

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ



دانشگاه صنعتی شاهرود

دانشکده مکانیک

گروه مکاترونیک

طراحی کنترل کننده تطبیقی عصبی - فازی به منظور کنترل مسیر و تعادل

ربات تعادلی دوچرخ

دانشجو: محمدحسن قیصریه‌ها

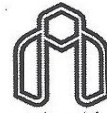
اساتید راهنما:

دکتر حسین خسروی

دکتر علیرضا الفی

پایان نامه جهت اخذ درجه کارشناسی ارشد

شهریورماه سال ۱۳۹۲



دانشگاه صنعتی مازندران

مدیریت تحصیلات تکمیلی

فرم شماره (۶)

باسمه تعالی

شماره:

تاریخ:

ویرایش:

فرم صورت جلسه دفاع از پایان نامه تحصیلی دوره کارشناسی ارشد

با تأییدات خداوند متعال و با استعانت از حضرت ولی عصر (عج) نتیجه ارزیابی جلسه دفاع از پایان نامه کارشناسی ارشد آقای محمدحسن قیصریه ها رشته مکترونیک گرایش مکترونیک تحت عنوان طراحی کنترل کننده تطبیقی عصبی - فازی به منظور کنترل مسیر و تعادل ربات تعادلی دوچرخ که در تاریخ ۹۲/۰۶/۲۶ با حضور هیأت محترم داوران در دانشگاه صنعتی مازندران برگزار گردید به شرح ذیل اعلام می گردد:

<input type="checkbox"/> قبول (با درجه: بسیار خوب امتیاز ۱۸/۸۸)	<input type="checkbox"/> دفاع مجدد	<input type="checkbox"/> مردود
------------------------------------------------------------------------	------------------------------------	--------------------------------

۲- بسیار خوب (۱۸/۹۹ - ۱۸)

۱- عالی (۲۰ - ۱۹)

۴- قابل قبول (۱۵/۹۹ - ۱۴)

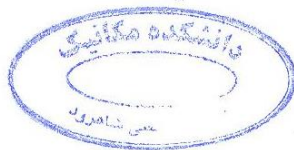
۳- خوب (۱۷/۹۹ - ۱۶)

۵- نمره کمتر از ۱۴ غیر قابل قبول

امضاء	مرتبه علمی	نام و نام خانوادگی	عضو هیأت داوران
	استادیار	حسین حسینی	۱- استاد راهنما اول
	استادیار	علیرضا الحقی	۲- استاد راهنما دوم
	استادیار	امیرحاجری	۳- نماینده شورای تحصیلات تکمیلی
	دانشیار	مهدی رحیمی	۴- استاد ممتحن
	دانشیار	مهدی رحیمی	۵- استاد ممتحن

امضاء

رئیس دانشکده:



تقدیم بہ آمان کہ اولین آموزگار انم بودند

مادرم زہرا

پدرم غلامحسین

ویدر بزرگم رستم

پاس خدایی را که ستودن برانزده می اوست

زبان از بیان سپاسی شایسته قاصر است لذا به حکم ادب و با الفاظی همیشگی تشکر می کنم از همه ی کسانی که یاریم دادند.

از استاد گران بایه جناب آقای دکتر خسروی که دست از حمایت برداشتند و همواره بار آسمانی های ارزشمند خود یاریم دادند و همچنین از

جناب آقای دکتر الفی که با اعتماد به شکر کوچک خود، امکان استفاده از امکانات آزمایشگاهی را برایم فراهم ساختند.

تعهدنامه

این جانب محمدحسن قیصریه‌ها دانشجوی دوره کارشناسی ارشد رشته مکترونیک دانشکده مکانیک دانشگاه صنعتی شاهرود نویسنده پایان‌نامه: طراحی کنترل‌کننده تطبیقی عصبی - فازی به منظور کنترل مسیر و تعادل ربات تعادلی دوچرخه، تحت راهنمایی دکتر حسین خسروی و دکتر علیرضا الفی متعهد می‌شوم .

- تحقیقات در این پایان‌نامه توسط این جانب انجام شده است و از صحت و اصالت برخوردار است .
- در استفاده از نتایج پژوهش‌های محققان دیگر به مرجع مورد استفاده استناد شده است .
- مطالب مندرج در پایان‌نامه تاکنون توسط خود یا فرد دیگری برای دریافت هیچ نوع مدرک یا امتیازی در هیچ جا ارائه نشده است .
- کلیه حقوق معنوی این اثر متعلق به دانشگاه صنعتی شاهرود می‌باشد و مقالات مستخرج با نام «دانشگاه صنعتی شاهرود» و یا « Shahrood University of Technology » به چاپ خواهد رسید .
- حقوق معنوی تمام افرادی که در به دست آمدن نتایج اصلی پایان‌نامه تأثیرگذار بوده‌اند در مقالات مستخرج از پایان‌نامه رعایت می‌گردد.
- در کلیه مراحل انجام این پایان‌نامه ، در مواردی که از موجود زنده (یا بافت‌های آن‌ها) استفاده شده است ضوابط و اصول اخلاقی رعایت شده است .
- در کلیه مراحل انجام این پایان‌نامه، در مواردی که به حوزه اطلاعات شخصی افراد دسترسی یافته یا استفاده شده است اصل رازداری ، ضوابط و اصول اخلاق انسانی رعایت شده است .

تاریخ

امضای دانشجو

مالکیت نتایج و حق نشر

- کلیه حقوق معنوی این اثر و محصولات آن (مقالات مستخرج ، کتاب ، برنامه‌های رایانه‌ای ، نرم‌افزارها و تجهیزات ساخته شده است) متعلق به دانشگاه صنعتی شاهرود می‌باشد . این مطلب باید به نحو مقتضی در تولیدات علمی مربوطه ذکر شود .
- استفاده از اطلاعات و نتایج موجود در پایان‌نامه بدون ذکر مرجع مجاز نیست.

چکیده فارسی:

پاندول معکوس به عنوان یک مسئله‌ی کلاسیک در دینامیک و کنترل شناخته می‌شود که اغلب به منظور بررسی کارایی نظریه‌های کنترلی مورد استفاده قرار می‌گیرد. در سال‌های اخیر، ایده‌ی ربات‌های تعادلی دوچرخ با الگو گرفتن از پاندول معکوس مطرح شده است، غیرخطی، ناپایدار و غیرهولونومیک بودن ربات موجب شده است به حجم بالایی از محاسبات برای مدل‌سازی و کنترل این سیستم نیاز باشد.

تاکنون مقالات ارائه‌شده از دیدگاه‌های مختلفی ربات تعادلی را مورد بررسی قرار داده‌اند، گروهی به معرفی این ربات به عنوان یک وسیله نقلیه پرداختند، گروهی کوشیده‌اند تا معادلات حرکتی جامعی برای آن استخراج کنند و گروهی نیز این ساختار را (چون گذشته) برای امتحان نظریه‌های کنترلی خود به کار برده‌اند.

در دهه‌ی گذشته روش‌های کنترل غیرخطی برای کنترل موقعیت، سرعت و تعادل ربات پیشنهاد و مورد بررسی قرار گرفته است، همچنین در سال‌های اخیر استفاده از روش‌های هوشمند برای کنترل ربات رشد چشمگیری داشته و تا جایی پیش رفته است که الگوریتم‌های هوشمند مقاوم و بهینه، برای کنترل ربات حتی بدون در نظر گرفتن معادلات دینامیکی آن پیشنهاد شده است.

این پایان‌نامه ابتدا به معرفی ربات تعادلی کوچکی به نام «زحل» پرداخته و سپس با بهره‌گیری از روش‌های غیرخطی (مد لغزشی) و هوشمند (شبکه عصبی فازی) به کنترل ربات می‌پردازد. لازم به ذکر است ربات زحل نه به عنوان یک وسیله‌ی نقلیه که تحت عنوان یک ریز ربات مطرح می‌شود. در پیاده‌سازی عملی نشان می‌دهیم که روش کنترل مد لغزشی به عنوان یک کنترل‌کننده‌ی مقاوم که بر پایه‌ی مدل سیستم ساخته می‌شود، توانایی کنترل ربات در بروز اغتشاشات خارجی را دارد، اما این روش نوسانات شدیدی را به دنبال داشته و خطای ردگیری زیادی دارد. پس به سراغ طراحی یک شبکه عصبی- فازی برای کنترل همزمان تعادل و مسیر حرکت ربات رفته و نشان می‌دهیم علی‌رغم توانایی کنترلی مناسب و نوسانات کم، این کنترل‌کننده‌ها در عمل زمان لازم برای آموزش را پیدا نکرده و به علت کوچک بودن ابعاد ربات و کم بودن تاخیر مکانیکی، نمی‌توانند با اغتشاشات خارجی مواجه شوند. برای برطرف کردن این مشکل، ترکیبی خطی از کنترل‌کننده‌ی مد لغزشی و شبکه عصبی- فازی ارائه‌شده است که تابع ترکیب‌کننده‌ای همواره به تلفیق خروجی کنترل‌کننده‌ها پرداخته و بر اساس ضریب خطای ورودی، درصدی از سیگنال‌های کنترلی را به خروجی منتقل می‌سازد.

کلمات کلیدی: ربات تعادلی، ربات متحرک، پاندول معکوس دوچرخ، عصبی فازی، مد لغزشی، ربات دوچرخ.

فهرست مطالب

عنوان.....صفحه

۱- فصل اول: مقدمه ۱

۱-۱- مقدمه ۲

۱-۲- مزایای ربات تعادلی در مقایسه با ربات‌های متحرک ۴

۱-۳- هدف پایان‌نامه ۷

۱-۴- ساختار پایان‌نامه ۸

۱-۵- جمع‌بندی ۱۰

۲- فصل دوم: مروری بر فعالیت‌های پیشین ۱۱

۲-۱- مقدمه ۱۲

۲-۲- بررسی مقالات و پایان‌نامه‌های پیشین ۱۳

۲-۳- جمع‌بندی و نتیجه‌گیری ۲۰

۳- فصل سوم: پیش‌زمینه ۲۲

۳-۱- روش مدولاسیون پهنای پالس ۲۳

۳-۲- روش کنترل مد لغزشی ۲۴

۳-۲-۱- مقدمه ۲۴

۳-۲-۲- سطح لغزش ۲۵

۳-۲-۳- طراحی کنترل‌کننده ۲۶

۳-۳- شبکه‌های عصبی فازی ۲۸

۳-۳-۱- شبکه‌های عصبی پیش‌خور ۲۸

۳-۳-۲- شبکه‌های عصبی بازگشتی ۲۹

۳-۴- منطق فازی ۲۹

۳-۴-۱- انواع شبکه عصبی-فازی و نورو-فازی ۳۰

۳-۴-۲- شبکه عصبی فازی استفاده‌شده در این پژوهش ۳۱

۳-۵- جمع‌بندی ۳۶

۴- فصل چهارم: مدل‌سازی دینامیکی و سینماتیکی ربات ۳۷

۴-۱- مقدمه ۳۸

۴-۲- استخراج معادلات سینماتیکی ربات ۳۹

۴-۳- روابط موقعیت زمان‌گسسته‌ی ربات ۴۱

۴۲	۴-۴- استخراج معادلات دینامیکی
۴۸	۵-۴- مدل سازی دینامیکی موتورها
۵۰	۶-۴- جمع بندی
۵۱	۵- فصل پنجم: انتخاب قطعات و ساخت ربات تعادلی
۵۲	۱-۵- معرفی ساختار
۵۳	۲-۵- مراحل ساخت
۵۵	۳-۵- انتخاب قطعات
۵۵	موتور
۵۶	باتری
۵۶	حسگرها
۵۷	بدنه (شاسی)
۵۸	۴-۵- طراحی مکانیکی
۵۸	۵-۵- طراحی الکترونیکی
۶۰	۶-۵- ساخت
۶۱	۶- فصل ششم: طراحی کنترل کننده و شبیه سازی عملکرد
۶۲	۱-۶- مقدمه
۶۲	۲-۶- طراحی مسیر
۶۳	۳-۶- طراحی کنترل کننده مد لغزشی تفکیک شده
۷۴	۴-۶- طراحی کنترل کننده عصبی - فازی
۸۰	۵-۶- تلفیق کنترل کننده مد لغزشی و کنترل کننده عصبی - فازی
۸۷	۶-۶- جمع بندی
۸۸	۷- فصل هفتم: نتیجه گیری و پیشنهادها
۸۹	۱-۷- نوآوری تحقیق
۸۹	۲-۷- نتیجه گیری
۹۰	۳-۷- پیشنهادها
۹۲	۸- منابع

فهرست شکل‌ها

- شکل ۱-۱: وضعیت تعادل ربات در صورت اضافه شدن بار نامتعادل ۶
- شکل ۲-۱: نمایش نحوه حرکت ربات تعادلی و ربات معمولی در سطح شیب‌دار ۶
- شکل ۳-۱: ربات تعادلی طراحی شده در این پایان‌نامه ۷
- شکل ۱-۲: ربات JOE ۱۳
- شکل ۲-۲: ربات Segway ۱۴
- شکل ۳-۲: ربات ساخته شده توسط کیم ۱۶
- شکل ۴-۲: ربات تعادلی طراحی شده به وسیله لونگ ۱۸
- شکل ۵-۲: ربات چارمر هنگام حمل یک ظرف سبک ۱۹
- شکل ۶-۲: نمودار زمانی ربات‌های تعادلی ساخته شده از سال ۲۰۰۰ تا کنون ۲۱
- شکل ۱-۳: شکل موج یک سیگنال PWM ۲۴
- شکل ۲-۳: پدیده‌ی وزوز ۲۷
- شکل ۳-۳: نمای کلی شبکه عصبی پیشخور ۲۸
- شکل ۴-۳: نمایی از یک شبکه عصبی بازگشتی ۲۹
- شکل ۵-۳: شبکه عصبی فازی استفاده شده در این پژوهش ۳۱
- شکل ۱-۴: شمای سینماتیک ربات تعادلی دوچرخ ۳۹
- شکل ۲-۴: نمایش وضعیت جابه‌جایی ربات ۴۱
- شکل ۳-۴: شمای سه بعدی ربات تعادلی دوچرخ ۴۴
- شکل ۴-۴: مدل مداری موتور مغناطیس دائم ۴۸
- شکل ۱-۵: بلوک دیاگرام ربات تعادلی زحل ۵۲
- شکل ۲-۵: ربات تعادلی CPG ۵۳
- شکل ۳-۵: موتور انتخاب شده برای ربات ۵۶
- شکل ۴-۵: حسگرهای ربات ۵۷
- شکل ۵-۵: چرخ ربات ۵۸
- شکل ۶-۵: نمای از فیبر مدار چاپی طراحی شده ۵۹
- شکل ۷-۵: ربات زحل ۶۰
- شکل ۱-۶: مسیر طراحی شده برای ربات تعادلی ۶۳
- شکل ۲-۶: بلوک دیاگرام کنترل کننده مد لغزشی برای کنترل مسیر ۶۸
- شکل ۳-۶: نتیجه ردگیری کنترل کننده مد لغزشی در محیط شبیه سازی ۶۸
- شکل ۴-۶: نتیجه ردگیری کنترل کننده مد لغزشی برای ربات واقعی ۶۹
- شکل ۵-۶: انحراف زاویه بدنه برای کنترل کننده مد لغزشی در محیط شبیه سازی ۷۰
- شکل ۶-۶: انحراف زاویه بدنه برای کنترل کننده مد لغزشی برای ربات واقعی ۷۰
- شکل ۷-۶: نمایش خطای دائمی زاویه ی بدنه ربات ایجاد شده بر اثر خطای انتگرال گیری ۷۱
- شکل ۸-۶: چگونگی اصلاح خطا با ضریب اصلاح $k_q = 0.05$ ۷۲
- شکل ۹-۶: چگونگی اصلاح خطا با ضریب اصلاح $k_q = 0.1$ ۷۲
- شکل ۱۰-۶: ایجاد نوسانات زیاد در سیگنال زاویه ، در صورت انتخاب ضریب تصحیح بزرگ ۷۲
- شکل ۱۱-۶: سیگنال اصلاح شده با انتخاب ضریب اصلاح مناسب ۷۳

- شکل ۶-۱۲: نتیجه ردگیری برای کنترل کننده مد لغزشی پس از اصلاح خطای انتگرال گیری ۷۳
- شکل ۶-۱۳: بلوک دیاگرام شبکه عصبی- فازی برای کنترل ربات ۷۵
- شکل ۶-۱۴: نتیجه شبیه سازی ردگیری با استفاده از شبکه عصبی فازی ۷۵
- شکل ۶-۱۵: نوسانات بدنه با استفاده از شبکه عصبی فازی در محیط شبیه سازی ۷۶
- شکل ۶-۱۶: نتیجه ردگیری در استفاده از کنترل کننده عصبی- فازی برای ربات واقعی ۷۶
- شکل ۶-۱۷: نوسانات بدنه در پیاده سازی عملی کنترل کننده ی فازی - عصبی ۷۷
- شکل ۶-۱۸: نمایش مسئله ردگیری برای ۹ گره در لایه ی قوانین ۷۸
- شکل ۶-۱۹: نمایش نوسانات بدنه در حالت استفاده از ۹ گره در لایه ی قوانین ۷۸
- شکل ۶-۲۰: پاسخ کنترل کننده عصبی - فازی به ورود اغتشاشات نابهنگام ۷۹
- شکل ۶-۲۱: پاسخ کنترل کننده مد لغزشی به ورود اغتشاشات نابهنگام ۸۰
- شکل ۶-۲۲: نحوه استفاده از ترکیب کننده برای تلفیق دو کنترل کننده ۸۱
- شکل ۶-۲۳: بلوک دیاگرام کنترل کننده ی ترکیب شده ی پیشنهادی ۸۳
- شکل ۶-۲۴: ردگیری با استفاده از کنترل کننده ترکیبی در حضور اغتشاشات ۸۳
- شکل ۶-۲۵: نوسانات بدنه در حضور اغتشاش خارجی و با بهره گیری از کنترل کننده ترکیبی ۸۴
- شکل ۶-۲۶: نمایش سیگنال های خطا در کنار نحوه ی به روز رسانی وزن های ترکیب کننده ۸۵
- شکل ۶-۲۷: نحوه ردگیری و خطای بدنه برای مسیر بیضوی ۸۵
- شکل ۶-۲۸: نحوه ردگیری و خطای بدنه برای مسیر مارپیچ ۸۶
- شکل ۶-۲۹: بلوک دیاگرام استفاده شده در متلب ۸۶

فهرست جدول‌ها

- جدول ۱-۴: علامت‌های اختصاری استفاده‌شده در مدل‌سازی سینماتیکی ۳۹
- جدول ۲-۴: نمادهای استفاده‌شده در مدل‌سازی دینامیکی ۴۵
- جدول ۳-۴: نمادهای مورد استفاده در مدل‌سازی موتور مغناطیس دائم ۴۸
- جدول ۱-۵: وزن قطعات مختلف ربات ۶۰
- جدول ۱-۶: پارامترهای طراحی کنترل‌کننده مد لغزشی ۶۸
- جدول ۲-۶: پارامترهای طراحی برای کنترل‌کننده عصبی - فازی ۷۴
- جدول ۳-۶: نتایج حاصل از شبیه‌سازی کنترل‌کننده با تعداد مختلفی گره در لایه‌ی قوانین ۷۷

۱- فصل اول

مقدمه

۱-۱- مقدمه

پاندول معکوس یک مسئله‌ی سنتی و مشهور در دینامیک و کنترل به شمار می‌رود که ناپایداری و ساختار غیرخطی آن باعث شده تا به عنوان پیش‌الگویی^۱ جهت بررسی نظریه‌های کنترلی مورد استفاده قرار گیرد [1] و [2]. گاری و آونگ برگشته^۲ یکی از مسائل اصلی مطرح‌شده با استفاده از قضیه پاندول معکوس است این ساختار از یک وسیله‌ی چرخ دار تشکیل شده که آونگی به صورت وارونه روی آن قرار گرفته و هدف اصلی آن کنترل زاویه‌ی پاندول از طریق جابه‌جایی وسیله‌ی زیرین است. این مسئله‌ی مشهور اگرچه ممکن است مثال‌های فراوانی در دنیای واقعی داشته باشد اما تا سالیان زیادی به عنوان یک الگوی آزمایشگاهی برای برقراری تعادل اجسام ناپایدار مطرح بود و نمود عملی از آن وجود نداشت. سال‌های ابتدایی قرن ۲۱ را می‌توان شروع ارائه‌ی الگوی جدیدی تحت عنوان پاندول معکوس دوچرخ^۳ دانست [3]. این مسئله به زودی تا آنجا رشد کرد که در فاصله زمانی اندکی به اختراع وسیله‌ی نقلیه دوچرخ^۴ انجامید [4].

با نگاهی دقیق‌تر حتی می‌توان پاندول معکوس دوچرخ را به عنوان الگویی ساده از بدن انسان تلقی کرد که با وجود ناپایداری، همواره نیاز به حفظ تعادل در هنگام ایستادن، راه رفتن و دویدن را دارد. مطرح‌شدن ایده‌ی ربات‌های تعادلی دوچرخ^۵ که با الگو گرفتن از پاندول معکوس بوده، افق جدیدی به روی دنیای ربات‌های متحرک گشوده است. اما این ربات‌ها سیستم‌هایی غیرخطی، ناپایدار، تزویج شده^۶ و درجه بالا هستند از این رو برای مدل‌سازی و کنترل آنان نیاز به حجم بالایی از محاسبات می‌باشد [5]، [6].

مسئله اصلی در کنترل ربات تعادلی، داشتن کمترین لرزش، مقابله با اغتشاشات خارجی و پیمودن مسیری مشخص شده است، اما باید در نظر داشت که کنترل این وسیله کار چندان ساده‌ای نیست

¹ Prototype

² Cart-Inverted pendulum

³ Two-Wheeled inverted pendulum

⁴ Two-Wheeled vehicle

⁵ Two-Wheeled balancing Robot

⁶ Coupled

چراکه علاوه بر غیرخطی بودن شدید معادلات سیستم و وجود عدم قطعیت‌های^۱ فراوان، تعداد بازخوردهای کنترلی، کمتر از درجه‌های آزادی سیستم بوده و برخورداری از قیود غیرهولونومیک^۲، کنترل سیستم را با چالش‌های شدیدی روبه‌رو می‌سازد [7].

اکثر تحقیقات صورت گرفته روی ربات تعادلی، یا به منظور نشان دادن توانایی یک کنترل‌کننده در متعادل‌سازی سیستم‌های ناپایدار غیرخطی است [7]، [8]، [9] و [10]. یا این ربات را تحت عنوان یک وسیله نقلیه دوچرخ معرفی نموده است [11]، [12]. اما در این بین کمتر به معرفی این ربات تحت عنوان یک ربات متحرک خودکار^۳ پرداخته شده است درحالی که ربات‌های متحرک حجم قابل توجهی از مطالعات رباتیک را به خود معطوف ساخته است [13].

ربات‌های متحرک اغلب در ساختارهای سه چرخ یا چهارچرخ معرفی می‌شوند [14] ولی در اکثر موارد تنها دو محور حرکتی برای ربات وجود دارد که مستقیم به موتورهای ربات متصل می‌شوند (به غیر از ربات‌های چند جهتی). وجود چرخ‌های هرز گرد در این گونه ربات‌ها اگرچه باعث برقراری تعادل است اما توانایی حرکت در مسیرهای ناهموار و همچنین امکان عبور از موانع کوچک را از ربات سلب می‌کند. علاوه بر آن به علت صرف‌نظر کردن از دینامیک چرخ‌های هرز گرد و وجود اصطکاک، عواملی چون: دینامیک‌های مدل نشده و عدم قطعیت‌های پارامتری وارد سیستم می‌شوند [15] که منجر به افزایش خطای کنترل‌کننده‌ها می‌گردد و لذا توان کنترلی بیشتری برای ردگیری می‌طلبد.

اگر ربات تعادلی دوچرخ را یک ربات متحرک خودکار در نظر بگیریم، علاوه بر حفظ تعادل، مسئله‌ی جدیدی تحت عنوان پیمایش مسیر^۴، در دستور کار قرار خواهد گرفت که فرایند کنترل را کمی پیچیده می‌نماید [8]. داشتن یک ربات متحرک با چنین پیچیدگی‌هایی چندان معقول به نظر نمی‌رسد بالاخص زمانی که ساختارهای سه چرخ و چهارچرخ شرایط بهتری را برای کنترل این گونه ربات‌ها رغم می‌زنند

¹ Uncertainty

² Nonholonomic

³ Automatic Mobile Robot

⁴ Tracking

اما باید در نظر داشت که اضافه شدن این پیچیدگی ها، محاسنی را نیز به دنبال خواهد داشت که در ادامه به شرح آن می پردازیم.

ربات های متحرک فراوانی وجود دارند که تحت عنوان ربات های هوشمند مسابقه ای مطرح می شوند، ربات تعقیب خط^۱، کاوشگر، فوتبالیست های سایز کوچک و سایز متوسط، امداد رسان^۲، ربات درخانه^۳، موش هوشمند^۴ و... نمونه هایی از این دسته به شمار می روند. در کنار آن دسته ی دیگری از ربات های متحرک تحت عنوان ریز ربات ها^۵ معرفی می شوند که معمولاً برای مقاصد کاوشگرانه، نمایشی، جاسوسی و ... مورد استفاده هستند. اغلب این ربات ها اگر چه برای کاربردهای خاص و روی سطوح تقریباً هموار مورد استفاده قرار می گیرند، اما به علت ابعاد کوچکشان همواره با مشکلاتی روبه رو هستند که لازم است پیش از مطرح کردن مزایای ساختار تعادلی، به عنوان کردن آنها بپردازیم.

داشتن ابعاد کوچک باعث می شود تا این ربات ها از طراحی مکانیکی ساده ای برخوردار باشند. معمولاً کویلینگ بین موتورها و چرخ ها به صورت مستقیم صورت گرفته و ربات از مکانیزه هایی نظیر: تعلیق، دیفرانسیل و... بی بهره است. لذا در موارد چهارچرخ، هیچگاه نمیتوان تماس هر چهار چرخ را با زمین تضمین کرد و وجود ناهمواری های جزئی باعث عدم درگیری یکی از چرخ ها با زمین می گردد. در صورتی که چرخ درگیر نشده، به یکی از محرک ها کوپل شده باشد؛ عملاً حرکت روی مسیر طراحی شده برای ربات ناممکن می گردد. یکی از دلایل مهم استفاده از ساختار سه چرخ توجه به همین امر می باشد. ساختار سه چرخ اغلب دارای دو چرخ محرک و یک چرخ هرزگرد است. وجود چرخ هرزگرد باعث می شود تا ربات در عبور از موانع کوچک و مواجهه با شیب ها و ناهمواری هر چند کم؛ با مشکل مواجه شود به علاوه به علت نیاز به برقراری تعادل مخصوصاً با افزایش ارتفاع ربات، حداقل فاصله ای

¹ Line follower

² Rescue

³ At-Home

⁴ Micro mouse

⁵ Micro Robot

میان تکیه گاه‌های ربات باید رعایت شود که قیودی به طراحی مکانیکی اضافه کرده و سبب بالارفتن ابعاد ربات می‌گردد. با این مقدمه به سراغ معرفی مزایای ساختار تعادلی در ربات‌های متحرک می‌رویم.

۱-۲- مزایای ربات تعادلی در مقایسه با ربات‌های متحرک

استفاده از مزایای ساختار تعادلی در طراحی ربات‌های متحرک می‌تواند باعث افزایش کارایی و قابلیت‌های حرکتی آنان شود. این مزایا عبارت‌اند از:

۱- تمرکز نیروی وزن روی نقاط محرک یا به عبارت دیگر حذف تکیه‌گاه‌های اضافی: به علت اینکه

ربات همواره روی دو تکیه‌گاه قرار گرفته است تماس این تکیه‌گاه‌ها با زمین در حالت

تضمین‌شده‌تری نسبت به حالت سه چرخ و مخصوصاً چهارچرخ دارد. به علاوه در حرکت روی

مسیرهای ناهموار احتمال گیر کردن چرخ‌های هرز گرد به برآمدگی‌ها و یا افتادن این چرخ در

فرورفتگی‌ها که مانع از حرکت ربات شده یا در حرکت آن اختلال ایجاد می‌کند وجود ندارد.

۲- قرار گرفتن مرکز جرم در راستای عمود: این کمک را به ربات می‌کند تا حتی در مسیرهای

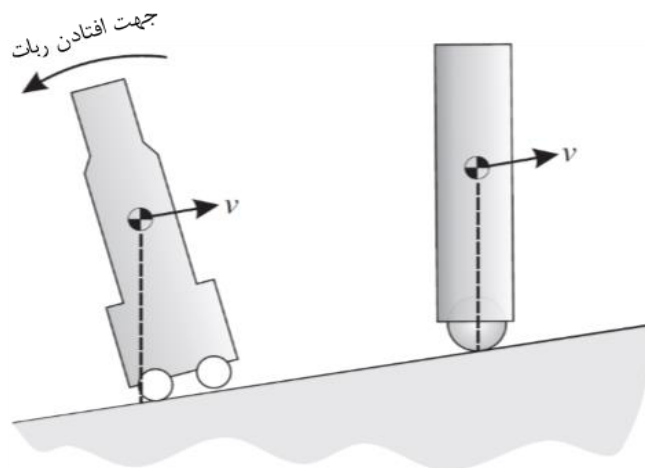
شیب‌دار بتواند تعادل خود را حفظ نماید. این در حالی است که در ربات‌های متحرک سه چرخ

و چهارچرخ با وجود تعادل ساختاری روی زمین مسطح، در صورت حرکت در مسیرهای شیب‌دار،

تعادل ربات با مشکل مواجه می‌شود، این مشکل در ربات‌های مرتفع و علی‌الخصوص در حرکت

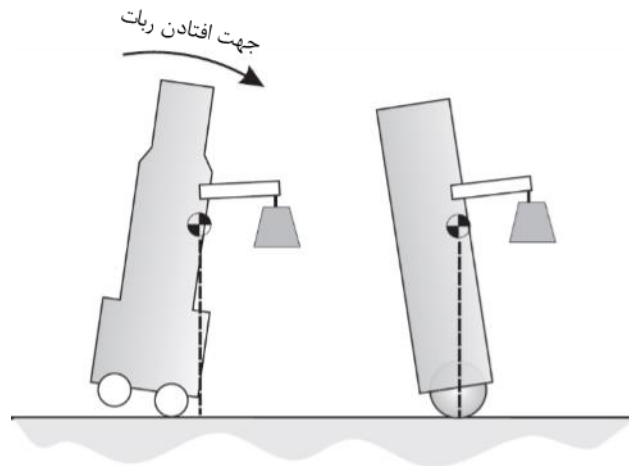
روی شیب‌های منفی به وضوح قابل مشاهده است. شکل (۱-۱) به درک واضح‌تری از این قضیه

کمک می‌کند.



