

یک الگوریتم تکاملی جدید مبتنی بر چرخه آب در طبیعت

^۱سید مرتضی حسینی، ^۲مجید وفايي جهان، ^۳جواد صدري

^۱گروه کامپیوتر- نرم افزار، دانشگاه آزاد اسلامی، واحد مشهد، seyedmorteza.hosseiny@gmail.com

^۲گروه کامپیوتر- نرم افزار، دانشگاه آزاد اسلامی، واحد مشهد، VafaeiJahan@mshdiau.ac.ir

^۳دانشکده مهندسی برق و کامپیوتر، دانشگاه بیرجند، javad.sadri@mail.mcgill.ca

چکیده

در سال‌های اخیر، متدهای فراابتکاری زیادی توسعه داده شده‌اند. بسیاری از این متدها الهام گرفته از فرایندهای طبیعی می‌باشند مانند الگوریتم ژنتیک، پرندگان، مورچه‌ها و بازپخت شبیه سازی شده. در این مقاله یک الگوریتم بهینه‌سازی جدید مبتنی بر چرخه آب در طبیعت ارائه شده است. در این الگوریتم همانند چرخه آب در طبیعت، با الگو برداری از تبخیر آب از اقیانوس، تشکیل ابر، تشکیل رودخانه و همچنین الگو برداری از سرریز آب از گودال‌ها در طبیعت الگوریتم جدیدی ارائه شده است که قابلیت بالایی در فرار از بهینه محلی و همچنین سرعت زیادی در رسیدن به بهینه سراسری دارد. پیاده‌سازی الگوریتم ارائه شده بر روی توابع آزمون استاندارد و نتیجه بدست آمده از آن بیانگر کیفیت مطلوب الگوریتم در دستیابی به بهترین حالت در هر تابع آزمون می‌باشد. همچنین مقایسات صورت گرفته بین الگوریتم ارائه شده با الگوریتم‌های مشهور دیگر نشان دهنده برتری الگوریتم ارائه شده نسبت به اکثر الگوریتم‌های مورد مقایسه در اکثر توابع آزمون می‌باشد.

واژه‌های کلیدی: بهینه‌سازی، الگوریتم‌های تکاملی، چرخه آب در طبیعت.

۱- مقدمه

برای بعضی مسائل خاص نسبت به الگوریتم‌های دیگر ارائه می‌دهند. در طرح ارائه شده سعی شده است که با بهره‌گیری و الهام از چرخه آب در طبیعت یک الگوریتم تکاملی جدید معرفی شود. هدف الگوبرداری از چرخه آب در طرح ارائه شده، طراحی یک الگوریتم جدید می‌باشد که همانند ریزش باران، جمعیت هر نسل هم دارای تنوع و پراکندگی مناسبی باشد و همچنین با شبیه سازی فرآیند حرکت آب‌های ریزشی به سمت رودخانه، الگوریتم سرعت همگرایی بالایی برای رسیدن به بهینه سراسری داشته باشد. علاوه بر این با الگوبرداری از اشباع شدن گودال‌ها در حین بارندگی فرار از بهینه محلی طراحی شده است که این امر موجب می‌شود که الگوریتم ارائه شده جستجوی مناسبی را در فضای مسئله داشته باشد و در بهینه محلی به دام نیفتد.

۲- استراتژی بهینه‌سازی مبتنی بر چرخه آب در طبیعت

همانطور که در بخش قبلی اشاره شد، بسیاری از الگوریتم‌های تکاملی با الگو برداری از طبیعت طراحی شده‌اند، همانند الگوریتم‌های ژنتیک، پرندگان، مورچگان و غیره. در این فصل، الگوریتم تکاملی جدیدی برای جستجوی عام معرفی می‌شود که از چرخه آب در طبیعت الهام گرفته شده است. هدف الگوبرداری از چرخه آب در طرح ارائه

در بسیاری از مسائل با فضای جستجو با بعدها زیاد، الگوریتم‌های بهینه‌سازی کلاسیک راه حل‌های مناسبی را ارائه نمی‌دهند، زیرا که فضای مسئله بصورت نمایی با افزایش اندازه مسئله، افزایش می‌یابد. بنابراین حل این مسئله با استفاده از تکنیک‌های قبلی اجرا پذیر نمی‌باشد.

در طول دهه‌های گذشته، پیشرفت بزرگی در الگوریتم‌هایی بر پایه رفتارهای طبیعی اتفاق افتاد [7,15,16,17]. در بسیاری از تحقیقات، این الگوریتم‌ها برای حل مسائل محاسباتی پیچیده از قبیل بهینه‌سازی توابع هدف [6]، هدف‌های کنترلی [14]، پردازش تصویر [4]، مدلسازی فیلترها [13] و غیره بکار برده شده‌اند. تاکنون الگوریتم‌های هوشمند مختلفی توسط محققان همانند الگوریتم ژنتیک [20]، بازپخت شبیه سازی شده [1]، مورچه‌ها [7]، پرندگان [15] و بسیاری دیگر ارائه شده است. این الگوریتم‌ها به طور گسترده‌ای برای آنالیز و بررسی بسیاری از مسائل تحقیقاتی مورد استفاده قرار گرفته‌اند [1,2,9].

با این حال، الگوریتم خاصی برای دستیابی به بهترین راه حل در تمام مسائل بهینه‌سازی وجود ندارد. بعضی از الگوریتم‌ها، راه حل‌های بهتری را

