

Sonar Dataset Classification using Multi-Layer Perceptron Neural Network Based on Dragonfly and Moth Algorithms

* F. Moosavi poor, M. R. Moosavi

* Professor, Faculty of Electrical Engineering, Iran University of Science and Technology, Tehran, Iran

(Received: 15/09/2021, Accepted: 09/08/2022)

ABSTRACT

One of the most complex areas of sonar research is the classification and recognition of the real target from the liar. Multi-layer perceptron neural networks (NNs) are the most popular and fastest classifier in this area. Train of these networks in remarkable in recent years. Classical algorithms for the training of NNs include: recursive methods, gradient descent, and Newton, etc. Some disadvantages of these methods are improper accuracy, trapping in local optimum, and low convergence rate. In recent years, metaheuristic algorithms combined for the training of NNs are proposed for dominating these defects. In this paper, two new meta-heuristic algorithms are used based on mimicking from animals (dragonfly and moth) for the training of NNs. Simulated results on Iris and Sejnowski datasets are shown Moth-Flame classification rate is 88% and has 30% improvement rather than old methods.

Keywords: Data classification, Sonar targets, Perceptron neural networks, Meta-heuristic algorithms.

* Corresponding Author Email: M_mosavi@iust.ac.ir

دسته‌بندی دادگان سونار با استفاده از شبکه‌های عصبی تعلیم‌یافته با الگوریتم‌های سنجاک

بهبودیافته و پروانه - شمع

سید فرید موسوی پور^۱، سید محمدرضا موسوی میرکلانی^{۲*}

۱- دانشجوی دکتری، ۲- استاد، دانشکده مهندسی برق، دانشگاه علم و صنعت، تهران، ایران

(دریافت: ۱۴۰۰/۰۶/۲۴، پذیرش: ۱۴۰۰/۱۰/۲۹)

چکیده

یکی از زمینه‌های بسیار پیچیده در تحقیقات حوزه سونار، دسته‌بندی و تشخیص اکوی اهداف واقعی از کاذب در این حوزه می‌باشد. شبکه‌های عصبی پرسپترون چندلایه ($MLP\ NN^1$) پرکاربردترین و سریع‌ترین دسته‌بندی‌کننده‌ها در این حوزه بوده و آموزش آنان در سال‌های اخیر بسیار مورد توجه قرار گرفته است. الگوریتم‌های سنتی آموزش شبکه‌های عصبی، شامل روش‌های بازگشتی، گرادیان نزولی و نیوتن و... دارای نقص‌های عدیده‌ای همچون دقت نامناسب، گیر افتادن در کمینه‌های محلی و سرعت همگرایی پایین است. اخیراً استفاده ترکیبی از روش‌های فراابتکاری تعلیم شبکه‌های عصبی به منظور غلبه بر این نقایص پیشنهاد شده‌اند. در این مقاله، از دو الگوریتم جدید فراابتکاری بر پایه تقلید از حیوانات سنجاک و پروانه - شمع به منظور بهبود در آموزش شبکه‌های عصبی استفاده شده است. نتایج شبیه‌سازی روی داده‌های پایگاه داده‌های سجنوفسکی و Iris نشان می‌دهد که شبکه عصبی آموزش یافته شده با روش پروانه - شمع، توانسته است دقت دسته‌بندی اهداف واقعی سوناری را تا ۸۸ درصد افزایش دهد که ۳۰ درصد از الگوریتم‌های قبلی تعلیم بیشتر است.

کلیدواژه‌ها: دسته‌بندی دادگان، اهداف سوناری، شبکه عصبی پرسپترون چندلایه، الگوریتم‌های فراابتکاری

۱- مقدمه

تشخیص آن سخت نیست. در صورتی که سیگنال هدف اصلی، دارای دامنه متغیر با توجه به نوع هدف می‌باشد.

بستر همه دریاها یکسان نبوده و بسته به نوع جنس بستر، اکوهای برگشتی می‌تواند ویژگی‌هایی داشته باشد که به ویژگی‌های هدف بسیار نزدیک باشد. به اکوهای سوناری که از این نوع بوده، ولی هدف واقعی نمی‌باشند، اکوی بدون هدف می‌گویند [۳].

تفکیک اکوی هدف اصلی از کاذب کار ساده‌ای نیست و با روش‌های ساده سنتی مثل عبور از صفر، انرژی سیگنال، اندازه حرکت [۴] و... نمی‌توان آن‌ها را از هم تفکیک نمود.

دریا به صورت کلی یک محیط متغیر با زمان است و ویژگی‌های اکوی سونار در طول زمان (شب و روز) یکسان نمی‌باشد. بنابراین، باید به دنبال ویژگی‌هایی از سیگنال اکو باشیم که کمترین وابستگی به این پارامترها و همچنین شوری، ویسکوزیته و... آب دریا داشته باشد.

جهت مستقل نمودن مسئله از این پارامترهای متغیر، باید از سامانه‌ای استفاده نمود که دسته‌بندی‌کننده آن تطبیق‌پذیری بالا و وابستگی حداقلی به این پارامترهای تداخلی داشته باشد. شبکه عصبی مصنوعی [۵] از نوع پرسپترون چندلایه برای دسته‌بندی

در سال‌های اخیر اهمیت دریاها و وجود منابع در زیر بستر آن‌ها توجه زیادی را در پدافند سایبری جلب کرده است. یکی از پیش‌نیازهای پدافند اطلاعاتی در حوزه دریا، دادگان زیرسطحی می‌باشد. مهم‌ترین ابزار در فن‌آوری زیرسطحی، سونار [۱] بوده که توانایی کشف و شناسایی و هدف‌گیری را بر عهده دارد. این فن‌آوری هم قابلیت عملیات در زیر سطح و هم روی سطح آب را دارد.

فن‌آوری سونار به صورت کلی چهار هدف را باید از هم تشخیص دهد که عبارت‌اند از: اکو بدون هدف، اکوی هدف، نویز و طنین. نویزها در سونار انواع مختلفی دارند و مشخصه آن‌ها این است که از جنس سیگنال پینگ ارسالی توسط ژنراتور سونار نمی‌باشند. به علت تفاوت منشأ و ذات آن با اکوی بازگشتی، تفکیک آن از سه نوع دیگر آسان است. انواع نویز شامل نویز حرارتی، محیطی، قاره‌ای و غیره است.

طنین [۲] حاصل برخورد پینگ ارسالی سیگنال سونار با کف و سطح دریا است. ذات و منشأ سیگنال طنین با سیگنال هدف یکسان است، اما چون طنین، دامنه یکسان و همگن دارد،

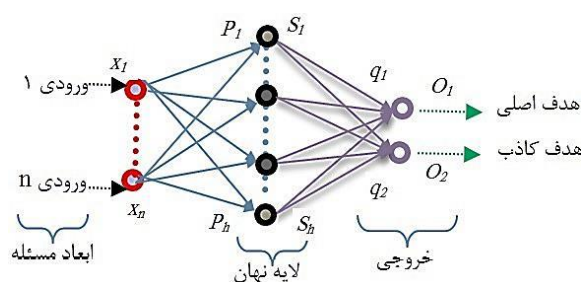
* رایانامه نویسنده مسئول: M_mosavi@iust.ac.ir

¹ Multi-Layer Perceptron Neural Network

سوم، الگوریتم‌های آموزش شبکه یعنی الگوریتم‌های سنجاقک و پروانه- شمع به صورت پایه شرح داده خواهند شد و در بخش چهارم، چگونگی تعلیم شبکه‌های عصبی مبتنی بر الگوریتم‌های بهینه‌سازی بیان می‌شود. در این بخش تغییراتی که باید در الگوریتم‌های اصلی برای بهبود هزینه محاسباتی صورت گیرد، ارائه می‌گردد. در بخش پنجم، نتایج شبیه‌سازی‌ها و در بخش ششم، نتیجه‌گیری شرح داده می‌شود.

