

ارسال مقاله: ۹۳/۴/۴

پذیرش مقاله: ۹۳/۹/۱۶

تنظیم بهینه و همزمان ساختار و پارامترهای شبکه عصبی با استفاده از الگوریتم آمیختار مبتنی بر جستجوی گرانشی برای کاربردهای دسته‌بندی و تقریب توابع

منصور شیخان^۱، مهدی عباس‌نژاد عربی^۲

^۱ دانشیار، گروه مهندسی مخابرات، دانشگاه آزاد اسلامی، واحد تهران جنوب

msheikh@azad.ac.ir

^۲ کارشناس ارشد، گروه مهندسی الکترونیک، دانشگاه آزاد اسلامی، واحد تهران جنوب

m_abasnejad@yahoo.com

چکیده: کارایی بهتر شبکه عصبی به پارامترهایی همچون تعداد گره‌های ورودی، تعداد لایه‌های میانی، تعداد نرون‌ها، و وزن اختصاص یافته به نرون‌ها بستگی دارد. روش‌های متفاوتی جهت به‌روزرسانی پارامترها و ساختار شبکه عصبی مصنوعی ارائه گردیده است. یکی از روش‌های متداول و مورد استفاده در شبکه‌های عصبی، روش پس‌انتشار خطا (EBP) است که در آن تنها وزن‌های شبکه عصبی به‌روزرسانی می‌شوند. در این مقاله، از الگوریتم جستجوی گرانشی (GSA) بدین منظور استفاده شده است، که یکی از روش‌های نوین جستجو و بهینه‌سازی مبتنی بر هوش جمعی می‌باشد. در روش پیشنهادی، از GSA به همراه نسخه باینری الگوریتم جستجوی گرانشی (BGSA) به‌صورت همزمان، جهت آموزش شبکه عصبی و نیز تعیین ساختار بهینه شبکه عصبی استفاده شده است. نتایج عملکرد روش پیشنهادی با روش‌هایی چون الگوریتم تجمع ذرات (PSO)، الگوریتم آمیختار PSO و نسخه باینری PSO (PSO-BPSO) و نیز روش EBP در کاربردهای دسته‌بندی و تقریب توابع مقایسه شده است. عملکرد برای کاربرد دسته‌بندی بر روی سه دادگان استاندارد گل زنبق، سرطان سینه و انواع شیشه و برای کاربرد تقریب توابع، درخصوص یک سیستم تخمین نوا برای سنتز گفتار فارسی مورد ارزیابی قرار گرفته است. از آنجا که تعداد ورودی‌ها به شبکه عصبی در کاربرد سیستم تخمین نوا زیاد است، از یک الگوریتم آمیختار تکاملی و هوش جمعی نیز برای انتخاب ویژگی‌های مناسب بهره گرفته شده است. نتایج شبیه‌سازی نشان می‌دهد که روش پیشنهادی با به‌کارگیری تعداد نرون‌های کمتر در لایه مخفی (به‌میزان ۲۵ تا ۶۸ درصد کاهش در تعداد این نرون‌ها در مقایسه با بسیاری از الگوریتم‌های مورد بررسی)، دقت دسته‌بندی و تقریب را به‌صورت قابل رقابتی (به‌ویژه هنگام کار با داده‌های آزمون) از خود ارائه می‌نماید.

واژه‌های کلیدی: الگوریتم جستجوی گرانشی، دسته‌بندی داده، شبکه عصبی مصنوعی، الگوریتم آمیختار انتخاب ویژگی، سنتز گفتار

فارسی

۱. مقدمه

شبکه عصبی مصنوعی می‌تواند به‌عنوان یک راه حل مؤثر برای طبقه‌بندی داده‌ها، بازشناسی الگو، و تقریب توابع مورد استفاده قرار گیرد. مهم‌ترین مزیت در این شبکه‌ها، قابلیت یادگیری^۱ است [۱]. یکی از مرسوم‌ترین انواع شبکه‌های عصبی، شبکه پرسپترون چندلایه^۲ (MLP) است [۲]. هنگام کار با این شبکه با دو مسئله روبرو هستیم: الف) انتخاب الگوریتم آموزش مناسب، ب) انتخاب معماری مناسب. روش پس‌انتشار خطا^۳ (EBP) یکی از الگوریتم‌های مرسوم آموزش شبکه‌های عصبی چندلایه است [۱]. روش EBP دارای یک عیب ذاتی می‌باشد و آن توقف در بهینه‌های محلی است، که این مشکل بیشتر در مسائل دسته‌بندی غیرخطی یا تقریب توابع پیچیده مشاهده می‌شود [۳]. دومین مشکل شبکه عصبی MLP، سرعت کند آن در همگرا شدن می‌باشد که به مقادیر اولیه وزن‌های شبکه عصبی و مقادیر اولیه نرخ یادگیری^۴ وابسته است [۴، ۵].

ساختار شبکه عصبی MLP از یک لایه ورودی، یک لایه خروجی و حداقل یک لایه میانی تشکیل شده است. تعیین ساختار مناسب برای شبکه عصبی یکی از مهم‌ترین مسائل مربوط در طراحی می‌باشد. اگر ساختار شبکه عصبی کوچک باشد، نمی‌تواند به کارایی خوبی دست یابد و در صورت بزرگ بودن بیش از اندازه، موجب اتصالات داخلی اضافه، از بین رفتن قابلیت تعمیم^۵ و پیچیدگی بی‌مورد می‌شود [۶-۷]. هدف همه روش‌های پیشنهادی، به‌دست آوردن یک روش سیستماتیک برای تعیین ساختاری مناسب برای شبکه‌های عصبی می‌باشد [۲]. تعداد لایه‌های میانی مورد نیاز و مناسب برای یک شبکه عصبی MLP، یک یا دو لایه می‌باشد [۸، ۹]. برای تعیین تعداد گره‌های لایه میانی، روش‌های گوناگونی پیشنهاد شده است. به‌عنوان نمونه، در [۱۰]، تعداد گره‌های لایه میانی چنین پیشنهاد شده است:

$$n_2 = 0.75 \times n_1 \quad (1)$$

که n_1 تعداد گره‌های ورودی و n_2 تعداد گره‌های لایه میانی است. در [۱۱] نیز تعداد گره‌های لایه میانی بر اساس (۲) پیشنهاد شده است:

$$n_2 = n_1 + n_3 + 1 \quad (2)$$

که n_1 تعداد گره‌های ورودی، n_2 تعداد گره‌های لایه میانی و n_3 تعداد گره‌های لایه خروجی می‌باشد. همچنین تعداد گره‌های میانی پیشنهادی در [۱۲] برابر n_1 و در [۱۳] $2n_1 + 1$ در نظر گرفته شده‌اند.

یکی از روش‌های تعیین تعداد گره‌های لایه میانی، استفاده از الگوریتم‌های هوش محاسباتی می‌باشد. این الگوریتم‌ها به صورت پویا، تعداد گره‌های شبکه عصبی را مشخص می‌نمایند و اساس کار آن‌ها چنین می‌باشد [۲]:

۱- یک گره اضافه می‌شود، اگر سرعت همگرایی کند باشد یا میانگین مربعات خطا^۶ (MSE) بیشتر از یک آستانه باشد.

۲- یک گره کم می‌شود، اگر تغییر در وزن یک نرون، تغییری در خروجی ایجاد ننماید.

بر اساس قوانین بالا و ترکیب آن با الگوریتم‌های هوش محاسباتی مانند تجمع ذرات^۷ (PSO) [۱۷-۱۴] یا الگوریتم وراثتی^۸ (GA) [۱۸، ۱۹] می‌توان به قابلیت و برتری این الگوریتم‌ها بر روش‌های استاتیک پی برد. با توجه به استفاده گسترده از شبکه عصبی MLP از یک سو و ضرورت تعیین ساختار مناسب برای این شبکه جهت جلوگیری از طولانی شدن زمان آموزش از سوی دیگر (به‌دلیل تخصیص تعداد نرون‌ها به‌میزان بیشتر از نیاز و در نتیجه، پیچیده شدن ساختار شبکه و افزایش تعداد ضرایب وزن شبکه)، در این پژوهش از ترکیب نسخه حقیقی الگوریتم جستجوی گرانشی^۹ (GSA) [۲۰] و نسخه باینری آن^{۱۰} (BGSA) [۲۱] به‌ترتیب برای تنظیم بهینه ضرایب وزن شبکه MLP و نیز تعیین تعداد نرون‌ها در لایه مخفی این شبکه استفاده شده است. GSA با الهام از مفاهیم قانون گرانش نیوتن ارائه شده است. از نمونه

6. Mean Square Error
7. Particle Swarm Optimization
8. Genetic Algorithm
9. Gravitational Search Algorithm
10. Binary GSA

1. Learning
2. Multi-Layer Perceptron
3. Error Back Propagation
4. Learning rate
5. Generalization

وزن شبکه MLP با هدف مشترک در این مقاله استفاده شده‌اند. در جدول (۱) تعدادی از الگوریتم‌های به‌کارگرفته‌شده، فهرست شده‌اند. همان‌گونه که مشاهده می‌شود، تعداد قابل توجهی از پژوهش‌ها در سال‌های اخیر از الگوریتم PSO به صورت تکی یا ترکیب آن با سایر الگوریتم‌ها بهره‌جسته‌اند. نکته قابل توجه این است که در هر دو الگوریتم PSO و GSA، بهینه‌سازی از طریق حرکت عامل‌ها در فضای جستجو انجام می‌شود، هر چند که روش حرکت در این دو الگوریتم متفاوت است. در این پژوهش، از GSA به‌عنوان الگوریتم نوین‌تر و نیز با سه مزیت مذکور در زیر نسبت به الگوریتم PSO استفاده شده است [۲۰]: الف) روال به‌روزرسانی در الگوریتم PSO بدون در نظر گرفتن کیفیت جواب‌ها انجام می‌شود (مقادیر برازش در این روال مهم نیستند)، حال آنکه در GSA نیرو متناسب با مقدار برازش است (عامل‌ها فضای جستجوی پیرامون خود را با توجه به نیرو مشاهده می‌کنند)، ب) روال به‌روزرسانی در الگوریتم PSO بدون در نظر گرفتن فاصله بین جواب‌ها انجام می‌شود، حال آنکه در GSA نیرو متناسب معکوس با فاصله بین جواب‌ها دارد، پ) در الگوریتم PSO، راستا و جهت یک عامل تنها با توجه به دو موقعیت بهینه محاسبه می‌شود، حال آنکه در GSA این جهت بر اساس نیروی کلی حاصل از تمام عامل‌های دیگر محاسبه می‌شود لازم به ذکر است که در پژوهش ارائه‌شده در [۲۶] توسط نویسندگان، GSA تنها برای تعیین مقادیر بهینه نرخ یادگیری و پارامترهای مراقبت و انتخاب در شبکه ARTMAP فازی به‌کار گرفته شد و بهینه‌سازی ساختار شبکه صورت پذیرفت. به‌همین ترتیب، در پژوهش ارائه‌شده در [۲۹] توسط نویسندگان نیز از عملگر فروپاشی^۷ در GSA [۴۴] استفاده شده و ضرایب وزن شبکه به‌صورت بهینه تعیین شده‌اند. در پژوهش مذکور نیز بهینه‌سازی ساختار شبکه صورت پذیرفت. اما در پژوهش حاضر، الگوریتم آمیختار GSA-BGSA برای بهینه‌سازی همزمان ساختار و ضرایب شبکه عصبی مورد استفاده قرار گرفته است.

