

یک روش تطبیقی جدید برای بهینه سازی دسته جمعی ذرات با استفاده از الگوریتم فرهنگی در محیط های پویا

سجاد هواسی^۱، محمدرضا میبیدی^۲، سمانه رحیمی^۳

^۱ گروه مهندسی کامپیوتر، دانشگاه آزاد اسلامی واحد قزوین، قزوین

Sajad_havaasi@yahoo.com

^۲ عضو هیئت علمی دانشکده مهندسی کامپیوتر و فن آوری اطلاعات دانشگاه صنعتی امیرکبیر، تهران

mmeybodi@aut.ac.ir

^۳ گروه مهندسی برق، دانشگاه آزاد اسلامی واحد قزوین، قزوین

Smmh_rahimi@yahoo.com

چکیده - تعداد قابل توجهی از مسائل بهینه سازی پویا هستند، پویایی در مسائلی که در دنیای واقعی رخ می دهد بیشتر دیده می شود. در این نوع مسائل، بهینه دائماً در حال تغییر است. برای مواجه شدن با این نوع مسائل الگوریتم باید بتواند بهینه سراسری را یافته و آن را در طول زمان دنبال کند.

در این مقاله، یک روش تطبیقی جدیدی برای بهینه سازی دسته جمعی ذرات با استفاده از دانش معیار الگوریتم فرهنگی پیشنهاد شده است. در این روش با توجه به الگوریتم *mQSO* روشی برای بهبود *gbest* هر دسته با استفاده از مفهوم ذرات کوانتوم مطرح گردیده که تنظیم تطابق یکی از پارامترهای این روش به عهده دانش معیار الگوریتم فرهنگی است. این پارامتر، فاصله ی اطلاعاتی مهمی را که از بهترین فرهنگ ذرات هر دسته استخراج می شود، برای هر بعد از ذره تعیین می کند.

نتایج حاصل از این روش پیشنهادی، بر روی معیار قله های متحرک ارزیابی شده و با نتایج حاصل از چندین الگوریتم معتبر مورد مقایسه قرار گرفته است. نتایج بدست آمده نشان دهنده ی کارایی خوب این روش نسبت به الگوریتم *mQSO* و سایر الگوریتم هاست.

کلید واژه- الگوریتم بهینه سازی دسته جمعی ذرات، الگوریتم فرهنگی، فضای باور، محیط های پویا، معیار قله های متحرک.

مسائل محسوب می شوند، ولی مشکل این الگوریتم ها این است که سرانجام به یک بهینه همگرا می شوند. بنابراین، تنوع لازم را در محیط از دست می دهند. در صورت تغییر در محیط، همگرا شدن به نقطه بهینه جدید در صورت امکان بسیار زمان گیر است. الگوریتم بهینه سازی دسته جمعی ذرات یا PSO^1 ، در سال ۱۹۹۵ توسط Kennedy و Eberhart معرفی شد [۱]. این الگوریتم، یک الگوریتم بهینه سازی است که از روی زندگی جمعی و گروهی پرندگان الهام می گیرد تا به راه حل بهینه برسد. تاکنون نسخه های متعددی از آن برای بهینه سازی در محیط های ایستا و پویا معرفی شده است. یک مشکل اساسی این الگوریتم از

۱- مقدمه

از آنجا که اکثراً مسائل دنیای واقعی پویا هستند و محیط در حال تغییر، ممکن است بهینه اش نیز تغییر کند، در این مسائل معمولاً چندین بهینه محلی وجود دارد که یکی از آنها بهینه سراسری است. بنابراین، الگوریتم ها باید سعی در دنبال کردن بهینه سراسری داشته باشند. از طرفی اکثر الگوریتم ها مانند الگوریتم فرهنگی و الگوریتم بهینه سازی دسته جمعی ذرات به سمت بهینه حرکت می کنند، به نظر انتخاب خوبی برای این

شرح مختصری از نوع خاص آن یعنی mQSO و کارهای انجام شده قبلی برای انطباق با محیط‌های پویا مطرح می‌شود. در بخش سوم به الگوریتم‌های فرهنگی و قسمت‌های مختلف آن پرداخته می‌شود. در بخش چهارم الگوریتم پیشنهادی طرح می‌گردد. در بخش پنجم نتایج آزمایش‌ها مورد بررسی قرار می‌گیرند و بخش نهایی به بیان نتیجه‌گیری می‌پردازد.

۲- الگوریتم بهینه‌سازی دسته جمعی ذرات

این الگوریتم با یک گروه از جواب‌های تصادفی شروع به کار می‌کند. سپس برای یافتن جواب بهینه در فضای مسئله با به‌روز کردن موقعیت و سرعت هر ذره به جستجو می‌پردازد. هر ذره به صورت چند بعدی با دو مقدار V_{ij} و X_{ij} که به ترتیب معرف مکان و سرعت مربوط به بعد j ام از i امین ذره هستند تعریف می‌شود. در هر مرحله از حرکت جمعیت، هر ذره با توجه به دو مقدار بهترین، به‌روز می‌شود. اولین مقدار، بهترین جواب از لحاظ شایستگی است که تاکنون برای هر ذره به طور جداگانه بدست آمده است. این مقدار بهترین تجربه فردی است که $pbest$ نامیده می‌شود. مقدار بهترین دیگر که توسط PSO بدست می‌آید، بهترین مقادیر است که تاکنون توسط تمام ذره‌ها در میان جمعیت بدست آمده است. این مقدار بهترین تجربه گروهی است که $gbest$ نامیده می‌شود. پس از یافتن دو مقدار $pbest$ و $gbest$ هر ذره سرعت و مکان جدید خود را با دو رابطه زیر به‌روز می‌کند:

$$X_{i,j}^{(t+1)} = X_{i,j}^t + v_{i,j}^{(t+1)} \quad (1)$$

$$v_{i,j}^{(t+1)} = w * v_{i,j}^t + c_1 r_1 (pbest_{i,j}^t - X_{i,j}^t) + c_2 r_2 (gbest^t - X_{i,j}^t) \quad (2)$$

