کاربرد شبکه‌های عصبی در عیب‌یابی سیستمهای توزیع الکتریکی

سهراب خان محمدی - علی‌اصغر حبیب زاده
دانشگاه تبریز

چکیده :

موضوع عیب‌یابی سیستمهای مختلف به دلایلی چون ایمنی (Safety) ، افزایش کیفیت و راندمان ، کاهش هزینه‌های بازرسی و غیره یکی از مسائل مهم مورد نظر در سیستمهای کنترل می باشد. در این رابطه روشهای مختلفی وجود دارند که عمدتا" بر مبنای پارامترهای اقتصادی و بهینه کردن فرایند عیب‌یابی ارائه شده‌اند. از آن جمله می توان به روشهای برنامه‌ریزی پویا (Dynamic Programming) اشاره نمود. یکی از دیدگاههای جدید در این ارتباط استفاده از شبکه‌های عصبی مصنوعی (Artificial Neural Networks) است ، که در اشکال مختلف مورد استفاده قرار میگیرد. در این مقاله بررسی شبکه‌های عصبی چند لایه (Multy layer) بعنوان یک سیستم هوشمند برای عیب‌یابی سیستمهای توزیع مورد نظر می‌باشد. برای این منظور از روش تصمیم‌گیری در شبکه‌های عصبی رقابتی (Competitive) برای پیاده نمودن سیستم انتشار تصمیم گیری در درخت عیب‌یابی (Diagnostic tree) استفاده می‌شود. در این روش رسم درخت عیب‌یابی و حرکت در شاخه های آن به حرکت در لایه‌های مختلف تصویر می شود.

شرح مقاله :

پیشرفتهای اخیر در طراحی سیستمهای بزرگ و پخش شده و گستردگی قابل توجه

آنها همراه با مسائل اقتصادی که در جوارشان مطرح است ، مهندسین سیستم را ناگزیر می سازد که روشهایی را برای تست و عیب‌یابی اتوماتیک سیستمها در زمان حقیقی طراحی کنند . در سیستمهای عیب‌یاب اتوماتیک جدید مسئله رفع عیب در زمان حقیقی و سریع برای دستیابی به کیفیت عملیاتی مورد نظر هدف اصلی است [۲،۱] و برای این منظور روشهای مختلفی ارائه شده است . یکی از مسائل عمده در طراحی سیستمهای عیب‌یاب مسئله هزینه عیب‌یابی در سطوح مختلف سیستمها است. برای مثال هزینه‌عیب یابی در سطح عناصر (مقره‌ها) بمراتب کمتر از هزینه عیب‌یابی در سطح مجموعه ترانس در یک مرکز توزیع محلی است . ولی از طرف دیگر زمان (در نتیجه هزینه) یافتن عنصر معیوب در سیستم متشکل از عناصر متعدد قابل توجه می باشد . بنابراین سعی می شود برای یافتن عامل اصلی عیب حتی‌المقدور عناصر کمتری مورد تست قرار گیرند .

روشهایی که تا بحال برای تست سیستمها بصورت ترتیبی بکار رفته است به دو گروه مختلف تقسیم می شوند :

الف : استفاده از برنامه‌ریزی پویا Dynamic Programming (DP)

ب : استفاده از روشهای ادراکی Heuristic

روش برنامه‌ریزی پویا DP با توجه به زمان محاسباتی طولانی و مقدار حافظه مورد نیاز آن در زمان حقیقی غیر عملی است [۳] . البته این روش می تواند در جوار روشهای ادراکی به عنوان یک روش مکمل مورد استفاده قرار گیرد . جوآنه بچتادووکان و همکارانش در مقاله خود تحت عنوان " مدلهای پویای درخت عیب‌یابی سیستمهای کامپیوتر " مدلهای مختلفی از سیستمهای معرفی شده عیب‌یاب را با یکدیگر مقایسه کرده‌اند [۴].

امروزه با پی بردن به امکانات یادگیری و محاسباتی سریع شبکه‌های عصبی ، این نظریه را در اغلب سیستمهای کنترل مورد آزمایش و تجزیه و تحلیل قرار می‌دهند. مسئله عیب‌یابی نیز از جمله مسائلی است که کارشناسان سیستمهای هوشمند را برای استفاده از شبکه‌های عصبی مورد توجه قرار داده است .در این زمینه می‌توان به کارهای انجام یافته در مراجع [۵ و۶] اشاره کرد .

در این مقاله نظریه جدیدی از کاربرد شبکه‌های عصبی رقابتی برای عیب‌یابی سیستمها معرفی می شود . در این روش برخلاف روشهای مرسوم ترتیب بندی حالتهای مختلف بردار احتمالات و ماتریس تست به صورت مقادیر معلوم و از پیش تعیین شده به سیستم تغذیه نمی شوند بلکه شبکه عیب‌یاب از یک حالت کاملا" گنگ در

مورد منابع عیب شروع کرده و پس از آموزشهای متعدد ، با توجه به موفقیت یا شکست در هر تست ، مقادیر احتمالی حالتهای عیب‌دار را که بصورت وزنهای اتصالات شبکه‌های عصبی در نظر گرفته شده‌اند به هنگام در می‌آورد. بدین ترتیب روش تجربه‌اندوزی یک متخصص با تجربه در عیب‌یابی سیستمها به شبکه عصبی منتقل می‌شود . با توجه به عمومی بودن روش ارائه شده شبکه عیب‌یاب سیستم توزیع بصورت عمومی مطرح و مورد بررسی قرار می‌گیرد .