شکل ۱-۱: نمایش نحوه حرکت ربات تعادلی و ربات معمولی در سطح شیب‌دار [35]

۳- امکان برقراری تعادل در صورت اضافه‌شده بار نامتعادل: به علت اهمیت مسئله حفظ تعادل و وجود فرآیند کنترلی برای دستیابی به آن، اضافه شدن بار نامتعادل تعادل ربات را برهم نخواهد زد و ربات همچنان در حالت تعادل باقی می‌ماند. این در حالی است که در نوع معمولی، به علت عدم توجه به این مسئله، بار اضافه‌شده می‌تواند باعث برهم زدن تعادل ربات گردد. این امر در شکل (۱-۲) به وضوح قابل مشاهده است.

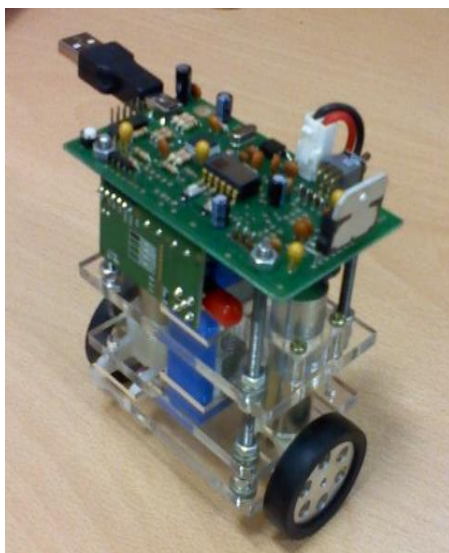


شکل ۲-۱ وضعیت تعادل ربات در صورت اضافه شدن بار نامتعادل [35]

۱-۳- هدف پایان نامه

این پژوهش دو هدف اصلی را دنبال می‌کند

۱- استفاده از مزایای ربات‌های تعادلی در طراحی ربات‌های متحرک: برای این منظور ربات تعادلی متحرکی به نام "زحل" معرفی شده است. این ربات دارای ساختاری کوچک، سبک و هوشمند است که علاوه بر پیمایش مسیر دلخواه قابلیت حفظ تعادل روی دو چرخ را دارد. از این رو آن را می‌توان به عنوان ساختاری جدید برای ریز ربات‌های هوشمند برشمرد. بدنه‌ی ربات از جنسی شفاف و شیشه‌ای انتخاب شده به گونه‌ای که تمام اتصالات و قطعات استفاده شده با چشم قابل رویت می‌باشد. ربات تعادلی طراحی شده در شکل (۱-۳) به نمایش در آمده است.



شکل ۱-۳: ربات تعادلی معرفی شده در این پایان نامه

۲- تلفیق روش‌های هوشمند و کلاسیک برای دستیابی به کنترل کننده‌ای مناسب در مواجهه با اغتشاشات خروجی: به این منظور، خروجی دو کنترل کننده‌ی مدل‌گزشی و شبکه‌ی عصبی-فازی با یکدیگر به صورت خطی ترکیب شده و بر اساس مناسب بودن جواب هر کنترل کننده در شرایط خاص، به ربات اعمال می‌گردد، این امر سبب می‌شود تا ربات علاوه بر کنترل مناسب در حالت بدون اغتشاش، بتواند در مقابل بروز اغتشاشات نیز از خود عمل مناسبی نشان دهد.

مقاصد اصلی کنترلی در ادامه بیان می‌گردد، ربات طراحی شده باید بتواند:

- ۱- تعادل خود را روی دوچرخ حفظ کند.
- ۲- در صورت ورود اغتشاشات محیطی محدود تعادل خود را از دست ندهد.
- ۳- مسیر طراحی شده را با کمترین خطای ممکن پیمایش کند.
- ۴- در حین پیمایش مسیر تعادل خود را از دست ندهد.
- ۵- قابلیت عبور از مسیرهای با شیب کم را داشته باشد.

۱-۴- ساختار پایان نامه:

در ادامه و با شروع فصل دوم، مروری بر فعالیتهای گذشته در زمینه‌ی ربات‌های تعادلی صورت گرفته و مقالات و پایان‌نامه‌های نوشته‌شده در این زمینه اجمالاً مورد بررسی قرار می‌گیرد.

در فصل سوم توضیحاتی در مورد مفاهیم اولیه‌ی به‌کاررفته در این پایان‌نامه بیان شده است، ابتدا روش پهنای پالس که برای کنترل محرک‌های ربات استفاده‌شده تعریف گردیده و در ادامه‌ی فصل مباحث اصلی کنترلی به ترتیب استفاده در پیاده‌سازی‌ها عنوان می‌گردد. بر این اساس: طراحی کنترل‌کننده مد لغزشی به تفصیل بیان‌شده، مروری اجمالی بر مفهوم شبکه‌های عصبی-فازی انجام‌گرفته و در نهایت نحوه طراحی یک شبکه عصبی-فازی استفاده‌شده برای کنترل ربات معرفی شده است.

فصل چهارم به استخراج معادلات کلی سیستم اختصاص دارد. در ابتدای این فصل و برای مشخص شدن قیود سینماتیکی حاکم بر سیستم، معادلات سینماتیکی ربات استخراج شده است، از طرفی به علت لزوم پیاده‌سازی نتایج شبیه‌سازی روی ربات، نیاز به دانستن برخی تبدیلات مختصات و همچنین معادلات سینماتیکی ربات در فضای گسسته است لذا معادلات سینماتیکی زمان گسسته نیز به دست آمده است. پس از استخراج معادلات سینماتیک به سراغ نوشتن معادلات حرکت می‌رویم. معادلات دینامیکی ربات تعادلی بر اساس روش معروف اوپلر-لاگرانژ استخراج شده است این معادلات ساختاری کاملاً غیرخطی دارد. در ادامه از فرم حالت معادلات استفاده شده و برای کاهش مختصات تعمیم‌یافته و حذف ضریب لاگرانژ از روشی مرسوم استفاده می‌شود و نهایتاً معادلات دینامیکی پایانی به دست می‌آید،

اما ورودی سیستم دینامیکی ربات در این حالت برابر با گشتاوری است که توسط محرک‌های الکتریکی تولید می‌شود و در عمل ولتاژ این محرک‌هاست که توسط کنترل‌کننده کنترل می‌گردد. پس لازم است تا معادله دینامیکی موتور نوشته‌شده و در معادلات سیستم قرار گیرد. پایان فصل چهارم به این امر اختصاص یافته است.

فصل پنجم به ساخت ربات اختصاص دارد که در آن قطعات استفاده‌شده برای ربات معرفی، نحوه‌ی انتخاب هر یک عنوان و چگونگی طراحی‌ها بیان گردیده است.

در فصل ششم به بیان نتایج شبیه‌سازی‌ها و مقایسه آن با پیاده‌سازی‌های عملی می‌پردازیم. در این فصل ابتدا از یک کنترل‌کننده مد لغزشی برای کنترل سرعت ربات استفاده می‌گردد، پس از آن نقایص این روش مشخص و از کنترل‌کننده‌ای عصبی- فازی برای کنترل موقعیت و سرعت ربات استفاده می‌شود. نتایج این روش با روش قبل مقایسه شده و در نهایت از تلفیق دو روش به روشی کارآمدتر برای ربات دست می‌یابیم تلفیق دوروش اگر چه ممکن است امری معمول باشد اما برای اولین بار در کنترل این ساختار مورد استفاده قرار گرفته است.

فصل هفتم نیز به عنوان آخرین فصل این پژوهش، به نتیجه‌گیری و بیان پیشنهادها اختصاص یافته است.

لازم به ذکر است تمامی مراحل بیان‌شده در این پایان‌نامه با کمک گرفتن از نرم‌افزار متلب و در محیط شبیه‌سازی صورت می‌گیرد. همچنین با استفاده از امکانات این نرم‌افزار در پردازش زمان واقعی¹ داده‌ها، اطلاعات ربات از طریق ارتباط USB به رایانه انتقال یافته و پس از محاسبه مقادیر در محیط شبه سازی²، نتایج نهایی به ربات ارسال می‌گردد.

¹ Real-time

² Simulink

۱-۵- جمع‌بندی:

در این فصل ابتدا مباحثی مقدماتی پیرامون پیدایش ربات‌های تعادلی عنوان شد، پس از آن مقایسه‌ی کوتاهی از ساختار تعادلی با ساختارهای سنتی ربات‌های تعادلی صورت گرفت. به لزوم استفاده از کنترل‌کننده‌های تطبیقی و مقاوم برای کنترل این‌گونه ربات‌ها اشاره شد، پس از بیان مزایای ربات تعادلی به مختصری از اهداف پایان‌نامه پرداخته شد و در پایان نیز مروری بر فصل‌های مختلف پایان‌نامه صورت گرفت.

۲- فصل دوم

مروری بر فعالیت‌های پیشین

۲-۱- مقدمه:

همواره در ابتدای هر تحقیق، فصلی به مرور فعالیت‌های پیشینیان اختصاص می‌یابد تا نشان دهد که پژوهش انجام‌شده بر پایه‌ی اطلاعاتی است که از پیشینیان گرفته شده است، علاوه بر آن ذهن خواننده را برای درک پیش‌زمینه‌ی موضوع و نحوه‌ی پیشرفت این موضوع در دنیای خارج آماده می‌نماید. از این رو در این فصل به بیان مختصری از فعالیت‌های انجام‌شده در زمینه ربات‌های تعادلی می‌پردازیم.

با توجه به این که ربات تعادلی دوچرخ ایده‌ای برگرفته از پاندول معکوس می‌باشد، پیشینه‌ی آن را می‌توان به آغاز تحقیقاتی نسبت داد که در سال‌های دور بر روی پاندول معکوس آغاز شد. "در حدود یک قرن پیش و در سال ۱۹۰۸ برای اولین بار شرایط پایداری یک پاندول معکوس به صورت عملی توسط استفانسون مورد بررسی قرار گرفت. وی عنوان کرد که با اعمال یک نوسان دائمی متناوب به پاندول معکوس می‌توان آن را در شرایط پایدار قرارداد."

با گسترش یافتن تحقیقات بر روی این مدل، لواسترن موفق شد در سال ۱۹۳۲ الگویی برای معادلات حرکت عمومی پاندول معکوس ارائه دهد. از سال ۱۹۶۰ دینامیک این سیستم به طور کامل شناخته شد و در این دهه تحقیقات زیادی در آنالیز یا تخمین پاسخ پاندول معکوس به ورودی‌های مختلف انجام گرفت. سرانجام مقاله‌ای در سال ۱۹۸۰ توانست کنترل‌کننده‌ای ارائه دهد که قادر به پایدارسازی پاندول معکوس برای محدوده‌ی وسیعی از ورودی‌ها بود.

آنچه عنوان شد به عنوان پیش‌زمینه‌ای از تحقیقاتی بود که در زمینه پاندول معکوس در سال‌های دور انجام‌شده است، اما آنچه در این پژوهش مورد بررسی قرار می‌گیرد، الگوی خاصی از ربات‌های متحرک است که بر پایه‌ی نظریه‌ی پاندول معکوس دوچرخ شکل گرفته است.

۲-۲- بررسی مقالات و پایان‌نامه‌های پیشین

آن گونه که «گراسر» و همکارانش ادعا می‌کنند [3]، برای اولین بار ایده‌ی طراحی وسیله‌ی نقلیه‌ی دوچرخ در حدود سال ۲۰۰۰ در دانشگاه ایالتی سوئیس مطرح و منجر به ساخت رباتی به نام JOE شد، بعدها مقاله‌ای تحت عنوان پاندول معکوس متحرک در اوایل سال ۲۰۰۲ منتشر شد که به انتشار نتایج این تحقیق پرداخته بود [3]. شکل (۱-۲) که برگرفته از این مقاله است؛ ربات JOE را نمایش می‌دهد.



شکل ۱-۲: ربات JOE

تاکنون تحقیقات گسترده‌ای روی این مدل از ربات‌ها انجام و پایان‌نامه‌ها و مقاله‌های متعددی در این زمینه منتشر شده است. حجم گسترده‌ای از مقالات منتشرشده، به بررسی و تحلیل دینامیکی ربات پرداخته‌اند [5]، [16]، [17] و [18]. دسته‌ای دیگر با در نظر گرفتن حالت کلی معادلات سیستم، به کنترل آن پرداخته‌اند [19]، [20] و دسته‌ی سوم شامل مقالاتی می‌شود که از دینامیک سیستم صرف‌نظر کرده و به کنترل سینماتیک آن پرداخته‌اند [5]، [13]. گراسر و همکارانش در معرفی ربات JOE، ابتدا به مدل‌سازی سیستم پیشنهادی توسط رایانه پرداخته، سپس مدل به دست آمده را حول نقطه کار خطی کرده و پس از تفکیک کردن معادلات از دو کنترل‌کننده‌ی ساده برای کنترل مجزای تعادل و

زاویه چرخش ربات استفاده نموده‌اند. در این تحقیق که مبنای تحقیقات بعدی نیز قرار گرفته است حسگر ژيروسکوپ برای محاسبه سرعت زاویه‌ای پاندول و کدکننده‌های^۱ افزایشی برای تعیین موقعیت چرخ‌ها استفاده شده است که اطلاعات را به یک پردازنده‌ی دیجیتالی ارسال می‌کند و کنترل‌های نهایی توسط این واحد انجام می‌شود.

در سال ۲۰۰۲ ساختار مشابهی به نام «سگ وی^۲» توسط «دن کمن» اختراع شد [4]. این وسیله قادر به حفظ تعادل سرنشینی بود که بر روی آن قرار می‌گرفت و برای این منظور از مجموع ۵ ژيروسکوپ و حسگرهای زاویه و لغزش استفاده کرده بود که هزینه ساخت آن را فوق‌العاده سنگین می‌کرد، علاوه بر این، برای حرکت ربات از سرو^۳ موتورهای قدرتمندی استفاده شده، که کنترل ربات را فوق‌العاده آسان می‌نماید، این وسیله دو سال پس از طرح ایده‌ی اولیه به عنوان یک محصول تجاری وارد بازار شد. شکل (۲-۲) نمونه‌ای از این ربات را نشان می‌دهد.



شکل ۲-۲: ربات Segway

¹ encoder

² Segway

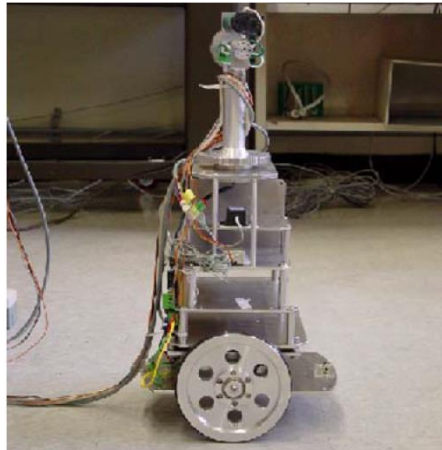
³ Servo Motor

در همین سال «اندرسون» موفق به ساخت ربات مشهوری به نام n-Bot شد که نحوه‌ی کار آن شباهت فراوانی به ربات JOE داشت. این ربات از یک حسگر اینرسی برای حفظ تعادل و از کدکننده‌های افزایشی به منظور مشخص کردن موقعیت موتورها استفاده کرده است.

در پایان سال ۲۰۰۳ پایان‌نامه‌ای تحت عنوان «ربات دوچرخ تعادلی خودکار» در دانشگاه شرقی استرالیا دفاع شد [21]، این پایان‌نامه اگر چه فعالیت خود را بر طراحی یک کالمن فیلتر برای نویز زدایی از اطلاعات واردشده به ربات توسط حسگرها متمرکز کرده بود، اما توانست نمونه‌ای آزمایشگاهی از ربات تعادلی با نام Eyebot ارائه دهد که یکی از کوچک‌ترین ربات‌های تعادلی به شمار می‌رود. در این پایان‌نامه پس از استخراج معادلات دینامیکی سیستم و لحاظ کردن معادلات موتور در آن، خطی سازی حول نقطه‌ی کار، روی معادلات انجام شده و از روش LQR برای کنترل سیستم استفاده شده است. این تحقیق را می‌توان اولین پژوهشی دانست که مدل غیرخطی موتورها را وارد سیستم کرده است.

در سال ۲۰۰۵ «پسک و فرانک»، برای اولین بار کنترل موقعیت و سرعت را در ربات تعادلی دوچرخ مورد بررسی قرار دادند [20]، این مقاله که صرفاً به تحلیل تئوری ساختار ربات تعادلی پرداخته، مدل غیرخطی ربات را استخراج کرده است و با استفاده از روش خطی سازی بازخوردی، راهکار نوینی برای کنترل موقعیت و سرعت ارائه می‌دهد.

«کیم» و همکاران در سال ۲۰۰۶ ربات تعادلی دوچرخ را تحت عنوان یک ربات هوشمند خانگی معرفی کردند و معادلات جامعی برای دینامیک این ربات به دست آوردند [22]. این مقاله از روش کین برای استخراج معادلات دینامیکی ربات استفاده کرده و به مدل سازی ربات در مسیرهای شیب‌دار نیز پرداخته است. این مقاله گرچه دربردارنده‌ی اطلاعاتی از شبیه‌سازی عملی نیز می‌باشد اما این شبیه‌سازی‌ها تنها برای چند مدل حرکتی خاص است و جنبه‌ی عمومی ندارد. ربات استفاده شده برای شبیه‌سازی عملی نتایج این پژوهش در شکل (۲-۳) قابل مشاهده می‌باشد.



شکل ۲-۳: ربات ساخته شده توسط کیم

ساختار ربات تعادلی همانند تمامی ربات‌های متحرکی که دارای دوچرخ محرک هستند به عنوان یک ساختار غیرهولونومیک شناخته می‌شود. (در فصل ۳ به توضیح مختصری در این باره خواهیم پرداخت). وجود قیود غیرهولونومیک، استخراج معادلات دینامیکی این ربات را با مشکلات فراوانی مواجه می‌کند. «سالرنو» و «آنجلس» با استفاده از نظریه‌ی سیستم‌های شبه هولونومیک، به ارائه خانواده‌ی جدیدی از ربات‌های تعادلی پرداخته‌اند [17]. وی در مقاله خود به بیان این مطلب پرداخته که اگر مرکز جرم ربات را در صفحه‌ی میانی چرخ‌ها تصور کنیم، می‌توان مدل‌سازی سیستم را توسط نظریه سیستم‌های شبه هولونومیک و با استفاده از ماتریس هولونومیک انجام داد. به علاوه در خانواده‌ی جدیدی از ربات‌های تعادلی که سالمو معرفی کرده است؛ مرکز جرم پایین‌تر از محور چرخ‌ها قرار می‌گیرد، این مسئله اگر چه تأثیر زیادی روی معادلات سیستم ندارد و به شبه هولونومیک بودن یا نبودن ربات کمکی نمی‌کند؛ اما باعث کمک به پایداری ربات شده و کنترل تعادل ربات را با چالش کمتری مواجه می‌کند.

در سال‌های بعد بارونق گرفتن استفاده از کنترل‌کننده‌های هوشمند و الگوریتم‌های مقاوم، مقالاتی در زمینه‌ی کنترل هوشمند [8]، [13]، [23] و برخط^۱ [24] این دسته از ربات‌ها منتشر شده است. [25]

نیز از کنترل‌کننده‌ی فازی-تطبیقی برای حفظ تعادل و کنترل مسیر حرکت ربات استفاده کرده است، «لی» در مقاله خود [26] کوشیده است تا با نوشتن معادلات در فرم ماتریسی، نشان دهد که ربات

¹ Online

تعادلی را می‌توان یک ربات متحرک فرض کرد که تنها ترم‌های مرتبط با زاویه انحراف بدنه‌ی آن به سیستم اضافه می‌شود و بر این اساس می‌توان سیستم را به یک ربات متحرک و یک پاندول معکوس تبدیل نمود. اگر چه وی تنها به نوشتن فرم کلی معادلات بسنده کرده و با در نظر گرفتن برخی مقادیر نامعلوم برای سیستم به اثبات تطبیقی و مقاوم بودن کنترل‌کننده‌ی فازی پیشنهادشده می‌پردازد، اما فرم معادلات استفاده‌شده در این مقاله به عنوان یک فرم کلی بعدها مورد استفاده سایر مقالات قرار می‌گیرد.

برای اولین بار ایده‌ی استفاده از کنترل‌کننده مد-لغزشی در پ ربات تعادلی، توسط «هوانگ» و همکارانش در سال ۲۰۱۰ میلادی مطرح شد [27]، در این مقاله، ربات تعادلی نه به عنوان یک ربات متحرک بلکه به عنوان فرمی خاص از پاندول معکوس دوچرخ مطرح و برای حرکت ربات یک مسیر خطی در نظر گرفته شده است. هوانگ نشان می‌دهد در صورتی که مدل ربات در سطح صاف، به دست آمده و بر اساس آن به طراحی یک کنترل‌کننده مد لغزشی بپردازیم، اثرات ناشی از قرار گرفتن ربات در شیب کم را می‌توان به صورت محدوده‌ای از تغییرات در ماتریس نیروهای گرانشی و اصطکاک تلقی کرد که در صورت محدود بودن این دسته از تغییرات، ربات توسط کنترل‌کننده مد لغزشی قابل کنترل است.

لی در ادامه‌ی تحقیقات خود در زمینه ربات‌های تعادلی، موفق به پیشنهاد یک کنترل‌کننده‌ی بهینه بر پایه‌ی ماشین‌های بردار پشتیبان^۱ می‌شود [23]. در این پژوهش علاوه بر کنترل موقعیت، تعادل و سرعت؛ کنترل تنظیم^۲ و ردگیری^۳ نیز مورد ارزیابی قرار گرفته و نتایج آن با سایر سیستم‌های بر پایه‌ی مدل^۴ مقایسه می‌گردد.

میرزای و معمارماهر در مقاله‌ای تحت عنوان «رویکردی بر ربات‌های تعادلی دوچرخ مبتنی بر پاندول معکوس» [28]. به بیان ویژگی‌ها و تفاوت‌های میان ربات‌های تعادلی ساخته‌شده پرداخته‌اند.

¹ Support Vector machine (SVM)

² Set-Point

³ Tracking

⁴ Model Base

«معمارباشی» با همکاری «چانگ» در سال ۲۰۱۱ با انتشار مقاله‌ای، نتایج عملی از شبیه‌سازی نظریه‌های کنترلی روی ربات خود را مطرح کردند [29]. در این مقاله از روش خطی سازی نقطه تعادل استفاده شده و ربات از طریق کنترل کننده‌ی بازخورد حالت به وضعیت تعادل در آمده است، این ربات که توسط یک میکروکنترلر ساده و با استفاده از حسگرهای شتاب^۱ و انحراف^۲ کنترل شده است و در ابعاد نسبتاً کوچکی ساخته شده اما تنها کنترل تعادل مورد توجه قرار گرفته است. مسئله قابل توجه در این ربات، استفاده از حسگرها و سخت افزارهای ارزان قیمت برای ساخت ربات است. علاوه بر آن نشان داده شده که با وجود استفاده از قوانین کنترلی ساده جهت پیاده‌سازی روی میکروکنترلر؛ ربات پایداری مناسبی دارد. «جوونگ» و همکاران با انتشار مقاله‌ای در سال ۲۰۱۲ به پیاده‌سازی عملی کنترل ناوبری و تعادل این ربات با بهره گرفتن از ترکیب حسگرهای ارزان قیمت می‌پردازند [30] در این پژوهش، انواع حسگرهای موجود در بازار از نظر قیمت، فاکتورگذاری شده و نشان داده می‌شود که با ترکیب چند حسگر ارزان قیمت می‌توان جوابی مورد قبول و قابل مقایسه با موارد مشابه گران قیمت به دست آورد. در این تحقیق فرم کلی ماتریسی معادلات مورد استفاده قرار گرفته و از یک شبکه RBF برای کنترل مسیر حرکت ربات استفاده شده است. علاوه بر این، مجموعه‌ای از کنترل کننده‌های تفکیک شده‌ی تناسبی مشتقی نیز وظیفه‌ی حفظ تعادل ربات را به عهده‌دارند. ربات مورد در این تحقیق در شکل ۲-۴ قابل مشاهده می‌باشد.

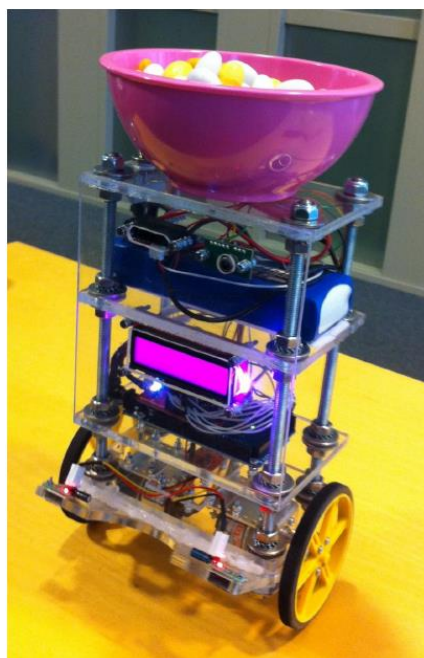


شکل ۲-۴: ربات تعادلی طراحی شده به وسیله لونگ

¹ Accelerometer

² Tilt Sensor

«ساندین» و «تورتسن» در پایان‌نامه‌ی خود به معرفی ربات تعادلی کوچک و جدیدی می‌پردازند که قابلیت حمل اشیاء کوچک را دارد [31]، این ربات که به منظور جابه‌جایی وسایل تزئینی و حمل اجناس روی یک میز نمایش طراحی شده، نمودی جالب از نگاهی جدید به ربات‌های تعادلی است. در استخراج معادلات این ربات، مدل دینامیکی موتورها لحاظ شده است اما مانند بسیاری از مقالات مشابه، در نهایت پس از خطی‌سازی حول نقطه‌ی کار، از کنترل‌کننده‌های خطی برای کنترل تعادل ربات استفاده شده است. این ربات به صورت مستقل دارای یک پردازنده‌ی دیجیتالی سرعت بالاست که اطلاعات حسگرها را گرفته و فرمان‌ها کنترلی را برای ربات فراهم می‌نماید. برای کنترل موتورهای این ربات از روش پهنای پالس استفاده شده و جهت جلوگیری از تأثیرات نویز، یک کالمن فیلتر برای حسگرهای ربات و از یک مشاهده‌گر کالمن در کنترل‌کننده‌ی LQC استفاده شده است. لازم به ذکر است هدف کنترلی این ربات، کنترل نقطه تنظیمی است که توسط یک کنترل‌کننده‌ی بی‌سیم به ربات داده می‌شود. شکل (۲-۵) این ربات را در حال حمل کاسه‌ی سبک وزنی نشان می‌دهد



شکل ۲-۵: ربات چارم‌هنگام حمل یک ظرف سبک

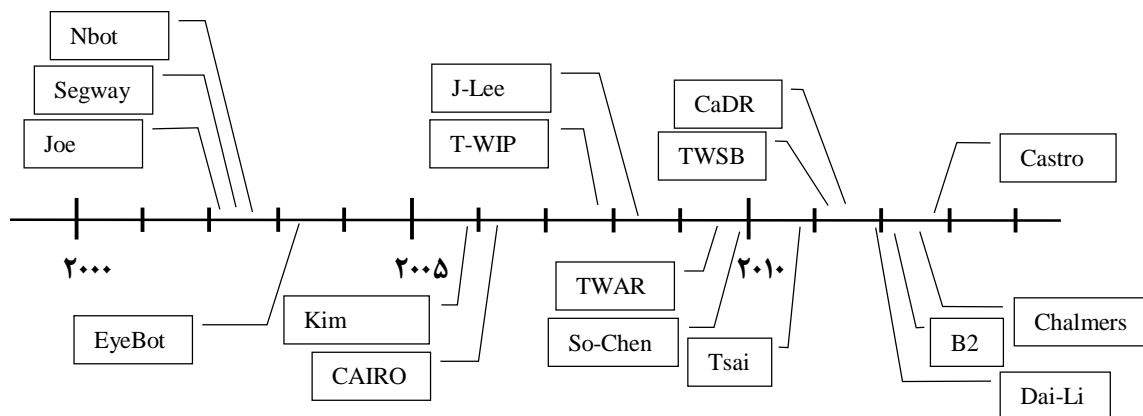
اگر چه ممکن است کنترل تعادل ربات دوچرخ اصلی ترین کار آن تلقی گردد اما اگر بخواهیم به عنوان یک ربات متحرک یا یک وسیله نقلیه متحرک به آن بنگریم لازم است تا بتواند علاوه بر حفظ تعادل به پیمایش مسیری دلخواه نیز پردازد.

در مواردی که ساختار تعادلی تحت عنوان یک وسیله نقلیه معرفی شده، تنها کنترل تعادل ربات مدنظر قرار گرفته، چرا که این جابه جایی راننده است که باعث برهم زدن تعادل وسیله و در نتیجه جابه جایی ربات برای ایجاد یک شتاب منفی و جبران سازی اختلال به وجود آمده می شود. «آیت الله زاده» [12] در پایان نامه ی خود به طور مفصل این قضیه را تحلیل کرده و با مدل سازی وسیله نقلیه دوچرخ در حالت های مختلف: روی سطح صاف، روی سطح شیب دار، بدون لغزش و با در نظر گرفتن لغزش چرخ ها، نمونه ای مناسب برای وسیله های نقلیه دوچرخ ارائه داده است، «نادری» نیز با استفاده از روش لاگرانژ، معادلات خودروی دوچرخ را برای حالات حرکت و سکون استخراج نموده و نهایتاً به کنترل آن پرداخته است.

«لاریمی» پای خود را اندکی فراتر نهاده و با ترکیب ربات تعادلی و یک بازوی دو لینکی، به معرفی یک بازوی متحرک دوچرخ می پردازد که به منظور حفظ تعادل، از یک چرخ دوار کمکی استفاده می کند [32].

