

دسترسی در سایت <http://jnrm.srbiau.ac.ir>

سال چهارم، شماره شانزدهم، زمستان ۱۳۹۷

شماره شاپا: ۲۵۸۸-۵۸۸۸

JNRM
JOURNAL OF
NON-LOCAL
RHEOLOGY

پژوهش‌های نوین در ریاضی



دانشگاه آزاد اسلامی، واحد علوم و تحقیقات

آرایه روش مبتنی بر گرادیان مزدوج برای آموزش شبکه عصبی تصمیم

محدثه نادرشاهی^۱، اعظم دخت صفی صمغ آبادی^{۲*}، رضا توکلی مقدم^۳

^(۱) دانشجوی دکتری، گروه مهندسی صنایع، دانشگاه پیام نور، تهران، ایران

^(۲) استادیار، گروه مهندسی صنایع، دانشگاه پیام نور، ص.پ. ۱۹۳۹۵-۳۶۹۷، تهران، ایران

^(۳) استاد، دانشکده مهندسی صنایع، پردیس دانشکده‌های فنی، دانشگاه تهران، تهران، ایران

تاریخ ارسال مقاله: ۹۷/۰۱/۰۸ تاریخ پذیرش مقاله: ۹۷/۰۶/۲۷

چکیده

شبکه عصبی تصمیم یک رویکرد جدید برای حل مسائل تصمیم‌گیری چندهدفه مبتنی بر شبکه‌های عصبی مصنوعی می‌باشد. با بکارگیری روش‌های ارزیابی غیردقیق، ظرفیت یادگیری در این شبکه افزایش یافته و حجم مجموعه داده آموزشی کاهش یافته است. لذا شرایط برای تصمیم‌گیرندگان تسهیل شده است. روش موجود برای آموزش پارامترهای این شبکه مبتنی بر روش گرادیان است. یکی از محدودیت‌های روش فعلی، سرعت همگرایی آن است. در این مقاله، جهت افزایش کارایی شبکه عصبی تصمیم، روشی مبتنی بر گرادیان مزدوج برای آموزش این شبکه توسعه داده شده است. نکته ویژه در آموزش شبکه عصبی تصمیم این است که ساختار و پارامترهای دو زیر شبکه پرسپترون چند لایه موجود در شبکه عصبی تصمیم در فرآیند آموزش یکسان باقی بماند. کارایی روش پیشنهادی در حل مسئله تصمیم‌گیری چند هدفه با توابع مطلوبیت متعدد خطی و غیرخطی ارزیابی می‌شود. نتایج روش پیشنهادی با برخی روش‌های مشابه مقایسه شده و نشان می‌دهد که در روش پیشنهادی، همگرایی نسبت به روش قبلی سریع‌تر بوده و نتایج مطلوب‌تر هستند.

واژه‌های کلیدی: آموزش شبکه عصبی تصمیم، روش گرادیان مزدوج، مسائل تصمیم‌گیری چندهدفه، تخمین تابع مطلوبیت.

۱- مقدمه

شبکه عصبی تصمیم^۱ DNN [۱] که نوع خاصی از شبکه‌های عصبی است، توسط چن و همکاران برای تخمین تابع مطلوبیت در مسایل تصمیم‌گیری چند هدفه پیشنهاد داده شد. در تحلیل مسائل تصمیم‌گیری چندهدفه به دلیل وجود توابع چندگانه متناقض، اغلب مطلوب است که به طور صریح ساختار کامل ارجحیت‌های DM^2 از طریق یک مدل تصمیم تجویزی مانند تابع مطلوبیت^۳ بدست آید [۲، ۱۵۲]. از طرفی در بکارگیری روش‌هایی که در آن‌ها ارزیابی ساختار ارجحیت‌های DM از طریق تابع مطلوبیت صورت می‌گیرد، تشخیص تابع مطلوبیت به سادگی میسر نمی‌باشد. برخی ساده‌سازی‌ها برای تابع مطلوبیت در مسائل چندهدفه توسط محققین در نظر گرفته شده است [۳-۸]. همچنین استفاده از فرم‌های متداول تابع مطلوبیت نیاز به مفروضات قوی مانند شرط استقلال میان معیارها دارد که در مسایل محدودی از مسایل دنیای واقعی این شرط تامین می‌شود.

در این میان روش‌های تعاملی به عنوان موثرترین روش‌ها برای ارزیابی یک مسئله $MODM^4$ شناخته شده‌اند. ماهیت متناقض اهداف، جستجو برای یافتن ارجح‌ترین جواب را نیازمند تعامل با تصمیم‌گیرنده و گرفتن بازخورد از وی به منظور ترسیم ساختار اطلاعات، ساخته است. روش‌های متعدد معرفی شده [۹-۱۱] هنوز به طور کامل رضایت بخش نبوده‌اند. هر یک از روش‌های تعاملی سنتی دارای یک سری مفروضات می‌باشند که این مفروضات، محدودیت‌هایی را در بکارگیری آن‌ها در عمل ایجاد می‌کنند. هم‌چنین ساختار ارجحیت‌های DM در عمل ممکن است بسیار پیچیده باشد، بنابراین به روش‌ها و ابزارهای توانمندی که قادر به اخذ این ساختارها و هدایت فرآیند حل در یافتن ارجح‌ترین جواب باشند، نیاز می‌باشد [۱۲، ۶۰۲].

یک رویکرد جدید در حل مسائل تصمیم‌گیری چندهدفه، بکارگیری شبکه‌های عصبی در تخمین و توصیف ساختار ارجحیت‌های تصمیم‌گیرنده می‌باشد. این رویکرد بر استخراج، نمایش و بکارگیری اطلاعات ارجحیت‌های بدست آمده از تصمیم‌گیرنده تمرکز دارد. در مقایسه با رویکردهای قبلی، تعمیم اطلاعات ارجحیت‌های تصمیم‌گیرنده و جستجوی ابتکاری برای راه‌حل‌های بهبود یافته از ویژگی‌های این رویکرد می‌باشد. در این رویکرد انواع متعدد ساختارهای ارجحیت غیرخطی نیز قابل تخمین هستند [۱۳، ۸۳۵]. بکارگیری شبکه‌های عصبی در حل مسائل تصمیم‌گیری چند هدفه نسبت به روش‌های مبتنی بر تخمین تابع مطلوبیت و روش‌های تعاملی متداول، چندین برتری دارد: اول اینکه در این رویکرد، لازم نیست فرض شود که تابع مطلوبیت ساختار خاصی دارد. دوم اینکه در جایی که روش‌های تعاملی تابع مطلوبیت را جزئی و موضعی ارزیابی می‌کنند، این رویکرد یک تابع ارزیابی شده کامل و کلی را تخمین می‌زند و سوم اینکه در این رویکرد، زمانی که اطلاعات کسب شده از تصمیم‌گیرنده تکمیل‌تر شود، شبکه عصبی قادر به تطبیق و بهبود نمایش ساختار ارجحیت‌ها می‌باشد [۱۴، ۱۵۴۳].

