

روشی جدید برای بهینه سازی ساختار شبکه عصبی با استفاده از

روش تکامل همکارانه

(نگرش طبیعی در بهینه سازی اتصالات نرونی)

علی کارساز

دانشجوی دکتری کنترل، دانشگاه فردوسی مشهد

a_karsaz1@yahoo.com

محمد رضا اکبرزاده توتونچی

استادیار دانشگاه فردوسی مشهد

akbarzadeh@kiaeee.org

چکیده: شبکه های عصبی مصنوعی به عنوان روشی

برگرفته از مکانیزم مغز انسان در زمینه های مختلف هوش

مصنوعی^۱ و محاسبات نرم^۲ کاربردهای وسیعی را دارند. از

آنجا که ساختار مغز انسان در طول سالهای متمادی، متکامل

شده است، میتوان انتظار داشت تا هر چه معماری شبکه عصبی

مصنوعی به ساختار طبیعی مغز انسان نزدیکتر باشد، از کارایی

بهرتر و بالاتری برخوردار بوده که در عین حال بسیاری از

کاربردها برای گریز از پیچیدگیهای این ساختار، ساده سازیها

و فرضیهائی صورت می پذیرد. در این مقاله، سعی شده یکبار

دیگر این نگرش به مغز انسان و چگونگی تکامل آن در طول

قرون متمادی احیاء شده و از این رهگذر روشی جدید که

کارایی خوبی در برخورد با مسائل متنوع و مختلف بویژه در

ردیابی دینامیک فرایند را داشته باشد، ارائه گردد. از سری

زمانی Macky-Glass به عنوان مثالی عمومی برای بررسی

کارایی آن استفاده شده است. نتایج بدست آمده بهبود از ۲ تا

۴ برابر بر حسب میزان خطای پیشبینی و با یک تعداد نرون

بکار رفته یکسان را نشان می دهد.

۱-مقدمه

مغز انسان بنابر نظریه داروین در طول قرون متمادی دست

خوش تحول و تکامل بوده، به طوری که ساختار و معماری

فعلی مغز انسان، شکل متکامل خود را به دست آورده است.

لذا در این مقاله سعی شده است که تا حد ممکن نگرش به

شبکه عصبی برگرفته از مکانیزم فعلی مغز انسان باشد، و نیز

تکامل و ارتقاء این ساختار و نرونها بر اساس قانون تکامل و

اصل بقا طبیعی صورت پذیرد.

انتخاب ساختار بهینه و تعداد نرونها، بازه بسیار وسیعی از

تحقیقات را در سالهای اخیر به خود اختصاص داده است،

عمده این روشها بر روی ساختار معروف پرسپترون چند لایه

صورت پذیرفته است، فرض پرسپترون چند لایه^۳ در اغلب

مقالات [3,4] محدودیتهای زیادی را در برخورد با مسائل

مختلف به وجود می آورد همانگونه که در شکل (۱) مشخص

است نرونها در مغز انسان به شکل کاملاً پیچیده و بدون نظم

خاصی با یکدیگر اتصال دارند. به گونه ای که هیچ نظم خاصی

را نمی توان برای آنها در نظر گرفت.

¹ _Artificial Intelligence

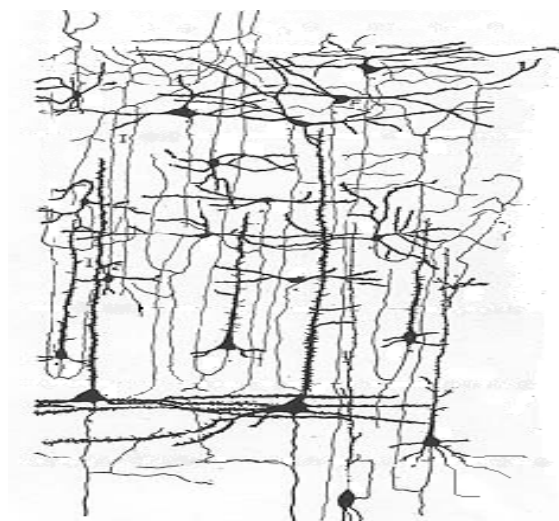
² _Soft Computing

³ -Multi Layer Perceptron

همگرایی آن به نقطه بهینه عمومی با صرف وقت کافی، قطعی می باشد، که با استفاده از عملگرهای برش، جهش و نیز جستجوی تصادفی صورت می پذیرد، یکی از اولین کارهای انجام شده در این زمینه که در آن وزنه های یک شبکه عصبی مصنوعی توسط *GA* تعیین می شود، توسط *Whiteley* صورت پذیرفته است [3].

