

معرفی الگوریتم بهبود یافته رقابت استعماری بر اساس رگرسیون چندمتغیره به کمک منطق فازی

محمد فیوضی^۱ سید کمال‌الدین موسوی مشهدی^۱ جواد حدادنیا^۲

^۱دانشکده مهندسی برق، دانشگاه علم و صنعت ایران، تهران، ایران
^۲دانشکده مهندسی برق و کامپیوتر، دانشگاه حکیم سبزواری، سبزوار، ایران

چکیده

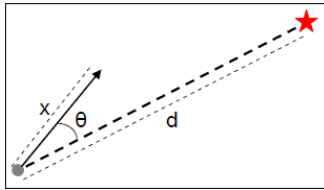
الگوریتم رقابت استعماری^۱ یکی از الگوریتم‌های بهینه‌سازی مبتنی بر جمعیت تصادفی است که از ایده تکامل در فرآیندهای سیاسی-اجتماعی بشر الهام گرفته شده است. در این الگوریتم بر پایه‌ی اطلاعات مربوط به بهترین کارایی قبل (کشورهای مستعمره و استعمارگر)، پارامترهای مسیر حرکت و اصل همگون‌سازی (جذب)، تنظیم می‌شوند. سپس تعداد کشورهای استعمارگر به همراه مستعمرات شان به صورت حرکت یک جمعیت از جواب‌ها، عمل جستجو را در فضای مساله انجام می‌دهند. در این روش هر یک از جواب‌های ممکن (کشورها) سعی می‌کنند به سمت بهترین تجربه بدست آمده (استعمارگر) همگرا شوند، بدنبال این فرایند نقطه بهینه تابع هدف توسط یک جستجوی هوشمند بدست می‌آید. تنظیم پارامترهای الگوریتم استاندارد نقش موثری در همگرایی به پاسخ بهینه در حل مسائل پیچیده با دامنه گسترده را دارد. سعی شد با استفاده از منطق فازی روشی برای کنترل پارامترهای ضریب جذب و هزینه مستعمرات به منظور افزایش دقت، سرعت و صحت الگوریتم رقابت استعماری ارائه شود، سپس توسط رگرسیون چندمتغیره، سیستم فازی در یک رابطه غیرخطی چندجمله‌ای خلاصه می‌شود که البته موجب کاهش حجم حافظه و محاسبه و افزایش سرعت اجرا را در پی دارد. بر این اساس در ابتدا با استفاده از دانش کافی و تسلط بر الگوریتم رقابت استعماری، سیستم فازی طراحی می‌شود، سپس رابطه چندجمله‌ای غیرخطی با استفاده از رگرسیون چندمتغیره استخراج می‌گردد. در هر مرحله از اجرای الگوریتم، موقعیت هر کشور (فضای جواب مساله) با توجه به ورودی‌های کنترلر فازی (نرمالیزه تکرار فرایند و تفاوت نقطه بهینه سراسری در هر تکرار) تعیین می‌شود تا در بدست آوردن موقعیت بعدی کشورها، توسط رابطه چندجمله‌ای غیرخطی، ضریب جذب و هزینه مستعمرات بدست آیند. روش پیشنهادی در حل مسائل معروف (مسائل محک) با الگوریتم‌های استاندارد^۳ ICA، GA^۳، PSO^۴ مقایسه شد و نتایج مقایسه، به منظور اثبات برتری روش پیشنهادی توسط شکل‌ها و جداولی ارائه می‌شود.

کلمات کلیدی: بهینه‌سازی، رقابت استعماری، بهینه سراسری و محلی، منطق فازی، رگرسیون چندمتغیره، جستجوی هوشمند.

۱- مقدمه

تکاملی^۵ (EA) زیادی برای حل مسائل بهینه‌سازی عمومی ارائه شده‌اند [۲، ۳]. این الگوریتم‌ها، با مدل کردن فرآیند تکامل در طبیعت، سعی در یافتن جواب بهینه برای مسئله را دارند [۴]. این موضوع از طریق به تکامل رساندن جمعیتی از راه حل‌های کاندید، مشابه فرآیند بیولوژیکی در تکامل جمعیت انجام می‌شود که می‌تواند با تغییرات محیطی سازگار گردد. الگوریتم‌های تکاملی گوناگونی تاکنون برای بهینه‌سازی ارائه شده‌اند که از آن میان می‌توان به الگوریتم ژنتیک (GA) ارائه شده توسط هالند^۶ در سال ۱۹۶۲

در بسیاری از مسائل بهینه‌سازی بویژه مسائل بزرگ انتخاب بهترین جواب از طریق جستجوی همه جانبه اگر چه غیرممکن نیست ولی کاری بسیار مشکل است. هدف مسائل بهینه‌سازی کاهش زمان جستجو و افزایش دقت و صحت جواب (بهینه تابع هدف) می‌باشد. امروزه با بزرگ شدن مسائل و پیچیده‌تر شدن آنها استقبال از روش‌های ابتکاری بطور چشم‌گیری افزایش یافته است [۱]. تاکنون الگوریتم‌های



شکل ۱- نحوه حرکت کشورها در فضای جستجو بر اساس الگوریتم ICA [۱۷]

در مسائل بهینه‌سازی با در نظر گرفتن تابع $f(x)$ سعی بر آن است تا آرگومان x به گونه‌ای یافته شود که هزینه متناظر با آن، بهینه (معمولاً کمینه) باشد. در یک مسئله بهینه‌سازی Nvar بعدی، یک کشور، یک آرایی $1 \times Nvar$ است. این آرایی به صورت رابطه (۱) تعریف می‌شود.

$$country = [p_1, p_2, p_3, \dots, p_{Nvar}] \quad (1)$$

در الگوریتم ICA برای شروع، تعداد "Ncountry" کشور اولیه ایجاد و "Nimp" از بهترین اعضای این جمعیت (کشورهای دارای کمترین مقدار تابع هزینه رابطه (۲)) به عنوان استعمارگر انتخاب می‌شوند.

$$cost_i = f(country_i) = f(p_1, p_2, p_3, \dots, p_{Nvar}) \quad (2)$$

باقیمانده "Ncol" از کشورها، مستعمراتی را تشکیل می‌دهند که هر کدام به یک امپراطوری تعلق دارند [۴]. استعمارگران با توجه به قدرتش، این مستعمرات را با رابطه (۳)، به سمت خود می‌کشند.

$$T.C_n = Cost(imperialist_n) + \xi \text{ mean } \{Cost(colonies\ of\ empire_n)\} \quad (3)$$

قدرت کل هر امپراطوری، با محاسبه قدرت هر دو بخش تشکیل دهنده آن یعنی قدرت کشور استعمارگر (قسمت اول معادله (۳))، به اضافه ξ برابر درصدی از میانگین قدرت مستعمرات آن، تعیین می‌شود. همانطور که در شکل (۱) دیده می‌شود، اگر فاصله بین کشور مستعمره تا استعمارگر برابر d باشد، حرکت کشور مستعمره به اندازه x و به سمت محل استعمارگر نظیر آن خواهد بود. البته این حرکت با زاویه θ منحرف می‌شود که مقدار حرکت x و زاویه θ به طور تصادفی تعیین می‌گردد. معمولاً مقدار زاویه θ به طور یکنواخت در بازه $[-\gamma, \gamma]$ و مقدار حرکت x به طور یکنواخت در بازه $[0, \beta d]$ انجام می‌شود مقادیر γ و β به عنوان پارامترهای الگوریتم ICA هستند که به ترتیب برابر ۴۵ درجه و "۲" پیشنهاد شده است [۱۷].

اگر در طول انجام الگوریتم و حرکت کشورها، یک کشور مستعمره قدرت بیشتری از استعمارگر نظیر خود پیدا کند، جای کشور مستعمره و استعمارگر عوض خواهد شد. به عبارت دیگر در مراحل بعدی اجرای الگوریتم، تمام کشورهای مستعمره استعمارگر قبلی، به استعمارگر جدید تعلق خواهند گرفت و حرکت این مستعمرات به سمت استعمارگر جدید خواهد بود. در هر مرحله از تکرار الگوریتم ICA، رقابتی استعماری میان استعمارگران برقرار است. در این رقابت، استعمارگری که نسبت به دیگر استعمارگران قدرت کمتری دارد، یکی از مستعمرات خود را از دست می‌دهد. اگر استعمارگری به دلیل از دست دادن مستعمرات خود، هیچ مستعمره‌ای نداشته باشد، آن استعمارگر خود به صورت

[۵، ۶، ۷] الگوریتم بهینه‌سازی حرکت دسته جمعی ذرات (PSO) توسط کندی^۷ و ابرهات^۸ در سال ۱۹۹۵ [۸، ۹، ۱۰]، الگوریتم شبیه سازی سرمایه‌های فلزات^۹ (SA) [۱۱-۱۴]، الگوریتم تکامل فرهنگی [۱۵] و سایر الگوریتم‌های موجود در این زمینه اشاره کرد. در سال ۲۰۰۷ الگوریتم جدیدی با نام رقابت استعماری توسط آتشیپ گرگری^{۱۱} و لوکاس^{۱۱} ارائه شد [۱۶] که نه از یک پدیده طبیعی، بلکه از یک پدیده اجتماعی-سیاسی الهام گرفته شده است [۴]. در این الگوریتم، کشورها (مستعمرات) با یک زاویه معین به سمت کشور استعمارگر حرکت می‌کنند تا در نهایت جواب یا نقطه بهینه تابع هدف بدست آید. می‌توان گفت که از جمله مشکلات مهم این الگوریتم احتمال گیر افتادن در دام بهینه محلی و تا حدودی پایین بودن سرعت همگرایی آن است.

در این تحقیق برای اولین بار با استفاده از منطق فازی، کنترل پارامترهای ضریب جذب (β) و هزینه مستعمرات (ξ) را توسط یک تابع غیرخطی هوشمند به منظور افزایش سرعت همگرایی، صحت و دقت الگوریتم استاندارد ICA ارائه کرده‌ایم، این امر موجب بهبود عملکرد الگوریتم ICA در حل مسائل بهینه‌سازی می‌شود. سپس توسط رگرسیون چند متغیره، سیستم فازی را به یک رابطه چندجمله‌ای غیرخطی تبدیل می‌شود. الگوریتم پیشنهادی را بر روی توابع محک آزمایش شده و نتایج را در جداول و شکل‌ها (نمودارها) بخش‌های بعد که همگی نشان دهنده برتری الگوریتم پیشنهادی می‌باشند، بررسی و اثبات می‌شود. در بخش دوم به بررسی الگوریتم استاندارد ICA پرداخته، در بخش سوم منطق فازی، در بخش چهارم ارزیابی الگوریتم پیشنهادی فازی رقابت استعماری، در بخش پنجم رگرسیون چند متغیره، سپس الگوریتم پیشنهادی و در آخر نتایج بدست آمده را بررسی و به اثبات بهبود کارایی الگوریتم پیشنهادی می‌پردازیم.