۲- شبکه پرسپترون چندلایه

شبکه‌های پرسپترون پرکاربردترین نوع شبکه عصبی در حل مسائل بوده که در شکل (۱) نشان داده شده است.



شکل (۱). ساختار شبکه عصبی پرسپترون چندلایه

در شکل (۱) تعداد لایه‌ها بدون احتساب ورودی، دو لایه در نظر گرفته شده است که یک لایه نهان و یک لایه خروجی می‌باشد.

شبکه عصبی MLP یک شبکه غیرخطی بوده که تحریک نرون‌های لایه نهان مطابق رابطه (۱) به دست می‌آیند.

$$P_j = \sum_{i=1}^n x_i * w_{ji} + b_j \quad j = 1, 2, \dots, h. \quad (1)$$

که در آن، w_{ji} وزن بین ورودی i و لایه نهان j بوده و b_j مقدار بایاس نرون نهان j است.

$$S_j = F(P_j) \quad j = 1, 2, \dots, h. \quad (2)$$

خروجی هر نرون نهان با استفاده از تابع F که یک تابع سیگموئید است، محاسبه می‌شود.

لایه خروجی در این مسئله، با توجه به نوع پایگاه داده می‌تواند دو یا سه نرون باشد و به صورت رابطه (۳) محاسبه می‌شوند:

$$q_m = \sum_{j=1}^h S_j * w_{mj} + B_m \quad m = 1, 2, \dots, h \quad (3)$$

مقدار نهایی خروجی‌ها نیز با استفاده از تابع G که یک تابع غیرخطی است، به دست می‌آید:

$$O_k = G(q_k) \quad k = 1, 2, \dots, m.$$

اهداف واقعی از کاذب پیشنهاد مناسبی است. این شبکه می‌تواند با یک پایگاه داده مرجع آموزش دیده و در مانورهای مختلف آزموده شود و در عمل، بسیار عملکرد مناسبی داشته باشد.

یادگیری در شبکه‌های عصبی پرسپترون، از نوع نظارت‌شده است که به سادگی قابل پیاده‌سازی می‌باشد. برای تعلیم شبکه، روش‌های سنتی مثل گرادینان نزولی^۱ [۶] یا پس انتشار^۲ از قدیم به کار برده می‌شدند که دارای ضعف‌های عمده مثل همگرایی کند و گیر افتادن در کمینه‌های محلی بود. اما با ظهور الگوریتم‌های فراابتکاری مثل ژنتیک^۳ [۷] و انتشار ذرات^۴ [۸] و جستجوی گرانشی^۵ [۹]، انقلابی در این حوزه اتفاق افتاد.

در سال‌های اخیر روش‌های فراابتکاری^۶ [۱۰] اولین انتخاب برای به دست آوردن پاسخ بهینه در مسائل مهندسی واقعی است. این الگوریتم‌های از چند جنبه برتری دارند که مهم‌ترین آن استفاده از عملگرهای تصادفی می‌باشد. این عملگرها سبب می‌شوند که رویه‌های متمایزی نسبت به روش‌های قطعی^۷ سنتی ایجاد شود. یک روش قطعی سنتی برای یک مسئله مشخص تنها یک پاسخ با یک روش رسیدن دارد. در صورتی که رفتار روش‌های فراابتکاری متفاوت بوده و می‌توانند از مسیرهای مختلف به چند جواب بهینه برسند.

مزیت دیگر الگوریتم‌های فراابتکاری جلوگیری از گیر کردن الگوریتم در بهینه‌های محلی^۸ است که به ذات تصادفی بودن آن برمی‌گردد. اگر الگوریتم در یک تکرار زمانی در تله بهینه محلی بیفتد، با استفاده از عملگرهای تصادفی از این محدوده خارج شده و به قسمت‌های دیگر فضای جستجو^۹ می‌رود. در [۱۱] و [۱۲] از دو الگوریتم تکاملی به نام‌های BBO و SFS و بهبود یافته‌های آن برای دسته‌بندی اهداف سوناری تحت شرایط آموزش متفاوت و پایگاه دادگان متفاوت استفاده شده است.

در این مقاله از دو الگوریتم جدید بر پایه تقلید از حیوانات سنجاقک^{۱۰} [۱۳] و پروانه^{۱۱} [۱۴] استفاده شده است تا دسته‌بندی به شکل بهتر و بهینه‌تری انجام شود.

ساختار کلی این مقاله به شرح ادامه است. در بخش دوم، معماری شبکه عصبی پرسپترون چندلایه و تنظیماتی که برای حل این مسئله مورد نیاز است، توضیح داده خواهد شد. در بخش

¹ Gradient Decent

² Back Propagation

³ Genetic Algorithm

⁴ Particle Swarm Optimization

⁵ Gravitational Search Algorithm

⁶ Meta-Heuristic

⁷ Deterministic

⁸ Local Optimum

⁹ Search Space

¹⁰ Dragonfly Algorithm

¹¹ Moth-Flame Optimization

جدول (۱). مقالات و روابط تخمین تعداد نرون لایه نهان

مرجع پیشنهادی	رابطه
[۱۷]	$N_h = N_i - 1$
[۱۸]	$N_h = \sqrt{N_i N_o}$
[۱۹]	$N_h = \log(N_i - 1) - N_o$
[۲۰]	$N_h = \frac{4N_i^2 + 3}{N_i^2 - 8}$

برای سهولت در محاسبات، مطابق مرجع [۱۸] تعداد نرون‌های نهان را ۱۱ در نظر می‌گیریم.

۳- الگوریتم‌های تکاملی

در این قسمت دو الگوریتم سنجاکک و پروانه - شمع شرح داده شده و نکات مهم این الگوریتم‌ها شامل مزایا و معایب آن‌ها نیز توصیف می‌شود.

۳-۱- الگوریتم سنجاکک

سنجاکک‌ها یکی از شناخته‌شده‌ترین حشرات هستند که در همه مناطق دنیا یافت می‌شوند. این حشره دو مرحله زندگی دارد. مرحله اول، لارو و مرحله دوم، بلوغ است. در مورد نوع انتشار (حرکت) دسته‌جمعی سنجاکک‌ها در طبیعت توجه زیادی صورت گرفته است. این انتشار شامل دو حالت کلی است: شکار و مهاجرت.

به حالت شکار، انتشار ایستا و به حالت مهاجرت انتشار پویا نیز می‌گویند. در انتشار ایستا، سنجاکک‌ها به گروه‌های کوچکی تقسیم شده و در محدوده کوچکی به عقب و جلو حرکت می‌کنند. هدف از این عملیات شکار طعمه‌ها است. حرکات مقطعی و تغییرات تصادفی در مسیر پرواز مهم‌ترین ویژگی انتشار ایستا است. در انتشار دینامیک تعداد بسیار زیادی از سنجاکک‌ها برای مهاجرت در یک جهت، مسیری طولانی را طی می‌کنند.

این دو نوع حرکت را به دو فاز الگوریتم‌های فراابتکاری یعنی اکتشاف^۱ و بهره‌برداری^۲ نسبت می‌دهیم. سنجاکک‌ها یک زیرانتشار ایجاد کرده و در مناطق مختلف به صورت ایستا حرکت می‌کنند. این هدف اصلی فاز اکتشاف است. در انتشار دینامیک سنجاکک‌ها در گروه‌های بزرگ‌تر و در یک جهت پرواز می‌کنند که این مشابه فاز بهره‌برداری است.

در مدل‌سازی حرکات سنجاکک‌ها پنج عملگر کلی داریم:

✓ جدا شدن: هر سنجاکک از سنجاکک‌های دیگر در همسایگی‌اش فاصله می‌گیرد.

مقدار مطلوب هر هدف اصلی یک و هدف کاذب صفر است. برای ارزیابی خطا، باید مقادیر خروجی به دست آمده از شبکه را از مقادیر خروجی مطلوب که در پایگاه داده ثبت شده است کم کرده و میزان خطا را به دست آوریم.