هم تکاملی، انجام الگوریتمهای تکاملی هم زمان بین دو یا چند گونه متفاوت با استفاده از توابع برازندگی ترکیبی بین آنها می باشد که نسبت به روشهایی که از یک الگوریتم واحد استفاده می کنند به دلیل تکامل هم زمان چند زیر سیستم و ترکیب نتایج آنها با همدیگر، دارای تواناییهای بسیار زیادی می باشد. از جمله کارهای متعددی که در زمینه هم تکاملی صورت پذیرفته می توان به روشی که مبتنی بر تکامل ساختار و وزنه های یک پرسپترون می باشد اشاره نمود. در این روش یک جمعیت اولیه برای ساختارهای مختلف و یک جمعیت نیز برای وزنه های مختلف در نظر گرفته شده، سپس با تعریف تابع برازندگی ترکیبی بر اساس روش دوستی⁷ ساختار و وزنه را بهبود می بخشد [5]. ضعف عمده این روش تاکید بر پرسپترون بودن شبکه است. چرا که ساختارهای بهینه متعددی را می توان تصور نمود که به شکل یک پرسپترون چند لایه پیش رو نمی توان آنها را نمایش داد. یک روش جدید، به نام هم تکاملی یکنواخت⁸ برای آموزش وزنه های یک شبکه عصبی برای هدایت یک روبات انجام شده است [6]. این متد در دو جمعیت متفاوت، یکی برای پیدا کردن وزنه های نرونها و دیگری برای بهینه سازی محیط جستجو عمل کرده، به این ترتیب یک محیط جستجو که قابلیت عمومی سازی بالاتری را دارا است.

هم تکاملی همکارانه یک پیشرفت در روشها و متدهای معمول محاسباتی محسوب می شود [7]. این نگرش با چندین زیر جمعیت که هر کدام یک مجموعه متفاوت از پارامترهای مسئله هستند، سروکار دارد، لذا در برخورد با مسائل تکاملی همکارانه چند نکته کلیدی را باید به خاطر داشت:



شکل ۱: یک نمونه از پیچیدگی اتصالات نرونها در کورتکس^۴ مغز انسان این در حالی است که در شبکه های عصبی مصنوعی پرسپترون، نرونها به صورت لایه ای و با اتصالات منظم و مشخص با یکدیگر اتصال داشته و اطلاعات به شکل پیش رو در میان لایه های آن انتقال می یابد، در صورتی که در مغز انسان اینگونه نیست و لذا فرض پرسپترون چند لایه برای مسائل متنوع یک نگرش ساده^۵ و راحت به مسئله می باشد.

تا کنون کارهای بسیار وسیعی در زمینه شبکه های عصبی صورت پذیرفته است. به عنوان مثال روشی برای بهبود میزان خطای شبکه و تنظیم تعداد نرونها لایه مخفی و ضرائب آن پیشنهاد شده است [1]. مهمترین روش در جهت یادگیری وزنه های یک شبکه عصبی با ساختار مشخص روش پس انتشار خطا^۶ بوده که در بسیاری از مقالات بر این روش به عنوان یک روش تحلیلی و سریع در تعیین وزنه ها تکیه شده است [2]. که دارای ایرادات متعددی بوده، گرفتار شدن در کمینه های محلی و حساسیت به وزنه های اولیه و نقطه شروع بهینه سازی به مانند اغلب روشهای بهینه سازی از آن جمله است. اما مشکل اساسیتر آنست که این روشها کاملاً تحلیلی بوده، و با مکانیزم یادگیری مغز انسان تطابق لازم را ندارد.

الگوریتمهای ژنتیک (*GA*) دارای مزایای متعددی بوده، از جمله این مزایا اینکه برای شروع جستجو و ایجاد فضای اولیه جستجو، به کمترین اطلاعات اولیه از مسئله نیاز داشته، و نیز

⁴ - Cortex

⁵ - Oversimplify

⁶ - Back Propagation Error

⁷ - Friend Ship

⁸ - Uniform Cooperative Coevolutionary

۱- تفکیک مسئله^۹: این بخش به تعیین تعداد بهینه از زیر سیستمهایی که در حل مسئله نقش ایفاء می کنند باز می گردد.