۲-۳- جمع بندی و نتیجه گیری

پیشرفت های زیادی در زمینه ی طراحی و مدل سازی و کنترل ربات تعادلی صورت گرفته است که قطعاً نمی توان به پوشش تمامی آن ها پرداخت. در این فصل به بررسی مهم ترین فعالیت های انجام شده در زمینه ربات های تعادلی پرداختیم و جمعاً ۴ پایان نامه و ۱۵ مقاله ی منتشر شده در این زمینه بررسی شد. شکل (۲-۶) نشان دهنده ی نمودار زمانی ربات های تعادلی ساخته شده است. اسامی مندرج در این نمودار بیان کننده ی نام ربات و یا در صورت نداشتن اسم خاص، بیانگر نام سازنده آن می باشد.



شکل ۲-۶: نمودار زمانی ربات های تعادلی ساخته شده از سال ۲۰۰۰ تاکنون

اگرچه روش های هوشمند جدیدی برای کنترل تطبیقی و مقاوم ربات های تعادلی پیشنهاد شده اما اکثر فعالیت های عملی بر پایه ی کنترل های سنتی و مدل خطی شده سیستم صورت گرفته است. به همین منظور در تحقیق جاری، سعی شده علاوه بر استفاده از معادلات غیرخطی سیستم و معرفی نمونه ای جدید با در نظر گرفتن دینامیک موتورها، به پیاده سازی روش های کنترلی هوشمند روی یک ربات تعادلی بپردازیم.

۳- فصل سوم

پیش زمینه

۳-۱- روش مدولاسیون پهنای پالس^۱:

به علت استفاده از باتری به عنوان منبع تغذیه، مقدار ولتاژ همواره ثابت است اما برای کنترل سرعت موتورها، باید ولتاژهای مختلفی تولید و به محرکها اعمال گردد. به همین منظور از روش مدولاسیون پهنای پالس استفاده شده که روشی ساده و دارای توان تلفاتی کم است، از طرفی سخت‌افزار مورد نیاز برای آن نیز در میکروکنترلرهای AVR موجود می‌باشد.

در این روش ولتاژ باتری در حالت کلید زنی و از طریق راه انداز به موتورها انتقال می‌یابد. اما به علت وجود پسماند، ولتاژ موتور در هر بار کلید زنی از صفر به سمت مقدار نهایی خود حرکت نمی‌کند. مقدار ولتاژ اعمال شده به موتور را میتوان میانگین سیگنالی دانست که در بازه‌های زمانی مختلف دوسر موتور وجود دارد و خوشبختانه میتوان نشان داد این مقدار وابسته به سیکل کاری^۲ (D) است. در ادامه به استخراج معادلات ولتاژ می‌پردازیم. سطح ولتاژ خروجی از طریق فرمول (۳-۱) محاسبه می‌گردد.

$$U = \frac{1}{T} \int_0^T f(t) dt \quad (۳-۱)$$

که T بازه زمانی یک دوره تناوب از سیگنال و $f(t)$ بیان‌کننده‌ی تابع زمانی شکل موج سیگنال است. اگر شکل موج PWM را به صورت شکل (۳-۱) در نظر بگیریم. داریم:

$$f(t) = \begin{cases} U_{\max} & 0 < t \leq D \times T \\ y_{\min} & D \times T < t \leq T \end{cases} \quad (۳-۲)$$

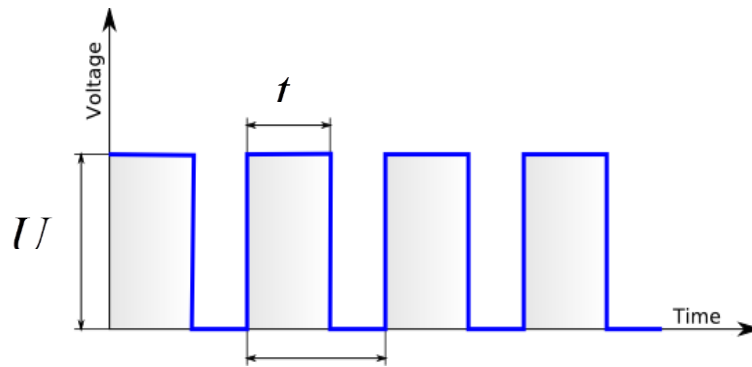
که در این معادله U_{\max} و y_{\min} به ترتیب بیان‌کننده‌ی ولتاژ باتری و ولتاژ پسماند، T کل عرض یک پالس و $D = \frac{t_{on}}{T}$ ، سیکل کاری یک دوره تناوب زمانی را بیان می‌کند.

بنابراین با جایگذاری (۳-۲) در (۳-۱) می‌توان معادله ولتاژ میانگین را به صورت زیر نوشت.

$$U = \frac{1}{T} \left(\int_0^{DT} U_{\max} dt + \int_{DT}^T y_{\min} dt \right) = \frac{DTU_{\max} + T(1-D)y_{\min}}{T} = DU_{\max} + (1-D)y_{\min} \quad (۳-۳)$$

¹ Pulse width Modulation (PWM)

² Duty-Cycle



شکل ۱-۳: شکل موج یک سیگنال PWM

با ناچیز فرض کردن ولتاژ پسماند، رابطه‌ی (۳-۳) را می‌توان به صورت (۴-۳) بازنویسی کرد

$$U = D \times U_{\max} \quad (4-3)$$

بر این اساس می‌توان نتیجه گرفت که با کنترل محدوده‌ی کاری پالس، می‌توان ولتاژهای مختلفی تولید کرد که با اعمال به موتورها باعث به وجود آمدن گشتاورهای مختلف می‌گردد.

۳-۲- روش کنترل مدل لغزشی^۱:

۳-۲-۱- مقدمه

عدم قطعیت و وجود دینامیک‌های مدل نشده در کنترل سیستم‌های غیرخطی می‌تواند باعث بروز خطای زیادی گردد. یکی از روش‌های مهم برای مبارزه با عدم قطعیت‌ها استفاده از روش‌های کنترل مقاوم^۲ می‌باشد.

مد لغزشی به عنوان یکی از روش‌های مقاوم و استاندارد برای مقابله با عدم قطعیت‌های پارامتریک و اغتشاشات خارجی در سیستم‌های غیرخطی معرفی می‌شود. که در اینجا با توجه به ساختار غیرخطی ربات و وجود دینامیک‌های مدل نشده، مورد استفاده قرار گرفته است.

مد لغزشی، سیستم‌های مرتبه بالا را به یک سیستم مرتبه اول تبدیل می‌کند. از این رو می‌توان با اعمال قوانین ساده، به کنترل سیستم پرداخت و این کنترل از مقاومت بالایی نیز برخوردار خواهد بود.

¹ Sliding-Mode

² Robust Control

ساختار اصلی کنترل مقاوم به این صورت است که به کنترل قسمت اصلی سیستم می‌پردازد (همانند قانون کنترل بازخوردی) و در کنار آن به کاهش اثر نامعینی‌های در نظر گرفته‌شده می‌پردازد. مد لغزشی یکی از مهم‌ترین روش‌های کنترل مقاوم به شمار می‌رود. این روش یک رویکرد مشخصی برای حفظ پایداری و رسیدن به رفتار مطلوب دارد که بر اساس مدل سیستم طراحی می‌شود. بدین صورت که با برقراری یک رابطه‌ی دو سویه میان مدل و عملکرد سیستم، تمام مراحل طراحی مشخص می‌شود.

۲-۲-۳- سطح لغزش^۱

در روش مد لغزشی از یک سوئیچ سرعت بالا در مسیر بازخورد استفاده می‌شود. برای مثال فرض کنید که ضریب تقویت هر یک از مسیرهای بازخورد، بر حسب قانونی که به حالت‌های سیستم وابسته است، بین دو مقدار سریعاً تغییر نماید.

انجام کلید زنی باعث می‌شود تا متغیرهای حالت سیستم غیرخطی، یک سطح خاص (که توسط ما تعیین می‌شود) را دنبال کنند و پس از برخورد، بر روی این سطح باقی بمانند. برای این منظور، زمانی که متغیرهای حالت سیستم، فراتر از سطح مطلوب قرار دارند، برای مسیر بازخورد یک ضریب انتخاب‌شده و به وضع مشابهی برای حالتی که متغیرهای حالت سیستم پایین تر از سطح باشند مقدار دیگری انتخاب می‌شود. این امر باعث هدایت سیستم به سمت سطح مطلوب شده و تضمین می‌کند که برای زمان‌های بعدی بر روی آن باقی بماند. از این رو به آن سطح لغزش یا صفحه‌ی کلیدزنی گفته می‌شود.

می‌توان گفت مسئله‌ی اصلی، طراحی کنترل کلید زنی به گونه‌ای است که بتواند متغیرهای حالت سیستم را به سمت سطح لغزش هدایت نموده و در سطح لغزش نگه دارد. برای این منظور از روش لیاپانوف استفاده می‌شود [14].

¹ Sliding surface

۳-۲-۳- طراحی کنترل کننده

طراحی کنترل کننده مد لغزشی شامل دو مرحله عمده است. اول: استفاده از قانون کنترل بازخورد حالت به منظور شناسایی شرایط لغزش است در این مرحله برای لحاظ کردن دینامیک‌های مدل نشده و اغتشاشات نیاز است تا قانون کنترل به صورت گسسته بیان شود. طراحی کنترل کننده در این مرحله بر مبنای روش لیاپانوف صورت می‌گیرد. مرحله دوم حل کردن پدیده‌ی وزوز^۱ است که مخصوصاً برای پیاده‌سازی‌های عملی غیرقابل اغماض می‌باشد. در ادامه به بیان هر یک از این مراحل خواهیم پرداخت. اگر سیستم مرتبه دوم را به فرم زیر در نظر بگیرید

$$\ddot{x}(t) = f(x, t) + u(t) \quad (۵-۳)$$

که $f(x, t)$ معمولاً غیرخطی و تابعی متغیر با زمان است، با وجود عدم قطعیت‌ها در سیستم، مقدار این تابع با $\tilde{f}(x, t)$ تخمین زده می‌شود. ورودی‌های کنترلی و $x(t)$ متغیرهای حالتی هستند که باید کنترل شوند و به سمت $x_d(t)$ هدایت شوند. فرض می‌شود که خطای تخمین $f(x, t)$ مقداری محدود و معین است که با $F = F(x, t)$ بیان می‌گردد.

$$|\tilde{f}(x, t) - f(x, t)| \leq F(x, t) \quad (۶-۳)$$

با تعریف خطای اندازه‌گیری، حال لازم است تا متغیر لغزش تعریف گردد. بر این اساس داریم:

$$s(t) = \left(\frac{d}{dx} + \gamma\right)\tilde{x}(t) = \dot{\tilde{x}}(t) + \gamma\tilde{x}(t) \quad (۷-۳)$$

که $\tilde{x}(t) = x - x_d$ و γ پهنای باند سیستم را در مقابله با عدم قطعیت‌ها بیان می‌کند.

مشتق متغیر لغزش را به دست می‌آوریم.

$$\dot{s}(t) = \ddot{\tilde{x}}(t) - \ddot{x}_d(t) + \gamma\dot{\tilde{x}}(t) \quad (۸-۳)$$

حال معادله (۵-۳) را در (۸-۳) جایگذاری می‌کنیم که بر این اساس خواهیم داشت

$$\dot{s}(t) = f(x, t) + u(t) - \ddot{x}_d(t) + \gamma\dot{\tilde{x}}(t) \quad (۹-۳)$$

¹ Chattering

از این رو قانون کنترلی $\tilde{u}(t)$ برای $\dot{s}(t) = 0$ به صورت زیر پیشنهاد می‌شود که می‌تواند به عنوان بهترین تخمین از کنترل کننده سیستم معادل، تفسیر شود.

$$\tilde{u}(t) = -\hat{f}(x, t) + \ddot{x}_d(t) - \gamma \dot{\tilde{x}}(t) \quad (10-3)$$

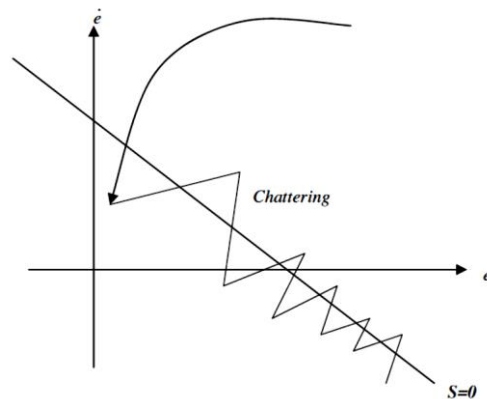
برای مواجهه با عدم قطعیت‌های موجود در f که لازمه‌ی رسیدن به شرایط مطلوب لغزشی است. باید

$$\frac{1}{2} \frac{d}{dt} (s(t)^2) \leq -\eta |s(t)|$$

باشد و قانون کنترل نهایی به صورت زیر تعریف می‌شود.

$$u(t) = \tilde{u}(t) - k(x, t) \operatorname{sgn}(s(t)) \quad (11-3)$$

که لازم است $k(x, t)$ به میزان کافی بزرگ در نظر گرفته شود به عنوان مثال: $k(x, t) = F(x, t) + \eta$. بر این اساس می‌توان مطمئن بود که با استفاده از قانون کنترلی (۱۱-۳)، متغیرهای حالت سیستم در زمانی محدود، به سمت صفحه $s(t) = 0$ میل می‌کنند. که در این صورت، خطای سیستم بعد از مدت زمانی محدود به سمت صفر میل خواهد کرد.



شکل ۲-۳: پدیده‌ی وزوز

از اینجا قسمت دوم کار شروع می‌شود، به علت وجود تابع $\operatorname{sgn}(\cdot)$ شاهد نوسانات فرکانس بالایی حول سطح لغزش خواهیم بود که به پدیده‌ی وزوز موسوم است. این پدیده در شکل (۲-۳) به نمایش درآمده است.

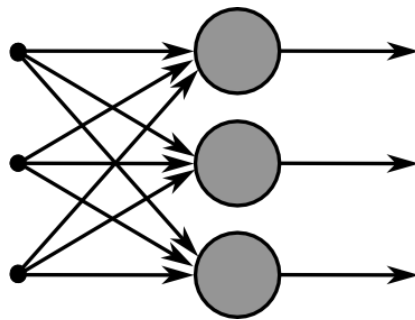
یکی از روش‌های معمول و کارآمد برای بهبود این نوسانات فرکانس بالا، استفاده از تابع $\operatorname{sat}(\cdot)$ به جای تابع $\operatorname{sgn}(\cdot)$ است که در این پژوهش نیز از آن استفاده شده است.

۳-۳- شبکه‌های عصبی فازی:

در سال‌های اخیر استفاده از شبکه‌های عصبی به عنوان یک کنترل کننده رونق فراوانی یافته است. و این به خاطر مزایای فراوانی است که این شبکه‌ها در خود جای داده‌اند. از مهم‌ترین آنان می‌توان به قابلیت یادگیری و همچنین امکان تخمین توابع پیچیده‌ی غیرخطی اشاره کرد. به طور کلی شبکه عصبی را می‌توان مجموعه‌ای از گره‌ها دانست که در لایه‌های مختلف قرار گرفته و توسط یک سری وزن به هم متصل می‌شوند. نحوه‌ی قرارگیری این لایه‌ها، نوع ارتباط سلول‌ها و همچنین روش‌های موجود برای به‌روزرسانی^۱ وزن‌ها، باعث ایجاد انواع مختلفی از شبکه عصبی می‌شوند که در ادامه به معرفی برخی از آنان خواهیم پرداخت

۳-۳-۱- شبکه‌های عصبی پیش‌خور^۲:

در این دسته از شبکه‌های عصبی، خروجی هر لایه، ورودی لایه‌ی بعد را می‌سازد و هیچ‌گونه بازگشتی در سیستم انجام نمی‌شود. برای اصلاح وزن‌ها در این شبکه، معمولاً از روش پس انتشار خطا^۳ استفاده می‌شود. مشکل عمده‌ی این شبکه‌ها را می‌توان برهم خوردن وزن‌ها، به علت آموزش بیش از اندازه^۴ دانست. نمای کلی یک شبکه پیش‌خور در شکل (۳-۳) نشان داده شده است.



شکل ۳-۳: نمای کلی شبکه عصبی پیش‌خور

¹ Update

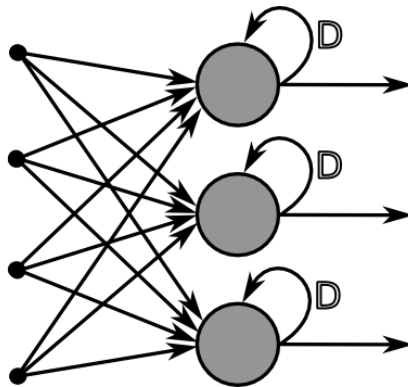
² Feed forward

³ Back propagation

⁴ Over training

۳-۳-۲- شبکه‌های عصبی بازگشتی^۱:

اگر خروجی لحظه قبلی سلول‌ها به عنوان یکی از ورودی‌های فعلی شبکه عصبی در نظر گرفته شود به شبکه عصبی حاصل، بازگشتی گفته می‌شود، این شبکه‌ها کاربرد فراوانی به عنوان تخمین گر در سیستم‌های کنترلی دارند. شکل (۳-۴) یک شبکه عصبی بازگشتی را نشان می‌دهد.



شکل ۳-۴: نمایی از یک شبکه عصبی بازگشتی

۳-۴- منطق فازی

اگر بخواهیم به روش سنتی تعدادی داده را دسته بندی کنیم، دسته های مشخصی داریم که هر یک از داده های ما به صورت قطعی در یکی از این دسته ها قرار می گیرد به عبارت دیگر می توان گفت: دسته بندی به صورت بله و خیر بوده و حالت دیگری برای آن وجود ندارد. اگر بخواهیم شاخص عضویت برای هر مجموعه مشخص کنیم، دسته بندی سنتی تنها دارای شاخص های عضویت ۰ (عدم عضویت در مجموعه) و ۱ (عضویت در مجموعه) می باشد. به این مفهوم تحت عنوان تابع عضویت نگاه خواهیم کرد و می گوئیم مقدار تابع عضویت برای عضویت در مجموعه، ۱ و در بقیه حالات ۰ است و این تابع هیچ مقدار سومی نخواهد داشت. اما در دنیای واقعی دسته بندی ها به صورت دیگری انجام می شود. فرض کنید می خواهیم مجموعه ی هوای سرد و گرم را از هم جدا کنیم، در چنین مواردی مرز مشخصی برای تصمیم گیری وجود ندارد و نمی توان به صورت قطعی مشخص کرد تا هوای مثلا بیشتر از ۲۵

¹ recurrent

درجه، گرم و کمتر از آن سرد تلقی شود، آنچه مشخص است وجود عدم قطعیت در این نوع طبقه بندی هاست و تفاوت اصلی میان خوشه بندی سنتی و فازی نیز از همین جا نشات میگیرد. در خوشه بندی فازی تابع عضویت میتواند مقادیری بین ۰ و ۱ را نیز داشته باشد.

۳-۴-۱- انواع شبکه عصبی-فازی^۱ و نورو-فازی^۲:

پیش از این به توضیح منطق فازی و توابع عضویت در کلاسه بندی فازی پرداختیم. چگونگی استفاده از این خاصیت در شبکه های عصبی و همچنین نوع ارتباط میان این دو مفهوم، سیستم های مختلفی را پدید می آورد که عموماً نورو فازی خوانده می شود اما حقیقت امر این است که "اختصار نورو فازی به سیستم ترکیبی حاصل از شبکه عصبی و سیستم استنتاجی فازی گفته شده که در آن شبکه عصبی به عنوان تعیین کننده ی پارامترهای سیستم فازی مورد استفاده قرار می گیرد". به این معنا که از شبکه عصبی برای یافتن پارامترهای بهینه ی توابع عضویت و یا انتخاب قوانین فازی استفاده می شود. اما این تنها ارتباط میان شبکه های عصبی و ساختارهای فازی نیست. "در مقابل نورو فازی، شبکه عصبی فازی قرار دارد که در آن از منطق فازی برای بهبود عملکرد شبکه عصبی استفاده می شود. در این شبکه، منطق فازی فرع بوده و تنها برای بهبود شرایط شبکه عصبی و یا اضافه نمودن مفهوم عدم قطعیت به شبکه مورد استفاده قرار می گیرد".

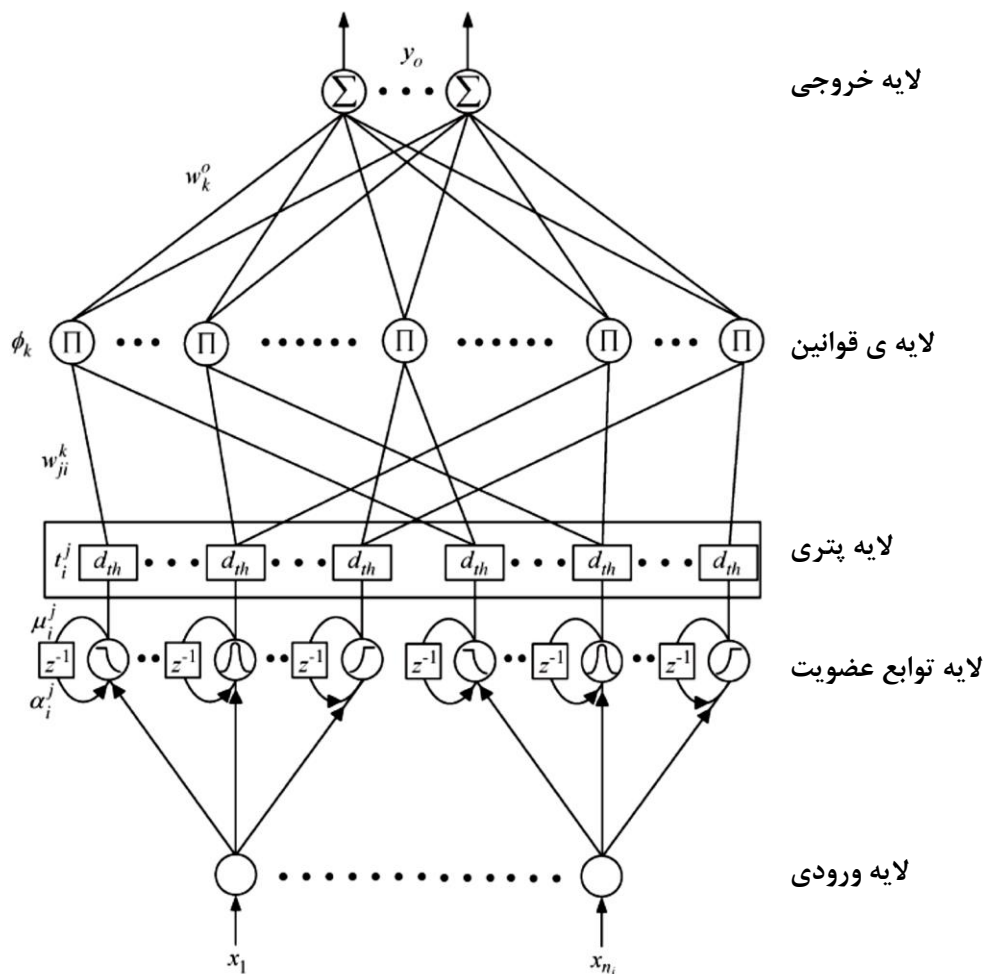
شبکه ی عصبی استفاده شده در این پژوهش اگر چه از نوع عصبی- فازی است اما می توان آن را تلفیقی از دو عبارت بالا دانست چرا که از منطق فازی فروع بوده و تنها به منظور وارد کردن مفهوم عدم قطعیت استفاده شده است اما در نهایت نوع شبکه به گونه ای است که در طول مراحل یادگیری به اصلاح پارامترهای توابع عضویت می پردازد. در ادامه توضیح جامع تری در این باره خواهیم داد.

¹ Neural-fuzzy network or fuzzy-neural network

² Neuro-Fuzzy

۳-۴-۲- شبکه عصبی فازی استفاده شده در این پژوهش:

در این پژوهش از یک شبکه‌ی عصبی بازگشتی استفاده شده که ایده کلی آن توسط [33] برای کنترل ربات متحرک مطرح شده است. شمای کلی این شبکه در شکل (۳-۵) قابل مشاهده است. همان طور که عنوان شد، علت عصبی فازی نامیده شدن این شبکه در این است که در لایه‌ی دوم، از توابع عضویت فازی استفاده شده است و در مراحل یادگیری پارامترهای آن توسط شبکه عصبی تنظیم می‌گردد.



شکل ۳-۵: شبکه عصبی فازی استفاده شده در این پژوهش [33]

شبکه فوق مجموعاً از ۵ لایه تشکیل شده است که وظیفه‌ی هر لایه به شرح زیر بیان می‌شود

۱- لایه اول: لایه‌ی ورودی‌هاست که از i سلول تشکیل شده و عیناً مقادیر هر یک از ورودی‌ها را

به لایه ۲ انتقال می‌دهد.

۲- لایه دوم: این لایه از تعداد z سلول تشکیل شده و دربردارنده‌ی قوانین عضویت فازی است. ورودی‌های این سلول را x_i و خروجی لحظه‌ی قبل تشکیل می‌دهد. بنابراین، ورودی توابع عضویت فازی را می‌توان به صورت زیر تعریف کرد.

$$r_i^j(n) = x_i(n) + \mu_i^j(n-1)\alpha_i^j \quad (12-3)$$

که در آن n بیان‌کننده‌ی شماره‌ی مرحله، α_i^j وزن حلقه‌ی بازخورد به خود^۱ و $\mu_i^j(n-1)$ خروجی لحظه‌ی قبل لایه‌ی ۲ است که توسط تابع عضویت گوسی به صورت زیر تعریف می‌شود.

$$net_j = -\frac{(r_i^j - m_i^j)^2}{(s_i^j)^2}$$

$$\mu_i^j [net_j(r_i^j(n))] = \exp(net_j(r_i^j(n))) \quad (13-3)$$

که m_i^j, s_i^j به ترتیب و مستقلاً برای $(i = 1, \dots, n_i; j = 1, \dots, n_j)$ بیان‌کننده‌ی انحراف معیار و میانگین توابع گوسی هستند.

۳- لایه سوم: لایه‌ی پتری است که در این شبکه به منظور تولید نشانه برای نشان دادن گره‌های فعال به کار می‌رود. این کار توسط قانون زیر صورت می‌گیرد.

$$t_i^j = \begin{cases} 1, & \mu_i^j \geq d_{th} \\ 0, & \mu_i^j < d_{th} \end{cases} \quad (14-3)$$

در اینجا بیان‌کننده انتقال و d_{th} آستانه‌ی متغیری است و که با توجه به خطا تغییر می‌کند و در آینده به توصیف آن خواهیم پرداخت.

۴- لایه‌ی چهارم: لایه‌ی قوانین است و قانون هر گره آن توسط \prod توصیف شده که به معنای ضرب تمامی سیگنال‌های ورودی و انتقال نتیجه‌ی حاصل به خروجی است

$$\phi_k = \begin{cases} \prod_{i=1}^{n_i} w_{ji}^k \mu_j^i, & t_i^j = 1 \\ 0, & t_i^j = 0 \end{cases} \quad (15-3)$$

¹ Self feed-back

که ϕ_k ($k = 1, \dots, n_y$) بیان کننده خروجی لایه‌ی قوانین می‌باشد و w_{ji}^k وزن میان لایه‌ی پتری و لایه‌ی قوانین را مشخص می‌کند. لازم به ذکر است که در پژوهش فوق این وزن واحد در نظر گرفته شده است. n_y نیز نشان دهنده‌ی تعداد نهایی قوانین است.

۵- لایه‌ی پنجم: آخرین لایه و لایه‌ی خروجی است که به وسیله \sum نمایش یافته است و این بدین معناست که مقدار خروجی‌ها، متناسب با مجموعی از ورودی‌ها است که به صورت زیر تعریف می‌شود.

$$y_o = \sum_{k=1}^{n_j} w_k^o \phi_k \quad (۱۶-۳)$$

مسئله اصلی در اصلاح خطای این شبکه، چگونگی تعیین بردار گرادیان از طریق روش بازگشتی است. هر یک از قسمت‌های الگوریتم آموزش را می‌توان به عنوان مشتقی از تابع خطایی دانست که بر اساس پارامترهای سیستم و طبق قانون زنجیره‌ای به دست می‌آید. بر این اساس بردار گرادیان، در مسیری خلاف مسیر حرکت خروجی‌های شبکه محاسبه خواهد شد. این روش به عنوان پس انتشار خطا شناخته می‌شود.

برای معرفی الگوریتم برخط آموزش شبکه با استفاده از روش گرادیان نزولی حمایت شده^۱، تابع انرژی سیستم به صورت

$$E = \frac{1}{2} \sum ((x_{d_i} - x_i)^2 + (\dot{x}_{d_i} - \dot{x}_i)^2) = \frac{1}{2} \sum (e_i^2 + \dot{e}_i^2) \quad (۱۷-۳)$$

معرفی می‌گردد. ذکر این نکته لازم است که مقدار d_{th} که در معادله (۱۴-۳) مطرح شده بود، با کمک گرفتن از (۱۷-۳) و به صورت زیر محاسبه می‌شود.

$$d_{th} = \frac{\alpha \exp(-\beta E)}{1 + \exp(-\beta E)} \quad (۱۸-۳)$$

¹ Supervised gradient-descent

که α و β ضرایب ثابت هستند. این رابطه بیان می‌کند که با افزایش خطای خروجی، آستانه‌ی طراحی شده برای تشخیص خطا در لایه پتری کاهش می‌یابد؛ تا فعالیت کنترلی بیشتری برای رسیدن به خروجی مطلوب صورت گیرد.

در صورتی که ورودی‌های شبکه عصبی فوق را سیگنال‌های خطا و مشتقات آن در نظر بگیریم و تعداد خروجی‌ها محدود به اندازه‌ی n_o باشد، اصلاح وزن بر اساس روش پس انتشار با استفاده از مراحل زیر صورت می‌گیرد.