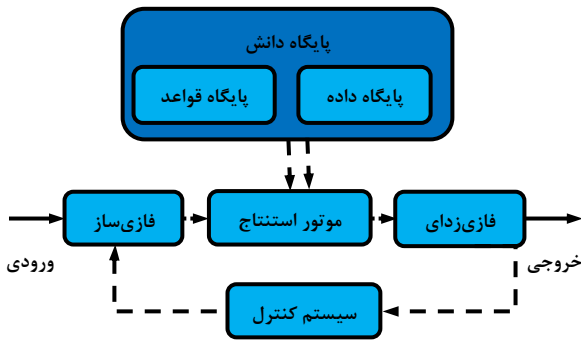
۲- الگوریتم استاندارد رقابت استعماری ICA

برای فهم بهتر الگوریتم ICA تلاش می‌کنیم رابطه‌ای میان الگوریتم GA و ICA برقرار کنیم. در الگوریتم GA تعدادی فرد وجود دارند که یک جمعیت را تشکیل می‌دهند. افراد جمعیت بر اثر عملگرهای ترکیب و جهش در فضای جستجو به دنبال پیدا کردن بهترین جواب جایجا می‌شوند. انتخاب والدین و انتخاب فرزندان جدید برای نسل بعد در الگوریتم GA براساس میزان برازش هر فرد انجام می‌شود [۴]. در الگوریتم ICA به جای افراد، کشورهایی وجود دارند که هر کشور مشابه یک فرد در الگوریتم GA دارای مشخصاتی است که مکان آن کشور را در فضای جستجو مشخص می‌کند.

در این مجموعه از کشورها که به عنوان نقاطی از فضای جستجو هستند، تعدادی کشور که میزان برازش بیشتری دارند به عنوان استعمارگر انتخاب می‌شوند. در الگوریتم ICA به جای اصطلاح میزان برازش از اصطلاح قدرت (Power) استفاده می‌شود. به این ترتیب کشورهای قدرتمند به عنوان استعمارگر و کشورهای ضعیف به عنوان مستعمره قرار می‌گیرند. هر چه قدرت استعمارگر بیشتر باشد، تعداد کشورهای مستعمره بیشتری را به خود اختصاص خواهد داد. در ابتدای الگوریتم، کشورها به طور تصادفی تولید می‌شوند و چند کشور قدرتمند به عنوان استعمارگر انتخاب می‌شوند. سپس سایر کشورها به طور تصادفی به یکی از استعمارگران منتسب می‌شوند، به طوری که تعداد مستعمرات هر استعمارگر متناسب با قدرتش خواهد بود.

حرکت در فضای جستجو به این شکل می‌باشد که هر کشور در راستای کشوری که مستعمره آن است به صورت تصادفی حرکت می‌کند. به عنوان مثال اگر یک استعمارگر را با ستاره و کشور مستعمره آن را با دایره نشان دهیم، حرکت کشور مستعمره به سمت کشور استعمارگر مطابق شکل (۱) خواهد بود [۱۷].

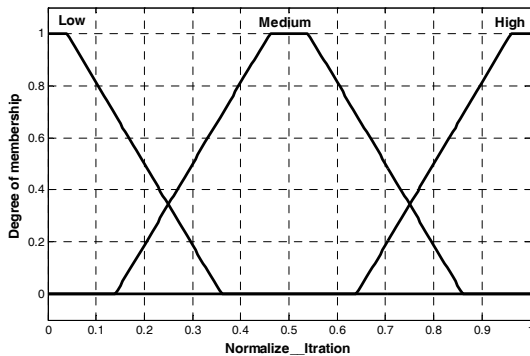
نقاط فضا به درجه عضویت (عددی بین ۰ یا ۱) می‌باشد [۱۹، ۲۰]. هر سیستم فازی تعدادی قانون دارد که بصورت عبارات IF_THEN در بانک قواعد آن ذخیره می‌گردد. مرحله پردازش که به آن موتور استنتاج گفته می‌شود بر اساس بانک قواعد فازی عمل می‌کند. ۲ نوع سیستم استنتاج فازی ممدانی و سوگنو وجود دارد. این ۲ نوع سیستم تا حدودی در روش تعیین فازی ساز ورودی‌ها و عملگرهای فازی شبیه هم هستند. اما در روش تعیین خروجی با هم تفاوت دارند [۲۱]. ما در طراحی سیستم مورد نظر، از سیستم استنتاج ممدانی استفاده کرده‌ایم. تمامی مراحل سیستم فازی در فلوچارت (۲) مشخص شده است.



شکل ۳- تمام مراحل یک سیستم فازی

۴-۱- پیاده‌سازی سیستم فازی

سیستم فازی طراحی شده ۲ ورودی (نرمالیزه تکرار فرآیند و تفاوت نطقه بهینه سراسری) و دو خروجی ضریب جذب و هزینه مستعمرات دارد. توابع عضویت در این سیستم از نوع ذوزنقه‌ای برای هر ورودی و خروجی مانند شکل (۴) می‌باشند.



شکل ۴- تابع عضویت برای سیستم فازی، اینجا تابع ورودی اول

محدوده ورودی اول طبق رابطه (۴) در بازه [۰ ۱] مشخص می‌شود.

$$Normalize_Iteration(i) = \frac{iter(i)}{\max(iter)} \quad (4)$$

محدوده ورودی دوم نیز طبق رابطه (۵) بر اساس شکل (۵) در بازه [0 1] مشخص می‌شود.

$$D_Best = \frac{1}{Npar} (MeanCost - MinimumCost) \quad (5)$$

مستعمره یک استعمارگر دیگر در خواهد آمد. مراحل الگوریتم ICA به همین ترتیب ادامه می‌یابد، تا تعداد استعمارگران به یک برسد. در این حالت تمام کشورها، مستعمره یک استعمارگر هستند و الگوریتم به پایان می‌رسد بدین ترتیب نقطه بهینه تابع هدف مشخص می‌شود، تمام مراحل و پارامترها این الگوریتم در فلوچارت (۱) و جدول (۱) بیان شده است.



شکل ۲- مراحل اجرای الگوریتم پیشنهادی

۳- روش پیشنهادی

مشکل عمده و اساسی الگوریتم رقابت استعماری بیان شد، ابتدا توسط سیستم فازی، کنترل کننده‌ای در قالب یک تابع غیرخطی به‌منظور برآورد و کنترل ضرایب جذب و هزینه مستعمرات طراحی می‌شود، سپس توسط رگرسیون چند متغیره، رابطه یا تابع غیرخطی برگرفته از تابع غیرخطی که نمایانگر رفتار سیستم فازی است، تعریف می‌شود. سپس رابطه غیرخطی در الگوریتم اصلی به جای سیستم فازی جایگذاری می‌شود، با این کار تمام سیستم فازی طراحی شده در یک رابطه چند جمله‌ای خلاصه می‌شود. در انتها به منظور اثبات همگرایی و کارایی الگوریتم پیشنهادی، از توابع محک استفاده می‌شود.

۴- منطق فازی

منطق فازی روش نسبتاً جدیدی است که در مقابل روش‌های مرسوم برای طراحی و مدل سازی سیستم‌هایی که نیازمند ریاضیات و آمار پیچیده هستند به کار می‌روند. منطق فازی اولین بار توسط پروفیسور زاده [۱۸] معرفی شد. در منطق کلاسیک همه متغیرها در عبارت باینری بیان می‌شوند. منطق فازی گزاره‌های صحیح بولی را با درجه درستی بیان می‌کند. تابع عضویت یک مجموعه فازی، در حالت خاص یک منحنی بوده و نشان دهنده چگونگی نگاشت هر یک از

دو خروجی سیستم فازی (ضریب جذب و هزینه مستعمرات) می‌باشند. دلایل انتخاب این دو پارامتر بعنوان خروجی‌ها، وابستگی زیاد الگوریتم استاندارد ICA به این پارامترها می‌باشد.

نتیجه می‌رسیم که برای یکسان بودن شانس مستعمره برای همه، باید ضریب جذب در ابتدا زیاد و هزینه مستعمره کم باشد [۱۶]. همچنین برای فرار از نقاط بهینه محلی باید در انتهای الگوریتم ضریب جذب کاهش یابد و نیز برای افزایش سرعت در رسیدن به قله سراسری و نقطه بهینه باید در انتهای الگوریتم ضریب هزینه مستعمره زیاد و در حد استاندارد قرار گیرد [۱۶]. به این منظور برای ایجاد یک رابطه غیرخطی مناسب بین این ورودی‌ها و خروجی‌ها در این مقاله از یک سیستم فازی استفاده می‌شود. قسمت دیگر سیستم فازی بانک قواعد آن می‌باشد که براساس مطالب بیان شده بصورت جدول (۲) طراحی می‌گردد.

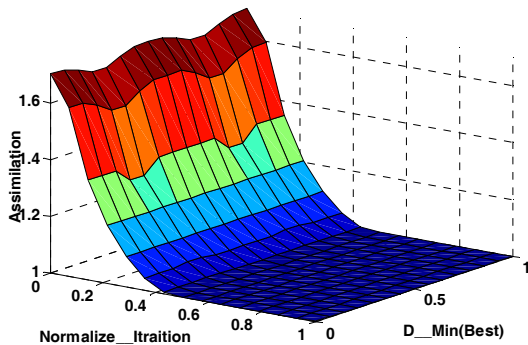
مرحله پردازش که به آن موتور استنتاج گفته می‌شود بر اساس بانک قواعد فازی عمل می‌کند. مشخصات سیستم فازی طراحی شده در جدول (۳) آمده است.

