

یک روش آموزش نوین در شبکه عصبی آشوبگونه

حامد رحیم اف^{*}، محمدرضا جاهدمطلق[†]، ناصر مزیننی[‡]

چکیده

از جمله سیستم‌های حافظه انجمنی مبتنی بر پویایی غیر تعادلی، مدل S-GCM است. این مدل نسبت به شبکه هاپفیلد از توانایی بالاتر ذخیره سازی و نرخ موفق تشخیص برخوردار است، ولی هنوز، به عنوان یک سیستم حافظه انجمنی دارای نقاط ضعفی است و آن نرخ ذخیره سازی و سرعت همگرایی پایین آن می‌باشد. در این مقاله، برای رفع مشکل S-GCM فرایند آموزش این شبکه را مورد توجه قرار داده ایم. از این رو بر اساس روش آموزش اسپارس و با در نظر گرفتن نحوه یادگیری انسان که امکان فراموشی آنچه که تکرار می‌گردد کم می‌شود، یک روش آموزش جدید در مدل S-GCM ارائه کرده و همگرایی آن را اثبات نموده ایم. نشان داده ایم که روش ما، جهت سنتز شبکه S-GCM در مقایسه با قانون یادگیری هب از گنجایش بیشتر ذخیره سازی اطلاعات و سرعت همگرایی بالاتری برخوردار می‌باشد.

کلمات کلیدی

حافظه انجمنی آشوبگونه، نگاشت بهم پیوسته سراسری متقارن، روش آموزش MIMS (More Iterate More Store)

A new learning method for chaotic neural network

Hamed Rahimov, Mohammad-Reza Jahedmotlagh, Nasser Mozayani

Iran University of Science and Technology, Computer Engineering Faculty

Abstract

Recently, there have been many studies on artificial neural network models with non-equilibrium dynamics. One of these models is S-GCM. In this paper, a new learning method for S-GCM is proposed. In the proposed method, we use modified sparse matrix for learning method. Both the theory analyses and computer simulation results show that the performance of S-GCM can be improved greatly by using our learning method. The method is like sparse method, with difference in sparse method. This method to recur the stored patterns and in result it will be dependent on the sequence of storing the patterns, on the other hand, primary patterns have more effect in creating the weight matrix in comparison to the patterns will be stored finally, it means they are recurred more and consequently they are stored and stick better in the memory. It seems this method of learning is more similar to the man's way of learning, as the patterns which we repetition during time we will keep them in our long-term memory better.

Keywords

Chaotic Associative Memory, Symmetric Globally Coupled Map (S-GCM), More Iterate More Store (MIMS) Learning

* Email Address: Hrahimov@iust.ac.ir

† Email Address: Jahedmr@iust.ac.ir

‡ Email Address: Mozayani@iust.ac.ir

۱- مقدمه

امروزه رفتار دینامیک آشوبگونه توجه فراوانی را در بسیاری از زمینه‌های تحقیقاتی به خود معطوف ساخته است که این امر سبب پیشرفت قابل ملاحظه‌ای در مطالعه آشوب شده است [1-7]. علاوه بر این، بر محققین مسلم شده است که رفتار دینامیک‌های آشوبگونه، نقش عمده‌ای را در نرونها و شبکه‌های عصبی بیولوژیک ایفا می‌کنند [8,9]. از این رو محققین در تلاشند تا بتوانند به کمک دینامیک آشوبگونه، رفتار نرونهای واقعی را با نرونهای مصنوعی مدل کنند. بنابراین تحقیق بر روی پردازش هوشمند اطلاعات مبتنی بر مغز، لزوماً یک طرح اساسی تلقی می‌گردد. باید در نظر داشت که مطالعه بر روی شبکه‌های عصبی آشوبگونه، این مطلب را پر رنگ تر می‌سازد که چنین شبکه‌هایی تنها به منظور مطالعه و شبیه‌سازی سیستم‌های عصبی و حسی بکار گرفته نمی‌شوند بلکه علاوه بر این ما را به سوی کاربردهای مهندسی مهم و مقرون بصرفه سوق می‌دهند. از اینرو، شبکه‌های عصبی آشوبگونه برای حل مسایل بفرنج بهینه‌سازی، طبقه‌بندی الگوها بر اساس دینامیک انجمنی و کاربردهای دیگری که به نظر می‌رسد در آینده نزدیک مطرح گردند، بکار گرفته می‌شوند [10].

Freeman [8,9] یک شبیه‌سازی کامپیوتری از دینامیک یک مدل توزیع شده از سیستم بویایی را مطرح کرد. مطالعه این سیستم نقش آشوب در سیستم‌های تشخیص بیولوژیک و بطور خاص نقش آشوب در تشخیص و تمایز بو در سیستم بویایی را نشان می‌دهد.

پس از نظریه وی، سیستم‌های حافظه انجمنی زیادی بر اساس دینامیک غیر تعادلی عمدتاً با قانون یادگیری همبستگی هب پیشنهاد شدند. از طرف دیگر ثابت شد که نوسانگرهای غیرخطی جفت شده با هم، قادر به تولید الگوهای پیچیده فضا-زمانی هستند [11,12]. بر مبنای این نظریه، Kaneko [13] مدل‌هایی را بر اساس عناصر آشوبگونه جفت شده پیشنهاد کرد. سپس Nara [14] سیستم حافظه انجمنی را بر اساس شبکه عصبی نامتقارن پیشنهاد نمود. Aihara [15,16]، با استفاده از معادلات تفاضلی قطعی، یک مدل آشوبگونه را ارائه کرد. پیچیدگی فضا-زمانی در این مدل توسط ساختار دینامیک شبکه تولید می‌شد. Adachi [17] نیز در همان سال بر مبنای مدل پیشنهادی آیهارا، مدلی پیشنهاد کرد. از سوی دیگر، مدل‌های Kaneko [13] بر پایه عناصر آشوبگونه بهم پیوسته استوار بود که هر عنصر آن شامل یک نگاشت لوجستیک وابسته به زمان و ارتباط آنها از نوع ارتباط سراسری بود. این مدل نگاشت بهم پیوسته سراسری^۱ نامیده می‌شد و ویژگی‌های قابل توجهی داشت. به عنوان مثال دارای انسجام پنهان در حالت آشفته بود. در عین حال این مدل، به دلیل استفاده از نگاشت لوجستیک درجه دوم از توانایی چندانی در پردازش اطلاعات برخوردار نبود. از این رو، Ishii [18] نگاشت بهم پیوسته سراسری متقارن^۲ (S-GCM) را ارائه نمود و از این مدل در کاربردهای پردازش اطلاعات نظیر حافظه انجمنی استفاده کرد.