کاربردهای این الگوریتم می‌توان به طراحی فیلتر [۲۲]، انتخاب ویژگی [۲۳]، جبران‌سازی توان راکتیو [۲۴]، آموزش شبکه‌های عصبی [۲۵]، و بهینه‌سازی پارامترهای آموزش در شبکه عصبی [۲۶] اشاره داشت. در BGSA، گره‌ها به‌عنوان سوئیچ باینری در نظر گرفته می‌شوند.

لازم به ذکر است که بر اساس اطلاع نویسندگان، تاکنون استفاده از این ترکیب برای بهینه‌سازی همزمان ساختار و ضرایب وزن شبکه عصبی MLP در مراجع گزارش نشده است. البته این روش برای بهینه‌سازی سایر ساختارهای شبکه‌های عصبی مانند شبکه‌های عصبی بازگشتی^۱ (RNN) نیز کاربرد دارد. در این مورد، عملکرد الگوریتم پیشنهادی در کاربردهای دسته‌بندی الگوها و تقریب توابع مورد بررسی قرار گرفته است. برای کاربرد دسته‌بندی الگوها، از سه دادگان استاندارد گل زنبق^۲، سرطان سینه^۳ و شیشه^۴ و برای کاربرد تقریب توابع، عملکرد یک سیستم تخمین اطلاعات نوا^۵ برای سنتز گفتار طبیعی فارسی مورد ارزیابی قرار گرفته است. لازم به ذکر است که برای کاهش تعداد ویژگی‌های ورودی به شبکه عصبی در کاربرد تقریب توابع، از یک الگوریتم آمیختار GA و الگوریتم اجتماع مورچگان^۶ (ACO) نیز بهره گرفته شده است.

ساختار مقاله نیز چنین است که در بخش ۲ کارهای مرتبط مرور شده‌اند و در بخش ۳، الگوریتم‌های GSA و BGSA شرح داده شده‌اند. بخش ۴ به معرفی روش پیشنهادی بهینه‌سازی شبکه عصبی بر اساس نسخه‌های مذکور GSA می‌پردازد. نتایج عملکرد حاصل در کاربردهای دسته‌بندی و تقریب توابع در بخش ۵ گزارش شده و با چند روش مقایسه شده است. بخش ۶ نیز به نتیجه‌گیری اختصاص یافته است.

۲. کارهای مرتبط

در سال‌های اخیر برای بهینه‌سازی ساختار و یا ضرایب وزن شبکه‌های عصبی از الگوریتم‌های مختلفی استفاده شده است که برخی از آنها برای بهینه‌سازی همزمان ساختار و ضرایب

5. Prosody
6. Ant Colony Optimization
7. Disruption

1. Recurrent Neural Network
2. Iris
3. Breast cancer
4. Glass

جدول (۱): برخی از الگوریتم‌های مورد استفاده در سالیان اخیر برای بهینه‌سازی ساختار و/ یا ضرایب وزن شبکه‌های عصبی

توضیح	شبکه عصبی بهینه‌شده	روش / الگوریتم
آموزش شبکه توسط ترکیب الگوریتم PSO (که در گام‌های اولیه از یک جستجوی سراسری به سرعت همگرا می‌شود) و روش EBP (که در اطراف نقطه بهینه‌یابی همزمان ساختار و ضرایب وزن یک MLP سه‌لایه با استفاده از الگوریتم‌های آمیختار PSO و استراتژی تکاملی وفقی تعیین همزمان مراکز و ضرایب وزن در شبکه‌های پایه- شعاعی (RBF) بهینه‌سازی پارامترهای آموزش شبکه (مانند نرخ یادگیری، پارامترهای مراقبت ^۲ و انتخاب)	MLP	آمیختار PSO و EBP [۴]
بهینه‌سازی ضرایب وزن شبکه	MLP	ترکیب نسخه‌های حقیقی و باینری PSO [۲۷]
تنظیم همزمان ساختار و ضرایب وزن شبکه	RBF ^۱	PSO چندهدفی متغیر با زمان [۲۸]
ترکیب قابلیت GSA در جستجو برای بهینه سراسری و سرعت جستجوی بالای PSO در گام‌های آخر برای آموزش مؤثرتر	Fuzzy ARTMAP	GSA [۲۶]
بهینه‌یابی روال خوشه‌بندی k-means مورد استفاده در آموزش RBF به کارگیری ۵ گونه از الگوریتم جستجوی هارمونی برای آموزش شبکه طراحی مجموعه‌های نامتجانس از شبکه‌های عصبی	MLP	GSA اصلاح شده [۲۹]
به کارگیری CFO توزیع شده برای بهینه‌یابی هر یک از اجزا در ساختار جمعی شبکه‌های عصبی	RNN	ترکیب PSO و CMA-ES ^۳ [۳۰]
شبکه‌های عصبی	MLP	آمیختار PSO و GSA [۳۱]
بهینه‌سازی ساختار شبکه برای تقریب تابع توسط شبکه عصبی تعیین ضرایب وزن و آستانه اولیه برای آموزش به روش EBP تعیین ساختار بهینه شبکه	RBF	اجتماع ماهی‌ها ^۴ [۳۲]
بهینه‌سازی ساختار و پارامترهای شبکه MLP با یک لایه مخفی	MLP	جستجوی هارمونی ^۵ [۳۳]
بهینه‌سازی ضرایب وزن شبکه آموزش شبکه عصبی و مقایسه نتایج با الگوریتم اجتماع زنبورها	ساختار جمعی شبکه‌های عصبی ^۶	جستجوی مبتنی بر سیستم ایمنی مصنوعی [۳۴]
بهینه‌سازی مرکز خوشه‌ها در لایه مخفی شبکه RBF آموزش هر شبکه ایستا در NNE	NNE	بهینه‌سازی نیروی مرکزی ^۷ (CFO) [۳۵]
	MLP	هرس اصلاح شده ^۸ [۳۶]
	MLP	GA [۳۷]
	MLP	GA [۳۸]
	MLP	ماشین یادگیری حدی بهینه ^۹ (O-ELM)
	شبکه عصبی موجک ^{۱۰}	PSO [۴۰]
	MLP	رقابت استعماری ^{۱۱} (ICA) [۴۱]
	RBF	PSO [۴۲]
	NNE	آمیختار نسخه آشوب PSO و گرادیان کاهش ^{۱۱} [۴۳]

1. Radial Basis Function
2. Vigilance
3. Covariance Matrix Adaptation-Evolution Strategy
4. Fish swarm
5. Harmony search
6. Neural Network Ensemble
7. Central Force Optimization
8. Modified pruning
9. Optimized-Extreme Learning Machine
10. Wavelet Neural Network
11. Imperialist Competitive Algorithm

۳. الگوریتم جستجوی گرانشی

۱.۳. مبانی الگوریتم

الگوریتم جستجوی گرانشی یکی از الگوریتم‌های ارائه شده برای بهینه‌سازی می‌باشد که از قانون گرانش نیوتن برای حل هر مسئله بهینه‌سازی که در آن پاسخ‌های ممکن به صورت یک موقعیت در فضا قابل تعریف باشند، استفاده می‌نماید. عامل‌های جستجوکننده در GSA، مجموعه‌ای از اجرام با چهار مشخصه‌اند: الف) موقعیت جرم، ب) جرم گرانشی فعال، پ) جرم گرانشی غیرفعال، ت) جرم اینرسی [۲۰]. اجرام فوق برگرفته از مفاهیم جرم گرانشی فعال و جرم اینرسی در فیزیک می‌باشند. اجرام گرانشی و اینرسی با الهام از مفاهیم فیزیک نیوتنی تعریف شده‌اند و با توجه به تابع برازندگی عامل‌ها تعیین می‌شوند. جرم گرانشی فعال، معیاری از مقدار نیروی گرانشی حول یک جسم است و جرم اینرسی معیاری از مقاومت جسم در مقابل حرکت است. جرم گرانشی غیرفعال نشان‌دهنده قدرت اثر متقابل در میدان گرانشی است. به طور کلی هر جرم نشان‌دهنده یک راه حل است و برای به‌روزرسانی و بهینه‌سازی اجرام، نیاز به قوانین مشخصی می‌باشد. قوانین حاکم بر این الگوریتم شامل قانون گرانش و قوانین حرکت می‌باشند.

قانون گرانش: هر ذره در جهان، ذرات دیگر را با نیروی گرانشی به سمت خود جذب می‌کند. مقدار این نیرو با جرم ذرات رابطه مستقیم و با مجذور فاصله دو جسم رابطه معکوس دارد.

قانون حرکت: سرعت فعلی هر جرم برابر است با مجموع ضربی از سرعت قبلی جرم و تغییرات سرعت آن جرم. تغییر سرعت یا شتاب هر جرم برابر است با نیروی وارد بر آن جرم تقسیم بر جرم اینرسی [۲۰].