در واقع ایده اصلی استفاده از شبکه‌های عصبی برای حل مسائل تصمیم‌گیری چند معیاره از آنجا ناشی می‌شود که ارجحیت‌های تصمیم‌گیرنده دارای توابعی ضمنی است، تعیین این توابع کار دشواری است. از طرفی یک شبکه عصبی پیشرو سه لایه (لایه‌های ورودی، پنهان و خروجی) با تعداد عصب کافی در لایه پنهان می‌تواند هر تابع پیوسته را تخمین بزند [۱۵، ۵۹۷]. بنابراین اساس رویکردهای مبتنی بر شبکه‌های عصبی برای حل مسائل تصمیم‌گیری چندهدفه، مدل‌سازی ساختار ارجحیت‌های تصمیم‌گیرنده با استفاده از یک شبکه عصبی پیشرو می‌باشد. در این شبکه تعداد ورودی‌های شبکه برابر با تعداد هدف‌ها می‌باشد و شبکه تنها یک خروجی دارد. پس از آموزش این شبکه عصبی توسط داده‌های آموزشی، شبکه قادر خواهد بود که تابع مطلوبیت مسئله را تخمین بزند.

1. Decision Neural Network
2. Decision Maker
3. Utility Function
4. Multi Objective Decision Making

محور سوم- چگونگی بکارگیری ساختار ارجحیت‌های تصمیم‌گیرنده برای جستجو به دنبال جواب بهبود یافته. شبکه عصبی تصمیم با ساختار ویژه‌اش، محدودیت‌های روش‌های مبتنی بر شبکه عصبی قبلی را مرتفع ساخته است. این شبکه از تکنیک‌های ارزیابی غیرمستقیم ارجحیت‌ها در آموزش شبکه عصبی بهره گرفته است، بنابراین ظرفیت یادگیری شبکه افزایش یافته است.

* این الگوریتم بر مبنای Golden Section Search و روش Polak-Ribiere می‌باشد.

** در این الگوریتم ابتدا یک جهت تولید می‌شود و سپس یک جستجوی چندبعدی در آن جهت انجام می‌شود. علاوه بر این حجم مجموعه داده آموزشی با بکارگیری روش‌های ارزیابی غیر دقیق کاهش می‌یابد، لذا شرایط برای تصمیم‌گیرندگان تسهیل شده است. با وجود مزایای DNN هنوز کارهای زیادی برای توسعه این شبکه به خصوص برای بهبود روش آموزش آن می‌توان انجام داد [۲۱، ص ۱۴۴].

استفاده از شبکه‌های عصبی برای حل مسائل تصمیم‌گیری چند هدفه قدمت طولانی ندارد. با این حال در چند دهه گذشته کارهایی در این زمینه شکل گرفته است [۲۳-۱۶] و [۱۴-۱۲] و [۲-۱]. در جدول ۱ مرور مختصری بر عمده تحقیقات انجام شده پیشین آرایه شده است.

در اغلب این روش‌ها از یک شبکه عصبی پیشرو استفاده شده است. تحقیقات این حوزه، در سه محور عمده صورت می‌پذیرند. موضوع این مقاله مربوط به محور دوم است:

محور اول- چگونگی اخذ اطلاعات مربوط به ارجحیت‌های تصمیم‌گیرنده بر روی مجموعه جواب‌های شدنی کارا (موثر)

محور دوم- چگونگی نمایش ساختار ارجحیت‌های تصمیم‌گیرنده به یک روش سیستماتیک: در این رویکرد اطلاعات تصمیم‌گیرنده برای آموزش یک شبکه عصبی بکار گرفته می‌شود تا اطلاعات مربوط به ارجحیت‌ها در آن ذخیره گردد.

جدول ۱- مروری بر عمده پیشینه تحقیق و دسته‌بندی آن‌ها

مرجع	نوع مساله	نوع شبکه	روش آموزش	تعداد داده‌های آموزشی	کیفیت آموزش	سرعت همگرایی	نوع داده‌های آموزشی
[۲]	گسسته	FFANN	الگوریتم یادگیری تطبیقی بهبودیافته*	p	خوب	بالا	صریح و دقیق
[۱۲]	پیوسته	FFANN	BP (با جواب‌های اولیه کارا)	p	خوب	پایین	صریح و دقیق
[۱۳]	پیوسته	FFANN	BP	p	ضعیف	پایین	صریح و دقیق
[۱۴]	گسسته	AF-ANN _۱	الگوریتم یادگیری تکراری**	p	خوب	بالا	صریح و دقیق
[۱]	پیوسته	DNN	BP (با روش بهینه‌سازی گرادیان نزولی)	p(p-1)/2	بسیار خوب	پایین	صریح و دقیق / غیر صریح و غیر دقیق
[۲۲]	پیوسته	DNN	BP با اصلاح نرخ یادگیری	p(p-1)/2	بسیار خوب	پایین	صریح و دقیق / غیر صریح و غیر دقیق
پژوهش حاضر	پیوسته	DNN	روش گرادیان مزدوج	p(p-1)/2	بسیار خوب	بالا	صریح و دقیق / غیر صریح و غیر دقیق

1. Adaptive Feed forward Artificial Neural Network

تصمیم و $f_i(x)$ ، $i=1, \dots, n$ -امین تابع هدف می‌باشد. با در نظر گرفتن مقدار تابع هدف $f_i(x)$ به صورت z_i ، مسئله تصمیم‌گیری چندهدفه به صورت زیر بازنویسی می‌شود:

$$\max z = [z_1, z_2, \dots, z_n]^T$$

$$s.t. \quad x \in X \quad (2)$$

$$z_i = f_i(x) \quad i=1, 2, \dots, n$$

تعریف ۱- بردار X یک جواب موجه^۳ در فضای تصمیم است اگر $x \in X$

تعریف ۲- یک بردار معیارها مانند $z \in Z$ کارا(موثر)^۴ نامیده می‌شود، اگر و تنها اگر جوابی مانند $\bar{z} \in Z$ وجود نداشته باشد که برای تمام i ها، $z_i \leq \bar{z}_i$ و حداقل برای یک i داشته باشیم $\bar{z}_i < z_i$.

تعریف ۳- مجموعه تمام راه حل‌های کارا را با N نمایش می‌دهیم و آن را مرز کارا^۵ می‌نامیم.

تعریف ۴- اگر $z \notin N$ در این صورت Z جواب ناکارا می‌باشد.

در مدل‌های تصمیم‌گیری چندهدفه، تابع کلی مطلوبیت تصمیم‌گیرنده، تابعی بر حسب مقادیر توابع هدف می‌باشد که بر حسب میزان مطلوب بودن راه حل مقداری بین صفر و یک به آن اختصاص می‌یابد. تابع مطلوبیت در برخی از روش‌های ارزیابی به صورت عینی محاسبه و با توجه به محدودیت‌های مدل اصلی بهینه می‌گردد و در برخی دیگر به صورت ضمنی مورد بررسی و بهینه شدن قرار می‌گیرد. زیرا تشخیص تابع مطلوبیت به سادگی میسر نیست.