به طوری که w وزن اینرسی، c_1 و c_2 ضرایب شتاب و r_1 و r_2 اعداد تصادفی در بازه (۰،۱) هستند.

در الگوریتم mQSO [۴] که گونه خاصی از الگوریتم بهینه‌سازی دسته‌جمعی ذرات است، تعداد از قبل تعیین شده‌ای دسته وجود دارد که هر دسته وظیفه یافتن و تعقیب کردن یک قله را بر عهده دارد. زیرا در محیط‌های پویا بهینه‌های محلی ممکن است پس از تغییر محیط به بهینه سراسری تبدیل شوند. بنابراین، تا جای امکان لازم است همه‌ی آنها زیر پوشش قرار

دست دادن تنوع پس از مدتی است که باعث همگرایی زودرس و گیر افتادن در بهینه محلی می‌شود و مشکل دیگر به‌خصوص برای محیط‌های پویا، بی‌استفاده شدن حافظه پس از تغییر محیط است.

الگوریتم‌های فرهنگی^۲ توسط Reynolds در سال ۱۹۹۴ مطرح شد. این الگوریتم از تکامل فرهنگ انسان‌ها و تأثیر پذیری افراد یک جامعه از آن و اثر آن در ایجاد نسل‌های آینده الهام گرفته شده است. این الگوریتم از حوزه دانش برای فرآیند جستجو استفاده می‌کند. اضافه شدن حوزه دانش در بهبود کارایی الگوریتم‌های تکاملی مؤثر است و فرآیند جستجو را هوشمندانه‌تر می‌کند. درواقع، اضافه شدن حوزه دانش مکانیزمی برای کاهش فضای جستجو از طریق هرس کردن قسمت‌های نامناسب آن است [۲]. این الگوریتم دارای دانش‌های مختلفی در فضای باور خویش است که به امر جستجو کمک می‌کند. در بخش‌های آتی به به آنها می‌پردازیم.

در این مقاله، یک الگوریتم بهینه‌سازی مبتنی بر الگوریتم mQSO پیشنهاد می‌گردد که هر دسته از جمعیت، فضای باور یا فرهنگ مربوط به خود را دارند.

این فضای باور که از دانش معیار بهره می‌برد، مقدار پارامتر با ارزشی را بصورت تطابقی که همان فاصله اطلاعاتی هر بعد از ذره است، در اختیار هر دسته قرار می‌دهد. هر دسته باید بر اساس تعداد تلاش‌های مشخصی بهترین ذره‌اش را بهبود دهد. این پارامتر اطلاعاتی مهم را الگوریتم فرهنگی در اختیار الگوریتم mQSO قرار می‌دهد. الگوریتم پیشنهادی تمام نیازهای محیط‌های پویا را برآورده می‌کند.

الگوریتم پیشنهادی بر روی سناریوهای مختلف معیار قله‌های متحرک (MPB)^۳ [۳]، که از معروف‌ترین بنچمارک‌های محیط‌های پویا است به کار رفته و کارایی آن با الگوریتم‌های mQSO [۴]، CellularPso [۵] و Adaptive mQSO [۶] مقایسه شده است. معیار مقایسه، خطای برون خطی^۴ بوده که یکی از معیارهای اصلی مقایسه الگوریتم‌های طراحی شده برای محیط‌های پویا است. نتایج آزمایش‌ها نشان می‌دهند که الگوریتم پیشنهادی از کارایی قابل قبولی برخوردار است.

ادامه این مقاله بدین ترتیب سازماندهی شده است: در بخش دوم مروری بر الگوریتم بهینه‌سازی دسته‌جمعی ذرات (PSO) و

Lung و Dumitrescu، دو جمعیت با اندازه‌های برابر را برای دنبال کردن بهینه متحرک در محیط‌های پویا معرفی کردند، که یکی از این جمعیت‌ها وظیفه حفظ تنوع در فضای جستجو و جمعیت دیگر مسئول پیدا کردن بهینه سراسری است. حفظ تنوع در جمعیت، با بکار بردن الگوریتم تکامل تفاضلی مبتنی بر ازدحام^۸ که برای بهینه‌سازی چند هدفه به کار می‌رود، صورت می‌پذیرد. در حالی که Pso، مستقیماً برای پیدا کردن بهینه سراسری مورد استفاده قرار می‌گیرد [۸].

Moyed Daneshyari و Gary G.yen در مرجع [۹]، روش ترکیبی مبتنی بر الگوریتم فرهنگی و الگوریتم دسته‌جمعی ذرات را معرفی کردند که در آن روش، الگوریتم فرهنگی به عنوان چارچوبی اطلاعاتی برای Pso نقش‌آفرینی می‌کند. این روش پیشنهادی از دانش‌های موجود در الگوریتم فرهنگی برای تشخیص تغییرات محیط، زمان مهاجرت بین دسته‌های ذرات و دافعه‌ی بین ذرات برای حفظ تنوع و انتخاب ذرات در سه سطح شخصی، محلی و سراسری بهره می‌برد.