سیستمی را با m جرم در نظر بگیرید که موقعیت هر جرم به صورت زیر تعریف می‌شود:

$$X_i = (x_i^1, \dots, x_i^d, \dots, x_i^n); i = 1, 2, \dots, m \quad (3)$$

که x_i^d موقعیت بُعد d از جرم m و n بُعد فضای پاسخ می‌باشد. برای به دست آوردن و تنظیم موقعیت هر جرم از تابع

برازندگی استفاده می‌شود، طوری که به اجرام با شایستگی بهتر، جرم بیشتری اختصاص داده می‌شود و جرم بیشتر نشان‌دهنده نیروی جذب بیشتر و حرکت آهسته‌تر است. مطابق (۴) اجرام گرانشی فعال (M_{ai})، گرانشی غیرفعال (M_{pi}) و اینرسی (M_{ii}) مانند آنچه در طبیعت وجود دارد، مساوی فرض می‌شوند.

$$M_{ai} = M_{pi} = M_{ii} = M_i; i=1, 2, \dots, N \quad (4)$$

که در آن M_i و نیز m_i که در تعریف M_i از آن استفاده شده است به ترتیب در (۵) و (۶) آورده شده‌اند:

$$M_i(t) = \frac{m_i(t)}{\sum_{j=1}^N m_j(t)} \quad (5)$$

$$m_i(t) = \frac{fit_i(t) - worst(t)}{best(t) - worst(t)} \quad (6)$$

که در این رابطه، $fit_i(t)$ بیانگر میزان برازندگی جرم m در زمان t است. برای مسائل کمینه‌یابی، $best(t)$ و $worst(t)$ به ترتیب بر اساس (۷) و (۸) تعریف می‌شوند:

$$best(t) = \min_{j \in \{1, \dots, m\}} fit_j(t) \quad (7)$$

$$worst(t) = \max_{j \in \{1, \dots, m\}} fit_j(t) \quad (8)$$

پس از محاسبه جرم، مقدار نیروی وارد شده بر جرم m توسط جرم m در بُعد d ، بر اساس (۹) محاسبه می‌شود.

$$F_{ij}^d(t) = G(t) \frac{M_{pi}(t) \times M_{aj}(t)}{R_{ij}(t) + \epsilon} (x_j^d(t) - x_i^d(t)) \quad (9)$$

که M_{aj} جرم گرانشی فعال جرم m ، M_{pi} جرم گرانشی غیرفعال جرم m و $G(t)$ ثابت گرانش در زمان t و R_{ij} مبین فاصله اقلیدسی بین دو جرم i و j می‌باشند.

$$R_{ij}(t) = \|X_i(t) - X_j(t)\|_2 \quad (10)$$

در (۹)، ϵ یک عدد بسیار کوچک است. نیروی وارد بر جرم m در جهت بُعد d در زمان t ($F_i^d(t)$)، برابر مجموع نیروهایی است که k جرم برتر جمعیت بر جرم وارد می‌کنند. مقصود از اجرام برتر، عامل‌هایی هستند که دارای برازندگی

رابطه‌ی نمایی جهت کاهش ثابت گرانش در حل بسیاری از مسائل مؤثر است [۲۰]. در شکل (۱)، الگوریتم GSA آورده شده است [۲۲].

۱	تعیین محیط سیستم و مقداردهی اولیه
۲	جایابی اولیه اجرام
۳	ارزیابی اجرام
۴	به‌روزرسانی پارامترهای G , $worst$ و m
۵	محاسبه نیروی وارد بر هر جرم
۶	محاسبه شتاب و سرعت هر جرم
۷	به‌روزرسانی موقعیت اجرام
۸	اگر شرط توقف برآورده نشده است، به مرحله ۳ برو.
۹	پایان

شکل (۱): الگوریتم GSA

۲.۳. نسخه باینری الگوریتم جستجوی گرانشی

مقدار هر بُعد در الگوریتم‌های باینری می‌تواند فقط مقدار 0 یا 1 را بگیرد و حرکت در یک بُعد نیز به این معنی است که مقدار هر متغیر در الگوریتم‌های باینری از 0 به 1 یا برعکس تغییر می‌نماید. الگوریتم BGSA نیز از این قانون تبعیت می‌کند. الگوریتم BGSA مبتنی بر GSA و تفاوت این دو الگوریتم تنها در بعضی از مفاهیم اولیه است. در BGSA موقعیت هر جرم با بیت 0 یا 1 نمایش داده می‌شود و برای تغییر مقدار این موقعیت از تابع احتمال بر اساس سرعت جرم استفاده می‌گردد. به بیان دیگر، به‌روزرسانی سرعت در BGSA بر پایه (۱۳) می‌باشد و بر اساس تابع احتمال موقعیت محاسبه می‌گردد.

برای پیاده‌سازی BGSA نیاز است تا علاوه بر مفاهیم اولیه GSA، دو مفهوم دیگر نیز در نظر گرفته شوند [۲۱]:

۱- مقدار بزرگ سرعت موجب می‌شود تا خروجی تابع احتمال، مقدار بزرگی باشد و در نتیجه این مقدار سبب تغییر موقعیت قبلی از 0 به 1 یا برعکس می‌گردد.

بیشتری می‌باشند. مقدار k با یک مقدار اولیه k_0 شروع شده و به صورت خطی کاهش می‌یابد [۲۰].

$$F_i^d(t) = \sum_{j \in kbest, j \neq i} rand_j(t) \times F_{ij}^d(t) \quad (11)$$

در (۱۱)، $rand_j$ یک عدد تصادفی با توزیع یکنواخت در بازه $[0,1]$ است که برای حفظ خصوصیت تصادفی بودن جستجو در نظر گرفته می‌شود [۲۰]. قدم‌های بعدی، محاسبه شتاب و محاسبه سرعت بر اساس قانون حرکت می‌باشد. مقدار شتاب ذره i ام در زمان t و بُعد d برابر است با:

$$a_i^d(t) = \frac{F_i^d(t)}{M_{ii}(t)} \quad (12)$$

روابط به‌روزرسانی سرعت و موقعیت نیز به ترتیب در (۱۳) و (۱۴) ارائه شده‌اند:

$$V_i^d(t+1) = rand_i \times V_i^d(t) + a_i^d(t) \quad (13)$$

$$x_i^d(t+1) = x_i^d(t) + V_i^d(t+1) \quad (14)$$

در روابط فوق $V_i^d(t)$ سرعت بُعد d عامل i در زمان t است. برای تنظیم ضریب گرانش از (۱۵) استفاده می‌شود. در (۱۵)، T مبین تعداد کل تکرارها است و α نیز نرخ نزول این تابع کاهشی را مشخص می‌کند.

$$G(t) = (1 + 0.02 \times t)^{-\alpha/T} \quad (15)$$

ثابت گرانش یک پارامتر مناسب برای کنترل توانایی‌های کاوش^۱ و بهره‌وری در این الگوریتم به‌شمار می‌آید. مقادیر بزرگ برای این پارامتر باعث تقویت توانایی کاوش الگوریتم و مقادیر کوچک آن باعث افزایش توانایی بهره‌وری الگوریتم می‌شود [۲۰]. در مراحل اولیه جستجو لازم است که الگوریتم به جستجوی نقاط جدیدی در فضای مسئله پرداخته و در مراحل پایانی با افزایش توان بهره‌وری به بهبود جواب‌های دیده‌شده بپردازد. گزینه مناسب برای ثابت گرانش آن است که با یک مقدار اولیه بزرگ شروع شده و با گذشت زمان مقدار آن کاهش یابد. طبق آزمایش‌های متعدد انجام‌شده، استفاده از

تصادفی تولید می‌شود [۲۵]. بنابراین ذرات به صورت (۱۹) نشان داده می‌شوند:

$$W_{jd} = [x_{11}, \dots, x_{n_h}, b_1^1, \dots, b_{n_h}^1, w_{11}, \dots, w_{n_h}, b_1^2, \dots, b_{n_h}^2] \\ ; j=1, \dots, m, \quad d=1, \dots, D \quad (19)$$

که m نشان‌دهنده تعداد ذرات، x نشان‌دهنده مقدار وزن‌های بین گره‌های ورودی و لایه میانی، b نشان‌دهنده بایاس و w بیانگر مقدار وزن‌های میان گره‌های لایه میانی و لایه خروجی می‌باشد. همچنین D نشان‌دهنده بُعد بردار می‌باشد که برابر است با:

$$D = (n_i + 1) \times n_h + (n_h + 1) \times n_o \quad (20)$$

در این رابطه، n_i مبین تعداد گره‌های ورودی، n_h مبین تعداد گره‌های لایه میانی، و n_o مبین تعداد گره‌های لایه خروجی است.

همچنین برای تعیین تعداد گره‌های میانی از BGSA استفاده می‌شود و برای هر گره در لایه میانی یک سوئیچ در نظر گرفته می‌شود و در صورتی که وجود یک نرون ضروری نباشد، به سوئیچ باینری مربوط به آن نرون، مقدار 0 اختصاص می‌یابد. با استفاده از الگوریتم BGSA یک بردار باینری بر اساس تعداد نرون‌ها به صورت (۲۱) تعریف می‌شود:

$$W_j = [\delta_1, \delta_2, \delta_3, \dots, \delta_{n_h}] ; j=1, \dots, m \quad (21)$$

تابع هزینه مورد استفاده (C) برای کمینه‌سازی توسط BGSA بر اساس (۲۲) در نظر گرفته شده است:

$$C = MSE + \eta \left(\frac{n_h}{\max_n_h} \right) \quad (22)$$

MSE در (۲۲) مبین میانگین مربعات خطا پس از آموزش شبکه و η ضریب توازن بین دو جمله سمت راست است تا تعداد نرون‌های لایه مخفی نیز در تابع هزینه به تناسب دیده شود. این مقدار در شبیه‌سازی‌های این پژوهش برابر ۰/۲ در نظر گرفته شده است.

۲- مقدار کوچک سرعت موجب می‌شود تا خروجی تابع احتمال، مقدار کوچکی باشد و در نتیجه این مقدار نشان‌دهنده این است که موقعیت قبلی مناسب می‌باشد و نیازی به تغییر ندارد.