جدول ۳- مشخصات سیستم فازی طراحی شده

استلزام	ممدانی	تعداد متغیرها	2in / 2out
اشتراک	min	تعداد توابع عضویت	3MF
اجتماع	max	نوع توابع عضویت	Trap MF
استنتاج	min	فازی زدا	مرکزقل

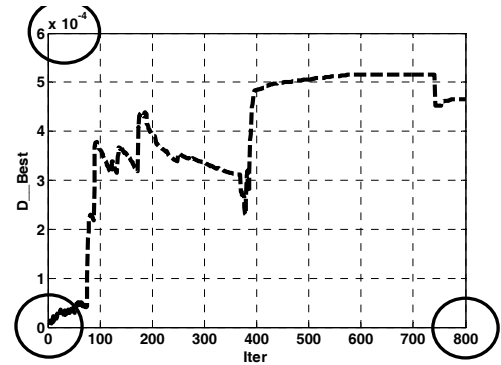
با ورود متغیرهای ورودی، سیستم فازی در داخل برنامه اصلی جایگذاری می‌شود، با توجه به بانک قواعد و روش‌های استنتاج و استلزام، سیستم فازی خروجی‌های مناسب را با توجه به جداول (۲ و ۳) (یعنی β و γ) ضرایب جذب و هزینه مستعمرات) به برنامه اصلی پاس می‌دهد. این منجر به بهبود کاوش و افزایش سرعت مشهود در همگرایی الگوریتم نهایی می‌شود، البته (رابطه سطحی بین ورودی‌ها و خروجی‌ها) در شکل (۷) بیان شده است.

پس از اعمال قوانین براساس توابع عضویت‌ها، Surface کنترلی نهایی بدست می‌آید. هر Surface نشان دهنده میزان تغییرات یک خروجی بر اساس دو ورودی است. در این سیستم به ازای هر خروجی یک Surface تعریف می‌شود، دو Surface در این سیستم متفاوت هستند. Surface با توجه به شکل‌های بدست آمده، صحت قوانین و توابع عضویت متغیرهای ورودی و خروجی و نیز نرمی و عملکرد آنها در شکل‌های (۷ و ۸) رسم شده است که به نمایش درآمده است.



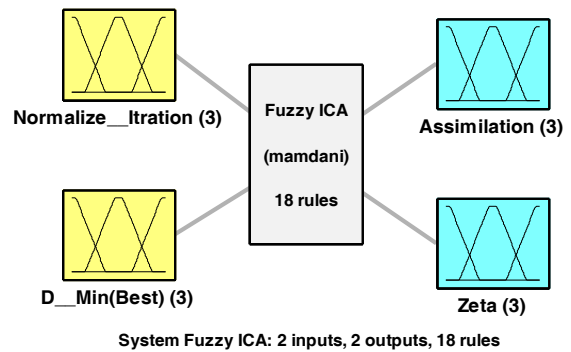
شکل ۷- سطح فیزیکی بین ورودی‌ها و خروجی اول

رابطه سطحی بین ورودی‌ها و خروجی دوم در شکل (۸) رسم شده است.



شکل ۵- محدوده مشخص شده برای ورودی دوم

تمام پارامترها و سیستم فازی طراحی شده در شکل (۶) مشخص شده است.



شکل ۶- سیستم فازی طراحی شده

الگوریتم رقابت استعماری که در [۴] ارائه شده است دارای تنظیمات و پارامترهایی خاص می‌باشد، که ما در این تحقیق از این پارامترها استفاده می‌نماییم. البته تمام مشخصات مذکور در جدول (۱) مشخص شده است.

جدول ۱- مشخصات سیستم فازی

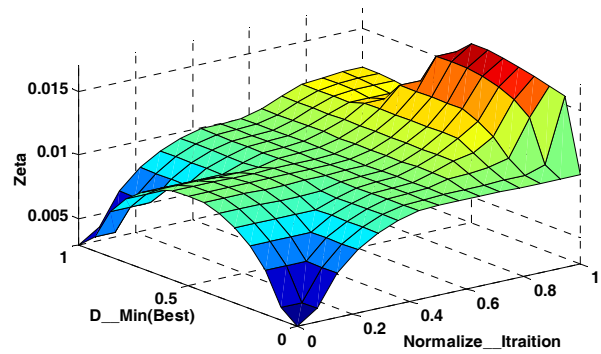
Parametr	Value
Num of Country	500
Num of Initial Imperialists	10
Num of Decades	800
Revolution Rate	0.4
Assimilation Coefficient	Set By Proposed ALG
Assimilation Angle Coefficient	0.05
Zeta	Set By Proposed ALG
Damp Ratio	0.99
Uniting Threshold	0.02

جدول ۲- بانک قواعد سیستم فازی طراحی شده

D_Min(Best)	High	Assimilation: High Zeta: High	Assimilation: Med Zeta: Med	Assimilation: Low Zeta: Med
	Medium	Assimilation: High Zeta: Med	Assimilation: High Zeta: Med	Assimilation: Med Zeta: High
Low	Assimilation: High Zeta: Low	Assimilation: Med Zeta: Med	Assimilation: Low Zeta: Med	Assimilation: Low Zeta: Med
		Low	Medium	High
		Normalize_Itration		

جدول ۴- صورت ریاضی توابع محک [۴]

Function	Relation	Range
Rastregin Or F1	$f(x) = \sum_{i=1}^D (x_i^2 - 10 \times \cos(2\pi x_i) + 10)$	(-10,10)
Rosenbrock Or F2	$f(x) = \sum_{i=2}^D (100 \times (x_{i+1} - x_i^2)^2 + (x_i - 1)^2)$	(-100,100)
Sphere Or F3	$f(x) = \sum_{i=1}^D x_i^2$	(-100,100)
Griewank Or F4	$f(x) = \frac{1}{4000} \times \sum_{i=1}^D x_i^2 - \prod_{i=1}^D \cos(\frac{x_i}{\sqrt{i}}) + 1$	(-600,600)



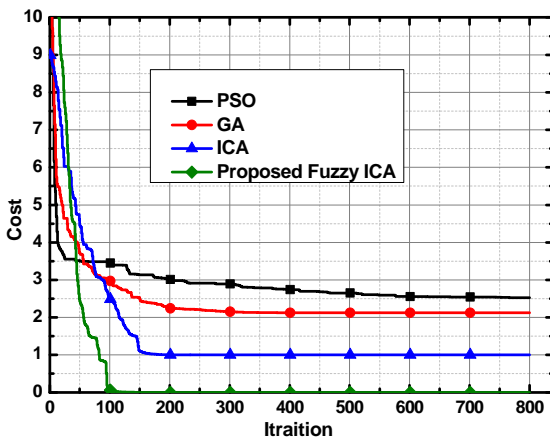
شکل ۸- سطح فیزیکی بیان کننده رابطه بین ورودی‌ها و خروجی دوم

اولین ارزیابی بر روی تابعی ۱۵ بعدی انجام می‌شود که برای اولین بار الگوریتم ICA بر روی آن پیاده‌سازی شد [۱۶، ۲۲]. الگوریتم استاندارد و نیز الگوریتم فازی ICA هر دو بر روی آن پیاده‌سازی شد، سپس به منظور مقایسه بهتر و جامع‌تر از دو الگوریتم ژنتیک (GA) استاندارد [۳۵] و همینطور الگوریتم بهینه‌سازی گروهبی ذرات (PSO) استاندارد [۳۶] (با پیش فرض آشنایی با این دو الگوریتم) به منظور اعمال روی توابع جدول [۴] استفاده می‌نماییم که پارامترهای تنظیمی این دو الگوریتم نیز در جدول (۵) معرفی شده‌اند.

جدول ۵- پارامترهای قابل تنظیم در الگوریتم‌های PSO، GA

Algorithm	Setting	Setting
GA	pCrossover(Recom)=0.7	pMutation=0.2
PSO	W = 0.729 C ₁ =1.4962	C ₂ =1.4962

پس از اعمال الگوریتم‌ها و همینطور پارامترهای تنظیمی آن‌ها، اولین نتیجه مقایسه در شکل (۱۰) رسم شده است.

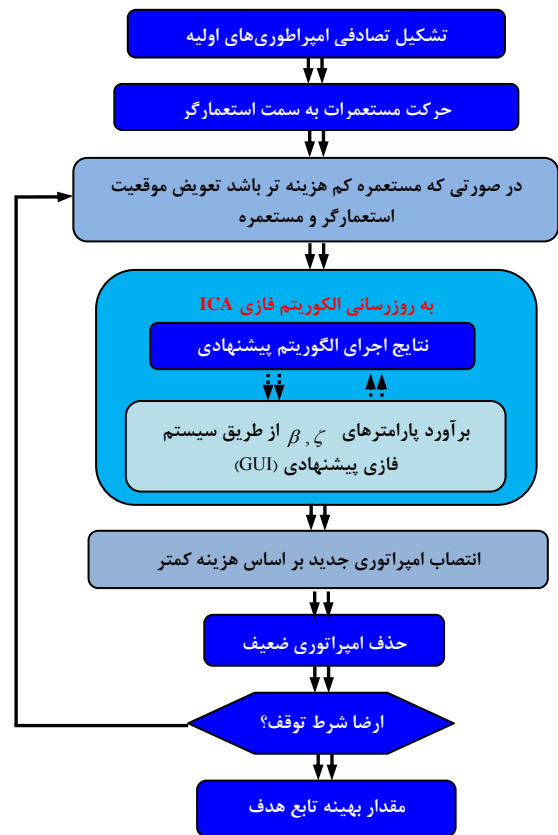


شکل ۱۰- ارزیابی الگوریتم فازی رقابت استعماری با الگوریتم‌های ICA استاندارد، PSO و GA در حل توابع از [۱۶، ۲۲]

به منظور بررسی بهتر و دقیق‌تر سیستم فازی طراحی شده برای بهبود عملکرد الگوریتم رقابت استعماری، نتایج اولین ارزیابی که در شکل (۱۰) مشخص شده‌اند، را بصورت لگاریتمی در شکل (۱۱) نمایش می‌دهیم. توابع جدول (۴)، دارای نقاط بهینه محلی فراوان و از طرف دیگر دارای نقطه بهینه سراسری هستند. به همین منظور از ۴ تابع مشهور جدول (۴) از [۴] در ابعاد ۵۰، ۱۰۰ و ۲۰۰ استفاده می‌کنیم، هدف مینیمم‌سازی این توابع در نقطه صفر است، نتایج بدست آمده در بهینه‌سازی تابع Rastregin در ۵۰، ۱۰۰ و ۲۰۰ بعد در شکل‌های (۱۲) و (۱۳) و (۱۴) مشاهده می‌شود.

