

رویکردی کارآمد مبتنی بر دنباله هندسی متغیر با زمان برای برقراری تعادل بین اکتشاف و بهره‌برداری در الگوریتم بهینه‌سازی ملخ

زهرا نخعی¹ و عباس قائمی بافقی²

¹ دانشجوی کارشناسی ارشد مهندسی کامپیوتر و عضو آزمایشگاه امنیت داده‌ها و ارتباطات در دانشگاه فردوسی مشهد

zahra.nakhai@um.ac.ir

² عضو هیات علمی گروه مهندسی کامپیوتر و سرپرست آزمایشگاه امنیت داده‌ها و ارتباطات در دانشگاه فردوسی مشهد

ghaemib@um.ac.ir

چکیده:

امروزه استفاده از الگوریتم‌های بهینه‌سازی بسیار گسترش یافته است. اکتشاف و بهره‌برداری دو ویژگی بسیار مهم در الگوریتم‌های بهینه‌سازی می‌باشند که برقراری تعادل میان این دو ویژگی بسیار اهمیت دارد و در الگوریتم بهینه‌سازی ملخ مورد توجه بوده است. اما الگوریتم بهینه‌سازی ملخ از یک ضریب تعادل بین این دو ویژگی استفاده می‌کند که رابطه بین آن‌ها به صورت خطی تغییر می‌کند و موجب می‌شود تعادل مناسب بین زمان رسیدن به همگرایی و یافتن بهینه سراسری برقرار نشود، یعنی با انتخاب ضریب بزرگ همگرایی زود هنگام رخ داده و احتمال گیر کردن در بهینه محلی افزایش می‌یابد و در صورت انتخاب ضریب کوچک، زمان همگرایی طولانی خواهد بود و در زمان محدود بهینه سراسری بدست نمی‌آید. در این مقاله با بکارگیری ضریب هندسی متناسب با زمان و شیوه‌ای جدید برای محاسبه آن، تعادل بین دو ویژگی اکتشاف و بهره‌برداری برای جلوگیری از همگرایی زود هنگام و دیر هنگام از یکسو و رسیدن به بهینه سراسری از سوی دیگر فراهم شده است. همچنین ده تابع برازش مختلف بعنوان تابع برازش در الگوریتم پایه و طرح پیشنهادی بررسی شده است. نتایج حاصل از مقایسه جنبه‌های مختلف شامل بهترین، بدترین و متوسط برازندگی و نیز انحراف معیار آن‌ها و زمان اجرا در ۵۰ بار اجرای مختلف نشان می‌دهد که عملکرد شیوه‌ی پیشنهادی نسبت به شیوه الگوریتم پایه در همه توابع برازش مورد بررسی ده برابر بهبود پیدا کرده است. با توجه به یکسان بودن تعداد تکرار در اجرای بهینه‌سازی‌ها، زمان همگرایی در همه یکسان است. بهترین حالت در تابع برازش ماتیا رخ می‌دهد که بهترین، بدترین و متوسط برازندگی و نیز انحراف معیار آن‌ها به ترتیب ۱۳-۱، ۸۹-۱، ۳۸-۱، ۴۷-۶، ۱۱-۱ و ۴۵-۱ است.

کلمات کلیدی: الگوریتم بهینه‌سازی ملخ، بهینه‌سازی، اکتشاف، بهره‌برداری، همگرایی.

در الگوریتم‌های فراابتکاری تعادل درست بین بهره‌برداری و اکتشاف بسیار مهم است. اکتشاف به دنبال یافتن جواب‌های متنوع و جدید در کل فضای جواب و بهره‌برداری به دنبال جستجو محلی و یافتن یک جواب بهتر پیرامون یک نقطه است. بنابراین اگر الگوریتمی فقط از اکتشاف استفاده کند آن الگوریتم به یک الگوریتم جستجو تصادفی تبدیل می‌شود که ممکن است هیچوقت به جواب نرسد و در مقابل اگر الگوریتمی فقط از بهره‌برداری استفاده کند در این صورت الگوریتم بسیار وابسته به نقطه شروع می‌شود که اگر نقطه شروع درست انتخاب نشود هیچوقت به جواب نخواهد رسید. در نتیجه الگوریتمی که فقط مبتنی بر یکی از مفاهیم بهره‌برداری و اکتشاف باشد کارآمدی لازم برای تبدیل به الگوریتم فراابتکاری را ندارد و الگوریتمی مناسب و قدرتمند است که ترکیب درستی از این دو ویژگی داشته باشد. در این مقاله با بکارگیری ضریب هندسی متناسب با زمان و شیوه‌ای جدید برای محاسبه آن، تعادل بین دو ویژگی اکتشاف و بهره‌برداری برای جلوگیری از همگرایی زود هنگام و دیر هنگام از یکسو و رسیدن به بهینه سراسری از سوی دیگر فراهم شده است. در ادامه مقاله بصورت زیر سازماندهی شده است:

در بخش دوم مروری بر کارهای انجام شده در این زمینه خواهیم داشت. سپس در بخش سوم به معرفی الگوریتم بهینه‌سازی ملخ می‌پردازیم. در بخش چهارم طرح پیشنهادی تشریح شده و در بخش پنجم نتایج ارزیابی و مقایسه ارائه شده است. در بخش ششم نتیجه‌گیری و کارهای آینده بیان و در انتها مراجع آورده می‌شود.

