

الله أكبر



دانشکده مهندسی برق و رباتیک

پایان نامه کارشناسی ارشد مهندسی رباتیک

بهینه سازی کنترل ربات با به کارگیری الگوریتم های هوشمند

مبتنی بر راهبرد کنترل ولتاژ

نگارنده:

یاسر زمانداری

استاد راهنما:

پروفسور محمدمهدی فاتح

شهریور ۱۳۹۷

شماره ۱۵۹۱، آ.ت.ب
تاریخ: ۲۷/۶/۱۳

باسمه تعالی



مدیریت تحصیلات تکمیلی

فرم شماره (۳) صورتجلسه نهایی دفاع از پایان نامه دوره کارشناسی ارشد

با نام و یاد خداوند متعال، ارزیابی جلسه دفاع از پایان نامه کارشناسی ارشد خانم / آقای یاسر زمانداری با شماره

دانشجویی ۹۴۰۸۹۰۴ رشته مهندسی برق- رباتیک گرایش کنترل ربات تحت عنوان: بهینه سازی کنترل ربات

با به کارگیری الگوریتم های هوشمند مبتنی بر راهبرد کنترل ولتاژ که در تاریخ ۱۳۹۷/۰۶/۱۲ با حضور هیأت

محترم داوران در دانشگاه صنعتی شاهرود برگزار گردید به شرح ذیل اعلام می گردد:

قبول (با درجه: خیلی خوب) مردود
نوع تحقیق: نظری عملی

عضو هیأت داوران	نام و نام خانوادگی	مرتبه علمی	امضاء
۱- استاد راهنمای اول	مسعود مهدی زاده رخی	استاد	
۲- استاد راهنمای دوم	---	---	---
۳- استاد مشاور	---	---	---
۴- نماینده تحصیلات تکمیلی	مسعود مهدی زاده رخی	استادیار	
۵- استاد ممتحن اول	علیرضا انصاری	دانشیار	
۶- استاد ممتحن دوم	سید علی حسینی ابوبکر	دانشیار	

نام و نام خانوادگی رئیس دانشکده: علیرضا انصاری
تاریخ و امضاء و مهر دانشکده: ۲۷/۶/۱۳

تبصره: در صورتی که کسی مردود شود حداکثر یکبار دیگر (در مدت مجاز تحصیل) می تواند از پایان نامه خود دفاع نماید (دفاع مجدد نباید زودتر از ۴ ماه برگزار شود).

به پاس محبت‌های بی‌دینشان که هرگز فروکش نمی‌کند

به پاس عاطفه سرشار و کرمای امیدبخش وجودشان

و به پاس قلب‌های بزرگشان

این مجموعه را تقدیم می‌کنم به

مادر و پدر عزیزم

شکر خدا که هر چه طلب کردم از خدا بر منتهای همت خود کامران شدم
و به مصداق «من لم یشکر المخلوق لم یشکر الخالق» بسی شایسته است

از استاد فرهیخته و پرمایه‌ام پروفیسور محمدمهدی فاتح

برای زحمات گران‌قدر و راهنمایی‌های کارسازشان

بی‌نهایت سپاسگذار باشم

از اساتید دانشکده مهندسی برق و رباتیک

از تمام اساتید و دوستان عزیزمی که علمشان را بی‌منت می‌بخشند

و

بهترین‌های زندگی‌ام

مادر و پدر فداکار و خواهران دلسوزم

از صمیم قلب سپاسگزارم

تعهد نامه

اینجانب یاسر زمانداری دانشجوی دوره کارشناسی ارشد رشته مهندسی رباتیک دانشکده مهندسی برق و رباتیک دانشگاه صنعتی شاهرود نویسنده پایان نامه بهینه سازی کنترل ربات با به کارگیری الگوریتم های هوشمند مبتنی بر راهبرد کنترل ولتاژ تحت راهنمایی پروفسور محمدمهدی فاتح متعهد می شوم

- تحقیقات در این پایان نامه توسط اینجانب انجام شده است و از صحت و اصالت برخوردار است.
- در استفاده از نتایج پژوهش های محققان دیگر به مرجع مورد استفاده استناد شده است.
- مطالب مندرج در پایان نامه 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 می باشد 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 پایان نامه تأثیرگذار بوده اند در مقالات مستخرج از پایان نامه رعایت می گردد.
- در کلیه مراحل انجام این پایان نامه ، در مواردی که از موجود زنده (یا بافت های آن ها) استفاده شده است ضوابط و اصول اخلاقی رعایت شده است.
- در کلیه مراحل انجام این پایان نامه، در مواردی که به حوزه اطلاعات شخصی افراد دسترسی یافته یا استفاده شده است اصل رازداری ، ضوابط و اصول اخلاق انسانی رعایت شده است.

۹۷/۰۶/۱۲

مالکیت نتایج و حق نشر

- کلیه حقوق معنوی این اثر و محصولات آن (مقالات مستخرج، کتاب، برنامه های رایانه ای، نرم افزار ها و تجهیزات ساخته شده است) متعلق به دانشگاه صنعتی شاهرود می باشد. این مطلب باید به نحو مقتضی در تولیدات علمی مربوطه ذکر شود.
- استفاده از اطلاعات و نتایج موجود در پایان نامه بدون ذکر مرجع مجاز نمی باشد.

چکیده:

در این پایان‌نامه، طراحی کنترل‌کننده بهینه با بهره‌گیری از الگوریتم‌های تکاملی هوشمند برای کنترل موقعیت بازوی ربات مبتنی بر راهبرد کنترل ولتاژ ارائه می‌شود. به‌عنوان مطالعه موردی یک قانون کنترل مقاوم با راهبرد کنترل ولتاژ برای سیستم بازوی دو-رابط رباتیک با حضور عدم قطعیت‌ها پیشنهاد می‌گردد. مزیت طرح کنترل پیشنهادی آن است که سیستم حلقه بسته تکرارپذیر است. در نتیجه اجرای الگوریتم‌های بهینه‌سازی میسر می‌گردد. در ادامه، برای نیل به اهداف کنترلی، ضرایب کنترل‌کننده با بهره‌گیری از برخی الگوریتم‌های تکاملی هوشمند، به‌طور بهینه به دست می‌آید. در روش‌های بهینه‌سازی پیشنهادی از تابع چند هدفه که مجموع میانگین مربعات خطای ردگیری و میانگین مربعات خطای سیگنال کنترل است، به‌عنوان تابع هدف استفاده می‌شود.

الگوریتم‌های ازدحام ذرات، کرم شب‌تاب و خفاش از نظر دقت، سرعت و کیفیت همگرایی مقایسه می‌شوند. به دلیل استفاده از راهبرد کنترل ولتاژ در طرح کنترلی، محاسبات مستقل از مدل دینامیکی ربات است. پایداری و مقاوم بودن سیستم کنترل پیشنهادی با الگوریتم‌های بهینه‌سازی محقق می‌گردد.

