



دانشکده مهندسی برق و رباتیک

گروه کنترل

پایان نامه کارشناسی ارشد

کنترل فازی مد لغزشی با شبکه‌های عصبی در ربات سه بازوی SCARA

سجاد شجاع مجید آباد

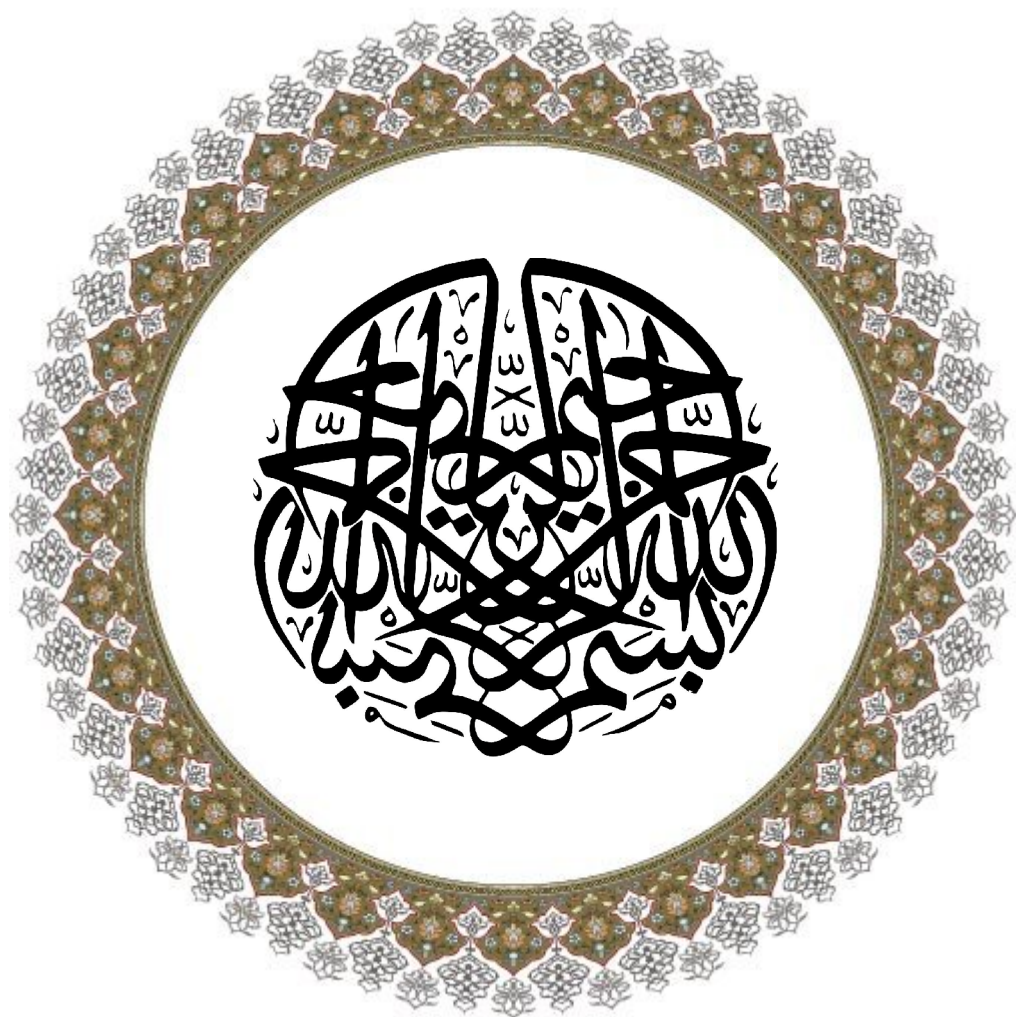
استاد راهنما

دکتر حیدر طوسی‌ان شاندیز

استاد مشاور

دکتر محمد مهدی فاتح

شهریور ۱۳۸۹



تقدیم به پدر و مادر عزیز و مهربانم

تشکر و قدردانی

با تقدیر و تشکر از اساتید ارجمند جناب آقای دکتر طوسی‌ان شاندیز و جناب آقای دکتر فاتح که با راهنمایی‌های ارزنده‌شان مرا در انجام این تحقیق یاری نمودند. همچنین از پدر و مادر عزیز و مهربانم که همواره باعث ایجاد انگیزه در من برای ادامه تحصیل و تلاش در این راه بوده‌اند و همه اساتید گرانقدر تشکر و قدردانی نموده توفیق و عزت همه را از خداوند متعال خواستارم.

سجاد شجاع مجید آباد

چکیده

در این پایان نامه، طراحی و توسعه کنترل کننده‌های عصبی-فازی-مد لغزشی بر روی ربات سه بازوی اسکارا بررسی شده است. اهداف اصلی عبارت اند از: طراحی کنترل کننده مستقل از مدل یا به عبارتی غلبه بر مشکل محاسبه قانون کنترل معادل، حذف نوسانات ناخواسته حاصل از جمله کلیدزنی، طراحی کنترل کننده مقاوم در برابر اصطکاک مفاصل و اغتشاش خارجی. در این راستا، از یک ساختار شبکه عصبی موازی با سیستم فازی برای تقریب بخش مربوط به مدل سیستم و جمله کلیدزنی استفاده می‌نماییم. همچنین عملکرد شبکه عصبی پرسپترون چند لایه و شبکه با توابع گوسین را در تقریب قانون کنترل مورد مقایسه قرار می‌دهیم. ضرایب شبکه عصبی به گونه‌ای تنظیم می‌شوند که خروجی سیستم فازی (برای شبکه پرسپترون) یا سطح لغزش (برای شبکه گوسین) به سمت صفر میل نماید. جواب‌های حاصل از این روش، با کنترل مد لغزشی مقایسه شده و عملکرد مطلوب آن را نشان می‌دهد. در ادامه و به منظور کاهش هزینه و حجم محاسبات کنترل کننده فوق، دو کنترل کننده دیگر نیز پیشنهاد می‌دهیم. در مورد اول که گریزی بر مفهوم طراحی کنترل کننده مد لغزشی و عصبی-فازی-مد لغزشی در حوزه زمان گسسته با سهولت پیاده‌سازی عملی دارد، به طراحی کنترل کننده مورد اول در حوزه گسسته می‌پردازیم. همچنین تأثیر زمان نمونه برداری بر همگرایی سیستم حلقه بسته بحث شده است. در مورد دوم، برای کاهش بار محاسباتی کنترل کننده عصبی-فازی-مد لغزشی، از سیستم فازی صرفنظر شده و شبکه عصبی با سطح لغزش غیرخطی ایفای نقش می‌نماید. در هر یک از موارد فوق، نتایج شبیه سازی پیاده شده بر روی بازوی سه رابط اسکارا، دال بر کارایی مطلوب و بهبود عملکرد سیستم حلقه بسته است. علاوه بر این کنترل کننده‌های پیشنهادی برای دسته وسیعی از سیستم های چند ورودی-چند خروجی قابل اعمال می باشند.

کلمات کلیدی: کنترل عصبی-فازی-مد لغزشی، مد لغزشی زمان گسسته، سطح لغزش غیرخطی، شبکه های عصبی، منطق فازی، ربات سه بازوی اسکارا.

فصل اول: مقدمه

- ۱-۱- تاریخچه ۱
- ۲-۱- ساختار پایان نامه ۴

فصل دوم: کنترل مد لغزشی زمان- پیوسته و زمان- گسسته

- ۱-۲- کنترل مد لغزشی پیوسته در زمان ۵
- ۱-۱-۲- طراحی قانون کنترل مد لغزشی ۵
- ۲-۱-۲- زمان رسیدن به سطح لغزش ۱۲
- ۲-۲- کنترل مد لغزشی زمان- گسسته ۱۴
- ۳-۲- تفاوت اساسی کنترل مد لغزشی زمان- پیوسته و زمان گسسته ۲۰
- ۴-۲- مزایا و قابلیت‌های کنترل مد لغزشی ۲۲

فصل سوم: دینامیک بازوی سه رابط SCARA

- ۱-۳- دینامیک بازوی ربات ۲۴
- ۱-۱-۳- معادلات لاگرانژ ۲۵
- ۲-۱-۳- انرژی جنبشی و پتانسیل ۲۶
- ۳-۱-۳- نیروی تعمیم یافته ۲۷
- ۴-۱-۳- محرک‌ها ۲۸
- ۴-۱-۳- اصطکاک ۲۸
- ۲-۳- معادلات حرکت ۳۱
- ۳-۳- ویژگی‌ها و معادلات بازوی سه رابط اسکارا ۳۴

فصل چهارم: کاربرد سیستم‌های فازی و شبکه‌های عصبی در شناسایی و کنترل

- ۱-۴- شبکه‌های عصبی ۳۸

۳۹	۴-۱-۱- کاربرد عمومی شبکه‌های عصبی
۴۱	۴-۱-۲- انواع شبکه‌های عصبی
۴۴	۴-۱-۳- مراحل شناسایی و مدل‌سازی
۴۵	۴-۱-۴- مثال شبیه سازی
۴۸	۴-۲- سیستم‌های فازی
۴۸	۴-۲-۱- کنترل فازی
۴۹	۴-۲-۲- شناسایی و مدل‌سازی فازی
۵۰	۴-۳- مقایسه سیستم‌های فازی و عصبی
۵۱	۴-۴- کنترل فازی- عصبی و عصبی- فازی

فصل پنجم: کنترل عصبی- فازی- مد لغزشی پیوسته و گسسته در زمان

۵۴	۵-۱- مروری بر تحقیقات انجام شده
۵۶	۵-۲- طراحی کنترل کننده زمان- پیوسته برای بازوی ربات
۵۶	۵-۲-۱- کنترل مد لغزشی پیوسته در زمان
۵۹	۵-۲-۲- نتایج شبیه‌سازی به ازای کنترل کننده مد لغزشی
۶۳	۵-۲-۳- طراحی کنترل کننده عصبی- فازی مد لغزشی
۶۳	۵-۲-۳-۱- ساختار کنترل کننده پیشنهادی
۶۵	۵-۲-۳-۲- محاسبه قانون کنترل کلیدزنی
۶۷	۵-۲-۳-۳- محاسبه قانون کنترل معادل
۶۷	۵-۲-۳-۳-۱- تقریب قانون کنترل معادل با شبکه عصبی MLP
۷۱	۵-۲-۳-۳-۲- نتایج شبیه‌سازی به ازای کنترل کننده NFSMC-MLP
۷۳	۵-۲-۳-۳-۳- تقریب قانون کنترل معادل با شبکه عصبی RBF
۷۸	۵-۲-۳-۳-۴- نتایج شبیه‌سازی به ازای کنترل کننده NFSMC-RBF

۵-۳-۳-۲-۵- موارد مهم در طراحی کنترل کننده NFSMC	۸۱
۵-۳- طراحی کنترل کننده زمان- گسسته بر روی بازوی ربات	۸۲
۵-۳-۱- طراحی کنترل کننده مد لغزشی زمان- گسسته	۸۲
۵-۳-۲- نتایج شبیه سازی به ازای کنترل کننده DSMC	۸۷
۵-۳-۳- کنترل عصبی- فازی مد لغزشی در حوزه زمان- گسسته (NFDSMC)	۹۰
۵-۳-۱- تقریب قانون کنترل معادل	۹۱
۵-۳-۲- تقریب قانون کنترل کلیدزنی	۹۲
۵-۳-۴- نتایج شبیه سازی به ازای کنترل کننده NFDSMC-MLP	۹۲
۵-۳-۵- تاثیر زمان نمونه برداری بر خروجی، سیگنال کنترلی و همگرایی سیستم	۹۵

فصل ششم: کنترل عصبی- مد لغزشی با سطح لغزش غیرخطی

۶-۱- مروری بر تحقیقات انجام شده	۹۸
۶-۲- تحلیلی بر سطح لغزش غیرخطی	۹۹
۶-۱-۲- زمان رسیدن به سطح لغزش	۱۰۱
۶-۲-۲- واگرایی سیگنال کنترلی	۱۰۲
۶-۳- طراحی کنترل کننده مد لغزش غیرخطی کراندار TSMC	۱۰۴
۶-۴- نتایج شبیه سازی به ازای کنترل کننده TSMC	۱۰۷
۶-۵- طراحی کنترل کننده عصبی- مد لغزش غیرخطی NTSMC	۱۱۰
۶-۶- نتایج شبیه سازی به ازای کنترل کننده NTSMC	۱۱۴
فصل هفتم: نتیجه گیری و پیشنهادات	۱۱۹

ضمائم

ضمیمه الف- برنامه شناسایی با استفاده از شبکه عصبی دو لایه (M-file)	۱۲۱
ضمیمه ب- بلوک های طراحی شده در SIMULINK	۱۲۳

ضمیمه ج- بلوک‌های طراحی شده در SIMULINK ۱۲۸

فهرست مراجع ۱۳۰

فهرست اشکال

شکل (۱-۲): قاعده حرکت مد لغزشی ۸

شکل (۲-۲): لایه مرزی برای کنترل مد لغزشی ۱۲

شکل (۳-۲): سیستم حلقه بسته شامل کنترل کننده زمان گسسته و سیستم تحت کنترل زمان پیوسته ۱۵

شکل (۱-۳): حرکت رابط k -ام ۲۶

شکل (۲-۳): مدل اصطکاک مفصل k -ام ۲۹

شکل (۳-۳): از راست به چپ: ۱- ربات Bosch Turboscara SR ۲- ربات ICOMATIC03 با سه درجه آزادی

۳- ربات Adept one چهار محوری ۳۵

شکل (۴-۳): ربات اسکارای سه محوری ۳۶

شکل (۱-۴): شناسایی سیستم با شبکه‌های عصبی ۴۰

شکل (۲-۴): توابع wavelet: از راست ۱- مکزیکی، ۲- مورلت، ۳- شانون ۴۳

شکل (۳-۴): تابع f (خط) و تقریب آن $\hat{f}$ (نقطه چین) با شبکه دولایه ۴۶

شکل (۴-۴): خروجی سیستم $y_p(k)$ (خط) و خروجی شبکه عصبی دولایه $\hat{y}_p(k)$ (نقطه چین) ۴۷

شکل (۵-۴): مدل فازی تاکاگی- سوگنو مرتبه اول با دو قانون ۵۳

شکل (۱-۵): مسیر مطلوب (نقطه چین) و زاویه مفصل (خط) حاصل از SMC ۶۱

شکل (۲-۵): گشتاور ورودی حاصل از SMC ۶۲

شکل (۳-۵): دو ساختار NFSMC ۶۴

شکل (۴-۵): نمایش دسته‌های فازی ورودی $F_S(s_i(t))$ و خروجی $F_S(u_{iFuzzy}(t))$ ۶۶

شکل (۵-۵): ساختار شبکه عصبی دولایه، به منظور تقریب قانون کنترل ۶۷

شکل (۶-۵): مسیر مطلوب (نقطه چین) و زاویه مفصل (خط) حاصل از NFSMC-MLP ۷۱

- شکل (۷-۵): گشتاور ورودی حاصل از NFSMC-MLP ۷۲
- شکل (۸-۵): ساختار شبکه عصبی RBF ۷۵
- شکل (۹-۵): کنترل مد لغزش، بر پایه شبکه عصبی RBF ۷۸
- شکل (۱۰-۵): مسیر مطلوب (نقطه چین) و زاویه مفصل (خط) حاصل از NFSMC-RBF ۷۹
- شکل (۱۱-۵): گشتاور ورودی حاصل از NFSMC-RBF ۸۰
- شکل (۱۲-۵): بردار مسیر مطلوب (نقطه چین) و زاویه مفصل (خط) حاصل از DSMC ۸۸
- شکل (۱۳-۵): ورودی های کنترل حاصل از DSMC ۸۹
- شکل (۱۴-۵): بردار مسیر مطلوب (نقطه چین) و زاویه مفصل (خط) حاصل از NFDSMC ۹۳
- شکل (۱۵-۵): ورودی های کنترل حاصل از NFDSMC ۹۴
- شکل (۱۶-۵): مسیر مطلوب مفصل اول و زاویه مفصل به ازای سه زمان نمونه برداری متفاوت (NFDSMC) ۹۵
- شکل (۱۷-۵): ورودی کنترلی مفصل اول به ازای سه زمان نمونه برداری متفاوت (NFDSMC) ۹۶
- شکل (۱۸-۵): سطح لغزش گسسته مفصل اول به ازای سه زمان نمونه برداری متفاوت (NFDSMC) ۹۷
- شکل (۱-۶): مسیر مطلوب (نقطه چین) و زاویه مفصل (خط) حاصل از TSMC ۱۰۸
- شکل (۲-۶): گشتاور ورودی حاصل از TSMC ۱۰۹
- شکل (۳-۶): مسیر مطلوب (نقطه چین) و زاویه مفصل (خط) حاصل از NTSMC ۱۱۵
- شکل (۴-۶): گشتاور ورودی حاصل از NTSMC ۱۱۶
- شکل (۵-۶): سطح لغزش غیرخطی NTSMC ۱۱۷

اشکال پیوست (شبیه سازی ها)

- شکل (ب-۱): سیستم حلقه بسته ۱۲۳
- شکل (ب-۲): مدل بازوی ربات سه رابط SCARA ۱۲۳
- شکل (ب-۳): نمایی از عملکرد موازی سیستم فازی و عصبی (NFSMC-MLP) ۱۲۴
- شکل (ب-۵): نمایی از عملکرد موازی سیستم فازی و عصبی (NFSMC-RBF) ۱۲۶

- شکل (ب-۶): بلوک های مربوط به شبکه عصبی RBF و آموزش آنلاین ۱۲۷
- شکل (ج-۱): بلوک های مربوط به کنترل کننده NFTSMC ۱۲۸
- شکل (ج-۲): بلوک های مربوط به شبکه عصبی MLP و آموزش آنلاین ۱۲۹

فصل اول: مقدمه

۱-۱- تاریخچه

در علم کنترل با مسئله طراحی کنترل کننده برای یک سیستم روبرو هستیم، به گونه‌ای که سیستم بتواند عملیات خاصی را که مورد نظر است انجام دهد. با توجه به تنوع سیستم‌ها همچنین اهداف کنترلی مختلفی که وجود دارد، محققین روش‌های کنترلی متنوعی را پایه‌ریزی کرده و گسترش داده‌اند. برای مثال، مدت‌ها از روش‌های کنترل خطی برای طراحی سیستم‌های کنترل استفاده می‌شد، اما امروزه به دلیل ماهیت غیرخطی و پیچیدگی سیستم‌های موجود، از روش‌های کنترل غیرخطی استفاده می‌گردد. هرچند هنوز هم در کاربردهای عملی، کنترل کننده‌های خطی به دلیل سادگی و ارزانی مورد توجه بسیاری قرار دارد. امروزه کنترل غیرخطی پیشرفت گسترده‌ای از لحاظ تئوری و عملی داشته است. در حوزه تئوری، گام‌های مهمی در بخش‌هایی از تکنیک‌های خطی‌سازی فیدبکی، کنترل تطبیقی غیرخطی و کنترل مد لغزشی برداشته شده است.

کنترل مد لغزشی^۱ (SMC) از سال ۱۹۷۰ تا به امروز به علت مقاوم بودن در برابر اغتشاشات و تغییر پارامترها مورد توجه فراوان قرار گرفته است. کنترل مد لغزشی یکی از انواع سیستم‌های کنترل ساختار متغیر^۲ (VSCS) می‌باشد. تحقیقات در این زمینه توسط امیلیانوف و دانشکده وی در سال ۱۹۵۷ شروع شد [۱-۲] و برای طراحی کنترل کننده‌های مقاوم تا به امروز ادامه یافته است. ایده

^۱ Sliding Mode Control

^۲ Variable Structure Control System

کنترل مد لغزشی تا قبل از انتشار مقاله [۳] و کتاب [۴] توسط اُتکین^۱ و ایتکیز^۲ در سال‌های ۱۹۷۷ و ۱۹۷۶ ناشناخته مانده بود. روش کنترلی مورد نظر، یکی از مهم‌ترین روش‌ها کنترلی در مقابل عدم قطعیت‌ها، مشخصه‌های غیرخطی و اغتشاشات خارجی محدود می‌باشد. SMC کاربردهای مختلفی در زمینه‌های الکترونیک قدرت [۵]، سیستم‌های الکتریکی قدرت [۶] و بازوی ربات [۷] داشته است. اخیراً بسیاری از کاربردهای عملی موفقیت آمیز روش SMC، نشان دهنده اهمیت این روش در سه دهه اخیر می‌باشد. از مقالات [۸-۹] می‌توان به عنوان شاهدی بر تمرکز روی SMC و کاربرد گسترده آن نام برد.

در دو دهه اخیر نیز کنترل SMC در دو زمینه پیشرفت زیادی داشته است. ۱- در سال ۱۹۹۳ به منظور طراحی کنترل کننده مد لغزشی دقیق‌تر با زمان همگرایی محدودتر، کنترل مد لغزشی غیرخطی^۳ (TSMC) برای اولین بار توسط ونکتارمن^۴ [۱۰] مطرح شد. این شاخه از SMC در ۱۰ سال گذشته کاربرد زیادی داشته است [۱۱-۱۵]. ۲- از سال ۱۸۸۷ به اینطرف، به منظور استفاده از پردازشگرهای سیگنال دیجیتال^۵ (DSP) و میکروکنترلرها به عنوان کنترل کننده ارزان قیمت در طراحی SMC، کنترل مد لغزشی در حوزه گسسته در زمان^۶ (DSMC) توسط محققان زیادی بسط داده شد [۱۶-۲۲].

¹ Utkin

² Itkis

³ Terminal sliding mode control

⁴ Venktaraman

⁵ Digital signal processors

⁶ Discrete sliding mode control

از سال ۱۹۶۰ به اینطرف، هوش مصنوعی^۱ به کاربرد های صنعتی راه یافته و بیشتر در زمینه دانش شخص خبره بر اساس تصمیم گیری برای طراحی و مانیتورینگ فرایندها و تولیدات صنعتی عمل نمود. این ابزار با پیشرفت های تکنولوژی کامپیوتر و ظهور کامپیوترهای شخصی ارتقاء یافته و بسیاری از کاربردهای هوش مصنوعی تحقق بخشیده شد. در سال ۱۹۸۰ با اختراع ریزپردازنده های فازی، منطق فازی^۲ در صنعت و خصوصاً در ژاپن کاربرد زیادی یافت. در این کشور، همچنین شبکه های عصبی^۳ و محاسبات تکاملی^۴ هم در دانشگاه و هم در صنعت مورد توجه قرار گرفت. در نتیجه این اتفاقات، هوش محاسباتی^۵ متولد شد. افزون بر این، در دهه های اخیر در کنار کاربردهای گسترده هوش محاسباتی در علوم دیگر، روش های جدید کنترلی تحت عنوان کنترل هوشمند مطرح شده و گسترش یافته است. یکی از ابزار کنترلی هوشمند، استفاده از منطق فازی می باشد که نخستین بار در سال ۱۹۶۵ توسط دکتر لطفی زاده تحت عنوان مجموعه های فازی مطرح شد، ابزار دیگر استفاده از شبکه های عصبی می باشد.

به منظور بهره گیری همزمان از مزایای SMC و کنترل هوشمند و همچنین برای رفع معایب هر یک از این دو، ترکیب های مختلف کنترل مد لغزشی با روش های منطق فازی و شبکه های عصبی تحت عناوین گوناگون از قبیل: کنترل فازی مد لغزشی^۶ [۲۷-۲۳]، کنترل عصبی مد لغزشی^۷ [۳۶-۲۸] و یا کنترل عصبی-فازی مد لغزشی^۸ [۴۲-۳۷] مورد تحقیق بسیاری در سال های اخیر به ویژه در زمینه

^۱ Artificial intelligence

^۲ Fuzzy logic

^۳ Neural network

^۴ Evolutionary computations

^۵ Soft computing

^۶ Fuzzy-sliding mode

^۷ Neuro-sliding mode

^۸ Neuro-fuzzy sliding mode

زمینه کنترل ربات گردیده است. توجه شود که شبکه‌های عصبی و سیستم فازی، اکثراً نه به عنوان روش، بلکه به عنوان ابزاری برای کنترل تطبیقی، کنترل مد لغزشی و شناسایی سیستم می‌باشند. فصل‌های بعد به مرور جامعتری بر تحقیقات صورت گرفته در این زمینه می‌پردازد.

۱-۲- ساختار پایان نامه

در تحقیق انجام گرفته در این پایان نامه به منظور طراحی کنترل کننده مستقل از مدل و مقاوم در برابر اغتشاش خارجی، دو ترکیب جدید از کنترل مد لغزشی با شبکه‌های عصبی و سیستم فازی ارائه شده که در کنترل ربات به کار می‌رود. نتایج شبیه سازی‌های انجام گرفته در این تحقیق، عملکرد مطلوب و توانمندی کنترل کننده‌های طراحی شده در کنترل ربات اسکارا را نشان می‌دهد.

در ادامه ساختار پایان نامه بدین شرح می‌باشد: در فصل ۲ به تشریح کنترل مد لغزشی در دو حوزه پیوسته و گسسته پرداخته شده که پایه و اساس طراحی در این تحقیق می‌باشد. معادلات دینامیکی مورد مطالعه که ربات سه بازوی اسکارا می‌باشد در فصل ۳ بررسی شده است. در فصل ۴ مرور جامعی از شبکه‌های عصبی و سیستم فازی به همراه یک مثال شناسایی بیان شده است. در بخش اول فصل ۵، کنترل NFSMC به ازای دو شبکه عصبی ^۱RBF و ^۲MLP بررسی، و در بخش دوم، طراحی کنترل کنترل کننده NFSMC متناظر در حوزه زمان گسسته مورد بحث قرار گرفته است. طراحی کنترل کننده مستقل از مدل جدید به ازای حذف سیستم فازی و سطح لغزش غیرخطی به همراه اثبات پایداری در فصل ۶ بیان می‌گردد. در پایان، فصل ۷ به نتیجه‌گیری و ارائه برخی پیشنهادات می‌پردازد.

^۱ Radial base function network

^۲ Multilayer perceptron network

فصل دوم: کنترل مد لغزشی زمان-پیوسته و زمان-گسسته

در این فصل به دلیل اهمیت کنترل مد لغزشی در علم کنترل، این روش در دو حوزه زمان-پیوسته و زمان-گسسته بررسی می شود.

۱-۲- کنترل مد لغزشی پیوسته در زمان

سیستم غیرخطی حلقه باز و تغییرپذیر با زمان زیر را در نظر بگیرید [۴۳]:

$$\dot{x}^{(n)} = f(x, t) + b(x, t)u(t) + d(t) \quad (1-2)$$

که در آن $x(t) = [x, \dot{x}, \dots, x^{(n-1)}]^T$ بردار حالت، $d(t)$ اغتشاش وابسته به زمان و کراندار $|d(t)| < D$ (به ازای $D > 0$)، ورودی کنترل و $f(x, t)$ و $b(x, t)$ توابع غیرخطی از بردار حالت و زمان می باشند. مسئله ردگیری، عبارت است از یافتن یک قانون کنترل به طوری که هرگاه مسیر مطلوب $x_d(t)$ داده شده باشد، خطای ردگیری $x(t) - x_d(t)$ ، با وجود عدم قطعیت های مدل، دینامیک های مدل نشده و اغتشاش، به سمت صفر میل کند.

۱-۱-۲- طراحی قانون کنترل مد لغزشی

روش طراحی SMC شامل دو مرحله مهم می باشد که بیانگر دو فاز مهم SMC است [۴۴]:

۱- فاز رسیدن به سطح لغزش^۱: در این فاز، حالت سیستم از هر نقطه اولیه در زمان محدود به سطح لغزش آورده می‌شود.

۲- فاز مد لغزشی^۲: حالت سیستم بر روی سطح لغزش، حرکت لغزشی دارد، یعنی سطوح لغزش در نقش جاذب^۳ عمل می‌کنند.

این فازها با دو مرحله طراحی زیر متناظر می‌باشند.

انتخاب سطح لغزش^۴: سطوح لغزش بر اساس مشخصه‌های دینامیکی مطلوب تجویز شده انتخاب می‌شوند ولی در بیشتر موارد سطح لغزش از نوع خطی انتخاب می‌گردد. با تعریف خطای ردیابی بصورت [۴۳]:

$$e(t) = x(t) - x_d(t) = [e, \dot{e}, \dots, e^{(n-1)}]^T \quad (2-2)$$

برای سیستم مرتبه n ، سطح لغزش^۵ به صورت زیر تعریف می‌شود:

$$s(x, t) = \left(\frac{d}{dt} + \lambda \right)^{(n-1)} e(t) = \sum_{k=0}^{n-1} \binom{n-1}{k} \lambda^k e^{(n-1-k)} \quad (3-2)$$

که در آن $\lambda > 0$ می‌باشد.

طراحی کنترل غیرپیوسته^۶: استراتژی کنترل غیر پیوسته به منظور رسیدن به سطح لغزش در زمان محدود شکل می‌پذیرد. هدف اصلی طراحی کنترل کننده کلیدزنی که حالت سیستم را به سطح

¹ Reaching phase

² Sliding mode phase

³ Attractor

⁴ Switching manifold selection

⁵ Sliding surface

⁶ Discontinuous

لغزش رسانده و روی آن حفظ نماید می‌باشد. برای کاراکتریزه کردن این وظیفه از روش لیاپانوف استفاده می‌گردد. روش لیاپانوف معمولاً برای تعیین خواص پایداری نقطه تعادل بدون حل معادله حالت استفاده می‌شود. در این قسمت برای بدست آوردن قانون کنترل، تابع لیاپانوف را بصورت زیر تعریف می‌کنیم (فرض شود که $n = 2$):

$$V(t) = 0.5s^2(t) \quad (۴-۲)$$

با $V(0) = 0$ و $V > 0$ برای $s > 0$.

بنابراین شرط کافی برای پایداری سیستم (۱-۲)،

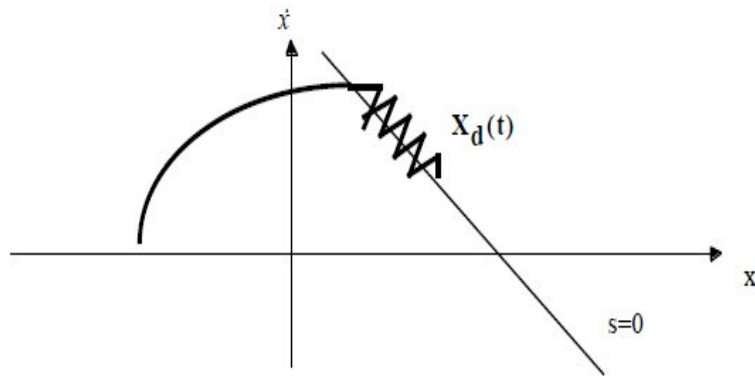
$$\dot{V}(t) = 0.5 \frac{d}{dt}(s^2(t)) \leq -\eta |s(t)| \quad (۵-۲)$$

می‌باشد. در نتیجه خواهیم داشت:

$$\dot{s}(t) \cdot \text{sgn}(s(t)) \leq -\eta \quad (۶-۲)$$

رابطه مذکور را شرط رسیدن به سطح لغزش گویند که در آن $\eta > 0$ و برقراری این شرط، سیستم (۱-۲) را به آنچه مد لغزش نامند، می‌رساند. مد لغزشی به این معناست که هرگاه مسیر حالت $x(t)$ به سطح لغزش $s = 0$ رسیده باشد، مستقل از عدم قطعیت‌های مدل، دینامیک‌های مدل نشده و اغتشاش، همچنانکه به سمت $x_d(t)$ می‌لغزد روی این سطح باقی بماند.

رسیدن به سطح لغزش به این معناست که یک مسیر سیستم $x(t)$ خارج از سطح لغزش قرار دارد اما شرط (۶-۲) را دنبال می‌کند. شکل ۱-۲ اصول کلی همگرایی SMC را برای یک سیستم درجه دو را نشان می‌دهد. همانطور که در این شکل نشان داده شده، در مرحله اول متغیر حالت $x(t)$ به سطح لغزش می‌رسد و در مرحله دوم بر روی سطح به سمت مقدار مطلوب $x_d(t)$ می‌لغزد.



شکل ۲-۱- قاعده حرکت مد لغزشی [۴۳]

حال باید قانون کنترلی که سیستم (۱-۲) را به مود لغزشی ببرد را بدست آوریم. بنابراین با مشتق گیری از معادله (۳-۲) داریم:

$$\dot{s} = x^{(n)} - x_d^{(n)} + \sum_{k=1}^{n-1} \binom{n-1}{k} \lambda^k e^{(n-k)} \quad (۷-۲)$$

با قرار دادن (۱-۲) و (۷-۲) در معادله (۶-۲) بدست می آید:

$$\text{sgn}(s) \left(f(x,t) + b(x,t)u(t) + d(t) - x_d^{(n)} + \sum_{k=1}^{n-1} \binom{n-1}{k} \lambda^k e^{(n-k)} \right) \leq -\eta \quad (۸-۲)$$

با ساده سازی رابطه مذکور، قانون کنترل مد لغزشی را به شکل زیر تعریف می کنیم:

$$u(t) = \hat{b}^{-1}(x,t) \left(G(x_d^{(n)} - \sum_{k=1}^{n-1} \binom{n-1}{k} \lambda^k e^{(n-k)} - K(x,t) \text{sgn}(s)) - \hat{f}(x,t) \right) \quad (۹-۲)$$

که در آن $K(x,t) > 0$ و $\hat{b}$ و $\hat{f}$ به ترتیب تقریبی از توابع b و f می باشند. برای تعیین ضریب G محدوده های زیر را تعریف می کنیم:

$$0 \leq \beta_{\min} \leq b.b^{-1} \leq \beta_{\max} \quad (۱۰-۲)$$

پس G را به صورت زیر تعریف می‌کنیم:

$$G = (\beta_{\min} \cdot \beta_{\max})^{-1/2} \quad (۱۱-۲)$$

در نهایت، حاشیه بهره با رابطه زیر تعریف می‌شود:

$$\beta = \left(\frac{\beta_{\max}}{\beta_{\min}} \right)^{1/2} \quad (۱۲-۲)$$

رابطه (۹-۲) را می‌توان به صورت زیر نیز در نظر گرفت:

$$\begin{aligned} u(t) &= \hat{b}^{-1}(x, t)(\tilde{u} - \hat{f}(x, t)) \\ \tilde{u} &= G(\hat{u} - K(x, t)\text{sgn}(s)) \\ \hat{u} &= x_d^{(n)} - \sum_{k=1}^{n-1} \binom{n-1}{k} \lambda^k e^{(n-k)} \end{aligned} \quad (۱۳-۲)$$

هدف طراحی، یافتن $K(x, t)$ به گونه‌ای است که شرط رسیدن به سطح لغزش (۶-۲) برآورده گردد. با قرار دادن (۱۳-۲) در (۸-۲) بدست می‌آید:

$$\text{sgn}(s)(\Delta f + (b\hat{b}^{-1}G - 1)\hat{u} + d - b\hat{b}^{-1}GK(x, t)\text{sgn}(s)) \leq -\eta \quad (۱۴-۲)$$

که $\Delta f = f - b\hat{b}^{-1}\hat{f}$ می‌باشد. با ساده‌سازی عبارت (۱۴-۲) داریم:

$$(\Delta f + (b\hat{b}^{-1}G - 1)\hat{u} + d)\text{sgn}(s) - b\hat{b}^{-1}G.K(x, t) \leq -\eta \quad (۱۵-۲)$$

نامعادله فوق همواره برقرار است اگر

$$b\hat{b}^{-1}G.K(x, t) \geq \eta + \left| \Delta f + (b\hat{b}^{-1}G - 1)\hat{u} + d \right| \quad (۱۶-۲)$$

با قدر مطلق گیری از هر یک از عبارات سمت راست رابطه فوق، نامعادله قویتری به دست می‌آید،

$$b\hat{b}^{-1}G.K(x,t) \geq \eta + |\Delta f| + |b\hat{b}^{-1}G - 1| \cdot |\hat{u}| + |d| \quad (۱۷-۲)$$

با جایگذاری حد پایین $\beta_{\min} = b\hat{b}^{-1}$ در معادله بالا و با توجه به $\beta_{\min} G = \beta^{-1}$ ، خواهیم داشت:

$$K(x,t) \geq \eta + \beta \left(|\Delta f| + |1 - \beta^{-1}| \cdot |\hat{u}| + |d| \right) \quad (۱۸-۲)$$

حد بالای محدوده‌ها را به این صورت اختصاص می‌دهیم:

$$|\Delta f| < F \quad |d| < D \quad |\hat{u}| < U$$

که از فرضیات مسئله می‌باشند. در نهایت شرط کافی برای اینکه سطح لغزش $s = 0$ یک ناحیه جذب باشد، به صورت زیر به دست می‌آید:

$$K(x,t) \geq \eta + \beta \left(F + |1 - \beta^{-1}| \cdot U + D \right) \quad (۱۹-۲)$$

به طور خلاصه طراحی روش SMC شامل مراحل زیر است:

۱- λ را به گونه ای انتخاب می‌کنیم که فرکانس های مدل نشده ν توسط فیلتر حذف گردد.

اگر $\nu_{\min}$ حد پایین ν باشد آنگاه λ طوری طراحی می‌شود که $\lambda \ll \nu_{\min}$ باشد.

۲- $\hat{u}$ و حد بالای آن را بدست می‌آوریم (U).

۳- مقادیر تخمینی f و b را بدست می‌آوریم ($\hat{b}$ و $\hat{f}$).

۴- G را از رابطه (۱۱-۲) به دست می‌آوریم.

۵- η و همچنین حدهای بالای F برای $|\Delta f|$ و D برای $|d|$ را تعیین می‌نماییم.

۶- $K(x,t)$ را محاسبه می‌کنیم.

ویژگی مهم SMC نسبت به سایر روش‌های کنترل غیرخطی این است که هرگاه سیستم در مد لغزشی باشد، سیستم حلقه بسته (۲-۱)، بدون توجه به اغتشاش و عدم قطعیت‌های مدل، از معادله دینامیکی خطی $s(x,t)=0$ پیروی می‌کند. نقطه ضعف روش SMC این است که تغییرات شدید ورودی کنترل u ممکن است موجب اشکال در سیستم تحت کنترل گردد. برای اجتناب از این مورد، یک لایه مرزی^۱ نزدیک سطح لغزش در نظر می‌گیریم که دینامیک‌های ورودی کنترل u را هموار ساخته و اطمینان می‌دهد که حالت‌های سیستم درون این لایه باقی بماند. پهنای باند با 2ϕ نشان داده می‌شود. اگر $|s|$ فاصله بین حالت x و سطح لغزش باشد، آنگاه حالت x درون باند قرار دارد اگر $|s| \leq \phi$ ، و بیرون از آن است اگر $|s| \geq \phi$ باشد. با جایگزین کردن $\text{sgn}(s)$ توسط $\text{sat}(s/\phi)$ ، قانون کنترل (۲-۱۳) به شکل زیر در می‌آید

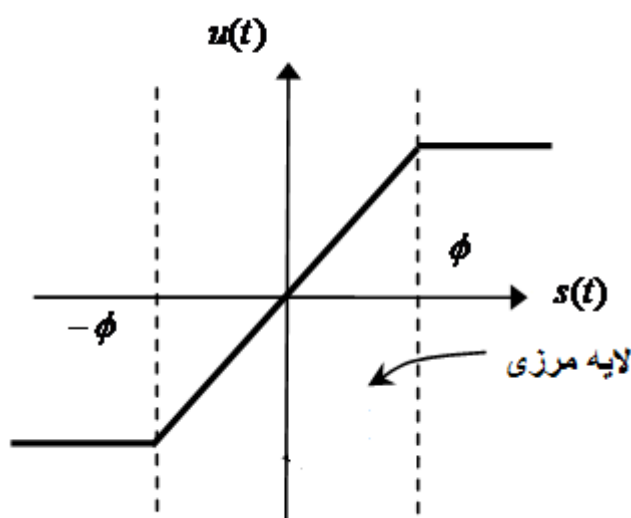
$$\begin{aligned} u(t) &= \hat{b}^{-1}(x,t)(\tilde{u} - \hat{f}(x,t)) \\ \tilde{u} &= G(\hat{u} - K(x,t)\text{sat}(s)) \end{aligned} \quad (2-20)$$

که تابع اشباع بدین صورت تعریف می‌گردد:

$$\text{sat}(w) = \begin{cases} w & \text{if } |w| < 1 \\ \text{sgn}(w) & \text{if } |w| > 1 \end{cases} \quad (2-21)$$

شکل ۲-۲، لایه مرزی را در اطراف خط لغزش یک سیستم درجه دو به همراه جمله $K.\text{sat}(s/\phi)$ از قانون کنترل (۲-۲۱) نشان می‌دهد.

^۱ Boundary layer



شکل ۲-۲- لایه مرزی برای کنترل مد لغزشی [۴۵]

در این حالت اگر K از رابطه (۲-۱۹) محاسبه شود، پایداری مجانبی حاصل می‌گردد. در نتیجه لایه مرزی یک ناحیه جذب می‌باشد و از نوسانات سیگنال کنترلی جلوگیری می‌کند. اما این بهبودی به قیمت افزایش خطای ردگیری و کاهش عملکرد حاصل می‌گردد.

۲-۱-۲- زمان رسیدن به سطح لغزش

زمانی را که طول می‌کشد تا $e(t)$ از حالت اولیه $e(0)$ به سطح لغزش برسد، زمان رسیدن به سطح لغزش نامند (t_{reach}). این زمان را می‌توانیم از روی شرط لغزش محاسبه نمود. با فرض $s(t) > 0$ از شرط لغزش داریم:

$$\dot{s}(t) \leq -\eta \quad (2-22)$$

با انتگرالگیری از رابطه (۲-۵) مابین $t = 0$ و $t = t_{reach}$ نتیجه می‌دهد:

$$s(t_{reach}) - s(0) \leq -\eta(t_{reach} - 0) \quad (2-23)$$

با توجه به اینکه $s(t_{reach}) = 0$ می‌باشد، بنابراین

$$t_{reach} \leq \frac{s(0)}{\eta} \quad (24-2)$$

به دست می‌آید. برای حالت $s(t) < 0$ ، نتیجه مشابه معادله (۲۴-۲) می‌باشد. بنابراین در حالت کلی داریم:

$$t_{reach} \leq \left| \frac{s(t=0)}{\eta} \right| \quad (25-2)$$

در نتیجه با شروع از هر شرط اولیه، مسیر حالت در زمان محدود و کوچکتر از $\frac{|s(t=0)|}{\eta}$ به سطح لغزش همگرا می‌شود [۴۶].

قونین لغزشی رایج در کنترل سیستم‌های غیرخطی عبارت اند از [۴۷]:

۱- قانون لغزش با بهره ثابت^۱

$$\dot{s} = -Q \text{sign}(s) \quad (26-2)$$

۲- قانون لغزش با بهره ثابت و تناسبی^۲ (قانون کنترل نمایی^۳)

$$\dot{s} = -Ks - Q \text{sign}(s) \quad (27-2)$$

۳- قانون لغزش توانی^۴

$$\dot{s}_i = -A|s_i|^{a_i} \text{sign}(s_i) \quad i=1,2,\dots,m \quad (28-2)$$

¹ Constant rate reaching law

² Constant plus proportional rate reaching law

³ Exponential reaching law

⁴ Power rate reaching law

در روبرو فوق Q و K ماتریس‌های قطری با المان‌های مثبت، A یک اسکالر مثبت و $a_i \in (0,1)$ می‌باشد. انتخاب مناسب این پارامترها شرط رسیدن به سطح لغزش را برآورده می‌کند. در نتیجه از قوانین لغزشی مذکور می‌توان در به دست آوردن قانون کنترلی استفاده کرد.