۲-۲- ساختار شبکه عصبی MLP

ساختار شبکه عصبی MLP در شکل (۱) نشان داده شده است. در این شبکه، $U = [u_1, \dots, u_n]^T$ بردار ورودی، n_h تعداد نرون‌ها در لایه میانی (مخفی)، W_h ماتریس وزن‌های لایه میانی، f_h تابع فعال‌ساز لایه میانی، $O_h = [o_1^h, \dots, o_{n_h}^h]^T$ بردار خروجی لایه میانی،

در روش پس انتشار خطا^۱ برای تطبیق مقادیر وزن، به منظور کمینه‌سازی خطای برآورد شبکه از بردار گرادیان استفاده می‌شود. سرعت همگرایی در این روش کند است. در اغلب رویکردهای مبتنی بر شبکه عصبی پیشرو برای حل مسائل تصمیم‌گیری چند هدفه، از روش‌های مبتنی بر گرادیان برای آموزش شبکه عصبی استفاده شده است. آقای چن برای آموزش DNN روش پس انتشار خطا (BP) را که مبتنی بر بردار گرادیان است با اصلاحات متناسب با ساختار ویژه این شبکه، معرفی نموده است [۱، ص ۲۲۱-۲۲۲]. در مقاله حاضر به منظور کاهش تعداد تکرار مراحل آموزش و افزایش نرخ همگرایی الگوریتم آموزشی DNN، روش کارآمد برنامه ریزی غیرخطی نامقید که مبتنی بر مشتق دوم می‌باشد، جهت طراحی یک الگوریتم یادگیری کارا برای این شبکه ارایه و به کار گرفته شده است.

ادامه این مقاله به این شرح می‌باشد: در بخش ۲، به بیان مفاهیم کلی، معرفی شبکه‌های MLP^2 ، DNN، نحوه محاسبات پیشرو در آنها و مقایسه توانمندی‌های DNN نسبت به شبکه‌های MLP در حل مسایل چند هدفه پرداخته می‌شود. در بخش ۳، روش پیشنهادی تشریح شده است. نحوه آموزش شبکه‌های عصبی تصمیم به روش پس انتشار خطا (BP) و روش پیشنهادی گرادیان مزدوج در این بخش ارائه می‌شود. در بخش ۴ روش پیشنهادی در چند مثال کاربردی پیاده‌سازی شده و نتایج آن ارایه شده است. بخش ۵ حاوی نتیجه‌گیری و پیشنهادات خواهد بود.

۲- مفاهیم کلی و تعریف مسئله

۲-۱- مدل‌های تصمیم‌گیری چندهدفه

یک مسئله تصمیم‌گیری چندهدفه در حالت کلی به صورت زیر نوشته می‌شود:

$$\max f(x) = [f_1(x), f_2(x), \dots, f_n(x)]^T$$

$$s.t. \quad x \in X \quad (1)$$

که در آن، n تعداد توابع هدف، X فضای متغیرهای

3. Feasible Solution
4. Non Dominated Solution
5. Efficient Frontier

1. Back Propagation Error
2. Multi Layer Perceptron

۲-۳- ساختار DNN

ساختار DNN در شکل (۲) نشان داده شده است. DNN در ازای دریافت ورودی‌های z_1 و z_2 (دو راه حل کارا بر حسب مقادیر اهداف) نسبت مطلوبیت راه حل اول به راه حل دوم را تولید می‌کند. به منظور محاسبه خروجی نهایی گام‌های لازم به صورت زیر می‌باشند:

گام اول: ورودی توابع فعال‌ساز در لایه میانی به ازای ورودی‌های z_1 و z_2 براساس رابطه (۳) بدست می‌آید:

$$net_{i1}^h = \sum_{j=1}^n W_j^h z_1 \quad i = 1, \dots, n_h \quad (7)$$

$$net_{i2}^h = \sum_{j=1}^n W_j^h z_2 \quad i = 1, \dots, n_h$$

گام دوم: خروجی لایه میانی با عبور دادن بردارهای net_{i1}^h و net_{i2}^h از توابع فعال‌ساز بر اساس رابطه (۴) برای هر جزء MLP بدست می‌آید:

$$o_{i1}^h = f_h(net_{i1}^h) \quad (8)$$

$$o_{i2}^h = f_h(net_{i2}^h)$$

W_o بردار وزن لایه خروجی، f_o تابع فعال‌ساز لایه خروجی و U خروجی شبکه عصبی MLP می‌باشد. الگوریتم پیشرو شبکه عصبی MLP برای ورودی U به صورت زیر می‌باشد:

گام اول: ورودی هر تابع فعال‌ساز در لایه میانی بر اساس رابطه (۳) بدست می‌آید:

$$net_i^h = \sum_{j=1}^n W_j^h u_j \quad i = 1, \dots, n_h \quad (3)$$

گام دوم: خروجی هر عصب لایه میانی با عبور دادن مقادیر net_i^h از توابع فعال‌ساز بر اساس رابطه (۴) بدست می‌آید:

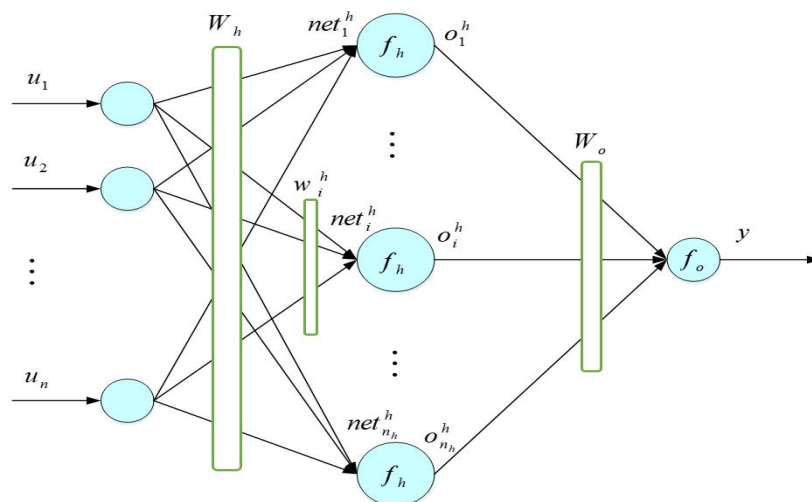
$$o_i^h = f_h(net_i^h) \quad i = 1, \dots, n_h \quad (4)$$

گام سوم: ورودی تابع فعال‌ساز در لایه خروجی بر اساس رابطه (۵) بدست می‌آید:

$$net^o = \sum_{i=1}^{n_h} W_i^o o_i^h \quad (5)$$

گام چهارم: خروجی شبکه‌های عصبی MLP به صورت رابطه (۶) بدست می‌آید:

$$y = MLP(U) = f_o(net^o) \quad (6)$$



شکل (۱): ساختار شبکه عصبی MLP

است. خروجی این دو شبکه به یک گره نهایی منتقل می‌گردد. خروجی این گره حاصل تقسیم خروجی‌های این دو شبکه پیشرو با ساختار همانند است. این نکته باعث می‌گردد که اطلاعات مربوط به مقایسات زوجی راه کارها که توسط تصمیم‌گیرنده ارائه می‌گردد، به راحتی برای این شبکه قابل استفاده گردد. زیرا هر گونه پیش پردازش اطلاعات مربوط به مقایسات زوجی، برای یافتن ارزش راه کارها، علاوه بر صرف وقت می‌تواند بر ساختار اطلاعات و آموزش شبکه عصبی تأثیر بگذارد. علاوه بر این، در این روش جداول مقایسات زوجی که بصورت ناقص تکمیل شده باشد، نیز می‌تواند در آموزش شبکه بکارگرفته شود. مزیت دیگر این شبکه، تعداد داده‌های آموزشی است که در هر مرحله مقایسه زوجی توسط تصمیم‌گیران به دست می‌آید.