۱. مقدمه

شیوه‌ها و الگوریتم‌های بهینه‌سازی شامل دو دسته الگوریتم‌های دقیق و الگوریتم‌های تقریبی می‌باشند. الگوریتم‌های دقیق قادر به یافتن جواب بهینه به صورت دقیق هستند اما در مورد مسائل بهینه‌سازی سخت کارایی کافی ندارند و زمان اجرای آن‌ها متناسب با ابعاد مسئله به صورت نمایی افزایش می‌یابد. الگوریتم‌های تقریبی در زمان کوتاه قادر به یافتن جواب‌های خوب (نزدیک به بهینه) برای مسائل بهینه‌سازی سخت هستند. الگوریتم‌های تقریبی شامل الگوریتم‌های ابتکاری، فراابتکاری و فوق‌ابتکاری می‌باشند. دو مشکل اصلی الگوریتم‌های ابتکاری، گیر کردن آن‌ها در نقاط بهینه محلی و همگرایی زود هنگام به این نقاط است.

الگوریتم‌های فراابتکاری برای حل مشکلات الگوریتم‌های ابتکاری ارائه شده‌اند. در واقع الگوریتم‌های فراابتکاری، یکی از انواع الگوریتم‌های بهینه‌سازی تقریبی هستند که دارای راه‌کارهای فرار از نقاط بهینه محلی هستند و قابلیت کاربرد در طیف گسترده‌ای از مسائل را دارند. کلمه ابتکاری به معنی کشف کردن بواسطه سعی خطا و کلمه فرا به معنی یک سطح بالاتر است، در نتیجه فراابتکاری به معنی بهره‌برداری از اکتشاف انجام شده برای حل یک مسئله می‌باشد. انواع گوناگونی از این نوع الگوریتم‌ها در دهه‌های اخیر توسعه یافته است که همه این‌ها زیر مجموعه الگوریتم‌های فراابتکاری می‌باشند. از الگوریتم‌های شناخته شده فراابتکاری بر پایه جمعیت می‌توان الگوریتم بهینه‌سازی ملخ را نام برد.

۲. مروری بر کارهای پیشین

الگوریتم بهینه‌سازی ملخ، با الهام از رفتار حمله ملخ‌ها در طبیعت برای حل مشکلات بهینه‌سازی بصورت ریاضی مدلسازی و ارائه شده است. اگرچه از ابداع الگوریتم بهینه‌سازی GOA توسط آقایان صارمی و میر جلیلی [۱] در سال ۲۰۱۷ مدت زیادی نمی‌گذرد، اما مورد توجه جامعه علمی بوده است و در سال‌های اخیر یکی از مهم‌ترین و اثربخش‌ترین تحقیقات شیوه‌های فراابتکاری الهام‌گرفته از طبیعت بوده است. همچنین شیوه‌های تعادل بین اکتشاف و بهره‌برداری در الگوریتم‌های فراابتکاری و تابع برازش مورد استفاده در آن‌ها برای الهام بخشی در طرح پیشنهادی مرور شده و در ادامه ارائه می‌شود.

۲-۲ شیوه‌های تعادل بین اکتشاف و بهره‌برداری در الگوریتم‌های فراابتکاری

الگوریتم‌های فراابتکاری مختلف از شیوه‌های مختلفی برای برقراری تعادل بین اکتشاف و بهره‌برداری استفاده کرده‌اند، از جمله می‌توان به موارد زیر اشاره کرد.

الگوریتم بهینه‌سازی ازدحام ذرات (PSO) به عنوان یک شیوه جستجوی غیر قطعی برای بهینه‌سازی تابعی ارائه شده که از حرکت دسته جمعی پرنده‌گانی که به دنبال غذا می‌باشند، الهام گرفته شده است. این الگوریتم با استفاده از ضریب W تعادل بین اکتشاف و بهره‌برداری برقراری می‌کند، که به صورت زیر تعریف می‌شود [۹]:

$$w_t = w_0 - \left[(w_0 - w_t) \times \frac{t}{T} \right], t = 1, 2, \dots, T \quad (1-2-2)$$

الگوریتم اجماع زنبور عسل (BCO) مبتنی بر الگوریتم جست‌وجو است که شبیه‌سازی رفتار جست‌وجوی غذای گروه‌های زنبور عسل است. این الگوریتم با استفاده از ضریب L تعادل بین اکتشاف و بهره‌برداری برقراری می‌کند، که به صورت زیر تعریف می‌شود [۱۰]:

$$L_k = \frac{L_k - L_{min}}{L_{max} - L_{min}} \quad (2-2-2)$$

الگوریتم بهینه‌سازی علف‌های هرز یک الگوریتم جدید قدرتمند با الهام از تکثیر و رشد علف‌های هرز است. این الگوریتم با استفاده از ضریب σ تعادل بین اکتشاف و بهره‌برداری برقراری می‌کند، که به صورت زیر تعریف می‌شود [۱۱]:

$$\sigma_t = \frac{(T-t)^\beta}{(T)^\beta} (\sigma_{initial} - \sigma_{final}) + \sigma_{final} \quad (3-2-2)$$

الگوریتم خفاش، نوعی الگوریتم هوش جمعی است که از رفتار خفاش‌های کوچک هنگام جهت‌یابی برای شکار، الهام گرفته شده است. این الگوریتم با استفاده از ضریب λ تعادل بین اکتشاف و بهره‌برداری برقراری می‌کند، که به صورت زیر تعریف می‌شود [۱۲]:

$$\lambda = [\lambda_{min} + (\lambda_{max} - \lambda_{min})] \times \text{Rand } j \in M, i \in N \quad (4-2-2)$$

الگوریتم‌های فراابتکاری دیگری مانند الگوریتم اجماع مورچه‌ها [۱۳]، الگوریتم جهش قورباغه، الگوریتم نیروی مرکزی [۱۴] و [۱۵]، الگوریتم بهینه‌سازی جغرافیای زیستی [۱۶]، الگوریتم شبیه‌ساز تبرید [۱۷]، الگوریتم تکثیر گیاه [۱۸]، الگوریتم چرخه آب [۱۹] و ... ارائه شده‌اند که از شیوه‌های مختلفی برای برقراری تعادل میان اکتشاف و بهره‌برداری استفاده کرده‌اند.