۲-۲- کنترل مد لغزشی زمان-گسسته^۱ (DSMC)

در حالی که طراحی فیدبک سیستمی در حوزه پیوسته در زمان، یک فرایند مقبول برای مهندسين کنترل می‌باشد، کاربردهای حوزه زمان-گسسته حلقه‌های فیدبکی با نرخ نمونه‌برداری به اندازه کافی سریع، به علت آسانی و دسترسی به پردازشگرهای سخت‌افزاری، موضوع روز می‌باشد. چهارچوب مفهومی طراحی فیدبک در حوزه زمان پیوسته قرار دارد. به عنوان مثال برای کنترل VSS و SMC، تفکر طراحی مد لغزشی برای سیستم پیوسته در زمان، با وجود گسسته گی‌ها و ویژگی‌های کلیدزنی، مفهومی پیوسته دارد. به هر حال، تحلیل کمی کنترل SMC با ویژگی‌های زمان پیوسته آن، با وجود مقبولیت کاربردهای حوزه زمان-گسسته امری مشکل و پیچیده می‌باشد. زمانی که مهندسين کنترل روش کنترل نمونه‌برداری را به کار می‌برند، انتخاب نرخ نمونه‌برداری مناسب امری ضروری می‌باشد. متأسفانه در کنترل SMC پیوسته در زمان، پهنای باند سیستم حلقه بسته راهنمای مفیدی برای انتخاب نرخ نمونه‌برداری نمی‌نماید؛ برای همین در دو دهه گذشته کنترل SMC در حوزه زمان-گسسته کاربرد زیادی یافته است. [۴۸] برای سیستم‌های دیجیتال، کنترل مد لغزشی گسسته، مستقیماً قابل پیاده‌سازی بوده و همچنین برای سیستم‌های آنالوگ با استفاده از واسطه‌هایی انجام می‌پذیرد که در شکل ۲-۳ نشان داده شده است. در این شکل، اجزای تشکیل دهنده سیستم حلقه بسته عبارتند از:

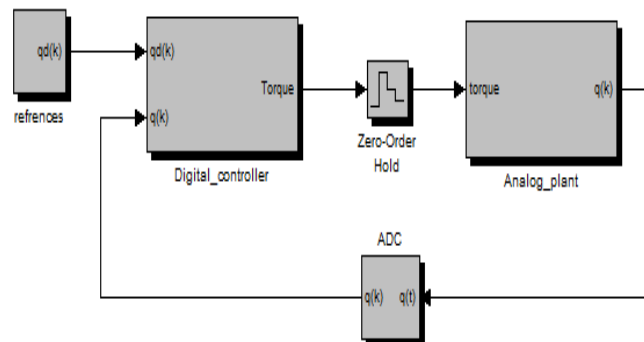
۱- سیستم آنالوگ (سیستم تحت کنترل)

¹ Discrete Sliding Mode Control

۲- AD : مبدل سیگنال آنالوگ به دیجیتال

۳- DA : مبدل سیگنال دیجیتال به آنالوگ

۴- کنترل کننده دیجیتال



شکل ۲-۳- سیستم حلقه بسته شامل کنترل کننده زمان گسسته و سیستم تحت کنترل زمان پیوسته

همانطور که اشاره شد، سیستم تحت کنترلی می تواند دیجیتالی نیز باشد. در این حالت دیگر نیازی نیست که AD و DA بکار روند. برای سیستم دیجیتال، طراحی کنترل کننده بصورت مستقیم و با استفاده از معادلات دیجیتالی توصیف کننده سیستم انجام می گیرد. ولی برای سیستم آنالوگ مانند بازوی ربات، باید معادلات دینامیکی سیستم به حوزه زمان- گسسته برده شوند، و طراحی کنترل کننده DSMC از روی این معادلات انجام گیرد. در فصل ۵ به مفهوم گسسته سازی معادلات سیستم و طراحی قانون کنترل DSMC می پردازیم. در این قسمت نیز به بحث همگرایی و انتخاب زمان نمونه برداری مناسب می پردازیم.

به دلیل آنکه سیگنال کنترلی زمان- گسسته در فواصل نمونه برداری ثابت است، لذا شرط لغزش زمان گسسته متناظر با $\dot{s}(t) \leq \eta T \operatorname{sgn}(s(t))$ ، در حوزه زمان پیوسته همگرایی سطح لغزش را برآورد نمی کند. بنابراین برای رفع مشکل همگرایی و ردیابی مطلوب کنترل کننده DSMC، خلاصه

ای از بارزترین شرایط همگرایی کنترل مد لغزشی زمان گسسته را بیان می‌کنیم [۱۹-۱۶] و [۵۵-۴۹]:

۱- قانون لغزشی پیوسته در زمان گسسته‌سازی شده^۱:

ساده‌ترین شرط حرکت لغزشی گسسته در زمان را از رابطه $s(t)\dot{s}(t) < 0$ می‌توان بصورت،

$$s(k)[s(k+1) - s(k)] < 0 \quad (29-2)$$

نوشت. بنابراین با توجه به شرط رسیدن به سطح لغزش حالت زمان-پیوسته و با استفاده از نگهدار

مرتبه صفر $\dot{s}(t) = \frac{s(k+1) - s(k)}{T}$ با زمان نمونه‌برداری T ، قانون لغزشی در حوزه دیجیتال به شکل زیر بدست می‌آید:

$$\dot{s}(t) < -\eta \operatorname{sgn}(s(t)) \rightarrow \frac{s(k+1) - s(k)}{T} \leq -\eta \operatorname{sgn}(s(k))$$

$$s(k+1) \leq s(k) - \eta T \operatorname{sgn}(s(k)) \quad (30-2)$$

شرط لغزش فوق برای حرکت شبه-لغزشی^۲، لازم است ولی کافی نمی‌باشد. علاوه بر این، شرط لغزش مذکور، همگرایی مسیر حرکت را به سمت هایپرپلن تضمین نمی‌کند، و ممکن است باعث افزایش نوسانات در اطراف هایپرپلن گردد [۵۴].

۲- قانون لغزشی سارپتورک^۳:

در سال ۱۹۸۷، شرط لازم و کافی برای حرکت لغزشی گسسته در زمان توسط سارپتورک بصورت،

¹ Discretized Continuous-Time Reaching Law

² Quasi Sliding Mode

³ Sarpturk Reaching Law

$$|s(k+1)| < |s(k)| \quad (۳۱-۲)$$

پیشنهاد شد. معادل رابطه (۳۰-۲) و (۳۱-۲) را می توان به شکل رابطه زیر نوشت:

$$\begin{aligned} [s(k+1) - s(k)] \operatorname{sgn}(s(k)) &< 0 \\ [s(k+1) + s(k)] \operatorname{sgn}(s(k)) &> 0 \end{aligned} \quad (۳۲-۲)$$

شرط اول بیانگر این است که سیستم حلقه بسته به سمت سطح لغزش حرکت می کند، در حالی که به ازای شرط دوم، سیستم حلقه بسته از سطح لغزش نمی تواند زیاد دور شود. به عبارت دیگر، شرط اول، کران پایین عمل کنترل و شرط دوم، کران بالای آن می باشد. این شرط می تواند همگرایی حرکت لغزشی به سمت هایپرپلن را تضمین نماید و برای اثبات پایدار سیستم حلقه بسته زمان گسسته، به این دو شرط یا معیار $V(k) = 1/2 s^2(k)$ ، رجوع می شود [۵۵].

۳- شرط لغزشی بارتزوویچ^۱:

به منظور طراحی کنترل کننده ای که بتواند همگرایی سیستم حلقه بسته را بهبود ببخشد، بارتزوویچ شرط زیر را پیشنهاد داد:

$$|s(k)| < q \quad (۳۳-۲)$$

در رابطه فوق $q > 0$ و پهنای باند مد شبه- لغزشی نامیده می شود. به ازای شرط فوق نیازی نیست که مسیر حرکت در هر لحظه نمونه برداری، تابع کلیدزنی را قطع کند. درواقع عبور از تابع مجاز است ولی ضروری نمی باشد [۵۴].

۴- قانون لغزشی خطی^۲:

^۱ Bartoszewicz

^۲ Linear Reaching Law

سپورگتون^۱ [۱۷] و هوئی و زاک^۲ در سال ۱۹۹۲، برای رسیدن به همگرایی مطلوب، از قانون لغزش خطی استفاده نمودند:

$$s(k+1) = \Phi s(k) \quad (۳۴-۲)$$

که در آن $0 \leq \Phi < 1$ می‌باشد. قانون کنترل فوق، تشابه زیادی با شرط همگرایی سارپتورک دارد. طراحی کنترل کنند با استفاده از رابطه (۳۴-۲) براحتی انجام می‌گیرد. نکته قابل توجه در رابطه مذکور، عدم حضور قسمت لغزشی است [۴۹].

۵- قانون لغزشی ژائو^۳:

همانطور که قبلاً اشاره شد، بدلیل اینکه قانون کنترل فقط در لحظات نمونه برداری تغییر می‌کند و در فواصل مابین زمان‌های نمونه‌برداری ثابت است، لذا شرط لغزش (۳۰-۲) توان همگراسازی حالت-های سیستم را ندارد. اکثر شرایط لغزشی قبلی که به آن‌ها اشاره شد کامل نیستند و ویژگی‌های کنترل مد لغزشی زمان-گسسته را ارضاء نمی‌کنند. برای غلبه بر این مشکل، ژائو در سال ۱۹۹۵، قانون کنترل لغزشی زمان-گسسته را بصورت زیر پیشنهاد داد [۱۸]:

$$s(k+1) = (1 - qT)s(k) - \varepsilon T \operatorname{sgn}(s(k)) \quad q > 0, \varepsilon > 0, 1 - Tq > 0 \quad (۳۵-۲)$$

از رابطه (۳۵-۲) مشاهده می‌شود که جمله $qTs(k)$ به قانون کنترل (۳۰-۲) افزوده شده است. این رابطه قانون کنترل شبه-لغزشی^۴ (QSMC) نامیده می‌شود. به ازای شرط لغزش مذکور، از شدت نوسانات در اطراف هایپرپلن کاسته می‌شود.

^۱ Spurgeon

^۲ Hui and Zak

^۳ Gao Reaching Law

^۴ Quasi-sliding mode

با اعمال شرط لغزش QSMC، باید علامت $s(k+1)$ مخالف علامت $s(k)$ باشد. با اعمال این ویژگی به معادله (۳۵-۲)، محدوده تغییرات سطح لغزش،

$$|s(k)| < \frac{\varepsilon T}{1 - qT} \quad (۳۶-۲)$$

خواهد بود. از نامعادله (۳۶-۲) چنین برمی آید که با کمتر شدن زمان نمونه برداری همگرایی سیستم حلقه بسته بهبود می آید. (یعنی $|s(k)| \rightarrow 0 \Rightarrow T \rightarrow 0$).

۶- قانون لغزشی نمایی

برای یک سیستم زمان پیوسته، قانون لغزشی نمایی بصورت،

$$\dot{s}(t) = -qs(t) - \eta \operatorname{sgn}(s(t)) \quad (۳۷-۲)$$

معمول می باشد [۵۶]. برای بیان معادله (۳۷-۲) در حوزه زمان-گسسته با زمان نمونه برداری T ، معادله زیر به دست می آید:

$$s(k+1) = -Tqs(k) - T\varepsilon \operatorname{sgn}(s(k)) \quad q > 0, \varepsilon > 0 \quad (۳۸-۲)$$

با استفاده از شرط (۳۲-۲)، پایداری سیستم حلقه بسته به ازای،

$$|s(k)| > \frac{T\varepsilon}{2 - Tq} \quad 2 - Tq > 0 \quad (۳۹-۲)$$

برقرار خواهد بود. تفاوت اساسی قانون کنترل نمایی و ژائو در محدوده همگرایی و پارامتر q می باشد $(0 < q_{Exponential} < 2/T - 1)$ و $(0 < q_{Gao} < 1/T)$. بعلاوه محدوده همگرایی (۳۹-۲)، به راحتی قابل اثبات می باشد [۵۶]؛ اما ژائو هیچ اثبات پایداری برای رابطه (۳۵-۲) ارائه ننموده است.

در کنترل مد لغزشی زمان-پیوسته (۳۷-۲) و معادلات ژائو (۳۵-۲) و نمایی (۳۸-۲)، η و q دو عامل موثر در دامنه نوسانات می‌باشد. بطوری که با افزایش این دو پارامتر، سیستم حلقه بسته عملکردی مقاوم تر داشته و ردیابی به نحو مطلوب انجام می‌شود؛ اما این افزایش باعث تقویت دامنه نوسانات ناخواسته می‌گردد. وجه تمایز دو کنترل کننده (۳۷-۲) با (۳۵-۲) و (۳۸-۲)، به ازای زمان نمونه برداری T می‌باشد. در حالت گسسته در زمان، زمان نمونه‌برداری تأثیری بر دامنه نوسانات نداشته؛ اما با تغییر T می‌توان فرکانس نوسانات زمان-گسسته را تنظیم نمود، در حالی که برای حالت زمان-پیوسته، نوسانات سیگنال کنترلی قابل کنترل نمی‌باشد.

۳-۲- تفاوت اساسی کنترل مد لغزشی زمان-پیوسته و زمان-گسسته

در این بخش و با بیان یک مثال ساده، به تفاوت اساسی این روش در دو حوزه می‌پردازیم. سیستم مرتبه اول خطی پیوسته در زمان زیر را در نظر بگیرید:

$$\dot{x}(t) = -ax(t) \quad (۴۰-۲)$$

۱- بررسی پایداری سیستم در حوزه زمان-پیوسته: اگر سطح لغزش را برای سیستم فوق بصورت

$$s(t) = e(t) = x(t) - x_d(t) = x(t) - 0$$

$$V(t) = 0.5s^2(t) \quad (۴۱-۲)$$

و با مشتق گیری از رابطه (۴۱-۲)، می‌توان نوشت:

$$\dot{V}(t) = s(t)\dot{s}(t) = -ax^2(t) \leq 0 \quad (۴۲-۲)$$

در نتیجه سیستم زمان-پیوسته (۴۲-۲)، پایدار است.

۲- بررسی پایداری سیستم زمان- گسسته: حال به بررسی پایداری سیستم در حالت گسسته در

زمان می پردازیم. با استفاده از نگهدار مرتبه صفر، معادل گسسته در زمان سیستم (۲-۴۰)

را می توان به صورت زیر نوشت:

$$x(k+1) = (1-aT)x(k) \quad (۲-۴۳)$$

با تعریف سطح لغزش $s(k) = e(k) = x(k) - x_d(k) = x(k) - 0$ و تابع لیانوف:

$$V(k) = 0.5s^2(k) \quad (۲-۴۴)$$

سیستم زمانی پایدار مجانبی خواهد بود که $\Delta V(k) = V(k+1) - V(k) \leq 0$ باشد. در نتیجه داریم:

$$\Delta V(k) = 0.5e^2(k+1) - 0.5e^2(k) \quad (۲-۴۵)$$

$$\Delta V(k) = 0.5((1-aT)^2 - 1)x^2(k) \quad (۲-۴۶)$$

برای اینکه $\Delta V(k) \leq 0$ شود، باید رابطه زیر برقرار باشد:

$$(1-aT)^2 - 1 < 0 \xrightarrow{T>0} T < \frac{2}{a} \quad (۲-۴۷)$$

با مقایسه رابطه (۲-۴۲) با روابط (۲-۴۷)، به این نتیجه می رسیم که سیستم زمان- پیوسته پایدار، در حوزه گسسته در زمان به شرطی پایدار خواهد بود که زمان نمونه برداری مناسب انتخاب گردد. در واقع زمان نمونه برداری می تواند عامل ناپایداری سیستم پایدار باشد.

توجه شود که اگر معیار پایداری را بصورت زیر تعریف کنیم:

$$\Delta V(k) = 0.5s(k)[s(k+1) - s(k)] \quad (۲-۴۸)$$

$$\Delta V(k) = -0.5aTx^2(k) \quad (۴۹-۲)$$

از رابطه فوق مشاهده می‌شود که معیار (۴۸-۲)، بدون هیچ محدودیت نمونه‌برداری پایداری سیستم را نتیجه می‌دهد. لذا این معیار منطقی نمی‌باشد. بنابراین معیار پایداری $\Delta V(k) = s(k)[s(k+1) - s(k)]$ ، تحلیل درستی از پایداری سیستم را ارائه نمی‌دهد. اما رابطه $\Delta V(k) = s^2(k)$ یا $\Delta V(k) = s^2(k+1) - s^2(k)$ ، معیار پایداری مناسبی را برای سیستم زمان-گسسته می‌باشد.

۴-۲- مزایا و قابلیت‌های کنترل مد لغزشی

در پایان، یک جمع‌بندی کلی از ویژگی‌های روش SMC را بیان می‌کنیم. از مزایای و قابلیت‌های کنترل مد لغزشی می‌توان به موارد ذیل اشاره کرد:

۱- قابلیت طراحی برای اکثر سیستم‌ها از قبیل سیستم‌های خطی و غیرخطی، سیستم‌های چند

ورودی-چند خروجی^۱، سیستم‌های گسسته در زمان، سیستم‌های تصادفی^۲.

۲- مقاوم بودن در برابر تغییرات پارامترها، اغتشاشات و عدم قطعیت‌های سیستم تحت کنترل.

علیرغم این قابلیت‌های کنترل مد لغزشی، این روش دارای معایبی نیز می‌باشد. که از آن جمله می‌توان موارد زیر را نام برد:

۱- محاسبه قانون کنترل معادل، به دلیل وابسته بودن به مدل ریاضی سیستم، امری دشوار می‌-

باشد و یکی از معایب اصلی این روش است.

¹ Multi-input Multi-output

² Stochastic systems

۲- با توجه به اینکه ورودی کنترل به علامت متغیرهای اندازه‌گیری شده در نزدیکی صفر (معمولا سطح لغزش) بستگی دارد، لذا SMC نسبت به نویز اندازه‌گیری حساس می‌باشد.

۳- در روش SMC، ماهیت کلیدزنی قانون کنترل منجر به بروز پدیده نوسانات فرکانس بالا^۱ می‌گردد. بطوریکه این قبیل نوسانات فرکانس بالا، دینامیک‌های فرکانس بالا و مدل نشده سیستم را تحریک کرده و ممکن است باعث آسیب رساندن به سیستم تحت کنترل شود.

۴- SMC ممکن است برای غلبه بر عدم قطعیت‌های پارامتری از یک سیگنال کنترلی بزرگ استفاده کند که این امر ممکن است باعث اشباع محرک‌ها گردد.

همه مزیت و معایب مذکور برای DSMC نیز برقرار است. اما برتری اساسی DSMC نسبت به SMC، پیاده‌سازی کم هزینه آن می‌باشد.

برای غلبه بر مشکلات مذکور در SMC و DSMC، راهکارهای متفاوتی پیشنهاد شده است، از این قبیل می‌توان به استفاده از یک لایه مرزی در اطراف سطح لغزش، یا استفاده از بهره تطبیقی برای کاهش نوسانات، و یا استفاده از شبکه‌های عصبی و فازی برای کاهش وابستگی به مدل و نوسانات ناخواسته می‌باشد که در فصل‌های بعد به تشریح آن‌ها می‌پردازیم.

^۱ chattering

فصل سوم: دینامیک بازوی سه رابط SCARA

در این فصل نحوه استخراج معادلات دینامیکی بازوی ربات و بطور خاص بازوی سه رابط اسکارا که مورد مطالعه در این تحقیق است، بررسی می‌شود. همچنین به علت حضور مولفه اصطکاک در معادله دینامیکی، خصوصیات و اثرات نیروی اصطکاک نیز شرح داده می‌شود.

۳-۱- دینامیک بازوی ربات

برای کنترل دقیق و سریع بازوی ربات باید مدل دینامیکی آن در دسترس باشد. در حالت کلی

دو روش برای به دست آوردن معادلات بازوی ماهر وجود دارد [۵۷]:

۱- فرمولاسیون لاگرانژ- اویلر^۱: این روش بر پایه مفهوم مختصات تعمیم یافته، انرژی و نیروی تعمیم یافته استوار است. مزیت این روش در این است که هر جمله در فرمول بسته نهایی، تعبیر فیزیکی ساده‌ای دارد. به عنوان نمونه در معادله نهایی جملات: ماتریس اینرسی، گرانش، اصطکاک و نیروهای کرویولیس و مرکزگرا ظاهر می‌شوند.

۲- فرمولاسیون بازگشتی نیوتون- اویلر^۲: این روش برای پیاده سازی کامپیوتری مفید بوده و با افزایش محورها، از لحاظ محاسباتی نسبت به فرمولاسیون لاگرانژ- اویلر کارایی بهتری دارد.

^۱ Lagrange-Euler formulation

^۲ Recursive Newton-Euler formulation

۳-۱-۱- معادلات لاگرانژ

سیستم‌های پیچیده از لحاظ دینامیکی را می‌توان با استفاده از روش متداول فرمولاسیون لاگرانژ^۱ مدل نمود. این فرمولاسیون بر پایه مختصات تعمیم یافته، انرژی و نیروهای تعمیم یافته استوار است. برای بازوی ربات n -محوری، برداری از n مفصل با متغیرهای q ، دسته مناسبی از مختصات تعمیم یافته می‌باشد. توجه شود که برای مفاصل لولایی^۲، q نشان دهنده زاویه مفصل و برای مفاصل کشویی^۳ نشان دهنده فاصله مفصل از مفاصل کشویی است. فرض کنید K و V به ترتیب نشان دهنده انرژی جنبشی و انرژی پتانسیل بازوی ربات باشند. بنابراین ما تفاضل انرژی جنبشی و پتانسیل را به عنوان تابع لاگرانژین به صورت زیر تعریف می‌کنیم:

$$L(q, \dot{q}) = K(q, \dot{q}) - V(q) \quad (۱-۳)$$

که در آن $\dot{q} = dq/dt$ بردار سرعت مفاصل می‌باشد. توجه شود که انرژی جنبشی K به موقعیت و سرعت بازو وابسته است، در حالی که انرژی پتانسیل V فقط به موقعیت بازو وابسته می‌باشد. معادلات کلی حرکت بازوی ربات را با استفاده از تابع لاگرانژین می‌توان به صورت زیر نوشت:

$$\frac{d}{dt} \frac{\partial}{\partial \dot{q}_i} L(q, \dot{q}) - \frac{\partial}{\partial q_i} L(q, \dot{q}) = F_i \quad 1 \leq i \leq n \quad (۲-۳)$$

که در این فرمول F_i نیروی تعمیم یافته^۴ اعمالی به مفصل i -ام می‌باشد. فرمولاسیون لاگرانژ بازوی ربات، شامل n معادله دیفرانسیلی غیرخطی مرتبه دو از بردار متغیرهای مفاصل q است. برای

^۱ Lagrangian formulation

^۲ Revolute joint

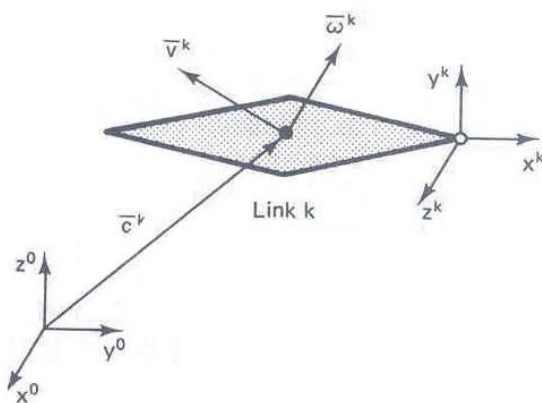
^۳ Prismatic joint

^۴ Generalized force

مشخص کردن جزئیات بیشتر این معادلات، باید معادلات مربوط به انرژی جنبشی K ، انرژی پتانسیل V و نیروی تعمیم F بررسی شوند.

۳-۱-۲- انرژی جنبشی و پتانسیل

محاسبه مجموع انرژی جنبشی $K(q, \dot{q})$ در تابع لاگرانژین بازوی ربات، پیچیده‌ترین قسمت می‌باشد. برای بدست آوردن عبارت توصیف کننده انرژی جنبشی، حرکت رابط k -ام بازوی ربات در شکل ۱-۳ نشان داده شده است.



شکل ۱-۳- حرکت رابط k -ام [۵۷]

در شکل فوق، رابط k -ام هم با سرعت خطی و هم با سرعت زاویه‌ای در فضای سه بعدی حرکت خواهد کرد. فرض کنید که $\bar{v}^k \in R^3$ سرعت خطی مرکز جرم رابط k -ام، و $\bar{\omega}^k \in R^3$ نشان دهنده سرعت زاویه‌ای اطراف رابط k -ام (با توجه به شکل آن) باشد. بنابراین مجموع انرژی جنبشی حاصل از جمع انرژی جنبشی رابط‌ها به صورت زیر در می‌آید:

$$K(q, \dot{q}) = \sum_{k=1}^n \frac{(\bar{v}^k)^T m_k \bar{v}^k + (\bar{\omega}^k)^T D_k \bar{\omega}^k}{2} \quad (۳-۳)$$

در این رابطه m_k نشان دهنده جرم بازوی k -ام و D_k تانسور لختی 3×3 رابط k -ام در مرکز جرم آن می‌باشد. برای به دست آوردن معادلات حرکت بازو باید انرژی جنبشی را به صورت تابعی از q و $\dot{q}$ بیان کنیم، در نتیجه باید تانسور لختی بررسی گردد.

۳-۱-۳- نیروی تعمیم یافته^۱

نیروهای تعمیم یافته، نیروهای باقیمانده‌ای هستند که به منظور مهار نیروهای مرتبط با انرژی پتانسیل (نیروهای گرانشی) و انرژی جنبشی (نیروهای داخلی) بر روی بازوی ربات عمل می‌کنند. نیروی تعمیم یافته می‌تواند با استفاده از کار تولید شده ناشی از جابجایی مجازی^۲ تعریف شود. جابجایی مجازی در واقع جابجایی کوچک لحظه ای δq می‌باشد. این جابجایی باید محدودیت‌های هندسی تحمیل شده توسط اتصالات بازو را ارضاء نموده ولی در محدودیت‌های گذرا صدق نکند. چون فرض بر این است که مستقل از گذر زمان رخ می‌دهد. کار تولید شده ناشی از جابجایی مجازی، کار مجازی^۳ نامیده می‌شود و با δW نمایش داده می‌شود. نیروی مربوط به جابجایی مجازی که کار مجازی تولید می‌کند را نیروی تعمیم یافته نامند. بنابراین نیروی تعمیم یافته F ، یک بردار $n \times 1$ ضمنی خواهد بود که با معادله زیر تعریف می‌شود:

$$\delta W = F^T \delta q \quad (۴-۳)$$

توجه شود که بسته به نوع مفاصل، مولفه‌های بردار نیروی تعمیم یافته، نیرو (برای مفصل کشویی) یا گشتاور (برای مفصل لولایی) خواهند بود.

^۱ Generalized force

^۲ Virtual displacement

^۳ Virtual work

۳-۱-۴- محرک‌ها^۱

هر مفصل ربات به وسیله محرکی از نوع موتور الکتریکی یا سیلندر هیدرولیکی و یا پنوماتیکی رانده می‌شود. فرض کنید $\tau \in \mathbb{R}^n$ ، نشان دهنده بردار گشتاور مفصل تولید شده توسط n محرک باشد. برای سهولت τ_k را "گشتاور" می‌نامیم، اما در واقع τ_k برای مفصل چرخان "گشتاور" و برای مفاصل کشویی "نیرو" می‌باشد (برای مفصل k -ام). بنابراین اثرات دینامیک‌های محرک‌ها، چرخ دنده‌ها و ابزار انتقال نیروی مکانیکی در τ خلاصه شده است. کار مجازی تولید شده به وسیله این گشتاور محرک به صورت زیر بیان می‌شود:

$$\delta W_1 = \tau^T \delta q \quad (۵-۳)$$

با مقایسه معادله (۵-۳) و (۴-۳)، می‌بینیم که $F = \tau$ می‌باشد. بنابراین در این مورد نیروی تعمیم یافته با گشتاور محرک برابر می‌باشد؛ اما این زمانی صادق است که سیستم دارای اصطکاک نباشد.

۳-۱-۵- اصطکاک

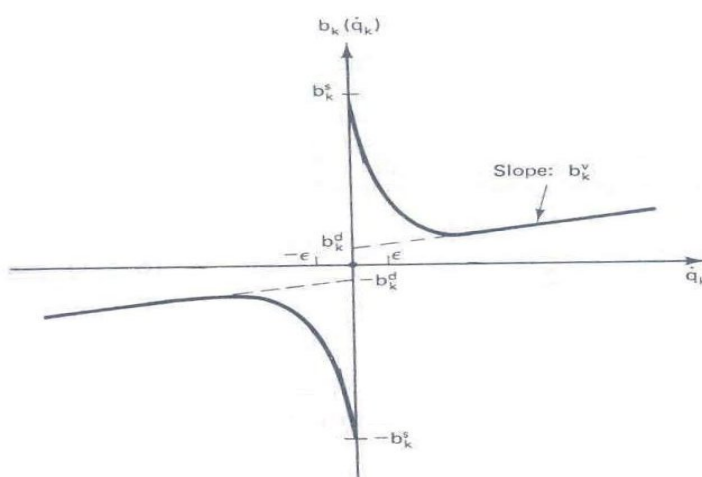
اثر نیروی اصطکاک^۲ به عنوان یک نیروی تعمیم یافته می‌تواند در مدلسازی بازوی ربات در نظر گرفته شود. اصطکاک نیروی پچیده غیرخطی است که مدلسازی دقیق آن دشوار بوده و در عین حال در بسیاری از موارد تاثیر قابل توجهی در دینامیک بازوی ربات دارد. سه جزء مجزای اصطکاک با استفاده از مدل نیروی اصطکاک برای مفصل k -ام به شکل زیر بیان می‌شوند [۵۷-۵۸]:

$$b_k(\dot{q}) = b_k^v \dot{q}_k + \text{sgn}(\dot{q}_k) \left[b_k^d + (b_k^s - b_k^d) \exp \frac{-|\dot{q}_k|}{\varepsilon} \right] \quad 1 \leq k \leq n \quad (۶-۳)$$

^۱ Actuators

^۲ Frictional force

در این رابطه sgn نشان دهنده تابع سیگنوم می‌باشد. قسمت اول در معادله (۶-۳) اصطکاک چسبندگی^۱ را نشان می‌دهد که در آن b_k^v ضریب اصطکاک چسبندگی برای مفصل k - ام است. قسمت دوم معادله (۶-۳) نشان دهنده اصطکاک دینامیک^۲ بوده و b_k^d ضریب نیروی دینامیک مفصل k - ام می‌باشد. و در نهایت، قسمت سوم معادله (۶-۳) اصطکاک استاتیکی^۳ یا اصطکاک کولمب^۴ را نمایش می‌دهد و در آن b_k^s ضریب نیروی استاتیکی مفصل k - ام و ϵ پارامتر مثبت کوچک می‌باشد. با توجه به معادله (۶-۳) زمانی که سرعت مفصل k - ام به صفر نزدیک می‌شود، نیروی اصطکاکی به $\pm b_k^s$ نزدیک خواهد شد. در نتیجه b_k^s می‌تواند به عنوان گشتاور یا نیروی مورد نیاز برای غلبه بر اصطکاک (زمانی که مفصل شروع به حرکت می‌نماید) تفسیر شود. همچنین واضح است که اصطکاک چسبندگی تابع خطی از q می‌باشد در حالی که اصطکاک دینامیک و استاتیک توابع غیرخطی و ناپیوسته از سرعت هستند. مدل نیروی اصطکاک به طور خلاصه در شکل ۲-۳ نشان داده شده است.



شکل ۲-۳- مدل اصطکاک مفصل k - ام [۵۷]

¹ Viscous friction

² Dynamic friction

³ Static friction

⁴ Coulomb

برای بازوهای رباتی که به خوبی طراحی شده اند، به خصوص بازو با تحریک مستقیم، ضرایب اصطکاک کوچکتر خواهند بود. توجه شود که معادله (۳-۶) مدل خشنی^۱ از نیروهای اصطکاک می باشد. نمایش جامعتری از نیروی اصطکاک می تواند به فرم $b(q, \dot{q})$ باشد که به موقعیت مفصل و سرعت آن نیز وابسته است. این نوع نمایش زمانی که برای مثال محور چرخان بازوی ربات به طور کامل دوار و یکنواخت نباشد، ممکن است ضروری باشد. کار مجازی انجام شده توسط ربات به واسطه نیروی اصطکاک مخالف حرکت به صورت زیر است:

$$\delta W_2 = -b(\dot{q})^T \delta q \quad (۳-۷)$$

نیروی کلی از ترکیب گشتاور محرک و نیروی اصطکاک به دست خواهد آمد. بنابراین از معادله (۳-۵) و (۳-۷) کار مجازی تولید شده توسط محرک و اصطکاک عبارت است از:

$$\delta W = \delta W_1 + \delta W_2 = [\tau - b(\dot{q})]^T \delta q \quad (۳-۸)$$

با مقایسه معادله (۳-۴) و (۳-۸) رابطه نیروی جامع به کار رفته برای بازوی ربات به فرم زیر به دست می آید:

$$F = \tau - b(\dot{q}) \quad (۳-۹)$$

به علت آنکه نیروی اصطکاک با حرکت تولید شده به وسیله محرک مخالفت می کند، لذا در معادله (۳-۹) علامت منفی به خود می گیرد.

۳-۲- معادلات حرکت

در این قسمت معادلات حرکت با در نظر گرفتن دو شرط خاص به دست می آیند [۵۹]. شرط اول اینکه انرژی جنبشی یک تابع درجه دوم از بردار $\dot{q}$ مطابق زیر باشد:

^۱ Rough model

$$K = \frac{1}{2} \sum_{i,j}^n M_{ij}(q) \dot{q}_i \dot{q}_j = \frac{1}{2} \dot{q}^T M(q) \dot{q} \quad (۱۰-۳)$$

که در آن $q \in \mathbb{R}^n$ و $M(q)$ ماتریس اینرسی $n \times n$ مثبت معین متقارن است. شرط دوم این است که انرژی پتانسیل مستقل $\dot{q}$ از باشد. البته باید توجه داشت که بازوهای ماهر رباتیک این شرط را برآورده می‌سازند. برای لاگرانژین L داریم:

$$L = K - V = \frac{1}{2} \sum_{i,j} M_{ij}(q) \dot{q}_i \dot{q}_j - V(q) \quad (۱۱-۳)$$

در نتیجه با استفاده از معادله (۱۱-۳)، معادلات حرکت به فرم زیر تعیین می‌شوند:

$$\frac{\partial L}{\partial \dot{q}_k} = \sum_j M_{kj}(q) \dot{q}_j \quad (۱۲-۳)$$

که در آن $k = 1, \dots, n$ و

$$\frac{d}{dt} \frac{\partial L}{\partial \dot{q}_k} = \sum_j M_{kj}(q) \ddot{q}_j + \sum_j \frac{d}{dt} M_{kj}(q) \dot{q}_j = \sum_j M_{kj}(q) \ddot{q}_j + \sum_{i,j} \frac{\partial M_{kj}}{\partial q_i} \dot{q}_i \dot{q}_j \quad (۱۳-۳)$$

همچنین برای جمله دیگر معادله حرکت می‌توان نوشت:

$$\frac{\partial L}{\partial q_k} = \frac{1}{2} \sum_{i,j} \frac{\partial M_{kj}}{\partial q_i} \dot{q}_i \dot{q}_j - \frac{\partial V}{\partial q_k} \quad (۱۴-۳)$$

بنابراین برای معادلات اوایلر- لاگرانژ خواهیم داشت:

$$\sum_j M_{kj}(q) \ddot{q}_j + \sum_{i,j} \left\{ \frac{\partial M_{ij}}{\partial q_i} - \frac{1}{2} \frac{\partial M_{kj}}{\partial q_k} \right\} \dot{q}_i \dot{q}_j + \frac{\partial V}{\partial q_k} = \tau_k \quad (۱۵-۳)$$

که در آن τ نشان دهنده گشتاور ورودی است. با تغییر ترتیب در جمع و استفاده از خاصیت تقارن می توان نشان داد:

$$\sum_{i,j} \left(\frac{\partial M_{kj}}{\partial q_i} \right) \dot{q}_i \dot{q}_j = \frac{1}{2} \sum_{i,j} \left\{ \frac{\partial M_{kj}}{\partial q_j} + \frac{\partial M_{ki}}{\partial q_j} \right\} \dot{q}_i \dot{q}_j \quad (۱۶-۳)$$

پس

$$\sum_{i,j} \left(\frac{\partial M_{kj}}{\partial q_i} - \frac{1}{2} \frac{\partial M_{i,j}}{\partial q_k} \right) \dot{q}_i \dot{q}_j = \frac{1}{2} \sum_{i,j} \left\{ \frac{\partial M_{kj}}{\partial q_j} + \frac{\partial M_{ki}}{\partial q_j} - \frac{\partial M_{ij}}{\partial q_k} \right\} \dot{q}_i \dot{q}_j \quad (۱۷-۳)$$

حال تعریف می کنیم:

$$C_{ijk} = \frac{1}{2} \left\{ \frac{\partial M_{kj}}{\partial q_j} + \frac{\partial M_{ki}}{\partial q_j} - \frac{\partial M_{ij}}{\partial q_k} \right\} \quad (۱۸-۳)$$

عبارات به فرم معادله (۱۸-۳) را نمادهای کریستوفل^۱ می نامند. باید توجه کرد که برای یک ثابت k داریم $C_{ijk} = C_{jik}$. حال اگر تعریف نماییم

$$G_k = \frac{\partial V}{\partial q_k} \quad (۱۹-۳)$$

که در این حالت G بردار جاذبه می باشد. در نهایت می توانیم معادلات اوایلر- لاگرانژ را بصورت زیر بنویسیم:

$$\sum_j M_{kj}(q) \ddot{q}_j + \sum_{i,j} C_{ijk}(q) \dot{q}_i \dot{q}_j + G_k(q) = \tau_k, \quad k = 1, \dots, n \quad (۲۰-۳)$$

^۱ Christoffel symbols

در سمت چپ معادله فوق، اگر در عبارت دوم داشته باشیم $i = j$ آنگاه عباراتی با ضریب q_i^2 تشکیل می شود که جانب مرکز نامیده می شوند. و اگر $i \neq j$ باشد، آنگاه عباراتی با ضریب $q_i q_j$ تشکیل می شود که بخش های کوریولیس نام دارند. معادله (۳-۲۰) را می توان به شکل ماتریسی زیر نشان داد:

$$M(q)\ddot{q} + C(q, \dot{q})\dot{q} + G(q) = \tau \quad (۲۱-۳)$$

که در آن $G, \tau, q \in \mathbb{R}^n$ و $M, C \in \mathbb{R}^{n \times n}$. عنصر k, j -ام ماتریس C به صورت زیر تعریف می شود

$$C_{ijk} = \sum_{i=1}^n C_{ijk}(q)\dot{q}_i = \sum_{i=1}^n \frac{1}{2} \left\{ \frac{\partial D_{kj}}{\partial q_j} + \frac{\partial D_{ki}}{\partial q_j} - \frac{\partial D_{ij}}{\partial q_k} \right\} \dot{q}_i \quad (۲۲-۳)$$

قضیه زیر یک رابطه مهم را بین ماتریس M اینرسی و ماتریس C بیان می کند.

قضیه ۳-۱- اگر تعریف کنیم $N = \dot{M}(q) - 2C(q, \dot{q})$ ، آنگاه N یک ماتریس پادمتقارن است که عناصر n_{jk} آن در رابطه $n_{jk} = -n_{kj}$ صدق می کنند.

اثبات: عنصر kj -ام ماتریس $\dot{M}$ طبق قاعده زنجیره ای بصورت زیر است:

$$\dot{M}_{kj} = \sum_{i=1}^n \frac{\partial M_{kj}}{\partial q_i} \dot{q}_i \quad (۲۳-۳)$$

بنابراین عنصر kj -ام ماتریس N می شود:

$$\begin{aligned} n_{kj} = \dot{M}_{kj} - 2C_{kj} &= \sum_{i=1}^n \left[\frac{\partial M_{kj}}{\partial q_i} - \left\{ \frac{\partial M_{kj}}{\partial q_j} + \frac{\partial M_{ki}}{\partial q_j} - \frac{\partial M_{ij}}{\partial q_k} \right\} \right] \dot{q}_i \\ &= \sum_{i=1}^n \left[-\frac{\partial M_{ki}}{\partial q_j} + \frac{\partial M_{ij}}{\partial q_k} \right] \dot{q}_i \end{aligned} \quad (۲۴-۳)$$

از آنجا که ماتریس M متقارن است، پس $M_{ij} = M_{ji}$ ، بنابراین با جابجایی k و j در (۲۴-۳) نتیجه می شود:

$$n_{jk} = -n_{kj} \quad (۳-۲۵)$$

که اثبات را کامل می‌نماید. □

مطالب ارائه شده در این بخش در مورد هر سیستم مکانیکی که انرژی جنبشی آن به فرم (۳-۱) و انرژی پتانسیل آن مستقل از $\dot{q}$ باشد قابل اجرا است.

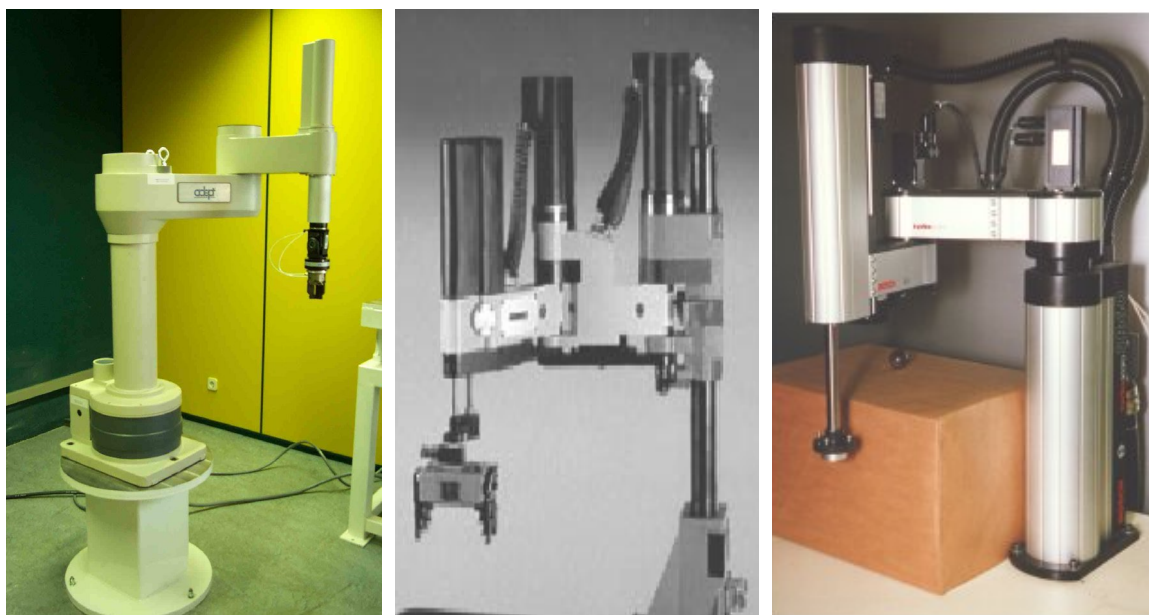
۳-۳- ویژگی‌ها و معادلات بازوی سه رابط اسکارا

اسکارا^۱ مخفف "بازوی رباتیک ماهر منعطف با قابلیت (سهولت) مونتاژ" می‌باشد. ربات اسکارا یک ترکیب جدید و مردم پسند است. و در سال ۱۹۸۱ میلادی ربات اسکارا با هدایت پروفسور "هیروشی مکینو" استاد دانشگاه یاماناشی ساخته شد. بازوی اسکارا دارای ساختار RRP می‌باشد. از مفاصل اول و دوم برای تعیین مختصات X و Y، و از مفصل سوم برای تعیین مختصات Z استفاده می‌شود. ربات اسکارا دارای قدرت مانور بسیاری در صفحه XY بوده و به همین علت کاملاً مناسب مونتاژ کاری می‌باشد. از این ربات علاوه بر مونتاژ در کارهای جابجایی اجسام نیز بهره‌برداری می‌شود. ویژگی ربات اسکارا اتصال دو رابط لولایی پی در پی با محورهای موازی، شبیه دست انسان می‌باشد. به همین دلیل به آن هنرمند می‌گویند. این ویژگی به ربات این قدرت را می‌دهد که در مساحتی محدود، باز و جمع شود. و به سرعت خود را از نقطه‌ای به نقطه دیگر برساند. این ربات تنها بر یک پایه محکم شده و برای نصب به جای کمتری نیاز دارد و به راحتی آماده کار می‌شود. شکل ۳-۳ مدل‌های متفاوتی از ربات اسکارای سه و چهار محوری را نشان می‌دهد. به عنوان مثال در این شکل از سمت چپ، بازوی اسکارای چهار-محوری (ماهر اول^۲) را نشان می‌دهد. این ربات اولین نمونه از نوع ربات‌های تجاری با عملگرهای تحریک مستقیم می‌باشد. چرخنده و مبدل‌های تبدیل توان در آن استفاده نشده است. در عوض، موتورهای DC بدون جاروبک با سرعت پایین و گشتار بالا برای درایو مستقیم مفاصل در

¹ Selective compliant articulated robot for assembly OR Selective compliance assembly robot arm

² Adept one

آن به کار رفته است. در نتیجه از اصطکاک و لقی چرخنده خبری نیست، و عملکرد سریع و دقیقی دارد [۵۷].



