

بهبود الگوریتم جستجوی گرانشی (GSA) به کمک منطق فازی

امید مخلص^(۱) - سید حمید ظهیری^(۲) - ناصر مهرشاد^(۳) - سید محمد رضوی^(۳)

(۱) کارشناس ارشد - مؤسسه آموزش عالی خراسان

(۲) دانشیار - دانشکده مهندسی برق، دانشگاه بیرجند

(۳) استادیار - دانشکده مهندسی برق، دانشگاه بیرجند

تاریخ دریافت: تابستان ۱۳۹۱

تاریخ پذیرش: بهار ۱۳۹۲

خلاصه: کارایی روش‌های جستجو و بهینه‌سازی هوش جمعی، تمایل محققین را برای استفاده از آن‌ها در مسائل مختلف پیچیده مهندسی به صورت چشمگیری افزایش داده است. از جمله الگوریتم‌های مبتنی بر هوش جمعی، الگوریتم جستجوی گرانشی (GSA) است که با الهام از قوانین فیزیکی جاذبه گرانشی و حرکت نیوتنی، افراد جامعه را که در واقع جرم‌های تصادفی در فضا هستند را به جستجو در فضا و می‌دارد. این مقاله به ارائه مدل جمعیتی جدیدی به نام Fuzzy Population GSA می‌پردازد که به اختصار آن را FPGSA می‌نامیم. روش پیشنهادی در واقع ترکیبی از دو کنترل‌کننده پارامتریک فازی و الگوریتم جستجوی گرانشی است که به جستجوی دقیق و منطقی فضا می‌پردازد. باید دانست در الگوریتم‌های هوش جمعی (و به تبع آن در روش GSA)، اندازه جمعیت عامل تاثیرگذاری بر پاسخ نهایی است به نحوی که با زیاد شدن این جمعیت پاسخ‌های بهتری حاصل شده و به تبع آن زمان اجرای الگوریتم طولانی‌تر می‌شود که برای رفع این مشکل پارامتری جدید به نام ضریب پراکندگی به الگوریتم افزوده‌ایم تا عملکرد بهتر سیستم را با کنترل این ضریب مورد ارزیابی قرار دهیم.

کلمات کلیدی: کنترل‌کننده پارامتریک فازی، الگوریتم جستجوی گرانشی (GSA)، ضریب پراکندگی، مدل جمعیتی فازی.

۱- مقدمه

در دهه‌های اخیر بررسی‌های بسیاری در زمینه بهینه‌یابی تصادفی صورت گرفته است که به دو دسته عمده از الگوریتم‌های جستجو کلاسیک طبقه‌بندی می‌شوند. دسته اول با الهام از پدیده‌های تکاملی طبیعی و بر مبنای عملگرهای تصادفی پیاده‌سازی می‌شوند. از جمله بارزترین آن‌ها، الگوریتم ژنتیک (GA) است [۱]. دسته دیگر، الگوریتم‌های مبتنی بر هوش جمعی است که در آن‌ها از رفتار اجتماعی افراد یک جامعه به صورت همکاری متقابل و برای رسیدن به یک هدف نهایی به کار گرفته می‌شود. از جمله این الگوریتم‌ها می‌توان به الگوریتم PSO و الگوریتم ACO اشاره کرد که در آن پارامترهای جستجوگر، موقعیت خود را در یک فضای جستجوی چند بعدی تا رسیدن به بهینه شدن تابع هدف، تغییر می‌دهند [۲-۴].

امروزه با توجه به تنوع و پیچیدگی‌های مسائل جدید، روش‌های کلاسیک برای دست‌یابی به پاسخ‌های بهینه در پاره‌ای از موارد دچار چالش می‌شوند. برای دستیابی به مناسب‌ترین پاسخ، الگوریتم‌های ابتکاری بیشماری با الهام از قوانین موجود در طبیعت پایه‌ریزی شده‌اند. از جمله این الگوریتم‌های ابتکاری، الگوریتم GSA است که با الهام از قوانین گرانشی نیوتن شکل می‌گیرد [۵-۶]. همزمان با توسعه این روش‌های بهینه‌یابی نوین و با هدف افزایش کارایی و بهره‌وری روش‌های بهینه‌یابی کلاسیک، روش‌های ترکیبی بسیاری مبتنی بر ترکیب روش‌های کلاسیک بهینه‌یابی تکاملی و یا هوش جمعی با کنترل‌کننده‌های غیرخطی معرفی شده‌اند [۷-۸]. از این جمله می‌توان به کنترل‌کننده‌های فازی [۷]، عصبی [۸] و ترکیبی از هر دو [۹] اشاره کرد.

$$F_{ij}^d(t) = G(t) \frac{M_{pi}(t) \times M_{aj}(t)}{R_{ij}(t)} (x_j^d(t) - x_i^d(t)) \quad (1)$$

$$R_{ij}(t) = \|x_i^d(t), x_j^d(t)\|_2 \quad (2)$$

در روابط بالا $M_{aj}(t)$ مقدار جرم گرانشی موثر (ذره) با موقعیت مکانی بهتر با توجه به تابع هدف، $M_{pi}(t)$ جرم گرانشی تأثیر پذیرنده (ذره) با موقعیت مکانی بدتر با توجه به تابع هدف، $G(t)$ ثابت گرانش در زمان t و $R_{ij}(t)$ فاصله اقلیدسی بین دو ذره هستند.

برای حفظ تصادفی بودن الگوریتم، نیروی وارد بر ذره λ در بعد d از طرف هر ذره را در یک عدد تصادفی با توزیع یکنواخت در بازه $[0, 1]$ ضرب کرده و کل نیروی وارده بر ذره λ در بعد d حاصل جمع نیروهای وارده از طرف همه ذره‌ها بجز خود ذره λ در نظر گرفته می‌شود.

$$F_i^d(t) = \sum_{j=1, j \neq i}^N \text{rand}_j \times F_{ij}^d(t) \quad (3)$$

با توجه به قانون دوم نیوتن چنانچه به ذره‌ای نیرو وارد شود، متناسب با آن نیرو شتاب می‌گیرد.

$$a_i^d(t) = \frac{F_i^d(t)}{M_{ii}(t)} \quad (4)$$

که $M_{ii}(t)$ جرم لختی مربوط به ذره λ است. سرعت بعدی یک ذره تابعی از سرعت فعلی و شتاب فعلی ذره است. بنابراین موقعیت و سرعت ذره با استفاده از روابط زیر قابل محاسبه خواهد بود.

$$v_i^d(t+1) = \text{rand}_j v_i^d(t) + a_i^d(t) \quad (5)$$

$$x_i^d(t+1) = x_i^d(t) + v_i^d(t+1) \quad (6)$$

اینجا نیز متغیر تصادفی یکنواخت در بازه $[0, 1]$ برای این در نظر گرفته شده است که خاصیت تصادفی جستجو از بین نرود. جرم‌های گرانشی و لختی با توجه به مقادیر یک تابع برازندگی محاسبه می‌شوند. جرم سنگین‌تر به معنی ذره موثرتر است.

