

# بررسی کارایی یک شبکه عصبی جدید برای مسیریابی در یک شبکه اتصالات توری

شادرخ سماوی\*، ولی‌اله طحانی\*\* و پژمان خدیوی\*\*\*  
دانشکده مهندسی برق و کامپیوتر، دانشگاه صنعتی اصفهان

(دریافت مقاله: ۷۹/۵/۴ - دریافت نسخه نهایی: ۸۰/۵/۱۵)

چکیده: مسیریابی، یکی از اساسی‌ترین قسمت‌های یک سیستم چندپردازنده مبتنی بر مبادله پیام است و تأثیر زیادی بر بازدهی سیستم دارد. الگوریتم‌های مسیریابی عصبی موجود که اساساً برای شبکه‌های رایانه‌ای تدوین شده‌اند، تعداد نرونهای زیادی را استفاده می‌کنند. این در حالی است که با توجه به توپولوژیهای خاصی که در سیستم‌های چند رایانه استفاده می‌شود، می‌توان تعداد نرونهای شبکه عصبی را کاهش داد. در این مقاله، یک شبکه عصبی برگشتی جدید که از تعداد کمی نرون استفاده می‌کند معرفی شده و سپس کارایی این شبکه عصبی پیشنهادی برای مسیریابی در یک شبکه اتصالات توری مورد بررسی قرار می‌گیرد.

واژگان کلیدی: مسیریابی، سیستم‌های چند رایانه، شبکه عصبی برگشتی، شبکه توری

## Performance Analysis of a New Neural Network for Routing in Mesh Interconnection Networks

S. Samavi, V. Tahani and P. Khadivi

Department of Electrical & Computer Engineering, Isfahan University of Technology

**Abstract:** Routing is one of the basic parts of a message passing multiprocessor system. The routing procedure has a great impact on the efficiency of a system. Neural algorithms that are currently in use for computer networks require a large number of neurons. If a specific topology of a multiprocessor network is considered, the number of neurons can be reduced. In this paper a new recurrent neural network and its energy function is introduced. This network requires a significantly smaller number of neurons compared to its counterparts. Also presented is the performance of this neural network.

**Keywords:** Routing, Multicomputer systems, Recurrent neural networks, Mesh interconnection networks.

### ۱- مقدمه

کارایی یک شبکه عصبی<sup>۳</sup> از نوع برگشتی<sup>۴</sup>، به همراه تابع انرژی آن است که برای مسیریابی در شبکه توری<sup>۵</sup> قابل استفاده است [۱].

ایده مسیریابی با استفاده از شبکه عصبی، نخستین بار توسط راش و وینارسک [۲] مطرح شد. در روش مطرح شده

مسیریابی<sup>۱</sup> در سیستم‌های چند پردازنده مبتنی بر مبادله پیام، یا چند رایانه<sup>۲</sup>ها، تأثیر زیادی در بازدهی سیستم دارد. به این منظور، الگوریتم‌های مسیریابی متعددی برای این سیستمها تدوین شده است. در این مقاله، هدف، معرفی و بررسی

\*\*\*- دانشجوی دکترا

\*\* - استاد

\* - استادیار

توسط این دو نفر، پیش از مسیریابی، باید تعداد گره‌های بین مبدا و مقصد را بدانیم. اگر تعداد گره‌های شبکه اتصالات  $n$  و تعداد گره‌های بین مبدا و مقصد  $m$  باشد، به یک شبکه عصبی هاپفیلد<sup>۶</sup> با  $m \times n$  نرون<sup>۷</sup> نیاز داریم، که به صورت یک ماتریس  $m \times n$  نمایش داده می‌شود. اگر، گره  $i$ ام شبکه اتصالات،  $j$ امین گرهی باشد که در مسیر بین مبدا و مقصد پیموده می‌شود، درایه  $(i, j)$  این ماتریس برابر یک خواهد بود. تابع انرژی شبکه بر اساس ظرفیت کانالها و با توجه به افزایش بازدهی سیستم نوشته می‌شود، با این فرض که در هر ستون از ماتریس بالا، نباید بیش از یک نرون روشن باشد.

ایراد روش راش و وینارسک، نیاز به دانستن تعداد گره‌های بین مبدا و مقصد است و این کار همیشه امکانپذیر نیست (اگر چه در شبکه‌هایی نظیر شبکه Mesh می‌توان به چنین آگاهی رسید). علاوه بر این، تعداد نرونهای شبکه عصبی در این روش، زیاد است. روشهای بعدی که در مراجع [۳-۵] مطرح شده‌اند نیز از تعداد زیادی نرون استفاده می‌کنند و به طور مشخص در [۳ و ۵]، تعداد نرونها توان دوم تعداد گره‌های شبکه اتصالات است. در مرجع [۶]، برای یک شبکه اتصالات با  $n$  گره، ابتدا یک شبکه عصبی هاپفیلد با  $(n-1)(n-2)$  نرون ارائه می‌شود. در مرجع مذکور سپس شبکه عصبی دیگری با  $n$  نرون معرفی شده‌است که خروجی آن برای تعیین مسیر نیاز به پردازش اضافی دارد. در همه روشهای بالا، شبکه عصبی، کل مسیر را از مبدا تا مقصد معرفی می‌کند.

