شناسایی سیستم فوق آشوب راسلر با استفاده از الگوریتم شعله-پروانه بهبودیافته با الگوریتم جستجوی ممنوعه

مصطفی سعادتی فر[1] ، مهدی یعقوبی[2*]، مهسا اسماعیل نیا[1]

چکیده

در این مقاله به مسئله_ی شناسایی سیستم های آشوبی به کمک الگوریتم شعله پروانه بهبود یافته با الگوریتم جستجوی ممنوعه پرداخته شده است. مسئله ی شناسایی سیستمهای آشوبی، یک مسئله با بهینه های محلی زیاد است. بدین منظور به یک الگوریتم بهینه سازی قدرتمند برای حل آن نیاز است. الگوریتم شعله-پروانه که از حرکت حلزونی پروانه در اطراف شمع الهام گرفته است دارای ویژگی های متعددی ازجمله توازن بین اکتشاف و استخراج می باشد. اما از نظر جستجوی محلی ضعیف است و نیاز به بهبود دارد. در این مقاله به منظور بهبود این الگوریتم، برای اولین بار از روش جستجوی ممنوعه به طور ترکیبی استفاده شده است. هدف بهبود توانایی استخراج و جلوگیری از به دام افتادن در بهینه ی محلی در الگوریتم شعله پروانه است. سپس از این الگوریتم برای شناسایی پارامترهای مدل سیستم آشوبی راسلر استفاده شده است. نتایج شبیه سازی ها بر روی مسئله توابع محک و شناسایی پارامتر سیستم آشوبی حاکی از برتری الگوریتم شعله پروانه ترکیبی باجستجوی ممنوعه نسبت به الگوریتمهای بهینه سازی ازدحام ذرات وشعله پروانه استاندارد است.

کلمات کلیدی: تخمین پارامتر ، سیستمهای فوق آشوب، الگوریتم شعله-پروانه، الگوریتم جستجوی ممنوعه

دریافت مقاله: ۱۴۰۳/۰۳/۵

پذیرش مقاله: ۱۴۰۳/۰۶/۶

۱-مقدمه

مفهوم آشوب یکی از مفاهیم جدید و بنیادی علم نوین است که در بسیاری از پدیده های دنیای واقعی اعم از سیستمهایی که دارای رفتاری به ظاهرتصادفی و بی نظم وسیستم هایی دارای رفتاری معین، می تواند رخ دهد[۱] سیستم های آشوبناک رفتارهای دینامیکی دارند که دارای برخی از ویژگی های خاص میباشند که از جمله آنها حساسیت بیش از حد سیستمهای آشوبناک به شرایط اولیه می باشد که موقعیت جدید سیستم به رفتارهای پیشین سیستم وابسته است و تغییری اندکی در شرایط اولیه چنین سیستم هایی، باعث تغییرات بسیار قابل توجه در شرایط نهایی می شود. از ویژگیهای دیگر آشوب، وجود طیف فرکانسی پیوسته می باشد.شناسایی سیستم علاوه بر اینکه در بحث ایجاد مدل برای طراحی کنترل کننده کاربرد دارد کاربردی مهم و چه بسا اصلی تر آن در ایجاد فهم بهتر از سیستم و شبیه سازی حالتهای مختلفی می باشد که ممکن است در عمل پیاده سازی آنها غیر ممکن و یا هزینه بر باشد. در مدلسازی سیستم یا یک پدیده، به صورت ریاضی توصیف می شود. در این سیستم ورودی و خروجی وجود دارد . برای توصیف این سیستم باید مدلی را در نظر گرفته شود که اگر همان ورودیها به آن اعمال شود اختلاف خروجی مدل با سیستم اصلی صفر شود . تخمین مدل دقیق امکان پذیر نیست، بنابراین سعی می شود این اختلاف کمینه گردد. هر چه این اختلاف کمتر شود مدل دقیق تر خواهد بود. این خطا را میتوان به صورت میانگین مربعات آن نوشت و به صورت تابع هزینه در آورد وسپس به کمک روشهای مناسب آن را کمینه نمود

[1]* پست الکترونیک نویسنده مسئول:yaghoobi@mshdiau.ac.ir

۱. کارشناسی ارشد رشته هوش مصنوعی و رباتیک ، دانشگاه آزاد اسلامی واحد مشهد

۲. دانشیار، دانشکده مهندسی برق، دانشگاه آزاد اسلامی واحد مشهد

به عمل بدست آوردن این مدل شناسایی سیستم گفته می شود می توان یک سیستم را به صورت معادلات دیفرانسیلی توصیف نمود. در این معادلات دیفرانسیل ضرایبی از بردارهای حالت وجود دارند که به آن ضرایب پارامترهای سیستم گفته میشود. فرض میشود که سیستمی با ضرایب معلوم و سیستمی با ضرایب مجهول وجود دارد. تابع هزینه به صورت اختلاف خروجی گرفته شده از دو سیستم بوسیله میانگین مربعات خطا[1] بین دو خروجی نوشته میشود. بنابراین میتوان با روش های بهینه سازی این تابع هزینه را کمینه نمود. از آنجایی که این مسئله دارای بهینه های محلی زیادی است، از الگوریتم های فرا اکتشافی می توان استفاده کرد زیرا احتمال به دام افتادن در بهینه های محلی در مقایسه با روش های مبتنی بر گرادیان خیلی پایینتر است.

الگوریتمهای فرا اکتشافی، روشهای جستجوی تصادفی هستند که تکامل طبیعی زیستی و یا رفتار اجتماعی موجودات که از طریق یادگیری، سازش و تکامل، هدایت می گردند، را تقلید می نمایند. این الگوریتمها با پیروی از قوانین احتمالی به جای قوانین قطعی و عدم نیاز به مشتقگیری و یا هر گونه اطلاعات کمکی و تنها با تابع هدف وشیوه تعیین ارزیابی از اطلاعات خام، جهت جستجو را مشخص می نمایند. با بکارگیری این روشها در زمان محاسباتی معقول، راه حلهای سریع و نزدیک جواب بهینه سراسری برای مسائل بهینه سازی پیچیده فراهم می آید، لیکن هیچ کدام از این روشها تضمینی برای یافتن جواب بهینه سراسری به دست نمی دهد. این الگوریتمها را میتوان به دو دسته، روشهای خط سیر[2] یا مبتنی بر جستجوی محلی و روشهای مبتنی برجمعیت تقسیم نمود. روش های مبتنی بر جستجوی محلی روی راه حل های واحد کار میکنند که جستجوی تپه نوردی[3]، جستجوی محلی کاوشگرانه[4]، جستجوی ممنوع و باز پخت شبیه سازی شده[5] ، نمونه هایی از روش مبتنی بر جستجوی محلی می باشند.در مقابل روشهای مبتنی بر جمعیت، فرآیندهای جستجویی را انجام میدهند که تکامل مجموعه ای از نقاط را در فضای جستجو توصیف می کنند و همزمان، نه تنها با یک نقطه بلکه با یک مجموعه از نقاط ،جستجو می نمایند. این روشها به دو رده ی بهینه سازی هوش جمعی[6]و محاسبه تکاملی[7] تقسیم می شوند هوش جمعی یک تکنیک مبتنی بر بررسی رفتار جمعی در سیستمهای غیر متمرکز و خود سازمانده است که معمولا" از جمعیتی از عاملهای ساده تشکیل شده که بطور محلی با یکدیگر و محیطشان تعامل دارند. با اینکه ساختار کنترلی متمرکزی برای تحمیل رفتار عاملها وجود ندارد، تعاملات محلی بین عاملها اغلب منجر به بروز یک رفتار سراسری می گردد. از مهمترین الگوریتمهای مبتنی بر هوش جمعی میتوان به الگوریتمهای بهینه سازی شعله-پروانه[8]، کلونی مورچه ها[9]، بهینه سازی ازدحام ذرات [10]، الگوریتم ژنتیک[11]، الگوریتم کرم شب تاب[12]اشاره نمود.

