

بکارگیری شبکه عصبی در تخمین جریانهای کنترل برداری موتور سنکرون آهنربای دائم

علی اکبر صفوی^۲
safavi@hafiz.shirazu.ac.ir

محمد فرخی^۱
farrokhi@iust.ac.ir

مسعود آزادی^۱
azadi@sun.iust.ac.ir

(۱) دانشکده مهندسی برق - دانشگاه علم و صنعت ایران، تهران ۱۶۸۴۴، ایران

(۲) دانشکده صنعت الکترونیک - دانشگاه شیراز، شیراز، ایران

کلمات کلیدی: موتور سنکرون آهنربای دائم، کنترل برداری، شبکه عصبی و یونت

چکیده: کنترل برداری موتورهای سنکرون یکی از پیشرفته‌ترین روشهای موجود برای این منظور است. یکی از مهمترین عوامل در این روش کنترلی، بدست آوردن تخمین خوبی از جریان برداری i_q است. در این مقاله، این جریان با استفاده از سه شبکه عصبی بدست خواهد آمد. این سه شبکه عبارتند از: پرسپترون چند لایه، شبکه با تابع پایه عریض، و شبکه یونت که شبکه‌ای بر مبنای توابع ویولت است. نتایج شبیه سازی این سه شبکه با یکدیگر مقایسه شده و نشان داده شده است که ویونت دارای دقت بسیار بالاتری است. علاوه بر آن، ویونت دارای ساختار چند وضوحی است و نیاز به زمان آموزش کمتری دارد که این امر باعث مناسب‌تر شدن آن در کاربردهای وصل-خط خواهد شد.

۱- مقدمه

موتورهای dc کاربرد وسیع در کنترل دقیق سرعت تا اواسط دهه ۱۹۷۰ داشته‌اند. اما همان محدودیت‌های موتور dc مانند مقاوم نبودن، باریک بودن محدوده سرعت و احتیاج به نگهداری زیاد بعلت وجود جاروبک‌ها و کموتاتورها، وجود دارند. پیشرفتهای اخیر در میکروپروسورها، مواد مغناطیسی، و تکنولوژی نیمه هادی‌ها باعث بکارگیری موتورهای ac در سیستمهای درایو با عملکرد بالا شده است. در میان انواع مختلف موتورهای ac، موتور سنکرون آهنربای دائم (PMSM) در سالهای اخیر محبوبیت بیشتری را بدلیل برخی مزیت‌ها داشته است. این مزیت‌ها شامل نسبت گشتاور به جریان بالا، توان بالا نسبت به وزن، راندمان بالا و مقاوم بودن است [1]. از نظر نوع کنترل، موتورهای PMSM به دو دسته کلاسه‌بندی می‌شوند:

- ۱- موتور سنکرون آهنربای دائم تغذیه شده با موج مثلثی که به نام موتور dc بدون جاروبک شناخته می‌شود.
- ۲- موتور سنکرون آهنربای دائم تغذیه شده با موج سینوسی که به عنوان موتور سنکرون آهنربای دائم بدون جاروبک شناخته می‌شود.

بعضی از معیارهای اصلی در سیستمهای درایو با عملکرد بالا عبارتند از: پاسخ سریع و دقیق و عدم حساسیت به تغییرات پارامترها. مشخصه عملکرد موتور PMSM با بکارگیری کنترل برداری، شبیه یک موتور dc تحریک مستقل خواهد شد. در موتور PMSM با کنترل برداری نه تنها مشخصه شار و گشتاور از هم مستقل می‌شوند، بلکه منجر به پاسخ سریع و کنترل ساده‌تر می‌شود [1] و [۱۰].

کنترل کننده‌های کلاسیک مانند تناسبی-انتگرال‌گیر (PI) یا تناسبی-انتگرال‌گیر-مشتق‌گیر (PID) کاربرد وسیعی در کنترل موتورهای ac و dc داشته‌اند. اما طراحی این نوع کنترل کننده‌ها هنگامی که مدل دقیق سیستم در دسترس نباشد مشکل خواهد بود. به علاوه عدم شناسایی دینامیک بار و فاکتورهای دیگر مانند نویز، دما، و اشباع بر روی عملکرد این نوع کنترل کننده‌ها، برای محدوده وسیعی از سرعت، تاثیر نامطلوبی می‌گذارد.

هرچند برخی از کنترل کننده‌های تطبیقی مانند کنترل کننده‌های تطبیقی مدل مرجع (MRAC)، کنترل کننده لغزشی (SMC)، کنترل کننده ساختار متغیر، و رگولاتور خود تنظیم وجود دارند که در موتورهای با تعداد پارامترهای نامشخص بکار برده

می‌شوند. اما این نوع کنترل‌کننده‌ها معمولاً بر مبنای پارامترهای مدل سیستم طراحی می‌شوند. از این‌رو در دسترس نبودن مدل دقیق سیستم باعث طراحی مشکل این نوع کنترل‌کننده‌ها می‌شود. امروزه توجه زیادی به شبکه‌های عصبی مصنوعی در مدل‌سازی سیستم‌ها و کنترل می‌شود. در درایوهای PMSM با عملکرد بالا شبکه عصبی می‌تواند نقش کلیدی در شناسایی سیستم و کنترل سرعت ایفا نماید، زیرا شبکه عصبی مزیت نگاشت غیرخطی کارآمد بین ورودی‌ها و خروجی‌ها را دارد. با وجود کارهای اخیر که بر روی کنترل موتور dc بوسیله شبکه عصبی انجام شده اما بکارگیری شبکه‌های عصبی در کنترل موتورهای ac در آغاز راه است [1].