شکل ۳-۳- از راست به چپ: ۱- ربات Bosch Turboscara SR 60 [۶۰] ۲- ربات ICOMATIC03 با سه درجه آزادی [۶۱] ۳- ربات Adept one چهار محوری [۶۲ و ۵۷]

بازوی سه رابط اسکارا نشان داده شده در شکل ۳-۴ را در نظر بگیرید. این شکل شامل سه مفصل است، مفصل اول و دوم از نوع لولایی و مفصل سوم از نوع کشویی می‌باشند. معادلات دینامیکی ربات اسکارا سه رابط را با استفاده از معادلات لاگرانژ- اویلر می‌توان بصورت زیر نوشت [۶۳ و ۵۷]:

$$\begin{bmatrix} M_{11} & M_{12} & M_{13} \\ M_{21} & M_{22} & M_{23} \\ M_{31} & M_{32} & M_{33} \end{bmatrix} \begin{bmatrix} \ddot{q}_1 \\ \ddot{q}_2 \\ \ddot{q}_3 \end{bmatrix} + \begin{bmatrix} C_{11} & C_{12} & C_{13} \\ C_{21} & C_{22} & C_{23} \\ C_{31} & C_{32} & C_{33} \end{bmatrix} \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \dot{q}_3 \end{bmatrix} + \begin{bmatrix} g_1 \\ g_2 \\ g_3 \end{bmatrix} + \begin{bmatrix} b_1(\dot{q}) \\ b_2(\dot{q}) \\ b_3(\dot{q}) \end{bmatrix} = u(t) \quad (۲۶-۳)$$

که در آن مولفه‌های ماتریس‌های M ، C و G عبارتند از:

$$M_{11} = l_1^2 \left(\frac{m_1}{3} + m_2 + m_3 \right) + l_1 l_2 (m_2 + 2m_3) \cos(q_2) + l_2^2 \left(\frac{m_2}{3} + m_3 \right)$$

$$M_{12} = M_{21} = l_1 l_2 \left(\frac{m_2}{2} + m_3 \right) \cos(q_2) - l_2^2 \left(\frac{m_2}{3} + m_3 \right)$$

$$M_{22} = l_2^2 \left(\frac{m_2}{3} + m_3 \right)$$

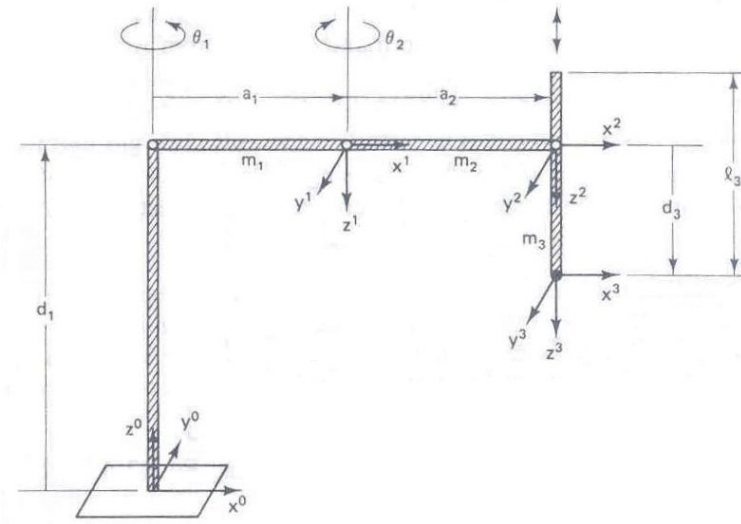
$$M_{13} = M_{23} = M_{31} = M_{32} = 0$$

$$M_{33} = m_3 \quad C_{11} = -\dot{q}_2 (m_2 + 2m_3) l_1 l_2 \sin(q_2)$$

$$C_{12} = -\dot{q}_2 \left(\frac{m_2}{2} + m_3 \right) l_1 l_2 \sin(q_2) = C_{21}$$

$$C_{22} = -\dot{q}_2 \left(\frac{m_2}{2} + m_3 \right) l_1 l_2 \sin(q_2)$$

$$C_{13} = C_{23} = C_{31} = C_{32} = C_{33} = 0 \quad g_1 = g_2 = 0 \quad g_3 = -m_3 g$$



شکل ۳-۴- ربات اسکارای سه محوری [۵۷]

در معادله (۲۶-۳)، بردار گشتاور ورودی، q_1, q_2 و q_3 زوایای مفصل ها، m_1, m_2 و m_3 جرم بازوها و l_1, l_2 و l_3 طول بازوهای رابط اول، دوم و سوم، $b(\dot{q})$ بردار اصطکاک و $u(t)$ بردار گشتاور ورودی می باشند. نحوه بدست آوردن معادله (۲۶-۳)، بطور کامل در [۵۷] ذکر شده است.

مدل دینامیکی ربات اسکارا سه رابط در معادله (۲۶-۳)، در مقایسه با مدل دینامیکی بازوی دو رابط نسبتاً پیچیده می باشد. دلیل این پیچیده گی، وجود بازوی سوم با حرکت کشویی می باشد. با توجه به معادله (۲۶-۳)، واضح است که حرکت مفصل سوم به طور کامل از مفاصل یک و دو مستقل می باشد. حرکت مفاصل یک و دو به جرم بازوی سوم وابسته اند. در واقع بازوی سوم مانند بار برای مفصل یک و دو عمل می کند [۵۷].

اگر بازوی ربات تحت اغتشاش خارجی نیز قرار بگیرد، در این حالت معادله دینامیکی به صورت زیر خواهد بود [۶۳]:

$$\begin{bmatrix} M_{11} & M_{12} & M_{13} \\ M_{21} & M_{22} & M_{23} \\ M_{31} & M_{32} & M_{33} \end{bmatrix} \begin{bmatrix} \ddot{q}_1 \\ \ddot{q}_2 \\ \ddot{q}_3 \end{bmatrix} + \begin{bmatrix} C_{11} & C_{12} & C_{13} \\ C_{21} & C_{22} & C_{23} \\ C_{31} & C_{32} & C_{33} \end{bmatrix} \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \dot{q}_3 \end{bmatrix} + \begin{bmatrix} g_1 \\ g_2 \\ g_3 \end{bmatrix} + F(\dot{q}) + t_d(t) = u(t) \quad (۲۷-۳)$$

در رابطه مذکور، $t_d(t)$ بردار اغتشاش خارجی و $b(\dot{q}) = F(\dot{q})$ بردار اصطکاک مفاصل می باشند.

وجود بردارهای اصطکاک در حالتی که مدل دقیقی از آن در دسترس نباشد، کنترل سیستم را پیچیده تر و سخت تر خواهد ساخت. بنابراین در فصل های بعد به طراحی کنترل کننده های مقاوم در برابر این عدم قطعیت ها پرداخته می شود.

فصل چهارم: کاربرد سیستم‌های فازی و شبکه‌های عصبی در شناسایی و

کنترل

در این فصل به دلیل اهمیت شبکه‌های عصبی و سیستم‌های فازی در علم شناسایی و کنترل، به مرور مختصری از ویژگی‌های آن‌ها می‌پردازیم. و همچنین برای نمایش قابلیت مدل‌سازی شبکه‌های عصبی یک مثال شبیه‌سازی با استفاده از نرم‌افزار MATLAB، در محیط M-file پیاده‌سازی می‌شود.

۱-۴-۱- شبکه‌های عصبی

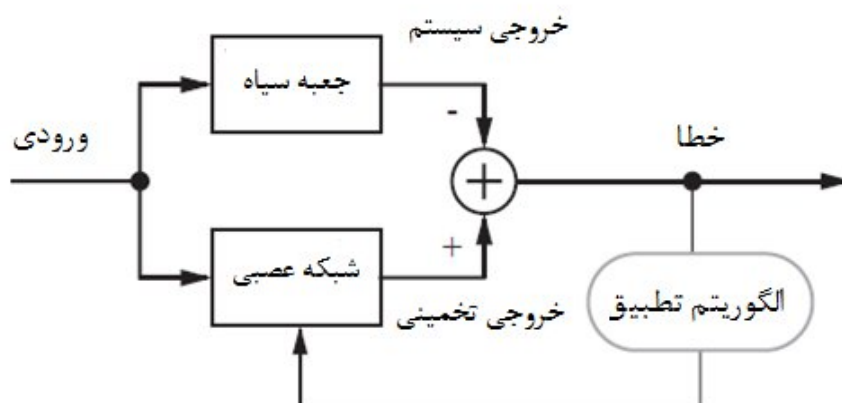
شبکه‌های عصبی در شناسایی و کنترل سیستم‌های دینامیک کاربرد بسیار موفقیت‌آمیزی داشته‌اند. به علت قابلیت مدل‌سازی، پرسپترون چندلایه کاربرد زیادی در مدل‌سازی سیستم‌های غیرخطی و کنترل کننده‌های غیرخطی داشته است. به عنوان نمونه: ۱- کنترل پیش‌بین مدل ۲- کنترل تطبیقی مدل مرجع ۳- خطی‌سازی فیدبکی ۴- کنترل تطبیقی مستقیم ۵- کنترل عصبی- مد لغزشی می‌توان اشاره کرد [۶۴-۶۵].

با توجه به خاصیت عمومی شبکه‌های عصبی به عنوان ابزار تقریب توابع، استفاده از شبکه در طراحی کنترل کننده امری طبیعی می‌باشد. خصوصاً زمانی که مدل ریاضی سیستم دینامیکی معلوم نباشد، شبکه‌های عصبی روش مناسبی برای طراحی کنترل کننده با استفاده از داده‌های ورودی و

خروجی را فراهم می‌کنند. به عبارت دیگر، شبکه عصبی می‌تواند به عنوان مدل جعبه سیاه برای سیستم به کار رود. همچنین به خاطر قابلیت یادگیری شبکه‌های عصبی، کنترل‌کننده‌هایی که با استفاده از شبکه طراحی می‌شوند، خاصیت تطبیقی دارند. و این برای تطبیق با تغییرات شرایط محیطی، مثلاً برای سیستم‌های متغیر با زمان مفید است. در عمل کنترل‌کننده‌های عصبی برای سیستم‌های تغییرپذیر با زمان رواج زیادی پیدا کرده است.

۲ ۴-۱-۱- کاربرد عمومی شبکه‌های عصبی:

- ۱- تقریب مدل دینامیکی سیستم، به منظور طراحی کنترل‌کننده مطلوب (شناسایی سیستم)
 - ۲- تقریب قانون کنترل اعمالی به سیستم تحت کنترل (کنترل سیستم)
 - ۳- تقریب مدل دینامیکی سیستم و تنظیم ضرایب کنترل‌کننده عصبی به طور همزمان (شناسایی و کنترل سیستم)
- یکی دیگر از زمینه‌هایی که شبکه‌های عصبی در آن کاربرد زیادی دارد، شناسایی سیستم می‌باشد. در روش‌های شناسایی کلاسیک، ابتدا مدل پیش‌فرضی در نظر گرفته شده و سپس شناسایی انجام می‌شود. و اگر مدل به دست آمده مطلوب نباشد دوباره با مدل پیش‌فرض جدیدی سیستم را شناسایی می‌کنیم. همچنین به علت فقدان تئوری جامع در مورد شناسایی سیستم‌ها در کنترل کلاسیک، شناسایی بصورت موردی انجام می‌شود. اما شناسایی با استفاده از شبکه‌های عصبی به صورت نسبتاً کلی بوده و برای فرایندهای متفاوت عملکرد قابل قبولی دارند.
- شکل ۳-۱ فرم کلی شناسایی سیستم‌های غیرخطی را نشان می‌دهد و در آن پارامترهای شبکه عصبی طوری تنظیم می‌شود که اختلاف خروجی مدل واقعی و شبکه کاهش یابد.



شکل ۴-۱- شناسایی سیستم با شبکه‌های عصبی [۶۵]

هدف از شناسایی فرایندها، به دست آوردن مدل دینامیکی سیستم، بدون داشتن اطلاعات قبلی از دینامیک سیستم می‌باشد که به شناسایی جعبه سیاه معروف است. الگوریتم یادگیری متداول در مدل‌سازی با شبکه، گرادیان کاهشی می‌باشد (الگوریتم پسانتشار خطا)^۱. ممکن است که شبکه عصبی آموزش دیده به ازای داده‌های جدید خروجی مطلوبی نداشته باشد اما در عمل کارایی مطلوبی برای سیستم‌های دینامیکی خوش-رفتار^۲ از خود نشان می‌دهند [۶۶].

دو روش متداول یادگیری شبکه عصبی عبارت‌اند از [۶۷]:

- ۱- یادگیری نظارتی^۳: در این حالت خروجی شبکه با مقدار مطلوب (ناظر) مقایسه می‌گردد و اختلاف این دو مقدار برای تنظیم ضرایب شبکه استفاده می‌شود. این نوع آموزش کاربرد گسترده‌ای در شناسایی و تقریب قانون کنترل دارد.

^۱ Backpropagation

^۲ Well-defined

^۳ Supervised learning

۲- یادگیری غیر نظارتی^۱: در این روش یادگیری مقدار مطلوب (ناظر) موجود نمی‌باشد و یادگیری بر اساس قوانین رقابتی انجام می‌گردد. در مقایسه با آموزش نظارتی کاربرد کمتری در مباحث کنترل دارد.

۴-۱-۲- انواع شبکه‌های عصبی

ساختارهای مختلفی از شبکه‌های عصبی در کنترل و شناسایی کاربرد دارند که از مهمترین آنها می‌توان به موارد زیر اشاره کرد:

۱- شبکه پرسپترون چندلایه (MLP)

این شبکه یکی از گونه‌های متداول و پرکاربرد در زمینه‌های مختلف می‌باشد. پرسپترون چندلایه به صورت باناظر و با استفاده از الگوریتم پسانتشار^۲ آموزش می‌بیند. در این نوع شبکه بسته به نوع کاربرد می‌توان تعداد لایه‌ها را بیشتر یا کمتر انتخاب نمود. به عنوان مثال، در جایی که سرعت مدل‌سازی مطرح است تعداد لایه‌ها کم انتخاب می‌گردد. و در حالتی که حجم محاسباتی و سرعت آموزش مهم نباشد، می‌توان تعداد لایه‌ها را بیشتر گرفت. پرسپترون دولایه و سه‌لایه، دو نوع معمول شبکه MLP در کنترل سیستم‌ها می‌باشند.

۲- شبکه با توابع گوسین (RBFN):

شبکه RBF نیز یکی از شبکه‌های معمول و پرکاربرد می‌باشد. این شبکه ساختاری سه‌لایه دارد. لایه اول یا همان لایه ورودی، که کار پردازشی بر عهده ندارد. لایه دوم، نگاشت غیرخطی ورودی‌ها به لایه پنهان، با استفاده از توابع گوسین می‌باشد. و در نهایت لایه سوم، مجموع خطی از خروجی‌های لایه پنهان است. این شبکه در مقایسه با MLP از سرعت یادگیری بالاتری برخوردار بوده و احتمال

^۱ Unsupervised

^۲ Back-propagation

کمتری دارد که الگوریتم یادگیری آن در مینیمم محلی بیافتد. شبکه RBF همانند شبکه MLP جزء شبکه‌های پیشخور می‌باشد. نحوه آموزش آنلاین و ساختار شبکه‌های MLP و RBF در فصل بعد ذکر خواهد شد.

۳- شبکه عصبی پسخور یا بازگشتی^۱ (RNN)

شبکه RNN شامل فیدبک‌هایی از خروجی به ورودی یا لایه‌های میانی است. در واقع این نوع شبکه‌ها دارای خواص دینامیک بوده و به دلیل همین خاصیت دینامیک، این نوع شبکه‌ها در شناسایی و کنترل سیستم‌های متغیر با زمان کاربرد فراوانی دارد. شبکه هاپفیلد یکی از نمونه‌های معروف شبکه‌های پسخور می‌باشد. البته با فیدبک گرفتن از هر نوع شبکه پیشخور، شبکه بازگشتی ساخته می‌شود. لذا شبکه‌های پسخور می‌تواند ساختارهای متنوعی داشته باشد.

۴- شبکه عصبی با توابع موجک^۲ (WNN)

"wavelet" به معنی موج کوچک می‌باشد. این موج کوچک باید حداقل نوسانات را داشته باشد و به سرعت از طرف منفی و مثبت به سمت صفر میل کند [۶۸]. چند نمونه از توابع wavelet عبارت‌اند از [۶۹]:

۱- تابع کلاه مکزیکی^۳: این تابع به صورت مشتق دوم تابع چگالی احتمال گوسی تعریف می‌شود. همانطور که در شکل ۴-۲ نشان داده شده، بیشترین مقدار انرژی تابع مورد نظر در مبدا

متمرکز است:

$$\psi(x) = (1 - 2x^2) \exp(-x^2) \quad (۱-۴)$$

^۱ Recurrent neural network

^۲ Wavelet neural network

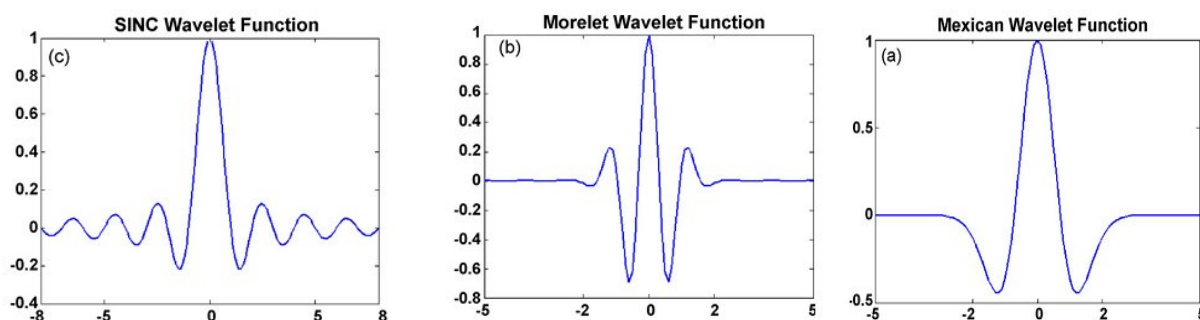
^۳ Mexican hat wavelet

۲- تابع مورلت^۱: تابع مورد نظر از توابع متناسب با تابع کساین و چگالی احتمال گوسی بدست می‌آید و همانند wavelet مکزیکی، بیشترین مقدار انرژی را در مبدا داراست:

$$\psi(x) = \exp(-x^2) \cos(5x) \quad (۲-۴)$$

۳- تابع سینک^۲ (شانون): این تابع از تابعی که متناسب با تابع کساین است، بدست می‌آید و بیشترین مقدار انرژی نیز در عریض‌ترین باند نزدیک به مبدا قرار دارد:

$$\psi(x) = \frac{\sin(\pi x)}{(\pi x)} \quad (۳-۴)$$



شکل ۲-۴: wavelet: از راست ۱- مکزیکی، ۲- مورلت، ۳- شانون [۶۹]

از ترکیب تئوری تبدیل wavelet با شبکه‌های عصبی پیشخور و پسخور، شبکه‌های جدید نگاشتی برای تقریب توابع بدست می‌آیند. این شبکه‌ها با ترکیباتی از قبیل WNN-MLP، WNN-RBFN و یا Neuro-Fuzzy Wavelet از کاربرد فراوانی در علم کنترل و شناسایی برخوردارند [۷۰].

^۱ Morlet wavelet

^۲ Sinc (Shannon) wavelet

۴-۱-۳- مراحل شناسایی و مدلسازی

مرحله اول: مرحله اول در شناسایی سیستم به دست آوردن داده‌های عملی و آزمایشگاهی، که در بازه کاملی از عملکرد سیستم قرار دارد، می‌باشد. در واقع در این مرحله باید به یک بینش فیزیکی از عملکرد سیستم برسیم تا بتوانیم داده‌های مطلوبی برای آموزش شبکه به دست آوریم [۶۶].

مرحله دوم: مرحله دوم انتخاب مدل شبکه عصبی می‌باشد. این انتخاب شامل انتخاب ورودی‌ها و خروجی‌های شبکه، انتخاب ساختار داخلی شبکه و بهره‌یادگیری می‌باشد. همانطور که قبلاً اشاره شد، سه نوع رایج خانواده شبکه‌های عصبی شامل: پرسپترون چند لایه، شبکه با توابع گوسین و شبکه‌های پس‌خور می‌باشند و در شناسایی و مدلسازی از قابلیت بالایی برخوردارند.

حال برای شناسایی، مدل سیستم را به صورت زیر در نظر بگیرید:

$$y(t) = g(x(t, \theta), \theta) + e(t) \quad (۴-۴)$$

که در آن $x(\theta, t)$ بردار رگرسیون و θ بردار شامل وزن‌ها می‌باشد. فرم بردار رگرسیون بصورت زیر می‌باشد:

$$x(\theta, t) = [y(t) \quad \cdots \quad y(t-n) \quad u(t-d) \quad \cdots \quad u(t-d-m) \quad e(t, \theta) \quad \cdots \quad e(t-k, \theta)]^T \quad (۵-۴)$$

بردار رگرسیون بسته به اطلاعاتی که از رفتار سیستم در دسترس است، انتخاب می‌شود. بنابراین مدل با استفاده از داده‌های به دست آمده از آزمایش آموزش می‌بیند.

مرحله سوم: صحت مدل به دست آمده باید بعد از آموزش تست شود. اگر نتایج به دست آمده راضی‌کننده نباشد، باید ساختار شبکه عصبی عوض شود. اگر داده‌های به دست آمده کافی نباشد، امکان پذیر نیست که با انتخاب مدل جدید بتوان به جواب رسید. در این حالت باید به جمع‌آوری داده‌های

زیادتر بپردازیم. بنابراین اگر ورودی‌هایی که بر رفتار سیستم اثر گذارند را درست نشناخته باشیم (در کنترل و مدلسازی) و یا تعداد نمونه‌ها کم باشد (در شناسایی الگو)^۱ شناسایی معتبر نخواهد بود.

۴-۱-۴- مثال شبیه‌سازی

سیستم غیرخطی زیر را در نظر بگیرید [۶۶]:

$$y_p(k+1) = f(y_p(k), y_p(k-1), \dots) + u(k) \quad (۴-۶)$$

در این رابطه $f(y_p(k), y_p(k-1), \dots)$ تابع غیرخطی که باید شناسایی گردد، و $u(k)$ ورودی سیستم است. تابع $f(y_p(k), y_p(k-1), \dots)$ نشان دهنده ساختار داخلی سیستم غیرخطی بوده و برای به دست آوردن $y_p(k+1)$ کاربرد دارد. واضح است که اگر بتوان پاسخ $y_p(k)$ سیستم غیرخطی را در محدوده ورودی $u(k) \in [\min \quad \max]$ و در ناحیه پایداری متغیرها نمونه‌برداری کرد، می‌توان از داده‌های به دست آمده با استفاده از شبکه‌های عصبی، مدل سیستم را شناسایی نمود. به عنوان مثال سیستم غیرخطی زیر را در نظر بگیرید:

$$y_p(k+1) = \frac{y_p(k)(y_p(k-1)+2)(y_p(k)+2.5)}{8.5+(y_p(k))^2+(y_p(k-1))^2} + u(k) \quad (۴-۷)$$

در سیستم فوق ناحیه عملکرد پایدار به ازای $u(k) \in [-2 \quad 2]$ می‌باشد. مدل مورد استفاده برای شناسایی در این مثال، شبکه عصبی دو لایه می‌باشد. مراحل کلی طراحی به طور خلاصه عبارتند از:

۱- شناسایی ورودی‌ها و خروجی‌های سیستم.

۲- نمونه‌برداری از ورودی‌ها و خروجی‌ها به منظور تولید داده.

۳- انتخاب نوع، ساختار و آموزش شبکه عصبی با یکی از الگوریتم‌های موجود.

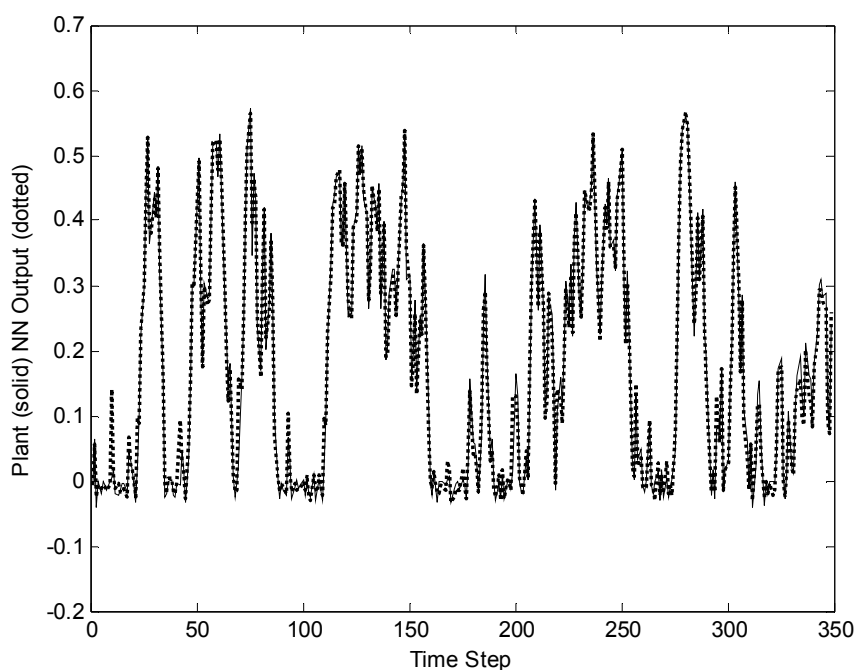
^۱ Pattern recognition

۴- تست مدل^۱ به ازای داده ها و ورودی های جدید. اگر مدل سازی مطلوب نباشد، در این حالت باید تعداد لایه ها، تعداد نرون ها، تابع فعالیت و یا ضریب یادگیری تغییر داده شود تا به جواب مطلوب برسیم.

مشخصات شبکه عصبی انتخابی: آموزش آفلاین، دو لایه، ۱۰ نرون در لایه اول و تک نرون خروجی،

۵۰۰ تا تکرار، دو ورودی $(y_p(k)$ و $(y_p(k-1)$ و تک خروجی $(\hat{f}(y_p(k), y_p(k-1), \dots))$.

شکل ۳-۴ مقدار واقعی و تقریبی تابع $f(y_p(k), y_p(k-1), \dots)$ را نشان می دهد. مشاهده می شود که مدل سازی بطور مطلوب انجام شده است. اما برای اینکه صحت و دقت مدل سازی بررسی شود، خروجی سیستم و شبکه عصبی به ازای سه ورودی با دامنه یا فرکانس های متفاوت، در شکل ۴-۴ رسم شده است.

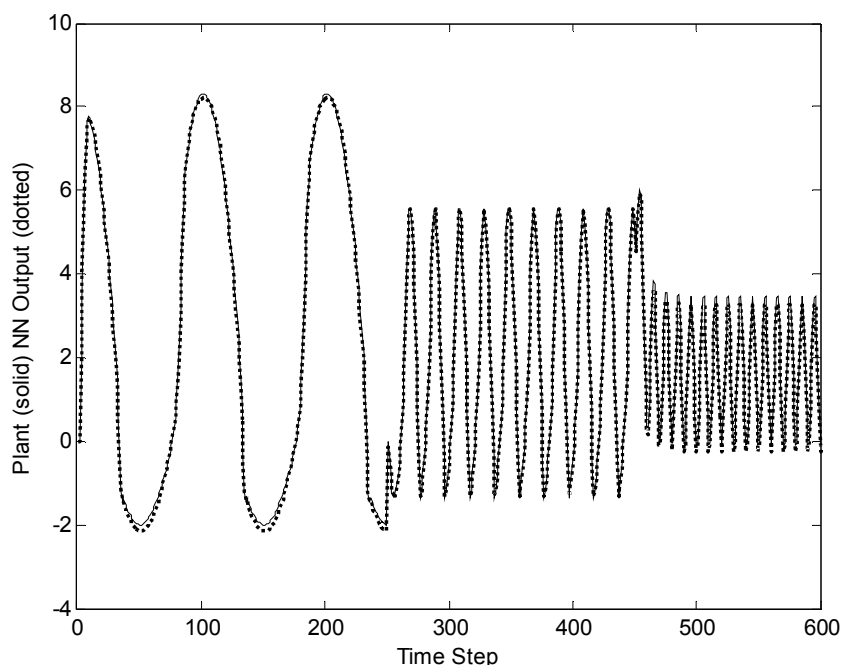


شکل ۳-۴- تابع f (خط) و تقریب آن $\hat{f}$ (نقطه چین) با شبکه دولایه

ورودی هایی که برای تست مدل سازی به کار رفته اند، عبارتند از:

¹ Model validation

$$\begin{cases} \text{if } 2 \leq k \leq 250 & \rightarrow u(k) = 2 \cos(0.02\pi k) \\ \text{if } 250 < k \leq 450 & \rightarrow u(k) = 1.2 \cos(0.1\pi k) \\ \text{if } 450 < k \leq 600 & \rightarrow u(k) = 1.2 \cos(0.2\pi k) \end{cases}$$



شکل ۴-۴- خروجی سیستم $y_p(k)$ (خط) و خروجی شبکه عصبی دولایه $\hat{y}_p(k)$ (نقطه چین)

از شکل فوق مشاهده می‌شود که به ازای ورودی‌های متفاوت، پاسخ‌های سیستم واقعی و شناسایی شده به هم نزدیک می‌باشند. در نتیجه مدل‌سازی به نحو مطلوبی انجام شده است.

زمانی که شناسایی به عنوان قسمتی از قانون کنترل انجام شود، در این حالت کنترل کننده، کنترل کننده تطبیقی نامیده می‌شود. و برای سیستم‌هایی با ویژگی‌های دینامیکی متغیر با زمان طراحی می‌شود. در این حالت شناسایی به صورت برخط (آنلاین) پیاده می‌شود. نکته‌ای که باید بدان توجه نمود این است که دقت شناساگر عصبی به مقدار اولیه وزن‌های شبکه و پارامتر یادگیری آن وابسته است. وزن‌ها را معمولاً مابین $[-0.5, 0.5]$ ، به صورت تصادفی انتخاب می‌کنند. اما برای انتخاب پارامتر یادگیری مطلوب، بهتر است که الگوریتم چندین بار به ازای پارامتر یادگیری مختلف تکرار گردد و بهترین جواب را از بین آنها انتخاب شود.

برنامه M-file شناسایی در پیوست (الف) ذکر گردیده است.

۴-۲- سیستم‌های فازی

سیستم‌های فازی نیز همانند شبکه‌های عصبی در پردازش سیگنال، شناسایی الگو، کنترل سیستم‌های خطی و غیرخطی و دسته‌بندی داده کاربرد گسترده‌ای دارند. دو کاربرد عمده سیستم‌های فازی در کنترل و شناسایی سیستم می‌باشد [۷۱].

۴-۲-۱- کنترل فازی

کنترل کننده‌های فازی با استفاده از مدل‌های هیوریستیک و ریاضی به دست آمده از منطق فازی قابلیت کنترل مسائل پیچیده‌تر را دارا می‌باشند، در حالی که کنترل کننده‌های کلاسیک از مدل ریاضی به فرم معادلات دیفرانسیلی بهره می‌جویند. اما زمانی که مدل سیستم به صورت غیرخطی یا مدل دقیق ریاضی در دسترس نباشد، کنترل کننده‌های فازی از اهمیت زیادی برخوردارند.

در کنترل، ساخت ماشینی که بتواند رفتارهای شخص خبره را تقلید کند، راه جدیدی را در طراحی کنترل کننده‌هایی برای سیستم‌های پیچیده که بدست آوردن مدل کمی برای آن‌ها آسان نیست، باز کرده است. کنترل فازی کاربرد زیادی در سیستم‌های عملی و صنعتی داشته و در اکثر موارد نسبت به کنترل کننده‌های کلاسیک کارایی بهتری دارد. به عنوان نمونه عملکرد PID فازی نسبت به PID معمولی به مراتب بهتر می‌باشد.

در حالت کلی کنترل کننده‌های فازی شامل دو نوع شناخته شده ممدانی و تاکاگی-سوگنو می‌باشند. که در کنترل مد لغزشی برای تنظیم بهره sgn از نوع ممدانی و برای تقریب قانون کنترل معادل، یکی از این دو با تنظیم تطبیقی پارامترها کاربرد دارند [۶۶، ۷۱].

۴-۲-۲- شناسایی و مدل سازی فازی

سیستم های فازی، نوع خاصی از توابع غیرخطی هستند، بنابراین اهمیتی ندارد که به عنوان کنترل کننده استفاده شوند یا تصمیم گیرنده یا پردازشگر سیگنال یا هر نوع سیستم دیگر، موضوع جالب دانستن قابلیت سیستم های فازی از نقطه نظر یک تابع تقریب می باشد. به عنوان مثال سیستم های فازی، کدام نوع از توابع غیرخطی را می توانند نشان دهند یا آن را تقریب بزنند و با چه دقتی؟ [۶۶].

قضیه تقریب عمومی: فرض کنید که ورودی، مجموعه مرجع U یک مجموعه بسته در $\mathbb{R}^n$ باشد، آنگاه برای هر تابع حقیقی داده شده $g(x)$ در U و $\varepsilon > 0$ دلخواه، یک سیستم فازی $f(x)$ به شکل رابطه [۷۱]

$$f(x) = \frac{\sum_{l=1}^M \bar{y}^l [\prod_{i=1}^n a_i^l \exp(-(\frac{x_i - \bar{x}_i^l}{\sigma_i^l})^2)]}{\sum_{l=1}^M [\prod_{i=1}^n a_i^l \exp(-(\frac{x_i - \bar{x}_i^l}{\sigma_i^l})^2)]} \quad (۸-۴)$$

(که در آن $\bar{y}^l$ و σ_i^l پارامترهای توابع عضویت ورودی و خروجی است) وجود دارد به نحوی که:

$$\sup_{x \in U} |f(x) - g(x)| < \varepsilon \quad (۹-۴)$$

این بدین معناست که سیستم های فازی با موتور استنتاج ضرب، فازی ساز منفرد، غیر فازی ساز میانگین مراکز و توابع تعلق گوسین، تقریب زنده های عمومی هستند. این قضیه خود بر اساس قضیه استون-ویراشتراس استنتاج شده است.

قضیه استون ویراشتراس: فرض کنید Z مجموعه ای از توابع پیوسته در مجموعه بسته U می باشد. اگر:

۱- Z یک جبر باشد، بدین معنا که تحت عمل جمع، ضرب و ضرب اسکالر بسته باشد.

۲- Z متمایز کننده نقاط U باشد، بدین معنا که برای هر $x \neq y$ و $x, y \in U$ ، یک $f \in Z$

وجود دارد به نحوی که $f(x) \neq f(y)$.

۳- Z در هیچ نقطه از U صفر نشود، بدین معنی که برای هر $x \in U$ ، یک $f \in Z$ وجود دارد

به نحوی که $f(x) \neq 0$.

آنگاه برای هر تابع حقیقی پیوسته $g(x)$ بر روی U و $E > 0$ دلخواه، یک $f \in Z$ وجود دارد به نحوی که

$$\sup_{x \in U} |f(x) - g(x)| < 0 \quad (۱۰-۴)$$

قضیه مذکور برای شبکه‌های MLP و RBF نیز برقرار می‌باشد. در واقع این شبکه‌ها نیز از خاصیت تقریب تقریب عمومی برخوردار می‌باشند [۶۷].

۳-۴- مقایسه سیستم‌های فازی و عصبی

همان‌طور که قبلاً گفته شد، طراحی کنترل کننده فازی براساس داده‌هایی به فرم منطق فازی انجام می‌گیرد. این زمانی است که داده‌های در دسترس، مجموعه‌ای از داده‌های زبانی و قوانین اگر-آنگاه می‌باشند. به عبارت دیگر کنترل فازی یک روش ریاضی برای بیان استراتژی کنترل به زبان طبیعی می‌باشد. این مسئله در کنترل سیستم‌های پیچیده و زمانی که عملگر انسانی کارایی مناسب‌تری دارد، پیش می‌آید [۶۶].

کنترل عصبی یکی از کنترل کننده‌های متداول بوده و طراحی این کنترل کننده‌ها با استفاده از داده‌های عددی بدست آمده از سیستم تحت کنترل انجام می‌شود. بنابراین فرق اساسی کنترل عصبی و کنترل فازی در نوع داده‌های به دست آمده از سیستم می‌باشد. داده‌های سیستم فازی به صورت زبانی بوده و باید داده‌های عددی به فرم زبانی تبدیل گردد، تا قابل استفاده در کنترل کننده فازی

باشد. اما برای طراحی کنترل کننده عصبی مستقیماً از داده‌های عددی به دست آمده از ورودی و خروجی سیستم استفاده می‌شود. زمانی می‌توان داده‌های عددی را برای سیستم فازی به کار برد که ضرایب سیستم فازی بصورت تطبیقی آموزش یابند. و این در واقع همان مفهوم عصبی- فازی یا تطبیقی- فازی می‌باشد. در حالت کلی تفاوت اساسی میان سیستم‌های فازی و عصبی عبارتند از:

۱- داده‌های مورد استفاده در سیستم فازی از نوع زبانی، یا زبانی به دست آمده از داده عددی می‌باشد، اما برای شبکه عصبی از نوع داده عددی است.

۲- شبکه عصبی بر خلاف سیستم‌های فازی قابلیت یادگیری دارد اما اگر بخواهیم سیستم فازی آموزش ببیند، مفهوم عصبی- فازی خواهد داشت.

۳- برای سیستم‌های جبری شبکه عصبی از توان مدلسازی بیشتری برخوردار است ولی برای مدلسازی رفتار انسان و در کل سیستم‌هایی که امکان مدلسازی آن‌ها به صورت ریاضی کم است، سیستم‌های فازی کارایی بیشتری دارند.

۴-۴- کنترل فازی- عصبی و عصبی- فازی

تا به حال شناسایی و کنترل فازی و عصبی به طور مجزا بحث شد. در حالت کلی انتخاب روش کنترل و یا شناسایی اغلب بسته به نوع داده در دسترس از سیستم می‌باشد. اگر داده‌ها به فرم عددی باشند، شبکه عصبی مطلوب‌تر بوده و اگر داده به فرم قوانین باشد، روش فازی مناسب خواهد بود. به بیانی دیگر، شبکه‌های عصبی قابلیت یادگیری و سیستم فازی قابلیت مدلسازی و بیان مفاهیم هیوریستیک را دارا می‌باشند. جمع کردن این دو خاصیت در روش‌های هوشمند در حالت کلی، می‌تواند به نتایج بهتری منجر شود و محدودیت‌های هر یک از دو روش را بهبود ببخشد.

شیوه‌های مختلفی برای ترکیب این دو روش وجود دارد. در شناسایی و طراحی کنترل کننده‌های پیچیده ترکیبات مختلفی از این دو روش را می‌توان به کار برد. اما در حالت کلی سیستم فازی و شبکه‌های عصبی به دو روش می‌توانند ترکیب شوند [۶۶]:

۱- سیستم فازی-عصبی^۱: هدف از معرفی مفاهیم فازی در شبکه‌های عصبی، بالا بردن قابلیت بیان مفاهیم هیوریستیک در شبکه عصبی است. در این حالت ورودی‌ها، وزن‌ها، عملگر جمع، تابع تحریک، و خروجی‌ها، به صورت فازی در نظر گرفته می‌شوند. مثلاً t -نرم و s -نرم را می‌توان به جای عملگر جمع در شبکه قرار داد یا نرون استاندارد به نرون فازی تغییر می‌کند. در این روش سیستم فازی-عصبی، قابلیت یادگیری را هم داراست.

۲- سیستم عصبی-فازی^۲: در حالت دوم شبکه‌های عصبی می‌تواند در مدلسازی فازی و کنترل، برای فراهم کردن قابلیت یادگیری سیستم فازی بکار رود. در این حالت، ضرایب نگاشت فازی با استفاده از الگوریتم‌های آموزش شبکه عصبی، تنظیم می‌شوند که همان مفهوم تطبیقی فازی را به همراه دارد و الگوریتم ANFIS^۳ خوانده می‌شود. این الگوریتم در سال ۱۹۹۳ توسط جانگ^۴ پیشنهاد شد [۷۲]. این ساختار در کنترل و همچنین برای کاربردهای دیگر مفید می‌باشد.

برای اینکه بتوانیم الگوریتم تطبیقی را بر روی سیستم فازی پیاده کنیم، سیستم فازی را به فرم عصبی در نظر می‌گیریم. به عنوان مثال، در شکل ۴-۵ ساختار سیستم فازی تاکاگی-سوگنو به

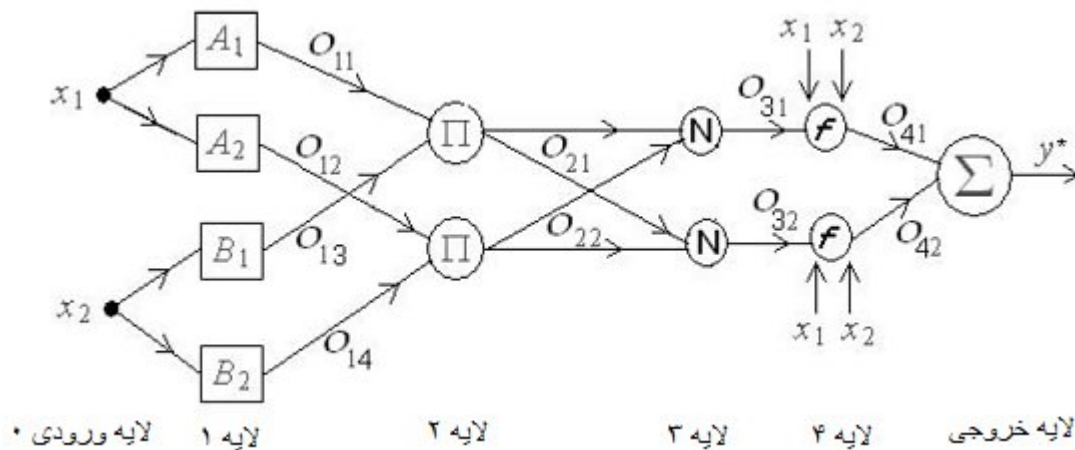
^۱ Fuzzy- neural system

^۲ Neural- fuzzy system

^۳ Adaptive neuro-fuzzy inference systems

^۴ Jang

صورت شبکه عصبی پیشخور پنج لایه نشان داده شده است. در نتیجه تنظیم ضرایب سیستم فازی راحت تر می گردد و برای تنظیم ضرایب می توان از روش گرادیان کاهشی استفاده کرد.



شکل ۴-۵- مدل فازی تاکاگی- سوگنو مرتبه اول با دو قانون [۳]

همانطور که قبلا اشاره شد، به علت اینکه سیستم فازی مفاهیم را به صورت زبانی و با استفاده از قوانین مدل می کند، رواج زیادی نسبت به شبکه های عصبی دارد. اما طراحی سیستم فازی از روی داده ها زمانبر بوده و همچنین نوشتن قوانین برای ورودی های زیاد و تنظیم توابع عضویت مشکل می باشد. اما با آموزش سیستم فازی، زمان طراحی کم، دقت و کارایی سیستم افزایش می یابد [۶۶].

در این تحقیق و در ادامه، شبکه های عصبی و سیستم های فازی بطور موازی با هم عمل می کنند و به صورت ترکیبات عصبی- فازی و فازی- عصبی نخواهند بود.

فصل پنجم: کنترل عصبی- فازى مد لغزشى پيوسته و گسسته در زمان

۵-۱- مرورى بر تحقيقات انجام شده

همانطور که پيشتر اشاره شد، کنترل مد لغزشى عملکرد مطلوبى در برابر اغتشاش خارجى و عدم قطعيت‌هاى مدل دارد. اما اين روش كنترلى قدرتمند داراى دو عيب اساسى: ۱- نوسانات شديد سيگنال كنترلى ۲- وابستگى به مدل رياضى سيستم تحت كنترل، است. با توجه قابليت سيستم‌هاى فازى و شبكه‌هاى عصبى در تقريب و آموزش، با تركيب كنترل مد لغزشى با تقريب‌گرهاى عصبى و فازى، مى‌توان از مزايای هر دو روش بهره برد. از تركيبات رايج اين نوع كنترل كننده ها مى‌توان كنترل فازى- مد لغزشى (FSMC)، كنترل عصبى- مد لغزشى (NSMC) و كنترل عصبى- فازى- مد لغزشى (NFSMC) را نام برد.

در زمينه كنترل FSMC، تحقيقات با ارزشى به انجام رسيده كه از برخى از آن‌ها مى‌توان به [۲۷-۲۳] اشاره كرد. به عنوان نمونه: در [۲۳-۲۴]، از سيستم فازى براى حذف نوسانات ناخواسته كنترل مد لغزشى اعمالى به دو تانكر مرتبط و بازوى ربات، استفاده شده است. همچنين در [۲۶-۲۷]، مفهوم سيستم فازى با تنظيم ضرايب تطبيقى، در كنترل بازوى ربات دو رابط و پل تحريك شده، پياده شده است.