کلمات کلیدی: کنترل بهینه ربات، کنترل هوشمند ربات، راهبرد کنترل ولتاژ، تکرارپذیری، الگوریتم

تکاملی هوشمند، بازوی رباتیک

فهرست مطالب

صفحه

فصل اول: مقدمه ۱

- ۱-۱- رباتیک و اهمیت کنترل ربات ۲
- ۲-۱- مروری بر کارهای انجام شده ۴
- ۳-۱- اهداف تحقیق ۱۰
- ۴-۱- ساختار پایان نامه ۱۱

فصل دوم: مدل سازی بازوی رباتیک دو-رابط ۱۳

- ۱-۲- مقدمه ۱۴
- ۲-۲- مدل سازی سینماتیکی ۱۴
- ۱-۲-۲- سینماتیک مستقیم ۱۴
- ۲-۲-۲- سینماتیک سرعت و ماتریس ژاکوبین ۱۵
- ۳-۲- مدل سازی دینامیکی ۱۶
- ۴-۲- مدل ربات دو-رابط ۱۶

فصل سوم: راهبرد کنترل ولتاژ ۱۹

- ۱-۳- مقدمه ۲۰
- ۲-۳- معادلات حرکت سیستم رباتیک ۲۱
- ۳-۳- طراحی کنترل کننده با راهبرد کنترل ولتاژ ۲۲
- ۴-۳- شبیه سازی سیستم کنترل ۲۵

فصل چهارم: بهینه سازی و الگوریتم های تکاملی هوشمند ۲۷

- ۱-۴- مقدمه ۲۸
- ۲-۴- بهینه سازی ۲۸
- ۳-۴- بیان ریاضی مسئله بهینه سازی ۳۱
- ۱-۳-۴- بردار طراحی ۳۲
- ۲-۳-۴- فضای طراحی ۳۲
- ۳-۳-۴- قیدهای طراحی ۳۲
- ۴-۴- پارامترهای کنترلی مشترک بین الگوریتم های بهینه سازی هوشمند ۳۲
- ۱-۴-۴- تابع هدف ۳۳
- ۲-۴-۴- معیار توقف و شاخص عملکرد الگوریتم های بهینه سازی هوشمند ۳۴
- ۵-۴- الگوریتم بهینه سازی ازدحام ذرات ۳۶
- ۱-۵-۴- عملکرد الگوریتم بهینه سازی ازدحام ذرات ۳۷

۳۹.....	۲-۵-۴- پیاده‌سازی الگوریتم بهینه‌سازی ازدحام ذرات
۴۱.....	۶-۴- الگوریتم بهینه‌سازی کرم شب‌تاب
۴۱.....	۱-۶-۴- عملکرد الگوریتم کرم شب‌تاب
۴۷.....	۲-۶-۴- پیاده‌سازی الگوریتم کرم شب‌تاب
۴۹.....	۷-۴- الگوریتم بهینه‌سازی خفاش
۵۰.....	۱-۷-۴- عملکرد الگوریتم خفاش
۵۱.....	۲-۷-۴- پیاده‌سازی الگوریتم خفاش

۵۳..... فصل پنجم: طراحی کنترل غیرخطی مقاوم

۵۴.....	۱-۵- مقدمه
۵۵.....	۲-۵- کنترل‌کننده غیرخطی مقاوم
۵۹.....	۳-۵- شبیه‌سازی روش کنترل غیرخطی مقاوم بر روی ربات
۵۹.....	۴-۵- بررسی عملکرد کنترل‌کننده غیرخطی مقاوم در ردگیری مسیر مطلوب

۶۳..... فصل ششم: بهینه‌سازی کنترل‌کننده غیرخطی مقاوم

۶۴.....	۱-۶- مقدمه
۶۴.....	۲-۶- بهینه‌سازی سیستم کنترل غیرخطی مقاوم
۶۷.....	۱-۲-۶- بهینه‌سازی با استفاده از الگوریتم بهینه‌سازی ازدحام ذرات
۶۸.....	۱-۱-۲-۶- بررسی عملکرد الگوریتم ازدحام ذرات در بهینه‌سازی کنترل‌کننده
۷۰.....	۲-۲-۶- بهینه‌سازی با استفاده از الگوریتم کرم شب‌تاب
۷۱.....	۱-۲-۲-۶- بررسی عملکرد الگوریتم کرم شب‌تاب در بهینه‌سازی کنترل‌کننده
۷۲.....	۳-۲-۶- بهینه‌سازی با استفاده از الگوریتم خفاش
۷۳.....	۱-۲-۲-۶- بررسی عملکرد الگوریتم خفاش در بهینه‌سازی کنترل‌کننده
۷۵.....	۳-۶- مقایسه عملکرد الگوریتم‌های تکاملی هوشمند در بهینه‌سازی سیستم کنترل

۷۹..... فصل هفتم: نتیجه‌گیری و پیشنهادها

۸۰.....	۱-۷- نتیجه‌گیری
۸۰.....	۲-۷- پیشنهادها
۸۱.....	فهرست منابع

فهرست اشکال

- شکل ۱-۲ ربات دو-رابط [۵۲] ۱۷
- شکل ۱-۳ دیاگرام کنترل ولتاژ موتور مفصل ربات ۲۱
- شکل ۲-۳ دیاگرام موتور مغناطیس دائم جریان مستقیم [۵۲] ۲۴
- شکل ۳-۳ سیستم کنترل ربات بر مبنای راهبرد کنترل ولتاژ [۵۲] ۲۴
- شکل ۴-۳ سیستم کنترل موتور [۵۲] ۲۴
- شکل ۵-۳ مسیر مطلوب ردگیری در شبیه‌سازی راهبرد کنترل ولتاژ ۲۵
- شکل ۶-۳ خطای ردگیری سیستم کنترل با راهبرد کنترل ولتاژ ۲۶
- شکل ۷-۳ ولتاژ موتورهای سیستم کنترل با راهبرد کنترل ولتاژ ۲۶
- شکل ۱-۴ تقسیم‌بندی الگوریتم‌های بهینه‌سازی [۵۵] ۲۹
- شکل ۲-۴ نمایش نموداری حرکت ذرات در روش ازدحام ذرات [۶۶] ۳۵
- شکل ۳-۴ فلوجارت الگوریتم بهینه‌سازی ازدحام ذرات ۳۸
- شکل ۱-۵ مسیر مطلوب ردگیری ۶۰
- شکل ۲-۵ ولتاژ ورودی موتورها ۶۰
- شکل ۳-۵ جریان موتورها ۶۰
- شکل ۴-۵ ولتاژ ورودی موتورها با اعمال محدودکننده ۶۱
- شکل ۵-۵ جریان ورودی موتورها با اعمال محدودکننده ۶۱
- شکل ۶-۵ خطای ردگیری کنترل‌کننده ۶۲
- شکل ۷-۵ نمودار تغییرات دینامیک‌های مدل نشده ۶۲
- شکل ۱-۶ مسیر مطلوب ردگیری در بهینه‌سازی ۶۶
- شکل ۲-۶ خطای ردگیری کنترل‌کننده بهینه با استفاده از الگوریتم بهینه‌سازی ازدحام ذرات ۶۹
- شکل ۳-۶ ولتاژ ورودی موتورها در حالت بهینه با استفاده از الگوریتم بهینه‌سازی ازدحام ذرات ۶۹
- شکل ۴-۶ جریان موتورها در حالت بهینه با استفاده از الگوریتم بهینه‌سازی ازدحام ذرات ۶۹
- شکل ۵-۶ نمودار همگرایی پارامترهای کنترل‌کننده بهینه با استفاده از الگوریتم بهینه‌سازی ازدحام ذرات ۷۰
- شکل ۶-۶ خطای ردگیری کنترل‌کننده بهینه با استفاده از الگوریتم بهینه‌سازی کرم شب‌تاب ۷۱
- شکل ۷-۶ ولتاژ ورودی موتورها در حالت بهینه با استفاده از الگوریتم بهینه‌سازی کرم شب‌تاب ۷۱

شکل ۶-۸ جریان ورودی موتورها در حالت بهینه با استفاده از الگوریتم بهینه‌سازی کرم شب‌تاب ۷۲

شکل ۶-۹ نمودار همگرایی پارامترهای کنترل‌کننده بهینه با استفاده از الگوریتم بهینه‌سازی کرم شب‌تاب ۷۲

شکل ۶-۱۰ خطای ردگیری کنترل‌کننده بهینه با استفاده از الگوریتم بهینه‌سازی خفاش ۷۳

شکل ۶-۱۱ ولتاژ ورودی موتورها در حالت بهینه با استفاده از الگوریتم بهینه‌سازی خفاش ۷۴

شکل ۶-۱۲ جریان ورودی موتورها در حالت بهینه با استفاده از الگوریتم بهینه‌سازی خفاش ۷۴