روشهای مسیریابی عصبی پیشنهاد شده در مراجع [۲-۶]، برای استفاده در شبکه‌های رایانه‌ای مطرح شده‌اند ولیکن روش خاصی برای سیستمهای چند رایانه تدوین نشده‌است. اینکه چرا به روشهای جدیدی برای سیستمهای چند رایانه نیاز است، به اختلافات موجود بین این دو ساختار (شبکه‌های رایانه‌ای و چند رایانه‌ها) باز می‌گردد. بعضی از این اختلافات عبارت‌اند از

۱- در یک سیستم چند رایانه، مجموعه گره‌ها (واحدهای پردازنده) برای اجرای یک برنامه، همکاری می‌کنند، اما، در

یک شبکه رایانه‌ای چنین همکاری وجود ندارد و اساساً، در آنها فقط ارتباط مطرح است.

۲- گره‌ها و خطوط ارتباطی از نظر حجم حافظه، سرعت پردازش، سرعت انتقال و ...، معمولاً در سیستمهای چند رایانه یکسان‌اند ولی، این مطلب در شبکه‌های رایانه‌ای در اغلب موارد صادق نیست.

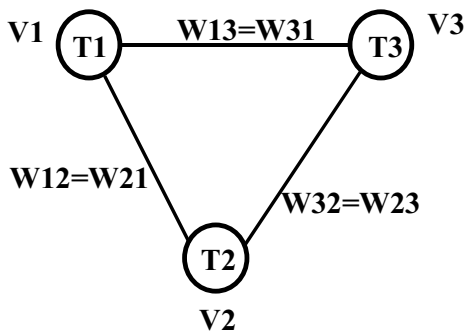
۳- توپولوژی شبکه‌هایی که، در سیستمهای چند رایانه استفاده می‌شوند، معمولاً از بین توپولوژیهای خاصی مانند توری، ابرمکعب<sup>۸</sup> و ... انتخاب می‌شود. این مسئله، به ساده شدن مسیریابی بسیار کمک می‌کند.

در روشهای کلاسیک نیز، در سیستمهای چند رایانه، از الگوریتمهای مسیریابی متفاوت با آنچه در شبکه‌های رایانه‌ای متداول است [۷]، استفاده می‌شود. در اغلب الگوریتمهای مسیریابی خاص سیستمهای چند رایانه، در هر گره، تنها گره بعدی که پیام باید به آن برود تعیین می‌شود و نه کل مسیر [۸ و ۹]. شبکه عصبی که در این مقاله ارائه می‌شود، با  $\left\lfloor \log_2 n \right\rfloor$  نرون برای یک شبکه توری با  $n^2$  گره، عمل مسیر یابی را انجام می‌دهد و در هر گره، تنها آدرس گره بعدی اعلام می‌شود.

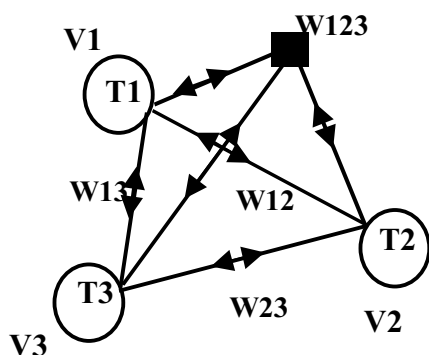
در ادامه مقاله، ابتدا، به معرفی یک شبکه عصبی برگشتی و تابع انرژی مورد نظر برای مسیریابی می‌پردازیم. سپس، عملکرد شبکه عصبی و تاثیر پارامترهای مختلف در آن بررسی می‌شود.

## ۲- شبکه عصبی پیشنهادی

در این بخش، یک شبکه عصبی به نام ابر شبکه با وزنهای نامتقارن<sup>۹</sup> را معرفی می‌کنیم. یکی از شبکه‌های عصبی مفید، شبکه عصبی هاپفیلد است. کاربرد عمده شبکه هاپفیلد در حل مسائل بهینه‌سازی نظیر مشکل فروشنده دوره‌گرد<sup>۱۰</sup> [۱۰] یا  $n$ -وزیر<sup>۱۱</sup> [۱۱] است، اگرچه، به عنوان حافظه رطبی نیز کاربرد دارد [۱۲]. به طور کلی، در حل بسیاری از مسائل بهینه‌سازی<sup>۱۲</sup> می‌توان از حداقل کردن یک تابع انرژی استفاده کرد. به این



شکل ۱- شبکه هاپفیلد با سه نرون



شکل ۲- شبکه HNN با سه نرون

یک ابرشاخه سه پایانه‌ای است. در شبکه HNN هر نرون، نه تنها از نرونهای دیگر، بلکه از حاصلضرب آنها نیز ورودی می‌گیرد. همان طور که در شکل‌های (۱) و (۲) ملاحظه می‌شود، شبکه HNN تعداد بیشتری وزن در مقایسه با شبکه هاپفیلد استفاده می‌کند که باعث افزایش درجه آزادی این سیستم می‌شود. به هر حال، در شبکه HNN هنوز محدودیت تقارن وزنها وجود دارد.

شبکه عصبی پیشنهادی AWHNN دارای ساختار مشابه با شبکه HNN است لیکن وزنه‌های این شبکه نامتقارن بوده و طبیعتاً معادله (۲) را ارضاء نمی‌کند. شکل (۳)، یک شبکه AWHNN با دو نرون را نشان می‌دهد.  $W_{ij..k}$  وزن مربوط به یک ورودی نرون  $i$  است که از حاصلضرب خروجی نرونهای  $j$  تا  $k$  به دست می‌آید. شکل (۴) ساختار نرون شماره ۱ در شکل (۳) را نشان می‌دهد. لذا می‌توان معادله (۳) را برای شبکه AWHNN در نظر گرفت

ترتیب، برای شبکه یک تابع انرژی با توجه به محدودیتها و نیازهای مسئله تعریف می‌شود و سپس از روی آن تابع، وزنه‌های شبکه هاپفیلد مشخص می‌شود. پس از آنکه شبکه به حالت پایدار رسید، تابع انرژی اولیه، به یک حداقل همگرا شده و خروجی نرونها جواب مسئله خواهد بود.