براى كنترل عصبى و كنترل كننده NSMC، مى‌توان به مطالعات بارزى از قبيل [۲۸-۲۹] و NSMC [۳۰-۳۶] اشاره نمود. براى مثال: در [۳۰]، شبكه عصبى موازى با كنترل كننده‌هاى

کلاسیک با آموزش ضرایب بصورت لغزشی، برای کنترل بازوی ربات به کار گرفته شده. در تحقیقی دیگر، کنترل ربات صنعتی با استفاده از دو کنترل کننده کلاسیک و عصبی موازی انجام شده که در آن شبکه عصبی برای تنظیم ضرایب خود، از سطح لغزش کمک می‌گیرد [۳۱]. در [۳۲]، تقریب قانون کنترل معادل با استفاده از روش گرادیان کاهشی مطرح شده است. در تحقیقات صورت گرفته در [۳۴-۳۶] و [۷۳-۷۴]، کنترل عصبی-مد لغزشی برای کنترل سیستم‌هایی از قبیل: سیستم الکترومکانیکی، بازوی ربات منعطف، ربات زیردریایی و ربات متحرک مطرح شده است. در اکثر موارد قبلی که ذکر نمودیم، پرسپترون چند لایه به کار گرفته شده؛ اما در [۷۵-۷۷]، مطالعات خوبی در زمینه استفاده از شبکه‌های عصبی با توابع گوسین، برای تقریب قسمتی از قانون کنترل در کنترل ارتعاشات و کنترل سیستم لرنز^۱ انجام شده است.

در زمینه کنترل NFSMC نیز تحقیقات با ارزشی به انجام رسیده است [۳۷-۴۲]. به عنوان نمونه، در [۳۷]، از ترکیب شبکه عصبی و سیستم فازی مجزا، برای کاهش اثر اصطکاک و اغتشاشات دامنه بزرگ استفاده شده. در تحقیقی دیگر، برای کنترل سیستم آشوبگر لرنز، تقریبگر عصبی با ویژگی‌های فازی به کار رفته است [۳۸]. در [۳۹]، شبکه عصبی-فازی با الگوریتم یادگیری خارج خط استفاده شده است؛ اما مفهومی که در این تحقیق دنبال می‌شود، کاربرد دو تقریبگر موازی برای تقریب کل قانون کنترل لغزشی می‌باشد. به عنوان نمونه، در [۷۸] دو شبکه عصبی موازی برای تقریب قانون کنترل کلیدزنی و معادل، بر روی سیستم دیددریایی^۲ تحت گرانت نیروی دریایی چین بصورت عملی پیاده شد. در تحقیقی مشابه، این کنترل کننده برای کنترل بازوی ربات منعطف به کار گرفته شد [۷۹]. در همین راستا، [۴۱-۴۲] از دو تقریبگر فازی و عصبی برای کنترل دو تانکر با کوپلینگ بالا و بازوی ربات استفاده می‌کنند. در تکمیل همین بحث و در ادامه، به طراحی کنترل کننده NFSMC به

^۱ lorenz

^۲ Seasaw

ازای دو تقریبگر RBF و MLP، و مقایسه نتایج آنها می‌پردازیم. علاوه بر این، طراحی این کنترل کننده را در حوزه گسسته بسط می‌دهیم.

۵-۲- طراحی کنترل کننده زمان- پیوسته برای بازوی ربات

۵-۲-۱- کنترل مد لغزشی پیوسته در زمان

معادله دینامیکی بازوی ربات سه رابط را بصورت زیر در نظر بگیرید:

$$M(q)\ddot{q} + C(q, \dot{q})\dot{q} + G(q) + F(\dot{q}) + u_d(t) = u(t) \quad (۱-۵)$$

که در آن $u_d \in \mathbb{R}^3$ اغتشاش خارجی بوده بطوریکه $|u_d| \leq U_D$. مسئله کنترل ردگیری به این صورت است که با داشتن مسیر مطلوب q_d ، یک قانون کنترل برای گشتاور ورودی u به گونه‌ای طراحی می‌کنیم که بردار موقعیت q بتواند مسیر مطلوب را ردگیری نماید. بردار خطای ردگیری در اینجا به صورت زیر تعریف می‌شود:

$$e = q_d - q \quad (۲-۵)$$

برای طراحی کنترل مد لغزشی، سطح لغزش خطی را به صورت زیر در نظر بگیرید:

$$s = \dot{e} + \lambda e \quad (۳-۵)$$

که در آن λ ماتریس قطری با مقادیر مثبت است. همچنین، بردار سرعت مرجع بصورت زیر تعریف می‌گردد [۴۳]:

$$\dot{q}_r = \dot{q}_d - \lambda e \quad (۴-۵)$$

بنابراین می‌توان سطح لغزش را بدین صورت نمایش داد:

$$s = \dot{q} - \dot{q}_r \quad (۵-۵)$$

برای اینکه حالت‌های سیستم $(e, \dot{e})$ در یک زمان محدود به سطح لغزش $s = 0$ رسیده، روی آن باقی بماند، باید قانون کنترل را به گونه ای طراحی نماییم که شرط لغزش زیر را برآورده نماید.

$$\frac{1}{2} \frac{d}{dt} [s^T M s] < -\eta (s^T s)^{1/2} \quad (۶-۵)$$

که η مثبت معین می‌باشد. اکنون طراحی کنترل کننده SMC را توسط لم ۵-۱ بیان می‌کنیم.

باید توجه داشت که پارامترهای M ، C ، G و F بازوی ربات، دارای عدم قطعیت می‌باشند. برای طراحی کنترل کننده مد لغزشی، پارامترهای $\hat{M}$ ، $\hat{C}$ ، $\hat{G}$ و $\hat{F}$ را به عنوان تخمینی از پارامترهای اصلی در نظر می‌گیریم و اختلاف مقادیر واقعی و تخمینی را به صورت $\Delta M = \hat{M} - M$ ، $\Delta C = \hat{C} - C$ ، $\Delta G = \hat{G} - G$ و $\Delta F = \hat{F} - F$ تعریف می‌کنیم.

لم ۵-۱: سیستم داده شده با معادله دینامیکی (۵-۱)، سطح لغزش و سرعت مرجع تعریف شده با معادلات (۵-۳) و (۵-۴) مفروض است. اگر قانون کنترل را بصورت زیر انتخاب نماییم،

$$u = \hat{f} - K \operatorname{sgn}(s) \quad (۷-۵)$$

بطوریکه

$$\hat{f} = \hat{M} \ddot{q}_r + \hat{C} \dot{q}_r + \hat{G} + \hat{F} \quad (۸-۵)$$

و

$$K_{ii} \geq |\Delta f|_i + U_{Di} + \eta_i \quad (۹-۵)$$

آنگاه شرط لغزشی (۵-۶) برآورده خواهد شد. در نامعادله آخر، K_{ii} نشان دهنده عناصر بردار بهره لغزشی K است ($\Delta f = \hat{f} - f$).

اثبات: تابع منتخب لیپانوف زیر را در نظر بگیرید [۴۳]:

$$V = \frac{1}{2} s^T M s \quad (۱۰-۵)$$

از آنجا که M مثبت معین است، به ازای $s \neq 0$ داریم $V > 0$ و با مشتق گیری از رابطه فوق و استفاده از تقارن M می توان نوشت:

$$\dot{V} = s^T M \dot{s} + \frac{1}{2} s^T \dot{M} s \quad (۱۱-۵)$$

از (۵-۵) نتیجه می گردد:

$$\dot{V} = s^T M (\ddot{q} - \ddot{q}_r) + \frac{1}{2} s^T \dot{M} s \quad (۱۲-۵)$$

با جایگزین کردن (۱-۵) در (۱۲-۵) می توان نوشت:

$$\dot{V} = s^T (u - C\dot{q} - G - F - u_d - M\ddot{q}_r) + \frac{1}{2} s^T \dot{M} s \quad (۱۳-۵)$$

با اعمال $\dot{q} = s + \dot{q}_r$ به معادله بالا و با توجه به اینکه $s^T \left[\frac{1}{2} \dot{M} - C \right] s = 0$ داریم:

$$\dot{V} = s^T (u - M\ddot{q}_r - C\dot{q}_r - G - F - u_d) \quad (۱۴-۵)$$

اکنون، جایگذاری (۷-۵) و (۸-۵) در معادله فوق، می دهد:

$$\dot{V} = s^T (\Delta M \ddot{q}_r + \Delta C \dot{q}_r + \Delta G + \Delta F - u_d) - s^T K \text{sign}(s) \quad (۱۵-۵)$$

در نتیجه داریم:

$$\dot{V} = s^T (\Delta f - u_d) - \sum_{i=1}^n K_{ii} |s_i| \quad (۱۶-۵)$$

سرانجام از رابطه (۹-۵) می‌توان نتیجه گرفت:

$$\dot{V} \leq - \sum_{i=1}^n K_{ii} |s_i| \quad (۱۷-۵)$$

بنابر رابطه فوق، V یک تابع لیاپانوف بوده و شرط لغزشی (۶-۵) نیز ارضا شده است. $\square$

۵-۲-۲- نتایج شبیه‌سازی به ازای کنترل کننده مد لغزشی

رابط سه رابط اسکارا (۱-۵)، با پارامترهای زیر مفروض است [۶۳]:

$$\begin{aligned} m_1 &= 1.0 \text{ kg} & m_2 &= 0.8 \text{ kg} & m_3 &= 0.5 \text{ kg} & g &= 9.8 \\ l_1 &= 1.0 \text{ m} & l_2 &= 0.8 \text{ m} & l_3 &= 0.6 \text{ m} \end{aligned}$$

چون پارامترهای جرم، اینرسی، مرکز جرم و اصطکاک بازوها دارای عدم قطعیت می‌باشند. لذا مقادیر

پارامترهای تخمینی را به شکل زیر انتخاب می‌کنیم [۲۶]:

$$\hat{M} = 0.95M \quad \hat{C} = 0.8C \quad \hat{G} = 0.9G \quad \hat{F} = 0.95F$$

با صرفنظر کردن از قسمت نمایی بردار اصطکاک، اصطکاک مفاصل را به شکل زیر مدل می‌کنیم

[۶۳]:

$$F(\dot{q}) = \begin{bmatrix} b_1^s \dot{q}_1 + b_1^d \operatorname{sgn}(\dot{q}_1) \\ b_2^s \dot{q}_2 + b_2^d \operatorname{sgn}(\dot{q}_2) \\ b_3^s \dot{q}_3 + b_3^d \operatorname{sgn}(\dot{q}_3) \end{bmatrix} \quad b_1^s = b_2^s = b_3^s = 12 \quad b_1^d = b_2^d = b_3^d = 0.2$$

تعیین پارامترهای اصطکاک امری مشکل است؛ ولی در طراحی قانون کنترل لغزشی $\hat{F} = 0.95F$

در نظر می‌گیریم. علاوه براین، اغتشاش خارجی اعمالی بر مفاصل بصورت زیر بیان می‌شود:

$$u_d(t) = [0.4 \sin(4t) \quad 0.4 \sin(4t) \quad 0.4 \sin(4t)]^T$$

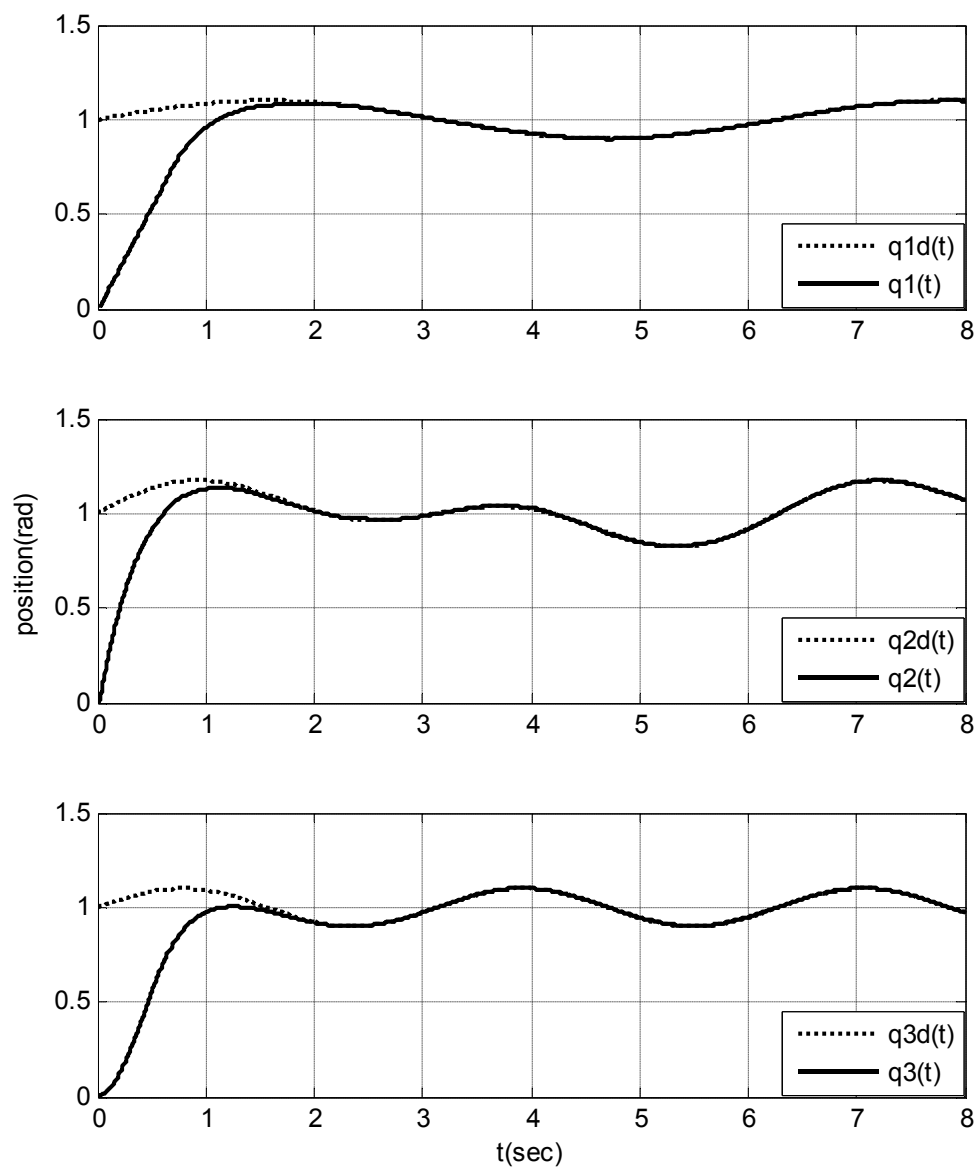
در نتیجه داریم: $U_d = [0.4 \quad 0.4 \quad 0.4]^T$. همچنین برای نمایش قابلیت کنترل کننده‌های طراحی شده در ردیابی ورودی پله و سینوسی، سیگنال مرجع به صورت ترکیبی از ورودی پله و سینوسی‌های کم دامنه در نظر گرفته می‌شود.

$$q_d(t) = [1 + 0.1 \sin(t) \quad 1 + 0.1(\sin(t) + \sin(2t)) \quad 1 + 0.1 \sin(2t)]^T$$

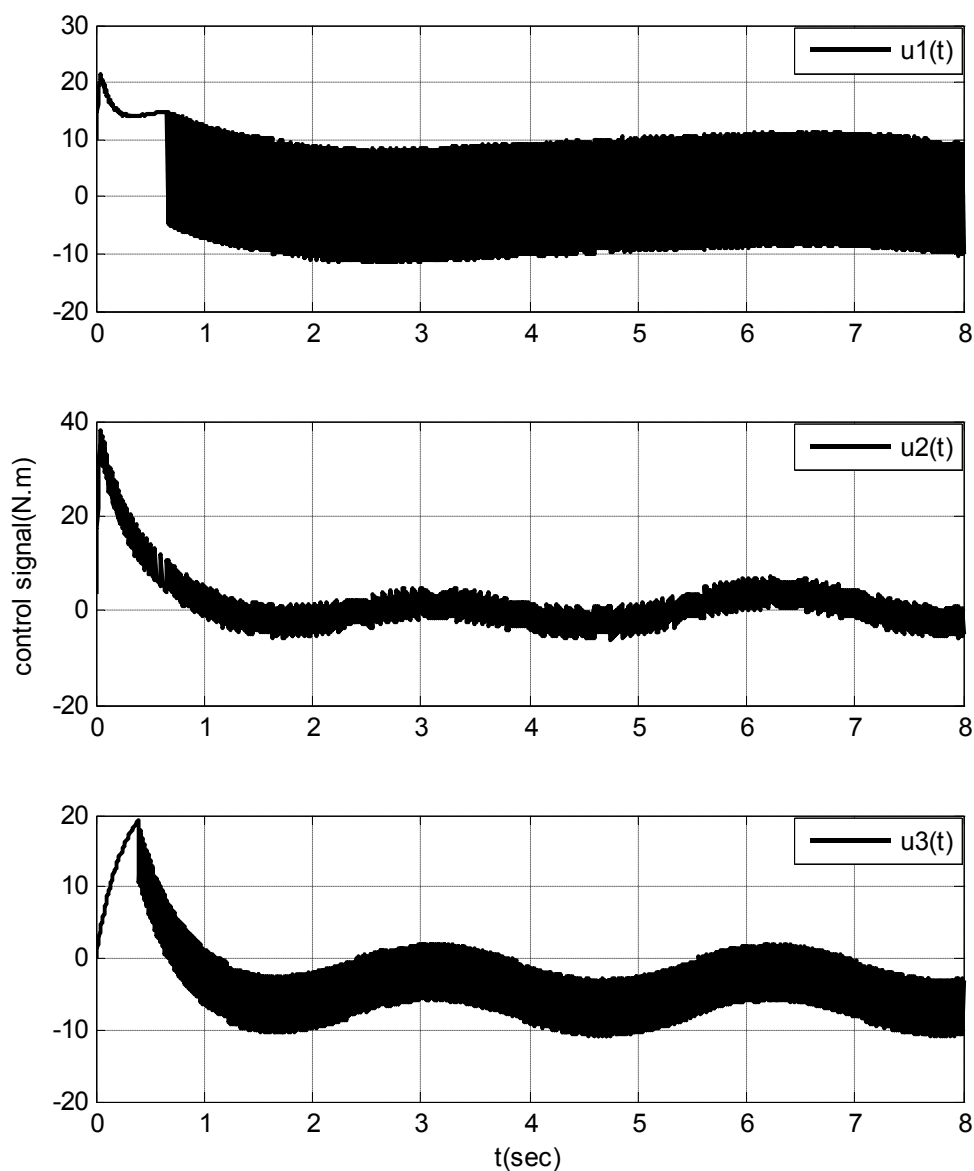
پارامترهای طراحی را نیز به صورت زیر در نظر می‌گیریم:

$$K = \begin{bmatrix} 10 & 0 & 0 \\ 0 & 3.6 & 0 \\ 0 & 0 & 3 \end{bmatrix}, \quad \lambda = \begin{bmatrix} 3 & 0 & 0 \\ 0 & 3 & 0 \\ 0 & 0 & 3 \end{bmatrix}$$

نتایج حاصل از طراحی کنترل مد لغزشی متداول در شکل‌های (۵-۱) و (۵-۲) آورده شده است که به ترتیب نشان دهنده موقعیت مفاصل و سیگنال کنترلی اعمالی به آن‌ها می‌باشد.



شکل ۵-۱- مسیر مطلوب (نقطه چین) و زاویه مفصل (خط) حاصل از SMC



شکل ۵-۲- گشتاور ورودی حاصل از SMC

همانطور که در شکل ۵-۱ نشان داده شده، کنترل مد لغزشی با وجود اغتشاش خارجی، عدم قطعیت‌های مدل و کوپلینگ شدید مابین بازوها (مثلاً اثر وزن بازوهای دوم و سوم روی بازوی اول) ردیابی مطلوب و مقاومی از خود نشان می‌دهد. زمان رسیدن q به مقدار مطلوب q_d ، برای هر سه مفصل تقریباً کمتر از 2sec است.

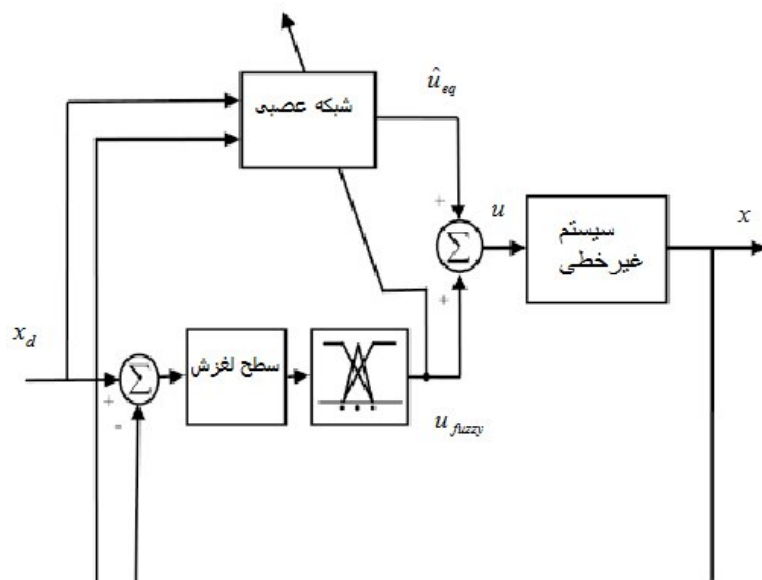
۵-۲-۳- طراحی کنترل کننده عصبی- فازی- مد لغزشی

همانطور که در بخش قبل مشاهده شد، برای طراحی کنترل کننده SMC مقادیر پارامترهای سیستم را $\hat{M} = 0.95M$ ، $\hat{C} = 0.8C$ ، $\hat{G} = 0.9G$ ، $\hat{F} = 0.95F$ در نظر گرفتیم؛ اما زمانی که مقادیر تخمینی از مقادیر واقعی فاصله داشته باشند ردیابی با خطا همراه خواهد بود. برای کاهش خطا می‌توان مقدار بهره کنترلی را بزرگ انتخاب نمود؛ ولی در این حالت دامنه نوسانات سیگنال کنترلی نیز تقویت خواهد شد. بنابراین در ادامه ما حالتی را در نظر می‌گیریم که مدل ریاضی سیستم در دسترس نباشد.

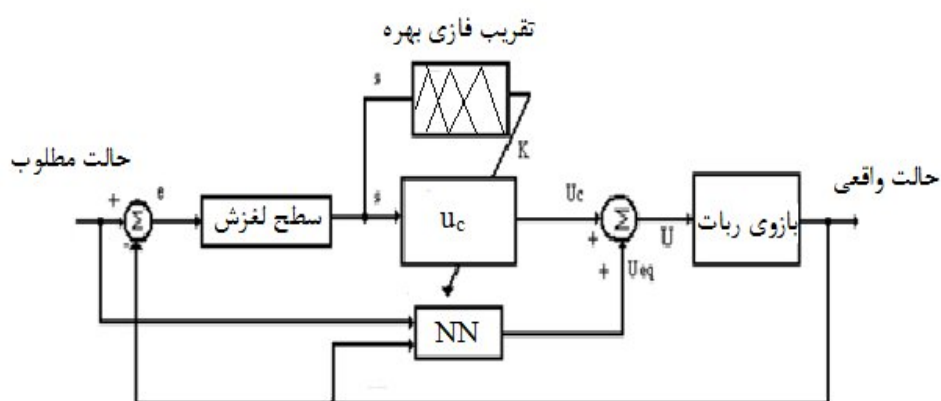
۵-۲-۳-۱- ساختار کنترل کننده پیشنهادی

در ساختار پیشنهادی، قانون کنترلی کلیدزنی^۱ با استفاده از سیستم فازی پیوسته و قانون کنترل معادل، در کنترل مد لغزشی، با استفاده از شبکه عصبی تقریب زده می‌شود. سیگنال کنترلی از مجموع خروجی شبکه عصبی و خروجی جمله کلیدزنی به دست می‌آید. خروجی کنترل کلیدزنی، برای تنظیم آنالین ضرایب شبکه عصبی بکار می‌رود. هدف از یادگیری شبکه عصبی، مینیمم کردن کنترل کارکتیو می‌باشد. این بدان علت است که در مد لغزش، قانون کنترل معادل برای حفظ سیستم بر روی سطح لغزش کافی بوده و ترم کلیدزنی برای جبران انحرافات از سطح لغزش ضروری می‌باشد. شماتیک کلی سیستم کنترل در شکل ۵-۳ نشان داده شده است.

¹ Corrective



الف) تقریب کل قانون کنترل کلیدزنی [۴۱]



ب) تقریب بهره قانون کنترل کلیدزنی [۴۲]

شکل ۵-۳- دو ساختار NFSMC

اکنون با توجه به شکل ۵-۳- الف، قانون کنترلی (۵-۷) را بصورت زیر در نظر بگیرید:

$$u(t) = u_{EQ}(t) + u_{SW}(t) \quad (۵-۱۸)$$

که در آن u_{EQ} ، قانون کنترل معادل و u_{SW} و u_C ، قانون کنترل کلیدزنی (کارکتیو) نامیده می

شوند. این دو جمله بصورت زیر تعریف می‌شوند:

$$\begin{aligned}
 u_{EQ}(t) &= \hat{M}(q)\ddot{q}_r + \hat{C}(q, \dot{q})\dot{q}_r + \hat{G}(q) + \hat{F}(\dot{q}) \\
 u_{SW}(t) &= -K \operatorname{sgn}(s)
 \end{aligned}
 \tag{۱۹-۵}$$

۵-۲-۳-۲- محاسبه قانون کنترل کلیدزنی

کنترل کننده (۵-۱۸)، باعث تولید نوسانات فرکانس بالا در سیگنال کنترلی می‌گردد. این نوسانات فرکانس بالا، به دلیل آنکه موجب تحریک دینامیک‌های فرکانس بالا می‌شوند، نامطلوب می‌باشند. برای حذف این نوسانات، سیستم فازی پیوسته u_{Fuzzy} برای تقریب u_C استفاده شده است [۲۳ و ۳۹ و ۴۱].

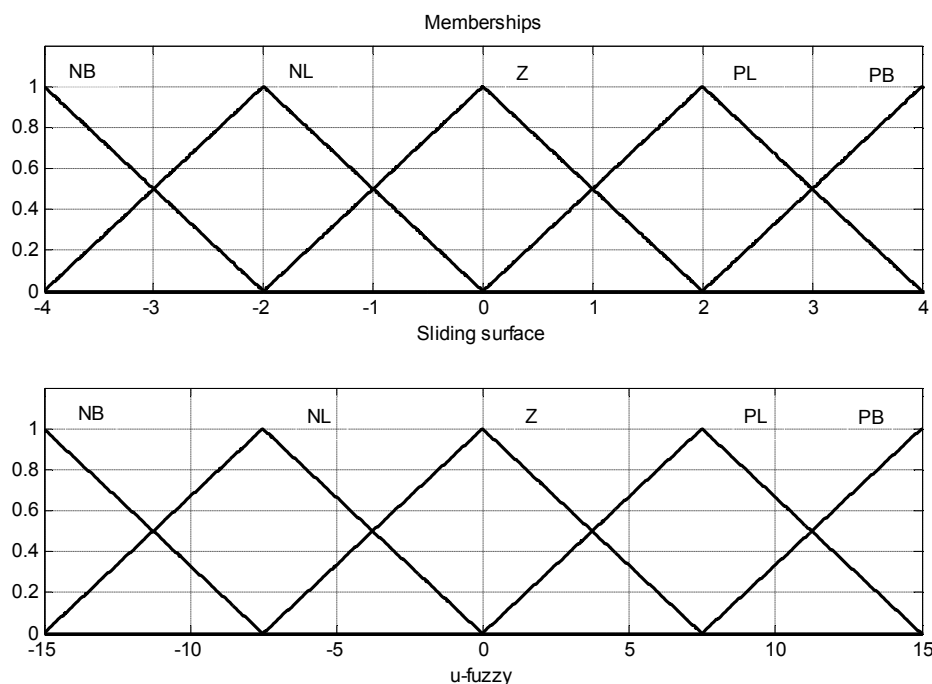
طراحی کنترل کننده فازی نیز با بسط سطح لغزش قطعی به سطح لغزش فازی بوسیله اصطلاحات زبانی انجام می‌پذیرد. برای تقسیم بندی بازه ورودی سیستم فازی (S)، مجموعه توابع عضویت بدین صورت معرفی می‌شود:

$$F_S(s_i(t)) = \{NB, NL, Z, PL, PB\} \tag{۲۰-۵}$$

برای خروجی کنترل u_{Fuzzy} ، مجموعه توابع عضویت، بطور مشابه بصورت زیر تعریف شده است:

$$F_S(u_{iFuzzy}(t)) = \{NB, NL, Z, PL, PB\} \tag{۲۱-۵}$$

توابع عضویت این دسته‌های فازی در شکل ۵-۴ نمایش داده شده است.



شکل ۴-۵- نمایش دسته‌های فازی ورودی $F_S(s_i(t))$ و خروجی $F_S(u_{iFuzzy}(t))$

از این دو دسته توابع عضویت، قوانین فازی را برای مفصل ۱-ام، می‌توانیم به شکل زیر بنویسیم:

Rule1: if $s_i(t)$ is NB then u_{iFuzzy} is PB

Rule2: if $s_i(t)$ is NL then u_{iFuzzy} is PL

Rule3: if $s_i(t)$ is Z then u_{iFuzzy} is Z

Rule4: if $s_i(t)$ is PL then u_{iFuzzy} is NL

Rule5: if $s_i(t)$ is PB then u_{iFuzzy} is NB

در مراحل قبلی، توابع عضویت ورودی- خروجی و قوانین فازی مشخص گردید. در این مرحله باید

از این قوانین فازی مقدار قطعی^۱ کنترلی تولید شود. برای بدست آوردن این مقدار کنترلی از روش

میانگین وزنی استفاده می‌کنیم:

$$u_{Fuzzy} = \frac{\sum_{i=1}^5 C_i \mu_i(s)}{\sum_{i=1}^5 \mu_i(s)} \quad (۱۹-۵)$$

^۱ Crisp

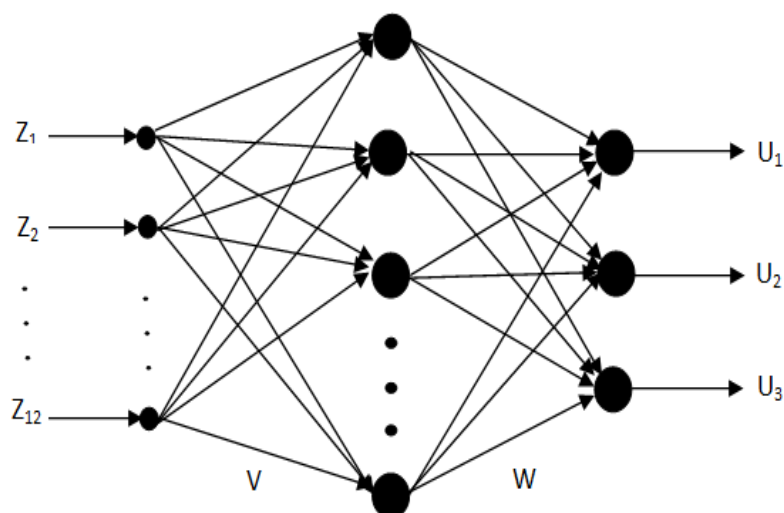
که در آن $C = [C_1, \dots, C_5]^T$ بردار مراکز توابع عضویت خروجی u_{Fuzzy} می‌باشد. در نهایت، خروجی موتور استنتاج فازی را برای بردار سطح لغزش s ، می‌توان بصورت زیر نوشت:

$$u_{Fuzzy} = -K \text{sign}(s) \quad (۲۰-۵)$$

۵-۲-۳-۳- محاسبه قانون کنترل معادل

۵-۲-۳-۱- تقریب قانون کنترل معادل با شبکه عصبی MLP

به منظور تقریب قانون کنترل معادل، شبکه عصبی بصورت دولایه انتخاب شده است. ساختار ورودی‌ها و خروجی‌های شبکه با توجه به قانون کنترل تعیین می‌گردد. شکل ۵-۵، نشان دهنده ساختار شبکه عصبی می‌باشد.



شکل ۵-۵ - ساختار شبکه عصبی دولایه، به منظور تقریب قانون کنترل [۶۷]

از شکل ۵-۵، فرم شبکه عصبی دولایه را می‌توان به صورت زیر در نظر گرفت:

$$U(t) = W\sigma(VZ(t)) + \delta \quad (۲۱-۵)$$

در رابطه بالا، W ماتریس ضرایب لایه اول (لایه میانی)، V ماتریس ضرایب لایه خروجی، δ خطای تقریب و $\sigma(x)$ تابع فعایت^۱ شبکه می‌باشند. همچنین $Z = [z_1 \ \dots \ z_{12}]^T$ بردار ورودی و $U = [U_1 \ U_2 \ U_3]^T$ بردار خروجی است. از شکل ۵-۵، چنین برمی‌آید که قانون کنترل معادل با استفاده از الگوریتم گرادیان تکراری به منظور مینیمم سازی خطای مجذور میانگین مابین مقدار مطلوب و واقعی، محاسبه خواهد شد.

در حالت کلی، از الگوریتم پس‌انتشار خطا بر اساس روش گرادیان کاهشی^۲ برای تنظیم ضرایب شبکه عصبی می‌توان استفاده کرد. فرایند آموزش، از الگوریتم گرادیان کاهشی تکراری^۳ برای مینیمم کردن خطای مجذور میانگین مابین مقدار خروجی واقعی و مطلوب استفاده می‌کند. پس اگر تابع خطا^۴ (تابع هزینه^۵) J_{NN} را به صورت زیر در نظر بگیریم:

$$J_{NN} = \frac{1}{2} (U_{EQ} - U_{NN})^T (U_{EQ} - U_{NN}) \quad (۲۲-۵)$$

که در آن U_{EQ} قانون کنترل معادل و U_{NN} خروجی شبکه عصبی دولایه می‌باشد. هدف ما مینیمم کردن تابع هزینه J_{NN} نسبت به تغییرات ضرایب شبکه عصبی می‌باشد. حال اگر معادلات یادگیری شبکه را به شکل زیر تعریف کنیم:

$$\dot{W}(t) = \mu \frac{\partial J_{NN}}{\partial W} \quad , \quad \dot{V}(t) = -\mu \frac{\partial J_{NN}}{\partial V} \quad (۲۳-۵)$$

^۱ Activation function

^۲ descent

^۳ iterative

^۴ Error function

^۵ Cost function

در رابطه فوق، μ بهره ثابت بوده و نشان دهنده پارامتر نرخ یادگیری الگوریتم BP می‌باشد. علاوه

براین، ترم‌های $\frac{\partial J_{NN}}{\partial W}$ و $\frac{\partial J_{NN}}{\partial V}$ با استفاده از قاعده زنجیره‌ای به صورت زیر به دست می‌آیند [۳۲]:

$$\begin{aligned}\frac{\partial J_{NN}}{\partial W} &= \frac{\partial J_{NN}}{\partial U_{NN}} \frac{\partial U_{NN}}{\partial W} \\ \frac{\partial J_{NN}}{\partial V} &= \frac{\partial J_{NN}}{\partial U_{NN}} \frac{\partial U_{NN}}{\partial \sigma} \frac{\partial \sigma}{\partial V}\end{aligned}\quad (24-5)$$

با جایگذاری رابطه (۲۱-۵) در معادله (۲۴-۵) و با انتخاب $\sigma = \tanh(x)$ داریم:

$$\begin{aligned}\left[\frac{\partial J_{NN}}{\partial W}\right]^T &= -\tanh(VZ)(U_{EQ} - U_{NN})^T \\ \left[\frac{\partial J_{NN}}{\partial V}\right]^T &= -Z(U_{EQ} - U_{NN})^T W[I - \tanh^2(VZ)]\end{aligned}\quad (25-5)$$

اکنون، با جایگزینی (۲۵-۵) در (۲۳-۵)، ضرایب شبکه عصبی بدین صورت تنظیم می‌شوند:

$$\begin{aligned}\dot{W} &= \mu(U_{EQ} - U_{NN})[\tanh(VZ)]^T \\ \dot{V} &= \mu(I - \tanh^2(VZ))W^T(U_{EQ} - U_{NN})Z^T, \mu > 0\end{aligned}\quad (26-5)$$

توجه شود که مقدار واقعی U_{EQ} در معادله بالا معلوم نمی‌باشد. لذا برای تنظیم ضرایب، نمی‌توان

این رابطه را بکار برد. برای غلبه بر این مشکل می‌توان از دو روش زیر استفاده نمود:

۱- جایگزینی عبارت نامعلوم $U_{EQ} - U_{NN}$ با سطح لغزش $s(t)$ [۳۲ و ۷۸]:

۲- استفاده از خروجی سیستم فازی u_{FUZZY} [۴۱].

با توجه به اینکه با گذشت زمان ($t \rightarrow \infty$) عبارت $U_{EQ} - U_{NN}$ به سمت صفر میل می‌کند

($U_{EQ} - U_{NN} \rightarrow 0$)، و همچنین به علت آنکه در کنترل مد لغزشی، سطح لغزش و قانون کلید زنی

(با سیستم فازی تقریب زده می‌شود) با گذشت زمان به سمت صفر میل می‌کنند ($s(t) \rightarrow 0$)

و $0 \rightarrow u_{FUZZY}$ ، در نتیجه در عوض عبارت $\hat{U} - U$ از $s(t)$ یا u_{FUZZY} می‌توان استفاده نمود. در نهایت ضرایب شبکه عصبی به صورت زیر در می‌آیند:

$$\begin{aligned}\dot{W} &= \mu u_{FUZZY} [\tanh(VZ)]^T \\ \dot{V} &= \mu (I - \tanh^2(VZ)) W^T u_{FUZZY} Z^T, \mu > 0\end{aligned}\quad (27-5)$$

با توجه به اینکه معادلات مذکور به صورت ماتریسی بیان شده اند، در محیط SIMULINK به راحتی پیاده سازی می‌شوند.

اثبات همگرایی الگوریتم گرادیان کاهشی: تابع خطای J_{NN} شبکه عصبی را در نظر بگیرید، آنگاه آهنگ تغییرات این تابع، عبارت است از:

$$\frac{\partial J(t)}{\partial t} = \frac{\partial J_{NN}(t)}{\partial W(t)} \frac{\partial W(t)}{\partial t} + \frac{\partial J_{NN}(t)}{\partial V(t)} \frac{\partial V(t)}{\partial t} \quad (28-5)$$

با جایگذاری رابطه (۲۳-۵) در معادله بالا،

$$\frac{\partial J(t)}{\partial t} = -\mu \left| \frac{\partial J_{NN}(t)}{\partial W(t)} \right|^2 - \mu \left| \frac{\partial J_{NN}(t)}{\partial V(t)} \right|^2 \quad (29-5)$$

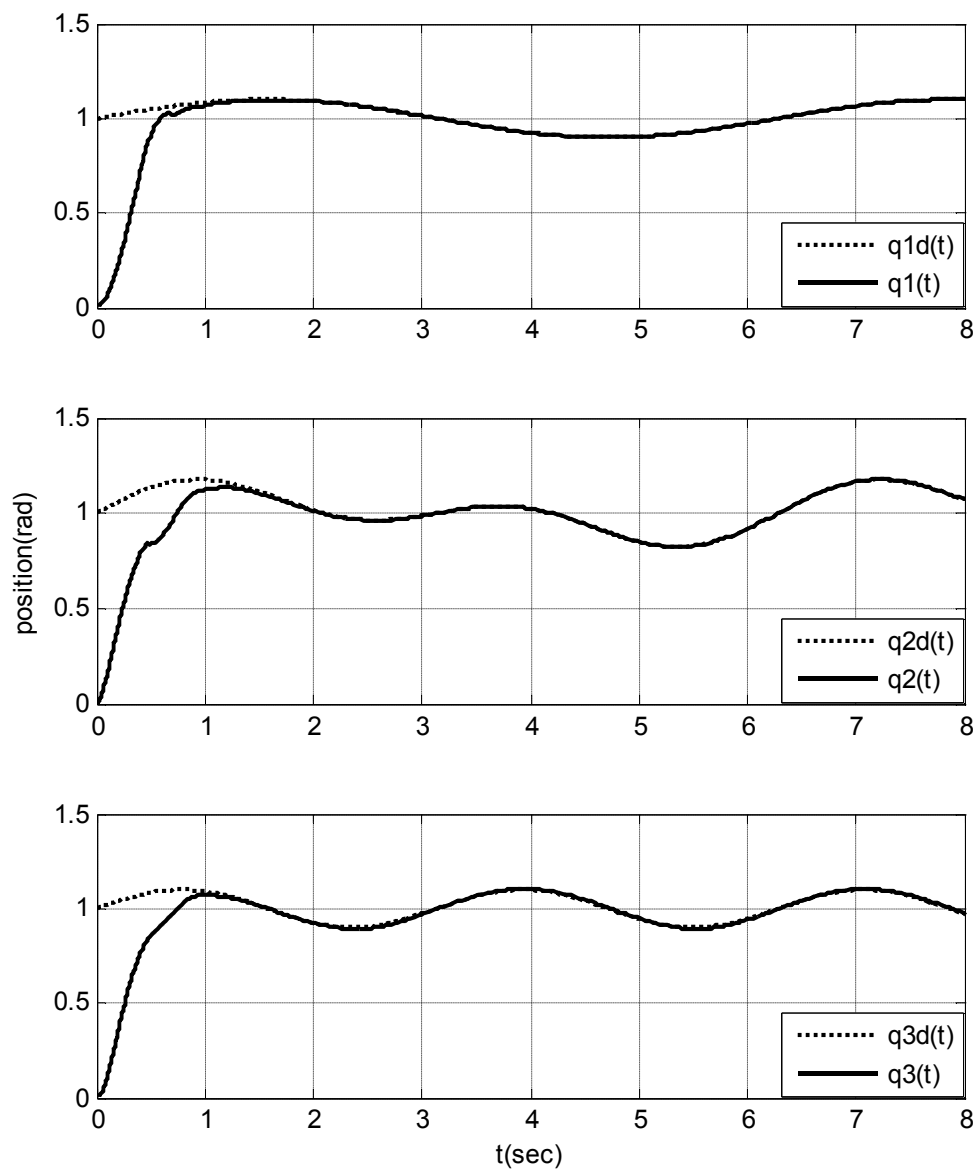
بدست می‌آید. از رابطه فوق نتیجه می‌گیریم که $\partial J(t)/\partial t \leq 0$. بنابراین با گذشت زمان و تنظیم ضرایب، مقدار J به سمت صفر میل خواهد کرد. اما این زمانی ممکن است که تعداد نرون ها، تعداد لایه ها، توابع فعالیت و پارامتر یادگیری بطور مناسبی انتخاب شوند.