۵- ارزیابی الگوریتم فازی ICA (Fuzzy ICA)

در الگوریتم ICA اساس کار بر روی تغییر وضعیت و جستجوی فضای مساله توسط استعمارگر می‌باشد. برای بررسی میزان صحت و دقت، الگوریتم‌ها در نرم‌افزار MATLAB® با استفاده از راهنمایی‌های [۱۸، ۱۹] بر روی توابع ریاضی مسائل محک [۴] مطابق با جدول (۴) شبیه‌سازی شدند و مورد ارزیابی قرار گرفته‌اند. فلوجارت الگوریتم فازی رقابت استعماری طبق فلوجارت (۳) می‌باشد. همچنین برای رسم نمودارهای بدست آمده از نرم‌افزار Origin® استفاده شد.

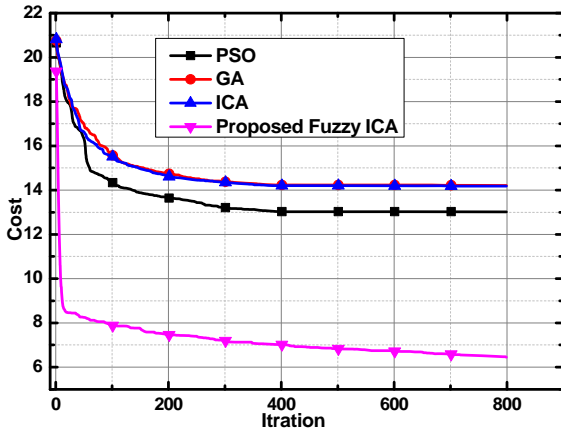


شکل ۹- روند اجرای الگوریتم رقابت استعماری پیشنهادی

توابع ارزیابی پیشنهادی در مرجع اصلی الگوریتم رقابت استعماری در جدول (۴) بیان شده‌اند.

تهیه شده است)، دقت مناسبی ندارد، و خیلی کندتر نیز به همگرایی رسیده است. شکل (۱۳) روند همگرایی الگوریتم پیشنهادی در بهینه‌سازی تابع Rastregin ۱۰۰ بعدی را نشان می‌دهد.

شکل ۱۴ نیز مشخص است که الگوریتم استاندارد و سایر الگوریتم‌ها، سرعت و مقداری را که بهینه کرده‌اند، ضعیف تر و کندتر از الگوریتم پیشنهادی بوده است.

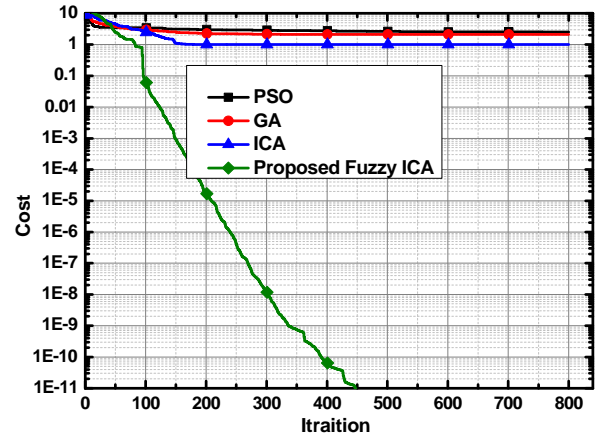


شکل ۱۴- تقابل الگوریتم پیشنهادی و الگوریتم‌های مذکور در ۲۰۰ بعد

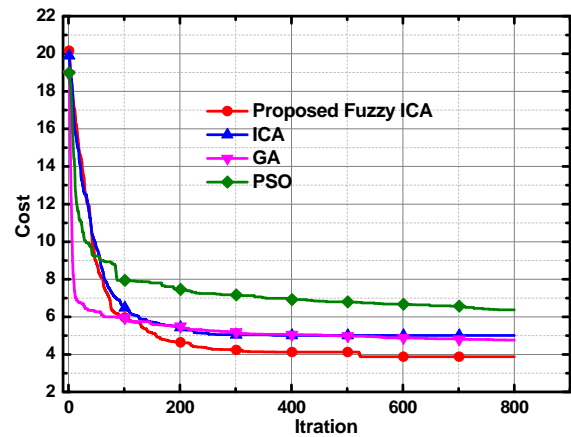
در نهایت در اشکال (۸ و ۹ و ۱۰ و ۱۱)، اجرای الگوریتم‌های مذکور در شرایط یکسان (تعداد تلاش الگوریتم برای همگرایی یا به عبارتی دیگر ایجاد شرط ارضاء یکسان) پس از میانگین گیری در ۱۰۰ بار اجرا، بهترین نتیجه به ازای ۸۰۰ بار اجرای هر الگوریتم دیده می‌شود. تا این مرحله، سیستم فازی پیشنهادی مورد ارزیابی قرار گرفت. براساس نتایج مشخص شده در اشکال (۸ و ۹ و ۱۰ و ۱۱) به خوبی برتری الگوریتم فازی رقابت استعماری مشخص می‌شود. پس می‌توان نتیجه گرفت که سیستم فازی طراحی شده که در الگوریتم رقابت استعماری جایگذاری شده است، به خوبی الگوریتم رقابت استعماری را هدایت کرده و با توجه به مقادیر ورودی برای این الگوریتم، خروجی‌های ضریب جذب و هزینه مستعمرات را کنترل و هدایت کرده است.

در پیاده‌سازی سیستم فازی مشکل اصلی که وجود دارد در مرحله اول این است که، خود سیستم فازی به علت انجام محاسبات مطمئناً در افزایش زمان محاسبات دخیل خواهد بود و نکته مهم تر این که برای تعمیم الگوریتم فازی رقابت استعماری در زبان‌های دیگر برنامه نویسی (واسط کاربری یا GUI که موجب طراحی سیستم فازی شده است، فقط در زبان MATLAB وجود دارد) مانند، ++C و Java، ... باید بتوانیم سیستم فازی را پیاده‌سازی نماییم، که البته استفاده از واسط کاربری (GUI) امکان پذیر نمی‌باشد و از طرف دیگر اگر سعی در نوشتن سیستم فازی به همان زبان کنیم، حتماً کاری سخت، طاقت فرسا و سرعت گیر خواهد بود، مضافاً بر این که در کنار کاهش حجم محاسبات به دلیل کاهش حجم حافظه، کاهش انرژی و افزایش سرعت در کنار کاهش چشمگیر هزینه‌ها را نیز می‌توان در نظر گرفت.

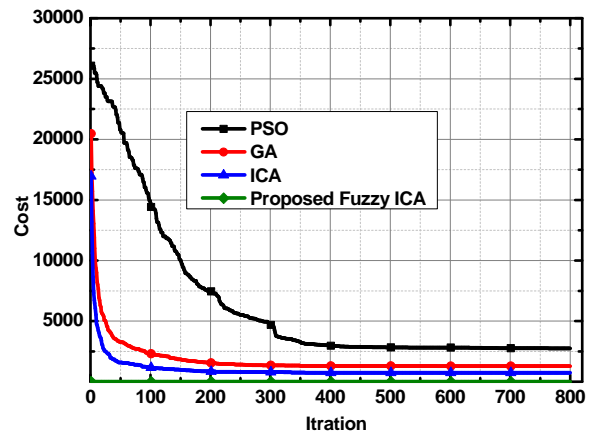
رگرسیون چندمتغیره یکی از الگوریتم‌های پیشنهادی است که می‌تواند اصطلاحاً سیستم فازی را در یک رابطه چند جمله‌ای خلاصه نماید. برای اعمال رگرسیون چند متغیره، باید به رفتار سیستم مورد نظر دسترسی داشته باشیم، یکی از پارامترهایی که موجب شناسایی و تعریف عملکرد سیستم فازی می‌شود، سطح فیزیکی یا Surface است، که بیان کننده رابطه بین ورودی‌ها و خروجی‌های یک سیستم (در اشکال ۵ و ۶) می‌باشد. در قسمت بعد به نحوه استخراج پارامترهای سیستم فازی یا خلاصه سازی سیستم فازی در یک رابطه غیرخطی چند جمله‌ای می‌پردازیم.



شکل ۱۱- نمایش لگاریتمی الگوریتم رقابت استعماری فازی در حل تابع [۱۶]، [۲۲] که در اولین ارزیابی الگوریتم رقابت استعماری بیان شده است



شکل ۱۲- تقابل الگوریتم پیشنهادی فازی رقابت استعماری و الگوریتم‌های مذکور در ۵۰ بعد



شکل ۱۳- تقابل الگوریتم پیشنهادی و الگوریتم‌های مذکور در ۱۰۰ بعد

در شکل‌های ۱۲ و ۱۳ و ۱۴ مقایسه عملکرد الگوریتم پیشنهادی با الگوریتم‌های مذکور را در بهینه‌سازی تابع Rastregin به ترتیب به ابعاد ۵۰، ۱۰۰ و ۲۰۰ نشان می‌دهد. در این شکل‌ها به خوبی مشخص است که الگوریتم پیشنهادی از همان ابتدا همگرا شده و کمترین مقادیر را در بهینه‌سازی بدست آورده است، اما الگوریتم‌های دیگر حتی ICA (با توجه به برنامه‌ای که در [۲۲]

۶- الگوریتم رقابت استعماری پیشنهادی

در این قسمت از الگوریتم پیشنهادی که در نهایت منجر به افزایش سرعت و دقت الگوریتم استعماری می‌شود، سعی بر آن داریم تا قسمت فازی الگوریتم رقابت استعماری فازی به یک تابع غیرخطی خلاصه شود، یعنی توسط یک رابطه چند جمله‌ای عملاً قسمت فازی حذف می‌شود و به جای آن از یک رابطه چند جمله‌ای استفاده می‌کنیم تا مقادیر جذب و هزینه متعمرات برآورد شوند تا از آنها در رابطه نهایی استفاده شوند. برای این کار لازم است تا از رفتار سیستم فازی اطلاع یابیم، سیستم فازی طراحی شده توسط سطح فیزیکی (Surface) در واسط کاربری (GUI) بیان کننده رابطه بین ورودی‌ها و خروجی‌های سیستم فازی است. با اعمال رگرسیون چند متغیره بر روی سطح فیزیکی، روابط چند جمله‌ای بدست خواهند آمد. به همین منظور در قسمت‌های بعدی ابتدا رگرسیون چند متغیره بحث و بررسی می‌شود سپس الگوریتم نهایی و در آخر ارزیابی الگوریتم پیشنهادی بیان می‌شود.