بر پایه مفاهیم فوق، یک تابع احتمال مناسب، تابعی است که به‌ازای مقادیر کوچک $|v_i^d|$ ، احتمال تغییر x_i^d نزدیک به صفر باشد و برای مقادیر بزرگ $|v_i^d|$ ، احتمال تغییر موقعیت x_i^d زیاد باشد. تابع $S(v_i^d)$ برای تبدیل $|v_i^d|$ به تابع احتمال به صورت (۱۶) پیشنهاد شده است [۲۱]:

$$S(v_i^d(t)) = |\tanh(v_i^d(t))| \quad (16)$$

خروجی این تابع در بازه [0,1] قرار دارد. با توجه به (۱۶)، ذرات بر اساس (۱۷) حرکت خواهند کرد:

$$\text{if } rand < S(v_i^d(t+1)) \text{ then } x_i^d(t+1) = \text{complement}(x_i^d(t)) \\ \text{else } x_i^d(t+1) = x_i^d(t) \quad (17)$$

برای به‌دست آوردن نرخ همگرایی بهتر نیاز به محدودیت سرعت است که مقدار این محدودیت طبق (۱۸) در نظر گرفته شده است. باید توجه داشت که مقدار R در الگوریتم BGSA بر اساس فاصله همینگ^۱ به‌دست می‌آید.

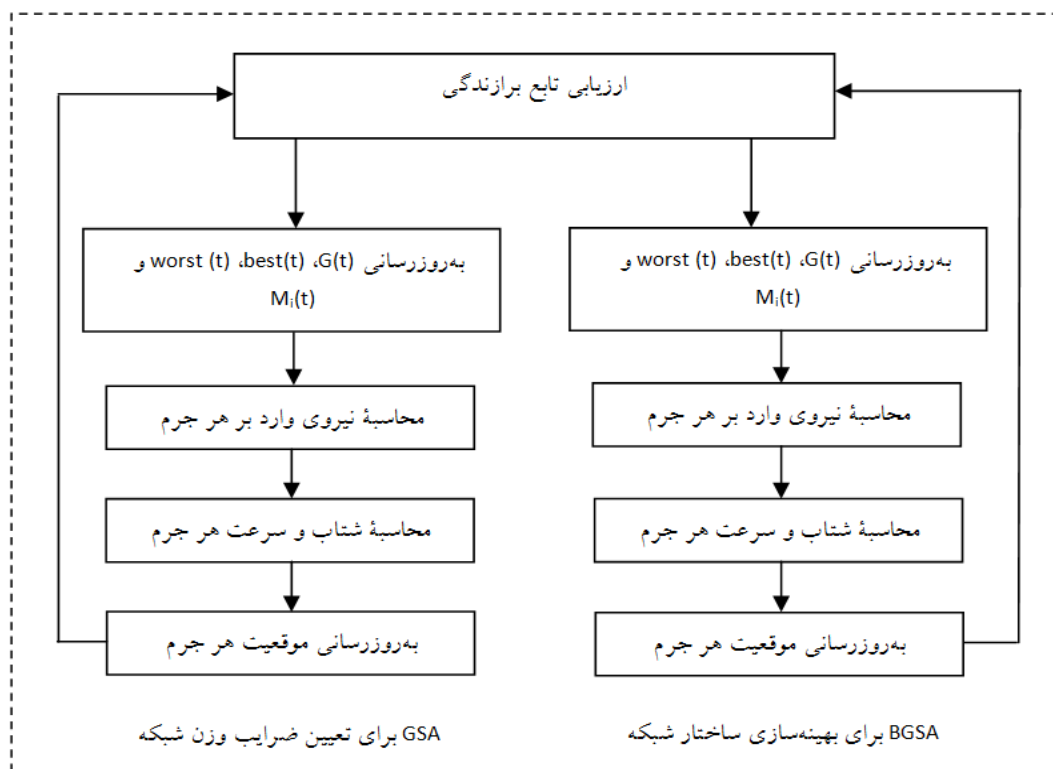
$$|v_i^d| < v_{\max} \quad ; \quad v_{\max} = 6 \quad (18)$$

۴. روش پیشنهادی برای بهینه‌سازی شبکه عصبی

به روش GSA-BGSA

در این بخش روش تنظیم ساختار و پارامترهای یک شبکه عصبی MLP شرح داده می‌شود. در فرایند آموزش برای دسته‌بندی داده‌ها/ تقریب توابع با استفاده از شبکه عصبی MLP، هدف نهایی جستجوی بهترین ضرایب وزن شبکه و نیز بهترین تعداد گره‌های لایه میانی به ترتیب با استفاده از الگوریتم GSA و BGSA است که به MSE کمینه منجر شود. در الگوریتم GSA هر ذره به صورت یک بردار وزنی با توجه به تعداد کل نرون‌های شبکه عصبی MLP و به صورت

1. Hamming distance



شکل (۲): روندنمای الگوریتم GSA-BGSA

استفاده از ترکیب دو الگوریتم می‌باشد. همچنین تابع برازندگی شبکه عصبی به‌عنوان تابع برازندگی ذرات در جهت محاسبه جرم ذرات می‌باشد.

۴- سپس شتاب، سرعت و موقعیت هر ذره بر اساس روابط (۱۲)، (۱۳) و (۱۴) محاسبه می‌شوند.

۵- این الگوریتم تا زمانی که به یک تعداد تکرار مشخص یا یک خطای مورد قبول برسد، تکرار می‌گردد و هنگام به پایان رسیدن الگوریتم، جرم با حداقل MSE به‌عنوان بهترین جرم در نظر گرفته می‌شود.

بدیهی است که تعداد تکرار یا میزان خطای مورد قبول توسط کاربر تعیین می‌شود. به‌عنوان نمونه در آزمایش‌های بخش ۵، بیشینه تعداد تکرار برابر ۱۰۰۰ در نظر گرفته شده است (هرچند که GSA-BGSA در تعداد گام‌های بسیار کمتری همگرا می‌شود). انتخاب این میزان بیشینه تعداد تکرار برای یکسان گرفتن شرایط برای تمام الگوریتم‌های مورد بررسی در این پژوهش (به‌ویژه برای EBP) می‌باشد.

الگوریتم GSA-BGSA جهت بهینه‌سازی وزن و ساختار شبکه عصبی در شکل (۲) نشان داده شده است. این دو الگوریتم به‌صورت مستقل از یکدیگر برای جستجو در فضای حل مسئله جهت یافتن پاسخ بهینه اجرا شده و در هنگام محاسبه کمینه تابع هزینه با یکدیگر ترکیب می‌شوند. مراحل ترکیب و اجرای این الگوریتم در ۵ مرحله چنین است:

۱- وزن‌ها و ساختار شبکه عصبی به‌صورت دو بردار در GSA و BGSA تعریف می‌شوند.

۲- موقعیت اجرام در دو الگوریتم مذکور به‌صورت تصادفی و در محدوده مشخص شده بارگذاری می‌شوند. همچنین پارامترهای اولیه الگوریتم GSA، همچون ضریب گرانش نیز تنظیم می‌شوند.

۳- با اجرای دو الگوریتم GSA و BGSA اجرام در فضا حرکت می‌کنند و سعی می‌کنند تابع هزینه را کمینه سازند. تغییر موقعیت اجرام به‌معنای بهینه‌سازی وزن‌ها و ساختار شبکه در جهت کاهش MSE در تکرار جاری با

۵. شبیه‌سازی و نتایج تجربی

تعداد گره‌های اولیه لایه میانی نیز برابر ۷۵٪ گره‌های ورودی می‌باشد.

• الگوریتم آمیختار PSO-BPSO

پارامترهای اولیه این الگوریتم نیز مانند الگوریتم PSO در نظر گرفته شده است. تعداد گره‌های اولیه لایه میانی نیز برابر ۷۵٪ گره‌های ورودی فرض شده، تا الگوریتم مورد نظر بتواند بر اساس این ساختار اولیه، ساختار بهینه را به دست آورد.

• آمیختار GSA-BGSA

پارامترهای اولیه و تعداد گره‌های اولیه لایه میانی در این الگوریتم نیز مانند GSA فرض شده است. همچنین ۷۰ درصد از دادگان به آموزش و ۳۰ درصد باقی‌مانده به آزمون اختصاص داده شده است.

۱.۵. نتایج برای دسته‌بندی دادگان گل زنبق

این مجموعه دادگان یکی از متداول‌ترین دادگان کلاسیک در زمینه بازشناسی الگو است که شامل ۱۵۰ اندازه‌گیری از چهار ویژگی سه نوع گل زنبق می‌باشد. در این دادگان، یکی از کلاس‌ها می‌تواند از دو کلاس دیگر به صورت خطی تفکیک شود، اما تفکیک پذیری دو کلاس دیگر به صورت غیرخطی است. در جداول (۲) و (۳) نرخ تشخیص صحیح داده‌های آموزشی و آزمون شبکه عصبی MLP توسط ۵ روش مذکور در طبقه‌بندی داده‌های گل زنبق آورده شده است. همان‌طور که مشاهده می‌شود در فاز آزمون، الگوریتم GSA-BGSA دارای بیشترین میانگین نرخ تشخیص درست می‌باشد، اما در دسته‌بندی داده‌های آموزشی، الگوریتم PSO عملکرد بهتری داشته است. نکته قابل توجه اینکه الگوریتم GSA-BGSA قابلیت تعمیم بهتری را با به کارگیری تعداد گره کمتر در لایه میانی نسبت به دیگر روش‌ها ارائه داده است. در شکل (۳) مقایسه‌ای میان ۵ الگوریتم از نقطه نظر MSE در ۱۰ مرتبه آزمون ارائه شده است. همان‌گونه که در شکل مشاهده می‌شود، الگوریتم پیشنهادی کمترین MSE را نتیجه می‌دهد.

ارزیابی عملکرد الگوریتم پیشنهادی در کاربردهای دسته‌بندی الگو و تقریب توابع در این بخش انجام می‌شود. برای کاربرد دسته‌بندی الگو از سه مجموعه- داده مرجع (داده‌های انواع گل زنبق، داده‌های سرطان سینه و دادگان انواع شیشه از تارگه مربوط به مجموعه- داده‌های یادگیری ماشین دانشگاه کالیفرنیا [۴۵]) استفاده شده است. برای کاربرد تقریب توابع نیز یک سیستم تخمین اطلاعات نوا مبتنی بر شبکه MLP بهینه شده برای سنتز گفتار فارسی مورد استفاده قرار گرفته است.