۲- دینامیک موتور PMSM

برای بکاربردن شبکه عصبی نیازی به مطالعه مدل سیستم نیست. ورودی‌ها و خروجی‌های شبکه عصبی می‌توانند متغیرهای سیستم باشند. ولی در مواردی ترجیح داده می‌شود که دینامیک سیستم مدل شود تا این که ورودی‌ها و خروجی‌های شبکه عصبی از طریق معین و ملموس انتخاب شوند [1].

در این مقاله برخلاف کنترل سرعت‌های مرسوم از تکنیک کنترل‌برداری همراه با شبکه عصبی برای بدست‌آوردن حساسیت گشتاور بالا در درایو PMSM استفاده شده است. استراتژی کنترل برداری در مختصات مبنای چرخان سنکرون است. مدل ریاضی موتور PMSM و بار توسط معادلات در مختصات مبنای چرخان سنکرون $q-d$ تعیین می‌شوند.

$$v_q = Ri_q + p\lambda_q + P\omega_r\lambda_d \quad (1)$$

$$v_d = Ri_d + p\lambda_d - P\omega_r\lambda_q \quad (2)$$

که در آنها $\lambda_q = L_q i_q$ ، $\lambda_d = L_d i_d + \lambda_m$ ، v_d و v_q ولتاژهای محور q و d هستند، و i_q و i_d به ترتیب جریان استاتور در راستای محورهای d و q می‌باشند. L_q و L_d اندوکتانس محورهای d و q ، R مقاومت استاتور، λ_m شار پیوندی مغناطیسی ثابت تولید شده بوسیله آهنربای دائم روتور، ω_r سرعت موتور، p تعداد جفت قطب‌ها، و P اپراتور مشتق‌گیر است. گشتاور الکتریکی به صورت زیر بیان می‌شود:

$$T_e = 3P/2[\lambda_m i_q + (L_d - L_q)i_d i_q] \quad (3)$$

و دینامیک موتور می‌تواند بصورت زیر بیان شود [1]:

$$T_e = Jp\omega_r + B\omega_r + T_L \quad (4)$$

که در آن T_L گشتاور بار، B ضریب میرایی، و J اینرسی روتور می‌باشد. گشتاور بار در معادله (۴) می‌تواند هر نوع مشخصه مکانیکی غیرخطی ناشناخته‌ای داشته باشد. معادله زیر می‌تواند مدلی برای بار غیرخطی باشد:

$$T_l = K_1\omega_r^2 + K_2\omega_r + K_3 \quad (5)$$

که در آن K_1 ، K_2 و K_3 ثابت‌اند. استراتژی کنترل‌برداری بر این اساس است که جریان محور d ، یعنی i_d ، صفر باشد تا شار پیوندی محور d ، یعنی λ_d ، فقط وابسته به شار پیوندی تولیدشده توسط آهنربای دائم روتور باشد. با این استراتژی کنترل، مدل ماشین ساده‌تر شده و به صورت زیر بیان می‌شود:

$$pi_q = 1/L_q(v_q - Ri_q - K_b\omega_r) \quad (6)$$

$$p\omega_r = 1/J[T_e - T_L - B\omega_r] \quad (7)$$

که در آن $K_b = P\lambda_m$ و گشتاور T_e در معادله (۴) متناسب با جریان محور q است و به فرم زیر بیان می‌شود:

$$T_e = K_T i_q \quad (8)$$

که در آن $K_T = 3p/2\lambda_m$. معادله (۸) مشابه معادله گشتاور برای موتور dc تحریک مستقل می‌باشد که در آن i_q مشابه جریان آرمیچر ماشین dc است. از این‌رو کنترل گشتاور دقیق موتور PMSM بوسیله کنترل جریان محور q ، یعنی i_q امکان‌پذیر خواهد بود. اکنون برای ساده‌تر ساختن عمل کنترل، معادلات موتور PMSM را می‌توان به فرم یک سیستم تک ورودی- تک خروجی در محدوده زمان با ترکیب معادلات (۵) تا (۸) بدست آورد.

$$L_q J \frac{d^2 \omega_r(t)}{dt^2} + (RJ + L_q B + K_2 L_q) \frac{d\omega_r(t)}{dt} + (RB + K_b K_T + K_2 R) \omega_r(t) + K_1 L_q \frac{d\omega_r^2(t)}{dt} + K_1 R \omega_r^2(t) + K_3 R - K_T v_q(t) = 0 \quad (9)$$

مدل زمان-گسسته موتور PMSM ساده شده را می‌توان با جایگزینی تمام مقادیر پیوسته با عنصر تفاضلی بدست آورد.