انتشار خطا در لایه‌ی خروجی را می‌توان به صورت

$$\delta_o = \sum \left(e_x \frac{\delta x}{\delta y} + e_x \frac{\delta \dot{x}}{\delta y} \right) \quad (19-3)$$

دانست که مقادیر $\frac{\delta x}{\delta y}$ و $\frac{\delta \dot{x}}{\delta y}$ از طریق زیر تخمین زده می‌شوند

$$\begin{aligned} \frac{\delta x}{\delta y} &= \text{sgn} \left(\frac{x(n) - x(n-1)}{y_o(n) - y_o(n-1)} \right) \\ \frac{\delta \dot{x}}{\delta y} &= \text{sgn} \left(\frac{\dot{x}(n) - \dot{x}(n-1)}{y_o(n) - y_o(n-1)} \right) \end{aligned} \quad (20-3)$$

با این اوصاف اصلاح وزن‌های لایه‌ی خروجی از طریق معادله ۲۱-۳ صورت می‌گیرد

$$\begin{aligned} \Delta w_k^o &= \eta_w \delta_o \phi_k \\ w_k^o(n+1) &= w_k^o(n) + \Delta w_k^o(n) \end{aligned} \quad (21-3)$$

با توجه به واحد در نظر گرفتن وزن‌های لایه‌ی قوانین، میزان انتشار خطا در این لایه به صورت:

$$\zeta_k = \begin{cases} \delta_o w_k^o, & \phi_k \neq 0 \\ 0, & \phi_k = 0 \end{cases} \quad (22-3)$$

محاسبه می‌گردد که با توجه به خروجی‌های شبکه قوانین و همچنین معادله (۲۲-۳)؛ المان خطا در لایه‌ی پتری به دست می‌آید.

$$\rho_j = \begin{cases} \sum_k \zeta_k \phi_k, & t_i^j = 0 \\ 0, & t_i^j = 0 \end{cases} \quad (24-3)$$

بنابراین قانون اصلاح وزن برای مقادیر m_i^j ، s_i^j و α_i^j را می‌توان به شرح زیر عنوان کرد.

$$\Delta m_i^j = \eta_m \rho_j \frac{2(r_i^j - m_i^j)}{(s_i^j)^2} \quad (25-3)$$

$$m_i^j(n+1) = m_i^j(n) + \Delta m_i^j(n)$$

$$\Delta s_i^j = \eta_s \rho_j \frac{2(r_i^j - m_i^j)^2}{(s_i^j)^3} \quad (26-3)$$

$$s_i^j(n+1) = s_i^j(n) + \Delta s_i^j(n)$$

$$\Delta \alpha_i^j = -\eta_\alpha \rho_j \frac{2(r_i^j - m_i^j)^2}{(s_i^j)^3} \mu_i^j(n-1) \quad (26-3)$$

$$\alpha_i^j(n+1) = \alpha_i^j(n) + \Delta \alpha_i^j(n)$$

که مقادیر η_α ، η_m و η_s به ترتیب ضریب حلقه بازخورد داخلی، نرخ آموزش میانه‌ها و نرخ آموزش

انحراف معیار توابع گوسی می‌باشد. که انتخاب شایسته‌ی آنان تأثیر فراوانی بر رفتار شبکه دارد.

الگوهای متفاوتی برای انتخاب این پارامترها عنوان شده. اما با توجه به [33]، نحوه انتخاب این مقادیر

به صورت زیر پیشنهاد می‌گردد

$$\eta_w = \frac{E(n)}{4 \left[\sum_{o=1}^{n_o} \sum_{k=1}^{n_y} \left(\frac{\partial E(n)}{\partial y_o} \frac{\partial y_o}{\partial w_k^o} \right)^2 + \varepsilon \right]}$$

$$\eta_m = \frac{E(n)}{4 \left[\sum_{i=1}^{n_i} \sum_{j=1}^{n_j} \sum_{o=1}^{n_o} \left(\frac{\partial E(n)}{\partial x_i} \frac{\partial x_i}{\partial y_o} \frac{\partial y_o}{\partial m_i^j} \right)^2 + \varepsilon \right]} \quad (27-3)$$

$$\eta_s = \frac{E(n)}{4 \left[\sum_{i=1}^{n_i} \sum_{j=1}^{n_j} \sum_{o=1}^{n_o} \left(\frac{\partial E(n)}{\partial x_i} \frac{\partial x_i}{\partial y_o} \frac{\partial y_o}{\partial s_i^j} \right)^2 + \varepsilon \right]}$$

۳-۵- جمع‌بندی:

در این فصل ابتدا به بیان روش PWM جهت کنترل موتورهای پرداخته شد. این روش به علت ثابت بودن ولتاژ تغذیه ربات و نیاز به سطوح ولتاژ مختلف برای کنترل موتورها، استفاده شده است. پس از آن به بیان چگونگی طراحی یک کنترل‌کننده مد لغزشی اشاره شد، روش مد لغزشی جزو اصلی‌ترین روش‌های کنترل مقاوم غیرخطی به شمار می‌رود که می‌تواند بر محدوده‌ی تعریف‌شده‌ای از عدم قطعیت‌ها غلبه کرده و سیستم را به شرایط پایدار برساند. در ادامه نیز مقدمه‌ای بر شبکه‌های عصبی و منطق فازی بیان شد و نهایتاً به مراحل طراحی یک شبکه عصبی فازی برای کنترل ربات اشاره شد.

۴- فصل چهارم:

مدل سازی سینمایی و دینامیکی ربات

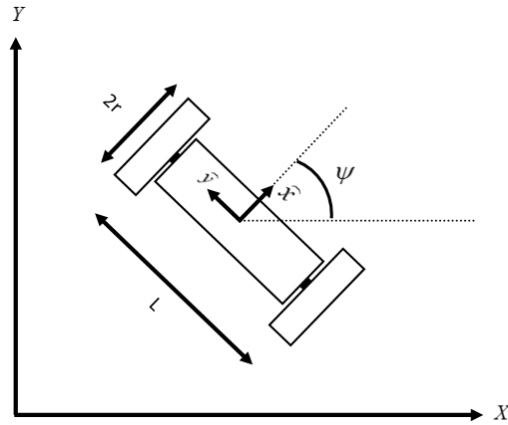
۴-۱- مقدمه:

سینماتیک علمی است که به بررسی حرکت ربات بدون در نظر گرفتن نیروهای مولد آن می‌پردازد. بنابراین در سینماتیک، حرکت بیشتر جنبه‌ی هندسی دارد. یکی از مشکلات مدل‌سازی سینماتیکی ربات متحرک دوچرخ و به طور کلی ربات‌های متحرک، وجود قیود غیرهولونومیک است. هر سیستم مکانیکی که ساختارش توسط مختصات تعمیم‌یافته توصیف شود؛ می‌تواند تعدادی قیود سینماتیکی داشته باشد. این قیود به صورت رابطه‌ای میان مختصات تعمیم‌یافته توصیف می‌شوند. اگر قیدها انتگرال پذیر باشند، آن‌ها را هولونومیک می‌نامند و می‌توان با حذف مختصات اضافی، ساختار سیستم را با تعداد مختصات کمتری بیان نمود [34]. معادلات جدید به دست آمده در این روش، قیدهای حذف‌شده را نیز در دل خود جای داده‌اند. متأسفانه در مورد قیدهای غیرهولونومیک، حذف مختصات اضافی امکان‌پذیر نیست و قیدها در کنار معادلات دینامیکی سیستم باید به صورت ضمنی مدنظر قرار گرفته شود.

سینماتیک مستقیم یک ربات، در واقع توصیف‌کننده موقعیت و سرعت نهایی ربات بر اساس مختصات تعمیم‌یافته و مشتقات آن است. بر همین اساس معادلات سینماتیکی ربات تعادلی دوچرخ متحرک شباهت زیادی به معادلات سینماتیکی یک ربات متحرک ساده دارد که در ادامه به استخراج این معادلات خواهیم پرداخت. از طرفی، به علت لزوم پیاده‌سازی عملی، معادلات موقعیت زمان گسسته‌ی ربات نیز لازم است که در ادامه‌ی مدل‌سازی سینماتیکی به این بخش نیز پرداخته می‌شود.

۲-۴- استخراج معادلات سینماتیکی ربات:

تفاوت اصلی میان ربات تعادلی و ربات‌های متحرک معمولی در محل قرارگیری مرکز جرم ربات است. بر خلاف ربات‌های متحرک دیگر، مرکز جرم ربات تعادلی در صفحه‌ی میانی دوچرخ قرار گرفته است. شمای سینماتیک یک ربات تعادلی در شکل (۱-۴) نشان داده شده است.



شکل ۱-۴: شمای سینماتیک ربات تعادلی دوچرخ

این ساختار از دو چرخ محرک تشکیل شده که به صورت جداگانه توسط موتورهای کنترل می‌شوند. برای کنترل موقعیت و زاویه‌ی ربات می‌بایست به کنترل مستقل هر یک از چرخ‌ها پرداخت. علامت‌های اختصاری استفاده شده در مدل‌سازی سینماتیکی که بعدها در مدل دینامیکی ربات نیز مورد استفاده قرار می‌گیرد در جدول (۱-۴) بیان شده است.

جدول ۱-۴: علامت‌های اختصاری استفاده شده در مدل‌سازی سینماتیکی

l	فاصله بین دو چرخ
r	شعاع چرخ‌ها
ψ	جابه‌جایی دورانی ربات حول محور Z
v	سرعت خطی ربات در راستای \hat{x}
ω	سرعت دورانی ربات حول Z
θ_L, θ_R	جابه‌جایی دورانی چرخ‌های راست و چپ
\hat{x}, \hat{y}	محورهای مختصات متصل به ربات

استخراج معادلات سینماتیکی را با دو فرض آغاز می‌کنیم.

۱- ربات روی یک سطح صاف حرکت می‌کند.

۲- چرخ‌ها دارای تماس کامل با سطح هستند و غلطش کامل دارند.

با توجه به عدم لغزش چرخ‌ها، ربات تعدادی قیود سینماتیکی دارد که میتواند به فرم کلی (۴-۱) نوشته شود.

$$\dot{Y} \cos(\psi) - \dot{X} \sin(\psi) = 0 \quad (۴-۱)$$

با توجه به شکل (۴-۱)، معادله سرعت‌های خطی و دورانی ربات به صورت زیر است.

$$\dot{X} = v \cos \psi \quad (۴-۲)$$

$$\dot{Y} = v \sin \psi \quad (۴-۳)$$

$$\dot{\psi} = \omega \quad (۴-۴)$$

اما سرعت خطی و دورانی به دست آمده، وابسته به سرعت چرخ‌های راست و چپ است. بر این اساس و با در نظر گرفتن مدل ربات داریم:

$$v = \frac{r}{2}(\dot{\theta}_R + \dot{\theta}_L) \quad (۴-۵)$$

$$\omega = \frac{r}{L}(\dot{\theta}_R - \dot{\theta}_L) \quad (۴-۶)$$

با جایگذاری معادلات (۴-۵) و (۴-۶) در معادلات (۴-۲)، (۴-۳) و (۴-۴)، رابطه‌ی ژاکوبین دوبرعی ربات برای حالتی که چرخ‌ها در مرکز ربات قرار گرفته‌اند به صورت زیر حاصل می‌شود.

$$\dot{X} = \frac{r}{2} \cos(\psi) \dot{\theta}_R + \frac{r}{2} \cos(\psi) \dot{\theta}_L \quad (۴-۷)$$

$$\dot{Y} = \frac{r}{2} \sin(\psi) \dot{\theta}_R + \frac{r}{2} \sin(\psi) \dot{\theta}_L$$

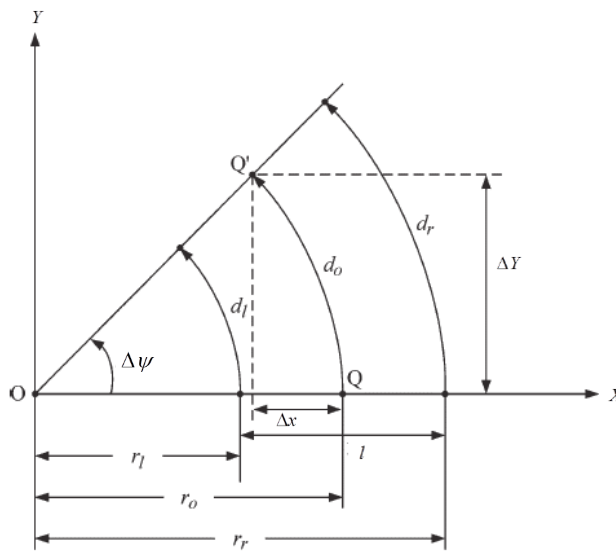
$$\dot{\psi} = \frac{r}{l} \dot{\theta}_R - \frac{r}{l} \dot{\theta}_L$$

این معادله را می‌توان به فرم ماتریسی و کلی (۴-۸) نوشت.

$$\begin{bmatrix} \dot{X} \\ \dot{Y} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} \frac{r}{2} \cos \psi & \frac{r}{2} \cos \psi \\ \frac{r}{2} \sin \psi & \frac{r}{2} \sin \psi \\ l & -l \end{bmatrix} \begin{bmatrix} \dot{\theta}_R \\ \dot{\theta}_L \end{bmatrix} \quad (۸-۴)$$

۳-۴- روابط موقعیت زمان گسسته‌ی ربات:

به علت استفاده از شبکه‌ی عصبی- فازی برای کنترل ربات و به منظور کنترل ردگیری ربات لازم است تا معادلات زمان گسسته‌ی حرکت، استخراج شود تا بتوان از طریق این معادلات به طراحی مسیر دلخواه و اعمال آن به کنترل کننده پرداخت. از این رو برای توصیف موقعیت ربات از شکل (۲-۴) استفاده می‌شود.



شکل ۲-۴: نمایش وضعیت جابه‌جایی ربات [33]

در صورتی که مکان فعلی و مکان بعدی ربات در نظر گرفته شود هر یک از چرخ‌های راست و چپ به ترتیب کمان‌های d_r و d_l را با شعاع‌های r_r و r_l می‌پیمایند و

$$\Delta \psi = \frac{\Delta t}{l} (v_r - v_l) \quad (۹-۴)$$

$$r_o = \frac{l}{2} \frac{v_r + v_l}{v_r - v_l} \quad (۱۰-۴)$$

به دست می‌آید. که Δt فاصله زمانی نمونه‌برداری و v_l و v_r سرعت خطی چرخ‌های راست و چپ هستند که برای آن‌ها مقداری نهایی تحت عنوان v_{\max} معرفی می‌گردد. این مقدار نهایی همان نهایت سرعت موتورها است که به عنوان قید در طراحی مسیر و اعمال سیگنال کنترلی به ربات تاثیرگذار است. بر اساس شکل (۴-۲)، موقعیت زمان گسسته‌ی ربات در دستگاه مرجع توصیف می‌گردد.

$$\begin{aligned} X(n_s+1) &= X(n_s) + r_o \{ \sin[\theta(n_s) + \Delta\theta(n_s)] - \sin\theta(n_s) \} \\ Y(n_s+1) &= Y(n_s) + r_o \{ \cos\theta(n_s) - \cos[\theta(n_s) + \Delta\theta(n_s)] \} \\ \theta(n_s+1) &= \theta(n_s) + \Delta\theta(n_s) \end{aligned} \quad (۱۱-۴)$$

اگر موقعیت مطلوب ربات در صفحه‌ی مختصات مرجع را به صورت $p_g^d = [x_d \quad y_d]^T$ در نظر بگیریم،

$$R = \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix}$$

متغیرهای حالت مطلوب سیستم که نیاز به کنترل دارند از ضرب ماتریس دوران

در مختصات مرجع ربات به دست می‌آید که مختصات محلی ربات را به ما خواهند داد.

$$\begin{bmatrix} \hat{x}_d \\ \hat{y}_d \end{bmatrix} = \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} x_d \\ y_d \end{bmatrix} \quad (۱۲-۴)$$

۴-۴ - استخراج معادلات دینامیکی:

دینامیک علم حرکت شناسی است و بر خلاف سینماتیک، عوامل مولد نیرو را نیز در معادلات دخیل می‌کند. برای تحلیل دینامیکی سیستم‌ها، روش‌های متعددی مانند نیوتن-اویلر، لاگرانژ، همیلتون، کین و ... وجود دارد. استفاده از هر کدام از این روش‌ها در نهایت منجر به استخراج معادلات حرکت می‌شود ولی بر اساس شرایط و پیچیدگی مسئله باید روش مناسب برای مدل‌سازی انتخاب شود تا محاسبات را چندان دشوار ننماید.

یکی از روش‌های مناسب برای تحلیل دینامیکی سیستم‌های پیچیده و مقید، روش اویلر-لاگرانژ است، در این روش تنها با دانستن توابع انرژی، می‌توان نسبت به نوشتن معادلات دینامیکی اقدام کرد. در ادامه به استخراج معادلات دینامیکی ربات از روش اویلر-لاگرانژ خواهیم پرداخت.

اگر انرژی پتانسیل هر عضو، u_i و انرژی جنبشی آن را t_i بنامیم، مجموع کل انرژی‌های جنبشی و پتانسیل را می‌توان به صورت (۴-۱۳) معرفی نمود

$$\begin{aligned} U &= \sum u_i \\ T &= \sum t_i \end{aligned} \quad (۴-۱۳)$$

لاگرانژین سیستم از تفاضل انرژی‌های جنبشی و پتانسیل حاصل می‌گردد.

$$L = T - U \quad (۴-۱۴)$$

برای به دست آوردن معادلات دینامیکی لازم است تا معادله اویلر لاگرانژ (۴-۱۵) برای هر یک از متغیرهای حالت نوشته شود.

$$\frac{d}{dt} \left(\frac{\delta L}{\delta \dot{q}_i} \right) - \frac{\delta L}{\delta q_i} = Q_i \quad (۴-۱۵)$$

که هر یک از عناصر این معادله در ادامه معرفی می‌گردند

$$q = \text{مختصات تعمیم یافته}$$

$$Q_i = \text{نیروی محرک در هر مختصه است که از } F \frac{\delta V_i}{\delta q_i} + M \frac{\delta \omega_i}{\delta \dot{q}_i} \text{ به دست می‌آید و در}$$

اینجا برابر با گشتاور اعمالی از طرف موتورها یا همان τ می‌باشد.

حال با در نظر گرفتن فرض‌های زیر به سراغ نوشتن معادلات دینامیکی ربات می‌رویم

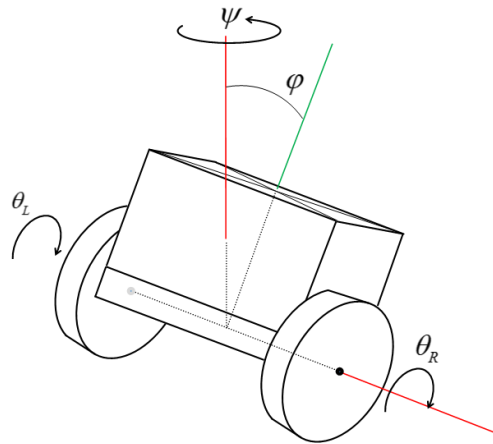
۱- ربات روی سطح افقی حرکت می‌کند

۲- چرخ‌ها دارای تماس کامل با سطح هستند و غلطش کامل روی سطح انجام می‌دهند

۳- نیروی اغتشاشی خارجی به ربات وارد نمی‌شود.

۴- از اصطکاک صرف نظر شده است.

در صورتی که ربات را به صورتی که در شکل (۳-۴) نمایش یافته است در نظر بگیریم، در خواهیم یافت که تنها انرژی پتانسیل سیستم مربوط به بدنه ربات و در حالتی است که بدنه‌ی به اندازه‌ی زاویه φ منحرف شده باشد. مابقی انرژی‌های پتانسیل به علت وجود نیروی عکس‌العملی سطح برابر با صفر می‌باشد.



شکل ۳-۴: شمای سه بعدی ربات تعادلی دوچرخ

از این رو میزان مجموع انرژی‌های پتانسیل سیستم برابر است با

$$U = m_b g d \cos \varphi \quad (۱۶-۴)$$

مجموع انرژی جنبشی ربات نیز به صورت زیر محاسبه می‌گردد.

$$T = \frac{1}{2} (I_w (\dot{\theta}_R^2 + \dot{\theta}_L^2) + m_w (\dot{x}_L^2 + \dot{x}_R^2) + MV^2 + I_{bz} \dot{\psi}^2 + I_{by} \dot{\varphi}^2) \quad (۱۷-۴)$$

در این رابطه سرعت خطی مرکز ربات را به صورت زیر مینویسیم:

$$V^2 = (\dot{x}_b + d \dot{\varphi} \cos \varphi)^2 \quad (۱۸-۴)$$

و همینطور برای محاسبه‌ی مقادیر جابه‌جایی خطی هر چرخ و جابه‌جایی خطی بدنه به صورت زیر عمل می‌کنیم:

$$\begin{aligned} x_L &= r \theta_L \\ x_R &= r \theta_R \\ x_b &= \frac{x_L + x_R}{2} \end{aligned} \quad (۱۹-۴)$$

با جایگذاری مقادیر (۴-۱۹) در معادله‌ی (۴-۱۸) و پس از آن بازنویسی معادله‌ی (۴-۱۷) انرژی جنبشی کل ربات به صورت زیر محاسبه می‌گردد.

$$T = \frac{1}{2} \left(I_{by} \dot{\varphi}^2 + (I_w + \frac{1}{r^2} M) (\dot{\theta}_L^2 + \dot{\theta}_R^2) + m_b r^2 \dot{\theta}_L \dot{\theta}_R + I_{bz} \dot{\psi}^2 + m_b d^2 \dot{\varphi}^2 \cos^2 \varphi \right) \quad (۴-۲۰)$$

کلیدیه‌ی نمادهای استفاده‌شده در مدل‌سازی دینامیکی در جدول ۴-۲ آورده شده است

جدول ۴-۲: نمادهای استفاده‌شده در مدل‌سازی دینامیکی

M	وزن کل ربات	l	فاصله بین دو چرخ
m_b	وزن بدنه	r	شعاع چرخ‌ها
m_w	وزن هر یک از چرخ‌ها	d	فاصله محور چرخ‌ها تا مرکز جرم بدنه
I_w	ممان اینرسی چرخ‌ها	V	سرعت خطی ربات در راستای \hat{x}
I_{bz}	ممان اینرسی بدنه حول محور Z	φ	جابه‌جایی دورانی بدنه حول محور \hat{y}
I_{by}	ممان اینرسی بدنه حول محور \hat{y}	ψ	جابه‌جایی دورانی ربات حول محور Z
θ_R	جابه‌جایی دورانی چرخ راست	θ_L	جابه‌جایی دورانی چرخ چپ
x_b	جابه‌جایی خطی مرکز جرم بدنه	x_L, x_R	جابه‌جایی خطی چرخ‌ها

در ادامه و با نوشتن (۴-۱۵) برای هر یک از مولفه‌ها، معادله حرکت سیستم حاصل می‌شود. [12]

$$\dot{v} = \frac{((I_{by} + m_b d^2)(\frac{\tau}{r} + m_b D_3(\dot{\varphi}^2 + \dot{\psi}^2)) - m_b D_1 \tau - g m_b D_3 - ((I_{bx} + I_{by}) D_2 - m_b D_3 D_1) \dot{\psi}^2)}{((I_{by} + m_b d^2)(M + \frac{2I_{wy}}{r^2}) - m_b^2 D_1^2)} \quad (۴-۲۱)$$

$$\ddot{\psi} = \frac{-\frac{L(\tau_1 - \tau_r)}{2r} + \dot{\psi}(2(I_{bx} - I_{by}) D_2 \dot{\varphi} - m_b D_3(2D_1 \dot{\varphi} - v))}{(I_{by} + 2I_{wz} + \frac{m_w L^2}{2} + \frac{I_{wy} L^2}{2r^2} + (I_{bx} - I_{by}) \sin^2 \varphi + m_b D_3^2)} \quad (۴-۲۲)$$

$$\ddot{\varphi} = \frac{m_b D_1(\frac{\tau}{r} + m_b D_3(\dot{\varphi}^2 + \dot{\psi}^2)) - (M + \frac{2I_{wy}}{r^2})(\tau - g m_b D_3 - ((I_{bx} - I_{by}) D_2 - m_b D_3 D_1) \dot{\psi}^2)}{((I_{by} + m_b d^2)(M + \frac{2I_{wy}}{r^2}) - m_b^2 D_1^2)} \quad (۴-۲۳)$$

که مقادیر $D_3 = d \sin \varphi$ و $D_2 = \sin \varphi \cos \varphi$ ، $D_1 = -d \cos \varphi$ معرفی می‌شوند.

این معادلات را همچنین می‌توان به فرم کلی حالت (۴-۲۴) نوشت که برای این منظور بردار مختصات

تعمیم‌یافته به صورت $q = [x \ y \ \psi \ \varphi \ \theta_R \ \theta_L]^T$ تعریف می‌گردد.

$$M(q)\ddot{q} + C(q, \dot{q}) + G(q) = P\tau + \sigma_i \quad (۲۴-۴)$$

$M(q)$ ماتریسی $n \times n$ اینرسی‌ها، $C(q, \dot{q})$ ماتریس $n \times n$ از ترم‌های شتاب کریولیس و گریز از مرکز، $G(q)$ برداری $n \times 1$ از ترم‌های جاذبه، P ترانهاده‌ی ماتریس ورودی‌ها و τ بردار گشتاورهای ورودی می‌باشد.

عبارت $\sigma_i = [\zeta]^T \lambda$ در این معادله از ضرب ترانهاده‌ی ماتریس قیود ($[\zeta]$) در ضریب لاگرانژ (λ) به دست می‌آید.

در صورتی که تعداد مختصات تعمیم‌یافته برابر با تعداد درجه‌های آزادی سیستم باشد $\sigma_i = 0$ می‌شود و معادله‌ی فوق به فرم معادله‌ی سیستم‌های غیر مقید تبدیل می‌گردد. اما زمانی که قیود هولونومیک وجود داشته باشند ماتریس قیود از حل معادله دیفرانسیل مرتبه اول زیر به دست می‌آید

$$[\zeta] \dot{q} = 0 \quad (۲۵-۴)$$

برای سیستم‌های غیرهولونومیک نیز معادله‌ی مرتبه اول قیود به فرم $\phi(t) + [\zeta] \dot{q} = 0$ معرفی می‌شود که اگر، وابستگی صریحی به زمان در سیستم وجود نداشته باشد، می‌توان از ترم زمانی این رابطه صرف‌نظر کرد. بنابراین با شرط عدم وابستگی صریح به زمان، ماتریس قیود سیستم‌های هولونومیک و غیرهولونومیک از حل معادله دیفرانسیل مرتبه اول $[\zeta] \dot{q} = 0$ به دست می‌آید.

اما به جای حل این معادله‌ی دیفرانسیل و به دست آوردن مقدار ضریب λ که کار چندان آسانی هم نیست، روش دیگری پیشنهاد گردیده [30] که به حذف ضریب لاگرانژ از معادلات سیستم خواهد انجامید.

اگر بردار سرعت ربات را $v = [V \ \psi \ \dot{\varphi}]^T$ معرفی کنیم، رابطه‌ی ژاکوبین ربات به صورت زیر نوشته خواهد شد.

$$\dot{q} = J(q)v \quad (۲۶-۴)$$

که در رابطه فوق $\dot{q} = [\dot{x} \quad \dot{y} \quad \dot{\psi} \quad \dot{\phi} \quad \dot{\theta}_R \quad \dot{\theta}_L]^T$ مشتق بردار مختصات تعمیم یافته است. برای حذف ضریب لاگرانژ از معادله سیستم لازم است تا معادله (۴-۲۶) را در (۴-۲۵) جایگذاری نماییم که بر این اساس $J^T \zeta^T \lambda = 0$ می‌گردد.

ماتریس ژاکوبین با توجه به مختصات تعمیم یافته تعریف می‌گردد.

$$J(q) = \begin{bmatrix} \cos \psi & 0 & 0 \\ \sin \psi & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ \frac{1}{r} & \frac{l}{2r} & 0 \\ \frac{1}{r} & -\frac{l}{2r} & 0 \end{bmatrix} \quad (۲۷-۴)$$

ماتریس قیود نیز از حل معادله (۴-۲۵) به دست می‌آید.

$$\zeta(q) = \begin{bmatrix} \sin \psi & -\cos \psi & 0 & 0 & 0 & 0 \\ \cos \psi & \sin \psi & \frac{1}{2} & 0 & -r & 0 \\ \cos \psi & \sin \psi & -\frac{1}{2} & 0 & 0 & -r \end{bmatrix} \quad (۲۸-۴)$$

حال جهت حذف ضریب لاگرانژ، طرفین معادله (۴-۲۸) را در J^T ضرب می‌کنیم

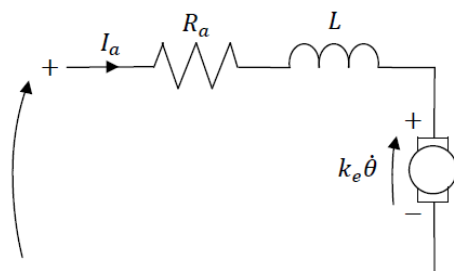
$$J^T M(q) \ddot{q} + J^T (C+G) = J^T P \tau \quad (۲۹-۴)$$

در صورت ادغام مشتق معادله (۴-۲۶) در معادله (۴-۲۹) معادله جدید به فرم زیر به دست می‌آید که بر حسب مشتق بردار سرعت‌های جدید ($\dot{v} = [\dot{V} \quad \dot{\psi} \quad \dot{\phi}]^T$) خواهد بود. از این طریق معادلات دینامیکی سیستم در فرم حالت به صورت زیر بیان می‌گردد.

$$\dot{v} = J^{-1} (M(q)^{-1} (P\tau - C - G) - \dot{J}v) \quad (۳۰-۴)$$

۴-۵- مدل سازی دینامیکی موتورها:

برای این ربات از دو موتور کوچک استفاده شده که معادله‌ی دینامیکی آنان در کل سیستم تأثیرگذار خواهد بود چرا که گشتاور اعمالی، به وسیله موتورها تولید می‌شود. از همین رو نیاز به استخراج معادلات دینامیکی موتور داریم.