۲- تخصیص اعتبار^{۱۰}: با توجه به نقش هر کدام از اعضاء زیر سیستم ها در حل نهائی، یک عدد که نشان دهنده میزان اعتبار آن عضو می باشد، به آن اختصاص می یابد.

۳- پراکندگی جمعیت^{۱۱}: یک استراتژی هم تکاملی بر اساس حفظ تنوع یا پراکندگی جمعیت با استفاده از تکنیکهای اشتراک میزان اعتبار [8]، برای دست یابی به هدف نهائی بنا شده است.

در سایر مراجع مانند [10,11,12]، اعتبار یابی زیر سیستم های مختلف با به کار گیری چند معیار صورت پذیرفته است. در مقاله ای که توسط Gonzales [14] انجام شده است یک الگوریتم ژنتیک چند منظوره یک جمعیت را کنترل می کند که هر عضو آن جمعیت یک شبکه عصبی است، در این روش از روش های متعارف نظیر خوشه سازی¹²OLS،¹³SVD استفاده شده است.

با توجه به مطالب عنوان شده، روشی جدید که هم به لحاظ تطبیق با دینامیک مسئله و هم به لحاظ ردیابی الگوهای ورودی دارای قابلیت است، معرفی می شود، این روش بر اساس تعریف و بهینه سازی دو ماتریس وزنها و اتصالات پایانه های سیناپسی با استفاده از عملگر برش سطری صورت می پذیرد. پس از آن به معرفی یک ساختار شناخته شده در بخش ۲-۶ با این روش به عنوان نمونه پرداخته خواهد شد، و در پایان در بخش نتایج شبیه سازی این مقاله روی یک مثال شناخته شده که قابلیت تغییر در دینامیک را دارد، به همراه مقایسه آن با سایر روشهای هم تکاملی عنوان می شود.

۲- تعریف مسئله

ایجاد یک ترکیب از نرونها به گونه ای که بتواند هر شبکه عصبی دیگر اعم از پیش رو و یا بازگشتی با هر تعداد نرون و پایانه سیناپسی و دندریتی دلخواه (متناسب با پیچیدگی و دینامیک سیستم مورد مطالعه) را بوجود آورد، هدف این بخش می باشد.

⁹ - Decomposition of Problem

¹⁰ - Credit Allocation

¹¹ -Population Diversity

¹² -Singular Value Decomposition

¹³ -Orthogonal Least Square

پایانه های سیناپسی نقطه اتصال یک آکسون از یک نرون به دندریت نرون مجاور می باشد، به این ترتیب دندریت ترمینال ورودی یک نرون و سیناپس پایانه خروجی آن نرون که از طریق آکسون به هسته نرون متصل است، می باشد.

۱.۲- مجموعه آموزش

مجموعه آموزش دارای $N_{example}$ عضو شامل بردار ورودی و خروجی ($[N_{in} \ N_{out}]$) می باشد.

$N_{example}$: تعداد بردارهای مجموعه آموزش

N_{in} : ابعاد بردار ورودی به شبکه عصبی

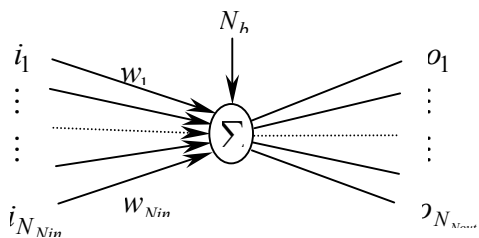
N_{out} : ابعاد بردار خروجی شبکه عصبی

به این ترتیب مجموعه آموزش دارای ابعاد زیر خواهد بود:

$$P_{N_{example} * (N_{in} + N_{out})} \quad (1)$$

۲.۲- ساختار نرونها

یک نرون با تعداد N_{Nin} دندریت ورودی و N_{Nout} تعداد سیناپسهای خروجی و یک بایاس N_b و یک تابع عملکرد مشخص می شود. (شکل (۲))



شکل ۲: نمایش نرون بکار رفته در روش اتصالات اتفاقی

۲.۳- نمایش ساختار

شامل تعداد نرونهای بکار رفته N_{num} ، وچگونگی اتصالات و وزنها آنها با یکدیگر می باشد. به این ترتیب که بین دو پایانه ورودی و خروجی مطابق (شکل ۳) که به ابعاد N_{in} , N_{out} می باشد، تعداد N_{num} نرون با مشخصاتی که در بند (۲.۲) معرفی شد بکار می رود.