۲-۲ تابع برازش

تابع برازش یا برازندگی با توجه به مسئله مورد بررسی در نظر گرفته می‌شود و میزان مطلوبیت راه‌حل‌های مختلف برای یک مسئله را مشخص می‌کند. در جدول ۱ توابع مختلف استفاده شده در کارهای مختلف بیان شده است که در این مقاله مورد استفاده قرار می‌گیرد.

۱-۲ کارهای مبتنی بر الگوریتم بهینه‌سازی ملخ

تحقیقات مختلفی اخیراً با استفاده از الگوریتم بهینه‌سازی ملخ انجام شده که بیشتر آن‌ها مربوط به انتخاب ویژگی است که از جمله می‌توان به موارد زیر اشاره کرد:

عملکرد و کارایی الگوریتم SVM عمدتاً به نوع کرنل و پارامترهای آن بستگی دارد. علاوه بر این، انتخاب زیرمجموعه ویژگی که برای آموزش مدل SVM استفاده می‌شود یک عامل مهم دیگر است که تاثیر عمده‌ای بر دقت طبقه‌بندی دارد. در سال ۲۰۱۸، الجار و همکارانش [۲]، مقاله‌ای با عنوان «انتخاب ویژگی و بهینه‌سازی ماشین بردار پشتیبانی به طور همزمان با استفاده از الگوریتم بهینه‌سازی ملخ» ارائه دادند. در همان سال ابراهیم و همکارانش [۴]، مقاله‌ای با عنوان «یک شیوه بهینه‌سازی ملخ برای انتخاب ویژگی و بهینه‌سازی پارامترهای SVM با استفاده از مجموعه داده‌های واقعی پزشکی» ارائه دادند که از الگوریتم بهینه‌سازی ملخ برای انتخاب ویژگی و بهینه‌سازی پارامترهای SVM بر روی مجموعه داده‌های پزشکی استفاده کرده‌اند.

۱-۲ کارهای مبتنی بر الگوریتم بهینه‌سازی ملخ

انتخاب ویژگی یک کار مرتبط با یادگیری ماشین چالش برانگیز است که با حذف داده‌های بی‌ربط، اضافی و نویزی در حالی که سطح قابل قبولی از دقت طبقه‌بندی را حفظ می‌کند. در سال ۲۰۱۸ مافارجا و همکارانش [۳]، مقاله‌ای با عنوان «پویایی جمعیت تکاملی و الگوریتم بهینه‌سازی ملخ برای مسائل انتخاب ویژگی» ارائه دادند که انتخاب ویژگی را با استفاده از الگوریتم بهینه‌سازی ملخ انجام داده‌اند. در این مقاله، الگوریتم بهینه‌سازی ملخ به عنوان یک استراتژی جست‌وجو برای طراحی یک روش انتخاب ویژگی مبتنی بر پوشش استفاده می‌شود. سینگ و همکارانش [۵]، در سال ۲۰۱۹ مقاله‌ای با عنوان «شیوه مبتنی بر الگوریتم بهینه‌سازی ملخ برای بهینه‌سازی طبقه‌بندی موثر و انتخاب ویژگی‌ها برای طبقه‌بندی سیگنال‌های EEG بیماری صرع» ارائه دادند که از الگوریتم بهینه‌سازی ملخ برای انتخاب ویژگی و بهینه‌سازی پارامترهای پنج الگوریتم یادگیری ماشین استفاده می‌کند. مافارجا و همکارانش [۶]، مقاله‌ای با عنوان «رویکردهای الگوریتم بهینه‌سازی ملخ باینری برای مسئله انتخاب ویژگی» ارائه دادند. در این مقاله ترکیب الگوریتم بهینه‌سازی ملخ باینری با روش‌های مختلف انتخاب ویژگی و مجموعه داده‌های مختلف را بررسی کرده‌اند. ذاکری و همکارانش [۷]، مقاله‌ای با عنوان «شیوه انتخاب ویژگی کارآمد با استفاده از الگوریتم بهینه‌سازی ملخ با ارزش واقعی» ارائه دادند. در این مقاله، یک روش انتخاب ویژگی جدید براساس مدل ریاضی تراکنش میان‌ملخ‌ها در یافتن منابع غذایی به نام GOFS پیشنهاد شده است.

در سال ۲۰۱۷، لوکاسیک و همکارانش مقاله‌ای با عنوان «خوشه‌بندی داده‌ها با الگوریتم بهینه‌سازی ملخ» ارائه دادند که از ترکیب الگوریتم خوشه‌بندی

جدول ۱: توابع برازش

ردیف	نام تابع	فرمول	بازه مقادیر
۱	Sphere	$f(x) = \sum_{i=1}^d x_i^2$	$x_i \in [-5.12, 5.12]$
۲	Rastrigin	$f(x) = 10d + \sum_{i=1}^d [x_i^2 - 10\cos(2\pi x_i)]$	$x_i \in [-5.12, 5.12]$
۳	Rosenbrock	$f(x) = \sum_{i=1}^d [100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2]$	$x_i \in [-2.048, 2.048]$
۴	Ackley	$f(x) = -a \exp\left(-b \sqrt{\frac{1}{d} \sum_{i=1}^d x_i^2}\right) - \exp\left(\frac{1}{d} \sum_{i=1}^d \cos(cx_i)\right) + a + \exp(1)$	$x_i \in [-32.768, 32.768]$
۵	Griewank	$f(x) = \sum_{i=1}^d \frac{x_i^2}{4000} - \prod_{i=1}^d \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$	$x_i \in [-600, 600]$
۶	Levy	$f(x) = \sin^2(3\pi x_1) + (x_1 - 1)^2 [1 + \sin^2(3\pi x_2)] + (x_2 - 1)^2 [1 + \sin^2(3\pi x_3)]$	$x_i \in [-10, 10]$
۷	Booth	$f(x) = (x_1 + 2x_2 - 1)^2 + (2x_1 + x_2 - 1)^2$	$x_i \in [-10, 10]$
۸	Matya	$f(x) = 0.26(x_1^2 + x_2^2) - 0.48x_1x_2$	$x_i \in [-10, 10]$
۹	Zakharov	$f(x) = \sum_{i=1}^d x_i^2 + \left(\sum_{i=1}^d 0.5ix_i\right)^2 + \left(\sum_{i=1}^d 0.5ix_i\right)^4$	$x_i \in [-5, 10]$
۱۰	Sum Square	$f(x) = \sum_{i=1}^d ix_i^2$	$x_i \in [-5.12, 5.12]$

