

الگوریتمی بر مبنای شبکه عصبی برای مسیریابی در شبکه‌های میان‌ارتباطی چندسطحی

مهران قاضی‌عسگر^{*} ، علی موقر رحیم‌آبادی[†]

چکیده

در این مقاله از یک الگوریتم یادگیری و بر مبنای شبکه‌های عصبی برای مسیریابی در شبکه‌های میان‌ارتباطی چندسطحی (MIN)⁽¹⁾ استفاده کردہ‌ایم. در این الگوریتم یکتابع انرژی بر اساس مسیر صحیح ما بین ورودی و خروجی تعریف کرده و با روش حرکت در خلاف جهت بردار گرادیان⁽²⁾ سعی می‌کنیم در نهایت این تابع به مقدار کمینه خود برسد. نتایج شبیه‌سازی نشان می‌دهد این روش مسیر صحیح ما بین ورودی و خروجی را پیدا کرده و از لحاظ کارآیی نیز قابل مقایسه با روش‌های دیگر می‌باشد. پارامترهای یک الگوریتم یادگیری معمولاً با استفاده از سعی و خطأ بدست می‌آیند؛ ولی ما در اینجا فرمول‌هایی را برای تعیین مقدار آنها بیان کرده‌ایم، حد آستانه بالا⁽³⁾ و حد آستانه پایین⁽⁴⁾ برای روش‌بودن و خاموش‌بودن نوروون‌ها نیز محاسبه شده‌اند، که طراحی شبکه‌های عصبی بر اساس این حدود موجب بهبود عملکرد شبکه می‌گردد. این الگوریتم را می‌توان به صورت سخت‌افزاری و یا نرم‌افزاری پیاده‌سازی نمود که پیاده‌سازی نرم‌افزاری آن مورد بررسی قرار خواهد گرفت.

کلمات کلیدی

شبکه‌های میان‌ارتباطی چندسطحی، شبکه‌های عصبی، مدل Hopfield، ضریب یادگیری، حد آستانه بالا، حد آستانه پایین.

A Neural Network Based Algorithm for Routing in Multistage Interconnection Networks

Mehran Ghaziasgar, Ali Movaghah Rahimabadi

Abstract

In this paper, we use a learning algorithm which is based on neural networks for routing in MINs(multistage interconnection networks). In this algorithm an energy function that is based on the appropriate path between input and output is defined, and using gradient descent method, the energy is minimized at the convergence of neural network. Simulation results show that this method finds the correct path between input and output, and the performance is comparable with other methods.

In general, parameters of a learning algorithm are achieved by trial and error, but here we suggest some formulas to find the value of parameters. Upper trigger point and lower trigger point for neurons being on or off are also calculated; designing neural network base on this points, gives the better functionality of the network. This algorithm can be implemented in hardware or software; the software implementation will be inspected.

Key words

MIN(multistage interconnection network), neural network, Hopfield model, learning rate, UTP(upper trigger point), LTP(lower trigger point).

* دانشجوی کارشناسی ارشد، دانشگاه آزاد اسلامی واحد اراک، gghmrm@yahoo.com

[†] دانشیار و عضو هیأت علمی دانشگاه صنعتی شریف، دانشکده مهندسی کامپیوتر، movaghah@sharif.edu

شبکه‌های میان ارتباطی چندسطحی [11]، اشاره کرد. استفاده از الگوریتم‌های یادگیری در مسیریابی شبکه‌های داده نیز مورد استفاده قرار گرفته است. [12,13].

در این مقاله تغییراتی در روش‌های استفاده شده در مراجع [12,13] داده‌ایم و روشی را برای مسیریابی در MINها مطرح کرده که تعداد نورون‌های مورد استفاده در آن نسبت به آنچه قبلًا بررسی شده است، [7,8]، کمتر می‌باشد، و همچنین سعی کرده‌ایم مقادیر مناسب برای پارامترهای مسأله را تعیین کنیم.

۲- مقدمات موردنیاز برای حل مسأله

در این بخش بطور خلاصه روش حرکت در خلاف جهت بردار گرادیان و ساختار کلی MINها را بیان می‌کنیم، در [1] روش‌های مختلف یادگیری و در [2] شبکه‌های میان ارتباطی به شکل گسترشده بررسی شده‌است.

۱-۲- حرکت در خلاف جهت بردار گرادیان

مبناًی کمینه کردن تابع انرژی حرکت در خلاف جهت بردار گرادیان می‌باشد فرض کنید انرژی تابعی از u_1, u_2, \dots, u_n باشد:

$$E(\mathbf{u}) = E(u_1, u_2, \dots, u_n) \quad (1)$$

که در آن E تابع انرژی است. حال می‌توان بردار گرادیان را تشکیل داد:

$$\frac{\partial E}{\partial \mathbf{u}} = \left(\frac{\partial E}{\partial u_1}, \frac{\partial E}{\partial u_2}, \dots, \frac{\partial E}{\partial u_n} \right) \quad (2)$$