در شرایط یکسان، شبکه عصبی MLP جهت دسته‌بندی داده‌ها توسط روش‌های GSA، GSA-BGSA، PSO، PSO-BPSO و روش متداول EBP آموزش داده می‌شود. تمام نتایج برای بیشینه تکرار ۱۰۰۰ برای هر الگوریتم آورده شده است. همچنین با توجه به تصادفی بودن الگوریتم‌های فوق، آزمایش‌های مربوط به کاربرد دسته‌بندی برای ۱۰ مرتبه تکرار شده است. مشخصات روش‌های به کار گرفته شده برای آموزش به شرح زیر است:

• الگوریتم EBP

در این روش، تعداد گره‌های میانی بر اساس [۱۰] برابر ۷۵٪ تعداد گره‌های ورودی در نظر گرفته شده است. همچنین نرخ یادگیری برابر ۰/۰۰۱ فرض شده است.

• الگوریتم PSO

در این روش ثابت‌های شتاب (C_1 و C_2) با هم برابر و مساوی ۱/۴ فرض شده‌اند. همچنین تابع اینرسی W به صورت خطی از ۰/۹ تا ۰/۲ کاهش می‌یابد. جمعیت اولیه ذرات (بردارهای وزن) نیز برابر ۳۰ در نظر گرفته شده است. در این الگوریتم تعداد گره‌های اولیه لایه میانی نیز برابر ۷۵٪ گره‌های ورودی فرض شده است.

• GSA

جمعیت اولیه اجرام (بردارهای وزن) برابر ۳۰ در نظر گرفته شده است. همچنین برای تنظیم ضریب گرانش در (۱۵)، مقدار $\alpha = 0.001$ در نظر گرفته شده است. در این الگوریتم،

۲.۵. نتایج برای دسته‌بندی دادگان سرطان سینه

این مجموعه شامل ۵۶۹ نمونه مربوط به توده‌های سرطانی موجود در سینه بیماران، از دانشگاه Wisconsin است. نمونه‌ها دارای ۳۱ بُعد می‌باشند که از تصاویر دیجیتالی شده به دست آمده و در دو کلاس خوش‌خیم و بدخیم دسته‌بندی می‌شوند. نرخ تشخیص صحیح داده‌های آموزشی و آزمون این دادگان نیز در جداول (۴) و (۵) ارائه شده است. همان‌طور که مشاهده می‌شود در دسته‌بندی داده‌های آموزشی، الگوریتم‌های GSA و GSA-BGSA عملکرد بهتری در مقایسه با سایرین داشته‌اند. در فاز آزمون نیز الگوریتم GSA-BGSA عملکرد تقریباً مشابهی با GSA داشته، که حکایت از قابلیت تعمیم خوب آن دارد و این در شرایطی است با به‌کارگیری تنها ۷ گره میانی (درمقابل ۲۲ گره میانی استفاده‌شده در حالت به‌کارگیری

GSA) این عملکرد حاصل شده است. در شکل (۴) نیز مقایسه‌ای میان ۵ الگوریتم از نقطه‌نظر MSE در ۱۰ مرتبه آزمون با دادگان سرطان سینه ارائه شده است که حاکی از عملکرد قابل قبول آمیختار GSA-BGSA با به‌کارگیری ساختار ساده‌تری از شبکه عصبی دارد.

۳.۵. نتایج برای دسته‌بندی دادگان انواع شیشه

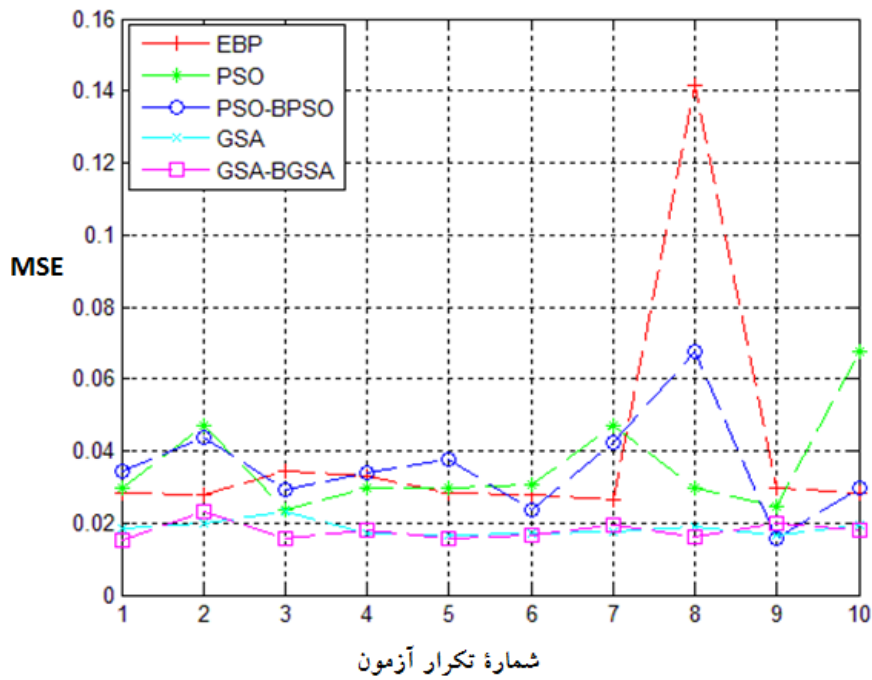
این دادگان مربوط به دسته‌بندی انواع شیشه و شامل ۲۱۴ نمونه در ۶ کلاس و بردارهای ویژگی آن دارای ۱۰ مؤلفه هستند. در جداول (۶) و (۷) نرخ تشخیص صحیح داده‌های آموزشی و آزمون شبکه عصبی MLP توسط ۵ روش مذکور برای این دادگان ارائه شده و در شکل (۵) نیز MSE در ۱۰ مرتبه آزمون این روش‌های بهینه‌یابی هوشمند ترسیم شده است.

جدول (۲): نرخ تشخیص صحیح داده‌های آموزشی برای دسته‌بندی دادگان گل زنبق در پنج روش آموزشی

الگوریتم آموزش و/یا بهینه‌سازی ساختار	تعداد گره‌های میانی	میانگین نرخ (%)	بیشینه نرخ (%)	کمینه نرخ (%)
EBP	۴	۹۷/۶۲	۹۹/۰۵	۹۳/۳۳
PSO	۴	۹۸/۳۸	۹۹/۰۵	۹۶/۱۹
PSO-BPSO	۴	۹۶/۸۰	۹۵/۰۵	۹۴/۲۹
GSA	۴	۹۷/۱۴	۹۸/۱۰	۹۴/۲۹
GSA-BGSA	۳	۹۵/۹۰	۹۹/۰۵	۹۳/۳۳

جدول (۳): نرخ تشخیص صحیح داده‌های آزمون برای دسته‌بندی دادگان گل زنبق در پنج روش آموزشی

الگوریتم آموزش و/یا بهینه‌سازی ساختار	تعداد گره‌های میانی	میانگین نرخ (%)	بیشینه نرخ (%)	کمینه نرخ (%)
EBP	۴	۹۳/۷۸	۹۵/۵۶	۷۷/۷۸
PSO	۴	۹۵/۱۱	۱۰۰/۰۰	۸۸/۸۹
PSO-BPSO	۴	۹۴/۹۵	۱۰۰/۰۰	۸۶/۶۷
GSA	۴	۹۶/۰۰	۹۷/۷۸	۹۵/۵۶
GSA-BGSA	۳	۹۷/۳۳	۱۰۰/۰۰	۹۵/۵۶



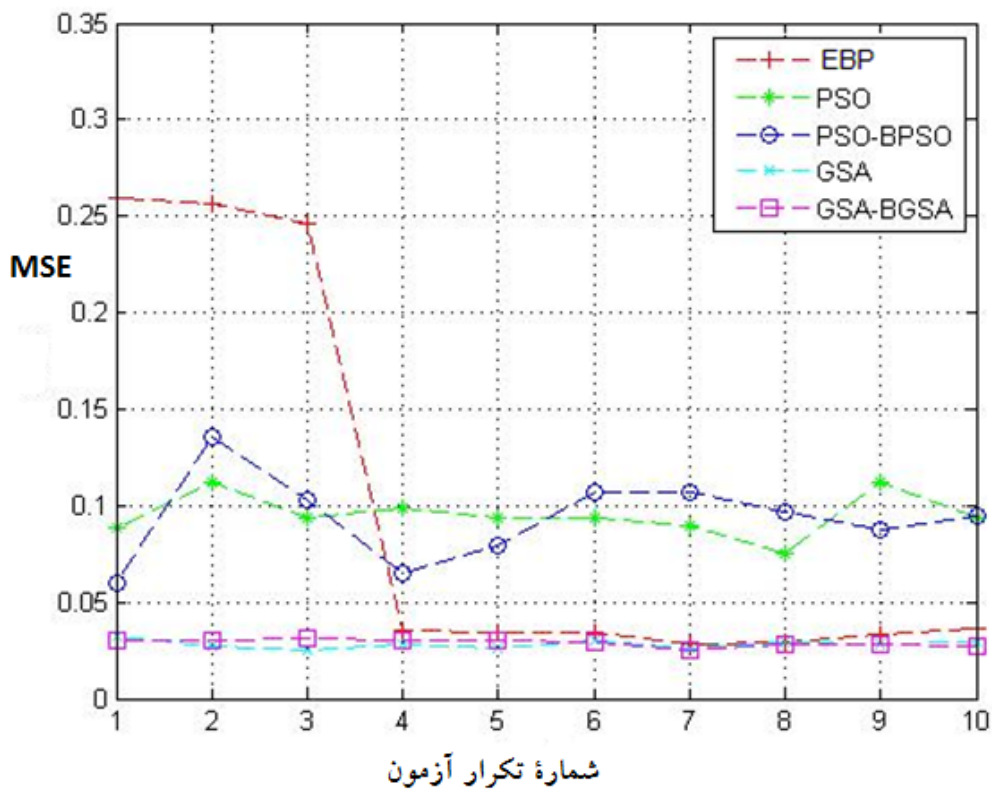
شکل (۳): MSE پنج روش آموزشی برای دادگان گل زنبق در ۱۰ بار تکرار آزمون

جدول (۴): نرخ تشخیص صحیح داده‌های آموزشی برای دسته‌بندی دادگان سرطان سینه در پنج روش آموزشی

الگوریتم آموزش و / یا بهینه‌سازی	تعداد گره‌های میانی	میانگین نرخ (%)	بیشینه نرخ (%)	کمینه نرخ (%)
EBP	۲۲	۸۷/۲۸	۹۹/۵۰	۵۸/۰۰
PSO	۲۲	۹۲/۴۰	۹۳/۲۵	۹۰/۵۰
PSO-BPSO	۸	۹۲/۳۶	۹۵/۷۵	۸۸/۷۵
GSA	۲۲	۹۹/۲۵	۹۹/۵۰	۹۹/۰۰
GSA-BGSA	۷	۹۸/۱۱	۹۸/۷۵	۹۵/۷۵

