مجله تحقیق در عملیات در کاربردهای آن، سال شانزدهم، شماره سوم (پیاپی ۶۲)، پاییز ۹۸، ص ص ۲۰-۱
شاپا ۷۲۸۶-۲۲۵۱

یک شبکه عصبی اصلاح شده کارا برای حل مسایل برنامه‌ریزی غیرخطی با محدودیت‌های هیبریدی

علیرضا ناظمی[1*]، سمیرا سوخت سرایی[2]، مرضیه مرتضایی[3]
۱- دانشیار، دانشگاه صنعتی شاهرود، گروه ریاضی کاربردی، شاهرود، ایران
۲- دانش آموخته کارشناسی ارشد، دانشگاه صنعتی شاهرود، گروه ریاضی کاربردی، شاهرود، ایران
۳- دانشجوی دکتری، دانشگاه صنعتی شاهرود، گروه ریاضی کاربردی، شاهرود، ایران

رسید مقاله: ۱۷ بهمن ۱۳۹۶
پذیرش مقاله: ۲۴ اسفند ۱۳۹۷

چکیده
یک مدل شبکه عصبی کارا برای حل یک مساله بهینه‌سازی غیرخطی با محدودیت‌های هیبریدی ارایه می‌شود. بر مبنای شرایط بهینگی کاروش-کان-تاکر و مفاهیمی از آنالیز محدب و بهینه‌سازی، ابتدا یک مدل شبکه ارایه می‌شود. ثابت می‌شود که مدل شبکه عصبی ارایه شده پایدار لیاپانف بوده و همگرای سراسری به جواب مساله بهینه‌سازی اصلی می‌باشد. با ارایه چندین مثال عددی، کارایی مدل ارایه شده برای حل مساله بهینه‌سازی با محدودیت‌های هیبریدی نشان داده می‌شود.

کلمات کلیدی: برنامه‌ریزی‌محدب، پایدار لیاپانوف، محدودیت‌های هیبریدی، شبکه عصبی، همگرای سراسری.

۱مقدمه
مسایل بهینه‌سازی درجه دوم در طیف وسیعی از کاربردهای علمی و مهندسی از جمله پردازش سیگنال، شناسایی سیستم، طراحی صافی، تقریب توابع، تجزیه و تحلیل رگرسیون و... به‌کار برده می‌شود [۱]-[۸]. در بسیاری از مسایل بهینه‌سازی کاربردی مثل برنامه‌ریزی سیستم‌های قدرت و مسیریابی سیستم‌های ارتباط از راه دور، تعداد متغیرهای تصمیم‌گیری و محدودیت‌ها به‌طور معمول بسیار زیاد هستند [۹]-[۱۴]. چالش اصلی زمانی به‌وجود می‌آید که نیاز به یک جواب زمان حقیقی[1] در یک مساله بهینه‌سازی با مقیاس بزرگ باشد [۱۵]. برای چنین کاربردهایی، روش‌های بهینه‌سازی کلاسیک ممکن است با توجه به افزایش بعد مساله و الگوریتم مورد استفاده در زمان محاسبات مناسب نباشد. یک راه حل بسیار امیدوارکننده، روش‌های بهینه‌سازی مبتنی بر شبکه‌های عصبی مصنوعی برای حل این نوع از مسایل است.

* عهده‌دار مکاتبات
آدرس الکترونیکی:web2_alireza.nazemi@shahroodut.ac.ir

[1] Real time

[Downloaded from jorlu.ir on 2026-11-29]
[DOR: 20.1001.1.22517286.2019.16.3.6.9]

شبکه‌ی عصبی از مک کلاچ و پژوهش‌های پیشگامانه‌ی نیم قرن پیش توسط پیتز [۱۶] ناشی می‌شود. از آن زمان به بعد مدل‌های شبکه عصبی مختلفی ایجاد شده است. ایده اصلی در استفاده از شبکه‌های عصبی مصنوعی برای مسایل بهینه‌سازی استفاده از یک تابع انرژی و یک سیستم دینامیکی است که این دو، بیان کننده‌ی مدل‌های شبکه عصبی مصنوعی متناظر مسایل بهینه سازی هستند. سیستم دینامیکی بیان شده یک دستگاه معادلات دیفرانسیل غیرخطی مرتبه اول است. انتظار می‌رود که برای یک نقطه‌ی آغازین نقطه‌ی تعادل دستگاه معادلات دیفرانسیل غیرخطی به‌دست آمده، جواب بهینه‌ی مساله بهینه‌سازی اصلی باشد. یک اصل اساسی در استفاده از تابع انرژی این است که برای همگرایی دستگاه معادلات دیفرانسیل به نقطه‌ی تعادل آن، تابع انرژی متناظر با آن باید نامنفی و مشتق آن نسبت به زمان باید منفی باشد. مزیت اصلی استفاده از شبکه‌های عصبی برای مسایل بهینه‌سازی این است که فرآیند یافتن جواب در یک سیستم دینامیکی به‌طور ذاتی به‌صورت موازی است؛ بنابراین روش شبکه عصبی مسایل بهینه‌سازی را به‌صورت در زمان[1] حل می‌کند که خیلی سریع‌تر از الگوریتم‌های کلاسیک روی کامپیوترهای دیجیتال عمل می‌کند. مزیت دیگر روش شبکه‌های عصبی برای حل مسایل بهینه‌سازی این است که بر خلاف بسیاری از روش‌های کلاسیک مانند روش سیمپلکس که وابسته به نقطه شروع هستند، این روش‌ها هیچ نوع وابستگی به نقطه شروع ندارند و به اصطلاح همگرای سراسری هستند. در حقیقت نقطه شروع چه داخل، روی مرز یا خارج ناحیه شدنی باشد، این روش‌ها همگرا به جواب هستند.

در سال‌های اخیر، شبکه‌های عصبی زمان پیوسته‌ی بسیاری برای حل مسایل بهینه‌سازی مختلفی توسعه یافته و برخی نتایج بسیار مهم نیز به دست آمده است. به‌عنوان مثال می‌توان به حل مسایل برنامه‌ریزی خطی و درجه دوم، مسایل بهینه‌سازی غیرخطی، مسایل نامساوی وردشی و مسایل مکمل خطی و غیرخطی اشاره کرد [۱۷]-[۵۳]. در سال ۲۰۱۳، ناظمی در [۳۳] یک مدل شبکه عصبی کارا برای حل صورت کلی مسایل بهینه‌سازی غیرخطی محدب پیشنهاد کرد. از آن زمان به بعد ناظمی و همکاران [۳۴]-[۴۰] موفق شدند با حل مسایل مختلف کارایی مدل ارایه شده را نشان دهند. هدف این مقاله توسعه و اصلاح یک شبکه عصبی جدید از مدل بیان شده در [۳۳] برای حل مسایل برنامه‌ریزی غیر خطی محدب با قیود هیبریدی است. این مدل ساخته شده دارای تعداد متغیرهای کم‌تر، پیچیدگی ساختاری پایین‌تر و شرایط همگرایی ساده‌تری است. این موضوع به‌عنوان نوآوری اصلی این مقاله است که در بخش چهارم از مقاله به‌طور مفصل مورد بررسی قرار خواهد گرفت.

ساختار این مقاله به هفت بخش تقسیم می‌شود. در بخش دوم صورت کاملی از یک مساله برنامه‌ریزی غیرخطی با محدودیت‌های هیبریدی بیان می‌گردد. یک مدل شبکه عصبی کارا در بخش سوم برای حل مساله اصلی پیشنهاد می‌شود. در بخش چهارم، مقایسه کاملی بین مدل ارایه شده و برخی مدل‌های موجود بیان می‌گردد. پایداری و همگرایی سراسری مدل بیان شده به‌طور کامل در بخش پنجم از مقاله مورد بحث قرار می‌گیرد. در بخش ششم، نتایج شبیه‌سازی و مثال‌های به‌کار رفته، کارایی شبکه عصبی پیشنهادی را به‌طور کامل تایید می‌کنند. در نهایت مقاله با نتیجه‌گیری در بخش هفتم به پایان می‌رسد.

[1] Online

[Downloaded from jorlu.ir on 2026-11-29]

[DOR: 20.1001.1.22517286.2019.16.3.6.9]

۲ بیان مساله

مساله‌ی برنامه‌ریزی غیرخطی با محدودیت‌های هیبریدی را به‌صورت زیر در نظر بگیرید:

$$\text{Min} \quad f(x) \tag{۱}$$

s.t.

$$Ax = b, \tag{۲}$$

$$Hx \le g, \tag{۳}$$

$$x \in Q, \tag{۴}$$

که در آن $f:\mathbb{R}^n \to \mathbb{R}$ محدب و به‌طور پیوسته دوبار مشتق‌پذیر است، $A \in \mathbb{R}^{l\times n}$، $H \in \mathbb{R}^{m\times n}$، $b \in \mathbb{R}^l$، $g \in \mathbb{R}^m$ و $\operatorname{rank}(A) = l$. همچنین

$$Q = \left\{ x \in \mathbb{R}^n : \quad \alpha_i \le x_i \le \beta_i \quad (i = ۱,\cdots,n) \right\},$$

که در آن β_i ها می‌توانند $+\infty$ و α_iها می‌توانند $-\infty$ باشند. فرض کنید

$$D = \operatorname{diag}(۱,۱,\cdots,۱) \in \mathbb{R}^{n\times n},$$

$$B = \begin{pmatrix} H \\ D \\ -D \end{pmatrix} \in \mathbb{R}^{(m+۲n)\times n},$$

$$\beta = \begin{pmatrix} \beta_۱ \\ \vdots \\ \beta_n \end{pmatrix} \in \mathbb{R}^n,$$

$$\alpha = \begin{pmatrix} -\alpha_۱ \\ \vdots \\ -\alpha_n \end{pmatrix} \in \mathbb{R}^n,$$

$$d = \begin{pmatrix} g \\ \beta \\ \alpha \end{pmatrix} \in \mathbb{R}^{m+2n}.$$