شکل ۶-۱۳ نمودار همگرایی پارامترهای بهینه‌کننده با استفاده از الگوریتم بهینه‌سازی خفاش ۷۴

فهرست جداول

- جدول ۱-۲ جدول دناویت-هارتنبرگ برای ربات دو-رابط ۱۵
- جدول ۲-۲ پارامترهای دینامیکی ربات دو-رابط ۱۷
- جدول ۱-۳ پارامترهای موتور جریان مستقیم مغناطیس دائم ۲۵
- جدول ۱-۵ پارامترهای موتور جریان مستقیم مغناطیس دائم ۵۹
- جدول ۱-۶ حدود پارامترها در بهینه‌سازی ۶۶
- جدول ۲-۶ پارامترهای داخلی الگوریتم بهینه‌سازی ازدحام ذرات ۶۸
- جدول ۳-۶ پارامترهای کنترل‌کننده بهینه و مشخصه‌های الگوریتم ازدحام ذرات ۶۸
- جدول ۴-۶ پارامترهای داخلی الگوریتم بهینه‌سازی کرم شب‌تاب ۷۰
- جدول ۵-۶ پارامترهای کنترل‌کننده بهینه و مشخصه‌های الگوریتم کرم شب‌تاب ۷۰
- جدول ۶-۶ پارامترهای داخلی الگوریتم بهینه‌سازی خفاش ۷۲
- جدول ۷-۶ پارامترهای کنترل‌کننده بهینه و مشخصه‌های الگوریتم خفاش ۷۳
- جدول ۸-۶ عملکرد الگوریتم بهینه‌سازی ازدحام ذرات برای حالت‌های مختلفی از ضرایب وزنی تابع هدف چندگانه ۷۵
- جدول ۹-۶ عملکرد الگوریتم بهینه‌سازی کرم شب‌تاب برای حالت‌های مختلفی از ضرایب وزنی تابع هدف چندگانه ۷۶
- جدول ۱۰-۶ عملکرد الگوریتم بهینه‌سازی خفاش برای حالت‌های مختلفی از ضرایب وزنی تابع هدف چندگانه ۷۶
- جدول ۱۱-۶ مقایسه عملکرد الگوریتم‌های PSO، FA و BA در بهینه‌سازی مشخصه‌ها ۷۸

فصل اول

مقدمه

در این فصل ابتدا مختصری از تاریخچه سیستم‌های رباتیک^۱ و سپس توضیحاتی در مورد ربات و انواع آن، مزایای استفاده از آن و اهمیت کنترل ربات ارائه می‌شود. در ادامه مروری بر کارهای انجام شده خواهد شد و پس از آن اهداف کلی پایان‌نامه بیان می‌شود. در پایان طرح کلی و ساختار پایان‌نامه ارائه خواهد شد.

۱-۱- رباتیک و اهمیت کنترل ربات

رباتیک دانشی بین‌رشته‌ای است که کاربردهای بسیاری از جمله در صنعت، تجارت و پزشکی پیدا کرده است. کلمه ربات از واژه ربتا^۲ در زبان چک گرفته شده که به معنای برده است و اولین بار در سال ۱۹۲۱ در نمایشنامه آقای کارل چاپک^۳ با موضوع طغیان ربات‌های کارگر علیه کشاورزان به کار رفت. از این رو ذهنیت اولیه مردم از ربات‌ها، موجوداتی خطرناک و مضر برای انسان و محیط زیست بود و بسیاری از نویسندگان و فیلم‌سازان در انتهای داستان‌های خود ربات‌ها را موجوداتی افسارگسیخته، آسیب‌رسان معرفی می‌کردند. ایزاک اسیموف^۴ دانشمند و نویسنده روسی-آمریکایی در آثار خود با هدف تغییر این ذهنیت اشتباه و حمایت از علم رباتیک، ربات را یک ماشین قابل اطمینان هماهنگ با انسان و تا حدودی باهوش معرفی نمود و در کتاب رونده^۵ به تعریف قوانین رباتیک پرداخت. بدین ترتیب در دهه ۱۹۳۰ اولین تلاش‌ها برای ساخت ربات صورت گرفت. در سال ۱۹۳۸ پلارد^۶ یک بازوی مکانیکی قابل برنامه‌ریزی برای پاشیدن رنگ ابداع کرد و مدتی بعد برای اولین بار گوئرتز^۷ با اضافه کردن قابلیت کنترل از راه دور به یک بازوی مکانیکی آن را برای کار با مواد رادیواکتیو محیا ساخت. بعدها در سال ۱۹۴۶ جورج دول^۸ کنترل‌کننده مغناطیسی با قابلیت تکرار را اختراع کرد. در همین زمان وسیله‌ای

¹ *Robotic systems*

² *Robota*

³ *Karel Čapek*

⁴ *Isaac Asimov*

⁵ *Runaround*

⁶ *Alfred Oliver Pollard*

⁷ *Gueretz*

⁸ *George Dole*

ساخته شد که با استفاده از حسگرها، حرکات دست شخص برای راه‌اندازی موتورهای ربات را اندازه‌گیری می‌کرد. هم‌زمان با انجام گرفتن این اختراعات، اکوت و موجلی، کامپیوتر اینیاک^۱ را در دانشگاه پنسیلوانیا ساختند و این گونه برای ارضای نیاز ماشین‌های مکانیکی به سیستم هوشمند کنترلی، در سال ۱۹۵۲ اولین ماشین کنترل عددی در دانشگاه MIT ساخته شد که منجر به ابداع اولین زبان برنامه‌نویسی برای ماشین‌ها شد. دو سال بعد جرج دول اولین ربات برنامه‌پذیر و هوشمند با قابلیت حرکت تکراری نقطه به نقطه را اختراع کرد. دناویت با همکاری هارتنبرگ ن ماتریس‌های تبدیل همگن را بیان کرد و شش سال بعد یعنی در سال ۱۹۶۲ اولین ربات صنعتی حافظه‌دار با بهره‌گیری از کنترل عددی و تکنولوژی عملگرهای الکتریکی ساخته شد. طراحی و توسعه ربات‌های متحرک بر اساس تمایل به حمل‌ونقل اتوماتیک در خطوط تولید صورت گرفت و در نهایت در اواخر دهه ۷۰ میلادی منجر به ساخت سیستم‌های حمل‌ونقل بدون راننده شد. با پیشرفت رباتیک در جهان و ساخت ربات‌های متنوع، در سال ۱۹۸۳ رباتیک در صنعت و علم به‌صورت عمومی مطرح شد و بسیاری از برنامه‌های درسی و آموزشی بر همین اساس شکل گرفت و تا به امروز همچنان ادامه دارد به‌طوری‌که بسیاری از کشورهای توسعه‌یافته جهان بیشتر برنامه‌های صنعتی و آموزشی خود را بر اساس علم رباتیک تدوین و پایه‌ریزی می‌کنند. از این‌رو تحقیقات گسترده در زمینه‌های مختلف کاربردهای ربات، انجام شده است.

امروزه بسیاری از جنبه‌های زندگی فردی و اجتماعی بشر، تحت تأثیر اتوماسیون قرار دارد. معمولاً به استفاده از ربات‌ها به‌ویژه در محیط‌های صنعتی اتوماسیون گفته می‌شود [۱]. ربات‌ها دستگاه‌های حساس و دقیقی هستند که بسیار سریع‌تر از انسان و به‌صورت اتوماتیک و برنامه‌ریزی‌شده کارهای محوله را انجام می‌دهند.