$$\omega_r(n+1) = \alpha \omega_r(n) + \beta \omega_r(n-1) + \gamma \omega_r^2(n) + \delta \omega_r^2(n-1) + \varepsilon v_q(n) + \vartheta \quad (10)$$

که در آن α و β و γ و δ و ε و ϑ توابعی از پارامترهای موتور هستند [3]. از معادله (10) برای بدست آوردن مدل معکوس سیستم استفاده میشود

$$v_q(n) = [\omega_r(n+1) - \alpha \omega_r(n) - \beta \omega_r(n-1) - \gamma \omega_r^2(n) - \delta \omega_r^2(n-1) - \vartheta] / \varepsilon \quad (11)$$

مدل زمان-گسسته محور q را می‌توان با جایگزینی $v_q(n)$ و $\omega_r(n)$ در معادله (6) بدست آورد

$$i_q(n) = A i_q(n-1) + B v_q(n) + C \omega_r(n) \quad (12)$$

$$A = 1 - R\Delta T / L_q, \quad B = \Delta T / L_q, \quad C = \Delta T K_b / L_q$$

که در آن ΔT زمان نمونه‌گیری است. بنابراین معادله جریان محور q به شکل زیر بدست می‌آید

$$i_q(n) = A i_q(n-1) + B [\omega_r(n) - (\alpha + \frac{\varepsilon C}{\beta}) \omega_r(n-1) - \beta \omega_r(n-2) - \gamma \omega_r^2(n-2) - \vartheta] / \varepsilon \quad (13)$$

طرف راست معادله (13) تابعی غیرخطی از سرعت ω_r است. هدف از بکارگیری شبکه عصبی، نگاشت غیرخطی بین جریان محور q ، یعنی $i_q(n)$ ، و سرعت $\omega_r(n)$ بر طبق معادله (13) است. معادله (13) ساختار شبکه عصبی برای کنترل سرعت موتور PMSM را مشخص می‌نماید [1].

۳- ساختار MLP برای PMSM

یکی از جنبه‌های مهم بکارگیری شبکه عصبی بدست آوردن ورودی‌ها و خروجی‌های آن است. دینامیک معکوس موتور PMSM که در معادله (13) توصیف شده است، ورودی‌ها و خروجی‌های شبکه عصبی بکار گرفته شده را بوجود می‌آورد. بر طبق معادله (13) ورودی‌های شبکه عصبی پیشنهادی، سرعت موتور در لحظه n ، $(n-1)$ و $(n-2)$ ، و همچنین جریان محور q در لحظه $(n-1)$ است. بنابراین بردار ورودی به صورت $[i_q(n-1), \omega_r(n-2), \omega_r(n-1), \omega_r(n)]$ در خواهد آمد. خروجی، جریان محور q در لحظه n ، یعنی $i_q(n)$ ، است. سپس باید تعداد لایه‌های مخفی و همچنین تعداد سلول‌ها در هر لایه را تعیین نمود که این کار عمدتاً با روش سعی و خطا صورت می‌گیرد. برای کاربرد این مقاله، یک لایه مخفی با سه سلول کفایت می‌کند. تابع تبدیل بکار رفته در لایه مخفی و لایه‌های خروجی از نوع توابع S-شکل هستند. قدم بعدی تعیین وزن‌ها و بایاس‌های شبکه عصبی از طریق آموزش (برای بدست آوردن هدف مطلوب با ورودی‌های داده شده) است. الگوریتم یادگیری پس-انتشار خطا برای سیستم موجود بکار می‌رود که مبتنی بر مینیمم کردن مجموع مربعات خطا بین خروجی‌های شبکه عصبی و تابع هدف است. مقدار اولیه وزن‌ها به صورت اتفاقی تعیین می‌شود [4].

۴- ساختار RBF برای PMSM

شکل ۱ شبکه عصبی با تابع پایه شعاعی را نشان می‌دهد. در این شبکه سیگنال‌های ورودی مستقیماً وارد سلول‌های لایه مخفی می‌شوند و هیچگونه وزنی بین ورودی‌ها و سلول‌های مخفی وجود ندارد. برخلاف شبکه‌های MLP، که دارای تابع فعالیت عمومی (یعنی توابع S-شکل) هستند، در این شبکه‌ها توابع فعالیت دارای عملکرد محلی‌اند. تعداد سلولهای لایه مخفی از روش سعی و خطا بدست می‌آید. در لایه خروجی تنها جمع‌کننده‌ها وجود دارند که ورودی این جمع‌کننده‌ها خروجی وزن‌دار سلول‌های لایه مخفی در وزن‌های مربوطه است. تعداد سلول‌های لایه خروجی برابر با تعداد خروجی‌ها است. در این مقاله توابع فعالیت برای ساختار RBF (φ ها) از نوع گوسی در نظر گرفته شده‌اند که از نوع توابع محلی‌اند. در تعلیم این شبکه علاوه بر تنظیم وزن‌ها نیاز به تنظیم مراکز توابع فعالیت نیز خواهد بود. وزن‌ها از روش گرادیان نزولی بر طبق کمترین مجموع مربعات خطا بدست می‌آید و مرکز توابع فعالیت نیز به طور اتفاقی با فواصل معین تعیین میشوند [4]. در این مقاله تعداد سلولهای لایه مخفی ۱۰۰

در نظر گرفته شده است.