در این صورت مساله (۱)-(۴) به‌صورت زیر تبدیل می‌شود:

$$\text{Min} \quad f(x) \tag{۵}$$

s.t.

$$Ax = b, \tag{۶}$$

$$Bx \le d. \tag{۷}$$

قضیه ۲-۱ [۳] $x^* \in \mathbb{R}^n$ یک جواب بهینه (۵)-(۷) است اگر و فقط اگر $u^* \in \mathbb{R}^{m+۲n}$ و $v^* \in \mathbb{R}^l$ وجود داشته باشند به‌طوری که $(x^{*T}, u^{*T}, v^{*T})^T$ در شرایط کاروش-کان-تاکر[1] (KKT) زیر صدق کند:

[1] Karush-Kuhn-Tucker

[Downloaded from jorlu.ir on 2026-11-29]

[DOR: 20.1001.1.22517286.2019.16.3.6.9]

$$\begin{cases} u^* \geq 0, \quad Bx^* - d \leq 0, \quad u^{*T}\left(Bx^* - d\right) = 0, \\ \nabla f\left(x^*\right) + B^T u^* + A^T v^* = 0, \\ Ax^* = b. \end{cases} \tag{۸}$$

x^* یک نقطه KKT مساله(۵)-(۷) نامیده می‌شود و زوج $(u^{*T}, v^{*T})^T$ بردار ضرایب لاگرانژ متناظر با x^* نامیده می‌شود.

قضیه ۲-۲ x^* یک جواب بهینه (۵)-(۷)است اگر و فقط اگر x^* یک نقطه KKT از(۵)-(۷)باشد. [۳]

۳ مدل شبکه عصبی

اگر چه انواع مختلفی از شبکه‌های عصبی برای محاسبه وجود دارند، اما تاکنون فقط دو مدل برای حل مسایل بهینه‌سازی محدب با قیود هیبریدی ارایه شده است [۴۱، ۴۹]. با عنایت به اینکه ساختار این دو مدل نیاز به l متغیر وضعیت برای برقراری قیود تساوی در شرایط بهینگی (۸) دارد، ساختار این دو مدل بیان شده پیچیده است و نیاز به ساده‌سازی این ساختارها وجود دارد؛ بنابراین هدف ما حذف این l متغیر وضعیت و ساختن یک مدل شبکه عصبی با ساختاری ساده‌تر، پایداری تضمین شده و همگرایی سراسری است. برای ساده‌سازی ساختار شبکه ابتدا لم زیر را در نظر می‌گیریم.

لم ۳-۱ x^* جواب بهینه‌ی (۵)-(۷) است اگر و تنها اگر $u^* \geq 0$ موجود باشد به‌طوری که $(x^{*T}, u^{*T})^T$ در شرایط زیر صدق کند:

$$\left(I - P\right)\left[\nabla f\left(x^*\right) + B^T u^*\right] + Q\left(Ax^* - b\right) = 0, \tag{۹}$$

$$(u^* + Bx^* - d)^+ - u^* = 0. \tag{۱۰}$$

برهان: اگر x^* جواب بهینه‌ی (۵)-(۷) باشد، مطابق شرایط بهینگی (۸) برای مسایل بهینه‌سازی محدب، $(x^{*T}, u^{*T}, v^{*T})^T$، $u^* \geq 0$ موجودند به‌طوری که در دستگاه زیر صدق کنند:

$$\nabla f\left(x^*\right) + A^T v^* + B^T u^* = 0, \tag{۱۱}$$

$$Ax^* = b, \tag{۱۲}$$

$$d - Bx^* \geq 0, \quad u^* \geq 0, \quad (u^*)^T\left(d - Bx^*\right) = 0. \tag{۱۳}$$

به آسانی دیده می‌شود که (۱۰) و (۱۳) معادل هستند. به این معنی که:

$$d - Bx^* \geq 0, \quad u^* \geq 0, \quad (u^*)^T\left(d - Bx^*\right) = 0 \Leftrightarrow (u^* + Bx^* - d)^+ - u^* = 0.$$

در ادامه، نشان می‌دهیم که جواب‌های (۹) ، (۱۱) و (۱۲) معادلند. از (۱۱) داریم:

$$-A\left(\nabla f\left(x^*\right) + A^T v^* + B^T u^*\right) = 0.$$

با جمع کردن معادله‌ی بالا با (۱۲) داریم:

$$Ax^* - A\left(\nabla f\left(x^*\right)\right) - AA^T v^* - AB^T u^* = b.$$

بنابراین:

[Downloaded from jorlu.ir on 2026-11-29]

[DOR: 20.1001.1.22517286.2019.16.3.6.9]

$$A^T v^* = A^T (AA^T)^{-1}\left(Ax^* - b\right) - A^T (AA^T)^{-1}\left(\nabla f\left(x^*\right) + B^T u^*\right). \qquad (۱۴)$$

با جایگذاری (۱۴) در (۱۱) داریم:

$$[I - A^T\left(AA^T\right)^{-1}A]\left(\nabla f\left(x^*\right) + B^T u^*\right) + A^T (AA^T)^{-1}\left(Ax^* - b\right) = ۰. \qquad (۱۵)$$

فرض کنید $P = A^T (AA^T)^{-1}A$ و $Q = A^T (AA^T)^{-1}$. می‌توان (۱۵) را به‌صورت زیر نوشت:

$$\left(I - P\right)\left[\nabla f\left(x^*\right) + B^T u^*\right] + Q\left(Ax^* - b\right) = ۰.$$

برعکس، اگر u^*موجود باشد به‌طوری که $(x^{*T}, u^{*T})^T$در (۹) صدق کند با ضرب A در طرفین (۹) داریم:

$$A\left(I - P\right)\left[\nabla f\left(x^*\right) + B^T u^*\right] + AQ\left(Ax^* - b\right) = ۰.$$

توجه کنید که $A\left(I - P\right) = ۰$ و $AQ = I$، داریم:

$$\left(I - P\right)\left(\nabla f\left(x^*\right) + B^T u^*\right) = ۰,$$
$$Ax^* - b = ۰.$$

فرض کنید $v^* = -(AA^T)^{-1}A\left(\nabla f\left(x^*\right) + B^T u^*\right)$؛ بنابراین داریم:

$$\nabla f\left(x^*\right) + A^T v^* + B^T u^* = \nabla f\left(x^*\right) + B^T u^* - A^T (AA^T)^{-1}A\left(\nabla f\left(x^*\right) + B^T u^*\right)$$
$$= \left(I - P\right)\left(\nabla f\left(x^*\right) + B^T u^*\right) = ۰.$$

بنابراین $(x^{*T}, u^{*T}, v^{*T})^T$وجود دارد به‌طوری که (۱۱) و (۱۲) در شرایط بهینگی (۸) صدق می‌کند. پس حکم ثابت شده است.

می‌توان با استفاده از لم ۳-۱ مدل شبکه‌عصبی زیر را برای به‌دست آوردن جواب بهینه‌ی مساله‌ی (۵)-(۷) به‌صورت زیر به‌کار برد:

$$\begin{cases} \dfrac{dx}{dt} = -\left(I - P\right)[\nabla f\left(x\right) + B^T\left(u + Bx - d\right)^+\right] - Q\left(Ax - b\right), \\ \dfrac{۲du}{dt} = -u + (u + Bx - d)^+, \end{cases} \qquad (۱۶)$$

که در آن:

$$P = A^T (AA^T)^{-1}A, \qquad Q = A^T (AA^T)^{-1},$$
$$(u + Bx - d)^+ = (\left[u + Bx - d\right]_۱^+, \left[u + Bx - d\right]_۲^+, \ldots, \left[u + Bx - d\right]_{m+۲n}^+),$$
$$[u + Bx - d]_k^+ = \max\{(u + Bx - d)_k\ , ۰\}, \quad k = ۱, ۲, \ldots, m + ۲n.$$

توجه ۳-۲ با استفاده از لم ۳-۱، به آسانی می‌بینیم که x^*جواب بهینه‌ی مساله‌ی (۵)-(۷) است اگر وتنها اگر $u^* \geq ۰$وجود داشته باشد به‌طوری که $(x^{*T}, u^{*T})^T$نقطه‌ی تعادل شبکه‌ی عصبی ارایه شده در (۱۶) باشد؛ بنابراین، زمانی که شبکه‌ی عصبی به نقطه‌ی تعادل همگراست، مسیر حرکت $x(t)$ به جواب بهینه‌ی مساله‌ی (۵)-(۷) همگرا می‌باشد.

[Downloaded from jorlu.ir on 2026-11-29]

[DOR: 20.1001.1.22517286.2019.16.3.6.9]

٤ مقایسه با برخی مدل‌های موجود

برای اینکه ببینیم مدل شبکه‌ی عصبی ارایه شده در (۱۶) چه مزیت‌هایی در حل مساله بهینه‌سازی (۵)-(۷) دارد، آن را با برخی مدل‌های موجود مقایسه می‌کنیم.