۱- برنامه‌ریزی تست :

مسئله ترتیب‌بندی یا برنامه‌ریزی تست با مجموعه چهارتایی $\{S,P,T,C\}$ مشخص می‌شود. در این مجموعه $S = \{s_0,\ldots,s_n\}$ یک مجموعه محدود از حالتهای سیستم است که در آن s_0 نشان دهنده حالت بدون عیب و s_i $(i \le i \le n)$ ها نشان دهنده حالتهای مختلف عیب‌دار است.

$P = [P(S_0),\ldots,P(S_m)]^T$ بردار احتمالات حالتهای مختلف از پیش تعیین شده است. $T =\{t_1,\ldots,t_n\}$ نشان دهنده مجموعه محدود تستهای ممکن و $C =[c_1,\ldots,c_n]^T$ بردار هزینه‌های تست است، که می تواند بر اساس زمان ، نیروی لازم و یا سایر هزینه‌های مختلف باشد. هر تست t_j , $1 \le j \le n$ با یک بردار ستونی d_j با $(m+1)$ عنصر مشخص می‌شود.

بدین ترتیب که اگر $d_{ij} = 1$ ، یعنی یک حالت عیب‌دار i با تست t_j کشف می‌شود. با این تفصیل مسلما" در هر ستونی $d_{0j} = 0$. در روش ترتیب‌بندی تست معمولا" فرض می‌شود که در هر لحظه زمانی فقط یکی از حالتهای عیب‌دار اتفاق می‌افتد. ماتریس دوتایی $D = [\ d_{ij}\ ]$ بصورت اطلاعات تجربی موجود در فایل ذخیره شده است . مسئله برنامه‌ریزی معمولا" برمی‌گردد به طراحی روشی که قادر باشد با استفاده از تستهای T عیبهای S را با حداقل هزینه (زمان) ممکن ،که از رابطه زیر بدست می‌آید ، مشخص کند [۳] .

$$J= P^TAC= \sum_{i=0}^{m} \sum_{j=1}^{n} a_{ij}P(S_i)C_i$$

که در آن $A=(a_{ij})$ یک ماتریس دوتایی binary است بطوریکه هر عنصر a_{ij} آن ، اگر تست j در مسیر یافتن حالت s_i بکار برده شود برابر با یک و در غیر اینصورت برابر با صفر خواهد بود . جدول زیر یک سیستم عیب‌یاب را با ۵

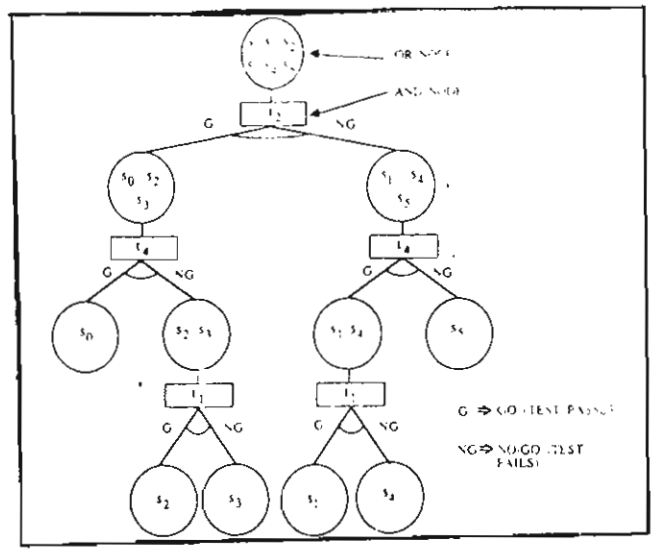
حالت عیب‌دار و مجموعه بردار تست (۵) حالتی همراه با بردار هزینه و احتمال وجود هر یک از حالتها نشان می‌دهد.

احتمالات	تستها t_1	t_2	t_3	t_4	t_5	حالتهای عیب‌دار
	هزینه‌ها 1	1	1	1	1	
0.70	0	0	0	0	0	S_0
0.01	0	1	0	0	0	S_1
0.02	0	0	1	1	0	S_2
0.10	1	0	0	1	1	S_3
0.05	1	1	0	0	0	S_4
0.12	1	1	1	1	0	S_5

جدول ۱- ماتریس تست - احتمال حالتهای عیب‌دار و هزینه‌های تست

درخت آلگوریتم تست بهینه برای جدول (۱) در شکل (۱) آمده است . در این شکل هر گره or مجموعه حالتهای ممکن در هر شاخه را نشان داده و گره‌های and نماینده تست بکار رفته هستند . گره ریشه شامل تمام حالتها است . اگر سیستم از تستی مانند تست z بگذرد (G) ، یعنی یکی از حالتهای غیر قابل کشف توسط آن تست مشاهده شده است و اگر تست مذکور پاسخ منفی بدهد (NG) ، یعنی یکی از حالتهای قابل کشف با آن تست رخ داده است .

برای مثال با توجه به جدول (۱) تست (۲) قادر به کشف حالتهای S_5 ، S_4، S_1 بوده ولی قادر نیست عیبهای واقع در حالتهای S_3 ، S_2 ، S_0 را کشف نماید. بنابراین در درخت تست شکل (۱) پاسخ مثبت این تست نشان دهنده این است که یکی از حالتهای عیب‌دار S_3 ، S_2 ، S_0 رخ داده است و پاسخ منفی آن نشان دهنده رخداد یکی از حالتهای S_5 ، S_4 ، S_1 می‌باشد .با حرکت به شاخه‌های پایین تر درخت تعداد حالتهای ممکن در هر گره or کاهش می‌یابد(واضحتر می‌شود) و نهایتا " در برگهای درخت فقط یک حالت برای تست باقی می‌ماند. بدین ترتیب با حرکت در شاخه‌های مختلف درخت و انجام تستهای متوالی عنصر عیب‌دار کشف می‌شود.