۵- ساختار ویونت برای PMSM

علیرغم توانایی‌های شبکه‌های عصبی، برخی مشکلات وجود دارند که باید قبل از کاربرد وسیع آنها در سیستم‌ها، رفع شوند. برای مثال غالباً یک تعبیر فیزیکی را نمی‌توان به مدل معادل با شبکه عصبی آموزش دیده نسبت داد. اغلب توابع بکار رفته در شبکه‌های عصبی توابع عمومی هستند که اجازه یادگیری محلی یا عبارتی بکارگیری بهتر از شبکه را نمی‌هند و همچنین همگرا شدن الگوریتم یادگیری معمولاً تضمین نمی‌شود. پیشرفت‌های اخیر و رشد سریع نظریه ویولت‌ها و آنالیز "چند-وضوحی" ابزار قدرتمند ریاضی را برای شبکه‌های عصبی مهیا می‌نمایند.

نظریه ویولت‌ها از کار گراسمن و مورلت [6] بر روی پرازش سیگنال، الهام گرفت. پس از آن مایر [7] اولین ویولت متعامد را معرفی کرد. امروزه برخی ویولت‌ها را به عنوان روشی جدید نمایش برای نمایش توابع در نظر می‌گیرند، بعضی دیگر آن را بعنوان روشی جدید در آنالیز فرکانس-زمان و برخی دیگر مبحث جدید ریاضیات می‌دانند. در این مقاله از ویولت‌ها به عنوان ابزاری قدرتمند برای توسعه شبکه‌های عصبی استفاده خواهد شد.

دو راه اصلی برای بکارگیری ویولت‌ها در شبکه‌های عصبی وجود دارد. اولین روش، که بنام شبکه ویولت نامیده می‌شود، یک شبکه عصبی $(1+1/2)$ لایه بر مبنای ویولت‌ها معرفی می‌کند و از روش یادگیری پس-انتشار خطا برای تعلیم آن استفاده می‌شود. دومین روش که بنام ویونت مشهور است، ویولت را در ساختاری شبیه به ساختار شبکه‌های RBF بکار می‌گیرد و برای یادگیری از روشی غیرتکراری و سلسله‌مراتبی سود می‌برد [2]. شکل ۲ ساختار یک ویونت با p ورودی و یک خروجی را نشان می‌دهد. همانطور که در شکل ۳ مشخص است در ساختار ویونت دو دسته تابع تابع وجود دارد. توابع مقیاس، یعنی ϕ ها و توابع ویولت، یعنی ψ ها.

ویولت‌ها توابع پایه محلی هستند و دارای خصوصیات متعامد بودن، واحد بودن، محلی بودن، و همچنین عملکرد سریع می‌باشند. نمونه‌هایی از توابع مقیاس و ویولت متناظر با آن در شکل ۳ نشان داده شده است. ویولت‌ها معمولاً در چهارچوب چند-وضوحی که توسط مالات [8] انجام شد، معرفی می‌شوند.

تابع $f(x) \in L^2(R)$ را در نظر بگیرید که در آن $L^2(R)$ فضای برداری از تمامی توابع قابل اندازه‌گیری، تک بعدی قابل انتگرال‌گیری است. همچنین فرض کنید v_m فضای برداری شامل تمامی تقریب‌های ممکن از $f(x)$ در وضوح m باشد. سپس توالی فضاهای $v_m (m \in \mathbb{Z})$ نشان‌دهنده سطح وضوح‌های متوالی از $f(x)$ است. تابع ϕ یک تابع مقیاس از تحلیل چند-وضوحی نامیده می‌شود و به فرم زیر بیان می‌شود:

$$\phi_{m,k}(x) = 2^{-m/2} \phi(2^{-m}x - k) \quad m, k \in \mathbb{Z} \quad (14)$$

که در آن 2^{-m} ، k ، و $2^{-m/2}$ به ترتیب ضرایب تاخیر، انتقال، و بهنجارکننده انرژی هستند. فرض کنید w_m مکمل متعامد از v_m به v_{m-1} باشد به طوری که $v_m \oplus w_m = v_{m-1}$ است. در این صورت توابع پایه متعامد (مشابه w_m ها) که ویولت نامیده می‌شوند، توسط $\psi_{m,k}$ ها نشان داده می‌شوند و به سادگی از $\phi_{m,k}$ ها بدست می‌آیند. دسته‌ای از ویولت‌ها را می‌توان به فرم زیر بیان نمود:

$$\psi_{m,k}(x) = 2^{-m/2} \psi(2^{-m}x - k) \quad m, k \in \mathbb{Z} \quad (15)$$

که در آن 2^{-m} ، k ، و $2^{-m/2}$ به ترتیب ضرایب تاخیر، انتقال، و بهنجارکننده ویولت‌ها هستند. آنگاه $L^2(R)$ می‌تواند بفرم زیر بیان گردد:

$$L^2(R) = \bigcup_{m \in \mathbb{Z}} v_m = \dots w_{-1} \oplus w_0 \oplus w_1 \dots = \bigoplus_{m \in \mathbb{Z}} w_m \quad (16)$$

و با شروع تقریب تابع $f(x)$ از وضوح صفر ($m = 0$) داریم

$$f(x) = f_0(x) + \sum_{m=-\infty}^0 \sum_{k=-\infty}^{\infty} d_{m,k} \psi_{m,k}(x) \quad (17)$$

$$f_0(x) = \sum_k a_{0,k} \phi_{0,k}(x) \quad (18)$$

برای ساختن یک شبکه عصبی پیشخور احتیاج به توصیف محل مرکز توابع پایه، نوع توابع پایه، و الگوریتم یادگیری است.