در حالتی که از FFANN استفاده شود، تعداد داده‌های آموزشی برابر k (تعداد راه حل‌ها) خواهد بود، اما در حالتی که از DNN استفاده نماییم، تعداد داده‌های آموزشی برابر با $k(k-1)/2$ (تعداد مقایسات) خواهد بود. در این حالت تعداد دفعات مراجعه به تصمیم‌گیرندگان کمتر خواهد شد و کیفیت آموزش افزایش خواهد یافت.

گام سوم: ورودی توابع فعال‌ساز در لایه خروجی به صورت رابطه (۹) بدست می‌آید:

$$net^o_1 = \sum_{i=1}^{n_h} W_i^o o_{i1}^h$$

$$net^o_2 = \sum_{i=1}^{n_h} W_i^o o_{i2}^h$$
(۹)

گام چهارم: خروجی شبکه‌های عصبی MLP به صورت رابطه (۱۰) بدست می‌آید:

$$y_1 = f_o(net^o_1)$$

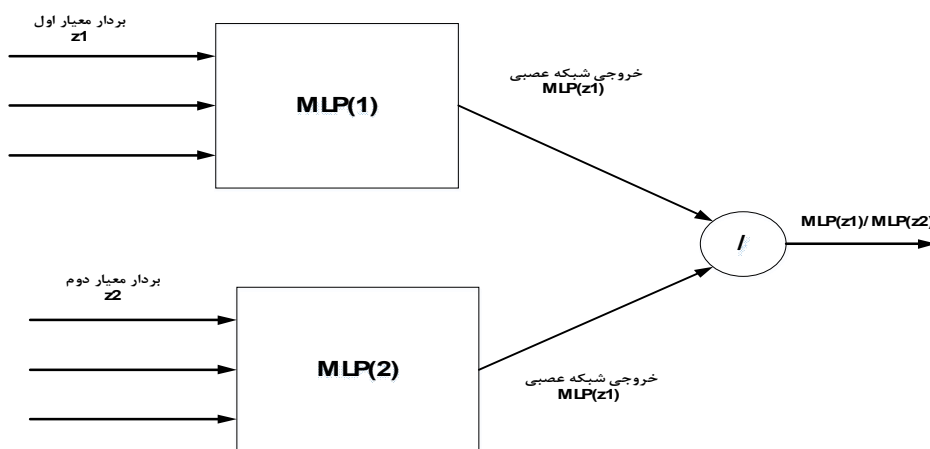
$$y_2 = f_o(net^o_2)$$
(۱۰)

گام پنجم: در نهایت خروجی شبکه عصبی DNN به صورت رابطه (۱۱) بدست می‌آید:

$$y_{DNN} = \frac{y_1}{y_2}$$
(۱۱)

۲-۴- بررسی توانمندی‌های DNN نسبت به MLP

ساختار DNN در شکل (۲) نشان داده شده است. این شبکه، شامل دو شبکه MLP با ساختار کاملاً یکسان



شکل (۲): ساختار DNN

$$E(k) = \frac{1}{2} e^2(k) = \frac{1}{2} (y_d(k) - y_{DNN}(k))^2 \quad (12)$$

بعد از محاسبه خطا، گام بعدی در الگوریتم پس انتشار خطا، محاسبه مشتق‌های مربعات خطا از طریق قانون زنجیری است. ابتدا مشتق خروجی شبکه عصبی نسبت به وزن‌های لایه آخر و میانی را بدست می‌آوریم:

$$\frac{\partial y_{DNN}}{\partial W^o} = \frac{\partial y_{DNN}}{\partial y_1} \frac{\partial y_1}{\partial W^o} + \frac{\partial y_{DNN}}{\partial y_2} \frac{\partial y_2}{\partial W^o} \quad (13)$$

با توجه به رابطه (۱۰) داریم:

$$\frac{\partial y_1}{\partial W^o} = \frac{\partial y_1}{\partial net_1^o} \frac{\partial net_1^o}{\partial W^o} = f'_o(net_1^o) o_1^h \quad (14)$$

$$\frac{\partial y_2}{\partial W^o} = \frac{\partial y_2}{\partial net_2^o} \frac{\partial net_2^o}{\partial W^o} = f'_o(net_2^o) o_2^h$$

که در آن، $\frac{\partial f_o}{\partial net_1^o} = f'_o(net_1^o)$ و $\frac{\partial net_1^o}{\partial W^o} = o_1^h$ و $\frac{\partial f_o}{\partial net_2^o} = f'_o(net_2^o)$ و $\frac{\partial net_2^o}{\partial W^o} = o_2^h$ می‌باشد.

با استفاده از روابط (۱۱) و (۱۳) و (۱۴) داریم:

$$\frac{\partial y_{DNN}}{\partial W^o} = \frac{1}{y_2} f'_o(net_1^o) o_1^h - \frac{y_1}{y_2^2} f'_o(net_2^o) o_2^h \quad (15)$$

وزن‌های متصل به I-امین نرون در لایه میانی و ورودی‌های (w_i^h) را در نظر می‌گیریم. به طریق مشابه برای این وزن‌ها داریم:

$$\frac{\partial y_{DNN}}{\partial w_i^h} = \frac{\partial y_{DNN}}{\partial y_1} \frac{\partial y_1}{\partial net_1^o} \frac{\partial net_1^o}{\partial o_{i1}^h} \frac{\partial o_{i1}^h}{\partial net_{i1}^h} \frac{\partial net_{i1}^h}{\partial w_i^h} + \frac{\partial y_{DNN}}{\partial y_2} \frac{\partial y_2}{\partial net_2^o} \frac{\partial net_2^o}{\partial o_{i2}^h} \frac{\partial o_{i2}^h}{\partial net_{i2}^h} \frac{\partial net_{i2}^h}{\partial w_i^h} \quad (16)$$

$$\frac{\partial y_{DNN}}{\partial w_i^h} = \frac{1}{y_2} f'_o(net_1^o) w_i^o f'_h(net_{i1}^h) z_1 \quad (17)$$

$$- \frac{y_1}{y_2^2} f'_o(net_2^o) w_i^o f'_h(net_{i2}^h) z_2$$

با توجه با نکات مذکور، توسعه روش‌های آموزش DNN می‌تواند ابزاری توانا برای حل مسایل چندهدفه با توابع مطلوبیت خطی یا غیر خطی ایجاد نماید. لذا هدف از این تحقیق معرفی روش‌های کارا برای آموزش شبکه DNN است تا با توجه به مزایای زیاد آن بتوان از این شبکه به طور کارآمد در حل مسایل تصمیم‌گیری چند معیاره بهره برد.