۳- الگوریتم بهینه‌سازی ملخ (GOA)

بنابراین، اگر این رفتار به صورت ریاضی مدل‌سازی شود، می‌توان الگوریتم جدیدی که از طبیعت الهام گرفته طراحی کرد. مدل ریاضیاتی بکار گرفته شده برای شبیه‌سازی رفتار گروهی ملخ‌ها به صورت زیر است:

$$X_i = S_i + G_i + A_i \quad (1-3)$$

که در آن موقعیت ملخ نام است، S_i تعامل اجتماعی، G_i نیروی جاذبه در ملخ نام و A_i حرکت افقی باد است. توجه داشته باشید که برای ایجاد رفتار تصادفی، معادله می‌تواند به صورت $X_i = r_1 S_i + r_2 G_i + r_3 A_i$ نوشته شود که در آن r_1 ، r_2 و r_3 اعداد تصادفی در بازه $[0, 1]$ می‌باشند. تابع S_i که تعامل اجتماعی را تعریف می‌کند به صورت زیر محاسبه می‌شود:

$$S_i = \sum_{j=1, j \neq i}^N s(d_{ij}) d_{ij} \quad (2-3)$$

که در آن d_{ij} فاصله بین ملخ نام و ملخ نام است و به صورت $d_{ij} = |x_j - x_i|$ محاسبه می‌شود، S تابعی است برای تعریف قدرت نیروهای اجتماعی نشان داده می‌شود و $d_{ij} = \frac{x_j - x_i}{d_{ij}}$ یک بردار واحد از ملخ نام به ملخ نام است. تابع S که نیروهای اجتماعی را تعریف می‌کند به صورت زیر محاسبه می‌شود:

$$S(r) = feT^{-r} - e^{-r} \quad (3-3)$$

الگوریتم بهینه‌سازی GOA برای اولین بار در سال ۲۰۱۷ توسط آقایان صامی و میرجلیلی [۱] ارائه شده است. الگوریتم ارائه شده با الهام از رفتار حمله ملخ‌ها در طبیعت برای حل مشکلات بهینه‌سازی بصورت ریاضی مدل‌سازی و ارائه شده است. ملخ‌ها حشره‌های کوچکی هستند، اما به دلیل آسبایی که به محصولات کشاورزی می‌زنند، آفت جدی محسوب می‌شوند. اگرچه ملخ‌ها معمولاً به صورت تکی در طبیعت دیده می‌شوند، اما آن‌ها متعلق به بزرگترین گروه از حیوانات هستند. اندازه گروه ملخ‌ها می‌تواند در مقیاس یک قاره بوده و برای کشاورزان کابوس باشد. جنبه منحصر به فرد دسته ملخ‌ها رفتار گروهی آن‌ها هم در بچگی و هم در بزرگسالی است. میلیون‌ها بچه ملخ مانند لوله‌های غلطان می‌پرند و حرکت می‌کنند و تقریباً همه سبزیجات در مسیرشان را می‌خورند. وقتی بزرگ می‌شوند، در هوا تشکیل گروه می‌دهند و فاصله‌های زیادی را برای مهاجرت طی می‌کنند. ویژگی اصلی گروه ملخ‌ها در مرحله لارو، حرکت آهسته و گام‌های کوچک آن‌ها است. در مقابل، حرکت زیاد و ناگهانی اصلی‌ترین ویژگی ملخ‌های بزرگتر است. جستجو برای غذا، دیگر ویژگی مهم گروه ملخ‌ها است. در مقاله اصلی الگوریتم ملخ ادعا شده که زندگی ملخ‌ها به ذات هر دو ویژگی بهره‌برداری و اکتشاف را دارند. به این صورت که ملخ‌های نابالغ حرکت‌های نرم و پیوسته‌ای دارند و در کنار آن‌ها ملخ‌های بالغ حرکت‌های کاملاً تصادفی و جهش‌دار دارند. بنابراین، به ترتیب نقش بهره‌برداری و اکتشاف را دارند. در نتیجه، مدل‌سازی الگوریتم ملخ‌ها منجر به ایجاد یک الگوریتم قدرتمند و مناسب می‌شود.