در این مقاله به مسئله ی تخمین پارامتر سیستم آشوبی راسلر پرداخته شده است. تحقیقات مختلفی در زمینه ی شناسایی پارامتر سیستم های آشوبی توسط نویسندگان ارائه شده اند. جاوید و همکاران به شناسایی پارامترهای مجهول در سیستم فوق آشوب چن[13]پرداختند. تابع هدف به صورت میانگین حداقل مربعات بین بردارهای حالت سیستم ها با مقادیری واقعی و بهینه تخمین زده شده، تعریف شد. الگوریتم PSO برای پیدا کردن بهترین مقدار پارامتر مجهول استفاده شده، به طوری که حداقل مقدار تابع هدف بدست می آید. نتایج شبیه سازی بر روی سیستم فوق آشوب چن صورت گرفته و نشان

[1] mean-square error
[2] Trajectory Meth
[3] Hill Climbing
[4] Explorative Local Search
[5] Simulated Annealing
[6] Swarm Intelligence
[7] Evolutionary Computation
[8] Moth-Flame
[9] Ant-Colony
[10] Particle-swarm optimization
[11] Genetic Algorithm
[12] Fire Fly Algorithm
[13] Chen

دهنده دقت وکارایی بالای این الگوریتم در زمینه شناسایی پارامتر سیستم های آشوب می باشد[۲]. در[۳] تخمین پارامتر برای سیستم های آشوبی با استفاده از الگوریتم جستجوی فاخته همراه با یادگیری مبتنی بر مقابله مطالعه شد. در [۴] تخمین پارامتر برای سیستم های آشوبی بوسیله¬ی الگوریتم تکامل تفاضلی و الگوریتم کلونی زنبور عسل ترکیبی انجام شد. تخمین پارامتر سیستم های دینامیکی آشوبی با استفاده از الگوریتم بهینه سازی توده ذرات مبتنی بر رفتار کوانتومی در [۵] پیشنهاد شد. ژانگ و همکاران در ۲۰۱۵ تخمین پارامتر سیستم آشوبی غیر خطی بوسیله ی استراتژی TLBO بهبود یافته انجام دادند[۶]. وانگ و همکاران در ۲۰۱۵ یک الگوریتم بهینه سازی جستجوی فاخته¬ی بهبود یافته برای تخمین پارامتر سیستم های آشوبی پیشنهاد نمودند [۷]. یک الگوریتم دسته¬ی گرگ برای شناسایی پارامتر سیستم¬های آشوبی توسط لی و وو در ۲۰۱۶ پیشنهاد شد[۸]. لازوس و همکاران در ۲۰۱۶ از الگوریتم ترکیبی هوشمند بهینه سازی توده ذرات و کلونی مورچگان برای تخمین پارامتر سیستم های آشوبی استفاده نمودند[۹]. رحیمی و همکاران در ۲۰۱۶ شناسایی رفتار آشوبی در موتور سنکرون مغناطیس دائم را بوسیله ی الگوریتم بهینه سازی خفاش مطالعه نمودند[۱۰]. در [۱۱]شناسایی پارامتر سیستم¬های آشوبی بر اساس الگوریتم کلونی زنبور عسل مصنوعی ترکیب شده با استراتژی جستجوی فاخته انجام شد. در [۱۲] تخمین پارامتر سیستم های آشوبی با استفاده از یک الگوریتم گرده افشانی گل ها انجام شد. در [۱۳]یک الگوریتم جستجوی فاخته ی ترکیبی برای پارامترهای نامعلوم و تخمین زمان های تاخیر سیستم های آشوبی استفاده شد. موسوی و الفی در (۲۰۱۸) در تحقیقی، مسئله برآورد پارامترهای سیستم های آشوب با استفاده از الگوریتم شبیه سازی (FA) با محاسبات کسری مورد بررسی قرار دادند[۱۴]. در [۱۵] مسئله ی تخمین پارامتر سیستم های آشوبی یک بعدی بوسیله ی الگوریتم بیهنه سازی توده ذرات تضمین شده مطالعه شد. تخمین پارامتر سیستم های آشوبی با استفاده از روش تخمین چگالی جاذب-های عجیب در فضای حالت بوسیله¬ی شگفته و همکاران در ۲۰۱۹ پیشنهاد شد[۱۶]. چن و همکاران در ۲۰۱۹ از الگوریتم گرده افشانی گل های سراسری بهبود یافته برای شناسایی پارامتر سیستم فوق آشوبی استفاده نمودند[۱۷]. در [۱۸] یوسری و همکاران در ۲۰۱۹ از الگوریتم بهینه سازی نهنگ آشوبی برای تخمین پارامترهای رفتار آشوبی در موتور سنکرون مغناطیس دائم استفاده نمودند. ابراهیمی و همکاران در ۲۰۱۹ شناسایی پارامتر سیستم غیر خطی را با استفاده از یک نگاشت لوزی بهبود یافته مبتنی بر الگوریتم بهینه سازی آشوبی ارائه دادند[۱۹]. در [۲۰]یک تابع هزینه مبتنی بر نگاشت خود سازمانده برای تخمین پارامتر سیستم های زمان گسسته آشوبی بوسیله ی موسی زاده و شکفته پیشنهاد شد.

الگوریتم شعله پروانه، الگوریتم است که از هوش جمعی برخوردار است. توازان بسیار بالای اکتشاف و استخراج این الگوریتم، بازدهی و کارایی آن را بالا برده است[۲۱]. در این مقاله به جهت جلوگیری از به دام افتادن در بهینه¬ی محلی و بالا بردن دقت الگوریتم شعله پروانه، با با یک الگوریتم مبتنی بر جستجوی محلی به نام جستجوی ممنوعه [۲۲] ترکیب شده است. سپس پارامترهای مدل فوق آشوب راسلر با الگوریتم پیشنهادی تخمین زده می شود. به منظور ارزیابی این روش نتایج آن با الگوریتم های بهینه سازی توده ذرات، شعله پروانه ی معمولی مقایسه شده است.

سازمان دهی این مقاله به این صورت است. در بخش دوم مسئله ی تخمین پارامتر سیستم آشوبی پرداخته شده است. در بخش سوم سیستم فوق آشوبی راسلر معرفی شده است. در بخش چهارم الگوریتم بهینه سازی شعله پروانه و ترکیب آن با الگوریتم جستجوی ممنوع توضیح داده شده است. شبیه سازی ها و نتایج تخمین پارامتر سیستم فوق آشوبی راسلر ارائه شده است. در نهایت در بخش پنجم نتیجه گیری و پیشنهادات آورده شده است

۲- تخمین پارامتر سیستم آشوبی

برای تخمین پارامترهای یک سیستم آشوبناک باید مدلی برای پیاده سازی سیستم به صورت زیر در نظر گرفته شود:

$$\dot{X} = F(X, X_0, \theta_0) \qquad (۱)$$

در این سیستم، بردار حالت به صورت $X = (x_1, x_2, ..., x_N)^T \in R^n$ و X0 به بردار حالت اولیه اشاره می کند $\theta_0 = (\theta_{10}, \theta_{20}, ...\theta_{d0})^T$ به عنوان پارامترهای اصلی مسئله در نظر گرفته می شوند. اگر سیستم مورد بررسی به صورت رابطه (۱)در نظر گرفته شود سیستم تخمین زده را می توان به صورت رابطه (۲) در نظر گرفت.

