

## ارائه مدل جدیدی از الگوریتم فازی حرکت دسته جمعی ذرات با افزایش دقت و صحت همگرایی در بهینه‌سازی توابع (FPSO)

محمد فیوضی<sup>۱</sup> جواد حدادنیا<sup>۱</sup> سیدکمال‌الدین موسوی مشهدی<sup>۲</sup> محمد علی‌پور<sup>۳</sup>

<sup>۱</sup> دانشکده مهندسی برق و کامپیوتر، دانشگاه حکیم سبزواری، سبزوار، ایران  
<sup>۲</sup> دانشکده مهندسی برق، دانشگاه علم و صنعت ایران، تهران، ایران  
<sup>۳</sup> دانشکده مهندسی پزشکی، دانشگاه صنعتی چالمرز، چالمرز، سوئد

### چکیده

الگوریتم حرکت دسته جمعی ذرات (PSO<sup>۱</sup>) با تنظیم مسیر حرکت یک جمعیت از ذرات در فضای مساله بر پایه اطلاعات مربوط به بهترین کارایی قبلی در هر ذره (P\_best) و بهترین کارایی قبلی مربوط به همسایگان هر ذره (g\_best, N\_best) عمل جستجو را در فضای مساله انجام می‌دهد. در این روش هر یک از ذرات سعی می‌کنند به سمت بهترین تجربه‌های فردی و گروهی که در آن نقاط روی داده است همگرا شوند، بدنبال این فرآیند نقطه بهینه تابع هدف مشخص می‌شود. مشکل اصلی الگوریتم PSO استاندارد افتادن در دام بهینه محلی است. تنظیم و کنترل پارامترهای الگوریتم استاندارد حرکت دسته جمعی ذرات (PSO) نقش تعیین کننده و موثری در همگرایی و رسیدن به پاسخ بهینه در حل مسائل پیچیده با دامنه گسترده دارد. در این مقاله با استفاده از منطق فازی روشی برای کنترل پارامترهای (W, C<sub>1</sub>, C<sub>2</sub>) در الگوریتم PSO ارائه شده است. هر ذره در هر بار اجرای الگوریتم، مقادیر C<sub>1</sub>, C<sub>2</sub>, W را با توجه به ورودی‌های کنترلر (نرمالیزه تکرار فرایند، تفاوت نقاط بهینه سراسری و محلی در هر تکرار و پارامتر NCBPE) با استفاده از یک سیستم فازی تعیین کرده و در بدست آوردن موقعیت بعدی ذرات از آن استفاده می‌کند. روش پیشنهادی در حل چند مساله با الگوریتم‌های PSO, CPSO, CFPSO مقایسه و نتایج آن ارائه می‌شود. نتایج آزمایش‌ها، برتری روش پیشنهادی به روش‌های مذکور را نشان می‌دهد.

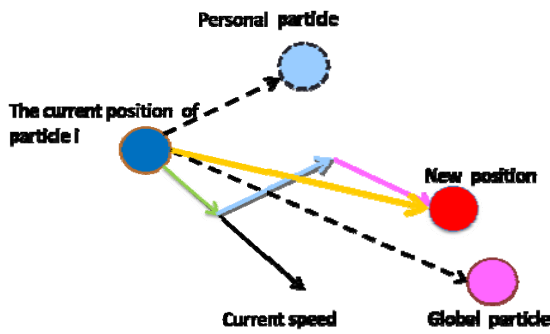
**کلمات کلیدی:** حرکت دسته جمعی ذرات، منطق فازی، بهینه محلی، بهینه سراسری، همگرایی.

### ۱- مقدمه

است [۳]. بهینه‌سازی گروه ذرات یکی از تکنیک‌های بهینه‌سازی ابتکاری می‌باشد که بر مبنای جمعیت کار می‌کند. ایده اصلی این روش در سال ۱۹۹۵ توسط دکتر ابرهات و دکتر کندی [۴] مطرح گردید که از رفتار دسته جمعی ماهی و پرندگان برای یافتن غذا الهام گرفته شده بود. فرض بر این است که یک گروه از پرندگان بصورت تصادفی در یک منطقه به دنبال غذا می‌گردند در حالیکه تنها در یک ناحیه جستجو غذا وجود دارد [۵]. پرندگان از مکان غذا اطلاعی ندارند و تنها میزان فاصله خود تا آن محل را می‌دانند [۶]. استراتژی بکار رفته این است که پرندگان بدنبال پرندایی حرکت می‌کنند که نزدیکترین فاصله تا غذا را دارد [۷]. در PSO استاندارد هر جواب مساله یک ذره نام دارد [۸]. هر ذره دارای یک مقدار

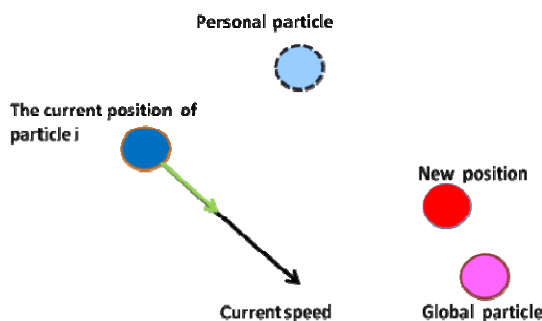
امروزه با بزرگ شدن مسائل و مشخص شدن اهمیت سرعت بالا برای رسیدن به پاسخ و همگرایی، استفاده از الگوریتم‌های جستجوی تصادفی به جای جستجوی همه جانبه در فضای مساله و یا الگوریتم‌های جستجوی هیروستیک<sup>۲</sup> (شهودی)، رشد چشمگیری داشته است [۱]. در مسائل بهینه سازی، یکی از اهداف مهم علاوه بر صحت و دقت نتایج، کاهش زمان جستجو، مهم می‌باشد [۲]. امروزه با بزرگ و پیچیده شدن مسائل، استقبال از روش‌های ابتکاری بطور چشم گیری افزایش یافته

در معادله ۱،  $W$  ضریب اینرسی است، برای اولین بار در [۲۳، ۲۴] پیشنهاد شد تا در طول برنامه بطور خطی بین ۰.۹ تا ۰.۴ کاهش یابد.  $C_1$  و  $C_2$ ، میزان تجربه شخصی و گروهی در بازه [۰-۱.۵] انتخاب می‌شود [۳، ۱۷، ۲۴]. دو عدد  $r_1$  و  $r_2$  نیز اعداد تصادفی در بازه [۰-۱] می‌باشد. همچنین مقدار نهایی سرعت هر پرنده، برای جلوگیری از واگرایی الگوریتم به یک بازه محدود می‌شود و  $V$  عضوی از دامنه  $[-V_{min}, V_{max}]$  می‌باشد. معادله ۲ نیز بردار موقعیت فعلی ذره را با توجه به سرعت جدید آن بروز رسانی می‌کند، نحوه بروز رسانی یک ذره در شکل ۱ نشان داده شده است.