N_{num} : تعداد کل نرونهای بکار رفته در شبکه

$$B = [b_1 \quad \dots \quad b_{num}]$$

$$WM : \begin{matrix} o_1 \\ \vdots \\ o_{in} \\ o_{in+1} \\ \vdots \\ o_T \end{matrix} \begin{bmatrix} i_1 & \dots & \dots & \dots & i_T \\ & I & & & \\ & & & & \\ & & & W & \end{bmatrix} \quad (4)$$

$$WM_{(N_{in}+N_{num})*(N_{num}*N_{Nin})}$$

B : بردار بایاس نرونها

WM : ماتریس وزنها

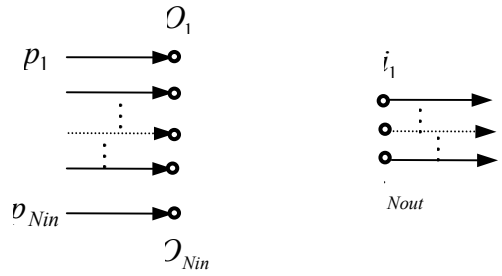
ورودیهای نرونها پس از ضرب شدن در وزنهایی که توسط ماتریس W مشخص می شود، با میزان بایاس جمع و تابع عملکرد نرون بر روی آن عمل کرده و خروجیهای مورد نظر به دست می آید.

۲.۶ نمایش یک مثال

برای روشنتر شدن موضوع یک ساختار MLP که دارای مشخصات زیر است با استفاده از تعریف مسئله آورده شده است. (شکل ۵) در بخش نتایج شبیه سازی یک ساختار پیچیده تر که با MLP قابل بیان نیست به عنوان ساختار بهینه بدست می آید که با ماتریسهای مورد نظر تعریف شده است.

جدول ۱: مقادیر پارامترها در مثال بالا

پارامتر	مقدار	پارامتر	مقدار
N_{in}	۴	I_T	۲۱
N_{out}	۱	O_T	۹
N_{num}	۵	WM	۹*۲۰
N_{Nin}	۴	CM	۲۱*۹
N_{nout}	۱		



شکل ۳: نمایش پایانه های ورودی به همراه پایانه های خروجی

۲-۴- ماتریس اتصالات

هر نرون مطابق شکل (۲) با اتصالات ورودی i و اتصالات خروجی o شناخته می شود، به این ترتیب هر ورودی (دندریت) i به هر خروجی (سیناپسی) o می تواند متصل شود، بنابراین تعداد دندریتهای ورودی با توجه به تعداد N_{out} دندریت مربوط به ترمینال خروجی (شکل ۳) از فرمول زیر محاسبه می گردد، و نیز تعداد سیناپس های خروجی با توجه به تعداد سیناپس ترمینال ورودی مسئله (شکل ۳) قابل محاسبه است:

$$I_T = N_{out} + N_{num} * N_{Nin} \quad (2)$$

$$O_T = N_{in} + N_{num} * N_{Nout}$$

I_T : تعداد کل دندریتهای قابل اتصال به پایانه های سیناپسی

O_T : کل سیناپسهایی که دندریتها به آنها می توانند متصل شوند.

به این ترتیب ماتریس اتصالات را به ابعاد $I_T * O_T$ تعریف می نمائیم:

$$CM : \begin{matrix} i_1 \\ \vdots \\ i_{Nout} \\ i_{Nout+1} \\ \vdots \\ i_T \end{matrix} \begin{bmatrix} o_1 & o_{Nin} & o_{Nin+1} & \dots & o_T \\ 1 & \dots & 0 & \dots & 0 & \dots \\ & \ddots & & & & \\ 0 & \dots & 1 & & & \\ & & & \ddots & & \\ & & & & \ddots & \\ & & & & & \ddots \end{bmatrix}_{I_T * O_T} \quad (3)$$

CM : ماتریس اتصالات

۲-۵- ماتریس وزنها

علاوه بر چگونگی اتصال دندریتها و سیناپها، وزن این اتصالات نیز حائز اهمیت است. در این مقاله دو ماتریس اتصالات (برای تعریف زیر فضای ساختار) و ماتریس وزنها برای تعریف زیر فضای وزنهایی شبکه عصبی تعریف می شود و نیز بین ورودیهای مجموعه آموزش و پایانه های $O_1 \dots O_{Nin}$ (شکل ۳) وزنهایی واحد را در نظر می گیریم.