با استفاده از مدل ارایه شده در [۲۳] از مدل شبکه‌ی عصبی زیر می‌توان برای حل (۵)-(۷) استفاده کرد:

$$\frac{dx}{dt} = -\Big(x - P_{\Omega}\left(x - \nabla f\left(x\right) - B^{T}u + A^{T}v\right)\Big), \tag{۱۷}$$

$$\frac{du}{dt} = -(u - \left(u + Bx - d\right)^{+}\Big), \tag{۱۸}$$

$$\frac{dv}{dt} = Ax - b, \tag{۱۹}$$

که در آن $\Omega \subset \mathbb{R}^{n}$ یک مجموعه محدب بسته و $P_{\Omega} : \mathbb{R}^{n} \to \Omega$ عملگر تصویر است [۲۴] که به‌صورت زیر تعریف می‌شود:

$$P_{\Omega} = \text{argmin}_{v \in \Omega} \| x - v \|.$$

توجه کنید که هیچ تضمینی برای پایداری مدل دینامیکی(۱۷)-(۱۹) برای زمانی که f یک تابع محدب است وجود ندارد [۲۴]؛ بنابراین دستگاه دینامیکی مطرح‌شده در (۱۷)-(۱۹) نمی‌تواند یک مساله برنامه‌ریزی خطی را حل کند. برای توضیحات بیش‌تر، مثال ۴-۱ در [۳۳] را ببینید.

نوع دیگری از مدل شبکه عصبی، مدل گرادیانی است. در مدل شبکه‌ی عصبی گرادیانی، یک مساله بهینه‌سازی مقید را می‌توان با یک مساله بهینه‌سازی نامقید تقریب زد. سپس می‌توان تابع انرژی را با استفاده از روش تابع جریمه ساخت. لازم به ذکر است که مدل شبکه‌ی عصبی گرادیانی نسبت به مدل‌هایی که با استفاده از تعریف مستقیم مشتقات تابع انرژی بیان شده‌اند دارای مزیت‌هایی است؛ اما نقطه ضعف آن در عدم تضمین همگرایی است [۵۲]، چون این مدل‌ها که بر مبنای تابع جریمه ساخته می‌شوند، نیاز به یک پارامتر قابل تنظیم[1] به‌نام پارامتر جریمه دارند. تقریب مساله مقید (۵)-(۷) با استفاده از مساله بهینه‌سازی نامقید زیر، یک نمونه از کاربرد روش تابع جریمه است:

$$\text{Min } E_{1}\left(x\right) = f\left(x\right) + \frac{\gamma}{2}\left\{\sum_{k=1}^{m+2n}[(Bx-d)_{k}^{+}]^{2} + \sum_{p=1}^{l}(Ax-b)_{p}^{2}\right\},$$

که در آن γ پارامتر جریمه است. مدل شبکه‌ی عصبی گرادیانی با روش زیر به‌دست می‌آید:

$$\frac{dx}{dt} = -\nabla E_{1}\left(x\right) = -(\nabla f\left(x\right) + \gamma\Big[B^{T}\left(Bx-d\right)^{+} + A^{T}\left(Ax-b\right)\Big]\Big). \tag{۲۰}$$

سیستم ارایه شده در (۲۰) به مدل کندی و چاو[2] معروف است که در [۱۸] بیان شده است. این شبکه‌ی عصبی به‌علت دارا بودن پارامتر جریمه‌ای متناهی، توانایی یافتن یک جواب بهینه دقیق را ندارد و اگر پارامتر جریمه‌ای خیلی بزرگ باشد، یافتن یک جواب بهینه دقیق دشوار است [۲۸]. همچنین می‌توان نشان داد که شبکه‌ی عصبی

[1] Adjustable
[2] Kennedy and Chua

[Downloaded from jorlu.ir on 2026-11-29]
[DOR: 20.1001.1.22517286.2019.16.3.6.9]

(۲۰) در بعضی مسایل برنامه‌ریزی محدب، همگرای سراسری به یک جواب بهینه دقیق نیست. برای توضیح بیش‌تر مثال ۵-۱ در [۳۰] را ببینید.

با شرط محدب اکید بودن $f(x)$ ، مدل شبکه عصبی گرادیانی زیر در [۳۱] برای حل (۵)-(۷) ارایه می‌شود:

$$\frac{dx}{dt} = -\left(\nabla f(x) + \frac{1}{2} B^T u^2 + A^T v\right), \tag{۲۱}$$

$$\frac{du}{dt} = \operatorname{diag}(u_1, ..., u_{m+2n})(Bx - d), \tag{۲۲}$$

$$\frac{dv}{dt} = Ax - b, \tag{۲۳}$$

با نقطه اولیه $(x_0^T, u_0^T, v_0^T)^T$ و $u_k(t_0) > 0 \ (k = 1, ..., m+2n)$. پایداری بر حسب لیاپانف و همگرایی سراسری این مدل با شرط محدب اکید بودن $f(x)$ تضمین می‌شود؛ بنابراین مدل نمی‌تواند مسایل برنامه‌ریزی خطی را حل کند.

می‌توانیم بین جواب مساله (۵)-(۷) و جواب یک مساله مینیمم‌سازی نامقید هم‌ارزتوسط یک تابع شایسته[1] ارتباط ایجاد کنیم.

تعریف ٤-۱ تابع $\phi : \mathbb{R}^2 \to \mathbb{R}$ یک تابع NCP نامیده می‌شود اگر در شرایط زیر صدق کند [۳۹]:

$$\phi(a,b) = 0 \Leftrightarrow a \geq 0, b \geq 0, ab = 0.$$

یک تابع NCP متداول، تابع فیشربرمیستر[2] است که قویاً نیمه هموار است و به‌صورت زیرتعریف می‌شود:

$$\phi_{FB}^{\varepsilon}(a,b) = (a+b) - \sqrt{a^2 + b^2 + \varepsilon}.$$

تعریف ٤-۲ تابع شایسته تابعی است که مینیمم سراسری آن منطبق بر جواب‌های NCPاست. [۳۹]

تابع شایسته معروف $\psi_{FB} : \mathbb{R} \times \mathbb{R} \to \mathbb{R}_+$ که به‌وسیله مجذور ϕ_{FB}^{ε} به‌دست می‌آید، به‌صورت زیر است:

$$\psi_{FB}(a,b) = \frac{1}{2} \| \phi_{FB}^{\varepsilon}(a,b) \|^2 .$$

اکنون بااستفاده از [۳۹] به راحتی درمی‌یابیم که جواب مساله‌ی (۵)-(۷) معادل نقطه تعادل شبکه عصبی زیر است:

$$\frac{d(x(t), u(t), v(t))}{dt} = -\nabla E(x(t), u(t), v(t)), \tag{۲۴}$$

که در آن:

$$E(x,u,v) = \frac{1}{2} \| \eta(x,u,v) \|_4^2,$$

$$\eta(x,u,v) = \begin{bmatrix} \nabla f(x) + A^T v + B^T u \\ b - Ax \\ \phi_{FB}^{\varepsilon}(u, d - Bx) \end{bmatrix}.$$

[1] Merit function
[2] Fisher Burmister

[Downloaded from jorlu.ir on 2026-11-29]

[DOR: 20.1001.1.22517286.2019.16.3.6.9]

پایداری و همگرایی سراسری این مدل نیز با شرط محدب اکید بودن $f(x)$ تضمین می‌شود.

در [۳۳] یک مدل شبکه عصبی کارا برای حل مساله (۵)-(۷) به‌صورت زیر بیان شده است:

$$\frac{d\left(x(t),u(t),v(t)\right)}{dt}=\phi(\left(x(t),u(t),v(t)\right), \tag{۲۵}$$

که در آن:

$$\phi(x,u,v)=\begin{pmatrix}-\left(\nabla f(x)+B^T(u+Bx-d)^+ + A^Tv\right)\\ (u+Bx-d)^+ - u\\ Ax-b\end{pmatrix}. \tag{۲۶}$$

در این مدل $x\in\mathbb{R}^n, u\in\mathbb{R}^{m+2n}$ و $v\in\mathbb{R}^l$ است؛ بنابراین شبکه عصبی (۲۵) دارای $m+3n+l$ نرون است؛ اما مدل شبکه عصبی (۱۶) دارای $m+3n$ نرون است. در حقیقت مدل (۱۶) یک اصلاح از مدل (۲۵) با تعداد نرون کم‌تر و ساختار ساده‌تر است.

۵ تحلیل پایداری

در این بخش نشان می‌دهیم که مدل شبکه‌ی عصبی (۱۶) همگرای سراسری است.

لم ۵-۱ فرض کنید $(x^{*T},u^{*T})^T\in\Omega^e$ یک نقطه‌ی تعادل شبکه‌ی عصبی (۱۶) باشد و

$$\begin{aligned}V(x,u)=&f(x)-f(x^*)+\frac{1}{2}\|(u+Bx-d)^+\|^2-\frac{1}{2}\|(u^*+Bx^*-d)^+\|^2\\&-(u-u^*)^Tu^*-(x-x^*)^T\left(\nabla f(x^*)+B^Tu^*\right)+\frac{1}{2}\|x-x^*\|^2+\frac{1}{2}\|u-u^*\|^2.\end{aligned} \tag{۲۷}$$

سپس

الف-

$$V(x,u)\geq\frac{1}{2}\|x-x^*\|^2+\frac{1}{2}\|u-u^*\|^2; \tag{۲۸}$$

ب-

$$\begin{aligned}\frac{\mathrm{d}V}{\mathrm{d}t}\leq&-\|(I-P)[\nabla f(x)+B^T\left(u+Bx-d\right)^+]+Q(Ax-b)\|^2\\&-\frac{1}{2}\|u-(u+Bx-d)^+\|^2.\end{aligned} \tag{۲۹}$$

برهان: الف- چون تابع $f(x)+\frac{1}{2}\|(u+Bx-d)^+\|^2$ محدب و مشتق‌پذیر است و

$$\nabla(f(x)+\frac{1}{2}\|\left(u+Bx-d\right)^+\|^2)=\begin{bmatrix}\nabla f(x)+B^T(u+Bx-d)^+\\(u+Bx-d)^+\end{bmatrix},$$