اقیانوس‌ها ریخته می‌شوند و قسمت دیگری از آن نیز بر روی زمین می‌بارند که در اثر جاذبه روی سطح زمین جاری شده و تبدیل به رواناب سطحی می‌شوند. بعضی از رواناب‌های سطحی وارد رودخانه‌ها شده و در نهایت به سمت اقیانوس‌ها حرکت می‌کنند؛ برخی دیگر نیز به صورت آب شیرین در دریاچه‌ها و رودخانه‌ها تجمع پیدا می‌کنند. تمام رواناب‌های بر روی سطح زمین جاری نمی‌شوند و قسمت زیادی از آنها به زیر زمین فرو می‌روند (نفوذ). بخشی از این آب نفوذ یافته، به اعماق پایین‌تر رفته و آبخوان‌ها را تغذیه می‌کنند (سنگ‌های زیرسطحی اشباع از آب) و حجم زیادی از آب شیرین را برای مدت‌های مدید نگهداری می‌کنند. بخشی از آب زیرزمینی نیز نزدیک به سطح زمین باقی می‌ماند و مجدداً به شکل تراوش به بدنه‌های آبی (واقیانوس‌ها) باز می‌گردند و مانند تخلیه آب زیرزمینی عمل می‌کنند. برخی دیگر نیز به صورت چشمه‌های آب شیرین از زیر زمین بیرون می‌آیند. در طول زمان، این حجم آب در حال حرکت است و هر از چند گاهی بخشی از آن به اقیانوس‌ها ختم می‌یابند که همان نقطه شروع است، و باز مجدداً این مسیر طی می‌شود. شکل ۱ نشان دهنده چرخه آب با جزئیات کامل می‌باشد.



شکل ۱ - چرخه کامل آب در طبیعت [6]

۲-۲- الگوریتم پیشنهادی مبتنی بر چرخه آب برای مسائل

تک هدفه

در ابتدا مفاهیم زیر تعریف می‌شوند:

- **ریزقطره:** یک مقدار از یک متغیر مسئله می‌باشد. هر متغیر مسئله یک پارامتر تاثیر گذار و مجهول از مسئله می‌باشد که مجموع n متغیر (در یک مسئله با n متغیر)، یک جواب از مسئله می‌باشد. هر ریزقطره یک نوع دارد که نوع آن وابسته به متغیر مسئله می‌باشد. به طور مثال، یک ریزقطره از نوع اول وابسته به متغیر اول مساله و یک ریزقطره از نوع دوم

شده، طراحی یک الگوریتم جدید می‌باشد که همانند ریزش باران، هم دارای تنوع و پراکندگی مناسبی باشد (قطرات بصورت تقریباً هماهنگ و یکدست ریزش می‌یابند) و همچنین دارای سرعت همگرایی بالایی برای رسیدن به بهینه سراسری داشته باشد.

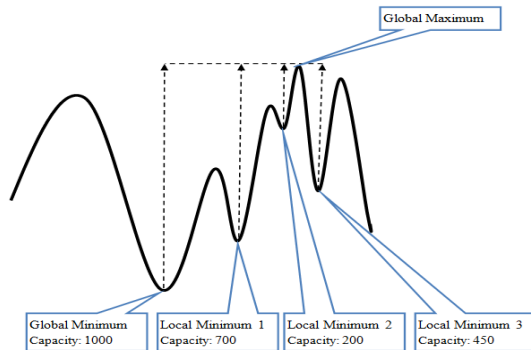
به طور خلاصه، این الگوریتم، شامل یک مجموعه‌ای از مقادیر اولیه برای هر پارامتر از مسئله می‌باشد. در هر دور از اجرای این الگوریتم، برای هر پارامتر از مسئله با یک اولویت مقادیری انتخاب می‌شوند و تشکیل یک مجموعه کوچکتر را می‌دهند. سپس با در نظر گرفتن شایستگی مقادیر، با ترکیب مقادیر مربوط به هر پارامتر، چندین جواب برای مسئله تولید می‌شود. سپس هر جواب از مسئله با استفاده از الگوریتم جستجوی محلی، در بهینه محلی خود قرار می‌گیرد. هر بهینه محلی با توجه به میزان برازندگی خود، یک مقدار گنجایش دریافت می‌کند که میزان ماندگاری آن جواب در تعداد اجراهای متوالی و همچنین احتمال کاندید شدن برای حرکت دیگر جواب‌ها به سمت خود را بیان می‌کند. در هر دور از اجرا، جواب‌هایی از مسئله که باید از مسئله خارج شوند به سمت جوابی با شایستگی بهتر حرکت می‌کنند و در طول مسیر اگر به شایستگی بهتری برخورد نکردند، به بهینه محلی انتخاب شده خود می‌رسند. هر دور یک جواب مسئله با بیشترین شایستگی ثابت می‌ماند و در پایان هر دور، مقادیر هر پارامتر از این جواب مسئله وارد مجموعه مقادیر اولیه می‌شوند. بعد از پایان f دور از الگوریتم، بهترین جواب دور f ام جواب بهینه مسئله می‌باشد.

۲-۱- مروری بر چرخه آب در طبیعت

چرخه آب، که با نام چرخه‌ی هیدرولوژی نیز شناخته می‌شود، وجود و حرکت آب را در رو و زیر زمین و بالای سطح زمین نشان می‌دهد. آب در زمین همیشه در حال حرکت است و همیشه در حال تغییر شکل می‌باشد، از مایع به بخار، به یخ و دوباره بازگشت به حالت اول. چرخه آب میلیاردها سال است که در حال کار است و زندگی همه چیز روی زمین به آب بستگی دارد.

توده‌های هوا، ابرها را در اطراف زمین به حرکت درمی‌آورند. قطعات ابر با یکدیگر برخورد کرده، رشد می‌کنند و به شکل بارش به سمت پایین می‌آیند. قسمتی از این بارش به شکل برف بر روی توده‌های برفی و یخچال‌ها تجمع می‌یابند. برف در نقاط گرم‌تر ذوب شده و به شکل نهر جاری می‌شود و تبدیل به برفاب می‌شوند. بارش‌ها بر روی سطح

در حین اجرای الگوریتم با توجه به ظرفیت خود، قطرات را در خود جای می‌دهد. شکل ۴ نشان دهنده چندین گودال می‌باشد که همانطور که مشخص است دارای گنجایش‌های متفاوتی می‌باشند.



شکل ۴- گودال‌های متفاوت در فضای مسئله با گنجایش‌های متفاوت

این نکته باید مورد توجه قرار گیرد که یک قطره پس از قرارگیری در بهینه محلی خود و شناسایی گودال، نماینده آن گودال می‌باشد. به عبارت دیگر آن گودال نیز همانند یک قطره دارای n ریزقطره و همچنین دارای یک ارزش (fitness) می‌باشد.