علاوه بر این Inoue [11] یک نرو-کامپیوتر آشوبگونه^۳ ارائه نمود که در این مدل یک نرون، ترکیبی از یک جفت نوسانگر متصل بود. نرو-کامپیوتر آشوبگونه بر اساس یک قانون قطعی عمل می‌کرد با اینحال توانایی جستجو و نیز حل مسایل بفرنج بهینه‌سازی را دارا بود. سیستم اصلاح شده S-GCM معرفی شده توسط Ishii و نیز نرو-کامپیوتر آشوبگونه معرفی شده توسط Inoue دو شبکه عصبی مصنوعی آشوبگونه رایج و اصلی در زمینه تشخیص الگو و حافظه انجمنی هستند لذا در سال ۹۹، Zhang [19] بر پایه تحلیل‌هایی از مفاهیم شبکه‌های عصبی آشوبگونه ارائه شده پیش از خود و نیز کاربرد آنها در پردازش اطلاعات، یک مدل تک بعدی، با عنوان نگاشت بهم پیوسته دو-مسیره^۴ معرفی نمود. این مدل ترکیبی از مدل نرو-کامپیوتر آشوبگونه و مدل S-GCM بود و تا حدودی بر ضعف مدل S-GCM در سرعت بازیابی اطلاعات و ضعف مدل نرو-کامپیوتر آشوبگونه در نرخ تشخیص داده‌ها غلبه می‌کرد و در عین حال، از توانایی هر دو مدل برخوردار بود.

در ادامه، Ishii [20] با بکارگیری توابع غیر یکنواخت و معادلات تفاضلی در مدل S-GCM باعث بهبود قابل ملاحظه در گنجایش ذخیره اطلاعات مدل S-GCM شد. Zheng [21] نیز در سال ۲۰۰۵ برای غلبه بر ضعف مدل S-GCM در سرعت بازیابی اطلاعات، یک پارامتر کنترلی جدید به این سیستم اضافه نمود و سرعت بازیابی اطلاعات را در مدل S-GCM بهبود بخشید. هرچند مدل S-GCM به لحاظ گنجایش اطلاعات برتری قابل ملاحظه‌ای نسبت به مدل هاپفیلد دارد، با اینحال هنوز مدل S-GCM در مقایسه با مدل هاپفیلد به مراتب از سرعت پایین تری در بازیابی اطلاعات برخوردار است [20,21].

جهت ذخیره‌سازی یک الگو، لازم است که مقدار تابع انرژی در نقطه متناظر با بردار الگو، حداقل باشد. به عبارت دیگر هر یک از الگوهای مورد نظر، یک نقطه حداقل را در منحنی تابع انرژی شبکه، اشغال می‌کند. در همین حال مایلیم که در صورت ذخیره‌سازی الگوهای جدید، الگوهای قدیم به عنوان نقاط حداقل محلی، دستخوش تغییر نگردند. نظر به اینکه ماتریس وزن شامل اطلاعاتی درباره الگوهای ذخیره شده است، می‌خواهیم این ماتریس را به نحوی محاسبه کنیم که مقدار حداقل تابع انرژی را در نقاط متناظر با الگوها به همراه داشته باشد. ثابت شده است که انتخاب ماتریس کوواریانس برای این منظور انتخاب مناسبی است [3]. علاوه بر این روش، روش دیگری نیز جهت ذخیره‌سازی الگوها در دسترس است که می‌تواند الگوها را در نقاط حداقل انرژی خود ذخیره کند. این روش که توسط Menhaj [3] مطرح شده است روش اسپارس نام دارد. نشان داده شده است که این روش جهت سنتز شبکه هاپفیلد در مقایسه با قانون هب از سرعت همگرایی بالاتری برخوردار است [3]. بر پایه این روش آموزش، ما یک روش آموزش جدید را ارائه داده ایم که استفاده از آن در شبکه S-GCM باعث افزایش قابلیت‌های این شبکه شده

می‌سازند. هر مدار یکسان، بسته به مقادیر پارامترها می‌تواند مداری دو سیکله، چهار سیکله، هشت سیکله، ... و یا مدار آشوبگونه باشد [18].
به منظور بکارگیری S-GCM در حافظه انجمنی و شناسایی الگو، ابتدا مشخصات این نگاشت را بیان می‌کنیم. ورودی این نگاشت همان شرایط اولیه سیستم و خروجی آن همان حالت سیستم بعد از چند تناوب زمانی است و بدین ترتیب پردازش اطلاعات توسط این نگاشت انجام می‌شود. مبنای طراحی سیستم‌های حافظه انجمنی بر پایه S-GCM اینست که فرآیند یادآوری الگوهای ذخیره شده به یک فرآیند تکاملی در سیستم بهم پیوسته تبدیل می‌شود.

ویژگی فضا-زمان جذب کننده‌های نگاشت بهم پیوسته سراسری متقارن توسط مقادیر پارامترها تعیین می‌شوند. α پارامتر انشعاب در این نگاشت و بیانگر قدرت آشوب هر نرون بوده و ε بیانگر قدرت اتصالی وزن بین نرونهاست. بنابراین اگر α بزرگ انتخاب شود، نگاشت بهم پیوسته سراسری متقارن، آشوبگونه شده و هنگامیکه ε بزرگ شود، نگاشت بهم پیوسته سراسری متقارن، یکنواخت و پایدار می‌شود.
مدل S-GCM با $\alpha = 3.4$ و $\varepsilon = 0.1$ در چهار خوشه فریز می‌گردد. وقتی پارامترها در این مقادیر تنظیم شوند، هر الگوی خروجی O تقریباً با هر الگوی ورودی I برابر بوده و نگاشت، نگاشتی همانی است. اما اگر $\alpha = 3.9$ و $\varepsilon = 0.05$ باشد هیچ بستگی بین ورودی و خروجی دیده نمی‌شود. در این حالت هر نرون دارای رفتار آشوبگونه بوده و نگاشت، فرایندی تصادفی از الگوی ورودی به الگوی خروجی می‌شود. بنابراین فرایند S-GCM در واقع، تبدیلی از الگوی ورودی به الگوی خروجی با دو وضعیت نگاشت همانی (حفظ الگو) و نگاشت تصادفی (تخریب الگو) در پردازش اطلاعات است.