برش و جهش معمولی، بهبود میابد، دومین فضای بهینه سازی ماتریس اتصالات است که با عملگر برش سطری و جهش معمولی ارتقا می یابند،

۳-۱- عملگر برش سطری:

همانگونه که ذکر شد یکی از محدودیتهای اعمال شده به مسئله این است که اجازه برقراری بیش از یک اتصال به هر دندریت در هر نرون دلخواه داده نشود (برقراری بیش از یک اتصال در این روش ممکن بوده ولیکن موجب پیچیده تر شدن مسئله میشود). لذا در هر سطر از ماتریس CM بیش از یک اتصال وجود ندارد، برای اینکه این محدودیت برقرار بماند در این بخش عملگر برش سطری انتخاب و اعمال می شود. به این ترتیب پس از انتخاب دو عضو جمعیت با احتمال P_c جهت برش یک سطر از CM اول بصورت احتمالی انتخاب شده و کلیه سطرهای پس از آن با ماتریس انتخاب شده دوم جایگزین می شود.

۳-۲- عملگر جهش سطری

برای گریز از محدودیت بالا در عملگر جهش نیز، پس از انتخاب سطر مورد نظر در ماتریس اتصالات جهت جهش با احتمال P_m کل سطر مورد نظر با سطری جدید که تنها یک ۱ در آن که به شکل رندم انتخاب می شود جایگزین می شود.

۴- نتایج شبیه سازی

سری $Mackey-Glass$ به عنوان یک مثال عمومی^{۱۴} که قابلیت تغییر در دینامیک را داراست، برای مقایسه نتایج شبیه سازی این روش با روشهای رایج پیشین آورده شده است:

$$\frac{dx(t)}{dt} = \frac{ax(t-\tau)}{1+x^{10}(t-\tau)} - bx(t) \quad (۷)$$

در این شبیه سازی $a=0.2, b=0.1, \tau=20$ در نظر گرفته شده است. مقادیر بالاتر τ درجه آشوب^{۱۵} سیستم را تغییر داده و دینامیک آن تغییر می یابد. در شکل (۶) نتایج شبیه سازی با روش پیشنهادی در پیش بینی سری $Mackey-Glass$ آورده شده است.

	o_1	o_2	o_3	o_4	o_5	o_6	o_7	o_8	o_9
i_1	1	0	0	0	0	0	0	0	0
i_2	0	1	0	0	0	0	0	0	0
i_3	0	0	1	0	0	0	0	0	0
i_4	0	0	0	1	0	0	0	0	0
i_5	1	0	0	0	0	0	0	0	0
i_6	0	1	0	0	0	0	0	0	0
i_7	0	0	1	0	0	0	0	0	0
i_8	0	0	0	1	0	0	0	0	0
i_9	1	0	0	0	0	0	0	0	0
i_{10}	0	1	0	0	0	0	0	0	0
$CM = i_{11}$	0	0	1	0	0	0	0	0	0
i_{12}	0	0	0	1	0	0	0	0	0
i_{13}	1	0	0	0	0	0	0	0	0
i_{14}	0	1	0	0	0	0	0	0	0
i_{15}	0	0	1	0	0	0	0	0	0
i_{16}	0	0	0	1	0	0	0	0	0
i_{17}	1	0	0	0	0	0	0	0	0
i_{18}	0	1	0	0	0	0	0	0	0
i_{19}	0	0	1	0	0	0	0	0	0
i_{20}	0	0	0	1	0	0	0	0	0
i_{21}	0	0	0	0	0	0	0	0	1

$$V_o = [o_1 \ o_2 \ o_3 \ o_4 \ o_5 \ o_6 \ o_7 \ o_8 \ o_9]^T$$

$$V_i = [i_1 \ i_2 \ \dots \ i_{20}]^T$$

$$B = [b_1 \ b_2 \ b_3 \ b_4 \ b_5]^T$$

$$P = [p_1 \ p_2 \ p_3 \ p_4]^T$$

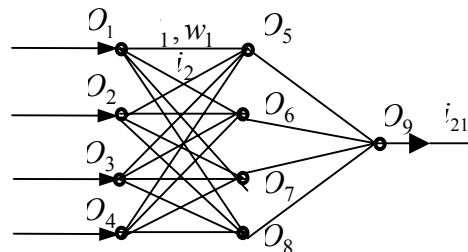
$$V_o(k+1) = \begin{bmatrix} P(k) \\ WM * V_i(k) \end{bmatrix} \quad (۵)$$