• **رودخانه:** رودخانه، با ارزش‌ترین گودال (گودالی با بهترین fitness) در هر دور از اجرا می‌باشد. در پایان هر دور از اجرا ریز قطرات تبخیر شده از اقیانوس توسط ریز قطرات رودخانه جایگزین می‌شوند.

الگوریتم ارائه شده مبتنی بر چرخه آب در طبیعت شامل ۶ بخش اصلی زیر می‌باشد:

- ۱- مقداردهی اولیه
- ۲- تبخیر
- ۳- میعان و شکل‌گیری باران
- ۴- جستجوی محلی
- ۵- جریان سطحی آب
- ۶- ورود آب رودخانه به اقیانوس

۱- **مقدار دهی اولیه:** همانند بسیاری از الگوریتم‌های تکاملی، این الگوریتم نیز با مجموعه‌ای از مقادیر اولیه تصادفی آغاز می‌شود. در اجرای اول از الگوریتم، اقیانوس توسط مقادیر اولیه تصادفی مقداردهی می‌شود.

وابسته به متغیر دوم مساله و الی آخر خواهد بود. هر ریزقطره فقط با یک ریزقطره هم نوع دیگر تعامل خواهد داشت.

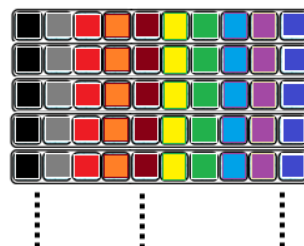
• **قطره:** هر عضو از جمعیت یا راه حل یک قطره نام دارد که با یک موقعیت و سرعت اولیه در فضای جستجوی مساله برای یافتن راه حل بهینه حرکت می‌کند. همچنین یک قطره دنباله‌ای از n ریزقطره می‌باشد که هر ریزقطره از یک نوع خاص می‌باشد. شکل ۲ نشان دهنده یک قطره متشکل از ۱۰ ریزقطره از ۱۰ نوع می‌باشد.



شکل ۲- نمایی از یک قطره که متشکل از ۱۰ ریزقطره از ۱۰ نوع می‌باشد

• **اقیانوس:** یک انباره بزرگ از ریزقطرات می‌باشد. در اقیانوس برای هر نوع ریز قطره، به تعداد k (به تعداد مساوی) ریزقطره وجود خواهد داشت. اندازه یک اقیانوس برابر تعداد ریزقطرات موجود در آن تقسیم بر n (تعداد متغیرهای مسئله) می‌باشد. به طور مثال اگر اندازه اقیانوس برابر a باشد، تعداد $(a \times n)$ ریزقطره در اقیانوس وجود خواهد داشت.

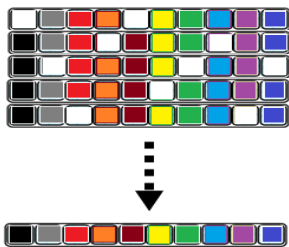
• **ابر:** همانند اقیانوس ولی در مقیاس کوچکتر می‌باشد. در ابر ریز قطرات تبخیر شده از اقیانوس که شایستگی بیشتری برای انتخاب دارند قرار خواهند گرفت. شکل ۳ نشان دهنده یک ابر و همچنین یک اقیانوس (با ابعاد بزرگتر) می‌باشد.



شکل ۳- نمایی از یک ابر (و یا اقیانوس) که از ۱۰ نوع ریزقطره تشکیل شده است.

• **گودال:** یک قطره پس از شکل‌گیری در ابر، توسط الگوریتم تپه نوردی در نزدیک‌ترین بهینه محلی به خود قرار خواهد گرفت. این بهینه محلی یک گودال نامیده می‌شود. هر گودال با توجه به عمق (fitness) خود دارای یک گنجایش قرارگیری قطرات می‌باشد که

۲- **تبخیر**: در تبخیر، در دور اول از اجراء ابر با استفاده از ریز قطرات موجود در اقیانوس بصورت تصادفی مقداردهی می شود. در دورهای بعدی از اجراء ریز قطرات انتخاب شده از اقیانوس برای تبخیر بر اساس یک اولویت انتخاب می شوند. این اولویت به این صورت می باشد که هر چه مقدار یک ریزقطره (از یک نوع) به مقدار ریزقطره موجود در رودخانه (با همان نوع) نزدیکتر باشد، احتمال انتخاب آن بیشتر خواهد بود. روشی که در این طرح برای اولویت دهی استفاده شده است، انتخاب یک تعداد تصادفی ریزقطره از هر نوع، از اقیانوس و سپس انتخاب ریزقطره ای از بین آنها که به ریزقطره موجود در رودخانه با همان نوع نزدیکتر باشد، می باشد. در این طرح، تعداد ریزقطره انتخاب شده از اقیانوس از قانون نما^۱ پیروی می کند. با پیروی از قانون نما در انتخاب تعداد ریزقطرات از اقیانوس، تعداد قطرات انتخابی کمتر، برای تعداد بیشتری انتخاب می شوند و همچنین برعکس. این امر موجب عدم قرار گیری جمعیت تولید شده در بهینه محلی در اوایل اجراء الگوریتم و همچنین شناسایی اطراف بهینه محلی توسط انتخاب تعداد زیاد قطرات انتخابی از اقیانوس می باشد. شکل ۵ نشان دهنده تعداد ریزقطره انتخاب شده از اقیانوسی با ظرفیت ۵۰۰ از هر نوع از ریزقطره، برای ۱۰۰۰۰۰ بار تکرار می باشد. همانطور که در شکل ۵ نشان داده می شود تعداد انتخاب ریزقطره از هر نوع از اقیانوس، برای مقادیر کمتر بیشتر می باشد و این امر باعث عدم همگرایی زودرس در جمعیت تولید شده و عدم به دام افتادن در بهینه محلی می شود



شکل ۶- تشکیل یک قطره از طریق الحاق ۱۰ ریزقطره در یک ابر که شامل ۱۰ نوع ریزقطره می باشد.