ربات‌ها انواع مختلفی دارند که بر اساس کاربرد شامل؛ ربات‌های صنعتی، ربات‌های پزشکی، ربات‌های خدماتی، ربات‌های نظامی، ربات‌های سرگرمی، ربات‌های آموزشی، ربات‌های توان‌بخشی، ربات‌های خانگی و ربات‌های مسابقه‌ای می‌شوند. در دسته‌بندی دیگری ربات‌ها به انواع: ربات‌های ایستا یا بستر

^۱ Eniac

ثابت، ربات‌های چرخ‌دار، پادار، ربات‌های پروازی، ربات‌های شناگر، ربات‌های کشسانی نرم یا انعطاف پذیر^۱، ربات‌های ماژولار^۲، ربات‌های گروهی یا همکار، ربات‌های نرم‌افزاری یا هوش مصنوعی، میکرو ربات‌ها و نانو ربات‌ها تقسیم می‌شوند. سیستم‌های رباتیک از قسمت‌های مکانیکی مفصل‌بندی شده، مجری نهایی، محرکه‌ها، حسگرها، کنترل‌کننده‌ها، سیستم انتقال نیرو، منبع تغذیه، پردازشگرها و نرم‌افزار تشکیل می‌شوند.

از مهم‌ترین قابلیت‌های ربات می‌توان به برنامه‌ریزی و کنترل از راه دور آن اشاره کرد. بر همین اساس ربات‌ها کاربردهای وسیعی در محیط‌هایی که انسان قادر به انجام کار در آن‌ها نیست، یافته‌اند. محیط‌های سخت و صنعتی، محیط‌های غیربهداشتی و سمی و محیط‌های خیلی سرد یا خیلی گرم از جمله محیط‌هایی هستند که برای کار انسان مفید نیست. همچنین می‌توان از ربات‌ها در کارهایی استفاده نمود که انسان قادر به درک، حل و انجام آن‌ها در سریع‌ترین زمان نیست. استفاده از ربات‌ها در انجام تست‌های حساس صنعتی و پزشکی، جراحی‌های دقیق بر روی اعضای بدن، کار با مواد رادیواکتیو، رنگرزی و قیرپاشی، جابجایی اشیا و حمل بار، مونتاژ دستگاه‌ها، برش فلزات، انواع جوش‌های صنعتی، چسباندن مواد و ... متداول است. همچنین استفاده از تجهیزات رباتیک باعث بهبود کمی و کیفی تولید، نزول هزینه‌ها و از همه مهم‌تر سهولت بیشتر فرآیند کنترل و در کل بالا بردن راندمان کاری شده است که این امر توجه محققان و صنعتگران را به طراحی سیستم‌های کنترل ربات با دقت و سرعت بالا و هزینه پایین‌تر جلب کرده است [۲].

۱-۲- مروری بر کارهای انجام‌شده

غالباً ربات برای انجام هر کاری در هر شکلی و تحت هر شرایطی نیاز به کنترل‌کننده دارد و نیروی محرکه اغلب ربات‌ها، توسط موتورهای الکتریکی تأمین می‌شود. ربات برای انجام هر کاری نیاز به کنترل‌کننده دارد. در بسیاری از کنترل‌کننده‌های طراحی شده قبلی، معادلات دینامیکی محرکه در نظر

¹ Flexible
² Modular

گرفته نشده است که این امر باعث ایجاد خطای دائمی و در بعضی موارد ناپایداری در سیستم کنترل می‌شود. از طرفی اکثر ربات‌ها بنا به الزامات مکانیکی خود نیاز به گشتاور بالا و سرعت پایین دارند در صورتی که موتورهای الکتریکی که برای حرکت دادن مفاصل ربات به کار گرفته می‌شوند دارای گشتاور پایین و سرعت بالایی هستند. در نتیجه برای عملکرد موردنظر، موتورهای الکتریکی به سیستم انتقال نیرو مجهز می‌شوند که یکی از عوامل مؤثر در پیچیدگی سیستم‌های رباتیک محسوب می‌شوند. همچنین ردگیری دقیق موقعیت بازوی ربات به دلیل پیچیدگی حاصل از مدل غیرخطی چندمتغیره آن و حضور عدم قطعیت‌ها، با مشکل مواجه می‌شود. بنابراین برای نائل شدن به عملکرد مطلوب کنترل‌کننده، باید این عوامل مهار و جبران شوند. برای حذف دینامیک‌های غیرخطی در ابتدا روش‌های کنترل گشتاور محاسباتی [۳] و کنترل خطی‌سازی پس‌خوری پیشنهاد شدند [۴]. این روش‌ها بر اساس مدل دینامیکی ربات بوده و نیازمند شناخت دقیق از مدل سیستم و در نتیجه نیازمند محاسبات سنگین هستند. همچنین در عمل حضور عدم قطعیت‌های غیر ساختاری مانند اغتشاش خارجی و دینامیک‌های مدل نشده و تا حدودی متغیر بودن پارامترهای سیستم از جمله بارهای گرانشی، عملکرد این کنترل‌کننده‌ها را با مشکل مواجه می‌کند. برای حل این مشکلات و مقابله با پیچیدگی‌های کنترل سیستم رباتیک روش‌های کنترل مجزا سازی^۱ مطرح شده است [۵]. خطی‌سازی فیدبکی که با عنوان روش کنترل گشتاور نیز شناخته می‌شود، یکی از روش‌هایی است که دینامیک غیرخطی ربات را به چند معادله دینامیکی خطی مرتبه دوم تبدیل می‌کند. روش خطی‌سازی پس‌خوردی روش مبتنی بر مدل است و مدل‌سازی، چه در اثر عدم دانش کافی در مدل کردن دینامیکی و چه در اثر گذشت زمان و تغییر پارامترها و چه در اثر اغتشاشات خارجی، همواره با عدم قطعیت روبرو است. کنترل تطبیقی، کنترل حالت لغزشی، کنترل غیرخطی مقاوم و کنترل هوشمند بر حسب مورد روش‌هایی هستند که محققان برای مقاوم کردن کنترل‌کننده‌های خطی‌سازی پس‌خوری در برابر عدم قطعیت‌ها ابداع کرده‌اند.

¹ Decoupling control

یکی از پرکاربردترین نظریه‌ها که برای کنترل سیستم‌های پیچیده با ساختار دینامیکی معلوم اما با پارامترهای نامعلوم مورد استفاده قرار می‌گیرد نظریه کنترل تطبیقی است [۶]. با این وجود، کنترل تطبیقی عملکرد ضعیفی در مقابله با عدم قطعیت‌های غیرساختاری مانند اغتشاش خارجی و دینامیک مدل نشده دارد [۷]. با توجه به اینکه سیستم‌های رباتیک همواره در معرض عدم قطعیت‌های غیرساختاری هستند و نظریه کنترل مقاوم توانایی غلبه بر عدم قطعیت‌ها به شکل مطلوب را دارد این روش مورد توجه قرار گرفت [۸]. قوانین کنترل مقاوم با استفاده از روش لیاپانوف تحلیل می‌شوند. کارهای ارزشمند بسیاری با استفاده از روش کنترل مقاوم در فضای مفصلی [۹] و در فضای کار ربات [۱۰، ۱۱] ارائه شده است. شرط لازم برای طراحی کنترل مقاوم، معلوم بودن کران عدم قطعیت است به طوری که پایداری سیستم حلقه بسته تضمین شود [۱۲]. اگر حدود عدم قطعیت کمتر از مقدار واقعی باشد، خطای ردگیری افزایش یافته و می‌تواند منجر به ناپایداری سیستم کنترل شود. از طرف دیگر، اگر حدود عدم قطعیت بزرگ‌تر از مقدار واقعی باشد، ممکن است اندازه سیگنال کنترل بیشتر از مقدار مجاز آن شده و پدیده اشباع رخ دهد. در این صورت علاوه بر اینکه کنترل کننده قادر به کنترل سیستم نخواهد بود، پدیده لرزش سیگنال کنترل نیز تقویت شده و ممکن است به سیستم آسیب برساند [۱۳-۱۵]. کنترل غیرخطی مقاوم بازوهای رباتیک نیز روش سودمندی برای بهبود عملکرد خطی سازی فیدبکی جهت غلبه بر عدم قطعیت است [۱۶-۱۸]. در این روش، طراح یک مدل نامی برای سیستم تخمین زده و یک جمله به قانون کنترل خطی سازی فیدبکی برای جبران عدم قطعیت اضافه می‌شود. این روش علاوه بر عدم قطعیت پارامتری بر عدم قطعیت ناشی از دینامیک مدل نشده و اغتشاش خارجی هم غلبه می‌کند که در بیشتر مواقع در مقایسه با کنترل تطبیقی عملکرد بهتری دارد.