یک شبکه عصبی دیگر که بر اساس شبکه هاپفیلد تعریف می‌شود، ابرشبکه<sup>۱۳</sup> [۱۳] است که آن را شبکه عصبی هاپفیلد تعمیم یافته<sup>۱۴</sup> [۱۴] هم می‌نامند. برای تعریف یک HNN ابتدا یک ابرگراف همبند<sup>۱۵</sup>،  $G = (\bar{V}, \bar{E})$ ، را در نظر می‌گیریم که مجموعه محدود  $\bar{V}$ ، رئوس گراف و مجموعه  $\bar{E} \subseteq 2^{\bar{V}}$  مجموعه ابرشاخه<sup>۱۶</sup> های گراف‌اند. یک ابرشاخه با دو عضو، یک ابرشاخه دوپایانه‌ای است. یک ابرشاخه با بیش از دو عضو، یک ابرشاخه چندپایانه‌ای نامیده می‌شود. بر این اساس، تابع انرژی،  $E$ ، یک HNN به صورت زیر تعریف می‌شود [۱۳]

$$E = K - \sum_{i=1}^N T_i V_i - \frac{1}{2!} \sum_{\substack{i,j \\ i \neq j}} W_{ij} V_i V_j -$$

$$\frac{1}{3!} \sum_{\substack{i,j,k \\ i \neq j \neq k}} W_{ijk} V_i V_j V_k - \dots \quad (1)$$

که در معادله (۱)،  $K$ ، مقداری ثابت و  $V_i$  خروجی نرون  $i$ ام،  $T_i$  سطح آستانه نرون  $i$ ام،  $W_{ij}$  وزن یک ابرشاخه دوپایانه‌ای و به همین ترتیب  $W_{ijk} \dots$  وزن یک ابرشاخه چندپایانه‌ای است. علاوه بر این،  $N$  اندازه مجموعه  $\bar{V}$  در ابرگراف  $G = (\bar{V}, \bar{E})$  است و

$$W_{ii...i} = 0, \quad W_{ij..i} = W_{ji..i} = \dots = W_{1...ji} \quad (2)$$

قابل توجه است که تابع انرژی یک شبکه هاپفیلد، تنها از سه عبارت اول معادله (۱) تشکیل شده است. به این ترتیب، شبکه هاپفیلد، حالت خاصی از شبکه HNN است که در آن، تنها خطوط ارتباطی با دو پایانه وجود دارد.

شکل (۱)، یک شبکه هاپفیلد با سه نرون را نشان می‌دهد. شکل (۲) نیز، یک شبکه عصبی با سه نرون از نوع HNN را نشان می‌دهد. مستطیل تیره‌رنگ در شکل (۲)، وزن مرتبط با

اثبات - می‌توانیم مشتق انرژی نسبت به زمان را به صورت زیر بنویسیم

$$\frac{dE}{dt} = \sum_i \frac{\partial E}{\partial V_i} \cdot \frac{dV_i}{dt} \cdot \frac{dU_i}{dt} \quad (4)$$

با توجه به فرض  $\frac{dU_i}{dt} = -\frac{\partial E}{\partial V_i}$ ، خواهیم داشت

$$\frac{dE}{dt} = -\sum_i \left(\frac{dU_i}{dt}\right)^2 \cdot \frac{dV_i}{dU_i} \quad (5)$$

از آنجا که تابع  $f(\cdot)$  غیر نزولی است،  $\frac{dV_i}{dU_i}$  همواره غیرمنفی

است و در نتیجه  $\frac{dE}{dt} \leq 0$  خواهد بود. برای توابع  $f(\cdot)$  گسسته

نیز می‌توان از اثبات مشابهی استفاده کرد [۱۱]. براین اساس، مقدار تابع انرژی شبکه AWHNN با هر بار تغییر وضعیت نرونی کم می‌شود و از آنجا که تعداد وضعیتهای ممکن برای نرونهای شبکه محدود است تابع انرژی شبکه دارای مقدار حداقلی است که به آن همگرا می‌شود.

در تعیین وزنهای شبکه AWHNN و برای حل مسئله

همگرایی بایستی سعی شود تا شرط  $\frac{dU_i}{dt} = -\frac{\partial E}{\partial V_i}$  برقرار شود.

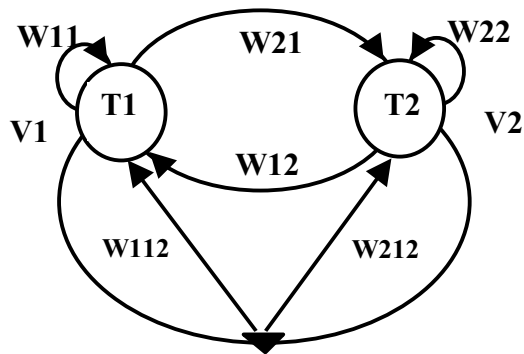
### ۳- تابع انرژی برای مسیریابی در شبکه اتصالات

#### توری

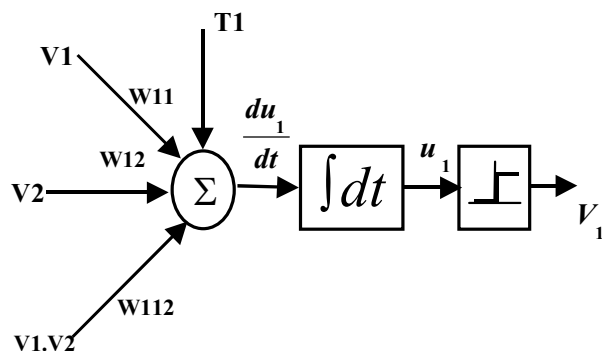
در این بخش کاربرد شبکه AWHNN در حل یک مسئله بهینه‌سازی معرفی می‌شود. مسئله مورد نظر، مسیریابی بهینه در یک شبکه توری است. به این منظور ابتدا تابع هدف مسئله (تابع انرژی شبکه) و سپس کارایی شبکه در رابطه با این مسئله بررسی می‌شود.