۳- الگوریتم فرهنگی

الگوریتم فرهنگی یک سیستم دو جمعیتی است، که دو فضای جستجو را ارائه می‌دهد. یکی فضای جمعیت^۹ است و دیگری فضای باور^{۱۰} که فرهنگ فضای جمعیت را ارائه می‌کند [۲].

فضای باور، در واقع اطلاعات فرهنگ افراد را مدل می‌کند. هر دو فضا به صورت موازی با هم تعامل دارند و از هم تاثیر می‌پذیرند. برای ارتباط دادن بین این دو فضا دو پروتکل ارتباطی تعریف می‌شود. یکی برای انتخاب گروهی از افراد تا فضای باور را شکل دهند و دیگری روشی برای تأثیر این فضای باور بر روی تولید افراد در فضای جمعیت است.

به‌طور کلی الگوریتم فرهنگی به شرح زیر عمل می‌کند: در هر نسل ابتدا افراد مانند الگوریتم ژنتیکی در فضای جمعیت وارد شده و توسط تابع شایستگی^{۱۱} ارزیابی می‌شوند. سپس توسط تابع پذیرش^{۱۲} افرادی را که مناسب شکل دادن به فضای باور هستند، انتخاب می‌کند و تجربیات افراد پذیرفته شده، برای ساختن و تغییر فضای باور بکار برده می‌شود. فرهنگ ایجاد شده در فضای باور، بر روی تکامل جمعیت در

بگیرند. ارتباط بین دسته‌ها به دو صورت محلی بر اساس rexccl و سراسری بر اساس rconv است [۴]. هنگامی که تمام دسته‌ها همگرا شده باشند و تعداد قله‌ها بیش از تعداد دسته‌ها باشد، دسته‌ای که کمترین مقدار شایستگی را دارد در فضای مسأله مقداردهی اولیه می‌شود. همین‌طور اگر دو دسته فاصله کمتر از rexccl از هم داشته باشند، دسته‌ای که شایستگی کمتری دارد، مقداردهی اولیه می‌شود. زیرا اگر هر دسته یک قله را زیر پوشش قرار دهد کفایت می‌کند. بنابراین، هنگامی که دو دسته به یک قله همگرا شدند، باید تنها یکی از آنها باقی بماند و دیگری در فضای مسأله مقداردهی اولیه شود تا شانس این را داشته باشد که به قله‌ای که هنوز زیر پوشش هیچ دسته‌ای قرار نگرفته همگرا شود. همچنین هنگامی که همه دسته‌ها همگرا شده باشند، هیچ دسته‌ای نمی‌تواند از قله‌ای که تحت پوشش آن است خارج شود زیرا تنوع، گستردگی دسته و سرعت ذرات کمتر از آن است که دسته‌ها بتوانند از بهینه‌هایی که در آن‌ها قرار گرفته‌اند به بیرون حرکت کنند. در این حالت مشکل از آنجا منجر می‌شود که وقتی تعداد قله‌ها بیش از دسته‌ها باشد و همه دسته‌ها همگرا شده باشند، برخی از قله‌ها هیچ وقت تحت پوشش قرار نمی‌گیرند. بنابراین در مرجع [۴] پیشنهاد شد که در این شرایط، دسته‌ای که بدترین مقدار شایستگی را دارد، باید در محیط مسأله مقداردهی اولیه شود تا شانس پیدا کردن قله‌های دیگر که احتمالاً بلندتر هستند نیز به وجود آید. در الگوریتم پیشنهادی ما از تکنیک‌های چنددستگی^۵، انحصار^۶ و ضد همگرایی^۷ استفاده شده است.

۲-۱- مروری بر کارهای گذشته در محیط‌های پویا

تحقیقات متعددی در زمینه بهینه‌سازی Pso در محیط‌های پویا انجام گرفته است. Blackwell و Branke، در مراجع [۴] و [۷]، روش چنددستگی ذرات، استفاده از مفاهیم ضد همگرایی و انحصار با هدف پوشش قله‌های مختلف و نیز ایجاد تنوع در محیط را معرفی کردند.

هاشمی و میبیدی در مرجع [۵]، با بهره‌گیری از تعاملات محلی در اتوماتای سلولی و تقسیم کردن جمعیت ذرات در داخل سلول‌های اتوماتای سلولی و مشخص کردن تعداد معینی ذره در هر سلول سعی بر حفظ تنوع در جمعیت نمودند.

۳-۲- فضای باور

در فضای باور، تجربیات عمومی شده افراد خوب از فضای جمعیتی، بدست آمده و این تجارب در طی نسل و نسل‌های بعدی شکل گرفته و ذخیره می‌شود [۲][۱۰].
دانش‌های مختلفی فضای باور را تشکیل می‌دهند که به شرح زیر است:

- دانش موقعیتی^{۱۵}
- دانش معیار^{۱۶}

دانش‌های دیگری نیز به مرور برای کاربردهای مختلف ایجاد شد [۱۱] که در این مقاله از دو مورد بالا استفاده می‌شود.

