International Congress of Interdisciplinary Studies in Science and Engineering
December 2017 - Iran, Tehran
کنگره بین المللی مطالعات میان رشته ای در علوم پایه و مهندسی
آذرماه ۱۳۹۶ - تهران
Archive of SID

الگوریتم مبتنی بر آموزش و یادگیری بهبود یافته

[۱]ابراهیم علی‌زاده ، [۲] علی ملکی، [۳] حمید امیری آرا

[۱] دانشگاه صنعتی مالک اشتر، پژوهشکده‌ی علوم و فناوری دفاعی شمال، ealizadeh@mut.ac.ir
[۲] دانشگاه صنعتی مالک اشتر، پژوهشکده‌ی علوم و فناوری دفاعی شمال، ali.maleki86@yahoo.com
[۳] دانشگاه صنعتی مالک اشتر، پژوهشکده‌ی علوم و فناوری دفاعی شمال، hamid.amiriara@gmail.com

چکیده

در این مقاله یک روش بهینه‌سازی کارآمد به نام "بهینه‌ساز مبتنی بر آموزش و یادگیری" جهت یافتن جواب قطعی برای مسائل بهینه‌سازی، معرفی شده و همچنین الگوریتم بهینه‌ساز مبتنی بر آموزش و یادگیری بهبودیافته (MTLBO) برای اصلاح این الگوریتم پیشنهاد شده است. الگوریتم TLBO به دو فاز تقسیم‌بندی می‌شود، فاز اول، که شامل فراگیری از استاد می‌باشد و فاز دوم، که شاگردان از تراکنش متقابل همدیگر آموزش می‌بینند. ما در الگوریتم MTLBO برای جلوگیری از گرفتار شدن الگوریتم در اکسترمم‌های محلی که یک نقطه ضعف برای الگوریتم TLBO است، یک فاز سوم با نام فاز جهش به آن اضافه می‌نماییم و همچنین تغییراتی در روابط تاثیر معلم بر دانش‌آموزان ایجاد می‌نماییم. میزان کارایی روش MTLBO برای بسیاری از توابع محک با خصوصیات متفاوت محاسبه و نتایج با چند روش‌های جمعیت محور مقایسه گردید. نتایج نشان می‌دهد که الگوریتم بهبود یافته پیشنهادی بهبود قابل قبولی در کارایی الگوریتم TLBO مرسوم دارد.

واژه های کلیدی : الگوریتم بهینه‌سازی، دانش‌آموز، معلم، الگوریتم مبتنی بر آموزش و یادگیری بهبود یافته.

۱- مقدمه

به یافتن مقادیری از پارامترهای تابع هدف که تابع را کمینه یا بیشنه می‌نمایید، بهینه‌سازی می‌گویند. کلیه مقادیر مناسب، جواب‌های ممکن و بهترین مقادیر پارامترها، جواب بهینه تابع نامیده می‌شود[۱]. الگوریتم‌های بهینه‌سازی شامل هر دو مسئله بیشینه‌سازی و کمینه‌سازی می‌باشند. بهینه‌سازی کاربرد زیادی در مسائل مهندسی مانند تخصیص منابع، تصمیم گیری‌ها، زمان بندی‌ها و ... دارد. در روش‌های سابق، حل مسائل بهینه‌سازی نیازمند تعداد محاسبات بیشماری بود اما با معرفی الگوریتم‌های هوش جمعی تا حدی از این محاسبات کاسته شد [۲و۳]. هوش جمعی نوعی روش هوش مصنوعی[1] مبتنی بر رفتارهای جمعی است که در آن اعضای گروه به‌طور محلی با یکدیگر و با گروه در تعامل هستند و با همکاری یکدیگر راه حل‌هایی را که تقریبا به جواب نزدیک هستند می‌یابند [۴و۵].

طبیعت مثال کامل و مطلوبی برای بهینه‌سازی می‌باشد و بیشتر الگوریتم‌های تکاملی از طبیعت الهام می‌گیرند تا مسائل پیچیده را از روش‌های ساده محاسبه نمایینند. هر الگوریتم با ایجاد یک جمعیت اولیه از جواب‌های مسئله آغاز می‌گردد و به صورت مکرر از نسلی به نسل دیگر در جهت جواب بهینه حرکت می‌کند [۶].

الگوریتم مبتنی بر آموزش- یادگیری، یک روش بهینه‌سازی کارآمد می‌باشد که برای اولین بار توسط آقای رائو و همکاران معرفی شد[۷]. این روش نیز مشابه سایر تکنیک‌های بهینه‌سازی موجود یک الگوریتم برگرفته از طبیعت است و بر اساس تاثیر یک مدرس روی یادگیری کلاس کار می‌کند. این الگوریتم از یک جمعیتی از جواب‌ها برای دستیابی به جواب کلی استفاده می‌نماید.