شکل ۱- روند بهینه‌سازی توسط معادلات (۱) و (۲)

حرکت پرندگان بدون قسمت اول معادله ۱، فرایندی خواهد بود که طی آن فضای جستجو به تدریج کوچک می‌شود، اما در مقابل، قسمت اول معادله ۱ باعث حرکت ذرات در مسیر عادی خود خواهد شد. با ترکیب این دو عامل سعی می‌شود تا موازنه‌ای بین جستجوی محلی و سراسری برقرار گردد. برای برقراری بهتر این موازنه برای اولین بار در [۲۳، ۱۷]  $W$  پیشنهاد گردید که ضریب حرکت در جستجوی سراسری را مشخص می‌کند، دو پارامتر  $C_1, C_2$  نیز ضریب حرکت در جستجوی محلی را مشخص می‌کنند.  $W$  رفته رفته کاهش می‌یابد ولی  $C_1, C_2$  همواره ثابت هستند، بنابراین با ادامه جستجو از میزان جستجوی سراسری کم شده و جستجوی محلی افزایش می‌یابد تا بهینه‌ترین جواب بدست آید. یکی از مشکلات شناخته شده الگوریتم PSO این است که برای پرنده‌ای که شامل اطلاعات  $g\_best$  است قسمت‌های دوم و سوم معادله ۱، صفر می‌شوند [۳]. بنابراین پرنده در راستای بردار حرکت قبلی خود حرکت خواهد کرد شکل ۲، ذره شامل  $g\_best$  ثابت خواهد شد و سایر پرندگان نیز به این پرنده همگرا می‌شوند.



شکل ۲- روند بهینه‌سازی الگوریتم با در نظر گرفتن مشکل اول

برای رفع این مشکل ابتدا در سال ۲۰۰۲، الگوریتم جدیدی با نام GCP SO ارائه گردید [۲۵]، در این روش یک پارامتر جدید به الگوریتم اضافه شده بود که

شایستگی است که توسط تابع شایستگی مساله بدست می‌آید [۹]. پرنده‌ای که به غذا نزدیکتر است شایستگی بیشتری دارد [۱۰]. در این زمینه، با توجه به اهمیت مساله، تحقیقات گسترده‌ای انجام شده است. در [۱۱] برای نرم کردن منحنی خروجی الگوریتم، تنها از گزاره‌های منطق فازی به صورت شرطی در بدنه الگوریتم استفاده شده است. توسط محققین [۱۲، ۱۳] از منطق فازی برای بهبود کارایی این الگوریتم در چند تابع محک استفاده شده است. در [۱۴] مدل جدیدی به نام CFPSO<sup>3</sup> توسط مولفین ارائه شد که تنها معادله سرعت، توسط توابع عضویت فازی کنترل می‌شود. در [۱۵] از یک الگوریتم پیشنهادی توسط الگوریتم‌های خوشه بندی استفاده شده است، بطوریکه یک تعریف جدید برای همسایگی پیشنهاد می‌شود که در همگرایی نهایی الگوریتم نقش موثری دارد [۱۶]. می‌توان گفت که از جمله مشکلات مهم این الگوریتم یکی افتادن در دام بهینه محلی و دیگری پایین بودن سرعت همگرایی است. در [۱۷] برای اجتناب از افتادن در دام بهینه محلی در حالتی که سرعت پرنده از یک حد مشخص کمتر و شایستگی بدست آمده قابل قبول نباشد، از یک تابع برای ایجاد شک در سرعت پرنده استفاده می‌شود.

در [۱۸] از یک تابع غیر خطی برای تغییر ضریب اینرسی استفاده می‌شود، در صورتی که میزان شایستگی بدست آمده قابل قبول نباشد این تابع، موجب افزایش ضریب اینرسی شده و با افزایش سرعت ذره‌ها امکان جستجوی سراسری بیشتر می‌شود. در [۱۹، ۲۰] سیستم‌های معرفی شده فازی ورودی‌هایی مانند (دفعات تکرار و اختلاف هر ذره از  $g\_best$ ) و خروجی (ضریب اینرسی<sup>۴</sup>، شتاب یا دخالت جمعی<sup>۵</sup> و یا فردی<sup>۶</sup>) یا ترکیبی از این پارامترها بوده‌اند. در روش پیشنهادی سیستم فازی دارای ۴ ورودی (نرمالیزه تکرار فرایند، تفاوت نقاط بهینه سراسری و محلی در هر تکرار و پارامتر NCBPE) و ۳ خروجی (ضریب اینرسی، دخالت (شتاب) جمعی و فردی) می‌باشد. در بخش دوم این تحقیق الگوریتم PSO استاندارد معرفی می‌شود، سپس در بخش سوم مقدمه‌ای از منطق فازی و کاربردهای آن در زمینه‌های مختلف ارائه می‌شود. در بخش چهارم الگوریتم پیشنهادی ارائه شده تا نهایتاً در بخش پنجم ارزیابی‌های مختلف بر روی الگوریتم پیشنهادی ارائه شود.

## ۲- الگوریتم PSO استاندارد

PSO استاندارد از جمله الگوریتم‌های جستجوی موازی مبتنی بر جمعیت است که با یک گروه از جواب‌های تصادفی (پرندگان) شروع به کار می‌کند [۲۱]. سپس برای یافتن جواب بهینه در فضای مسئله با به روز کردن مکان پرنده‌ها به جستجو ادامه می‌دهد. هر پرنده به صورت چند بعدی (بسته به نوع مساله) با دو بردار  $V_i[t]$  و  $X_i[t]$  که به ترتیب معرف موقعیت فعلی و سرعت در لحظه فعلی ( $t$ ام) از پرنده نام هستند مشخص می‌شود.

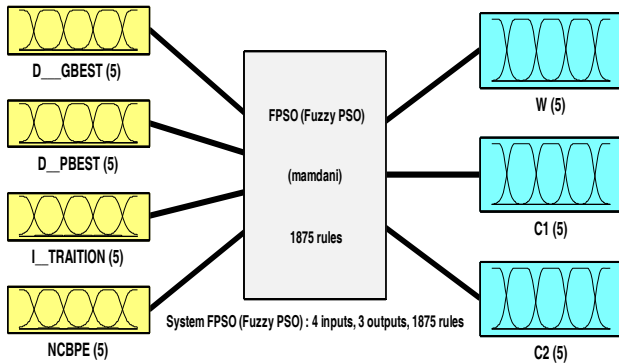
در هر مرحله از حرکت جمعیت، مکان هر پرنده با دو مقدار بهترین به روز می‌شود. اولین مقدار، بهترین تجربه‌ای است که خود پرنده تا به حال بدست آورده است و با  $p\_best$  نشان داده می‌شود دومین مقدار، بهترین تجربه‌ای است که در بین تمامی پرنده‌ها بدست آمده است و با  $g\_best$  نشان داده می‌شود. در هر تکرار، الگوریتم بعد از یافتن دو مقدار بالا، سرعت و موقعیت جدید پرنده را براساس معادلات ۱ و ۲ بروز رسانی می‌کند [۲۲].

$$V_i[t+1] = W V_i[t] + C_1 r_1 (x_i^{pbest}[t] - x_i[t]) + C_2 r_2 (x_i^{gbest}[t] - x_i[t]) \quad (1)$$

$$X_i[t+1] = X_i[t] + V_i[t+1] \quad (2)$$

#### ۴-۱- مشکلات الگوریتم

با توجه به مطالب ارائه شده در بخش قبل و آشنایی با الگوریتم پرندگان می‌توان گفت که دو مشکل مهم روش استاندارد یکی گیر افتادن ذرات در دام بهینه محلی و دیگری پایین بودن سرعت همگرایی است. در اینجا روشی را ارائه می‌کنیم که در واقع یک تابع غیرخطی برای ضریب اینرسی، ضریب تجربه فردی و ضریب تجربه گروهی توسط سیستم فازی ایجاد و تنظیم شود. بطور کلی می‌توان سیستم فازی طراحی شده را توسط شکل ۴ نشان داد.