#### ۳-۱- تابع انرژی برای مسیریابی در یک شبکه توری

هدف، تعیین تابع انرژی ای است که برای مسیریابی در شبکه توری دو بعدی به کار می‌آید. شکل (۵) یک شبکه توری دو بعدی را نشان می‌دهد. شبکه توری دو بعدی را که در هر کدام از ابعادش  $2^n$  گره وجود دارد در نظر می‌گیریم. در این شبکه،



شکل ۳- یک شبکه AWHNN با دو نرون



شکل ۴- ساختار نرون شماره ۱ در شکل (۳)

$$\frac{dU_i}{dt} = T_i + \sum_{j=1}^N W_{ij} V_j + \sum_{j=1}^N \sum_{k=1}^N W_{ijk} V_j V_k +$$

$$\sum_{j=1}^N \sum_{k=1}^N \sum_{h=1}^N W_{ijkh} V_j V_k V_h + \dots \quad (3)$$

که  $U_i$  ورودی نرون نام است و شبکه زمان پیوسته در نظر گرفته شده است. عمل شبکه AWHNN مانند شبکه هاپفیلد با انجام تغییر حالت در نرونهای شبکه صورت گرفته تا نهایتاً به حالتی دائمی همگرا شود. قضیه (۱) همگرایی شبکه عصبی را اثبات می‌کند.

**قضیه ۱-** فرض کنید  $U_i$ ، ورودی نرون  $i$ ،  $V_i = f(U_i)$  که  $f(\cdot)$  تابعی غیرنزولی است و  $E$ ، تابع انرژی یک شبکه عصبی AWHNN باشد، در این صورت اگر  $\frac{dU_i}{dt} = -\frac{\partial E}{\partial V_i}$ ، آن گاه،  $\frac{dE}{dt} \leq 0$  خواهد بود و  $E$  به یک مقدار حداقل همگرا می‌شود.

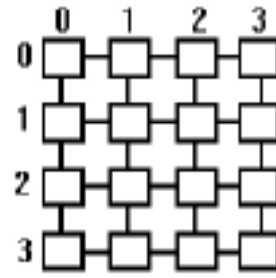
$$K_2 \left[ \left( \sum_{i=0}^{n-1} 2^i N_i^V - \sum_{i=0}^{n-1} 2^i C_i^V \right)^2 + \left( \sum_{i=0}^{n-1} 2^i N_i^H - \sum_{i=0}^{n-1} 2^i C_i^H \right)^2 - 1 \right]^2 \quad (10)$$

معادله (۱۰)، تابع انرژی یک شبکه عصبی از نوع AWHNN است که نرونهای آن،  $N_i^V$  ها و  $N_i^H$  ها هستند. به این ترتیب، یک شبکه توری با  $2^{2n}$  گره، تنها به  $2n$  نرون نیاز دارد و این در مقایسه با روشهای قبلی که به  $2^{4n}$  نرون نیاز دارند، مزیت بزرگی محسوب می‌شود. آدرس گره بعدی  $(N^V, N^H)$  می‌تواند کوچکتر یا بزرگتر از آدرس گره جاری و یا آدرس گره مقصد باشد و در نتیجه استفاده از توان دوم یا قدر مطلق تفاضل این آدرسها در تابع انرژی اجتناب ناپذیر است. قابل توجه است که در تابع انرژی معادله (۱۰)، مربع فاصله بین دو گره منظور می‌شود چراکه به این ترتیب محاسبه مشتق تابع انرژی به منظور محاسبه وزنهای بسیار ساده تر است.

با کمک قضیه (۱) و معادلات (۳) و (۱۰)، می‌توان وزنهای شبکه عصبی AWHNN را برای حل مسئله مسیریابی محاسبه کرد. اگر  $(C^V, C^H)$  آدرس گره  $C$  و  $(D^V, D^H)$  آدرس گره  $D$  در شبکه توری باشد، وزنهای مربوطه را می‌توان محاسبه کرد. نتایج این محاسبات در جدول (۱) نشان داده شده است. همان طور که در جدول (۱) دیده می‌شود، آدرس گره مقصد تنها در سطوح آستانه نرونها ظاهر می‌شود. با توجه به این مطلب، پیاده سازی شبکه عصبی در هر گره ساده خواهد بود چرا که، وزنهای شبکه عصبی، تنها بر اساس آدرس گره فعلی تعیین می‌شوند و هرگز تغییر نمی‌کنند. آنچه در هر بار استفاده از شبکه تغییر می‌کند، آدرس گره مقصد است و آن نیز تنها در سطوح آستانه نرونها ظاهر می‌شود و می‌تواند به عنوان یک ورودی برای شبکه در نظر گرفته شود.

