

مقایسه کارایی الگوریتم‌های فراابتکاری بر روی برخی توابع خاص با انطباق پارامترهای فازی

حسین اقبالی،* سمانه اصغری کنارسری**

تاریخ دریافت: ۱۴۰۱/۲/۳۰ تاریخ پذیرش: ۱۴۰۱/۳/۲۸ نوع مقاله: پژوهشی

چکیده در این تحقیق، مقایسه کارایی الگوریتم‌های فراابتکاری (PSO^۱ و ICA^۲) بر روی برخی توابع خاص با انطباق پارامترهای فازی صورت گرفته است. با الگوریتم‌های فازی با در نظر گرفتن هر یک از این توابع تست به عنوان تابع هدف یا بردنزاگی، راک خود را که رسیدن به مینیمم سراسری هر یک از این توابع تست است، انجام می‌دهد. دغدغه در گرفتن عدم قطعیت در این پژوهش، مدل‌سازی واقعی‌تری از مسئله ایجاد تدشاهیلوای ایجادشده از این مدل قابلیت بیشتری برای پیاده‌سازی داشته باشند. درنیا تحقیق به بهبود در سرعت همگرایی به مینیمم سراسری هر یک از این توابع پرداخته و میزان و روش OSP و ACI با پارامترهای فازی استفاده کردیم که ضمن کاهش زمان محاسباتی تا افزایش سرعت همگرایی، مقایسه‌ای هم بین کارایی OSP فازی و ACI فازی انجام گرفت. نتیجه رسیدیم که الگوریتم ACI فازی نسبت به الگوریتم OSP فازی سرعت همگرایی بیشتری برخوردار است.

واژگان کلیدی: پارامتر پویا، الگوریتم فراابتکاری، منطق فازی، توابع تست.

h.eghbali@eyc.ac.ir

*. مهندسی صنایع، دانشگاه ایوان کی (نویسنده مسئول مکاتبات)

** مهندسی کامپیوتر، دانشگاه ایوان کی

1. Particle Swarm Optimization
2. Imperialist Competitive Algorithm

مجله مهندسی سیستم و بهره‌وری، سال اول، شماره ۲، بهار ۱۴۰۱، ص ۷۵ - ۸۹

مقدمه

این پژوهش به مقایسه کارایی الگوریتم‌های فراابتکاری (PSO و ICA) بر روی برخی توابع خاص، با انطباق پارامترهای فازی پرداخته است و با در نظر گرفتن عدم قطعیت، مدل‌سازی واقعی‌تری از مسئله ایجاد می‌کند. این روش‌های پیشنهادی بر روی برخی توابع تست اعمال شده‌اند که از تابع های آماده در متلب هستند. توابع مورد استفاده برای تست کردن الگوریتم‌های بهینه‌سازی است. به منظور نشان دادن عملی بودن روش پیشنهادی الگوریتم‌ها، از چهار تابع Ackley, rastrigin, rosenbrock و تابع کروی یا همان sphere استفاده کردیم. توابع تست برای ارزیابی ویژگی‌های الگوریتم‌های بهینه‌سازی سودمند هستند. برخی اوقات، تصورات اشتباهی در مورد مقایسه "منطق فازی" و "واقعیت" وجود دارد. به عنوان مثال، آیا حل مسائل مختلف در محیط فازی به واقعیت نزدیک‌تر است یا در محیط کلاسیک؟ آیا جواب‌هایی که در محیط کلاسیک و قاطع حاصل می‌شوند، به جواب‌های واقعی نزدیک هستند؟ در این تحقیق و مطالعات مشابه درمی‌یابیم که، پدیده‌های دنیای واقعی به دلیل عدم قطعیت بایستی در محیط فازی بررسی شوند (اقبالی و همکاران، ۱۳۹۱، ۱۳۹۳). به هر حال، روش پیشنهادی در این تحقیق یک روش نوین برای انطباق پارامتر پویا در بهینه‌سازی روش‌های فراابتکاری است که یکی از کاربردهای ریاضیات فازی را در علم کامپیوتر به تصویر می‌کشد. دو مشکل اصلی الگوریتم‌های ابتکاری، گیرافتادن آنها در نقاط بهینه محلی و همگرایی زودرس به این نقاط است. الگوریتم‌های فراابتکاری برای حل این مشکلات الگوریتم‌های ابتکاری ارائه شده‌اند. در واقع، الگوریتم‌های فراابتکاری برای یافتن پاسخ بهینه به کار می‌روند.

سیستم استنتاج فازی (FIS)

صنعت سیستم فازی همانند عملکرد مغز انسان که توانایی تفسیر اطلاعات حسی مبهم و ناقص را دارد، روشی سیستماتیک برای قضاوت از چنین اطلاعات زبانی را فراهم می‌آورد. سیستم فازی، فناوری جدیدی است که طراحی و مدل‌سازی یک سیستم نیازمند به ریاضیات پیچیده و پیشرفته را با استفاده از مقادیر زبانی و دانش فرد خیره ممکن می‌سازد. ساختار اساسی سیستم استنتاج فازی شامل دو زیرسیستم اصلی فازی‌ساز و پایگاه قواعد و یک زیرسیستم فرعی غیرفازی‌ساز است. فازی-ساز و غیرفازی‌ساز وظیفه تبدیل اطلاعات بیرونی به مقادیر فازی و بر عکس را برعهده دارند (مرادی و همکاران، ۱۳۹۱). مرکز سیستم‌های فازی، پایگاه دانش بوده به طوری که مؤلفه کلیدی سیستم‌های مبتنی بر دانش، مجموعه قواعد اگر-آنگاه (IF-THEN rules) به دست آمده از دانش یا تجارب انسانی است (کوهساری و همکاران، ۱۳۹۴؛ وانگ، ۱۹۹۷؛ باتونکو و کریوشا، ۲۰۰۲؛ کلس و کروز، ۲۰۰۲).