شکل ۴- سیستم فازی طراحی شده

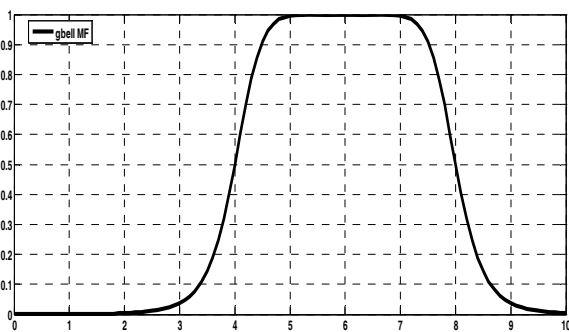
#### ۴-۲- متغیرهای ورودی

متغیرهای ورودی شامل، نرمال شده دفعات تکرار و اختلاف با بهترین بهینه سراسری و گروهی و همینطور پارامتر NCBPE در هر تکرار می‌باشند. برای فازی سازی هر کدام از متغیرهای فازی از توابع زنگوله‌ای بخاطر حساسیت کمتر به نویز طبق معادله ۳ و شکل ۵ استفاده می‌شود.

(۳)

$$f(x; a, b, c) = \frac{1}{1 + \left| \frac{x-c}{a} \right|^{2b}}$$

در این معادله a نقطه شروع، b نقطه اوج زنگوله و c نقطه مرکز زنگوله می‌باشد.



شکل ۵- تابع عضویت زنگوله‌ای

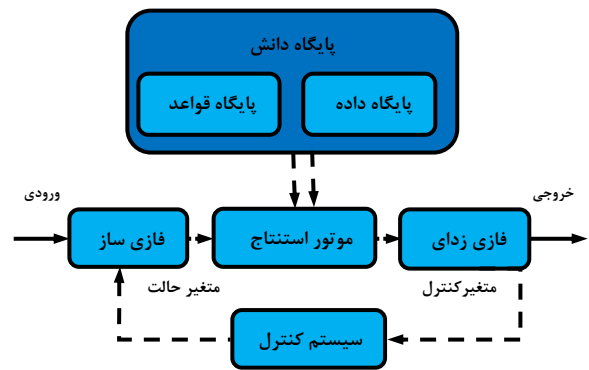
ورودی اول توسط کاربر تعیین می‌شود، بهمین خاطر از نرمال شده‌ی آن بصورت معادله ۴ استفاده می‌کنیم که حاصل عددی بین [0-1] می‌باشد.

باعث جستجوهای تصادفی پرنده best حول جواب بهینه خواهد شد. دومین مشکلی که الگوریتم PSO استاندارد دارد، همگرایی خطی آن می‌باشد که ممکن است بسیار زمان بر باشد [۲۶]. امروزه این الگوریتم در بسیاری از کاربردها از قبیل انتخاب ویژگی [۲۷]، تعیین وزن شبکه‌های عصبی [۲۹]، جایابی قطعات و مسیر یابی ادوات FPGA [۳۰]، طراحی کنترل کننده PID [۳۱]، جایابی ادوات SVC و TCSC در شبکه‌های قدرت استفاده می‌شود [۳۲، ۳۳]. در اینجا از روش ترکیبی الگوریتم PSO استاندارد و منطق فازی برای حل مشکلات مذکور استفاده می‌کنیم.

#### ۳- سیستم‌های فازی

منطق فازی تکنولوژی نسبتاً جدیدی است که در مقابل روش‌های مرسوم برای طراحی و مدل سازی سیستم‌هایی که نیازمند ریاضیات و احتمالات نسبتاً پیشرفته هستند به کار می‌رود. 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 نقاط فضا به درجه عضویت (عددی بین ۰ یا ۱) می‌باشد [۳۴، ۳۵].

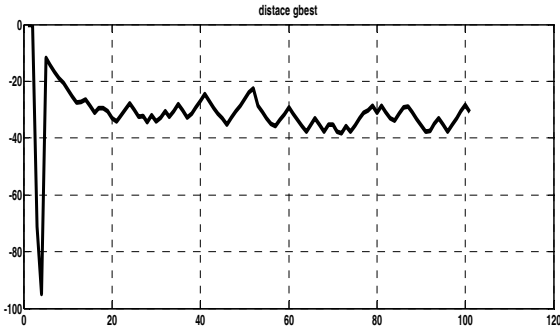
هر سیستم فازی تعدادی قانون دارد که بصورت عبارات IF\_THEN در بانک قواعد آن ذخیره می‌گردد. مرحله پردازش که به آن موتور استنتاج گفته می‌شود براساس بانک قواعد فازی عمل می‌کند. ۲ نوع سیستم استنتاج فازی ممدانی و سوگنو وجود دارد. این ۲ نوع سیستم تا حدودی در روش تعیین فازی سازی ورودی‌ها و عملگرهای فازی شبیه هم هستند. اما در روش تعیین خروجی با هم تفاوت دارند [۳۶]. ما در طراحی سیستم مورد نظر، بر حسب نتایج مورد انتظار از سیستم استنتاج ممدانی استفاده کرده‌ایم. تمامی مراحل سیستم فازی در شکل ۳ مشخص شده است.



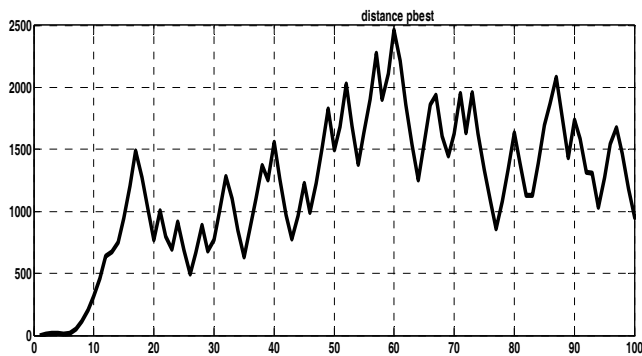
شکل ۳- ساختار یک کنترل کننده فازی

#### ۴- روش ترکیبی فازی یا (Fuzzy PSO) FPSO

در قسمت ۴-۱ ابتدا مساله توضیح داده می‌شود. در قسمت ۴-۲ متغیرهای ورودی‌ها و خروجی‌ها به همراه توابع عضویت شان تعریف می‌شوند و در قسمت ۴-۳ قوانین فازی تعریف و در انتها براساس توابع عضویت، قوانین فازی و موتور استنتاج، خروجی را استخراج کرده و بر روی surface بحث می‌کنیم.



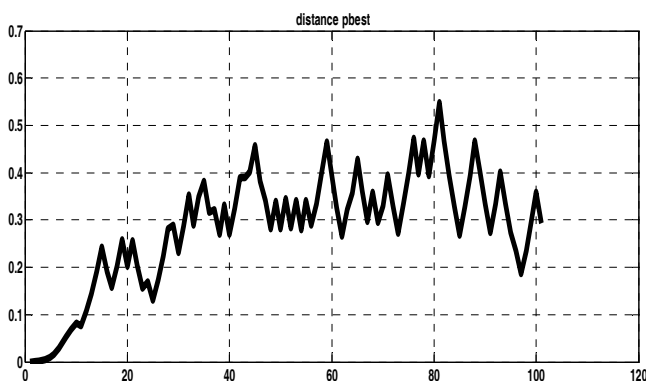
شکل ۷- رفتار تابع نرمال کننده D\_g best در هر تکرار در مسیر همگرایی



شکل ۸- رفتار تابع نرمال کننده D\_p best در هر تکرار

در معادله ۷  $NCBPE$  مقدار نرمال "۳" مقدار بهترین، بدترین و حالت فعلی شایستگی تابع می‌باشد که این ورودی نیز به ۵ مجموعه فازی به نام‌های  $N\_FAR$ ,  $N\_NEAR$ ,  $PROPER$ ,  $P\_NEAR$ ,  $P\_FAR$  در بازه  $[0, 1]$  در شکل ۹ مقدار به خود می‌گیرد [۲، ۳۸].