به این میزان خطا، شایستگی می‌گویند که مقدار ایده‌آل آن باید صفر باشد.

$$E_k = \sum_{i=1}^m (O_i^k - d_i^k)^2 \quad (4)$$

فرآیند یادگیری به منظور به‌روزرسانی وزن‌ها در زمان پسین به صورت رابطه (۵) به دست می‌آید:

$$w_i(n+1) = w_i(n) - \eta \frac{\partial E_k(n)}{\partial w_i(n)} \quad (5)$$

که در آن، η ضریب یادگیری $w_i(n+1)$ بوده و n زمان فعلی می‌باشد. به رابطه (۵)، معادله یادگیری نیز می‌گویند و مشابه همان نیز برای بایاس‌ها اعمال می‌شود.

معمولاً از سه روش می‌توان الگوریتم‌های تکاملی را برای آموزش شبکه‌های عصبی استفاده نمود. اولین روش استفاده از الگوریتم‌های تکاملی برای پیدا کردن ترکیب وزن‌ها و بایاس‌ها است، به صورتی که حداقل خطا برای یک شبکه به دست آید. دومین روش پیدا کردن ساختار مناسب برای یک شبکه عصبی در مسئله خاص است، به صورتی که تعداد لایه و تعداد نرون‌های نهان مشخص شود. روش آخر استفاده از روش‌های تکاملی برای تنظیم پارامترهای الگوریتم آموزش بر پایه شیب است که مواردی مثل نرخ آموزش و اندازه حرکت را تنظیم می‌کند.

در این مقاله، راه‌حل اول را انتخاب می‌کنیم، زیرا ساختار شبکه ثابت بوده و وظیفه الگوریتم، یادگیری محدود به پیدا کردن وزن‌ها و بایاس‌های شبکه است. هزینه محاسباتی این روش مشخص بوده و به دلیل عدم تغییر ساختار شبکه تغییر نمی‌کند. در روش دوم به دلیل تغییرات زیاد ساختار شبکه، برای مسئله دسته‌بندی کاربردی نبوده و عملاً تلف هزینه محاسباتی است.

پایگاه داده‌های موردنظر سجنوفسکی [۱۵ و ۱۶] و Iris هستند. در پایگاه داده سجنوفسکی، تعداد ابعاد مسئله ۶۰ می‌باشد که همان تعداد ورودی‌های شبکه عصبی است و در خروجی به دو نرون هدف اصلی و کاذب می‌رسیم. به علت آنکه تعداد ابعاد مسئله زیاد است، باید یک مقدار منطقی برای تعداد لایه‌های نهان پیدا نمود. در این بین تعیین تعداد نرون‌های لایه پنهان یک مسئله تجربی - علمی است. در جدول (۱) نمونه‌ای از مقالات و روابط پیشنهادی برای تخمین تعداد نرون‌های لایه نهان آمده است.

¹ Exploration

² Exploitation

در انتشار ایستا، جهت گیری خیلی کم بوده و همگرایی برای حمله به طعمه زیاد است. سنجاکک‌ها با جهت گیری بالا و همگرایی کم، مشغول اکتشاف در فضای جستجو بوده و اگر جهت دهی کم و همگرایی زیاد باشد، مشغول بهره برداری هستند. برای گذر از فاز جستجو به بهره برداری و برعکس باید شعاع همسایه‌ها را نسبت به تعداد تکرارهای زمانی افزایش دهیم.

در شبیه سازی، شعاع را به صورت تجربی از رابطه (۱۳) به دست می آوریم.

$$r_d = \frac{UB_d - LB_d}{10} \quad (13)$$

که در آن، UB کران بالا و LB کران پایین فضای جستجو در بعد d هستند. ضمانت همگرایی الگوریتم در این است که تغییر وزن عملگرها به صورت تطبیقی در طول زمان صورت می گیرد. به صورتی که این وزن‌ها تأثیر مستقیم در اکتشاف و بهره برداری در فضای جستجو دارند.

سنجاکک اصلی تمایل دارد که سنجاکک‌های بیشتری در همسایگی آن باشد تا مسیر پرواز را به این وسیله تنظیم کند. به عبارت دیگر، حیطة نفوذ همسایه‌ها باید افزایش یابد تا آنجا که در مرحله نهایی همه سنجاکک‌ها به یک دسته تبدیل شده و به مقدار بهینه جامع همگرا شوند.

منبع غذایی و دشمن از بهترین و بدترین راه حل در هر تکرار زمانی به دست می آید. این ایده سبب می شود که سنجاکک‌ها به بهترین راه حل نزدیک و از بدترین راه حل دور شوند.

برای بهبود تصادفی بودن از یک تابع گام‌های تصادفی، به نام لووی^۱ استفاده می کنیم. در این مورد، موقعیت سنجاکک با استفاده از تابع رابطه (۱۴) به روزرسانی می شود.

$$X_{t+1} = X_t + \text{levy}(d) \times X_t \quad (14)$$

که در آن، d بعد مسئله بوده و t زمان فعلی است. خود تابع لووی به صورت رابطه (۱۵) به دست می آید:

$$\text{levy}(x) = 0.01 \times \frac{r_1 \times \sigma}{|r_2|^{\frac{1}{2}}} \quad (15)$$

که منظور از $\Gamma(x)$ مقدار $\Gamma(x-1)!$ است.

۳-۱-۱- الگوریتم سنجاکک بهبود یافته

$$\sigma = \left(\frac{\Gamma(1+\beta) \times \sin\left(\frac{\pi\beta}{2}\right)}{\Gamma\left(\frac{1+\beta}{2}\right) \times \beta \times 2^{\left(\frac{\beta-1}{2}\right)}} \right)^{1/\beta} \quad (16)$$

✓ جهت گیری: نشان دهنده تطبیق سرعت هر سنجاکک با دیگر سنجاکک‌ها در همسایگی اش است.

✓ همگرایی: گرایش اجزا به مرکز ثقل کل گروه را می نامند.

✓ هر انتشاری نیازمند بقا است و بنابراین اجزای آن باید به سمت منابع غذایی جذب شوند و سمت دیگر از دشمنان فاصله بگیرند. پس عملگرهای ۴ و ۵، غذا و دشمن هستند.

مدل سازی هر عملگر به صورت زیر نشان داده شده است:

عملگر جدا شدن:

$$S_i = - \sum_{j=1}^N X - X_j \quad (6)$$

که در آن، X_j نشان دهنده موقعیت همسایه j ام است.

عملگر جهت گیری:

$$A_i = \frac{\sum_{j=1}^N \Delta X_j}{N} \quad (7)$$

که در آن، ΔX_j مبین سرعت همسایه j ام است.

عملگر همگرایی:

$$C_i = \frac{\sum_{j=1}^N X_j}{N} - X \quad (8)$$

عملگر غذا:

$$F_i = X^+ - X \quad (9)$$

که در آن، X^+ نشان دهنده موقعیت منبع غذا است.

عملگر دشمن:

$$E_i = X^- + X \quad (10)$$

که در آن، X^- مبین موقعیت دشمن است.

رفتار موقعیت سنجاکک‌ها شامل ترکیب این پنج الگو است. برای به روزرسانی موقعیت در فضای جستجو و شبیه سازی حرکت از دو بردار موقعیت و سرعت استفاده می شود. بردار سرعت از رابطه (۱۱) به دست می آید.