شکل ۴-۴: مدل مداری موتور مغناطیس دائم

مدل مداری موتور جریان مستقیم مغناطیس دائم^۱ در شکل (۴-۴) نشان داده شده است و نمادهای مورد استفاده برای مدل سازی در جدول (۴-۳)، بیان شده است

جدول ۴-۳: نمادهای مورد استفاده در مدل سازی موتور مغناطیس دائم

T_e	گشتاور الکترومغناطیسی موتور	emf	ولتاژ برگشتی موتور
T_l	گشتاور بار متصل به موتور	U	ولتاژ تغذیه موتور
r_g	ضریب تبدیل گیربکس	I_a	جریان موتور
J_m	ممان اینرسی موتور	R_a	مقاومت الکتریکی موتور
b_m	ضریب اصطکاک ویسکوز موتور	L_a	اندوکتانس القایی موتور
K	ثابت مغناطیسی موتور	ϕ	شار مغناطیسی موتور
k_m	ثابت الکتریکی موتور	k_e	ثابت الکترومغناطیسی موتور

در نظر بگیرید که روتور در حال چرخش باشد، در این حالت بر روی سیم‌های موتور یک ولتاژ القا می‌شود که این ولتاژ به سرعت موتور و میدان مغناطیسی حاصل از قطب‌های مغناطیسی بستگی دارد. و می‌توان آن را به صورت $emf = K \dot{\phi}$ نشان داد اما به علت استفاده از مغناطیس‌های دائم در این

¹ Permanent magnet dc motor

نوع از موتورهای جریان مستقیم؛ مقدار شار ϕ عددی ثابت است که باعث می‌شود مقدار ولتاژ القایی متناسب با سرعت موتور باشد.

$$emf = K_e \dot{\theta} \quad (۳۱-۴)$$

از طرفی با استفاده از قانون کریشهف و نوشتن معادله در مدل مداری شکل (۴-۴) خواهیم داشت:

$$U = R_a I_a + k_e \dot{\theta} + L \frac{dI_a}{dt} \quad (۳۲-۴)$$

به علت کوچک‌تر بودن ثابت زمانی موتور از ثابت زمانی سیستم‌های مکانیکی، می‌توان از اندوکتانس سلف در معادله (۳۲-۴) صرف‌نظر کرد.

$$U = R_a I_a + k_e \dot{\theta} \quad (۳۳-۴)$$

مطابق قانون لنز میدانیم که به سیم حامل جریان در یک میدان مغناطیسی نیرو وارد می‌شود. حال با فرض عبور جریان از سیم‌های روتور، نیروی واردشده به این سیم‌ها از طرف میدان مغناطیسی، باعث به وجود آمدن گشتاور می‌شود. این گشتاور متناسب با جریان روتور و میدان مغناطیسی است که با توجه به ثابت بود شار مغناطیسی، می‌توان رابطه‌ی گشتاور الکترومغناطیسی حاصل را به صورت زیر نوشت

$$T_e = k_m I_a \quad (۳۴-۴)$$

با محاسبه مقدار I_a از معادله (۳۳-۴) و جایگذاری در معادله (۳۴-۴)، گشتاور الکترومغناطیسی موتور بر حسب ولتاژ ورودی و سرعت موتور به دست می‌آید.

$$T_e = k_m \left(\frac{U - K_e \dot{\theta}}{R_a} \right) \quad (۳۵-۴)$$

گشتاور به دست آمده علاوه بر غلبه بر گشتاور بار می‌باید بر اینرسی موتور و اصطکاک موجود نیز غلبه کند.

$$T_e = J_m \ddot{\theta} + b_m \dot{\theta} + r_g T_l \quad (۳۶-۴)$$

اما با توجه به کوچک بودن اینرسی و اصطکاک موجود در موتور می توان از مقدار آن‌ها چشم‌پوشی کرد و با توجه به مقاوم بودن قانون کنترل پیشنهادی این مقادیر را به عنوان دینامیک‌های مدل نشده و یا یک اغتشاش در سیستم تلقی کرد. در نتیجه معادله (۴-۳۶) به (۴-۳۷) تغییر می‌یابد.

$$T_e = r_g T_l \quad (۴-۳۷)$$

مقدار $K_e \dot{\theta}$ در معادله‌ی (۴-۳۵) که معرف ولتاژ پسماند موتور است، عدد کوچکی است که می‌توان به نحوی مشابه از آن صرف‌نظر نمود که بر این اساس مقدار گشتاور الکترومغناطیسی، ضربی از ولتاژ اعمالی به موتور می‌گردد.

$$T_e = U \left(\frac{k_m}{R_a} \right) \quad (۴-۳۸)$$

از جایگذاری (۴-۳۸) در (۴-۳۷) داریم:

$$T_l = \frac{T_e}{r_g} = U \frac{k_m}{r_g R_a} \quad (۴-۳۹)$$

معادله (۴-۳۵) گشتاور تولیدی توسط هر یک از محرک را بیان می‌کند. که با جایگذاری مقدار T_l هر موتور در معادلات (۴-۲۵)، (۴-۲۶)، (۴-۲۷) معادله کلی سیستم به دست می‌آید.

۴-۶- جمع‌بندی:

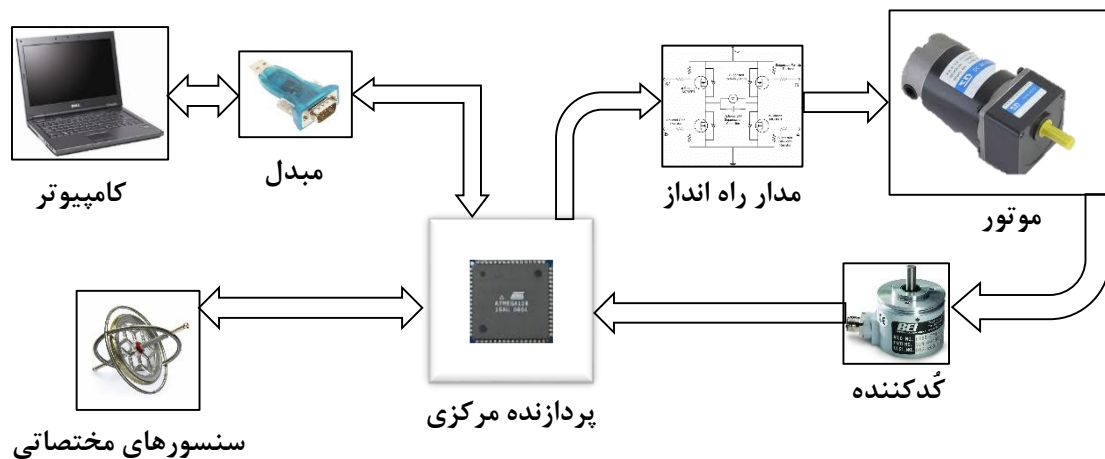
از آنجا که روش کنترلی پیشنهادی یک روش بر پایه‌ی مدل تلقی می‌شود و از طرف دیگر برای تعیین موقعیت ربات نیاز به کنترل سینماتیکی آن داریم، معادلات سینماتیکی زمان گسسته و زمان پیوسته‌ی ربات در این بخش محاسبه‌شد. پس از آن به مدل‌سازی دینامیکی ربات با استفاده از روش اویلر- لاگرانژ پرداختیم. سپس معادلات را به فرم کلی حالت بازنویسی کردیم که در ادامه‌ی آن از راهکار ارائه شده برای حذف ضریب لاگرانژ استفاده شد. در نهایت مدل دینامیکی موتور dc تحلیل و رابطه‌ی میان ولتاژ ورودی و گشتاور خروجی آن تعیین گشت، که از جایگذاری آن در معادلات اصلی، معادلات کلی ربات به دست می‌آید.

۵- فصل پنجم

انتخاب قطعات و ساخت ربات تعادلی

۵-۱- معرفی ساختار:

همان طور که پیش از این نیز عنوان شد، ربات تعادلی دوچرخ، وسیله‌ای است که توانایی حفظ تعادل خود بر روی دو چرخ را دارد. و درعین حال باید بتواند مسیری از پیش تعیین شده را پیمایش کند. مجموعه‌ی کلی ربات را می‌توان به صورت بلوک دیاگرام شکل (۵-۱) نشان داده که در ادامه به معرفی و طراحی هر یک از بخش‌های آن خواهیم پرداخت.



شکل ۵-۱: بلوک دیاگرام ربات تعادلی زحل

نحوه‌ی کار ربات به این صورت است که اطلاعات از طریق USB رایانه گرفته شده و به وسیله‌ی یک مبدل دوطرفه، به متوالی تبدیل می‌شود. این اطلاعات مستقیم وارد پردازنده مرکزی می‌گردد و بر اساس آن، فرمان‌های کنترلی برای کنترل موتورها به مدار راه‌انداز اعمال می‌گردد، مدار راه‌انداز به کنترل موتورها از طریق پالس‌های دریافتی پرداخته و ربات الگوی حرکتی خواسته شده توسط رایانه را انجام می‌دهد.

اما لازم است تا اطلاعات ربات، اعم از زاویه‌ی بدنه، موقعیت موتورها، سرعت زاویه‌ای موتورها و ... به رایانه ارسال گردد تا در تصمیمات بعدی مدنظر قرار گیرد. برای این منظور اطلاعات حسگرهای مختلفی که از مجموعه‌ای شتاب سنج وژیروسکوپ و ... تشکیل شده است؛ از درگاه‌های آنالوگ و دیجیتال به پردازنده وارد می‌شود این اطلاعات با مجموعه اطلاعات دریافتی از کدکننده‌های افزایشی موجود سر راه موتورها ترکیب و نهایتاً به رایانه ارسال می‌گردد.

۵-۲- مراحل ساخت:

برای طراحی و ساخت این ربات مراحل خاصی طی شده که می‌توان مراتب آن را به شرح زیر برشمرد

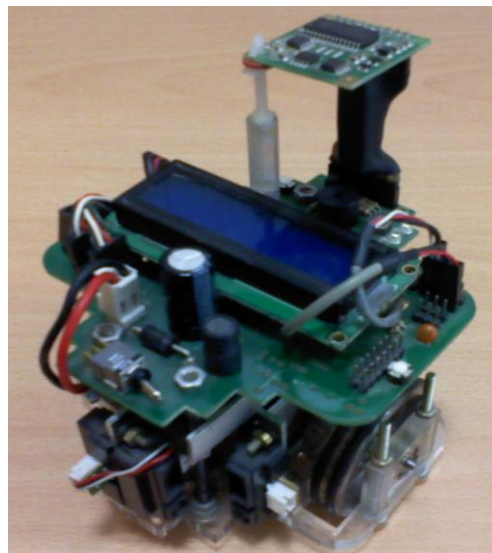
۱- انتخاب قطعات

۲- طراحی مکانیکی

۳- طراحی مدارات الکترونیکی

۴- ساخت

برای این پژوهش یک طراحی اولیه صورت گرفت که منجر به ساخت ربات تعادلی شکل (۵-۲) شد. این ربات در ابتدای ساخت، با انگیزه‌های شخصی، به نام ^۱CPG نام‌گذاری شد.



شکل ۵-۲: ربات تعادلی CPG

CPG اگرچه تمامی موارد خواسته‌شده را در خود جای داده بود، اما بروز برخی مشکلات و عدم دقت مناسب در طراحی و انتخاب قطعات آن، باعث شد تا در پیاده‌سازی‌های اولیه از خود ضعف نشان داده و ما را مجبور به طراحی ربات دیگر نماید. ربات جدید می‌بایست معایب ربات قبل را برطرف می‌نمود که در ادامه به شرح برخی از این معایب می‌پردازیم.

¹ Creative Power Group's

یکی از بزرگ‌ترین عیوب این ربات طول بیش از حد آن بود که به علت کم بودن قطر چرخ‌ها، امکان نوسانات کمی را به ربات می‌داد و عملاً ربات در یک انحراف ۱۰ درجه‌ای با زمین برخورد می‌کرد. از دیگر مشکلات اساسی ربات CPG، کند بودن موتورها و عدم پاسخ سریع به تغییرات ولتاژ بود. به عبارت دیگر، موتورهای انتخاب شده از نوع موتورهای معمولی ZGA12 بود؛ دارای مقاوت پایین و ضریب اندوکتانس بالا بود که باعث عدم توانایی کنترل‌کننده‌ها در برقراری تعادل ربات می‌شد. سایر مشکلات این ربات به صورت لیست وار در ادامه بیان شده و برای هر یک توضیح مختصری نیز داده شده است.

- ۱- استفاده از رگولاتورهای کلید زنی: به منظور پایین آوردن توان اتلافی ربات، از رگولاتورهای سری LM2576-5v استفاده شده بود اما به علت وجود نوسانات فرکانس بالا در خروجی رگولاتور، با وجود فیلترهای متعدد سلفی-خازنی همچنان ارتباط سریال میان قطعات ربات دچار مختل می‌شد.
- ۲- برای ارتباط میان قطعات از درگاه I2C نرم افزاری استفاده شده بود که سرعت کمی داشت و میبایست از درگاه سخت سخت افزاری استفاده شود.
- ۳- جنس چرخ دنده‌های گیربکس، پلاستیکی بود و با توجه به نوسانات کنترلی، سریعاً خورده شده و خروجی گیربکس دارای لقی زیادی می‌شد.
- ۴- چرخ‌ها از جنس فولاد تراشیده شده بود و علاوه بر مشکل زنگ زدگی، به علت دارا بودن وزن زیاد، تاخیر زیادی در راه اندازی و پاسخ موتور ایجاد می‌کرد.
- ۵- ربات بیش از حد عریض بود و قدرت مانور زیادی نداشت.
- ۶- تنها از سنسور شتاب سنج استفاده شده بود که می‌بایست با دوبار انتگرال گیری از مقادیر خروجی آن به سرعت‌ها و موقعیت‌ها می‌رسیدیم که خطای زیادی داشت.
- ۷- موتورهای داری کدکننده نبود و از کدکننده‌های خارجی سر راه موتور استفاده شده بود که پاسخ مطلوبی نداشت.

تمامی مراحل طراحی و کنترل بر روی این ربات تست شد و پس از مشخص شدن کلیه نواقص اقدام به طراحی ربات دیگری شد که «زحل» نام گرفت.

زحل اگر چه از نظر ظاهری شباهتی به CPG نداشت اما با فرآیندی مشابه، شبیه‌سازی و کنترل می‌شد و این بار نتایجی به مراتب بهتر دریافت شد که در بخش بعد به توصیف آن خواهیم پرداخت. با این مقدمه از طراحی CPG گذشته و به سراغ بیان مراحل طراحی ربات زحل می‌رویم.

۵-۳- انتخاب قطعات:

موتور: موتورهای ربات را می‌توان اصلی‌ترین بخش آن به شمار آورد، چرا که در عمل، نحوه‌ی پاسخ موتور به خروجی‌های کنترلی شما، تعیین‌کننده‌ی پاسخ سیستم هستند. برای این ساختار نیاز به موتوری با توانایی تولید گشتاور بالا^۱، اتلاف پایین، سرعت متوسط و پاسخ سریع است. علاوه بر آن سائز موتور باید کوچک و حتی‌الامکان سبک باشد. و امکان اتصال کدکندهای افزایشی را داشته باشد. لذا پس از بررسی‌های انجام‌شده از مجموعه‌ی آماده‌ی MOT1058 استفاده شد که از مجموعه‌ی موتور، گیربکس و کدکننده، تشکیل شده است. مشخصات اصلی MOT1058 در جدول (۵-۱) بیان شده است.

جدول ۵-۱: مشخصات موتور الکتریکی

واحد	مقدار	عنوان
Ω	۱۱	مقاومت آرمیچر
RPM^2	۱۳۴۰۰	سرعت نامی موتور
	۱۴۱:۱	ضریب تبدیل گیربکس
PPR^3	۱	خروجی کدکننده

شکل (۵-۳) شکل ظاهری موتور انتخاب‌شده را نشان می‌دهد.

^۱ گشتاوری که چندین برابر مقدار مورد نیاز برای به حرکت در آوردن ربات باشد.

^۲ Rote Per Minutse

^۳ Pulse Per Rote



شکل ۳-۵: موتور انتخاب شده برای ربات

باتری: برای تغذیه‌ی ربات از یک باتری لیتیم پلیمر ۳ سلولی استفاده شده که مجموعاً ولتاژ $11/7$ ولت را برای تغذیه ربات فراهم می‌کند. علت اصلی این انتخاب، وزن و حجم کم این نوع باتری‌ها در مقابل ظرفیت جریان دهی آنان می‌باشد.

حسگرها: اگرچه انتخاب موتوری مناسب به رفتار کنترلی ربات خیلی کمک می‌کند اما این حسگرها هستند که باید اطلاعات دقیقی از وضعیت فعلی ربات استخراج نمایند، پس عدم انتخاب حسگرهای صحیح ممکن است باعث خطا در حلقه‌ی بازخورد و برهم خوردن تعادل ربات گردد. شاید در شبیه‌سازی با نرم‌افزار متلب بتوان با انتگرال گرفتن از شتاب‌ها به سرعت‌ها و موقعیت‌ها دست یافت اما در عمل این امر امکان‌پذیر نیست، چرا که به علت گسستگی زمانی میان نمونه‌برداری‌ها، خطای انتگرال‌گیری به وجود آمده که در مدت زمان کوتاهی باعث غیرقابل اطمینان بودن اطلاعات حسگرها می‌شود. از این رو لازم است تا علاوه بر انتخاب حسگرهای شتاب سنج وژیروسکوپ، موقعیت ربات نیز توسط یک حسگر تعیین گردد.

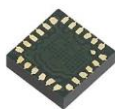
سایر مختصات ربات، اعم از موقعیت افقی، عمودی و زاویه انحراف ربات در صفحه را با استفاده از کدکننده‌های افزایشی می‌توان با دقت زیادی محاسبه کرد اما در این میان زاویه‌ی بدنه ربات با محور عمودی که ارتباط مستقیم با حفظ تعادل ربات دارد را تنها می‌توان با انتگرال‌گیری از خروجی حسگرژیروسکوپ به دست آورد که قبلاً به بیان مشکلات این روش پرداختیم.

برای برطرف کردن این مشکل، علاوه برژیروسکوپ، یک حسگر زاویه نیز برای تشخیص انحراف دقیق بدنه از محور عمودی، در نظر گرفته شده است. در ادامه به معرفی حسگرهای انتخاب شده می‌پردازیم.

یکی از معروفترین مدارهای مجتمع موجود، تراشه‌ی MPU-9150 ساخت شرکت InvenSense است که اولین حسگر ۹ محوره به حساب می‌آید و برای ربات زحل از آن استفاده شده است.

در این تراشه یک ژيروسکوپ ۳ محوره، یک شتاب سنج ۳ محور و یک قطب‌نمای ۳ محور قرار گرفته است. اگر چه اطلاعات خروجی حسگرها به صورت آنالوگ تولید می‌شود. اما نحوه‌ی ارتباط با این تراشه از طریق درگاه I2C است و علت آن استفاده از یک مبدل آنالوگ به دیجیتال و یک پردازنده‌ی پرسرعت داخلی است.

اما برای تشخیص زاویه‌ی انحراف، تراشه‌ها و ماژول‌های زیادی وجود دارد که همگی از سرعتی فوق‌العاده پایین برخوردار هستند، در میان تراشه‌های موجود، تنها تراشه‌ای که قدرت تشخیص بالا داشته و با سرعتی معادل ۱۰ هرتز، توانایی اندازه‌گیری زاویه را دارد، تراشه‌ی TCA100T است که در شکل (۴-۵) همراه با تراشه‌ی MPU-9150 به نمایش در آمده است.



MPU-9150



TCA100T

شکل ۴-۵: حسگرهای ربات

بدنه (شاسی): برای ساخت بدنه ربات از نوع خاصی از پلاستیک‌های فشرده معروف به پلکسی استفاده شده که علاوه بر قابلیت برش‌کاری توسط لیزر، دارای استحکام نسبتاً بالا و وزن کم هستند.

چرخ: چرخ‌های ربات از جنس آلومینیوم و با قطر ۳,۸ سانتیمتر طراحی شده‌اند که پس از قرار گرفتن روکش پلاستیکی، قطرش به ۴,۵ سانتیمتر می‌رسد. شکل (۵-۵) چرخ‌های ربات را نشان می‌دهد.



شکل ۵-۵: چرخ ربات

۵-۴- طراحی مکانیکی:

همان طور که در مقدمه اشاره شد، ربات طراحی شده دارای ساختاری کوچک و سبک است و به عنوان یک ریز ربات معرفی می‌گردد به همین خاطر، در طراحی مکانیکی سعی شده تا علاوه توجه به ظرافت ربات، نیازهای اصلی سیستم نیز برطرف گردد.

با توجه به طراحی‌های صورت گرفته؛ ابعاد نهایی ربات $10 * 5 * 12$ سانتیمتر می‌باشد که به ترتیب بیان‌کننده فاصله بیرونی دوچرخ، طول ربات و ارتفاع آن است.

چینش قطعات نیز به گونه‌ای بوده که مرکز جرم حتی‌الامکان در صفحه میانی دوچرخ قرار گیرد. برای این منظور باتری ربات که همراه با موتورها، بیش‌ترین وزن ربات را تشکیل می‌دهند، در مرکز ربات قرار گرفته است. لازم به ذکر است وزن نهایی ربات طراحی شده در حدود ۳۸۰ گرم می‌باشد.

تمامی قطعات اعم از بدنه و بسته‌ای موتور به وسیله نرم‌افزار AutoCAD طراحی و سپس به وسیله لیزر بر روی پلکسی پیاده‌سازی شده‌اند

۵-۵- طراحی الکترونیکی:

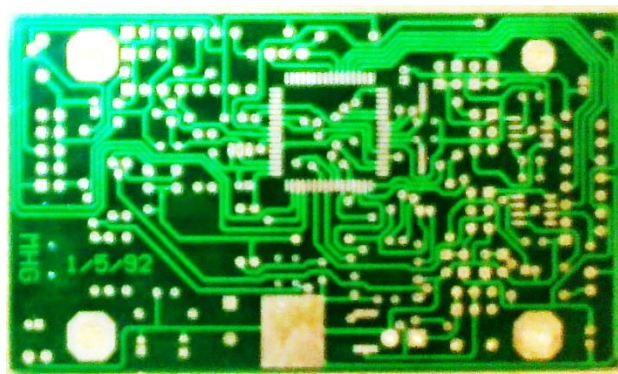
هر یک از قطعات مورد استفاده در ربات نیازمند مدار واسطی جهت راه‌اندازی است و لازم است تا مدارات لازم برای هر یک طراحی شود. مدار تغذیه به گونه‌ای طراحی شده که نیاز به دو سطح ولتاژ ثابت ۵ و ۳,۳ ولتی برای راه‌اندازی میکروکنترلر و حسگرهای ربات را بر طرف سازد. برای کنترل موتورها نیز از

تراشه‌ی راه‌انداز معروف L298 استفاده‌شده که پالس PWM تولیدی توسط میکروکنترلر را به موتورها ارسال می‌کند.

جهت برقراری ارتباط بین میکروکنترلر و رایانه از یک واسط متوالی FT232BM استفاده‌شده و همچنین درگاهی برای اتصال ماژول‌های بی‌سیم نیز تعبیه شده است.

میکروکنترلر ربات زحل، یک تراشه‌ی ATMEGA 128 از سری‌های AVR است که به خاطر سهولت در برنامه‌نویسی انتخاب‌شده است.

فیبر مدار چاپی ربات نیز در نرم‌افزار Altium-Designer به صورت دو لایه طراحی شده است که شکل (۵-۶) نمایی از یک طرف برد طراحی شده را نشان می‌دهد.



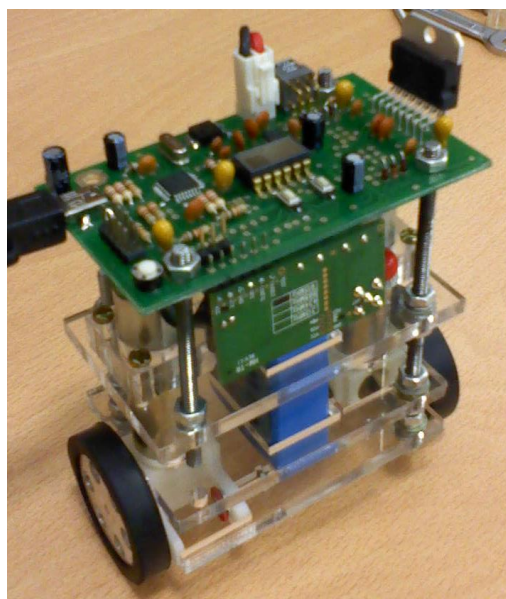
شکل ۵-۶: نمای از فیبر مدار چاپی طراحی شده

۵-۶- ساخت:

در این مرحله تمامی قطعات الکترونیکی و مکانیکی ربات مونتاژ شده و ربات آماده بهره‌برداری می‌گردد.

ربات نهایی در شکل (۵-۷) نشان داده شده و مشخصات و وزن هر یک از قسمت‌های ربات در جدول

(۵-۲) به تفصیل بیان شده است.



شکل ۵-۷: ربات زحل

جدول ۵-۱: وزن قطعات مختلف ربات

وزن هر یک از چرخ‌ها	۱۲ گرم
وزن هر یک از موتورها	۷۹ گرم
وزن باتری	۸۵ گرم
وزن مدارات الکترونیکی	۶۳ گرم
وزن بدنه و اتصالات	۵۶ گرم
وزن کلی ربات	۳۸۶ گرم

۶- فصل ششم

طراحی کنترل کننده و شیه سازی عملکرد

۶-۱- مقدمه:

برای کنترل ربات تعادلی دو هدف اصلی وجود دارد: برقراری تعادل و پیمایش مسیر دلخواه. رسیدن به هر دوی این اهداف گرچه امر غیرممکنی نیست اما باید در نظر داشت که به خاطر ساختار خاص این ربات و آمیخته بودن معادلات آن، مسئله کنترل با چالش‌های شدیدی روبه‌روست زیرا برای کنترل زمانی مسیر حرکت، ربات نیاز به شتاب گرفتن در یک جهت خاص را دارد و از طرفی این شتاب باعث برهم خوردن تعادل ربات می‌گردد. به علاوه وجود عدم قطعیت‌های فراوان در مدل‌سازی سیستم و همچنین صرف‌نظر کردن از برخی شرایط خاص مثل شیب مسیر، باعث به وجود آمدن دینامیک‌های مدل نشده در سیستم می‌شود که فرآیند کنترلی را با مشکل مواجه می‌کند.

در این فصل ابتدا به تحلیل کنترل‌کننده‌های مجزای مد لغزشی برای کنترل تعادل و مسیر حرکت سیستم پرداخته شده که توسط قوانین ماتریس همبندی^۱ خروجی کنترل‌کننده‌ها با هم ترکیب و به سیستم اعمال شده است.

پس از آن از شبکه تطبیقی عصبی فازی عنوان شده در فصل ۲ برای کنترل ربات استفاده شده است. نتایج به دست آمده از این شبکه با روش قبل مقایسه و در پایان نشان داده می‌شود ترکیب هر دو روش به نتیجه‌ی بهتری خواهد انجامید.

در پایان کنترل‌کننده‌ی نهایی برای کنترل مسیر پیشنهادی بر روی ربات واقعی آزمایش شده و نتایج تئوری و عملی با یکدیگر مقایسه می‌گردد.

۶-۲- طراحی مسیر:

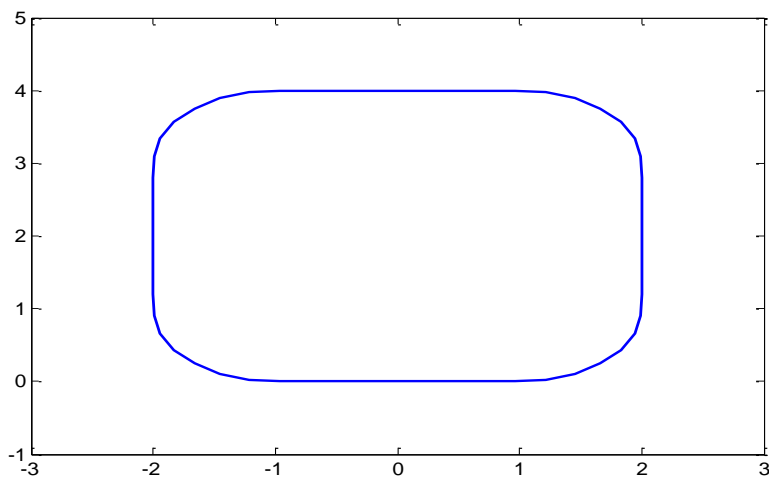
طراحی مسیر حرکت ربات تعادلی دوچرخ تفاوت چندانی با طراحی مسیر برای یک ربات ندارد چرا که مسیر طراحی شده به بیان سرعت خطی و دورانی ربات می‌پردازد و نمی‌توان برای زاویه‌ی بدنه‌ی ربات با محور عمود بر جاذبه مسیری طراحی کرد. لذا می‌توان گفت کنترل ردگیری ربات در صفحه‌ی مختصات XY همراه با کنترل نقطه تنظیم^۲ برای زاویه‌ی φ صورت می‌گیرد.

¹ Coupling

² Setpoint

برای تمامی طراحی‌ها، پیاده‌سازی‌ها و شبیه‌سازی‌ها از یک مسیر مستطیلی شکل به گوشه‌های خمیده استفاده شده است که معادله آن در ادامه بیان شده است.

$$X_d = \begin{cases} 0.1t & 0 \leq t < 10 \\ 1 + \cos\left(\frac{2\pi(t-10)}{T} - \frac{\pi}{2}\right) & 10 \leq t < 30 \\ 2 & 30 \leq t < 50 \\ 1 + \cos\left(\frac{2\pi(t-30)}{T} - \frac{\pi}{2}\right) & 50 \leq t < 70 \\ 1 - 0.1(t-70) & 70 \leq t < 90 \\ -1 + \cos\left(\frac{2\pi(t-50)}{T} - \frac{\pi}{2}\right) & 90 \leq t < 110 \\ -2 & 110 \leq t < 130 \\ -1 + \cos\left(\frac{2\pi(t+10)}{T} - \frac{\pi}{2}\right) & 130 \leq t < 150 \\ -1 + 0.1(t-150) & 150 \leq t < 160 \end{cases} \quad Y_d = \begin{cases} 0 & 0 \leq t < 10 \\ 1 + \sin\left(\frac{2\pi(t-10)}{T} - \frac{\pi}{2}\right) & 10 \leq t < 30 \\ 1 - 0.1(t-30) & 30 \leq t < 50 \\ 3 + \sin\left(\frac{2\pi(t-30)}{T} - \frac{\pi}{2}\right) & 50 \leq t < 70 \\ 4 & 70 \leq t < 90 \\ 3 + \sin\left(\frac{2\pi(t-50)}{T} - \frac{\pi}{2}\right) & 90 \leq t < 110 \\ 3 - 0.1(t-110) & 110 \leq t < 130 \\ 1 + \sin\left(\frac{2\pi(t+10)}{T} - \frac{\pi}{2}\right) & 130 \leq t < 150 \\ 0 & 150 \leq t < 160 \end{cases}$$



شکل ۶-۱: مسیر طراحی شده برای ربات تعادلی

مسیر طراحی شده در شکل (۶-۱) به نمایش در آمده است.

مختصات به دست آمده از مسیر طراحی شده پس از تبدیل به مختصات دستگاه محلی ربات توسط معادله (۴-۱۲) به عنوان مقادیر مطلوب به کنترل کننده اعمال می‌شود.