برای اینکه بتوان به سمت کمینه این تابع حرکت کرد، اگر هر کدام از مؤلفه‌های بردار گرادیان مثبت باشد، متغیر مربوط به آن مؤلفه را کم می‌کنیم، و اگر هر کدام از مؤلفه‌های بردار گرادیان منفی باشد، متغیر مربوط به آن مؤلفه را زیاد می‌کنیم:

$$\mathbf{u}_{i+1} = \mathbf{u}_i - \alpha \frac{\partial E}{\partial \mathbf{u}_i} \Rightarrow \Delta \mathbf{u}_i = -\alpha \frac{\partial E}{\partial \mathbf{u}_i} \quad (3)$$

در معادله بالا α که عددی مثبت است، ضریب یادگیری^۵ نام دارد و \mathbf{u} نمایانگر مقادیر متغیرها در مرحله i از اجرای الگوریتم است؛ بدین مفهوم که الگوریتم با مقادیر اولیه \mathbf{u}_0 شروع شده و مرحله به مرحله ادامه پیدا می‌کند؛ با استفاده از مقادیر متغیرها در مرحله i مقادیر متغیرها در مرحله $i+1$ بدست می‌آید. الگوریتم تا جایی ادامه پیدا می‌کند که مقدار تابع انرژی به حد مطلوب برسد.

۲-۲- شبکه‌های میان ارتباطی چندسطحی

شبکه میان ارتباطی چندسطحی از مجموعه‌ای سوئیچ $a \times b$ تشکیل می‌شود، هر سوئیچ $a \times b$ دارای a خط ورودی و b خط خروجی است، این سوئیچ می‌تواند هر کدام از خطوط ورودی را به هر کدام از خطوط خروجی متصل نماید. این سوئیچ‌ها در طبقات مختلفی در کنار هم قرار می‌گیرند، ورودی‌های شبکه، ورودی سوئیچ‌ها در طبقه اول و

۱- مقدمه

تاکنون تحقیقات زیادی برای یافتن الگوریتم‌ها و ساختارهایی که بتواند مانند انسان قادر به یادگیری و حل مسائل باشد، انجام شده است؛ دسته‌ای از این ساختارها شبکه‌های عصبی نام دارد. شبکه‌های عصبی مجموعه‌ای از واحدهای محاسباتی کوچک می‌باشد که عملکرد هر کدام بسیار ساده است ولی وقتی در کنار هم قرار می‌گیرند، می‌توانند مسائل بزرگی را حل کنند، با ایده گرفتن از سلولهای عصبی یک موجود زنده هر کدام از این واحدهای محاسباتی کوچک نورون نیز نامیده می‌شود. یادگیری در انسان عمولاً بدون دانستن روابط ریاضی پیچیده موجود در یک مسأله صورت می‌گیرد؛ ولی موقعی که بخواهیم یک مسأله را با استفاده از ماشین حل کنیم ناچار به دانستن روابط ما بین متغیرهای مسأله هستیم، الگوریتم‌های یادگیری سعی می‌کند با کمک تکنیک‌های خاصی روابط ما بین متغیرها را در یک مسأله خاص تخمین زده و راه حل را به شکلی که بتواند در ماشین پیاده‌سازی شود، ارائه کند.

یکی از کاربردهای مهم شبکه‌های عصبی استفاده از آنها در حل مسائل بهینه سازی است؛ در اینگونه مسائل با توجه به معیار خاصی به دنبال یافتن بهترین جواب مسأله هستیم؛ ولی در بسیاری اوقات یافتن این جواب بسیار وقت‌گیر است و یافتن جواب قابل قبول در زمان کمتر اهمیت خاصی پیدا می‌کند.

ساختار و مدلی که برای حل اینگونه مسائل مورد استفاده زیادی قرار گرفته است شبکه Hopfield می‌باشد، که توسط J.Hopfield ارائه شده است [4]. یکی از خصوصیات مهم این ساختار سادگی پیاده‌سازی آن در سخت‌افزار است، و در نتیجه نورون‌ها بطور موازی محاسبات خود را انجام داده و سرعت یافتن جواب افزایش می‌یابد، می‌توان از این ساختار برای حل مسأله فروشنده دوره‌گرد که از دسته مسائل بغرنچ (NP) به حساب می‌آید استفاده نمود [5]، در این روش شبکه عصبی مورد استفاده همواره بهترین جواب ممکن را تولید نمی‌کند ولی نتایج بدست آمده نشان می‌دهد جواب محاسبه شده تا حد زیادی تزدیک به جواب بهینه است و نکته مهم این است که این جواب در زمان مناسبی بدست می‌آید. عملکرد شبکه Hopfield بطور کامل با معادلات دیفرانسیل مربوط به آن توصیف می‌شود و به گونه‌ای عمل می‌کند که سیستم به سمت کمینه کردن تابع انرژی مربوط به شبکه حرکت می‌کند، حال اگر این تابع انرژی متناظر با شرایط مسأله تعریف گردد به طوری که کمینه تابع انرژی متناظر با جواب مسأله باشد سیستم در جهت یافتن راه حل حرکت خواهد کرد.