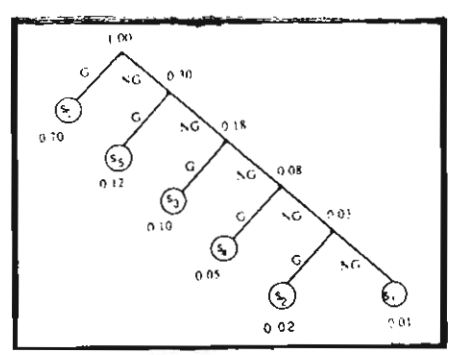


شکل ۱ - درخت تست بهینه برای سیستم جدول (۱) [۲]

روشهای مختلفی برای یافتن ترتیب بهینه تستها وجود دارد که از آن جمله می‌توان به بهره‌گیری از روش هوفمن [۲] اشاره کرد . در این روش فرض می‌شود که با هر تست t_i ای حالت عیب‌دار s_i کشف می‌شود . بدین ترتیب بهترین ترتیب برای انجام تستها ترتیبی است که از قانون اولویت‌بندی زیر پیروی کند.

$$P(S_1)/C_1 \geq P(S_2)/C_2 \geq \ldots \geq P(S_m)/C_m \qquad (۲)$$

شکل (۲) درخت دوتایی روش هوفمن را برای ترتیب‌بندی تستهای مختلف جدول (۱) نشان می‌دهد .



شکل ۲ - درخت دوتایی هوفمن برای عیب‌یابی سیستم جدول (۱)

با توجه به شکل ملاحظه می‌شود که ترتیب‌بندی تستها بر اساس اولویت‌بندی تست حالت با احتمال وقوع عیب بیشتر استوار است .

در این مقاله الگوی هوفمن به گونه‌ای دیگر ، با رده‌بندی حالتهای مختلف عیب‌دار در سطوح مختلف و با استفاده از شبکه‌های عصبی رقابتی بکار برده می‌شود.

۲- شبکه‌های عصبی مصنوعی و کاربرد آن در عیب‌یابی سیستمها :

یک شبکه عصبی مصنوعی متشکل است از تعداد زیادی سلولهای محاسباتی که به لایه‌های مختلف تقسیم‌بندی شده‌اند . گرچه شبکه‌های عصبی مصنوعی به هیچ وجه کارآیی انسان را ندارند ولی با این حال بخاطر قابلیت یادگیری Learning و محاسبات موازی Parallel و سریع ، امکانات جالبی برای بازشناسی الگوها و طبقه بندی آنها ارائه می‌دهند . در این بخش شبکه‌های عصبی تک لایه و چند لایه و نیز شبکه‌های رقابتی معرفی می‌شوند و در بخش بعد مدل جدیدی از یادگیری به نام انتشار لایه‌ای Layer Propagation معرفی می‌گردد که در آن شبکه عصبی با استفاده از بهنگام درآوردن اتصالات هر لایه پس از نتیجه تست هر سطح از حالتهای عیب‌دار ، برای عیب‌یابی سریعتر و خطای کمتر در انتخاب حالتها ، آموزش داده می‌شود.

۲-۱- شبکه‌های عصبی تک لایه :

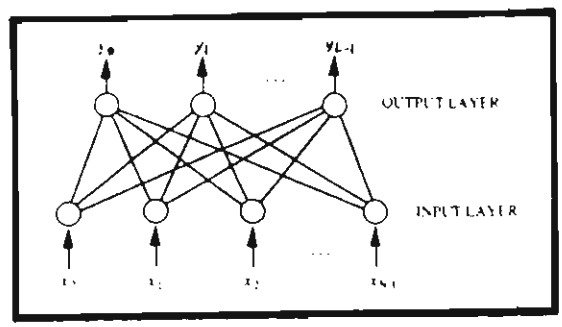
یک شبکه عصبی تک لایه مشتمل است بر یک لایه ورودی و یک لایه خروجی . لایه ورودی به صورت بافر Buffer عمل کرده و محتوی داده‌های سیستم است . سلولهای اصلی محاسباتی در لایه خروجی هستند . خروجیهای این شبکه از رابطه زیر بدست می‌آیند .

$$Y_j = f\left(\sum_{i=0}^{N-1} W_{ij} X_i - \theta_j \right) \qquad j = 0,\ldots,L-1 \qquad (۳)$$

که در آن X_i ها ورودیها ، Y_j ها خروجیها ، W_{ij} وزن اتصالات ، θ_j ها مقادیر بایاس bias یا انباره داخلی threshold بوده و تابع f نشان دهنده

قانون فعالیت activation rule سلولها است که معمولا" یکی از چهار تابع پله‌ای ، شیب ، خطی و سیگموید است .

شکل (۳) یک شبکه عصبی تک لایه را نشان می دهد . در این شکل تعداد سلولهای لایه ورودی N و تعداد عناصر لایه خروجی L فرض شده است [۶].