۶-۳ - طراحی کنترل کننده مد لغزشی تفکیک شده:

همان طور که در بخش‌های قبلی اشاره شد، ربات تعادلی را می‌توان ترکیبی از دو وسیله‌ی ربات متحرک و پاندول معکوس دانست. بر این اساس اگر بتوان معادلات سیستم را به صورت تفکیک شده نوشت، با

اعمال دو کنترل کننده‌ی مجزا می‌توان به کنترل سیستم پرداخت. با همین دید، در ادامه‌ی این بخش به جداسازی معادلات ربات تعادلی پرداخته و توسط روش مد لغزشی بیان شده در فصل ۳، برای هر یک، قانون کنترلی پیشنهاد می‌دهیم.

لازم به ذکر است، برای کنترل میسر حرکت ربات تعادلی، نیاز به یک کنترل سینماتیکی است که پیش از طراحی کنترل کننده‌ی مد لغزشی لازم است تا مورد بررسی قرار گیرد. با این مقدمه به سراغ طراحی کنترل کننده می‌رویم

از معادله (۴-۲۳) داریم: $J^T M(q)\ddot{q} + J^T (C+G) = J^T P\tau$ که برای کنترل به روش مد لغزشی لازم است این معادله را به فرم (۶-۱) بنویسیم.

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

که برای تحقق این امر، هر یک از عناصر معادله به صورت زیر تعریف می‌شوند.

$$f(x) = \begin{bmatrix} f_1(x) \\ f_2(x) \end{bmatrix}, g(x) = \begin{bmatrix} g_1(x) \\ g_2(x) \end{bmatrix} = \begin{bmatrix} 0_{4 \times 2} \\ (J^T M J)^{-1} J^T P \end{bmatrix}, u = \begin{bmatrix} \tau_1 \\ \tau_2 \end{bmatrix} \quad (۶-۲)$$

جهت تسهیل در مدل‌سازی سیستم لازم است تا با یک تبدیل، گشتاورهای τ_1, τ_2 که نشان‌دهنده‌ی گشتاور موتورها هستند را به دو گشتاور C_ψ, C_y تبدیل نماییم. این تبدیل بر اساس [3] به فرم زیر پیشنهاد می‌گردد

$$u = \begin{bmatrix} \tau_1 \\ \tau_2 \end{bmatrix} = \begin{bmatrix} 0.5 & 0.5 \\ 0.5 & -0.5 \end{bmatrix} \begin{bmatrix} C_\psi \\ C_y \end{bmatrix} \quad (۶-۳)$$

که در این صورت می‌توان معادله (۶-۱) را به فرم زیر نوشت.

$$\dot{x} = f(x) + g(x) \begin{bmatrix} 0.5 & 0.5 \\ 0.5 & -0.5 \end{bmatrix} \begin{bmatrix} C_\psi \\ C_y \end{bmatrix} \quad (۶-۴)$$

با توجه به معادله (۶-۴) می‌توان دو معادله سینماتیکی و دینامیکی ربات را به صورت زیر بیان نمود

$$\dot{q} = \begin{bmatrix} \dot{x} & \dot{y} & \dot{\psi} & \dot{\phi} \end{bmatrix}^T = \begin{bmatrix} v \cos \psi & v \sin \psi & \omega_\psi & \omega_\phi \end{bmatrix}^T \quad (۶-۵)$$

$$\dot{v}_r = \begin{bmatrix} \dot{v} \\ \dot{\omega}_\psi \\ \dot{\omega}_\varphi \end{bmatrix} = \begin{bmatrix} f_{21}(x) \\ f_{22}(x) \\ f_{23}(x) \end{bmatrix} + \begin{bmatrix} g_{21}(x) & 0 \\ g_{22}(x) & 0 \\ 0 & g_{23}(x) \end{bmatrix} \begin{bmatrix} C_\psi \\ C_y \end{bmatrix} \quad (6-6)$$

بر این اساس می‌توان دو کنترل مجزا بر روی C_ψ و C_y انجام داده و نتایج به دست آمده را دوباره با استفاده از تبدیل (6-3) تلفیق کرده و به سیستم اعمال کرد.

هدف اصلی برای طراحی این کنترل‌کننده، تعقیب یک مسیر زمانی و درعین‌حال نگاه‌داشتن زاویه‌ی بدنه در نزدیکی صفر می‌باشد.

اگر بردار $q_d = [x_d \ y_d \ \psi_d \ \varphi_d]^T$ را بردار مقادیر مطلوب در نظر بگیریم. بردار خطای حالت $\tilde{q} = [\tilde{x} \ \tilde{y} \ \tilde{\psi} \ \tilde{\varphi}]^T$ به صورت:

$$\tilde{q} = q_d - q = \begin{bmatrix} x_d - x \\ y_d - y \\ \psi_d - \psi \\ \varphi_d - \varphi \end{bmatrix} \quad (7-6)$$

تعریف می‌گردد. با استفاده از معادله سینماتیکی (6-4) می‌توان نوشت:

$$\begin{aligned} \dot{x}_c &= v_d \cos(\psi) \\ \dot{y}_c &= v_d \sin(\psi) \\ \dot{\psi}_c &= \omega_d \end{aligned} \quad (8-6)$$

که v_d و ω_d مقادیر مطلوب سرعت خطی و سرعت دورانی ربات هستند که در امر ردگیری مقادیر آنان وابسته به زمان می‌باشد.

حال می‌بایست به بیان دو قانون کنترل برای C_ψ و C_φ پردازیم به گونه‌ای که با رفتن $t \rightarrow \infty$ مقادیر $\tilde{x} \rightarrow 0$ ، $\tilde{y} \rightarrow 0$ ، $\tilde{\psi} \rightarrow 0$ میل کرده و مقدار φ نیز روی صفر باقی بماند.

برای این امر ابتدا خطای محاسبه‌شده را توسط بردار دوران به فضای ربات انتقال می‌دهیم که در این صورت داریم:

$$e = \begin{bmatrix} e_1 \\ e_2 \\ e_3 \end{bmatrix} = \begin{bmatrix} \cos \psi & \sin \psi & 0 \\ -\sin \psi & \cos \psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \tilde{x} \\ \tilde{y} \\ \tilde{\psi} \end{bmatrix} \quad (9-6)$$

با مشتق گیری از بردار خطا مقادیر زیر به دست می آید.

$$\begin{aligned} \dot{e}_1 &= \omega e_2 - v + v_d \cos e_3 \\ \dot{e}_2 &= -\omega e_1 + v_d \sin e_3 \\ \dot{e}_3 &= \omega_d - \omega \end{aligned} \quad (10-6)$$

حال نیاز به یک متغیر کمکی است که به صورت زیر تعریف می گردد.

$$\bar{e}_3 = e_3 + k_3 e_2 \quad (11-6)$$

در زمان صفر نبودن φ ، مشتق (11-6) به دست می آید:

$$\dot{\bar{e}}_3 = \dot{e}_3 + k_3 \dot{e}_2 = \omega_d - \omega + k_3 (-\omega e_1 + v_d \sin e_3) \quad (12-6)$$

به منظور پایدارسازی سیستم، قوانین کنترل v, ω به صورت زیر انتخاب می گردند.

$$\begin{aligned} \omega_\varphi &\triangleq \beta_1 = -k_\alpha \alpha \\ v &= \beta_2 = k_1 e_1 + v_d \cos e_3 \\ \omega &= \beta_3 = \frac{1}{1+k_3 e_1} (k_2 \bar{e}_3 + v_d \frac{\sin e_3}{\bar{e}_3} e_2 + \varphi v_d \sin e_3 + \omega_r) \end{aligned} \quad (13-6)$$

بنابراین خطای حلقه بسته ی سیستم به دست می آید.

$$\begin{aligned} \dot{e}_1 &= -k_1 e_1 + \omega e_2 \\ \dot{e}_2 &= -\omega e_1 + v_r \sin e_3 \\ \dot{\bar{e}}_3 &= -k_2 \bar{e}_3 - v_r \frac{\sin e_3}{\bar{e}_3} e_2 \end{aligned} \quad (14-6)$$

برای اثبات پایداری خطای حلقه بسته، یک تابع لیاپانوف محدود به صورت زیر پیدا می شود.

$$V_1 = (e_1^2 + e_2^2 + \bar{e}_3^2) / 2 \quad (15-6)$$

که مشتق رابطه ی فوق $V_1 = -k_1 e_1^2 - k_2 \bar{e}_3^2 \leq 0$ است و نیمه معین منفی^۱ به حساب می آید. بر اساس

لم باربالات^۲ و اصل لاسال^۳، مقادیر e_1, \bar{e}_3 با رفتن زمان به سمت بی نهایت، به صفر میل می کند.

¹ Semi negative definite

² Barbalat's lemma

³ LaSalle's Principle

حال به طراحی کنترل کننده‌ی مد لغزشی می‌پردازیم که با استفاده از آن بدنه‌ی ربات سرعت زاویه‌ای مجازی β_3 را دنبال نماید. این کنترل کننده بر اساس رابطه‌ی ساده و تفکیک‌شده‌ی حرکت دورانی بدنه، طراحی خواهد شد.

$$\dot{\omega} = f_{23} + g_{22}C_y \quad (16-6)$$

صفحه لغزش به صورت زیر پیشنهاد می‌گردد.

$$S_\eta = \omega - \beta_3 \quad (17-6)$$

مشتق صفحه‌ی لغزش به دست می‌آید

$$\dot{S}_\eta = \dot{\omega} - \dot{\beta}_3 \quad (18-6)$$

با جایگذاری (16-6) در (17-6) داریم:

$$\dot{S}_\eta = f_{23} + g_{33}C_y - \dot{\beta}_3 \quad (19-6)$$

که نهایتاً قانون کنترل C_y برای $S_\eta \rightarrow 0$ به صورت زیر پیشنهاد می‌گردد.

$$C_y = g_{23} \left(-f_{23} + \dot{\beta}_3 - k_{\eta 1} \text{sgn}(S_\eta) - k_{\eta 2} S_\eta \right) \quad (20-6)$$

حال لازم است تا به طراحی کنترل کننده‌ای برای کنترل گشتاور C_θ بپردازیم.

به روشی مشابه، $S_\alpha = \omega_\alpha - \beta_1$ و $S_v = v - \beta_2$ معرفی می‌شوند. و این بار معادله سطح لغزش به

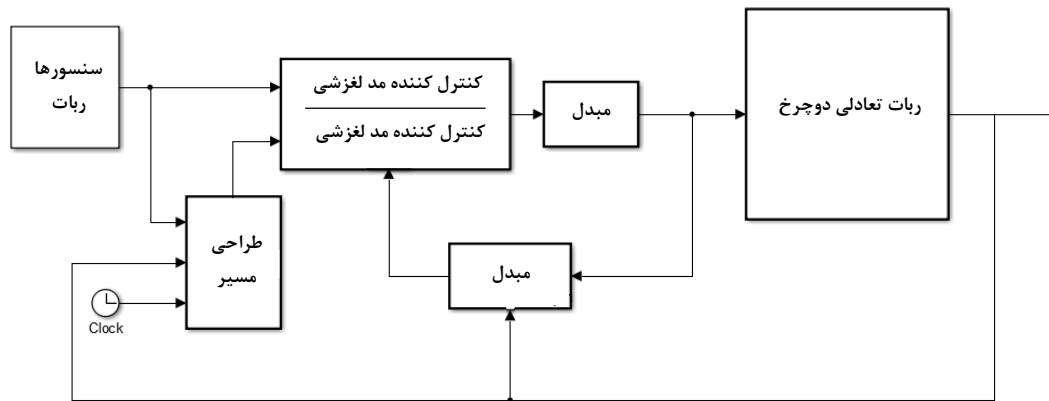
صورت (21-6) معرفی می‌شود.

$$S_1 = r_1 S_v + r_2 S_\alpha \quad (21-6)$$

لذا می‌توان به پیشنهاد قانون کنترل C_θ به صورت زیر پرداخت.

$$C_\theta = \frac{1}{r_1 g_{11}} \left(-r_1 f_{22} + r_1 \dot{\beta}_2 - r_2 \dot{S}_\alpha - k_{s1} \text{sgn}(S_1) - k_{s2} S_1 \right) \quad (22-6)$$

بر اساس طراحی انجام شده، کنترل کننده‌ی ما از دو بخش مجزا تشکیل شده که قوانین مربوط به هر کدام در (۶-۲۰) و (۶-۲۲) بیان شده است. بلوک دیاگرام کنترلی استفاده در این روش برای کنترل ربات را می‌توان به صورت شکل (۶-۲) در نظر گرفت.

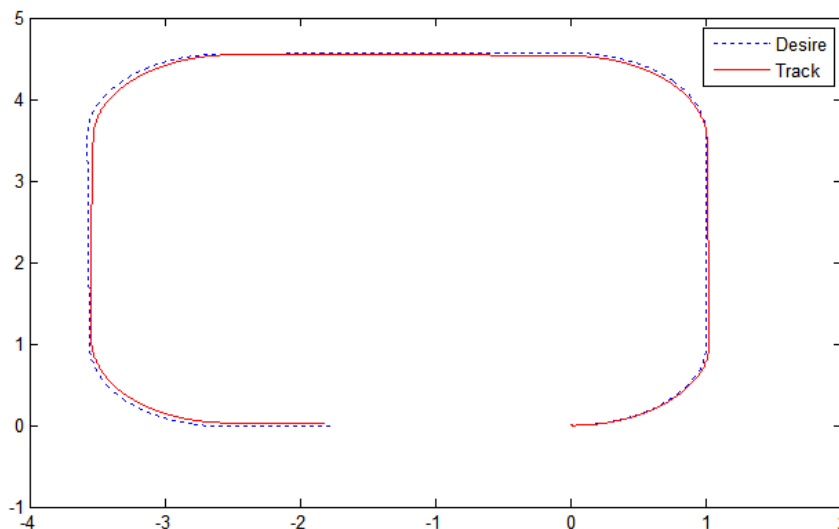


شکل ۶-۲: بلوک دیاگرام کنترل کننده مد لغزشی برای کنترل مسیر

در صورتی که کنترل کننده مد لغزشی طراحی شده را برای پارامترهای مطرح شده در جدول (۶-۱) انجام دهیم. نتیجه ردگیری در محیط شبیه سازی به صورت شکل (۶-۳) خواهد بود.

جدول ۶-۱: پارامترهای طراحی کنترل کننده مد لغزشی

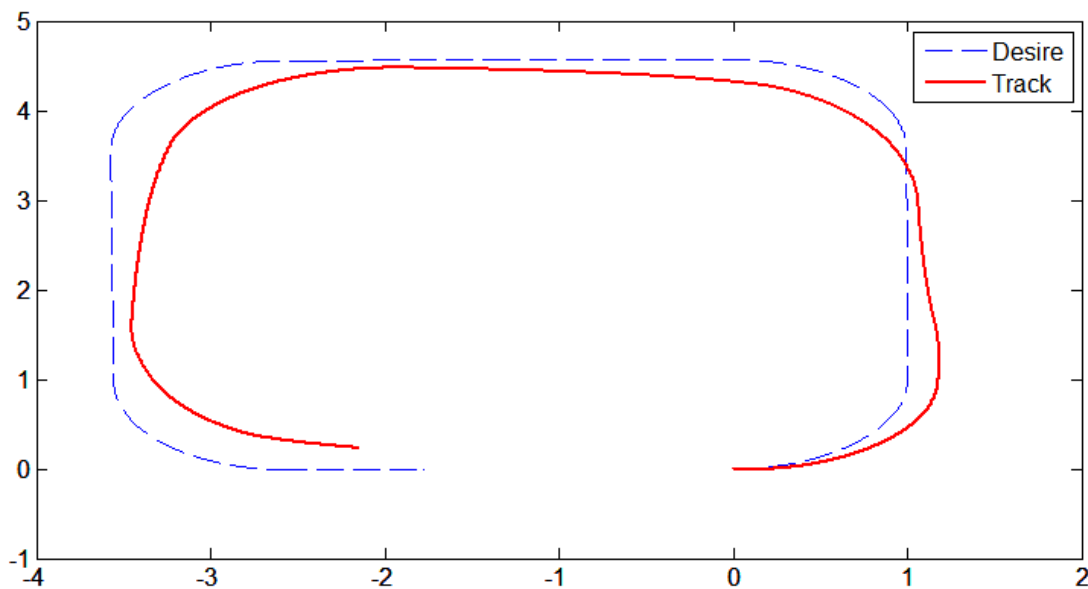
$r_1 = 1$	$k_{\eta_1} = k_{\eta_2} = 10$	$k_\alpha = 2$	$k_2 = 0.3$
$r_2 = 0.35$	$k_{s_1} = k_{s_2} = 10$	$k_1 = 0.2$	$k_3 = 0.75$



شکل ۶-۳: نتیجه ردگیری کنترل کننده مد لغزشی در محیط شبیه سازی

بر اساس این شکل می‌توان نتیجه‌گیری کرد که ربات ردگیری مناسبی داشته و توانسته از مسیر مطلوب پیروی نماید، لازم به ذکر است پارامترهای کنترل‌کننده در این حالت به صورت جدول (۶-۱) در نظر گرفته شده است.

در صورتی که همین کنترل‌کننده را بر روی ربات واقعی تست کنیم نتایج به هیچ‌وجه مطلوب نبوده و به فرم شکل (۶-۴) خواهد بود.



شکل ۶-۴: نتیجه ردگیری کنترل‌کننده مدلفزشی برای ربات واقعی

آنچه در شکل (۶-۴) مشاهده می‌کنیم؛ نشان از عدم توانایی کنترل‌کننده در دنبال کردن مسیر مطلوب دارد. اگرچه ربات توانسته با تقریب، نسبت به دنبال کردن مسیر اصلی اقدام نماید. اما در نهایت خطای بزرگی ایجاد شده که این ردگیری را از مرتبه اطمینان خارج می‌سازد.

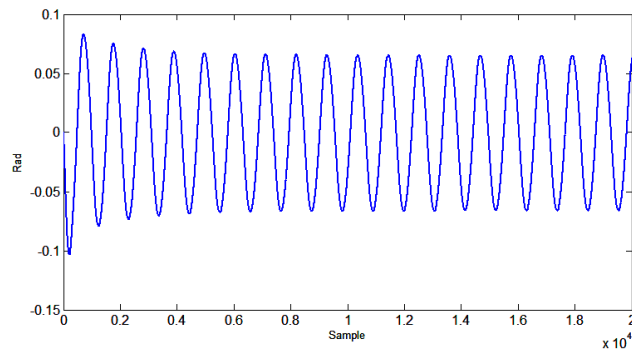
برای ارزیابی روش‌ها، میانگین خطای ردگیری به صورت

$$e_{ir} = \frac{1}{lT} \sum_{n=1}^T \sqrt{e_x^2(n) + e_y^2(n)} \quad (۶-۲۳)$$

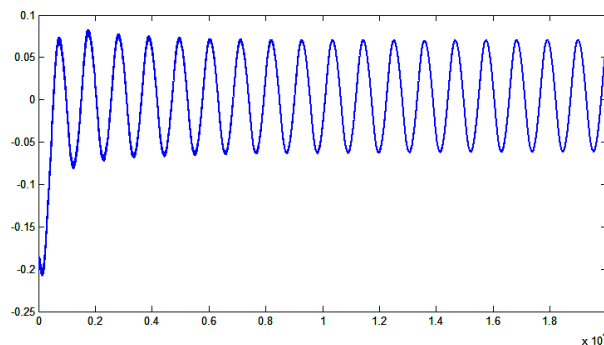
تعریف می‌گردد.

بر این اساس خطای ردگیری برای شکل (۳-۶) در حدود 1×10^{-2} و برای شکل (۴-۶) در حدود ۰,۷ محاسبه می‌گردد.

از طرفی علاوه بر ردگیری انجام‌شده، زاویه بدنه ربات نیز بایستی مورد بررسی قرار گیرد، که این مورد برای دو حالت شبیه سازی و عملی در شکل (۵-۶) و (۶-۶) نمایش داده شده است



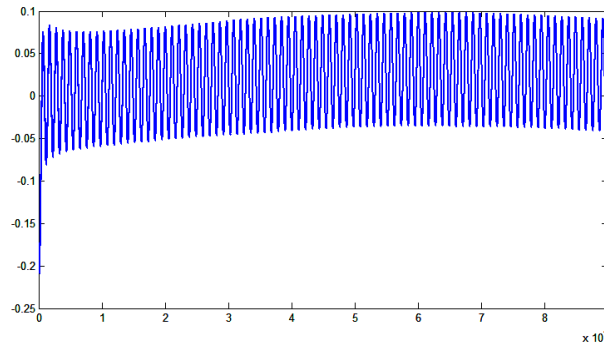
شکل ۵-۶: انحراف زاویه بدنه برای کنترل کننده مد لغزشی در محیط شبیه سازی



شکل ۶-۶: انحراف زاویه بدنه برای کنترل کننده مد لغزشی برای ربات واقعی

تفاوت عمده‌ی مشاهده شده در این دو سیگنال که یکی از محیط شبیه سازی و دیگری از انتگرال گیری از سیگنال سرعت زاویه‌ای بدنه‌ی ربات به دست می‌آید، مربوط به لحظه‌ی راه‌اندازی ربات است که بستگی به شرایط اولیه و همچنین لقی موجود در موتورها دارد.

اما اگر سیگنال شکل (۶-۶) را در بازه زمانی بزرگ‌تری مورد مطالعه قرار دهیم متوجه ایجاد یک خطای دائمی در اندازه زاویه پاندول می‌شویم که ناشی از خطای انتگرال گیری می‌باشد. این امر در شکل (۷-۶) نمایش داده شده است.



شکل ۶-۷: نمایش خطای دائمی زاویه‌ی بدنه ربات ایجاد شده بر اثر خطای انتگرال گیری

برای رفع این مشکل، می‌توان از حسگر زاویه کمک گرفت، این حسگر اگر چه دارای دقت بالایی است اما سرعت پاسخ کمی دارد، لذا نمی‌توان آن را جایگزین ژيروسکوپ کرد بلکه تنها برای اصلاح خطای انتگرال گیری از آن استفاده می‌کنیم. برای این امر فرض می‌کنیم که حسگر زاویه در هر ثانیه n_s و انتگرال‌گیری از سیگنال ژيروسکوپ، n_g نمونه در ثانیه تولید نماید. برای اصلاح مقدار خطای انتگرال‌گیری، شیب خط گذرنده از میانگین نمونه‌های سیگنال خروجی حسگر را با شیب به دست آمده از سیگنال حسگر زاویه، مقایسه می‌کنیم و خطای به وجود آمده را به صورت

$$M_g = \frac{1}{n_g} \sum_{i=1}^{n_g} (n_g^i - n_g^{i-1})$$

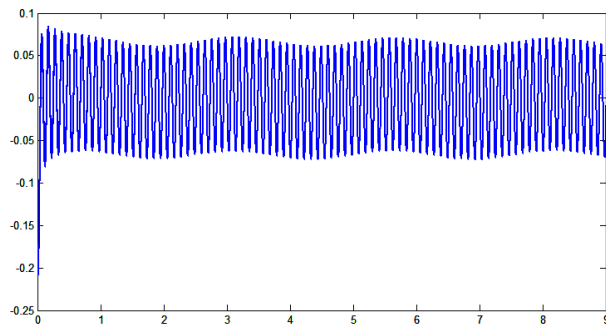
$$M_s = \frac{1}{n_s} \sum_{i=1}^{n_s} (n_s^i - n_s^{i-1}) \quad (۲۴-۶)$$

$$e_M = M_g - M_s$$

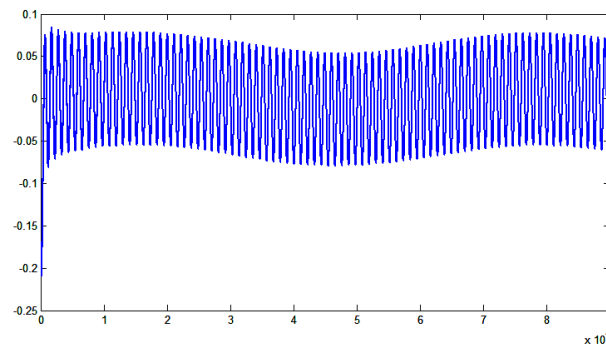
محاسبه می‌نماییم، حال اگر مطابق (۲۵-۶) ضربی از خطای به دست آمده را با سیگنال φ_g حاصل از انتگرال‌گیری جمع کنیم، این امر موجب اصلاح تدریجی خطای انتگرال‌گیری می‌گردد.

$$\varphi = \varphi_g + k_q e_M \quad (۲۵-۶)$$

لازم به ذکر است این فرآیند می‌بایست در هر بازه‌ی زمانی D_t ، از سرگیری شود. نتایج به دست آمده برای چند k_q مختلف و در بازه‌های نمونه‌برداری مختلف در تصاویر (۶-۸) و (۶-۹) نمایش یافته است.

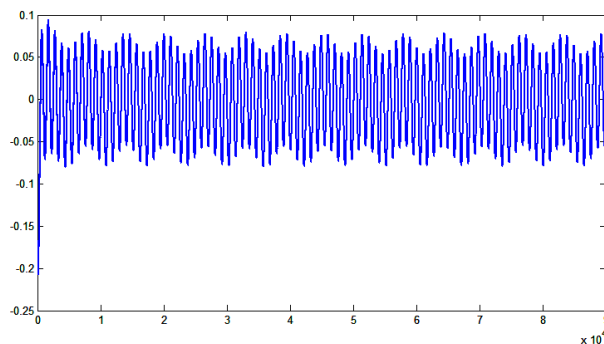


شکل ۶-۹: چگونگی اصلاح خطا با ضریب اصلاح $k_q = 0.1$



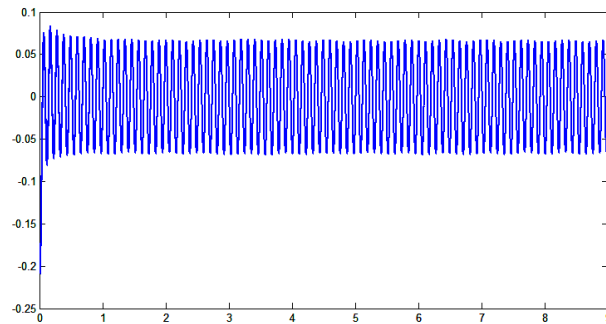
شکل ۶-۸: چگونگی اصلاح خطا با ضریب اصلاح $k_q = 0.05$

همان‌گونه که در تصاویر مشاهده می‌گردد، با افزایش ضریب، میزان اصلاح افزایش یافته و زاویه φ با دقت بالاتری اندازه‌گیری می‌شود، اما زیاد کردن بیش از حد مقدار k_q باعث نوسانی شدن سیگنال φ می‌گردد. این امر در شکل (۶-۱۰) برای $k_q = 15$ نشان داده شده است.



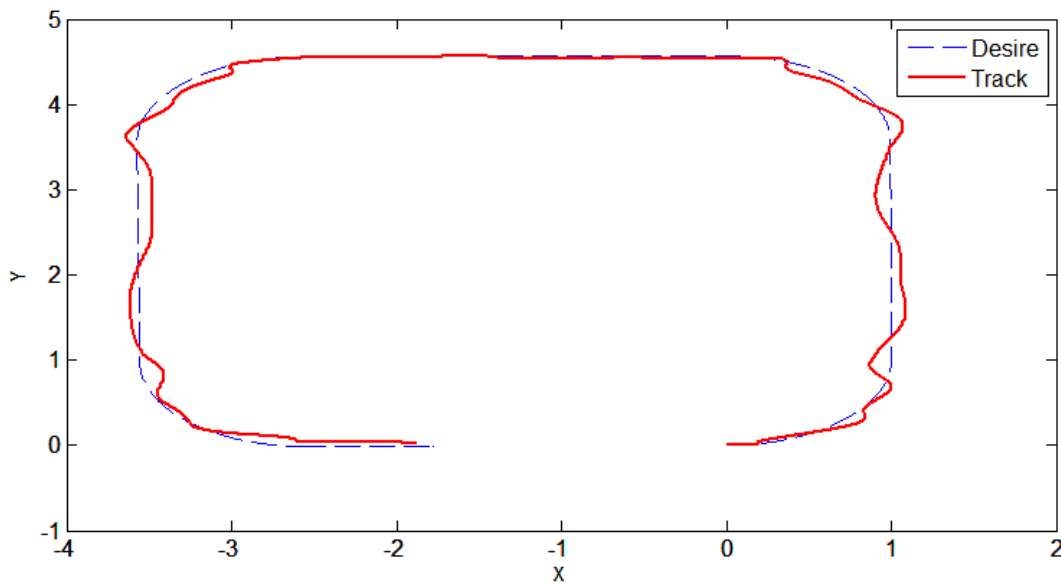
شکل ۶-۱۰: ایجاد نوسانات زیاد در سیگنال زاویه، در صورت انتخاب ضریب تصحیح بزرگ

در این پژوهش و بر اساس تجربیات عملی مقدار بهینه‌ی $k_q = 0.3$ و $D_T = 300ms$ پیشنهاد می‌گردد که در این صورت و برای حرکت روی مسیر تعیین‌شده، نوسانات بدنه، در شکل (۶-۱۱) نشان داده شده است.



شکل ۶-۱۱: سیگنال اصلاح شده با انتخاب ضریب اصلاح مناسب

حال با توجه به اصلاح صورت گرفته، نتایج ردگیری را برای ربات تکرار می‌کنیم.



شکل ۶-۱۲: نتیجه ردگیری برای کنترل کننده مد لغزشی پس از اصلاح خطای انتگرال گیری

در این حالت خطای ردگیری به ۰/۰۵ کاهش می‌یابد.

اگر فرض کنیم مقدار حاصل، حدی مطلوب برای خطای مسیریابی است؛ بازهم وجود نوسانات زیاد در بدنه‌ی ربات که در حد ۱۰ درجه می‌باشد، امر مطلوبی به نظر نمی‌رسد.

لذا، کنترل کننده مد لغزشی را کنار گذاشته و به سراغ طراحی یک شبکه عصبی - فازی برای کنترل ربات می‌رویم.