شبکه‌های Hopfield به شکل‌های مختلف در مسیریابی مورد استفاده قرار گرفته است. از آن جمله می‌توان به مسیریابی در شبکه‌های میان ارتباطی تصادفی و نوری [6,7]، مسیریابی در شبکه‌های میان ارتباطی چندسطحی [8,9,10]، و کنترل ترافیک در

آن را مقدار کوچکی کمتر از ۰.۵ قرار می‌دهیم. برای یافتن مسیر صحیح هر دو شرط زیر باید برآورده شود:

۱. هیچ تداخلی در خطوط ارتباطی رخ ندهد.
۲. در هر طبقه حداکثر یک نورون روش باشد

حال تابع انرژی مطابق با فرمول (۴) را در نظر بگیرید که در آن β یک ضریب ثابت و مثبت است:

$$E = \sum_{k=1}^{(K-1)} \sum_{i=0}^{(n_{k-1}-1)} \sum_{j=0}^{(n_k-1)} w_{kij} u_{(k-1)i} u_{kj} + \beta \sum_{k=1}^{K-2} \left(\left(\sum_{i=0}^{n_k-1} u_{ki} \right)^2 - 1 \right) \quad (4)$$

کمینه عبارت اول در تابع انرژی وقتی رخ می‌دهد که مسیر ارتباطی ما میان نورون i در طبقه $k-1$ و نورون j در طبقه k اشغال نباشد و یا اگر این خط اشغال است یکی از این دو نورون خاموش باشند، و در نتیجه مسیر از این خط ارتباطی استفاده نخواهد کرد؛ پس این عبارت متناظر با شرط اول مسئله می‌باشد؛ یعنی در صورت معتبر بودن مسیر، مقدار عبارت اول در فرمول (۴) نزدیک به صفر است. عبارت دوم در تابع انرژی موقعی صفر است که مجموع مقادیر نورون‌ها در هر طبقه برابر یک باشد، در موقع همگرایی فقط یک نورون مقدار یک داشته و بقیه صفر خواهند بود؛ پس این عبارت متناظر با شرط دوم مسئله می‌باشد. با توجه به مطالب فوق می‌توان گفت کمینه تابع انرژی متناظر با حل مسئله است.

۲-۳- بررسی عملکرد شبکه عصبی و تعیین پارامترهای الگوریتم

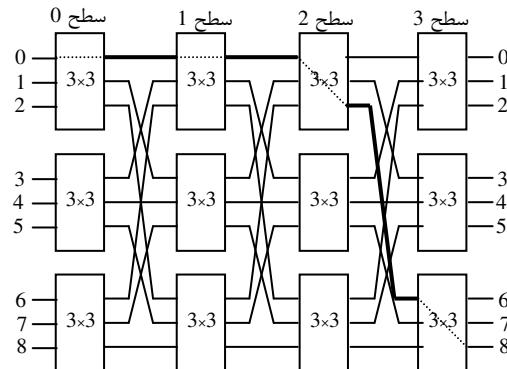
در هنگام شروع محاسبات یک جفت ورودی، خروجی به شبکه عصبی داده می‌شود. به نورون متناظر با سوئیچ ورودی مورد نظر در طبقه ورودی (طبقه ۰) مقدار ۱ و به بقیه نورون‌ها در طبقه ورودی مقدار صفر داده می‌شود. به نورون متناظر با سوئیچ خروجی مورد نظر در طبقه خروجی (طبقه $K-1$) مقدار ۱ و به بقیه نورون‌ها در طبقه خروجی مقدار صفر داده می‌شود. به هر کدام از سوئیچ‌ها در طبقه‌ای به غیر از طبقه ورودی یا خروجی مقدار $1/n_k$ داده می‌شود؛ بدین مفهوم که احتمال انتخاب هر کدام از سوئیچ‌ها مساوی با دیگری است. در ضمن جمع مقادیر نورون‌ها در هر طبقه هم برابر با ۱ خواهد بود. در قسمت اول جدول (۱) مقادیر اولیه بردارها، برای مسیری که در شبکه (۱) دیدید، آورده شده است. وزن‌ها مطابق با مطالب گفته شده در بخش (۱-۳) مقداردهی می‌شوند:

$$w_{kij} = \begin{cases} (j/2n_k) & \text{در صورت آزاد بودن خط} \\ 1 & \text{در صورت اشغال بودن خط} \end{cases} \quad (5)$$

حال می‌توان نوشت:

خروجی‌های شبکه، خروجی سوئیچ‌ها در طبقه آخر هستند، که اتصال آنها از طریق سوئیچ‌های میانی برقرار می‌گردد.

هر سوئیچ در طبقه k می‌تواند (و نه لزوماً) با هر سوئیچ در طبقه $k+1$ ارتباط داشته باشد، در شکل (۱) شبکه میان ارتباطی ۴ سطحی با ۹ ورودی و ۹ خروجی نشان داده شده است؛ که در آن ورودی شماره ۹ صفر به خروجی شماره ۸ متصل شده است. مسیریابی در شبکه باید به گونه‌ای صورت گیرد که در خطوط ارتباطی داخل پیش نیاید؛ بدین مفهوم که مثلاً دو جفت ورودی و خروجی از خط ارتباطی مشترکی استفاده نکند.