۳- روش پیشنهادی

۳-۱- آموزش DNN

ابتدا تعداد محدودی جواب اولیه (راه حل موثر) برای مسئله تصمیم‌گیری چندهدفه مورد نظر تولید و با مقایسه زوجی این جواب‌ها تعدادی داده آموزشی برای شبکه عصبی تولید می‌شود. نتیجه هر مقایسه یک داده آموزشی به صورت (z_i, z_j, α_{ij}) خواهد بود که در آن، z_i و z_j دو ورودی متمایز و α_{ij} نسبت خروجی‌های تابع ارزیابی مطلوبیت متناظر با ورودی‌های z_i و z_j می‌باشد. اگر تعداد جواب‌های اولیه k تا باشد، تعداد داده‌های آموزشی $\frac{k(k-1)}{2}$ خواهد بود.

فرآیند یادگیری برای یک شبکه عصبی، به این صورت است که با در دست داشتن پاسخ مطلوب y_d برای دو بردار ورودی مورد نظر یک فرمول تطبیق‌پذیر برای وزن‌های شبکه بدست می‌آید به نحوی که خروجی شبکه برای این دو بردار ورودی بقدر کافی به y_d نزدیک شود و یا اینکه شبکه عصبی دانش لازم را از پاسخ مطلوب y_d کسب نماید. برای تنظیم وزن‌ها از الگوریتم گردیان نزولی (GD¹) و گرادیان مزدوج (CG¹) استفاده می‌شود که نتایج حاکی از آن است که روش گرادیان مزدوج که در این تحقیق ارایه شده است، دارای سرعت همگرایی و کیفیت جواب‌های بهتری نسبت به روش گرادیان نزولی [۱] می‌باشد.

مربع خطای بین پاسخ مطلوب و خروجی شبکه در مرحله k ، به عنوان تابع معیار عملکرد شبکه در نظر گرفته می‌شود:

2. Gradient Descend
3. Conjugate Gradient

$$\begin{aligned} \Delta W^o(k-1) &= W^o(k-1) - W^o(k-2) \\ \Delta W^h(k-1) &= W^h(k-1) - W^h(k-2) \end{aligned} \quad (21)$$

در رابطه (۱۹) J_h بر اساس مشتقات جزئی $\frac{\partial y_{DNN}}{\partial W^h}$ و J_o نیز بر اساس مشتقات جزئی $\frac{\partial y_{DNN}}{\partial W^o}$ محاسبه می‌گردد.

۳-۲- ساختار شبکه‌های MLP در DNN

در این مقاله در ساختار DNN، شبکه‌های MLP سه لایه در نظر گرفته شده‌اند. تعداد نرون‌ها در لایه اول این شبکه‌ها، برابر تعداد اهداف، در لایه پنهان برابر سه و در لایه خروجی برابر با یک می‌باشد. توابع فعال‌ساز برای لایه پنهان یکسان و از تابع سیگموئید تک قطبی استفاده شده است. در لایه خروجی نیز تابع فعال‌ساز خطی بکارگرفته شده است.

۴- مثال‌های عددی نمونه (نتایج عددی)

در این بخش چندین مثال کاربردی برای نشان دادن کارایی روش پیشنهادی ارائه می‌گردد. در دو مثال اول قابلیت تخمین تابع ارزیابی مطلوبیت با استفاده از روش پیشنهادی نشان داده شده و در مثال سوم شبکه عصبی DNN در حل یک مسئله تصمیم‌گیری چندهدفه به کار برده می‌شود.

۴-۱- مثال اول - تابع مطلوبیت خطی

در مثال اول، روش پیشنهادی برای تخمین تابع مطلوبیت زیر به کار برده می‌شود:

$$v(z_1, z_2) = 0.85z_1 + 0.15z_2 \quad (22)$$

که در آن z_1, z_2 اعداد تصادفی بین صفر و یک هستند. ابتدا هشت جواب اولیه تولید کرده و با استفاده از این هشت جواب به تعداد $28 = (8-1)/2$ داده آموزشی برای شبکه عصبی بدست می‌آوریم. با استفاده از این داده‌های آموزشی، وزن‌های شبکه عصبی را با روش‌های آموزشی مختلفی که در بخش قبل مطرح گردید تنظیم می‌کنیم. نتایج روش پیشنهادی بعد از ۵۰۰۰ تکرار در جدول (۲) نشان داده شده است. در این جدول $v(z)$

که در آن W_i^o وزن اتصال خروجی و $i=1$ -امین نرون لایه میانی در MLP(1) و MLP(2) است. $f'_h(\text{net}_{i1}^h)$ مشتق تابع فعال‌ساز در لایه میانی نسبت به net_{i1}^h می‌باشد.

۳-۱-۱- روش گرادیان نزولی

برای آموزش از طریق گرادیان از رابطه زیر استفاده می‌کنیم:

$$\begin{aligned} W^h &= W^h + \eta \frac{\partial y_{DNN}}{\partial W^h} e \\ W^o &= W^o + \eta \frac{\partial y_{DNN}}{\partial W^o} e \\ e &= y_d - y_{DNN} \end{aligned} \quad (18)$$

که در آن e خطا بوده و به صورت $e = y_d - y_{DNN}$ تعریف می‌شود، y_{DNN} خروجی DNN و y_d خروجی مطلوب می‌باشد.

۳-۱-۲- روش گرادیان مزدوج

روشی که در اینجا برای بهبود الگوریتم آموزش شبکه عصبی تصمیم پیشنهاد می‌شود، روش گرادیان مزدوج است. فرم معادله اساسی الگوریتم CG همانند الگوریتم گرادیان بدین صورت است: قانون تنظیم وزن‌های DNN بر اساس این الگوریتم، بصورت زیر انتخاب می‌شود.

$$\begin{aligned} W^o(k+1) &= W^o(k) - \\ & J_o(k) + \alpha_o(k-1) \Delta W^o(k-1) \\ W^h(k+1) &= W^h(k) - \\ & J_h(k) + \alpha_h(k-1) \Delta W^h(k-1) \end{aligned} \quad (19)$$

که در آن، تنظیم پارامتر $\alpha(k)$ به صورت معادلات (۲۰) می‌باشد.

$$\begin{aligned} \alpha_o(k) &= \frac{(J_o(k+1))^T J_o(k+1)}{(J_o(k))^T J_o(k)} \\ \alpha_h(k) &= \frac{(J_h(k+1))^T J_h(k+1)}{(J_h(k))^T J_h(k)} \end{aligned} \quad (20)$$

شکل (۴) نشان داده شده است. MSE مجذور مربعات خطایی است که بر اساس خروجی DNN بدست می‌آید (شکل ۳) ولی خطای تعریف شده در جدول ۲ براساس خروجی MLP می‌باشد. همانطور که دیده می‌شود روش پیشنهادی آموزش براساس روش CG نتایج بسیار بهتری نسبت به روش آموزش مقاله مرجع داشته است.