$$M_{ai}(t) = M_{pi}(t) = M_{ii}(t) = M_i(t), \quad i=1, \dots, N \quad (7)$$

$$m_i(t) = \frac{\text{Value}_i(t) - \text{worst}(t)}{\text{best}(t) - \text{worst}(t)} \quad (8)$$

$$M_i(t) = \frac{m_i(t)}{\sum_{j=1}^N m_j(t)} \quad (9)$$

در رابطه‌ی بالا $\text{Value}_i(t)$ مقدار برازندگی تابع ارزیابی برای ذره λ در زمان t است. همچنین $\text{best}(t)$ و $\text{worst}(t)$ که به ترتیب نشان دهنده‌ی برازندگی قوی‌ترین و ضعیف‌ترین ذره‌های جمعیت در زمان t هستند که با استفاده از روابط زیر به دست می‌آیند.

$$\text{best}(t) = \min_{i \in \{1, 2, \dots, N\}} \text{Value}_i(t) \quad (10)$$

$$\text{worst}(t) = \max_{i \in \{1, 2, \dots, N\}} \text{Value}_i(t) \quad (11)$$

همان‌گونه که اشاره شد عامل توقف در الگوریتم GSA همان $G(t)$

در برخی دیگر کاربردها هدف افزایش و یا کنترل پارامتریک مؤلفه‌های جستجوگر در این الگوریتم‌های کلاسیک است. کنترل فازی پارامترها، از جمله پدیده‌ترین روش در این زمینه است. از این جمله می‌توان به استفاده از روش کنترل فازی بر مؤلفه‌های PSO اشاره کرد [۱۰]. در مرجع [۱۱] دیگر روش پرکاربرد هوش جمعی یعنی ACO تحت تأثیر روش‌های فازی و کنترل پارامتریک قرار گرفته است. در مرجع [۱۲] از ترکیب الگوریتم GA و روش فازی و نیز در مرجع [۱۳] از ترکیب کنترل کننده فازی پارامتریک، جهت طبقه‌بندی داده‌ها استفاده شده است.

آنچه که در این مقاله بدان پرداخته شده است، استفاده از دو کنترل کننده فازی است که با عملکردی موازی، سبب بهبود عملکرد GSA و نیز افزایش بهره‌وری و دقت این الگوریتم است که همان‌گونه که در ادامه بدان پرداخته خواهد شد با افزودن پارامتری جدیدی به نام ضریب پراکندگی و کنترل فازی آن این مهم به خوبی محقق شده است. پیکربندی مقاله به این صورت است که در بخش دوم الگوریتم جستجوی گرانشی استاندارد و ساختار پارامتری استاندارد آن توصیف خواهد شد. بخش سوم به معرفی الگوریتم فازی جدید مبتنی بر کنترل پارامتریک پرداخته شده و به لزوم تعریف پارامتری جدید برای کنترل تراکم جمعیت پرداخته می‌شود. بخش چهارم به ارزیابی روش پیشنهادی پرداخته و در نهایت بخش پنجم به بحث و نتیجه‌گیری نهایی اختصاص یافته است.

۲- الگوریتم جستجوی گرانشی

الگوریتم جستجوی گرانشی (GSA) در زمره‌ی الگوریتم‌های مبتنی بر هوش جمعی طبقه‌بندی می‌شود که یک فضای جستجوی چندبعدی را برای پیدا کردن مقدار اکسترمم تابع هدف مورد جستجو قرار می‌دهد. در این الگوریتم، از مشابهت حرکت سینماتیک و کلاسیک اجرام در میدان گرانشی الهام گرفته شده است. GSA در واقع با استفاده از طرح قوانین گرانش و حرکت در یک سیستم مصنوعی با زمان گسسته ساخته می‌شود که در آن عوامل جستجوکننده، مجموعه‌ای از اجرام هستند که هر جرم در واقع محل و وضعیت سایر اجرام را درک کرده و به طریقی می‌تواند با توجه به تبادل نیرو بین اجرام دیگر، به تبادل اطلاعات پرداخته و سیستمی مصنوعی را ایجاد کند [۵].

پیکربندی الگوریتمی GSA به صورت زیر تعریف می‌شود. بردار موقعیت اولیه شامل مجموعه‌ای از N جرم است که موقعیت هر جرم نقطه‌ای است که می‌تواند شامل d بعد در فضا باشد $(X_i = (x_{i1}, \dots, x_{id}, \dots, x_{iN}))$ که موقعیت بعد d از جرم λ با x_{i^d} نشان داده می‌شود.

در این سیستم در زمان t و در هر بعد d نیرویی به اندازه‌ی $F_{ij}^d(t)$ از سوی جرم j به جرم i وارد می‌شود. مقدار این نیرو از روابط (۱) و (۲) محاسبه می‌شود.

$$D_p = \sum_{i=1}^N \left(\frac{d_i}{N} \right) \quad (i=1, \dots, N) \quad (14)$$

$$D_k = \frac{(D_k - \min(d_i(t)))}{(\max(d_i(t)) - \min(d_i(t)))} \quad (i=1, \dots, N) \quad (15)$$

۳-۲-۲، G(t) پارامتر معرف ثابت گرانش

همان گونه که در رابطه‌ی (۱۲) اشاره شد، $G(t)$ در GSA استاندارد، رابطه‌ای نمایی در نظر گرفته شده است که در واقع عامل همگرا کننده‌ی الگوریتم در طی سپری شدن تکرارهای الگوریتم می‌باشد. مقادیر بزرگ برای این پارامتر باعث تقویت توانایی جستجوی الگوریتم و مقادیر کوچک آن باعث افزایش توانایی بهینه‌یابی GSA می‌شود [۵].

۳-۳-۳ طراحی سیستم فازی برای کنترل پارامترهای G(t)

همان گونه که اشاره شد، انتخاب ثابت گرانش یا $G(t)$ در مسئله امری حیاتی است به طوری که مقدار پارامتری آن در صورتی که بزرگتر باشد موجب تقویت توانایی کاوش و برعکس اگر کوچکتر در نظر گرفته شود، موجب تقویت توانایی بهره‌وری می‌گردد [۵].

اولین گام در طراحی یک سیستم فازی، انتخاب ورودی‌ها و خروجی‌های مناسب با توجه به شرایط مسئله برای سیستم فازی است. با توجه نکات اشاره شده در مورد همگرایی الگوریتم GSA استاندارد و نیز نیاز مبرم این الگوریتم جهت همگرایی یکنواخت بسته به شرایط جمعیت، در این مقاله دو ساختار فازی زیر جهت کنترل پارامتری همگرایی الگوریتم GSA در نظر گرفته شده است.

الف- سیستم اول دارای دو ورودی و یک خروجی است. ورودی‌ها شامل $Value_i(t)$ یا همان مقدار برازندگی تابع هدف و D_k یا همان ضریب پراکندگی است. آنچه که توسط این سیستم فازی و ورودی‌های آن کنترل می‌شود، همان مقدار G_0 است که در GSA استاندارد (رابطه‌ی (۱۲)) به صورت ثابت فرض شده بود.

ب- در سیستم فازی دوم ورودی‌ها، $Fitt$ یا همان مقدار نرمالیزه شده برازندگی تابع هدف است که به صورت زیر قابل محاسبه خواهد بود.

$$Fitt = \frac{(F_{best} - F_{min})}{(F_{max} - F_{min})} \quad (16)$$

مقادیر F_{min} و F_{max} به ترتیب حداقل و حداکثر برازندگی مورد تعریف برای عامل جرم در دامنه‌ی فضا است که هرگاه خارج آن قرار گیرد در واقع خارج از فضای مسئله است. ورودی دوم مقدار t یا همان تکراری است که سیستم در آن قرار دارد. خروجی مقدار α است که در رابطه‌ی (۱۲) به صورت ثابت فرض شده بود.