### ۲-۳- بررسی عملکرد شبکه عصبی

در این بخش به بررسی کارایی روش عصبی ارائه شده



شکل ۵- یک Mesh دوبعدی

هر گره را با آدرسی به شکل  $X = (X^V, X^H)$  نشان می‌دهیم که  $X^V$  و  $X^H$ ، هر کدام عددی  $n$  بیتی است و می‌توان نوشت

$$X^V = 2^{n-1} X_{n-1}^V + 2^{n-2} X_{n-2}^V + \dots + X_0^V \quad (6)$$

$$X^H = 2^{n-1} X_{n-1}^H + 2^{n-2} X_{n-2}^H + \dots + X_0^H$$

حال، پیام  $M$  را در نظر گرفته که در گره  $C$  است و به مقصد  $D$  می‌رود. در این صورت، گره  $N$  که پیام به آن می‌رود به شرطی انتخاب می‌شود که

(الف) فاصله پیام با مقصد کمتر شود.

(ب)  $C$  و  $N$  همسایه باشند.

ابتدا، تابع  $d(X, Y)$  را تعریف می‌کنیم که مربع فاصله دو گره  $X$  و  $Y$  را به ما می‌دهد

$$d(X, Y) = (X^V - Y^V)^2 + (X^H - Y^H)^2 \quad (7)$$

گره های  $N$  و  $C$  همسایه اند اگر، فاصله آنها برابر واحد باشد، در نتیجه

$$d(N, C) = 1 \Rightarrow (N^V - C^V)^2 + (N^H - C^H)^2 = 1 \quad (8)$$

با توجه به شرایط (الف) و (ب) و معادلات (۷) و (۸) می‌توان تابع انرژی را به صورت زیر نوشت

$$E = K_1 d(N, D) + K_2 [d(N, C) - 1]^2 \quad (9)$$

که با جایگزینی خواهیم داشت

$$E = K_1 [(N^V - D^V)^2 + (N^H - D^H)^2] + K_2 [(N^V - C^V)^2 + (N^H - C^H)^2 - 1]^2 = K_1 \left[ \left( \sum_{i=0}^{n-1} 2^i N_i^V - \sum_{i=0}^{n-1} 2^i D_i^V \right)^2 + \left( \sum_{i=0}^{n-1} 2^i N_i^H - \sum_{i=0}^{n-1} 2^i D_i^H \right)^2 \right] +$$

جدول ۱- وزنهای شبکه عصبی AWHNN برای مسیریابی در یک شبکه اتصالات توری

$W_{N_i^H N_j^H} = -[2^{i+j+1}(K_1 - 2K_2) + 2^{i+j+2} K_2 \times (2^{2j} - 2^{j+1} C^H + 3C^{H^2} + C^{V^2} - 2^j C^H)]$	$W_{N_i^V N_j^V} = -[2^{i+j+1}(K_1 - 2K_2) + 2^{i+j+2} K_2 \times (2^{2j} - 2^{j+1} C^V + 3C^{V^2} + C^{H^2} - 2^j C^V)]$
$W_{N_i^H N_j^V} = K_2 \times C^H \times 2^{i+j+2} (2^j - 2C^V)$	$W_{N_i^V N_j^H} = K_2 \times C^V \times 2^{i+j+2} (2^j - 2C^H)$
$W_{N_i^H N_j^H N_k^H} = -2^{i+j+k+2} K_2 (2^k + 2^{j+1} - 3C^H)$	$W_{N_i^V N_j^V N_k^V} = -2^{i+j+k+2} K_2 (2^k + 2^{j+1} - 3C^V)$
$W_{N_i^H N_j^H N_k^V} = -2^{i+j+k+2} K_2 (2^k - 3C^V)$	$W_{N_i^V N_j^V N_k^H} = -2^{i+j+k+2} K_2 (2^k - 3C^H)$
$W_{N_i^H N_j^V N_k^V} = 2^{i+j+k+2} K_2 \times C^H$	$W_{N_i^V N_j^H N_k^H} = 2^{i+j+k+2} K_2 \times C^V$
$W_{N_i^H N_j^H N_k^H N_h^H} = -2^{i+j+k+h+2} K_2$	$W_{N_i^V N_j^V N_k^V N_h^V} = -2^{i+j+k+h+2} K_2$
$W_{N_i^H N_j^H N_k^V N_h^V} = -2^{i+j+k+h+2} K_2$	$W_{N_i^V N_j^V N_k^H N_h^H} = -2^{i+j+k+h+2} K_2$
$T_{N_i^H} = K_2 \times 2^{i+2} (C^H - C^{H^3} - C^H C^{V^2}) - 2^{i+1} K_1 D^H$	$T_{N_i^V} = K_2 \times 2^{i+2} (C^V - C^{V^3} - C^V C^{H^2}) - 2^{i+1} K_1 D^V$

متوقف می‌کنیم که

$$\sum \Delta U_i(t) < Err \quad (14)$$

و Err نیز یکی از پارامترهای شبکه عصبی خواهد بود که در این بخش تاثیر آن را بر کارایی مطالعه می‌کنیم. از آنجا که در تابع انرژی ارائه شده، مربع فاصله بین گره بعدی و مقصد  $d(N, D)$  وجود دارد، در مشتق تابع انرژی و در نتیجه در  $\Delta U_i(t)$  تاثیر از فاصله بین مبدا و مقصد وجود خواهد داشت. نتایج اولیه شبیه‌سازی نشان داده است که استفاده از معادله (۱۴) مناسب نیست و رابطه زیر برای توقف محاسبات نتایج بهتری از خود نشان می‌دهد

$$\sum \Delta U_i(t) < Err \times dist \quad (15)$$

که  $dist$  فاصله (و نه مربع فاصله که در تابع انرژی به کار رفته بود) بین مبدا  $C$  و مقصد  $D$  است. و البته هر چه مقدار  $\Delta t$  کمتر باشد تاثیرات  $dist$  کمتر است. ضریب Err در رابطه (۱۵)، با تعداد نرونها متناسب خواهد بود، زیرا در طرف دیگر نامساوی، مجموع تغییرات ورودیهای تمام نرونها را قرار داده‌ایم. به این ترتیب، انتظار داریم که با افزایش اندازه شبکه