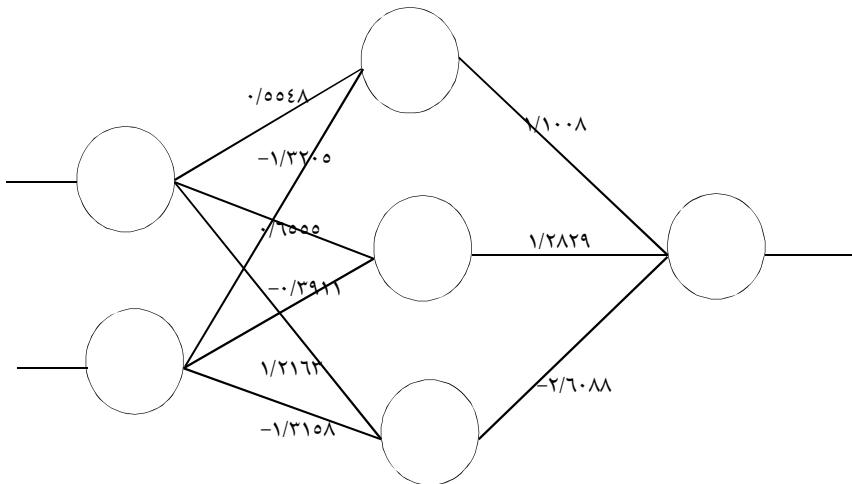
مقدار تابع مطلوبیت (رابطه ۲۴) بوده و $MLP(z)$ خروجی شبکه عصبی MLP در DNN می‌باشد. در شکل (۳) آورده در MLP شبکه عصبی با روش‌های مختلف آموزشی در MSE شده است. نمودار

جدول ۲- نتایج مثال اول با روش پیشنهادی CG

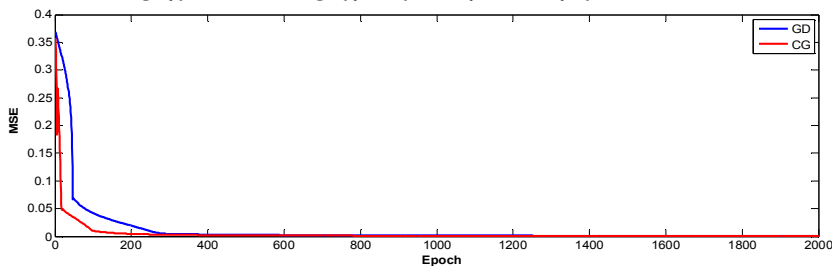
Output of MLP	MLP/K	V	Error
۰/۴۰۲۴	۰/۴۵۰۵	۰/۳۵۹۵	۰/۰۱۸
۰/۷۴۳۶	۰/۸۳۲۴	۰/۶۶۵۸	۰/۲۴۷
۰/۶۸۳۳	۰/۷۶۵۰	۰/۶۱۲۴	۰/۳۴۰
۰/۵۳۲۳	۰/۵۹۵۰	۰/۴۷۴۵	۰/۱۹۸
۰/۸۶۸۱	۰/۹۷۱۹	۰/۷۷۵۴	۰/۰۰۳
۰/۲۸۰۷	۰/۳۱۴۳	۰/۲۵۱۰	۰/۱۰۷
۰/۸۱۹۷	۰/۹۱۷۷	۰/۷۳۰۹	۰/۱۸۱
۰/۷۱۹۴	۰/۸۰۵۴	۰/۶۴۰۸	۰/۲۷۱

$$\times \text{Error} = |MLP(z)K - v(z)| / v(z), K = \sum v(z) / \sum MLP(z)$$

شکل (۳): شبکه MLP در DNN برای تابع مطلوبیت (۲۴)



شکل (۴): نمودار MSE در مثال اول با روش‌های مختلف آموزش



نوسانات از بین می‌رود ولی در عوض سرعت همگرایی کمتر می‌شود.

۳-۴- مثال سوم

در این مثال یک مسئله تصمیم‌گیری سه هدفه در نظر گرفته شده‌است. کاربرد مؤثر شبکه عصبی DNN در حل این مسئله را نشان داده و نتایج با برخی روش‌های دیگر مقایسه می‌شود.

$$\begin{aligned}
 \max z_1 &= 2x_2 + 5x_3 + 5x_4 - 2x_5 + 5x_6 \\
 \max z_2 &= -x_1 - 2x_2 + 4x_5 - x_6 \\
 \max z_3 &= 5x_1 + 3x_2 - 2x_3 - x_5 - x_6 \\
 \text{subject to} \\
 7x_4 + 2x_5 + 6x_6 &\leq 28 \\
 3x_1 + 4x_6 &\leq 23 \\
 4x_1 + 4x_3 + x_4 &\leq 23 \\
 x_2 + 6x_3 + 7x_4 + 4x_6 &\leq 23 \\
 2x_1 + 5x_2 + 5x_3 + 5x_4 + 8x_5 &\leq 29 \\
 x_j &\geq 0 \quad 1 \leq j \leq 6
 \end{aligned} \tag{۲۴}$$

۲-۴- مثال دوم - تابع مطلوبیت غیرخطی

در این مثال یک تابع غیرخطی به صورت زیر، به عنوان تابع مطلوبیت در نظر گرفته می‌شود:

$$v = L - \left(\sum \lambda_i (z_i^* - z)^p \right)^{\frac{1}{p}} \tag{۲۳}$$

که در آن،

$$p = 4, L = 1, \lambda_1 = 0.220, \lambda_2 = 0.472, \lambda_3 = 0.308$$

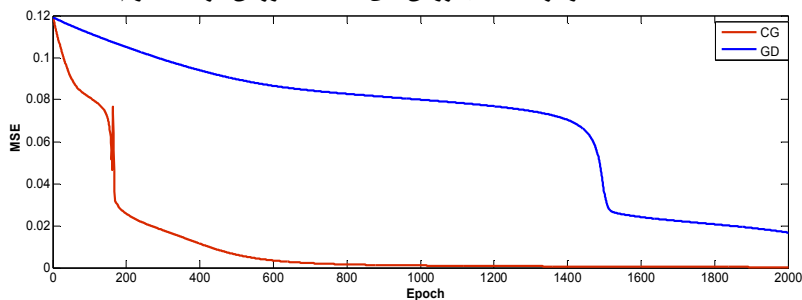
$$z^* = (1, 1, 1)^T \text{ و}$$

مشابه مثال اول، هفت جواب اولیه بدست آورده و با ۲۱ نمونه آموزشی پارامترهای شبکه عصبی را با روش‌های مختلف تنظیم می‌کنیم. نتایج در جدول (۳) آورده شده‌است. نمودار MSE با توجه به روش‌های مختلف آموزش در شکل (۵) نشان داده شده است. همانطور که دیده می‌شود نتایج روش پیشنهادی بسیار بهتر می‌باشد. نوسان مقدار MSE حول مقدار بهینه در این الگوریتم‌ها طبیعی است. در واقع در همه این الگوریتم‌ها یک مصالحه‌ای بین سرعت و پایداری باید صورت بگیرد. اگر مقدار اولیه گام آموزشی را کوچک انتخاب کنیم این