فرض کنید $\{\chi^1, \chi^2, \dots, \chi^m | \chi^k \in \{1, -1\}^N\}$ مجموعه‌ای از الگوهای یادگیری و χ_i^k مقدار i امین عنصر در k امین الگو باشد. m تعداد الگوها است. قانون یادگیری این الگوها هم، همان قانون یادگیری اصلاح شده هب باشد:

$$w_{ij} = \frac{1}{N} \sum_{k=1}^m \chi_i^k \chi_j^k \quad (2)$$

شبکه نرونی با ساختار زیر یک سیستم حافظه انجمنی است که با استفاده از مدل S-GCM حاصل می‌شود [18]:

$$x_i(t+1) = (1-\varepsilon) \cdot f_i[x_i(t)] + \frac{\varepsilon}{N} \cdot \sum_{j=1}^N f_j[x_j(t)] \quad (3)$$

$$f_i(x_i(t)) = \alpha_i(t) \cdot x_i(t)^3 - \alpha_i(t) \cdot x_i(t) + x_i(t)$$

در این مدل، آشوب هر نرون α_i به صورت زیر کاهش می‌یابد [18]:

$$\alpha_i(t+16) = \alpha_i(t) + [\alpha_i(t) - \alpha_{\min}] \cdot \tanh(\beta \cdot E_i(t)) \quad (4)$$

$$E_i(t) = -x_i(t) \cdot \sum_{j=1}^N w_{ij} \cdot x_j(t)$$

که در آن α_{\min} ، β و ε پارامترهای ثابتی هستند و $E_i(t)$ انرژی داخلی سیستم می‌باشد. در اینجا $\beta = 2$ ، $\alpha_{\min} = 3.4$ و $\varepsilon = 0.1$ در نظر

است. اساس مقاله ما بر پایه بکارگیری این روش آموزش در شبکه S-GCM است که در ادامه به تشریح روش آموزش خود خواهیم پرداخت.

ساختار این مقاله به صورت زیر می‌باشد: در بخش دوم، به اجمال، دینامیک انجمنی آشوبگونه مدل S-GCM را شرح می‌دهیم. در بخش سوم بطور مختصر روش آموزش اسپارس را توضیح می‌دهیم و در ادامه روش آموزش خود را معرفی کرده و بطور کامل تشریح و همگرایی آن را اثبات می‌کنیم. خواهیم دید که استفاده از این روش در شبکه S-GCM در مقایسه با قاعده هب و نیز قاعده اسپارس، کارکرد بهتری دارد. سپس، در بخش چهارم نتایج پیاده سازی‌های مدل S-GCM را با دو روش آموزش هب و روش پیشنهادی مقایسه نموده و معایب و مزایای روش پیشنهادی را عنوان می‌کنیم. در نهایت در آخرین بخش، نتایج مقاله را ارائه خواهیم کرد.

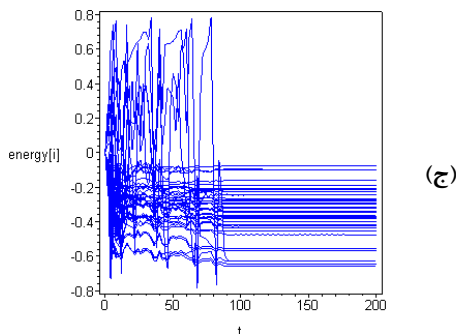
۲- مدل S-GCM با قاعده یادگیری هب

در شبکه‌های حافظه انجمنی معمول، بازیابی نتایج، بطور عمدتاً توسط حالات تعادل پایدار، توصیف می‌شوند. به عبارت دیگر یک حالت پایدار، متناظر با یک الگوی ذخیره شده است. با این حال یک شبکه عصبی آشوبگونه یک سیستم دینامیک است و بطور مشهود نشان دهنده رفتار دینامیک انجمنی غیر پریودیک است و عملیات بازیابی از اطلاعات جزئی اولیه، ممکن است به عنوان یک فرایند دینامیک جستجوی اطلاعات نگریسته شود [10]. این فرایند جستجو می‌تواند به صورت زیر تفسیر گردد: از آنجا که تعریف ماتریس حافظه خود انجمنی، همان قانون اصلاح شده هب و همان تعریفی است که در حافظه خود انجمنی شبکه‌های عصبی معمول بکار می‌رود، بنابراین ورودی‌های فیدبک از نرونهای اصلی بواسطه اتصالات خودبازگشتی سبب می‌گردند که شبکه به یک الگوی ذخیره شده، همگرا گردد. از این رو این ایده به نظر می‌آید که رفتار دینامیک پایدار بهم پیوسته که ماتریس خودانجمنی از خود نشان می‌دهد و نیز رفتار ناپایدار نوسانگرهای دینامیک بهم پیوسته، در مجموع بیانگر فرایند جستجوی حافظه هستند و این فرایند دینامیک جستجوی حافظه می‌تواند به عنوان یک حافظه انجمنی در تحول دینامیک غیرخطی بکار گرفته شود [10].
در مدل توسعه یافته Kaneko که Ishii ارائه نمود از نگاشت لوجستیک درجه سوم به عنوان نوسانگرهای دینامیک استفاده می‌شود [18]:

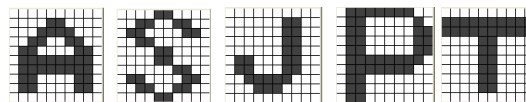
$$x_i(t+1) = (1-\varepsilon) \cdot f[x_i(t)] + \frac{\varepsilon}{N} \cdot \sum_{j=1}^N f[x_j(t)] \quad (1)$$

$$f(x) = \alpha \cdot x^3 - \alpha \cdot x + x \quad x \in [-1, 1]$$

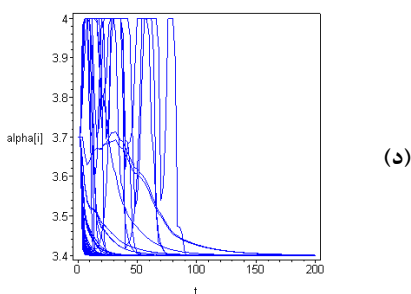
مدل S-GCM دارای دو پارامتر α و ε است. زمانی که این پارامترها مقادیر خاصی داشته باشند، سیستم در بستر جذب خوشه‌ای قرار می‌گیرد [18]. نرونهای متعلق به یک خوشه، مداری یکسان را



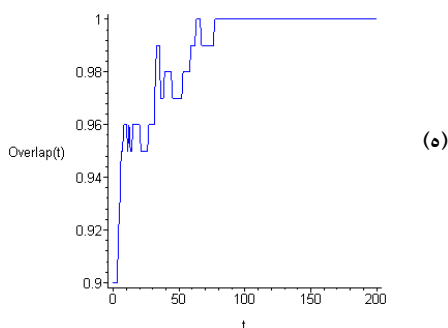
گرفته شده اند. پویایی هر α_i بین $\alpha_{\min} = 4$ و α_{\max} در هر دو واحد زمانی کنترل می‌شود. تابع سیگموئید اعمال شده در مدل، برخلاف مدل هاپفیلد، فقط برای کنترل پارامتر α_i (پارامتر انشعاب) بکار می‌رود و بطور غیر مستقیم روی x_i هر نرون تاثیر می‌گذارد.



شکل (۱): الگوهای یادگیری ۱۰۰ بیتی



جهت ارزیابی فرایند انجمن به ذکر یک مثال در این سیستم می‌پردازیم. در شکل (۱) پنج الگوی صد بیتی دودویی نشان داده شده است که با قانون یادگیری هب به این سیستم آموخته شده اند. در اینجا نیز $\beta = 2 \cdot \alpha_{\min} = 3.4$ و $\varepsilon = 0.1$ در نظر گرفته شده اند. هر یک از الگوها را ۱۰٪ نویزی می‌کنیم و به عنوان ورودی به شبکه می‌دهیم. سیستم، بعد از چندین واحد زمانی قادر به بازیابی الگوهای مورد نظر خواهد بود. در شکل (۲) سری زمانی خروجی نرونها، انرژی، مقادیر α و نیز سری زمانی همپوشانی الگوی هدف با خروجی شبکه مربوط به بازیابی الگوی حرف A که ۱۰٪ نویز تصادفی به آن اعمال شده، نشان داده شده است. در منحنی (ب) سری زمانی خروجی همه نرونها در طول زمان بازیابی، رسم شده است. رفتار آشوبگونه در مراحل اولیه کاملا مشهود است و با گذشت زمان این رفتار آرام می‌شود و سرانجام سیستم در مراحل نهایی انجمن در یک چرخه حدی پایدار که در اینجا یک جذب کننده چهار خوشه ای است، قرار می‌گیرد. در منحنی (ج) نیز این مساله با تغییرات مقدار α در طول فرایند انجمن، قابل بیان است.



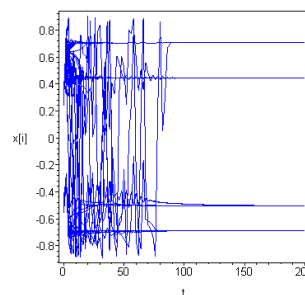
شکل (۲): نگاشت S-GCM با کنترل α (الف) الگوی نویزدار ورودی، (ب) سری زمانی خروجی نرونها، (ج) سری زمانی تابع انرژی، (د) سری زمانی α ، و (ه) سری زمانی همپوشانی خروجی شبکه با الگوی هدف

در این سیستم، اگر E_i را i امین جزء انرژی سیستم بدانیم، هدف کمینه کردن مقدار کل تابع انرژی $(E = \sum_i E_i)$ است [18]. این سیستم، فرایند جستجو برای کمینه محلی را با کوچک و منفی کردن هر جزء انرژی انجام می‌دهد.

منحنی (ج) سری زمانی α_i هر نرون را نشان می‌دهد. همانطور که دیده می‌شود، در مراحل اولیه شناسایی، بعضی از مقادیر α_i بزرگ هستند که باعث می‌شود نرون به دنبال یک وضعیت خاص بگردد. به هر حال، با گذشت زمان، این مقادیر کوچک شده و بالاخره همه آنها برابر با α_{\min} خواهند شد که موجب می‌شود سیستم، معادل S-GCM گشته و همانطور که در منحنی (ه) نشان داده شده، الگوی هدف را بازیابی کند.



(الف)



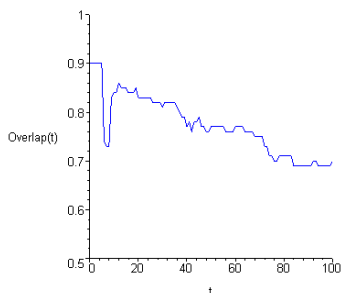
۳- روش آموزش پیشنهادی (MIMS)

ثابت شده است که انتخاب ماتریس کوواریانس جهت آموزش الگوها به شبکه حافظه انجمنی انتخاب مناسبی است [3]. علاوه بر این روش، روش‌های دیگری نیز جهت ذخیره سازی الگوها در دسترس است. به عنوان مثال روش اسپارس که در مرجع [3] آورده شده است می‌تواند

۴- نتایج پیاده سازی

در ادامه قصد داریم مدل S-GCM با قاعده یادگیری هب و اسپارس را با مدل S-GCM با قاعده یادگیری پیشنهادی مقایسه نماییم. برای این منظور، چهار مجموعه تست را که هر یک حاوی ۱۰۰ الگوی تصادفی از میان ۱۲۰ الگوی مرجع انتخاب می‌کنیم. سپس برای چهار مجموعه ذخیره، الگوی حرف A را که ۱۰٪ نویز تصادفی به آن اعمال شده است و اولین نمونه ذخیره شونده در چهار مجموعه است به عنوان ورودی به هر سه سیستم می‌دهیم.