توابع پایه در یک ویولت، ویولت‌ها و توابع مقیاس هستند. بنابراین فقط محل مرکز توابع پایه و الگوریتم یادگیری برای شبکه باید مشخص گردد. محل مرکز توابع پایه برای سطح‌های وضوح متفاوت در شکل ۴ نشان داده شده است. معادله (۱۷) ساختار اصلی یک ویولت را توصیف می‌کند، که در آن چگونگی عملکرد متقابل هر ویولت را در کل شبکه تشریح می‌نماید. همچنین این معادله نشان می‌دهد که توابع مقیاس فقط در مرحله ابتدایی تقریب برای تولید f_0 بکار می‌روند و پس از آن فقط از ویولت‌ها استفاده می‌شود. ابتدا تقریب اول از تابع f بدست آورده می‌شود، یعنی f_0 ، سپس تقریب بهتری به نام f_{-1} با استفاده از ویولت‌های با ضریب تاخیر یکسان با تابع مقیاس (در اینجا $m=0$) بدست آورد. اضافه نمودن ویولت‌های با وضوح بالاتر (در اینجا $m=-1$) باعث بهتر شدن تقریب، یعنی f_{-2} ، نسبت به تقریب قبل می‌شود. این روند تا بدست آوردن تابع هدف، یا بدست آوردن دقت دلخواه از تقریب، ادامه می‌یابد. تعداد ویولت‌هایی که باید بکار روند و موقعیت آنها در هر سطح از تقریب در شکل ۴ آمده است. برای پیدا نمودن ضرایب شبکه $a_{m,k}$ و $d_{m,k}$ ، که وزنه‌های شبکه هستند، روش زیر بوسیله بخشی و استفانپولوس [9] پیشنهاد شده است. برای سادگی، حالت تک بعدی را در نظر می‌گیریم و فرض می‌کنیم که اطلاعات ورودی و خروجی نمونه‌های تکی از تابع پیوسته هستند.

$$\mathbf{x} = [x_1 \quad x_2 \quad \dots \quad x_n]^T \quad \text{و} \quad \mathbf{F} = [F(x_1) \quad F(x_2) \quad \dots \quad F(x_n)]^T \quad (19)$$

برای اجتناب از تکرار، تابع پایه (که تابع مقیاس یا ویولت است) توسط علامت $\theta(\mathbf{x})$ نشان داده می‌شود. ضرایب مشابه a و d توسط ضریب عمومی c نشان داده می‌شوند. در هر سطح از تقریب، مسئله تقریب می‌تواند به صورت زیر فرموله شود:

$$\bar{F}(x) = \sum_{k=1}^K c_k \theta_k(x) \quad (20)$$

که در آن k تعداد توابع پایه بکار رفته برای این تقریب بخصوص است. برای مثال، بدترین تقریب، F_0 ، فقط توسط توابع مقیاس محاسبه می‌شود و بنابراین \bar{F} برابر F_0 می‌باشد. از آن جایی که F_0 و تمامی c_k ها مشخص نیستند، ممکن است F به عنوان تقریبی از F_0 برای حل مسائل فرض شود. عموماً در هر سطحی از وضوح:

$$F_{m-1}(x) = F_m(x) + \sum_k d_{m,k} \psi_{m,k}(x) \quad m, k \in Z \quad (21)$$

که در آن F_m (به عنوان تقریب محاسبه شده قبل) مشخص است. سپس، در هر حالت، F تقریبی از F_{m-1} فرض می‌گردد. بنابراین معادله (۲۱) به صورت زیر بیان می‌شود:

$$\bar{F}(x) = \sum_k d_{m,k} \psi_{m,k}(x) \quad m, k \in Z \quad (22)$$

که در آن $\bar{F}(x) = F(x) - F_m(x)$ که فرم یکسانی با معادله (۲۰) دارد. همچنین معادله (۲۰) را می‌توان به فرم بردار-ماتریسی نیز نوشت:

$$\bar{\mathbf{F}} = \mathbf{A} \mathbf{C} \quad (23)$$

که در آن

$$\bar{\mathbf{F}} = \begin{bmatrix} \bar{F}(x_1) \\ \vdots \\ \bar{F}(x_n) \end{bmatrix}, \quad \mathbf{A} = \begin{bmatrix} \theta_1(x_1) & \dots & \theta_k(x_1) \\ \vdots & \ddots & \vdots \\ \theta_1(x_n) & \dots & \theta_k(x_n) \end{bmatrix}, \quad \mathbf{C} = \begin{bmatrix} c_1 \\ \vdots \\ c_k \end{bmatrix} \quad (24)$$