۴- طرح پیشنهادی

نتایج تجربی و مدل‌سازی‌ها از رفتار همگرایی الگوریتم‌های بهینه‌سازی نشان می‌دهد که ویژگی بهره‌برداری میل به افزایش سرعت همگرایی دارد در حالی که ویژگی اکتشاف میل به کاهش نرخ همگرایی دارد. از سوی دیگر، اکتشاف خیلی بالا احتمال رسیدن به بهینه سراسری را افزایش می‌دهد ولی کارایی کاهش می‌یابد. در حالی که بهره‌برداری خیلی بالا منجر به دام افتادن الگوریتم در بهینه محلی می‌شود. بنابراین یک مقدار صحیحی از میزان اکتشاف و بهره‌برداری وجود ندارد و بسیاری از الگوریتم‌ها از این دو ویژگی به طور نامطلوبی استفاده می‌کنند. در مرحله اکتشاف سعی می‌شود راه‌حل‌های جدید کشف و شناسایی شوند اما در مرحله بهره‌برداری سعی می‌شود جست‌وجو در فضای راه‌حل فعلی صورت گیرد و در صورت امکان جواب به دست آمده بهینه‌تر گردد. اکتشاف و بهره‌برداری در الگوریتم‌های بهینه‌سازی فراابتکاری در هم تنیده شده‌اند. تنوع بالا در جمعیت سبب بالا رفتن قابلیت اکتشاف الگوریتم فراابتکاری می‌گردد و اگر تنوع در جمعیت از دست برود باعث می‌شود که الگوریتم فراابتکاری دچار همگرایی زود هنگام گردد. در این مقاله از یک ضریب هندسی برای برقراری تعامل بین اکتشاف و بهره‌برداری جهت جلوگیری از همگرایی زود هنگام استفاده می‌شود که در ادامه شرح داده شده است.

بهنگام‌سازی موقعیت ملخ‌ها در طرح پیشنهادی همانند الگوریتم پایه بصورت زیر است:

$$x_i^d = \beta \left(\sum_{j \neq i}^N \beta \frac{Ub_d - lb_d}{2} s(|x_j^d - x_i^d|) \frac{x_j - x_i}{d_{ij}} \right) + T_d^A \quad (1-4)$$

پارامتر β دو بار در معادله بالا استفاده شده است بدلیل زیر:

- ضریب β داخل سیگما ناحیه جاذبه-دافعه و ناحیه خنثی بین ملخ‌ها را کاهش می‌دهد.
 - ضریب β بیرون سیگما تعادل بین اکتشاف و بهره‌برداری را برقرار می‌کند.
- برای برقراری تعادل بین دو ویژگی اکتشاف و بهره‌برداری ضریب بهنگام‌سازی بصورت دنباله هندسی در نظر گرفته شده است، که بصورت زیر محاسبه می‌شود:

$$\beta = \beta_{max} - \left(\frac{it}{MaxIt} \right)^w (\beta_{max} - \beta_{min}) \quad (2-4)$$

که در آن β_{max} حداکثر مقدار و β_{min} حداقل مقدار، w ضریب هندسی، it تکرار فعلی و $MaxIt$ حداکثر تعداد تکرار را نشان می‌دهند. در طرح پیشنهادی بهنگام‌سازی ضریب هندسی مطابق شکل ۱ به گونه‌ای انجام می‌شود که در نیمه ابتدایی روند بهینه‌سازی، تغییرات آن کم باشد که ملخ‌های نابالغ فرصت اکتشاف بیشتری داشته باشند و در نیمه انتهایی روند بهینه‌سازی، تغییرات آن زیاد می‌شود که فرصت بهره‌برداری بیشتر باشد.

برای این منظور بهنگام‌سازی ضریب w به صورت زیر محاسبه می‌شود:

$$w = W_{max} - it \frac{W_{max} - W_{min}}{MaxIt} \quad (3-4)$$

که در آن W_{max} حداکثر و W_{min} حداقل مقدارها، it تکرار فعلی و $MaxIt$ حداکثر تعداد تکرار را نشان می‌دهند.

که در آن f نشان دهنده شدت جذب و l مقیاس طول جذب است. عامل G_i در معادله (۱-۳) به صورت زیر محاسبه می‌شود:

$$G_i = -g e_g \quad (4-3)$$

که در آن g ثابت گرانش و e_g نشان دهنده یک بردار واحد به سمت مرکز زمین است. عامل A_i در معادله (۱-۳) به صورت زیر محاسبه می‌شود:

$$A_i = u e_w \quad (5-3)$$

که در آن u ثابت جسم شناور و e_w بردار واحد در مسیر باد است. ملخ‌های نوزاد هیچ بالی ندارند. در نتیجه، حرکت آن‌ها به شدت وابسته به مسیر باد است. با جایگزین کردن S ، G و A در معادله (۱-۳) این معادله می‌تواند به صورت زیر تعریف شود:

$$X_i = \sum_{j \neq i}^N s(|x_j^d - x_i^d|) \frac{x_j - x_i}{d_{ij}} - g e_g + u e_w \quad (6-3)$$

این مدل ریاضیاتی نمی‌تواند مستقیم برای حل مسائل بهینه‌سازی استفاده شود، بیشتر به این دلیل که ملخ‌ها به سرعت به ناحیه خنثی می‌رسند و یک نقطه مشخص را تحت پوشش قرار نمی‌دهند. نسخه اصلاح شده این معادله به صورت زیر ارائه شده است تا مسائل بهینه‌سازی را حل کند:

$$x_i^d = c \left(\sum_{j \neq i}^N c \frac{Ub_d - lb_d}{2} s(|x_j^d - x_i^d|) \frac{x_j - x_i}{d_{ij}} \right) + T_d^A \quad (7-3)$$

که در آن Ub_d محدوده بالا در بعد d ، lb_d محدوده پایین در بعد d ، c مقدار بعد d در هدف (بهترین راه‌حل بدست آمده) و c ضریب کاهش برای کوچک کردن ناحیه خنثی و ناحیه جاذبه-دافعه است.

معادله ۳-۷ نشان می‌دهد که موقعیت بعدی یک ملخ بر اساس موقعیت فعلی، موقعیت هدف و موقعیت دیگر ملخ‌ها تعریف می‌شود. توجه کنید که اولین عامل در این معادله، موقعیت فعلی ملخ با توجه به دیگر ملخ‌ها است. در واقع، توجه داشته باشید که شرایط تمام ملخ‌ها برای تعیین موقعیت عوامل جست‌وجو حول هدف در نظر گرفته شود.