یک مجموعه فازی با یک تابع عضویت که مقادیر آن بین صفر و یک تغییر می‌کند، تعریف می‌شود. تابع عضویت دارای اعضایی با درجه عضویت متغیری بر اساس مقادیر تابع عضویت است. به طور کلی، دو نوع سیستم استنتاج فازی مطرح است: الف) سیستم استنتاج ممدانی (ممدانی، ۱۹۷۵)، ب) سیستم استنتاج سوگنو (سوگنو و تاکاجی، ۱۹۸۵). این دو سیستم استنتاج فازی خیلی مشابه هم هستند، عمده تفاوت آنها در این است که تابع عضویت خروجی در روش سوگنو ثابت و یا خطی است و نیازی به غیرفازیساز در آن وجود ندارد (آرشدیپ و آمیرت، ۲۰۱۲؛ کوهساری و همکاران، ۱۳۹۴).

رفنار سیستم‌های فازی به اطلاعات طراحی شامل: انتخاب توابع عضویت و قوانین کنترلی بستگی دارند. در روش‌های طراحی سنتی، اطلاعات طراحی مبتنی بر تجربه افراد خبره است که از طریق آزمون سعی و خطا تعیین می‌شود. بنابراین، طراحی یک سیستم فازی مناسب زمان بر است. بنابراین، انتخاب بهینه قوانین فازی و یا شکل توابع عضویت، مسئله بسیار مهمی است. و این مسئله بدون نیاز به تجربیات افراد خبره از اهمیت ویژه‌ای برخوردار است (عابدی نیا و امجدی، ۱۳۹۳). در این شرایط، می‌توان اطلاعات و داده‌های غیرقطعی را در قالب اعداد فازی بیان کرد (حسامی و مولایی، ۱۳۹۴).

الگوریتم‌های پیشنهادی

روش حل این پژوهش، بدین صورت است که ابتدا الگوریتم‌های مورد نظر (الگوریتم بهینه‌سازی ازدحام ذرات (کندی و ابرهات، ۱۹۹۵) و الگوریتم رقابت استعماری (لوکاس، ۲۰۰۷؛ آتش پز گرگری، ۱۳۸۷)) را فازی کرده، سپس برای تست کردن روش‌های پیشنهادی و اینکه بفهمیم که آیا الگوریتم‌های پیشنهادی خوب عمل کرده‌اند یا خیر، الگوریتم‌های پیشنهادی را بر روی یک مسئله اعمال یا تست کرده‌ایم:

۱. الگوریتم PSO فازی شده

محققان در PSO معمولی، میزان پارامترهای c_1 و c_2 را ۰٫۵ و ۲٫۵ در نظر می‌گیرند. به خاطر ثابت فرض شدن این دو متغیر، تنوع در ذرات کم است و به همگرایی خوبی منجر نمی‌شود (انگلبرت، ۲۰۰۵؛ یانگ و همکاران، ۲۰۰۷) و برای رفع این مشکل (همگرایی و تنوع)، یعنی بهبود همگرایی و تنوع در PSO، تطبیق پارامتر پویا در PSO انجام دادیم؛ یعنی پارامترهای c_1 و c_2 را فازی کردیم تا معیار همگرایی و تنوع در الگوریتم PSO بهبود یابند و این الگوریتم، قدرت همگرایی و تنوع بیشتری در جست و جوی نقاط بهینه مسئله داشته باشد (ملین و همکاران، ۲۰۱۳).

جدول ۱: میزان پارامترها برای هر روش

پارامترها	PSO ساده	فازی PSO ₁	فازی PSO ₂	فازی PSO ₃
C1	۰,۵	پویا	پویا	پویا
C2	۲,۵	پویا	پویا	پویا

فرمول‌های مورد استفاده در الگوریتم‌های پیشنهادی FPSO :

- برای اندازه‌گیری تکرارهای الگوریتم در معادله، درصدی از تکرارها استفاده شده است؛ یعنی هنگامی که الگوریتم شروع به تکرار می‌کند، مقدارش کم در نظر گرفته می‌شود و هنگامی که تکرار تمام می‌شود، مقدارش زیاد یا نزدیک ۱۰۰ درصد در نظر گرفته می‌شود. این ایده به صورت زیر نشان داده می‌شود:

معادله (۱) :

$$\text{Iteration} = \frac{\text{Current Iteration}}{\text{Maximum of Iteration}}$$

Current Iteration : تکرار جاری

Maximum of Iteration : کل تکرارهای الگوریتم

- اندازه‌گیری تنوع در معادله (۲) نشان داد که اندازه‌گیری درجه پراکندگی ذرات در زمانی که ذرات نزدیک به هم هستند، در آنجا کمترین تنوع وجود دارد و زمانی که ذرات از هم دورند بیشترین تنوع را دارد. معادله تنوع می‌تواند به عنوان متوسط فاصله اقلیدسی بین هر یک از ذرات و بهترین ذره در نظر گرفته شود.