شکل (۱): شبکه میان ارتباطی ۴ سطحی

۳- معرفی راه حل

۱-۳- تعریف تابع انرژی

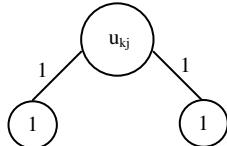
برای تعریف تابع انرژی در یک شبکه میان ارتباطی K سطوح را از ۰ تا $K-1$ شماره‌گذاری می‌کنیم و به ازای هر کدام از سوئیچ‌ها یک نورون در نظر می‌گیریم، نورون‌ها در هر طبقه نیز از ۰ تا n_k شماره‌گذاری می‌شوند که برابر با تعداد نورون‌ها در طبقه k است. با این تعاریف نورون j ام در طبقه k را u_{kj} می‌نامیم، هر مسیر مابین یک ورودی و خروجی در این شبکه از $K-1$ خط ارتباطی می‌گذرد و در هر طبقه دقیقاً از یک سوئیچ برای مسیریابی استفاده می‌کند. حال فرض کنید مقدار نورون‌ها u_{kj} ، همواره مقداری میان صفر و یک باشد؛ که مقدار صفر بیانگر خاموش بودن نورون و مقدار ۱ بیانگر روشن بودن نورون است. در موقع همگرایی در هر طبقه از شبکه عصبی دقیقاً باید یک نورون روشن باشد که نشانگر سوئیچ مورد استفاده در مسیریابی می‌باشد و در صورتی که نتوان مسیری مابین ورودی و خروجی مورد نظر پیدا کرد باید همه نورون‌ها خاموش شوند.

در ادامه برای هر کدام از خطوط ارتباطی یک وزن در نظر می‌گیریم، نورون i در طبقه $K-1$ از طریق یک خط ارتباطی به نورون j در طبقه k متصل می‌شود و ما وزن آن را w_{kij} می‌نامیم. اگر این خط قبلًا توسط یک مسیر اشغال شده باشد و یا از ابتدا در ساختار شبکه موجود نباشد، مقدار آن را برابر یک و در صورت آزاد بودن خط مقدار

مقادیر نورون‌ها همواره باید مثبت باشند، برای اینکه مثبت بودن آنها تضمین شود مقدار $\max_{u_{kj}}$ باید بیشتر از مقدار u_{kj} باشد، در

نتیجه:

$$\max_{u_{kj}} \leq u_{kj} \Rightarrow \alpha \leq \frac{u_{kj}}{2} \quad (12)$$



شکل (۲): حالت یک نورون در موقع همگرایی، برای محاسبه ضریب یادگیری

برای آنکه سرعت همگرایی الگوریتم بیشتر شود باید α را تا حد ممکن بزرگ در نظر بگیریم و فرمول های (۱۰) و (۱۱) حد بالایی از α را در اختیار ما می‌گذارند که البته باید کمینه این دو حد را به عنوان ضریب یادگیری در هر مرحله انتخاب نمود. با انتخاب ضریب یادگیری مناسب علاوه بر اینکه مثبت بودن مقدار هر نورون تضمین می‌شود، جمع تمام مقادیر تمام u_{kj} ها همواره بین صفر و یک باقی خواهد ماند.

برای تعیین پارامتر دوم مسأله یعنی β فرض کنید هیچ مسیر صحیحی ما بین ورودی و خروجی موجود نباشد، حال مسیری را در نظر بگیرید که در آن فقط یکی از خطوط ارتباطی، اشغال باشد. اگر تمام نورون‌ها در طبقه‌ای که به خط اشغال متصل است، صفر شوند مقدار انرژی متناظر با این طبقه، برابر با $E_1 = \beta$ است، و اگر در این طبقه یک سوئیچ برای مسیریابی انتخاب شود، مقدار انرژی متناظر با این طبقه برابر با $E_2 = 1 - \beta$ است. با توجه به اینکه حالت اول حالت مورد نظر ما می‌باشد باید انرژی حالت اول از دوم کمتر باشد، تا بتوان به سمت جواب صحیح حرکت کرد:

$$E_1 < E_2 \Rightarrow \beta < 1 \quad (13)$$

اگر بخواهیم با توجه به شرایط مطرح شده، مقدار انرژی متناظر با طبقه‌ای که به خط اشغال متصل است را بحسب آوریم، فرض کنید مقدار نورونی که در طبقه مذکور با خط اشغال در ارتباط است برابر e باشد. پس با توجه به اینکه می‌دانیم بقیه نورون‌ها در این طبقه صفر هستند و با استفاده از فرمول (۴) مقدار انرژی این طبقه برابر است با:

$$E = e + \beta(1-e)^2 \quad (14)$$

کمینه فرمول (۱۴) موقعی رخ می‌دهد که $e = (2\beta - 1)/2\beta$ باشد، و چون انتظار داریم E ، کمینه شود و e هم باید مثبت باشد می‌توان گفت:

$$2\beta > 1 \Rightarrow \beta > 0.5 \quad (15)$$

با ترکیب فرمول‌های (۱۳) و (۱۵) :

$$0.5 < \beta < 1 \quad (16)$$

ترکیب فرمول‌های (۱۰) و (۱۶) و فرمول (۱۶) ما را در انتخاب پارامترهای مسأله کمک می‌کند.