$$\begin{bmatrix} V_i(k+1) \\ output(k) \end{bmatrix} = CM * V_o(k) \quad (۶)$$

$P(k)$: بردار مجموعه آموزش در لحظه k ام $k: 1 \dots N_{example}$

$V_o(k)$: مقدار بردار خروجی در لحظه k

$V_i(k)$: مقدار بردار ورودی در لحظه k



شکل ۵: مثالی از یک پرسپترون سه لایه

$$I_T = 1 + 5 * 4 = 21$$

$$O_T = 4 + 5 * 1 = 9$$

به این ترتیب با مشخص بودن بردار $[P \ V_i]^T$ که در آن P داده های ورودی مجموعه آموزش بوده، مقادیر اولیه $V_i = 0$ می باشد. بنابراین بردار V_o مشخص می شود.

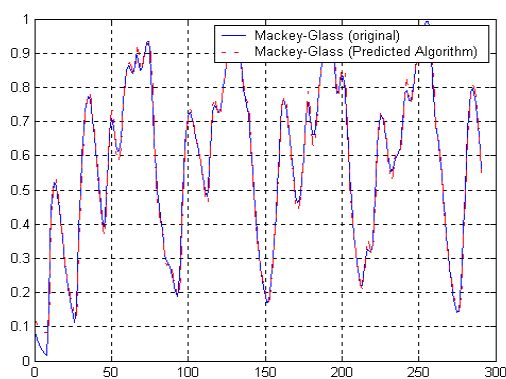
۳- الگوریتم بهینه سازی

دو فضای موثر برای بهینه سازی هم زمان عبارتند از: ماتریس وزنهای که بوسیله یک الگوریتم بهینه سازی ژنتیک با عملگرهای

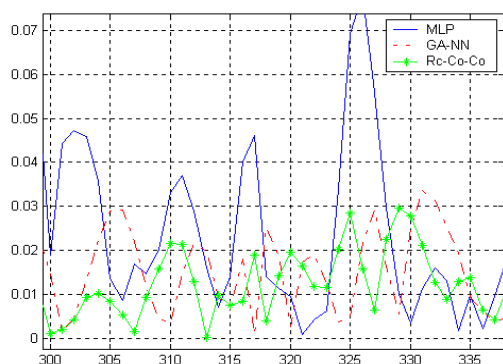
^{۱۴} -Benchmark

^{۱۵} - Chaos

	Specification	mse
MLP	$Max_epoch = 50000$ $Lr = 0.01$ $Neuron - Layer1 = 4$ $Neuron - Layer2 = 1$	1.3602
GA-NN	$ngeneration = 2000$ $npop = 10$ $P_C = 0.7$ $P_m = 0.005$ $Neuron - Layer1 = 4$ $Neuron - Layer2 = 1$	0.934
Rc-Co-Co	$ngeneration - space1 = 50$ $ngeneration - space2 = 1200$ $npop = 10$ $P_m = 0.005$ $P_C = 0.7$ $N_{num} = 5$	0.403
Co-Co-NN	$max_epoch = 11200$ $npop = 10$ $Neuron - Layer1 = 20$ $Neuron - Layer2 = 26$	0.405



