارایی موفقیت برای ۲۶ تابع آزمون

۶ نتیجه‌گیری

در این مقاله، پژوهشگران برای رفع مشکل زمان‌بر بودن جست‌وجو برای یافتن استراتژی مناسب برای تولید بردار آزمایشی در مسایل تکامل تفاضلی و پارامترهای مناسب آن استراتژی، روشی خود تطبیقی ارایه کردند که در آن استراتژی انتخاب بردار آزمایشی و پارامتر CR متناسب هر استراتژی را در افراد قرار داده‌اند تا تکامل پیدا کند، به این شکل که در طول دوره‌ی یادگیری، نرخ موفقیت هر استراتژی (که در استخر استراتژی ذخیره شده‌اند) و مقادیر مناسب پارامتر یادشده را در حافظه‌هایی نگهداری کرده‌اند و پس از اتمام دوره‌ی یادگیری، احتمال انتخاب هر استراتژی و مقادیر مناسب پارامتر بازترکیبی را به‌روزرسانی کرده‌اند. احتمال انتخاب هر استراتژی، متناسب با نرخ تولید بردارهای آزمایشی که موفق به ورود به نسل بعد شده‌اند، به‌روزرسانی شده‌است. مقادیر CR نیز با توجه به حساسیت بالای آن، طبق یک تابع توزیع نرمال با واریانس کم تغییر داده شده‌است و برای هر استراتژی در طی دوره‌ی یادگیری ذخیره شده‌است. به‌روزرسانی مقدار این پارامتر به این شکل است که در پایان دوره‌ی یادگیری مقادیر آن برای هر کدام از استراتژی‌ها با میانه‌ی مقادیر این پارامتر در آن دوره، جاکزین می‌شود. انتخاب استراتژی‌هایی با ویژگی‌های متفاوت در استخر یادشده، سبب شده‌است این روش در مراحل مختلف تکامل بتواند استراتژی مناسب با مقدار پارامتر مناسب برای آن مرحله را انتخاب کند. آزمایش‌ها بر روی ۲۶ تابع آزمون محبوب و مطرح در تکامل تفاضلی انجام شده‌است و مقایسه‌ها با ۸ روش معمولی و تطبیقی مطرح، انجام شده‌است. به نظر می‌رسد نقطه‌ی قوت اصلی این مقاله در همین مقایسه‌ها و آزمایش‌ها بوده است چرا که با توجه به کارهای انجام‌شده در تکامل تفاضلی که مطالعه شد، تقریباً تمامی جنبه‌های آزمایش‌ها و مقایسه‌ها را مطرح نموده‌است و در انتخاب توابع آزمون نیز کوتاهی نکرده‌است.

از کارهای آتی این مقاله می‌توان انتظار داشت اندازه‌ی بهینه‌ی استخر استراتژی و معیارهای انتخاب استراتژی‌های استخر را با توجه به ویژگی‌های مساله بررسی کنند. همچنین تطبیق پارامتر F و NP نیز می‌تواند انجام گیرد. در کل ایده‌ی ساده ولی کار پژوهشی قوی انجام‌شده توسط این پژوهشگران می‌تواند آموزه‌ای برای کارهای آتی در زمینه‌ی تکامل تفاضلی و علاقه‌مندان به تکنیک‌های تکاملی باشد.

- [1] P. J. Angeline, "Adaptive and self-adaptive evolutionary computation," In Computational Intelligence: A Dynamic System Perspective, M. Palaniswami, Y. Attikiouzel, R. J. Marks, D. Fogel, and T. Fukuda, Eds. New York: IEEE Press, 1995, pp. 152–161.
- [2] J. E. Smith and T. C. Fogarty, "Operator and parameter adaptation in genetic algorithms," *Soft Comput.*, vol. 1, no. 2, pp. 81–87, 1997.
- [3] R. Storn and K. V. Price, "Differential evolution-A simple and efficient heuristic for global optimization over continuous Spaces," *J. Global Optim.*, vol. 11, pp. 341–359, 1997.
- [4] A. K. Qin, V. L. Huang, and P. N. Suganthan "Differential Evolution Algorithm With Strategy Adaptation for Global Numerical Optimization," *Evolutionary Computation, IEEE Transactions on*, vol. 13, no. 2, 398-417, 2009.
- [5] K. Price, R. Storn and J. Lampinen, "Differential Evolution — A Practical Approach to Global Optimization," Berlin, Germany: Springer-Verlag, 2005.
- [6] J. Lampinen and I. Zelinka, "On stagnation of the differential evolution algorithm," In Proc. 6th Int. Mendel Conf. Soft Comput., P. Osmera, Ed., pp. 76–83, 2002.
- [7] R. Gämperle, S. D. Müller, and P. Koumoutsakos, "A parameter study for differential evolution," In Advances in Intelligent Systems, Fuzzy Systems, Evolutionary Computation, A. Grmela and N. E. Mastorakis, Eds. Inter-laken, Switzerland: WSEAS Press, pp. 293–298, 2002.
- [8] D. Zaharie, "Control of population diversity and adaptation in differential evolution algorithms," In Proc. Mendel 9th Int. Conf. Soft Comput., R. Matousek and P. Osmera, Eds., Brno, Czech Republic, pp. 41–46, 2003.
- [9] S. Das, A. Konar, and U. K. Chakraborty, "Two improved differential evolution schemes for faster global search," In ACM-SIGEVO Proc. Genetic Evolut. Comput. Conf., Washington, DC, pp. 991–998, 2005.
- [10] J. Kennedy, and E. Russell, "Particle swarm optimization," In Proceedings of 1995 IEEE International Conference on Neural Networks, pp. 1942-1948, 1995.
- [11] Y. Shi, and R. C. Eberhart, "Empirical study of particle swarm optimization." In Evolutionary Computation, CEC 99, Proceedings of the 1999 Congress on, IEEE, vol. 3, pp. 1945-1950, 1999.