$$\begin{aligned} \frac{\partial E}{\partial u_{kj}} &= \sum_i w_{kij} u_{(k-1)i} + \sum_i w_{(k+1)ji} u_{(k+1)i} \\ &\quad + 2\beta \left(\sum_i u_{ki} - 1 \right) \end{aligned} \quad (6)$$

حال با توجه به فرمول (۳) تغییرات مقدار هر نورون، در هر مرحله از اجرای الگوریتم بدست می‌آید:

$$\Delta u_{kj} = -\alpha(S_1 + S_2 + S_3) \quad (7)$$

که در فرمول بالا α ضریب یادگیری و S_1, S_2, S_3 به ترتیب عبارت اول، دوم و سوم در فرمول (۶) هستند. برای اینکه یادگیری به شکل مناسی صورت گیرد تمام مقادیر u_{kj} باید همواره اعدادی بین صفر و یک باشند. همچنانی جمع کلیه مؤلفه‌های یک بردار در یک طبقه نیز همواره باید کوچکتر یا مساوی با یک باشد. با ابتدا تمام مقادیر u_{kj} به همان شکلی است مطابق با آنچه گفته شد، در شروع S_3 عددی کوچکتر یا مساوی صفر، واضح است که S_1 و S_2 مثبت هستند. حال پارامترها باید به گونه‌ای تعیین شوند که این شرایط همواره باقی بماند.

اجازه دهید حداقلر مقداری که می‌تواند به یک نورون اضافه شود را \max^+ در نظر بگیریم، با توجه به فرمول (۷)، منفی بودن S_3 و مثبت بودن S_1 و S_2 داریم:

$$\max^+ = -\alpha S_3 \quad (8)$$

مقدار بالا همواره عددی مثبت است. برای اینکه تضمین شود جمع کلیه نورون‌ها در هر طبقه همواره کوچکتر یا مساوی یک است، فرض کنید در یک طبقه $\sum_i u_{ki} = \theta$ باشد. پس حداقل مجموع کل نورون‌ها در هر طبقه، پس از اجرای هر مرحله از الگوریتم برابر با فرمول (۹) است:

$$\max(\theta + \sum_i \Delta u_{ki}) = \theta + n_k \max^+ = \theta - \alpha S_3 n_k \quad (9)$$

با جایگذاری S_3 از فرمول‌های (۶) و (۷)، و اینکه مقدار فرمول (۹) باید کوچکتر از یک باشد به نتیجه زیر می‌رسیم:

$$\theta + 2\alpha\beta(1-\theta)n_k \leq 1 \Rightarrow \alpha \leq \frac{1}{2\beta n_k} \quad (10)$$

حال اجازه دهید حداقلر مقداری که در یک مرحله از اجرای الگوریتم از مقدار یک نورون کم می‌شود را \max_- بنامیم، با توجه به فرمول (۷) این مقدار برابر با $(S_1 + S_2) / \alpha$ است. u_{kj} با طبقات قبل و بعد خود در ارتباط است و جمع مقادیر نورون‌ها در هر کدام از این طبقات حداقل ۱ است، پس در شرایطی که \max_- از لحاظ قدر مطلق به حداقل خود می‌رسد می‌توان گفت u_{kj} با یک نورون روشن در طبقه قبلی و با یک نورون روشن در طبقه بعدی و با وزن ۱ در ارتباط است (شکل ۲). با این شرایط می‌توان نوشت:

$$\begin{aligned} \max_- &= (\alpha S_1 + \alpha S_2) \leq (\alpha + \alpha) \\ &\Rightarrow \max_- = 2\alpha \end{aligned} \quad (11)$$

برای بدست آوردن حد آستانه بالا (یعنی مقداری که اگر ورودی نورون بیشتر از آن باشد نورون را روشن فرض می‌کنیم)، u_{kj} را متناظر با سوئیچی در نظر بگیرید که باید برای مسیریابی استفاده شود، درنتیجه مقدار آن باید به سمت ۱ نزدیک شود، فرض کنید مقدار u_{kj} در آستانه همگرایی، a باشد. مانند قبل مقداری که در هر مرحله به u_{kj} اضافه می‌شود را با sum^+ نشان می‌دهیم، در موقع همگرایی تمام نورون‌های دیگر در طبقه k به مقدار صفر رسیده‌اند، با این فرض sum^+ برابر با فرمول (۲۱) است:

$$sum_a^+ = -2\alpha\beta \left(\sum_i u_{ki} - 1 \right) = 2\alpha\beta(1-a) \quad (21)$$