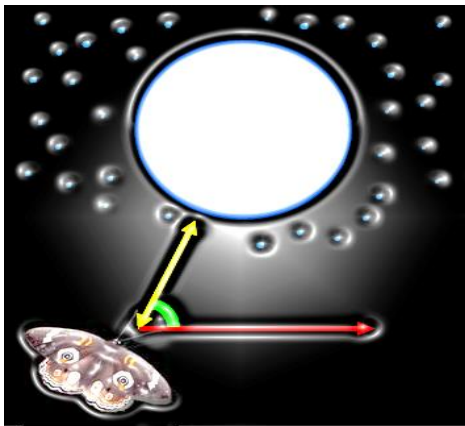
$$\Delta X_{t+1} = (sS_i + aA_i + cC_i + fF_i + eE_i) + w\Delta X_t \quad (11)$$

بردار موقعیت، جهت حرکت سنجاکک‌ها را با استفاده از رابطه (۱۲) نشان می دهد.

$$X_{t+1} = X_t + \Delta X_{t+1} \quad (12)$$

با استفاده از این پنج عملگر رفتارهای اکتشاف و بهره برداری در بهینه سازی به دست می آیند. فرض می کنیم، همسایگان در یک فاصله دویبعدی (دایره) یا سه بعدی (کره) قرار می گیرند. سنجاکک‌ها تمایل دارند که پروازشان را وقتی جهت دهی کنند که جدا شدن و همگرایی در انتشار دینامیک مقدار زیادی است.

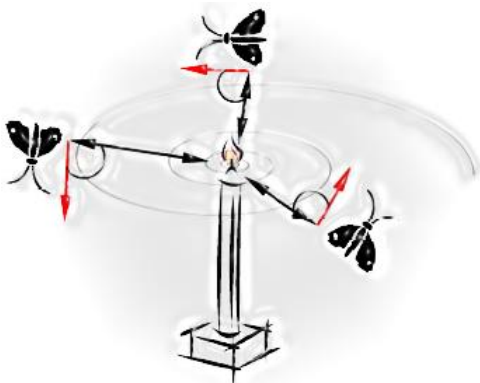
¹ Levy



شکل (۲). نحوه جهت‌یابی پروانه با استفاده از نور ماه.

حال اگر این حالت را برای نورهای مصنوعی در نظر بگیریم، پروانه‌ها آن را با نور ماه اشتباه گرفته و گول می‌خورند. اگر آن نور نزدیک به پروانه باشد، حشره سعی کرده تا نسبت به آن زاویه را حفظ کرده و یک زاویه نسبی نسبت به آن منبع نور ایجاد کرده و مطابق این زاویه حرکت می‌کند. بنابراین، پروانه طی یک حرکت مداری نزدیک شونده، دور نور می‌چرخد. این حرکت، یک حرکت دورانی مداری است. در نهایت، مطابق شکل (۳) پروانه به سمت یک شعله همگرا می‌شود. این مسیر یک مدار است که با گذر زمان شعاع آن کاهش می‌یابد.

این نوع رفتار پروانه نسبت به شعله در روابط (۱۷) تا (۲۰) مدل می‌شود.



شکل (۳). نحوه حرکت پروانه نسبت به شعله شمع

فرض می‌کنیم که راه‌حل‌های ما پروانه‌ها هستند و متغیر مسئله نیز، موقعیت پروانه‌ها در فضای جستجو است. بر اساس ابعاد متغیرها، حرکت پروانه نیز چندبعدی است. ماتریس M که شامل موقعیت پروانه‌ها است به صورت رابطه (۱۷) تشکیل می‌شود:

$$M = \begin{bmatrix} m_{1,1} & m_{1,2} & \dots & \dots & m_{1,d} \\ m_{2,1} & m_{2,2} & \dots & \dots & m_{2,d} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ m_{n,1} & m_{n,2} & \dots & \dots & m_{n,d} \end{bmatrix} \quad (17)$$

الگوریتم سنجاکک از لحاظ میزان محاسبات سه برابر دیگر روش‌های سنتی فراابتکاری حجم داشت. در مسئله دسته‌بندی دادگان سونار، با کمی ژرف‌نگری مشخص شد که سه ضریب این الگوریتم، در تقریباً تمامی تکرارهای زمانی صفر یا نزدیک به صفر می‌باشند. این سه ضریب شامل: جدا شدن، همگرایی و غذا بود. در مسئله خاص دسته‌بندی، چون حداکثر سه نوع خروجی داریم، صفر شدن ضریب همگرایی یا جدا شدن محتمل است. در شبیه‌سازی نرم‌افزاری، برای خروجی‌ها سه مقدار صفر یا ۱ یا -۱ در نظر گرفته می‌شود. بنابراین، چون تابع شایستگی یک سیگموئید است، خروجی آن تنها به این سه عدد نزدیک می‌شود و از اعداد بسیار بزرگ مثبت یا منفی دور است. در مورد ضریب غذا، این ضریب مقدار بسیار ناچیزی داشت و نشان‌دهنده این بود که موقعیت سنجاکک و منبع غذا نزدیک به هم یا منطبق بر هم است. به عبارت واضح‌تر، الگوریتم در همان تکرارهای اول منبع غذا را پیدا کرده و آن را گم نمی‌کند.

با توجه این دلایل، یک الگوریتم سنجاکک از نوع بهبودیافته تعریف شد که تنها بر اساس ضرایب دشمن و جهت‌گیری، به‌روزرسانی می‌شود و بقیه ضرایب صفر در نظر گرفته می‌شود. از لحاظ تئوری میزان محاسبات پیدا کردن موقعیت، باید کاهش حدود ۵۰ درصدی داشته باشد. در نرم‌افزار شبیه‌سازی Matlab به این دلیل که زمان زیادی برای عملیات شرطی صرف می‌شود، این صرفه‌جویی زمانی در کل الگوریتم به ۸ درصد رسید. عملکرد نرخ دسته‌بندی الگوریتم نیز، به‌صورت غیر محسوس در هر دو پایگاه داده سجنوفسکی و Iris، به ترتیب ۲ و ۴ درصد افزایش پیدا کرد.

الگوریتم سنجاکک بهبودیافته برای کاربردهای دسته‌بندی با تعداد خروجی زیاد مناسب‌تر است و تعداد ابعاد مسئله در دقت نرخ دسته‌بندی آن تأثیر محسوس ندارد. در بحث همگرایی الگوریتم نیز هر دو الگوریتم سنجاکک و نوع بهبودیافته آن مشابه هم عمل می‌کنند. در ادامه همه نتایجی که ارائه خواهند شد بر اساس سنجاکک بهبودیافته است.

۳-۲- الگوریتم پروانه - شمع

یکی از رفتارهای خیلی جالب پروانه، نحوه جهت‌یابی ویژه آن در شب است. پروانه‌ها با استفاده از نور ماه در شب پرواز می‌کنند. آن‌ها از یک مکانیزم مختصات متقاطع، برای جهت‌یابی استفاده می‌نمایند. در این روش، مطابق شکل (۳) پروانه به‌وسیله ثابت نگه داشتن زاویه حرکت خود نسبت به ماه پرواز می‌کند. این مکانیزم برای طی مسافت‌های بسیار طولانی در یک خط سیر مستقیم بسیار مفید است [۲۱ و ۲۲]. با اینکه ماه فاصله زیادی تا پروانه دارد، این مکانیزم، حرکت در خط راست را ضمانت می‌کند.

برخی از نکات قابل توجه این مدل سازی:

- یک پروانه می تواند در همسایگی هر شعله، با تغییر t (ضریب نزدیکی) در هر نقطه ای همگرا شود.
- مقدار کم ضریب نزدیکی، نشان دهنده فاصله کمتر تا شعله است.
- هر چه پروانه به شعله نزدیک تر شود، سرعت به روزرسانی موقعیت زیاد می شود.

برای اینکه فاز بهره برداری تضمین شود، میزان ضریب نزدیکی را متغیر با زمان در نظر گرفته و یک عدد تصادفی بین $[r, 1]$ در نظر می گیریم به صورتی که r یک متغیر خطی است که از ۱- به ۲- می رسد.