جمعیت به عنوان گروهی از یادگیران یا دانش‌آموزان یک کلاس در نظر گرفته می‌شوند. یک معلم تلاش می‌کند تا با آموزش به یادگیران، سطح دانش کلاس را افزایش دهد و دانش‌آموز به نمره یا رتبه خوبی مطابق با توانایی خودش دست یابد. در حقیقت یک معلم خوب کسی است که دانش‌آموز خود را به سطح دانش خود برساند. معلم یک شخص با دانش بالا در جامعه بوده که علم خود را با دانش‌آموزان خود تقسیم می‌کند، به طوری که بهترین جواب (بهترین عضو جمعیت) در همان تکرار به عنوان معلم عمل می‌کند. اما لازم است به این نکته اشاره شود که شاگردان مطابق با کیفیت آموزش ارائه شده توسط مدرس و وضعیت شاگردان حاضر در کلاس دانش کسب می‌کنند. به علاوه شاگردان از تراکنش متقابل بین خودشان (بحث گروهی، ارائه‌ها و...) که به وضعیتشان کمک می‌کند آموزش می‌بینند.

[1] Swarm Intelligence

www.SID.ir

Archive of SID

در الگوریتم TLBO مرسوم با مشکل به دام افتادن الگوریتم در مینیمم محلی مواجه هستیم برای حل این مشکل در این مقاله الگوریتم مبتنی بر آموزش و یادگیری بهبود یافته[2]MTLBO پیشنهاد می‌نماییم که در آن فاز جدیدی به الگوریتم TLBO به نام فاز جهش را اضافه می‌نماییم. این فاز کمک می‌کند تا بتوان از تله مینیمم محلی فرار کرده و تنوع را در جمعیت حفظ نماییم. بر خلاف الگوریتم TLBO، الگوریتم پیشنهادی MTLBO همه‌سویی[3]جمعیت را تضمین می‌نماید. یک جمعیت در صورتی همه‌سو نامیده می‌شود که یک احتمال غیر صفر برای تولید هر راه حل از هر حالت جمعیت وجود داشته باشد.

الگوریتم MTLBO به طور خاص مخصوص مسائل پیوسته می‌باشد ولی با ایجاد تمهیدات خاص می‌توان از آن در مسائل گسسته و باینری نیز استفاده نمود.

در ادامه مقاله به صورت زیر سازماندهی شده است:

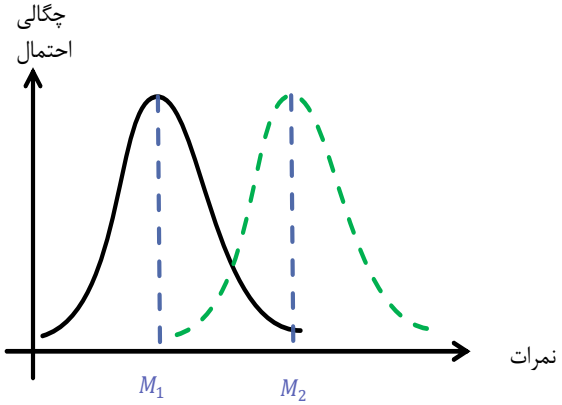
- در بخش دوم به صور کامل به بررسی فلسفه الگوریتم MTLBO می‌پردازیم.
- در بخش سوم، روابط بین معلم با دانش‌آموزان و دانش‌آموزان با یکدیگر را به صورت مدل ریاضی در الگوریتم MTLBO بیان می‌کنیم.
- در بخش چهارم، فلوچارت الگوریتم MTLBO را نمایش داده و بخش‌های مختلف آن را معرفی می‌نماییم.
- در بخش پنجم، الگوریتم MTLBO را با الگوریتم‌های متفاوت دیگر با استفاده از توابع محک گوناگون مقایسه کرده و کارایی این الگوریتم‌ها را با یکدیگر مقایسه می‌نماییم.
- در بخش آخر به نتیجه‌گیری می‌پردازیم.

۲- معرفی الگوریتم MTLBO

این الگوریتم براساس تاثیر یک معلم روی خروجی دانش‌آموزان در یک کلاس می‌باشد و به طور کلی در یک کلاس معلم فردی تعیین می‌شود که دارای مقدار بهتری می‌باشد و سطحی بالاتری نسبت به دانش‌آموزان دارد و می‌تواند دانش‌آموزان را با دانش خود سهیم نماید.