حال مقداری که در هر مرحله از u_{kj} کم می‌شود را در نظر بگیرید، برای محاسبه مقدار کم کننده، sum^- ، فرض کنید N ماتریس تعداد سوئیچ‌ها در هر طبقه باشد با توجه به فرمول (۵) حداقل وزن خطوط ارتباطی آزاد برابر با فرمول (۲۲) است:

$$Max(w_{kij}) = \frac{N-1}{2N} \quad N \geq 1 \quad (22)$$

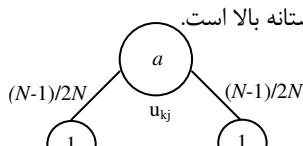
u_{kj} با طبقات قبل و بعد خود در ارتباط است و جمع مقداری نورون‌ها در هر کدام از این طبقات نیز حداقل ۱ است، پس در شرایطی که sum^- از لحاظ قدر مطلق به حداقل خود می‌رسد می‌توان گفت u_{kj} با یک نورون روشن در طبقه قبلی و با یک نورون روشن در طبقه بعدی و با حداقل وزن در ارتباط است (شکل ۴) با این شرایط حداقل مقدار کم کننده از لحاظ قدر مطلق برابر با فرمول (۲۳) است:

$$sum_a^- = \alpha \left(\frac{N-1}{2N} + \frac{N-1}{2N} \right) = \alpha \left(\frac{N-1}{N} \right) \quad (23)$$

حد آستانه موقعی رخ می‌دهد که مقدار sum_a^+ و sum_a^- مساوی شوند، پس از فرمول‌های (۲۳ و ۲۱) نتیجه می‌گیریم:

$$2\beta(1-a) = \frac{N-1}{N} \Rightarrow a = \frac{2\beta - (N-1/N)}{2\beta} \quad (24)$$

که در آن a حد آستانه بالا است.



شکل (۴): حالت یک نورون در موقع همگرایی، برای محاسبه حد آستانه بالا

۳-۳- تعیین حد آستانه بالا و حد آستانه پایین

طراحی نورون‌ها با استفاده از حد آستانه بالا و پایین موجب بهبود عملکرد شبکه می‌گردد [۳]. اگر مقدار ورودی یک نورون را u و خروجی آن را v ، حد آستانه بالا را a و حد آستانه پایین را b در نظر بگیریم، می‌توان روشن بودن و یا خاموش بودن نورون را از فرمول (۱۷) بدست آورد:

$$v = \begin{cases} 1 & u \geq a \\ 0 & u \leq b \\ no\ change & a < u < b \end{cases} \quad (17)$$

برای بدست آوردن حد آستانه پایین (یعنی مقداری که اگر ورودی نورون کمتر از آن باشد نورون را خاموش فرض می‌کنیم)، u_{kj} متناظر با سوئیچی در نظر بگیرید که مقدار آن باید به سمت صفر نزدیک شود، فرض کنید مقدار u_{kj} در آستانه همگرایی، b باشد، با توجه به فرمول (۷) مقداری که در هر مرحله به u_{kj} اضافه می‌شود برابر با $-\alpha\delta_3$ خواهد بود و آن را با sum^+ نشان می‌دهیم، اگر فرض کنیم تمام نورون‌های دیگر در طبقه k به مقدار صفر رسیده باشند، بیشترین مقدار sum^+ بدست می‌آید، با این فرض sum^+ برابر با فرمول (۱۸) است:

$$sum_b^+ = -2\alpha\beta \left(\sum_i u_{ki} - 1 \right) = 2\alpha\beta(1-b) \quad (18)$$

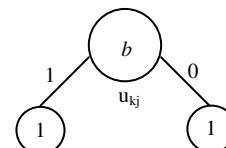
و اما با توجه به همان فرمول (۷) مقدار $(S_1 + S_2)\alpha$ از مقدار u_{kj} کم می‌شود که آن را با sum^- نشان می‌دهیم، u_{kj} با طبقات قبل و بعد خود در ارتباط است و جمع مقادیر نورون‌ها در هر کدام از این طبقات حداقل ۱ است. با توجه به اینکه u_{kj} به سمت صفر حرکت می‌کند حداقل با یک خط اشغال در ارتباط است، پس در شرایطی که sum^- از لحاظ قدر مطلق به حداقل خود می‌رسد می‌توان گفت با یک نورون روشن با وزن ۱، و با یک نورون روشن با وزن صفر در ارتباط است (شکل ۳). و داریم:

$$sum_b^- = \alpha(1+0) = \alpha \quad (19)$$

حد آستانه موقعی رخ می‌دهد که مقدار sum_b^+ و sum_b^- مساوی شوند، پس از فرمول‌های (۱۸ و ۱۹) نتیجه می‌گیریم:

$$2\beta(1-b) = 1 \Rightarrow b = \frac{2\beta-1}{2\beta} \quad (20)$$

که در آن b حد آستانه پایین است.



شکل (۳): حالت یک نورون در موقع همگرایی، برای محاسبه حد آستانه پایین

۴- نتایج شبیه سازی

در قسمت اول جدول (۱) مقادیر اولیه نورون‌های متناظر با شبکه شکل (۱) نشان داده شده‌است که در آن می‌خواهیم از ورودی شماره صفر به خروجی شماره ۸ برویم، تغییرات شبکه پس از ۵۰ مرحله و پس از ۱۰۰ مرحله نیز در جدول (۱) آورده شده‌است. اگر $\beta=0.9$ باشد، با توجه به فرمول‌های (۲۰ و ۲۴)، $a=0.63$ و $b=0.44$ می‌باشد و می‌بینید سوئیچ‌های اول در طبقات اول و دوم به حدود مورد نظر برای روشن شدن و انتخاب در مسیریابی رسیده‌اند. پس از مسیریابی این