برای جلوگیری از گیر کردن الگوریتم در کمینه های محلی، پروانه ها و شعله ها، باید در یک بردار جمع شده و از لحاظ شایستگی مرتب شوند. به تعداد عوامل جستجو با رتبه بالاتر، انتساب به شعله صورت می گیرد. به عبارت واضح تر، اگر یک پروانه شایسته تر از یک شعله باشد، آن پروانه برتر، شعله در نظر گرفته می شود و پروانه ضعیف تر حذف می شود. از سوی دیگر با گذشت زمان باید تعداد شعله ها کاهش یافته و به یک شعله برسد. زیرا همه پروانه ها می خواهند به یک موقعیت در هر بعد همگرا شوند که همان شعله اصلی است. برای تحقق این امر تعداد شعله ها در هر تکرار زمانی با رابطه (۲۰) کاهش می یابد.

$$F_{no} = \text{round} \left(N - l * \frac{N - 1}{T} \right) \quad (19)$$

که در آن، l اعداد تکرار فعلی، N حداکثر تعداد پروانه ها است و T تعداد کل تکرار زمانی است.

یکی از ویژگی های بسیار مهم این الگوریتم، پیچیدگی محاسباتی آن است. این امر به تعداد پروانه ها، تعداد متغیرها، حداکثر تعداد تکرار زمانی و مکانیسم مرتب کردن^۱ شعله ها در هر تکرار زمانی است. مرتب کردن خود می تواند یکی از مهم ترین عوامل بهینه سازی پیچیدگی الگوریتم باشد.

۴- تعلیم شبکه عصبی با الگوریتم تکاملی

جهت تعلیم یک شبکه عصبی از نوع پرسپترون چندلایه، لازم است که الگوریتم های تکاملی، مقادیر بهینه شده وزن یال ها و بایاس نرون ها را به عنوان خروجی به سیستم بدهند.

۴-۱- نمایش تعلیم شبکه عصبی توسط الگوریتم تکاملی

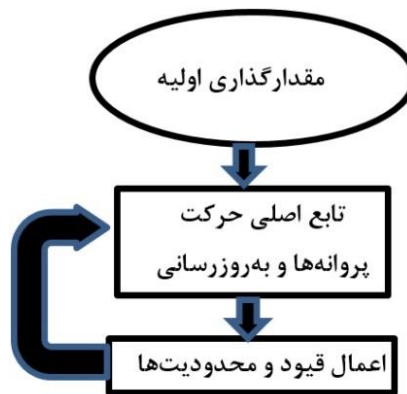
در مسائل مختلف که جهت حل به الگوریتم های تکاملی ارجاع داده می شوند، سه روش ارائه پارامترهای مجهول داریم: نمایش باینری، ماتریس و بردار. در نمایش باینری، هر عنصر به صورت

که در آن، n تعداد پروانه ها و d تعداد ابعاد (متغیرهای مسئله) است. پس از به دست آوردن مقدار موقعیت پروانه ها، از تابع شایستگی برای به دست آوردن شایستگی هر پروانه استفاده می کنیم و در رابطه (۱۸) ماتریس OM تشکیل می شود.

$$OM = \begin{bmatrix} OM_1 \\ OM_2 \\ \vdots \\ OM_n \end{bmatrix} \quad (18)$$

برای شعله ها، مشابه پروانه ها، دو ماتریس F و OF مشابه همان ابعاد تشکیل می شود. نکته قابل توجه این است که هم پروانه ها و هم شعله ها می توانند هر دو راه حل های ما باشند. تفاوت بین آن ها در رفتار و به روزرسانی آن ها در تکرارهای زمانی است. در اصل پروانه ها عوامل جستجوی واقعی مسئله در فضای جستجو هستند، ولی شعله ها بهترین موقعیت هایی هستند که از زمان های قبل به دست آمده اند. هر پروانه به دور شعله می چرخد و اگر مقدار شایستگی مناسب تری به دست آمد، راه حل بهتری پیدا کرده است.

این الگوریتم دارای سه قسمت اصلی است که در شکل (۴) نشان داده شده است.



شکل (۴). فلوجارت الگوریتم پروانه - شمع

تابع مقدار گذاری اولیه یک جمعیت تصادفی از پروانه ها و مقدار شایستگی آن ها تولید می کند. تابع اصلی، حرکت پروانه ها در فضای جستجو را به روزرسانی می نماید. در تابع قیود نیز اگر محدودیت ها و قیود رعایت نشوند، الگوریتم را از چرخه خارج می کند.

برای حرکت پروانه به دور شمع از یک مسیر مداری لگاریتمی به صورت رابطه (۱۹) استفاده می شود.

که در آن، D فاصله اقلیدسی پروانه i ام از شعله زام است. ضریب b برای شکل دهی به مسیر لگاریتمی است و t یک عدد تصادفی بین $[-1]$ و $[+1]$ است.

^۱ Sorting

در شکل (۵)، مقدار مطلوب با استفاده از پایگاه داده مشخص شده و مقدار تخمینی راه‌حل از طریق شبکه عصبی به دست می‌آید. در ادامه، این دو از هم کم شده و مقدار خطا شکل می‌گیرد. این مقدار خطا و مدل‌سازی که الگوریتم فراابتکاری از مسئله داشته است، سبب ایجاد تخمین جدید از پارامترهای شبکه برای شبکه پرسپترون چندلایه شده و دوباره چرخه تکرار می‌شود.

۵- شبیه‌سازی و جزئیات آن

در این قسمت به منظور ارزیابی کارایی الگوریتم‌های سنجاکک بهبودیافته و پروانه - شمع در آموزش شبکه عصبی پرسپترون چندلایه، از الگوریتم‌های PSO و جستجوی گرانشی را هم برای مقایسه عملکرد استفاده می‌کنیم.

جدول (۲). پارامترها و مقادیر اولیه الگوریتم‌های آموزش

الگوریتم	پارامتر	مقدار
GSA	تعداد ذرات	۳۰
	تعداد تکرار زمانی	۵۰۰
PSO	تعداد ذرات	۳۰
	تعداد تکرار زمانی	۵۰۰
	وزن اینرسی اولیه	۰/۲
	C_1	۱
	C_2	۱
پروانه - شمع	تعداد پروانه و شمع اولیه	۳۰
	تکرار زمانی	۵۰۰
	b	۱
سنجاکک بهبودیافته	تعداد سنجاکک‌ها	۳۰
	تکرار زمانی	۵۰۰

پارامترها و مقادیر اولیه هر الگوریتم بر اساس جدول (۲) تنظیم شده است. در ابتدا، دادگان Iris و سجنوفسکی به این الگوریتم‌ها اعمال شده و عملکرد دسته‌بندی کننده‌ها از نظر نرخ دسته‌بندی، اجتناب از گیر کردن در کمینه محلی و سرعت همگرایی آزموده می‌شود.

نرخ دسته‌بندی، میزان دقت الگوریتم تکاملی تعلیم‌دهنده شبکه عصبی را نشان می‌دهد. مقادیر میانگین و P-Value نیز نشان‌دهنده قدرت الگوریتم در مواجهه با بهینه‌های محلی می‌باشد.

هر الگوریتم ۱۰ بار اجرا شده و نرخ دسته‌بندی، میانگین خطا و مقدار P-Value و زمان محاسبه در جداول (۳) و (۴) به ترتیب برای مجموعه دادگان Iris و سجنوفسکی نمایش داده شده است.

مطابق [۲۴] برای سنجش عملکرد الگوریتم‌های فراابتکاری، علاوه بر میانگین و انحراف معیار دقت دسته‌بندی، یک آزمون

رشته‌ای از بیت‌های باینری صفر و یک نمایش داده می‌شوند. هر کدام از این شیوه‌های نمایش، مزایا و معایب خود را دارند که در برخی از مسائل نمود پیدا می‌کنند [۲۳].

به‌صورت کلی اعداد ما به‌صورت باینری نیستند و برای تبدیل آن‌ها باید از ساختارهای رمزنگاری استفاده نمود که هزینه محاسباتی را بالا می‌برند. بنابراین، استفاده از روش باینری برای شبکه‌های عصبی بزرگ و پیچیده به‌صرفه نمی‌باشد. معمولاً روش برداری منطقی‌ترین روش در دسته‌بندی دادگان سوناری به نظر می‌رسد.