در نتیجه:

$$f(x)-f(x^*)+\frac{1}{2}\|(u+Bx-d)^+\|^2-\frac{1}{2}\|(u^*+Bx^*-d)^+\|^2$$

[Downloaded from jorlu.ir on 2026-11-29]

[DOR: 20.1001.1.22517286.2019.16.3.6.9]

$$\geq (x-x^*)^T\left(\nabla f\left(x^*\right)+B^Tu^*\right)+(u-u^*)^Tu^*.$$

بنابراین:

$$V(x,u)\geq \frac{1}{2}\| x-x^*\|^2+\frac{1}{2}\| u-u^*\|^2 .$$

ب- برای اثبات (۲۹) داریم:

$$\begin{aligned}
\frac{dV}{dt}&=\frac{\partial V}{\partial x}\frac{dx}{dt}+\frac{\partial V}{\partial u}\frac{\mathrm{d}u}{\mathrm{d}t}\\
&=-[\nabla f\left(x\right)+B^T(u^*+Bx^*-d)^+-\nabla f\left(x^*\right)-B^Tu^*+x-x^*]^T\\
&\quad\times\{(I-P)[\nabla f\left(x\right)+B^T\left(u+Bx-d\right)^+]+Q(Ax-b)\}\\
&\quad-\frac{1}{2}[u-(u+Bx-d)^+]^T[\left(u+Bx-d\right)^+-2u^*+u]\\
&=-[\nabla f\left(x\right)-\nabla f\left(x^*\right)+x-x^*+B^T(u+Bx-d)^+-B^Tu^*]^T\\
&\quad\times\left\{(I-P)\left[\nabla f\left(x\right)-\nabla f\left(x^*\right)+B^T\left(u+Bx-d\right)^+-B^Tu^*\right)\right]+P\left(x-x^*\right)\right\}\\
&\quad-\frac{1}{2}[u-(u+Bx-d)^+]^T[u-(u+Bx-d)^++2\left(u+Bx-d\right)^+-2u^*]\\
&=-\{\left[\nabla f\left(x\right)-\nabla f\left(x^*\right)\right]^T(I-P)\left[\nabla f\left(x\right)-\nabla f\left(x^*\right)\right]\\
&\quad+\left[\nabla f\left(x\right)-\nabla f\left(x^*\right)\right]^T(I-P)\left[B^T\left(u+Bx-d\right)^+-B^Tu^*\right]\\
&\quad+[\nabla f\left(x\right)-\nabla f\left(x^*\right)]^TP\left(x-x^*\right)\\
&\quad+(x-x^*)^T(I-P)\left[\nabla f\left(x\right)-\nabla f\left(x^*\right)\right]\\
&\quad+(x-x^*)^T(I-P)[B^T\left(u+Bx-d\right)^+-B^Tu^*]+(x-x^*)^TP\left(x-x^*\right)\\
&\quad+[B^T(u+Bx-d)^+-B^Tu^*]^T(I-P)\left[\nabla f\left(x\right)-\nabla f\left(x^*\right)\right]\\
&\quad+[B^T(u+Bx-d)^+-B^Tu^*]^T(I-P)[B^T\left(u+Bx-d\right)^+-B^Tu^*]\\
&\quad+[B^T(u+Bx-d)^+-B^Tu^*]^TP\left(x-x^*\right)\}\\
&\quad-\frac{1}{2}\| u-(u+Bx-d)^+\|^2+[(u+Bx-d)^+-u]^T[\left(u+Bx-d\right)^+-u^*].
\end{aligned}$$

توجه کنید که:

$$(I-P)^2=I-P, P^2=P, P\left(I-P\right)=0,$$

و

$$(u+Bx-d)^+-u=Bx-d+(-u-\left(Bx-d\right))^+.$$

بنابراین داریم:

$$\begin{aligned}
\frac{dV}{dt}&=-[\nabla f\left(x\right)-\nabla f\left(x^*\right)]^T(I-P)^2\left[\nabla f\left(x\right)-\nabla f\left(x^*\right)\right]\\
&\quad-2\left[\nabla f\left(x\right)-\nabla f\left(x^*\right)\right](I-P)^2[B^T\left(u+Bx-d\right)^+-B^Tu^*]
\end{aligned}$$

[Downloaded from jorlu.ir on 2026-11-29]

[DOR: 20.1001.1.22517286.2019.16.3.6.9]

$$
\begin{aligned}
&-[B^T(u+Bx-d)^+ - B^T u^*]^T (I-P)^2 [B^T\left(u+Bx-d\right)^+ - B^T u^*\right] \\
&-(x-x^*)^T P^2\left(x-x^*\right)-(x-x^*)^T\left[\nabla f\left(x\right)-\nabla f\left(x^*\right)\right] \\
&-(x-x^*)^T [B^T\left(u+Bx-d\right)^+ - B^T u^*\right] \\
&-\frac{1}{2}\| u-(u+Bx-d)^+ \|^2 +[Bx-d+(-u-Bx+d)^+]^T[\left(u+Bx-d\right)^+ - u^*\right]
\end{aligned}
$$

$$
= -\|\left(I-P\right)[\nabla f\left(x\right)-\nabla f\left(x^*\right)]+\left(I-P\right)\left[B^T\left(u+Bx-d\right)^+ - B^T u^*\right]+P\left(x-x^*\right)\|^2
$$

$$
\begin{aligned}
&-(x-x^*)^T\left[\nabla f\left(x\right)-\nabla f\left(x^*\right)\right]-(x-x^*)^T[B^T\left(u+Bx-d\right)^+ - B^T u^*\right] \\
&-\frac{1}{2}\| u-(u+Bx-d)^+ \|^2 +(Bx-d)^T(u+Bx-d)^+ -(Bx-d)^T u^* \\
&+[(-u-\left(Bx-d\right))^+]^T(u+Bx-d)^+ -[(-u-\left(Bx-d\right))^+]^T u^* \\
&= -\|\left(I-P\right)[\nabla f\left(x\right)+B^T\left(u+Bx-d\right)^+\right]+Q\left(Ax-b\right)\|^2 \\
&-(x-x^*)^T\left[\nabla f\left(x\right)-\nabla f\left(x^*\right)\right]+(x-x^*)^T B^T u^* -(x-x^*)^T B^T(u+Bx-d)^+ \\
&-\frac{1}{2}\| u-(u+Bx-d)^+ \|^2 +(Bx-d)^T[\left(u+Bx-d\right)^+\right]-(Bx-d)^T u^* \\
&+[(-u-\left(Bx-d\right))^+]^T(u+Bx-d)^+ -[(-u-\left(Bx-d\right))^+]^T u^* \\
&= -\|\left(I-P\right)[\nabla f\left(x\right)+B^T\left(u+Bx-d\right)^+\right]+Q\left(Ax-b\right)\|^2 \\
&-\frac{1}{2}\| u-(u+Bx-d)^+ \|^2 -(x-x^*)^T[\nabla f\left(x\right)-\nabla f\left(x^*\right)]-\left[(u+Bx-d)^+\right]^T u^* \\
&-(Bx^*-d)^T u^* +(Bx^*-d)^T(u+Bx-d)^+ .
\end{aligned}
$$

چون تابع $f\left(x\right)$ محدب است؛ بنابراین داریم:

$$(x-x^*)^T\left[\nabla f\left(x\right)-\nabla f\left(x^*\right)\right]\geq 0 .$$

به‌علاوه به آسانی می‌توان تحقیق کرد که:

$$
\begin{aligned}
[(-u-\left(Bx-d\right))^+]^T(u+Bx-d)^+ &= 0, \\
(Bx^*-d)^T u^* &= 0, \\
[(-u-\left(Bx-d\right))^+]^T u^* &\geq 0, \\
g(x^*)^T[(u+Bx-d)^+]^T &\leq 0 .
\end{aligned}
$$

بنابراین:

$$\frac{dV}{dt}\leq -\|\left(I-P\right)[\nabla f\left(x\right)+B^T\left(u+Bx-d\right)^+\right]+Q\left(Ax-b\right)\|^2 -\frac{1}{2}\| u-(u+Bx-d)^+ \|^2 \leq 0 .$$

توجه ۵-۲ $V\left(x,y\right)$ معرفی شده در (۲۷) یک تابع لیاپانوف است، بنابراین، شبکه‌ی عصبی (۱۶) پایدار به مفهوم لیاپانوف است.

[Downloaded from jorlu.ir on 2026-11-29]

[DOR: 20.1001.1.22517286.2019.16.3.6.9]

قضیه ۵-۳ برای هر نقطه‌ی اولیه‌ی $(x(0)^T,u(0)^T)^T\in\mathbb{R}^{m+3n}$، یک جواب پیوسته منحصربه‌فرد $(x(t)^T,u(t)^T)^T\in\mathbb{R}^{m+3n}$ از (۱۶) برای $t\geq 0$ وجود دارد.

برهان: چون $f(x)$ روی $\mathbb{R}^n$ به‌طور پیوسته مشتق‌پذیر است، $\nabla f(x)$ پیوسته لیپ شیتز محلی است. درنتیجه $(I-P)[\nabla f(x)+B^T\left(u+Bx-d\right)^+]+Q(Ax-b)$ و $(u+Bx-d)^+-u$ پیوسته لیپ شیتز محلی هستند. از قضیه‌ی وجودی جواب موضعی برای معادلات دیفرانسیل معمولی [۵۵] نتیجه می‌شود یک جواب منحصر به‌فرد از (۱۶) روی $[t_0,T)$وجود دارد. بنا به لم ۵-۱می‌دانیم که V نسبت به t یک تابع نزولی است؛ لذا:

$$\frac{1}{2}\|x-x^*\|^2+\frac{1}{2}\|u-u^*\|^2\leq V\left(x(0),u(0)\right)\leq V\left(x(t),u(t)\right),\quad \forall t\geq 0, \tag{۳۰}$$

نشان می‌دهد مسیر حالت شبکه‌ی عصبی $(x(t)^T,u(t)^T)^T$ از(۱۶) روی $[t_0,T)$ کراندار است. درنتیجه $T=+\infty$. این اثبات را کامل می‌کند.