۱-۱-۱- دانش موقعیتی

این قسمت از دانش برای بهینه‌سازی توابع اعداد حقیقی در محیط‌های ایستا معرفی شد، که شامل تعدادی از افراد خوب است که بهترین آن‌ها برای تأثیر گذاری در تولید نسل بعدی، در نظر گرفته می‌شود.

۱-۲-۱- دانش معیار

این دانش، مجموعه بازه‌های خوب و امیدبخش را که از مجموعه‌ای از ذرات خوب استخراج شده است، برای هر بعد از مسأله نگهداری می‌کند. این دانش طبق رابطه زیر است:

$$Norm = \{X_1, \dots, X_D\} \quad (3)$$

در این جا، D معرف تعداد ابعاد مسأله است و هر X_i به صورت زیر تعریف می‌شود:

$$X_i = [l_i \ u_i \ \quad l_i \ \quad U_i] \quad (4)$$

در این جا l_i و u_i به ترتیب حد بالا و حد پایین بعد i ام هستند، l_i و U_i مقدار تابع شایستگی در آن حدود است [۱۲].
[۱۳].

دانش معیار طبق روابط زیر به‌روز می‌شود:

$$l_i^{t+1} = \begin{cases} x_{j,i} & \text{if } x_{j,i} \leq l_i^t \text{ Or } f(\overline{x_j}) < l_i^t \\ l_i^t & \text{Otherwise} \end{cases} \quad (5)$$

$$u_i^{t+1} = \begin{cases} x_{k,i} & \text{if } x_{k,i} \geq u_i^t \text{ Or } f(\overline{x_k}) < u_i^t \\ u_i^t & \text{Otherwise} \end{cases} \quad (6)$$

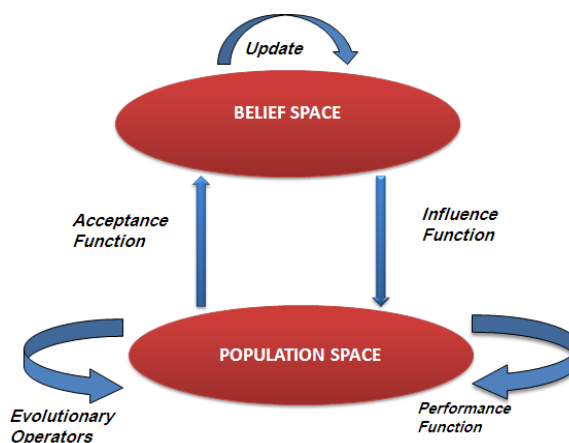
فضای جمعیت تأثیر می‌گذارد. این تأثیر با تغییر دادن عملگر جهش^{۱۳} و اعمال آن عملگر در تولید فرزندان صورت می‌گیرد. شبه کد این الگوریتم در شکل ۱ آورده شده است [۱۰].
اجزای الگوریتم فرهنگی عبارتند از: فضای جمعیت، فضای باور، تابع پذیرش و تابع تأثیر^{۱۴}، که هر کدام در بخش‌های آینده شرح داده خواهد شد. برای درک بیشتر الگوریتم چارچوب آن در شکل ۲ نشان داده شده است.

```

Begin
t=0;
Initialize POP(t); /* Initialize population*/
Initialize BLF(t); /* Initialize Belief Space*/
repeat
    Evaluate Pop(t);
    Vote (BLF(t),Accept(POP(t)));
    Adjust (BLF(t));
    Evolve (POP(t), Influence(BLF(t)));
    t=t+1;
    Select POP(t) from POP(t-1);
Until (termination condition achieved)
End

```

شکل ۱: شبه کد الگوریتم فرهنگی



شکل ۲: چارچوب الگوریتم فرهنگی

۳-۱- فضای جمعیت

این فضا در واقع فضای اصلی جمعیت است و با مقداردهی اولیه کار خود را شروع کرده و استخراج فرهنگ و ذخیره‌ی آن در فضای باور در این قسمت انجام می‌گیرد.

۴- الگوریتم پیشنهادی

در این الگوریتم هر دسته از دو نوع ذره، یعنی ذرات معمولی و ذرات کوانتوم همانند مرجع [۴] تشکیل شده است. اما عملکرد ذرات کوانتوم با آنچه در مرجع [۴] ارائه گردیده، متفاوت است. در ابتدا هر دسته از ذرات و نیز فضای باور مقداردهی اولیه می شوند. ذرات معمولی در هر تکرار از اجرای الگوریتم و بر اساس سرعت قبلی، بهترین تجربه شخصی و بهترین تجربه گروهی خود را به روز می کنند. در الگوریتم پیشنهادی، ذرات موجود در تمام دسته ها موقعیت خود را به روز کرده و همزمان دانش معیار در فضای باور هم ساخته می شود. پس از اجرای الگوریتم و به روز شدن موقعیت Gbest، دانش معیار برای هر بعد از ذره، شعاع یا فاصله اطلاعاتی را بر حسب $R(j)=ui-li$ محاسبه می کند.