معادله (۲):

$$\text{Diversity (S(t))} = \frac{1}{n_s} \sum_{i=1}^{n_s} \sqrt{\sum_{j=1}^{n_x} (x_{ij}(t) - x_{\square ij}(t))^2}$$

ns : بیان گر تعداد ذره‌ها

X_{ij} : ذره i ام در بعد j ام

X_{□ij} : بهترین ذره i ام در بعد j ام

- اندازه‌گیری خطا تبه وسیله معادله (۳) تعریف شده است؛ به طوری که برابر است با میانگین تفاوت بین برازندگی هر ذره و برازندگی بهترین ذره.

معادله (۳):

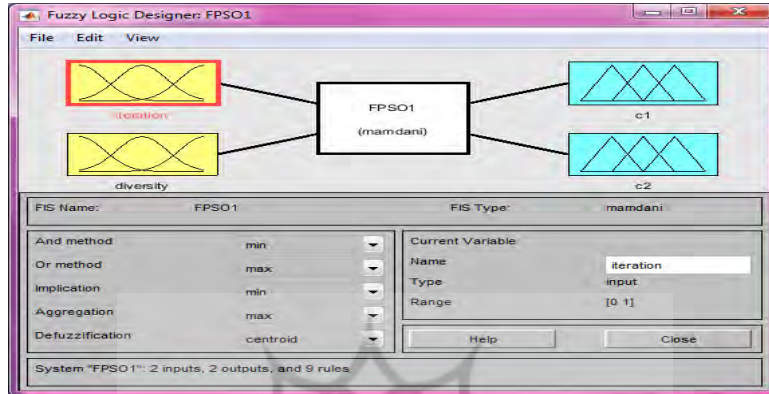
$$\text{Error} = \frac{1}{n_s} \sum_{i=1}^{n_s} (\text{Fitness}(x_i) - \text{MinF})$$

n_s : بیان گر تعداد ذره‌ها

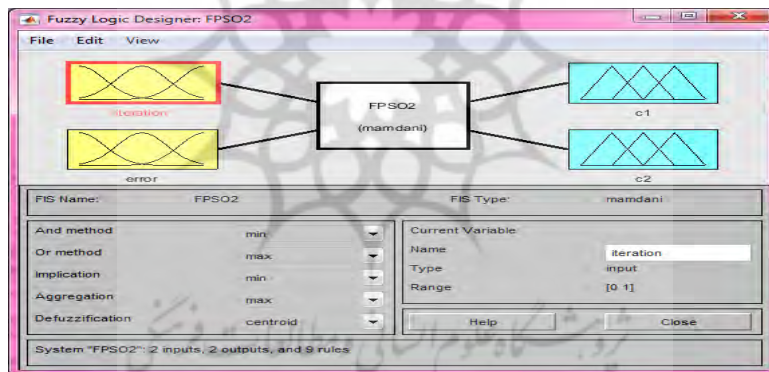
X_i : ذره i ام

$\text{Fitness}(x_i)$: برازندگی ذره i ام

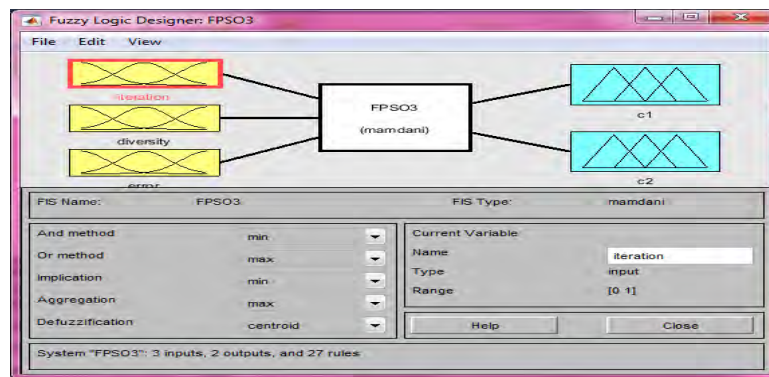
MinF : برازندگی بهترین ذره (کمترین برازندگی)



شکل ۱: سیستم فازی FPSO₁



شکل ۲: سیستم فازی FPSO₂



شکل ۳: سیستم فازی FPSO₃

در روش پیشنهادی FPSO₁، در هنگام طراحی سیستم فازی برای فازی کردن پارامترهای c_1 و c_2 در الگوریتم PSO، از دو ورودی تنوع و تکرار استفاده شده است؛ یعنی ورودی سیستم فازی پیشنهادی برای تطبیق پارامتر دادن c_1 و c_2 ، معیارهای تکرار و تنوع بوده است. الگوریتم PSO در هنگام شروع یعنی در اوایل تکرار و شروع بهتر است فضای جست و جو را برای یافتن و بهره‌برداری کردن بهترین ذره کاوش کند و برای محقق شدن این امر نیز بایستی دو معیار تکرار و تنوع لحاظ شود. به همین دلیل، چون سیستم پیشنهادی اول این دو معیار (تکرار و تنوع) را لحاظ کرده است، پس نتیجه بهتری را در رسیدن به مینیمم سراسری هر یک از این توابع تست می‌دهد.

۲. الگوریتم ICA فازی شده

محققان در ICA معمولی، میزان پارامتر بتا را ۱ یا ۲ در نظر می‌گیرند. به خاطر ثابت فرض شدن این متغیر، تنوع در ذرات کم است و به همگرایی خوبی منجر نمی‌شود (خبازی و همکاران، ۲۰۰۹؛ قاسمی، ۲۰۱۴) و برای رفع این مشکل (همگرایی و تنوع)، یعنی بهبود همگرایی و تنوع در ICA، تطبیق پارامتر پویا در ICA انجام دادیم؛ یعنی پارامتر بتا را فازی کردیم تا معیار همگرایی و تنوع در الگوریتم ICA بهبود یابد و این الگوریتم، قدرت همگرایی و تنوع بیشتری در جست و جوی نقاط بهینه مسئله داشته باشد.