۵-۲-۳-۳-۲- نتایج شبیه‌سازی به ازای کنترل کننده NFSMC-MLP

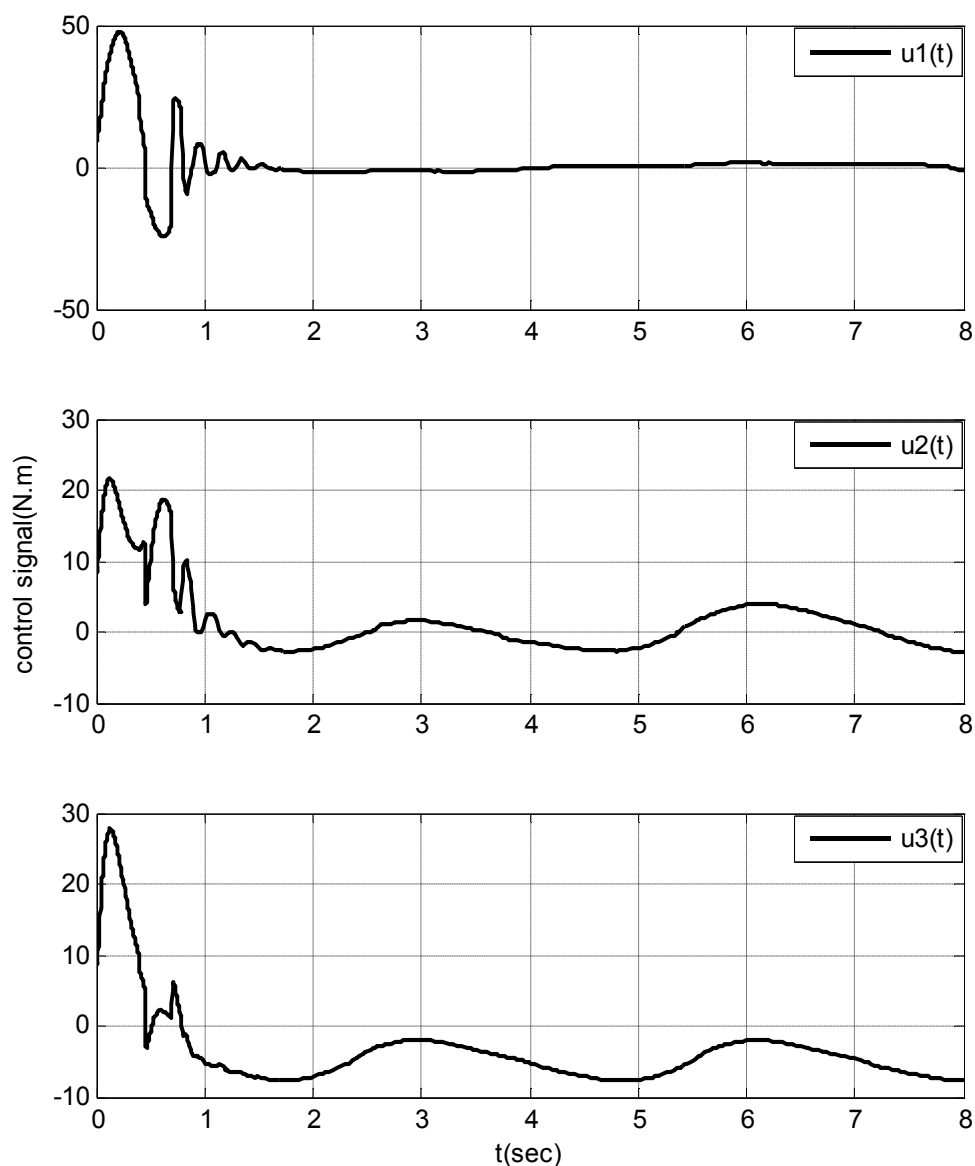
در طراحی کنترل کننده NFSMC، تعداد نرون‌های لایه اول ۸، لایه دوم ۳ و پارامتر یادگیری شبکه عصبی را $\eta = 5$ در نظر می‌گیریم. به ازای این مقادیر، موقعیت و سیگنال کنترلی به ترتیب

در شکل‌های (۵-۶) و (۵-۷) نشان داده شده است (بردار خروجی $U = [u_{eq1}, u_{eq2}, u_{eq3}]^T$ و بردار

ورودی $Z = [q_1, q_2, q_3, \dot{q}_1, \dot{q}_2, \dot{q}_3, q_{1d}, q_{2d}, q_{3d}, \dot{q}_{1d}, \dot{q}_{2d}, \dot{q}_{3d}]^T$ شبکه عصبی).



شکل ۵-۶- مسیر مطلوب (نقطه چین) و زاویه مفصل (خط) حاصل از NFSMC-MLP



شکل ۵-۷- گشتاور ورودی حاصل از NFSMC-MLP

همانطور که در شکل ۵-۶ نشان داده شده، به ازای کنترل کننده NFSMC، ردیابی به نحو مطلوب و مستقل از مدل سیستم انجام می‌گیرد. همچنین نوسانات سیگنال کنترلی ناشی از SMC، حذف شده است. لازم به ذکر است که ردیابی برای مفصل سوم در اکستریم‌های مقدار مرجع با کمی خطا صورت می‌گیرد (کمتر از ۰.۰۴). این خطا را می‌توان با تغییر تعداد نرون‌ها، تعداد لایه‌ها و پارامتر یادگیری کاهش داد.

در یادگیری شبکه عصبی MLP، پارامتر یادگیری μ که نشان دهنده مقیاس تغییرات ضرایب در هر مرحله از آموزش می‌باشد، تأثیر بسزایی بر کارایی الگوریتم یادگیری دارد. به عنوان مثال، مقدار بزرگ بهره یادگیری ممکن است باعث ایجاد نوسانات نامطلوب^۱ و نوسانات ناپایدار^۲ شود. همچنین مقادیر کوچک بهره یادگیری، باعث کند شدن الگوریتم یادگیری می‌گردد [۶۷].

در روش کنترلی NFSMC، نرخ یادگیری شبکه عصبی MLP با دامنه سیگنال کنترلی و خطای ردیابی ارتباط مستقیم دارد. بطوری که اگر مقدار μ را زیاد بگیریم دامنه سیگنال کنترلی افزایش یافته (کران بالا) و خطای ردیابی کمتر می‌شود (کران پایین) و بالعکس. اما همانطور که در بند قبل اشاره شد، مقدار بزرگ μ نیز ممکن است باعث نوسانی شدن سیستم گردد. بنابراین برای کاهش خطا و انتخاب سیگنال کنترلی مطلوب، در ابتدا مقدار μ را کوچک در نظر می‌گیریم و آن را تا حدی که خطا حذف گردد افزایش می‌دهیم.

۵-۲-۳-۳- تقریب قانون کنترل معادل با شبکه عصبی RBF

شبکه عصبی RBF در حالت استاندارد شامل سه لایه با نقش‌های کاملاً متفاوت است. لایه ورودی، شامل گره‌هایی از ورودی‌هایی است که به شبکه متصل می‌شوند. لایه اول در واقع همان بردار ورودی بوده و دارای نرون نیست. لایه دوم (لایه پنهان) تبدیل غیرخطی‌ای است که ورودی‌ها را به فضای پنهان^۳ می‌نگارد و به همین دلیل، در بسیاری از کاربردها، لایه پنهان دیمانسیون بالایی^۴ دارد. و در نهایت، لایه خروجی که خطی بوده و خروجی لایه پنهان را مدیریت می‌کند.

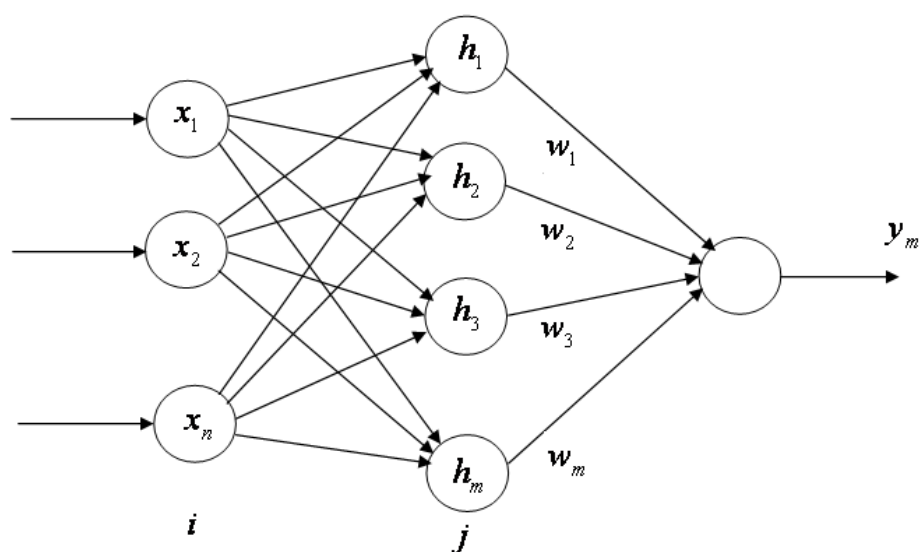
^۱ chattering

^۲ Unstable oscillations

^۳ Hidden space

^۴ High dimensionality

شبکه RBF، نگاشت مابین ورودی‌ها و خروجی‌ها را با استفاده از تقریبگرهای محلی^۱ تعیین می‌کند که به عنوان یکی از مزیت‌های مهم این شبکه است. این شبکه به دلیل اینکه در مقایسه با MLP ضرایب کمتری دارد، به سرعت آموزش می‌بیند و به نمونه‌های یادگیری کمتری نیاز دارد. ساختار کلی شبکه RBF در شکل ۸-۵ نشان داده شده است. در این شبکه تابع فعالیت لایه میانی به صورت تابع کرنل^۲ فاصله مابین بردار ورودی و بردار مراکز تعریف می‌شود و بیانگر رابطه میان لایه ورودی و پنهان می‌باشد. فرم نرمال تابع فاصله RBF به صورت $h_j(\|X - C_j\|)$ تعریف می‌شود که وابسته به فاصله مابین بردار ورودی X و بردار مراکز C_j می‌باشد. معمولاً h_j نرم اقلیدسی در نظر گرفته می‌شود [۷۵-۷۶].



شکل ۸-۵- ساختار شبکه عصبی RBF [۷۵]

در شکل فوق، خروجی شبکه به صورت ترکیب خطی از توابع پایه تقریب زده می‌شود.

^۱ Local approximators

^۲ Kernel function

$$y_m(X) = \sum_{j=1}^m w_{kj} h_j (\|X - C_j\|) \quad (30-5)$$

که در آن h_j تابع پایه، w_{kj} بردار ضرایب و m تعداد گره‌های لایه پنهان می‌باشند. فرم معمول توابع پایه مورد استفاده در تابع گوسین را بشکل زیر می‌توان نشان داد:

$$h_j(\|X - C_j\|) = \exp\left(-\frac{\|X - C_j\|^2}{2\sigma_j^2}\right), X = [x_1, x_2, \dots, x_n]^T, j = 1, 2, \dots, m \quad (31-5)$$

در رابطه فوق $\sigma = [\sigma_1 \quad \sigma_2 \quad \dots \quad \sigma_m]$ و $C_j = \begin{bmatrix} C_{j1} & \dots & C_{jn} \\ \vdots & \ddots & \vdots \\ C_{j1} & \dots & C_{jn} \end{bmatrix}$ به ترتیب بردار ضریب

پراکندگی و ماتریس موقعیت مراکز توابع گوسین هستند. با توجه به شکل ۵-۸ خروجی شبکه عصبی بشکل زیر نوشته می‌شود:

$$y_m = w_1 h_1 + w_2 h_2 + \dots + w_j h_j + \dots + w_m h_m = W.H \quad (32-5)$$

همانند شبکه MLP، برای شبکه RBF نیز می‌توانیم با تعریف تابع انرژی،

$$J = \frac{1}{2} (y_d(k) - y_m(k))^2 \quad (33-5)$$

و با استفاده از الگوریتم گرادیان کاهشی (۳۴-۵)، مقادیر پارامترهای w ، c و σ را تنظیم کنیم.

$$\dot{w}_i = -\eta_w \frac{\partial J}{\partial w_i} \quad \dot{C}_{ij} = -\eta_c \frac{\partial J}{\partial C_{ij}} \quad \dot{\sigma}_i = -\eta_\sigma \frac{\partial J}{\partial \sigma_i} \quad (34-5)$$

اما تابع انرژی (۳۳-۵)، برای شبکه RBF معمول نمی‌باشد و در اکثر مقالات کنترلی، تابع انرژی شبکه

RBF بصورت (۳۶-۵) تعریف می‌گردد. همچنین نوع ورودی‌ها را از $X = [x_1, x_2, \dots, x_n]^T$ به

$s = [s_1, s_2, \dots, s_f]^T$ تغییر می‌دهیم که در آن f تعداد سطوح لغزش می‌باشد [۷۵]. بنابراین معادلات توصیف کننده شبکه مورد نظر را برای بازوی ربات سه رابط بصورت زیر تعریف می‌نماییم:

$$\begin{aligned} u_{EQ1} &= w_{11}h_{11} + w_{12}h_{12} + \dots + w_{1m}h_{1m} \\ u_{EQ2} &= w_{21}h_{21} + w_{22}h_{22} + \dots + w_{2m}h_{2m} \\ u_{EQ3} &= w_{31}h_{31} + w_{32}h_{32} + \dots + w_{3m}h_{3m} \end{aligned} \quad (۳۵-۵)$$

در عبارت فوق، $h_{ij} = \exp(-\frac{\|s_i - C_j\|^2}{\sigma_j})$ و m تعداد نرون‌های شبکه می‌باشد. در ادامه برای

اینکه عملکرد کنترل کننده کندتر نگردد، به تنظیم w_{ij} ها بسنده کرده و C_j و σ_j ها مقادیر ثابتی خواهند داشت. قانون تطبیقی برای تنظیم ضرایب w_{ij} ، به منظور دسترسی به مقادیر وزنی بهینه و به دست آوردن همگرایی پایدار کنترل کننده، تنظیم می‌شوند. شبکه RBFN با استفاده از ضرایب تنظیم شده و قانون تطبیقی، می‌تواند نگاشت غیرخطی مابین متغیر ورودی لغزشی و قانون کنترل را تقریب بزند.

بر اساس قضیه لیاپانوف، هدف قانون کنترل، رساندن $s^T(t)\dot{s}(t) \rightarrow 0$ می‌باشد. پس توابع وزنی با استفاده از روش زیر تنظیم می‌شوند [۷۵-۷۶]. برای این کار، تابع انرژی را بصورت

$$J(t) = s^T(t)\dot{s}(t) \quad (۳۶-۵)$$

در نظر می‌گیریم. از روش گرادیان کاهشی و قاعده زنجیرهای، برای ضرایب شبکه عصبی می‌توان نوشت:

$$dw_{ij} = -\eta \frac{\partial J(t)}{\partial w_{ij}(t)} = -\eta \frac{\partial J(t)}{\partial u_{EQ_i}(t)} \frac{\partial u_{EQ_i}(t)}{\partial w_{ij}(t)} \quad (۳۷-۵)$$

در عبارت فوق، η پارامتر یادگیری می‌باشد. از رابطه (۳-۵)، می‌دانیم که سطح لغزش مستقل از

$$u_{EQ_i}(t) \text{ می‌باشد. بنابراین } \frac{\partial s_i}{\partial u_{EQ_i}} = 0 \text{ خواهد بود. لذا،}$$

$$\frac{\partial J(t)}{\partial u_{EQ_i}} = \frac{\partial s_i(t)}{\partial u_{EQ_i}} \dot{s}_i(t) + s_i(t) \frac{\partial \dot{s}_i(t)}{\partial u_{EQ_i}} = 0 + s_i(t) \frac{\partial \dot{s}_i(t)}{\partial u_{EQ_i}} \quad (۳۸-۵)$$

همچنین با جایگذاری (۱-۵) در مشتق سطح لغزش،

$$\dot{s}(t) = \ddot{q} - \ddot{q}_d + \lambda \dot{e} = M^{-1}(u - C\dot{q} - G - F) - \ddot{q}_d + \lambda \dot{e} \quad (۳۹-۵)$$

بدست می‌آید. با توجه به اینکه $u = u_{EQ} + u_{SW}$ ، و با جایگذاری (۳۸-۵) در (۳۷-۵) داریم:

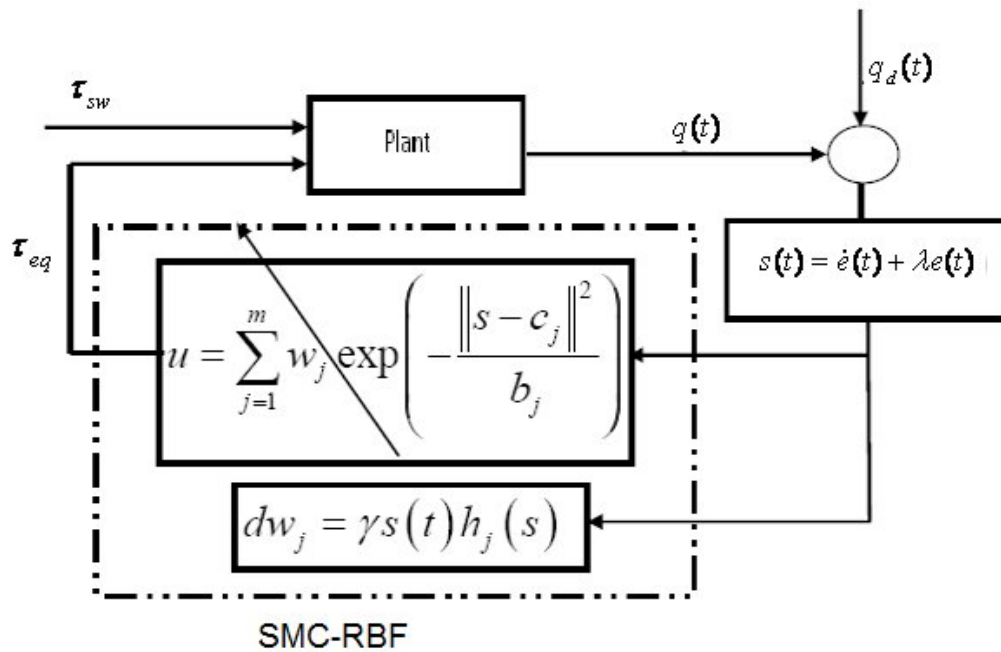
$$\frac{\partial J(t)}{\partial u_{EQ_i}} = -as_i(t) \quad (۴۰-۵)$$

در رابطه بالا، a درایه ای از ماتریس M^{-1} بوده که مقدار آن معلوم نمی‌باشد (پارامتر سیستم).

سرانجام، با جایگزینی (۳۹-۵) در (۳۶-۵) و فرض $\gamma = \eta a$ ، الگوریتم یادگیری شبکه RBF بشکل،

$$dw_{ij} = \gamma s_i(t) h_{ij}(t) \quad (۴۱-۵)$$

بدست می‌آید. شکل ۹-۵، ساختار کامل سیستم کنترلی SMC-RBF را نشان می‌دهد.



شکل ۵-۹- کنترل مد لغزش، بر پایه شبکه عصبی RBF [۷۵]

۵-۲-۳-۳-۴- نتایج شبیه‌سازی به ازای کنترل کننده NFSMC-RBF

به منظور تقریب قانون کنترل با استفاده از شبکه RBF، در مجموع از ۱۵ نرون استفاده شده که سهم هر مفصل ۵ نرون می‌باشد (می‌توانستیم برای هر سه مفصل کل نرون‌ها را ۵ عدد انتخاب کنیم؛

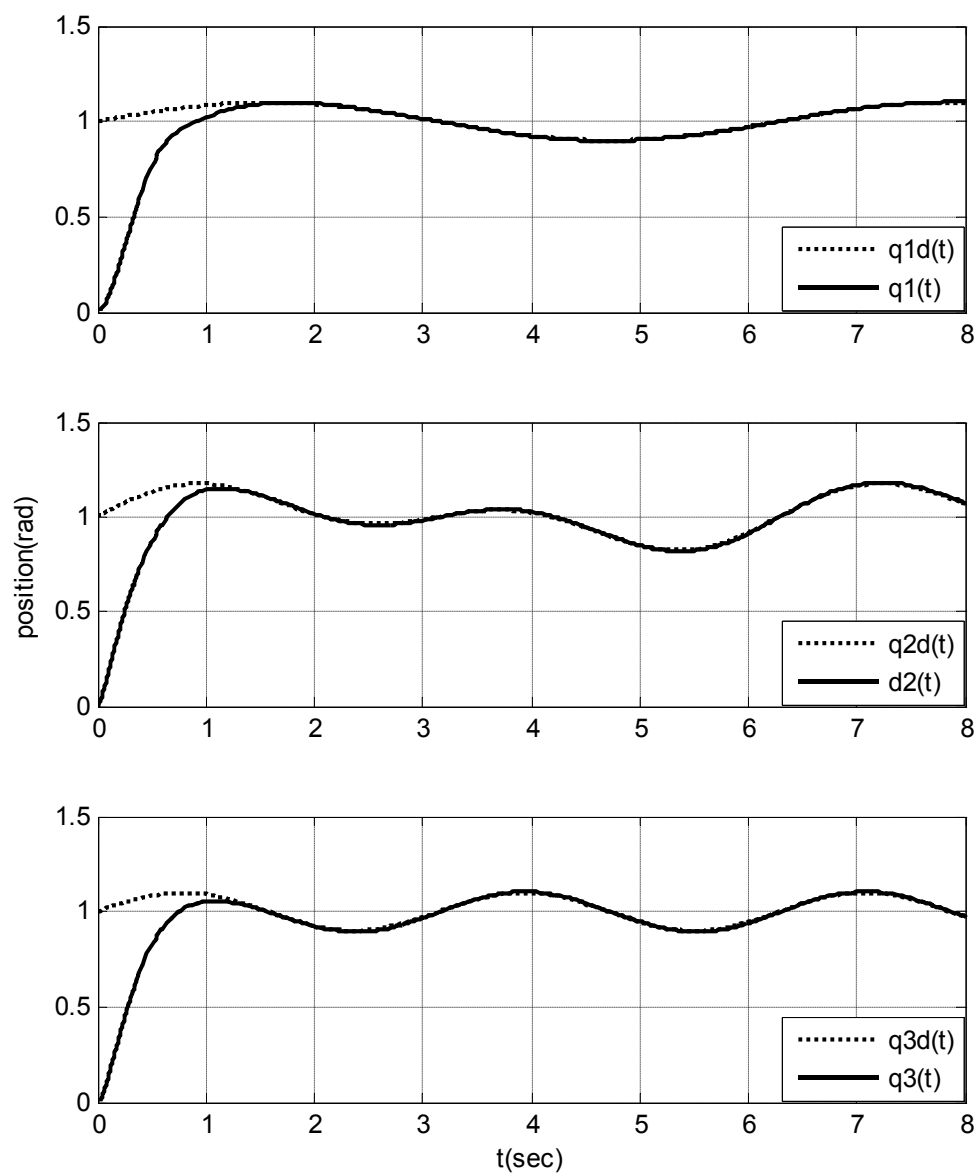
ولی در این حالت عبارت $h = \exp(-\frac{\|s - C\|^2}{\sigma})$ ، در لحظات اولیه تقریباً صفر می‌شود که این

مطلوب نمی‌باشد). مقادیر پارامترهای توابع گوسین را نیز بشکل زیر در نظر می‌گیریم:

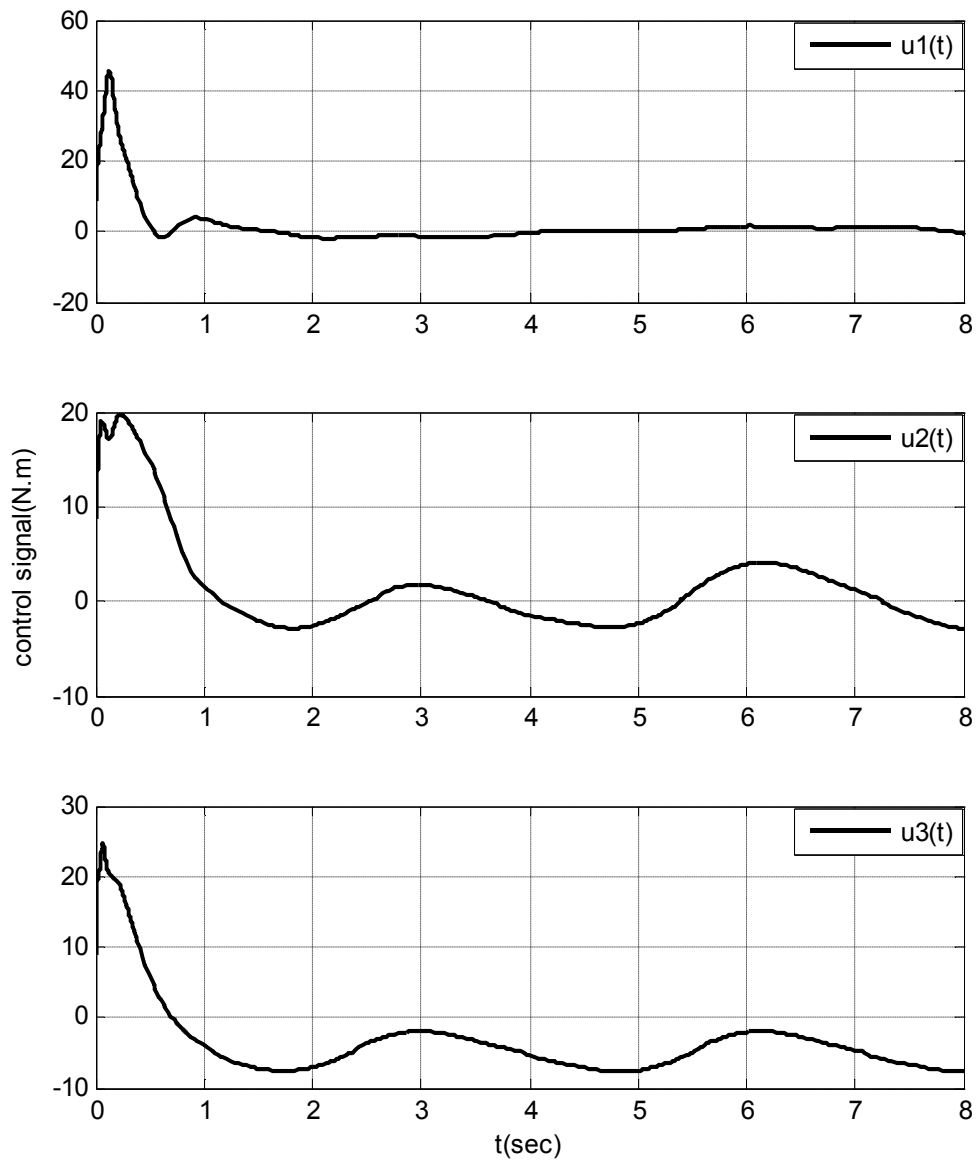
$$C = \begin{bmatrix} -3 & -1.5 & 0 & 1.5 & 3 \\ -3 & -1.5 & 0 & 1.5 & 3 \\ -3 & -1.5 & 0 & 1.5 & 3 \end{bmatrix} \quad \sigma = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \end{bmatrix} \quad \gamma = 0.6$$

همانند بخش قبل، بردار موقعیت و سیگنال کنترلی مفاصل به ترتیب در شکل‌های ۵-۱۰ و ۵-۱۱

نشان داده شده است.



شکل ۵-۱۰- مسیر مطلوب (نقطه چین) و زاویه مفصل (خط) حاصل از NFSMC-RBF



شکل ۵-۱۱- گشتاور ورودی حاصل از NFSMC-RBF

با توجه به اینکه توابع انرژی $(J_{MLP} = \frac{1}{2}(U_{EQ} - U_{NN})^T(U_{EQ} - U_{NN}))$ و $(J_{RBF} = s^T \dot{s})$

اختلاف قانون کنترل واقعی و مطلوب $(U_{EQ} - U_{MLP} = u_{Fuzzy})$ و $(U_{EQ} - U_{MLP} = s)$ و تعداد

نرون‌ها را برای شبکه های عصبی MLP و RBF در کنترل کننده های NFSMC-MLP و NFSMC-

RBF متفاوت در نظر گرفتیم؛ لذا مقایسه این دو کنترل کننده چندان منطقی به نظر نمی رسد. اما

در مجموع، بدلیل آنکه کنترل کننده NFSMC-MLP برای تنظیم شبکه MLP از خروجی سیستم فازی بهره می‌جوید، ردیابی مطلوبتری (خطای کمتر) خواهد داشت. همچنین با توجه به همگرایی سریعتر شبکه RBF نسبت به MLP، کنترل کننده NFSMC-RBF با تلاش کنترلی کمتری نسبت به NFSMC-MLP به جواب می‌رسد.

در نهایت، ما موفق به طراحی کنترل کننده NFSMC، مقاوم در برابر اغتشاش خارجی و مستقل از مدل به ازای دو شبکه عصبی MLP و RBF شدیم.

۵-۲-۳-۳-۵- موارد مهم در طراحی کنترل کننده NFSMC

۱- در طراحی شبکه‌های عصبی MLP و RBF، با افزایش تعداد نرون و لایه‌ها ممکن است خطای ردیابی کمتر شود. اما افزایش تعداد نرون‌ها و لایه‌ها باعث کند شدن کنترل کننده آنلاین می‌گردد و ممکن است باعث ناپایداری گردد. در نتیجه شمار نرون‌ها باید در حد مطلوب انتخاب گردد بطوریکه بین خطای مدلسازی و حجم محاسبات تعادل برقرار باشد.

۲- مسئله مذکور در بند قبل برای توابع عضویت (قوانین فازی) سیستم فازی نیز برقرار می‌باشد. در واقع تعداد نرون‌ها در شبکه عصبی با تعداد قوانین فازی متناظر می‌باشند. اگر تعداد توابع عضویت زیادتر انتخاب شوند، دقت تقریب بهره لغزشی افزایش می‌یابد؛ ولی باعث کند شدن کنترل کننده می‌گردد. طبیعی است که تعداد توابع عضویت کم هم دقت تقریب پایینی خواهند داشت. در نتیجه بهتر است که تعداد توابع عضویت (قوانین) همانند نرون‌ها طوری انتخاب گردند که بین سرعت و دقت تقریب بالانس برقرار شود.

۳- مسئله مهم دیگری که در طراحی با آن روبرو هستیم، مقداردهی اولیه شبکه عصبی می‌باشد. مثلاً برای شبکه MLP، به ازای شرایط اولیه ماتریسی $V(0)=0$ و $W(0)=0$ در رابطه (۵-۲۷)، خروجی شبکه عصبی صفر خواهد بود. در واقع شبکه عصبی تحریک نمی‌شود.

گردد. این مسئله به راحتی و با مقداردهی به یکی از ماتریس‌های W یا V در بازه $[۱ \ ۱-]$ یا $[۰.۵ \ ۰.۵-]$ قابل حل است.

۵-۳- طراحی کنترل کننده زمان- گسسته بر روی بازوی ربات

همانطور که قبلاً اشاره شد، کنترل مد لغزشی گسسته در زمان در دو دهه گذشته مطرح شده [۲۲-۱۶] و در چند سال اخیر توسعه یافته است [۸۷-۸۰]. مزیت این روش کنترلی نسبت به حالت پیوسته آن، در سهولت پیاده‌سازی آن می‌باشد، بطوری که با استفاده از میکروپروسسورها و با کمترین هزینه قابل پیاده‌سازی است. علاوه بر این، در مقالات، ترکیبات شبکه‌های عصبی و سیستم‌های فازی با کنترل مد لغزشی در حوزه زمان گسسته، به مراتب کمتر از حالت زمان پیوسته آن کار شده است؛ اما از نمونه‌های بارز می‌توان به [۵۴ و ۵۶] و [۸۵-۸۳] اشاره نمود. همچنین [۸۶]، از جمله محدود کارهایی است که روش DSMC را در کنترل ربات مطرح نموده است. در این قسمت و در ادامه، به طراحی کنترل کننده DSMC، NFDSMC و تأثیر رمان نمونه‌برداری بر عملکرد سیستم حلقه بسته، بر روی بازوی رابط اسکارا می‌پردازیم (کنترل کننده‌ها و اثبات‌هایی که در ادامه این فصل بحث می‌شود، برای اولین بار در حوزه گسسته بر روی بازوی ربات پیاده شده و مختص این پایان نامه است).

۵-۳-۱- طراحی کنترل کننده مد لغزشی زمان- گسسته

با تغییر متغیر $x_1 = q_1$ ، $x_2 = \dot{q}_1$ ، $x_3 = q_2$ ، $x_4 = \dot{q}_2$ ، $x_5 = q_3$ ، $x_6 = \dot{q}_3$ ، معادله (۵-۱) بازوی ربات، در فضای حالت بصورت،

$$\begin{aligned}
\begin{bmatrix} \dot{x}_1(t) \\ \dot{x}_3(t) \\ \dot{x}_5(t) \end{bmatrix} &= \begin{bmatrix} x_2(t) \\ x_4(t) \\ x_6(t) \end{bmatrix} \\
\begin{bmatrix} \dot{x}_2(t) \\ \dot{x}_4(t) \\ \dot{x}_6(t) \end{bmatrix} &= \begin{bmatrix} M_{11} & M_{12} & M_{13} \\ M_{21} & M_{22} & M_{23} \\ M_{31} & M_{32} & M_{33} \end{bmatrix}^{-1} \left(\begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} - \begin{bmatrix} C_{11} & C_{12} & C_{13} \\ C_{21} & C_{22} & C_{23} \\ C_{31} & C_{32} & C_{33} \end{bmatrix} \begin{bmatrix} x_2(t) \\ x_4(t) \\ x_6(t) \end{bmatrix} \right. \\
&\quad \left. - \begin{bmatrix} g_1 \\ g_2 \\ g_3 \end{bmatrix} - \begin{bmatrix} F_1 \\ F_2 \\ F_3 \end{bmatrix} \right) \quad (42-5)
\end{aligned}$$

بدست می‌آید. حال به منظور طراحی کنترل کننده مد لغزشی در حوزه گسسته در زمان، باید رابطه (42-5) را در حوزه گسسته بیان کنیم. در نتیجه با استفاده از نگهدار مرتبه صفر $\dot{x}(t) = (x(kT+T) - x(kT))/T$ ، معادلات دینامیکی بازوی ربات در حوزه زمان گسسته به شکل زیر بدست می‌آیند [۸۶]:

$$\begin{aligned}
\begin{bmatrix} x_1(k+1) \\ x_3(k+1) \\ x_5(k+1) \end{bmatrix} &= \begin{bmatrix} x_1(k) \\ x_3(k) \\ x_5(k) \end{bmatrix} + T \begin{bmatrix} x_2(k) \\ x_4(k) \\ x_6(k) \end{bmatrix} \\
\begin{bmatrix} x_2(k+1) \\ x_4(k+1) \\ x_6(k+1) \end{bmatrix} &= \begin{bmatrix} x_2(k) \\ x_4(k) \\ x_6(k) \end{bmatrix} + T \begin{bmatrix} M_{11} & M_{12} & M_{13} \\ M_{21} & M_{22} & M_{23} \\ M_{31} & M_{32} & M_{33} \end{bmatrix}^{-1} \left(\begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} - \begin{bmatrix} C_{11} & C_{12} & C_{13} \\ C_{21} & C_{22} & C_{23} \\ C_{31} & C_{32} & C_{33} \end{bmatrix} \begin{bmatrix} x_2(k) \\ x_4(k) \\ x_6(k) \end{bmatrix} \right. \\
&\quad \left. \times \begin{bmatrix} x_2(k) \\ x_4(k) \\ x_6(k) \end{bmatrix} - \begin{bmatrix} g_1 \\ g_2 \\ g_3 \end{bmatrix} - \begin{bmatrix} F_1 \\ F_2 \\ F_3 \end{bmatrix} \right) \quad (43-5)
\end{aligned}$$

اکنون با فرض $X_{2N-1} = [x_1(k), x_3(k), x_5(k)]^T$ و $X_{2N} = [x_2(k), x_4(k), x_6(k)]^T$

معادلات نهایی سیستم با اعمال اغتشاش خارجی به فرم،

$$\begin{aligned}
X_{2N-1}(k+1) &= X_{2N-1}(k) + TX_{2N}(k) \\
X_{2N}(k+1) &= X_{2N}(k) + TM(X_{2N-1}(k))^{-1} [u(k) \\
&\quad - C(X_{2N-1}(k), X_{2N}(k))X_{2N}(k) - G(X_{2N-1}(k)) - F(X_{2N}(k))] \\
&\quad + u_d(k)
\end{aligned} \quad (44-5)$$

در می‌آیند. در رابطه فوق، T زمان نمونه‌برداری، x_{2i-1} و x_{2i} به ترتیب موقعیت و سرعت مفاصل i -ام می‌باشند. اینک سطح لغزش گسسته در زمان را به صورت زیر در نظر بگیرید:

$$s(k) = e_{2N}(k) + \lambda e_{2N-1}(k) \quad (45-5)$$

که در آن $e_{2N-1}(k) = [e_1(k), e_3(k), e_5(k)]^T$ ، $e_{2N}(k) = [e_2(k), e_4(k), e_6(k)]^T$ به ترتیب خطای ردیابی موقعیت و خطای ردیابی سرعت، $s(k)$ تابع لغزش و $\lambda = \text{diag}\{\lambda_1, \lambda_2, \lambda_3\}$ با عناصر مثبت می‌باشد. بنابراین با توجه به معادلات (۳۸-۲) و (۳۹-۲)، قانون لغزشی نمایی را می‌توان به صورت زیر نوشت [۵۶]:

$$s(k+1) = (I_{3 \times 3} - Tq)s(k) - T\varepsilon \text{sgn}(s(k)) \quad (46-5)$$

که در آن T زمان نمونه‌برداری و $q_1 > 0$ ، $q_2 > 0$ ، $q_3 > 0$ ، $\varepsilon_1 > 0$ ، $\varepsilon_2 > 0$ ، $\varepsilon_3 > 0$ و $1 - Tq_1 > 0$ ، $1 - Tq_2 > 0$ و $1 - Tq_3 > 0$ است. در رابطه مذکور ε نرخ رسیدن و q نرخ تقریب نامیده می‌شوند. با انتخاب مقادیر مناسب q و ε می‌توان به پاسخ مطلوب رسید. حال برای کنترل بازوی ربات سه رابط در حوزه گسسته در زمان، لم ۵-۲ را مطرح می‌کنیم.

لم ۵-۲: سیستم داده شده با معادله دینامیکی (۴۴-۵)، سطح لغزش (۴۵-۵) و (۴۶-۵) مفروض است. اگر قانون کنترل را بصورت زیر انتخاب نماییم،

$$u(k) = \hat{C}X_{2N}(k) + \hat{G} + \hat{F} + \hat{M} \left(-\lambda e_{2N}(k) + \frac{x_{(2N)d}(k+1) - x_{(2N)d}(k)}{T} - qs(k) - \varepsilon \text{sign}(s(k)) \right) \quad (47-5)$$

آنگاه به ازای نامعادله های (۵۳-۵) و (۵۵-۵)، پایداری سیستم حلقه بسته تضمین می شود.

اثبات: با جایگزین کردن رابطه (۴۷-۵) در (۴۴-۵)، معادلات خطای ردیابی بصورت،

$$e_{2N-1}(k+1) = e_{2N-1}(k) + Te_{2N}(k) \quad (48-5)$$

$$e_{2N}(k+1) = e_{2N}(k) - \Delta x_{2Nd}(k) + TM^{-1}\hat{M}(\hat{M}^{-1}\Delta h - \lambda e_{2N}(k) + \frac{1}{T}\Delta x_{2Nd}(k) - qs(k)) - TM^{-1}\hat{M}\varepsilon \text{sgn}(s(k)) \quad (49-5)$$

در می آید که در آن $\Delta h = (\hat{C} - C)X_{2N}(k) + (\hat{G} - G) + (\hat{F} - F)$ است. و از قرار دادن روابط (۵-۴۸) و (۴۹-۵) در (۴۵-۵)، می توان نوشت:

$$s(k+1) = s(k) + TM^{-1}\Delta h + T(I_{3 \times 3} - M^{-1}\hat{M}) \left(\lambda e_{2N}(k) - \frac{1}{T}\Delta x_{2Nd}(k) \right) - TqM^{-1}\hat{M}s(k) - TM^{-1}\hat{M}\varepsilon \text{sgn}(s(k)) \quad (50-5)$$

اکنون با تعریف تابع لیاپانوف $V(k) = 1/2 s^T(k)s(k)$ یا با استفاده از دو شرط $s^T(k)(s(k+1) + s(k)) > 0$ و $s^T(k)(s(k+1) - s(k)) < 0$ را تعیین می کنیم.^۱

۱- از شرط اول همگرایی $s^T(k)(s(k+1) - s(k)) < 0$ داریم:

^۱ اقتباسی از رفرنس [۸۷].

$$\begin{aligned}
& Ts^T(k)M^{-1}\Delta h + Ts^T(k)\left(I_{3\times 3} - M^{-1}\hat{M}\right)\left(\lambda e_{2N}(k) - \frac{1}{T}\Delta x_{2Nd}(k)\right) \\
& - Ts^T(k)qM^{-1}\hat{M}s(k) - Ts^T(k)M^{-1}\hat{M}\varepsilon \operatorname{sgn}(s(k)) < 0
\end{aligned} \tag{۵۱-۵}$$

با نرم‌گیری از طرفین رابطه (۵۱-۵)، $\|y\|=1$ ، $y \in R^n$ و $\|M\|=\|My\|$ و ساده‌سازی آن، کران پایین ε به شکل زیر به دست می‌آید:

$$\|\varepsilon\| > \left\| \hat{M}^{-1}\Delta h \right\| + \frac{\left\| I_{3\times 3} - M^{-1}\hat{M} \right\|}{\left\| M^{-1}\hat{M} \right\|} \left\| \lambda e_{2N}(k) - \frac{1}{T}\Delta x_{2Nd}(k) \right\| - \|q\|\|s\| \tag{۵۲-۵}$$

با فرض اینکه $\left\| I_{3\times 3} - M^{-1}\hat{M} \right\| < \delta_{M1}$ ، $\left\| M^{-1}\hat{M} \right\| < \delta_{M2}$ ، $\|q\|=q_i$ ، $q_i=q_j$ و $\left\| \hat{M}^{-1}\Delta h \right\| < \delta_h$ مقادیری کراندار و مثبت اند، در نتیجه داریم:

$$\|\varepsilon\| > \delta_h + \frac{\delta_{M1}}{\delta_{M2}} \left\| \lambda e_{2N}(k) - \frac{1}{T}\Delta x_{2Nd}(k) \right\| - q_i\|s\| \tag{۵۳-۵}$$

۲- همچنین به ازای شرط دوم $s^T(k)(s(k+1) + s(k)) > 0$ می‌توان نوشت:

$$\begin{aligned}
& 2s^T(k)s(k) + Ts^T(k)M^{-1}\Delta h \\
& + Ts^T(k)\left(I_{3\times 3} - M^{-1}\hat{M}\right)\left(\lambda e_{2N}(k) - \frac{1}{T}\Delta x_{2Nd}(k)\right) \\
& - Ts^T(k)qM^{-1}\hat{M}s(k) - Ts^T(k)M^{-1}\hat{M}\varepsilon \operatorname{sgn}(s(k)) > 0
\end{aligned} \tag{۵۴-۵}$$

به همان روش قبلی

$$\begin{aligned}
\|\varepsilon\| &< \frac{1}{\|s(k)M^{-1}\hat{M}\|} \left\| \frac{2}{T} s^T(k)s(k) + s^T(k)M^{-1}\Delta h \right. \\
&\quad \left. + s^T(k)\left(I_{3 \times 3} - M^{-1}\hat{M}\right)\left(\lambda e_{2N}(k) - \frac{1}{T}\Delta x_{2Nd}(k)\right) \right\| - q_i \|s(k)\| \\
&< \left(\frac{2}{T\delta_{M2}} - q_i \right) \|s\| + \delta_h + \frac{\delta_{M1}}{\delta_{M2}} \left\| \lambda e_{2N}(k) - \frac{1}{T}\Delta x_{2Nd}(k) \right\|
\end{aligned} \tag{۵۵-۵}$$

در نتیجه به ازای شرایط $2 - q_i T \delta_{M2} > 0$ ، (۵۳-۵) و (۵۵-۵)، پایداری سیستم حلقه بسته تضمین می‌گردد. $\square$

برای حالت ایده ال (عدم قطعیت و اغتشاش خارجی نباشد)، شروط (۵۳-۵) و (۵۵-۵) به شکل،

$$\|s\| > \frac{T}{2 - Tq_i} \|\varepsilon\| \tag{۵۶-۵}$$

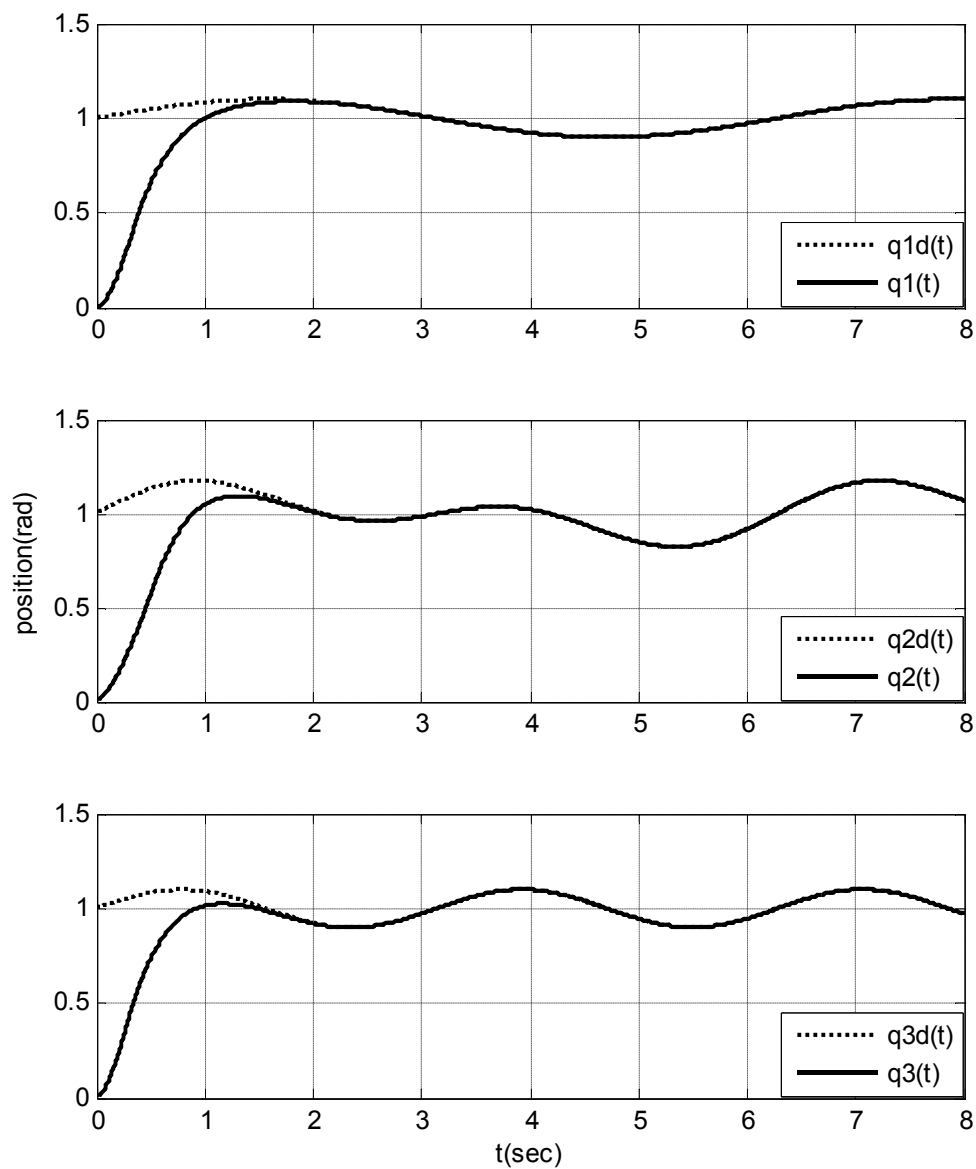
در می‌آید. در عبارت فوق سطح لغزش $s(k)$ زمانی به سمت صفر میل می‌کند که $T\|\varepsilon\|$ به سمت صفر میل نماید. اما با توجه به اینکه $\|\varepsilon\|$ و T مقداری غیر صفر می‌باشد لذا سطح لغزش $s(k)$ به سمت صفر نخواهد کرد؛ ولی هر اندازه که بتوانیم فرکانس نمونه‌برداری را بزرگتر انتخاب نماییم ($T \rightarrow 0$)، پاسخ‌های DSMC به حالت پیوسته نزدیکتر خواهد شد ($s(k) \rightarrow 0$). در بخش آخر این فصل به این ویژگی مهم می‌پردازیم.