یک معلم خوب، یک میانگین بهتر برای دانش‌آموزان تولید می‌نماید. در هر مرحله و تکرار معلم کسی است، که بهترین فرد کلاس باشد و بهترین تابع هدف را دارد. البته در هر مرحله ممکن است معلم تغییر نماید. این الگوریتم جهت استفاده در امور مهندسی گزینه بسیار مناسبی می‌باشد.

در این الگوریتم از فرآیند آموزش و یادگیری که در کلاس درس اتفاق می‌افتد، الهام گرفته شده است. بدین نحو که وقتی معلم درس را ارائه می‌دهد، در نهایت ارزیابی کرده و دانش‌آموزان نیز نمره‌ای را کسب می‌نمایند. اگر نمودار شکل ۱ نمودار توزیع احتمال نمرات افراد باشد، نمودار سمت چپ مربوط به کلاس اول و نمودار سمت راست مربوط به کلاس دوم یا معلم دیگر می‌باشد. با مقایسه سطح نمرات این دو کلاس در می‌یابیم که معلم دوم موفق‌تر عمل کرده است چرا که میانگین سطح کلاس آن که M_2 می‌باشد، از میانگین کلاس اول M_1 بالاتر می‌باشد.



شکل ۱- نمودار توزیع احتمال نمرات دانش‌آموزان

[2] Modify Teaching Learning Based Optimization
[3] Ergodicity

www.SID.ir

در این نمودار برای نمایش توزیع نمرات از توزیع گوسی استفاده شده است اما می‌توان از هر توزیع دیگری نیز استفاده نمود. طبق رابطه چگالی احتمال توزیع گوسی میانگین همان M_1 و M_2 می‌باشد، در این الگوریتم به پراکندگی نمرات یا واریانس پرداخته نمی‌شود و مهم‌ترین نکته همان میانگین نمرات دانش‌آموزان کلاس می‌باشد.

۳- مدل ریاضی الگوریتم

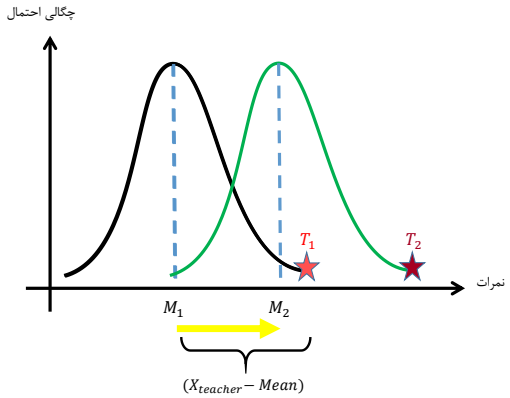
به‌طور کلی الگوریتم MTLBO به سه فاز بهینه‌سازی تقسیم می‌گردد:

- فاز معلم
- فاز دانش‌آموز
- فاز جهش

که در ادامه به معرفی این دو بخش می‌پردازیم.

۳-۱- فاز معلم

در فاز اول، معلم سعی می‌کند تا میانگین کلاس را به سطح خود برساند و سطح دانش‌آموزان در این مرحله به سمت معلم تغییر می‌نماید (شکل ۲). در این الگوریتم معلم از بین دانش‌آموزان انتخاب می‌شود، یعنی کسی که اطلاعاتش از بقیه بیشتر باشد و بهتر باشد به عنوان معلم انتخاب خواهد شد.



شکل ۲- نمودار جابجایی در فاز معلم

در نمودار شکل ۲، T_1 به عنوان معلم کلاس انتخاب شده و سعی می‌کند میانگین سطح کلاس یعنی M_1 را به سطح خودش برساند. اما در واقعیت این امکان پذیر نمی‌باشد که همه دانش‌آموزان به سطح معلم برسند بلکه نهایت به سطح میانگین جدید M_2 خواهند رسید. در اینجا به یک جامعه آماری دیگر خواهیم رسید که دارای میانگین M_2 می‌باشد و معلم جدید، همان بهترین فرد در جمعیت آماری جدید یا T_2 انتخاب خواهد گشت. در این مرحله معلم جدید یا همان T_2 سعی خواهد کرد که میانگین سطح کلاس را به خودش برساند و این روال بدین صورت ادامه خواهد داشت تا جمعیت بهتر شود و به عنوان فاز اول الگوریتم مطرح می‌شود.