جدول (۵): نرخ تشخیص صحیح داده‌های آزمون برای دسته‌بندی دادگان سرطان سینه در پنج روش آموزشی

الگوریتم آموزش و / یا بهینه‌سازی	تعداد گره‌های میانی	میانگین نرخ (%)	بیشینه نرخ (%)	کمینه نرخ (%)
EBP	۲۲	۸۵/۵۰	۹۶/۴۵	۶۰/۹۵
PSO	۲۲	۹۱/۱۸	۹۲/۹۰	۸۹/۳۵
PSO-BPSO	۸	۹۰/۴۲	۹۳/۴۹	۸۶/۳۹
GSA	۲۲	۹۶/۲۹	۹۴/۰۷	۹۵/۸۶
GSA-BGSA	۷	۹۶/۱۸	۹۷/۰۴	۹۵/۸۶



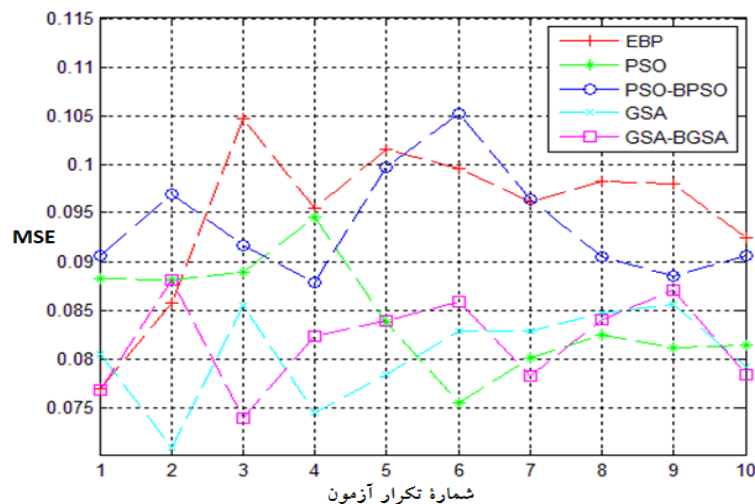
شکل (۴): MSE پنج روش آموزشی برای دادگان سرطان سینه در ۱۰ بار تکرار آزمون

جدول (۶): نرخ تشخیص صحیح داده‌های آموزشی برای دسته‌بندی دادگان انواع شیشه در پنج روش آموزشی

الگوریتم آموزش و / یا بهینه‌سازی ساختار	تعداد گره‌های میانی	میانگین نرخ (%)	بیشینه نرخ (%)	کمینه نرخ (%)
EBP	۷	۸۶/۰۶	۹۱/۳۳	۷۸/۶۶
PSO	۷	۶۶/۰۰	۷۴/۶۶	۵۴/۰۰
PSO-BPSO	۵	۵۹/۵۷	۶۷/۳۳	۵۰/۰۰
GSA	۷	۶۹/۳۳	۷۳/۳۳	۶۳/۳۳
GSA-BGSA	۵	۶۲/۱۸	۷۲/۰۰	۵۵/۳۳

جدول (۷): نرخ تشخیص صحیح داده‌های آزمون برای دسته‌بندی دادگان انواع شیشه در پنج روش آموزشی

الگوریتم آموزش و / یا بهینه‌سازی ساختار	تعداد گره‌های میانی	میانگین نرخ (%)	بیشینه نرخ (%)	کمینه نرخ (%)
EBP	۷	۶۶/۴۰	۷۱/۸۵	۵۷/۸۱
PSO	۷	۶۲/۶۵	۷۱/۸۵	۵۱/۵۶
PSO-BPSO	۵	۶۰/۹۳	۶۸/۷۵	۴۸/۴۳
GSA	۷	۶۹/۳۷	۷۶/۵۶	۶۵/۶۲
GSA-BGSA	۵	۶۸/۶۰	۷۱/۸۵	۶۸/۷۵



شکل (۵): MSE پنج روش آموزشی برای دادگان انواع شیشه در ۱۰ بار تکرار آزمون

تولید گفتار طبیعی در هر زبانی ضروری است. درخصوص تخمین اطلاعات نوا در زبان فارسی تلاش‌هایی در دو دهه اخیر انجام گرفته است که می‌توان به‌عنوان نمونه به کارهای نویسنده و همکاران در مراجع [۴۶-۴۹] اشاره داشت. لازم به ذکر است که در هیچیک از کارهای ارائه‌شده در مراجع [۴۹-۴۶] از ساختار بهینه شبکه و نیز آموزش با GSA استفاده نشده است. در این راستا، در مرجع [۴۹] از BGSA برای انتخاب ویژگی‌های ورودی به شبکه عصبی RNN بهره گرفته شد. در مرجع [۴۸] نیز از الگوریتم GA-ACO برای انتخاب ویژگی‌های ورودی به شبکه عصبی RNN استفاده شد. در مراجع [۴۶، ۴۷] نیز اساساً انتخاب ویژگی انجام نشده و از الگوریتم EBP برای آموزش شبکه استفاده شد. در تکمیل کارهای قبلی نویسنده، در پژوهش حاضر از شمای بلوکی ارائه‌شده در شکل (۶) که در واحد مولد نوای گفتار آن، MLP بهینه‌شده توسط GSA-BGSA کار می‌کند، بهره گرفته شده است. همان‌گونه که در شکل (۶) مشاهده می‌شود، برای کاهش تعداد ورودی‌های به شبکه عصبی از الگوریتم آمیختار GA-ACO استفاده شده است [۵۰].

الگوریتم GA-ACO با تولید یک جمعیت در GA و تعدادی مورچه در ACO کار خود را آغاز می‌کند. GA زیرمجموعه‌های ویژگی را تولید کرده و زیرمجموعه‌های

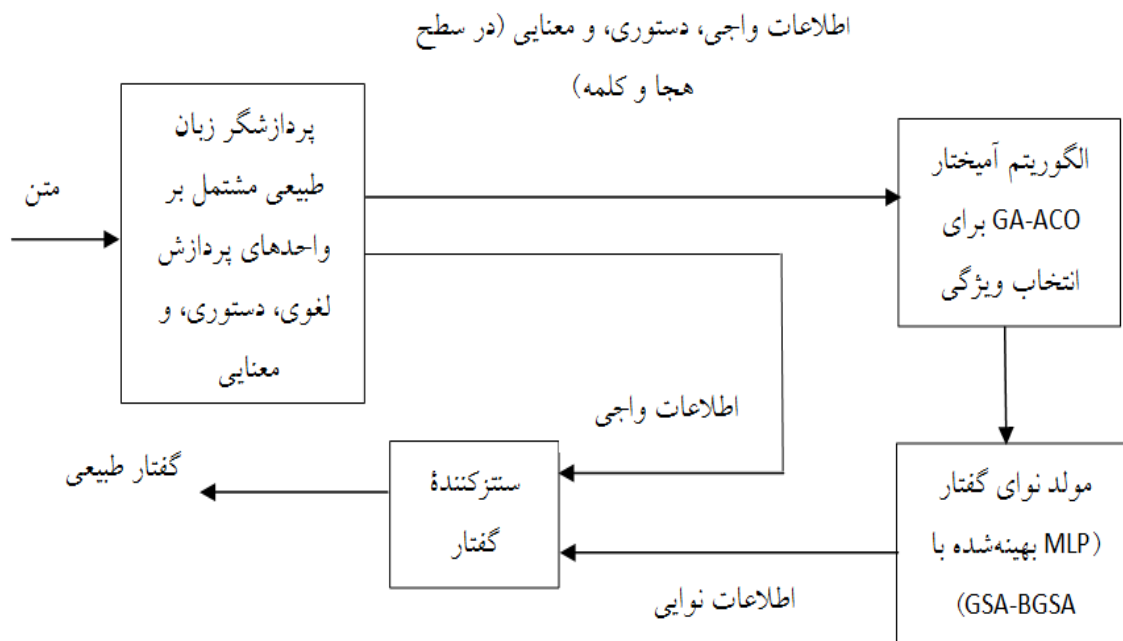
همان‌گونه که در جدول (۶) مشاهده می‌شود در دسته‌بندی داده‌های آموزش، الگوریتم EBP برای این دادگان بهتر از دیگر الگوریتم‌ها عمل نموده است. اما در دسته‌بندی داده‌های آزمون، GSA و GSA-BGSA دارای نرخ تشخیص بهتری هستند (البته GSA-BGSA با به‌کارگیری شبکه‌ای با تعداد نرون‌های کمتر در لایه مخفی). این موضوع دلالت بر گیرافتادن الگوریتم EBP در بهینه‌های محلی می‌باشد. همان‌طور که در شکل (۵) نیز مشاهده می‌شود، GSA-BGSA بهترین مقدار MSE را در برخی از تکرار آزمون‌ها به‌عنوان رقیب GSA داشته است، البته با در نظر گرفتن این نکته که این مقدار از MSE با تعداد گره‌های کمتری در لایه میانی شبکه عصبی به‌دست آمده است.