قضیه ۵-٤ مسیر حالت شبکه‌ی عصبی (۱۶) برای هر نقطه‌ی اولیه‌ی $(x(0)^T,u(0)^T)^T\in\mathbb{R}^{m+3n}$ به یک نقطه‌ی تعادل همگرا است. به‌ویژه، شبکه‌ی عصبی(۱۶) با هر نقطه‌ی اولیه‌ی $(x(0)^T,u(0)^T)^T\in\mathbb{R}^{m+3n}$ هنگامی که Ω^e نقطه‌ی تعادل منحصر به‌فرد دارد پایدار مجانبی سراسری است.

برهان: تعریف کنید:

$$D(x,u)=\|(I-P)[\nabla f(x)+B^T\left(u+Bx-d\right)^+]+Q(Ax-b)\|^2$$
$$+\frac{1}{2}\|u-(u+Bx-d)^+\|^2.$$

در این صورت $D(x,u)=0$ اگر و تنها اگر $(x^T,u^T)^T$ یک نقطه‌ی تعادل شبکه‌ی عصبی (۱۶) باشد. از اثبات قضیه‌ی ۵-۳ نتیجه می‌گیریم مسیر حالت $(x(t)^T,u(t)^T)^T$ شبکه‌ی عصبی (۱۶) کراندار است؛ بنابراین، دنباله‌ی صعودی $\{t_n\}$ با $\lim_{n\to\infty}t_n\to\infty$ و نقطه‌ی حدی $(\hat{x}^T,\hat{u}^T)^T$ وجود دارند به‌طوری که $\lim_{n\to\infty}x(t_n)\to\hat{x}$ و $\lim_{n\to\infty}y(t_n)\to\hat{u}$. حال ثابت می‌کنیم $(\hat{x}^T,\hat{u}^T)\in\Omega^e$ و مسیر حالت $(x(t)^T,u(t)^T)^T$ همگرای سراسری به نقطه‌ی تعادل $(\hat{x}^T,\hat{u}^T)^T$ است. در مرحله اول، ثابت می‌کنیم که $D(\hat{x},\hat{u})=0$؛ یعنی $(\hat{x}^T,\hat{u}^T)^T$ یک نقطه‌ی تعادل شبکه‌ی عصبی (۱۶) است. فرض کنید چنین نباشد (برهان خلف)؛ یعنی $D(\hat{x},\hat{u})>0$ چون $D(x,u)$ به ترتیب نسبت به x و u پیوسته است، $\varepsilon>0$، $q>0$ و $\varepsilon-$همسایگی:

$$B\left((\hat{x},\hat{u}),\varepsilon\right)=\{(x^T,u^T)^T\ :\ \|x-\hat{x}\|+\|u-\hat{u}\|\leq\varepsilon\},$$

از $(\hat{x}^T,\hat{u}^T)^T$ وجود دارند به‌طوری که برای هر $(x^T,u^T)^T\in B\left((\hat{x},\hat{u}),\varepsilon\right)$، $D(x,u)>q$. از طرفی با توجه به اینکه $\lim_{n\to\infty}x(t_n)\to\hat{x}$ و $\lim_{n\to\infty}u(t_n)\to\hat{u}$، عدد صحیح مثبت N وجود دارد به طوری که برای هر $n\geq N$، $\|x(t_n)-\hat{x}\|\leq\frac{1}{4}\varepsilon$ و $\|u(t_n)-\hat{u}\|\leq\frac{1}{4}\varepsilon$. از (۱۶) و کراندار بودن $(x(t)^T,u(t)^T)^T$ نتیجه می‌گیریم $\dot{x}$ و $\dot{u}$نیز کراندار هستند که این کران با M نشان داده می‌شود. با در نظر گرفتن:

[Downloaded from jorlu.ir on 2026-11-29]

[DOR: 20.1001.1.22517286.2019.16.3.6.9]

$$t \in \left[t_n - \frac{\varepsilon}{8M}, t_n + \frac{\varepsilon}{8M} \right],\ n \geq N,$$

داریم:

$$\begin{aligned} &\| x(t) - \hat{x} \| + \| u(t) - \hat{u} \| \\ &\quad \leq \| x(t) - x(t_n) \| + \| u(t) - u(t_n) \| + \| x(t_n) - \hat{x} \| + \| u(t_n) - \hat{u} \| \\ &\quad = \| \dot{x}(\xi_1) \| \times |t - t_n| + \| \dot{u}(\xi_2) \| \times |t - t_n| + \| x(t_n) - \hat{x} \| + \| u(t_n) - \hat{u} \| \\ &\quad \leq 2M |t - t_n| + \frac{\varepsilon}{2} \leq \varepsilon. \end{aligned}$$

از این رو، برای هر $t \in \left[t_n - \frac{\varepsilon}{8M}, t_n + \frac{\varepsilon}{8M} \right]$، $n \geq N$ داریم $(x^T, u^T)^T \in B\left((\hat{x}, \hat{u}), \varepsilon\right)$؛ بنابراین برای هر $t \in \left[t_n - \frac{\varepsilon}{8M}, t_n + \frac{\varepsilon}{8M} \right]$، $n \geq N$ نتیجه می‌شود $D\left(x(t), u(t)\right) > q$. چون $\lim_{n \to \infty} t_n = \infty$ و اندازه‌ی لبگ[1] مجموعه‌ی $t \in \cup_{n \geq N} \left[t_n - \frac{\varepsilon}{8M}, t_n + \frac{\varepsilon}{8M} \right]$ بی‌نهایت است، پس داریم:

$$\int_0^\infty D\left(x(t)\ u(t)\right)\ dt = \infty. \qquad (31)$$

از طرفی:

$$\begin{aligned} \int_0^\infty D\left(x(t)\ u(t)\right)\ dt &= \lim_{s \to \infty} \int_0^s D\left(x(t)\ u(t)\right)\ dt \leq -\lim_{s \to \infty} \int_0^s \dot{V}\left(x(t)\ u(t)\right)\ dt \\ &= \lim_{s \to \infty} \left[V\left(x(0), u(0)\right) - V\left(x(s), u(s)\right) \right] \leq V\left(x(0), y(0)\right), \end{aligned}$$

که (۳۱) را نقض می‌کند؛ لذا $D(\hat{x}, \hat{u}) = 0$؛ یعنی $(\hat{x}^T, \hat{u}^T)^T \in \Omega^e$. در مرحله دوم، ثابت می‌کنیم مسیر حالت $(x(t)^T, u(t)^T)^T$ همگرای سراسری به نقطه‌ی تعادل $(\hat{x}^T, \hat{u}^T)^T$ است. تابع لیاپانوف زیر را تعریف کنید:

$$\begin{aligned} \hat{V}(x,u) &= f(x) - f(\hat{x}) + \frac{1}{2} \| (u + Bx - d)^+ \|^2 - \frac{1}{2} \| (\hat{u} + B\hat{x} - d)^+ \|^2 - (u - \hat{u})^T \hat{u} \\ &\quad - (x - \hat{x})^T \left(\nabla f(\hat{x}) + B^T \hat{u} \right) + \frac{1}{2} \| x - \hat{x} \|^2 + \frac{1}{2} \| u - \hat{u} \|^2 . \end{aligned}$$

داریم $\hat{V}(\hat{x}, \hat{u}) = 0$. چون $\lim_{n \to \infty} x(t_n) \to \hat{x}$ و $\lim_{n \to \infty} u(t_n) \to \hat{u}$، به ازای هر $\varepsilon > 0$، t_k وجود دارد به طوری که $\hat{V}\left(x(t_k), u(t_k)\right) < \varepsilon$. بنا به لم ۵-۱ داریم:

$$\frac{1}{2} \| x(t) - \hat{x} \|^2 + \frac{1}{2} \| u(t) - \hat{u} \|^2 \leq \hat{V}\left(x(t), u(t)\right),$$

و $\hat{V}$ نزولی است؛ بنابراین، برای هر $t \geq t_k$ داریم:

$$\frac{1}{2} \| x(t) - \hat{x} \|^2 + \frac{1}{2} \| u(t) - \hat{u} \|^2 \leq \hat{V}\left(x(t), u(t)\right) \leq \hat{V}\left(x(t_k), u(t_k)\right) \leq \varepsilon,$$

[1] Lebesgue measure

[Downloaded from jorlu.ir on 2026-11-29]

[DOR: 20.1001.1.22517286.2019.16.3.6.9]

یعنی $\lim_{n\to\infty}x(t)=\hat{x}$ و $\lim_{n\to\infty}u(t)=\hat{y}$؛ لذا مسیر حالت شبکه‌ی عصبی (۱۶) همگرای سراسری به نقطه‌ی تعادل $(\hat{x}^T,\hat{u}^T)^T$ است. به‌ویژه، اگر $\Omega^e=\{(x^{*T},u^{*T})^T\}$، آنگاه بنا به تحلیل فوق مسیر حالت $(x(t)^T,u(t)^T)^T$ با هر نقطه‌ی اولیه‌ی $(x(\circ)^T,u(\circ)^T)^T$ به $(x^{*T},u^{*T})^T$ میل خواهد کرد. پس این شبکه‌ی عصبی پایدار مجانبی سراسری است.