اندازه گام جابجایی برابر است با $(X_{teacher} - Mean)$ که در این رابطه $X_{teacher}$ همان میانگین جدید و مطلوب می‌باشد و Mean میانگین فعلی می‌باشد. بنابراین رابطه ریاضی برای فاز معلم بدین صورت خواهد شد:

$$X_{new} = X_{old} + (X_{teacher} - Mean) \quad (۱)$$

این رابطه یک حالت ایده‌آل می‌باشد و تنوع جمعیتی ندارد. بنابراین با اضافه کردن دو ضریب تصادفی و یک ضریب وزن خواهیم داشت:

$$X_{new} = X_{old} + W * r(X_{teacher} - T_f Mean) \quad (۲)$$

که در آن r یک بردار تصادفی بین صفر و یک بوده که میزان موفقیت یک دانش‌آموز در درک مطالب یاد داده شده توسط استاد را نشان می‌دهد. بدین صورت که اگر r برابر صفر باشد یعنی دانش‌آموز از مطالبی که معلم به او آموزش داده چیزی یاد نگرفته است و اگر r برابر یک باشد یعنی تمام مطالب معلم را فراگرفته است. همچنین T_f یا ضریب معلم[4] نشان دهنده ضریب موفقیت معلم می‌باشد که $T_f \epsilon \{1,2\}$ می‌باشد. اگر ضریب

[4] Teacher Factor

www.SID.ir

Archive of SID

معلم یک باشد، یک یادگیری معمولی را خوهیم داشت ولی اگر این ضریب ۲ باشد شتاب یادگیری بیشتر خواهد گشت. همچنین در این رابطه W تابع وزن‌دهی می‌باشد و می‌تواند بصورت رابطه

$$W = W_{Max} - \frac{[(W_{Max} - W_{Min}) \times iter]}{Max_{iter}} \quad (۳)$$

تغییر نمایید، که در آن W_{max} وزن اولیه، و W_{min} وزن نهایی است و Max_{iter} تعداد تکرارها (مراحل) و $iter$ تعداد تکرارها تا لحظه کنونی (تا این مرحله) می‌باشد.

در الگوریتم MTLBO زمانی که W را با رابطه ارائه شده بصورت خطی از یک مقدار نسبتا بزرگ به یک مقدار کوچک کاهش می‌دهیم، نسبت به الگوریتم TLBO که در آن W مقداری ثابت و برابر یک دارد، عملکرد بهتری خواهیم داشت زیرا دانش آموز در ابتدا بدون توجه به معلم به جستجوی سراسری پرداخته و با گذشت چند دوره حضور در کلاس به جستجوی محلی و یادگیری از معلم می‌پردازد. در حالتی که W بزرگتر است، توانایی جستجوی سراسری بیشتر و در حالت W کوچکتر، توانایی جستجو محلی بیشتر است.

۲-۳- فاز دانش‌آموز

فاز دانش‌آموز بعد از فاز معلم اجرا می‌شود و دانش‌آموزان می‌توانند از یکدیگر نیز آموزش ببینند و بر روی یکدیگر تاثیر بگذارند و این تعامل باعث می‌شود که سطوح دانش‌آموزان ارتقا پیدا نماید. این فاز که نام دیگر آن فاز یادگیری می‌باشد همان طور که در شکل ۳ مشخص می‌باشد دانش‌آموزان با تعامل و بحث و گفتگو با یکدیگر سعی در افزایش سطح دانش هم دارند.

با توجه به شکل ۳ دو دانش‌آموز تصادفی از بین جمعیت انتخاب خواهد شد که در آن دانش‌آموز اول یعنی x_i می‌خواهد از دانش‌آموز دوم یعنی x_j آموزش ببیند، بسته به میزان نمره این دو دانش‌آموز دو حالت برای اثر پذیری دانش‌آموز x_i از دانش‌آموز x_j به وجود خواهد آمد:

- حالت اول: اگر نمرات دانش‌آموز در حال یادگیری از دانش‌آموز آموزش‌دهنده بدتر باشد.
- حالت دوم: اگر نمرات دانش‌آموز در حال یادگیری از آموزش‌دهنده بهتر باشد.

حالت اول: اگر نمرات دانش‌آموز در حال یادگیری از دانش‌آموز آموزش‌دهنده بدتر باشد.

در این حالت یک دانش‌آموز ضعیف (x_i) می‌خواهد از یک دانش‌آموز با نمرات بهتر آموزش (x_j) ببیند در نتیجه بهترین حالت این است که سطح دانش‌آموز x_i دقیقا برابر x_j گردد، که در این صورت رابطه ریاضی برای این حالت عبارت خواهد بود با $x_i^{new} = x_j$ یا:

$$x_i^{new} = x_i + (x_j - x_i) \quad (۴)$$

مشابه فاز معلم برای افزایش تنوع جمعیتی به این رابطه ایده‌آل یک ضریب تصادفی اضافم می‌نماییم:

$$x_i^{new} = x_i + \mathrm{r}(x_j - x_i) \quad (۵)$$

که در این رابطه r یک بردار تصادفی بین صفر و یک بوده که میزان موفقیت دانش‌آموز x_i در درک مطالب یاد داده شده توسط دانش‌آموز x_j را نشان می‌دهد.