الگوریتم پیشنهادی با استفاده از یک ذره کوانتوم، یک موقعیت را در شعاع $R(j)$ در اطراف Gbest هر دسته در نظر می گیرد و با استفاده از رابطه زیر تا Try_number بار موقعیت Gbest را بهبود می دهد.

$$Q_{i,j} = Gbest_{i,j} + R(j) \times N(-1,1) \quad (11)$$

که در رابطه (۱۱)، $Q_{i,j}$ مؤلفه j ام از موقعیت ذره کوانتوم دسته نام است. در این رابطه $R(j)$ نقش مهمی را در کارایی ذره کوانتوم بازی می کند. به این معنا که، اگر مقدار آن کوچک باشد، توانایی جستجوی محلی الگوریتم بسیار بالا می رود. اما باعث کاهش طول گام حرکت در هر تکرار و کاهش سرعت همگرایی می شود. در صورتی که مقدار این شعاع زیاد باشد، احتمال یافتن موقعیت های بهتر در آن کاهش می یابد و از قدرت جستجوی محلی کم می شود. در الگوریتم پیشنهادی این شعاع به صورت تطابقی توسط فضای باور الگوریتم فرهنگی تنظیم می شود.

در واقع مقدار این شعاع بر اساس دانش معیار هر بعد از ذره و فاصله اطلاعاتی مفید و امید بخش، تعیین می شود. در الگوریتم پیشنهادی، برای شناسایی تغییر در محیط، یک نقطه در ابتدای اجرای الگوریتم در فضای مسئله در نظر گرفته شده و در پایان هر تکرار، شایستگی آن سنجیده می شود. در صورتی که تغییری در محیط رخ داده باشد، مقدار شایستگی این نقطه تغییر می کند.

$$L_i^{t+1} = \begin{cases} f(\bar{x}_j) & \text{if } x_{j,i} \leq l_i^t \text{ Or } f(\bar{x}_j) < l_i^t \\ L_i^t & \text{Otherwise} \end{cases} \quad (7)$$

$$U_i^{t+1} = \begin{cases} f(\bar{x}_k) & \text{if } x_{k,i} \geq u_i^t \text{ Or } f(\bar{x}_k) < u_i^t \\ U_i^t & \text{Otherwise} \end{cases} \quad (8)$$

۳-۳- تابع پذیرش

این تابع افراد شایسته را در هر نسل برای شکل دهی فضای باور، انتخاب می کند. در مرجع [۱۴]، تعداد افراد انتخابی به صورت پویا طبق رابطه زیر پیشنهاد شد:

$$|A| = \left\lfloor \text{popsize} * p_{\text{accept}} + \frac{\text{popsize}(1 - p_{\text{accept}})}{g} \right\rfloor \quad (9)$$

که در آن p_{accept} پارامتری تجربی است و در این مقاله، ۰.۲ فرض شده است. g شمارنده نسل است و در محیط های پویا پس از تغییر محیط، g به ۱ مقداردهی می شود. popsize نیز تعداد کل فضای جمعیت است.

۳-۴- تابع تأثیر

در فضای باور، برای تغییر دادن افراد و نزدیک کردن آن ها به باور سراسری (بهینه کل) بکار برده می شوند که این تغییرات با استفاده از تابع تأثیر، تحقق می یابد [۲].

فضای باور با استفاده از عملگر جهش بر روی فضای جمعیت تأثیر می گذارد، این تأثیر از دو راه ممکن است، یکی اندازه جهش و دیگری جهت جهش.

در این مقاله، تلاش شده است با توجه به دانش موقعیتی و دانش معیار از مراجع [۲][۱۳]، تابع تأثیر استخراج شود. این جهش طبق رابطه زیر است:

$$x'_{j,i} = \begin{cases} x_{j,i} + |size(I_i).N(0,1)| & \text{if } x_{j,i} < l_i \text{ Or } x_{j,i} < s_i \\ x_{j,i} - |size(I_i).N(0,1)| & \text{if } x_{j,i} > u_i \text{ Or } x_{j,i} > s_i \\ x_{j,i} + size(I_i).N(0,1) & \text{Otherwise} \end{cases} \quad (10)$$

که در رابطه بالا $size(I_i) = u_i - l_i$ و $N(0,1)$ متغیر تصادفی با توزیع نرمال با میانگین ۰ و انحراف معیار ۱ است. در این مقاله، تابع تأثیری را که براساس جهش بالا انجام می شود $Influence_Ns$ می نامیم.

الگوریتم پیشنهادی در شکل ۳ آورده شده است:

```

For each swarm i
Initialize BLFi \ Belief Space for each swarm
For each Particle j
    initialize  $x_{i,j}$ ,  $v_{i,j}$ 
    Pbesti,j =  $x_{i,j}$ 
End for
Gbest = arg min f(Pbesti,j)
End for
Initialize Test_point
Repeat:
for each swarm i
for each particle j
    Update  $X_{i,j}(t+1)$  and Pbesti,j(t+1)
End for
Update(BLFi) \ according to Normative Knowledge
Calculate R(j) based on Normative Knowledge
Gbest = arg min f(Pbesti,j)
End for
for each swarm i
For cnt=1 to Try_number
    Update  $Q_{i,j}$  based on equation (11)
    If  $f(Q_{i,j}) > f(Gbest_i)$  THEN
        Gbesti =  $Q_{i,j}$ 
    End for
End for
Evaluate Test_point
IF new value is different from last iteration THEN
    change environment
    Reevaluate each particle attractor
    Update swarm attractor
    Update(BLFi) \ according to Normative Knowledge
    Calculate R(j) based on equation (11)
    Execute exclusion and anti-convergence[4]
Until stopping criterion is met
    
```

شکل ۳: شبه کد الگوریتم پیشنهادی.