٦ شبیه‌سازی‌های عددی

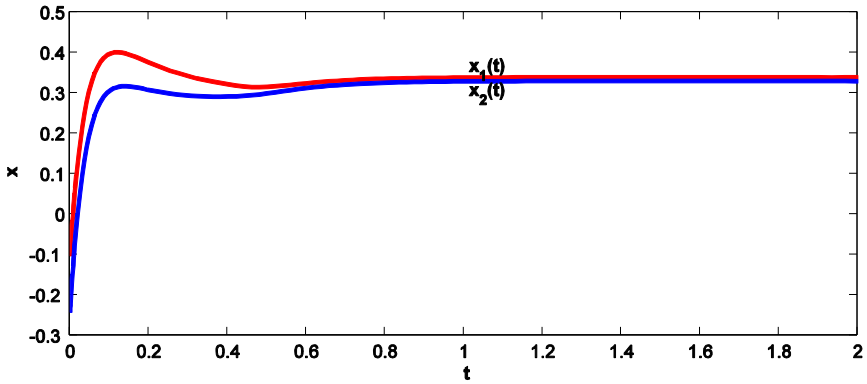
برای اثبات کارایی و موثر بودن شبکه عصبی پیشنهادی، در این بخش، چندین مثال را توسط شبکه عصبی (۱۶) آزمایش می‌کنیم. شبیه‌سازی توسط نرم افزار مطلب Matlab 7 اجرا می‌شود. حل کننده معادله دیفرانسیل معمولی ode45 است. دقت جواب مثال‌های زیر توسط نرم‌افزار لینگو Lingo 11 احراز شده است.

مثال ٦-١

$$\text{Min } f(x)=\circ/۴x_۲+x_۱^۲+x_۲^۲-x_۱x_۲+\frac{۱}{۳\circ}x_۱^۳$$

$$\text{s.t.}\begin{cases}x_۱\geq\circ,\\ x_۲\geq\circ,\\ x_۱+\circ/۵x_۲\geq\circ/۴,\\ \circ/۵x_۱+x_۲\geq\circ/۵.\end{cases}$$

جواب بهینه $x^*=(\circ/۳۳۹۵,\circ/۳۳۰۳)^T$ است. شبکه عصبی (۱۶) برای حل این مساله برنامه ریزی ریاضی به‌کار برده می‌شود. نتایج شبیه‌سازی نشان می‌دهد که مسیرهای شبکه عصبی (۱۶) همگرا به $y^*=(x^{*T},u^{*T})^T$ هستند. شکل ۱ نشان می‌دهد که مسیرها همگرا به جواب بهینه x^* هستند.



شکل ۱. رفتارهای گذرای $X_۱$ و $X_۲$ از شبکه عصبی (۱۶) در مثال (۶-۱).

[Downloaded from jorlu.ir on 2026-11-29]

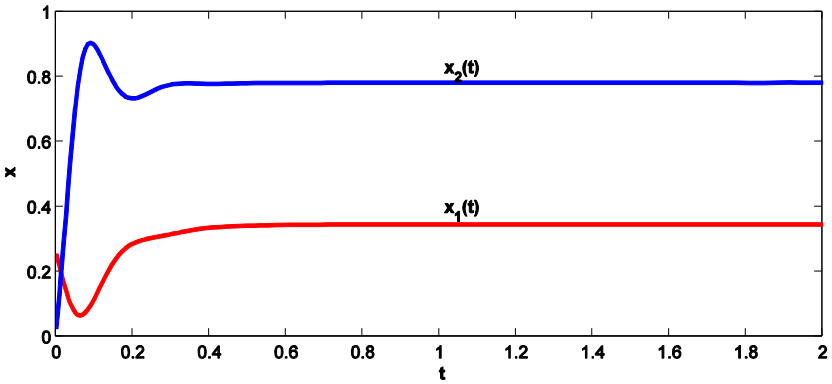
[DOR: 20.1001.1.22517286.2019.16.3.6.9]

مثال ٦-٢

$$\text{Min} \quad f(x)=\frac{۱}{۴}x_۱^۴+۰/۵x_۱^۲+\frac{۱}{۴}x_۲^۴+۰/۵x_۲^۲-۰/۹x_۱x_۲$$

$$\text{s.t.}\begin{cases}-x_۱-x_۲\geq -۲,\\ x_۱-x_۲\geq -۲,\\ x_۱-۳x_۲=-۲,\\ ۰\leq x\leq ۱.\end{cases}$$

نتایج شبیه‌سازی نشان می‌دهد که مسیرهای شبکه عصبی (۱۶) همگرا به جواب بهینه $x^*=(۰/۳۴۶۰,۰/۷۸۲۲)^T$ است. به‌عنوان مثال، شکل ۲ نشان می‌دهد که مسیرهای شبکه عصبی همگرا به جواب بهینه مساله هستند.



شکل ۲. رفتارهای گذرای $X_۱$ و $X_۲$ از شبکه عصبی (۱۶) در مثال (۶-۲).

مثال ٦-٣

$$\text{Min} \quad f(x)=\frac{۳}{۲}\left(x_۱^۲+x_۲^۲\right)+۲\left(x_۳^۲+x_۴^۲\right)-\ln\left(x_۱x_۴\right)+۳x_۱x_۲+۴x_۳x_۴-۲x_۱-۳x_۴$$

$$\text{s.t.}\begin{cases}x_۱+x_۲=۱,\\ x_۳+x_۴=۱,\\ x\in Q,\end{cases}$$

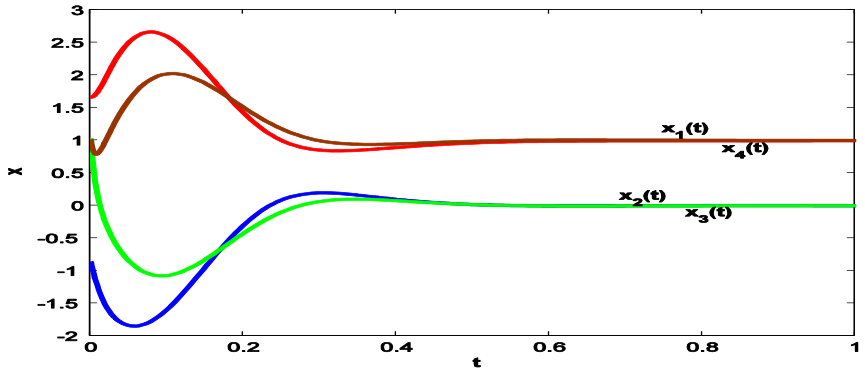
که در آن:

$$Q=\{x\in\mathbb{R}^۴\mid ۰\leq x_i\leq ۱۰ \quad i=۱,۲,۳,۴\}.$$

این مساله فقط یک جواب $x^*=(۱,۰,۰,۱)^T$ دارد. شکل ۳ مسیرهای وضعیت $(x_۱(t),x_۲(t),x_۳(t),x_۴(t))^T$ را نشان می‌دهد. این بردار به x^* همگرا است.

[Downloaded from jorlu.ir on 2026-11-29]

[DOR: 20.1001.1.22517286.2019.16.3.6.9]



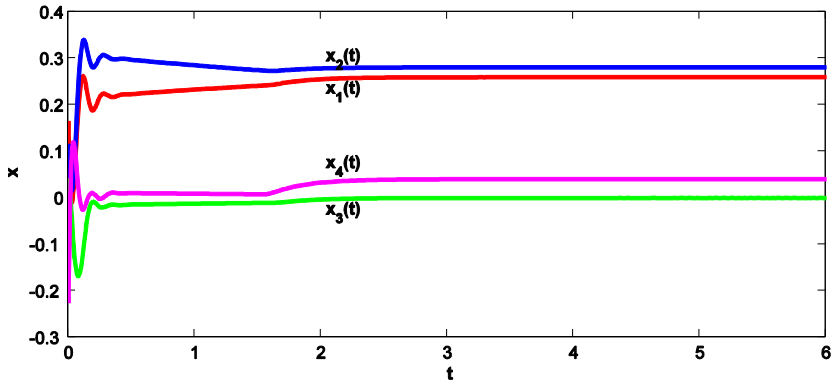
شکل ۳. رفتارهای گذرای X_1 ، X_2 ، X_3 و X_4 از شبکه عصبی (۱۶) در مثال (۶-۳).

مثال ٦-٤

$$\text{Min} \quad f(x)=۰/۴x_۱+x_۱^۲+x_۲^۲-x_۱x_۲+۰/۵x_۳^۲+۰/۵x_۴^۲+\frac{x_۱^۳}{۳۰}$$

$$\text{s.t.}\begin{cases} x_۱+x_۲-x_۴=۰/۵, \\ x_۱+۰/۵x_۲-x_۳=۰/۴, \\ x\geq ۰. \end{cases}$$

جواب بهینه $x^*=(۰/۲۵۹۵,۰/۲۸۰۹,۰,۰/۰۴۰۴)^T$ است. شکل ۴ نشان می‌دهد که مسیرها همگرا به جواب بهینه هستند. نتایج شبیه‌سازی نشان می‌دهند که نقطه شروع خارج ناحیه شدنی است؛ اما مسیرهای شبکه عصبی همگرا به جواب بهینه هستند.



شکل ٤. رفتارهای گذرای X_1 ، X_2 ، X_3 و X_4 از شبکه عصبی (۱۶) در مثال (۶-۴).