یکی دیگر از روش‌های کنترل مقاوم برای غلبه بر عدم قطعیت‌های پارامتری و اغتشاش خارجی، روش کنترل حالت لغزشی است که باعث سرعت در پاسخ دینامیکی و سادگی محاسباتی می‌شود [۱۹]. این روش به دلیل استفاده از تابع علامت در مقاوم کردن سیگنال ورودی با پدیده لرزش مواجه است که باعث تحریک دینامیک‌های مدل نشده سیستم در فرکانس‌های بالا، عملکرد نامطلوب سیستم و

فروسدگی قطعات می‌شود. همچنین نظریه ساختار متغیر از جمله روش‌هایی است که بسیار مورد توجه قرار گرفته است [۲۰]. لازمه استفاده از این روش نیز معلوم بودن کران بالای نامعینی است. همچنین قانون کنترل شامل کلیدزنی است که در نهایت این روش نیز سبب ایجاد لرزش سیگنال کنترل می‌شود. برای حل این معضل روش‌های هموارسازی سیگنال کنترل مانند روش‌های تطبیقی و پیشنهاد شدند [۲۱]. یک روش دیگر، روش کنترل مقاوم پس گام است که این روش بر اساس فیدبک‌های بازگشتی از حالت‌های سیستم بوده به طوری که پایداری سیستم حلقه بسته حفظ شود. این روش نیز مانند هر روش مقاوم دیگر احتیاج به معلوم کردن کران عدم قطعیت دارد [۲۲].

در سال‌های اخیر روش‌های کنترل هوشمند با ظهور منطق فازی و شبکه‌های عصبی، به عنوان روش‌های پرکاربردی در مهندسی کنترل مورد توجه واقع شده‌اند. قوانین فازی دانش و تجربه کارشناسان خبره را به صورت زبانی بیان می‌کنند. کنترل فازی یک روش مستقل از مدل است و به همین دلیل استفاده از آن برای تخمین و توصیف سیستم‌هایی که مدل ریاضی دقیقی از آن‌ها نداریم مفید است [۲۳]. از جمله مهمترین دلایل کاربردی شدن روش‌های کنترل فازی در سیستم‌های رباتیک، سادگی در طراحی و عدم وابستگی به مدل ریاضی سیستم و به ویژه استفاده از قوانین زبانی برای به کارگیری افراد با تجربه است [۲۴]. پیاده‌سازی کنترل فازی معمولی ساده‌تر است اما احتیاج زیادی به قواعد کارشناسان خبره دارد. در بیشتر موارد کنترل‌کننده‌های فازی برای سیستم حالت سعی و خطا دارد یعنی به صورت جعبه سیاه بکار گرفته می‌شود [۲۵]. کنترل مقاوم فازی ربات به کمک روش مستقیم لیاپانوف از جمله راه‌حلی با ساختار ساده برای استفاده از مزایای کنترل فازی است [۲۶]. در بسیاری از روش‌های کنترلی برای سیستم‌های غیرخطی همراه با عدم قطعیت از سیستم‌های فازی به عنوان تقریب‌گر عمومی استفاده شده است [۲۷، ۲۸]. کنترل عصبی یکی دیگر از روش‌های کنترل هوشمند و غیرخطی مستقل از مدل است که کاربردهای موثری در زمینه کنترل ربات داشته است [۲۹-۳۲]. دقت این سیستم‌ها رابطه مستقیم با نحوه آموزش نرون‌ها در شبکه دارد و از آموزش‌های تکراری برای تنظیم پارامترها در دینامیک سیستم‌های پیچیده استفاده می‌شود.

در سال‌های اخیر به دلیل توانایی‌های روش‌های تطبیقی و مقاوم، توجه ویژه‌ای به این نوع کنترل‌کننده‌ها شده است و تلاش‌ها مبنی بر تخمین دقیق از عدم قطعیت‌ها و کاهش خطای ردیابی و چترینگ سیگنال کنترل منجر به استفاده از توابع ریاضی نظیر سری فوریه و سری تیلور در طراحی در این نوع کنترل‌کننده‌ها شده است [۴۰، ۴۱].

در پژوهش‌های بسیاری، ترکیب روش‌های کنترلی با هدف بهره‌جستن از مزایای روش‌های مختلف، سبب افزایش کارایی و دقت سیستم‌های کنترل در سال‌های اخیر شده است. ترکیب کنترل تطبیقی، مقاوم و فازی از مزایای هر سه نوع کنترل‌کننده برخوردار خواهد بود [۳۳]، این در حالی است که دینامیک‌های غیر ساختاری در اختیار باشند. به هر حال این روش حجم محاسباتی بالایی دارد. در برخی موارد هم به دلیل پیچیدگی‌های خاص سیستم، به دست آوردن یک قانون کنترل کارآمد برای سیستم با استفاده از کنترل‌کننده‌های تطبیقی مقاوم مقدور و مقرون‌به‌صرفه نیست. در بسیاری از روش‌های مد لغزشی-فازی سیستم فازی برای معرفی سطح لغزش بکار گرفته شده است [۳۴]. در بعضی موارد هم از سیستم فازی یا عصبی برای تخمین حدود عدم قطعیت استفاده شده است [۳۵]. کنترل نرو-فازی مقاوم نیز برای دستیابی به موقعیت دقیق ربات پیاده‌سازی شده است [۳۶]. طرح کنترل فازی تطبیقی مد لغزشی برای بازوی ربات پیشنهاد شده است که تا حدودی لرزش سیگنال حذف شده است [۳۷]. در طرح دیگری کنترل مقاوم ربات با استفاده از تخمین گر فازی تطبیقی، به تخمین و جبران عدم قطعیت‌ها دست یافته است [۳۸]. با وجود توانایی‌های منحصر به فرد، تضمین پایداری سیستم‌های کنترل فازی و عصبی دشوار است [۳۹]. در جهت بهینه‌سازی کنترل‌کننده‌ها نیز تلاش‌هایی شده است. استفاده از الگوریتم‌های بهینه‌سازی هوشمند با هدف بهینه‌کردن عملکرد سیستم کنترل ربات مورد توجه قرار گرفته است. در طرحی با پیشنهاد یک قانون کنترل پیچیده و استفاده از الگوریتم بهینه‌سازی توده ذرات (PSO) به بهینه‌سازی پارامترهای کنترل‌کننده پرداخته است [۴۲]. با استفاده از الگوریتم PSO کنترل فازی عصبی حالت لغزشی را باهدف بهینه‌سازی کنترل گشتاور انجام داده است [۴۳]. در طرح دیگری با فرض حرکت تکراری ربات به کنترل بهینه تکراری گسسته زمان بازوی ربات پرداخته است [۴۴]. از

الگوریتم هوشمند بیولوژیک (BIA) به صورت کاربردی برای کنترل ربات متحرک استفاده شده است [۴۵]. برای بهینه‌سازی عملکرد ربات‌های خودکار، از الگوریتم کرم شبتاب استفاده شده است [۴۶،۴۷]. الگوریتم بهینه‌سازی خفاش (BA) نیز از جمله جدیدترین روش‌های بهینه‌سازی است که به تازگی در مسائل کنترل و رباتیک مورد استفاده قرار می‌گیرد. با استفاده از BA یک مسیر بهینه برای ربات متحرک برنامه‌ریزی شده است [۴۸]. پارامترهای کنترل‌کننده PI به منظور کنترل موقعیت مطلوب بازوی رباتیک و بهبود عملکرد آن با استفاده از الگوریتم BA تنظیم شده است [۴۹]. با استفاده از الگوریتم خفاش یک قانون کنترل بهینه برای کنترل و شکل‌گیری سیستم چند رباتی به دست آمده است [۵۰].