۵- ارزیابی

ارزیابی کارایی الگوریتم پیشنهادی، این الگوریتم همراه با سه الگوریتم شناخته شده به نام‌های mQSO [۴]، CellularPso و Adaptive mQSO [۶] بر روی MPB مورد مقایسه قرار گرفته‌اند. نتایج با توجه به پارامترهای جدول ۱، که با نام سناریو ۲ در مسأله MPB معروف است، آورده شده است [۳]. تنها پارامترهای متفاوت، تعداد قله‌ها و فرکانس تغییر است که برای ارزیابی بهتر بین الگوریتم‌ها از ۱ قله تا ۲۰۰ قله در فرکانس‌های تغییر ۵۰۰، ۱۰۰۰، ۲۵۰۰ و ۵۰۰۰ در نظر گرفته شده است.

جدول ۱: پارامترهای MPB

M تعداد قله‌ها
فرکانس تغییر
بین ۱ تا ۲۰۰
۵۰۰، ۱۰۰۰، ۲۵۰۰ و ۵۰۰۰

میزان تغییر ارتفاع ۷۰
میزان تغییر عرض ۱۰
شکل قله Cone
تابع اولیه ندارد
s طول جابجایی ۱۰
N تعداد ابعاد ۵
محدوده مکانی قله‌ها [۰.۰, ۱۰۰.۰]
محدوده پارامتر ارتفاع [۳۰۰, ۷۰۰]
محدوده پارامتر عرض [۱, ۱۲]
مقدار ارتفاع اولیه قله‌ها ۵۰

در آزمایش‌های الگوریتم پیشنهادی، ۱۰ دسته وجود دارد و تعداد ذرات در هر دسته برابر ۲۰ و مقدار Try_number نیز برابر ۵ و مقدار این پارامتر برای بهترین دسته برابر ۵۰ در نظر گرفته شده است. تنظیمات انحصار و ضد همگرایی مطابق مرجع [۴] انجام شده است.

آزمایش‌ها ۳۰ بار تکرار شده‌اند و متوسط مقدار خطای برون خطی و خطای استاندارد [۳] الگوریتم پیشنهادی به همراه سه الگوریتم دیگر، در جداول ۲، ۳، ۴ و ۵ در فرکانس‌های ۵۰۰، ۱۰۰۰، ۲۵۰۰ و ۵۰۰۰ با تعداد قله‌های مختلف ارائه شده است. در این جداول، خطای استاندارد در کنار خطای برون خطی درون پرانتز و برای مقایسه بهتر، نتایج برتر پررنگ‌تر نمایش داده شده است.

جدول ۲: مقایسه خطای برون خطی و خطای استاندارد در فرکانس ۵۰۰.

M \	MQSO10(5,5q)	Adaptive mQSO	Cellular PSO	الگوریتم پیشنهادی
۱	۳۳.۶۷ (۳.۴۲)	۳۰.۲ (۰.۲۲)	۱۳.۴۶ (۰.۷۳)	۹.۶۱ (۰.۱۲)
۵	۱۱.۹۱ (۰.۷۶)	۵.۷۷ (۰.۲۲)	۹.۶۳ (۰.۴۹)	۷.۵۲ (۰.۳۴)
۱۰	۹.۶۲ (۰.۳۴)	۵.۳۷ (۰.۲۲)	۹.۳۵ (۰.۳۷)	۶.۲۵ (۰.۳۵)
۲۰	۹.۰۷ (۰.۲۵)	۶.۸۲ (۰.۲۲)	۸.۸۴ (۰.۲۸)	۶.۸۰ (۰.۲۸)
۳۰	۸.۸۰ (۰.۲۱)	۷.۱۰ (۰.۲۲)	۸.۸۱ (۰.۲۴)	۷.۹۸ (۰.۴۲)
۴۰	۸.۵۵ (۰.۲۱)	۷.۰۵ (۰.۲۲)	۸.۹۴ (۰.۲۴)	۷.۴۲ (۰.۳۰)
۵۰	۸.۷۲ (۰.۲۰)	۸.۹۷ (۰.۲۲)	۸.۶۲ (۰.۲۳)	۸.۱۹ (۰.۴۱)
۱۰۰	۸.۵۴ (۰.۲)	۷.۳۴ (۰.۲۲)	۸.۵۴ (۰.۲۱)	۸.۱۴ (۰.۶۲)
۲۰۰	۸.۱۹ (۰.۱۸)	۷.۴۸ (۰.۲۲)	۸.۲۸ (۰.۱۸)	۷.۲۰ (۰.۲۱)

جدول ۳: مقایسه خطای برون خطی و خطای استاندارد در فرکانس ۱۰۰۰.