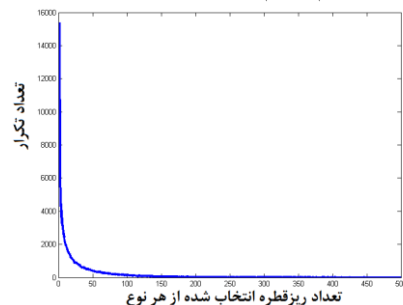
تعداد قطره تولید شده در ابر برابر با مجموع $\frac{1}{3}$ از اندازه ابر و یک عدد تصادفی از $\frac{1}{3}$ از اندازه ابر می باشد که بصورت زیر می باشد:

$$n2 = \frac{C}{3} + \text{Rand}(\frac{C}{3}) \quad (2)$$

$n2$ برابر تعداد قطره انتخابی و C برابر با اندازه ابر می باشد. لازم به ذکر می باشد که علت انتخاب معادله (۲) برای تعداد قطره انتخابی در ابر، کاملاً تجربی می باشد و پس از آزمایشات زیاد بر روی توابع آزمون مختلف بدست آمده است.

۴- **جستجوی محلی**: قطرات تشکیل شده در ابرها، پس از ریزش باید در بهینه محلی خود جای بگیرند. برای این منظور برای هر قطره ریزشی یک بردار سرعت در نظر گرفته می شود که مقدار اولیه آن، برابر با طول قدم اولیه ای می باشد که از ورودی دریافت می شود. در هر تکرار بردار جابجایی یک قطره با بردار سرعت آن جمع می شود و سپس به مقدار بردار سرعت آن اضافه می شود. معادله حرکت یک قطره بصورت معادله

۲- **تبخیر**: در تبخیر، در دور اول از اجراء ابر با استفاده از ریز قطرات موجود در اقیانوس بصورت تصادفی مقداردهی می شود. در دورهای بعدی از اجراء ریز قطرات انتخاب شده از اقیانوس برای تبخیر بر اساس یک اولویت انتخاب می شوند. این اولویت به این صورت می باشد که هر چه مقدار یک ریزقطره (از یک نوع) به مقدار ریزقطره موجود در رودخانه (با همان نوع) نزدیکتر باشد، احتمال انتخاب آن بیشتر خواهد بود. روشی که در این طرح برای اولویت دهی استفاده شده است، انتخاب یک تعداد تصادفی ریزقطره از هر نوع، از اقیانوس و سپس انتخاب ریزقطره ای از بین آنها که به ریزقطره موجود در رودخانه با همان نوع نزدیکتر باشد، می باشد. در این طرح، تعداد ریزقطره انتخاب شده از اقیانوس از قانون نما^۱ پیروی می کند. با پیروی از قانون نما در انتخاب تعداد ریزقطرات از اقیانوس، تعداد قطرات انتخابی کمتر، برای تعداد بیشتری انتخاب می شوند و همچنین برعکس. این امر موجب عدم قرار گیری جمعیت تولید شده در بهینه محلی در اوایل اجراء الگوریتم و همچنین شناسایی اطراف بهینه محلی توسط انتخاب تعداد زیاد قطرات انتخابی از اقیانوس می باشد. شکل ۵ نشان دهنده تعداد ریزقطره انتخاب شده از اقیانوسی با ظرفیت ۵۰۰ از هر نوع از ریزقطره، برای ۱۰۰۰۰۰ بار تکرار می باشد. همانطور که در شکل ۵ نشان داده می شود تعداد انتخاب ریزقطره از هر نوع از اقیانوس، برای مقادیر کمتر بیشتر می باشد و این امر باعث عدم همگرایی زودرس در جمعیت تولید شده و عدم به دام افتادن در بهینه محلی می شود



شکل ۵- توزیع پراکنده، تعداد ریز قطرات انتخاب شده از اقیانوس

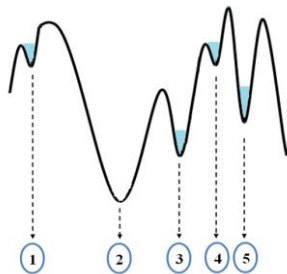
یک راه بسیار ساده برای دستیابی به توزیع قانون نمایی، استفاده از رابطه معادله (۱) می باشد که عبارت است از ۴ بار متوالی انتخاب تصادفی از اندازه اقیانوس.

$$n1 = \text{Rand}(\text{Rand}(\text{Rand}(\text{Rand}(S)))) + \alpha; \quad (1)$$

1. ^۱ Power law

چه اندازه گنجایش یک گودال بیشتر باشد با احتمال بیشتری برای دفعات بیشتر در اجرای الگوریتم موثر خواهد بود (حذف نخواهد شد) و همچنین با احتمال بیشتری جزء گودال انتخابی خواهد شد که آب‌های سرازیری به سمت آن حرکت می‌کنند.

به طور مثال، ۵ گودال انتخاب می‌شوند و عمل شناسایی و حذف گودال‌ها در دو دور از اجرا نشان داده خواهد شد. شکل ۷ (a) نشان دهنده پنج گودال در دور اول از اجرای الگوریتم می‌باشد که ۴ گودال ۱، ۳، ۴، ۵ شناسایی شده است و هر کدام دارای مقداری آب می‌باشند (در هر دور به میزان تصادفی از میانگین گنجایش گودال‌های فعال در آن دور، به آب‌های موجود در آن گودال‌ها، آب اضافه می‌شود). همانطور که مشاهده می‌شود ظرفیت گودال‌های ۱ و ۴ به علت گنجایش کم پر شده است و باید سرریز صورت گیرد. شکل ۷ (b) نشان دهنده عمل سرریز گودال‌های ۱ و ۴ می‌باشد. آب سرریزی در گودال شماره ۱، می‌تواند به سمت گودال‌های ۳ و ۵ (به علت داشتن ظرفیت بیشتر) (گودال ۲ هنوز شناسایی نشده است)) سرازیر شود. با فرض اینکه گودال ۳ برای قرار گیری آب سرریز شده از گودال یک انتخاب شود، آب سرریزی در طی حرکت به سمت گودال ۳، وارد گودال ۲ می‌شود (گودال ۲ شناسایی می‌شود) و چون ظرفیت گودال ۲ از مقدار آب سرریزی بیشتر است، آب سرریزی در گودال ۲ باقی می‌ماند و از ادامه حرکت باز می‌ایستد. سپس به اندازه آب سرریزی به مقدار آب موجود در گودال ۲ (چون به تازگی شناسایی شده است مقدار آب موجود در آن برابر صفر می‌باشد) اضافه می‌شود و گودال ۱ از بین گودال‌های فعال حذف می‌شود.