رویکرد کلی سیستم‌های فازی پیشنهادی برای کنترل پارامتریک الگوریتم GSA، در شکل (۱) نشان داده شده است.

یا ثابت گرانش است که در آن α و G_0 ، مقادیر ثابتی در نظر گرفته شده و T و t به ترتیب بیانگر کل تکرارها و تکرار فعلی است. به علت اهمیت این پارامتر، در ادامه مقاله بیشتر بدان پرداخته خواهد شد [۵].

$$G(t) = G_0 \cdot e^{-a\left(\frac{t}{T}\right)} \quad (12)$$

۳-۲-۳ FPGSA (Fuzzy Populated GSA): ضرورت تعریف

پارامترهای جدید و حیاتی

آنچه که در این مقاله به عنوان الگوریتم FPGSA معرفی شده است، در واقع الگوریتم بهینه‌یابی گرانشی است مبتنی بر کنترل فازی اندازه جمعیت است که با هدف جلوگیری از ایستایی بخش عمده‌ای از جرم‌ها در برخی از تکرارها شکل می‌گیرد. به بیان دیگر الگوریتم GSA مشابه سایر الگوریتم‌های هوش جمعی، با افزایش جمعیت در پاره‌ای از تکرارها دچار نابسامانی در نتیجه‌گیری می‌شود. افزایش جمعیت و به تبع آن قرار گرفتن تراکم بخشی از جمعیت در ناحیه‌ای از فضا می‌تواند سبب از دست رفتن بخش عمده‌ای از تجربیات برخی از جرم‌ها شده و در نتیجه‌ی آن همگرایی به اکستریم‌های محلی و پایین آمدن بهره‌وری تا حد قابل ملاحظه‌ای شود. همین ایستایی و تراکم می‌تواند همان عاملی باشد که همان طور که در ادامه نشان داده خواهد شد، الگوریتم GSA استاندارد را در برخی از تکرارها تا حد قابل ملاحظه‌ای دچار اختلال در همگرایی و در برخی از موارد دچار واگرایی‌های لحظه‌ای می‌کند. همان گونه که در ادامه نشان داده می‌شود الگوریتم GSA در برخی از توابع به شدت تحت تأثیر تعداد تکرارهای الگوریتم است که این خود به علت تأثیر پارامتری تعداد تکرار نهایی (T) در همگرا کننده‌ی گرانشی تعریف شده در رابطه‌ی (۱۲) است. البته نباید از همگرایی بسیار مناسب و توانمندی نهایی الگوریتم GSA چشم‌پوشی کرد آنچه که در ادامه خواهیم دید همگرایی در پاره‌ای از مواقع سریعتر و یکنواخت‌تر روش پیشنهادی در تعداد تکرارهای بسیار اندک خواهد بود.

۳-۱-۳، D_k پارامتر معرف ضریب پراکندگی

در این مقاله مشابه کار انجام شده در [۹]، در راه کنترل ایستایی جمعیت نیاز به تعریف پارامتری است که به عنوان ورودی یکی از کنترل کننده‌های فازی مورد استفاده قرار گیرد. D_k پارامتری است که معرف میزان پراکندگی بخشی از اجرام در بخشی از دامنه فضا است که برای کنترل کردن فازی جمعیت از جرم‌ها مورد استفاده قرار داده‌ایم و طی فرآیند زیر به صورت نرمالیزه شده، محاسبه و در نهایت به عنوان ورودی در اختیار یکی از سیستم‌های فازی قرار می‌گیرد. روند محاسبه در روابط زیر خلاصه می‌شود که در آن N تعداد کل جمعیت بوده و d_i تفاضل بهترین موقعیت از بدترین موقعیت است. تعریف این پارامتر به نوعی می‌تواند تعریف حساسیت فاصله بین المان‌های جستجوگر باشد.

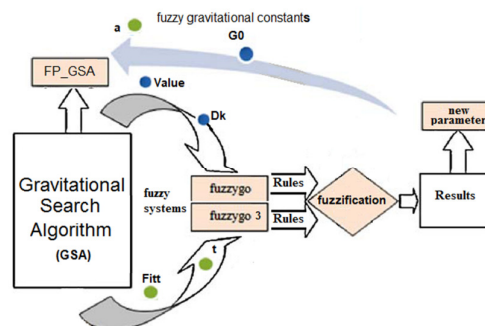
$$d_i = \text{abs}(Value_i(t) - \text{worst}) \quad (i = 1, \dots, N) \quad (13)$$

انجام شدخ مشابه در طراحی این نوع از کنترل کننده‌های فازی انتخاب شده‌اند [۱۶-۱۵].

از طرف دیگر شرایط همپوشانی قواعد و سطوح تصمیم سیستم‌های فازی پیشنهادی که به ترتیب در شکل‌های (۴) و (۵) نشان داده شده است نشان از انتخابی موثر برای توابع عضویت مسئله دارد.

دو سیستم فازی مطرح شده با دریافت ورودی‌های بیان شده، خروجی‌های مربوطه را کنترل کرده که به تبع آن، بر رفتار الگوریتم GSA تأثیر می‌گذارند. بر طبق آزمایش‌های متعدد انجام شده، برای هر یک سیستم‌های فازی پیشنهادی، ۹ قانون در نظر گرفته شده است که با توجه به عملکرد پارامترهای ورودی و خروجی الگوریتم تنظیم شده‌اند. این تنظیم پارامتریک با توجه به شرایط الگوریتم در مراحل اولیه جستجو و نیز با توجه به لزوم جستجوی نقاط اولیه جدید در فضای مسئله در مراحل پایانی با افزایش توان بهره‌وری و با هدف بهبود جواب‌های دیده شده صورت می‌گیرد که تأثیر آن را می‌توان در جستجوی محلی، جستجوی سراسری، سرعت همگرایی و بهینه‌یابی در تعداد گام‌های تکرار کمتر به وضوح مشاهده کرد.

به عنوان مثال در سیستم فازی اول اگر مقدار $Value_i(t)$ به مقدار بهینه تابع هدف نزدیک شده باشد (البته در $[0, 1]$) این بدین معنی است که در گام‌های آخر هستیم و باید دقت جستجو افزایش یابد و همچنین اگر مقدار D_k کم باشد این بدان معنی است که جرم‌ها در نقطه‌ای نامناسب قرار دارند در این صورت منطقی‌ترین شرایط آن است که مقدار پارامتر G_0 افزایش یابد. و یا در صورت عدم بهبود همگرایی در مراحل انتهایی‌تر الگوریتم، α می‌باید کاهش یابد. بهمین ترتیب با استدلال‌های مشابه دو سیستم فازی مستقل مورد نظر را به همراه ۹ قاعده IF-THEN فازی به صورت شکل (۲) پیشنهاد می‌کنیم.