در نمودارهای شکل (۳) سری زمانی همپوشانی الگوی هدف با خروجی شبکه نشان داده شده است. همانطور که در این شکل آمده است. خروجی مدل S-GCM با قاعده اسپارس بدون هیچ تغییری عینا مانند ورودی داده شده به آن است (منحنی ب). خروجی مدل S-GCM با قاعده هب، رفته رفته از الگوی هدف دورتر می‌گردد (منحنی الف). در حالیکه مدل S-GCM با قاعده MIMS سه بیت تصحیح خطا انجام داده است (منحنی ج). بطور کلی مدل S-GCM با قاعده اسپارس هنگامیکه نرخ ذخیره بالاست، قادر به تصحیح خطا نبوده و الگوی ورودی را عینا در خروجی تولید می‌کند و مدل S-GCM با قاعده هب نیز به هیچ الگوی هدفی همگرا نمی‌گردد چراکه نرخ ذخیره سازی از حد اشباع شبکه فراتر رفته است، در حالیکه مدل ما برای الگوهایی که در ابتدا ذخیره می‌شوند حتی هنگامیکه نرخ ذخیره بالاست، تصحیح خطا را انجام می‌دهد. برای سایر مجموعه‌های تست نیز نتایج مشابهی بدست می‌آید.



(الف)

الگوها را در نقاط حداقل انرژی خود ذخیره کند. نشان داده شده است که این روش جهت سنتز شبکه هاپفیلد در مقایسه با قانون اصلاح شده هب از گنجایش بیشتر ذخیره سازی اطلاعات و سرعت همگرایی بالاتری برخوردار است. این روش با روابط زیر خلاصه می‌شود. فرض کنید $\{\chi^1, \chi^2, \dots, \chi^m \mid \chi^k \in \{1, -1\}^N\}$ مجموعه‌ای از الگوهای یادگیری و χ_i^k مقدار i امین عنصر در k امین الگو باشد. قانون یادگیری این الگوها، در معادله (۵) آمده است:

$$t_{ij} = \frac{1}{N} \prod_{k=1}^m (\chi_i^k + \chi_j^k) \quad (5)$$

$$T = [t_{ij}], \quad W = [w_{ij}], \quad W = T^T \cdot T$$

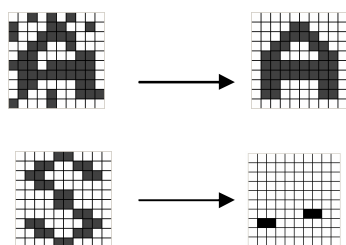
که در آن m تعداد الگوها و N تعداد نرون‌های شبکه است. ماتریس وزن‌های بدست آمده از رابطه بالا را ماتریس اسپارس می‌نامند.

ما بر اساس روش اسپارس و با رویکرد نحوه یادگیری انسان که با تکرار یک مطلب، قدرت یادگیری آن مطلب افزایش می‌یابد [4]، یک روش آموزش دیگر پیشنهاد کرده ایم. روابط بکار رفته در روش ما عینا مانند معادلات بکار رفته در روش اسپارس (یعنی معادله (۵)) است، با این تفاوت که در روش اسپارس، ابتدا ماتریس T از روی تمام الگوهای ذخیره شونده ساخته می‌شود و سپس حاصل ضرب $T^T \cdot T$ به عنوان ماتریس وزن شبکه در نظر گرفته می‌شود. ولی در روش پیشنهادی با اضافه شدن هر الگوی جدید، مقدار قبلی ماتریس T نیز در نظر گرفته شده و حاصل ضرب $T^T \cdot T$ محاسبه شده و به مقدار قبلی ماتریس وزن شبکه اضافه می‌شود و این روند تا آموزش تمام الگوها به شبکه ادامه می‌یابد. این نحوه آموزش الگوها به شبکه سبب می‌گردد که الگوهای ابتدایی بیشتر تکرار شده و لذا تاثیر بیشتری در ایجاد ماتریس وزن شبکه داشته باشند. روابط بکار رفته در روش آموزش پیشنهادی را می‌توان در معادله (۶) خلاصه نمود:

$$t_{ij} = \sum_{k=m}^1 \sum_{s=k}^1 \frac{1}{N} \prod_{k=1}^m (\chi_i^{(s)} + \chi_j^{(s)}) \quad (6)$$

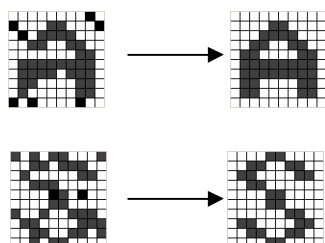
$$T = [t_{ij}], \quad W = [w_{ij}], \quad W = T^T \cdot T$$

ماتریس وزن بدست آمده از این روش را ماتریس MIMS می‌نامیم. همانطور که می‌دانیم، ساختار بیولوژیک سیستم یادگیری انسان بنحوی است که الگوهایی که در طول زمان بیشتر تکرار می‌شوند، بهتر در حافظه بلند مدت ذخیره می‌شوند. در روش آموزش ما نیز، نحوه ایجاد ماتریس وزن شبکه بگونه‌ای است که باعث می‌شود روش به ترتیب ذخیره سازی الگوها و در نتیجه به تکرار الگوهای ذخیره شونده وابسته گردد، بدین معنا که الگوهای ابتدایی در مقایسه با الگوهایی که در انتهای فرایند آموزش ذخیره می‌شوند تاثیر بیشتری در ایجاد ماتریس وزن شبکه داشته و یا به عبارت دیگر بیشتر تکرار شده و در نتیجه در حافظه بهتر ذخیره شده و حک گردند. در قسمت ضمایم، اثبات همگرایی این روش آمده است.