(a) نمایش ۵ گودال در دور اول از اجرای الگوریتم

(۶) می‌باشد. این مراحل به تعداد β بار تکرار می‌شود که β از ورودی دریافت می‌شود.

۵- جریان آب در سطح: همانطور که بیان شد هر گودال دارای یک گنجایش می‌باشد که هر چه عمق یک گودال بیشتر باشد (دارای *fitness* بهتری باشد) دارای گنجایش بیشتری خواهد بود. در هر دور از اجرا، گنجایش تمامی گودال‌ها بصورت زیر محاسبه خواهد شد:

$$C_i = \frac{(f_i - f_{min}) \times Const}{(f_{max} - f_{min})} \quad (3)$$

که در آن f_i *fitness* گودال i ام می‌باشد و f_{max} اندازه بزرگترین *fitness* بین گودال‌ها و f_{min} ، اندازه کوچک‌ترین *fitness* بین گودال‌ها می‌باشد. $Const$ یک عدد ثابت می‌باشد که نشان دهنده حداکثر ظرفیت یک گودال برای یک تابع هدف می‌باشد و C_i نیز گنجایش گودال i ام می‌باشد. در هر دور از اجرا به یک مقدار تصادفی از میزان میانگین گنجایش گودال‌ها، به میزان قطرات موجود در هر گودال اضافه می‌شود. اگر میزان قطرات موجود در یک گودال از گنجایش آن گودال بیشتر بود آن گودال از بین گودال‌ها حذف می‌شود و آب موجود در این گودال در سطح جاری می‌شود. جهت حرکت آب سرازیر شده به سمت یک گودال با گنجایش بیشتر نسبت به گودالی که آب از آن سرازیر شده است، می‌باشد. این گودال از بین گودال‌های موجود با گنجایش بیشتر بصورت تصادفی با یک اولویت مشخص می‌شود. این اولویت به اینصورت می‌باشد که هرچه میزان گنجایش یک گودال بیشتر باشد احتمال انتخاب آن گودال بیشتر خواهد بود. معادله (۴) بیانگر احتمال انتخاب یک گودال بین گودال‌های موجود با گنجایش بیشتر می‌باشد

$$S_i = \frac{C_i}{\sum_{j=0}^k C_j} \quad (4)$$

که در آن S_i برابر احتمال انتخاب یک گودال، C_i گنجایش گودال i ام و k برابر با تعداد گودال‌هایی می‌باشد که دارای گنجایش بیشتر نسبت به گودال شامل آب سرازیر شده می‌باشد. پس از انتخاب گودال، آب سرازیر شده به سمت گودال انتخابی حرکت می‌کند. اگر در طی مسیر حرکت، به گودالی برخورد که دارای گنجایش بیشتری نسبت به گودال مبدا بود، به میزان آب سرازیر شده، به مقدار آب آن گودال اضافه می‌شود و حرکت قطع خواهد شد، در غیر اینصورت آنقدر حرکت می‌کند که به گودال انتخاب شده برسد و به میزان آب سرازیر شده به مقدار آب آن گودال اضافه می‌شود. در پایان حرکت گودال مبدا حذف خواهد شد. هر

موقعیت فعلی قطره هستند. $X_{select}(t)$ موقعیت فعلی گودالی با ظرفیت بیشتر نسبت به گودالی که آب از آن سرازیر شده است، می باشد و R ضریب افزایش سرعت می باشد. در یک فرآیند تکراری با ارزیابی معیار شایستگی و برزاندگی قطرات برای بهبود بهترین تجربه گروهی (رودخانه) در طی حرکت، دو تعامل بین قطرات صورت می گیرد: یکی کاندید شدن یک گودال (که یک قطره نماینده آن می باشد) برای قبول آب های سرریزی که از طرف گودال های دیگر می آید و دیگری افزایش شایستگی ریزقطرات موجود در اقیانوس بوسیله رودخانه می باشد.

$$X_i = (X_{i1}, X_{i2}, \dots, X_{iD}) \quad (5)$$

$$V_i = (V_{i1}, V_{i2}, \dots, V_{iD})$$

$$V_i(t+1) = R \times V_i(t) \quad \text{و} \quad (6)$$

$$X_i(t+1) = X_i(t) + V_i(t+1)$$

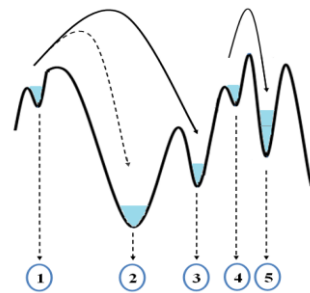
$$V_i(t+1) = V_i(t) + \text{Rand}(X_{select}(t) - X_i(t)) \quad \text{و} \quad (7)$$

$$X_i(t+1) = X_i(t) + V_i(t+1)$$

به این ترتیب در هر دور از اجرای الگوریتم تعدادی از قطرات وارد اقیانوس می شوند و در تشکیل نسل جدید نقش ایفا می کنند و تعدادی دیگر به نسل بعدی منتقل می شوند که می توانند به عنوان راهنما برای قطرات جدید نقش ایفا کنند.

۳- پیاده سازی طرح پیشنهادی و آنالیز نتایج

در این بخش به ارائه دستاوردها و نتایج شبیه سازی و مقایسه الگوریتم ارائه شده با الگوریتم های دیگر در توابع آزمون استاندارد رایج که در مقایسه روش های مختلف بهینه سازی استفاده می شوند؛ می پردازیم. آنالیزهای صورت گرفته شامل نتایج اجرای الگوریتم برای هر تابع آزمون بر حسب تعداد اجرا، میانگین و انحراف معیار جواب های بدست آمده از هر دور از اجرای الگوریتم برای رسیدن به جواب بهینه سراسری و مقایسه با الگوریتم های تکاملی دیگر در رسیدن به بهترین جواب و انجام آزمون های آماری بر روی آنها می باشد. ۶ تابع آزمون برای انجام مقایسه در این بخش مورد استفاده قرار گرفتند. این توابع در مرجع [21] استفاده شده اند که همه آنها مینیمم سازی هستند و در پایین لیست شده اند.