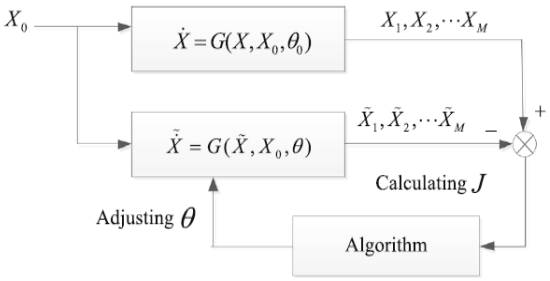
$$\dot{\tilde{X}} = F(\tilde{X}, X_0, \theta) \qquad (۲)$$

در این رابطه $\tilde{X} = (\tilde{x}_1, \tilde{x}_2, ..., \tilde{x}_n)^T \in R^n$ به بردار حالت اشاره میکند و $\theta = (\theta_1, \theta_2, ..., \theta_d)^T$ مجموعه پارامترهای تخمین زده شده را معرفی می کند. براین اساس، تابع برازندگی یا تابع هدف به صورت زیر تعریف می شود:

$$\min J = \frac{1}{M}\sum_{k=1}^{M} \| X_k - \tilde{X}_k \|^2 \qquad (۳)$$

در این رابطه M ، طول داده های سری زمانی برای تخمین پارامترها می باشد X_k و $\tilde{X}_k$ به ترتیب بردار اصلی و بردارهای تخمین زده در لحظه k می باشند.

سیستم های آشوبناک به خاطر دینامیک ناپایدار ذاتی شان به راحتی تخمین زده نمی شوند. علاوه بر این، به دلیل متغیرهای چندگانه ای که در مسئله وجود دارد و همچنین وجود نقاط بهینه چندگانه، فرآیندتخمین پارامترها ممکن است به راحتی در بهینه محلی گیر کند. به همین دلیل، جستجو در الگوریتم یا ترکیبی از الگوریتم هایی که خطر درگیر شدن با بهینه محلی را کم کند مورد توجه قرار می گیرد. تخمین پارامترهای یک سیستم آشوبناک را می توان شامل پروسه ای به صورت شکل ۱ در نظر گرفت.

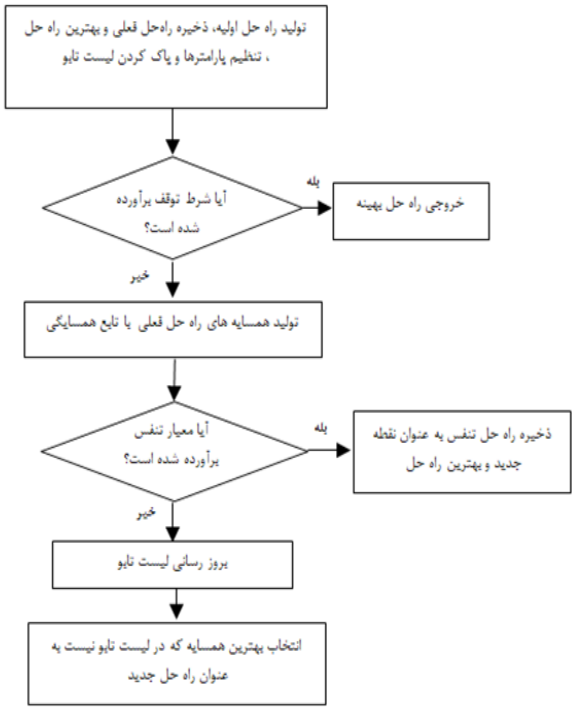


شکل(۱) بلوک دیاگرام تخمین پارامترهای ناشناخته یک سیستم آشوبناک [۱۷]

۳- الگوریتم ترکیبی

الگوریتم شعله پروانه از لحاظ توانایی جستجوی سراسری قوی است. اما از نظر جستجوی محلی ضعیف است و نیاز به بهبود دارد. هدف اصلی این مقاله، بهبود الگوریتم شعله پروانه از نظر جستجوی محلی و توانایی فرار از بهینه ی محلی به کمک تکنیک جستجوی ممنوعه است. تکنیک جستجوی ممنوعه یک تکنیک با توانایی جستجوی محلی قوی است. این تکنیک به توانایی جستجوی محلی و حرکت متنوع الگوریتم شعله پروانه در فضای جستجو کمک میکند. همچنین کمک میکند که احتمال به دام افتادن در بهینه های محلی در الگوریتم اصلی کم شود. با توجه به این ویژگی ها انتظار می رود که الگوریتم شعله پروانه بهبود یافته بتواند در کاربرد مسائل بهینه سازی موثرتر واقع شود. در این بخش ابتدا الگوریتم جستجوی ممنوعه معرفی شده و سپس الگوریتم شعله پروانه توضیح داده می شود. در نهایت ترکیب این دو الگوریتم پیشنهاد شده است. در بخش های بعدی نیز شبیه سازی ها و نتایج آن بر روی توابع محک و شناسایی پارامتر سیستم آشوبی راسلر داده شده است

۳-۱- جستجوی ممنوعه



شکل (۲) فلوچارت الگوریتم جستجوی ممنوعه

جستجوی ممنوعه[1] (TS) یک الگوریتم بهینه‌سازی فراابتکاری است که برای اولین بار در سال ۱۹۸۶ توسط گلوور Glover معرفی شد. براساس واژه‌نامه‌ی وِبستر، امروزه این واژه در معنای «ممنوعیت ایجاد شده به دلیل فرهنگ اجتماعی برای ایجاد اقدام حفاظتی» یا «ممنوعیت چیزی که دارای ریسک است»، به کار می‌رود . معنای اخیر واژه تابو، با تکنیک جستجوی ممنوعه کاملاً سازگار است. ریسکی که درالگوریتم جستجوی ممنوعه از آن اجتناب می‌شود، خطر مسیرهای نامناسب است. درشکل (۲) فلوچارت این الگوریتم نشان داده شده است.ذکر شود. شکل (۱) نمونه‌ای از چگونگی تنظیم یک شکل را نمایش می‌دهد.