$$NCBPE = \frac{CBPE - CBPE_{min}}{CBPE_{max} - CBPE_{min}} \quad (7)$$

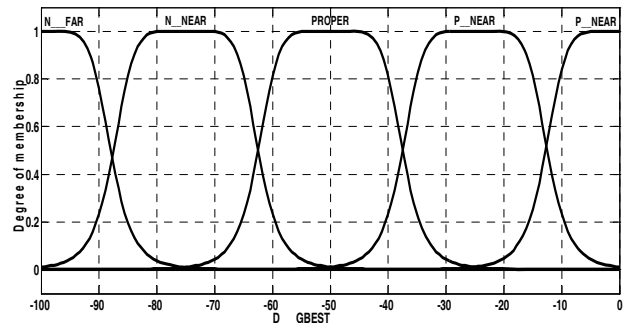


شکل ۹- رفتار تابع  $NCBPE$  در هر تکرار در مسیر همگرایی

در الگوریتم بهینه‌سازی ذرات در حالت عادی به خاطر ماهیت الگوریتم سعی بر پیدا کردن نقطه بهینه در صورت پیدا شدن نقاط بهینه سراسری را دارد.

$$\frac{iter(i)}{maxiter} \quad (4)$$

این ورودی به ۵ مجموعه فازی به نام‌های  $FIRST$ ,  $FIRST\_MED$ ,  $MED$ ,  $END\_MED$ ,  $END$  به ترتیب به معنی ابتدا، میانه ابتدا، میانه، میانه تا انتها و انتهای تکرار در شکل ۶ تقسیم می‌گردد. البته ناگفته نماند از آنجا که در این سیستم تمام توابع عضویت یک نوع (زنگوله‌ایی) و متغیرهای ورودی و خروجی همه دارای ۵ تابع عضویت هستند از ترسیم این مجموعه‌ها خود داری می‌نماییم. ورودی دوم اختلاف بهینه سراسری ( $D\_gbest$ ) است. در الگوریتم بهینه‌سازی ذرات در حالت عادی به خاطر ماهیت الگوریتم سعی بر پیدا کردن نقطه بهینه صورت پیدا شدن نقاط بهینه سراسری را دارد.



شکل ۶- تابع عضویت ورودی بعنوان نمونه

اختلاف بهینه سراسری بدست آمده در هر تکرار با تکرار قبلی به سمت صفر میل می‌کند، تا جایی که اختلاف بهینه سراسری هر مرحله با مرحله قبل صفر شود که در این صورت نقطه بدست آمده حتماً بهینه سراسری است. توسط معادله ۵.

$$D\_gbest = \frac{1}{N} \sum_{i=1}^N (X_i - X_{gbest}) \quad (5)$$

مشخص شد که نرمال شده این ورودی در بازه  $[-100, 0]$  می‌تواند باشد، که دارای ۵ مجموعه فازی با نام‌های  $N\_FAR$ ,  $N\_NEAR$ ,  $PROPER$ ,  $P\_NEAR$ ,  $P\_FAR$  در شکل ۷ به ترتیب به معنی منفی دور، منفی نزدیک، مناسب، مثبت نزدیک و مثبت دور توسط توابع عضویت زنگوله‌ایی تقسیم می‌گردد. ورودی سوم اختلاف بهینه محلی ( $D\_pbest$ ) است. این ورودی نیز به ۵ مجموعه فازی به نام‌های  $N\_FAR$ ,  $N\_NEAR$ ,  $PROPER$ ,  $P\_NEAR$ ,  $P\_FAR$  در بازه  $[0, +100]$  توسط معادله ۶ (شکل ۸ بیانگر رفتار نرمال شده اختلاف بهینه محلی در هر تکرار می‌باشد) توسط توابع عضویت زنگوله‌ایی به ۵ قسمت تقسیم می‌شود.

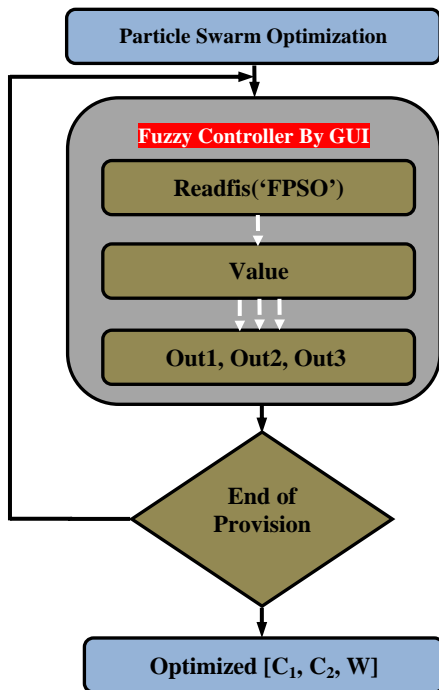
$$D\_pbest = \frac{1}{N} \sum_{i=1}^N (X_i - X_{pbest}) \quad (6)$$

ورودی چهارم پارامتر  $NCBPE$  می‌باشد.  $CBPE$  میزان شایستگی ذره در حالت فعلی را نشان می‌دهد و  $CBPE_{min}$  بهترین شایستگی می‌باشد که ذره تا بحال بدست آورده است و  $CBPE_{max}$  بدترین شایستگی هست که ذرات تا بحال به دست آورده‌اند.

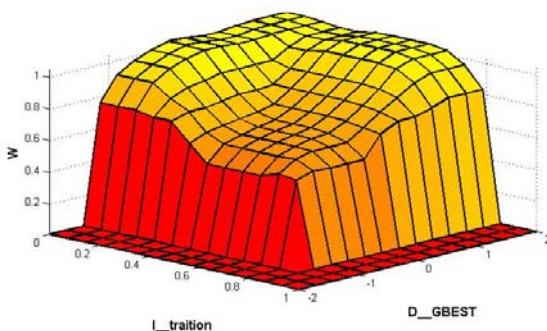
### ۳-۴- متغیرهای خروجی

سیستم فازی طراحی شده، در استلزام از ممدانی، اشتراک از minimum، در اجتماع از maximum و در عملیات استنتاج از اجتماع نتایج حاصل از استلزام هر یک از قوانین (correlation-minimum) استفاده می‌نماید. همچنین به منظور بدست آوردن یک مقدار Crisp (عددی برحسب ورودی) از دی فازی ساز مرکز ثقل استفاده می‌گردد، این مقادیر به صورت خروجی از طریق تابع evalu مجدداً به الگوریتم برنامه پاس داده می‌شود (شکل ۱۰).

پس از اعمال قوانین براساس توابع عضویت‌ها، surface کنترل بدست می‌آید. هر surface نشان دهنده میزان تغییرات یک خروجی براساس دو ورودی است. در این سیستم به ازای هر خروجی یک surface تعریف می‌شود، البته surface ها در این سیستم همانند یکدیگر هستند که برای اجتناب از تکرار فقط یک surface با توجه به شکل‌های بدست آمده، صحت قوانین و توابع عضویت متغیرهای ورودی و خروجی و نیز نرمی و عملکرد آنها در شکل ۱۱ به نمایش درآمده است.