برای برقراری تعادل بین اکتشاف و بهره‌برداری، پارامتر c لازم است تا میزان تکرار را کاهش دهد. این ضریب بهره‌برداری را افزایش می‌دهد در حالیکه میزان تکرار نیز افزایش می‌یابد. ضریب c که مقدار ناحیه خنثی را نسبت به تعداد تکرار کاهش می‌دهد به صورت زیر محاسبه می‌شود:

$$C = C_{max} - it \frac{C_{max} - C_{min}}{MaxIt} \quad (8-3)$$

که در آن C_{max} حداکثر مقدار و C_{min} حداقل مقدار، it تکرار فعلی و $MaxIt$ حداکثر تعداد تکرار را نشان می‌دهند. پارامتر c دو بار در معادله ۳-۷ استفاده شده است بدلیل زیر:

- ضریب C داخل سیگما ناحیه جاذبه-دافعه و ناحیه خنثی بین ملخ‌ها را کاهش می‌دهد.
- ضریب C بیرون سیگما تعادل بین اکتشاف و بهره‌برداری را برقرار می‌کند.

الگوریتم را می‌توان براساس دقت حصول نتیجه و یا ارزش برازندگی بهترین راه‌حل بدست آمده تعیین کرد و کارایی روش براساس تعداد تکرار یا زمان اجرای روند بهینه‌سازی تا رسیدن به همگرایی بدست می‌آید. طرح پیشنهادی با الگوریتم پایه [۱] براساس توابع برازش مختلف ارزیابی شده است.

۵-۱ تنظیم پارامترها

ارزیابی طرح پیشنهادی بر روی ده تابع برازش مختلف که در جدول ۱ بیان شد و با مقادیر اولیه تعداد جمعیت ۵۰، ملخ، تعداد تکرار ۱۰۰ در نظر گرفته شد. برای اطمینان از نتایج ارزیابی آزمون‌ها ۵۰ بار اجرا شده که مقادیر اولیه در هر اجرا برای الگوریتم پایه و طرح پیشنهادی و در همه توابع برازش یکسان تنظیم شده است. پارامترهای کنترلی نقش مهمی در نتیجه بهینه‌سازی دارد. در این مقاله مقادیر مناسب با سعی و خطا تعیین شده است که برای β_{max} مقدار ۰.۱، β_{min} مقدار ۰.۰۰۰۰۰۱، W_{max} مقدار ۰.۲، W_{min} مقدار ۰.۵، انتخاب شده است.

۵-۲ نتایج حاصل و مقایسه

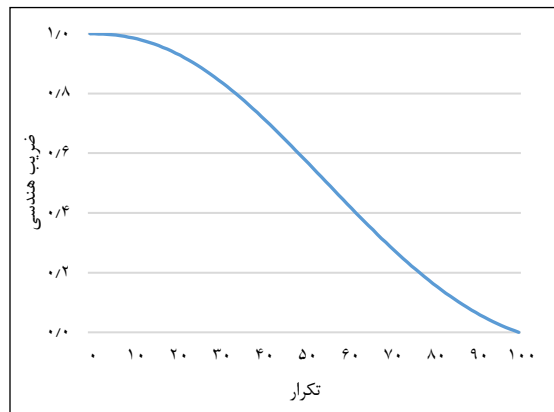
در جدول ۲ الگوریتم پایه و طرح پیشنهادی براساس توابع برازش مختلف و پارامترهای مختلف بررسی شده است. چنانچه ملاحظه می‌شود عملکرد شیوهی پیشنهادی نسبت به شیوه الگوریتم پایه در همه توابع برازش مورد بررسی ده برابر بهبود پیدا کرده است. بهترین حالت در تابع برازش ماتیا رخ می‌دهد که بهترین، بدترین و متوسط برازندگی و نیز انحراف معیار آن‌ها به ترتیب 1.89×10^{-11} ، 1.38×10^{-11} ، 1.45×10^{-11} و 6.47×10^{-11} است.

۶. نتیجه گیری و کار آینده

یکی از چالش‌های الگوریتم بهینه‌سازی ملخ همگرایی زود هنگام آن است که دلیل برقراری تعادل بین دو ویژگی اکتشاف و بهره‌برداری است. در این مقاله یک طرح پیشنهادی برای برقراری تعادل بین دو ویژگی اکتشاف و بهره‌برداری الگوریتم بهینه‌سازی ملخ برای جلوگیری از همگرایی زود هنگام ارائه شد. نتایج ارزیابی نشان می‌دهد که طرح پیشنهادی نتایج بسیار بهتری نسبت به الگوریتم پایه دارد.

این مقاله می‌تواند در دو بخش توسعه الگوریتم بهینه‌سازی ملخ و نیز کاربردهای آن ادامه یابد، که از جمله می‌توان موارد زیر را نام برد:

۱. تابع برازش وفق پذیر به گونه‌ای که در حین روند اجرای بهینه‌سازی پارامترها تنظیم شود.
۲. بکارگیری ایده‌های دیگر الهام گرفته از طبیعت مانند Simulated Annealing، Chaotic و Temporal Neural Network در بهبود الگوریتم بهینه‌سازی ملخ
۳. بکارگیری در دسته‌بندی داده‌های حجیم مانند سیستم‌های تشخیص نفوذ
۴. الگوریتم بهینه‌سازی ملخ کنترل پذیر به گونه‌ای که بسته به اهمیت دقت و سرعت در کاربرد بتوان تعامل بین آن‌ها را برقرار کرد.
۵. پردازش داده‌ها در اینترنت اشیا برای هوشمندسازی مانند بهره‌برداری رفتار ترافیکی شهر هوشمند و مصرف انرژی خانه هوشمند



شکل ۱: نمودار روند تغییر ضریب هندسی طرح پیشنهادی

الگوریتم بهینه‌سازی ملخ یک الگوریتم فراابتکاری هوش جمعی است که یکی از مهم‌ترین الگوریتم‌های فراابتکاری برای حل مسائل پیچیده می‌باشند. در این نوع از الگوریتم‌ها یک عضو جمعیت در تعامل و همکاری با سایر اعضا می‌تواند کارایی لازم را داشته باشد و از طرفی همکاری بین افراد جمعیت برای یافتن جواب بهینه یکی از ویژگی‌های مهم این الگوریتم‌ها به شمار می‌رود. در الگوریتم ۱ مراحل الگوریتم پیشنهادی بهینه‌سازی ملخ بیان شده است.