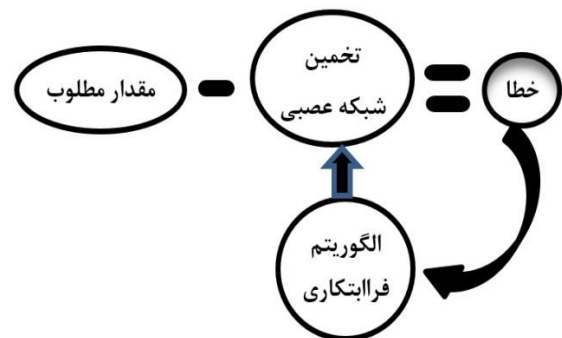
پارامترهایی که باید مقدار بهینه آن‌ها برای تعلیم به دست آید، شامل وزن‌های یال‌ها از ورودی به نرون‌های نهان و وزن یال‌های نرون‌های نهان، به خروجی است. بایاس‌ها نیز، برای نرون‌های نهان و چه خروجی موردنظر هستند. تعداد ورودی‌ها برابر ابعاد داده‌های پایگاه داده است. تعداد نرون‌های نهان نیز اگر از حدی بیشتر باشد، موجب بیش آموزش شبکه شده و همچنین ساختار آن را پیچیده می‌کند. تعداد کم نرون‌های نهان نیز خود مخرب بوده و می‌تواند به واگرایی شبکه بیانجامد.

عوامل جستجوی الگوریتم، شامل بردارهای وزن یال‌های بین لایه ورودی و نهان (W_{hi}) و بایاس‌های نرون‌های نهان (b_h) و نرون‌های خروجی (B_m) است. بنابراین، ورودی تابع اصلی الگوریتم تکاملی به‌صورت رابطه (۲۱) می‌باشد.

$$P_i = [\overrightarrow{W_{hi}} \overrightarrow{W_{mj}} \overrightarrow{b_h} \overrightarrow{B_m}] \quad (20)$$

روش‌های یادگیری یک هدف دارند و آن هم تعلیم شبکه عصبی مصنوعی است. این فرآیند دو قسمت اصلی دارد که یکی آموزش شبکه بوده و دیگری آزمون است. در قسمت آموزش با استفاده از روش‌های تصادفی (اعتبارسنجی متقاطع^۱) ۶۰-۷۰ درصد پایگاه داده به شبکه عصبی و الگوریتم تکاملی ارائه شده و اصطلاحاً شبکه تعلیم می‌بیند. سپس، بقیه پایگاه داده تحت آزمون قرار گرفته و عملکرد شبکه و میزان دقت و همگرایی آن مشخص می‌شود.

در شکل (۵) نمودار تعلیم شبکه عصبی با استفاده از الگوریتم‌های فراابتکاری نشان داده شده است.



شکل (۵). بلوک دیاگرام مدل‌سازی با شبکه پرسپترون چندلایه

^۱ Cross Validation

از دیگر عوامل بسیار مؤثر بر ارزیابی، زمان محاسبات است. دقت دسته‌بندی مؤثرترین عامل تئوری کیفیت سیستم دسته‌بندی سوناری می‌باشد، اما زمان محاسبات از لحاظ عملیاتی بسیار مهم است؛ زیرا روشی که سریعاً پاسخ ندهد از لحاظ عملیاتی قابل استفاده نمی‌باشد. برای محاسبه زمان محاسباتی از نرم‌افزار Matlab و مطابق تنظیمات مطرح شده در جدول (۲) استفاده شده است. کامپیوتر مورد استفاده یک لپ‌تاپ با پردازنده ۵ هسته‌ای Intel با فرکانس ۱/۸ GHz و حافظه رم ۴ GB استفاده شده است.

همان‌طور که از ظاهر جدول (۳) که مربوط به پایگاه داده Iris است، مشاهده می‌شود، روش‌های قدیمی جستجوی گرانشی و PSO از لحاظ کیفی سطح متوسطی داشته و هزینه محاسباتی متوسطی نیز دارند، اما روش سنجاکک بهبودیافته با اینکه تعداد تکرار زمانی به اندازه PSO دارد، زمان محاسبات چهار برابری داشته و از سوی دیگر نیز بهبود دسته‌بندی آن بسیار محسوس نیست. بهترین روش از لحاظ نرخ دسته‌بندی و نیز زمان محاسبات روش پروانه - شمع بوده که هم دقت بسیار بالاتری را داراست و نیز هزینه محاسباتی بسیار کمتری دارد.

جدول (۳). نتایج حاصل از اعمال دسته‌بندی کننده‌های مختلف بر روی دادگان Iris

الگوریتم	نرخ دسته‌بندی (%)	متوسط MSE	P-Value	زمان برحسب ثانیه
جستجوی گرانشی	۵۹	۰/۸۴۰۷	۶/۷۸ E - ۱۱۰	۱۸۱
PSO	۴۷	۰/۳۹۶۵	۱/۲۶ E - ۸۳	۱۸۲
سنجاکک بهبودیافته	۵۸	۰/۴۶۰۶	۵/۸۵ E - ۸۴	۵۸۴
پروانه - شمع	۸۸	۰/۲۶۰۳	۱/۲۶۱ E - ۸۳	۱۵۶

سنجوفسکی ۱۵ برابر شده است و ورودی‌ها نیز ۳۰ درصد افزایش یافته است، اما زمان کلی محاسبات برای ۱۰ بار اجرا تنها ۴ برابر شده است. کمترین متوسط میانگین مربع خطا هم در هر دو پایگاه داده مربوط به الگوریتم پروانه - شمع می‌باشد. این سنججه همراه با نرخ دسته‌بندی، برتری این الگوریتم نسبت به بقیه الگوریتم‌های یادگیری را اثبات می‌کند.

جدول (۴). نتایج حاصل از اعمال دسته‌بندی کننده‌های مختلف بر روی دادگان سنجوفسکی

الگوریتم	نرخ دسته‌بندی (%)	متوسط MSE	P-Value	زمان برحسب ثانیه
جستجوی گرانشی	۶۷	۰/۳۰۷۷	۱/۴۹ E - ۹۷	۵۵۲
PSO	۳۷	۰/۶۲۱۲	۱/۵۶۶ E - ۱۱۰	۴۴۰
سنجاکک بهبودیافته	۵۷	۰/۳۹۶۳	۹/۳۲ E - ۸۴	۱۳۶۴
پروانه - شمع	۸۱	۰/۲۷۰۸	۱/۲۵۹ E - ۸۳	۴۱۹

آماري نیز نیاز است. این آزمون آماری باید نشان دهد که الگوریتم پیشنهادی چقدر نتایج بهتری را نسبت به الگوریتم‌های معیار ارائه می‌دهد و آیا نتایج این الگوریتم به اندازه کافی از دیگر الگوریتم‌های سنجشی متفاوت است یا خیر؟

آزمون Wilcoxon's یک آزمون آماری غیر پارامتری است که خروجی آن به P-Value مشهور است [۲۵].

پایگاه دادگان سنجوفسکی و Iris استفاده شده، از نوع استاندارد بوده و از همه داده‌ها به صورت ترکیب اعتبارسنجی متقاطع استفاده شده است. به این صورت که یک ضریب از تعداد داده‌ها (مثلاً برای سنجوفسکی، ۱۳) را انتخاب کرده و داده‌ها به صورت دسته‌های ۱۳ تایی درهم‌ریخته می‌شوند (ابعاد و خروجی‌ها تغییر نمی‌کنند). این عمل تصادفی بودن اطلاعات ورودی را تضمین می‌کند. از سوی دیگر ابعاد زیاد پایگاه داده سنجوفسکی سبب می‌شود که همگرایی و دقت الگوریتم‌ها بهتر ارزیابی شود.