- [12] Y. Shi, and R. C. Eberhart, "Particle swarm optimization: developments, applications and resources." In *Evolutionary Computation, Proceedings of the 2001 Congress on*, IEEE, vol. 1, pp. 81-86, 2001.
- [13] R. Storn and K. Price, "Differential evolution-a simple and efficient adaptive scheme for global optimization over continuous spaces," Berkeley: ICSI, 1995. Available: <http://http.icsi.berkeley.edu/~storn/litera.html>
- [14] K. V. Price, "An introduction to differential evolution," In *New Ideas in Optimization*, D. Corne, M. Dorigo, and F. Glover, Eds. London, U.K.: McGraw-Hill, pp. 79-108, 1999.
- [15] J. Rönkkönen, S. Kukkonen, and K. V. Price, "Real parameter optimization with differential evolution," In *Proc. IEEE Congr. Evolut. Comput.*, Edinburgh, Scotland, pp. 506-513, 2005.
- [16] L. Ingber, "Simulated annealing: Practice versus theory." *Mathematical and computer modelling*, vol.18, no. 11, pp. 29-57, 1993.
- [17] W. H. Press, S. A. Teukolsky, B. P. Flannery, and W. T. Vetterling, "Numerical recipes in C," Cambridge: Cambridge University, pp. 545-555, 1992.
- [18] Nikolaus Hansen, and A. Ostermeier, "Convergence Properties of Evolution Strategies with the Derandomized Covariance Matrix Adaptation: The $(\mu/\mu_{-I}, \lambda)$ -ES," In *Proceedings of the 5th European Congress on Intelligent Techniques and Soft Computing (EUFIT'97)*, pp. 650-654, 1997.
- [19] J. Liu, and J. Lampinen, "A fuzzy adaptive differential evolution algorithm," *Soft Comput.*, vol. 9, no. 6, pp. 448-462, 2005.
- [20] D. Zaharie, and D. Petcu, "Adaptive pareto differential evolution and its parallelization," in *Proc. 5th Int. Conf. Parallel Process. Appl. Math.*, Czestochowa, Poland, pp. 261-268, 2003.
- [21] H. A. Abbass, "The self-adaptive Pareto differential evolution algorithm," in *Proc. Congr. Evolut. Comput.*, Honolulu, HI, pp. 831-836, 2002.
- [22] A. K. Qin, and P. N. Suganthan, "Self-adaptive differential evolution algorithm for numerical optimization," in *Proc. IEEE Congr. Evolut. Comput.*, Edinburgh, Scotland, pp. 1785-1791, 2005.
- [23] M. G. H. Omran, A. Salman, and A. P. Engelbrecht, "Self-adaptive differential evolution," in *Lecture Notes in Artificial Intelligence*. Berlin, Germany: Springer-Verlag, pp. 192-199, 2005.
- [24] J. Teo, "Exploring dynamic self-adaptive populations in differential evolution," *Soft Comput.*, vol. 10, no. 8, pp. 637-686, 2006.

- [25] J. Brest, S. Greiner, B. Boskovic, M. Mernik, and V. Zumer, "Self-adapting control parameters in differential evolution: A comparative study on numerical benchmark problems," *IEEE Trans. Evolut. Comput.*, vol. 10, no. 6, pp. 646–657, 2006.
- [26] R. Storn, "On the usage of differential evolution for function optimization," in *Proc. Biennial Conf. North Amer. Fuzzy Inf. Process. Soc.*, Berkeley, CA, pp. 519-523, 1996.
- [27] J. E. Baker, "Reducing bias and inefficiency in the selection algorithm," In *Proc. 2nd Int. Conf. Genetic Algorithms*, Cambridge, pp. 14–21, 1987.
- [28] J. J. Liang, P. N. Suganthan, and K. Deb, "Novel composition test functions for numerical global optimization," In *Proc. IEEE Swarm Intell. Symp.*, Pasadena, CA, pp. 68–75, 2005.
- [29] N. Hansen, "Compilation of results on the 2005 CEC benchmark function set," May 4, 2006 [Online]. Available: http://www.ntu.edu.sg/home/epnsugan/index_files/CEC-05/compareresults.pdf