جدول ۲: میزان پارامتر بتا برای هر روش

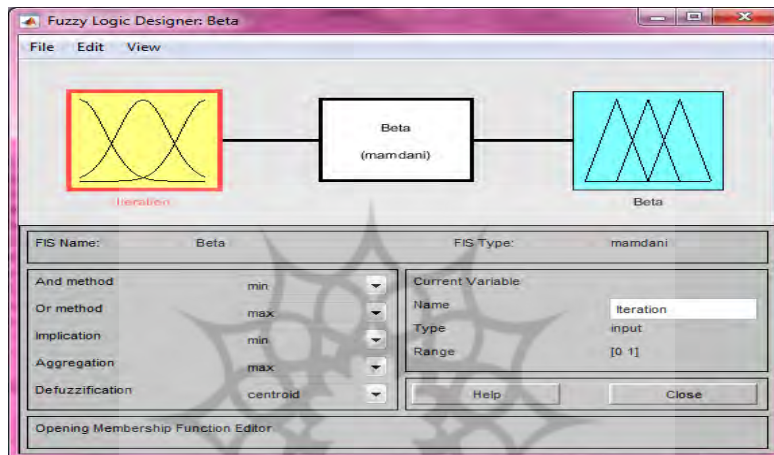
پارامترها	ICA ساده	ICA فازی
Beta	۱ یا ۲	پویا

فرمول مورد استفاده در الگوریتم پیشنهادی FICA:

- برای اندازه‌گیری تکرارهای الگوریتم در معادله (۴) از درصدی از تکرارها استفاده شده است؛ یعنی هنگامی که الگوریتم شروع به تکرار می‌کند، مقدارش کم در نظر گرفته می‌شود و هنگامی که تکرار تمام می‌شود، مقدارش زیاد یا نزدیک ۱۰۰ درصد در نظر گرفته می‌شود. این ایده به صورت زیر نشان داده می‌شود:
معادله (۴):

$$\text{Decade Factor} = \frac{\text{Current Decade}}{\text{Maximum of Decades}}$$

Current Decade: تکرار جاری
Maximum of Decade: کل تکرارهای الگوریتم



شکل ۴: سیستم فازی FICA

۳. اعمال الگوریتم‌های پیشنهادی بر روی توابع تست

توابع تست، از توابع بهینه‌سازی با نقاط بهینه سراسری مشخصی هستند، که برای استفاده در مسائل بهینه‌سازی به کار می‌روند و برای تست روش‌های مختلف بهینه‌سازی از این توابع استفاده می‌شود. به طور جزئی‌تر بخواهیم مسئله را روشن کنیم، مثلاً یک نمونه از توابع تست، تابع اکلای است، این تابع دارای مینیمم سراسری در میدا خود می‌باشد. حال ما روش پیشنهادی را بر روی این تست می‌کنیم تا ببینیم روش بهینه‌سازی پیشنهادی ما، این نقطه مینیمم سراسری را پیدا می‌کند؟ یا در بهینه‌های محلی گیر می‌افتد و موفق به یافتن این بهینه سراسری نمی‌شود. برای تست روش پیشنهادی نیز از چهار تابع Ackley, rastrigin, rosenbrock و تابع کروی یا همان

sphere استفاده کردیم. جزئیاتی از مشخصات توابع تست در نظر گرفته شده‌مان از جمله فرمول هر تابع، نموداری از تابع هدف، محدوده‌های متغیرهای هدف و مختصات بهینه سراسری در شکل ۵ آمده است.

نام تابع	نمودار	فرمول	مینیمم سراسری	دامنه جستجو
Ackley's function		$f(x, y) = -20 \exp \left(-0.2 \sqrt{0.5 (x^2 + y^2)} \right) - \exp (0.5 (\cos (2\pi x) + \cos (2\pi y))) + e + 20$	$f(0, 0) = 0$	$-5 \leq x, y \leq 5$
Sphere function		$f(x) = \sum_{i=1}^n x_i^2$	$f(x_1, \dots, x_n) = f(0, \dots, 0) = 0$	$-\infty \leq x_i \leq \infty, 1 \leq i \leq n$
Rosenbrock function		$f(x) = \sum_{i=1}^{n-1} [100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2]$	$\text{Min} = \begin{cases} n=2 \rightarrow f(1, 1) = 0, \\ n=3 \rightarrow f(1, 1, 1) = 0, \\ n>3 \rightarrow f(\underbrace{1, \dots, 1}_{(n) \text{ times}}) = 0 \end{cases}$	$-\infty \leq x_i \leq \infty, 1 \leq i \leq n$
Rastrigin function		$f(x) = An + \sum_{i=1}^n [x_i^2 - A \cos(2\pi x_i)]$	minimum at $f(0, \dots, 0) = 0$	$5.12 \leq x_i \leq 5.12$

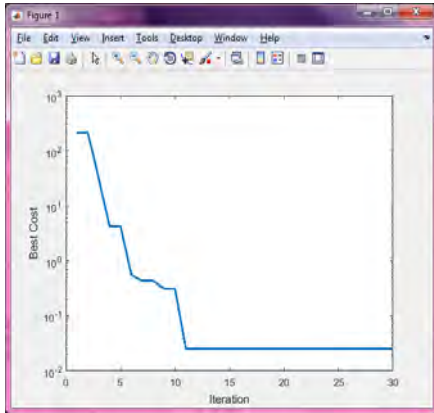
شکل ۵: توابع خاص در نظر گرفته شده برای تست روش‌های پیشنهادی