۳-۲-۱-ساختار کلی جستجوی ممنوعه

برای رسیدن به جواب بهینه در یک مسئله بهینه‌سازی، الگوریتم جستجوی ممنوعه ابتدا از یک جواب اولیه شروع به حرکت می‌کند. سپس الگوریتم بهترین جواب همسایه را از میان همسایه‌های جواب فعلی انتخاب می‌کند. در صورتی که این جواب در فهرست ممنوعه قرار نداشته باشد، الگوریتم به طرف جواب همسایه حرکت می‌کند؛ در غیراین‌صورت الگوریتم معیاری به نام معیار تنفس را چک خواهد کرد. بر اساس معیار تنفس اگر جواب همسایه از بهترین جواب یافت شده تا کنون بهتر باشد، الگوریتم به سمت آن حرکت خواهد کرد، حتی اگر آن جواب در فهرست ممنوعه باشد. پس از حرکت الگوریتم به جواب همسایه، فهرست ممنوعه بروزرسانی می‌شود؛ به این معنا که حرکت قبل که بوسیله‌ی آن به جواب همسایه حرکت کردیم در فهرست ممنوعه قرار داده می‌شود تا از بازگشت مجدد الگوریتم به آن جواب و ایجاد سیکل جلوگیری شود. در واقع فهرست ممنوعه ابزاری در الگوریتم جستجوی ممنوعه است که توسط آن از قرار گرفتن الگوریتم در بهینه‌ی محلی جلوگیری می‌شود. پس از قرار دادن حرکت قبلی در فهرست ممنوعه، تعدادی از حرکت‌هایی که قبلاً در فهرست ممنوعه قرار گرفته بودند از فهرست خارج می‌شوند. مدت زمانی که حرکت‌ها در فهرست ممنوعه قرار می‌گیرند توسط یک پارامتر که زمان ممنوعه نام دارد تعیین می‌شود. حرکت از جواب فعلی به جواب همسایه تا جایی ادامه می‌یابد که شرط خاتمه دیده شود. حافظه الگوریتم می تواند از دو نوع recency و یا frequency باشد. جستجوی ممنوع، در فرایند کوتاه مدت مجسم می¬شود. این حافظه لیستی با ابعاد N رکورد می¬باشدکه N تا از ساختار کلی جستجوی ممنوعه اغلب جوابگوی مسائل بزرگ نیست. بنابراین به منظور افزایش قدرت الگوریتم از استراتژی‌های فهرست کاندید، تقویت و تنوع بخشی استفاده می شود.

۳-۲-الگوریتم شعله-پروانه

الگوریتم شعله پروانه یک الگوریتم بهینه سازی هوش اجتماعی است که توسط میر جلیلی و همکاران در ۲۰۱۵ معرفی شد. ایده اصلی الگوریتم شعله پروانه، مکانیزم هدایت پروانه ها در طبیعت به نام جهت گیری عرضی است.

پروانه ها دو مرحله مهم در زندگی خود تجربه می کنند:

۱- پروانه کرم[2] ۲- بزرگسال[3] . مهمترین حقیقت در پروانه، حرکت آنها در شب است. آنها با استفاده از نور ماه در شب پرواز می کنند. آنها از یک جهت یابی عرضی برای حرکت در شب استفاده می کنند. در این مکانیزم پروانه در یک زاویه ثابت نسبت به ماه حرکت می کند. این مکانیزم برای حرکت مستقیم در مسیرهای طولانی بسیار مناسب است در الگوریتم فرا ابتکاری شعله-پروانه جواب های کاندید پروانه ها هستند و متغیرهای جواب های کاندید موقعیت مکانی پروانه در یک جهت است. بنابراین تعداد متغیرهای پروانه می تواند به طور دلخواه انتخاب شود. از آنجا که الگوریتم بهینه سازی شعله-پروانه یک الگوریتم مبتنی بر جمعیت است، در هر مرحله یک تعداد جواب کاندید بدست می آید. جواب های کاندید در دو دسته پروانه ها[4] و شعله[5] یک MFOها تعریف می شوند. از آنجا که الگوریتم الگوریتم مبتنی بر جمعیت است، مجموعه ای از پروانه هادر یک ماتریس به شرح زیر نمایش داده میشوند:

[1] Tabu Search
[2] Larva
[3] Adult
[4] Moth
[5] Flame

$$M = \begin{pmatrix} m_{1,1} & \cdots & m_{1,d} \\ \vdots & \ddots & \vdots \\ m_{n,1} & \cdots & m_{n,d} \end{pmatrix} \quad (۴)$$

که در آن nتعداد پروانه ها و d تعداد متغیرها یا ابعاد می باشد. برای همه پروانه ها ، ما فرض می کنیم که یک آرایه برای نگهداری مقدار شایستگی متناظر با بردارموقعیت آنها به شکل روبرو وجود دارد.

$$OM = \begin{bmatrix} OM_1 \\ OM_2 \\ \vdots \\ OM_n \end{bmatrix} \quad (۵)$$

که در آنn تعداد پروانه ها می باشد. مقدار شایستگی، مقدار محاسبه شده توسط تابع شایستگی برای هر پروانه می باشد. بردار موقعیت هر شاپرک (ردیف اول در ماتریس M بعنوان مثال) به تابع برازش داده می شود وخروجی تابع برازش، بعنوان مقدار شایستگی پروانه مربوطه لحاظ می شود.

یکی دیگر از مولفه های کلیدی در این الگوریتم، شعله های آتش است. که برای نمایش آنها یک ماتریس شبیه به ماتریس زیر در نظرگرفته می شود:

$$F = \begin{pmatrix} f_{1,1} & \cdots & f_{1,d} \\ \vdots & \ddots & \vdots \\ f_{n,1} & \cdots & f_{n,d} \end{pmatrix} \quad (۶)$$

که در آن nتعداد پروانه ها و d تعداد متغیرها یا ابعاد می¬باشد. برای شعله ها همچنین فرض می¬شود که یک آرایه برای نگهداری مقدار شایستگی منتاظر با آنها به شرح زیر وجود دارد:

$$OF = \begin{bmatrix} OF_1 \\ OF_2 \\ \vdots \\ OF_n \end{bmatrix} \quad (۷)$$

که در آن n تعداد پروانه ها می باشد. در اینجا باید خاطر نشان ساخت که شاپرکها و شعله های آتش هر دو راه حل هستند. تفاوت بین آنها نحوه رفتار و شیوه بروزرسانی آنها در هرتکرار می باشد. شاپرک ها عامل های جستجوی واقعی هستند که دراطراف فضای جستجو حرکت می کنند، در حالی که شعله های آتش، بهترین مکان هایی هستند که تاکنون توسط شاپرک ها به دست آمده اند. بنابراین، هر شاپرک، در اطراف یک شعله جستجو میکند و در صورت یافتن راه حل بهتر آن را بروز می کند. با استفاده از این مکانیزم، شاپرک هرگز بهترین راه حل خود را از دست نمی دهد. به منظورمدل کردن ریاضی رفتار حرکت مارپیچی شاپرک حول یک شعله، موقعیت هرشاپرک باتوجه به شعله و بارابطه زیر بروز رسانی می شود:

$$M_i = S\left(M_i, F_j\right) \quad (۸)$$

نشان Fj امین پروانه می باشد و i نشان دهنده Miکه تابع مارپیچی می باشد. یک مارپیچ S امین شعله و jدهنده لگاریتمی به عنوان مکانیسم بروزرسانی اصلی شاپرک ها انتخاب شده است. نوسان دامنه مارپیچ نباید از فضای جستجو تجاوز کند. یک مارپیچ لگاریتمی برای الگوریتم MFO:بصورت زیرتعریف شده است

$$S\left(M_i, F_j\right) = D_i \cdot e^{bt} \cdot \cos\left(2\pi t\right) + F_j \quad (۹)$$

که درآن Di نشان دهنده فاصله شاپرک i ام از شعله j ام می باشد وbیک ثابت برای تعریف کردن شکل مارپیچ لگاریتمی وt یک عدد تصادفی دربازه [-۱،۱] است. مقدارD بصورت زیر محاسبه می شود:

$$D_i = \left|F_j - M_i\right| \quad (۱۰)$$

که در آن Mi شاپرک i ام وFj شعله j ام رانشان می دهد وِ Di فاصله بین شاپرک i باشعله j را نمایش می-دهد. معادله (۱۰) جایی است که مسیر پرواز مارپیچی شاپرک ها شبیه سازی شده است. همانطور که در این معادله دیده می شود، موقعیت بعدی یک شاپرک با توجه به شعله تعریف می شود. پارامتر t در معادله مارپیچ تعریف می کند که چقدر موقعیت بعدی شاپرک باید نزدیک شعله باشد. (مقدارt=-1، نزدیک ترین نقطه به شعله است، درحالیکه t=1 دورترین رانشان می دهد). بنابراین یک ابربیضی را می

توان دراطراف شعله درتمام جهات فرض کرد و موقعیت بعدی آن درداخل این فضا می باشد. حرکت مارپیچی جزءاصلی روش پیشنهادی است; زیرا نشان می دهد که چگونه شاپرک ها موقعیت خود را در اطراف شعله های آتش و نه لزوما درفضای بین آنها بروزرسانی می کنند. بنابراین، اکتشاف و بهره برداری فضای جستجو میتواند تضمین شود.