۴.۵. نتایج تقریب اطلاعات نوا به‌منظور سنتز گفتار فارسی در کاربرد تقریب توابع

به‌عنوان یک نمونه کاربردی جهت بررسی عملکرد شبکه عصبی بهینه‌شده با آمیختار GSA-BGSA برای تقریب توابع، کارکرد MLP بهینه در نقش سیستم تولید اطلاعات نوای گفتار در این بخش مورد ارزیابی قرار می‌گیرد. نوای گفتار مشتمل بر اطلاعات تغییرات فرکانس گام، انرژی و دیرش^۲ قطعات گفتار و درنگ بین قطعات می‌باشد که به‌کارگیری آن برای

گرفته شده است. پس از اعمال الگوریتم انتخاب ویژگی این تعداد به ۱۸ ویژگی کاهش می‌یابد. لذا تعداد ورودی‌ها به MLP برابر ۱۸ و تعداد خروجی‌های MLP نیز ۹ می‌باشد که مبین اطلاعات نوا در سطح هجای جاری بدین ترتیب می‌باشند:

حاصل گردآوری شده و در انتهای تکرارها ارزیابی می‌شوند. برای مشاهده جزئیات الگوریتم آمیختار GA-ACO و تنظیم پارامترهای مربوط به مراجع [۴۸، ۵۰] مراجعه شود. تعداد ویژگی‌ها در سطح کلمه و هجا به الگوریتم انتخاب ویژگی به ترتیب برابر ۱۶ و ۲۱ (در مجموع ۳۷ ویژگی) در نظر



شکل (۶): شمای بلوکی یک سیستم تبدیل متن به گفتار مشتمل بر واحد انتخاب ویژگی مبتنی بر الگوریتم آمیختار GA-ACO و واحد نوای مبتنی بر MLP بهینه شده با GSA-BGSA

ورودی با نرخ میانگین ادای حدود ۴ هجا در ثانیه بوده‌اند [۴۶]. لازم به ذکر است که سه نوع اطلاعات نوایی مطرح در این سیستم (تغییرات دوره تناوب گام، انرژی، و دیرش قطعات) دارای گستره‌های پویای^۲ مختلف هستند، لذا یک معیار اعوجاج به‌عنوان تابع هدف برای کمینه‌سازی (مانند MSE در رابطه (۲۲)) به‌صورت زیر در نظر گرفته شده است:

$$J(k) = \sum_{j=0}^3 \{T[p_j(k)] - O[p_j(k)]\}^2 + \{T[l_e(k)] - O[l_e(k)]\}^2 + \sum_{j=0}^3 \{T[d_j(k)] - O[d_j(k)]\}^2 \quad (23)$$

در رابطه (۲۳)، $p_j(k)$ مبین ضریب مرتبه^۳ لام در بسط چندجمله‌ای لژاندر گسسته مربوط به منحنی تغییرات دوره

چهار پارامتر مبین الگوی دوره تناوب گام (با استفاده از چهار ضریب اول بسط چندجمله‌ای لژاندر گسسته^۱ [۵۱])،

- یک پارامتر مبین مقدار لگاریتم انرژی،
- یک پارامتر مبین میزان دیرش هجا،
- یک پارامتر مبین میزان دیرش واکه درون یک هجا،
- یک پارامتر مبین زمان شروع واکه درون هجا، و
- یک پارامتر مبین میزان دیرش درنگ بین هجاها.

دادگان آموزشی برای سیستم مولد نوای مبتنی بر شبکه عصبی مشتمل بر ۴۰۰ جمله بوده که از انواع جملات خبری، پرسشی، امری و تعجبی و با تنوع مثبت، منفی، کوتاه و بلند در نظر گرفته شده‌اند. طول ۲۵۰ جمله از این تعداد کمتر از ۱۵ هجا و طول بقیه کمتر از ۵۰ هجا بوده است. جملات

۶. نتیجه‌گیری

یکی از مشکلات و نقاط ضعف شبکه‌های عصبی مرسوم، مشخص‌نودن ساختار بهینه و ندانستن این مسئله است که آیا ساختار در نظر گرفته‌شده، بهترین پاسخ را خواهد داد؟ این مقاله یک روش پیشنهادی بر اساس الگوریتم جستجوی گرانشی حقیقی و باینری را معرفی نموده است که دارای مزیت‌هایی نسبت به برخی از الگوریتم مشابه مورد بررسی بوده است. از مهم‌ترین مزیت‌های الگوریتم GSA-BGSA اتخاذ رویکردی پویا جهت یافتن ساختاری بهینه همزمان با به‌روزرسانی وزن‌های شبکه عصبی می‌باشد.

برای ارزیابی عملکرد الگوریتم پیشنهادی، نتایج دسته‌بندی روی سه دادگان استاندارد به نام‌های گل زنبق، سرطان سینه، و انواع شیشه به‌دست آمده و با روش‌های PSO، EBP و GSA (برای تعیین ضرایب وزن) و PSO-BPSO (برای تعیین همزمان ساختار و ضرایب وزن) مقایسه شده است. در ضمن برای بررسی قابلیت تعمیم شبکه عصبی طراحی‌شده در کاربردهای تقریب توابع، نتایج شبیه‌سازی یک سیستم مولد نوا به کمک شبکه عصبی MLP بهینه‌شده گزارش شده و مقادیر RMSE در تخمین پارامترهای نوا با مقادیر متناظر در صورت به‌کارگیری الگوریتم‌های EBP و PSO-BPSO مقایسه شده است. برای کاهش تعداد ورودی‌ها به شبکه، از الگوریتم آمیختار GA-ACO استفاده شده است.

نتایج شبیه‌سازی‌ها نشان داد که روش پیشنهادی با به‌کارگیری تعداد نرون‌های کمتر در لایه مخفی (به‌میزان ۲۵ تا ۶۸ درصد کاهش در تعداد این نرون‌ها در مقایسه با بسیاری از الگوریتم‌های مورد بررسی)، دقت دسته‌بندی و تقریب را به صورت قابل رقابتی (به‌ویژه هنگام کار با داده‌های آزمون) از خود ارائه می‌نماید.

در تحقیقات آتی سعی داریم با تکامل و ارتقای الگوریتم پیشنهادی از طریق به‌کارگیری نسخه‌های اصلاحی GSA، قابلیت‌های این روش را در کاربردهای دسته‌بندی و تقریب توابع مورد بررسی قرار دهیم.

تناوب گام در هجای k ام می‌باشد. اندازه لگاریتم انرژی هجای k ام نیز با $le(k)$ نشان داده شده است. برای هجای k ام نیز $d_0(k)$ مبین میزان دیرش هجا، $d_1(k)$ مبین میزان دیرش واکه درون هجا، $d_2(k)$ مبین زمان آغاز واکه درون هجا، و $d_3(k)$ مبین میزان دیرش درنگ بین هجاها می‌باشد. $T[p_j(k)]$ ، $T[le(k)]$ و $T[d_j(k)]$ نیز مقادیر مطلوب (هدف) پارامترهای مذکور هستند. $O[p_j(k)]$ ، $O[le(k)]$ و $O[d_j(k)]$ نیز خروجی‌های واقعی را در فرآیند آموزش نشان می‌دهند. هنجارسازی این پارامترها نیز چنین انجام می‌شود:

$$T[p_j(k)] = [p_j(k) - m_{pj}] / \sigma_p; \quad 0 \leq j \leq 3 \quad (24)$$

$$T[le(k)] = [le(k) - m_{le}] / \sigma_e \quad (25)$$

$$T[d_j(k)] = [d_j(k) - m_{dj}] / \sigma_{dj}; \quad 0 \leq j \leq 3 \quad (26)$$

در این روابط m_x و σ_x^2 به ترتیب مبین میانگین و واریانس پارامتر x هستند که برای هر یک از پارامترهای مذکور قابل محاسبه می‌باشند.

مقادیر جذر میانگین مربعات خطا^۱ (RMSE) پارامترهای نوایی (که روی ۱۲۰ جمله آزمون میانگین‌گیری شده) برای سه آزمایش (یعنی به‌کارگیری الگوریتم‌های EBP، PSO-BPSO و GSA-BGSA) در جدول (۸) گزارش شده‌اند. همان‌گونه که در جدول (۸) مشاهده می‌شود، الگوریتم آمیختار GSA-BGSA عملکرد بهتری در مقایسه با الگوریتم هم‌ارز خود در این پژوهش، یعنی PSO-BPSO، ارائه نموده و این در حالی است که تعداد نرون‌های پیشنهادی آن برای لایه مخفی نیز یکی کمتر است. الگوریتم‌های مذکور در مقایسه با الگوریتم استاندارد EBP نیز عملکرد بهتری را روی داده‌های آزمون با به‌کارگیری ساختار سبک‌تری برای شبکه ارائه می‌نمایند.

جدول (۸): RMSE تخمین پارامترهای نوایی در سه آزمایش مختلف برای بهینه‌سازی شبکه و انجام انتخاب ویژگی با الگوریتم GA-ACO

RMSE تخمین	RMSE تخمین زمان	RMSE تخمین دیرش	RMSE تخمین دیرش	RMSE تخمین لگاریتم انرژی (dB)	RMSE تخمین دوره تناوب گام (میلی ثانیه بر قاب)	تعداد گره‌های میانی	الگوریتم آموزش و/یا بهینه‌سازی ساختار
دیرش درنگ بین هجایی (میلی ثانیه)	آغاز واکه درون هجا (میلی ثانیه)	واکه درون هجا (میلی ثانیه)	هجا (میلی-ثانیه)	۴/۳۵	۱/۱۲	۱۴	EBP
۳۸/۱	۷/۰	۴۱/۱	۵۴/۶	۴/۱۸	۱/۰۷	۱۰	PSO-BPSO
۳۰/۲	۵/۸	۳۲/۹	۴۴/۳	۳/۹۰	۰/۹۸	۹	GSA-BGSA

optimization", Expert Systems with Applications, Vol. 38, pp. 4972-4977, 2011.

[6] Yao, X., "Evolving artificial neural networks", Proceedings of IEEE, Vol. 87, pp. 1423-1447, 1999.

[7] Leung, F. H. F., Lam, H. K., Ling, S. H., Tam, P. K. S., "Tuning of the structure and parameters of a neural network using an improved genetic algorithm", IEEE Transactions on Neural Networks, Vol. 14, pp. 79-88, 2003.

[8] Chester, D. L., "Why two hidden layers are better than one?", Proceedings of the International Joint Conference on Neural Networks, Vol. 1, pp. 1265-1268, San Diego, CA, USA, 1990.

[9] Zhang, X., "Time series analysis and prediction by neural networks", Optimization Methods and Software, Vol. 4, pp. 151-170, 1994.

[10] Salchenberger, L. M., Cinar, E. M., Lash, N. A., "Neural networks: A new tool for predicting thrift failures", Decision Sciences, Vol. 23, pp. 899-916, 1992.

[11] Subramanian, V., Hung, M. S., Hu, M. Y., "An experimental evaluation of neural networks for classification", Computers & Operations Research, Vol. 20, pp. 769-782, 1993.

[12] Kang, S., "An investigation of the use of feed-forward neural networks for forecasting", Ph.D. Thesis, Kent State University, Kent, OH, USA, 1991.

[13] Hecht-Nielsen, R., Neurocomputing, Addison-Wesley, Menlo Park, CA, USA, 1990.