می‌پردازیم و تاثیر پارامترهای  $K_1$ ،  $K_2$  و اندازه ابعاد شبکه اتصالات توری (و به تبع آن تعداد نرونهای شبکه عصبی) بر روی کارایی را مطالعه می‌کنیم. برای شبیه‌سازی (یا پیاده‌سازی) این شبکه عصبی، از آنجا که هدف حداقل کردن تابع انرژی  $E$  است، باید تغییرات انرژی نسبت به زمان را صفر کنیم. اگر از روشهای عددی برای این منظور استفاده شود به معادله زیر می‌رسیم [۱۱]

$$U_i(t + dt) = U_i(t) + \frac{dU_i}{dt} \cdot dt \quad (11)$$

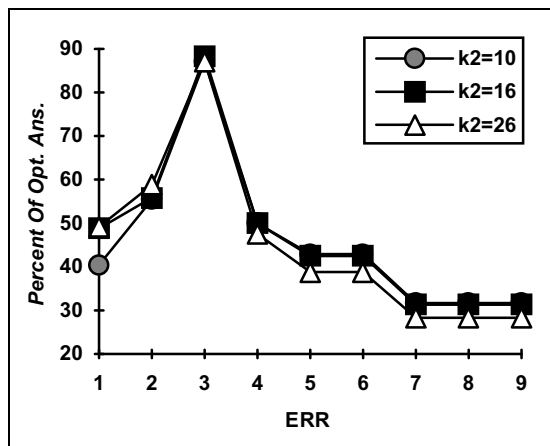
که در آن  $\frac{dU_i}{dt}$  از معادله (۳) به دست می‌آید. اگر زمان را به قسمتهای کوچک  $\Delta t$  تقسیم کنیم خواهیم داشت

$$U_i(t + \Delta t) = U_i(t) + \Delta U_i(t) \cdot \Delta t \quad (12)$$

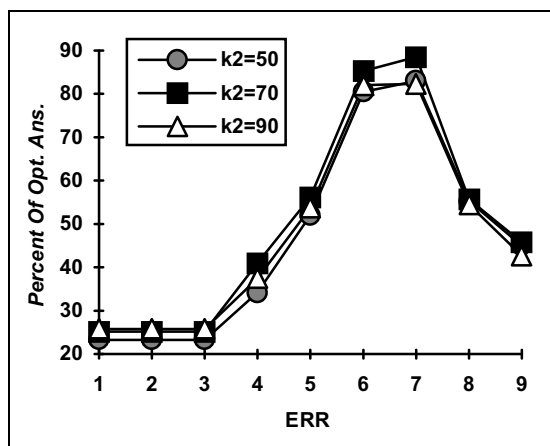
سپس با استفاده از معادله زیر وضعیت نرونها را تغییر می‌دهیم

$$\text{If } U_i > 0 \text{ then } V_i = 1 \text{ Else } V_i = 0 \quad (13)$$

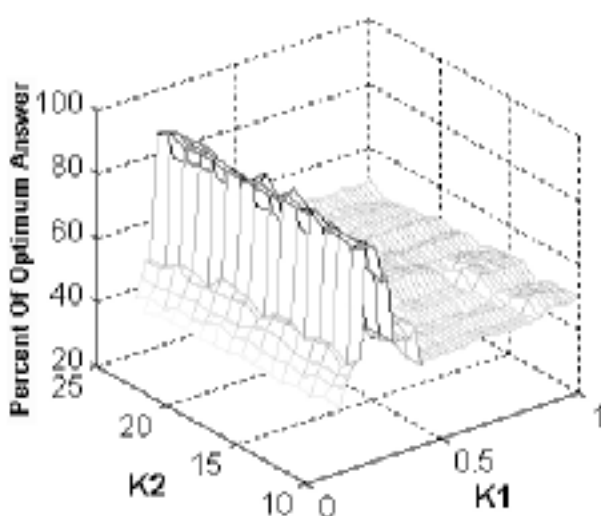
شبکه تا جایی باید به تغییر وضعیت نرونها ادامه دهد که  $\Delta U_i(t)$  ها صفر شوند. در عمل به دلیل گسسته بودن زمان ممکن است به این وضعیت نرسیم. پس، عملیات را هنگامی



شکل ۶- تغییرات درصد جوابهای درست نسبت به تغییر در مقدار Err برای یک شبکه توری دوبعدی با ۴ گره در هر بعد



شکل ۷- تغییرات درصد جوابهای درست نسبت به تغییر در مقدار Err برای یک شبکه توری دوبعدی با ۵ گره در هر بعد