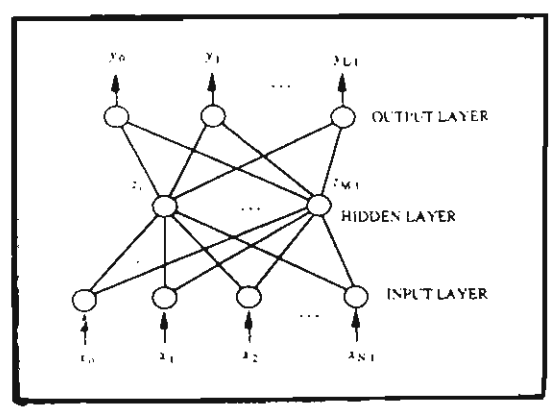


شکل ۳ - شبکه عصبی تک لایه

۲-۲- شبکه‌های عصبی چند لایه :

شبکه‌های عصبی تک لایه قادرند فقط الگوهایی را طبقه‌بندی کنند که بصورت خطی قابل جدا سازی هستند [۷] . برای طبقه‌بندی سیستمهای پیچیده‌تر از شبکه‌های عصبی چند لایه استفاده می شود . شبکه‌های چند لایه شامل یک لایه ورودی ، یک لایه خروجی و یک یا چند لایه مخفی هستند .

شکل (۴) یک شبکه عصبی مصنوعی سه لایه را نشان می دهد .



شکل ۴ - شبکه عصبی چند لایه

در این حالت خروجی سلولهای لایه مخفی از رابطه زیر بدست می‌آید :

$$Z_j = f\left(\sum_{i=0}^{N-1} W_{ij}X_i - \theta_j \right) \qquad j = 0,\ldots,M-1 \qquad (۴)$$

که در آن ها x_i ها ورودیهای شبکه ، z_j ها خروجیهای لایه مخفی ، W_{ij} ها وزن اتصالات بین عناصر لایه ورودی و لایه مخفی و θ_j مقادیر بایاس یا انباره داخلی سلولهای لایه مخفی هستند . در این شکل تعداد سلولهای لایه مخفی M تا در نظر گرفته شده‌است . خروجیهای شبکه از رابطه زیر بدست می‌آیند :

$$Y_j = f\left(\sum_{j=0}^{M-1} W_{jk}Z_j - \theta_k \right) \qquad K = 0,\ldots,L-1 \qquad (۵)$$

که در آن W_{jk} ها وزنهای اتصالات بین سلولهای لایه مخفی و لایه خروجی و θ_k ها مقادیر بایاس یا انباره داخلی سلولهای لایه خروجی می باشند .

۳-۲- انتشار بازگشتی :

برای اینکه شبکه عصبی قادر باشد الگوهای مختلف را طبقه‌بندی کند باید وزن اتصالات موجود شبکه بگونه‌ای بهنگام درآیند که شبکه بتواند طبقه‌بندی مورد نظر را به بهترین وجه انجام دهد .

روشی که برای این منظور بکار می رود آلگوریتم انتشار بازگشتی یا قانون دلتای تعمیم یافته Generalized delta rule (GDR) است [۸ ، ۹] .

۴-۲- شبکه‌های عصبی رقابتی :

شبکه‌های عصبی رقابتی مجموعه‌ای از سلولهای لایه‌ای هستند که در آنها سلولهای هر لایه برای فعال شدن به رقابت می پردازند . در فرایند رقابت ،هر سلولی سعی می کند در رقابت پیروز شده و پس از پیروزی ، از فعال شدن سلولهای دیگر مجموعه جلوگیری کند . قانون فعالیت هر سلول رقابتی از رابطه زیر پیروی می‌کند.

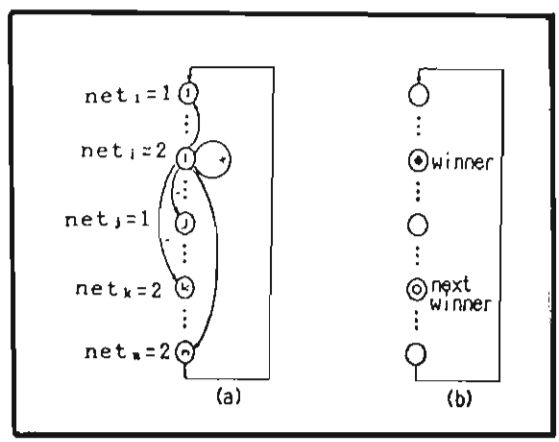
$$Y_j = \begin{cases} 1 \ , \ net_j > 0 \ \text{ and } net_j = \max\{net_i\} \\ 0 \ , \ \text{Otherwise} \end{cases} \qquad (۶)$$

که در آن قانون فعالیت سلولها پله‌ای در نظر گرفته شده و مقادیر خالص آنها بصورت زیر بدست می آید .

$$Net_j = \Sigma X_i W_{ij} + \theta \qquad (۷)$$

در این رابطه net_j مقدار خالص سلول j ، x_i خروجی سلول i در لایه قبلی ، W_{ij} وزن اتصال سلول i به سلول j و θ مقدار انباره داخلی است. در این بررسی مقدار انباره داخلی صفر در نظر گرفته شده است.

شکل (۵) یک لایه از سلولهای رقابتی را نشان می‌دهد.