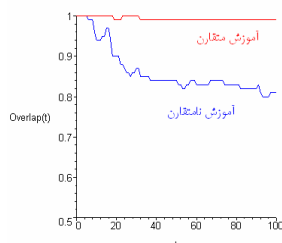
شکل (۶): فرایند بازیابی الگوها با ترتیب ذخیره سازی AS

به منظور آنکه سیستم برای الگوهای انتهایی نیز دارای نرخ تشخیص مناسبی باشد، یک اصلاح جزئی در الگوریتم آموزش اعمال می‌کنیم. برای این منظور ما یکبار الگوها را از ابتدا به انتها و در ادامه، همان الگوها را از انتها به ابتدا در سیستم ذخیره می‌سازیم. برای آنکه نتایج این تغییر در فرایند آموزش را ببینیم از همان الگوهای مثال قبل استفاده می‌کنیم و در ادامه الگوها را با ترتیب SAAS به سیستم آموزش می‌دهیم. نتایج فرایند بازیابی در شکل (۷) آورده شده است.

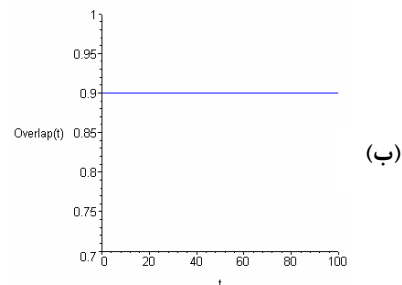


شکل (۷): فرایند بازیابی الگوها با ترتیب ذخیره سازی SAAS

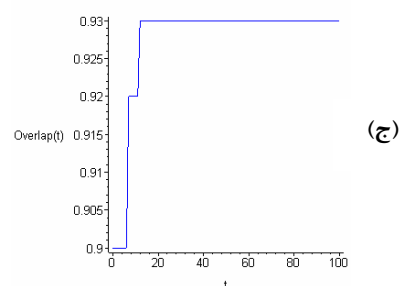
همانطور که در شکل (۷) مشاهده می‌شود، سیستم از الگوهای نویزی هر دو الگو به الگوی هدف همگرا می‌گردد. ما این مساله را با آموزش سایر مجموعه‌های دوتایی مانند AP، JA، SP و JP نیز بررسی نمودیم و به نتایج مشابهی دست یافتیم. برای بررسی نحوه تاثیر اصلاح الگوریتم در نرخ ذخیره بالاتر، این مساله را هنگامیکه پنج الگوی شکل (۱) با ترتیب AJTPS و نیز AJTPSSPTJA در سیستم ذخیره می‌گردند، بررسی می‌کنیم. چون تغییر الگوریتم سبب بالا رفتن نرخ تشخیص الگوهای انتهایی می‌گردد ما در این مثال فرایند بازیابی حرف S را در دو حالت آموزش در شکل (۸) بررسی می‌کنیم. همانطور که ملاحظه می‌کنید در الگوریتم اصلاح شده، میزان فاصله گرفتن از الگوی هدف مناسب بوده و آنرا به عنوان یک فرایند بازیابی موفق می‌توان در نظر گرفت در حالیکه در الگوریتم پایه چنین نیست.



شکل (۸): فرایند بازیابی الگوی حرف S در مجموعه پنج تایی با دو نوع روش آموزش (مقارن و نامتقارن)



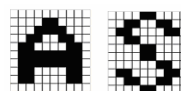
(ب)



(ج)

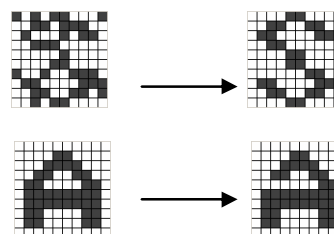
شکل (۳): سری زمانی همپوشانی الگوی هدف با خروجی شبکه، (الف) هب، (ب) اسپارس، (ج) روش پیشنهادی (MIMS)

حال قصد داریم حساسیت مدل با روش آموزش پیشنهادی را به ترتیب ذخیره سازی مورد توجه قرار دهیم. برای این منظور دو الگوی شکل (۴) را یکبار با ترتیب AS و یکبار با ترتیب SA در سیستم ذخیره می‌سازیم.



شکل (۴): دو الگو جهت آموزش به سیستم

وقتی ترتیب ذخیره سازی به گونه ای است که الگوی حرف S در ابتداست، همانطور که در شکل (۵) آورده شده است، سیستم به خوبی از الگوهای نویزدار S به الگوی هدف S همگرا می‌گردد در حالیکه در این حالت با دادن خود الگوی حرف A به عنوان ورودی، سیستم از الگوی هدف A شروع می‌شود. این سناریو برای حالتی که ترتیب ذخیره سازی با A شروع می‌شود، همانطور که در شکل (۶) آورده شده است، کاملاً برعکس است. یعنی، سیستم به خوبی از الگوهای نویزدار A به الگوی هدف A همگرا می‌گردد در حالیکه در این حالت با دادن خود الگوی حرف S به عنوان ورودی، سیستم از الگوی هدف S شروع می‌شود.



شکل (۵): فرایند بازیابی الگوها با ترتیب ذخیره سازی SA

ضمایم

برای اثبات همگرایی، تابع انرژی را به صورت معادله (۷) پیشنهاد می‌دهیم:

$$E = - \sum_{k=m}^1 \sum_{s=k}^1 \sum_i \sum_j W_{ij} \cdot (\chi_i^{(s)} + \chi_j^{(s)})^2 \quad (۷)$$