روش کمترین-مربعات برای معادله (۲۳) عبارت است از

$$\mathbf{C} = ((\mathbf{A}^T \mathbf{A})^{-1} \mathbf{A}^T) \bar{\mathbf{F}} = \mathbf{A}^+ \bar{\mathbf{F}} \quad (25)$$

که در آن \mathbf{A}^+ معکوس کاذب \mathbf{A} است. اگر $\mathbf{A}^T \mathbf{A}$ ماتریس همانی باشد، معادله (۲۵) به صورت زیر ساده می‌شود:

$$\mathbf{C} = \mathbf{A}^T \bar{\mathbf{F}} \quad (26)$$

پیدا کردن ضرایب با استفاده از معادله (۲۶) روشی کلاسیک است ولی اگر یک دسته بزرگ از اطلاعات نمونه در دسترس باشد، آنگاه هر ضریب شبکه می‌تواند از حاصلضرب داخلی تابع هدف f و ویولت مربوطه بدست آید [2]:

$$a_{m,k} = \langle \phi_{m,k}, F \rangle, \quad d_{m,k} = \langle \psi_{m,k}, F \rangle \quad (27)$$

۶- نتایج شبیه‌سازی

با توجه به معادله (۱۳) ورودی‌های شبکه عصبی عبارتند از: سرعت موتور در لحظه n ، $(n-1)$ ، و $(n-2)$ ، و جریان محور q در لحظه $(n-1)$. همچنین خروجی شبکه، جریان محور q در لحظه n است. بنابراین شبکه‌های عصبی بکاررفته در این مقاله دارای چهار ورودی و یک خروجی هستند.

شبکه MLP دارای سه سلول در لایه مخفی و یک سلول در لایه خروجی است. توابع تبدیل بکار رفته در لایه مخفی و لایه خروجی از نوع توابع s -شکل هستند. ضریب یادگیری $\eta = 0.21$ ، تعداد دوره‌ی تعلیم شبکه ۲۰۰۰ مرتبه، و تعداد الگوهای ورودی ۱۲۰۰ الگو می‌باشند. بعد از تعلیم شبکه، مجموع مربعات خطا برابر با $1/8814$ بدست آمده است. خروجی شبکه همراه با خروجی دلخواه را می‌توان در شکل ۵ مشاهده کرد. برای بدست آوردن نتیجه بهتر از این شبکه می‌بایست تعداد الگوهای ورودی و تعداد دوره‌های آموزش را افزایش داد که این امر مستلزم وقت و حافظه بالا خواهد بود.

در شبکه RBF تعداد سلولهای لایه مخفی را ۱۰۰ سلول در نظر گرفته‌ایم. تابع تبدیل بکار رفته در این شبکه تابع گوسی، از خانواده‌ی توابع پایه محلی، است. مرکز توابع فعالیت بطور اتفاقی و با فاصله ۱ در نظر گرفته شده است. همچنین تعداد الگوهای ورودی در این شبکه ۱۲۰۰ الگو می‌باشد. بعد از تعلیم شبکه، مجموع مربعات خطا به مقدار $0/62$ بدست آمد. خروجی شبکه همراه با خروجی دلخواه را در شکل ۶ می‌توان مشاهده کرد.

شبکه ویونت بکار رفته در این مقاله، ابتدا در سطح وضوح صفر، که دارای یک سلول با تابع مقیاس می‌باشد، اولین تخمین را برای تابع موردنظر به ما می‌دهد. در این مرحله در واقع اولین وزن شبکه $(a_0, 0)$ و در نتیجه $f_0(x)$ بدست می‌آید. در مرحله بعد یعنی سطح رزولوشن ۱، تعداد ۱۵ سلول با توابع فعالیت ویولت به شبکه اضافه می‌شود و دیگر وزنه‌های شبکه، که $d_{0,0}$ تا $d_{0,14}$ هستند، بدست می‌آیند. بنابراین دومین تخمین از تابع موردنظر با استفاده از توابع فعالیت ویولت با دقت بسیار خوبی در دسترس خواهد بود. در این مرحله مجموع مربعات خطا $0/41$ شده است. مهمترین مزیت این شبکه نسبت به دیگر شبکه‌های عصبی ساختار چند-وضوحی آن است. به عنوان مثال، اگر در دومین تخمین، مجموع مربعات خطا به اندازه قابل قبولی کاهش نیافته باشد، در این صورت با اضافه کردن ۲۴۰ سلول با توابع ویولت در سطح وضوح ۲ به تخمین بهتری می‌توانیم برسیم. در واقع در شبکه ویونت، همگرایی همواره تضمین شده است. تعداد کل سلول‌های شبکه ویونت، که اولین سلول آن با تابع مقیاس و مابقی با تابع فعالیت ویولت هستند، از رابطه $(2^n)^m$ بدست می‌آید که در آن n تعداد ورودی‌ها و m سطح وضوح شبکه است. خروجی شبکه ویونت در سطح وضوح ۱ همراه با خروجی دلخواه را در شکل ۷ می‌توان مشاهده کرد.