شکل ۶: نتایج شبیه سازی روش اتصالات اتفاقی



شکل ۷: نمایش قدر مطلق میزان خطاها در قسمتی از مجموعه تست در روشهای مختلف

چنانچه در شکل (۷) مشخص است. برای فرایند یادگیری شبکه عصبی از تعداد ۱۰۰۰ نمونه اول سری به عنوان مجموعه آموزش^{۱۶} استفاده شده، سپس از ۵۰۰ نمونه بعدی به عنوان مجموعه تست^{۱۷} استفاده شده است. نتایج شبیه سازی با نتایج سایر کارها از جمله روش شبکه عصبی معمولی چند لایه با آموزش پس انتشار خطا (MLP)، روش شبکه عصبی-الگوریتم ژنتیک (GA-NN)، [3] آمده و نتایج با روش پیشنهادی (Rc-Co-Co) در این مقاله مقایسه شده اند. مقادیر پارامترهای در نظر گرفته شده مطابق جدول (۲) است. متد هم تکاملی همکارانه (Co-Co-NN) [5] که با تعداد بیشتری نرون انجام شده و مقادیر بهینه تعداد تکرار آموزش و تعداد نرون در هر لایه مشخص شده است، برای مقایسه با روش پیشنهادی آورده شده است. در سه مورد نخست از یک شبکه عصبی با مشخصات مثال بخش (۲-۶) (با تعداد یکسان نرون) استفاده شده است، تا بتوان نتایج شبیه سازی را با یکدیگر مقایسه نمود. و در روش (Co-Co-NN) بهینه سازی با صرف تعداد بیشتری نرون صورت پذیرفته است. کارایی روش پیشنهاد شده با بکار گیری ۵ نرون تقریباً برابر کارایی روش (Co-Co-NN) با تعداد ۵۱ نرون است. به این ترتیب روش پیشنهادی با بهینه سازی اتصالات بین نرونی در استفاده بهینه از نرونها بسیار موثر است. با استفاده از دستور *ttest2*، *Matlab* داده های شبیه سازیها مورد بررسی قرار گرفت. نتایج بدست آمده پائین تر بودن *mse* های روش پیشنهادی را با یک بازه اطمینان ۹۵٪ نشان می دهد.

با افزایش میزان آشوب سری *Mackey-Glass* می توان دینامیک سیستم را آشفته تر کرده، نتایج را بررسی نمود که بهبود چشمگیر روش مورد نظر در دنبال سازی دینامیک سیستم را نشان می دهد.

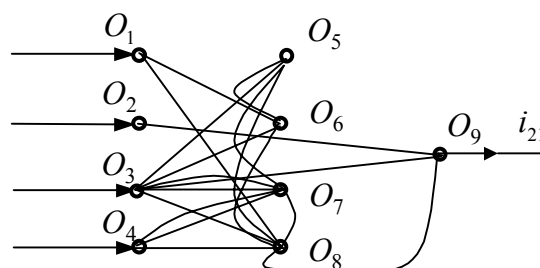
جدول ۲: نتایج شبیه سازی با روشهای مختلف

¹⁶ -Training Set

¹⁷ _Validation Set

- [1] K. Peng, S. Ge & C. Wen, **An Algorithm to Determine Neural Network Hidden Layer Size and Weight Coefficients**, Proceedings of the 15th IEEE Int. Symposium on Intelligent Control, July, pp. 48-54, 2000.
- [2] Simon Haykin, **Neural Network: A Comprehensive Foundation**, prentice Hall, New Jersey, 1999.
- [3] D. Whiteley, **Applying Genetic Algorithm to Neural Networks Learning**, Proceeding of 7th Conference of the society of Artificial Intyelligence and Simulation of Behavior, Sussex, England, Pitman Publishing, pp. 320-331, 1986.
- [4] X. Yao. **A review of Evolutionary Artificial Neural Networks**, International journal of Intelligent Systems, Vol. 8, pp. 549-567, 1989.
- [5] M. R. Akbarzadeh. M. Trabi., **Genetic Searching in Finding Preceptron Optimum Weights and Stractur, Co.Co.GA Algorithm**, 11 th ICEE May 2003, Vol. 1.
- [6] A. Berlanga A. Sanchis, **A General Learning Co-Evolution Method to Generalize Autonomous Robot Navigation Behavior**, SCA-LAB, Universidad, 30, Madrid, SPAIN, 2000 IEEE.
- [7] Potter, M.; De Jong, K. (2000), **Cooperation Coevolution: an Architecture for Evolving Coadapted Subcomponents**, Evolutionary Computation, 8(1), 1-29.
- [8] Golberg, D.:Richardson J. (1987), **Genetic Algorithm with Sharing for Multimodel Function Optimization**, In Grefenstette (ed.), Proceeding of Second International Conference on Genetic Algorithm, pp. 41-49. Lawrece Erlbaum Associates.
- [9] Beasley, D.; Martin R.. (1993). **A Sequential Niche Technique for Multimodel Function Operation**, Evolutionary Computation, 1. pp. 101-125.
- [10] Garcia, N., Hervás, C., J. (2002). **Multi-objective Cooperative Coevolution of Artificial Neural Networks**, Neural networks, 15, pp. 1259-1278.
- [11] Giordana, A. Neri, F. (1996) **Search-intensive Concept Induction**, Evolutionary Computation, 3(4), 375-416.
- [12] Rosin, C.; belew, R. (1997), **New Methods for Competitive Coevolution**, Evolutionary Computation, 5(1), 1-29.
- [13] Rivas, V.; Castillo, P. A.; Merelo-Guervos JJ. (2002), **Evolved RBF Networks for Time-Series Forecasting and Function Approximation**, J. Merelo Guervos et al. editor. Lecture Notes in Computer Science, LNCS. Springer-Verlag, pp. 505 September.
- [14] Gonzalez, J.; Ortega, J.; Pomares, H.; Fernandez, F.; Diaz, A. (2003) **Multiobjective Evolutionary Opyimization of Size, Shape and Position Parameters of Radial Basis Function Networks for Function Approximation**, IEEE Transaction on Neural Network, Vol.14, n.6, pp.1478-1495