• شاپرک می تواند با تغییر t به هرنقطه ای درمحدوده شعله همگرا شود.

• مقادیر کمتر t ، منجر به فاصله نزدیک تر به شعله میشود.

• فرکانس بروزرسانی موقعیت در هر دو طرف شعله با نزدیکتر شدن شاپرک به شعله افزایش می یابد.

روش بروزرسانی موقعیت پیشنهادی میتواند بهره برداری دراطراف شعله های آتش را تضمین کند. به منظور بهبود احتمال یافتن راه حلهای بهتر، بهترین راه حل های بدست آمده تاکنون به عنوان شعله در نظر گرفته شده است. بنابراین ماتریس F درمعادله (۱۰) همیشه شامل n تا ازبهترین راه حل های اخیر است.

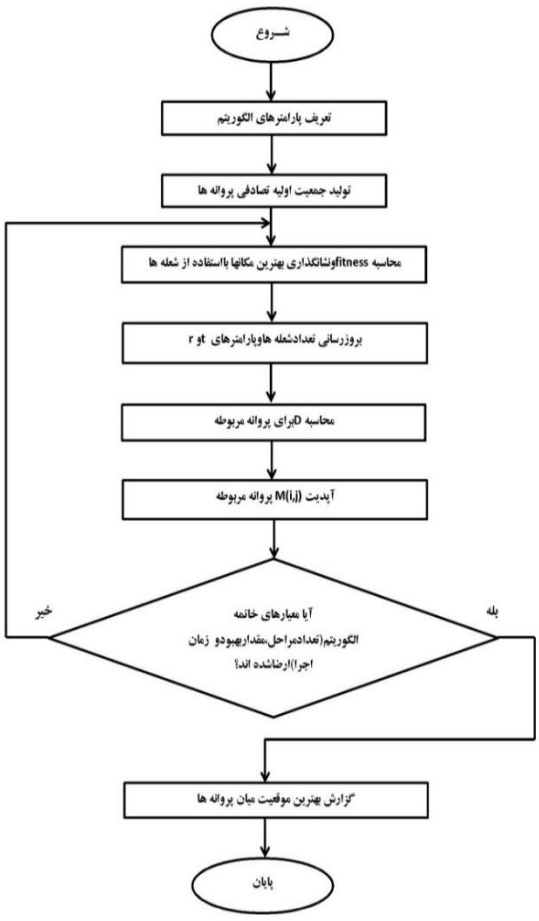
پروانه ها نیاز دارند در حین بهینه سازی، موقعیت خود را با درنظر گرفتن ماتریس شان بروزرسانی کنند. بمنظور تاکید بیشتر بر اکتشاف، فرض می شود که t یک عددتصادفی دربازه [r,1] است که درآن r به طورخطی از ۱- تا ۲- درطول تکرارکاهش می یابد. توجه داشته باشید که r به عنوان ثابت همگرایی نامیده میشود. با استفاده ازاین روش، شاپرک ها تمایل دارند که شعله های مربوطه خود رابادقت بیشتری نسبت به تعدادتکرارها موردبهره برداری قراردهند. دلیل اینکه چرا یک شعله خاص به هرکدام از شاپرکها اختصاص داده می شود، جلوگیری از گیر افتادن دربهینه محلی است. اگر تمام پروانه ها به سمت یک شعله جذب شوند، همه آنها به یک نقطه در فضاهای جستجو نزدیک می¬شوند؛ زیرا آنها فقط میتوانند به سوی شعله و نه به سمت خارج پرواز کنند. با این حال، ملزم کردن آنها برای حرکت در اطراف شعله های مختلف، باعث افزایش اکتشاف فضای جستجو و کاهش احتمال گیرافتادن در بهینه محلی میشود.

یکی دیگر از نگرانی های اینجا این است که بروز رسانی موقعیت شاپرک ها با توجه به مکان¬های مختلف در فضای جستجو میتواند استخراج را کاهش دهد. برای حل این نگرانی، یک مکانیسم تطبیقی برای تعداد شعله های آتش ارائه شده است.

فرمول زیر در این زمینه مورد استفاده قرار میگیرد:

$$\text{flame no}= round\left(N - l * \frac{N-1}{T}\right) \qquad (۱۱)$$

که درآن L شمارنده تکرارفعلی الگوریتم است وN حداکثر تعداد شعله ها می باشد وT حداکثر تعداد تکرارهای الگوریتم را نشان می دهد. شکل ۲-۱۵ نشان می دهد که N شعله درشروع تکرارالگوریتم وجوددارد، با این حال، شاپرک ها موقعیت خود را فقط باتوجه به بهترین شعله



شکل (۳): فلوچارت الگوریتم شعله-پروانه(Mirjalili, 2015).

درمراحل نهایی تکرار بروز می کنند. کاهش تدریجی تعدادی از شعله های آتش، اکتشاف واستخراج ازفضای جستجو را متعادل می کند.

همانطورکه در بالا توضیح داده شد، تابع P تا زمانی که تابع T مقدار true برگرداند، اجرا می شود. پس از خاتمه تابع P، بهترین شاپرک به عنوان بهترین تقریب بهینه ازجواب مطلوب برگشت داده می شود.

شکل ۳ فلوچارت پردازشی الگوریتم شعله-پروانه را نشان می دهد.

۳-۳- الگوریتم بهینه سازی شعله پروانه ترکیب شده با جستجوی ممنوع پیشنهادی

در این مقاله، به منظور بهبود در جستجوی محلی و سراسری الگوریتم شعله- پروانه از روش جستجوی تابو استفاده شده است. برای این منظور چند عمل تعریف شده و در هر بار از اجرا عملی که فراخوانی می شود به لیست ممنوع وارد می شود . سپس از میان عمل های دیگر یک عمل به اعمال شده است.

در این مسئله بهینه سازی ، الگوریتم جستجوی ممنوعه ابتدا از یک جواب بدست آمده از الگوریتم شعله پروانه ، شروع به حرکت میکند. اگر این حرکت به موقعیت بهتری دست یابد ، این موقعیت به جای موقعیت قبلی شعله پروانه قرار می گیرد . در غیر این صورت اتفاقی نمی افتد. پس از هر حرکت الگوریتم به سمت جواب همسایه موقعیت بدست آمده از الگوریتم شعله پروانه ، آن حرکت به فهرست ممنوعه اضافه میشود و در تکرار بعدی این حرکت در نظر گرفته نمی شود و حرکتی دیگر که خارج از فهرست ممنوعه است اجرا می شود. این به این دلیل است که از تکرار سیکل جلوگیری و از به دام افتادن در بهینه محلی اجتناب شود.