۵-۳-۲- نتایج شبیه‌سازی به ازای کنترل کننده DSMC

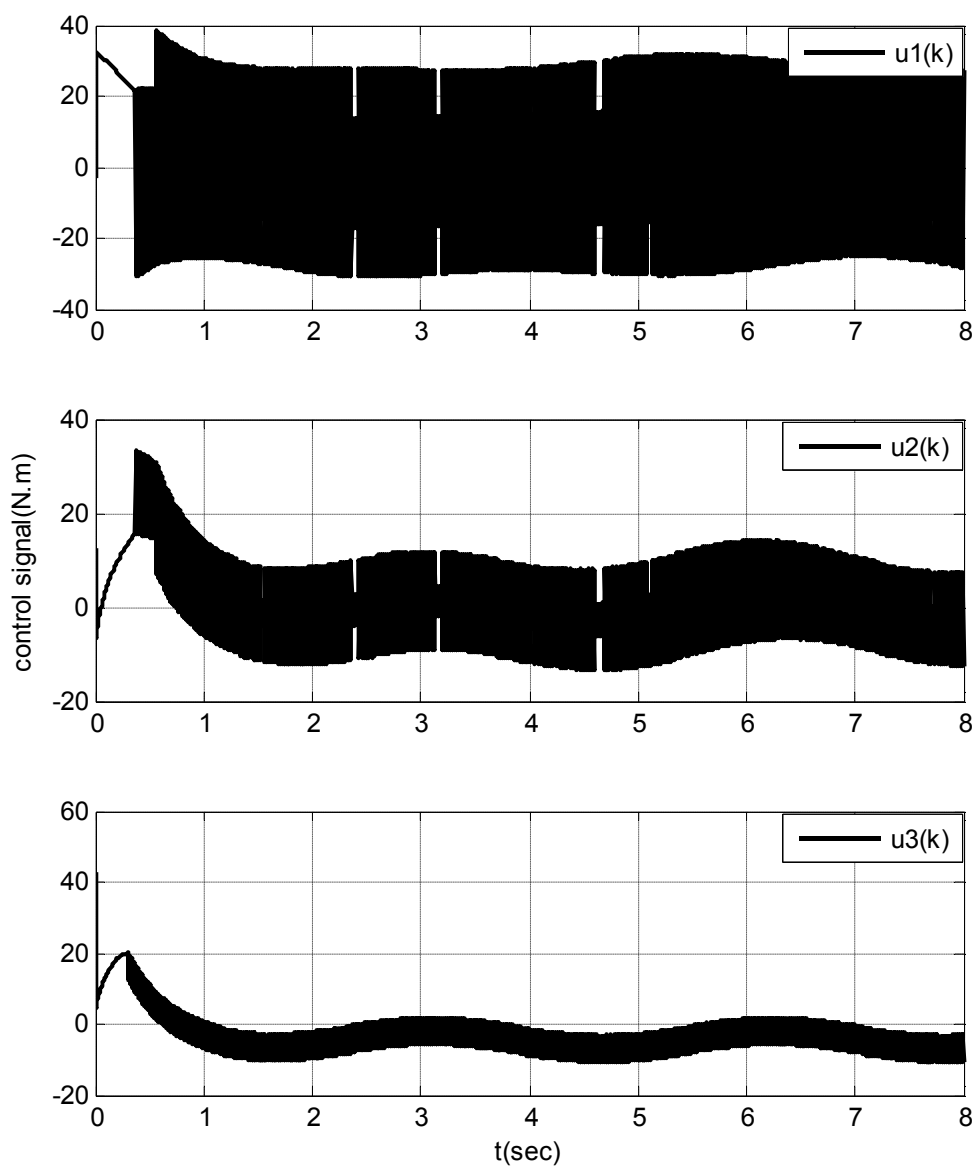
برای نمایش قابلیت کنترل کننده مذکور، پارامترهای بازوی ربات، عدم قطعیت‌ها، اغتشاش خارجی و مقادیر مرجع را همانند بخش قبل در نظر می‌گیریم. پارامترهای کنترل کننده DSMC را نیز بصورت،

$$q = \begin{bmatrix} 2 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 2 \end{bmatrix} \quad \varepsilon = \begin{bmatrix} 8 & 0 & 0 \\ 0 & 8 & 0 \\ 0 & 0 & 8 \end{bmatrix} \quad T = 2m \text{ sec}$$

در نظر می‌گیریم. شکل‌های ۱۲-۵ و ۱۳-۵ به ترتیب بردار موقعیت و سیگنال کنترلی مفاصل ربات سه رابط را نشان می‌دهد. از این شکل‌ها، مشاهده می‌شود که عملکرد کنترل کننده DSMC به ازای زمان نمونه‌برداری $T = 0.002 \text{ sec}$ مطلوب است و همانند کنترل کننده SMC از قابلیت بالایی برخوردار است.



شکل ۱۲-۵- بردار مسیر مطلوب (نقطه چین) و زاویه مفصل (خط) حاصل از DSMC



شکل ۵-۱۳- ورودی‌های کنترل حاصل از DSMC

در پایان این بخش، نکته قابل توجه‌ای که می‌توان بدان اشاره کرد، این است که قانون کنترل (۵-۴۷)، با قانون کنترل نمایی بدست آمده از شرط لغزش $\dot{s}(t) = -qs(t) - \varepsilon \operatorname{sgn}(s(t))$ ، در حوزه زمان پیوسته متناظر می‌باشد:

$$u(t) = \hat{C}X_{2N}(t) + \hat{G} + \hat{F} + \hat{M}(-\lambda e_{2N}(t) + \dot{x}_{(2N)d}(t) - qs(t) - \varepsilon \text{sign}(s(t))) \quad (57-5)$$

$$\dot{x}_{(2N)d}(t) = \frac{x_{(2N)d}(k+1) - x_{(2N)d}(k)}{T} \quad \text{که در آن } t \rightarrow k \text{ و}$$

۵-۳-۳- کنترل عصبی- فازی مد لغزشی در حوزه زمان- گسسته (NFDSMC)

همانطور که در بخش قبل مشاهده شد، کنترل کننده DSMC نیز همان معایب کنترل کننده SMC را دارا می باشد و تنها مزیتی که نسبت به آن برخوردار است، سهولت پیاده سازی عملی آن می باشد. به منظور رفع معایب کنترل کننده DSMC و طراحی کنترل کننده مستقل از مدل و متناظر با روش NFDSMC، از شبکه های عصبی و سیستم فازی استفاده می کنیم. در این قسمت صرفاً برای مطالعه تأثیر نمونه برداری بر روی کنترل کننده NFDSMC، فقط شبکه MLP را به کار برده و از بحث بر روی شبکه RBF صرف نظر می نماییم.

$$u(k) = \hat{C}X_{2N}(k) + \hat{G} + \hat{F} + \hat{M} \left(-\lambda e_{2N}(k) + \frac{x_{(2N)d}(k+1) - x_{(2N)d}(k)}{T} - qs(k) - \varepsilon \text{sign}(s(k)) \right) \quad (58-5)$$

همانند حوزه زمان پیوسته، با تعریف،

$$u(k) = u_{EQ}(k) + u_{SW}(k) \quad (59-5)$$

که در آن

$$u_{EQ}(k) = \hat{C}X_{2N}(k) + \hat{G} + \hat{F} + \hat{M} \left(-\lambda e_{2N}(k) + \frac{x_{(2N)d}(k+1) - x_{(2N)d}(k)}{T} - qs(k) \right) \quad (60-5)$$

$$u_{SW}(k) = -\varepsilon \hat{M} \operatorname{sgn}(s(k)) \quad (۶۱-۵)$$

است. اکنون به طرز مشابهی می‌توانیم مقادیر $u_{EQ}(k)$ و $u_{SW}(k)$ را با استفاده از شبکه عصبی و سیستم فازی تقریب بزنیم.

۵-۳-۳-۱- تقریب قانون کنترل معادل

برای آموزش شبکه عصبی در حوزه گسسته در زمان، باید معادلات مشتقی (۵-۲۷) را بصورت تفاضلی بیان کنیم. بنابراین داریم:

$$\begin{aligned} W(k+1) &= W(k) + \mu s(k) [\tanh(V(k)Z(k))]^T \\ V(k+1) &= V(k) + \mu (I - \tanh^2(V(k)Z(k))) W^T(k) s(k) Z^T(k) \end{aligned} \quad (۶۲-۵)$$

در رابطه فوق، μ پارامتر یادگیری حالت دیجیتال، $Z(k)$ و $s(k)$ ورودی و سطح لغزش نمونه برداری شده و $V(k)$ ، $W(k)$ ضرایب وزنی لایه اول و دوم است. همچنین توجه شود که مقادیر پارامتر یادگیری μ به ازای حالت آنالوگ و دیجیتال با همدیگر متفاوت می‌باشند. به عبارت دیگر $\mu_{analog} > 0$ و $0 < \mu_{digital} < 1$. بعلاوه با مقایسه روابط تنظیم شبکه عصبی (۵-۲۷) و (۵-۶۲)، رابطه تقریبی $\mu_{digital} = T \mu_{analog}$ برای دو حوزه آنالوگ و دیجیتال برقرار است. به عنوان مثال، اگر $\mu_{analog} = 5$ در نظر گرفته شود، برای عملکرد مشابه شبکه عصبی در حوزه زمان-گسسته به ازای $T_{sampling} = 2msec$ ، بهتر است مقدار $\mu_{digital}$ برابر 0.002×5 انتخاب گردد.

۵-۳-۳-۲- تقریب قانون کنترل کلیدزنی

ساختار سیستم فازی و توابع عضویت را همانند حالت زمان-پیوسته در نظر می‌گیریم. تفاوت اساسی در نوع ورودی و خروجی سیستم فازی می‌باشد. در واقع برای سیستم فازی از مقادیر نمونه برداری شده $s(k)$ به جای $s(t)$ ، استفاده می‌کنیم. اینک با تعریف مجموعه‌های فازی،

$$F_S(s_i(k)) = \{NB, NL, Z, PL, PB\}$$

$$F_S(u_{i \text{ FUZZY}}(k)) = \{NB, NL, Z, PL, PB\}$$

قوانین فازی برای مفصل i -ام را همانند حالت پیوسته، به شکل زیر می‌توان نوشت:

rule1: if $s_i(k)$ is NB then $u_{i \text{ FUZZY}}$ is PB

rule2: if $s_i(k)$ is NL then $u_{i \text{ FUZZY}}$ is PL

rule3: if $s_i(k)$ is Z then $u_{i \text{ FUZZY}}$ is Z

rule4: if $s_i(k)$ is PL then $u_{i \text{ FUZZY}}$ is NL

rule5: if $s_i(k)$ is PB then $u_{i \text{ FUZZY}}$ is NB

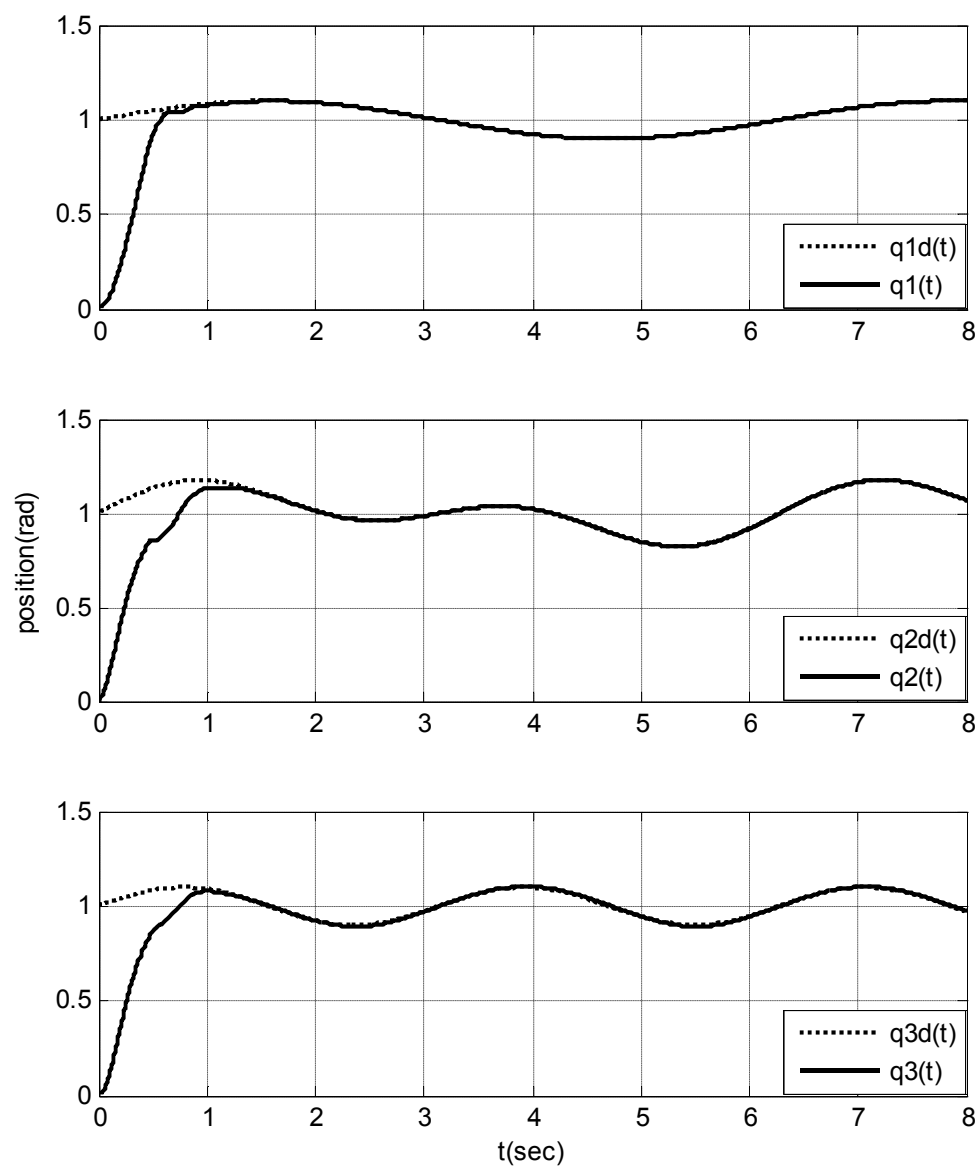
طبیعی است که خروجی سیستم فازی نیز بصورت مقادیر در فواصل نمونه برداری خواهد بود.

۵-۳-۴- نتایج شبیه‌سازی به ازای کنترل کننده NFDSMC-MLP

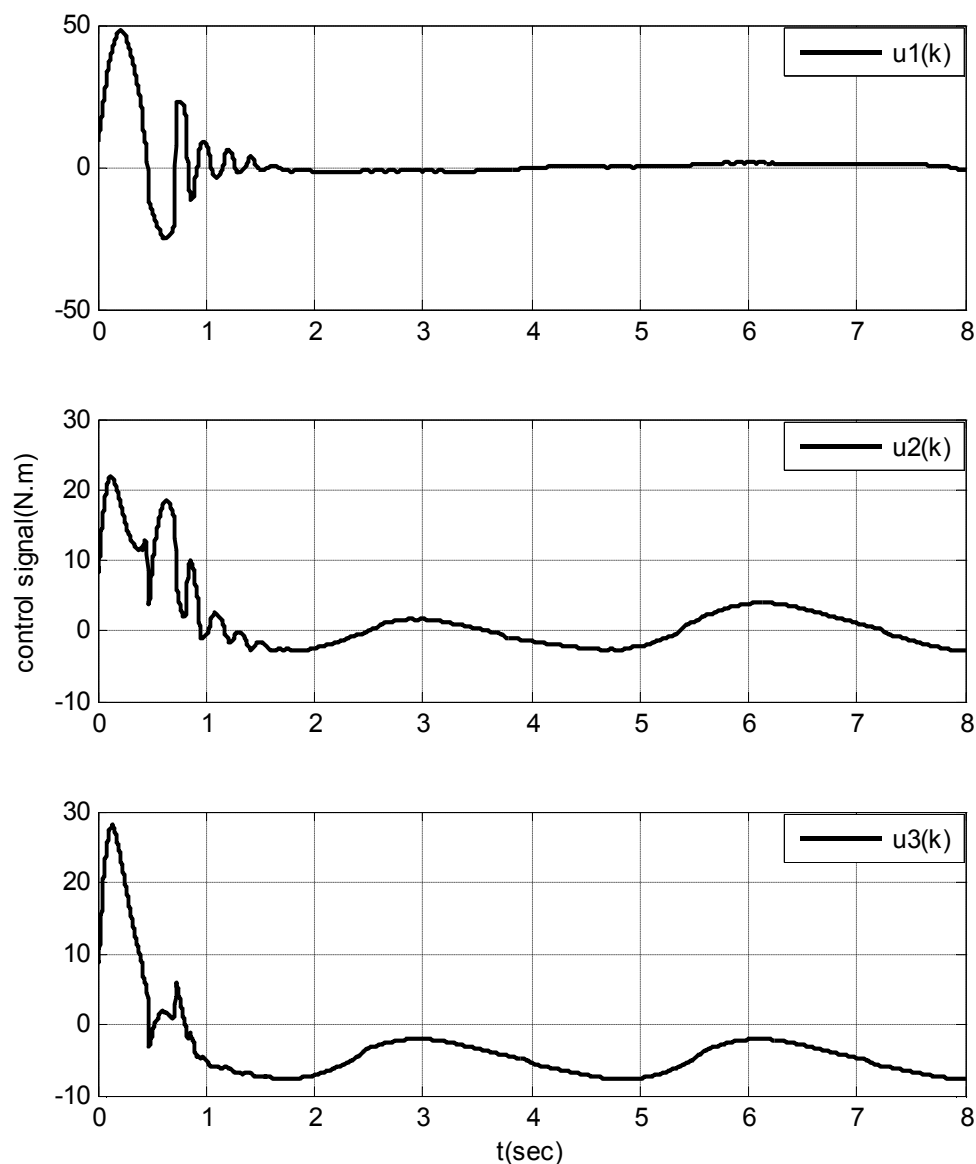
برای نشان دادن بهبودی حاصل شده توسط روش پیشنهادی در این قسمت، نتایج حاصل از

طراحی کنترل کننده NFDSMC-MLP، در شکل‌های ۵-۱۴ و ۵-۱۵ نمایش داده می‌شود. مقدار

پارامتر یادگیری را $\mu_{digit} = 0.01$ انتخاب می‌کنیم.



شکل ۵-۱۴- بردار مسیر مطلوب (نقطه چین) و زاویه مفصل (خط) حاصل از NFDSMC



شکل ۵-۱۵- ورودی‌های کنترل حاصل از NFDSMC

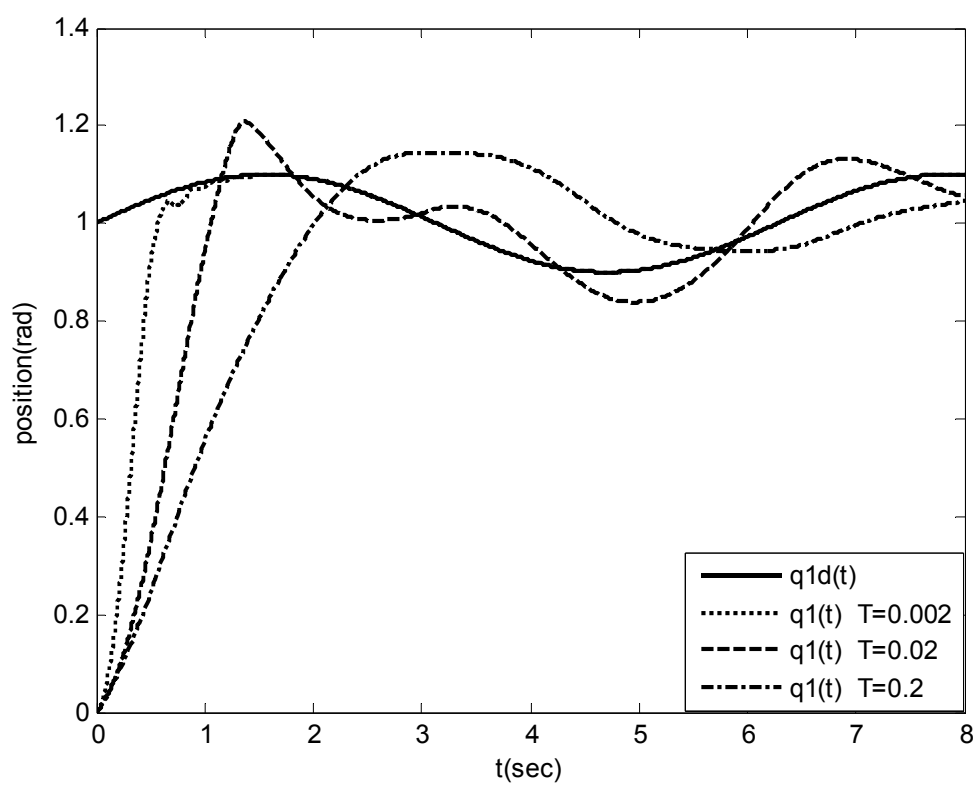
با توجه به شکل‌های ۵-۱۴ و ۵-۱۵، به این نتیجه می‌رسیم که کنترل کننده NFDSMC همانند کنترل کننده NFSMC، عملکرد قابل قبولی دارد.

۵-۳-۵- تأثیر زمان نمونه‌برداری بر خروجی، سیگنال کنترلی و همگرایی سیستم

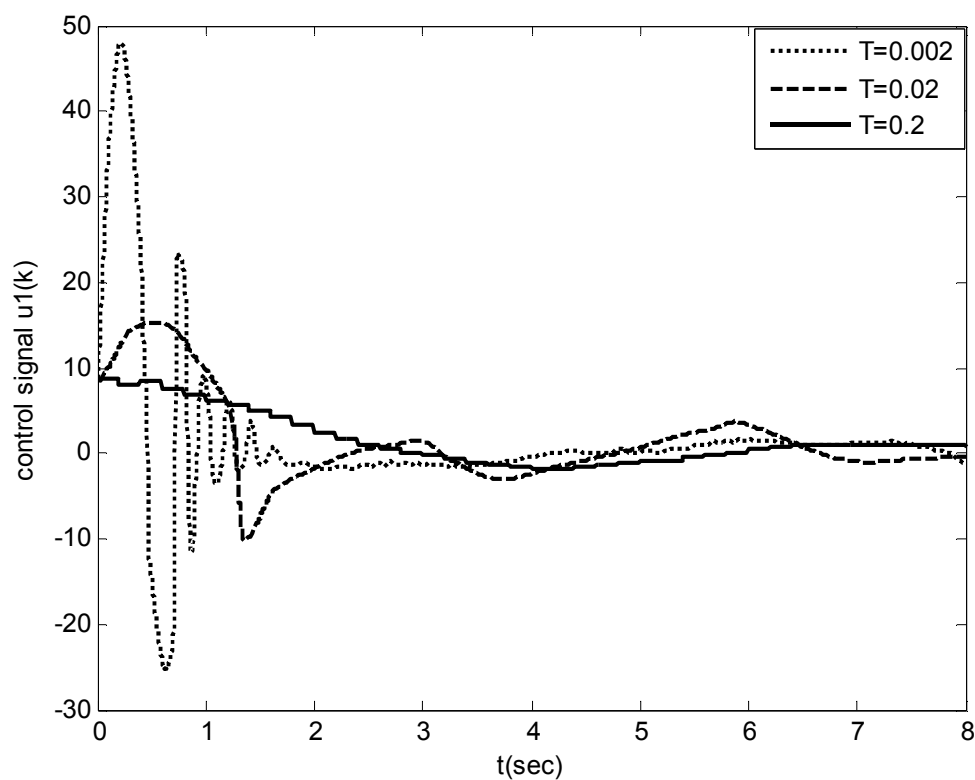
برای بررسی تأثیر زمان نمونه‌برداری بر بردار موقعیت، سیگنال کنترلی و سطح لغزش به ترتیب در شکل‌های ۵-۱۶، ۵-۱۷ و ۵-۱۸، به ازای سه زمان نمونه‌برداری $T = 0.002$ ، $T = 0.02$ و

$T = 0.2$ نشان داده شده است. همانطور که از شکل ۵-۱۶ برمی آید، هرچه زمان نمونه برداری کمتر

گردد ردیابی مطلوب تر خواهد بود.

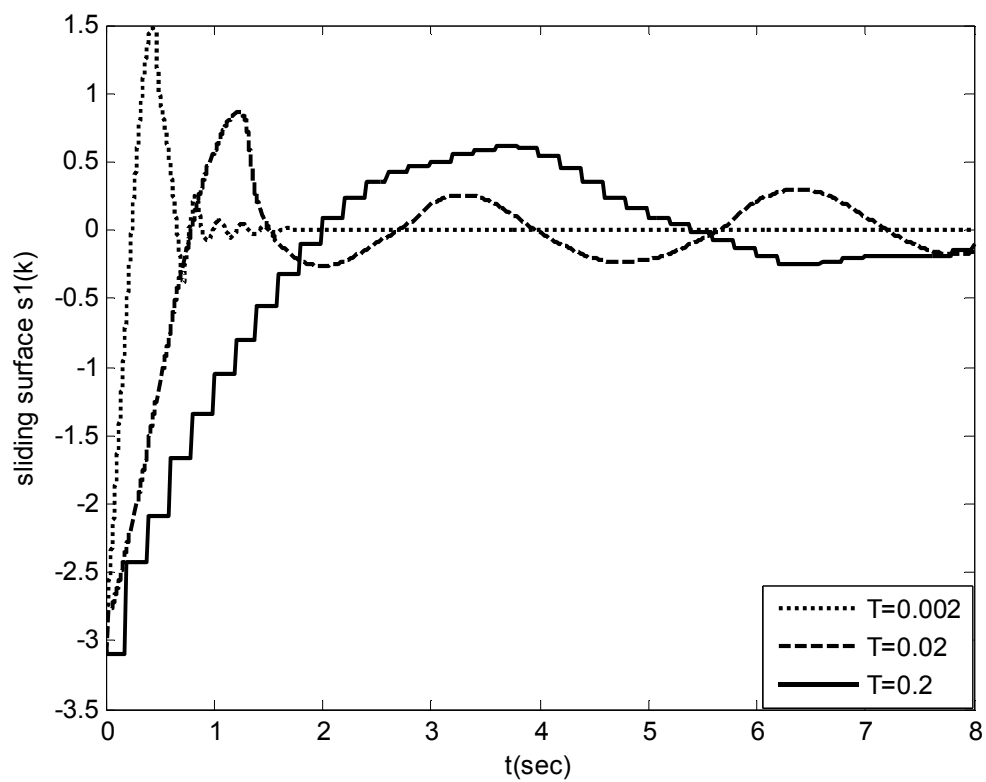


شکل ۵-۱۶- مسیر مطلوب مفصل اول و زاویه مفصل به ازای سه زمان نمونه برداری متفاوت (NFDSMC)



شکل ۶-۱۷- ورودی کنترلی مفصل اول به ازای سه زمان نمونه برداری متفاوت (NFDSMC)

همچنین با توجه به شکل ۵-۱۸، با کاهش زمان نمونه برداری T ، همگرایی سیستم حلقه بسته بهبود می یابد. این شکل در واقع تاییدی بر رابطه (۵-۵۶) در قانون کنترل (۵-۴۷) می باشد؛ اما باید توجه نمود که نرخ نمونه برداری یک سیستم کنترل، محدود به سریعترین دینامیک آن است.



شکل ۵-۱۸- سطح لغزش گسسته مفصل اول به ازای سه زمان نمونه برداری متفاوت (NFDSMC)

طراحی‌های مربوط به این فصل در محیط SIMULINK انجام شده و بلوک‌های مربوط به آن‌ها در ضمیمه (ب) آورده شده است.

فصل ششم: کنترل عصبی-مد لغزشی با سطح لغزش غیرخطی

۶-۱- مروری بر تحقیقات انجام شده

همانطور که در فصل پیش نشان داده شد، عملکرد کنترل کننده عصبی-فازی-مد لغزشی با سطح لغزش خطی، از لحاظ مقاومت در برابر اغتشاش و مستقل از مدل بودن، مطلوب می‌باشد. اما کنترل کننده مورد نظر به دلیل عملکرد همزمان شبکه عصبی و سیستم فازی، دارای حجم محاسباتی بالایی است. بنابراین به منظور کاهش حجم محاسبات و حفظ استقلال از مدل سیستم، بهتر است که یکی از تقریبگرها حذف گردد. همچنین به منظور حفظ سرعت پاسخ کنترل کننده قبلی (NFSMC)، سطح لغزش بصورت غیرخطی در نظر گرفته می‌شود. کنترل کننده طراحی شده (NTSMC)^۱ عملکردی مقاوم، سریع، مستقل از مدل با حجم محاسبات کمتر خواهد داشت. همچنین برخلاف کنترل کننده قبلی، پایداری این روش اثبات می‌شود. بدین منظور، به مرور مختصری از پیشینه کنترل مد لغزشی غیرخطی^۲ می‌پردازیم.

کنترل کننده مد لغزشی خطی، خطای سیستم را صفر می‌کند، اما دینامیک‌های خطا در زمان محدودی همگرا نمی‌شوند. برای همگرایی سریعتر می‌توان بهره کنترل کننده را افزایش داد، ولی افزایش بهره باعث اشباع سیگنال کنترلی شده و در عمل نامطلوب می‌باشد. برای رفع این مسئله

^۱ Neuro- terminal sliding mode control

^۲ Nonlinear sliding mode (Terminal sliding mode)

کنترل مد لغزش غیرخطی اولین بار توسط ونکترامن [۱۰] و من^۱ [۸۸] پیشنهاد شد. این روش همانند کنترل مد لغزشی خطی در مقابل عدم قطعیت‌ها مقاوم است. علاوه بر این، خطای ردیابی به ازای کنترل مد لغزشی غیرخطی در زمان محدودی به سمت صفر میل می‌کند. با معرفی ترم غیرخطی به سطح لغزش و طراحی مناسب کنترل کننده، خطای ردیابی و بهره کنترل کننده مد لغزشی غیرخطی به طور قابل ملاحظه‌ای در مقابل کنترل کننده مد لغزشی خطی کاهش می‌یابد. اما در کاربردهای اولیه کنترل کننده مد لغزشی، سیگنال کنترلی ممکن است نامحدود^۲ گردد [۹۰-۸۹]. این موضوع در کنترل بازوی ربات مشهود است. برای حل مشکل واگرایی سیگنال کنترلی، در [۹۲-۹۱] سطوح لغزش متفاوتی پیشنهاد شده است. به طور کلی به دلیل همگرایی سریع و ردیابی دقیق، کنترل مد لغزشی با سطح لغزش غیرخطی در سال‌های اخیر کاربرد زیادی داشته است که از آن جمله می‌توان به [۱۱-۱۵] و [۹۷-۹۲] اشاره کرد. در مقالات [۹۳-۹۲]، کنترل مد لغزشی غیرخطی به همراه تقریبگرهای عصبی یا فازی بر روی بازوی ربات پیاده شده است. در مورد اول [۹۲]، از کنترل کننده TSMC به همراه یک شبکه RBF با سطح لغزشی جدید، برای طراحی کنترل کننده مستقل از مدل و مقاوم بر روی بازوی ربات IMI Zebra بهره می‌جوید. در مورد دوم [۹۳]، شبکه عصبی-ویولت-فازی با ضرایب خطی شده به همراه کنترل کننده TSMC، بر روی PUMA-500 اعمال شده است. در این تحقیق و در ادامه، به طراحی کنترل کننده TSMC به همراه شبکه MLP دو لایه، بر روی بازوی سه رابط SCARA می‌پردازیم تا جایگزین مناسبی برای کنترل کننده NFSMC باشد.

۶-۲- تحلیل بر سطح لغزش غیرخطی

سیستم مرتبه دوم زیر را در نظر بگیرید [۴۱]:

$$\ddot{x}(t) = f(x) + g(x)u \quad (۶-۱)$$

^۱ Man

^۲ Singularity

که در آن u ورودی سیستم و $g(x)$ و $f(x)$ توابع غیرخطی از x می‌باشند. سطح لغزش غیر خطی را برای سیستم فوق به شکل زیر تعریف می‌کنیم:

$$\Psi(e) = \{e \in R \mid s(e) = \dot{e} + \Lambda(e) = 0\} \quad (2-6)$$

که در آن $e = x - x_d$ خطای ردیابی، $\Lambda(e)$ تابع غیرخطی از خطا می‌باشد، و با استفاده از لم ۲-۵ تعیین می‌شود.

لم ۱-۶: سطح لغزش Ψ تعریف شده در معادله (۲-۶) را با فرض اینکه $\Lambda(e)$ تابعی پیوسته و $e\Lambda(e) > 0, \forall e \neq 0$ در نظر بگیرید. بنابراین بر روی سطح Ψ ، خطای خروجی e ، به طور مجانبی به سمت صفر میل خواهد کرد.

اثبات: اگر حرکت لغزشی بر روی سطح لغزش باشد ($\Psi = 0$)، آنگاه از رابطه (۲-۶) می‌توان نوشت:

$$\dot{e} = -\Lambda(e) \quad (3-6)$$

همچنین با تعریف تابع لیاپانوف $V = 0.5e^2$ ، و مشتق گیری از آن داریم:

$$\dot{V}(t) = e\dot{e} = -e\Lambda(e) \quad (4-6)$$

برای آنکه رابطه (۴-۶) منفی گردد، کافی است که $e\Lambda(e) > 0, \forall e \neq 0$ باشد. در نتیجه خطای ردیابی محدود بوده و به سمت صفر میل خواهد کرد. $\square$

از جمله توابعی که در شرط $e\Lambda(e) > 0, \forall e \neq 0$ صدق می‌کنند، می‌توان به موارد زیر اشاره کرد:

$$1- \Lambda(e) = \tanh(e) \quad \text{تابع تانژانت هایپربولیک: تابعی فرد بوده و به راحتی می‌توان نشان داد}$$

که شرط مذکور را ارضاء می‌نماید [۴۱]:

$$e\Lambda(e) = \begin{cases} e \tanh(e) > 0 & e(t) > 0 \\ e \tanh(e) < 0 & e(t) < 0 \end{cases} \Rightarrow \dot{V}(t) = -e\Lambda(e) < 0 \quad (5-6)$$

۲- $\Lambda(e) = e^T$, $T = 2k + 1$, $k = 0, 1, 2, 3, \dots$ تابعی با توان طبیعی و فرد: تابع مورد نظر

به ازای $k = 0$ ، نشان دهنده سطح لغزش خطی بوده ولی به ازای $k = 0, 1, 2, 3, \dots$ بیانگر سطح لغزش غیر خطی خواهد بود.

$$e\Lambda(e) = ee^T = e^{T+1} > 0 \Rightarrow \dot{V}(t) = -e\Lambda(e) < 0 \quad (6-6)$$

سرعت سیستم حلقه بسته به ازای سطح لغزش مذکور در مقایسه با سطح لغزش خطی کندتر می باشد و با توجه به اینکه هدف از تعریف سطح لغزش غیرخطی، کاهش زمان گذرای سیستم حلقه بسته می باشد، لذا تعریف مذکور معقول نخواهد بود.

۳- $\Lambda(e) = e^p$, $p = \frac{p_1}{p_2}$, $p_2 > p_1$, p_1, p_2 odd تابعی با توان کسری و فرد:

سطح لغزش مذکور زمان همگرایی سریعتری داشته و بسیار پر کاربرد می باشد [۸۸-۸۹]

$$e\Lambda(e) = ee^{\frac{p_1}{p_2}} = e^{\frac{p_1+p_2}{p_2}} > 0 \Rightarrow \dot{V}(t) = -e\Lambda(e) < 0 \quad (7-6)$$

۶-۲-۱- زمان رسیدن به سطح لغزش

از فصل قبل، معادله دینامیکی بازوی ربات n- را می توان بصورت زیر نوشت [۹۲]:

$$\ddot{q} = M^{-1}(q)[u - C(q, \dot{q})\dot{q} - G(q) - F(\dot{q})] \quad (8-6)$$

برای طراحی کنترل مد لغزشی ترمینال، سطح لغزش غیرخطی را به شکل

$$s = \dot{e} + \Lambda e^p \quad (9-6)$$

با $e^p = [e_1^p \ \dots \ e_n^p]^T$ و $e = q - q_d$ در نظر می‌گیریم که در آن $p = \frac{p_1}{p_2}$ ، p_1 و p_2 اعداد

طبیعی فرد ($p_2 > p_1$) و Λ ماتریس قطری مثبت می‌باشد. از فرمول (۶-۹) برای بازوی i -ام، سطح

لغزش را به صورت زیر می‌نویسیم:

$$s_i = \dot{e}_i + \Lambda_{ii} e_i^p \quad (۱۰-۶)$$

بر روی سطح لغزش غیرخطی ($s_i = 0$) داریم:

$$\dot{e}_i + \Lambda_{ii} e_i^p = 0 \quad (۱۱-۶)$$

با حل معادله دیفرانسیلی فوق، زمان رسیدن به سطح لغزشی غیرخطی به شکل زیر بدست می‌آید:

$$t_{reach} = \frac{e_i(0)^{1-p}}{\Lambda_{ii}(1-p)} \quad (۱۲-۶)$$

طبیعی است که مقدار مینیمم t_{reach} ، به ازای $dt_{reach}/dp = 0$ بدست می‌آید.

۶-۲-۲- واگرایی سیگنال کنترلی

در این قسمت به طراحی قانون کنترل ربات n -رابط، به ازای سطح لغزش غیرخطی می‌پردازیم.

اگر شرط لغزش را بصورت زیر در نظر بگیریم [۹۲]:

$$\dot{s} \leq -\eta \text{Sign}(s) \quad (۱۳-۶)$$

که در آن η ضریب لغزش می‌باشد. با مشتق‌گیری از رابطه فوق

$$\dot{s} = \ddot{e} + p\Lambda \text{diag}\{e^{p-1}\}\dot{e} \quad (۱۴-۶)$$

و با استفاده از شرط لغزش (۶-۱۳)، قانون کنترل به شکل زیر بدست می‌آید:

$$u = C(q, \dot{q})\dot{q} + G(q) + F(\dot{q}) + M(q)[\ddot{q}_d - p\Lambda \text{diag}\{e^{p-1}\}\dot{e} - K\text{Sign}(s)] \quad (۱۵-۶)$$

در رابطه فوق، K بهره لغزش مطلوب بوده و محدوده آن از شرط لیاپانوف تعیین می‌گردد. قانون کنترل (۱۵-۶) به دست آمده از سطح لغزش غیر خطی، به ازای دو حالت زیر ممکن است بی‌کران گردد:

۱- زمانی که حرکت لغزشی بر روی سطح لغزش می‌باشد ($s = 0$): در این حالت با جایگذاری

$$\dot{e} = -\Lambda e^p$$

داریم:

$$u = C(q, \dot{q})\dot{q} + G(q) + F(\dot{q}) + M(q)[\ddot{q}_d - p\Lambda \text{diag}\{e^{2p-1}\}\Lambda - K\text{Sign}(s)] \quad (۱۶-۶)$$

برای آنکه عبارت فوق، به ازای $e = 0$ واگرا نگردد، باید $2p - 1 > 0$ انتخاب شود. بنابراین باید مقدار p بصورت $0.5 < p < 1$ انتخاب گردد.

۲- زمانی که حرکت در فاز رسیدن به سطح لغزش باشد: به عبارت دیگر زمانی که از سطح

لغزش فاصله داریم ($s \neq 0$). در این حالت اگر خطا صفر گردد ($e = 0$)، قانون کنترلی (۶-۶)

(۱۵) به ازای $0.5 < p < 1$ واگرا خواهد شد.

بنابراین سطح لغزشی غیرخطی تعریف شده در فرمول (۹-۶)، باید به گونه‌ای تصحیح گردد. برای فائق

آمدن بر این مسئله، چندین مورد سطح لغزشی غیرخطی پیشنهاد شده است که از بارزترین آنها می‌توان به دو مورد زیر اشاره کرد:

۱- در [۹۱]، سطح لغزش غیرخطی

$$s = e + \Lambda \dot{e}^p \quad (۱۷-۶)$$

پیشنهاد شد. سطح لغزش مذکور از سال ۲۰۰۲ به بعد بسیار پرکاربرد است، و مقالات [۱۱-۱۵] می‌تواند شاهدهی بر این مدعا باشد.

۲- برای کنترل بازوی ماهر n-رابط، سطح لغزش غیرخطی

$$s = \dot{e} + \Lambda e^p = \dot{q} - \dot{q}_r \quad (18-6)$$

در [۹۲] پیشنهاد شد. که در آن $\dot{q}_{ri} = \dot{q}_{di} + \lambda_i(e_i)$ $i=1,2,\dots,n$ و پارامتر $\lambda(e) = [\lambda_1(e_1) \quad \lambda_2(e_2) \quad \dots \quad \lambda_n(e_n)]^T$ و به صورت زیر تعریف می‌شود:

$$\lambda_i(e_i) = \begin{cases} \Lambda_{ii} e_i^p & \text{if } S_i = 0 \text{ or } S_i \neq 0, |e_i| > e_{si} \\ K_{1i} e_i + K_{2i} e_i^2 & \text{if } S_i \neq 0, 0 < e_i \leq e_{si} \\ K_{1i} e_i - K_{2i} e_i^2 & \text{if } S_i \neq 0, -e_{si} \leq e_i \leq 0 \end{cases} \quad (19-6)$$

که در آن $K_{1i} = (2-p)e_{si}^{p-1}$ ، $K_{2i} = (p-1)e_{si}^{p-2}$ ، e_{si} ثابت مثبت است. با تعریف سطوح لغزش جدید، سیگنال کنترلی چه در فاز رسیدن و چه بر روی سطح لغزش کراندار می‌ماند.