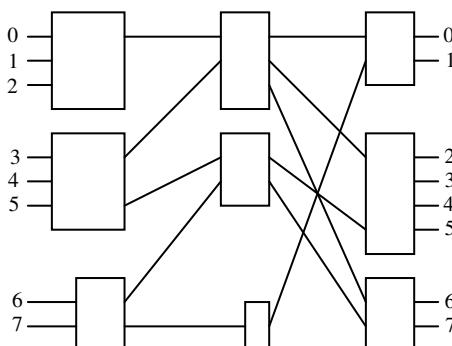
دو خروجی مشترک و یا دو ورودی مشترک وجود دارد، در نظر گرفته نشده‌اند، چرا که فرض می‌شود دو پردازنده در یک زمان نمی‌توانند به یک حافظه متصل شوند و همچنین یک پردازنده هم در یک زمان از دو حافظه استفاده نمی‌کند.

ستون سوم جدول (۳)، E_{nn} ، میانگین تعداد پیام‌های را نشان می‌دهد که توسط شبکه عصبی مطرح شده در [۶]، مسیریابی می‌شوند. برای مسیریابی، یک مجموعه پیام به شبکه عصبی داده می‌شود. این اعداد با تولید ۱۰۰۰ جفت ورودی و خروجی تصادفی به طول M بدست آمدند.

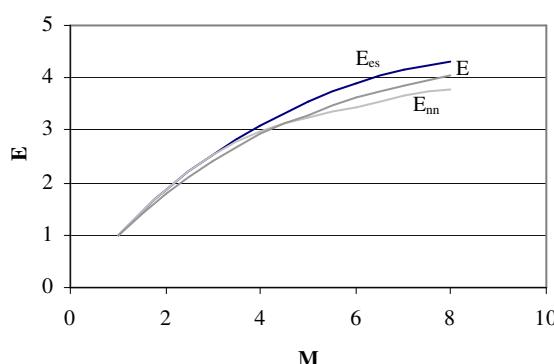
ستون چهارم جدول (۳)، E ، میانگین تعداد پیام‌های را نشان می‌دهد که توسط الگوریتم مطرح شده در این مقاله، مسیریابی می‌شوند. الگوریتم، پیام‌ها را یکی پس از دیگری و در صورت امکان مسیریابی می‌کند این اعداد با تولید ۱۰۰۰ جفت ورودی و خروجی تصادفی به طول M بدست آمده‌اند، همان‌طور که دیده می‌شود این اعداد نزدیک به اعداد بدست آمده از الگوریتم جستجوی کامل هستند.

جدول (۳): میانگین تعداد پیام‌های مسیریابی شده

M	E_{es}	E_{nn}	E
1	1.00000	1.00	1.000
2	1.85714	1.87	1.810
3	2.52487	2.51	2.411
4	3.08276	2.98	2.952
5	3.53214	3.24	3.301
6	3.89566	3.45	3.623
7	4.15982	3.66	3.861
8	4.32857	3.78	4.063



شکل (۵): شبکه میان ارتباطی تصادفی [۶]



شکل (۶): تعداد پیام‌های مسیریابی شده بر حسب طول مجموعه پیام

درخواست و تغییردادن وزن‌های مورد استفاده فرض کنید ورودی صفر بخواهد به خروجی صفر متصل شود. تغییرات شبکه در این حالت در جدول (۲) نشان داده شده است. همانطور که می‌بینید این‌بار در طبقه اول، سوئیچ دوم برای مسیر انتخاب می‌شود.

جدول (۱): تغییرات مقادیر نورون‌ها برای اتصال ورودی ۰ به خروجی ۸

سطح 0	سطح 1	سطح 2	سطح 3
1.00	0.33	0.33	0.00
0.00	0.33	0.33	0.00
0.00	0.33	0.33	1.00

۱. مقادیر اولیه نورون‌ها			
$k=0$	$k=1$	$k=2$	$k=3$
1.00	1.00	0.81	0.00
0.00	0.00	0.00	0.00
0.00	0.00	0.00	1.00

۲. مقادیر نورون‌ها پس از ۱۰۰ مرحله			
1.00	0.97	0.79	0.00
0.00	0.02	0.02	0.00
0.00	0.00	0.00	1.00

جدول (۲): تغییرات مقادیر نورون‌ها برای اتصال ورودی ۰ به خروجی ۸

سطح 0	سطح 1	سطح 2	سطح 3
1.00	0.33	0.33	1.00
0.00	0.33	0.33	0.00
0.00	0.33	0.33	0.00

۱. مقادیر اولیه نورون‌ها			
سطح 0	سطح 1	سطح 2	سطح 3
1.00	0.00	1.00	1.00
0.00	0.91	0.00	0.00
0.00	0.00	0.00	0.00

۲. مقادیر نورون‌ها پس از ۱۰۰ مرحله			
1.00	0.00	1.00	1.00
0.00	0.00	0.00	0.00
0.00	0.00	0.00	0.00

یکی از کاربردهای شبکه‌های میان ارتباطی اتصال پردازنده‌ها به حافظه‌ها می‌باشد، با این ساختار در هر برره زمانی ممکن است تعدادی پردازنده بخواهد به تعدادی حافظه متصل شوند، که هر پردازنده با یک پیام درخواست خود را مطرح می‌کند پس در هر برره زمانی مجموعه‌ای از پیام‌ها به مسیریاب می‌رسد، تعداد پیام‌هایی که در این مجموعه وجود دارد اندازه این مجموعه پیام نامیده می‌شود.