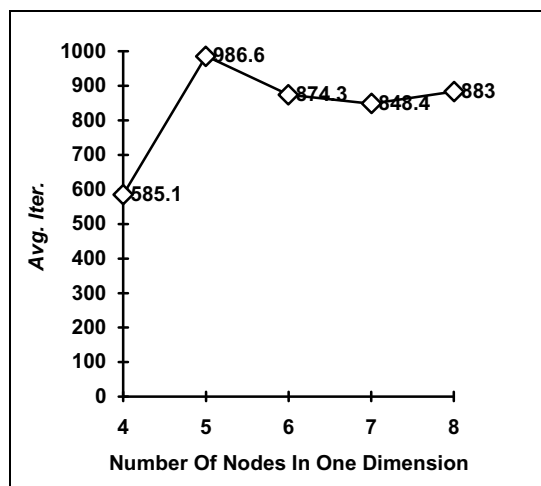


شکل ۸- درصد جوابهای درست برحسب تغییرات k1 و k2

اتصالات و به تبع آن افزایش تعداد نرونها، مقدار مناسب برای Err نیز زیاد شود چرا که در این صورت، طرف چپ نامساوی رابطه (۱۵) افزایش یافته است. نتایج شبیه‌سازیها این مطلب را تایید می‌کند. در شکل‌های (۶) و (۷) نمونه‌ای از این نتایج دیده می‌شود. در شکل (۶)، تغییرات درصد جوابهای درست نسبت به تغییر در مقدار Err برای یک شبکه اتصالات توری دو بعدی با چهار گره در هر بعد ترسیم شده است. همان طور که دیده می‌شود، بهترین جواب در نزدیکی مقدار  $Err=3$  به دست می‌آید. شکل (۷) همین نمودار را برای یک شبکه توری دو بعدی با پنج گره در هر بعد نشان می‌دهد و می‌بینیم که این بار  $Err=7$  بهترین جواب را می‌دهد. در هر دو نمودار مقدار  $K_1$  ثابت در نظر گرفته شده و مقادیر مختلفی به  $K_2$  نسبت داده شده است. قابل توجه است که،  $K_1$  و  $K_2$  استفاده شده در این نمودارها الزاما مناسبترین مقادیر نیستند و صرفا تاثیر تغییر در Err مورد نظر بوده است.

در تابع انرژی عبارتی که ضریب  $K_1$  دارد تاثیرات نامناسبی روی کارایی شبکه عصبی ایجاد می‌کند. مقدار این عبارت در مقایسه با عبارتی که ضریب  $K_2$  دارد و همسایه بودن گره بعدی با گره جاری را تضمین می‌کند، بسیار بزرگ است. عبارت با ضریب  $K_2$  دارای مقدار حداقل صفر است در صورتی که مربع فاصله دو گره در شبکه توری دو بعدی با چهار گره در هر بعد می‌تواند تا هجده هم برسد. لازم است که برای رفع این مشکل نسبت  $K_2$  به  $K_1$  بزرگ انتخاب شود. با بزرگتر شدن ابعاد شبکه، این نسبت باید افزایش یابد زیرا میانگین مربع فاصله دو گره در شبکه اتصالات افزایش می‌یابد. شکل (۸)، نموداری سه بعدی است که درصد جوابهای درست را برحسب تغییرات  $K_1$  و  $K_2$  نشان می‌دهد. این نمودار برای یک شبکه توری دو بعدی با چهار گره در هر بعد رسم شده است.

بهترین مقادیری که برای این دو ضریب از روی نمودار می‌توان انتخاب کرد،  $K_1=0.25$  و  $K_2=16$  است. مقدار Err استفاده شده در این نمودار 2.5 است. بزرگ بودن  $K_2$  نسبت

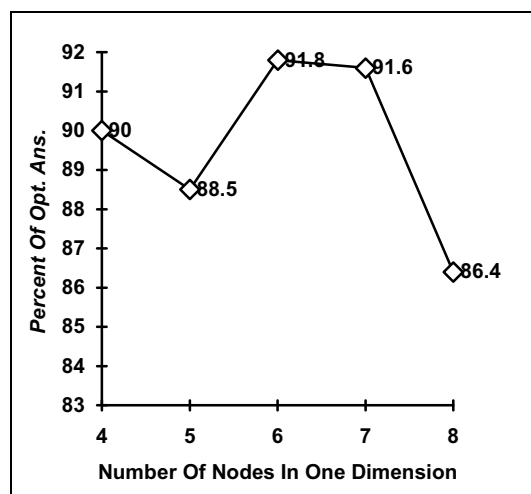


شکل ۱۰- تعداد متوسط تکرار محاسبات در هر بار محاسبه آدرس گره بعدی بر حسب تغییر در اندازه شبکه

تعداد بسیار کمتری نرون نسبت به شبکه های عصبی قبلی نیاز دارد. برای مسیریابی در یک شبکه اتصالات توری  $2^n \times 2^n$  با استفاده از روش ارائه شده، تنها به  $2n$  نرون نیاز است در حالی که روشهای موجود به  $2^{4n}$  نرون نیاز دارند. در این روش، آدرس گره مقصد که برای هر پیام متفاوت است و تنها در سطح آستانه نرون ها ظاهر شده و در خروجی هیچ نروسی ضرب نمی شود. این امر باعث ساده شدن پیاده سازی شبکه عصبی می شود. همچنین، در هر بار محاسبه مسیر، به جای تعیین کل مسیر، تنها آدرس گره بعدی که پیام باید به آن برود به دست می آید. روش مسیریابی ارائه شده در این مقاله می تواند به سادگی برای شبکه های اتصالات توری با ابعاد بیشتر نیز استفاده شود و تنها نیاز به تغییراتی جزئی در تابع انرژی دارد. روش ارائه شده برای مسیریابی، تنها آدرس مبدا و مقصد را در تصمیم گیری خود دخالت می دهد و اطلاعات مربوط به ترافیک و خرابی در شبکه را در نظر نمی گیرد. با استفاده از شبکه عصبی ارائه شده در این مقاله روش مسیریابی تحمل پذیر خطایی در [۱۵] ارائه شده است.