الگوریتم ۱: الگوریتم پیشنهادی بهینه‌سازی ملخ

۱. تولید تصادفی جمعیت ملخ‌ها در فضای جستجو
۲. تعیین مقدار پارامترهای مسئله از جمله حداقل مقدار ($Cmin$) و حداکثر مقدار ($Cmax$) و حداکثر تعداد تکرار ($MaxIt$)، حداقل مقدار ($Wmin$) و حداکثر مقدار ($Wmax$) و حداکثر تعداد اجرا
۳. محاسبه برازندگی هر ملخ با استفاده تابع برازش
۴. مشخص کردن بهترین ملخ (ملخ با مقدار برازش بهتر) در متغیر T
۵. بهنگام‌سازی پارامترهای کنترلی با استفاده از معادلات (۲-۴) و (۳-۴).
۶. نرمال‌سازی فاصله ملخ‌ها در بازه [۱،۴]
۷. بروزسانی مکان هر ملخ با استفاده از معادله (۱-۴).
- در صورتی که ملخ از فضای جستجو خارج شده به فضای جستجو برگردانده شود.
۸. بروزسانی T در صورتی که ملخی با برازندگی بهتر پیدا شود.
۹. اضافه نمودن شماره تکرار داخلی الگوریتم ($I=I+1$)
۱۰. تا وقتی که به شرط خاتمه نرسیده است ($I < MaxIt$) برو به گام ۵.
۱۱. برگرداندن بهترین ملخ بعنوان جواب نهایی.

۵. ارزیابی و مقایسه نتایج

الگوریتم‌های بهینه‌سازی با تلاش برای بهبود راه‌حل‌ها در هر تکرار به دنبال همگرایی به سمت راه‌حل بهینه می‌باشند که پس از چند تکرار، جستجو به ناحیه‌ای بهینه از فضای تصمیم‌گیری امکان پذیر می‌رسند. میزان همگرایی یک

جدول ۲: مقایسه نتایج الگوریتم بهینه‌سازی ملخ با طرح پیشنهادی

طرح پیشنهادی			الگوریتم پایه				پارامترها	
انحراف معیار	بدترین جواب	میانگین جوابها	بهترین جواب	انحراف معیار	بدترین جواب	میانگین جوابها		بهترین جواب
۱,۸۹۹۱۲E-۱۱	۸,۰۲۰۷E-۱۱	۲,۷۰۱۷۴E-۱۱	۶,۷۴۱۸۵E-۱۳	۲,۷۷۱۴۸E-۱۰	۱,۰۰۲۹۳E-۰۹	۳,۲۷۱۸E-۱۰	۲,۹۰۰۳۶E-۱۲	Sphere
۰,۴۶۰۵۷۶۵۴۶	۰,۹۹۴۹۵۰۷۶	۰,۲۹۸۴۸۷۷۲۳	۳,۸۹۸۳۹E-۱۱	۰,۴۸۲۴۲۹۰۰۳	۰,۹۹۴۹۵۹۳۳۹	۰,۳۵۸۱۸۵۳۳۶	۳,۰۹۹۵۶E-۰۹	Rastrigin
۰,۰۴۹۷۵۵۲۶۴	۰,۲۲۳۵۱۵۵۸۱	۰,۰۱۹۳۴۸۳۸۱	۱,۱۸۷۲۳E-۱۲	۰,۰۳۸۳۲۰۷۱۴	۰,۱۹۷۰۵۴۹۶	۰,۰۰۹۳۸۰۹۷۱	۱,۹۶۱۴۲E-۱۲	Rosenbrock
۴,۲۸۹۰۷E-۰۵	۰,۰۰۰۲۰۲۴۲۸	۸,۷۱۲E-۰۵	۵,۰۴۲۹۶E-۰۷	۰,۰۰۰۱۱۶۵۹۴	۰,۰۰۰۵۵۰۷۲	۰,۰۰۰۲۹۹۸۳۹	۸,۳۶۶E-۰۵	Ackley
۰,۰۱۱۲۰۳۰۵۶	۰,۰۵۹۱۷۸۱۷۷	۰,۰۱۱۴۰۵۷۳	۲,۵۶۲۷۲E-۰۸	۰,۰۱۰۰۵۰۶۷۶	۰,۰۴۶۸۳۵۰۷۶	۰,۰۱۰۹۹۶۸۴۴	۱,۲۴۵۲۴E-۰۷	Griewank
۲,۲۲۱۲E-۱۱	۹,۷۱۲۷۸E-۱۱	۲,۰۱۴۴E-۱۱	۱,۵۶۱۷۹E-۱۳	۲,۷۸۶۱۶E-۱۰	۱,۲۴۴۹۵E-۰۹	۲,۶۷۶۰۱E-۱۰	۴,۶۵۰۰۱E-۱۲	Levy
۳,۰۰۹۵۲E-۱۰	۱,۴۶۲۸۸E-۰۹	۳,۱۷۸۸۵E-۱۰	۱,۸۰۶۴E-۱۳	۸,۹۱۰۹۳E-۰۹	۶,۲۶۸۹۹E-۰۸	۴,۳۲۹۷۱E-۰۹	۱,۰۷۷۵۴E-۱۰	Booth
۱,۴۶۴۹۹E-۱۱	۶,۴۷۲۰۶E-۱۱	۱,۳۸۰۴۶E-۱۱	۱,۸۹۱۹۵E-۱۳	۲,۶۰۱۲۲E-۱۰	۱,۶۲۶۲E-۰۹	۱,۶۷۹۶۱E-۱۰	۲,۶۲۱۲۹E-۱۲	Matya
۵,۵۵۷۲۸E-۱۱	۲,۹۹۸۴۸E-۱۰	۵,۳۷۳۴۳E-۱۱	۴,۷۹۷۷۲E-۱۲	۶,۳۳۹۳۶E-۱۰	۲,۵۰۰۲۶E-۰۹	۶,۲۹۲۰۸E-۱۰	۲,۶۳۳۶۵E-۱۴	Zakharov
۱,۷۷۳۱E-۱۰	۷,۹۵۵۶۵E-۱۰	۱,۸۵۹۶۹E-۱۰	۵,۷۳۹۴۸E-۱۳	۱,۸۲۷۴۴E-۰۹	۹,۶۷۱۶۹E-۰۹	۱,۹۴۷۳۴E-۰۹	۱,۶۳۸۵۵E-۱۱	Sum Square