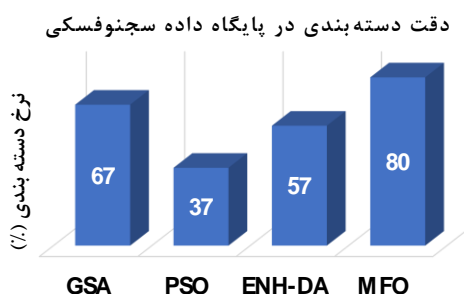
برای هر الگوریتم ۷۰ درصد پایگاه داده‌ها جهت تعلیم و ۳۰ درصد باقیمانده برای آزمون استفاده می‌شود. برخی از پژوهش‌ها از ۱۰۰ درصد پایگاه داده برای تعلیم استفاده کرده‌اند که این کار سبب می‌شود دقت الگوریتم‌ها به هم بسیار نزدیک شده و اختلاف آن‌ها به خوبی آشکار نشود.

جدول (۴) نیز مشابه جدول (۳) مقایسه ۴ سنججه بین الگوریتم‌های یادگیری فراابتکاری در پایگاه داده پیچیده‌تر سنجوفسکی است. مقدار P-Value باید کمتر از ۰/۰۵ باشد تا نتایج از دید آماری قابل قبول باشد. نکته دیگر نرخ دسته‌بندی است که بر اساس درصد بیان شده است و به جز روش جستجوی گرانشی بقیه روش‌ها با پیچیده‌تر شدن پایگاه دادگان افت محسوسی پیدا کرده است. با اینکه تعداد ابعاد در پایگاه

۱-۵- دادگان سونار

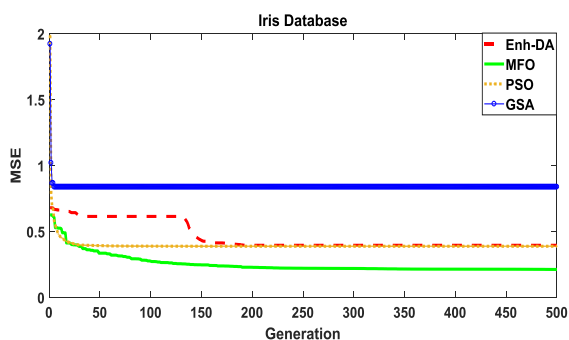
دادگان متفاوت و شرایط آموزش ۸۰ درصدی شبکه عصبی جهت دسته‌بندی اهداف سوناری استفاده شده است و تأکید بر پیاده‌سازی بر روی FPGA بوده است. در این پژوهش سعی بر این است که سخت‌ترین شرایط لحاظ گردد، زیرا در اکثر موارد عملیاتی اطلاعات پایگاه داده مرجع تکمیل نمی‌باشد.

در شکل (۷) نرخ دسته‌بندی روش‌های یادگیری در پایگاه داده سجنوفسکی نشان داده شده است. مقدار نرخ دسته‌بندی نسبت به پایگاه داده Iris، کمی کاهش یافته است و این نشان‌دهنده این مسئله است که با افزایش ابعاد (به‌جز روش جستجوی گرانشی) عملکرد الگوریتم‌ها به‌صورت شدید تضعیف نمی‌شود.



شکل (۷): نرخ دسته‌بندی بر اساس پایگاه داده سجنوفسکی

شکل (۸) نمودار همگرایی در پایگاه داده Iris نشان می‌دهد. سریع‌ترین همگرایی در الگوریتم جستجوی گرانشی رخ داده است، ولی میزان میانگین مربع خطای آن از همه بیشتر است. از سوی دیگر کمترین میانگین مربع خطا متعلق به الگوریتم پروانه شمع است. این الگوریتم اگرچه همگرایی دیرنگامی دارد، اما به دلیل اینکه الگوریتم ذاتاً سریعی است، برای کاربردهای عملیاتی نسبت به سنجاکک بهبودیافته برتری مطلق دارد. الگوریتم‌های PSO و سنجاکک بهبودیافته از لحاظ همگرایی در یک سطح قرار دارند.



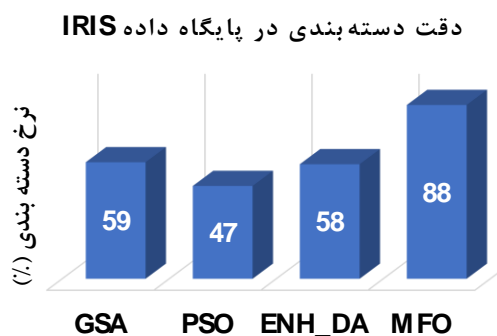
شکل (۸): نمودار همگرایی الگوریتم‌ها در پایگاه داده Iris

در شکل (۹) پایگاه داده سجنوفسکی مدنظر قرار گرفته و ضعیف‌ترین عملکرد برای PSO است. عملکرد جستجوی گرانشی اندکی بهتر از سنجاکک بهبودیافته می‌باشد. فاز اکتشاف در الگوریتم پروانه- شمع در همه تکرارهای زمانی به‌خوبی عمل کرده و به همین دلیل میانگین مربع خطا به‌صورت تدریجی

پایگاه داده سجنوفسکی دارای ۲۰۸ نمونه بوده که هر نمونه شامل ۶۰ باند فرکانسی (بعد مسئله) می‌باشد. این پایگاه داده فقط دو نوع هدف یعنی حقیقی و کاذب را شناسایی می‌کند. هدف واقعی در اصل یک سیلندر فلزی است و هدف کاذب نیز یک صخره سنگی هم‌اندازه سیلندر می‌باشد. طول سیلندر ۵ ft بوده و همراه با صخره در کف ماسه‌ای دریا قرار دارند. پالس ارسالی یک چیرپ FM پهن باند می‌باشد. اکوهای دریافتی از فاصله ۱۰m دارای SNR به مقدار ۴-۱۵ dB می‌باشند. از سوی دیگر پایگاه داده Iris دارای ۱۵۰ نمونه می‌باشد و تعداد ابعاد مسئله آن ۴ می‌باشد، ولی خروجی آن سه نوع هدف شامل حقیقی، کاذب و طنین را شناسایی می‌کند. در ابتدا شبیه‌سازی هر بعد از مسئله نرمال‌سازی می‌شود و سپس به الگوریتم‌های شبکه عصبی اعمال می‌گردد، زیرا اگر مقادیر نرمال‌سازی نشود، شبکه عصبی سریعاً اشباع شده و بی‌اثر می‌گردد.

۲-۵- دسته‌بندی اهداف سوناری

نمودار شکل (۶) متعلق به پایگاه داده Iris است که روش پروانه شمع با دقت ۸۸ درصد برترین روش و پس‌از آن جستجوی گرانشی با ۵۹ درصد و با فاصله کمی هم سنجاکک بهبودیافته قرار دارد. الگوریتم سنجاکک بهبودیافته از لحاظ میزان محاسبات و به‌هیچ‌عنوان قابل قیاس با روش‌های پروانه - شمع و یا PSO نمی‌باشد و حدود سه برابر آن‌ها از لحاظ زمانی هزینه دارد. در جستجوی گرانشی، برعکس سه الگوریتم دیگر هرچه پیچیدگی کمتر شده است، دقت دسته‌بندی آن نیز کمی کاهش یافته است. این رفتار غیرعادی در سیستم‌های واقعی که پیچیدگی‌های زیادی دارند، برای تشخیص هدف بسیار مضر می‌باشد.