۶-۴- طراحی کنترل کننده عصبی - فازی:

پیش از این در مورد نحوه‌ی طراحی و جزئیات کنترل کننده عصبی-فازی در فصل ۳ مفصلاً توضیح داده شد. شبکه عصبی شکل (۳-۵) را در نظر بگیرید. می‌خواهیم از این شبکه به عنوان کنترل کننده سیستم استفاده کنیم. برای این منظور بردار خطای به دست آمده از معادله (۶-۱۰) و مشتقات خطای به دست آمده از (۶-۱۱) را به عنوان ورودی‌های شبکه عصبی در نظر می‌گیریم. با کمتر شدن خطای تعریف شده از مقدار d_{th} تعریف شده، اصلاح وزن و اصلاح توابع عضویت فازی متوقف شده و این نشان از پایداری سیستم دارد. بر این اساس کافی است مقدار مطلوب برای φ را صفر و سایر مقادیر مطلوب را از معادله طراحی مسیر و با کمک گرفتن از (۶-۲۵) به دست آوریم.

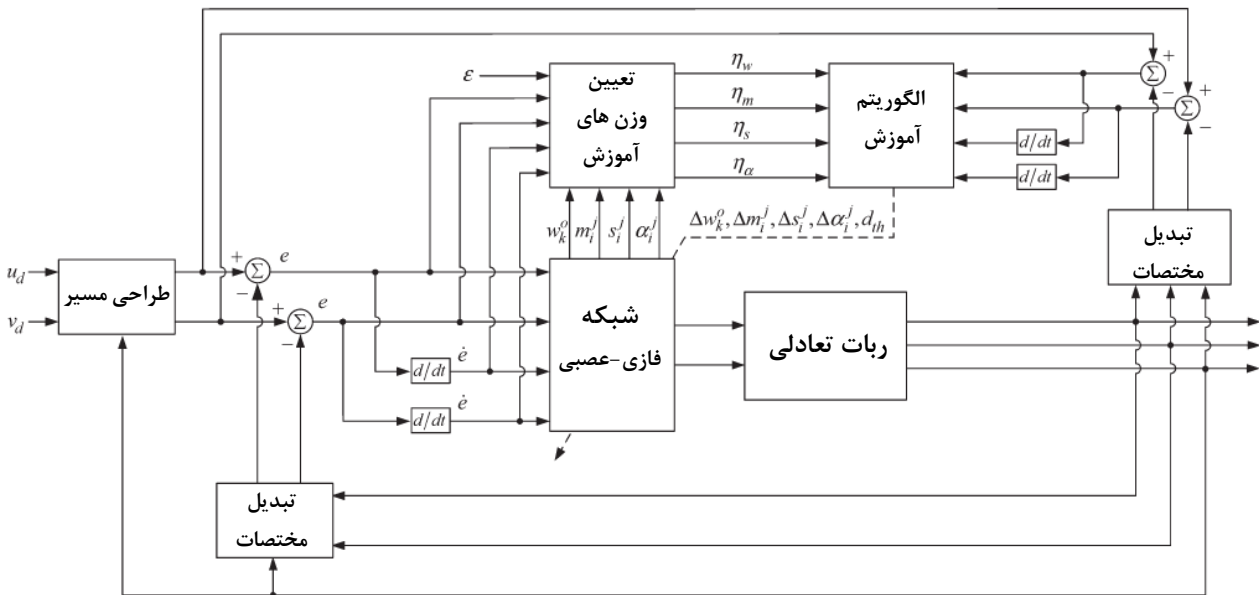
$$\begin{bmatrix} \dot{x}_d \\ \dot{y}_d \\ \dot{\psi}_d \\ \dot{\phi}_d \end{bmatrix} = \begin{bmatrix} \cos \psi_d & 0 & 0 \\ \sin \psi_d & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} v_d \\ \dot{\psi}_d \\ \dot{\phi}_d \end{bmatrix} \quad (۶-۲۶)$$

لازم به ذکر است، به علت پیاده‌سازی شبکه عصبی به صورت زمان گسسته، معادلات جابه‌جایی زمان گسسته مورد نیاز است که در (۴-۱۱) به بیان آن پرداخته شد. بنا بر توصیفات صورت گرفته، بلوک دیاگرام کنترل کننده عصبی - فازی به صورتی است که در شکل (۶-۱۳) مشاهده می‌کنید.

جدول ۶-۲: پارامترهای طراحی برای کنترل کننده عصبی - فازی

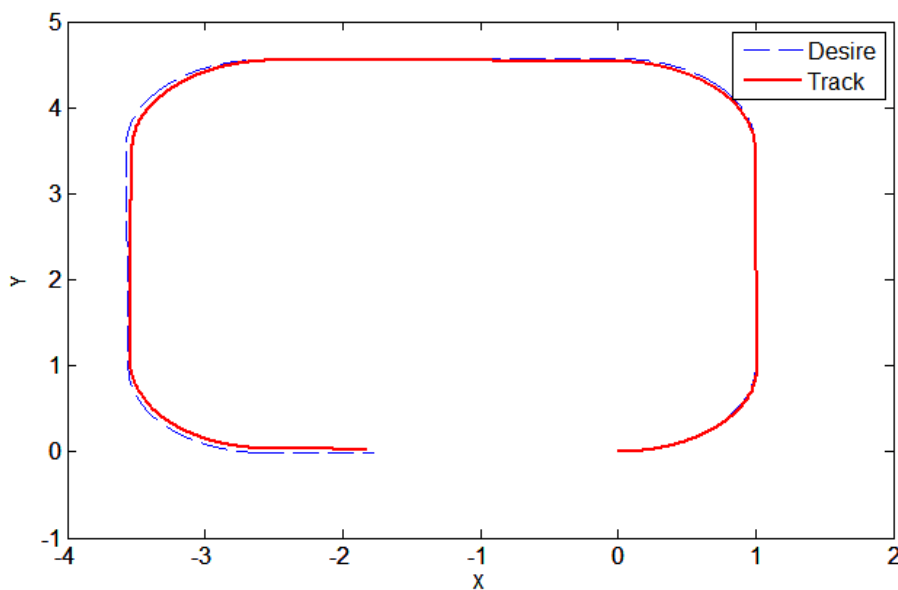
$\varepsilon = 0.5$	$\Delta t = 0.1s$
$n_i = 8$	$n_o = 2$

پارامترهای طراحی شبکه را به صورت جدول (۳-۶) طراحی کرده و به شبیه‌سازی عملکرد ربات می‌پردازیم.



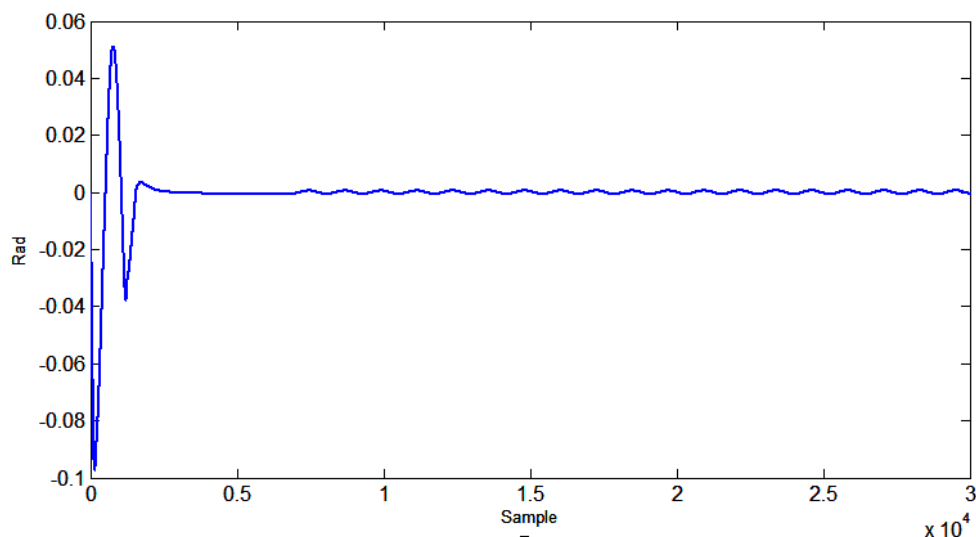
شکل ۱۳-۶: بلوک دیاگرام شبکه عصبی-فازی برای کنترل ربات

در حالت اول، کنترل‌کننده را روی مدل رایانه‌ای شبیه‌سازی می‌کنیم که نتیجه آن در شکل (۶-۱۴) نشان داده شده است.



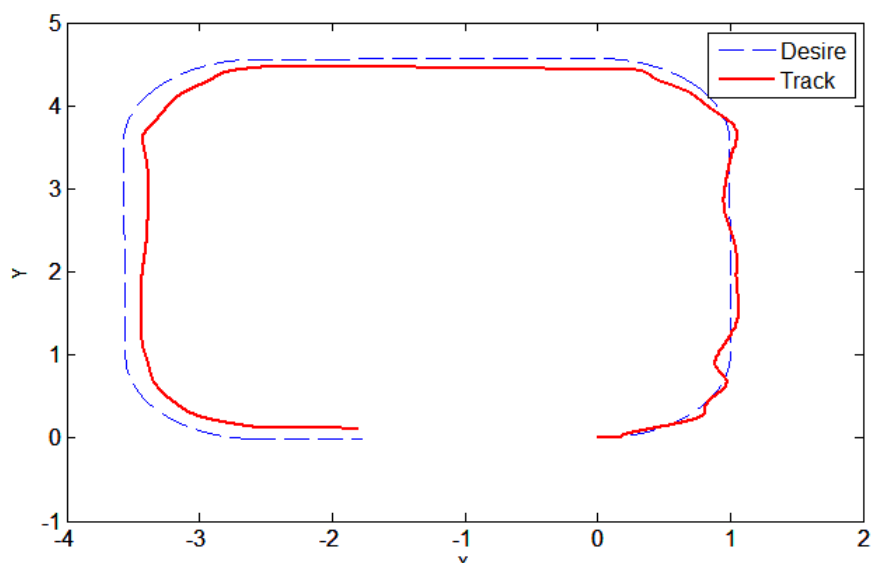
شکل ۱۴-۶: نتیجه شبیه‌سازی ردگیری با استفاده از شبکه عصبی فازی

خطای ردگیری در این حالت 0.003 است که نتیجه‌ای بهتر از کنترل مد لغزشی به حساب می‌آید. اگر نگاهی به نوسانات بدنه در این حالت بیندازیم، نوسانات بدنه نیز در مقایسه با روش قبل به شدت کاهش یافته است، شکل (۶-۱۵) این نوسانات را نشان می‌دهد.

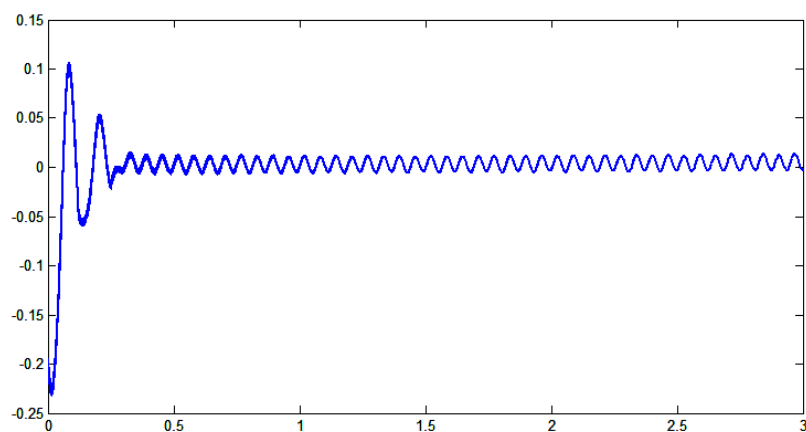


شکل ۶-۱۵: نوسانات بدنه با استفاده از شبکه عصبی فازی در محیط شبیه سازی

در پیاده‌سازی عملی، نتایج به کلی متفاوت است. این در تصاویر (۶-۱۶) و (۶-۱۷) قابل مشاهده می‌باشد. لازم به ذکر است در پیاده‌سازی عملی، اصلاح لازم برای تصحیح خطای انتگرال‌گیری که در قسمت قبل به آن پرداخته شد، صورت گرفته است.



شکل ۶-۱۶: نتیجه ردگیری در استفاده از کنترل کننده عصبی- فازی برای ربات واقعی



شکل ۶-۱۷: نوسانات بدنه در پیاده سازی عملی کنترل کننده‌ی فازی - عصبی

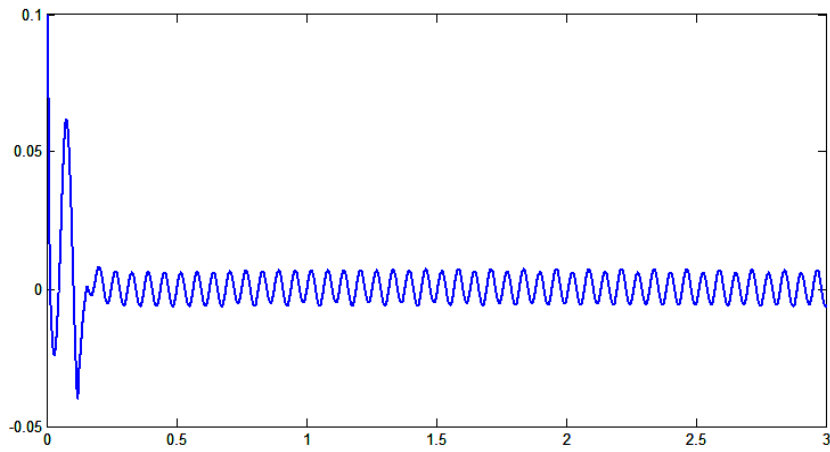
در این حالت خطای ردگیری در حدود ۰/۱ است. اما در ابتدای حرکت ربات، شاهد نوسانات شدید بدنه (در حدود ۲۰ درجه) هستیم که به علت عدم آموزش شبکه صورت می‌گیرد. این نوسانات ممکن است باعث برهم خوردن تعادل ربات و قرار گرفتن ربات در وضعیتی غیرقابل کنترل شود.

نتایج مسیریابی برای انتخاب‌های مختلفی از n_j (تعداد سلول‌های لایه‌ی قوانین) تکرار شده که در زیر در جدول (۴-۶) قابل مشاهده است.

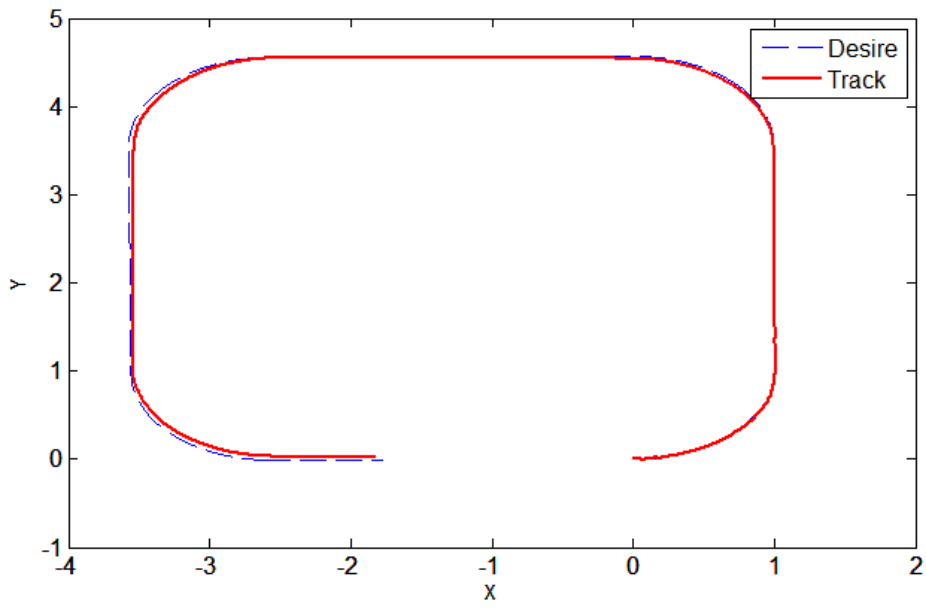
جدول ۳-۶: نتایج حاصل از شبیه سازی کنترل کننده با تعداد مختلفی گره در لایه‌ی قوانین

تعداد n_j	میانگین مربع خطای ردگیری	بیش‌ترین نوسان بدنه
۵	۰/۱	۲۰ درجه
۷	۰/۰۳	۱۱ درجه
۹	۰/۰۰۱	۱۰ درجه
۱۲	۰/۰۰۴	۱۳ درجه
۱۵	۰/۰۲	۱۳ درجه

بر این اساس بهترین تعداد برای لایه‌ی قوانین، ۹ عدد می‌باشد که برای این تعداد، نتایج عملی تکرار شده و در شکل (۶-۱۷) و (۶-۱۸) ارائه گردیده است.



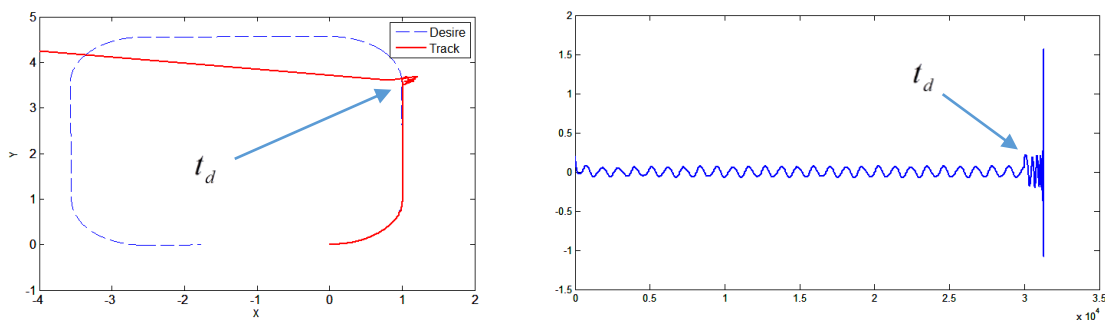
شکل ۶-۱۹: نمایش نوسانات بدنه در حالت استفاده از ۹ گره در لایه ی قوانین



شکل ۶-۱۸: نمایش مسئله ردگیری برای ۹ گره در لایه ی قوانین

اگرچه نتایج به دست آمده ، نشان‌دهنده‌ی توانای کنترل‌کننده عصبی- فازی در حفظ تعادل ربات و پیمایش مسیر مربوطه است. اما توجه به این نکته حائز اهمیت است که اگر اغتشاش یا نیروی خارجی به ربات وارد شود، به علت مدت زمان اندکی که برای اصلاح ساختار شبکه وجود دارد، سیستم از حالت تعادل خارج شده و ادامه‌ی فرآیند کنترلی انجام نمی‌شود. در واقع ساختار کوچک و سبک ربات زحل باعث شده تا حفظ تعادل آن نیاز به واکنش سریع از طرف کنترل‌کننده داشته باشد.

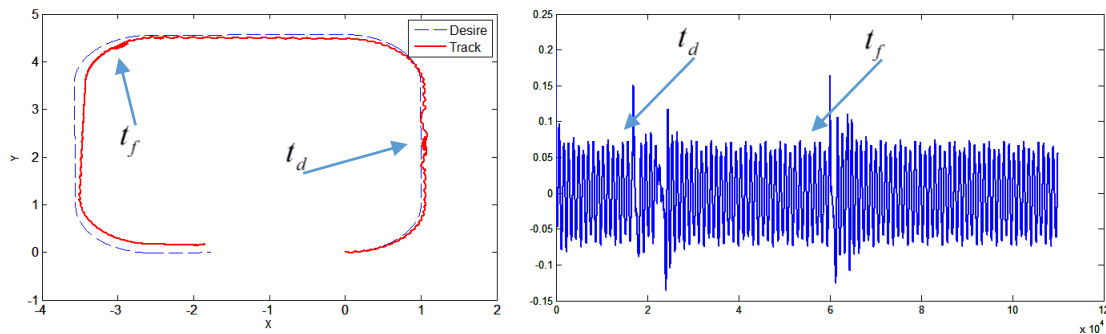
برای اثبات این موضوع در میان مسیر ، یک نیروی خارجی توسط دست به سیستم اعمال می‌شود، و در ادامه‌ی مسیر نیز مانع کوچکی به ارتفاع $2mm$ قرار داده‌شده و نتایج سیستم برای دو حالت کنترل مد لغزشی و شبکه عصبی- فازی با یکدیگر مقایسه شده است.



شکل ۶-۲۰: پاسخ کنترل‌کننده عصبی- فازی به ورود اغتشاشات نابهنگام

شکل (۶-۲۰) انحراف بدنه و مسیر حرکت ربات برای کنترل‌کننده‌ی فازی عصبی با وجود اغتشاشات خارجی را نشان می‌دهد، در لحظه‌ی t_d ضربه‌ای کوچکی توسط دست به ربات وارد شده که باعث برهم خوردن تعادل آن می‌گردد. در واقع به علت عدم خروج ربات از نقطه‌ی تعادل خود، شبکه‌ی استفاده‌شده تنها برای بازه‌ای حول نقطه‌ی تعادل آموزش می‌بیند، لذا اگر یک عامل خارجی باعث فاصله گرفتن ربات از نقطه‌ی تعادل آن شود، فرآیند آموزش شبکه از سر گرفته می‌شود، اما برای این آموزش مدت زمان نسبتاً زیادی لازم است که متأسفانه در اختیار شبکه قرار نمی‌گیرد، لذا ربات پس از مدت زمان مشخصی سرنگون شده و در این حالت باقی می‌ماند.

شکل (۶-۲۱) زاویه‌ی انحراف و مسیر حرکت ربات، برای کنترل‌کننده‌ی مد لغزشی را نشان می‌دهد.



شکل ۶-۲۱: پاسخ کنترل‌کننده مد لغزشی به ورود اغتشاشات ناپهنگام

اگرچه پاسخ این کنترل‌کننده در مقایسه با کنترل‌کننده‌ی عصبی-فازی دارای خطای بیشتری است، اما، به علت طراحی کنترل‌کننده‌ی مد-لغزشی بر اساس مدل دینامیکی سیستم، دارای پایداری بهتری در مواجهه با اغتشاشات خارجی است، همان‌گونه که در شکل (۶-۲۱) مشخص است، نیروی مشابهی در t_d توسط دست به ربات اعمال می‌شود، همچنین در نقطه‌ی t_f یک مانع کوچک سر راه ربات قرار گرفته که ربات باید بتواند از آن عبور کند.

تحلیل انجام‌شده نشان می‌دهد کنترل‌کننده مد لغزشی علی‌رغم وجود نوسانات بالاتر نسبت به کنترل‌کننده‌ی عصبی در مواجهه با اغتشاشات خارجی بهتر عمل می‌نماید.

لازم به توضیح است به علت پیاده‌سازی عملی این مراحل، امکان برقراری شرایط دقیقاً یکسان برای هر دو کنترل‌کننده وجود نداشته اما سعی شده است تا ربات در شرایطی یکسان مورد آزمایش قرار گیرد.

۶-۵- تلفیق کنترل‌کننده‌ی مد لغزشی و کنترل‌کننده عصبی - فازی

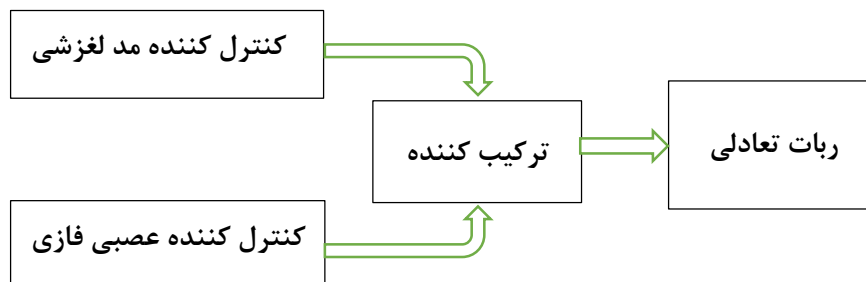
اگر بخواهیم مزایای کنترل‌کننده‌ی عصبی فازی و قدرت مد لغزشی در مبارزه با اغتشاشات را در کنار هم قرار دهیم. باید به طراحی ترکیبی از این کنترل‌کننده‌ها بپردازیم.

برای این منظور پیشنهاد می‌شود تا از هر دو کنترل‌کننده در کنار هم بهره برده و خروجی آنان را به نحوی تلفیق کنیم که بتواند به کنترل سیستم بپردازد.

بر اساس نتایج به دست آمده در مرحله قبل، اگر بتوان ساختاری ارائه داد که در مواقع بروز اغتشاشات خارجی، مد لغزشی و در شرایط عادی شبکه عصبی فازی را در حلقه‌ی کنترل قرار دهد. به این هدف دست یافته‌ایم.

اما نباید فراموش کرد که این جابه‌جایی می‌بایست خیلی نرم و بدون سوئیچ زنی انجام شود. برای این کار، بردار خروجی کننده‌ی مد لغزشی را $O_s = [\tau_{rs} \quad \tau_{ls}]^T$ و بردار خروجی شبکه کنترلی را $O_N = [\tau_{rN} \quad \tau_{lN}]^T$ معرفی می‌کنیم.

ترکیب‌کننده‌ای به صورت شکل (۶-۲۲) مورد استفاده قرار می‌گیرد که در ادامه به بررسی روند کاری این ترکیب‌کننده خواهیم پرداخت.



شکل ۶-۲۲: نحوه استفاده از ترکیب‌کننده برای تلفیق دو کنترل‌کننده

خروجی ترکیب‌کننده را به صورت

$$O_x = w_1 O_s + w_2 O_N \quad (۶-۲۷)$$

معرفی می‌کنیم. که w_1 و w_2 هر دو اعدادی مثبت‌اند. با این روش در واقع ترکیبی خطی از خروجی کنترل‌کننده‌ها برای کنترل خروجی به کار برده می‌شود. اگر داشته باشیم:

$$w_1 + w_2 = 1 \quad (۶-۲۸)$$

می‌توان نتیجه گرفت با تغییر یکی از وزن‌ها، وزن دیگر خودبه‌خود تغییر می‌کند. در اینجا ما وزن مرجع را w_1 فرض می‌کنیم. با توجه به ناپیوسته بودن سیستم، تغییرات w_1 را به صورت

$$\Delta w_1 = w_1(i) - w_1(i-1) \quad (۶-۲۹)$$

محاسبه می‌کنیم. برای به دست آوردن قانون اصلاح این وزن‌ها، نیاز به معرفی یک سیگنال خطا داریم. سیگنال خطای کل را به صورت زیر معرفی می‌شود.

$$error = \frac{\left(\sqrt{e_x^2 + e_y^2 + e_w^2} + \sqrt{e_\phi^2}\right)}{2} \quad (30-6)$$

همچنین تغییرات خطا به روشی مشابه به صورت زیر تعریف می‌گردد.

$$\Delta_{error} = error(n) - error(n-1) \quad (31-6)$$

حال روش اطلاع وزن را این‌گونه معرفی می‌کنیم.

$$\left\{ \begin{array}{ll} \Delta_{error} > 0 & \begin{array}{l} \text{if } \Delta w_1 > 0 \quad w_1 \downarrow \\ \text{if } \Delta w_1 < 0 \quad w_1 \uparrow \end{array} \\ \Delta_{error} < 0 & \begin{array}{l} \text{if } \Delta w_1 > 0 \quad w_1 \uparrow \\ \text{if } \Delta w_1 < 0 \quad w_1 \downarrow \end{array} \end{array} \right.$$

این قانون به بیان این مطلب اشاره دارد که در صورت افزایش تغییرات خطا، تغییرات وزن خروجی با شیبی معکوس دنبال شود. که با توجه به آن، قانون اصلاح وزن را به فرم (32-6) پیشنهاد می‌دهیم.

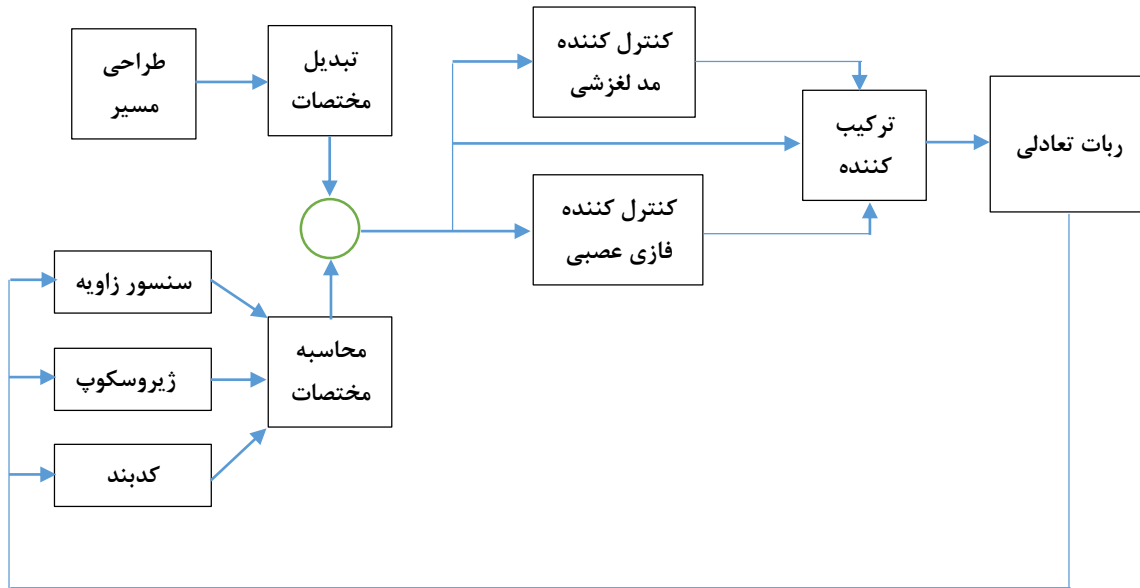
$$w_1(n+1) = w_1(n) + step \times \text{sgn}(\Delta_{error}) \quad (32-6)$$

که $step$ بیان‌کننده‌ی میزان اصلاح وزن در هر مرحله است و ممکن است با توجه به تأخیر سیستم، مقدارهای متفاوتی انتخاب شود. این پژوهش مقدار $step = 0.1$ در نظر گرفته می‌شود. w_2 نیز با قرار دادن مقدار $w_1(n+1)$ در معادله‌ی (38-6) به دست می‌آید.

اما وجود تابع $\text{sgn}(\cdot)$ در قانون اصلاح وزن، سبب نوسانات شدید وزن‌ها برای رساندن خطای خروجی به سمت صفر می‌گردد، که البته در عمل امکان‌پذیر نیست.