جدول ۳- نتایج مثال دوم با روش CG

Output of MLP	MLP/K	V	Error
۰/۵۰۵۸	۰/۴۸۳۷	۰/۵۲۸۸	۰/۰۱۱٪
۰/۳۳۳۶	۰/۳۱۹۱	۰/۳۴۹۰	۰/۰۶۲٪
۰/۴۹۰۷	۰/۴۶۹۳	۰/۵۱۲۹	۰/۰۲۹٪
۰/۵۷۴۶	۰/۵۴۹۵	۰/۶۰۱۱	۰/۰۴۳٪
۰/۲۹۷۲	۰/۲۸۴۲	۰/۳۱۰۶	۰/۰۵۲٪
۰/۴۶۰۵	۰/۴۴۰۴	۰/۴۸۰۸	۰/۱۴۴٪
۰/۳۷۳۷	۰/۳۵۷۴	۰/۳۹۱۳	۰/۱۵۲٪

شکل (۵): نمودار MSE با روش‌های مختلف آموزش در مثال دوم



می‌شود. نتایج بدست آمده در جدول (۷) آرایه شده و با برخی روش‌های دیگر مقایسه شده است. همانطور که دیده می‌شود روش پیشنهادی در این مقاله جواب دقیق‌تری داده است.

max MLP (z')

$$z'_1 = \frac{z_1 + 7.25}{33.1 + 7.25}$$

$$z'_2 = \frac{z_2 + 16.412}{14.5 + 16.412}$$

$$z'_3 = \frac{z_3 + 9.207}{39.250 + 9.207}$$

$$z_1 = 2x_2 + 5x_3 + 5x_4 - 2x_5 + 5x_6$$

$$z_2 = -x_1 - 2x_2 + 4x_5 - x_6$$

$$z_3 = 5x_1 + 3x_2 - 2x_3 - x_5 - x_6$$

subject to

$$7x_4 + 2x_5 + 6x_6 \leq 28$$

$$3x_1 + 4x_6 \leq 23$$

$$4x_1 + 4x_3 + x_4 \leq 23$$

$$x_2 + 6x_3 + 7x_4 + 4x_6 \leq 23$$

$$2x_1 + 5x_2 + 5x_3 + 5x_4 + 8x_5 \leq 29 \quad (24)$$

$$x_j \geq 0 \quad 1 \leq j \leq 6$$

تابع مطلوبیت به صورت زیر می‌باشد:

$$v = 50 - \left(\sum_{i=1}^3 \lambda_i (z_i^{\max} - z_i) \right)^{\frac{1}{4}} \quad (25)$$

که در آن، $\lambda_1 = 0.319, \lambda_2 = 0.416, \lambda_3 = 0.265$ جواب بهینه آن در جدول (۴) نشان داده شده است.

برای حل این مسئله به کمک روش پیشنهادی، ابتدا هفت جواب اولیه از این مسئله را به صورت جدول (۵) بدست می‌آوریم، که در آن z'_i ها مقادیر نرمالیزه شده z_i ها می‌باشند (برای نرمال سازی از روش نرمال سازی ماکس-مین که در مدل ریاضی مساله آورده شده است استفاده می‌شود. برای جزئیات بیشتر به مقاله [۲۱] مراجعه کنید). با استفاده از این جواب‌های اولیه مشابه مثال‌های اول و دوم، تابع ارزیابی مطلوبیت (۲۵) را با شبکه عصبی تخمین می‌زنیم. همچنین برای نشان دادن قابلیت روش پیشنهادی علاوه بر داده‌های جدول (۵)، داده‌های غیردقیقی به صورت جدول (۶) را وارد مسئله کرده و بار دیگر با مجموعه‌های جدید، مسئله (۲۴) را حل می‌کنیم. با در نظر گرفتن مقادیر بیشینه و کمینه برای هر یک از جواب‌ها، مسئله (۲۴) بر اساس روش پیشنهادی به صورت مدل ریاضی (۲۶) بازنویسی

جدول (۴): جواب بهینه تابع مطلوبیت مثال سوم [۲۱]

V^*	Z_3	Z_2	Z_1	X_6	X_5	X_4	X_3	X_2	X_1
۴۲/۴۲۳	۱۸/۹۵۶	-۰/۱۷۸	۱۶/۵۱۳	۲/۳۳۱	۱/۵۰۳	۱/۵۷۲	.	.	۴/۵۵۸

جدول (۵): جواب‌های اولیه برای مسئله مثال سوم [۲۱]

$V'(Z)$	تابع هدف سوم		تابع هدف دوم		تابع هدف اول		راه حل
	Z'_3	Z_3	Z'_2	Z_2	Z'_1	Z_1	
۰/۳۵۷۹۷	۰/۷۶۰۵۲	۲۷/۶۴۵۴	۰/۱۵۷۳۳	-۱۱/۵۴۸۶	۰/۷۸۳۲۶	۲۴/۳۵۴۶	۱
۰/۱۶۱۱۷	۰/۱۰۸۷۷	-۳/۹۳۶۳	۰/۹۸۹۹۳	۱۴/۱۸۸۶	۰/۰۳۸۵۸	-۵/۶۹۳۲	۲
۰/۲۵۵۰۱	۰/۰۲۷۲۵	-۷/۸۸۶۵	۰/۶۰۳۹۶	۲/۲۵۷۵	۰/۷۴۶۲۴	۲۲/۸۶۱۰	۳
۰/۲۷۳۰۸	۰/۴۸۲۹۶	۱۴/۱۹۵۹	۰/۷۷۲۱۸	۷/۴۵۷۵	۰/۰۶۵۲۴	-۴/۶۱۷۵	۴
۰/۳۴۶۶۷	۰/۳۳۰۸۳	۶/۸۲۴۲	۰/۲۳۳۰۴	-۹/۲۰۸۳	۰/۹۱۲۵۰	۲۹/۵۶۹۴	۵
۰/۳۴۷۸۵	۰/۸۹۲۳۹	۳۴/۰۳۵۴	۰/۳۲۷۸۷	-۶/۲۷۶۷	۰/۲۳۷۳۰	۲/۳۲۴۹	۶
۰/۳۰۷۷۰	۰/۷۶۵۷۷	۲۷/۹۰۰۱	۰/۵۷۴۲۶	۱/۳۳۹۵	۰/۱۰۰۹۷	-۳/۱۷۵۸	۷

جدول (۶): داده‌های خاص مربوط به مثال سوم [۲۱]