سپاسگزاری

این تحقیق در قالب طرح پژوهشی "روش آمیختار از الگوریتم‌های تکاملی و هوش جمعی به منظور پیش‌پردازش داده‌ها در سیستم‌های سنتز گفتار" و با حمایت دانشگاه آزاد اسلامی - واحد تهران جنوب انجام پذیرفته است. نویسندگان این مقاله از هم‌فکری اعضای هیئت تحریریه مجله علمی ترویجی محاسبات نرم کمال سپاسگزاری را دارند.

مراجع

[1] Hinton, G. E., Williams, R. J., Rumelhart, D. E., "Learning representations by back-propagating errors", Nature, Vol. 323, pp. 533-536, 1986.

[2] Vosniakos, G. C., Benardos, P. G., "Optimizing feedforward artificial neural network architecture", Engineering Applications of Artificial Intelligence, Vol. 20, pp. 365-382, 2007.

[3] Gori, M., Tesi, A., "On the problem of local minima in back-propagation", IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 14, pp. 76-86, 1992.

[4] Zhang, J. R., Zhang, J., Lok, T. M., Lyu, M. R., "A hybrid particle swarm optimization-back-propagation algorithm for feedforward neural network training", Applied Mathematics and Computation, Vol. 185, pp. 1026-1037, 2007.

[5] Zhao, L., Qian, F., "Tuning the structure and parameters of a neural network using cooperative binary-real particle swarm

- algorithm*", Proceedings of the First Joint Congress on Fuzzy and Intelligent Systems, pp. 1-6, Ferdowsi University of Mashhad, Iran,
- [۲۵] دهباشیان، ظهیری، "آموزش شبکه عصبی MLP در طبقه‌بندی داده‌ها با استفاده از روش GSA"، فصلنامه مهندسی برق و مهندسی کامپیوتر ایران، سال ۸، شماره ۴، صفحه ۲۷۴-۲۶۷، ۱۳۸۹.
- [26] Sheikhan, M., Sharifi Rad, M., "Gravitational search algorithm-optimized neural misuse detector with selected features by fuzzy grids based association rules mining", Neural Computing and Applications, Vol. 23, pp. 2451-2463, 2013.
- [27] Yu, J., Wang, S., Xi, L., "Evolving artificial neural networks using an improved PSO and DPSSO", Neurocomputing, Vol. 71, pp. 1054-1060, 2008.
- [28] Qasem, S. N., Shamsuddin, S. M., "Radial basis function network based on time variant multi-objective particle swarm optimization for medical diseases diagnosis", Applied Soft Computing, Vol. 11, pp. 1427-1438, 2011.
- [29] Sheikhan, M., Jadidi, Z., "Flow-based anomaly detection in high-speed links using modified GSA-optimized neural network", Neural Computing and Applications, Vol. 24, pp. 599-611, 2014.
- [30] Subrahmanya, N., Shin, Y. C., "Constructive training of recurrent neural networks using hybrid optimization", Neurocomputing, Vol. 73, pp. 2624-2631, 2010.
- [31] Mirjalili, S. A., Mohd Hashim, S. Z., Moradian Sardroudi, H., "Training feedforward neural networks using hybrid particle swarm optimization and gravitational search algorithm", Applied Mathematics and Computation, Vol. 218, pp. 11125-11137, 2012.
- [32] Shen, W., Guo, X., Wu, C., Wu, D., "Forecasting stock indices using radial basis function neural networks optimized by artificial swarm algorithm", Knowledge-Based Systems, Vol. 24, pp. 378-385, 2011.
- [33] Kulluk, S., Ozbakir, L., Baykasoglu, A., "Training neural networks with harmony search algorithms for classification problems",
- [14] Eberhart, R. C., Kennedy, J., "Particle swarm optimization", Proceedings of the IEEE International Conference on Neural Networks, Vol. 4, pp. 1942-1948, Piscataway, NJ, USA, 1995.
- [15] Kennedy, J., Eberhart, R. C., "A discrete binary version of the particle swarm algorithm", Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics, Vol. 5, pp. 4104-4108, Piscataway, NJ, USA, 1997.
- [16] Lin, C. J., Hsieh, M. H., "Classification of mental task from EEG data using neural networks based on particle swarm optimization", Neurocomputing, Vol. 72, pp. 1121-1130, 2009.
- [17] Kiranyaz, S., Ince, T., Yildirim, A., Gabbouj, M., "Evolutionary artificial neural networks by multi-dimensional particle swarm optimization", Neural Networks, Vol. 22, pp. 1448-1462, 2009.
- [18] DAM, M., Saraf, D. N., "Design of neural networks using genetic algorithm for on-line property estimation of crude fractionator products", Computers and Chemical Engineering, Vol. 30, pp. 722-729, 2006.
- [19] Yao, X., Liu, Y., "A new evolutionary system for evolving artificial neural network", IEEE Transactions on Neural Networks, Vol. 8, pp. 694-713, 1997.
- [20] Rashedi, E., Nezamabadi-pour, H., Saryazdi, S., "GSA: A gravitational search algorithm", Information Sciences, Vol. 179, pp. 2232-2248, 2009.
- [21] Rashedi, E., Nezamabadi-pour, H., Saryazdi, S., "BGSA: Binary gravitational search algorithm", Natural Computing, Vol. 9, pp. 727-745, 2010.
- [۲۲] راشدی، نظام‌آبادی پور، "طراحی فیلترهای IIR به وسیله الگوریتم جستجوی گرانشی"، شانزدهمین کنفرانس مهندسی برق ایران، دانشگاه تربیت مدرس، تهران، ایران، ۱۳۸۷.
- [۲۳] راشدی، نظام‌آبادی پور، "انتخاب ویژگی با استفاده از الگوریتم جستجوی گرانشی"، سومین کنفرانس فناوری اطلاعات و دانش، دانشگاه فردوسی، مشهد، ایران، ۱۳۸۶.
- [24] Rashedi, E., Nezamabadi-pour, H., Saryazdi, S., Farsangi, M. M., "Allocation of static var compensator using gravitational search

- [43] Zhao, Z. S., Feng, X., Lin, Y. Y., Wei, F., Wang, S. K., Xiao, T. L., Cao, M. Y., Hou, Z. G., "Evolved neural network ensemble by multiple heterogeneous swarm intelligence", *Neurocomputing*, Vol. 149, Part A, pp. 29-38, 2015.
- [44] Sarafrazi, S., Nezamabadi-pour, H., Saryazdi, S., "Disruption: A new operation in gravitational search algorithm", *Scientia Iranica*, Vol. 18, pp. 539-548, 2011.
- [45] <http://www.ics.uci.edu/~mllearn/MLRepository.htm>, S. Hettich, C. L. Blake, C. J. Merz, D. J. Newman, Department of Information and Computer Science, University of California, Irvine, CA, USA, 1998 [Online].
- [۴۶] شیخان، نصیرزاده، دفتریان، "طراحی و پیاده‌سازی سیستم تبدیل متن به‌گفتار طبیعی برای زبان فارسی"، مجله دانشکده مهندسی دانشگاه فردوسی مشهد، سال ۱۷، شماره ۲، صفحات ۴۸-۳۱، ۱۳۸۴.
- [۴۷] شیخان، "تولید خودکار نوای گفتار به‌کمک مدل آمیختار عصبی-آماری با امکان انتخاب واحد در سنتز"، مجله مهندسی پزشکی زیستی، دوره اول جدید، شماره ۳، صفحات ۲۴۰-۲۲۷، ۱۳۸۶.
- [48] Sheikhan, M., "Synthesizing suprasegmental speech information using hybrid of GA-ACO and dynamic neural network", *Proceedings of 5th Conference on Information and Knowledge Technology*, pp. 175-180, University of Shiraz, Iran, 2013.
- [49] Sheikhan, M., "Generation of suprasegmental information for speech using a recurrent neural network and binary gravitational search algorithm for feature selection", *Applied Intelligence*, Vol. 40, pp. 772-790, 2014.
- [50] Sheikhan, M., Mohammadi, N., "Neural-based electricity load forecasting using hybrid of GA and ACO for feature selection", *Neural Computing and Applications*, Vol. 21, pp. 1961-1970, 2012.
- [51] Chen, S. H., Wang, Y. R., "Vector quantization of pitch information in Mandarin speech", *IEEE Transactions on Communications*, Vol. 38, pp. 1317-1320, 1990.
- Engineering Applications of Artificial Intelligence, Vol. 25, pp. 11-19, 2012.
- [34] Pasti, R., De Castro, L. N., "The influence of diversity in an immune-based algorithm to train MLP networks", *Lecture Notes in Computer Science*, Vol. 4628, pp. 71-82, 2007.
- [35] Chao, M., Xin, S. Z., Min, L. S., "Neural network ensembles based on copula methods and distributed multi-objective central force optimization algorithm", *Engineering Applications of Artificial Intelligence*, Vol. 32, pp. 203-212, 2014.
- [36] Zeng, Q., Huang, H., "A stable and optimized neural network model for crash injury severity prediction", *Accident Analysis & Prevention*, Vol. 73, pp. 351-358, 2014.
- [37] Yu, F., Xu, X., "A short-term load forecasting model of natural gas based on optimized genetic algorithm and improved BP neural network", *Applied Energy*, Vol. 134, pp. 102-113, 2014.
- [38] Unal, M., Onat, M., Demetgul, M., Kucuk, H., "Fault diagnosis of rolling bearings using a genetic algorithm optimized neural network", *Measurement*, Vol. 58, pp. 187-196, 2014.
- [39] Matias, T., Souza, F., Araújo, R., Antunes, C. H., "Learning of a single-hidden layer feedforward neural network using an optimized extreme learning machine", *Neurocomputing*, Vol. 129, pp. 428-436, 2014.
- [40] Dheeba, J., Singh, N. A., Selvi, S. T., "Computer-aided detection of breast cancer on mammograms: A swarm intelligence optimized wavelet neural network approach", *Journal of Biomedical Informatics*, Vol. 49, pp. 45-52, 2014.
- [41] Duan, H., Huang, L., "Imperialist competitive algorithm optimized artificial neural networks for UCAV global path planning", *Neurocomputing*, Vol. 125, pp. 166-171, 2014.
- [42] Mao, X., Sun, L., Hui, G., Xu, L., "Modeling research on wheat protein content measurement using near-infrared reflectance spectroscopy and optimized radial basis function neural network", *Journal of Food and Drug Analysis*, Vol. 22, pp. 230-235, 2014.