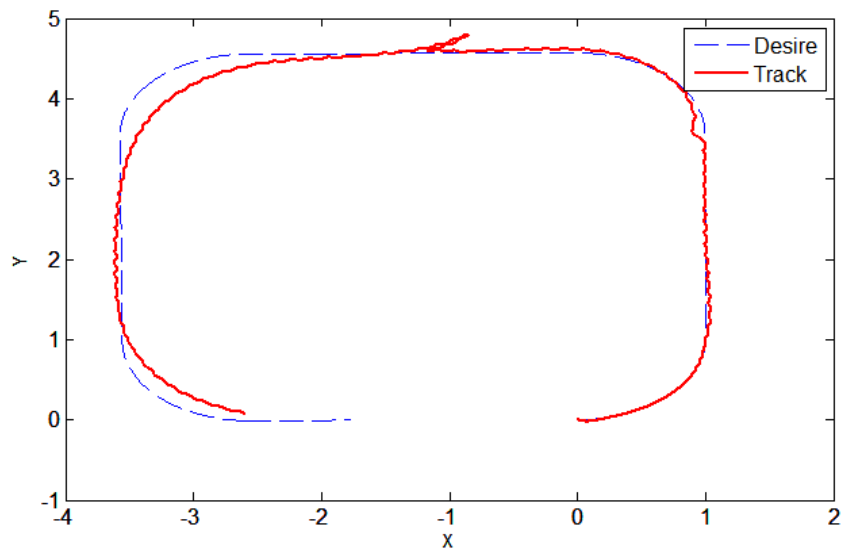
برای رفع این مشکل می‌توان یک آستانه برای خطا معرفی کرد، تا در صورت بزرگ‌تر شدن مقدار خطای خروجی از این آستانه، اصلاح وزن صورت گیرد. مقدار این آستانه برای نمونه‌ی عملی طراحی شده برابر 0/1 می‌باشد.

با این اوصاف، بلوک دیاگرام نهایی کنترل کننده‌ی استفاده‌شده، به شکل (۶-۲۲) می‌باشد.

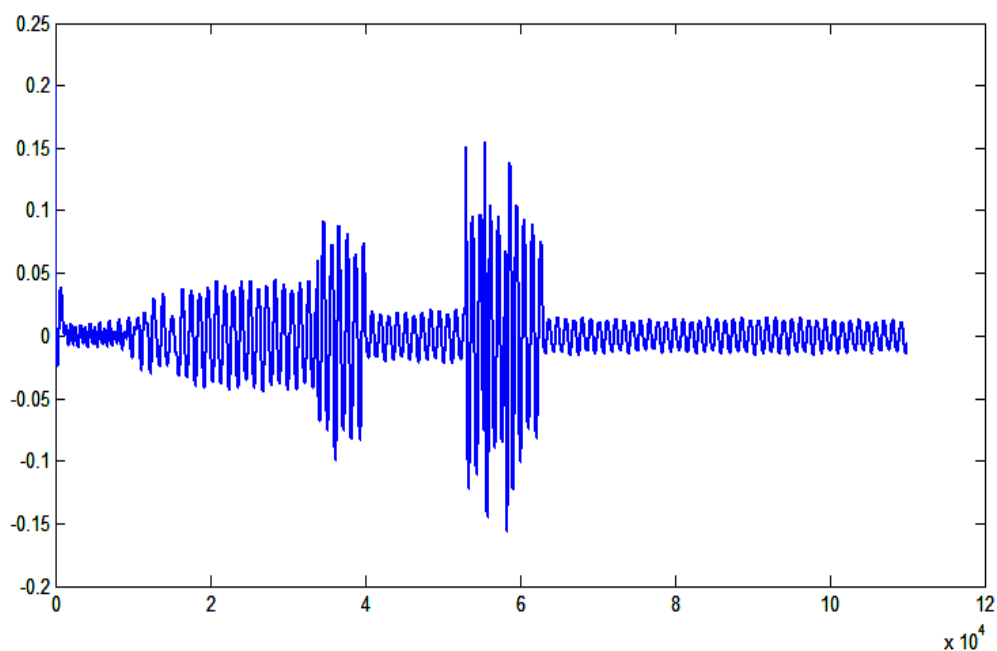


شکل ۶-۲۳: بلوک دیاگرام کنترل کننده‌ی ترکیب شده‌ی پیشنهادی

نتایج استفاده از این کنترل کننده برای کنترل مسیر ربات زحل، نشان‌دهنده‌ی قدرت مناسب کنترل کننده در مقابله با اغتشاشات خارجی و همچنین برقراری تعادل در حالت عادی است.



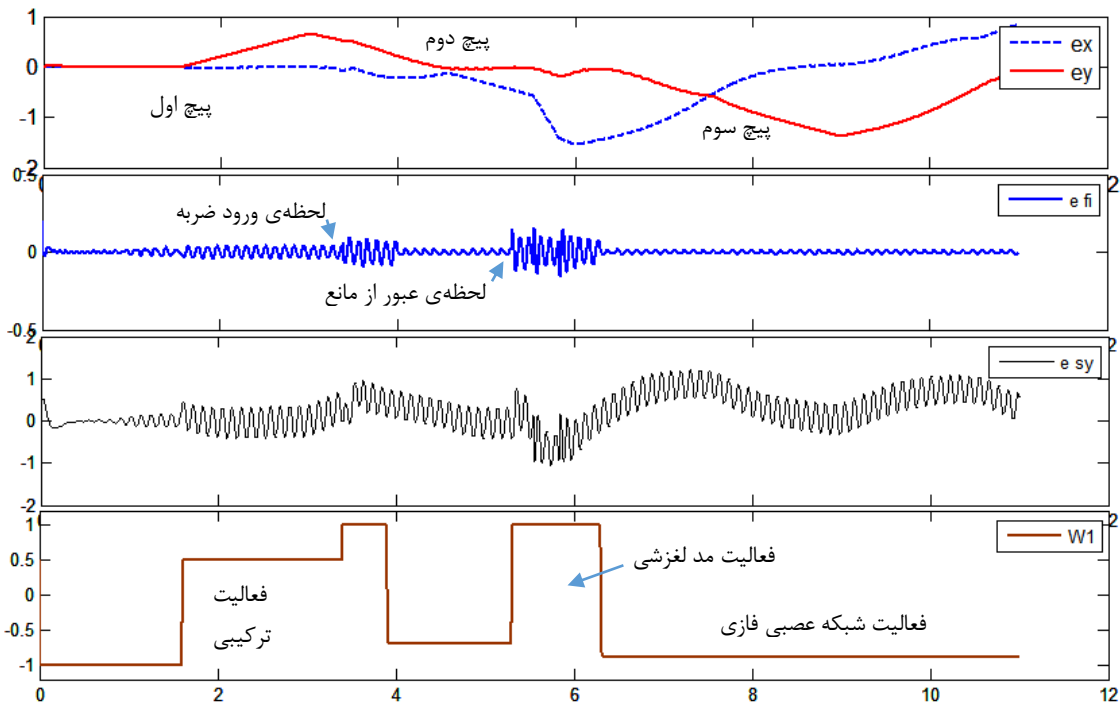
شکل ۶-۲۴: ردگیری با استفاده از کنترل کننده ترکیبی در حضور اغتشاشات



شکل ۶-۲۵: نوسانات بدنه در حضور اغتشاش خارجی و با بهره‌گیری از کنترل کننده ترکیبی

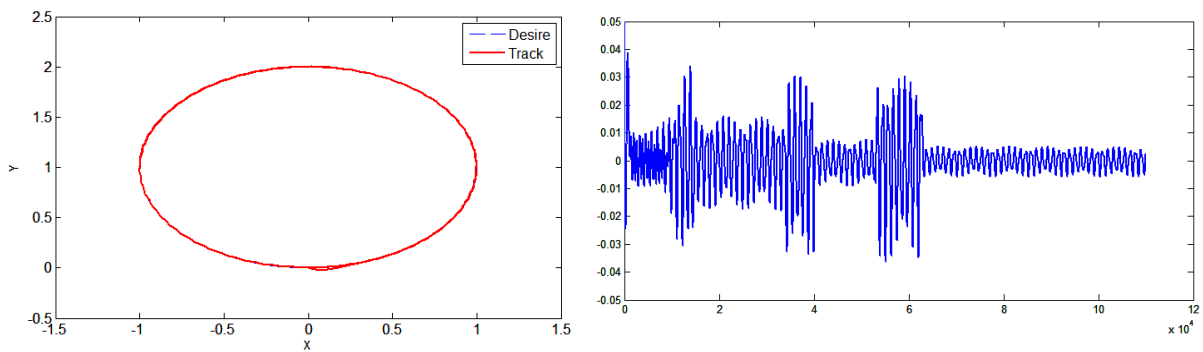
شکل (۶-۲۴) نشان‌دهنده‌ی وضعیت ردگیری برای مسیر مکعبی با وجود اغتشاشات خارجی است. در شکل (۶-۲۴) و (۶-۲۵)، تأثیر اغتشاش خارجی بر انحراف مسیر و نوسانات به وجود آمده در بدنه، به وضوح قابل مشاهده است.

اگر چه میزان مربع خطا در این روش بدون حضور اغتشاش 0.002 و با حضور آن 0.01 به دست می‌آید، اما نمایش مسیر طی شده و نوسانات بدنه، حاکی از آن است که روش جدید به خوبی توانسته است ارتباط موثر میان دو روش، مد لغزشی و شبکه عصبی فازی برقرار نماید. برای درک بهتر نحوه عملکرد این ترکیب کننده، نحوه‌ی به‌روزرسانی وزن w_1 ، در کنار سیگنال‌های خطا در شکل (۶-۲۶) نمایش یافته است.

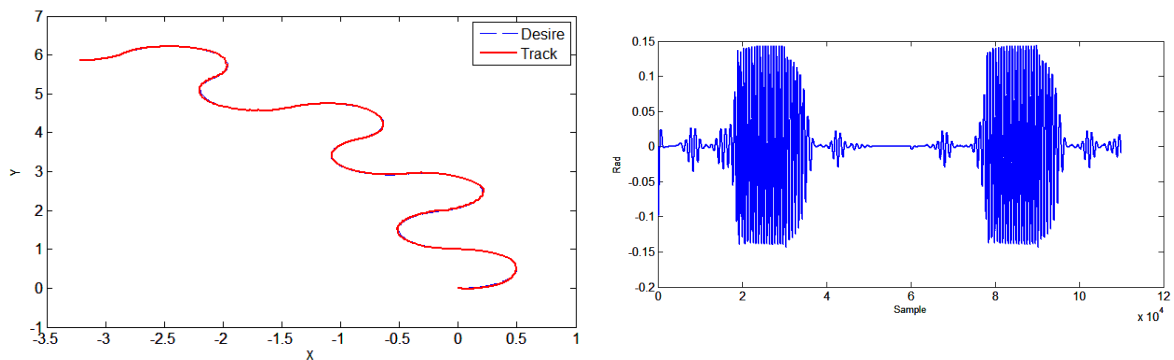


شکل ۶-۲۶: نمایش سیگنال های خطا در کنار نحوه‌ی به روز رسانی وزن های ترکیب کننده

همان طور که در ابتدای این فصل نیز عنوان شد، تمامی شبیه‌سازی‌ها و پیاده‌سازی‌ها بر اساس یک الگوی حرکتی مستطیلی انجام می‌شود که در شکل (۶-۱) نشان داده‌شد. این امر با توجه به سختی پیمایش این مسیر و تنها به منظور یکسان و قابل مقایسه بودن نتایج صورت گرفته است. لذا برای نشان دادن قدرت کنترل کننده در پیمایش مسیرهای دیگر، نمونه‌ای از ردگیریها و نوسانات بدنه برای سه مسیر؛ بیضی، مارپیچ در شکل‌های (۶-۲۷) و (۶-۲۸) مشاهده می‌گردد. که خطای ردگیری آنان در حدود 9×10^{-4}



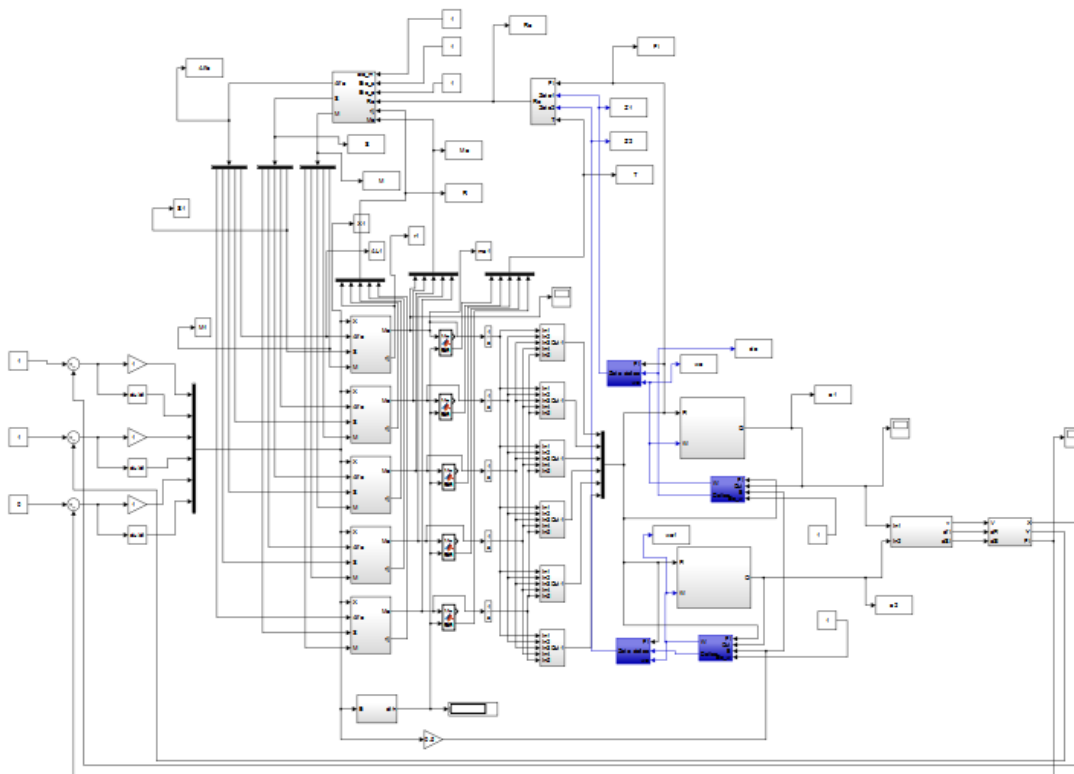
شکل ۶-۲۷: نحوه ردگیری و خطای بدنه برای مسیر بیضی



شکل ۶-۲۸: نحوه ردگیری و خطای بدنه برای مسیر مارپیچ

لازم به ذکر است به علت بزرگ بودن مسیر طی شده، خطای ردگیری در تصاویر چندان قابل مشاهده نیست

بلوک دیاگرام کلی استفاده شده در متلب برای کنترل برخط این ربات در شکل ۶-۲۹ نشان داده شده است.



شکل ۶-۲۹: بلوک دیاگرام استفاده شده در متلب

۶-۶- جمع‌بندی:

در این فصل ابتدا اهداف کنترلی ربات بیان شده و پس از آن مسیری برای کنترل ردگیری ربات پیشنهاد گردید. پس از آن، مدل دینامیکی ربات به دو بخش مجزا تفکیک شده و برای هر کدام کنترل‌کننده‌ی جداگانه‌ای به روش مد لغزشی طراحی شد. نشان داده شد که کنترل‌کننده‌ی مد لغزشی علاوه بر طراحی بر اساس مدل سیستم و مقاومت در برابر اغتشاشات خارجی، دارای نوسانات نسبتاً بالایی است. برای برطرف کردن این مشکل، یک کنترل‌کننده عصبی-فازی پیشنهاد شد و در پیاده‌سازی‌های عملی نشان داده شد که این کنترل‌کننده، به جز در لحظه‌ی شروع، پاسخ به مراتب بهتری نسبت به روش مد لغزشی دارد اما درعین حال در مقابل اغتشاشات خروجی مقاوم نبوده و به علت نیاز به مدت زمان کافی برای آموزش شبکه؛ سیستم از حالت تعادل خارج می‌گردد که برای رفع این مشکل، تلفیق خطی دو کنترل‌کننده معرفی و نتایج شبیه‌سازی و پیاده‌سازی آن عنوان گردید.

۷- فصل هفتم:

نتیجہ گیری و پیشہاوا

۷-۱- نوآوری تحقیق:

این تحقیق به معرفی و کنترل ریز ربات تعادلی با نام «زحل» اختصاص یافت. این ربات کوچک‌ترین ربات تعادلی متحرک به شمار می‌رود، ابعاد کوچک و وزن سبک آن باعث شده تا کنترل همزمان تعادل و ردگیری دشوار باشد. به همین منظور از دو روش مد لغزشی و شبکه‌ی فازی عصبی برای کنترل ربات بهره گرفته شد و برای اولین بار تلفیقی از این کنترل‌کننده‌ها با روشی نوین ارائه گردید و به طور عملی پیاده‌سازی شد تا اثبات شود این روش به پایداری سیستم کمک خواهد کرد

۷-۲- نتیجه‌گیری:

این پایان‌نامه به بررسی یک روش سنتی و یک روش نوین کنترلی برای کنترل ربات تعادلی پرداخت، هر دوی این روش‌ها به صورت تئوری قابل اثبات و شبیه‌سازی‌اند، اما با پیاده‌سازی واقعی روش‌ها بر روی مدل‌های ساخته شده می‌توان به نقایص و ضعف‌های کلی هر روش پی برد. از طرف دیگر، طراحی‌های مکانیکی فراوانی صورت گرفته و مدل‌های زیادی برای ربات تعادل پیشنهاد شده است اما کمتر سعی شده تا سیستم مکانیک و کنترل در راستای هم طراحی و محدودیت‌های هر یک در دیگری لحاظ شود. اگرچه روش‌های هوشمند و جدید به بیان راه‌کارهایی برای کنترل سیستم‌های نامعلوم می‌پردازند، اما این پژوهش نشان می‌دهد، اطلاعات نسبی از معادلات حاکم بر سیستم و طراحی روش‌هایی که مدل سیستم را نیز در خود لحاظ کرده باشند، می‌تواند سرعت پاسخ بهتری به ما بدهد.

ربات تعادلی به عنوان سیستمی ناپایدار، غیرخطی و تزویج شده، شناخته می‌شود؛ که نیازمند پاسخ سریع کنترل‌کننده جهت جلوگیری از برهم خوردن تعادل خود دارد. اگرچه سیستم‌هایی که از الگوریتم‌های شناسایی و یادگیری برخط استفاده می‌کنند، قدرت زیادی در کنترل این گونه سیستم‌ها دارند، اما همواره لازم است تا یک روش پشتیبان تضمین‌شده برای سیستم وجود داشته باشد که در صورت ورود اغتشاشات و اختلالات ناخواسته بتواند سیستم را در شرایط پایداری حفظ کند.

در این پژوهش از دو کنترل کننده برای کنترل ربات تعادلی «زحل» استفاده شد و نشان داده شد که اگر چه هر دو توانایی کنترل سیستم را دارند، اما هر یک مزایایی دارند که لازم است برای به دست آوردن حداکثر این مزایا، تلفیقی از دو روش ارائه شود.

کنترل های بر مبنای مدل، اگر چه سنتی و سخت هستند، اما به خاطر این که قسمتی از سیستم را در قلب خود جای داده اند، سرعت پاسخ بیشتری نسبت به سیستم های هوشمند دارند، اما اگر بتوان مدت زمان لازم برای آموزش یک الگوریتم هوشمند را توسط کنترل کننده ی دیگری تأمین کرد، پس از گذشت دوره ی آموزش و در صورت عدم بروز اختلالی در سیستم، فرآیند کنترلی بهتری خواهیم داشت.

این پایان نامه نشان می دهد که یک شبکه عصبی فازی مقاوم، اگر چه توانای کنترلی بیشتری در پایداری ربات دارد، اما در بروز اختلالات ناخواسته سرعت واکنش کمتری نسبت به روش مد لغزشی دارد. لذا گامی رو به جلو برداشته شد و از تلفیق دو کنترل کننده، روشی جدید برای کنترل ربات ارائه شد، تا در مواقع لزوم انتقال میان کنترل کننده ها با الگوریتم معرفی شده انجام پذیرد.

از مجموع طراحی، شبیه سازی و پیاده سازی های انجام شده می توان به این نتیجه رسید که نگاه دینامیکی به معادلات حرکت سیستم برای طراحی کنترل کننده های واقعی لازم است. لذا ترکیب کنترل کننده ی مد لغزشی به عنوان یک کنترل کننده ی مقاوم و تطبیقی با شبکه های عصبی فازی می تواند مزایای هر دو را در خود جای داده و پاسخی بهتر نسبت به هر یک خواهد داشت.

۷-۳- پیشنهادها:

بر اساس قسمت های مختلف عنوان شده در پایان نامه، پیشنهادها را می توان به سه بخش تقسیم کرد

بخش اول: پیشنهادها کنترلی

۱- کنترل ناوبری ربات تعادلی با استفاده از تلفیق روش های هوشمند و سنتی

۲- استفاده از شبکه فازی عصبی برای کنترل پارامترهای صفحه لغزش

۳- طراحی زاویه‌ی مطلوب بدنه، با توجه به برآیند نیروها و در نظر گرفتن شتاب حرکتی ربات

۴- استفاده از پردازش شکل برای کنترل موقعیت نهایی ربات

۵- پیاده‌سازی روش کنترل ولتاژ^۱ بر روی این ساختار

بخش دوم: پیشنهادها ساخت

۱- پیاده‌سازی الگوریتم‌های بیان‌شده بر روی میکروکنترلر ها به گونه‌ای که ربات نیازی به اتصال

به رایانه نداشته باشد

۲- استخراج بهینه‌ترین پارامترهای طراحی برای ربات تعادلی.

۳- طراحی سیستمی برای کاهش اثر لقی موتورها در سیستم مکانیکی

بخش سوم: پیشنهاد کاربردی

۱- استفاده از ربات تعادلی به عنوان یک موش هوشمند یا ربات فوتبالیست

¹ On the voltage control Method

۸- منابع

- [1] Mark W. Spong, P. C. (1999). Nonlinear Control of the Inertia Wheel Pendulum. *Automatica*, 37, 1845.
- [۲] الاسلام، م. ک. (۲۰۰۸). طراحی کنترلر فازی برای کنترل ربات پاندول معکوس چرخ دار با استفاده از الگوریتم ژنتیک. شانزدهمین کنفرانس سالانه مهندسی مکانیک. کرمان.
- [3] Grasser, A. D. (2002). JOE: A Mobile, Inverted Pendulum. *IEEE Industrial Electronics*, 49, 107.
- [4] Segway, "<http://www.segway.com>," [Online].
- [5] He Bin, L. W. (2010). The Kinematics Model of a Two-wheeled Self-balancing Autonomous Mobile Robot and Its Simulation. *Second International Conference on Computer Engineering and Applications*.
- [۶] بناب، ش. ت. (۱۳۹۰). تحلیل دینامیکی و طراحی ربات تعادلی دوچرخ با استفاده از کنترل کننده PD. چهاردهمین کنفرانس دانشجویی مهندسی برق ایران. کرمانشاه.
- [7] Jung, H.-J. L. (2008). Control of a Mobile Inverted Pendulum Robot System. *International Conference on Control, Automation and Systems*. Seoul.
- [8] Ching-Chih Tsai, Z.-Z. W. (2011). Intelligent Adaptive Trajectory Tracking Using Fuzzy Basis Function Networks for Self-Balancing Two-Wheeled Mobile Robots. *International Conference on System Science and Engineering*. Macau.
- [9] Wu Junfeng, Z. W. (2011). Research on Control Method of Two-wheeled Self-balancing Robot. *Fourth International Conference on Intelligent Computation Technology and Automation*. China.
- [10] Ong Yin Chee, M. S. (2006). Design And Development of Two Wheeled Autonomous Balancing Robot. *4th Student Conference on Research and Development*. Malaysia.
- [11] Laurent Vermeiren, A. T.-T. (2011). Modeling, control and experimental verification on a two-wheeled vehicle with free inclination: An urban transportation system. *Control Engineering Practice*, 19, 744.
- [۱۲] آیت الله زاده، س. (۱۳۹۰). تحلیل دینامیکی، شبیه سازی و کنترل وسیله ی دوچرخ تک محور. دانشکده مکانیک، دانشگاه صنعتی شریف.
- [13] Qin Yang, L. Y. (2011). Balance Control of Two-Wheeled Self-Balancing Mobile Robot Based on TS Fuzzy Model. *6th International Forum on Strategic Technology*.
- [14] Fuquan Dai, F. L. (2012). Development of a Coaxial Self-Balancing Robot Based on Sliding Mode Control. *International Conference on Mechatronics and Automation*. chengdu, china.
- [15] Parhi, D. R. (2003). Navigation of Mobile Robots Using a Fuzzy Logic Controller. *Journal of Intelligent and Robotic Systems*, 42(3), 255.
- [16] A.M. Almeshal, K. G. (2013). Dynamic modelling and stabilization of a new configuration of two-wheeled machines. *Robotics and Autonomous Systems*, 25, 217.

- [17] Angeles, A. S. (2007). A New Family of Two-Wheeled Mobile Robots: Modeling and Controllability. *IEEE TRANSACTIONS ON ROBOTICS*, 23(1), 169.
- [18] Kui Peng, Xiaogang Ruan, a. G.. (2012). Dynamic Model and Balancing Control for Two-Wheeled Self-Balancing Mobile Robot on the Slopes. *10th World Congress on Intelligent Control and Automation*. Beijing, China.
- [19] Ching- Chih Tsai, S.-Y. J. (2010). Trajectory Tracking and Regulation of a Self-Balancing Two-Wheeled Robot : A Backstepping Sliding-Mode Control Approach. *SICE Annual Conference*. Taipei.
- [20] Kaustubh Pathak, J. a. (2005). Velocity and Position Control of a Wheeled Inverted Pendulum by Partial Feedback Linearization. *IEEE TRANSACTIONS ON ROBOTICS*, 21, 505.
- [21] Ooi, R. C. (2003). Balancing a Two-Wheeled Autonomous Robot. *The University of Western Australia School of Mechanical Engineering*.
- [22] YEONHOON KIM, S. H. (2005). Dynamic Analysis of a Nonholonomic Two-Wheeled Inverted Pendulum Robot. *Journal of Intelligent and Robotic Systems*, 44, 25.
- [23] Zhijun Li, Y. ., (2010). Support vector machine optimal control for mobile wheeled inverted pendulums with unmodelled dynamics. *Neurocomputing*, 73, 2773.
- [24] S. W. Nawawi, M. N. (2008). Real-Time Control of a Two-Wheeled Inverted Pendulum Mobile Robot. *PROCEEDINGS OF WORLD ACADEMY OF SCIENCE, ENGINEERING AND TECHNOLOGY*, 29, 214.
- [25] Zhijun Lia, C. (2009). Adaptive fuzzy logic control of dynamic balance and motion for wheeled inverted pendulums. *Fuzzy Sets and Systems*, 160, 1787.
- [26] Zhijun Li, Y. Z. (2008). Adaptive Robust Dynamic Balance and Motion Control of Mobile Wheeled Inverted Pendulums. *7th World Congress on Intelligent Control and Automation*. chongqing.
- [27] Jian Huang, Z.-H. G. (2010). Sliding-Mode Velocity Control of Mobile-Wheeled Inverted-Pendulum Systems. *IEEE TRANSACTIONS ON ROBOTICS*, 26(4), 750.
- [28] ماهر، م. ت. (۱۳۹۰). رویکردی بر رباتهای تعادلی دوچرخه مبتنی بر پاندول معکوس. چهاردهمین کنفرانس دانشجویی مهندسی برق ایران. کرمانشاه.
- [29] Hamid Reza Memarbashi, J.-Y. C. (2011). Design and parametric control of co-axes driven two-wheeled balancing robot. *Microsyst Technol*, 17, 1215.
- [30] Hyungjik Lee, S. J. (2012). Balancing and navigation control of a mobile inverted pendulum robot using sensor fusion of low cost sensors. *Mechatronics*, 22, 95.
- [31] CHRISTIAN SUNDIN, F. T. (2012). Autonomous balancing robot: Design and construction of a balancing robot. *Department of Signals and Systems CHALMERS UNIVERSITY OF TECHNOLOGY*.
- [32] لاریمی، م. (۱۳۹۱). دینامیک و کنترل ربات دوچرخه بازو دار. دانشکده مکانیک، دانشگاه خواجه نصیرالدین طوسی..
- [33] Rong-Jong Wai, I. a.-M. (2009). Design of Dynamic Petri Recurrent Fuzzy Neural Network and Its Application to Path-Tracking Control of Nonholonomic Mobile Robot. *IEEE TRANSACTIONS ON INDUSTRIAL ELECTRONICS*, 56(7), 2667.
- [34] M, B. (2003). Modeling and Model Verification of an Intelligent Self-Balancing Two-Wheeled Vehicle for an Autonomous Urban Transportation System. *The*

Conference on Computational Intelligence, Robotics, and Autonomous Systems.
Singapore.

[۳۵] Nagarajan, U .(2012) *.Fast and Gracefull Balancing Mobile Robots .*Carnegie Mellon University: The Robotics Institute School of Computer Science.

Abstract

Inverted pendulum know as classic problem in dynamic and control that almost used to satisfying control theorem. In recent year the Balancing robot idea proposed subject to inverted pendulum. Robot are Nonlinear, unstable and Nonholonomic so its need a complex calculation for modelling and control

Published article studied Robot in different view. Some of paper define it as vehicle, some other try to modelling and so many used it to determine the control system.

Different Nonlinear method used to balance, speed and trajectory control of robot also intelligence control method growth and developed such that robust and adaptive control was suggested regardless dynamic model in past decades.

This thesis discuss about little balancing robot named as<Zohal> and proposed the mixture of classic and intelligence nonlinear method to control the robot. It should be note that Zohal introduced as tiny Robot not a vehicle.

Experimental result show the power of Sliding-mod as a nonlinear and model based method to defend disturbance but it had a large scale fluctuation cause the big trajectory error.so we have designed a neural fuzzy network to control both of balancing and trajectory, however the good control performance and low fluctuation achieve but it hadn't ability to deal with disturbance; because the time require for trained will not provide so imbalance is cussed by smallness of robot.

To resolved this problem a linear combination of sliding mode and Neural-Fuzzy network used that combine the outputs of both controller based on the error's co-efficient, it's mean at a same time percent of each control signal applied to the output due to the goodness of effect.

Keyword: balancing robot, mobile robot, inverted pendulum, neural fuzzy, sliding mod, two wheeled robot



Shahrood University of Technology

Faculty Mechanic Engineering

**Adaptive neural-fuzzy controller design for trajectory and
balance control of two wheeled balancing robot**

Mohammad Hassan Gheisarieha

Supervisors:

Hossein khosravi

Alireza Alfi

Date: September 2013