هدف از اعمال الگوریتم جستجوی ممنوعه بهبود در جستجوی محلی الگوریتم شعله پروانه است که با حرکت های آن بتوان جستجوی محلی حول همسایگی موقعیت های بدست آمده از شعله پروانه را انجام داد. این جستجوی محلی با یک توزیع تصادفی یکنواخت حول موقعیت انجام شده است. به منظور اعمال الگوریتم جستجوی ممنوعه در شعله-پروانه باسعی وخطا به این نتیجه رسیدیم که به تعداد ۱۰ درصد از کل جمعیت، جستجوی محلی انجام می شود.بطورمثال اگر جمعیت ۵۰ باشد، به تعداد ۵ مورد جستجوی محلی انجام شده و اگر موقعیت بهتری یافت شود، آن موقعیت جایگزین می شود. در غیر این صورت در موقعیت شعله-پروانه تغییری داده نمی شود. این مراحل تا اتمام تعداد کل تکرارهای الگوریتم ادامه می یابد.

۱) ایجاد جمعیت اولیه تصادفی

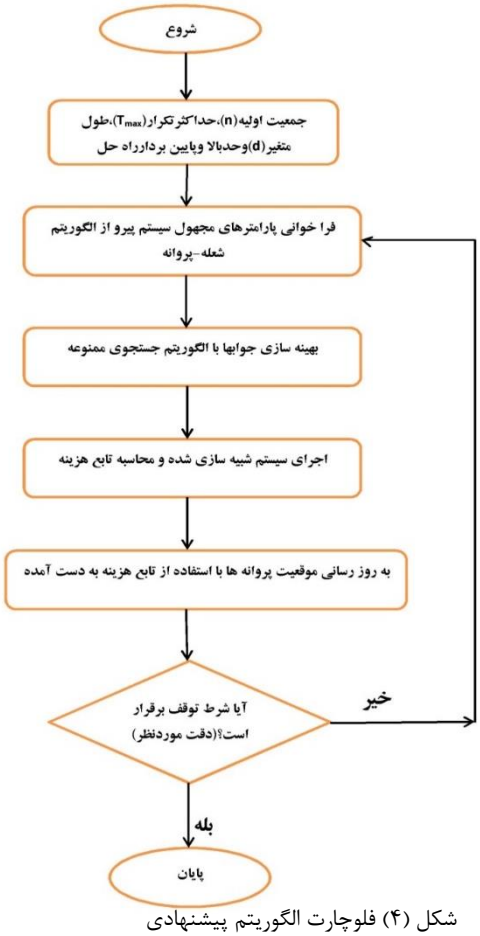
۲) فرا خوانی پارامترهای مجهول سیستم پیرو از الگوریتم شعله-پروانه

۳) بهینه سازی جوابها باالگوریتم جستجوی ممنوعه

۴) اجرای سیستم شبیه سازی شده و محاسبه تابع هزینه

۵) به روز رسانی موقعیت پروانه ها با استفاده از تابع هزینه به دست آمده

۶) پایان الگوریتم اگر به معیار مورد نظر رسیده باشد، در غیر این صورت بازگشت به مرحله ۲



شکل (۴) فلوچارت الگوریتم پیشنهادی

جدول ۱- نتایج بدست آمده مربوط به کمینه سازی توابع محک با استفاده از الگوریتم¬های **MFOTS,MFO,PSO**

MFOTS			MFO			PSO			توابع محک
MIN	MEAN	STD	MIN	MEAN	STD	MIN	MEAN	STD	معیار ارزیابی
0.00021579	9.25252	9.370193	19.99848	19.9985	3.5874920	1.6603	10.9996	9.65792	Ackley
0	0	0	0	0	0	6.3440	8.6139	1.249	Weierstrass
0.0344	0.1499	0.1302	0.03649	0.2435	0.1534	0.044	0.1848	0.195	Griewank
0.00718	39.9715	56.11	2.8091	46.73	67.53	9038.83	56.1146	25112.68	BentCigar
2.484e-08	0.00604	0.001696	6.65e-06	4.90008	15.49	0.12049	0.2636	0.148592	Discus

۴- شبیه سازی ونتایج

در این بخش ابتدا شبیه سازی¬ها بر روی توابع محک انجام می شود. سپس در زیر بخش دوم شبیه سازی ها برای شناسایی سیستم آشوبی راسلر انجام شده است.

۴-۱- نتایج شبیه سازی توابع محک

در این بخش پنج تابع محک از CEC2014 انتخاب شده است]۲۳[. توابع محک انتخابی شامل توابع محک Ackley, Weierstrass, Griewank, BentCigar, Discus هستند. الگوریتم شعله پروانه ترکیب شده با جستجوی ممنوع با الگوریتم های شعله پروانه و بهینه سازی توده ذرات مقایسه شده است.تعداد بیشینه تکرار برابر با ۲۰۰ و اندازه جمعیت کل برابر با ۵۰ در نظر گرفته شده است. هر الگوریتم ۳۰ بار اجرا شده است. نتایج مربوط به کمترین، میانگین و انحراف معیار مربوط به این اجراها در جدول ۱ نشان داده شده اند.

از جدول ۱ مشاهده می شود که در تمام توابع محک مقدار کمترین تابع هزینه ی بدست آمده از الگوریتم MFOTS در مقایسه با الگوریتم های MFO و PSO کمتر بدست آمده است. از نظر مقدار میانگین اجراها الگوریتم پیشنهادی نیز موفقتر از دو الگوریتم دیگر عمل کرده است. از نظر مقدار انحراف معیار نیز الگوریتم پیشنهادی در مقایسه با دو الگوریتم دیگر در مجموع مقادیر کمتری بدست آورده است. بنابراین عملکرد و کارایی الگوریتم پیشنهادی در کمینه سازی توابع محک در مقایسه با MFO,PSO به نتیجه رسیده است.

۴-۲- نتایج شبیه سازی مربوط به شناسایی پارامتر سیستم راسلر

سیستم آشوبی راسلر را می توان به صورت زیر توصیف نمود:

$$\begin{cases} \dot{x}_1 = -x_2 - x_3 \\ \dot{x}_2 = x_1 + a x_2 \\ \dot{x}_3 = (x_1 - c)x_3 + b \end{cases} \qquad (۱۲)$$

پارامترهای اصلی می $a = 0.2, b = 0.2, c = 5.7$ باشند. تعداد نمونه ها ۳۰۰ در نظر گرفته شده است.

مقدار بیشینه تکرار برابر با ۱۰۰، اندازه جمعیت برابر با ۵۰، در نظر گرفته شده است. در ادامه نتایج مربوط به ۱۵ اجرا آورده شده است. در جدول ۲ نتایج مقایسه ای مربوط به بهترین، میانگین و بدترین تابع هزینه بدست آمده از تخمین پارامتر سیستم راسلر به کمک الگوریتم های بهینه سازی توده ذرات، شعله پروانه معمولی و شعله پروانه به معمولی و شعله پروانه بهبود یافته به کمک جستجوی ممنوعه آورده شده است.