در روش FPSO

نحوه انجام این کار نیز به این شکل است که پارامترهای c_1 و c_2 را فازی کرده، سپس به الگوریتم PSO داده، یعنی همان سه الگوریتم پیشنهادی PSO فازی را هر یک به طور جداگانه بر روی توابع تست اعمال شده؛ یعنی یک بار الگوریتم FPSO₁ را بر روی هر یک از این توابع تست گفته شده امتحان می‌کند و یک بار الگوریتم FPSO₂ پیشنهادی را بر روی هر یک از این توابع تست گفته شده امتحان می‌کند و بار آخر الگوریتم FPSO₃ را بر روی هر یک از این توابع تست گفته شده امتحان می‌کند. نحوه تست کردن این توابع با استفاده از الگوریتم‌های پیشنهادی نیز مانند اعمال دیگر الگوریتم‌های بهینه‌سازی بر روی توابع تست است؛ یعنی این توابع تست به عنوان تابع هدف الگوریتم بهینه‌سازی پیشنهادی مان یعنی همان FPSO در نظر گرفته می‌شود و این الگوریتم با در نظر گرفتن هر یک از این توابع تست به عنوان تابع هدف یا برازندگی، کار خود را که رسیدن به مینیمم سراسری هر یک از این توابع تست است، انجام می‌دهد.

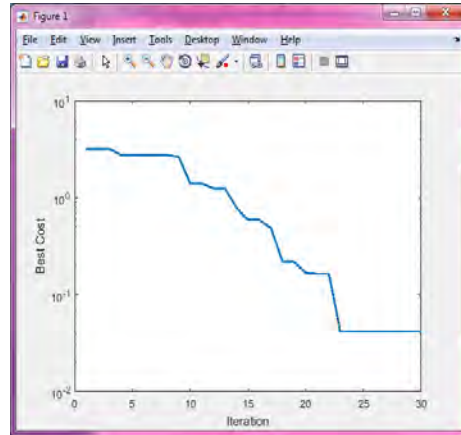
نتیجه اعمال الگوریتم پیشنهادی $FPSO_1$ بر روی توابع تست

تابع Rosenbrock :



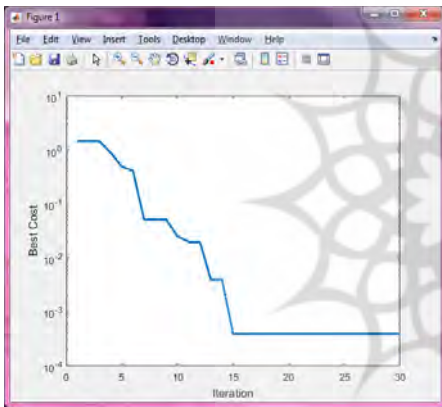
شکل ۸: الگوریتم $FPSO_1$ بر روی تابع Rosenbrock

تابع Ackley :



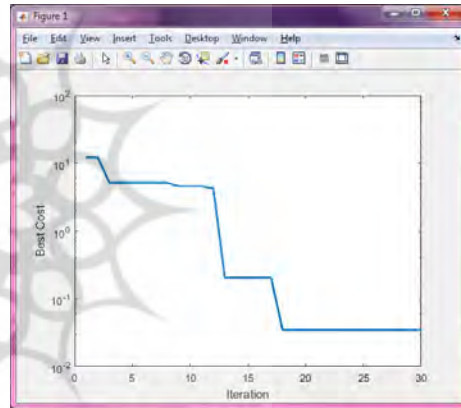
شکل ۶: الگوریتم $FPSO_1$ بر روی تابع Ackley

تابع Sphere :



شکل ۹: الگوریتم $FPSO_1$ بر روی تابع Sphere

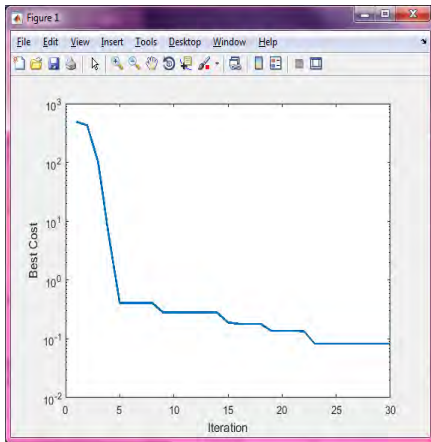
تابع Rastrigin :



شکل ۷: الگوریتم $FPSO_1$ بر روی تابع Rastrigin

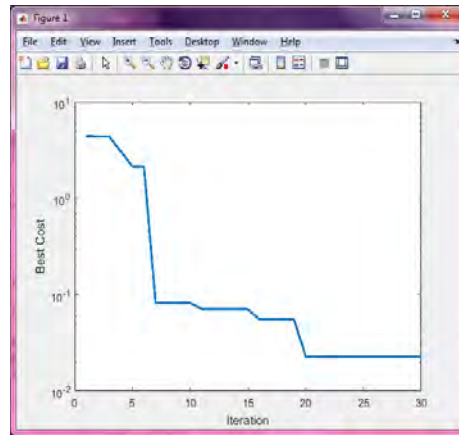
نتیجه اعمال الگوریتم پیشنهادی $FPSO_2$ بر روی توابع تست