۶-۱- رگرسیون چند متغیره

یکی از هدف‌های اصلی پژوهش‌های آماری ایجاد روابطی است تا پیش بینی یک یا چند متغیر را بر حسب سایرین میسر گرداند. مسئله اصلی رگرسیون عبارت است از تعیین میانگین شرطی مقدار Y به ازای مقادیر مفروضی از X . رابطه (۶) نشان دهنده معادله رگرسیون خطی می باشد.

$$\mu_{Y|X} = \alpha + \beta x \quad (6)$$

با توجه به اینکه برخلاف بودن رگرسیون تأکید کردیم، هدف از حل مسئله ضرایب α ، β از روی داده‌های نمونه می‌باشد، به طوری که خط $y = \alpha + \beta x$ بهترین برازش ممکن برای داده‌های مفروض شود. یکی از روش‌هایی که معمولاً برای تعیین ضرایب α ، β مورد استفاده قرار می‌گیرد روش کمترین مربعات خطا (Least Square Error) است که در اوایل قرن نوزدهم توسط ریاضیدان فرانسوی آدرین لژاندر (Adrien Legendre) پیشنهاد شده است [۲۳]. مسائل متعددی وجود دارند که در آنها می‌توان متغیری را بر حسب متغیر دیگر به صورت دقیق پیش گویی کرد، اما با در نظر گرفتن اطلاعات بیشتر می‌توان پیشگویی‌های بهتری انجام داد. بدین ترتیب رگرسیون چند متغیره تکامل یافته رگرسیون تک متغیره می باشد. گرچه روابط متعددی موجودند که می‌توان از آنها برای بیان روابط رگرسیونی بین بیش از دو متغیره، استفاده کرد [۲۴]، البته استفاده از رگرسیون چند متغیره در علوم متفاوت همچون مراجع [۲۵-۳۴] رایج بوده است، که کاربردهای متعددی از قبیل تخمین، پیش بینی و برازش منحنی دارد. رایجترین آنها معادلات خطی، به صورت رابطه (۷) می‌باشند.

$$\mu_{Y|X_1, X_2, X_3, \dots, X_k} = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \dots + \beta_k x_k \quad (7)$$

به دلایل متعدد معادلات رگرسیون خطی مورد توجه خاصی هستند، این معادلات به سادگی در سایر اعمال ریاضی وارد می‌شوند و اغلب، تقریب خوبی برای معادلات پیچیده هستند. در رابطه (۷)، Y متغیری تصادفی است که می‌خواهیم مقادیر آن را برحسب مقدارهای معلوم x_1, x_2, \dots, x_k پیشگویی کنیم، و $\beta_0, \beta_1, \dots, \beta_k$ ضرایب رگرسیون چندگانه، اعداد ثابتی هستند که باید از روی داده‌های مشاهده شده Y ، تعیین شوند. ضرایب رگرسیون چندگانه به کمک روش

کمترین مربعات خطا برآورد می‌شود. اگر n نقطه داده به صورت رابطه (۸) داشته باشیم.

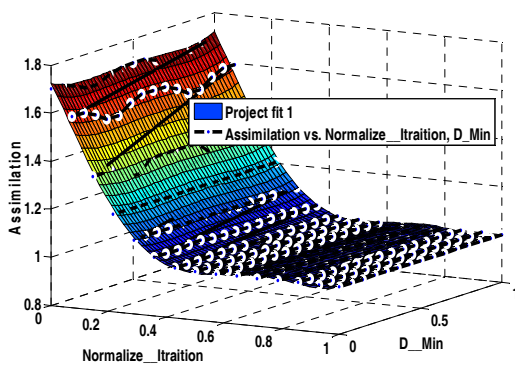
$$\{(x_{i1}, x_{i2}, \dots, x_{ik}, y_i) ; i = 1, 2, \dots, n\} \quad (8)$$

در این صورت هدف از بکارگیری روش کمترین مربعات خطا، یافتن مقادیری از $\beta_0 + \beta_1 x_1 + \beta_2 x_2 + \dots + \beta_k x_k$ می‌باشد که به ازای آنها کمیت q در رابطه (۹) مینیمم شود. در این معادله و معادلات بعدی، x_{i1} ، x_{i2} ، \dots ، x_{ik} ، y_i ، $i = 1, 2, \dots, n$ مقادیر متغیر x می‌باشد.

$$q = \sum_{i=1}^n [\beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \dots + \beta_k x_{ik}]^2 \quad (9)$$

برای مینیمم کردن q ، نسبت به β ها، مشتق جزئی می‌گیریم و با برابر صفر قرار دادن این مشتق‌های جزئی و ساده سازی آن‌ها، مجموعه روابط (۱۰) را بدست می‌آوریم:

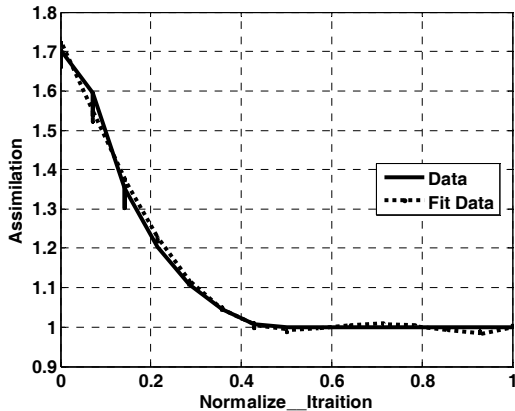
$$\begin{aligned} \sum_{i=1}^n y_i &= \beta_0 n + \beta_1 \sum_{i=1}^n x_{i1} + \beta_2 \sum_{i=1}^n x_{i2} + \dots + \beta_k \sum_{i=1}^n x_{ik} \\ \sum_{i=1}^n x_{i1} y_i &= \beta_0 \sum_{i=1}^n x_{i1} + \beta_1 \sum_{i=1}^n x_{i1}^2 + \beta_2 \sum_{i=1}^n x_{i1} x_{i2} + \dots + \beta_k \sum_{i=1}^n x_{i1} x_{ik} \\ \sum_{i=1}^n x_{ik} y_i &= \beta_0 \sum_{i=1}^n x_{ik} + \beta_1 \sum_{i=1}^n x_{ik} x_{i1} + \beta_2 \sum_{i=1}^n x_{ik} x_{i2} + \dots + \beta_k \sum_{i=1}^n x_{ik}^2 \end{aligned} \quad (10)$$



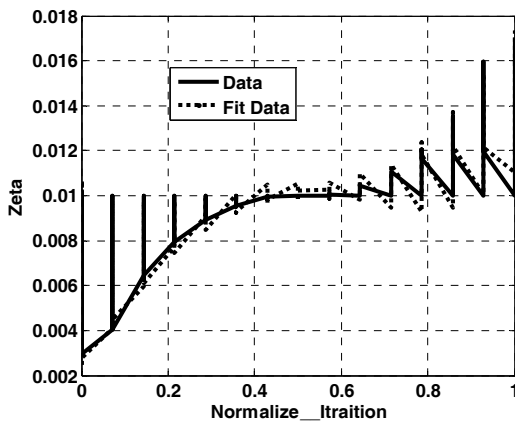
شکل ۱۵- خروجی سطح فیزیکی توسط رگرسیون چند متغیره برای خروجی اول بر اساس شکل (۷)، رابطه ۱۰ و سیستم فازی مورد نظر

بدین ترتیب از حل $k+1$ معادله فوق می‌توان متغیرهای مجهول $\beta_0, \beta_1, \dots, \beta_k$ و با قرار دادن آن‌ها در معادله (۷) تقریبی خطی برای داده‌های مورد نظر بدست آورد. اکنون با توجه به مباحث مطرح شده، می‌توان رگرسیون چندگانه را برای داده‌های خروجی کنترل کننده فازی مورد بحث که نمودارهای ورودی - خروجی آن در شکل (۵) و (۶) رسم شده بود، انجام داد. برای افزایش دقت در تقریب داده در تقریب داده‌های حاصل از خروجی سیستم فازی مورد بحث، آن را مطابق شکل (۱۲) و (۱۳) به نواحی متعدد تقسیم می‌کنیم.

مقایسه نماییم، به همین منظور به بررسی قطعات جانبی صفحات سیستم فازی و صفحه طراحی بدست آمده توسط رگرسیون چند متغیره می‌پردازیم. در شکل (۱۴) رابطه بین ورودی اول و یکی از خروجی‌های (هزینه مستعمرات) سیستم اصلی فازی مشخص شده است. در شکل (۱۵) رابطه بین همان متغیر ورودی از سیستم فازی و خروجی دیگر (ضریب جذب) مشخص شده است. با کمی دقت مشخص می‌شود که مقادیر مشخص شده و بیان شده توسط روش رگرسیون چند متغیره با مقادیر سیستم فازی در هر ۲ خروجی تقریباً برابر با مقادیر اصلی می‌باشند. نکات بیان شده در هر ۲ شکل (۱۷) و (۱۸) مشخص شده‌اند. در آخر پس از بحث و ارائه سیستم فازی پیشنهادی، ارزیابی پیشنهادی را ارائه می‌نماییم.