شکل ۱۰- بلوک دیاگرام اجرایی الگوریتم پیشنهادی



شکل ۱۱- نمایشگر سطح ورودی‌ها براساس خروجی اول

### ۵- ارزیابی الگوریتم پیشنهادی

در الگوریتم PSO اساس کار بر روی تغییر وضعیت فعلی پرنده می‌باشد. برای بررسی درست الگوریتم‌ها، در نرم افزار MATLAB<sup>®</sup> الگوریتم‌های پیشنهادی

متغیرهای این سیستم ضریب اینرسی (w)، شتاب یا ضرایب یادگیری ( $C_1, C_2$ ) می‌باشد (شکل ۴). مقادیر بزرگ W امکان جستجوی فرامحلی و مقادیر کوچک آن امکان جستجوی محلی را فراهم می‌کند [۳۷]. در زمینه کنترل ضریب اینرسی با استفاده از سیستم فازی می‌توان به کارهای انجام شده (تا آنجا که مولفین اطلاع دارند) از جمله [۳۸] اشاره کرد که این متغیر در بازه [0-1.2] بصورت استاندارد تعریف می‌شود که به ۵ مجموعه فازی با توابع عضویت زنگوله‌ای به نام‌های  $AF(w), AFM(w), ALM(w), AL(w)$ ، به ترتیب به معنی: کم، میانه‌کم، میانه، میان‌زیاد و زیاد تقسیم می‌شود.  $C_1, C_2$  ضرایبی هستند که به ترتیب بیانگر بیشترین گام حرکت به سمت جواب محلی و جواب سراسری می‌باشد. متغیر دوم خروجی سیستم  $C_1$  یا ضریب شتاب یادگیری (تجربه شخصی) است. این متغیر در بازه [0-1, 1.5] بصورت استاندارد تعریف می‌شود، دارای ۵ تابع عضویت با نام‌های  $AF(c_1), AFM(c_1), M(c_1), ALM(c_1), AL(c_1)$ ، به ترتیب به معنی: کم، میانه‌کم، میانه، میان‌زیاد و زیاد می‌باشد. متغیر سوم خروجی  $C_2$  می‌باشد، این متغیر در بازه [0-1, 1.5] بصورت استاندارد دارای ۵ تابع عضویت گوسی با نام‌های  $F(c_2), AFM(c_2), M(c_2), ALM(c_2), AL(c_2)$ ، به ترتیب به معنی: کم، میانه‌کم، میانه، میان‌زیاد و زیاد می‌باشد. برقراری تعادل بین این ۲ ضریب می‌تواند نقش به‌سزایی در سرعت پاسخ الگوریتم داشته باشد [۳۸]. با استفاده از منطق فازی در این تحقیق ضرایب  $C_1, C_2, W$  در هر تکرار الگوریتم کنترل می‌شود.

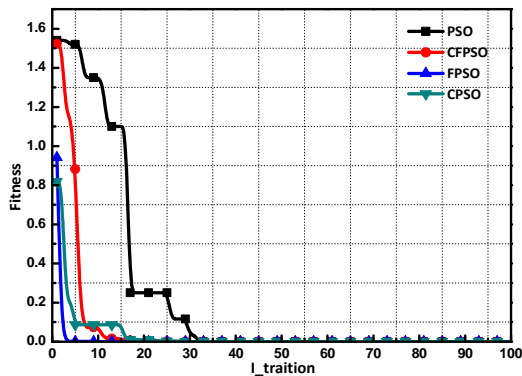
### ۴-۴- قواعد فازی

مهمترین قسمت از یک سیستم فازی، بانک قواعد آن می‌باشد. با داشتن قوانینی درست، می‌توان یک کنترل مناسب را از سیستم انتظار داشت. در این تحقیق تمام حالات ممکن بین ورودی‌ها و خروجی‌ها در بانک قوانین در نظر گرفته شده است. براساس قاعده Decouple ۲۵ مجموعه تحت عنوان بانک قواعد، هر کدام در ۷۵ قانون ایجاد شد، در کل  $1875 (5 \times 5 \times 5 \times 3)$  قانون بدست آمد که در جدول ۱ تنها قواعد بانک اول ارائه شده است. ابتدا مقادیر ورودی توسط الگوریتم حرکت دسته جمعی پرنده‌گان تولید می‌شوند، سپس وارد سیستم فازی می‌شوند.

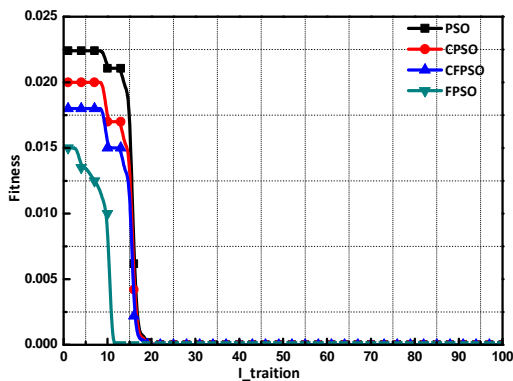
جدول ۱- بانک قواعد با ساختار Index

5 5 1 5, 5 5 1 (1) : 1	4 5 3 5, 3 3 3 (1) : 1
4 5 1 5, 5 4 2 (1) : 1	5 5 3 5, 3 3 3 (1) : 1
3 5 1 5, 5 3 3 (1) : 1	5 5 4 5, 1 2 4 (1) : 1
2 5 1 5, 5 4 2 (1) : 1	4 5 4 5, 2 2 5 (1) : 1
1 5 1 5, 5 5 1 (1) : 1	3 5 4 5, 2 3 3 (1) : 1
5 5 2 5, 4 1 5 (1) : 1	2 5 4 5, 2 2 4 (1) : 1
4 5 2 5, 4 2 4 (1) : 1	1 5 4 5, 2 1 5 (1) : 1
1 5 2 5, 4 5 2 (1) : 1	5 5 5 5, 1 1 5 (1) : 1
2 5 2 5, 4 4 2 (1) : 1	4 5 5 5, 1 1 5 (1) : 1
3 5 2 5, 4 3 3 (1) : 1	3 5 5 5, 1 1 5 (1) : 1
1 5 3 5, 3 3 3 (1) : 1	2 5 5 5, 1 1 5 (1) : 1
2 5 3 5, 3 3 3 (1) : 1	1 5 5 5, 1 1 5 (1) : 1
3 5 3 5, 3 3 3 (1) : 1	

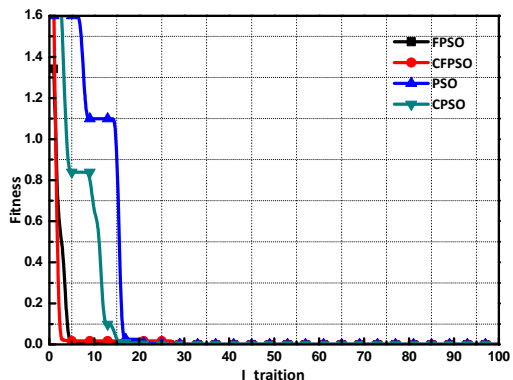
غیرگویایی می‌شود. لذا از نرم‌افزار "ORIGIN<sup>®</sup>" برای ترسیم این نمودارها استفاده شده است براساس نتایج این شبیه‌سازی‌ها برای توابع محک مورد نظر، با ابعاد مختلف (۲۰ و ۵۰ و ۱۰۰ و ۲۰۰)، بهترین جواب و متوسط جواب بدست آمده بیان شده‌اند. محروافقی "I\_traction" معرف نقطه همگرایی در تکرار فرآیند سیستم های معرفی شده و محور عمودی مشخص کننده ارزیابی (Fitness) توابع معرفی شده می‌باشند که در نمودارهای (۱ و ۳ و ۴) بیان می‌شوند. با توجه به نمودارهای (۱ و ۳ و ۴) و جداول (۲ و ۳ و ۴) و در نظر داشتن ماهیت تصادفی الگوریتم به خوبی نتیجه گرفته می‌شود که روش ارائه شده براساس سیستم فازی دارای سرعت همگرایی بالاتر در تعداد دفعات تکرار کمتر در مقایسه با الگوریتم استاندارد و الگوریتم‌های بررسی شده می‌باشد. این موضوع به خوبی بهینه بودن روش و مهم‌تر از آن پیاده‌سازی صحیح سیستم فازی و قوانین را بیان می‌کند.