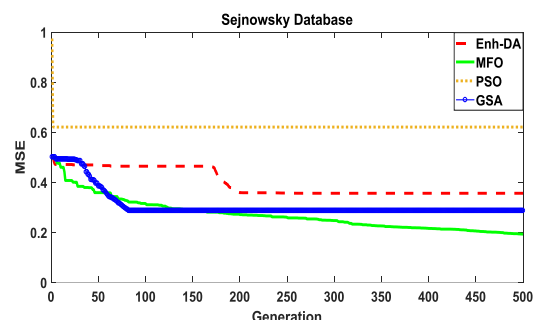


شکل (۶): نرخ دسته‌بندی بر اساس پایگاه داده Iris

روش قدیمی PSO با نرخ دسته‌بندی ۳۷ درصد هیچ ارزش عملیاتی ندارد. در برخی از مقالات، روش‌های قدیمی [۱۲] و [۲۶] تحت شرایط بسیار خاصی (آموزش ۱۰۰ درصد پایگاه داده) با روش‌های جستجوی گرانشی و دیگر الگوریتم‌های نوین مقایسه شده‌اند. در [۱۱] از الگوریتم‌های فراابتکاری بهبودیافته با پایگاه

- [7] H. Holland, "Genetic Algorithms," Scientific American, vol. 267, pp. 66-72, 1992.
- [8] R. C. Eberhart & J. Kennedy, "A New Optimizer using Particle Swarm Theory," International Symposium on Micro Machine and Human Science, pp. 39-43, 1995.
- [9] E. Rashedi, H. Nezamabadi-pour, & S. Saryazdi, "GSA: A Gravitational Search Algorithm," Information Sciences, vol. 179, pp. 2232-2248, 2009.
- [10] Y. Wang, L. Yuan, M. Khishe, A. Moridi, & F. Mohammadzade, "Training RBF NN Using Sine-Cosine Algorithm for Sonar Target Classification," Archive of Acoustics, vol. 45, no. 4, pp. 753-764, 2020.
- [11] M. Khishe, M. R. Mosavi, & A. Moridi, "Chaotic Fractal Walk Trainer for Sonar Dataset Classification using Multi-Layer Perceptron Neural Network and its Hardware Implementation," Applied Acoustics, vol. 137, pp. 121-139, 2018.
- [12] M. Kave, M. Khishe, & M. R. Mosavi, "Design and Implementation of a Neighborhood Search Biogeography based Optimization Trainer for Classifying Sonar Dataset using Multilayer Perceptron Neural Network," Analog Integrated Circuits and Signal Processing, vol. 100, no. 2, pp. 405-428, 2019.
- [13] S. Mirjalili, "Dragonfly Algorithm: A New Meta-heuristic Optimization Technique for Solving Single-Objective, Discrete, and Multi-Objective Problems," Neural Computing and Applications, vol. 27, no. 4, pp. 1053-1073, 2015.
- [14] S. Mirjalili, "Moth-Flame Optimization Algorithm: A Novel Nature-Inspired Heuristic Paradigm," Knowledge-Based Systems, vol. 89, pp. 228-249, 2015.
- [15] R. P. Gorman & T. J. Sejnowski, "Analysis of Hidden Units in a Layered Network Trained to Classify Sonar Targets," Neural Networks, vol. 1, pp. 75-89, 1988.
- [16] <http://archive.ics.uci.edu/ml/datasets>.
- [17] S. Tamura & M. Tateishi, "Capability of Four-Layered Neural Network: Four Layers versus Three," IEEE Trans. Neural Netw. vol. 8, pp. 251-255, 1997.
- [18] K. Shibata & Y. Ikeda, "Effect of Number of Hidden Neurons on Learning in Large-scale Layered Neural Networks," in Proceedings of the ICROS-SICE International Joint Conference, pp. 5008-5013, 2009.
- [19] D. Hunter, Y. Hao, M.S. Pukish, J. Kolbusz, & B.M. Wilamowski, "Selection of Proper Neural Network Sizes and Architectures - A Comparative Study," IEEE Trans. Ind. Inform., vol. 8, pp. 228-240, 2012.

کاهش می‌یابد.



شکل (۹): نمودار همگرایی الگوریتم‌ها در پایگاه داده سجنوفسکی

۶- نتیجه‌گیری

در این مقاله، از دو روش فرابتنکاری جدید به نام‌های سنجاچک بهبودیافته و پروانه - شمع به منظور آموزش شبکه عصبی پرسپترون چندلایه استفاده شده است. برای ارزیابی عملکرد دسته‌بندی الگوریتم‌ها از مجموعه داده‌های سجنوفسکی و Iris استفاده شده است و نتایج حاصله با الگوریتم‌های قدیمی جستجوی گرانشی و PSO مقایسه شده‌اند. معیارهای سنجش علاوه بر دقت دسته‌بندی و سرعت همگرایی، زمان محاسبات نیز می‌باشد که جهت کاربردهای واقعی حائز اهمیت است. نتایج حاصله نشان می‌دهد که با استفاده از دسته‌بندی کننده طراحی شده با پروانه شمع نرخ دسته‌بندی برای مجموعه داده‌های پیچیده‌تر نسبت به دیگر الگوریتم‌های جستجوی گرانشی، PSO و سنجاچک بهبودیافته به ترتیب ۱۳، ۴۳ و ۲۳ درصد افزایش یافته است.

۷- مراجع

- [1] U. Robert, "Principle of Underwater Sound," 3rd Edition, New York, Prentice Hall, 2002.
- [2] M. R. Mosavi, M. Khishe, & M. Aghababaei, "Modeling and Mitigation of Active Sonar Clutter," Noshahr University of Marine Science and Technology, 2015. (In Persian)
- [3] M. Fialkowski & C. Gauss, "Methods for Identifying and Controlling Sonar Clutter," IEEE Journal of Oceanic Engineering, vol. 35, no. 2, pp. 125-138, April 2010.
- [4] C. A. Wilgenbusch, "Detection and Feature Extraction of Mine-like Objects from Sonar Signals," Master's Thesis, Naval Postgraduate School Monterey, 2001.
- [5] J. M. Benitez, J. L. Castro, & I. Requena, "Are Artificial Neural Networks Black Boxes?," IEEE Transaction on Neural Networks, vol. 8, no. 5, pp. 1156-1164, 1997.
- [6] N. Karayiannis, "Reformulated Radial Basis Neural Networks Trained by Gradient Descent," IEEE Transactions on Neural Networks, vol. 10, no. 3, pp. 657-671, 1999.

- [24] S. García, D. Molina, M. Lozano, & F. Herrera, "A Study on the Use of Non-parametric Test for Analyzing the Evolutionary Algorithms' Behaviour," A Case Study on the CEC' 2005 Special Session on Real Parameter Optimization. *Journal of Heuristics*, vol. 15, no. 6, pp. 617-644, 2009.
- [25] S. Mirjalili & A. Lewis, "S-Shaped Versus V-Shaped Transfer Functions for Binary Particle Swarm Optimization," *Swarm and Evolutionary Computation*, vol. 9, pp. 1-14, 2013.
- [26] M. R. Mosavi, M. Khishe, A. Ghamgosar, & M. J. Ghalandari, "The Use of Radial Basis Function Networks Based on Leader Mass Gravitational Search Algorithm for Sonar Dataset Classification," *Journal Of Electronical & Cyber Defence*, vol. 4, no. 2, pp. 39-52, 2016. (In Persian)
- [20] K. Gnana Sheela & S. N. Deepa, "Review on Methods to Fix Number of Hidden Neurons in Neural Networks," *Mathematical Problems in Engineering*, vol. 2013, pp. 1-11, 2013.
- [21] K. J. Gaston, J. Bennie, T. W. Davies, & J. Hopkins, "The Ecological Impacts of Night Time Light Pollution: a Mechanistic Appraisal," *Biological Reviews*, vol. 88, pp. 912-927, 2013.
- [22] K. D. Frank, C. Rich, & T. Longcore, "Effects of Artificial Night Lighting on Moths," *Ecological Consequences of Artificial Night Lighting*, pp. 305-344, 2006.
- [23] M. R. Mosavi, M. Khishe, A. Ghamgosar, & M. J. Ghalandari, "Classification of Sonar Data Set using the Gray Wolf Optimizer Algorithm," *Journal of Electronics Industries*, vol. 7, no. 1, pp. 27-41, 2016. (In Persian)