جدول ۲-نتایج تابع هزینه الگوریتم های PSO,MFO,MFOTS بر روی سیستم آشوبی راسلر

الگوریتم	بهترین	میانگین	بدترین
PSO	$1.6198e^{-13}$	$6.84721e^{-11}$	$3.1719e^{-10}$
MFO	$1.6383e^{-16}$	$5.4864e^{-13}$	$7.3591e^{-12}$
MFOTS	$6.0488e^{-20}$	$1.4902e^{-17}$	$6.7194e^{-17}$

۱

جدول ۳- مقایسه نتایج تخمین پارامترها در بهترین حالت

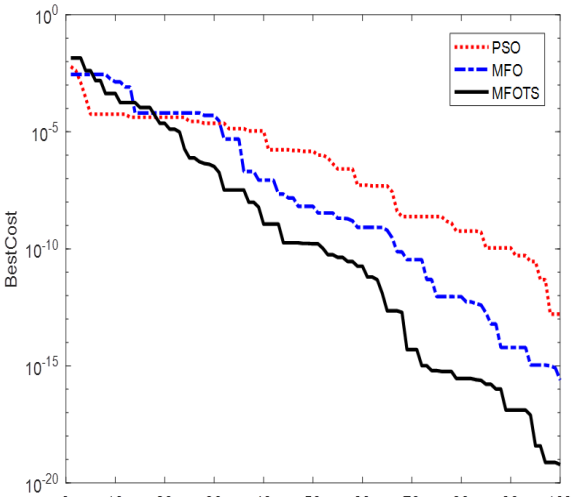
الگوریتم	a	b	c
PSO	0.200000152184943	0.199999591473626	5.699998930534023
MFO	0.200000004982759	0.200000018141152	5.699999937234080
MFOTS	0.200000000010356	0.199999998008726	5.699999996969637

از جدول ۳ مشاهده شده است که دقت تخمین بدست آمده از الگوریتم MFOTS در مقابل با دو روش PSO و MFO بیشتر است. درشکل ۵ منحنی تابع هزینه بدست آمده از سه روش نشان داده است. این شکل بهترین حالت را نشان می دهد

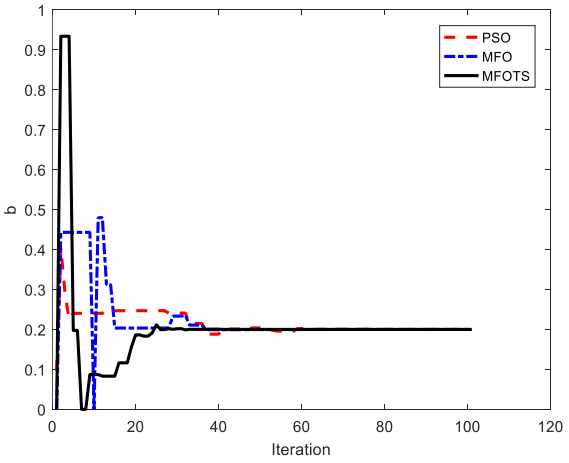
اماهدف ما بابالابردن می باشد وهمانطور که میبینیم،درپایان بهترین نتیجه از آن MFOTS می باشد.

در شکل های ۶ الی ۸ منحنی تخمین پارامترهای a,b,c به ترتیب نشان داده شده است.

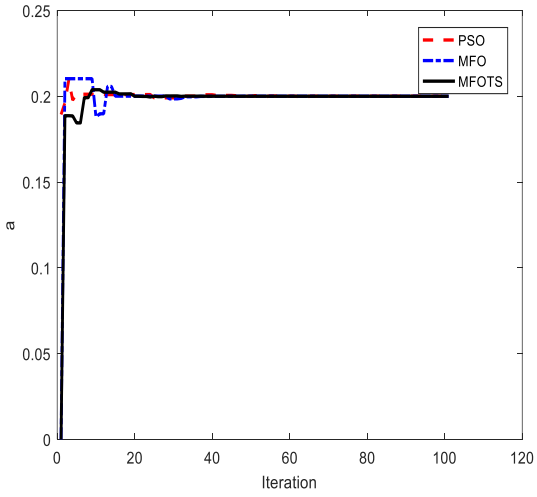
از شکل ۵ مشاهده می¬شود که روش پیشنهادی با تعدادتکرارکمتری در مقابل با دو روش دیگر به مقدار تابع هزینه کمتری رسیده است. درابتدا تا تکرار ۲۰ ام تابع هزینه PSO در مقابل ۲ روش دیگر کمتر بوده، اما پس از آن الگوریتم پیشنهادی MFOTS توانسته است که نتیجه¬ی بهتری از نظر کمترین تابع هزینه بدست آورد.



شکل (۵)- منحنی تابع هزینه بدست آمده از سه روش



شکل(۷)- منحنی تخمین پارامتر b بر حسب تعداد تکرار در حالت بهترین نتیجه



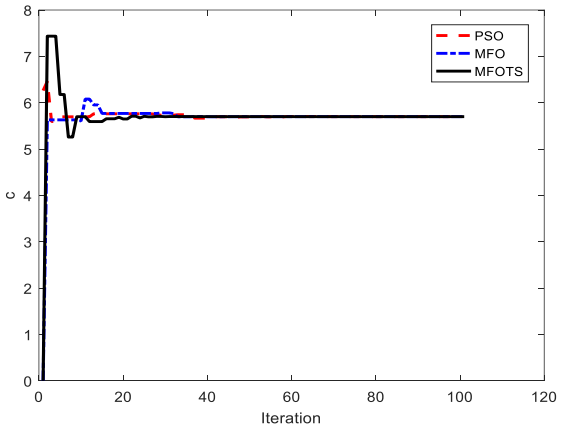
شکل(٦): منحنی تخمین پارامتر a بر حسب تعداد تکرار در حالت بهترین نتیجه

جدول ۳- نتایج تابع هزینه الگوریتم های PSO,MFO,MFOTS باآشوب لورنز

الگوریتم	بهترین	میانگین	بدترین
PSO	$1.3056e^{-9}$	$6.911e^{-7}$	$3.174207e^{-7}$
MFO	$6.5714e^{-14}$	$2.1556e^{-13}$	$2.6008e^{-12}$
MFOTS	$1.6567e^{-16}$	$1.927e^{-15}$	$1.2415e^{-13}$

جدول ٤: نتایج تابع هزینه الگوریتم های PSO,MFO,MFOTS باآشوب لوو

الگوریتم	بهترین	میانگین	بدترین
PSO	$2.6022e^{-07}$	$1.4087e^{-06}$	$3.8999e^{-06}$
MFO	$1.027e^{-12}$	$3.7824e^{-08}$	$9.6668e^{-08}$
MFOTS	$7.1416e^{-15}$	$5.3908e^{-12}$	$5.25e^{-10}$



شکل(۸): منحنی تخمین پارامتر c بر حسب تعداد تکرار در حالت بهترین نتیجه

از شکل های ۶ الی ۸ در مجموع نتیجه میشود که روش پیشنهادی الگوریتم شعله- پروانه بهبود یافته توسط جستجوی ممنوعه توانسته است پارامترها را در تعداد تکرار کمتری تخمین بزند.

پیاده سازی روی دو سیستم فوق آشوب دیگر، به نام لورنز و لوو انجام شد ونتایج بشرح ذیل بدست آمد.

باتوجه به جدول ۱، ۳ و ۴ به این نتیجه می¬رسیم که الگوریتم پیشنهادی نه تنها عملکرد بهتری در مقابل با دو روش دیگر در تخمین پارامتر سیستم فوق آشوب راسلر داشته است، بلکه این موفقیت را برای سیستم های آشوبی دیگر نیز میتوان تعمیم داد.