نتایج ماتریسهای بهینه شده پس از تنها ۵۰ تکرار در روش پیشنهادی در ادامه آمده است. ماتریس بهینه شده اتصالات، ارتباطات بین نرونی را با ساختار شکل (۸) نشان می دهد.



شکل ۸: شکل بهینه شده اتصالات بین نرونی توسط الگوریتم پیشنهادی

	o_1	o_2	o_3	o_4	o_5	o_6	o_7	o_8	o_9
i_1	0	0	0	0	0	1	0	0	0
i_2	0	0	1	0	0	0	0	0	0
i_3	0	0	0	0	0	0	1	0	0
i_4	0	0	0	0	0	0	0	1	0
i_5	0	0	1	0	0	0	0	0	0
i_6	0	0	0	0	0	0	0	1	0
i_7	1	0	0	0	0	0	0	0	0
i_8	0	0	0	0	0	0	0	0	1
i_9	0	0	1	0	0	0	0	0	0
i_{10}	0	0	0	1	0	0	0	0	0
i_{11}	0	0	0	1	0	0	0	0	0
i_{12}	0	0	1	0	0	0	0	0	0
i_{13}	0	0	0	1	0	0	0	0	0
i_{14}	0	0	0	0	0	0	1	0	0
i_{15}	0	0	1	0	0	0	0	0	0
i_{16}	1	0	0	0	0	0	0	0	0
i_{17}	0	0	0	1	0	0	0	0	0
i_{18}	0	1	0	0	0	0	0	0	0
i_{19}	0	0	0	0	0	0	0	0	1
i_{20}	0	0	0	0	0	0	0	1	0
i_{21}	0	0	0	0	0	1	0	0	1

 $B = [0.55 \quad 0.58 \quad 1.96 \quad -1.6 \quad 1.85]$

	o_1	o_2	o_3	o_4	o_5
i_1	-0.19	0	0	0	0
i_2	.48	0	0	0	0
i_3	.15	0	0	0	0
i_4	-.59	0	0	0	0
i_5	0	1.84	0	0	0
i_6	0	1.69	0	0	0
i_7	0	-.35	0	0	0
i_8	0	1.72	0	0	0
i_9	0	0	-.75	0	0
i_{10}	0	0	1.42	0	0
i_{11}	0	0	1.82	0	0
i_{12}	0	0	-1.87	0	0
i_{13}	0	0	0	1.35	0
i_{14}	0	0	0	-.61	0
i_{15}	0	0	0	0.45	0
i_{16}	0	0	0	0.09	0
i_{17}	0	0	0	0	1.73
i_{18}	0	0	0	0	-0.46
i_{19}	0	0	0	0	-0.78
i_{20}	0	0	0	0	1.82

 $T =$

۵- نتیجه گیری

با نگاهی به ساختار پیچیده و به دور از هر چارچوب منظم در مغز انسان، به عنوان یک ساختار بهینه می توان روشی را برای باز نگری ایجاد شبکه های عصبی مصنوعی ابداع نمود. این روش که مبتنی بر تکامل همکارانه بوده، ارتباطات بین نرونی را ارتقا بخشیده و به ساختاری بهینه دست می یابد. بر این اساس با تعداد نرون بسیار پائین می توان در تخمین توابع به خوبی عمل نمود. نتایج شبه سازی بهبود در تخمین توابع استاندارد مانند Macky-Glass را از ۲ تا ۴ برابر (با تعداد نرون یکسان) نشان می دهند.