M \	MQSO10(5,5q)	Adaptive mQSO	Cellular PSO	الگوریتم پیشنهادی
۱	۱۸۶۰ (۱.۶۳)	۲.۳۳ (۰.۳۱)	۶.۷۷ (۰.۳۸)	۴.۶۷ (۰.۲۳)
۵	۶.۵۶ (۰.۳۸)	۲.۹۰ (۰.۳۲)	۵.۳۰ (۰.۳۲)	۴.۱۱ (۰.۳۸)
۱۰	۵.۷۱ (۰.۲۲)	۴.۵۶ (۰.۴۰)	۵.۱۹ (۰.۱۳)	۳.۶۷ (۰.۱۲)
۲۰	۵.۸۵ (۰.۱۵)	۵.۳۶ (۰.۴۷)	۵.۲۳ (۰.۱۸)	۴.۱۱ (۰.۱۶)
۳۰	۵.۸۱ (۰.۱۵)	۵.۲۰ (۰.۳۸)	۵.۳۳ (۰.۱۶)	۴.۸ (۰.۲۲)
۴۰	۵.۷۰ (۰.۱۴)	۵.۲۵ (۰.۲۷)	۵.۶۱ (۰.۱۶)	۵.۳۱ (۰.۲۶)
۵۰	۵.۸۷ (۰.۱۳)	۶.۰۶ (۰.۱۴)	۵.۵۵ (۰.۱۴)	۵.۲۸ (۰.۱۷)
۱۰۰	۵.۸۳ (۰.۱۳)	۴.۷۷ (۰.۴۵)	۵.۵۷ (۰.۱۲)	۴.۷۹ (۰.۳۱)
۲۰۰	۵.۵۴ (۰.۱۱)	۵.۷۵ (۰.۲۶)	۵.۵۰ (۰.۱۲)	۴.۹۵ (۰.۲۸)

جدول ۵: مقایسه خطای برون خطی و خطای استاندارد در فرکانس ۵۰۰.

M \	MQSO10(5,5q)	Adaptive mQSO	Cellular PSO	الگوریتم پیشنهادی
۱	۳.۸۲ (۳.۴۲)	۰.۵۱ (۰.۰۴)	۲.۵۴ (۰.۱۶)	۱.۳۸ (۰.۳۹)
۵	۱.۹۰ (۰.۳۵)	۱.۰۹ (۰.۱۴)	۱.۷۲ (۰.۱۷)	۱.۲۰ (۰.۲۳)
۱۰	۱.۹۱ (۰.۳۱)	۱.۵۱ (۰.۱۰)	۱.۷۶ (۰.۱۳)	۱.۴۱ (۰.۱۷)
۲۰	۲.۵۶ (۰.۲۳)	۲.۰۰ (۰.۱۵)	۲.۵۹ (۰.۱۰)	۲.۴۲ (۰.۳۳)
۳۰	۲.۶۸ (۰.۲۹)	۲.۱۹ (۰.۱۷)	۲.۹۵ (۰.۱۲)	۲.۴۱ (۰.۱۸)
۴۰	۲.۶۵ (۰.۲۱)	۲.۲۸ (۰.۱۲)	۳.۱۱ (۰.۱۰)	۲.۴۳ (۰.۱۶)
۵۰	۲.۶۳ (۰.۲۰)	۲.۴۳ (۰.۱۳)	۳.۲۲ (۰.۱۱)	۲.۴۲ (۰.۰۸)
۱۰۰	۲.۵۲ (۰.۲)	۲.۶۸ (۰.۱۲)	۳.۳۹ (۰.۱۰)	۲.۳۸ (۰.۰۶)
۲۰۰	۲.۳۶ (۰.۱۷)	۲.۶۲ (۰.۱۰)	۳.۳۶ (۰.۰۹)	۲.۳۴ (۰.۰۲)

دلیل این که نتایج مربوط به Adaptive mQSO در قله های پایین تر برتری محسوسی دارد، این موضوع است که در این الگوریتم تعداد دسته ها به صورت تطبیقی و براساس تعداد قله های یافت شده تنظیم می شود. بدین صورت این الگوریتم می تواند براساس قله های یافت شده تعداد ارزیابی ها از تابع شایستگی را کنترل کند و باعث به هدر نرفتن تعداد ارزیابی ها شود. با این وجود الگوریتم پیشنهادی توانسته با آن الگوریتم رقابت کند و این نشان از مؤثر بودن ذره کوانتوم پیشنهادی است که در آن شعاع تطبیقی بر اساس فضای باور الگوریتم فرهنگی ایجاد شده است.

۶- نتیجه گیری

در این مقاله، یک الگوریتم براساس بهینه سازی تجمعی ذرات (PSO) برای محیط های پویا با رعایت کلیه الزامات محیط پویا و نیز ذره کوانتوم ابتکاری پیشنهاد شد تا بتواند اطراف بهترین ذره از هر دسته را با تعداد دفعات مشخصی جستجو کند. این شعاع جستجو را دانش معیار از فضای باور الگوریتم فرهنگی فراهم می نماید.

نتایج بر روی معیار حرکت قله ها با چند روش شناخته شده ی دیگر مورد مقایسه قرار گرفت. نتایج آزمایش ها نشان داد که الگوریتم پیشنهادی از کارایی قابل قبولی برخوردار است. برای کار آینده و بهبود این الگوریتم می توان تعداد دسته ها و فضای

همان طور که مشاهده می شود، الگوریتم پیشنهادی در فرکانس های کمتر، یعنی ۵۰۰ و ۱۰۰۰، عملکرد بهتری نسبت به سه الگوریتم دیگر دارد. علت این برتری این است که الگوریتم پیشنهادی تنوع مناسبی را ایجاد می کند. البته در اینجا نقش ذره کوانتوم کم رنگ تر است، زیرا محیط با سرعت بیشتری تغییر می کند و فرصت برای آن ذره برای بدست آوردن موقعیت های بهتر اطراف gbest کمتر بوجود می آید.

جدول ۴: مقایسه خطای برون خطی و خطای استاندارد در فرکانس ۲۵۰۰.