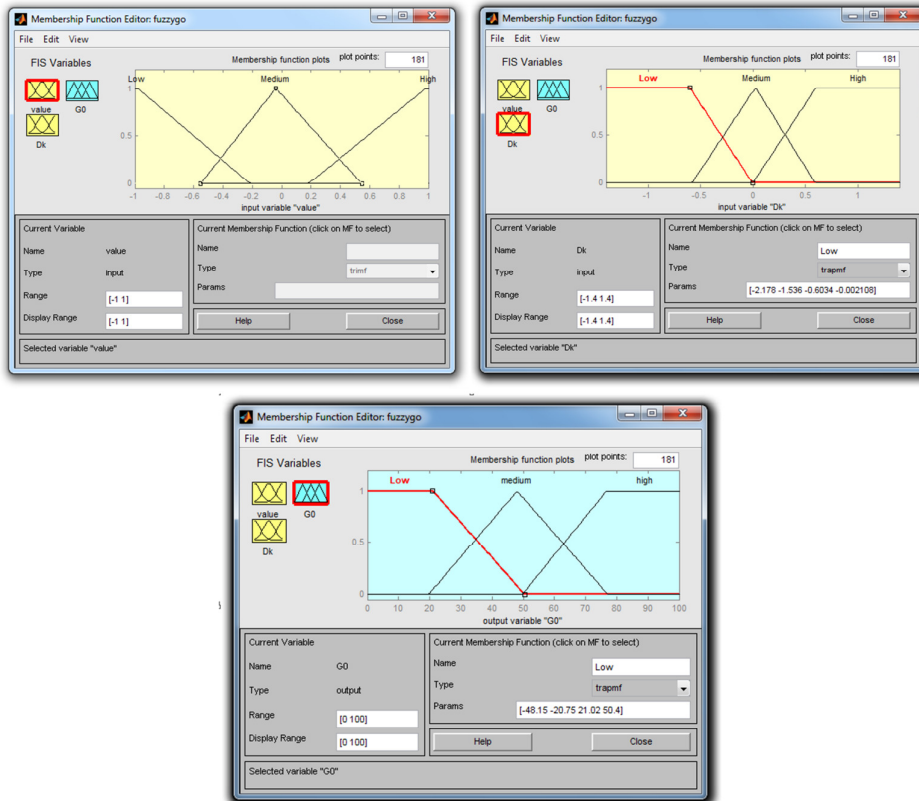


شکل (۱): روش پیشنهادی برای فازی‌کردن پارامتریک
Fig (1): The proposed method for parametric fuzzification

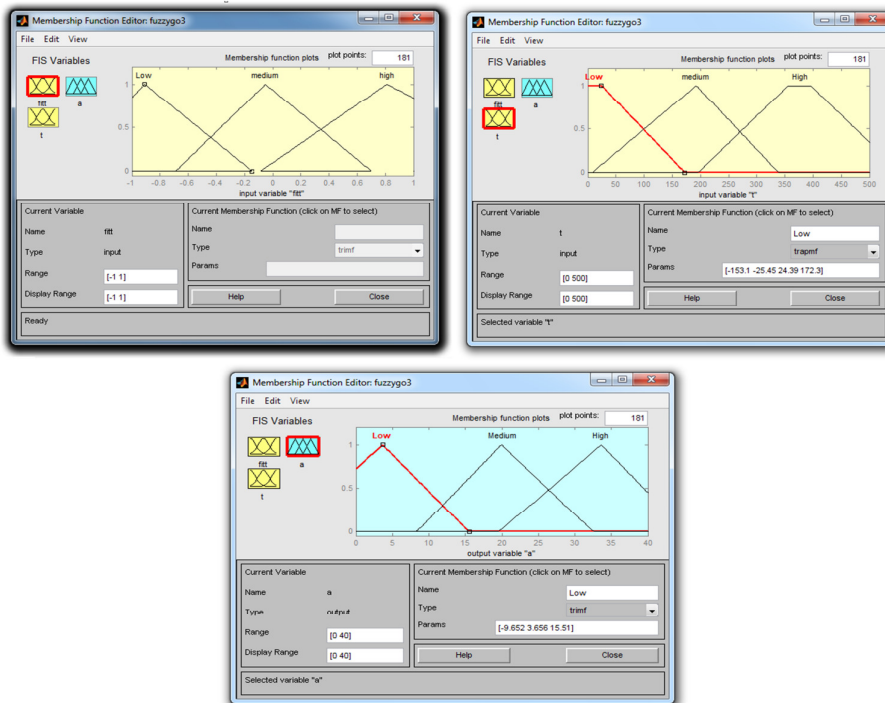
۳-۴- تعریف توابع عضویت

توابع عضویت فازی برای حل مسائل مختلف می‌باید بسته به شرایط ساختاری مسئله تعریف شوند و در واقع تعریف درست و بجا توابع عضویت در یک سیستم فازی، تأثیر مستقیمی بر پاسخ مسئله دارد [۱۴].

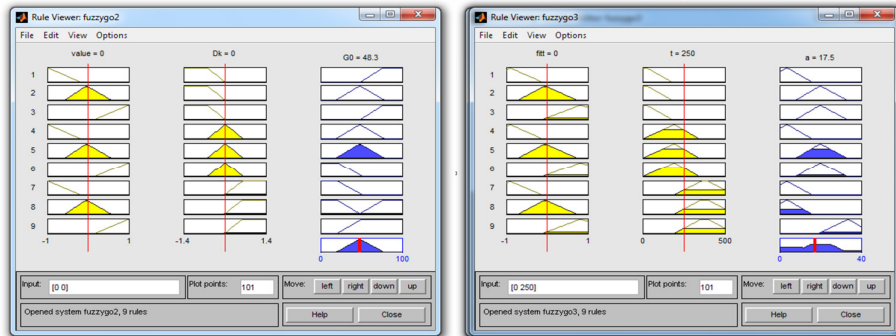
توابع عضویتی که برای این دو سیستم فازی در این مقاله در نظر گرفته شده است، در شکل‌های (۲) و (۳) نشان داده شده است. با توجه به تنوع بیشتر توابع عضویت در یک سیستم فازی، انتخاب‌های بیشماری برای توابع عضویت در سیستم‌های فازی مورد نظر وجود دارد. اما آنچه که به کارآیی بهتر روش پیشنهادی منجر می‌شود بیشتر به خاطر استخراج قواعد مؤثر و منطقی در کنترل پارامترهای تأثیرگذار در فرایند جستجوی GSA است. یعنی دانش موجود و مبهم در مکانیزم جستجو در GSA منجر به ایجاد یکسری قواعد فازی (و معتبر برای هر نوع استفاده از GSA) شده است که در هر تکرار می‌تواند با کنترل پارامترهای آن نسبت به ارزیابی جستجوی سراسری و محلی مبادرت نماید. انتخاب نوع مثلثی توابع عضویت بر اساس دیگر کارهای



شکل (۲): توابع عضویت در نظر گرفته شده برای سیستم فازی اول
 Fig. (2): Fuzzy membership functions for the first Fuzzy system

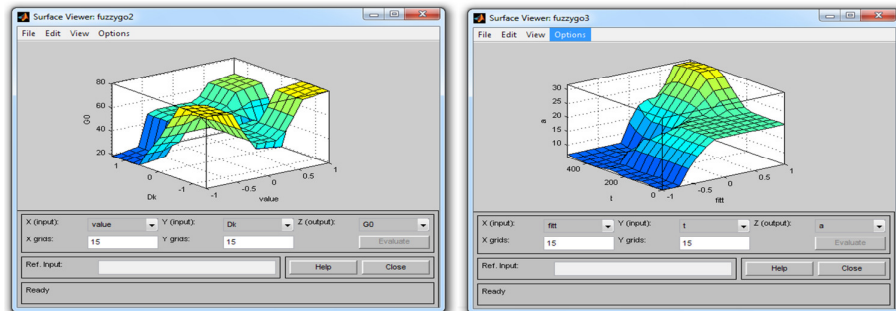


شکل (۳): توابع عضویت در نظر گرفته شده برای سیستم فازی دوم
 Fig. (3): Fuzzy membership functions for the Second Fuzzy system