شکل ۱۲- نمودار همگرایی تابع همگرایی Shepherd ۱۰۰ بعدی



شکل ۱۳- نمودار همگرایی تابع همگرایی Rastregin ۱۰۰ بعدی



شکل ۱۴- نمودار همگرایی تابع همگرایی Schawefel ۱۰۰ بعدی

(PSO, CPSO, CFPSO, FPSO (Proposed ALG)) شبیه‌سازی شدند و با استفاده از توابع محک استاندارد مورد ارزیابی قرار گرفته‌اند. این توابع دارای نقاط بهینه محلی فراوان و از طرف دیگر دارای نقطه بهینه سراسری نیز هستند. به همین منظور از ۴ تابع مشهور، Rosenber (۸-۱۱) معرفی شده‌اند استفاده می‌کنیم که هدف مینیمم سازی این توابع در نقطه صفر است.

$$f_{sph}(x) = \sum_{i=1}^{\dim(x)} x_i^2 \quad (8)$$

$$f_{sds}(x) = \sum_{i=1}^{\dim(x)} \sum_{j=1}^i x_j^2 \quad (9)$$

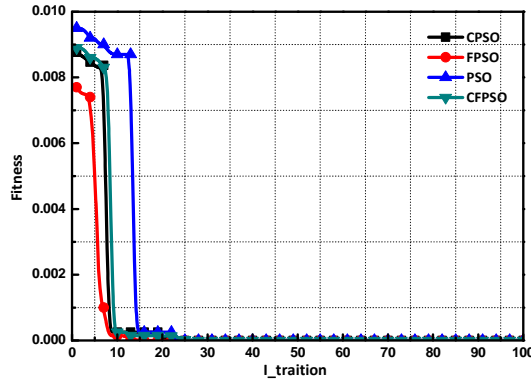
$$f_{ros}(x) = \sum_{i=1}^{\dim(x)-1} [100(x_{i+1} - x_i)^2 + (x_i - 1)^2] \quad (10)$$

$$f_{ras}(x) = \sum_{i=1}^{\dim(x)} [x_i^2 + 10 - 10\cos(2\pi x_i)] \quad (11)$$

در این ارزیابی، ۵۰ پرنده با ۲۰، ۵۰، ۱۰۰، ۲۰۰ بعد (متغیر)، و  $C_1=C_2=1.5$  و  $W=1.2$  بصورت پیش فرض برای شروع اولیه در نظر گرفته شده‌اند، سپس الگوریتم وارد قسمت فازی شده و از این مرحله تا انتهای الگوریتم کار به صورت فازی انجام می‌شود. بعد از ۱۰۰ بار تکرار نتایج مورد نظر بدست می‌آید. برای مقایسه بهتر شبیه سازی های بالا را ۵۰ بار اجرا کرده، نتایج ارائه شده در جداول (۲ و ۳ و ۴) میانگین نتایج بدست آمده و نمودارهای (۱ و ۳ و ۴) بهترین مقادیر از این ۱۰۰ بار تکرار می‌باشد. این جداول بیانگر مقایسه بین الگوریتم فازی پیشنهادی (FPSO<sup>8</sup>)، الگوریتم PSO استاندارد، الگوریتم CPSO<sup>9</sup> در [۱۲] و الگوریتم CFPSO<sup>3</sup> در [۱۳، ۲۷] می‌باشد. در الگوریتم CPSO [۱۲] با اضافه کردن یک قسمت به معادله ۲، رابطه جدیدی برای سرعت در معادله ۱۱ بدست آمده است.

$$v_{i+1} = k[av_i + c_1 r_1 (x_{ipbest} - x_i)] + \sum \varphi(h) c_2 r_2 (x_{igbest} - x_i) \quad (12)$$

که در این معادله  $\varphi(h)$  همان توابع عضویت فازی بوده که در بدنه اصلی برنامه معرفی شده است. الگوریتم معرفی شده در [۱۳، ۲۷] نیز به همین ترتیب با اضافه کردن یک قسمت در معادله ۲، معادله ۱۲ دوباره بدست آمده است، با این تفاوت که در این معادله، قسمت  $\varphi(h)$  یا تابع عضویت فازی از یک نوع متفاوت از [۱۲] می‌باشد. برای ارزیابی الگوریتم پیشنهادی از نرم افزار Matlab استفاده شد، قسمت الگوریتم PSO بر پیشنهاد Eberhart نوشته شد، سپس توسط واسط کاربری (GUI) قسمت فازی پیاده‌سازی شد، به منظور ارتباط بین قسمت فازی و واسط کاربری از توابع "Readfis" برای فراخوانی قسمت فازی (با نامگذاری توسط یک متغییر) و Evalfis به منظور برآورد سازی مقادیر محاسبه شده ورودی الگوریتم PSO (چهار ورودی سیستم فازی) مقدار Crisp (مقدار عددی منتج از ورودی‌های PSO و سیستم فازی) بدست می‌آید. سپس با توجه به خروجی‌های سیستم فازی مقادیر مناسب  $W$ ،  $C_1$ ،  $C_2$  به الگوریتم PSO به منظور (هدایت صحیح ذرات)؛ ارزیابی تابع شایستگی پاس داده می‌شود. ارزیابی توابع منجر به تولید بهترین مقدار شایستگی توابع و نمودارهایی می‌شود. ترسیم این نمودارها توسط نرم‌افزار Matlab به‌همراه تابع hold on منجر به نمودارهای نسبتاً



شکل ۱۵- نمودار همگرایی تابع همگرایی Rosenberg ۱۰۰ بعدی

جدول ۲- نتایج همگرایی توابع محک در ۲۰ بعد

Function	FPSO average	FPSO best	CFPSO average	CFPSO best	CPSO average	CPSO best	PSO average	PSO best
F <sub>Shepher</sub>	0.059	6.66e-16	0.326	3.51e-9	0.118	3.51e-10	0.337	0.337
F <sub>Rastregin</sub>	0.033	6.66e-16	0.0935	4.82e-11	0.237	2.83e-15	0.225	0.225
F <sub>Schawefel</sub>	0.019	5.26e-12	0.998	6.35e-2	0.998	5.425e-4	0.156	0.156
F <sub>Rosenberg</sub>	0.025	4.325e-15	0.860	3.25e-7	2.658	8.352e-5	0.125	0.125

جدول ۳- نتایج همگرایی توابع محک در ۵۰ بعد

Function	FPSO average	FPSO best	CFPSO average	CFPSO best	CPSO average	CPSO best	PSO average	PSO best
F <sub>Shepher</sub>	0.057	7.24e-10	0.638	5.214e-6	0.0995	5.512e-10	0.558	0.585
F <sub>Rastregin</sub>	0.046	8.223e-8	0.259	5.326e-9	0.345	3.458e-8	0.325	0.289
F <sub>Schawefel</sub>	0.119	6.758e-9	1.258	8.548e-5	1.897	7.231e-5	0.875	0.358
F <sub>Rosenberg</sub>	0.025	5.275e-12	1.325	4.289e-6	3.587	5.736e-4	0.536	3.003