شکل ۵ - یک لایه از سلولهای رقابتی

(a) - اتصالات محرک و بازدارنده (b) - سلول با مقدار خالص بیشتر برنده می‌شود

۳- شبکه عصبی سیستم عیب‌یابی :

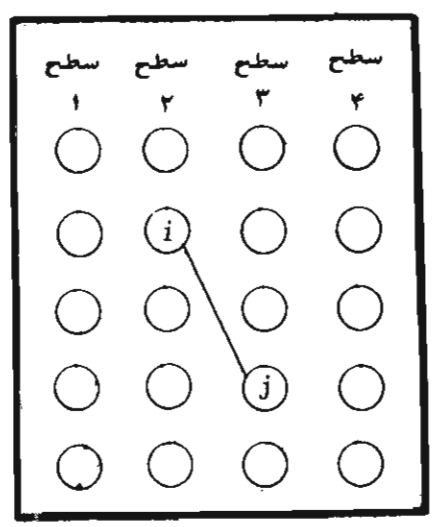
تیمو سورسا و سایر همکارانش در مقاله خود تحت عنوان " شبکه‌های عصبی در تشخیص عیب فرایندها " [۶] کاربرد سه نوع شبکه عصبی استاندارد را ، که در آنها وزن اتصالها با روش GDR یا خلاف انتشار Counter Propagation تغییر می یابند ، مورد بررسی قرار داده‌اند .

در این مقاله روش جدیدی از کاربرد شبکه‌های عصبی برای عیب‌یابی سیستمها معرفی می شود . همانگونه که قبلا" ذکر گردید بخاطر عمومی بودن روش ، آنرا بصورت کلی مورد بررسی قرار می دهیم نه مختص سیستمهای توزیع .

در این بررسی از یک شبکه عصبی ۴ لایه که هر کدام دارای ۵ سلول هستند استفاده شده است . شکل (۶) این سیستم را نشان می دهد . هرلایه از این شبکه نشان دهنده سطح مورد نظری از سیستم است که امکان دارد دچار عیب شود . وزن اتصالها بصورت زیر تعریف می شوند .

$W_{ij} = P($ حالت S_j منبع ایجاد عیب در حالت S_i است $)$

در حالت کلی تعداد حالتهای (زیر سیستمهای) هر لایه از لایه قبلی بیشتر بوده و یک حالت ممکن است منبع عیب چندین حالت در لایه قبلی باشد .



شکل ۶ - شبکه عصبی سیستم عیب‌یابی

۱-۳- روش یادگیری :

در این قسمت روش خاصی از یادگیری تعدیل شده برای سیستمهای عیب‌یابی معرفی می شود .بدین ترتیب که در مرحله اول وزنها مقادیر دلخواه انتخاب می‌شوند . در سیستمهایی چون بازیابی اطلاعات یا طبقه‌بندی الگوها معمولا" این مقادیر تصادفی و اغلب بین صفر و یک هستند .

در اینجا با توجه به اینکه قبل از آموزش تاثیر وجود عیب در هر حالت در حالتهای سطح بالاتر کاملا" گنگ و تمام حالتها یکسان می مانند ، وزن اولیه تمام اتصالات 0.5 (احتمال ۵۰ درصد) در نظر گرفته می شوند . در صورت وقوع یک عیب در زیر سیستم i ام سطح اول (بالاترین سطح) ، سلول i ام لایه (۱) فعال شده و بقیه سلولها را غیر فعال می کند . سپس با توجه به وزن اتصالات مقادیر خالص هر یک از سلولهای لایه (۲) محاسبه شده و پس از فرایند رقابت سلول z ام این لایه که بیشترین مقدار خالص را دارد فعال می شود . فعال شدن این سلول نشان دهنده این است که حالت S_j بعنوان منبع عیب حالت کاندید شده است. پس از تست این حالت اگر عملا" عیبی مشاهده شود وزن اتصال سلول i به سلول z افزایش می یابد (مقدار آن به یک نزدیکتر می شود) و اگر عیبی مشاهده نشود وزن آن کاهش یافته (به صفر نزدیکتر شده) و کاندید بعدی تست می شود. این عمل تا رسیدن به سطح ۴ ادامه می یابد .

۲-۳- تعدیل اتصالات با توجه به نتیجه تست :

در صورت استفاده از GDR تغییر وزن از رابطه زیر بدست خواهد آمد .

$$\hat{W}_{ij} = W_{ij} + \alpha\,(X_{jd} - X_{ja})W_{ij} \qquad (۸)$$

که در آن $\hat{W}_{ij}$ وزن جدید ، W_{ij} وزن قدیم ، α ضریب تغییر ، X_{jd} سطح فعالیت مورد انتظار سلول z و X_{ja} سطح فعالیت واقعی آن است که از روابط زیر بدست می‌آیند :

$$X_{ja} = \begin{cases} 1 & \text{اگرحالت z دارای عیب باشد} \\ 0 & \text{غیره} \end{cases} \qquad (۹)$$

$$X_{jd} = \begin{cases} 1 & \text{اگرحالت z منتخب برای منبع عیب باشد} \\ 0 & \text{غیره} \end{cases}$$

با توجه به رابطه (۸) در صورتیکه یک حالت کاندید شده بعنوان منبع عیب بعد از تست تائید شود خواهیم داشت :