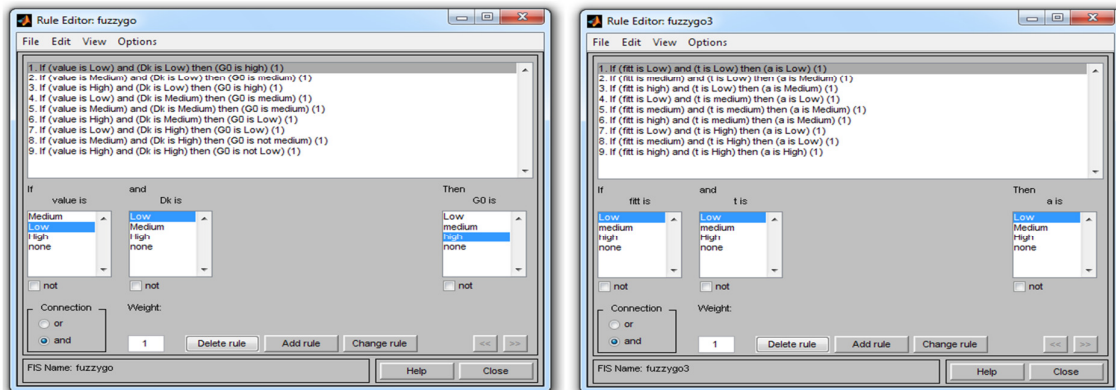
شکل (۴): شماتیک قواعد فازی برای هر دو سیستم فازی

Fig (4): Schematic of the fuzzy rules for both of fuzzy systems



شکل (۵): شماتیک سطحی متناظر با هر دو سیستم فازی

Fig. (5): Schematic of surface corresponding to both of fuzzy systems



شکل (۶): قوانین فازی پیشنهادی برای هر دو سیستم فازی

Fig. (6): Proposed Fuzzy rules for both of fuzzy systems

۴- توابع ارزیابی

در این مقاله جهت مینیمم‌سازی از برخی توابع ارزیابی پرکاربرد تک مدل و چندمدل (جدول (۲)) که در مقالات و مراجع بیشماری به کار رفته‌اند، جهت ارزیابی و مقایسه مابین الگوریتم FPGSA و GSA و PSO استاندارد بهره گرفته‌ایم [۱۷].

در این ارزیابی‌ها مشخصه‌های پارامتری یکسانی که شرح آن در جدول (۱) آمده است، برای الگوریتم‌های GSA، FPGSA و PSO مورد استفاده قرار گرفته است. همانگونه که اشاره شد، جهت مقایسه بهتر رفتار روش پیشنهادی، یکی از پرکاربردترین الگوریتم‌های هوش جمعی

یعنی PSO را نیز با ساختار پارامتری مشابه مورد استفاده قرار داده‌ایم. این پارامترهای ساختاری مشابه برای روش PSO عبارتست از تعداد جمعیت اولیه ۳۰ عامل جستجو، ابعاد فضای جستجو با توجه به ابعاد توابع ارزیابی و نیز کل تعداد تکرارهای الگوریتم ۵۰۰ مرتبه مشابه دو الگوریتم دیگر در نظر گرفته شده است. جهت بروز رسانی موقعیت ذرات از رابطه‌ی (۱۷) و جهت محاسبه سرعت ذرات از رابطه‌ی (۱۸) استفاده شده است. برای تنظیم بهتر ضریب وزن اینرسی (W) در رابطه‌ی (۱۸)، مقادیری خطی از ۰/۹ به ۰/۲ به صورت کاهشی قرار داده شده است. R_1 و R_2 اعداد تصادفی با توزیع یکنواخت بین صفر و

از الگوریتم‌ها محاسبه شده و میانگین^۱ (avt) بهترین جواب‌های دیده شده، میانه^۲ بهترین جواب‌های دیده شده در کل تکرارها الگوریتم (nor) گزارش شده است. و نیز به جهت مقایسه‌ی پراکندگی پاسخ‌ها، بهترین پاسخ (bv) و بدترین پاسخ (wv) نیز به ازای اجرای ۱۰ مرتبه‌ای هر یک از الگوریتم‌ها گزارش شده است. جهت مقایسه همگرایی بهبود یافته‌ی روش پیشنهادی در مقایسه با دو الگوریتم دیگر، در شکل (۷) معیار بهترین جواب دیده شده تا آن لحظه^۳ به کل تکرارها برای همه توابع به تصویر کشیده شده است.

یک هستند. همچنین بهترین موقعیتی است که تا کنون توسط جمعیت دیده شده و X_{L-best} بهترین موقعیتی است که ذره تاکنون به دست آورده است. در ضمن پارامترهای اختصاصی الگوریتم PSO در اینجا $C_1 = C_2 = 2$ فرض شده‌اند [۱۸-۱۹].

$$X_{ij} = X_{ij} + V_{ij} \quad (۱۷)$$

$$V_{ij} = W \cdot V_{ij} + C_1 \cdot R_1 (X_{L-best} - X_{ij}) + C_2 \cdot R_2 (X_{G-best} - X_{ij}) \quad (۱۸)$$

۴-۱- نتایج آزمایشات

نتایج پیاده‌سازی توابع ارزیابی در جدول (۳) گزارش شده است. برای اطمینان از صحت درستی نتایج برای ۱۰ مرتبه اجرای مستقل هر یک

Table (1): Characteristic parameters of FPGSA and standard GSA algorithms

جدول (۱): مشخصات پارامتری الگوریتم‌های FPGSA و GSA استاندارد

PSO	GSA	FPGSA	
30	30	30	سایز جمعیت اولیه
			ابعاد اجرای بهینه یابی
			G0
			α
			تکرار اجرای الگوریتم