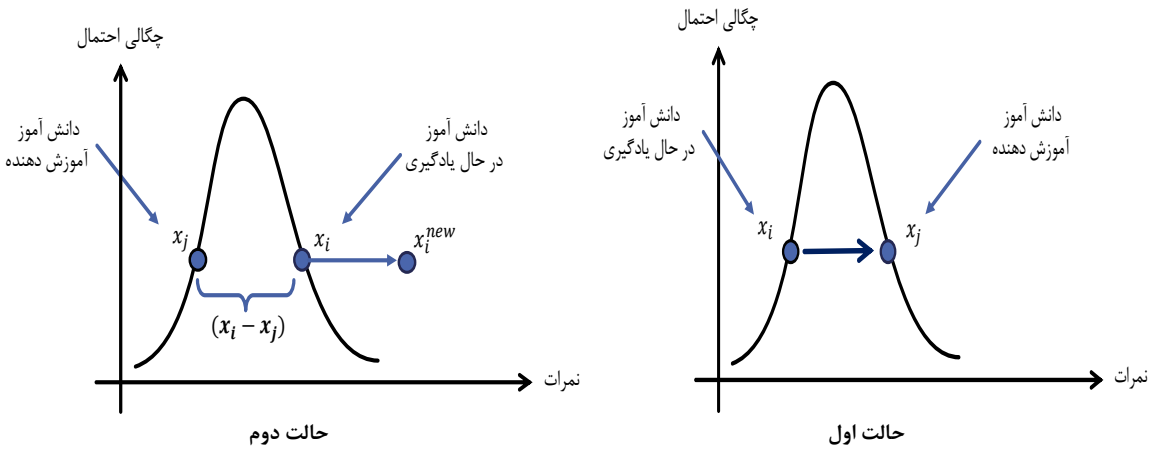
حالت دوم: اگر نمرات دانش‌آموز در حال یادگیری از آموزش‌دهنده بهتر باشد

همان طور که در شکل ۳ مشخص می‌باشد در این حالت دانش‌آموز در حال یادگیری که همان x_i می‌باشد از دانش‌آموز در حال آموزش یعنی x_j دارای نمرات بهتری می‌باشد. بدین منظور برای بهبود موقعیت x_i باید تا جای ممکن از x_j فاصله بگیرد. بنابراین به اندازه طول گام یعنی $(x_i - x_j)$ از آن فاصله می‌گیرد و رابطه ریاضی برای این حالت به صورت زیر خواهد بود:

$$x_i^{new} = x_i + \mathrm{r}(x_i - x_j) \quad (۶)$$

که در این رابطه نیز r یک بردار تصادفی بین صفر و یک می‌باشد.

www.SID.ir



شکل ۳- حالت اول: اگر نمرات آموزش‌گیرنده از آموزش‌دهنده بدتر باشد. حالت دوم: اگر نمرات آموزش‌گیرنده از آموزش‌دهنده بهتر باشد.

باید به این نکته اشاره کرد که در هر دو فاز معلم و دانش‌آموز با بدست آمدن دانش‌آموز جدید (x) مقدار تابع هدف محاسبه می‌شود و اگر این مقدار از مقدار تابع هدف دانش‌آموز قدیم بهتر باشد، دانش‌آموز جدید جایگزین دانش‌آموز قدیم می‌شود در غیر این صورت دانش‌آموز قدیم بدون تغییر می‌ماند.

۳-۳- فاز جهش

در هر تکرار یک بردار جهش با دانش آموزان اصلاح شده به صورت رابطه

$$x_i^{mut} = x_{rand1} + r(x_{rand2} - x_{rand3}) \qquad (۷)$$

تولید می‌شود، که در آن x_{rand1}، x_{rand2} و x_{rand3} سه دانش‌آموز تصادفی غیر هم‌شکل و r (ضریب مقیاس‌بندی جهش) یک عدد تصادفی در بازه $[0,1]$ است. در نهایت عضو جهش یافته x_i^{mut} در صورتی که بهتر از x_i^{new} باشد، پذیرفته خواهد شد.