شکل ۱۷- نمایش مقادیر مشخص شده ورودی اول و خروجی اول توسط رگرسیون چند متغیره بر اساس مقادیر واقعی در سیستم فازی

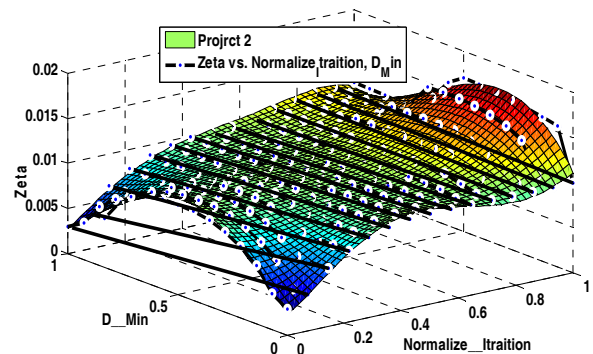


شکل ۱۸- نمایش مقادیر مشخص شده ورودی اول و خروجی دوم توسط رگرسیون چند متغیره بر اساس مقادیر واقعی در سیستم فازی

۷- ارزیابی الگوریتم

در این قسمت نیز برای شبیه‌سازی از نرم‌افزار MATLAB® استفاده شد. ابتدا سیستم فازی بر اساس مطالعه روی مقاله اصلی الگوریتم رقابت استعماری در [۴] طراحی شد، سپس با استفاده از دانش فرد خیره، تسلط بر الگوریتم استعماری و منطق فازی، سیستمی فازی به منظور بهبود عملکرد الگوریتم رقابت استعماری طراحی شد، سپس سیستم فازی طراحی شده پیاده‌سازی شد. در بخش (۵) سیستم فازی مورد ارزیابی قرار گرفت تا صحت سیستم فازی طراحی شده مورد بررسی قرار گیرد. پس از اثبات عملکرد سیستم فازی، از جعبه ابزار sftool در نرم افزار MATLAB® استفاده شد؛ تا اینکه توسط رابطه‌ای غیرخطی، عملکرد

تا برآوردی دقیق از شکل‌های (۷ و ۸) که معرفی کننده سیستم فازی هستند، بدست آید. در روابط (۱۱) توابع بدست آمده با استفاده از رگرسیون دوگانه و نواحی آنها آمده است. در شکل (۱۶) خروجی رگرسیون چند متغیره برای خروجی دوم مشخص شده است.



شکل ۱۶- خروجی سطح فیزیکی توسط رگرسیون چند متغیره برای خروجی دوم، بر اساس شکل (۸)، رابطه ۱۰ و سیستم فازی مورد نظر

برای اعمال رگرسیون خطی از جعبه ابزار sftool® در نرم‌افزار MATLAB® استفاده شده است. با اعمال رگرسیون چند متغیره، رابطه (۱۱) استخراج می‌شود، رابطه (۱۱) رابطه‌ای تقریبی یا معادله‌ای غیرخطی از سیستم فازی طراحی شده است.

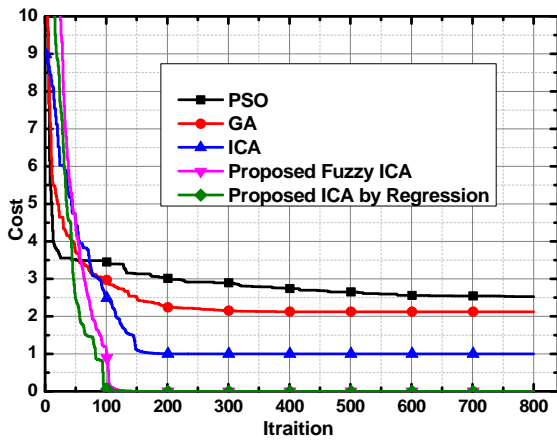
$$F(x, y)_{\text{Assimilation \& Zeta}} = P_{00} + P_{10}x + P_{01}y + P_{20}x^2 + P_{11}xy + P_{02}y^2 + P_{30}x^3 + P_{21}x^2y + P_{12}xy^2 + P_{03}y^3 + P_{40}x^4 + P_{31}x^3y + P_{22}x^2y^2 + P_{13}xy^3 + P_{04}y^4 + P_{50}x^5 + P_{41}x^4y + P_{32}x^3y^2 + P_{23}x^2y^3 + P_{14}xy^4 + P_{05}y^5 \quad (11)$$

مقادیر معرفی شده یا مقادیر عددی ثابت توسط جعبه ابزار در جدول (۶) مشخص شده‌اند.

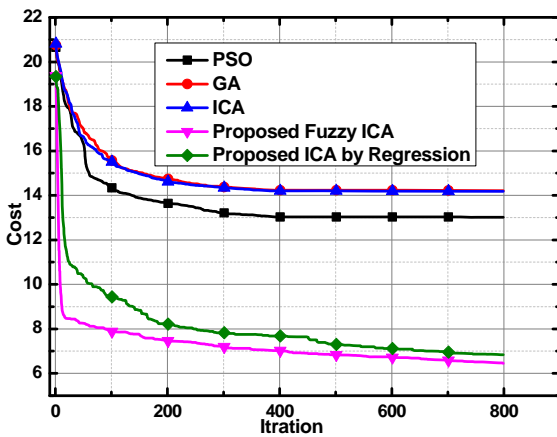
جدول ۶- پارامترهای معرفی شده توسط جعبه ابزار Sftool

Varies	Assimilation Fun	Zeta Fun
P ₀₀	1.725	0.002453
P ₁₀	-2.333	0.01841
P ₀₁	-0.4146	0.02594
P ₂₀	-2.992	0.07839
P ₁₁	0.9056	-0.1748
P ₀₂	1.675	0.01049
P ₃₀	19.12	-0.2922
P ₂₁	-0.6865	0.1788
P ₁₂	-2.674	0.2362
P ₀₃	-2.521	-0.08615
P ₄₀	-24.09	0.3014
P ₃₁	0.328	0.09883
P ₂₂	0.6865	-0.363
P ₁₃	3.536	-0.03292
P ₀₄	1.26	0.0581
P ₅₀	9.57	-0.0974
P ₄₁	2.071e-013	-0.08906
P ₃₂	-0.328	0.05029
P ₂₃	3.435e-014	0.119
P ₁₄	-1.768	-0.02324
P ₀₅	1.157e-013	-0.008066

به منظور بررسی عملکرد جعبه ابزار ناگزیر باید صفحات اصلی (متعلق به سیستم فازی) و صفحات بدست آمده از رابطه رگرسیون چند متغیره، با یکدیگر را



شکل ۲۰- تقابل الگوریتم پیشنهادی با سایر الگوریتم‌های بهینه‌سازی



شکل ۲۱- تقابل الگوریتم بهینه‌سازی پیشنهادی با سایر الگوریتم‌های بهینه‌سازی برای بهینه‌سازی تابع Rstregin ۱۰۰ بعدی

در جدول (۷) به خوبی مشهود است که، الگوریتم پیشنهادی در بهینه‌سازی توابع مذکور برای ۲۰ بعد عملکرد برتری داشته است. در جدول (۸) نیز نتایج تقابل بهینه‌سازی الگوریتم پیشنهادی با سایر الگوریتم‌ها برای بهینه‌سازی توابع [۴] به ازای ۵۰ بعد مشخص شده است.

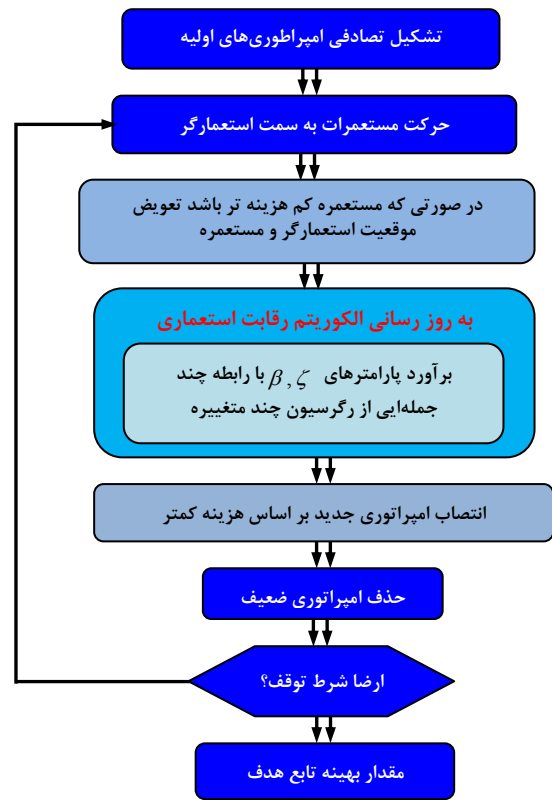
جدول ۸- تقابل الگوریتم‌های بهینه‌سازی به منظور در بهینه‌سازی توابع مذکور برای بهینه‌سازی ۵۰ بعدی توابع [۴]

Fcn by 50 Dimension	F1	F2	F3	F4
GA	5.01315	0.00124	-0.9359	356.296
PSO	6.36284	0.107	-0.9359	454.921
ICA	4.76591	0.00124	0.9359	250.932
Fuzzy ICA	3.87501	8.028E-11	-0.9359	152.456
MRICA	3.98162	5.788E-7	-0.9359	158.173

جدول ۹- تقابل بهینه‌سازی الگوریتم پیشنهادی با الگوریتم‌های [۴] به‌منظور بهینه‌سازی توابع مذکور به ازای ۱۰۰ بعد

Fcn by 100 Dimension	F1	F2	F3	F4
GA	9.15734	49.30344	-0.959	1271.8
PSO	9.54876	1077.2126	-0.959	2752.7
ICA	8.63942	1.19571	-0.939	1271.8
Fuzzy ICA	4.41572	0.00106	-0.959	-0.939
MRICA	5.43394	0.01274	-0.959	-0.939

سیستم فازی توسط چند جمله‌ای (۱۱) بیان گردید که در جدول (۸) مشخص و خلاصه شد. سپس رابطه چند جمله‌ای (۱۱) در الگوریتم اصلی هنگام بروز رسانی الگوریتم جاسازی شد تا مقادیر ضریب جذب و هزینه مستعمرات از این طریق برآورد شدند. الگوریتم نهایی مانند فلوجارت (۴) می‌باشد.



شکل ۱۹- روند اجرایی الگوریتم نهایی معرفی شده

به منظور بررسی و عملکرد الگوریتم پیشنهادی، از توابع محک معرفی شده در جدول (۴) از مرجع [۴] استفاده می‌نماییم. سپس نتایج مشخص شده را با الگوریتم‌های GA, PSO, Fuzzy ICA و الگوریتم پیشنهادی در ابعاد (۱۵ و ۲۰ و ۵۰ و ۱۰۰ و ۲۰۰) بررسی می‌نماییم. در انتها نتایج در اشکال (۱۶ و ۱۷) و جدول (۶ و ۷ و ۸ و ۹) بیان می‌شوند. در شکل (۲۰) نتایج اجرای الگوریتم پیشنهادی در حل تابع Rastregin ۱۵ بعدی، همان تابعی که اولین بار توسط [۴] معرفی شده است، مشخص می‌شود.