مسئله اصلی در به کارگیری الگوریتم‌های تکاملی هوشمند به عنوان تکنیکی برای بهینه‌سازی، مبتنی بر مسیر بودن این الگوریتم‌ها است. لذا در بهینه‌سازی عملکرد کنترل‌کننده‌ها با استفاده از این الگوریتم‌ها، سیستم حلقه بسته کنترل باید تکرارپذیر باشد تا عملیات بهینه‌سازی مفید و مثمر ثمر واقع شود که در بسیاری از مسایل کنترل بهینه به این مهم توجه نشده است.

گشتاور ورودی به مفاصل پارامتری است که در اکثر روش‌های کنترل ربات محاسبه می‌شود. این گشتاور توسط محرک به سیستم اعمال می‌شود و مسئله مهم برای ایجاد مقدار مطلوبی از این پارامتر، تحریک مناسب محرک‌ها است. همچنین برای پیاده‌سازی قانون کنترل در روش‌های کنترل گشتاور، به سیگنال‌های پس‌خورد موقعیت، سرعت و شتاب نیاز است که این سیگنال‌ها آغشته به نویز حسگرها هستند و باعث تضعیف عملکرد سیستم کنترل می‌شوند [۵۱]. از آنجایی که کنترل ربات در فضای کار پیچیده‌تر از کنترل آن در فضای مفصلی است، کنترل گشتاور ربات در فضای کار با پیچیدگی و محاسبات سنگین روبرو است زیرا در فضای کار به ماتریس ژاکوبین احتیاج داریم [۵۳].

راهبرد کنترل ولتاژ با عملکرد مستقل از مدل ربات، راهگشای این مشکلات شده است. براساس این روش بازوی رباتیک به عنوان بار روی موتورهای الکتریکی که به عنوان محرک مفاصل ربات هستند در

نظر گرفته می‌شوند. در واقع کنترل روی ولتاژ موتورها اعمال شده و ورودی سیستم، ولتاژ موتور و خروجی آن، موقعیت زاویه‌ای موتور است. عملاً این راهبرد مستقل از دینامیک پیچیده و غیرخطی همراه با تزویج زیاد ربات است [۵۲]. زیرا هم مدل موتور در مقایسه با مدل ربات بسیار ساده‌تر است و هم برای هر مفصل فقط موتور آن مفصل به‌طور مجزا از سایر موتورها کنترل می‌شود و این کار باعث مجزا سازی کامل، محاسبات کم و سرعت و دقت بالا می‌شود. گشتاور موردنیاز برای بازوی ربات به‌صورت گشتاور بار روی موتور در معادلات می‌آید و برای پیاده‌سازی قانون کنترل فقط به پس‌خورد جریان موتور و موقعیت آن نیاز است که اندازه‌گیری آن‌ها به‌مراتب ساده‌تر از اندازه‌گیری سرعت و شتاب است. راهبرد کنترل ولتاژ در مقایسه با راهبرد کنترل گشتاور تا حد خیلی زیادی مصون از اغتشاشات خارجی است لذا با تضمین تکرارپذیری سیستم، بسیار مناسب برای عملیات بهینه‌سازی با استفاده از روش‌های خارج از خط^۱ مانند الگوریتم‌های فراابتکاری^۲ هوشمند می‌باشد.

۱-۳- اهداف تحقیق

با توجه به مزایای راهبرد کنترل ولتاژ نسبت به راهبرد کنترل گشتاور، در این پایان‌نامه راهبرد کنترل ولتاژ برای کنترل موقعیت بازوی ماهر دو-رابط در فضای مفصلی انتخاب می‌شود. همچنین، به دلیل مصونیت راهبرد کنترل ولتاژ از اغتشاشات خارجی، با پیشنهاد یک قانون کنترل مقاوم تکرارپذیر، از مزایای الگوریتم‌های تکاملی هوشمند جهت بهینه‌سازی کنترل‌کننده پیشنهادی استفاده شده و در نهایت به مقایسه عملکرد الگوریتم‌های تکاملی ازدحام ذرات، کرم شبتاب و خفاش مورد استفاده برای بهینه‌سازی کنترل‌کننده می‌پردازد.

¹ Offline

² Meta-heuristic

۱-۴- ساختار پایان نامه

در فصل ۲ مروری اجمالی بر مدل سازی ربات آمده است. در فصل ۳ راهبرد کنترل ولتاژ و به کارگیری آن در بازوی ربات عنوان شده است. در فصل ۴ بهینه سازی و روش های منتخب از الگوریتم های تکاملی هوشمند ارائه شده است. در فصل ۵ کنترل غیرخطی مقاوم بازوی ربات با راهبرد کنترل ولتاژ بیان شده است. در فصل ۶ بهینه سازی کنترل کننده پیشنهادی با به کارگیری الگوریتم های تکاملی هوشمند ارائه شده است. و در پایان روش های بهینه سازی با یکدیگر مقایسه شده است.

فصل دوم

مدل سازی بازوی رباتیک دو-رابط

۲-۱- مقدمه

در این فصل با استفاده از مدل‌سازی ریاضی، مدل بازوی ماهر دو-رابط ارائه شده است [۵۴]. مدل‌سازی ربات‌ها شامل دو بخش مدل‌سازی سینماتیکی و مدل‌سازی دینامیکی است. در مدل‌سازی سینماتیکی بدون در نظر گرفتن نیروهای وارده به سیستم روابط مربوط به حرکت ربات را به دست می‌آورند. مهم‌ترین گام در مدل‌سازی سینماتیکی تشکیل جدول دناویت-هارتنبِrg است. از داده‌های این جدول به عنوان روشی منظم برای انجام سینماتیک مستقیم استفاده می‌شود. مدل‌سازی دینامیکی نیز به دو روش فرمول‌بندی نیوتن-اویلر و فرمول‌بندی لاگرانژی انجام می‌شود. در روش نیوتن-اویلر از قانون دوم نیوتن در حرکت و در روش لاگرانژ از قضیه کار و انرژی برای توصیف سیستم‌های دینامیکی استفاده می‌شود.

۲-۲- مدل‌سازی سینماتیکی

۲-۲-۱- سینماتیک مستقیم

مفاصل ربات‌ها به دو نوع لولایی و کشویی تقسیم می‌شوند. مفصل لولایی (که به اختصار با R نمایش داده می‌شود) امکان چرخش نسبی بین دو رابط را فراهم می‌آورد. مفصل کشویی (که با نماد P نشان داده می‌شود) اجازه حرکت نسبی طولی بین دو رابط را می‌دهد. با توجه به چگونگی ترتیب مفاصل ربات‌ها، پیکربندی‌های مختلفی از آن‌ها وجود دارد. پیکربندی‌های متداول عبارتند از: هنرمند، استوانه‌ای، اسکارا، استنفورد و کروی. در این پایان‌نامه از ربات دو-رابط در سیستم کنترل استفاده شده است.

دستورالعمل دناویت-هارتنبِrg روشی منظم و منسجم برای مدل‌سازی سینماتیکی انواع ربات‌ها است. آنالیز سینماتیک مستقیم (تعیین موقعیت و جهت مجری نهایی با استفاده از متغیرهای مفاصل) با استفاده از این دستورالعمل انجام می‌شود. برای انجام این کار باید ماتریس‌های تبدیل و دوران و بردار انتقال را معرفی کرد و طی یک الگوریتمی جدول دناویت-هارتنبِrg را تشکیل داد [۵۴].

پارامترهای دناویت-هارتنبرگ برای ربات دو-رابط در جدول (۱-۲) آمده است.

a_i : طول امتداد x_i از O_i تا محل تقاطع محورهای x_i و Z_{i-1} است.

d_i : طول امتداد Z_{i-1} از O_{i-1} تا محل تقاطع محورهای x_i و Z_{i-1} است. هرگاه مفصل i کشویی باشد d_i متغیر است.