Piscataway, NJ: Institute of Electrical and Electronics Engineers (IEEE), pp. 1942–1948.

مراجعه:

- [10] Bozorg-Haddad, O., Afshar, A., and Mariño, M. A. (2006). "Honey-bees mating optimization (HBMO) algorithm: A new heuristic approach for water resources optimization." *Water Resources Management*, 20(5), pp. 661–680.
- [11] Mehrabian, A. R. and Lucas, C. (2006). "A novel numerical optimization algorithm inspired from weed colonization." *Ecological Informatics*, 1(4), pp. 355–366.
- [12] Yang, X. S. and Gandomi, A. H. (2012). "Bat algorithm: A novel approach for global engineering optimization." *Engineering Computations*, 29(5), pp.464–483.
- [13] Gambardella, L. M. and Dorigo, M. (1996). "Solving symmetric and asymmetric TSPs by ant colonies." *Proceedings of the 1996 IEEE International Conference on Evolutionary Computation (ICEC'96)*, Nagoya University, Japan, May 20–22, Piscataway, NJ: Institute of Electrical and Electronics Engineers (IEEE), pp.622–627.
- [14] Mahmoud, K. R. (2011). "Central force optimization: Neldermead hybrid algorithm for rectangular microstrip antenna design." *Electromagnetics*, 31(8), pp. 578–592.
- [15] Liu, Y. and Tian, P. (2015). "A multi-start central force optimization for global optimization." *Applied Soft Computing*, 27(C), 92–98.
- [16] Simon, D. (2008). "Biogeography-based optimization." *IEEE Transactions on Evolutionary Computation*, 12(6), pp.702–713.
- [17] Kirkpatrick, S., Gelatt, C. D., & Vecchi, M. P. (1983). *Optimization by simulated annealing*. Science, 220(4598), 671–680.
- [18] Sulaiman, M., Salhi, A., Selamoglu, B. I., and Kirikchi, O. B. (2014). "A plant propagation algorithm for constrained engineering." *Mathematical Problems in Engineering*, 2014, pp. 1–10.
- [19] Eskandar, H., Sadollah, A., Bahreininejad, A., and Hamdi, M. (2012). "Water cycle algorithm-a novel metaheuristic optimization method for solving constrained engineering optimization problems." *Computer and Structures*, 110–111, pp. 151–166.
- [1] Saremi, S., Mirjalili, S., & Lewis, A. (2017). Grasshopper optimization algorithm: theory and application. *Advances in Engineering Software*, 105(2017), pp. 30-47.
- [2] Aljarah, I., Ala'M, A. Z., Faris, H., Hassonah, M. A., Mirjalili, S., & Saadeh, H. (2018). Simultaneous feature selection and support vector machine optimization using the grasshopper optimization algorithm. *Cognitive Computation*, 10(3), pp. 478-495.
- [3] Mafarja, M., Aljarah, I., Heidari, A. A., Hammouri, A. I., Faris, H., Ala'M, A. Z., & Mirjalili, S. (2018). Evolutionary population dynamics and grasshopper optimization approaches for feature selection problems. *Knowledge-Based Systems*, 145(2018), pp. 25-45.
- [4] Ibrahim, H. T., Mazher, W. J., Ucan, O. N., & Bayat, O. (2018). A grasshopper optimizer approach for feature selection and optimizing SVM parameters utilizing real biomedical data sets. *Neural Computing and Applications*, pp. 1-10.
- [5] Singh, G., Singh, B., & Kaur, M. (2019). Grasshopper optimization algorithm-based approach for the optimization of ensemble classifier and feature selection to classify epileptic EEG signals. *Medical & biological engineering & computing*, 57(6), pp. 1323-1339.
- [6] Mafarja, M., Aljarah, I., Faris, H., Hammouri, A. I., Ala'M, A. Z., & Mirjalili, S. (2019). Binary grasshopper optimisation algorithm approaches for feature selection problems. *Expert Systems with Applications*, 117(2019), pp. 267-286.
- [7] Zakeri, A., & Hokmabadi, A. (2019). Efficient feature selection method using real-valued grasshopper optimization algorithm. *Expert Systems with Applications*, 119(2019), pp. 61-72.
- [8] Łukasik, S., Kowalski, P. A., Charytanowicz, M., & Kulczycki, P. (2017, September). Data clustering with grasshopper optimization algorithm. In *2017 Federated Conference on Computer Science and Information Systems (FedCSIS)*. 11(2017). pp. 71-74.
- [9] Kennedy, J. and Eberhart, R. (1995). "Particle swarm optimization." *Proceeding of International Conference on Neural Networks*, Perth, Australia, November 27 to December 1,