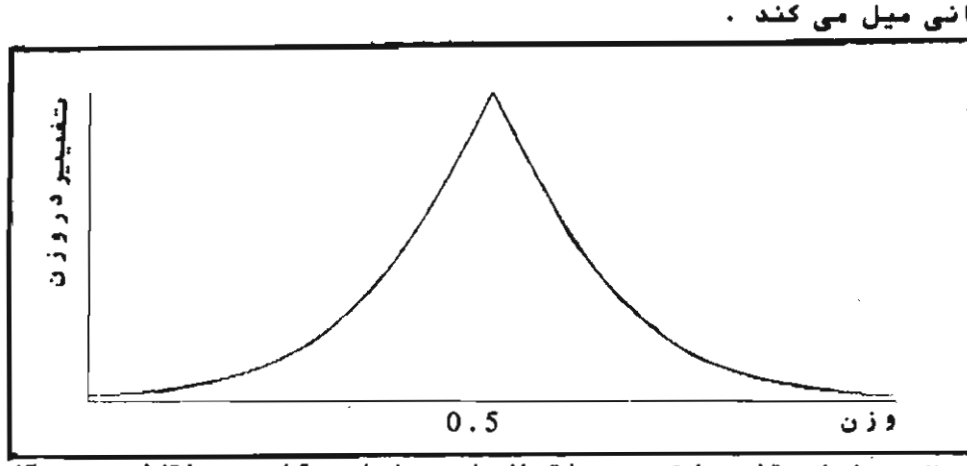
$$X_{jd} = X_{ja} = 1 \quad \Longrightarrow \quad \hat{W}_{ij} = W_{ij} \qquad (10)$$

یعنی پس از تائید صحت تشخیص وزن اتصال مربوطه ثابت می ماند در صورتیکه از دیدگاه یک کارشناس خبره این مسئله بعنوان یک تجربه مثبت تلقی شده و در صورت تکرار عیب مزبور حالت ز با احتمال بیشتری بعنوان منبع عیب در نظر گرفته می شود . برای پیاده کردن این منظور ، بصورت هوش مصنوعی در سیستم شبکه عصبی برای تعدیل وزن اتصالها، از تابع زیگموید بصورت زیر استفاده می شود .

$$\hat{W}_{ij} = W_{ij} \pm \alpha\left(1 - \frac{1}{1 + e^{-\beta(W_{ij} - 0.5)}}\right) \qquad \text{if } W_{ij} \geq 0.5$$

$$\hat{W}_{ij} = W_{ij} \pm \frac{\alpha}{1 + e^{-\beta(W_{ij} - 0.5)}} \qquad \text{if } W_{ij} < 0.5 \qquad (11)$$

که در آن علامت مثبت برای وقتی است که تست موفق شود و علامت منفی برای وقتی که تست موفق نشود . در این روابط ضریب α برای تعدیل وزنها بصورت آرام Smooth بکار رفته ($o<\alpha<1$) و ضریب β ، ($\beta>1$) برای افزایش اثر وزن پیشین هراتصال در تعدیل آن است . شکل (۷) منحنی تغییرات وزن اتصالها را به ازای مقادیر آنها نشان می دهد. همانگونه که از شکل پیداست شیب تغییرات وقتی که مقدار وزن نزدیک o.5 است بیشتر از نقاط کرانی (نزدیک صفر یا یک) می باشد. علت این انتخاب این است که در اثر آموزش شبکه وزن اتصالها به یکی از این نقاط کرانی میل می کند .



شکل ۷ - منحنی تغییرات وزن اتصالها به ازای مقادیر مختلف وزن آنها

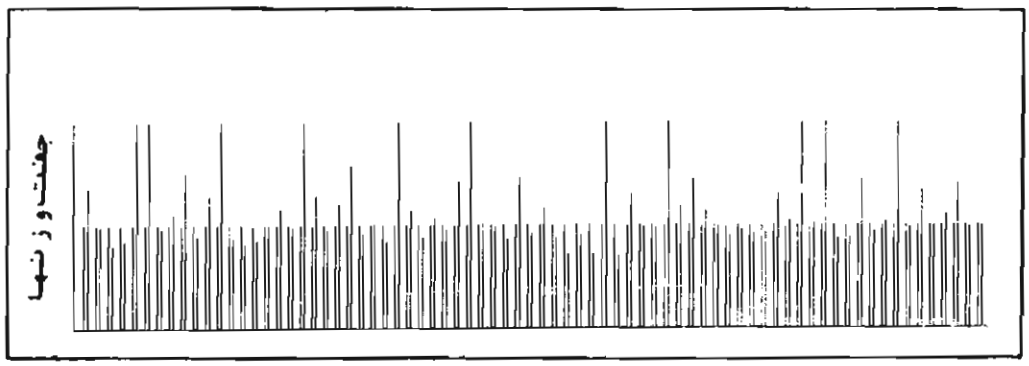
۴- نتایج مدلسازی :

روش پیشنهاد شده برای یک شبکه عصبی ۴ لایه و هر لایه شامل ۵ سلول مدلسازی شده است . عیبهای مختلف بصورت تصادفی به سیستم اعمال شده و کاندیدهای منابع عیب بترتیب در لایه های بعد تست شده و با توجه به نتیجه تست وزن اتصالها بهنگام درآمده‌اند . شبکه ۴۰۰۰ بار آموزش داده شده است . نتیجه آموزشها برای دو حالت مختلف در شکلهای (۸) و (۹) نشان داده می شود.