در شکل (۲۱) نتایج تقابل الگوریتم پیشنهادی با سایر الگوریتم‌های بهینه‌سازی در بهینه‌سازی تابع Rstregin ۱۰۰ بعدی را نمایش می‌دهد.

به منظور اثبات نتایج الگوریتم پیشنهادی، از نمایش اشکال در بهینه‌سازی توابع [۴]، خود دارای می‌کنیم و نتایج حاصل از میانگین گیری ۱۰۰ مرتبه‌ای را توسط جداول (۶ و ۷ و ۸ و ۹) نمایش می‌دهیم. در جدول (۷) مقادیر بهینه شده سایر الگوریتم‌های بهینه‌سازی بهمراه الگوریتم پیشنهادی برای بهینه‌سازی ۲۰ بعد توابع [۴] نمایش داده شده است.

جدول ۷- تقابل بهینه‌سازی الگوریتم پیشنهادی با سایر الگوریتم در توابع ۲۰ بعدی جدول [۴]

Fcn by 20 Dimension	F1	F2	F3	F4
GA	1.841728	0.0046	-0.9359	73.372
PSO	3.905521	0.84278	-0.9359	33.828
ICA	1.3944E-13	1.127E-22	-0.9359	27.858
Fuzzy ICA	5.4176E-15	2.5E-53	-0.9359	4.9748
MRICA	1.50921E-14	6.074E-29	-0.9359	26.207

توجه به ماهیت تصادفی روش حل تغییر می‌دهد. از طرفی چون چندین پارامتر در تعیین نقطه بهینه سراسری دخیل می‌باشند، منطق فازی یکی از بهترین روش‌ها برای ایجاد تناسب و تعادل در بین پارامترهای الگوریتم استاندارد رقابت استعماری (ICA) برای حل توابع غیرخطی به منظور دستیابی به نقطه بهینه سراسری تابع هدف می‌باشد. اگر چه محققین تمایل زیادی در استفاده از ابزار منطق فازی داشته اما چالش جدی فهم دقیق منطق و پیاده‌سازی قوانین فازی مساله می‌باشد. ترکیب سه ایده بهینه‌سازی رقابت استعماری، منطق فازی و همینطور رگرسیون چند متغیره می‌تواند پاسخی هوشمند برای این چالش محسوب شود. نتایج شبیه سازی نشان می‌دهد که الگوریتم استاندارد استعداد گیر افتادن در بهینه محلی را دارد. استفاده از منطق فازی و به طبع آن رگرسیون چند متغیره به شرط فهم دقیق موضوع و پیاده‌سازی صحیح قوانین فازی می‌تواند صحت پاسخ دقیق مساله را افزایش، و زمان حل را به شدت کاهش دهد. در نهایت تمام تلاش‌ها برای بهبود عملکرد الگوریتم رقابت استعماری بود که البته منجر شد تا الگوریتم استعماری بهبود یافته بر اساس رگرسیون چند متغیره یا Improved ICA by based on Fuzzy Logic Multivariable Regression معرفی شود.

مراجع

[۱] م. شیبانی، مدل‌های ترکیبی بر پایه اتوماتای یادگیر سلولی و محاسبات تکاملی و کاربردهای آنها، پایان نامه کارشناسی‌ارشد مهندسی کامپیوتر، دانشکده مهندسی کامپیوتر، دانشگاه صنعتی امیرکبیر، ۱۳۸۵.

[2] H. Sarimveis, and A. Nikolakopoulos, "A Line Up Evolutionary Algorithm for Solving Nonlinear Constrained Optimization Problems," *Journal of Computers and Operations Research*, vol. 32, no. 6, pp. 1499-1514, 2005.

[3] H. A. Abbass, R. Sarker, and C. Newton, "PDE: A Pareto-frontier Differential Evolution Approach for Multi-objective Optimization Problems," *Proc. IEEE Intl Conf. Genetic Programming*, pp. 503-511, 2001.

[۴] م. عبدچیری، م. صنیعی آبا، و م. بهرامی، "الگوریتم رقابت استعماری با شعاع حرکت تطبیقی"، دانشگاه تربیت مدرس کنفرانس، ۱۳۸۶.

[5] H. Muhlenbein, M. Schomisch, and J. Born, "The Parallel Genetic Algorithm as Function Optimizer," *Proc. IEEE Intl Conf. Genetic Programming*, pp. 270-278, 1991.

[6] M. Melanie, *An Introduction to Genetic Algorithms*, Massachusetts, MIT Press, 1999.

[7] P. J. Angeline, "Evolving Fractal Movies," *Proc. IEEE Intl Conf. Genetic Programming*, pp. 503-511, 1997.

[8] J. Kennedy, and R. C. Eberhart, "Particle Swarm Optimization," *Proc. IEEE Intl Conf. Neural Networks*, pp. 1942-1948, 1995.

[9] J. Kennedy, and R. C. Eberhart, "Particle Swarm Optimization," *Proc. IEEE Intl Conf. Neural Networks*, pp. 1942-1948, 1995.

[10] X. Yang, J. Yuan, J. Yuan, and H. Mao, "A Modified Particle Swarm Optimizer with Dynamic Adaptation,"

در ادامه نیز به بررسی و تقابل الگوریتم‌های بهینه سازی در جدول (۹) برای بهینه‌سازی توابع مذکور به ازای توابع ۱۰۰ و ۲۰۰ بعدی می‌پردازیم.

در جدول (۹) نتایج اجرای الگوریتم پیشنهادی نیز در یک شرایط یکسان (شرط ارضاء) مشخص شده است. نکته‌ی قابل تامل در این بررسی، تفاوت چشم گیر، همگرایی و دقت دیگر الگوریتم‌های بحث شده در این ارزیابی در مقایسه با الگوریتم پیشنهادی است، به عنوان مثال در سطر آخر جدول (۹) و شکل (۲۱) به خوبی مشخص است که الگوریتم پیشنهادی دارای یک سرعت بالاتر در همگرایی و همینطور دقت برتر در کاوش بمنظور بدست آوردن نقطه بهینه را دارد. در جدول (۱۰) نیز تقابل الگوریتم پیشنهادی با سایر الگوریتم‌های بهینه سازی در ۲۰۰ بعد نشان داده شده است.

جدول ۱۰- تقابل بهینه‌سازی الگوریتم پیشنهادی با الگوریتم‌های [۴] به‌منظور بهینه‌سازی توابع مذکور به ازای ۱۰۰ و ۲۰۰ بعد

Fcn by 200 Dimension	F1	F2	F3	F4
GA	14.187	7022.902	-0.939	4480.04
PSO	14.24	195.15	-0.959	8035.5
ICA	13.029	40.171	-0.939	2296.2
Fuzzy ICA	6.43	1.70	-0.959	1269.6
MRICA	6.879	5.56	-0.959	2181.3

در تمامی این ارزیابی‌ها سعی شد تا الگوریتم‌های بررسی شده به درستی پیاده شوند، تا در نتایج کمترین خطا و نهایت دقت بوجود آید. با توجه به نتایج از قبیل شکل‌های (۲۰) و (۲۱) و همینطور جداول (۷) و (۸) و (۹) و (۱۰)، به راحتی میتوان برتری الگوریتم پیشنهادی را نتیجه گرفت.

۸- بحث و نتیجه‌گیری

روند ارائه و تکامل در الگوریتم‌های بهینه‌سازی رو به رشد است، و همواره سعی بر این است تا نتایج این پیشرفت‌ها منجر به افزایش سرعت همگرایی و بهبود عمق کاوش در بهینه‌سازی شود. یکی از این الگوریتم‌ها، الگوریتم رقابت استعماری است، که در نوع خود قابل تحسین است. متأسفانه این الگوریتم نیز مشکل گیر افتادن در بهینه محلی، کندی سرعت در بهینه‌سازی و مهم تر از همه تنظیم صحیح پارامترهایش را در شرایط متفاوت دارد. ایده استفاده از منطق فازی برای رفع این مشکلات تحت عنوان اولین مدل فازی الگوریتم رقابت استعماری (تا زمان نگارش این اثر موردی در مراجع معتبر، نمایه نشده است) در اینجا مطرح شد. با استفاده از نرم افزار MATLAB® الگوریتم فازی پیشنهادی پیاده شد و سپس با توجه به مسائل مشهور محک ارزیابی گسترده انجام شد، سپس با توجه به مشکلاتی که در مورد پیاده‌سازی سیستم‌های فازی شد، از رگرسیون چند متغیره برای پیاده‌سازی و اصطلاحاً خلاصه‌سازی سیستم فازی در یک چند جمله‌ای غیرخطی استفاده شد. در ارزیابی نهایی طبق فلوجارت ۴، نحوه صحیح پیاده‌سازی الگوریتم پیشنهادی ارائه شد، سپس بحث و بررسی پیرامون ارزیابی الگوریتم پیشنهادی در شکل‌ها و جداولی ارائه شدند. با توجه به نتایج بدست آمده می‌توان نتیجه گرفت که الگوریتم پیشنهادی تقریباً در تمامی توابع محک استفاده شده بهترین بوده و برتری محسوسی نسبت به دیگر الگوریتم‌های مذکور دارد. این الگوریتم در تشخیص بهینه‌های محلی موفق بوده و توانسته خود را از دام آنها رها کند و بهینه سراسری را پیدا کند. در بعضی از نمودارها همانند شکل‌های (۲۰) و (۲۱) رفتار نسبتاً غیرخطی و حتی قابل تاملی از توابع دیده می‌شود که این بخاطر مقادیر کاملاً تصادفی و غیرقابل پیش بینی تولیدی توسط الگوریتم در تلاش برای همگرایی است. می‌توان گفت که استفاده از توابع غیرخطی رفتار الگوریتم استاندارد را با