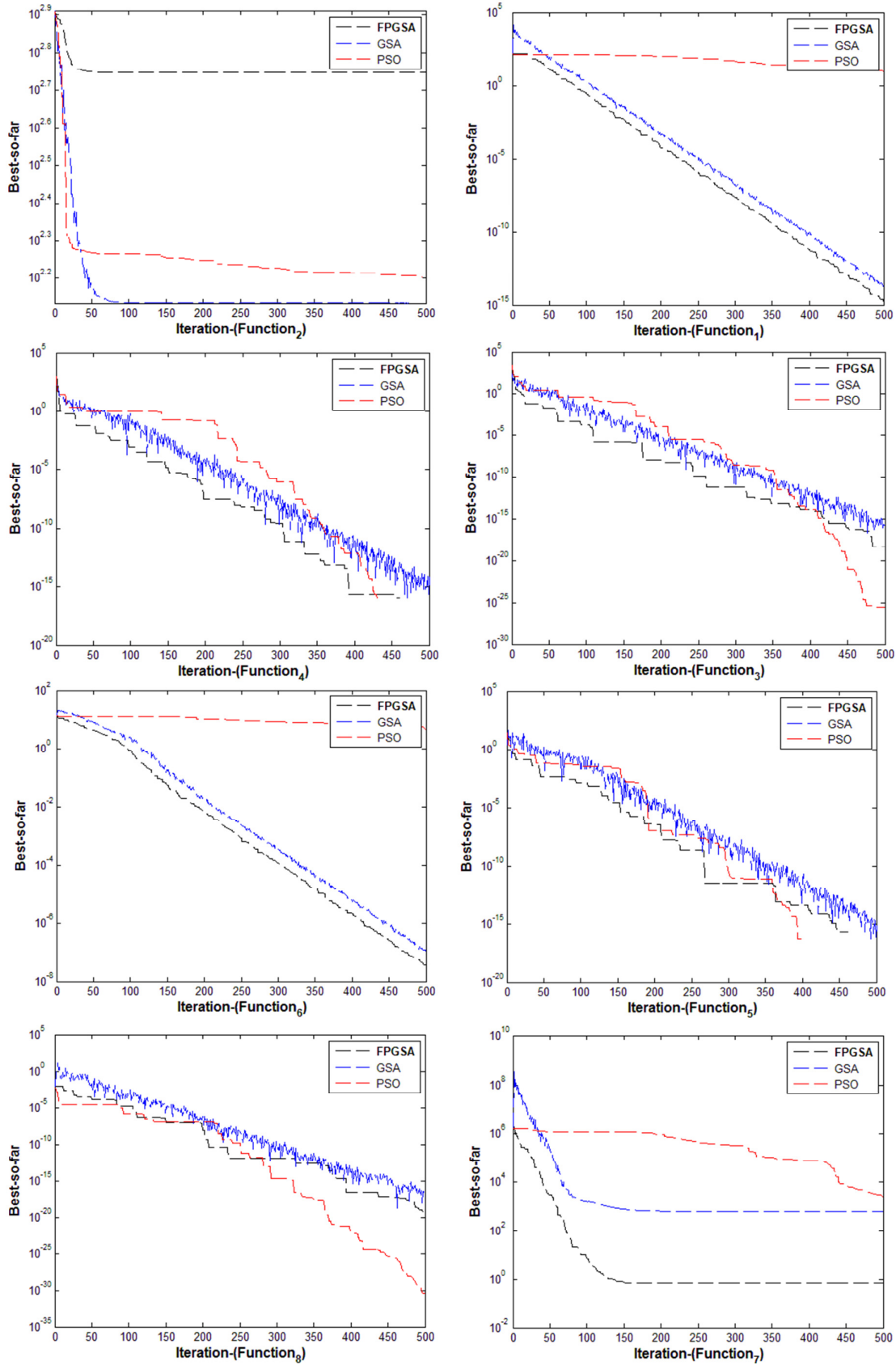
Table (2): Considered Standard test functions with Unimodal and Multimodal shapes

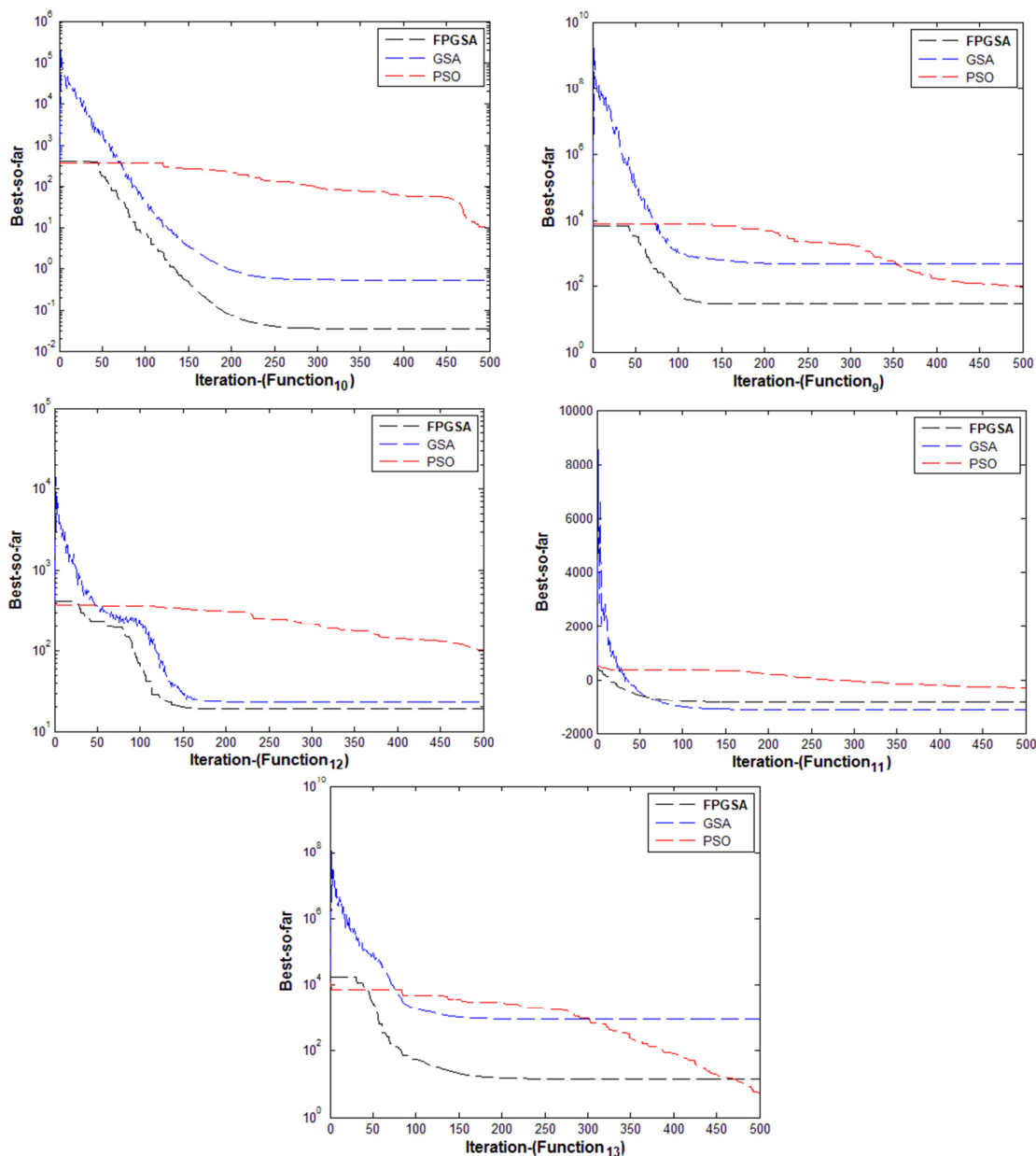
جدول (۲): توابع ارزیابی استاندارد انتخابی از نوع تک مدل یا چند مدل جهت ارزیابی هر سه روش