تابع Rosenbrock:



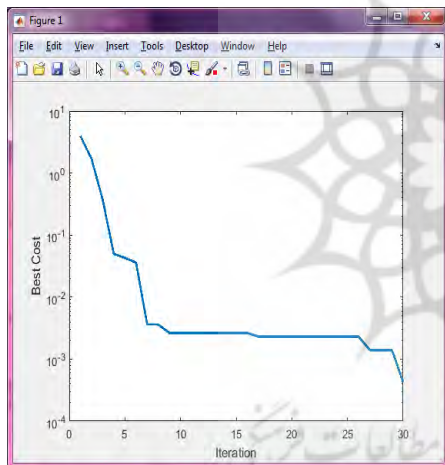
شکل ۱۲: الگوریتم $FPSO_2$ بر روی تابع Rosenbrock

تابع Ackley:



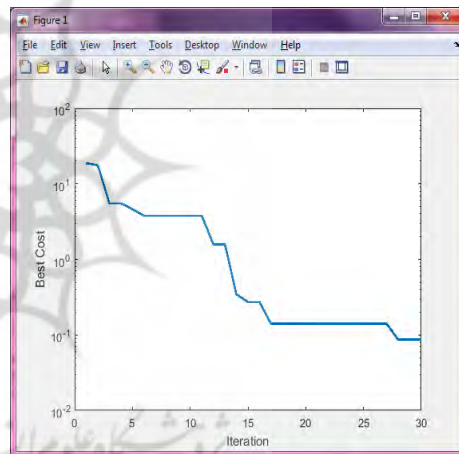
شکل ۱۰: الگوریتم $FPSO_2$ بر روی تابع Ackley

تابع Sphere:



شکل ۱۳: الگوریتم $FPSO_2$ بر روی تابع Sphere

تابع Rastrigin:

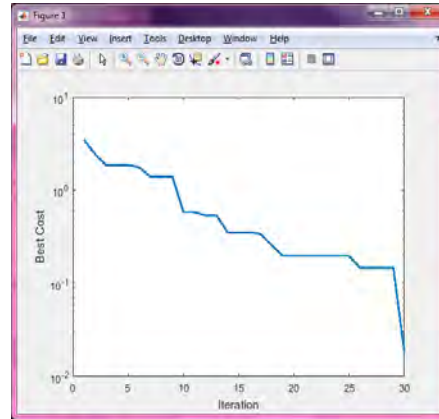
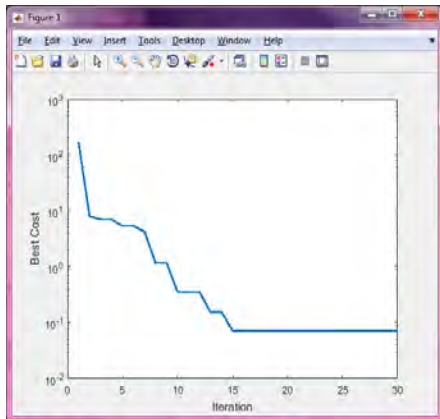


شکل ۱۱: الگوریتم $FPSO_2$ بر روی تابع Rastrigin

نتیجه اعمال الگوریتم پیشنهادی $FPSO_3$ بر روی توابع تست

تابع: Rastrigin

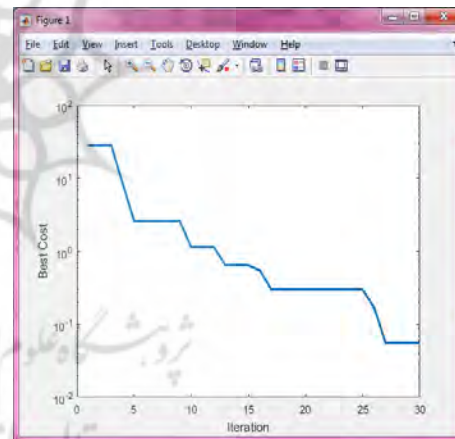
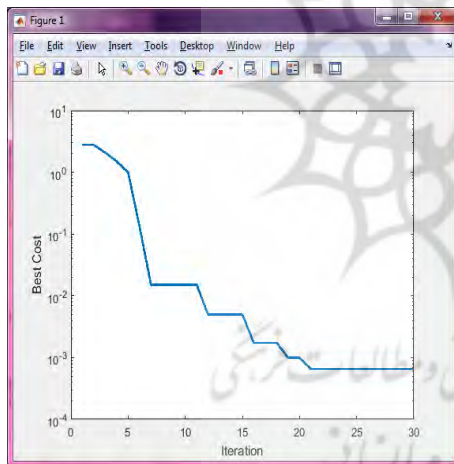
تابع: Ackley



شکل ۱۴: الگوریتم $FPSO_3$ بر روی تابع Ackley شکل ۱۵: الگوریتم $FPSO_3$ بر روی تابع Rastrigin

تابع: Sphere

تابع: Rosenbrock



شکل ۱۶: الگوریتم $FPSO_3$ بر روی تابع Rosenbrock شکل ۱۷: الگوریتم $FPSO_3$ بر روی تابع Sphere