- [26] S. Chehreh Chelgani, J. C. Hower, and B. Hart. "Estimation of Free-swelling Index based on Coal Analysis using Multivariable Regression and Artificial Neural Network," *Journal of Fuel Processing Technology*, vol. 9, no. 2, pp. 349-355, 2011.
- [27] A. R. Jafari, and T. Babadagli, "Estimation of Equivalent Fracture Network Permeability using Fractal and Statistical Network Properties," *Journal of Petroleum Science and Engineering*, vol. 5, no. 2, pp. 110-123, 2012.
- [28] L. dos Santos Coelho, and V. Cocco Mariani, "Firefly Algorithm Approach based on Chaotic Tinkerbell Map Applied to Multivariable PID Controller Tuning," *Journal of Computers and Mathematics with Applications*, vol. 64, no. 2, pp. 2371-2381, 2012.
- [29] N. Azam Baleghy, and S. K. Mousavi Mashhadi, "Design and Implementation Fuzzy Controller in the Frost-free Refrigerator by using Multivariate Regression," *Proc. IEEE Intl Conf. Electrical Engineering*, pp. 156-162, 2012.
- [30] L. Yong-song, Y. Jian-min, C. Jian-ping, and X. Jing, "Analysis of 3D In-situ Stress Field and Query System's Development based on Visual BP Neural Network," *Journal of Procedia Earth and Planetary Science*, vol. 5, no. 2, pp. 64-69, 2012.
- [31] K. Salahshoor, and M. Hamz ehnejad, "A Novel Online Affine Model Identification of Multivariable Processes using Adaptive Neuro-fuzzy Networks," *Journal of Chemical Engineering Research and Design*, vol. 8, no. 8, pp. 155-169, 2010.
- [32] D. Ansari, J. Nilsson, R. Andersson, S. Regner, B. Tingstedt, and B. Andersson, "Artificial Neural Networks Predict Survival from Pancreatic Cancer after Radical Surgery," *American Journal of Surgery*, vol. 20, no. 5, pp. 1-7, 2013.
- [33] Q. Lu, and M. Mahfouf, "Multivariable Self-organizing Fuzzy Logic Control using Dynamic Performance Index and Linguistic Compensators," *Journal of Engineering Applications of Artificial Intelligence*, vol. 25, no. 2, pp. 1537-1547, 2012.
- [34] T. A. Choudhury, N. Hosseinzadeh, and C. C. Berndt, "Artificial Neural Network Application for Predicting in-flight Particle Characteristics of an Atmospheric Plasma Spray Process," *Journal of Surface and Coatings Technology*, vol. 20, no. 5, pp. 4886-4895, 2011.
- [35] E. Kinnear, L. Kenneth, L. Spector, and P. J. Angeline, *Advances in Genetic Programming*, Ph. D. Dissertation, MIT, Cambridge, MA, 1999.
- [36] J. Kennedy, R. Eberhart, "Particle Swarm Optimization," *Proc. IEEE Intl Conf. Neural Networks*, pp. 1942-194, 1995.
- Journal of Applied Mathematics and Computation*, vol. 18, no. 2, pp. 1205-1213, 2007.
- [11] I. Tokuda, K. Aihara, and T. Nagashima, "Adaptive Annealing for Chaotic Optimization," *Journal of Physics*, vol. 58, no. 4, pp. 34-46, 1998.
- [12] A. Ingber, "Simulated Annealing: Practice versus Theory," *Journal of Mathematics, Computer and Modeling*, vol. 18, no. 11, pp. 29-57, 1993.
- [13] M. F. Cardoso, R. L. Salcedo, S. F. Azevedo and D. Barbosa, "A Simulated Annealing Approach to the Solution of Minlp Problems," *Journal of Computer, Chemistry and Engineering*, vol. 21, no. 12, pp. 1349-1364, 1997.
- [14] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi, "Optimization by Simulated Annealing," *Journal of Science*, vol. 22, no. 2, pp. 671-680, 1983.
- [15] B. Franklin, and M. Bergerman, "Cultural Algorithms: Concepts and Experiments," *Proc. IEEE Intl Conf. Evolutionary Computation*, pp. 1245-1251, 2000.
- [16] A. Gargari, and C. Lucas, "Imperialist Competitive Algorithm: An Algorithm for Optimization Inspired by Imperialistic Competition," *Proc. IEEE Intl Conf. Evolutionary Computation*, pp. 4661-4667, 2007.
- [۱۷] م. ح. سیگاری، و ک. لوکس، "کاربرد الگوریتم رقابت استعماری برای انتخاب ویژگی در سیستم تشخیص چهره،" *قطب کنترل و پردازش هوشمند، دانشکده مهندسی برق و کامپیوتر، دانشگاه تهران، ۱۳۸۶.*
- [18] L. A Zadeh, "Fuzzy Sets," *Proc. IEEE Intl Conf. Information and Control*, pp. 338-346, 1965.
- [19] M. Sugeno, *Industrial Applications of Fuzzy Control*, Elsevier Book, New York, 1985.
- [20] J. Liebowitz, *the Handbook of Applied Expert Systems*, Prentice-Hall, 1999.
- [21] Li-Yeh Chuang, Sheng-Wei Tsai, and Cheng-Hong Yang, "Improved Catfish Particle Swarm Optimization with Fuzzy Adaptation," *Proc. IEEE Intl Conf. Electrical Engineering*, pp. 20-24, 2009.
- [22] International Biometrics Group, <http://www.atashpaz.com>, May 2003.
- [۲۳] ف. جان، محاسبات فازی، انتشارات دانش نگار، تهران، ۱۳۸۶.
- [24] P. J. Rousseeuw, A. M. Leroy, P. J. Rousseeuw, and A. M. Leroy, *Fuzzy Logic Analysis*, Wiley and Sons, 1987.
- [25] P. L. P. Wasantha, P. G. Ranjith, and D. R. Viète, "Constitutive Models Describing the Influence of the Geometry of Partially-spanning Joints on Jointed Rock Mass Strength: Regression and Fuzzy Logic Analysis of Experimental Data," *Journal of Expert Systems with Applications*, vol. 39, no. 3, pp. 7663-7672, 2012.

اطلاعات بررسی مقاله:

تاریخ ارسال: ۹۲/۱۲/۲۳

تاریخ اصلاح: ۹۳/۸/۲۵

تاریخ قبول شدن: ۹۳/۹/۳

نویسنده مرتبط: محمد فیوضی، دانشکده مهندسی برق، دانشگاه علم و صنعت ایران، تهران، ایران.



محمد فیوضی در حال حاضر در دانشگاه علم و صنعت ایران، آزمایشگاه رباتیک و سیستم‌های غیر خطی (دکتر موسوی مشهدی)، دانشکده مهندسی برق مشغول به فعالیت می‌باشد. مدرک کارشناسی و کارشناسی ارشد خود را به ترتیب در سال‌های ۱۳۸۶ و ۱۳۹۱ از دانشگاه‌های آزاد

اسلامی واحد ساوه در رشته مهندسی برق، و دانشگاه حکیم سبزواری (تربیت معلم سبزوار) در رشته مهندسی برق بیوالکترونیک اخذ نموده است و در حال حاضر دانشجوی سال دوم مقطع دکتری مهندسی برق دانشگاه علم و صنعت ایران می‌باشند. زمینه‌های حرفه‌ای مورد علاقه ایشان، پردازش سیگنال (تصویرهای دیجیتالی و بیولوژیکی)، بازشناسی الگو، شبکه‌های عصبی مصنوعی و تطبیقی فازی عصبی، بهینه‌سازی، منطق فازی، محاسبات نرم، هوش مصنوعی کنترل مدل پیش بین و سیستم‌های غیر خطی می‌باشد. از ایشان تا به حال تعدادی مقاله در مجلات داخلی، خارجی و کنفرانس‌های مرتبط به چاپ رسیده است.

آدرس پست الکترونیکی ایشان عبارت است از:

mohammad_fiuzy@elec.iust.ac.ir



سید کمال‌الدین موسوی مشهدی در سال ۱۳۴۸ در مشهد متولد شده‌اند. ایشان مدارک کارشناسی ارشد و دکتری خویش را از دانشگاه فردوسی مشهد به ترتیب در سال‌های ۱۳۷۸ و ۱۳۸۵ اخذ نموده‌اند. ایشان هم‌اکنون در دانشگاه علم و صنعت ایران در تهران با سمت استاد یاری در

دانشکده مهندسی برق مشغول به فعالیت می‌باشند. علایق مطالعاتی ایشان، کنترل خطی و غیر خطی، کنترل فازی، مدل سازی غیر خطی و کاربردهایش، آشوب، کنترل تطبیقی، کنترل فرآیندهای اتفاقی، کنترل چندمتغیره و محاسبات نرم می‌باشد.

آدرس پست الکترونیکی ایشان عبارت است از:

sk_mousavi@iust.ac.ir



جواد حدادنیا در حال حاضر دانشیار دانشکده مهندسی برق و کامپیوتر دانشگاه حکیم سبزواری (تربیت معلم سبزوار سابق) می‌باشد. او مدرک کارشناسی را در رشته مهندسی برق - الکترونیک در سال ۱۳۷۲ و مدارک کارشناسی ارشد و دکتری را از دانشکده مهندسی برق و

کامپیوتر دانشگاه صنعتی امیرکبیر تهران به ترتیب در سال‌های ۱۳۷۴ و ۱۳۸۱ اخذ نموده است. ایشان از سال ۱۳۸۱ با مرتبه استادیاری در دانشگاه حکیم سبزواری (تربیت معلم سبزواری سابق) مشغول به فعالیت می‌باشند. او در سال‌های ۱۳۸۰-۱۳۸۱ در دانشگاه Windsor در کشور کانادا به عنوان محقق مشغول به فعالیت بوده‌اند. علایق مطالعاتی ایشان شامل؛ شبکه‌های عصبی مصنوعی، پردازش تصاویر دیجیتالی، بینایی ماشین و انسان، بازشناسی الگو، هوش مصنوعی و محاسبات نرم می‌باشد. از ایشان تا به حال شمار زیادی مقالات در ژورنال‌های بین‌المللی به چاپ رسیده است، ضمن اینکه هم‌اکنون سردبیر مجله هوش مصنوعی در دانشگاه مربوطه می‌باشند، ایشان هم‌اکنون عضو دائم انجمن‌های بین‌المللی IEEE, SPIE, CIPPR, and IEICE می‌باشند.

آدرس پست الکترونیکی ایشان عبارت است از:

haddadnia@hsu.ac.ir

¹ Imperialist Competitive Algorithm

² Assimilation

³ Genetic Algorithm

⁴ Particle Swarm Optimization

⁵ Evolutionary Algorithm

⁶ John Holland

⁷ Russell Eberhart

⁸ John Kennedy

⁹ Simulated Annealing

¹⁰ Esmail Atashpaz-e-Gargari

¹¹ Caro Lucas

¹² Lotfi-e-Askar Zadeh