در شکل (۸) وزن اولیه کلیه اتصالها مساوی و برابر با 0.5 در نظر گرفته شده است . در شکل (۹) مقادیر اولیه وزنها متفاوت و براساس تجربیات قبلی فرض شده است . در هر دو شکل قسمت (الف) وزن اتصالها را در شرایط اولیه و پس از آموزش بصورت جفت بارها نشان می دهد . در هر جفت ، بار طرف چپ شرایط اولیه و بار طرف راست بعد از آموزش است . قسمت (ب) احتمال موفقیت در رسیدن به منبع اصلی عیب در سطح ۳ را نشان می دهد . مشاهده می‌شود که با افزایش تعداد آموزشها این احتمال به سمت یک میل می کند. قسمت (ج) دراین شکلها نشان دهنده تعداد تستهای ناموفق قبل از رسیدن به منبع اصلی عیب در سیستم است . با مقایسه دو شکل (۸) و (۹) ملاحظه می شود که با شروع از یک شرایط اولیه واضحتر سرعت آموزش شبکه افزایش قابل ملاحظه‌ای می‌یابد .

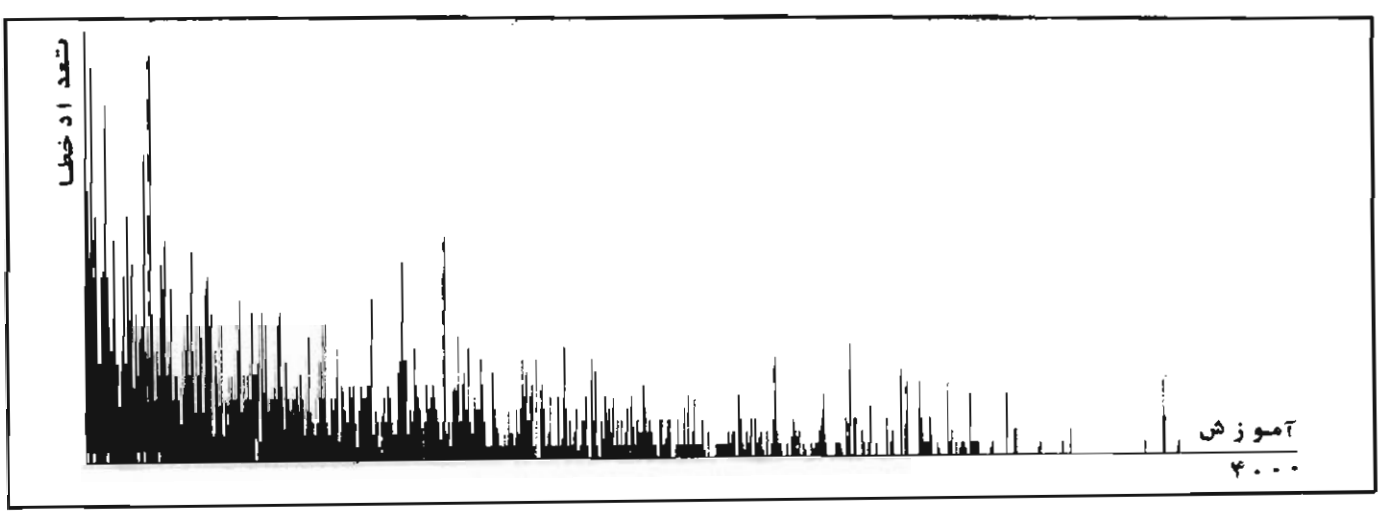
نتیجه :

در این مقاله از یک شبکه عصبی رقابتی برای عیب‌یابی سیستمها استفاده شد. بخاطر عمومی بودن روش ، آنرا بصورت کلی بررسی نموده و شبکه‌های توزیع را بعنوان یک مسئله مجرد در این رابطه مطرح نکردیم . برای آموزش شبکه عصبی به نحوی که تجربه‌اندوزی یک متخصص خبره را بصورت هوش مصنوعی تصویر کند از تابع زیگموید برای تعدیل وزن اتصالها استفاده کردیم ، در حالیکه این تابع بطور سنتی بعنوان قانون فعالیت سلولها بکار می رود . نتایج بررسیها نشان میدهد که شبکه‌های عصبی مصنوعی بخاطر قابلیت آموزش و محاسبات موازی در آنها ، آلترناتیو قابل توجهی برای استفاده در طراحی سیستمهای عیب‌یاب هستند . مطالعات انجام یافته در این مقاله نشان می دهد که هر چه اطلاعات اولیه شبکه واضحتر و منطبق با تجربیات قبلی باشد ، سرعت آموزش و در نتیجه کارآیی آن بیشتر خواهد بود .