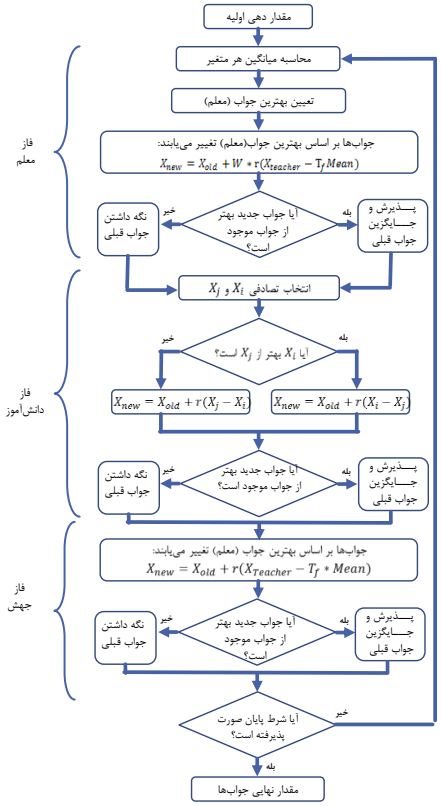
۴- فلوچارت الگوریتم MTLBO

الگوریتم MTLBO روندی مشابه سایر الگوریتم‌های بهینه‌سازی دارد که شامل پنج بخش اصلی می‌باشد. که به ترتیب به قرار زیر می‌باشند:

- تعریف مسئله
- پارامترهای الگوریتم
- آماده‌سازی
- حلقه اصلی (شامل فاز معلم، فاز دانش‌آموز و فاز جهش)
- نتایج

این بخش‌ها به ترتیب در فلوچارت الگوریتم در شکل ۴نمایش داده شده است.

www.SID.ir



شکل ۴- فلوچارت الگوریتم MTLBO

۵- نتایج شبیه‌سازی و مقایسه عملکرد

در این بخش، توانایی الگوریتم MTLBO با اجرا آن برای بهینه‌سازی تعدادی از توابع محک با بعدها و فضای جستجوی متفاوت مورد ارزیابی می‌گردد. نتایج حاصل از الگوریتم MTLBO با الگوریتم TLBO و مابقی الگوریتم‌های بهینه‌ساز موجود در مقالات مقایسه می‌گردد. توابع محک‌های در نظر گرفته شده دارای خواص متفاوتی مانند بدون قید و چند قیده، تفکیک پذیر و عدم تفکیک پذیر و با قاعده و بدون قاعده می‌باشند.

۵-۱- آزمایش اول

برای تعیین پیچیدگی محاسباتی و ثبات الگوریتم TLBO، هشت تابع محک متفاوت در این بخش مورد امتحان قرار گرفته است. نتایج حاصل از به‌کارگیری الگوریتم MTLBO پیشنهادی با الگوریتم‌های بهینه‌ساز شناخته شده دیگر مقایسه گردید.

جزئیات تابع محک‌ها در پیوست الف آورده شده است. برای هماهنگی میان تمامی الگوریتم‌های مقایسه شده، زمانی که اختلاف میان مقدار تابع شایستگی بدست آمده از الگوریتم و مقدار نقطه بهینه مطلق کمتر از ۰.۱ ٪ گردید، الگوریتم متوقف می‌شود برای مثال اگر نقطه بهینه صفر باشد، آن جوابی مورد قبول می‌گردد که از نقطه بهینه به اندازه کمتر از ۰,۰۰۱ اختلاف داشته باشد. هر یک از توابع محک به تعداد ۱۰۰ بار با الگوریتم MTLBO محاسبه گردید و میانگین تعداد ارزیابی تابع[5] و درصد موفقیت در جدول ۱ نمایش داده شده است.

۵-۲- آزمایش دوم

[5] Mean Number of Function Evaluations

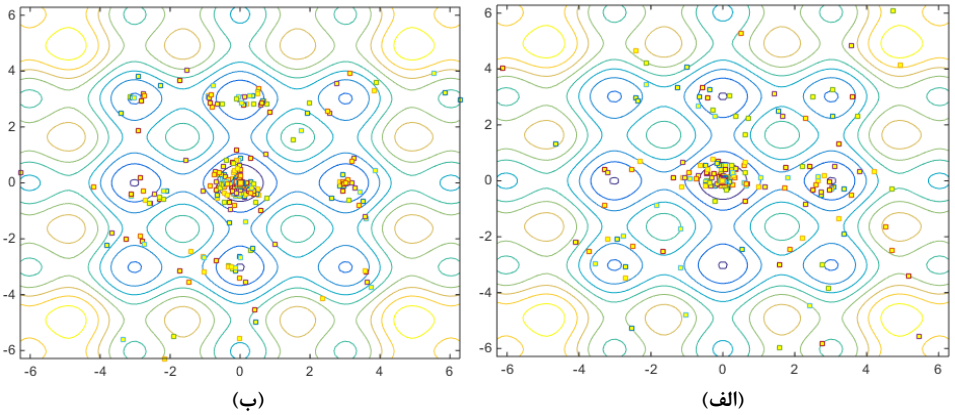
www.SID.ir

جدول ۱- نتایج حاصل از به‌کارگیری الگوریتم MTLBO با الگوریتم‌های بهینه‌ساز شناخته شده دیگر