۵- نتیجه گیری

در این مقاله مسئله تخمین پارامتر سیستم آشوبی بررسی شده است که از کاربردهای آن می توان به مخابرات ایمن،رباتهای موبایلی،رمزنگاری وامنیت اطلاعات و... اشاره نمود. شناسایی سیستم، در قالب یک مسئله بهینه سازی بیان شد. در این مسئله بهینه سازی تابع هدف به صورت حداقل مربعات خطا بین دو بردار حالت سیستم اصلی و سیستم با پارامترهای تخمین زده شده تعریف می شود که باید بهینه گردد. الگوریتم های بهینه سازی تکاملی به دلیل محاسبات ساده ریاضی و نیفتادن در نقاط بهینه محلی برای حل این مسئله بهینه سازی نسبت به روش های کلاسیک مناسب تر می باشند. بنابراین این روشهای تکاملی و هوشمند بوسیله محققان بسیاری در سالهای اخیر مورد توجه قرار گرفت و تحقیقات وسیعی در زمینه همزمان سازی و شناسایی سیستمهای آشوبی صورت گرفت. در این مقاله یک الگوریتم بهینه سازی شعله پروانه ترکیب شده با جستجوی ممنوع پیشنهاد شد. الگوریتم شعله پروانه دارای توانایی جستجوی سراسری خوبی است. اما برای بهبود در جستجوی محلی آن از روش جستجوی ممنوع به صورت ترکیب شده با آن استفاده شد. به منظور ارزیابی این روش، این الگوریتم در مسئله ی تخمین پارامتر سیستم آشوبی به کار برده شد. سپس با دو روش بهینه سازی توده ذرات و شعله پروانهی معمولی مقایسه شد. نتایج نشان داد که الگوریتم MFOTS ، نسبت به روش های مشابه خود یعنی PSOو الگوریتم MFO بدلیل داشتن توانایی استخراج قویتر و در نتیجه برخورداری از دقت بالاتر، با موفقیت توانسته است پارامترهای سیستم آشوبی را بادقت وتعدادتکرار کمتر، تخمین بزند. به عنوان پیشنهاد در کارهای آینده می توان از الگوریتم شعله پروانه ترکیبی با جستجوی ممنوع بهبود یافته با نگاشت های آشوبی جهت

تخمین پارامتر سیستمهای آشوبی با معادلات پیچیده تر

استفاده کرد.

٦- مراجع:

[1] Chen, G. and Dong, X., 1999. From chaos to order: Methodologies prespectives and applications world scientific Singapore 1998. In *Handbook of Chaos Control*. Weinheim.

[2] Chang, W.D., Cheng, J.P., Hsu, M.C. and Tsai, L.C., 2012, December. Parameter identification of nonlinear systems using a particle swarm optimization approach. In *2012 Third International Conference on Networking and Computing* (pp. 113-117). IEEE.

[3] Xiang-Tao, L. and Ming-Hao, Y., 2012. Parameter estimation for chaotic systems using the cuckoo search algorithm with an orthogonal learning method. *Chinese Physics B*, *21*(5), p.050507.

[4] Li, X. and Yin, M., 2014. Parameter estimation for chaotic systems by hybrid differential evolution algorithm and artificial bee colony algorithm. *Nonlinear Dynamics*, *77*(1), pp.61-71.

[5] Ko, C.N., Jau, Y.M. and Jeng, J.T., 2015. Parameter Estimation of Chaotic Dynamical Systems Using HEQPSO. *J. Inf. Sci. Eng.*, *31*(2), pp.675-689.

[6] Zhang, H., Li, B., Zhang, J., Qin, Y., Feng, X. and Liu, B., 2016. Parameter estimation of nonlinear chaotic system by improved TLBO strategy. *Soft Computing*, *20*(12), pp.4965-4980.

[7] Wang, J., Zhou, B. and Zhou, S., 2016. An improved cuckoo search optimization algorithm for the problem of chaotic systems parameter estimation. *Computational intelligence and neuroscience*, *2016*.

[8] Li, H. and Wu, H., 2016. An oppositional wolf pack algorithm for parameter identification of the chaotic systems. *Optik*, *127*(20), pp.9853-9864.

[9] Lazzús, J. A., Rivera, M., & López-Caraballo, C. H. (2016). Parameter estimation of Lorenz chaotic system using a hybrid swarm intelligence algorithm. *Physics Letters A*, *380*(11-12), 1164-1171.

[10] Rahimi, A., Bavafa, F., Aghababaei, S., Khooban, M.H. and Naghavi, S.V., 2016. The online parameter identification of chaotic behaviour in permanent magnet synchronous motor by self-adaptive learning bat-inspired algorithm. *International Journal of Electrical Power & Energy Systems*, *78*, pp.285-291.

[11] Ding, Z., Lu, Z. and Liu, J., 2018. Parameters identification of chaotic systems based on artificial bee colony algorithm combined with cuckoo search strategy. *Science China Technological Sciences*, *61*(3), pp.417-426.

[12] Xu, S., Wang, Y. and Liu, X., 2018. Parameter estimation for chaotic systems via a hybrid flower pollination algorithm. *Neural Computing and Applications*, *30*(8), pp.2607-2623.

[13] Wei, J. and Yu, Y., 2017. An effective hybrid cuckoo search algorithm for unknown parameters and time delays estimation of chaotic systems. *IEEE Access*, *6*, pp.6560-6571.

[14] Mousavi, Y. and Alfi, A., 2018. Fractional calculus-based firefly algorithm applied to parameter estimation of chaotic systems. *Chaos, Solitons & Fractals*, *114*, pp.202-215.

[15] Sheludko, A.S., 2018. Parameter estimation for one-dimensional chaotic systems by guaranteed algorithm and particle swarm optimization. *IFAC-PapersOnLine*, *51*(32), pp.337-342.

[16] Shekofteh, Y., Panahi, S., Boubaker, O. and Jafari, S., 2019. Parameter Estimation of Chaotic Systems Using Density Estimation of Strange Attractors in the State Space. In *Recent Advances in Chaotic Systems and Synchronization* (pp. 105-124). Academic Press.

[17] Chen, Y., Pi, D. and Wang, B., 2019. Enhanced global flower pollination algorithm for parameter identification of chaotic and hyper-chaotic system. *Nonlinear Dynamics*, *97*(2), pp.1343-1358.

[18] Yousri, D., Allam, D. and Eteiba, M.B., 2019. Chaotic whale optimizer variants for parameters estimation of the chaotic behavior in Permanent Magnet Synchronous Motor. *Applied Soft Computing*, *74*, pp.479-503.

[19] Ebrahimi, S.M., Malekzadeh, M., Alizadeh, M. and HosseinNia, S.H., 2019. Parameter identification of nonlinear system using an improved Lozi map based chaotic optimization algorithm (ILCOA). *Evolving Systems*, pp.1-18.

[20] Mousazadeh, A. and Shekofteh, Y., 2020. Cost function based on the self-organizing map for parameter estimation of chaotic discrete-time systems. *Engineering Applications of Artificial Intelligence*, *94*, p.103817.

[21] Mirjalili, S., 2015. Moth-flame optimization algorithm: A novel nature-inspired heuristic paradigm. *Knowledge-based systems*, *89*, pp.228-249.

[22] Glover, F. and Laguna, M., 1998. Tabu search. In *Handbook of combinatorial optimization* (pp. 2093-2229). Springer, Boston, MA.

[23] Liang, J. J., Qu, B. Y., & Suganthan, P. N. (2013). Problem definitions and evaluation criteria for the CEC 2014 special session and competition on single objective real-parameter numerical optimization. *Computational Intelligence Laboratory, Zhengzhou University, Zhengzhou China and Technical Report, Nanyang Technological University, Singapore*, *635*, 490.