۷- نتیجه‌گیری

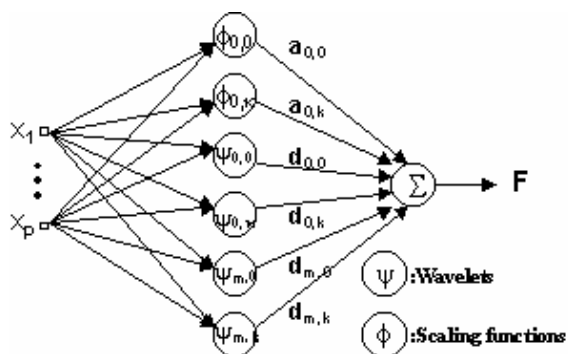
پیشرفتهای اخیر در میکروپروسورها، مواد مغناطیسی، تکنولوژی نیمه‌هادی‌ها و همچنین روشهای کارآمد و نوین کنترلی مانند کنترل برداری و همچنین بکارگیری هوش مصنوعی مانند شبکه‌های عصبی در کنترل موتورهای AC، امکان کنترل سرعت و گشتاور این موتورها را، همانند موتورهای dc تحریک مستقل مهیا می‌سازد. در میان شبکه‌های عصبی مصنوعی شبکه ویونت دارای مزیت‌هایی مانند، عملکرد چند-وضوحی، سرعت، ساختار ساده، و همگرایی تضمین شده است. همان‌گونه که نتایج شبیه‌سازی مشاهده می‌شود، شبکه RBF و شبکه ویونت دارای خروجی بهتر نسبت به شبکه MLP هستند. همچنین شبکه ویونت در مقایسه با شبکه RBF دارای ساختاری ساده‌تر و عملکردی چند-وضوحی است و با تعداد سلولهای کمتر و سریع‌تر از شبکه RBF به جواب قابل قبولی رسیده است.

مراجع

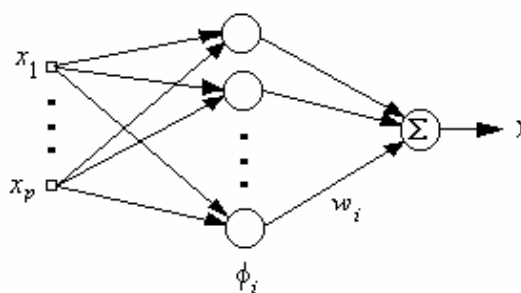
- [1] M. A. Rahman, M. A. Hoque, "On-line adaptive artificial neural network based vector control of permanent magnet synchronous motors" *IEEE Trans. On Energy Conversion*, vol. 13, no. 4, pp. 311_318, December 1998.
- [2] A. A. Safavi, J. A. Romagnoli, "Application of wavelet-based neural network to the modeling and optimisation of an experimental distillation", *Engineering Applications of Artificial Intelligence*, vol. 10, no. 3, pp. 301-313, 1997.

- [3] Peter Vas, *Sensorless Vector and Direct Torque Control*, Oxford University Press , 1998.
- [4] Simon Haykin, *Neural Networks*, Printice Hall N.J., 1999.
- [5] G.K.Dubey, *Power Semiconductor Controlled Drives*, Prentice Hall, 1989.
- [6] A. Grossman and J. Morlet, "Decomposition of Hardy function into square itegrable wavelets of constant shape", *SIAM Journal of Mathematics Annals*, vol. 15, pp.723-736, 1984.
- [7] Y. Meyer, " Principe d'incertitude, bases hilbertiennes et algebras d'operateurs, Seminaire Bourbaki, nr. 662, 1985-1986.
- [8] S.G. Mallat, "A theory for multiresolution signal decomposition: the wavelet representation", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 11, no. 7, pp. 674-693, 1989.
- [9] B. R. Bakshi and G. Stephanopoulos, "Wave-net: a multiresolution, hierarchcal neural network with localized learning", *AICHE Journal*, vol. 39, no. 1, pp.57-81, 1993.

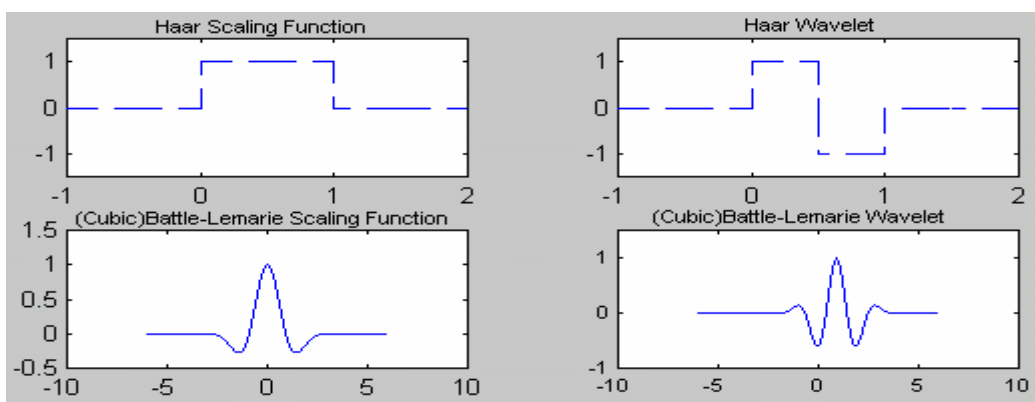
[۱۰] جعفر سلطانی و فریدالدین کتیرایی، " کنترل سرعت موتور سنکرون مغناطیس دائم تغذیه شده با اینورترهای ولتاژ، توسط کامپیوتر. "قسمت دوم_کنترل به روش برداری، هفتمین کنفرانس مهندسی برق ایران، صفحات ۱۴۶ الی ۱۵۳، اردیبهشت ۱۳۷۸.