	De Jong		Goldstein and Price		Branin		Martin and Gaddy	
	MNFE	Succ %	MNFE	Succ %	MNFE	Succ %	MNFE	Succ %
GA	10 160	100	5662	100	7325	100	2488	100
ANTS	6 000	100	5330	100	1936	100	1688	100
Bee Colony	868	100	999	100	1657	100	526	100
GEM	746	100	701	100	689	100	258	100
TLBO	1 070	100	452	100	443	100	422	100
Improve TLBO	862	100	439	100	435	100	357	100

	Rosenbrock (D = 2)		Rosenbrock (D = 2)		Rosenbrock (D = 4)		Hyper sphere (D = 6)	
	MNFE	Succ %	MNFE	Succ %	MNFE	Succ %	MNFE	Succ %
GA	10 212	100	–	–	–	–	15 468	100
ANTS	6 842	100	7 505	100	8 471	100	22 050	100
Bee Colony	631	100	2 306	100	28 529	100	7 113	100
GEM	572	100	2 289	100	82 188	100	423	100
TLBO	669	100	1 986	100	21 426	100	417	100
Improve TLBO	550	100	1892	100	18 692	100	396	100

برای تعیین میزان توانایی جستجوی الگوریتم MTLBO و تعیین نقطه بهینه قطعی از میان تعداد زیادی از نقاط بهینه محلی، تابع محک جعبه تخم مرغ[6] که در پیوست ب معرفی شده است، در نظر می‌گیریم.در شکل ۵ (الف) و (ب) به ترتیب نقاط تعیین شده از هر دوره الگوریتم‌های TLBO و MTLBO بر روی نمودار کانتور[7] این تابع محک رسم شده است. همانطور که از مقایسه این دو شکل مشخص است، الگوریتم MTLBO تقریبا تمامی نقاط بهینه محلی را مورد بررسی قرار داده است. این بدین معنی است که قابلیت جستجوی این الگوریتم نسبت به الگوریتم TLBO بیشتر شده و احتمال تعیین نقطه مینیمم محلی به‌جای دستیابی به نقطه بهینه قطعی کاهش می‌یابد.



شکل ۵-الف: نقاط تعیین شده از هر دوره الگوریتم TLBO ب: نقاط تعیین شده از هر دوره الگوریتم MTLBO

۶- نتیجه‌گیری

یک روش بهبود یافته برپایه الگوریتم بهینه‌سازی، TLBO که بر پایه فلسفه مراحل آموزش- یادگیری ارائه گردید و کارایی آن به وسیله آزمایشات با مسائل محک گوناگون با خصوصیات متفاوت بررسی گردید. بر خلاف الگوریتم TLBO، الگوریتم MTLBO پیشنهادی همه‌سویی[8] جمعیت را تضمین می‌نماید و از به دام افتادن الگوریتم در مینیمم محلی جلوگیری می‌نماید. کارایی الگوریتم MTLBO همچنان برای ملاک‌های متعدد

[6] Egg Create
[7] Contour
[8] Ergodicity

www.SID.ir

مانند میانگین تعداد ارزیابی تابع هدف و میزان جستجو مورد آزمایش قرار گرفت. نتایج کارایی بهتر MTLBO را در مقابل روش TLBO و روش‌های بهینه‌سازی دیگر برای تابع معیار ثابت نشان می‌دهد.

۷- مراجع

[1] J.F. Goncalves, J.J.M. Mendes, M.G.C. Resende, A genetic algorithm for the resource constrained multi-project scheduling problem, European Journal of Operational Research 189 (2008) 1171–1190.
[2] Z. Wu, G. Ding, K. Wang, M. Fukaya, Application of a genetic algorithm to optimize the refrigerant circuit of fin-and tube heat exchangers for maximum heat transfer or shortest tube, International Journal of Thermal Science 47 (2008) 985–997.
[3] S. Suresh, P.B. Sujit, A.K. Rao, Particle swarm optimization approach for multi objective composite-beam design, Composite Structure 81 (2007) 598–605.
[4] R.E. Perez, K. Behdinan, Particle swarm approach for structural design optimization, Computers and Structures 85 (2007) 1579–1588.
[5] Shilane. D, Martikainen. J. S and Dudoit. S, A general frame-work for statistical performance comparison of evolutionary computation algorithms, Information Sciences, Journal, 178, 2870-2879, 2008.
[6] Vallada. E and Ruiz. R, Genetic algorithms with path re linking for the minimum tardiness permutation flowshop problem, Omega, 38, 57 -67, 2011.
[7] Rao RV, Savsani VJ, Vakharia DP. "Teaching-learning-based optimization: a novel method for constrained mechanical design optimization problems," Comput-Aided Design 2011; 43:303–15.
[8] Ahrari, A. and Atai, A.A. ‘‘Grenade explosion method a novel tool for optimization of multimodal functions’’, Appl. Soft Comput 2010, 10, pp. 1132–1140.