(b) سرریز آب از دو گودال با ظرفیت محدود در همان دور اول و شناسایی گودالی با ظرفیت بهتر

شکل ۷- چگونگی عمل جستجو در فضای حالت توسط آب های سرریزی (پر شدن ظرفیت گودال ها و سرریزی آن ها)

۶- ورود جریان آب رودخانه به اقیانوس: همانطور که بیان شد در هر دور از اجرا گودال با بیشترین گنجایش رودخانه نامیده می شود. آب موجود در رودخانه در هر دور وارد اقیانوس می شود و به اینصورت ریز قطرات تبخیر شده از اقیانوس توسط ریزقطرات رودخانه مقداره می شوند. یک رودخانه تا زمانی که گودالی با گنجایش بیشتر نسبت به آن یافت نشد به حیات خود ادامه می دهد و اگر گودالی با گنجایش بیشتر نسبت به رودخانه یافت شد آن گودال به عنوان رودخانه انتخاب می شود و رودخانه قبلی نیز به عنوان یک گودال عادی در نظر گرفته می شود. در پایان β دور از اجرای الگوریتم، رودخانه بهترین جواب الگوریتم برای مساله خواهد بود. شکل ۸ فلوجارت الگوریتم مبتنی بر چرخه آب را نشان می دهد. در این الگوریتم بر خلاف الگوریتم پرندگان، ذرات (قطرات) دارای حافظه برای نگهداری بهترین موقعیت خود در حین جستجو نیستند و دانش راه حل های خوب برای هر قطره حفظ نمی شود و تنها بهترین راه حل از بین تمامی راه حل ها برای دور بعدی از جستجو حفظ خواهد شد. هر قطره قابلیت دستیابی به مکان فعلی هر قطره دیگر را دارد، بنابراین در هر لحظه مکان بهترین قطره مشخص می باشد. اگر فضای جستجوی مسئله D بعدی باشد موقعیت و سرعت، قطره α م در جمعیت را می توان با بردار D بعدی به صورت معادله ۵ نمایش داد. رابطه ۶ بیانگر سرعت و موقعیت حرکت قطرات در هنگام جستجوی محلی و رابطه ۷ بیانگر سرعت و موقعیت حرکت قطرات در هنگام حرکت در سطح (سرریز آب از گودال) می باشد. در معادله های ۶ و ۷، $V_i(t)$ سرعت و $X_i(t)$

Particle swarm optimization(PSO)	[12]
Self-adaptive learning based PSO(SLPSO)	[22]

برای اجرا در توابع آزمون، اندازه اقیانوس برابر ۳۰۰ و اندازه ابر برابر ۸۰ و تعداد تکرار الگوریتم برابر ۲۰۰ در نظر گرفته شده است. همچنین مقادیر اولیه را برای بردار سرعت قطرات و همچنین آبهای سرریزی برابر ۰،۰۰۱ و ضرب افزایش سرعت بردار سرعت برابر ۲ و حداکثر عدم موفقیت الگوریتم در یافتن بهینه محلی برای هر قطره ۱۰۰۰ در نظر گرفته شده است.

۳-۱- چگونگی رفتار الگوریتم در رسیدن به بهینه سراسری در روند اجرا

در این بخش الگوریتم ارائه شده به تعداد ۲۰۰ بار برای هر تابع آزمون تکرار می‌شود و میانگین جواب‌های بدست آمده و همچنین انحراف معیار در این تعداد تکرار محاسبه و ارائه می‌شود. این مراحل برای ۵۰ بار تکرار شده است و میانگین داده‌های بدست آمده محاسبه شده است. جدول ۳ نشان دهنده نتایج اجرای الگوریتم ارائه شده بر روی توابع آزمون و میانگین جواب‌های بدست آمده و انحراف معیار آنها می‌باشد.

جدول ۳- میانگین هزینه و انحراف معیار اجرای الگوریتم ارائه شده در توابع آزمون در ۲۰۰ اجرای اول

انحراف معیار در ۲۰۰ اجرای اول	میانگین هزینه توابع در ۲۰۰ تکرار اول (برای ۵۰ بار اجرای مستقل)
F1	-12197.8422
F2	2.0278
F3	13.3245
F4	-95.6551
F5	-75.9376
F6	9828503072

همانطور که در جدول ۳ مشاهده می‌شود برای ۲۰۰ اجرای اول الگوریتم، میانگین جواب‌های بدست آمده بسیار مطلوب می‌باشند) با توجه به اینکه دوره‌های اول از اجرا جواب‌هایی بسیار دور از جواب بهینه سراسری تولید می‌کنند) و انحراف معیار میانگین از جواب بهینه سراسری عدد نسبتاً کوچکی می‌باشد.

$$F1 = \sum_{i=1}^n -x_i \sin(\sqrt{|x_i|})$$

$$F2 = \sum_{i=1}^n (x_i^2 - 10 \cos(2\pi x_i) + 10)$$

$$F3 = \frac{1}{4000} \sum_{i=1}^n x_i^2 - \prod_{i=1}^n \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$$

$$F4 = - \sum_{i=1}^n \sin(x_i) \sin^{20}\left(\frac{i \times x_i}{\pi}\right)$$

$$F5 = \frac{1}{n} \sum_{i=1}^n (x_i^4 - 16x_i^2 + 5x_i)$$

$$F6 = \sum_{i=1}^{n-1} [100(x_i^2 - x_{i+1})^2 + (x_i - 1)^2]$$

بعضی از خصوصیات اصلی توابع آزمون ارائه شده در جدول ۱ لیست شده اند. این خصوصیات شامل فضای جستجوی، ابعاد و مقدار مینیمم سراسری هر تابع آزمون می‌باشد.