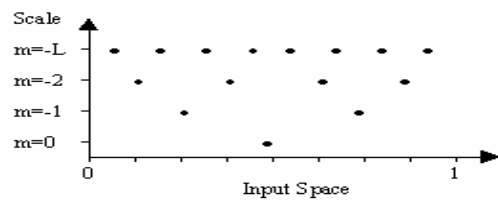
شکل ۲ - ساختار ویونت با P ورودی و یک خروجی



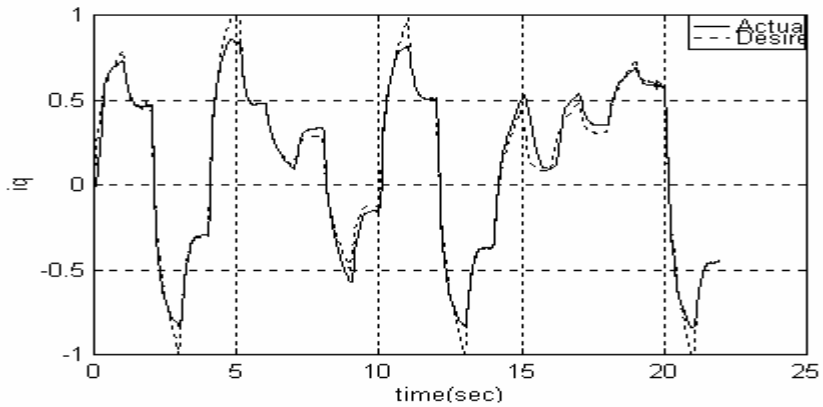
شکل ۱- شبکه با تابع پایه شعاعی (RBF)



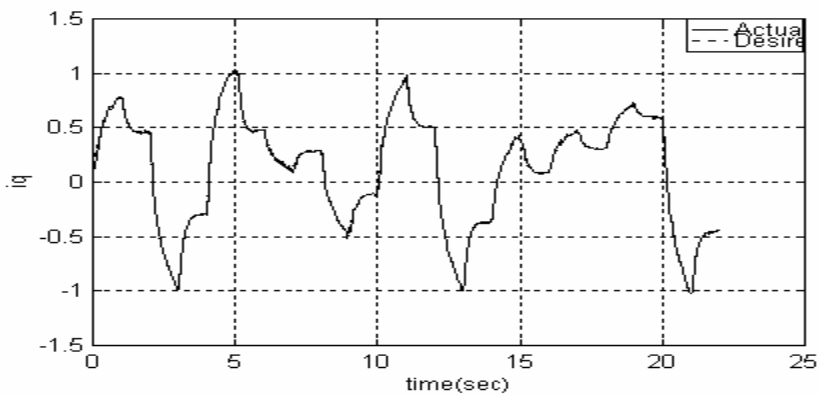
شکل ۳- نمونه‌هایی از توابع مقیاس و ویولت متناظر با آن



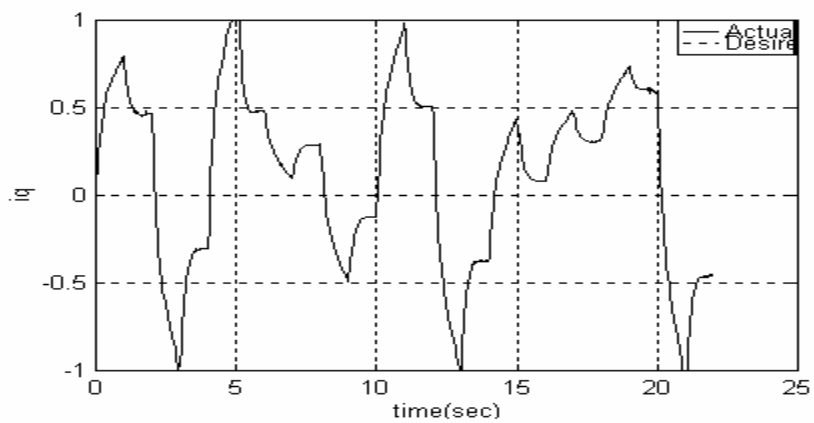
شکل ۴ - محل مرکز توابع پایه برای سطوح های رزولوشن متفاوت



شکل ۵ - خروجی شبکه MLP و خروجی دلخواه بعد از تعلیم شبکه



شکل ۶ - خروجی شبکه RBF و خروجی دلخواه بعد از تعلیم شبکه



شکل ۷ - خروجی شبکه ویونت و خروجی دلخواه بعد از تعلیم شبکه