پیوست الف: معرفی توابع محک

تعدادی از توابع محک معروف که در مقالات از آن‌ها برای ارزیابی الگوریتم‌های بهینه‌ساز استفاده می‌گردد در جدول ۲ گردآوری شده است. منظور از ستون ابعاد در این جدول بعد جواب‌های مسئله می‌باشد و در ستون محدوده جستجو محدوده فضای جستجو فرض شده در آزمایشات این مقاله می‌باشد.

جدول ۲- معرفی توابع محک[۸]

	تابع معیار	فرمول	ابعاد	محدوده جستجو
۱	De Jong	$Fmin = 3905.93 - 100(x_1^2 - x2)^2 - (1 - x_1)^2$	۲	[-۲,۰۴۸,۲,۰۴۸]
۲	GoldStein–Price	$Fmin = [1 + (x_1 + x_2 + 1)^2(19 - 14x_1 + 3x_1^2 - 14x_2 + 6x_1x_2 + 3x_2^2)][30 + (2x_1 - 3x_2)^2(18 - 32x_1 + 12x_1^2 + 48x_2 - 36x_1x_2 + 27x_2^2)]$	۲	[-۲,۲]
۳	Branin	$Fmin = \left(x_2 - \frac{5.1}{4\pi 2}x_1^2 + \frac{5}{\pi}x_1 - 6\right)^2 + 10(1 - \frac{1}{8\pi})\cos x_1 + 10$	۲	[-۵,۱۰]
۴	Martin and Gaddy	$Fmin = (x_1 - x_2)^2 + \left[\frac{x_1 + x_2 - 10}{3}\right]^2$	۲	[۰,۱۰]
۵	Rosenbrock	$Fmin = 100(x_1^2 - x_2)^2 + (1 - x_1)^2$	۲	[-۱,۲,۱,۲]
۶	Rosenbrock	$Fmin = 100(x_1^2 - x_2)^2 + (1 - x_1)^2$	۲	[-۱۰,۱۰]
۷	Rosenbrock	$Fmin = \sum_{i-1}^{D} 100(x_i^2 - x_{i+1})^2 + (1 - x_i)^2$	۳	[-۱۰,۱۰]
۸	Hyper Sphere	$Fmin = \sum_{i-1}^{D} x_i^2$	۶	[-۵,۱۲,۵,۱۲]

پیوست ب: تابع محک جعبه تخم مرغ

تابع محک جعبه تخم مرغ در این بخش معرفی می‌گردد. رابطه این تابع محک به صورت

$$F_{min} = x_1^2 + x_2^2 + \left(25 * (\sin(x_1)^2 + \sin(x_2)^2)\right) \quad (۸)$$

می‌باشد. از آنجایی که این تابع محک دارای نقاط اکسترمم‌های محلی زیادی است، معمولا از این تابع محک برای آزمایش عدم انتخاب نقطه بهینه محلی بجای نقطه بهینه قطعی یا همان قدرت جستجوی الگوریتم استفاده می‌گردد.

www.SID.ir

International Congress of Interdisciplinary Studies in Science and Engineering
December 2017 - Iran, Tehran
کنگره بین المللی
مطالعات میان رشته ای در علوم پایه و مهندسی
آذرماه ۱۳۹۶ - تهران
Archive of SID

Modified Teaching Learning Based Optimization

Ebrahim Alizadeh, Ali Maleki, Hamid Amiriara

Shomal instiute of defense scinece and technology, University of Malek-Ashtar, E-mail: ealizadeh@mut.ac.ir
Shomal instiute of defense scinece and technology, University of Malek-Ashtar, E-mail: ali.maleki86@ yahoo.com
Shomal instiute of defense scinece and technology, University of Malek-Ashtar, E-mail: hamid.amiriara@gmail.com

Abstract. In this paper teaching- learning based optimizer (TLBO) as effective optimization method for finding exact response for optimization problems is introduced also MTLBO is proposed to modify this algorithm. TLBO is divided to two phases: first phased include training from teacher and in second phase, students learn to each other. In MTLBO, we added third phase to TLBO to prevent local extreme and also change the effect of teacher on students. The accuracy of MTLBO for mast of benchmarks function is calculated and the results are compared to some other algorithms. The results show that the proposed algorithm has more accuracy with respect to TLBO.

Keywords: optimization algorithm, student, teacher, teaching- learning based optimization.

www.SID.ir