α_i : زاویه بین Z_i و Z_{i-1} که حول x_i اندازه‌گیری می‌شود.

θ_i : زاویه بین x_i و x_{i-1} که حول Z_{i-1} اندازه‌گیری می‌شود. هرگاه مفصل i لولایی باشد θ_i متغیر است.

جدول ۱-۲ دناویت-هارتنبرگ برای ربات دو-رابط [۵۴]

	θ	d	a	α
رابط ۱	θ_1^*	0	1	0
رابط ۲	θ_2^*	0	1	0

۲-۲-۲- سینماتیک سرعت و ماتریس ژاکوبین

روابط سرعت ربات به وسیله ماتریس ژاکوبین $J(q)$ به دست می‌آیند. این ماتریس یکی از مهم‌ترین کمیت‌های آنالیز و کنترل حرکت ربات محسوب می‌شود. ماتریس ژاکوبین در طراحی مسیرهای هموار، تعیین ترکیب‌های تکین، تبدیل نیروها و گشتاورها از مجری نهایی به مفاصل به کار می‌رود. در حقیقت این ماتریس را می‌توان به‌عنوان یک ماتریس تبدیل در نظر گرفت که بردار سرعت مفاصل \dot{q} را به بردار سرعت در فضای دکارتی یا همان فضای کار \dot{x} تبدیل می‌کند. رابطه سرعت‌ها در فضای مفصلی و فضای کار به‌صورت زیر است:

$$\dot{x} = J(q)\dot{q} \quad (1-2)$$

روابط به دست آوردن ژاکوبین در [۵۴] ارائه شده است.

۲-۳- مدل سازی دینامیکی

در این بخش رفتار دینامیکی بازوی ربات را بررسی می‌کنیم. رفتار دینامیکی به نرخ تغییر حالت و وضعیت بازو نسبت به گشتاور اعمالی به وسیله موتورهای محرک اطلاق می‌شود و همان‌طور که اشاره شد این معادلات دیفرانسیل به دو روش قابل حصول هستند. به‌طور مثال برای به دست آوردن مدل دینامیکی ربات از روش لاگرانژ ابتدا انرژی‌های جنبشی و پتانسیل ربات را محاسبه می‌کنیم. سپس لاگرانژین سیستم را تشکیل می‌دهیم. آنگاه با استفاده از معادلات اویلر-لاگرانژ معادله دینامیکی ربات به دست می‌آید [۵۴]. معادله کلی نهایی بعد از محاسبات به شکل زیر ظاهر می‌شود:

$$D(q)\ddot{q} + C(q, \dot{q})\dot{q} + G(q) = \tau \quad (۲-۲)$$

که ماتریس $D(q)\ddot{q}$ ماتریس اینرسی، $C(q, \dot{q})\dot{q}$ بردار نیروی گریز از مرکز و کورلیوس و $G(q)$ بردار نیروی گرانشی برای بازوی ربات می‌باشند. q و \dot{q} و \ddot{q} به ترتیب بردار موقعیت، سرعت و شتاب مفصل ربات است.

۲-۴- مدل ربات دو-رابط

طبق رابطه (۲-۲) برای ربات دو-رابط داریم:

$$\begin{bmatrix} D_{11} & D_{12} \\ D_{21} & D_{22} \end{bmatrix} \begin{bmatrix} \ddot{q}_1 \\ \ddot{q}_2 \end{bmatrix} + \begin{bmatrix} C_{11} & C_{12} \\ C_{21} & C_{22} \end{bmatrix} \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \end{bmatrix} + \begin{bmatrix} g_1 \\ g_2 \end{bmatrix} = \begin{bmatrix} \tau_1 \\ \tau_2 \end{bmatrix} \quad (۳-۲)$$

بعد از محاسبات با روش لاگرانژی درایه‌های ماتریس‌های فوق عبارت‌اند از:

$$\begin{aligned} D_{11} &= m_1 l_{c1}^2 + I_1 + m_2 [l_1^2 + l_{c2}^2 + 2l_1 l_{c2} \cos(q_2)] + I_2 \\ D_{12} &= D_{21} = m_2 l_1 l_{c2} \cos(q_2) + m_2 l_{c2}^2 + I_2 \\ D_{22} &= m_2 l_{c2}^2 + I_2 \end{aligned} \quad (۴-۲)$$

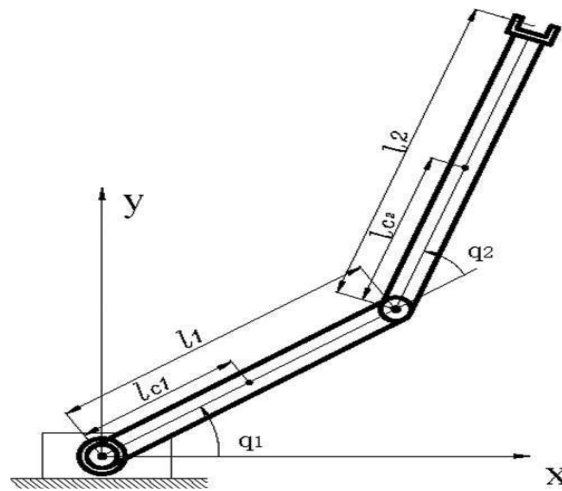
$$\begin{aligned}
C_{11} &= -m_2 l_1 l_{c2} \sin(q_2) \dot{q}_2 \\
C_{12} &= -m_2 l_1 l_{c2} \sin(q_2) (\dot{q}_1 + \dot{q}_2) \\
C_{21} &= m_2 l_1 l_{c2} \sin(q_2) \dot{q}_1 \\
C_{22} &= 0
\end{aligned}
\tag{۵-۲}$$

$$\begin{aligned}
g_1 &= m_1 l_{c1} g \cos(q_1) + m_2 g (l_{c2} \cos(q_1 + q_2) + l_1 \cos(q_1)) \\
g_2 &= m_2 l_{c2} g \cos(q_1 + q_2)
\end{aligned}
\tag{۶-۲}$$

در معادلات فوق، $\tau = [\tau_1 \quad \tau_2]^T$ بردار گشتاور ورودی، q_1 و q_2 زوایای مفصل‌ها، m_1 و m_2 جرم بازوها و طول بازوهای رابط اول و دوم و I_1 و I_2 ممان اینرسی رابط‌ها و l_{c1} و l_{c2} فاصله از مرکز جرم رابط‌ها می‌باشند. این پارامترها به‌طور خلاصه در جدول (۲-۲) آمده است.

جدول ۲-۲ پارامترهای دینامیکی ربات دو-رابط

$g=9.8$	$l(m)$	$m(kg)$	$l_c(m)$	$I(kgm^2)$
رابط ۱	1	1	0.5	0.12
رابط ۲	1	2	0.5	0.25



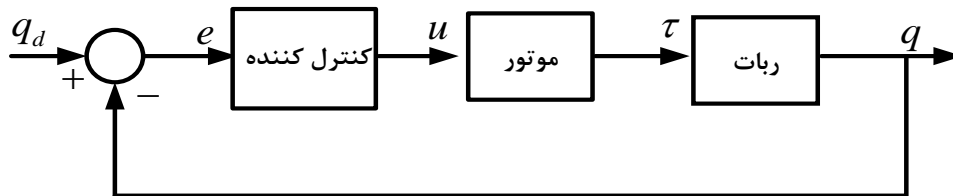
شکل ۱-۲ ربات دو-رابط [۵۲]