[Downloaded from jorlu.ir on 2026-11-29]

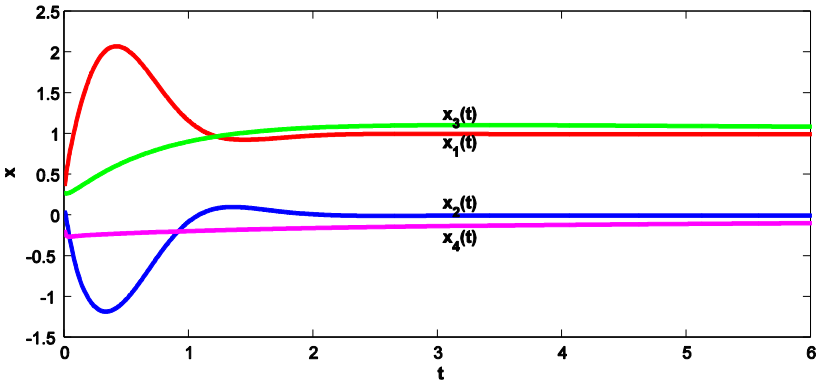
[DOR: 20.1001.1.22517286.2019.16.3.6.9]

مثال ٦-٥

$$\text{Min} \quad f(x)=\frac{۳}{۲}\left(x_۱^۲+x_۲^۲\right)+۲\left(x_۳^۲+x_۴^۲\right)-\ln\left(x_۱x_۴\right)+۳x_۱x_۲+۴x_۳x_۴-۲x_۱-۳x_۴$$

$$\text{s.t.}\begin{cases} x_۱+x_۲=۱, \\ x_۳+x_۴=۱, \\ ۰/۱\le x_۱\le ۱۰, \quad ۰\le x_۲\le ۱۰, \\ ۰\le x_۳\le ۱۰, \quad ۰/۱\le x_۴\le ۱۰. \end{cases}$$

شبیه‌سازی عددی نشان می‌دهد که مسیرهای خروجی از شبکه عصبی (۱۶) همگرا به جواب بهینه $x^*=(۱,۰,۰,۱)^T$ هستند. به‌عنوان مثال شکل ۵ همگرایی را نشان می‌دهد.



شکل ۵. رفتارهای گذرای $x_۱$، $x_۲$، $x_۳$ و $x_۴$ از شبکه عصبی (۱۶) در مثال (۶-۵).

مثال ٦-٦

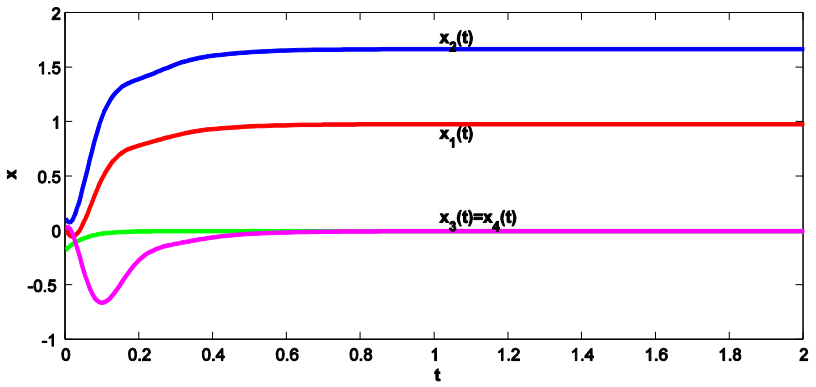
$$\text{Min} \quad f(x)=۰/۴x_۱+x_۱^۲+x_۲^۲-x_۱x_۲+۰/۵x_۳^۲+۰/۵x_۴^۲+\frac{۱}{۳۰}x_۱^۳$$

$$\text{s.t.}\begin{cases} -x_۱+x_۲-x_۳\le ۲, \\ ۳x_۱+x_۲-x_۳-x_۴\le ۱۸, \\ \frac{۱}{۳}x_۱+x_۲-x_۴=۲, \\ x_۱,x_۲,x_۳,x_۴\ge ۰. \end{cases}$$

جواب بهینه برای این مساله $x^*=(۰/۹۸۲۰,۱/۶۷۲۷,۰,۰)^T$ است. همگرایی مسیرهای گذرای $x(t)$ بر مبنای شبکه عصبی (۱۶) در شکل ۶ نشان داده می‌شود.

[Downloaded from jorlu.ir on 2026-11-29]

[DOR: 20.1001.1.22517286.2019.16.3.6.9]



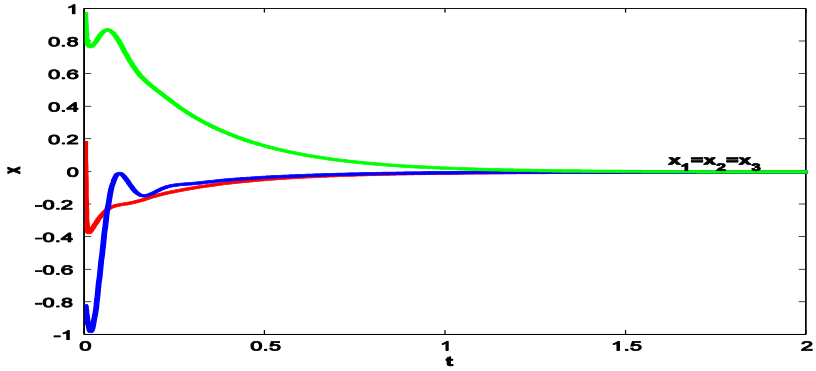
شکل ٦. رفتارهای گذرای x_1 ، x_2 ، x_3 و x_4 از شبکه عصبی (۱۶) در مثال (۶-۶).

مثال ٦-۷

$$\text{Min} \quad f(x)=11x_1^2+x_2^2+x_3^2+2x_1x_2+6x_1x_3$$

$$\text{s.t.}\begin{cases}2x_1+2x_2+x_3=0,\\ -x_1+x_2\le 1,\\ 3x_1+x_3\le 4,\\ 0\le x_i\le 6.\end{cases}$$

جواب بهینه $x^*=(0,0,0)^T$ است. همگرایی مسیرها به x^* در شکل ۷ نشان داده می‌شود.



شکل ۷. رفتارهای گذرای x_1 ، x_2 و x_3 از شبکه عصبی (۱۶) در مثال (۶-۷).

مثال ٦-۸

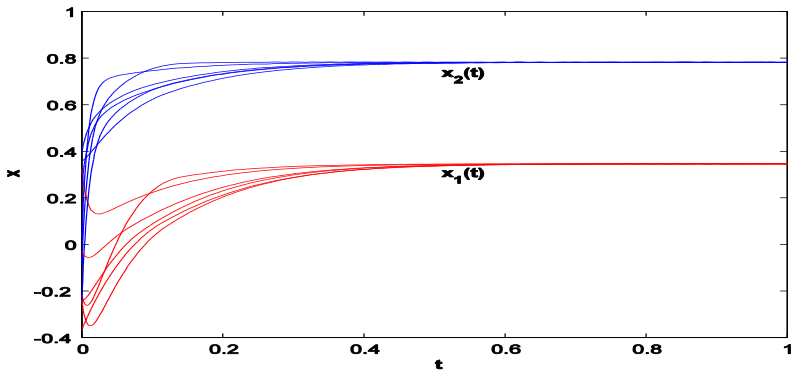
$$\text{Min} \quad f(x)=\frac{1}{4}x_1^4+\frac{1}{2}x_1^2+\frac{1}{4}x_2^4+\frac{1}{2}x_2^2-0/9x_1x_2$$

$$\text{s.t.}\begin{cases}x_1+x_2\le 2,\\ -x_1+x_2\le 2,\\ x_1-3x_2\le -2,\\ x_1,x_2\ge 0.\end{cases}$$

[Downloaded from jorlu.ir on 2026-11-29]

[DOR: 20.1001.1.22517286.2019.16.3.6.9]

قضیه ۴-۵ همگرایی سراسری شبکه عصبی پیشنهادی را به جواب بهینه $x^* = (۰/۳۴۶۲, ۰/۷۸۱۴)^T$ تضمین می‌کند. شکل ۸ همگرایی مسیرهای شبکه عصبی (۱۶) را با ۶ نقطه شروع متفاوت نشان می‌دهد.



شکل ۸. رفتارهای گذرای X_1 و X_2 از شبکه عصبی (۱۶) در مثال (۶-۸).

۷ نتیجه‌گیری

در این مقاله یک شبکه عصبی جدید برای حل مساله برنامه‌ریزی غیر خطی با محدودیت‌های هیبریدی ارایه شده است. شبکه عصبی پیشنهاد شده به هیچ پارامتر جریمه‌ای نیاز ندارد و ساختار آن نیز ساده است؛ بنابراین مدل پیشنهادی برای اجرا در سخت افزار بسیار مناسب است. به‌علاوه، نشان دادیم که مدل شبکه عصبی ارایه شده پایدار به مفهوم لیاپانوف بوده و به جواب دقیق مساله اصلی همگرا می‌شود. از طرفی، به دلیل کاربرد وسیع مساله برنامه‌ریزی درجه دوم، مدل پیشنهادی و نتایج به‌دست آمده در این مقاله، هم از نظر تئوری و هم از نظر کاربرد بسیار مهم و قابل توجه می‌باشند. با ارایه چندین مثال نشان دادیم که مدل ارایه شده همگرای سراسری بوده و هیچ وابستگی به نقطه آغازین ندارد. نتایج شبیه‌سازی عددی نیز به روشنی همگرایی و خواص مدل پیشنهادی را نشان می‌دهند.

منابع

[۸] ناصری،س.، طالشیان جلودار،ف.، تقی نژاد،ن.،خلیلی، ف.، (۱۳۹۱). مساله برنامه ریزی درجه دوم با ضرایب فازی: یک روش حل مبتنی بر اصل گسترش.مجله تحقیق در عملیاتدر کاربردهای آن، ۹(۳۵)، ۹-۲۵.