تابع	عنوان تابع ارزیابی	رابطه‌ی تابع	دامنه تک یا مالتی مدل تابع
F_1	Sphere Function	$F_1(x) = \sum_{i=1}^n X_i^2$	$[-5.12, 5.12]^n$
F_2	Schwefel 2.26 Function	$F_2(x) = \sum_{i=1}^n x_i \sin(x_i)$	$[-4, 5]^2$
F_3	Booth Function	$F_3(x) = (x_1 + 2x_2 - 7)^2 + (2x_1 + x_2 - 5)^2$	$[-10, 10]^2$
F_4	Bohachevsky 1 Function	$F_4(x) = x_1^2 + 2x_2^2 - 0.3 \cos(3\pi x_1) - 0.4 \cos(4\pi x_2) + 0.7$	$[-100, 100]^2$
F_5	Bohachevsky 2 Function	$F_5(x) = x_1^2 + 2x_2^2 - 0.3 \cos(3\pi x_1) * \cos(4\pi x_2) + 0.3$	$[-100, 100]^2$
F_6	Ackley Function	$F_6(x) = 20 + e - 20 e^{-0.2 \sqrt{\frac{1}{n} \sum_{i=1}^n x_i^2}} - e \frac{1}{n} \sum_{i=1}^n \cos(2\pi x_i)$	$[-32, 32]^n$
F_7	Dixon and price Function	$F_7(x) = (x_1 - 1)^2 + \sum_{i=0}^n i(2x_i^2 - x_{i-1})^2$	$[-10, 10]^n$
F_8	Matyas Function	$F_8(x) = 0.26(x_1^2 + x_2^2) - 0.48x_1x_2$	$[-10, 10]^2$
F_9	Rosenbrock Function	$F_9(x) = \sum_{i=1}^{n-1} [100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2]$	$[-2.04, 2.04]^n$
F_{10}	Sum squares function	$F_{10}(x) = \sum_{i=1}^n ix_i^2$	$[-10, 10]^n$
F_{11}	Trid Function	$F_{11}(x) = \sum_{i=1}^n (x_i - 1)^2 - \sum_{i=2}^n x_i x_{i-1}$	$[-10, 10]^n$
F_{12}	Rastrigin Function	$F_{12}(x) = \sum_{i=1}^n (X_i^2 - 10 \cos(2\pi x_i) + 10)$	$[-5.12, 5.12]^n$
F_{13}	Powell Function	$F_{13}(x) = \sum_{i=1}^n (x_{4i-3} + 10x_{4i-2})^2 + 5(x_{4i-1} - x_{4i})^2 + (x_{4i-3} - x_{4i-1})^4 + 10(x_{4i-3} - x_{4i})^4$	$[-4, 5]^n$

Table (3): The minimization result (The best, worst, average and median) of implemented functions over 10 runs
 جدول (۳): نتایج می‌نیمم‌سازی توابع ۱ تا ۱۳ با بعد ۳۰. بهترین، بدترین مقدار دیده‌شده و میانگین و میانه نتایج برای ۱۰ بار اجرای مستقل الگوریتم‌ها

FPGSA	GSA	PSO	معیار	تابع
2.03×10^{-15}	1.17×10^{-14}	8.39×10^{-3}	بهترین مقدار (bv)	F ₁
2.91×10^{-15}	2.42×10^{-14}	1.94	بدترین مقدار (wv)	
5.42×10^{-14}	1.86×10^{-13}	8.31×10^{-1}	میانگین نتایج (avr)	
2.88×10^{-14}	1.88×10^{-13}	5.25×10^{-1}	میانه نتایج (mor)	
417.54	115.41	270.83	بهترین مقدار (bv)	F ₂
518.23	175.29	405.32	بدترین مقدار (wv)	
357.95	137.55	345.40	میانگین نتایج (avr)	
448.105	135.61	337.94	میانه نتایج (mor)	
1.19×10^{-17}	5.49×10^{-18}	3.15×10^{-30}	بهترین مقدار (bv)	F ₃
9.05×10^{-17}	7.52×10^{-17}	7.40×10^{-27}	بدترین مقدار (wv)	
3.29×10^{-16}	1.93×10^{-15}	2.12×10^{-27}	میانگین نتایج (avr)	
2.68×10^{-16}	8.06×10^{-16}	9.12×10^{-28}	میانه نتایج (mor)	
1.11×10^{-16}	7.70×10^{-16}	1.11×10^{-16}	بهترین مقدار (bv)	F ₄
4.44×10^{-16}	1.11×10^{-15}	2.62×10^{-16}	بدترین مقدار (wv)	
2.01×10^{-15}	1.95×10^{-14}	1.17×10^{-16}	میانگین نتایج (avr)	
2.21×10^{-15}	1.55×10^{-14}	1.19×10^{-16}	میانه نتایج (mor)	
1.11×10^{-16}	2.22×10^{-16}	2.76×10^{-16}	بهترین مقدار (bv)	F ₅
5.55×10^{-16}	4.44×10^{-15}	5.59×10^{-16}	بدترین مقدار (wv)	
3.21×10^{-15}	1.70×10^{-14}	1.20×10^{-15}	میانگین نتایج (avr)	
3.05×10^{-15}	5.28×10^{-15}	4.48×10^{-15}	میانه نتایج (mor)	
3.42×10^{-18}	3.92×10^{-18}	1.21	بهترین مقدار (bv)	F ₆
4.20×10^{-8}	8.59×10^{-7}	2.46	بدترین مقدار (wv)	
3.82×10^{-7}	7.29×10^{-7}	1.83	میانگین نتایج (avr)	
3.77×10^{-7}	6.95×10^{-7}	1.94	میانه نتایج (mor)	
0.66	0.68	1.36	بهترین مقدار (bv)	F ₇
132.16	411.23	184.6	بدترین مقدار (wv)	
32.34	123.56	79.03	میانگین نتایج (avr)	
11	11.38	14.37	میانه نتایج (mor)	
1.42×10^{-20}	1.54×10^{-18}	1.11×10^{-30}	بهترین مقدار (bv)	F ₈
7.63×10^{-19}	8.36×10^{-17}	3.68×10^{-26}	بدترین مقدار (wv)	
2.89×10^{-18}	1.53×10^{-16}	3.86×10^{-26}	میانگین نتایج (avr)	
2.05×10^{-18}	3.25×10^{-17}	1.91×10^{-25}	میانه نتایج (mor)	
26.85	26.31	40.77	بهترین مقدار (bv)	F ₉
76	584.65	516.65	بدترین مقدار (wv)	
39.04	92.32	195.55	میانگین نتایج (avr)	
28.60	27.85	84.85	میانه نتایج (mor)	
2.6×10^{-3}	4.8×10^{-3}	1.26	بهترین مقدار (bv)	F ₁₀
2.96	4.22	47.35	بدترین مقدار (wv)	
1.01	1.53	13.65	میانگین نتایج (avr)	
0.656	1.17	12.47	میانه نتایج (mor)	
-875.20	-1064	-693.44	بهترین مقدار (bv)	F ₁₁
-533.54	-832	-320.31	بدترین مقدار (wv)	
-715.06	-905.90	-481.99	میانگین نتایج (avr)	
-721.94	-867	-488.30	میانه نتایج (mor)	
7.95	10.94	32.44	بهترین مقدار (bv)	F ₁₂
25.86	21.88	110.81	بدترین مقدار (wv)	
18.20	17.52	59.15	میانگین نتایج (avr)	
17.41	18.40	67.89	میانه نتایج (mor)	
70.18	69.30	29.86	بهترین مقدار (bv)	F ₁₃
1287.12	1299.18	55.47	بدترین مقدار (wv)	
512.31	598.43	49.39	میانگین نتایج (avr)	
479.34	566.70	45.78	میانه نتایج (mor)	





شکل (۷): میانگین بهترین همگرایی مقدار دیده شده برای توابع ۱ تا ۱۳

Fig. (7): Comparison of performance of GSA, PSO and FPGSA for minimization of F1 to F13

توابع عملگری نزدیک به یکدیگر دارند. علاوه بر این، نتایج آزمایش‌ها روی تقریباً اکثر توابع (جز برخی از توابع تک مدل مثل F_2 و F_{11}) نشان دهنده آن است که الگوریتم FPGSA نسبت به GSA از توانایی بهره‌وری بهینه‌یابی بالاتری برخوردار شده است. این موضوع می‌تواند تأکیدی بر نحوه کنترل ثابت گرانش به صورت فازی بجای استفاده از مقادیر ثابتی برای آن باشد. اینکه در دو تابع نتایج بهتری حاصل نشده است می‌تواند بدین علت باشد که بعضی از قواعد فازی در برخی شرایط اکتیو نمی‌شوند یعنی شرایطی که کنترل فازی دخالت کند مهیا نمی‌شود. اما همان طور که وضوح مشخص است، این شرایط در توابع مالتی مدل مناسب‌تر است.