یکی از معیارهایی که در مورد یک روش مسیریابی موردنظر قرار دارد، این است که با داشتن مجموعه پیامی به طول M مسیریاب چه تعدادی از این پیام‌ها را بدون تداخل مسیریابی می‌کند، و چه تعدادی باید منتظر بمانند [۶]. برای اینکه حداقل پیام‌ها مسیریابی شوند باید جستجوی کامل را انجام داد؛ یعنی اگر کل مجموعه نتواند بدون تداخل مسیریابی شود باید زیر مجموعه‌ای از آن را با حداقل اندازه ممکن مسیریابی کرد. واضح است که پیاده سازی این الگوریتم در عمل بسیار وقت‌گیر است ولی می‌تواند معیاری برای بررسی کارآیی الگوریتم‌های دیگر باشد، چرا که این روش حداقل کلیه مجموعه‌های مسیریابی می‌کند [۶].

ستون اول جدول (۳) نشانگر طول مجموعه پیام و ستون دوم این جدول، E_{es} ، میانگین تعداد پیام‌هایی است که توسط الگوریتم جستجوی کامل در شبکه میان ارتباطی تصادفی شکل (۵) مسیریابی می‌شوند. (منظور از تصادفی این است که الگوی خاصی برای اتصال ممکن است که یکدیگر وجود ندارد). این اعداد با تولید کلیه مجموعه‌های ممکن به طول M بدست آمده‌اند؛ البته مجموعه پیام‌هایی که در آنها

- [13] Lee, S., L., Chang, S., "Neural networks for routing of communication networks with unreliable components", IEEE transactions on neural networks, Vol. 4, No 5, pp. 854-863, 1993.

در شکل (۶) سه مورد مطرح شده در جدول (۳) با هم مقایسه شده‌اند.

۵- نتیجه

^۱ Multistage interconnection network

^۲ Gradient descent method

^۳ UTP(upper trigger point)

^۴ LTP(lower trigger point)

^۵ Learning rate

^۶ Message set

در این مقاله روشی را برای مسیریابی در شبکه‌های میان‌ارتباطی چندسطوحی بیان کردیم، پارامترهای مختلف مسئله را مورد بررسی قرار داده و نشان دادیم که این روش بدرستی کار می‌کند، و با استفاده از شبیه‌سازی نرم‌افزاری عملکرده آن را بررسی کردیم. از این روش می‌توان برای حل دیگر مسائل بهینه‌سازی نیز بهره برد.

پیاده‌سازی سخت‌افزاری این روش با استفاده از مدل Hopfield امکان‌پذیر است، پیاده‌سازی سخت‌افزاری و یا استفاده از این روش در کاربردهای دیگر نیاز به تحقیقات بیشتری دارد.

مراجع

- [1] Kekman, S., *Learning and Soft Computing Support Vector Machines, Neural Networks, and Fuzzy Logic Models*, Massachusetts Institute of Technology: MIT press, 2001.
- [2] Duato, J., Yalamanchili, S., Ni, L., *Interconnection Networks: An Engineering Approach*, IEEE Computer Society, 1997.
- [3] Xia, G., Tang, Z., Li, Y., Wang, J., "A binary Hopfield neural network with hysteresis for large crossbar packet-switches", Elsevier, Neurocomputing 67, pp. 417-425, 2005.
- [4] Hopfield, J., "Neurons with graded response have collective computational properties like those of two-state neurons", Proc. Nat. Acad. Sci. USA, Vol. 81, pp. 3088-3092, 1984.
- [5] Hopfield, J., Tank, D., "Neural's computation of decisions in optimization problems", Biol. Cybern., Vol. 52, pp. 141-152, 1985.
- [6] Goudreau, M. W., Giles, C. L., "Neural network routing for random multistage interconnection networks", Neurals information processing system 4, pp. 722-729, 1992.
- [7] Goudreau, W., Lee Giles, C., "Routing in optical multisatge interconnection networks: a neural network solution", Journal of light way technology, Vol. 13, No. 6, pp 1111-1115, 1995.
- [8] Brown, T., X., "Neural networks for switching", IEEE communications magazine, Vol. 27, No.11, pp. 73-81, 1989.
- [9] Brown, T., X., Liu, K., H., "Neural network design of a Banyan network controller", IEEE journal on selected areas of communication, Vol. 8, No. 8, pp. 1428-1438, 1990.
- [10] Melsa, P., J., Kenney, J., B., Rohrs, C., E., "A neural network solution for routing in three stage interconnection networks", International symposium on circuits and systems, pp. 483-486, 1990.
- [11] Funabiki, N., Takefuji, Y., Lee, K., C., "Comparisions of seven neural network models on traffic control problems in multistage interconnection networks", IEEE transactions on computers 42, pp. 497-501, 1993.
- [12] Rauch, H., E., Winarske T., "Neural networks for routing communication traffic", IEEE control systems magazine, pp. 26-31, 1988.