قضیه: هر تغییری در χ در خلال پردازش موجب کاهش E به اندازه $-4 \cdot \sum_{k=m}^1 \sum_{s=k}^1 \Delta \chi_r^{(s)} \sum_j W_{rj} \cdot \chi_j^{(s)}$ خواهد شد.
فرض کنید که تغییری در خروجی یک نرون χ_r صورت گیرد. بنابراین معادله (۷) به صورت زیر در می‌آید:

$$E_1 = - \sum_{k=m-s}^1 \sum_{i \neq j} \left(\sum_j W_{ij} \cdot (\chi_i^{(s)} + \chi_j^{(s)})^2 + \sum_j W_{rj} \cdot (\chi_r^{(s)} + \chi_j^{(s)})^2 + \sum_i W_{ir} \cdot (\chi_i^{(s)} + \chi_r^{(s)})^2 \right)$$

$$E_2 = - \sum_{k=m-s}^1 \sum_{i \neq j} \left(\sum_j W_{ij} \cdot (\chi_i^{(s)} + \chi_j^{(s)})^2 + \sum_j W_{rj} \cdot (\chi_r^{(s)} + \chi_j^{(s)})^2 + \sum_i W_{ir} \cdot (\chi_i^{(s)} + \chi_r^{(s)})^2 \right)$$

بنابراین تغییر تابع انرژی به صورت زیر خواهد بود:

$$E_2 - E_1 = - \sum_{k=m-s}^1 \left(\sum_j W_{rj} \cdot [(\chi_r^{(s)} + \chi_j^{(s)})^2 - (\chi_r^{(s)} + \chi_j^{(s)})^2] + \sum_i W_{ir} \cdot [(\chi_i^{(s)} + \chi_r^{(s)})^2 - (\chi_i^{(s)} + \chi_r^{(s)})^2] \right)$$

$$= - \sum_{k=m-s}^1 \left(\sum_j W_{rj} \cdot [\chi_r^{2(s)} + \chi_j^{2(s)} + 2\chi_r^{(s)} \chi_j^{(s)} - \chi_r^{2(s)} - \chi_j^{2(s)} - 2\chi_r^{(s)} \chi_j^{(s)}] + \sum_i W_{ir} \cdot [\chi_i^{2(s)} + \chi_r^{2(s)} + 2\chi_i^{(s)} \chi_r^{(s)} - \chi_i^{2(s)} - \chi_r^{2(s)} - 2\chi_i^{(s)} \chi_r^{(s)}] \right)$$

$$= - \sum_{k=m-s}^1 \left(2 \cdot \sum_j W_{rj} \cdot \chi_j^{(s)} \cdot \Delta \chi_r^{(s)} + 2 \cdot \sum_i W_{ir} \cdot \chi_i^{(s)} \cdot \Delta \chi_r^{(s)} \right)$$

به دلیل متقارن بودن ماتریس W داریم: $w_{ij} = w_{ji}$ و لذا رابطه بالا به صورت معادله (۸) در می‌آید:

$$\Delta E = -4 \cdot \sum_{k=m}^1 \sum_{s=k}^1 \Delta \chi_r^{(s)} \sum_j W_{rj} \cdot \chi_j^{(s)} \quad (۸)$$

ابتدا نشان می‌دهیم قضیه هنگامیکه در سیستم تنها یک الگو ذخیره شده است، برقرار است:

مجموع وزن دار ورودی‌های وارد شده به گره r ام می‌باشد.

حالت اول: $\chi_{r_1} = +1$ و $\chi_{r_2} = -1$: چون χ_r از $+1$ به -1 تغییر یافته است، بنابراین $\sum_j \chi_i \cdot w_{rj} < 0$ و $\Delta \chi_r = -1 - 1 = -2 < 0$ لذا

$$\Delta E < 0$$

حالت دوم: $\chi_{r_1} = -1$ و $\chi_{r_2} = +1$: چون χ_r از -1 به $+1$ تغییر یافته است، بنابراین $\sum_j \chi_i \cdot w_{rj} > 0$ و $\Delta \chi_r = +1 - (-1) = +2 > 0$ لذا

$$\Delta E < 0$$

نتایج پیاده سازی‌های ما نشان می‌دهد که در الگوریتم اصلاح شده، نرخ تشخیص الگوهای انتهایی افزایش می‌یابد. این بهبود در نرخ تشخیص بخصوص زمانیکه نرخ ذخیره پایین است بیشتر نمود می‌یابد. علت اینکه بازم نرخ تشخیص الگوهای انتهایی نسبت به الگوهای ابتدایی پایین تر است باید گفت که استفاده از این روش هم سبب نمی‌گردد که حساسیت به ترتیب ذخیره الگوها از بین برود، فقط تاحدودی ضریب الگوهای انتهایی در ایجاد ماتریس وزن بیشتر می‌گردد ولی در هر حال بدلیل ساختار الگوریتم، بازم الگوهای ابتدایی تاثیر بیشتری در ایجاد ماتریس وزن شبکه دارند.

ویژگی مهم دیگر در مدل ما اینست که زمانیکه سیستم الگوی هدف را با موفقیت بازبایی می‌کند، تفاوت بین خروجی سیستم و الگوی هدف ناچیز است. حتی زمانی که تعداد الگوهای ذخیره شده در سیستم برابر با بعد شبکه سیستم باشد. در حالیکه در سایر مدل‌ها زمانیکه سیستم در یک حالت اشباع باشد بین الگوی هدف و الگوی خروجی، تفاوت قابل رویت است.

۵- نتیجه گیری

نحوه یادگیری انسان بگونه ای است که الگوهایی که در طول زمان بیشتر تکرار شوند بهتر در حافظه بلند مدت ماندگار می‌گردند. از این رو، نحوه ایجاد ماتریس وزن شبکه در روش یادگیری ابداعی بگونه‌ای است که روش به ترتیب ذخیره سازی الگوها و در نتیجه به تکرار الگوهای ذخیره شونده وابسته است، بدین معنا که الگوهای ابتدایی بدلیل تکرار، در مقایسه با الگوهایی که در انتها ذخیره می‌شوند تاثیر بیشتری در ایجاد ماتریس وزن شبکه دارند و یا به عبارت دیگر بیشتر تکرار شده و در نتیجه در حافظه بهتر ذخیره می‌شوند.