جدول ۱- خصوصیات اصلی توابع آزمون

تک هدفه

ابعاد	مقدار مینیمم سراسری	فضای جستجو	نام تابع
30	-12569.5	$[-500, 500]^n$	F1
30	0	$[-5.12, 5.12]^n$	F2
30	0	$[-600, 600]^n$	F3
100	-99.2784	$[0, \pi]^n$	F4
100	-78.33236	$[-5, 5]^n$	F5
100	0	$[-5, 10]^n$	F6

الگوریتم‌هایی که در این بخش مورد مقایسه قرار گرفته اند در جدول

۲ لیست شده اند.

جدول ۲- الگوریتم‌های مورد ارزیابی قرار گرفته

نام الگوریتم	مرجع
Evolutionary programming with adaptive Levy mutation (ALEP)	[18]
orthogonal genetic algorithm with quantization(OGA/Q)	[19]
Level-Set Evolution Algorithm(LEA)	[21]
bacterial swarm optimization(BSO)	[3]
ABFOA1	[5]
group search optimizer(GSO)	[10]
Genetic Algorithm(GA)	[11]

F3	ABFOA1	2.5372	46.96521✓
	GA	0.6509	-
	PSO	20.7863	12.80622✓
	GSO	1.0179	-
	SLPSO	0	24.74433✓
	WATER CYCLE	0	-7.56929✓
			N/A
			0
F4	ALEP	0.024	2.3935✓
	LEA	6.104×10^{-16}	4.80686✓
	BSO	0.2565	-8.9770✓
	ABFOA1	0.1914	-13.0065✓
	GA	1.0038	-
	PSO	0.2323	66.58127✓
	GSO	0.030792	-2.84728✓
	WATER CYCLE	0.00181	1.772877
		0.05120	4.62758✓
F5	OGA/Q	-92.83	-
	LEA	-93.01	1291.214✓
	WATER CYCLE	-99.3373	-
			1339.407✓
			0
F6	OGA/Q	-78.300029	-2.85771✓
	LEA	-78.310	-1.2919
	WATER CYCLE	-78.31822	0
F6	LEA	0.5609	-
	BSO	15.471	25.23492✓
	ABFOA1	4.572	-
	GA	338.5616	41.20395✓
	PSO	37.3582	-10.5543✓
	GSO	49.8359	-6.62243✓
	WATER CYCLE	0.04906	-
			8.218194✓
		-11.667✓	
		0	

✓ مقادیر T برای درجه آزادی ۴۹ در سطح معنی دار ۰،۰۵، دارای اهمیت می باشد.

۲-۳- مقایسه با الگوریتم های تکاملی دیگر بر اساس دست یابی

به بهینه سراسری

در این بخش به تحلیل و مقایسه بهترین جواب های تولید شده برای الگوریتم چرخه آب در مقایسه با الگوریتم های تکاملی مختلف دیگر می پردازیم. بدین منظور الگوریتم چرخه آب را به طور مستقل ۵۰ بار با تعداد تکرار ۲۰۰، اجرا می کنیم و سپس میانگین و انحراف معیار بهترین جواب ها را محاسبه می کنیم. سر انجام مقدار آزمون T را بین الگوریتم چرخه آب و الگوریتم های دیگر محاسبه می کنیم. داده های مربوط به میانگین بهترین جواب ها و انحراف معیار الگوریتم های مختلف دیگر از مراجع [21],[10],[5],[22] بدست آمده اند. جدول ۴ خلاصه ای از نتایج مقایسه مقایسه الگوریتم چرخه آب با الگوریتم های تکاملی مختلف دیگر را نشان می دهد. در این جدول، تمامی مقادیر T که برای درجه آزادی ۴۹ و سطح معنی دار ۰،۰۵ دارای اهمیت می باشند با علامت ✓ مشخص شده اند. مقادیر منفی T بیانگر عملکرد بهتر الگوریتم چرخه آب نسبت به الگوریتم دیگر در یک تابع آزمون می باشد و همچنین برعکس.

جدول ۴- مقایسه نتایج بدست آمده از الگوریتم چرخه آب با الگوریتم های مختلف دیگر بر حسب دستیابی به بهینه سراسری

اندازه آزمون T	میانگین بهترین جواب ها در ۵۰ اجرای مستقل	نام الگوریتم مورد مقایسه	نام تابع آزمون
F1	-11619.0473	ALEP	
	115.4762✓	OGA/Q	
	-	LEA	
	126.0509✓	GA	
	-	PSO	
	126.4090✓	GSO	
	-	WATER CYCLE	
	11.40832✓	-12569.5	
-44.3643✓			
-			
3.744579✓			
0			
F2	-19.9834✓	ALEP	
	N/A	OGA/Q	
	-4.28549✓	LEA	
	-24.6852✓	BSO	
-			



[2] F.V.D. Bergh, A.P. Engelbrecht, A study of particle swarm optimization particle trajectories, *Information Sciences* 176, 937–971, 2006.

[3] A. Biswas, S. Dasgupta, S. Das, and A. Abraham, “Synergy of PSO and bacterial foraging optimization: A comparative study on numerical benchmarks,” in *Proc. 2nd Int Symp. Hybrid Artificial Intell. Syst. (HAIS) Advances Soft Computing Ser.*, vol. 44. Germany, Springer-Verlag, *Innovations in Hybrid Intelligent Systems*, ASC, pp. 255–263, 2007.

[4] O. Cordon, S. Damas, J. Santamarı, A fast and accurate approach for 3D image registration using the scatter search evolutionary algorithm, *Pattern Recognition Letters* 27, 1191–1200, 2006.

[5] S.Dasgupta, S.Das, A.Abraham, “Adaptive Computational Chemotaxis in Bacterial Foraging Optimization: An Analysis”, *IEEE Trans. Evol. Comput*, vol 13, pp: 919 – 941, 2009.

[6] U.S. Department of the Interior, url: “ga.water.usgs.gov/edu/watercyclefarsi.html”

[7] M. Dorigo, V. Maniezzo, A. Colorni, The ant system: optimization by a colony of cooperating agents, *IEEE Transactions on Systems, Man, and Cybernetics – Part B* 26 (1), 29–41, 1996.