به  $K_1$  صحت بحث انجام شده روی ویژگیهای دو ضریب را تایید می کند. با تغییر اندازه شبکه اتصالات، بهترین مقدار برای  $Err$ ،  $K_1$  و  $K_2$  می تواند با استفاده از آزمون و خطا و به منظور افزایش درصد جوابهای بهینه تعیین می شود.

شکلهای (۹) و (۱۰) به ترتیب، تغییر در درصد جوابهای درست و تغییر در تعداد متوسط تکرار محاسبات<sup>۱۷</sup> در هر بار محاسبه آدرس گره بعدی را بر حسب تغییر در اندازه شبکه اتصالات نشان می دهند. مشاهده می شود که در روش ارائه شده همواره درصد جوابهای بهینه بیش از ۸۶٪ است. همچنین به نظر می رسد که متوسط تکرار در محاسبات با تعداد نرونها متناسب باشد.



شکل ۹- درصد جوابهای درست بر حسب تغییر در اندازه شبکه

## نتیجه گیری

شبکه اتصالات توری، یکی از پر کاربردترین شبکه های اتصالات در سیستمهای چند رایانه است. در این مقاله، پس از تعریف یک شبکه عصبی برگشتی، تابع انرژی برای مسیریابی در شبکه توری دو بعدی ارائه شد. شبکه عصبی معرفی شده در این مقاله از شبکه عصبی هاپفیلد الهام گرفته شده است و محدودیتهای شبکه هاپفیلد در انتخاب وزنها و اتصال نرونها را ندارد. به این ترتیب، شبکه عصبی ارائه شده، می تواند برای پیاده سازی تعداد بیشتری از توابع انرژی استفاده شود. شبکه عصبی ای که برای مسیریابی پیام در این مقاله ارائه شده است به



- |                   |   |   |
|-------------------|---|---|
| 1. routing        | 8. hypercube                                      | 13. Hyper Neural Network (HNN)          |
| 2. multi-computer | 9. Asymmetric Weight Hyper-Neural Network (AWHNN) | 14. generalized hopfield neural network |
| 3. neural network | 10. Traveling Salesman Problem (TSP)              | 15. connected hypergraph                |
| 4. recurrent      | 11. N-Queen                                       | 16. hyper-edge                          |
| 5. mesh           | 12. optimization                                  | 17. iteration                           |
| 6. Hopfield       |   |   |
| 7. neuron         |   |   |

## مراجع

1. سماوی، ش.، و خدیوی، پ.، "مسیریابی در شبکه Mesh با استفاده از شبکه‌های عصبی،" مجموعه مقالات هشتمین کنفرانس مهندسی برق ایران، جلد اول (الکترونیک و کامپیوتر)، صص. ۳۳۴-۳۴۰، دانشگاه صنعتی اصفهان، اردیبهشت ۱۳۷۹.
2. Rauch, H. E., and Winarske, T., "Neural Networks For Routing Communications Traffic," *IEEE Cont. Syst. Mag.*, pp. 26-31, April 1988.
3. Kamoun, F., and Mehmet Ali, M. K., "A Neural Network Shortest Path Algorithm For Optimum Routing In Packet-Switched Communication Networks," *Global Telecommunication Conference*, Vol. 1, pp. 120-124, 1991.
4. Baba, T., and Funabili, N., "A Routing Algorithm Based On Greedy Neural Network For Non-Sequential Activation Scheduling Problems," *IEEE International Conference On Systems, Man, and Cybernetics*, Vol. 3, pp. 2285-2290, 1998.
5. Pierre, S., Said, H., and Probst, W. G., "A Neural Network Approach For Routing In Computer Networks," *IEEE Canadian Conference On Electrical and Computer Engineering*, Vol. 2, pp. 826-829, 1998.
6. Wang, J., "A Dual Neural Network For Shortest Path Routing," *International Conference On Neural Network*, Vol. 2, pp. 1295-1298, 1997.
7. Tanenbaum, A. S., *Computer Networks*, New Jersey, Prentice-Hall, 1996.
8. Boppana, R. V., and Chalasani, S., "Fault Tolerant Wormhole Routing Algorithms For Mesh Networks," *IEEE Tran. On Computers*, Vol. 44, No. 7, pp. 848-864, July 1995.
9. Talia, D., "Message Routing Systems For Transputer Based MultiComputers," *IEEE Micro*, pp. 62-72, June 1993.
10. Abe, S., *Neural Networks and Fuzzy Systems*, Boston: Kluwer Academic Publishers, 1997.
11. Takefuji, Y., *Neural Network Parallel Computing*, Boston: Kluwer Academic Publishers, 1992.
12. Aleksander, I., and Morton, H., *An Introduction To Neural Computing*, Chapman & Hall, 1991.
13. Rai, S., and Deng, W., "Hyper Neural Network-An Efficient Model For Test Generation in Digital Circuits," *IEEE Tran. On Computers*, Vol. 45, No. 1, pp. 115-121, Jan. 1996.
14. Ortega, J., Prieto, A., Lloris, A., and Pelayo, F. J., "Generalized Hopfield Neural Network For Concurrent Testing," *IEEE Tran. On Computers*, Vol. 42, No. 8, pp. 898-912, August 1993.
15. سماوی، ش.، و خدیوی، پ.، "مسیریابی تحمل پذیر خطا در شبکه های اتصالات توری با استفاده از تحلیل کمان،" نشریه بین المللی علوم و مهندسی، دانشگاه علم و صنعت، ۱۳۸۰.