۳-۶- طراحی کنترل کننده مد لغزش غیرخطی کراندار TSMC

به منظور طراحی کنترل کننده مد لغزشی غیرخطی $s(t)$ ، ضروری است که سطح لغزش غیرخطی را برای بازوی ماهر n-رابط بصورت زیر تعریف کنیم [۹۳]:

$$s = e + \lambda^{-1} \dot{e}^p \quad (20-6)$$

که در آن $e(t) = q(t) - q_d(t)$ بردار خطای ردیابی بازوی ربات، $1 < p < 2$ و گویا، $\dot{e}^p = [\dot{e}_1^p \quad \dot{e}_2^p \quad \dots \quad \dot{e}_n^p]^T$ ، $\lambda = \text{diag}\{\lambda_1, \lambda_2, \dots, \lambda_n\}$ (ماتریس مثبت و قطری) به ازای $i = 1, \dots, n$ در ادامه، مشتق $s(t)$ به شکل،

$$\dot{s} = \dot{e} + p\lambda^{-1} \text{diag}\{\dot{e}^{p-1}\} \ddot{e} \quad (21-6)$$

بدست می‌آید. عبارت فوق به ازای $1 < p < 2$ کراندار خواهد بود. اکنون، با بازنویسی رابطه (۵-۱)، بصورت زیر بدست می‌آید:

$$\ddot{e} = u + f(q, \dot{q}, \ddot{q}) - \ddot{q}_d - \ddot{u}_d \quad (۲۲-۶)$$

در رابطه بالا، تابع $f(q, \dot{q}, \ddot{q})$ به شکل،

$$f(q, \dot{q}, \ddot{q}) = (I_{n \times n} - M(q))\ddot{q} - C(q, \dot{q})\dot{q} - G(q) - F(\dot{q}) \quad (۲۳-۶)$$

تعریف می‌گردد. فرض می‌کنیم که $\hat{f}(q, \dot{q}, \ddot{q})$ به عنوان تقریبی از $f(q, \dot{q}, \ddot{q})$ در دسترس بوده و $\Delta f = f - \hat{f}$ معلوم باشد. اکنون طراحی کنترل کننده TSMC را توسط لم ۶-۱ بیان می‌کنیم.

لم ۶-۲: سیستم داده شده با معادله دینامیکی (۵-۱)، سطح لغزش غیرخطی (۶-۲۰) و معادله (۶-۲۲) مفروض است. اگر قانون کنترل بصورت،

$$u = -\hat{f}(q, \dot{q}, \ddot{q}) + \ddot{q}_d - \frac{1}{p} \lambda \text{diag}\{\dot{e}^{1-p}\} \dot{e} - K \text{sgn}(s) \quad (۲۴-۶)$$

انتخاب شود. آنگاه e و $\dot{e}$ در زمان محدودی به سمت صفر میل خواهد کرد.

اثبات: تابع منتخب لیاپانوف زیر را در نظر بگیرید:

$$V = \frac{1}{2} s^T s \quad (۲۵-۶)$$

با مشتق‌گیری از رابطه فوق و جایگزینی معادله (۶-۲۱) در آن داریم:

$$\dot{V} = s^T \dot{s} = s^T (\dot{e} + p \lambda^{-1} \text{diag}\{\dot{e}^{p-1}\} \ddot{e}) \quad (۲۶-۶)$$

از جایگذاری رابطه (۶-۲۴) در (۶-۲۲)، می‌توان نوشت:

$$\ddot{e} = \Delta f - \frac{1}{p} \lambda \dot{e}^{2-p} - K \operatorname{sgn}(s) - u_d \quad (27-6)$$

بنابراین با اعمال رابطه (27-6) به رابطه (26-6)، داریم:

$$\dot{V} = p s^T \lambda^{-1} \operatorname{diag}\{\dot{e}^{p-1}\} (\Delta f - K \operatorname{sgn}(s) - u_d) \quad (28-6)$$

و با ساده‌سازی آن

$$\dot{V} = p \lambda \operatorname{diag}\{\dot{e}^{p-1}\} (s^T (\Delta f - u_d) - s^T K \operatorname{sgn}(s)) \quad (29-6)$$

سرانجام با انتخاب $K_{ii} \geq |\Delta f|_i + U_{Di} + \eta_i$ ، می‌توان نتیجه گرفت:

$$\dot{V} = -p \lambda \operatorname{diag}\{\dot{e}^{p-1}\} \eta |s| \quad (30-6)$$

با توجه به اینکه $p-1 = \frac{p_1 - p_2}{p_2}$ ، $p = \frac{p_1}{p_2}$ ، $p_1 > p_2$ مقدار مثبت و زوج است، لذا

$$\dot{V} \leq 0 \text{ در نتیجه } \dot{e}_i^{p-1} > 0, \quad i = 1, 2, \dots, n \quad \square \text{ خواهد بود.}$$

۴-۶- نتایج شبیه‌سازی به ازای کنترل کننده TSMC

برای نشان دادن کارایی قانون کنترل پیشنهادی در این بخش، آن را در مورد بازوی سه رابط اسکارا با پارامترهای مفروض در فصل ۵، به کار می‌بریم. پارامترهای کنترل TSMC را به شرح زیر انتخاب می‌کنیم:

$$p = \frac{11}{7} \quad \lambda^{-1} = \begin{bmatrix} \frac{1}{3} & 0 & 0 \\ 0 & \frac{1}{3} & 0 \\ 0 & 0 & \frac{1}{3} \end{bmatrix} \quad K = \begin{bmatrix} 5 & 0 & 0 \\ 0 & 5 & 0 \\ 0 & 0 & 5 \end{bmatrix}$$

مقدار $\hat{f}$ نیز بصورت $\hat{f}(q, \dot{q}, \ddot{q}) = (I_{3 \times 3} - \hat{M}(q))\ddot{q} - \hat{C}(q, \dot{q})\dot{q} - \hat{G}(q) - \hat{F}(\dot{q})$ می‌باشد که در آن

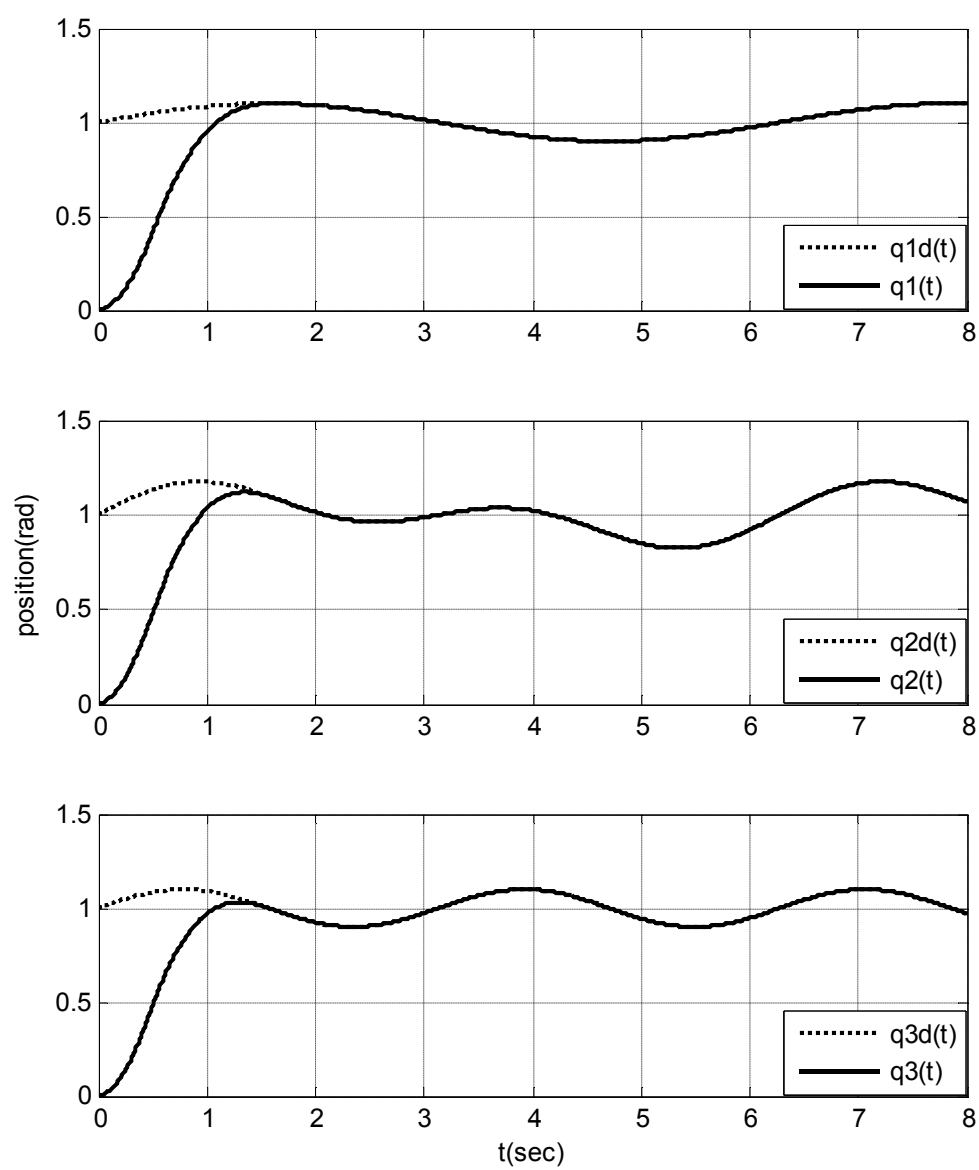
$$\hat{M} = 0.95M \quad \hat{C} = 0.8C \quad \hat{G} = 0.9G \quad \hat{F} = 0.95F$$

انتخاب می‌کنیم.

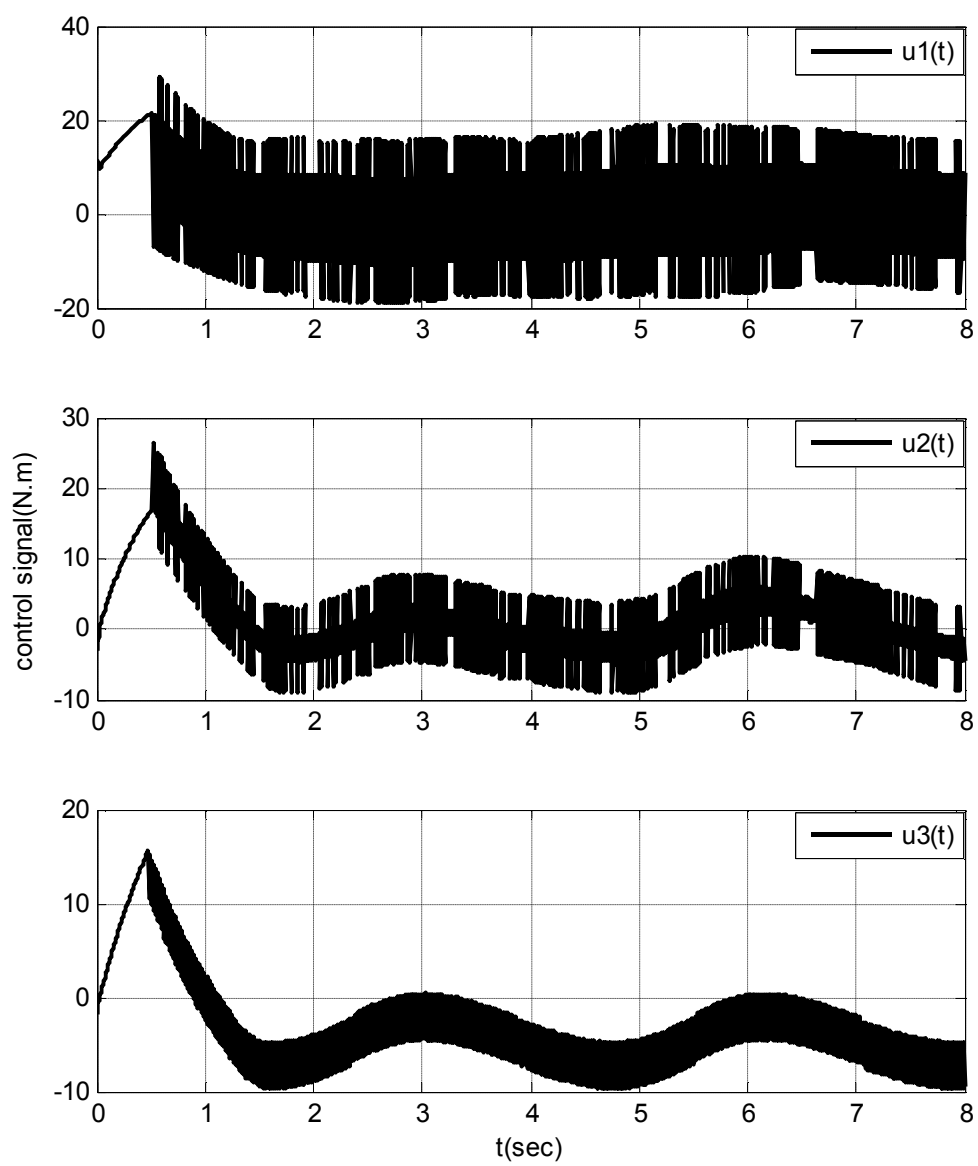
بردار موقعیت و گشتاور مفاصل حاصل از TSMC به ترتیب در شکل‌های ۶-۱ و ۶-۲ نشان داده شده است. همانطور که در شکل ۶-۱ نشان داده شده، به ازای کنترل کننده TSMC، ردیابی دقیق تر و سریعتر از کنترل کننده SMC انجام می‌پذیرد. همچنین نکته قابل توجه در رابطه با قانون کنترل (۶-۲۴)، وابستگی آن به شتاب مفصلی می‌باشد. برای رفع این مشکل، قانون کنترل را می‌توانستیم به شکل زیر پیشنهاد دهیم:

$$u(t) = \hat{C}(q, \dot{q})\dot{q} + \hat{G}(q) + \hat{F}(\dot{q}) + \hat{M}(q) \left(\ddot{q}_d - \frac{1}{p} \lambda^{-1} \text{diag}\{\dot{e}^{1-p}\} \dot{e} - K(q, t) \text{sgn}(s) \right) \quad (۶-۳۱)$$

اما با توجه به اینکه این تحقیق در محیط شبیه سازی انجام می‌شود، به رابطه (۶-۲۴) بسنده می‌کنیم.



شکل ۶-۱- مسیر مطلوب (نقطه چین) و زاویه مفصل (خط) حاصل از TSMC



شکل ۶-۲- گشتاور ورودی حاصل از TSMC

۵-۶- طراحی کنترل کننده عصبی- مد لغزش غیرخطی NTSMC

در فصل قبل با استفاده از شبکه عصبی موازی با سیستم فازی موفق شدیم یک کنترل کننده SMC مستقل از مدل و بدون نوسان با وجود اغتشاش خارجی طراحی نماییم؛ اما همانطور که قبلاً اشاره شد، کنترل کننده مورد نظر دارای دو عیب اساسی است:

۱- کنترل کننده NFSMC از لحاظ محاسباتی حجیم می باشد.

۲- به دلیل وجود سیستم فازی، اثبات پایداری آن مشکل است.

در این بخش، برای کاهش حجم محاسبات از سیستم فازی صرفنظر شده است. و بجای آن $Ksat(s/L)$ را به کار می‌بریم. همچنین برای حفظ سرعت پاسخ از سطح لغزش غیرخطی استفاده می‌گردد. برای این منظور از رابطه (۶-۲۴) استفاده کرده و تابع f را با یک شبکه MLP دو لایه تقریب می‌زنیم. بنابراین قانون کنترل را به فرم زیر پیشنهاد می‌دهیم [۳۴ و ۹۳]:

$$u = -\hat{f} + \ddot{q}_d - \frac{1}{p} \lambda \text{diag}\{\dot{e}^{1-p}\} \dot{e} - Ksat(s) \quad (۳۲-۶)$$

که در آن $K = \text{diag}\{k_{11}, k_{22}, \dots, k_{nn}\}$ و $\hat{f} = \hat{W}^T \sigma(\hat{V}^T x)$

$$sat(s) = \begin{cases} s/L & \|s\| < L \\ \text{sgn}(s/L) & \|s\| > L \end{cases} \quad (۳۳-۶)$$

تعریف می‌گردد. توجه شود که

$$f(x) = W^T \sigma(V^T x) + \varepsilon(x) \quad (۳۴-۶)$$

می‌باشد، که در آن $x = [q, \dot{q}, \ddot{q}]^T$ و $\varepsilon(x)$ به ترتیب، بردار ورودی و خطای تقریب شبکه عصبی است. خطاهای ناشی از ضرایب وزنی را نیز بصورت $\tilde{W} = W - \hat{W}$ و $\tilde{V} = V - \hat{V}$ تعریف می

نماییم. همچنین، تابع فعالیت $\sigma(x)$ رابطه (۳۴-۶) را بصورت بسط سری تیلور آن مانند زیر در نظر می گیریم:

$$\sigma(\hat{V}^T x) = \sigma(\hat{V}^T x) + \sigma'(\hat{V}^T x) \tilde{V}^T x + O(\hat{V}^T x)^2 \quad (۳۵-۶)$$

که در آن، $\sigma(z) \equiv \frac{d\sigma(z)}{dz}_{z=\hat{z}}$ ماتریس ژاکوبین و $O(z)^2$ نشان دهنده عبارت درجه دوم این تقریب می باشد. اکنون طراحی کنترل کننده را بصورت لم زیر بیان می کنیم:

لم ۳-۶: معادلات دینامیکی سیستم (۱-۵) و سطح لغزش (۲۰-۶) را در نظر بگیرید. برای سرعت کراندار و پیوسته، کنترل کننده (۳۲-۶) کراندار سیستم حلقه بسته را می تواند تضمین کند. و قوانین تطبیقی عصبی به وسیله روابط،

$$\dot{\hat{W}} = \alpha \sigma(\hat{V}^T x) s_1^T - \alpha \sigma'(\hat{V}^T x) \hat{V}^T x s_1^T \quad (۳۶-۶)$$

$$\dot{\hat{V}} = \beta x s_1^T \hat{W}^T \sigma'(\hat{V}^T x) \quad (۳۷-۶)$$

داده شده اند. که در روابط فوق $s_1 = p\lambda^{-1} \text{diag}\{\dot{e}^{p-1}\} s$ ، α و β ثابت های مثبت می باشند. علاوه بر این، حالت های سیستم در رنج وسیعی به سطح لغزش غیرخطی همگرا خواهند شد.

اثبات: بدون استفاده از خواص بازوی ربات، تابع لیاپانوف را بشکل زیر در نظر می گیریم:

$$V_L = \frac{1}{2} s^T s + \frac{1}{2\alpha} \text{tr}(\tilde{W}^T \tilde{W}) + \frac{1}{2\beta} \text{tr}(\tilde{V}^T \tilde{V}) \quad (۳۸-۶)$$

محاسبه مشتق (۳۸-۶) بر روی سطح لغزش (۲۰-۶) نتیجه می دهد

$$\dot{V}_L = s^T \dot{s} + \frac{1}{\alpha} \text{tr}(\tilde{W}^T \dot{\tilde{W}}) + \frac{1}{\beta} \text{tr}(\tilde{V}^T \dot{\tilde{V}}) \quad (۳۹-۶)$$

از روابط (۲۱-۶)، (۲۲-۶) و (۳۲-۶) داریم:

$$\begin{aligned}\dot{V}_L = & ps^T \lambda^{-1} \text{diag}\{\dot{e}^{p-1}\} \left(W^T \sigma(V^T x) + \varepsilon(x) - \hat{W}^T \sigma(\hat{V}^T x) - u_d - Ksat(s) \right) \\ & + \frac{1}{\alpha} \text{tr}(\tilde{W}^T \dot{\tilde{W}}) + \frac{1}{\beta} \text{tr}(\tilde{V}^T \dot{\tilde{V}})\end{aligned}\quad (۴۰-۶)$$

با جایگذاری (۳۵-۶) در عبارت فوق

$$\begin{aligned}& W^T \sigma(V^T x) + \varepsilon(x) - \hat{W}^T \sigma(\hat{V}^T x) - u_d - Ksat(s) \\ & = W^T \sigma(V^T x) + \varepsilon(x) - \tilde{W}^T \sigma(V^T x) + \tilde{W}^T \sigma(V^T x) - \hat{W}^T \sigma(\hat{V}^T x) - u_d - Ksat(s) \\ & = \tilde{W}^T \sigma(V^T x) + \tilde{W}^T (\sigma(V^T x) - \sigma(\hat{V}^T x)) - u_d + \varepsilon(x) - Ksat(s) \\ & = \tilde{W}^T \left(\sigma(\hat{V}^T x) + \sigma'(\hat{V}^T x) \tilde{V}^T x + O(\hat{V}^T x)^2 \right) + \tilde{W}^T \left(\sigma'(\hat{V}^T x) \tilde{V}^T x + O(\hat{V}^T x)^2 \right) \\ & \quad - u_d + \varepsilon(x) - Ksat(s) \\ & = \tilde{W}^T \sigma(\hat{V}^T x) - \tilde{W}^T \sigma'(\hat{V}^T x) \hat{V}^T x + \hat{W}^T \sigma'(\hat{V}^T x) \tilde{V}^T x \\ & \quad + \tilde{W}^T \sigma'(\hat{V}^T x) \tilde{V}^T x + WO(\hat{V}^T x)^2 - u_d + \varepsilon(x) - Ksat(s)\end{aligned}$$

به علت آنکه

$$\begin{aligned}s_1^T \tilde{W}^T \sigma(\hat{V}^T x) &= \text{tr}(\tilde{W}^T \sigma(\hat{V}^T x) s_1^T) \\ s_1^T \tilde{W}^T \sigma'(\hat{V}^T x) \hat{V}^T x &= \text{tr}(\tilde{W}^T \sigma'(\hat{V}^T x) \hat{V}^T x s_1^T) \\ s_1^T \tilde{W}^T \sigma'(\hat{V}^T x) \tilde{V}^T x &= \text{tr}(\tilde{V}^T x s_1^T \tilde{W}^T \sigma'(\hat{V}^T x))\end{aligned}$$

در نتیجه می‌توان نوشت:

$$\begin{aligned}\dot{V} = & \text{tr} \left(\tilde{W}^T \left(\alpha^{-1} \dot{\tilde{W}} + \sigma(\hat{V}^T x) s_1^T - \sigma'(\hat{V}^T x) \hat{V}^T x s_1^T \right) \right) \\ & + \text{tr} \left(\tilde{V}^T \left(\beta^{-1} \dot{\tilde{V}} + x s_1^T \tilde{W}^T \sigma'(\hat{V}^T x) \right) \right) \\ & + s_1^T \left(\tilde{W}^T \sigma'(\hat{V}^T x) \tilde{V}^T x + WO(\hat{V}^T x)^2 + \varepsilon(x) - u_d - Ksat(s) \right)\end{aligned}\quad (۴۱-۶)$$

اگر روابط تطبیقی (۳۶-۶) و (۳۷-۶) را در عبارت فوق جایگذاری کنیم:

$$\dot{V} = s_1^T \left(\tilde{W}^T \sigma'(\hat{V}^T x) V^T x + W O(\hat{V}^T x)^2 + \varepsilon(x) - u_d - K \text{sat}(s) \right) \quad (42-6)$$

با فرض اینکه

$$\|\tilde{W}^T \sigma'(\hat{V}^T x) V^T x\| < \varepsilon_1, \quad \|W O(\hat{V}^T x)^2\| < \varepsilon_2, \quad \|\varepsilon(x)\| < \varepsilon_{NN}$$

و با توجه به اینکه $p-1 = \frac{p_1 - p_2}{p_2}$ ($p = \frac{p_1}{p_2}, p_1 > p_2$) مقداری مثبت و زوج است، لذا

$$\dot{e}_i^{p-1} > 0, \quad i = 1, 2, \dots, n \text{ در نتیجه:}$$

۱- در ناحیه $\|s\| > L$

$$\dot{V} \leq p \lambda^{-1} \dot{e}^{p-1} \left(\|s\| (\|\varepsilon_1 + \varepsilon_2\| + \|\varepsilon_{NN}\| + U_D) - \sum_{i=1}^n k_{ii} \|s_i\| \right) \quad (43-6)$$

بنابراین به ازای $i = 1, 2, 3$

$$k_{ii} > \|s\| (\|\varepsilon_1 + \varepsilon_2\| + \|\varepsilon_{NN}\| + U_D) \quad (44-6)$$

$\dot{V} \leq 0$ را تضمین می کند.

۲- از طرف دیگر در ناحیه $\|s\| < L$

$$\dot{V} \leq p \lambda^{-1} \dot{e}^{p-1} \left(\|s\| (\|\varepsilon_1 + \varepsilon_2\| + \|\varepsilon_{NN}\| + U_D) - \sum_{i=1}^n k_{ii} \|s_i\|^2 / L \right) \quad (45-6)$$

برای

$$L (\|\varepsilon_1 + \varepsilon_2\| + \|\varepsilon_{NN}\| + U_D) / \min_i (k_{ii}) < \|s_i\| \leq L \quad (46-6)$$

$$\square. \dot{V} \leq 0$$

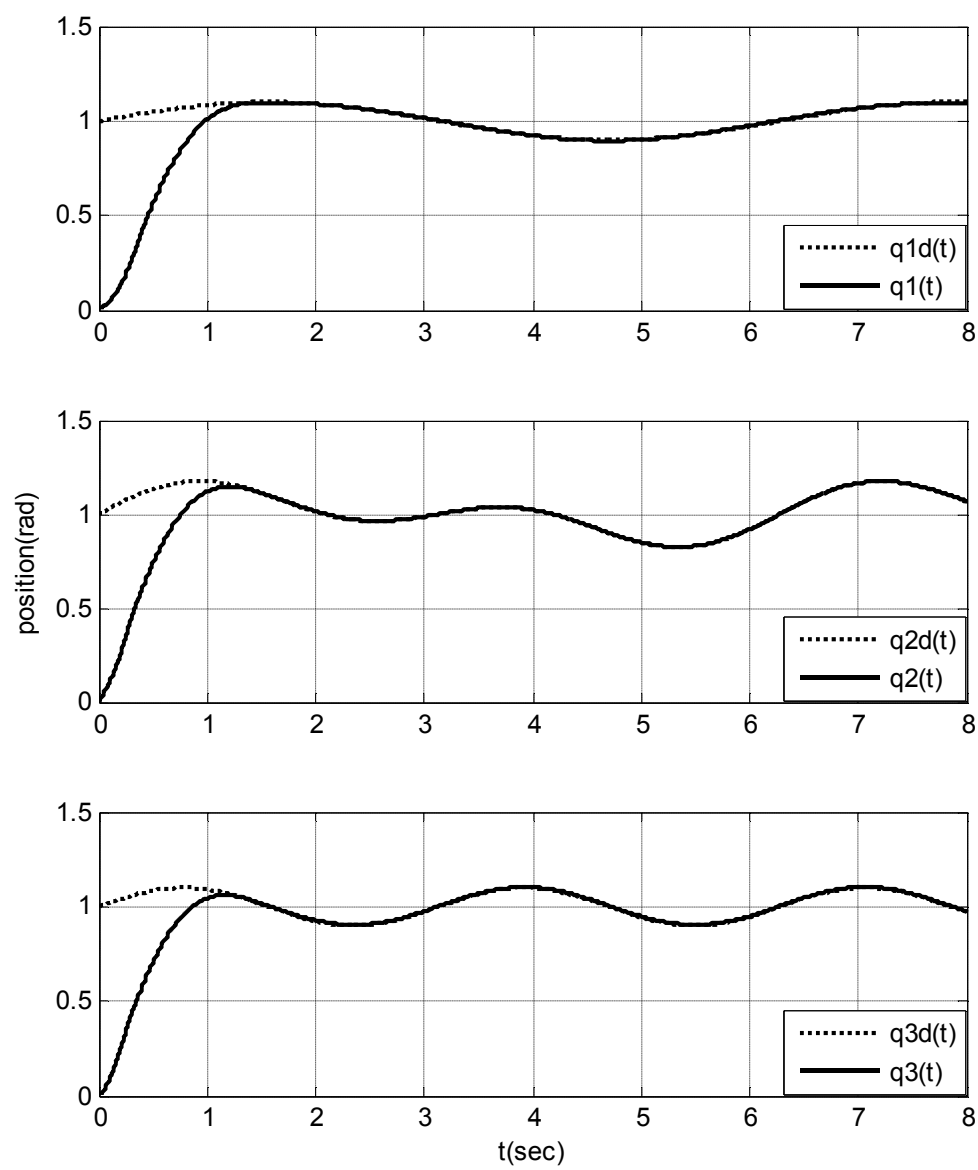
۶-۶- نتایج شبیه‌سازی به ازای کنترل کننده NTSMC

مقادیر پارامترهای کنترل کننده NTSMC را

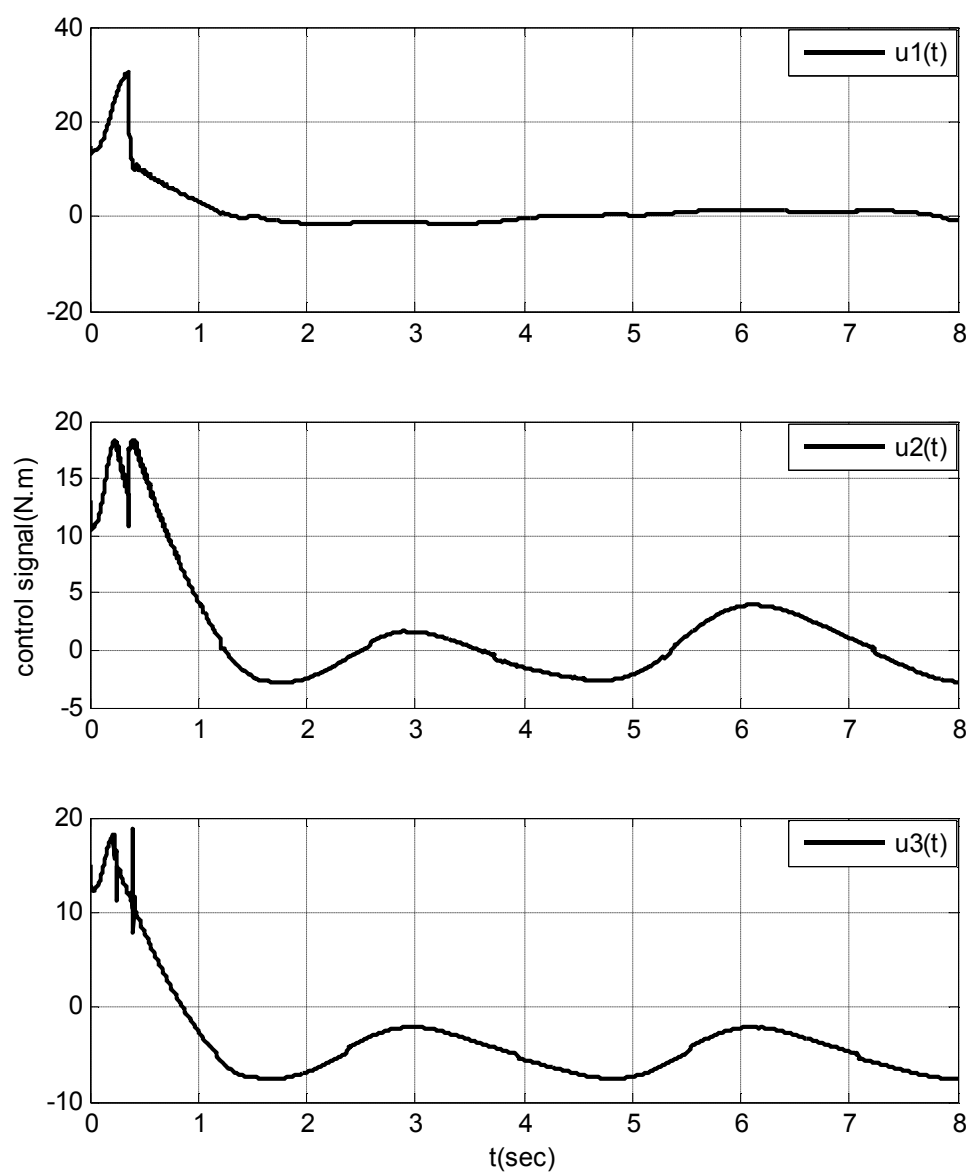
$$L = 0.02, \quad K = \begin{bmatrix} 14 & 0 & 0 \\ 0 & 12 & 0 \\ 0 & 0 & 14 \end{bmatrix}, \quad \lambda^{-1} = \begin{bmatrix} 1/3 & 0 & 0 \\ 0 & 1/3 & 0 \\ 0 & 0 & 1/3 \end{bmatrix}$$

انتخاب می‌کنیم. توجه شود که مقدار L را تا آستانه نوسانی شدن سیگنال کنترلی کوچک در نظر می‌گیریم. برای شبکه عصبی دو لایه، تعداد نرون‌های لایه میانی ۱۰ و لایه خروجی را ۳ نرون انتخاب می‌کنیم. همچنین مقادیر پارامترهای $\alpha = 0.08$ ، $\beta = 0.08$ با زمان نمونه‌برداری $T = 2ms$ به ازای بردار ورودی $x(t) = [q \quad \dot{q} \quad q_d \quad \dot{q}_d]^T$ می‌باشد (بردار ورودی را می‌توانستیم $x(t) = [q \quad \dot{q} \quad \ddot{q}]^T$ نیز انتخاب کنیم؛ اما برای حفظ استقلال کنترل کننده از شتاب مفصلی، بردار ورودی قبلی معقول تر می‌باشد).

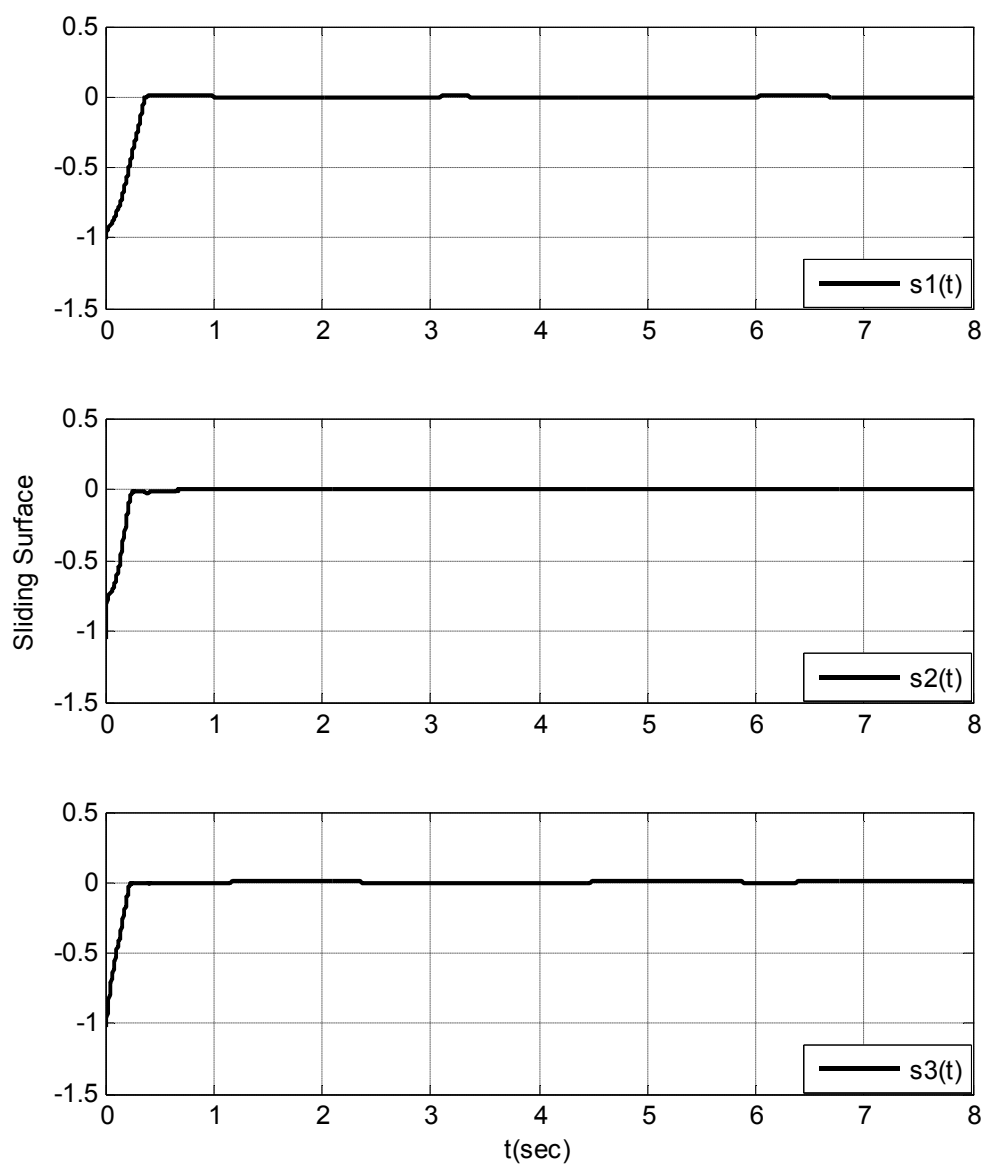
موقعیت زاویه ای مفاصل و گشتاور اعمالی به ترتیب در شکل‌های ۶-۳، ۶-۴ نشان داده شده است. شکل ۶-۳، با توجه به عدم استفاده از مدل سیستم و وجود اغتشاش خارجی، توانایی کنترل کننده پیشنهادی در ردگیری مسیر ورودی مطلوب را نشان می‌دهد. همچنین شکل ۶-۵ بردار سطح لغزش s را نشان می‌دهد.



شکل ۳-۶- مسیر مطلوب (نقطه چین) و زاویه مفصل (خط) حاصل از NTSMC



شکل ۴-۶- گشتاور ورودی حاصل از NTSMC



شکل ۵-۶- سطح لغزش غیرخطی NTSMC

همانطور که در شکل‌های ۳-۶، ۴-۶ و ۵-۶ دیده می‌شود، کنترل کننده NTSMC در مقایسه با کنترل کننده‌های NFDSMC و NFSCM، از دقت بهتری در ردیابی ورودی مرجع برخوردار است؛ و بدلیل حذف سیستم فازی، حجم محاسبات کاهش یافته و پایداری سیستم حلقه بسته به اثبات می‌رسد.

در این فصل نیز مانند فصل قبل، برای شبیه‌سازی از نرم افزار SIMULINK استفاده شده که بخش مهم بلوک های مربوطه در ضمیمه (ج) آورده شده است.

فصل هفتم: نتیجه‌گیری و پیشنهادات

در این پایان نامه، اهدافی از قبیل: طراحی کنترل‌کننده مستقل از مدل ریاضی، مقاوم در برابر اغتشاش خارجی، حذف نوسانات ناخواسته و تنظیم آنالین پارامترهای کنترلی، به ازای کنترل‌کننده عصبی-فازی-مد لغزش بر روی بازوی ربات اسکارا را مورد مطالعه قرار دادیم. بدین منظور سه کنترل‌کننده مستقل از مدل طراحی نمودیم.

برای کنترل‌کننده اول (NFSMC)، از ساختار شبکه عصبی موازی با سیستم فازی استفاده شد. در این راستا، شبکه‌های عصبی پیشخور MLP و RBF، وظیفه تقریب قانون کنترل، و سیستم فازی، تقریب جمله کلیدزنی را بر عهده داشتند. تنظیم ضرایب شبکه عصبی بصورت آنالین و با استفاده از الگوریتم پس انتشار خطا و بر پایه خروجی سیستم فازی (برای شبکه MLP) یا سطح لغزش (برای شبکه RBF) صورت پذیرفت. نتایج شبیه‌سازی عملکرد مطلوب کنترل‌کننده NFSMC را نشان می‌دهد. برای کارهای آتی: مطالعه پایداری تکنیک مذکور، استفاده از شبکه‌های عصبی برای هر دو قسمت تقریب زنی و تنظیم بهره، بر روی ربات اسکارا، تقریب قانون کنترل معادل با استفاده از شبکه RNN و مقایسه عملکرد آن با شبکه‌های MLP و RBF را می‌توان پیشنهاد داد.

برای کنترل‌کننده دوم (NFDSMC)، کنترل‌کننده متناظر کنترل‌کننده اول را در حوزه گسسته در زمان مطرح نمودیم. همچنین تاثیر زمان نمونه‌برداری بر همگرایی سیستم حلقه بسته به ازای کنترل‌کننده‌های DSMC و NFDSMC، بررسی شد. نتایج شبیه‌سازی بیانگر توانایی کنترل‌کننده مستقل از مدل پیشنهادی در این حوزه می‌باشد. برای کارهای آینده در این زمینه، می‌توان قانون

کنترلی جدیدی پیشنهاد داد که همگرایی سیستم حلقه بسته به سطح لغزش گسسته در زمان را بهبود بخشد.

برای کنترل کننده سوم (NTSMC)، طراحی کنترل کننده اول را بدون استفاده از سیستم فازی و با تعریف سطح لغزش غیرخطی انجام دادیم. علاوه براین، قانون کنترل معادل با استفاده شبکه عصبی تقریب زده شد؛ اما با این تفاوت که ضرایب شبکه عصبی نه از الگوریتم پس انتشار خطا، بلکه از پایداری تابع لیاپانوف تنظیم شدند. همچنین به دلیل حذف سیستم فازی، بار سنگین محاسباتی کنترل کننده تا حدودی کاهش یافته و خطای ردیابی کمتر شده است. نتایج شبیه سازی ارائه شده نیز بهبودی‌های حاصل شده را نشان داد. برای کارهای آینده می‌توان استفاده از تابع انرژی غیرخطی برای شبکه عصبی، به منظور تسریع آموزش را مورد مطالعه قرار داد.