[۵۳] اعتمادی، م.، (۱۳۸۶). حل مساله تخصیص تعمیم یافته به روش شبکه عصبی.مجله تحقیق در عملیات در کاربردهای آن، ۴(۱۳)، ۱-۱۰.

[1] Agrawal, S. K., Fabien, B. C. (1999). Optimization of Dynamic Systems, Kluwer Academic Publishers, Netherlands.

[2] Avriel, M. (1976). Nonlinear Programming: Analysis and Methods, Englewood Cliffs, NJ: Prentice-Hall.

[3] Bazaraa, M. S., Sherali, H. D., Shetty, C. M. (1993). Nonlinear Programming- Theory and Algorithms, 2nd ed. New York: Wiley.

[4] Bertsekas, D. P. (1989). Parallel and Distributed Computation: Numerical Methods, Prentice-Hall, Englewood Cliffs, NJ.

[5] Cornuejols, G., Tutuncu, R. (2006). Optimization Methods in Finance, Cambridge University Press.

[6] Boyd, S., Vandenberghe, L. (2004). Convex Optimization, Cambridge University Press, Cambridge.

[Downloaded from jorlu.ir on 2026-11-29]

[DOR: 20.1001.1.22517286.2019.16.3.6.9]

[7] Fletcher, R. (1981). Practical Methods of Optimization, New York: Wiley.
[9] Yoshikawa, T., (1990). Foundations of robotics: Analysis and control. Cambridge, MA: MIT Press.
[10] Kalouptisidis, N., (1997). Signal processing systems, theory and design. New York: Wiley.
[11] Harker, P. T., Pang, J. S. (1990). Finite-dimensional variational inequality and nonlinear complementarity problems: A survey of theory, algorithms, and applications, Math. Progr. ser. B 48, 161-220.
[12] He, B. S., Liao, L. Z. (2002). Improvements of some projection methods for monotone nonlinear variational inequalities, Journal of Optimization Theory and Applications, 112, 111-128.
[13] Solodov, M. V., Tseng, P. (1996). A Modified projection-type methods for monotone variational inequalities, SIAM Journal on Control Optimization, 34,1814-830.
[14] Tseng, P. (2000). A modified forward-backward splitting method for maximal monotone mappings, SIAM Journal on Control Optimization, 38, 431-446.
[15] Cichocki, A., Unbehauer, R. (1993). Neural networks for optimization with bounded constraints, IEEE Transactions on Neural Networks, 4, 293-304.
[16] McCulloch, W., Pitts, W. (1943). A logical calculus of the ideas immanent in nervous activity. Bull. Math. Biophys., 5, 115–133.
[17] Tank, D. W., Hopfield, J. J. (1986). Simple neural optimization networks: An A/D converter, signal decision circuit, and a linear programming pircuit, IEEE Transactions on Circuits and Systems, 33, 533-541.
[18] Kennedy, M. P., Chua, L. O. (1988). Neural networks for nonlinear programming, IEEE Transactions on Circuits and Systems, 35, 554-562.
[19] Cichocki, A., Unbehauer, R. (1993). Neural networks for optimization with bounded constraints, IEEE Transactions on Neural Network, 4, 293-304.
[20] Effati, S., Nazemi, A. R. (2006). Neural network models and its application for solving linear and quadratic programming problems, Applied Mathematics and Computation, 172, 305-331.
[21] Effati, S., Ghomashi, A., Nazemi, A.R. (2007). Application of projection neural network in solving convex programming problems, Applied Mathematics and Computation, 188, 1103-1114.
[22] Forti, M., Nistri, P., Quincampoix, M. (2006). Convergence of neural networks for programming problems via a nonsmoothLojasiewicz inequality, IEEE Transactions on Neural Networks, 17, 1471-1486.
[23] Friesz, T. L., Bernstein, D. H., Mehta, N. J., Tobin, R. L., Ganjlizadeh, S. (1994). Day-to-day dynamic network disequilibria and idealized traveler information systems. Oper.ation Research, 42, 1120─1136.
[24] Gao, X. B., Liao, L.Z., Qi, L. Q. (2005). A novel neural network for variational inequalities with linear and nonlinear constraints, IEEE Transactions on Neural Networks, 16, 1305-1317.
[25] Hu, X., (2009). Applications of the general projection neural network in solving extended linear-quadratic programming problems with linear constraints, Neurocomputing, 72 ,1131-1137.
[26] Hu, X., Wang, J. (2007). Design of general projection neural networks for solving monotone linear variational inequalities and linear and quadratic optimization problems, IEEE Transactions on Systems, Man, and Cybernetics, Part B, 37, 1414-1421.
[27] Liao, L.,Qi, H., Qi, L.(2001). Solving nonlinear complementarity problems with neural networks: a reformulation method approach, Journal of Computational and Applied Mathematics, 131, 343-359
[28] Lillo, W. E., Loh, M. H., Hui, S., Z $\breve{a}$ k, S. H. (1993). On solving constrained optimization problems with neural networks: A penalty method approach, IEEE Transactions on Neural Networks, 4, 931-939.
[29] Liu, Q. S., Wang, J. (2008). A one-layer recurrent neural network with a discontinuous hard-limiting activation function for quadratic programming, IEEE Transactions on Neural Networks, 19, 558-570.
[30] Maa, C. Y., Shanblatt, M. A. (1992). Linear and quadratic programming neural network analysis, IEEE Transactions on Neural Networks, 3, 580-594.
[31] Malek, A., Hosseinipour-Mahani, N., Ezazipour, S. (2010). Efficient recurrent neural network model for the solution of general nonlinear optimization problems, Optimization Methods and Software, 25, 1–18.
[32] Nazemi, A. R. (2012). A dynamic system model for solving convex nonlinear optimization problems, Communications in Nonlinear Science and Numerical Simulation, 17, 1696-1705.
[33] Nazemi, A. R., (2013). Solving general convex nonlinear optimization problems by an efficient neurodynamic model, Engineering Applications of Artificial Intelligence, 26, 685–696.
[34] Arjmandzadeh, Z,. Safi, M. R, Nazemi, A. R. (2017). A high performance neural network model for solving random interval linear programming problems, Neural Networks.

[Downloaded from jorlu.ir on 2026-11-29]

[DOR: 20.1001.1.22517286.2019.16.3.6.9]

[35] Nazemi, A. R., (2014). A neural network model for solving convex quadratic programming problems with some applications, Engineering Applications of Artificial Intelligence, 32, 54-62.
[36] Nazemi, A. R., (2011). A dynamical model for solving degenerate quadratic minimax problems with constraints, Journal of Computational and Applied Mathematics, 236, 1282-1295.
[37] Nazemi, A. R., Omidi, F. (2012). A capable neural network model for solving the maximum flow problem, Journal of Computational and Applied Mathematics, 236, 3498-3513.
[38] Nazemi, A. R., Omidi, F.(2013). An efficient dynamic model for solving the shortest path problem, Transportation Research Part C: Emerging Technologies, 26, 1-19.
[39] Nazemi, A. R., Effati, S. (2013). An application of a merit function for solving convex programming problems, Computers and Industrial Engineering, 66, 212–-221.
[40] Nazemi, A. R., Sharifi, E. (2013). Solving a class of geometric programming problems by an efficient dynamic model, Communications in Nonlinear Science and Numerical Simulation, 18, 692-709
[41] Tao, Q., Cao, J., Xue, M., Qiao, H., Tao, Q., Cao, J., Sun, D. (2001). A high performance neural network for solving nonlinear programming problems with hybrid constraints, Physics Letters A, 288, 88–94.
[42] Xia, Y., Wang, J. (2004). A general projection neural network for solving monotone variational inequality and related optimization problems, IEEE Transactions on Neural Networks, 15, 318-328.
[43] Xia, Y., Wang, J. (2004). A recurrent neural network for nonlinear convex optimization subject to nonlinear inequality constraints, IEEE Transactions on Circuits and Systems, 51, 447-458
[44] Xia, Y., Wang, J. (2000). On the stability of globally projected dynamical systems, Journal of Optimization Theory and Applications, 106, 129-150.
[45] Xia, Y., Feng, G., Wang, J. (2004). A recurrent neural network with exponential convergence for solving convex quadratic program and related linear piecewise equation, Neural Networks, 17, 1003-1015
[46] Xia, Y., Feng, G. (2005). An improved network for convex quadratic optimization with application to real-time beamforming, Neurocomputing, 64, 359-374
[47] Xue, X., Bian, W. (2007). A project neural network for solving degenerate convex quadratic program, Neurocomputing, 70, 2449-2459.
[48] Xue, X., Bian, W. (2009). A project neural network for solving degenerate quadratic minimax problem with linear constraints, Neurocomputing, 72, 1826-1838.
[49] Yashtini, M., Malek, A. (2008). A discrete-time neural network for solving nonlinear convex problems with hybrid constraints, Applied Mathematics and Computation, 195, 576-584.
[50] Yang, Y., Cao, J. (2006). Solving quadratic programming problems by delayed projection neural network, IEEE Transactions on Neural Networks, 17, 1630-1634.
[51] Yang, Y., Cao, J. (2006). A delayed neural network method for solving convex optimization problems, International Journal of Neural Systems, 16, 295-303.
[52] Yang, Y., Cao, J. (2008). A feedback neural network for solving convex constraint optimization problems, Applied Mathematics and Computation, 201, 340–-350.
[54] Ortega, T. M., Rheinboldt, W. C. (1970). Iterative solution of nonlinear equations in Several variables, Academic Press, New York.
[55] Miller, R. K., Michel, A. N. (1982). Ordinary Differential Equations, Academic Press, NewYork.

[Downloaded from jorlu.ir on 2026-11-29]

[DOR: 20.1001.1.22517286.2019.16.3.6.9]