M \	MQSO10(5,5q)	Adaptive mQSO	Cellular PSO	الگوریتم پیشنهادی
۱	۷.۶۴ (۰.۶۴)	۰.۸۷ (۰.۱۱)	۴.۱۵ (۰.۲۵)	۲.۱۷ (۰.۲۳)
۵	۳.۲۶ (۰.۲۱)	۲.۱۶ (۰.۱۹)	۲.۸۵ (۰.۲۴)	۱.۹۸ (۰.۱۱)
۱۰	۳.۱۲ (۰.۱۴)	۲.۴۹ (۰.۱۰)	۲.۸۲ (۰.۱۴)	۱.۷۹ (۰.۳۳)
۲۰	۳.۵۸ (۰.۱۳)	۲.۷۳ (۰.۱۱)	۳.۴۱ (۰.۱۴)	۳.۱۸ (۰.۱۵)
۳۰	۳.۶۳ (۰.۱۰)	۳.۲۴ (۰.۱۸)	۳.۶۲ (۰.۱۲)	۲.۹۵ (۰.۲۳)
۴۰	۳.۵۵ (۰.۱۰)	۳.۲۷ (۰.۲۳)	۳.۸۴ (۰.۱۲)	۳.۲۲ (۰.۱۷)
۵۰	۳.۶۳ (۰.۱۰)	۳.۶۸ (۰.۱۵)	۳.۸۶ (۰.۱۰)	۳.۳۳ (۰.۱۰)
۱۰۰	۳.۵۸ (۰.۰۸)	۳.۵۳ (۰.۱۴)	۴.۱۰ (۰.۱۱)	۳.۱۹ (۰.۱۲)
۲۰۰	۳.۳۰ (۰.۰۶)	۳.۰۷ (۰.۱۲)	۳.۹۷ (۰.۱۰)	۲.۹۷ (۰.۰۳)

الگوریتم پیشنهادی در این فرکانس هم نتایج قابل رقابتی با دیگر الگوریتم ها دارد و نتایجی نزدیک به نتایج الگوریتم برتر Adaptive mQSO دارد. در اینجا نقش ذره کوانتوم پیشنهادی محسوس تر است، زیرا الگوریتم توانسته عملکرد بهتری نسبت به mQSO ارائه کند.

باور هر دسته را به صورت تطبیقی در نظر گرفت، به طوری که به ازای هر قله یافت شده در فضای مسأله یک دسته ایجاد شود.

مراجع

- [1] Kennedy, J., Eberhart, R.C., "Particle Swarm Optimization", *IEEE International Conference on Neural Networks*, Piscataway, NJ, vol. IV, pp. 1942-1948, 1995.
- [2] P. Engelbrecht, *Computational Intelligence An Introduction*, Wiley, Second Edition, 2007.
- [3] <http://www.aifb.unikarlstruhe.de/~jbr/MovPeaks/>
- [4] T. Blackwell and J. Branke, "Multiswarm, Exclusion, and Anti-Convergence in Dynamic Environment", in *IEEE Transaction on Evolutionary Computation*, vol. 10, No. 4, pp. 459-472, 2006.
- [5] B. Hashemi and M. R. Meybodi, "Cellular PSO: A PSO for Dynamic Environments", in *Advances in Computation and Intelligence, Lecture Notes in Computer Science*, vol. 5821, pp. 422-433, 2009.
- [6] T. Blackwell and J. Branke, "Particle Swarms for Dynamic Optimizatin Problems", in *swarm Intelligence*, pp. 193-217, 2008.
- [7] Blackwell, T., Branke, J., "Multi-Swarm Optimization in Dynamic Environments", *Applications of Evolutionary Computing*, 489-500, 2004.
- [8] Lung, R.I., Dumitrescu, D., "A Collaborative Model for Tracking Optima in Dynamic Environments", *IEEE Congress on Evolutionary Computation*, pp. 564-567, 2007.
- [9] M. Daneshyari and G.G. Yen, "Cultural-based particle swarm for dynamic optimization problems", *International Journal of Systems Science*, pp. 1-21, 2011, DOI=<http://dx.doi.org/10.1080/00207721.2011.605965>.
- [10] R. Reynolds, "An Introduction to Cultural Algorithms", in *Proceedings Of the 3rd Annual on Evolutionary Programming*, World Scientific, River Edge, NJ, pp. 131-139, 1994.
- [11] R. Reynolds, S. Saleem, "The Impact of Environmental Dynamic on Cultural Emergence", *Festschrift, in Honor of John Holland*, Oxford University Press, pp. 1-10, 2003.
- [12] Reynolds, R.G, Bin Peng, Mostafa Z. Ali, "The Role of Culture in the Emergence of Decision Making Roles", *International joint conference on Understanding Complex Systems*, University of Illinois-Urbana Champaign, pp. 27-42, 2005.
- [13] Y. Wu, X-L. Huang, X-Z. Gao, "A Cultural Particle Swarm Optimization Algorithm", *Sixth International Conference on Natural Computation*, pp. 2505-2509, 2010.
- [14] Saleh M. Saleem, "Knowledge-Based Solution to Dynamic Optimization Problems using Cultural Algorithms", *PhD thesis*, Wayne State University, Detroit, Michigan, 2011.

زیر نویس‌ها

¹ Particle Swarm Optimization
² Cultural Algorithms
³ Moving peak Benchmark
⁴ Offline Error
⁵ Multi Swarm