جدول ۴- نتایج همگرایی توابع محک در ۱۰۰ بعد

Function	FPSO average	FPSO best	CFPSO average	CFPSO best	CPSO average	CPSO best	PSO average	PSO best
F <sub>Shepher</sub>	0.116	8.330e-15	0.038	2.834e-14	0.0258	8.331e-13	0.923	9.283e-13
F <sub>Rastregin</sub>	0.226	1.546e-12	0.208	1.565e-12	0.548	1.817e-08	0.485	1.6493e-06
F <sub>Schawefel</sub>	0.230	4.102e-19	0.658	1.58e-15	0.835	1.266e-11	0.975	1.37543e-06
F <sub>Rosenberg</sub>	0.672	6.221e-13	0.8275	3.17e-09	0.897	5.052e-08	0.776	6.036e-07

جدول ۵- نتایج همگرایی توابع محک در ۲۰۰ بعدی

Function	FPSO average	FPSO best	CFPSO average	CFPSO best	CPSO average	CPSO best	PSO average	PSO best
F <sub>Shepher</sub>	0.569	4.251e-3	1.0019	3.531e-2	1.564	5.145e-2	1.0563	1.852
F <sub>Rastregin</sub>	0.437	2.001e-5	0.528	7.564e-3	0.9214	4.553e-3	0.891	0.852
F <sub>Schawefel</sub>	0.620	5.802e-4	3.5483	1.648e-8	2.654	2.423e-2	0.9915	2.0127
F <sub>Rosenberg</sub>	0.782	2.587e-6	5.312	8.565e-3	3.423	5.569e-1	0.753	3.645

optimization," *the 3rd International Conference on Machine Learning and Cybernetics*, Shanghai, pp. 398-407, 2004.

[7] B. Al-kazemi, and C. K. Mohan, "Training feed forward neural networks using multi-phase particle swarm optimization," *the 9th International conference on Neural Information*, Bambamie, pp. 25-28, 2002.

[8] D. Merwe, and A. Engelbrecht, *Data clustering using particle swarm optimization*; <http://cirg.cs.up.ac.za/publications/CEC2003d.pdf>.

[9] T. R. Machado, and H. S. Lopes, "A hybrid particle swarm optimization model for the traveling salesman problem," *Adaptive and Natural Computing Algorithms Springer*, vol. 12, no. 10, pp. 255-258, 2005.

[10] V. G. Gudise, and G. K. Venayagamoorthy, "FPGA placement and routing using particle swarm," *Proc. IEEE Int Conf. Computer Society Annual Symposium on VLSI Emerging trends in VLSI Design (ISLVI 2004)*, pp. 514-518, 2004.

[11] T. Niknam, and H. D. Mojarrad, "A new fuzzy adaptive particle swarm optimization for non smooth," *Economic dispatch journal Energy*, vol. 21, no. 3, pp. 1746-1778, 2010

[12] X. Meng, and L. Jia, "A New Kind of PSO Convergent Fuzzy Particle Swarm Optimization and Performance Analysis," *Proc. Fourth Int Conf. Networked Computing and Advanced Information Management*, pp. 1251-1257, 2008.

[13] X. Meng, and L. Jia, "Study on Convergent Fuzzy Particle Swarm Optimization and Performance Analysis," *Proc. The 3rd Int IEEE Conf. Innovative Computing Information and Control (ICICIC'2008)*, pp. 9780-9785, 2008.

[14] X. Meng, and L. Jia, "A New Kind of PSO-Convergent Fuzzy Particle Swarm Optimization and Performance Analysis," *Proc. Fourth Int Conf. Networked Computing and Advanced Information Management*, pp. 25-28, 2008.

[15] M. Alizadeh, E. Fotoohi, V. Roshanae, and E. Safavieh, "Clustering Based Fuzzy Particle Swarm Optimization," *Proc. 28th North American Fuzzy Information Processing Society, Annual Conf. (NAFIPS2009)*, pp. 14-17, 2009.

[16] N. S. Niasar, J. Shanbezade, M. M. Perdam, and M. Mohajeri, "Discrete Fuzzy Particle Swarm Optimization for Solving Traveling Salesman Problem," *International Conference on Information and Financial Engineering*, pp. 75-81, Iran, 2009.

[17] X. Cui, "Document, Clustering using particle swarm optimization," *IEEE Journal*, vol. 10, no. 4, pp. 185-191, 2005.

[18] L. Hongbo, and M. Abraham, "Fuzzy Adaptive Turbulent Particle Swarm Optimization," *IEEE Conference*, pp. 39-47, Turkey, 2005.

[19] J. Liebowitz, *The Handbook of Applied Expert Systems*, CRC Press, ISBN: 0849331064, pp. 131-135, 1999.

براساس نمودارهای حاصل از ارزیابی توابع به خوبی دیده می‌شود که ۲ مشکل اصلی الگوریتم استاندارد در ۴ تابع محک به طور قابل ملاحظه‌ای برطرف شده و برتری واقعی روش ارائه شده به الگوریتم استاندارد، الگوریتم CPSO در مرجع [۱۲] الگوریتم CFPSO در مراجع [۱۳، ۲۶] را توسط نمودارهای فوق به اثبات رساندیم.

## ۶- نتیجه گیری

با توجه به نتایج بدست آمده از جداول (۲ و ۳ و ۴ و ۵)، نتیجه می‌گیریم که الگوریتم پیشنهادی در تمامی توابع محک استفاده شده بهترین بوده و اختلاف محسوسی براساس نمودارهای (۱ و ۲ و ۳ و ۴) نسبت به الگوریتم‌های مذکور وجود دارد که برتری الگوریتم پیشنهادی را نشان می‌دهد. این الگوریتم در تشخیص بهینه‌های محلی موفق بوده و توانسته خود را از دام بهینه‌های محلی رها کند و بهینه سراسری را پیدا کند. در بعضی از نمودارها همانند نمودارهای (۴ و ۲) رفتار نسبتاً غیرخطی و حتی قابل تاملی از توابع دیده می‌شود که این بخاطر مقادیر کاملاً تصادفی و غیر قابل پیش‌بینی تولیدی توسط الگوریتم در تلاش برای همگرایی است. می‌توان گفت که استفاده از توابع غیر خطی رفتار الگوریتم استاندارد را با توجه به ماهیت تصادفی‌اش 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 حرکت دسته جمعی ذرات برای حل توابع غیر خطی 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 دو ایده هوش جمعی و منطق فازی می‌تواند پاسخی برای این چالش محسوب شود. نتایج شبیه‌سازی نشان می‌دهد که الگوریتم بهینه‌سازی جمعی پرندگان استعداد گیر افتادن در بهینه محلی را دارد. استفاده از منطق فازی 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.

## مراجع

[۱] م. رستمی شهر بابکی، و م. نظام‌آبادی پور، "روش جدیدی برای الگوریتم pso باینری"، پردازش تصویر، بخش مهندسی برق، دانشگاه شهید بهنر کرمان، چهاردهمین کنفرانس ملی برق ایران (ICE 2006)، ص ۱۲۱-۱۲۰، ۱۳۸۵.

[۲] م. نورزوی بیرامی، و م. میبیدی، "بهبود بهینه‌سازی گروه ذرات با استفاده از منطق فازی"، دانشکده مهندسی کامپیوتر و فناوری اطلاعات دانشگاه صنعتی امیرکبیر، کنفرانس داه‌کاوی (ICDM 2008)، تهران، ص ۶۱-۵۴، ۱۳۸۷.