[8] W. Du, B. Li, Multi-strategy ensemble particle swarm optimization for dynamic optimization, *Information Sciences* 178, 3096–3109, 2008.

[9] I. Ellabib, P. Calamai, O. Basir, Exchange strategies for multiple ant colony system, *Information Sciences* 177, 1248–1264, 2007.

[10] S. He, Q.H.Wu, J. R. Saunders, “Group Search Optimizer: An Optimization Algorithm Inspired by Animal Searching Behavior”, *IEEE Trans. Evol. Comput*, vol 13, pp: 973 – 990, 2009.

[11] J. H. Holland, *Adaption in Natural and Artificial Systems*. Ann Arbor, MI, 1975.

[12] J. Kennedy and R. C. Eberhart, “Particle swarm optimization,” in *Proc. IEEE Int. Conf. Neural Netw.*, vol. 4. Piscataway, NJ: IEEE Press, pp. 1942–1948, 1995.

[13] A. Kalinlia, N. Karabogab, Artificial immune algorithm for IIR filter design, *Engineering Applications of Artificial Intelligence* 18, 919–929, 2005.

[14] C. Karakuzu, Fuzzy controller training using particle swarm optimization for nonlinear system control, *ISA Transactions* 47, 2, 229–239, 2008.

[15] J. Kennedy, R.C. Eberhart, Particle swarm optimization, in: *Proceedings of IEEE International*

همانطور که در جدول ۴ مشاهده می شود الگوریتم چرخه آب در اکثر مقایسات، دارای عملکرد بهتری نسبت به الگوریتم های تکاملی دیگر در توابع آزمون مختلف در دستیابی به بهترین جواب می باشد.

۴- نتیجه گیری

آنچه در این نوشتار مورد بررسی قرار گرفت ارائه یک الگوریتم بهینه سازی جدید بر مبنای مدل سازی ریاضی فرایند چرخه آب در طبیعت بود. به طور خلاصه، این الگوریتم، شامل یک مجموعه ای از مقادیر اولیه برای هر پارامتر از مسئله می باشد. در هر دور از اجرای این الگوریتم، برای هر پارامتر از مسئله با یک اولویت مقادیری انتخاب می شوند و تشکیل یک مجموعه کوچکتر را می دهند. سپس با در نظر گرفتن شایستگی مقادیر، با ترکیب مقادیر مربوط به هر پارامتر، چندین جواب برای مسئله تولید می شود. سپس هر جواب از مسئله با استفاده از الگوریتم جستجوی محلی، در بهینه محلی خود قرار می گیرد. هر بهینه محلی با توجه به میزان برانزنگی خود، یک مقدار گنجایش دریافت می کند که میزان ماندگاری آن جواب در تعداد اجراهای متوالی و همچنین احتمال کاندید شدن برای حرکت دیگر جواب ها به سمت خود را بیان می کند. در هر دور از اجرا، جواب هایی از مسئله که باید از مسئله خارج شوند به سمت جوابی با شایستگی بهتر حرکت می کنند و در طول مسیر اگر به شایستگی بهتری برخورد نکردند، به بهینه محلی انتخاب شده خود می رسند. هر دور یک جواب مسئله با بیشترین شایستگی ثابت می ماند و در پایان هر دور مقادیر هر پارامتر از این جواب مسئله وارد مجموعه مقادیر اولیه می شوند. بعد از پایان n دور از الگوریتم، بهترین جواب دور n ام جواب بهینه مسئله می باشد. نتایج آزمایش روش پیشنهادی بر روی توابع آزمون مختلف نشان می دهد که الگوریتم معرفی شده در یافتن جواب بهینه توابع کاملاً موفق عمل می کند. مقایسه نتایج حاصله توسط الگوریتم مطرح شده با روش های رایج بهینه سازی از برتری نسبی این الگوریتم حکایت دارد. الگوریتم معرفی شده به عنوان نسخه اولیه یک الگوریتم مبتنی بر فرآیند چرخه آب می باشد. بنابراین مطمئناً می توان اصلاحاتی در آن نیز ایجاد نمود.

۵- مراجع

[1] A. Badr, A. Fahmy, A proof of convergence for ant algorithms, *Information Sciences* 160, 267–279, 2004.



چهارمین کنفرانس مهندسی برق و الکترونیک ایران

4th Iranian Conference on Electrical and Electronics Engineering (ICEEE2012)

دانشگاه آزاد اسلامی کتاباد- ۷ و ۸ و ۹ شهریور ماه ۱۳۹۱



- [19] Y. W. Leung and Y. P. Wang, "An orthogonal genetic algorithm with quantization for global numerical optimization," *IEEE Trans. Evol. Comput.*, vol. 5, no. 1, pp. 41–53, Feb. 2001.
- [20] K.S. Tang, K.F. Man, S. Kwong, Q. He, Genetic algorithms and their applications, *IEEE Signal Processing Magazine* 13, 6, 22–37, 1996.
- [21] Y.Wang, C.Dang, "An Evolutionary Algorithm for Global Optimization Based on Level- Set Evolution and Latin Squares" . *IEEE Trans. Evolutionary Computation*, 11(5): 579-595 ,2007.
- [22] Y.Wang, B.Li, T.Weise, J.Wang, B.Yuan, Q.Tian, " Self-adaptive learning based particle swarm optimization". *Inf. Sci*, 181, 20, 4515-4538, 2011.
- Conference on Neural Networks, vol. 4, pp. 1942-1948, 1995.
- [16] M.Vafaei Jahan, M.R. Akbarzadeh Totonchi, "From Local Search to Global Conclusions: Migrating Spin Glass-based Distributed Portfolio Selection", *Journal of IEEE Transaction on Evolutionary Computation*, No. 14, pp:591-601, August 2010.
- [17] S. Kirkpatrick, C.D. Gelatto, M.P. Vecchi, Optimization by simulated annealing, *Science*, 220, 671–680, 1983.
- [18] C. Y. Lee and X. Yao, "Evolutionary programming using mutations based on the levy probability distribution," *IEEE Trans. Evol. Comput.*, vol. 8, no. 1, pp. 1–13, Feb. 2004.