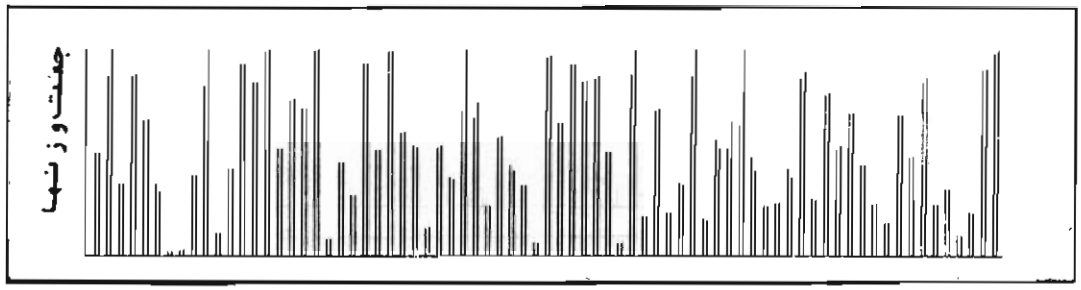
الف

ب

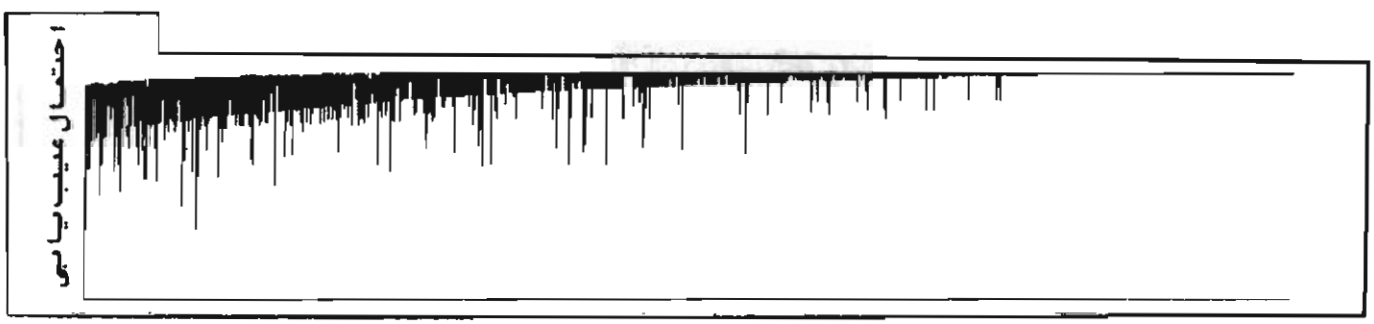


ج

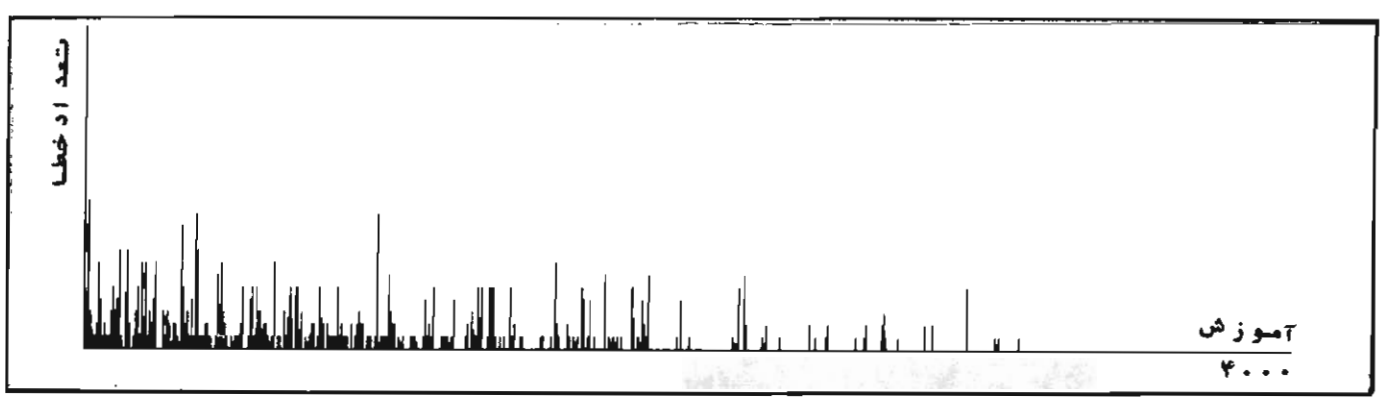
شکل ۸ - شرایط اولیه "کاملا" کنگ (وزن اولیه تمام اتصالها 0.5)



الف



ب



ج

شکل ۹ - شرایط اولیه بر اساس تجربیات قبلی

منابع :

1- T.W. Williams, K.P. Parker, "Design for testability a survay,"Proc. IEEE, Vol. 71, No.1, pp 98-112, Jan.1983.

2- K.R. Pattipati, M.G. Alexandridis," Application of heuristic search and information theory to sequential fault diagnosis," IEEE trans. on man, and cybernetics, Vol. 20, No.4, pp 872-887, system, July/August 1990.

3- M . R . Garey," optimal binary identification procedures , " SIAM J. Appl . Math . , Vol .23, no. 2, pp 173-186.

4- Juanne B . Dugan ,S. J. Bavuso , M.A. Boyd , " Dynamic fault tree models for fault tolerant computer systems ," IEEE trans. on reliability - vol 41, no. 3 , pp 363-376 , sep .1992.

5- Neural network professional II : User,s Guide , Neuralware Inc . Pittsburgh , PA , 1989 .

6- T . Sorsa , H . N . Koivo, H . Koivisto, " Neural networks in process faukt diagnosis , " IEEE trans , on systems , man cybernetics , Vol. 21, no . 4 , pp 815-825 , July/Aug . 1991.

7- R . P . Lippmann , " An introduction to computing with neural nets " IEEE ASSP Mag ., vol . 4, pp 4-22 , Apr . 1987

8- D.E. Rumelhart, J.L. McClelland & PDP Research group . " prallel distributed processing , explorations in the micro structure of cognition," MIT press , 1986 (1988).

9- K. S. Narendra , K. Parthasarathy ," Identification and control of dynamical systems using neural networks ," IEEE trans.on neural networks vol. 1, no.1, pp 4-27 March 1990.

10-D . E . Rumelhart , D. Zipser ," Feature discovery by competitive learning ." Cognitive science , 11,1987 .

11-S . Grossberg ." Competitive learning : from interactive activation to adptive resonance ," Cognitive science , 11 , 1987.