[۳] ح. نظام‌آبادی پور، و م. رستمی شهر بابکی، "تعمیمی بر الگوریتم GCBPSO". دوازدهمین کنفرانس مهندسی کامپیوتر ایران، ص ۲۹-۳۵، ۱۳۸۵.

[4] R. Kennedy, "Eberhart, Particle Swarm Optimization," *IEEE, 33rd Applied Imagery*, pp. 1942-1948, 1995.

[5] H. A. Firip, and E. Goodman, "Swarmed feature selection," *IEEE Conf of the 33rd Applied Imagery Pattern Recognition, AIPR'04*, pp. 251-259, 2004.

[6] H. B. Lip, Y. Y. Tang, J. Meng, and Y. Jp, "Neural networks learning using best model particle swarm

*Proc. 3rd International Conf. Technical and physical Problems in Power Engineering*, pp. 5-9, 2006.

[34] M. Sugeno, *Industrial Applications of Fuzzy Control*, Elsevier, Book, New York, 1985.

[35] J. Liebowitz, *The Handbook Of Applied Expert Systems*, CRC Press, ISBN: 0849331064, 1999 Krishnamoorthy.

[36] L.-Y. Chuang, S.-W. Tsai, and C.-H. Yang, "Improved Catfish Particle Swarm Optimization with Fuzzy Adaptation," *IEEE FUZZY Proceeding conference*, 20-24, KORAE, 2009.

[37] Y. Shi, and R. C. Eberhart, "Fuzzy adaptive particle swarm optimization," *IEEE proceedings Conference*, pp. 101-106, 2001.

[38] F. Herrera, and M. Lozano, "Fuzzy adaptive genetic algorithm: design," *Journal of Soft Computing*, vol. 7, no. 4, pp. 545-562, 2009.



**محمد فیوضی** در حال حاضر در دانشگاه حکیم سبزواری (تربیت معلم سبزواری سابق) در آزمایشگاه پردازش سیگنال، بخش مهندسی پزشکی، دانشکده مهندسی برق و کامپیوتر مشغول به فعالیت می‌باشد. مدرک کارشناسی را از دانشگاه آزاد اسلامی واحد ساوه در رشته مهندسی برق - قدرت در سال ۱۳۸۶ اخذ نموده و در حال حاضر دانشجوی ترم آخر رشته مهندسی پزشکی - بیو الکترونیک در دانشگاه حکیم سبزواری می‌باشد. زمینه‌های حرفه‌ای مورد علاقه ایشان، پردازش سیگنال (تصویرهای دیجیتال و بیولوژیکی)، بازشناسی الگو، شبکه‌های عصبی مصنوعی و تطبیقی فازی عصبی، بهینه‌سازی، منطق فازی، محاسبات نرم و هوش مصنوعی 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@yahoo.com



**جواد حدادنیای** در حال حاضر استاد دانشکده مهندسی برق و کامپیوتر دانشگاه حکیم سبزواری (تربیت معلم سبزواری 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 دیجیتال، 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 سر دبیر 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@sttu.ac.ir

[20] C. Krishnamoorthy, and S. Rajeev, *Artificial Intelligence and Expert Systems for Engineers*, CRC Press LLC ISBN: 0849391253, pp. 24-29, 1996.

[21] J. Kennedy, and R. Mendes, "Population Structure and Particle Swarm Performance," *Proc. IEEE Conf. Evol. Computing*, pp. 1671-1676, 2002.

[22] Y. Shi. Eberhart, "Parameter selection in particle swarm optimization," *Proc. Annual Conf. Evolutionary Programming*, pp. 174-78, March 1998.

[23] Y. Shi. Eberhart, "Empirical, study of particle swarm optimization," Conference on Evolutionary Programming Washington DC, pp. 21-27, USA, 6-9 July, 1999.

[24] L. Hongbo, and M. Abraham, "Fuzzy Adaptive Turbulent Particle Swarm Optimization," *IEEE Conference*, pp. 39-47, Slovakia, 2005.

[25] X. Feng, J. Zhang, and Z. Yang, *Adaptive, Particle Swarm Optimization on Individual Level*, IEEE, conference, China, pp. 1215- 1218, 2002.

[26] X. Meng, and L. Jia, "Study on Convergent Fuzzy Particle Swarm Optimization," *Proc. Fourth International Conf. Networked Computing and Advanced Information Management*, 87-94, 2008.

[27] H. A. Firip, and E. Goodman, "Swarmed feature selection," *IEEE Conference Proceedings of the 33rd Applied Imagery Pattern Recognition Workshop*, pp. 112-118, Canada, 2004.

[28] H. B. Liu, Y. Tang, J. Meng, and Y. Jp, "Neural networks learning using vbest model particle swarm optimization," *Proc. The 3rd International Conf. Machine Learning and Cybernetics*, pp. 3157-3159, 2004.

[29] B. Al-kazemi, and C. K. Mohan, "Training feed forward neural networks using multi-phase particle swarm optimization," *Proceedings of the 10th International conference on Neural Information*, pp. 2615-2619, Sweden, 2002.

[30] V. G. Gudise, and G. K. Venayagamoorthy, "FPGA placement and routing using particle swarm optimization" *the IEEE Computer Society*, pp. 307-308, Iran, 2005.

[31] M. Nasri, H. Nezamabadi-pour, and M. M. Farsangi, "Design of a PID controller using PSO algorithm Incorporating fuzzy objective function," *Proc. The 6th Iranian Conf. Fuzzy Systems and 1st Islamic Conference on Fuzzy Systems*, pp. 157-168, 2006.

[32] M. M. Farsangi, H. Nezamabadi-pour, and K. Y. Lee, "Multi-objective VAr planning with SVC for a large power system using PSO and GA," *IEEE Power Systems Conference & Exposition*, pp. 274-279, Pakistan, 2006.

[33] S. Mollazei, M. M. Farsangi, and H. Nezamabadi-pour "Allocation of TCSC to enhance total transfer capability using guaranteed convergence particle swarm optimization,"



**سیدکمال‌الدین 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 استادیاری در دانشکده مهندسی برق مشغول به فعالیت می‌باشند. علایق مطالعاتی ایشان، کنترل خطی و غیر خطی، کنترل فازی، مدل‌سازی غیرخطی و کاربردهایش، آشوب، کنترل تطبیقی، کنترل فرآیندهای اتفاقی، کنترل چند متغیره و محاسبات نرم می‌باشد.

آدرس پست‌الکترونیکی ایشان عبارت است از:

sk\_mousavi@iust.av.ir



**محمد علی پور** متولد ۱۳۶۲ و دانش آموخته کارشناسی مهندسی پزشکی و کارشناسی ارشد الکترونیک - پردازش سیگنال به ترتیب از دانشگاه‌های صنعتی سهند تبریز و تربیت معلم سبزوار می‌باشد. او هم‌اکنون در رشته مهندسی پزشکی و در مقطع دکتری در دانشگاه صنعتی چالمرز - سوئد مشغول به تحصیل است پردازش سیگنال و تصویر (عموماً تصاویر پزشکی)، شناسایی الگو و بهینه‌سازی از زمینه‌های تحقیقاتی مورد علاقه وی می‌باشد.

آدرس پست‌الکترونیکی ایشان عبارت است از:

mohammad.alipoor@ymail.com

#### اطلاعات بررسی مقاله:

تاریخ ارسال: ۹۰/۷/۱۵

تاریخ اصلاح: ۹۱/۸/۲۴

تاریخ قبول شدن: ۹۱/۹/۵

نویسنده مرتبط: محمد فیوضی، دانشکده مهندسی برق و کامپیوتر، دانشگاه

حکیم سبزوار، سبزوار، ایران.