۲-۴- تحلیل نتایج

همان‌گونه که اشاره شد، نتایج بهینه‌سازی توابع تک مدل و چند مدل در جدول (۳) و نیز مقایسه‌ی همگرایی در شکل (۷)، نشان دهنده آن است که الگوریتم FPGSA در پاره‌ای از مواقع خیلی سریع‌تر از الگوریتم‌های PSO و GSA به جواب بهینه همگرا می‌شود. البته نباید از عملکرد این دو الگوریتم‌های در جستجوی بهینه فضا و سرعت همگرایی در حل برخی از مسائل مالتی مدل چشم‌پوشی کرد. نتایج ارائه شده در جدول (۳)، نشان دهنده‌ی عملکرد مشابه GSA و FPGSA در حل مسایل مالتی مدل با ابعاد کم (مثل F_1, F_6, F_9, F_{10} و F_{12}) است. همان طور که ملاحظه می‌شود این دو الگوریتم در این

۵- نتیجه گیری

مقابل موقعیت‌های غیرعادی برخی از اجرام، تصمیم‌گیری مناسب از خود نشان دهد. با توسعه روش فوق بر دیگر بخش‌های این الگوریتم می‌توان رفتار این الگوریتم کارآمد را باز هم هوشمندانه‌تر و دقیق‌تر کرد.

در این مقاله با ارائه یک روش جدید فازی، اولاً امکان جستجوی موازی را به الگوریتم استاندارد اضافه کرده، ثانياً با افزودن پارامتری جدید به نام ضریب پراکندگی و کنترل فازی مناسب این پارامتر نه تنها سرعت همگرایی الگوریتم به مراتب بهبود می‌یابد بلکه به دلیل کنترل تکرار به تکرار جمعیت، مانع از سکون و عدم تحرک برخی از اجرام در طی فرایند اجرای الگوریتم و به تبع آن مانع از دست رفتن تجربیات همه اجرام می‌شود. در نهایت می‌توان اشاره کرد که استفاده از قابلیت‌های توابع فازی نه تنها می‌تواند رفتار الگوریتم GSA استاندارد را به مراتب بهبود بخشیده بلکه و این امکان را برای الگوریتم فراهم کند که در

پی‌نوشت:

- 1- Average
- 2- Median
- 3- Best-so-far

References

- [1] D.H. Kim, A. Abraham, "A hybrid genetic algorithm and bacterial foraging approach for global optimization", *Information Sciences*, Vol. 177, No. 18, pp. 3918–3937, Sep. 2007.
- [2] Y.L.Lin, W.D.Chang, J.G.Hsieh, "A particle swarm optimization approach to nonlinear rational filter modeling", *Information Sciences*, Vol. 34, No. 2, pp.1194–1199, Feb.2008
- [3] J. Kennedy, R.C. Eberhart, "Particle swarm optimization", *IEEE/ICNN*, Vol. 4, pp. 1942–1948, 1997.
- [4] S.G. Li, Y.L. Rong, "The reliable design of one-piece flow productionsystem using fuzzy antcolonyoptimization", *Com. and Ope. Res.*, Vol. 36, No. 5, pp. 1656 - 1663, May 2009.
- [5] E. Rashedi, H. Nezamabadi-pour, S. Saryazdi, "GSA: A gravitational search algorithm", *Information Sciences*, Vol. 179, pp. 2232–2248, 2009.
- [6] R.A. Formato, "Central force optimization: A new nature inspired computational framework for multidimensional search and optimization", *Studies in Computational Intelligence*, Vol. 129, pp. 221–238, 2008.
- [7] Y.Shi, R. Eberhart, Y. Chen, "Implementation of evolutionaryfuzzy systems", *IEEE Trans. on Fuzzy Sys.*, Vol. 7, No. 2, pp. 109-119, 1999.
- [8] Z .Yao ,P.Sen, G.E .Hearn, "An on-line trained adaptive neural controller", *IEEE on Control Systems*, Vol. 15, No. 5, pp. 67-75, Oct. 1995.
- [9] L .Yonggon, S.H.Zak, "Designing a genetic neural fuzzy antilock-brake-system controller", *IEEE Trans. on Evolutionary Computation*, Vol. 6, Issue: 2
- [10] O .Mokhlessi, S.H .Zahiri, N. Mehrshad, "Fuzzy optimum PSO: PSO with optimized fuzzy controllers", *Majlesi Jou. of Elec. Eng.*, Vol. 6, No. 3, 2012.
- [11] S.G. Li, Y.L. Rong, "The reliable design of one-piece flow productionsystem using fuzzy ant colonyoptimization", *Com. and Ope. Res.*, Vol. 36, pp. 1656 - 1663, 2009.
- [12] M. Stenes, H. Robous, "GA-fuzzy modeling and classification: Complexity and performance", *IEEE Trans. on Fuzzy Sys.*, Vol. 8, No. 5, pp. 509-522, 2000.
- [13] R.-C.David, E.M .Petriu, S .Preitl, M .Radac, "Fuzzy control systems with reduced parametric sensitivity based on simulated annealing", *IEEE Trans. on Ind. Elec.*, pp. 3049-3061, Aug. 2012.
- [14] Y.T. Juang, S.L. Tung, H.C. Chiu, "Adaptive fuzzy particle swarmoptimization for global optimizationof multimodalfunctions", *Information Sciences*, Vol. 181, pp. 4539–4549, 2011.
- [15] H. Liu, A. Abraham, "Fuzzy adaptive turbulent particle swarm optimization", *IEEE/ICHIS*, Nov. 2005.
- [16] S.Mitra, S.K.Pal, "A fuzzy sets in pattern recognition and machine intelligence", *Fuzzy Sets and Systems*, Vol. 156, No. 3, pp. 381-386, Dec. 2005.
- [17] T.A.Jones, "A MATLAB functions to analyze directional (azimuthal) data—I: Single-sample inference", *Com. and Geosciences*, Vol. 32, pp. 166–175, 2006.
- [18] T. Niknam, "A new fuzzy adaptive hybrid particle swarm optimization algorithm for non-linear, non-smooth and nonconvex economic dispatch problem", *Appl. Ene.*, Vol. 87, No. 1, pp. 327-339, Jan. 2010.
- [19] H. Shayeghi, A. Jalili, H.A. Shayanfar, "Multi-stage fuzzy load frequency control using PSO", *Inf. Ene. Conv. and Man.*, Vol. 9, No. 10, pp. 2570–2580, Oct. 2008.