مشکل عمده این روش، الگوهاییست که در انتهای فرایند آموزش ذخیره می‌گردند چراکه نرخ موفق تشخیص در مورد الگوهای انتهایی نسبتا پایین می‌آید. به منظور آنکه سیستم برای الگوهای انتهایی نیز دارای نرخ تشخیص مناسبی باشد، یک اصلاح جزئی در الگوریتم آموزش اعمال کردیم. برای این منظور ما یکبار الگوها را از ابتدا به انتها و در ادامه، همان الگوها را از انتها به ابتدا در سیستم ذخیره ساختیم. نتایج پیاده سازی‌های ما نشان داد که استفاده از این روش، نرخ تشخیص الگوهای انتهایی را افزایش می‌دهد. این بهبود در نرخ تشخیص بخصوص زمانیکه نرخ ذخیره پایین است بیشتر نمود می‌یابد. البته باید گفت که روش آموزش ما در مقایسه با روش آموزش هب، پایه بستر جذب کننده کوچکتری دارد.

سپاسگزاری

این پژوهش با حمایت مالی مرکز تحقیقات مخابرات ایران (ITRC) انجام شده است.

- [17] Ishii, S., Fukumizu, K., and Watanabe, S., "A network of chaotic elements for information processing", *Neural Networks*, Vol. 9, Issue 1, pp. 25-40, 1996.
- [18] Zhang, Y., Yang, L., and He, Z., "Chaotic neural network for associative memory", *J. Appl. Sci.*, Vol. 17, Issue 3, pp. 259-266, 1999.
- [19] Zhang, Y., Yang, L., and He, Z., "Study of chaotic neural network and its applications in associative memory", *Neural Processing Letters*, Vol. 9, Issue 2, pp. 163-175, 1999.
- [20] Ishii, S., and Sato, M., "Associative memory based on parametrically coupled chaotic elements", *Physica D*, Vol. 121, pp. 344-366, 1998.
- [21] Zheng, L., and Tang, X., "A new parameter control method for S-GCM", *Pattern Recognition Letters*, Vol. 26, Issue 7, pp. 939-942, 2005.

در نتیجه دیده می‌شود که به ازای هر تغییری در χ_r تابع انرژی کاهش می‌یابد. حال با فرض اینکه قضیه برای زمانیکه $m-1$ الگو در سیستم ذخیره است، برقرار بوده ثابت می‌کنیم که برای m الگو نیز صادق است:

$$-4 \cdot \sum_{k=m}^1 \Delta \chi_r^{(s)} \sum_j w_{rj} \cdot \chi_j^{(s)} = -4 \cdot \left(\underbrace{\sum_{k=m-1}^1 \sum_{s=k}^1 \Delta \chi_r^{(s)} \sum_j w_{rj} \cdot \chi_j^{(s)}}_{\text{Supposed}} + \underbrace{\sum_{s=m-1}^1 \Delta \chi_r^{(s)} \sum_j \chi_j^{(s)} \cdot w_{rj} + \Delta \chi_r^{(m)} \sum_j \chi_j^{(m)} \cdot w_{rj}}_{\text{Proved}} \right)$$

لذا طبق استقرا قضیه برای m الگو نیز برقرار است.

زیر نویس‌ها

- ¹ Global Coupled Map
- ² Symmetric Global Coupled Map
- ³ Chaotic Neuro-Computer
- ⁴ Two-way Coupled Map

مراجع

- [1] Schuster, H.G., and Just, W., *Deterministic Chaos: An Introduction*. Fourth Edition, WILEY-VCH, 2005.
- [2] Afraimovich, V., and Hsu, S.B., *Lectures on Chaotic Dynamical Systems: Studies in Advanced Mathematics*. American Mathematical Society, International Press, 2003.
- [3] Menhaj, M.B., *Fundamentals neural networks: Computational intelligence*, 2nd edition, AKU press, Vol. 1, pp. 222-229, 2002.
- [4] Calvin, W.H., and Ojemann, G.A., *Conversations with Neil's Brain: The Neural Nature of Thought & Language*, Addison-Wesley, 1994.
- [5] Sinha, S., Munakata, T., and Ditto, W.L., "Chaos Computing: Implementation of Fundamental Logical Gates by Chaotic Elements", *IEEE TRANS*, Vol. 49, No. 11, 2002.
- [6] Li, S., Mou, X., Cai, Y., Ji Z., and Zhang, J., "On the security of a chaotic encryption scheme: problems with computerized chaos in finite computing precision", *Computer Physics*, pp. 52-58, 2003.
- [7] Sinha, S., Murali, K., and Ditto, W.L., "Computing with chaos", *Physical Review. E, Statistical, Nonlinear, And Soft Matter Physics*, Vol. 68, Issue 1, Part 2, 2004.
- [8] Yao, Y., and Freeman, W.J., "Model of biological pattern recognition with spatially chaotic dynamics", *Neural Networks*, Vol. 3, Issue 2, pp. 153-170, 1992.
- [9] Skarda, C.A., and Freeman, W.J., "How brains make chaos in order to make sense of the world", *Behavior Brain Science*, Vol. 10, pp. 161-195, 1987.
- [10] Adachi, M., and Aihara, K., "Associative Dynamics in a Chaotic Neural Network", *Neural Networks*, Vol. 10, Issue 1, pp. 83-98, 1996.
- [11] Inoue, M., and Nagayoshi, A., "A chaos neuro-computer", *Phys. Lett. A*, Vol. 158, pp. 373-376, 1991.
- [12] Inoue, M., and Fukushima, S., "A neural network of chaotic oscillators", *Progress. Letters*, Vol. 87, Issue 3, 1992.
- [13] Kaneko, K., "Clustering, coding, switching, hierarchical ordering, and control in a network of chaotic elements", *Physica D*, Vol. 41, pp. 137-172, 1990.
- [14] Nara, S., Davis, P., and Totsuji, H., "Memory search using complex dynamics in a recurrent neural network model", *Neural Networks*, Vol. 6, pp. 963-973, 1993.
- [15] Chen, L., and Aihara, K., "Chaotic simulated annealing by a neural network model with transient chaos", *Neural Networks*, Vol. 8, Issue 6, pp. 915-930, 1995.
- [16] Adachi, M., Aihara, K., and Kotani, M., "An analysis of associative memory dynamics with a chaotic neural network", *Proceedings of the International Symposium on Nonlinear Theory and its Applications*, pp. 1169-1172, 1993.