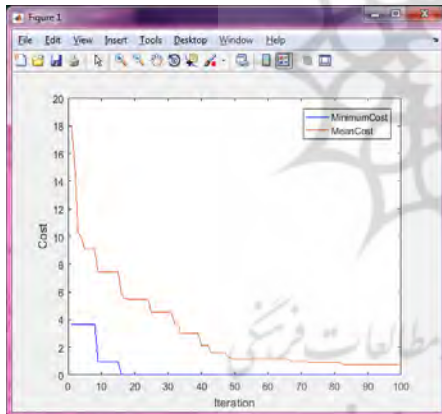
✓ همان طور که در شکل‌های بالا مشاهده می‌کنید، به نتایجی دست یافتیم که نشان می‌دهد الگوریتم $FPSO_1$ نسبت به الگوریتم‌های $FPSO_2$ و $FPSO_3$ از لحاظ سرعت همگرایی و گیر نکردن در دام مینیمم‌های محلی بهتر است. همان طور که مشاهده می‌شود الگوریتم $FPSO_1$ خیلی کم‌تر در دام مینیمم‌های محلی افتاده است. منحنی مینیمم هزینه $FPSO_1$ دارای شیب تندتری نسبت به منحنی مینیمم هزینه $FPSO_2$ و $FPSO_3$ است و این امر نشان می‌دهد که سرعت همگرایی $FPSO_1$ نسبت به $FPSO_2$ و $FPSO_3$ بیشتر است.

در روش FICA

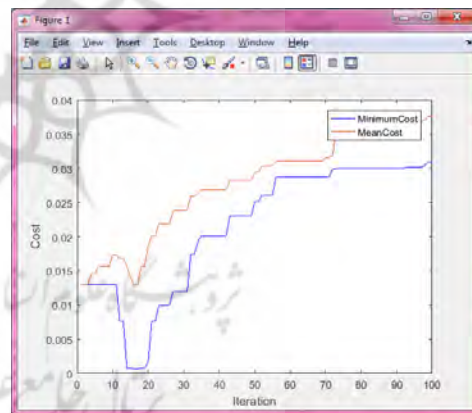
نحوه انجام این کار نیز به این شکل است که پارامتر بتا را فازی کرده، سپس به الگوریتم ica داده، یعنی همان الگوریتم پیشنهادی ica فازی را بر روی توابع تست گفته شده اعمال می‌کند. نحوه تست کردن این توابع با استفاده از الگوریتم‌های پیشنهادی نیز مانند اعمال دیگر الگوریتم‌های بهینه‌سازی بر روی توابع تست است؛ یعنی این توابع تست به عنوان تابع هدف الگوریتم بهینه‌سازی پیشنهادی مان یعنی همان FICA در نظر گرفته می‌شود و این الگوریتم با در نظر گرفتن هر یک از این توابع تست به عنوان تابع هدف یا برازندگی، کار خود را که رسیدن به مینیمم سراسری هر یک از این توابع تست است، انجام می‌دهد.

نتیجه اعمال الگوریتم پیشنهادی FICA بر روی توابع تست

تابع: Ackley

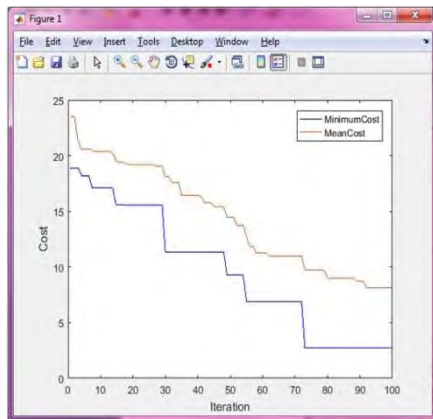


شکل ۱۹: الگوریتم FICA بر روی تابع Rastrigin

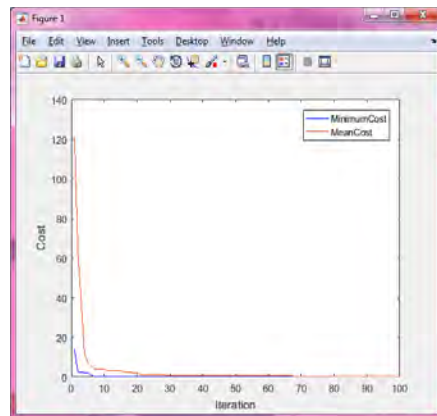


شکل ۱۸: الگوریتم FICA بر روی تابع Ackley

تابع: Rastrigin



شکل ۲۱: الگوریتم FICA بر روی تابع Sphere



شکل ۲۰: الگوریتم FICA بر روی تابع Rosenbrock

همان طور که در شکل‌های بالا مشاهده می‌کنید، به نتایجی دست یافتیم که نشان می‌دهد، الگوریتم FICA نسبت به الگوریتم $FPSO_1$ از لحاظ سرعت همگرایی و گیر نکردن در دام مینیمم‌های محلی بهتر است؛ زیرا فاصله بین منحنی مینیمم و متوسط بسیار ناچیز است که این امر نشان می‌دهد، در الگوریتم FICA تمام کشورها در نهایت به نقطه بهینه میل می‌کنند و خیلی کمتر در دام مینیمم‌های محلی افتاده است. همان طور که در شکل‌ها مشاهده می‌شود، منحنی مینیمم هزینه الگوریتم FICA دارای شیب تندتری نسبت به منحنی مینیمم هزینه الگوریتم $FPSO_1$ است و این امر نشان می‌دهد که سرعت همگرایی FICA نسبت به $FPSO_1$ بیشتر است.