راه حل	z_1'	z_2'	z_3'	V'
۸	۱	۰	۰	۰/۰۹۶۹
۹	۰	۱	۰	۰/۰۹۶۵
۱۰	۰	۰	۱	۰/۰۹۵۹
۱۱	۱	۱	۱	۱

جدول (۷): مقایسه نتایج بدست آمده در مثال سوم

مرجع [۲۱]	روش پیشنهادی CG	مقادیر واقعی
تعداد مقایسه	۲۴	۲۴
تعداد داده‌های آموزشی	۲۴	۲۴
مقدار نهایی V	۴۲/۱۶۶	۴۲/۳۹۷۶
z_1	۱۸/۷۱۹	۱۷/۶۶۶۷
z_2	-۳/۱۷۸	-۱/۳۲۳۳
z_3	۲۲/۳۹۵	۱۸/۹۷۰

۵- نتیجه‌گیری

در بسیاری از مسایل چند هدفه دنیای واقعی، فرض جمع‌پذیری تابع مطلوبیت برقرار نمی‌باشد و تخمین توابع غیرخطی با شیوه‌های متداول کاری پیچیده و توام با خطای بسیار می‌باشد، توانمندی شبکه‌های عصبی در تخمین توابع پیچیده و غیرخطی در تحقیقات بسیاری بررسی و تایید شده‌است. لذا در این تحقیق به منظور بهبود تخمین توابع مطلوبیت- به خصوص توابع غیرخطی پیچیده- رویکردی نوین برای آموزش شبکه‌های عصبی تصمیم‌ارایه می‌گردد. این شبکه‌ها به صورت کارآمد می‌توانند به تخمین تابع مطلوبیت ضمنی تصمیم‌گیرنده نایل آمده و با استفاده از رویکردهای تعاملی جواب‌هایی نزدیک به جواب بهینه تابع مطلوبیت مدل را با توجه به محدودیت‌های مدل بیابند. در این تحقیق رویکرد ارایه شده برای حل مثال‌های عددی به کار گرفته شد. نتایج با روش‌های معرفی شده در تحقیقات پیشین مقایسه گردیدند. نتایج عددی حاکی از آن است که رویکرد پیشنهادی قادر است نسبت به روش‌های پیشین نتایج بهتری ارایه نماید و می‌تواند در حل مسائل کاربردی مورد استفاده قرار بگیرد. همانطور که با شکل‌ها و جداول مختلف نشان داده شد، این روش قابلیت همگرایی

سریع‌تری نسبت به روش‌های پیشین دارد و می‌تواند جواب‌های بهتر را در زمان کوتاه‌تری بیابد.

فهرست منابع

- [10] Geoffrion, M, Dyer, J, S and A. Feinberg, An interactive approach for multi-criterion optimization, with an application to the operation of an academic department, *Manage. Sci.*, 1972, vol. 19, no. 4, pp. 357–368.
- [11] Gardiner, L, R and Steuer, R, E, Unified interactive multiple objective programming, *Eur. J. Oper. Res.*, 1994, vol. 74, pp. 391–406.
- [12] Sun, M, Stam, A, Steuer, R. E., Interactive multiple objective programming problems using Tchebycheff programs and artificial neural networks. *Computer & Oper. Res.*, 2000, 27, PP.601–620.
- [13] Sun, M, Stam, A, Steuer, R. E., “Solving multiple objective programming problems using feed-forward artificial neural networks: The interactive FFANN procedure,” *Manage. Sci.*, 1996, vol. 42, No. 6, pp. 835–849.
- [14] Malakooti, B and Zhou, Y, Feed-forward artificial neural networks for solving discrete multiple criteria decision making problems, *Manage.Sci.*, , 1994, vol. 40, No. 11, pp. 1542–1560.
- [15] R. Hecht-Nielsen, “Theory of the backpropagation neural networks,” in *Proc. Int. Joint Conf. Neural Networks*, 1989, pp. 593–611.
- [16] Shih, H.S., Wen, U.P., Lee, E.S., Lan, K.M., Hsiao, H.C., A neural network approach to multi objective and multilevel programming problems, *comput & math*, 2004, vol. 48, pp. 95–108.
- [17] Huang, H.Z, Tian, Z, J. ZUO, M, Intelligent interactive multi objective optimization method and its application to reliability optimization, *IIE Transactions* 2005, 37, 983–993.
- [1] Chen, J. & Lin, S, A Neural Network Approach- Decision Neural Network (DNN) for preference assessment, *IEEE Trans. Syst. Man, Cyber net*, 2004, vol 34, No. 2, pp.219-225.
- [2] Wang, J., & Malakooti, B., A feed forward neural network for multiple criteria decision making, *Computer. Oper. Res.*, 1992, Vol. 19, No. 2, pp. 151-167.
- [3] Keeney, R. L., Raffia, H, Decisions with multiple objectives. Cambridge Univ. Press, U.K., 1993.
- [4] Bell, D. E., Consistent assessment procedures using conditional utility functions. *Oper. Res.*, 1979, vol 27.PP. 1054-1066.
- [5] Dyer, J. S., & Sarin, R. K., “Measurable multi attribute value functions”, *Ops Res* , 1979, vol. 27, PP. 810-822.
- [6] Farquhar, P. H., “A fractional hypercube decomposition theorem for multi attribute utility functions”, *Ops Res.* 1975, Vol. 23, pp. 941-967.
- [7] Fishburn, P. G., “von Neumann Morgenstern utility functions on two attributes”, *Ops Res*, 1974, Vol. 22, pp. 35-45.
- [8] Kirkwood, C. W., “Parametrically dependent preferences for multi attributed consequences”, *Ops Res*, 1976, Vol. 24, pp. 92-103.
- [9] Benayoun, R, de Montgolfier, J, Tergny, J and Laritchev, O, I, Linear programming with multiple objective functions: Step method (STEM), *Math. Program*, 1971, vol. 1, pp. 366–387.

[18] Gen, M, Ida, K. and Kobuchi, R, Neural network technique for fuzzy multiobjective linear programming, computer and industrial engineering”, 1998, vol 35, NO. 3, PP. 543-546.

[19] Golmohammadi, D, Neural network application for fuzzy multi-criteria decision making problems, Int. J. production Economics 131, 2011, pp. 490–504.

[20] Golmohammadi, D, A neural network decision-making model for job-shop scheduling, International Journal of Product Res, 2013, Vol. 51, No. 17, pp. 5142–5157.

[21] Chen, J. & Lin, S., “An interactive neural network-based approach for solving multiple criteria decision-making problems”, Decision support systems, 2003, vol. 36, pp. 137-146.

[22] Gal, T., “Nonessential objective within network approaches for MCDM”, European Journal of Operation Research, 2006, vol. 168, pp. 584-592.

[23] Singh, R.K., Choudhury, A.K., Tiwari, M.K., Shankar, R, “Improved Decision Neural Network (IDNN) based consensus method to solve a multi-objective group decision making problem”, Advanced Engineering Informatics, 2007, vol 21, pp. 335-348.