ضمیمه الف

برنامه شناسایی با استفاده از شبکه عصبی دو لایه (M-file)

```

u=rands(1, 351)*2; % u(k).....[-2, 2]
yp(1)=0;
yp(2)=0;
for k=2:351
yp(k+1)=yp(k)*(yp(k-1)+2)*(yp(k)+2.5)/(8.5+yp(k)^2+yp(k-1)^2)+u(k);
out(k-1)=(yp(k+1)-u(k))/10; %normalizing output training data
in(k-1)=yp(k)/10; %normalizing input training data
end;
plantin=[in(1:349); in(2:350)]; plantout=out(1:349);
net = newff(minmax(plantin),[10 1],{'tansig' 'purelin'},'trainlm','learnngdm','mse')
net = init(net);
net.iw{1,1}
net.b{1}
net.trainParam.epochs = 500;
net.trainParam.goal = 0.0001;
net=train(net,plantin,plantout);
trainedout=sim(net,plantin);
figure
plot(plantout,'k'); %Actual plant output
hold on;
plot(trainedout,'k','LineWidth',2); %Trained neural network output
axis([0, 350, -0.2 ,0.7]);
xlabel('Time Step');
ylabel('Plant (solid) NN Output (dotted)');
%.....test.....
yp(1)=0.0; yp(2)=0.0;out1(1)=0; out1(2)=0;

```

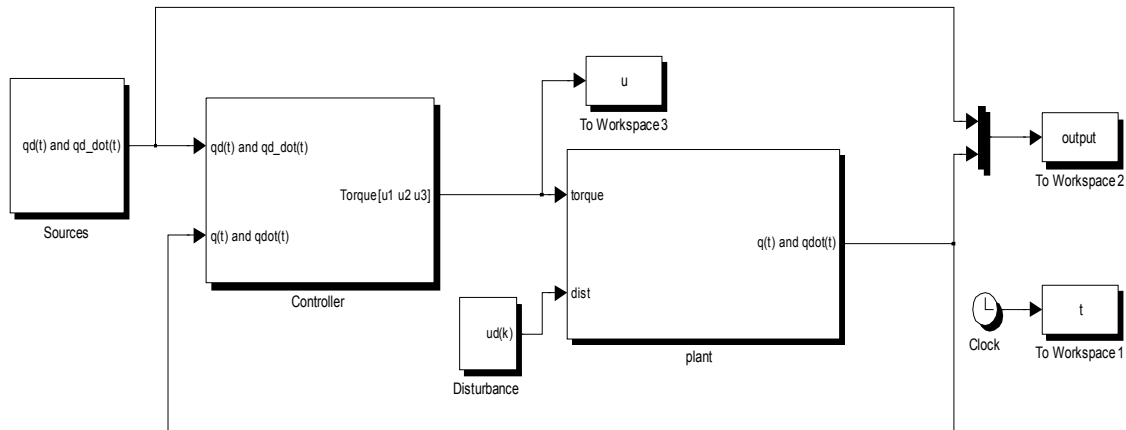
```

for k=2:600
if (k<=250)u(k)=2.0*cos(2*pi*k*0.01);
end
if k>=250 & k<=450
    u(k)=1.2*sin(2*pi*k*0.05);
end;
if k>=450
    u(k)=1.2*sin(4*pi*k*0.05);
end
yp(k+1)=yp(k)*(yp(k-1)+2)*(yp(k)+2.5)/(8.5+yp(k)^2+yp(k-1)^2)+u(k);
out1(k)=yp(k)/10;
out1(k-1)=yp(k-1)/10;
nnout(k+1)=10*sim(net,[out1(k);out1(k-1)])+u(k);
end
figure
plot(yp,'k');
hold on;
plot(nnout, 'k','LineWidth',2);
axis([0, 600, -4.0, 10.0]);
xlabel('Time Step');
ylabel('Plant (solid) NN Output (dotted)');

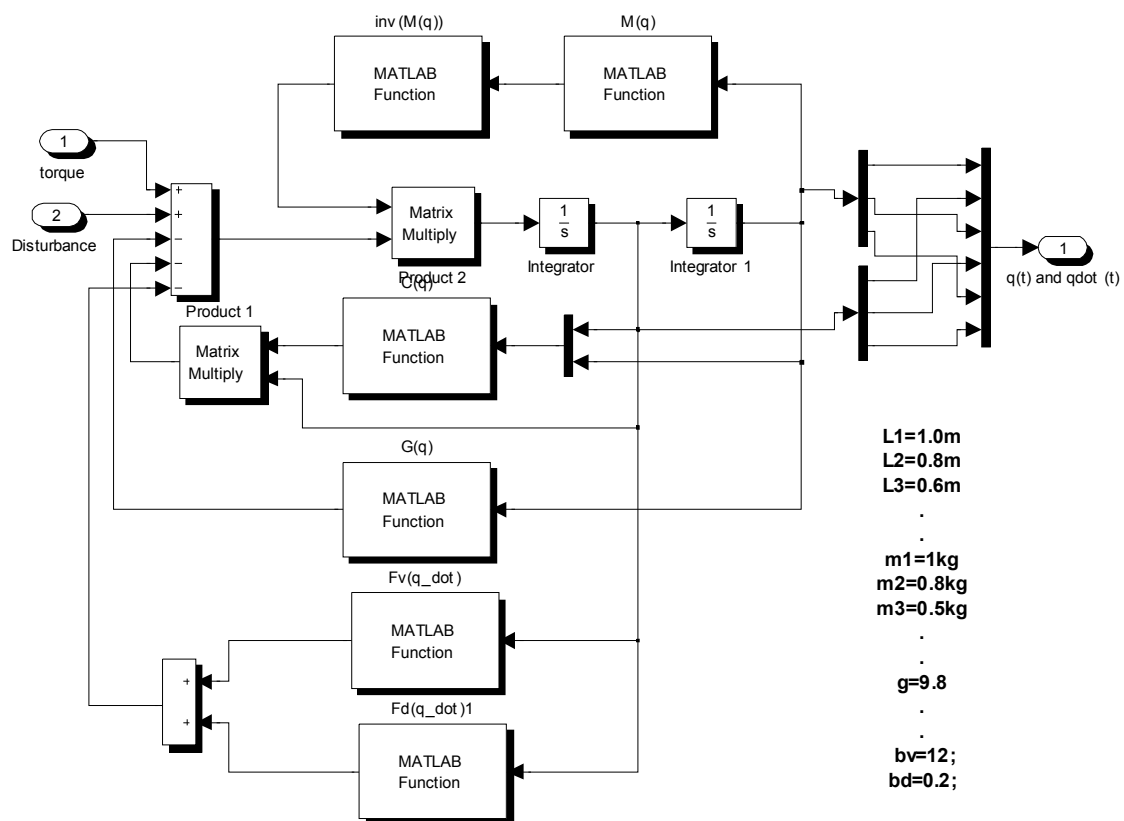
```

ضمیمه ب

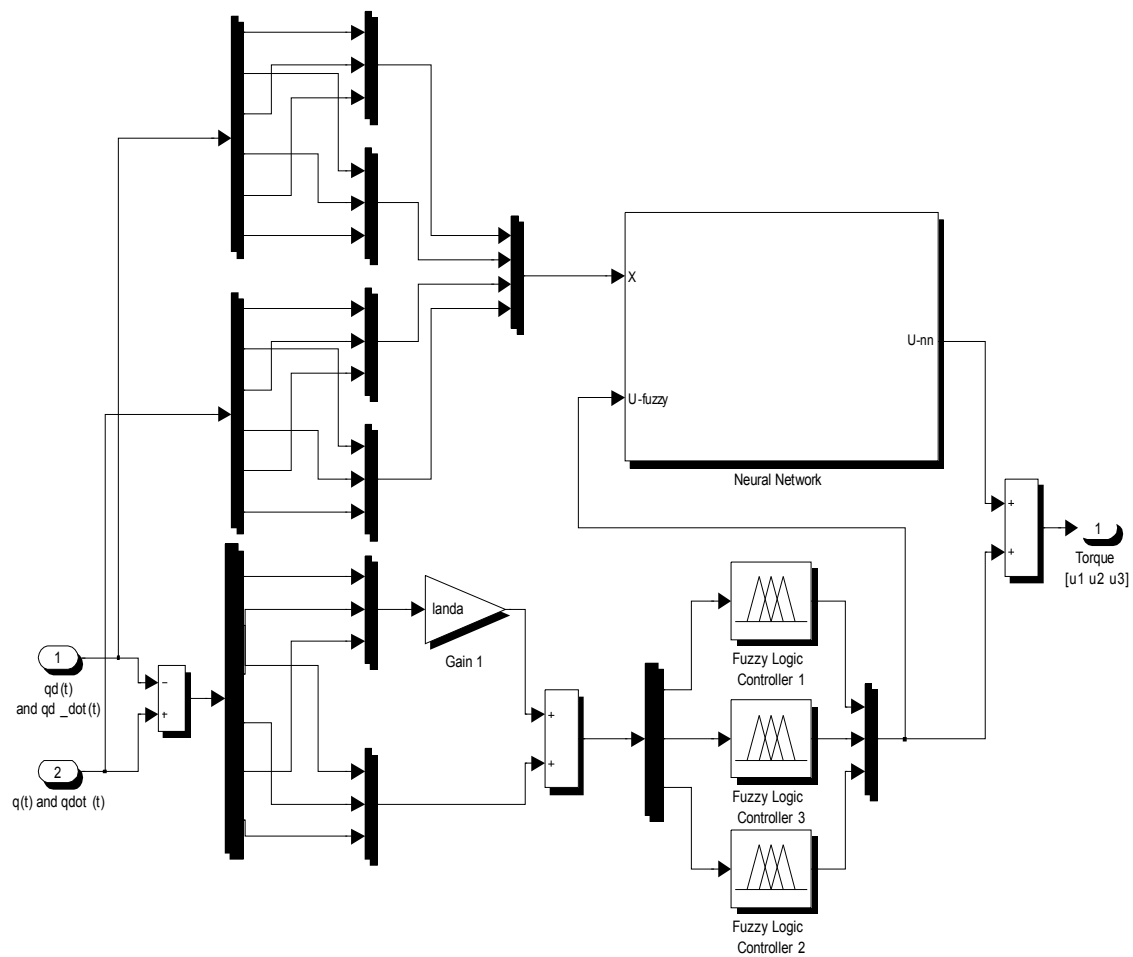
بخشی از بلوک‌های طراحی شده در SIMULINK



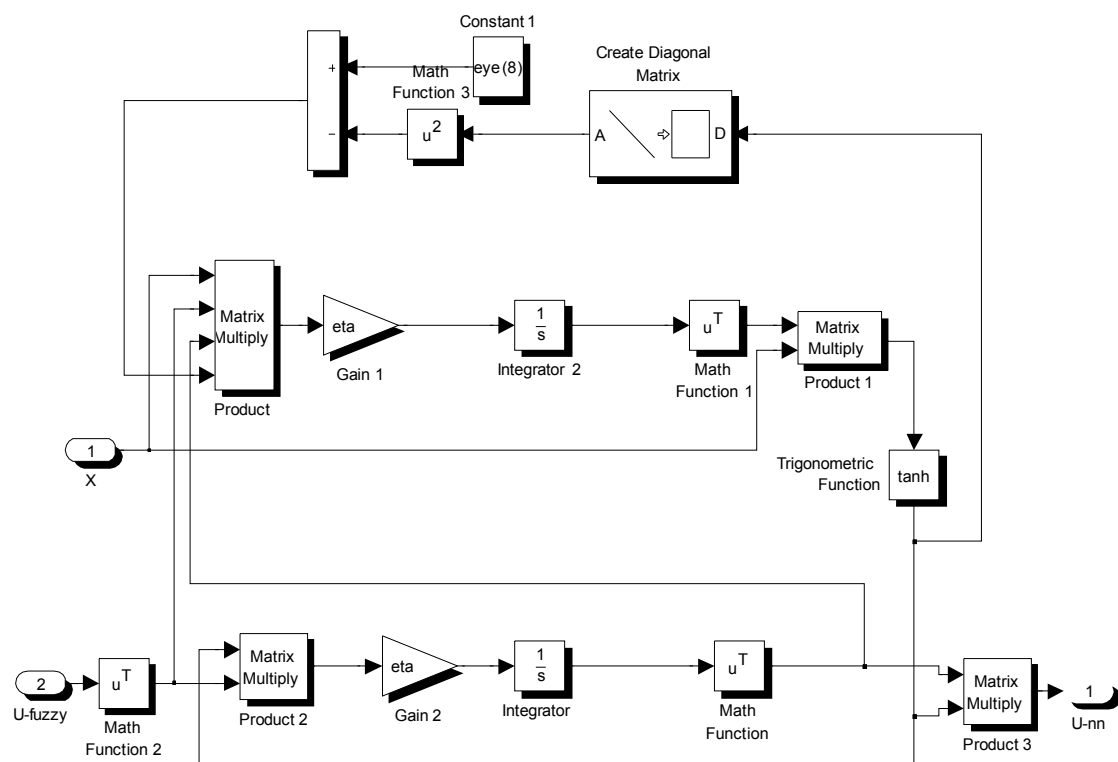
شکل ب-۱- سیستم حلقه بسته



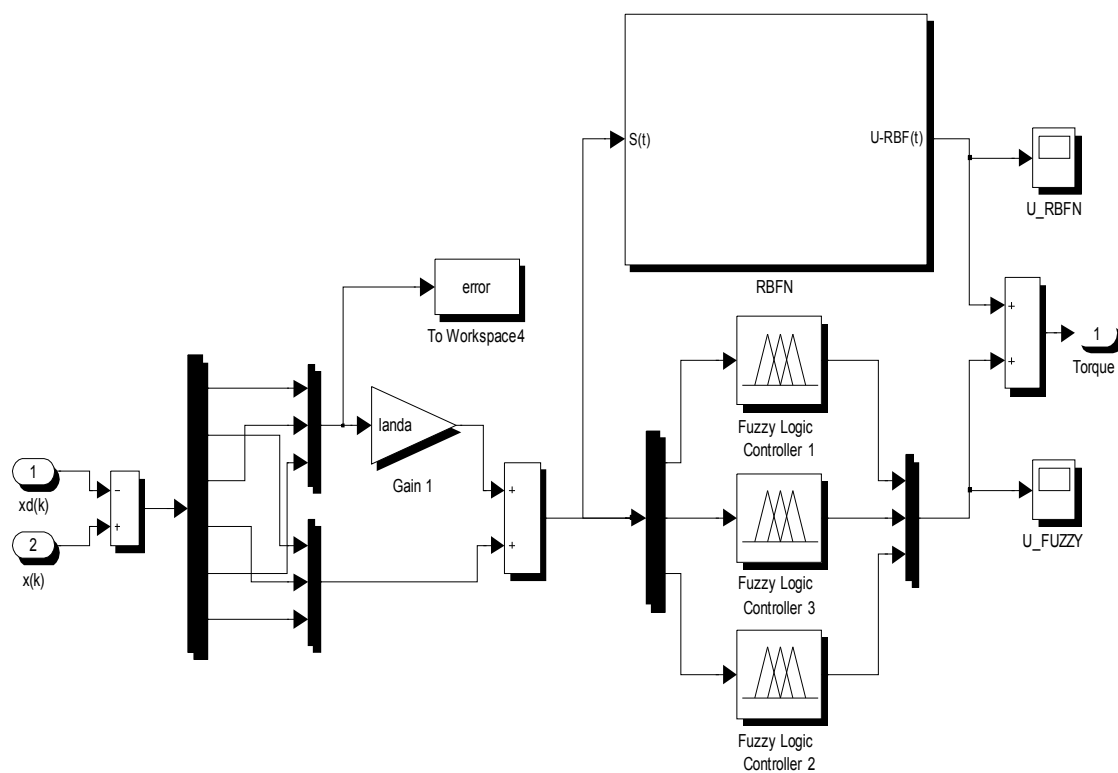
شکل ب-۲- مدل بازوی ربات سه رابط SCARA



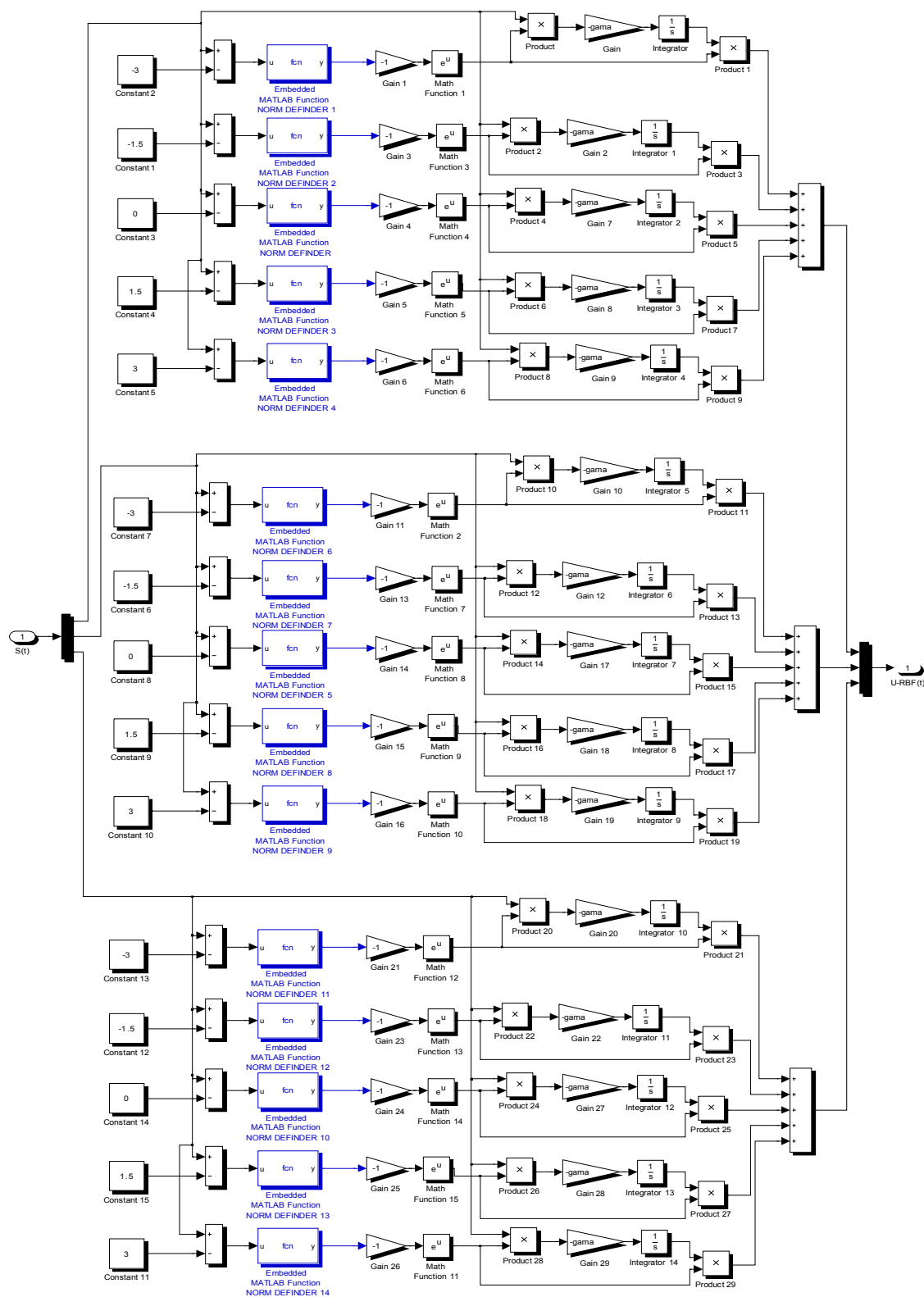
شکل ب-۳- نمایی از عملکرد موازی سیستم فازی و عصبی (NFSMC-MLP)



شکل ب-۴- بلوک‌های مربوط به شبکه عصبی MLP و آموزش آنلین



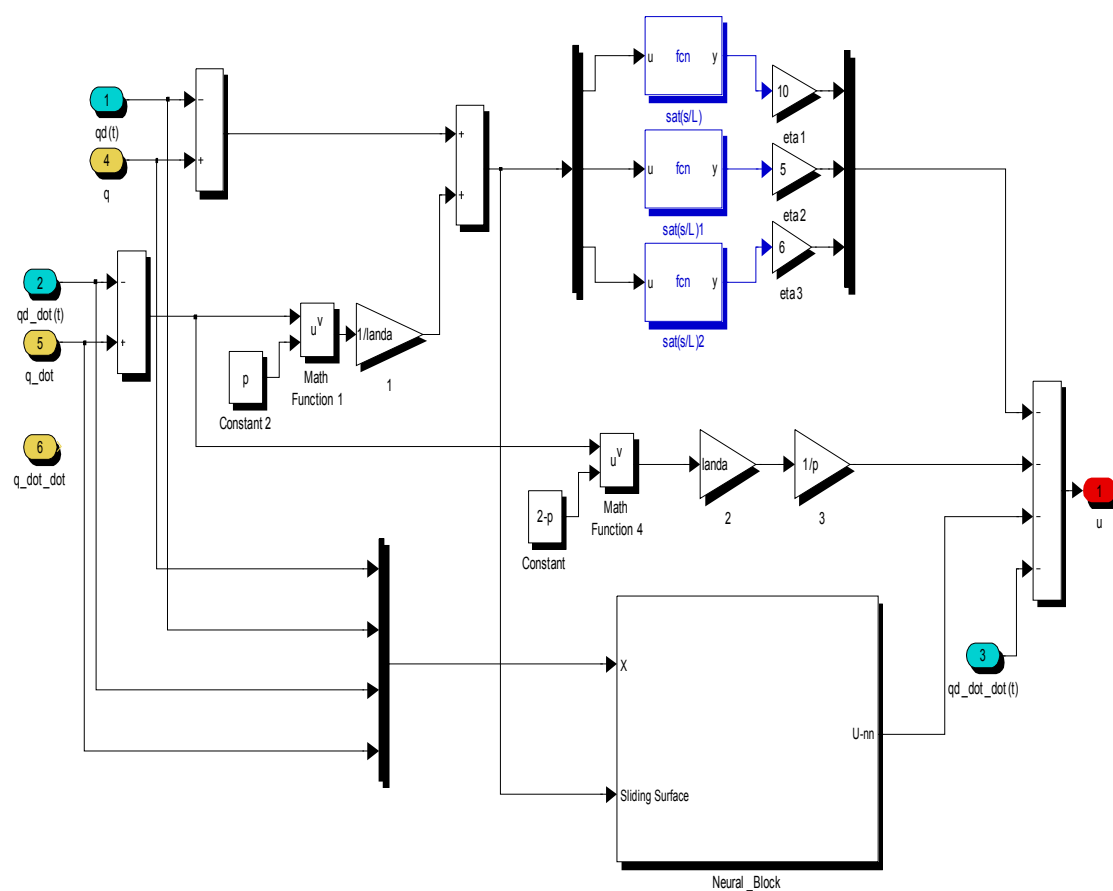
شکل ب-۵- نمایی از عملکرد موازی سیستم فازی و عصبی (NFSMC-RBF)



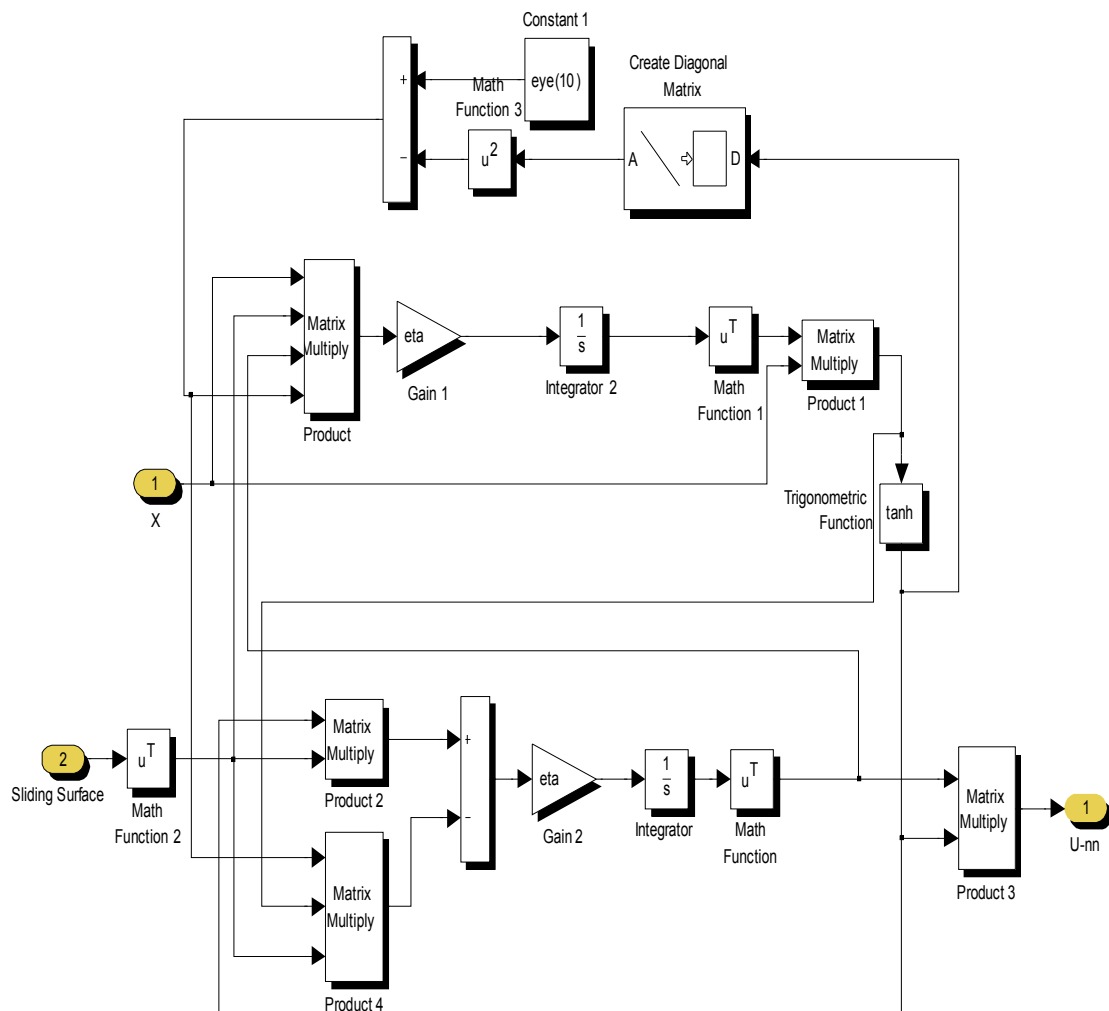
شکل ب-۶- بلوک‌های مربوط به شبکه عصبی RBF و آموزش آنالین

ضمیمه ج

بخشی از بلوک‌های طراحی شده در SIMULINK



شکل ج-۱- بلوک‌های مربوط به کنترل کننده NFTSMC



شکل ج-۲- بلوک‌های مربوط به شبکه عصبی MLP و آموزش آنالین

- [1] S. V. Emelyanov, "Method of designing complex control algorithm using an error and its first time derivative only", *Automation and Remote Control*, 18(10), 1957.
- [2] S. V. Emelyanov, I. A. Burovoi, "Mathematical models of process in technology and development of variable structure control system", *Metallurgy*, 18(07), 1964.
- [3] V. I. Utkin, "Variable structure system with sliding mode" *IEEE Trans. On Automatic Control*, 22(2), 212-221, 1977.
- [4] U. Itkis, *Control Systems of Variable Structure*, Wiley, New York, 1976.
- [5] F. Boudjema, J. L. Abatut. Sliding mode: A New Way to Control Series Resonant Converters. In *Proc of IEEE Conference Industrial Electronics Society*, IEEE Press, Pacific Grove, Florida, USA, vol. 2, pp. 938-943, 1990.
- [6] M. E. Aggoune, F. Boudjema, A. Bensenousi, A. Hellal, M. R. Elmesai, S. V. Vadri, "Design of Variable Structure Voltage Regulator Using Pole Assignment Technique" *IEEE Trans. On Automation Control*, Vol. 39, No. 10, pp. 2106-2110, 1994.
- [7] D. Boukhetala, F. Boudjema, T. Madani, M. S. Boucherit, N. K. Msiridi "A New Decentralized Variable Structure Control for Robot Manipulators" *International Journal of Robotics and Automation*, vol. 18, no. 1, pp. 28-40, 2003.
- [8] A. Bartoszewicz, O. Kaynak, V. I. Utkin, "special section on sliding mode control on industrial applications" *IEEE Trans. On Industrial Electronics*, Vol. 55, No. 11, 3806-4074, 2008.
- [9] A. Bartoszewicz, R. J. Patton, "Sliding mode control: Special issue" *International Journal of Adaptive Control and Signal Processing*, 21, 635-822, 2007.
- [10] S. T. Venkataraman and S. Gulati, "Control of nonlinear systems using terminal sliding modes" *J. Dyn. Syst. Meas. Control*, vol. 115, no. 3, pp. 554-560, Sep. 1993.
- [11] D. Zhao S. Li, F. Gao and Q. Zhu, "Robust adaptive terminal sliding mode based synchronised position control for multiple motion axes system" *IET Control Theory and Appl*, Vol. 3, No. 1, pp. 136-150, 2009.
- [12] L. Hui and J. Li, "Terminal Sliding Mode Control for Spacecraft Formation Flying" *IEEE Trans. On Aerospace And Electronic Systems*, Vol. 45, No. 3, July, 2009.
- [13] D. J. Kim, K. H. Park and Z. Bien, "Hierarchical Longitudinal Controller for Rear-End Collision Avoidance" *IEEE Trans. On Industrial Electronics*, Vol. 54, No. 2, April, 2007.
- [14] E. C. Chang, T. J. Liang, J. F. Chen and F. J. Chang, "Real-time implementation of grey fuzzy terminal sliding mode control for PWM DC-AC Convertors" *IET Power Electron*, Vol. 1, No. 2, pp. 235-244, 2008.
- [15] Y. Feng, J. Zheng, X. Yu and N. V. Truong, "Hybrid Terminal Sliding-Mode Observer Design Method for a Permanent-Magnet Synchronous Motor Control System" *IEEE Trans. On Industrial Electronics*, Vol. 56, No. 9, September, 2009.

- [16] B. Bandyopadhyay, Discrete-time Sliding Mode Control: A Multirate Output Feedback Approach, Springer-Verlag Berlin Heidelberg New York, 2006.
- [17] S. K. Spurgeon, "Hyperplane design techniques for discrete-time variable structure control system" *Int. J. of Control* 55(2), pp. 445-456, 1992
- [18] W. Gao, Y. Wang and A. Homaifa, "Discrete time variable structure control system" *IEEE Trans. On Industrial Electronics*, Vol. 42, pp. 117-122, 1995.
- [19] A. Bartoszewicz, "Discrete time quasi-sliding mode control strategies" *IEEE Trans. On Industrial Electronics*, Vol. 45(3), pp. 633-637, 1998.
- [20] K. Furuta, "Sliding mode control of a discrete system" *Systems and Control Letters*, 14, pp. 145-152, 1990.
- [21] W. C. Su, S. V. Drakunov, U. Ozguner, "An $O(T^2)$ boundary layer in sliding mode for sampled-data systems" *IEEE Trans. On Automatic Control*, Vol. 45(3), pp. 482-485, 2000.
- [22] G. Golo, C. Milosavljevi, "Robust discrete-time chattering free sliding mode control" *Systems and Control Letters*, 41, pp. 19-28, 2000.
- [23] A. Boubakir, F. Boudjema and S. Laboid "A Fuzzy Sliding Mode Controller Using Nonlinear Sliding Surface Applied to the Coupled Tanks System" *International Journal of Fuzzy Systems*, Vol. 10, No. 2, June, 2008.
- [24] M. M. Abdelhameed "Enhancement sliding mode controller by fuzzy logic with application to robotic manipulators" *Mechatronics*, Vol. 15, pp. 439-458, 2005.
- [25] N. Yagiz, Y. Hacioglu "Robust control of a spatial robot using fuzzy sliding modes" *Mathematical and Computer Modelling*, Vol. 49, pp. 114-127, 2009.
- [26] Y. Guo and P. Y. Woo, "An Adaptive Fuzzy Sliding Mode Controller for Robotic Manipulators" *IEEE Trans. On Systems, Man, Cybernetics*, Vol. 33, No. 2, March, 2003.
- [27] X. L. Ning, P. Tan, D. Y. Huang and F. L. Zhou "Application of adaptive fuzzy sliding mode control to a seismically excited highway bridge" *Structural Control and Health Monitoring*, 16:639-656, March, 2009.
- [28] F. Passold, "Applying RBF Neural Nets for Position Control of an Inter/SCARA Robot" *Int. J. of Computers. Communications & Control*, Vol. 4, No. 2, pp. 148-157, 2009.
- [29] F. Passold and M. R. Stemmer, "Feedback Error Learning Neural Network Applied to a SCARA Robot" *Fourth International Workshop on Robot Motion and Control*, pp. 197-202, 2004.
- [30] A. V. Topalov, O. Kaynak, and G. Aydin "Neuro-adaptive sliding-mode tracking control of robot manipulators" *International Journal of Adaptive Control and Signal Processing*, 21:674-691, May 2007.
- [31] Z. H. Jiang, Taiki Ishita, "A Neural Network Controller for Trajectory Control of Industrial Robot Manipulator" *Journal of Computers*, Vol. 3, No. 8, August, 2008.
- [32] M. Ertugrul and O. Kaynak, "Neuro-sliding Mode Control of Robotic Manipulators" *Mechatronics*, vol 10, no 1-2, pp 239-263, 2000.

- [33] M. Baric, I. Petrovic, N. Peric, "Neural network based sliding mode control of electronic throttle" *Engineering Applications of Artificial Intelligence*, 18, pp 951-961, 2005.
- [34] Y. Tang, F. Sun, Z. Sun, "Neural network control of flexible-link manipulators using sliding mode" *Neurocomputing* 70, pp 288-295, 2006.
- [35] L. C. Hung, H. Y. Chung, "Decoupled control using neural network based sliding mode controller for nonlinear systems" *Expert Systems with Applications* 32, pp 1168-1182, 2007.
- [36] T. Chatchanayuenyong, M. Parnichkun, "Neural network based time optimal sliding mode control for an autonomous underwater robot" *Mechatronics* 16, pp 471-478, 2006.
- [37] V. M. Hung, U. J. Na, "Adaptive Neural Fuzzy Control for Robot Manipulator Friction and Disturbance Compensator" *International Conference on Control, Automation and Systems*, Seoul, Korea, Oct, 2008.
- [38] A. Bagheri, J. J. Moghaddam, "Decoupled adaptive neuro-fuzzy (DANF) sliding mode control system for a Lorenz chaotic problem" *Expert Systems with Applications*, Vol. 36, pp. 6062-6068, 2009.
- [39] C. Elmas, O. Ustun, "A hybrid controller for the speed control of a permanent magnet synchronous motor derive" *Control Engineering Practice*, Vol. 16, pp. 260-270, 2008.
- [40] F. J. Lin, R. J. Wai, "Adaptive and fuzzy neural network sliding-mode controllers for motor-quick-return servomechanism" *Mechatronics*, Vol. 13, pp. 477-508, 2003.
- [41] A. Boubakir, F. Boudjema and S. Laboid, "A Neuro-fuzzy-sliding Mode Controller Using Nonlinear Sliding Surface to the Couple Tanks System" *International Journal of Automation and Computing*. pp. 72-80, February, 2009.
- [42] A. G. Ak and G. Cansever, "NN Approaches on Fuzzy Sliding Mode Controller Design for Robot Trajectory Tracking" *IEEE Multi-Conference on Systems and Control*, Saint Petersburg, Russia, July, 2009.
- [43] J.J.E. Slotine, and W. Li, *Applied Nonlinear Control*. Englewood Cliffs, NJ Prentice-Hall, 1991.
- [44] X. Yu, O. Kaynak, "Sliding-Mode Control With Soft Computing: A Survey" *IEEE Trans. On Industrial Electronics*, Vol. 56, No. 9, September, 2009.
- [45] A. R. Husain, M. N. Ahmad and M. H. M. Yatim, "Chattering free Sliding Mode Control for an Active Magnetic Bearing System" *World Academy of Science, Engineering and Technology*, 39, 2008.
- [46] B. Wing, K. Ling, H. Ho, C. Lu and K. Lam, *Control of Chaos in Nonlinear Circuits and Systems*, World Scientific Publishing Co. Pte. Ltd, 2009.
- [47] B. Bandyopadhyay, F. Deepak and K. S. Kim, *Sliding Mode Control Using Novel Sliding Surfaces*, Springer-Verlag Berlin Heidelberg, 2009.
- [48] K. D. Young, V. I. Utkin, U. Uzguner, "A Control Engineer's Guide to Sliding Mode Control" *IEEE Trans. On Control System Technology*, Vol. 7, No. 3, 1999.
- [49] G. Monses, discrete time sliding mode control, PhD thesis, 2002.

- [50] X. Yu, G. Chen, "Discretization behaviors of equivalent control based sliding-mode control system" IEEE Trans. On Automatic Control, Vol. 48, pp. 1641-1646, 2003.
- [51] K. C. Veluvolu, Y. C. Soh and W. Cao, "Robust discrete-time nonlinear sliding mode state estimation of uncertain nonlinear systems" International Journal of Robust Nonlinear Control, 17:803-828, 2007.
- [52] Y. Zheng, Y. Jing, "Approximation law for discrete-time variable structure control systems" Journal of Control Theory and Applications 3, 291-296, March 2006.
- [53] M. C. Saaj, B. Bandyopadhyay and H. Unbehauen, "A new algorithm for discrete time sliding mode control using fast output sampling feedback" IEEE Trans. On Industrial Electronics, Vol. 49, pp. 518-523, 2002.
- [54] J. De Jesus Rubio, W. Yu, "a new discrete-time sliding-mode control with time-varying gain and neural identification" International Journal of Control, Vol. 79, No. 4, 338-348, April 2006.
- [55] C. Vivekanandan, R. Prabhakar and M. Gnanambigai, "A Redefined Quasi Sliding Mode Control Strategy" World Academy of Science, Engineering and Technology, 39, 2008.
- [56] N. Jianjun, F. Yongling and Q. Xiaoye, "Design and Application of Discrete Sliding Mode Control with RBF Network-Switching Law" Chinese Journal of Aeronautics, 22, 123-145, 2009.
- [57] R. J. Schilling, Fundamentals of Robotics Analysis & Control, Prentice-Hall of India, New Delhi, 2003.
- [58] H. Chaoui, P. Sicard and W. Gueaieb, "ANN-based Adaptive Control of Robotic Manipulators With Friction and Joint Elasticity" IEEE Trans. On Industrial Electronics, Vol. 56, No. 8, August, 2009.
- [59] M. W. Spong, S. Hutchinson and M. Vidyasagar, Robot Dynamics and Control, Second Edition, Wiley, New York, January 28, 2004.
- [60] G. Incerti, "Trajectory tracking for SCARA robots with compliant transmission: a technique to improve the positioning precision" Proc of the 12th IFTOMM Congress, Besancon, France, 2007.
- [61] A. Visioli, G. Legnani, "On the Trajectory Tracking Control of Industrial SCARA Robot Manipulators" IEEE Trans. On Industrial Electronics, Vol. 49, No. 1, February, 2002.
- [62] M. G. Sanz, I. Egana and J. Villanueva, "Interval Modelling of a SCARA Robot for Robust Control" Mediterranean Conference on Control and Automation, Lisbon, Portugal, July 9-12, 2002.
- [63] R. J. Wai, C. Y. Tu and K. Y. Hsieh, "Adaptive tracking control of robot manipulator" International Journal of System Science, vol. 35, no. 11, pp. 615-627, September, 2004.
- [64] M. T. Hagan, H. B. Demuth, "Neural Networks for Control" Invited Tutorial, American Control Conference, San Diego, pp. 1642-1656, June, 1999.

- [65] M. Hagan, H. Demuth and O. De Jesus, "An Introduction to the use of Neural Networks in Control Systems", *International Journal of Robust and Nonlinear Control*, Vol. 12, No. 11, pp. 959-985, September, 2002.
- [66] H. T. Nguyen, N. R. Prasad, C. L. Walker and E. A. Walker, "A First Course in FUZZY and NEURAL CONTROL", CRC press, 2002.
- [67] M. M. Gupta, L. Jin and N. Homma, *Static and Dynamic Neural Networks from Fundamentals to Advanced Theory*, John Wiley & Sons, 2003.
- [68] G. Lekutai, "Adaptive Self-Tuning Neuro Wavelet Network Controllers", PhD thesis, 1997.
- [69] A. Banakar, M. F. Azeem, "Artificial wavelet neural network and its application in neuro-fuzzy models" *Applied Soft Computing*, 8, pp. 1463-1485, November, 2007.
- [70] W. Jun, P. Hong, "Constructing Fuzzy Wavelet Network Modelling" *International Journal of Information Technology*, Vol. 11, No. 6, 2005.
- [71] L. X. Wang, "A Course in Fuzzy Systems and Control", Prentice-Hall, 1997.
- [72] J. S. R. Jang, "adaptive network based fuzzy inference system" *IEEE Trans. On Systems, Man and Cybernetics*, 23(3), 665-684, 1993.
- [73] B. S. Park, S. J. Yoo, J. B. Park, Y. H. Choi, "Adaptive Neural Sliding Mode Control of Nonholonomic Wheeled Mobile Robots with Model Uncertainty" *IEEE Transaction on Control Systems and Technology*, Vol. 17, No. 1, January, 2009.
- [74] Y. Liu, Y. Li, "Sliding Mode Adaptive Neural-Network Control for Nonholonomic Mobile Modular Manipulators" *Journal of Intelligent and Robotic Systems*, 19, May, 2005.
- [75] Z. Q. Gu, G. A. Vio and S. O. Oyadiji, "Vibration control of nonlinear system using RBF-based sliding mode control" *Int. J. Appl. Math. and Mech*, 3(4): 56-68, May, 2007.
- [76] S. J. Huang, K. S. Huang and K. C. Chiou "Development and application of a novel radial basis function sliding mode controller" *Mechatronics*, Vol. 13, pp. 313-329, 2003.
- [77] H. Guo, S. Lin, J. Liu "A radial basis function sliding mode controller for chaotic Lorenz system" *Physics Letters A* 351, pp 257-261, 2006.
- [78] C. H. Tsai and H. Y. Chung, "Neuro-Sliding Mode Control With Its Applications to Seesaw Systems" *IEEE Trans. Neural Networks*, Vol. 15, No. 1, January, 2004.
- [79] Z. Yu, Y. Tangwen and S. Zengqi, "Neuro-Sliding-Mode Control of Flexible-Link Manipulators Based on Singularity Perturbated Model" *Tsinghua Science and Technology*, Vol. 14, No. 4, August, 2009.
- [80] L. Wu, H. Gao, "Sliding mode control of two dimensional systems in rossler model" *IET Control Theory and Appl*, Vol. 2, No. 4, pp. 352-364, 2008.
- [81] B. Bandyopadhyay, D. Fulwani, "High-Performance Tracking Controller for Discrete Plant Using Nonlinear Sliding Surface" *IEEE Trans. On Industrial Electronics*, Vol. 56, No. 9, September, 2009.
- [82] S. Janardhanan and B. Bandyopadhyay, "on discretization of continuous time terminal sliding mode" *IEEE Trans. On Automatic Control*, Vol. 51, No. 9, pp. 1532-1536, September, 2006.

- [83] K. Abidi, J. X. Xu, and J. H. She "A Discrete-Time Terminal Sliding-Mode Control Approach Applied to a Motion Control Problem" *IEEE Trans. On Industrial Electronics*, Vol. 56, No. 9, September, 2009.
- [84] H. Abid, M. Chtourou, A. Toumi, "Robust Fuzzy Sliding Mode Controller for Discrete Nonlinear Systems" *International Journal of Computers, Communications and Control*, Vol. 3, No. 1, pp. 6-20, 2008.
- [85] M. O. Efe, "Discrete Time Fuzzy Sliding Mode Control of a Biochemical Process" *Proc of the 9th Wseas International Conference on Automatic Control, Modeling and Simulation*, Istanbul, Turkey, May 27-29, 2007.
- [86] M. L. Corradini, G. Orlando, "Control of mobile robots with uncertainties in the dynamical model: a discrete sliding mode approach with experimental results" *Control Engineering Practice*, Vol. 10, pp. 23-34, 2002.
- [87] H. Salarieh, S. M. M. Kashani, G. Vossoughi, A. Alasty, "Stabilizing unstable fixed points of discrete chaotic systems via quasi-sliding mode method" *Communications in Nonlinear Science and Numerical Simulation*, Vol. 14, pp. 839-849, 2009.
- [88] Z. H. Man, A. P. Paplinski, and H. R. Wu, "A robust MIMO terminal sliding mode control scheme for rigid robotic manipulator," *IEEE Tans. Autom. Control*, vol. 39, no. 12, pp. 2464–2469, Dec, 1994.
- [89] Y. Tang, "Terminal sliding mode control for rigid robots," *Automatica*, vol. 34, no. 1, pp. 51–56, Jan, 1998.
- [90] R. M. Zhihong, M. O'Day, "A Robust Adaptive Terminal Sliding Mode Control for Rigid Robotic Manipulators" *Journal of Intelligent and Robotic Systems*, 24: 23-41, 1999.
- [91] Y. Feng, X. Yu and Z. Man, "Non-singular terminal sliding mode control of rigid manipulators" *Automatica*, 38(12), 2159-2167, July 2002.
- [92] L. Wang, T. Chai and L. Zhai, "Neural-Network-Based Terminal Sliding-Mode Control of Robotic Manipulators Including Actuator Dynamics" *IEEE Trans. On Industrial Electronics*, Vol. 56, No. 9, September, 2009.
- [93] C. K. Lin "Nonsingular Terminal Sliding Mode Control of Robot Manipulators using Fuzzy Wavelet Networks" *IEEE Trans. On Fuzzy Systems*, Vol. 14, No. 6, December, 2006.
- [94] M. Jin, J. Lee, P. H. Chang and C. Choi, "Practical Nonsingular Terminal Sliding Mode Control of Robot Manipulators for High-Accuracy Tracking Control" *IEEE Trans. On Industrial Electronics*, Vol. 56, No. 9, September, 2009.
- [95] S. Yu, X. Yu, B. Shirinzadeh and Z. Man, "Continuous finite-time control for robotic manipulators with terminal sliding mode" *Automatica*, 41, 1957-1964, July 2005.
- [96] S. Khoo, L. Xie and Z. Man "Robust Finite-Time Consensus Tracking Algorithm for Multirobot Systems" *IEEE Trans. On Mechatronics*, Vol. 14, No. 2, April, 2009.
- [97] Y. Wu, B. Wang and G. D. Zong, "Finite-Time Tracking Controller Design for Nonholonomic Systems With Extended Chained Form" *IEEE Trans. On Circuits And Systems*, Vol. 52, No. 11, November, 2005.

Abstract

In this thesis, neuro-fuzzy-sliding mode controllers are studied for the three-link SCARA robot. The main goals are to design a free model controller and to eliminate the chattering that is produced by the switching term and also to obtain a robust controller against the joint friction and external disturbance. In this method, a neural network used to approximate the equivalent control law and in a same time to evaluate the switching law, a fuzzy system is utilized. The weights of the neural network are updated such that the fuzzy system output (for the multilayer perceptron network) and the sliding surface (for the radial base network) of the neuro-fuzzy-sliding mode controller approaches to zero. The simulation result of this method is compared with conventional sliding mode. Moreover, in order to design a flexible and low cost controller and also to reduce the computational burden, two other controllers are proposed. In the first one, the previous neuro-fuzzy-sliding mode controller is redefined in discrete time domain and is compared with discrete sliding mode. Additionally, the sampling time effects on the closed loop system convergence are discussed. For the second one, to design a faster controller the fuzzy system is neglected and the nonlinear sliding surface is used to compensate fuzzy system specifications. In all of three above cases, presented simulation results demonstrate the performance improvement in each cases. The proposed controllers also are applicable to large class of MIMO systems.

Keywords: Neuro-fuzzy-sliding mode control, Discrete sliding mode, Nonlinear sliding surface, Neural Network, Fuzzy logic, SCARA robot.



Shahrood University of Technology

Department of Electrical and Robotic Engineering

M.Sc.thesis

Fuzzy Sliding Mode Control with Neural Network on the Three Link SCARA Robot

By:

S. Shoja Majidabad

Supervisors:

Dr. H. T. Shandiz

Advisors:

Dr. M. M. Fateh

September 2010