نتیجه‌گیری

بنابر مقایسه‌ای که بین بهترین روش پیشنهادی $FPSO$ (یعنی $FPSO_1$) و FICA صورت گرفت، می‌توان گفت که در این پژوهش، الگوریتم رقابت استعماری فازی به نتیجه بهتری نسبت به الگوریتم بهینه‌سازی ازدحام ذرات فازی رسیده است و کیفیت نتایج را از لحاظ سرعت همگرایی بهبود می‌بخشد؛ زیرا الگوریتم FICA خیلی کمتر در دام مینیمم‌های محلی افتاده است. نتایج حاصل از آزمایشات، کارایی و سرعت همگرایی بالای الگوریتم پیشنهادی در الگوریتم رقابت استعماری فازی را تأیید می‌کنند.

منابع

- A.Abedinya, N.Amjady, *Fuzzy Stabilizer Design in Multi-Machine Power Systems Using Harmonic Search Algorithm*, Journal of Modeling in Engineering, Vol. 12, No. 36.1393. (in Persian)
- Arshdeep, K., Amrit, K, *Comparison of Mamdani-Type and Sugeno-Type Fuzzy Inference Systems for Air Conditioning System*, International Journal of Soft Computing and Engineering (IJSCE) ISSN: 2231-2307, Vol. 2, Issue-2, 2012.
- E. A. G. a. C. Lucas, *Imperialist Competitive Algorithm: An Algorithm for Optimization Inspired by Imperialist Competition*, IEEE Congressman Evolutionary Computation ,pp. 4661-4667, 2007.
- E.Atashpaz Gargari, *Development of Social Optimization Algorithm and its Performance Evaluation*, M.Sc., Faculty of Electrical and Computer Engineering, University of Tehran.1387. (in Persian)
- Engelbrecht, A, *Fundamentals of computational swarm intelligence*, University of Pretoria, South Africa, 2005.
- H.Eghbali, M.Eghbali, A.Vahidyan kamyad, R.Aminlo, *Dietary Optimization of Patients with Hepatitis Fuzzy Approach*, Journal of Operations Research and Its Applications, Ninth Year, No.2.1-20, 1391. (in Persian)
- H.Eghbali, M.a.Basuti, A.Esapour, *Eliminating Energy Protein Malnutrition in HIV-infected Patients Using Fuzzy Multi-Objective Linear Programming*, First National Conference on Extrinsic Algorithms and its Applications in Science and Engineering, Pardisan-Fereidunkar Institute of Higher Education.1393. (in Persian)
- H.Eghbali, T.Faghani, B.Ardestani, *Optimizing energy intake and fat intake in athletes nutrition program using fuzzy multi-objective linear programming model*, Proceedings of the Second National Conference on Industrial Engineering and Sustainable Management, Islamic Azad University, Isfahan ,1393. (in Persian)
- H.Kohsari, A.Najafi, H.Alielahi, M.Adampira, *Investigation of Factors Affecting Dynamic Density Operation in Fuzzy-Based Grain Soils*, Journal of Modeling in Engineering, Volume 13, Issue 43. 1394. (in Persian)
- H.Moradi, M.Tamana, H.Ansari, M.Naderyanfar, *Evaluation of Fuzzy Inference Systems for Estimating Hourly Reference Transpiration Evaporation Case Study: Fariman Region*, Journal of Water and Soil Conservation Research (Agricultural and Natural Sciences), Volume 19, Number 1,1391. (in Persian)
- J. Kennedy and R.C. Eberhart, *Particle swarm optimization*, in: Proceedings of IEEE International Conference on Neural Networks, Piscataway: IEEE, pp. 1942–1948, 1995.

- Khabbazi, A., Atashpaz-Gargari, E. and Lucas, C, *Imperialist competitive algorithm for minimum bit error rate beamforming*, International Journal of Bio-Inspired Computation, 1 , 125–133, 2009.
- Klose, A. and Kruse, R, *Enabling neuro-fuzzy classification to learn from partially labeled data*, In IEEE World Congress on Computational Intelligence. IEEE International Conference on fuzzy Systems, psge 32-42, 2002.
- Li-Xin Wang, *A Course in fuzzy systems and control*, Prentice Hall, NJ, 1997.
- Butenkov, S. and Krivsha, V, *Classification using Fuzzy Geometric Features*, . Proc. IEEE Conf. ICAIS'02, Divnomorskoe, Russia, 89-91, 2002.
- Mamdani, E. H., Assilian, S, *An experiment in linguistic synthesis with a fuzzy logic controller*, Int J Man Mach Stud, 7 , pp. 1-13, 1975.
- Melin, Patricia; Olivas ,Frumen; Castillo, Oscar; Valdez, Fevrier; Soria, Jose and Mario Valdez, *Optimal design of fuzzy classification systems using PSO with dynamic parameter adaptation through fuzzy logic*, Expert Systems with Applications 40, 3196–3206, 2013.
- S. G ,M. M. G ,R. M ,M. G. Mojtaba Ghasemi, *Application of imperialist competitive algorithm with its modified techniques for multi-objective optimal power flow problem: A comparative study*, Information Sciences, 281p. 225 – 242, 2014.
- S.Hesami, Z.Molaei, *Scheduling Optimization in Lean Thinking Construction Projects*, Journal of Modeling in Engineering, Thirteenth Year, Issue 04.1394. (in Persian)
- Sugeno, M., Takagi, T, *Fuzzy identification of systems and its application to modelling and control*, IEEE Trans Syst Man Cybern, No. 15, pp. 116–132, 1985.
- X. Yang, J. Yuan, J. Yuan and H. Mao, *A modified particle swarm optimizer with dynamic adaptation*, Applied Mathematics and Computation, Volume 189, Issue 2, pp. 1205-1213, 2007.