فصل سوم

راهبرد کنترل و لتاژ

همان‌طور که در فصل اول اشاره شد، راهبرد کنترل گشتاور به دلیل استفاده از مدل ربات و وجود عملگرها همواره شامل پیچیدگی دینامیک ربات و مشکلات پیاده‌سازی عملی است. در مقابل، راهبرد کنترل ولتاژ معایب کنترل گشتاور را نداشته و به دلیل سادگی و سرعت بالای محاسبات و مقاوم کردن سیستم کنترل ربات مورد توجه قرار گرفته است [۵۲]. ایده اصلی راهبرد کنترل ولتاژ از آنجا نشأت می‌گیرد که به ربات و موتورهای آن، به صورت یک سیستم واحد بنام سیستم رباتیک توجه شود. آنگاه در واقع کنترل ربات به کنترل موتورهای مفاصل ربات تبدیل می‌گردد. از آنجاکه مفاصل ربات توسط موتورهای آن به حرکت درمی‌آیند، مسئله کنترل ربات تبدیل به مسئله کنترل موتورهای آن می‌شود. با این دیدگاه کنترلی، به جای کنترل گشتاور مفاصل می‌توان به کنترل ولتاژ موتورهای مفاصل پرداخت. در این صورت سیگنال‌های ورودی، ولتاژ موتورهای ربات خواهد بود. موتور الکتریکی دستگاهی است که توسط منبع ولتاژ تغذیه می‌شود. بنابراین حتی برای کنترل جریان موتور، باید ولتاژ آن را کنترل نماییم. موتور مغناطیس دائم DC به دلیل داشتن دینامیک خطی، به خوبی و به آسانی قابل کنترل است. مدل سیستم ربات بسیار غیرخطی، چند ورودی-چند خروجی، و دارای تزویج و عدم قطعیت زیاد است. مدل موتور به صورت کلی بسیار ساده‌تر از مدل ربات است. بنابراین، طراحی کنترل‌کننده بر مبنای مدل موتور، بسیار ساده‌تر از طراحی آن بر مبنای مدل ربات است. در این نوع طراحی، ربات به عنوان بار موتور دیده می‌شود. برای کنترل موقعیت هر مفصل، در واقع زاویه موتور آن مفصل کنترل می‌شود. بدین ترتیب، کنترل سیستم چند متغیره ربات، تبدیل به کنترل تک تک موتورهای آن می‌شود که یک سیستم یک ورودی-یک خروجی است و با کنترل جداگانه هر مفصل به سادگی می‌توان ربات مورد نظر را کنترل نمود. بهره‌مندی از مجزا سازی کامل، محاسبات کم و سرعت در انجام محاسبات، سادگی طراحی و دقت بالای آن و مقاوم بودن سیستم کنترل نسبت به دینامیک‌های ربات، علت برتری راهبرد کنترل ولتاژ ربات بر راهبرد کنترل گشتاور است. این راهبرد به دلیل صرف نظر نکردن از دینامیک محرکه‌ها عملکرد

کنترلی بهتری خواهد داشت و برای ردگیری با سرعت بالا مناسب است. شکل (۳-۱) دیاگرام کنترل ولتاژ موتور مفصل ربات را نشان می‌دهد. در این فصل به طراحی و شبیه‌سازی کنترل‌کننده مبنی بر مدل با راهبرد کنترل ولتاژ می‌پردازیم.



شکل ۳-۱ دیاگرام کنترل ولتاژ موتور مفصل ربات

۳-۲- معادلات حرکت سیستم رباتیک

موتورهای dc به دلیل آنکه قابلیت کنترل پذیری بالایی دارند در صنعت دارای کاربردهای فراوانی می‌باشند. معادله حرکت موتورهای dc مغناطیس دائم ربات به صورت ماتریسی زیر است:

$$J_m \ddot{\theta}_m + B_m \dot{\theta}_m + r \tau_l = \tau_m \quad (۱-۳)$$

که در آن J_m ماتریس قطری $n \times n$ ممان اینرسی، B_m ماتریس قطری $n \times n$ ضریب میرایی، θ_m بردار $n \times 1$ موقعیت زاویه‌ای، τ_m بردار گشتاور موتورها و r ماتریس قطری $n \times n$ ضرایب چرخ‌دنده-های موتورها می‌باشند. τ_l از معادله دینامیکی ربات به صورت زیر به دست می‌آید:

$$D(q) \ddot{q} + C(q, \dot{q}) \dot{q} + G(q) = \tau_l \quad (۲-۳)$$

رابطه بین موقعیت موتور و متغیرهای ربات را می‌توان به صورت زیر نوشت:

$$q = r \theta_m \quad (۳-۳)$$

معادله kvl در مدار الکتریکی موتور به صورت زیر است:

$$R I_a + L \dot{I}_a + K_b \dot{\theta}_m = v \quad (۴-۳)$$

که در آن L ماتریس قطری $n \times n$ اندوکتانس، R ماتریس قطری $n \times n$ مقاومت آرمیچر، K_b ماتریس قطری $n \times n$ ضریب عکس‌العمل آرمیچر، و I_a بردار $n \times 1$ جریان و v بردار $n \times 1$ ولتاژ موتورها می‌باشند. رابطه بین گشتاور و جریان نیز در موتور جریان مستقیم مغناطیس دائم به صورت

$$\tau_m = K_m I_a \quad (5-3)$$

است که K_m ماتریس قطری $n \times n$ ضرایب گشتاور موتورها است. در موتور جریان مستقیم مغناطیس دائم K_m و K_b با یکدیگر برابرند.

از معادلات (1-3) تا (5-3) فرم فضای حالت معادلات ربات با خطی سازی فیدبکی به دست می‌آیند:

$$\dot{X} = H(X) + bv \quad (6-3)$$

که در آن،

$$X = [q \dot{q} I_a]^T, \quad b = [0 \ 0 \ L^{-1}] \quad (7-3)$$

و تابع برداری $H(X)$ به صورت زیر است:

$$H(X) = \begin{bmatrix} x_2 \\ (j_m r^{-1} + rD(x_1))^{-1} (-(B_m r^{-1} + rC(x_1, x_2))x_2 - rG(x_1) + k_m x_3) \\ -L^{-1}(k_b r^{-1} x_2 + R x_3) \end{bmatrix} \quad (8-3)$$

همانطور که مشاهده می‌شود معادلات حرکت سیستم رباتیک با در نظر گرفتن دینامیک موتورها نشان‌دهنده یک سیستم مرتبه ۳ غیرخطی چندمتغیره بسیار پیچیده است. همچنین در این روابط ماتریس‌های مدل ربات ظاهر شده‌اند که از لحاظ عملی به‌طور دقیق در دسترس نیستند.

۳-۳- طراحی کنترل‌کننده با راهبرد کنترل ولتاژ

بر اساس راهبرد کنترل ولتاژ از مدل موتور در ساده‌سازی معادلات استفاده می‌شود و روش کنترلی

بر اساس راهبرد کنترل ولتاژ که ربات را به سیستم تک ورودی-تک خروجی تبدیل کرده است و کنترل

هر مفصل را جداگانه انجام می‌دهد ارائه می‌شود. هدف، کنترل موقعیت زاویه‌ای موتور است و سیگنال کنترل، ولتاژ ترمینال موتور است. بنابراین معادلات برای موتور و موقعیت یک موتور برای یک مفصل نوشته می‌شوند و در نهایت قابل تعمیم به بردار $1 \times n$ برای n رابط و موتور مفصل آن‌ها هستند. بنابراین، معادله دینامیکی سیستم همان معادله (۳-۴) است. با توجه به رابطه (۳-۳) می‌توانیم (۳-۴) را بر حسب متغیر مفصل ربات به صورت زیر بنویسیم:

$$RI_a + L\dot{I}_a + K_b r^{-1} \dot{q} = v \quad (۹-۳)$$

باید یک قانون کنترل ارائه دهیم تا خطای ردگیری موقعیت مفصل را به صفر برساند. قانون کنترل زیر را در نظر بگیرید،

$$RI_a + L\dot{I}_a + K_b r^{-1} \dot{\phi} = v \quad (۱۰-۳)$$

$$\dot{\phi} = \dot{q}_d + K_p(q_d - q)$$

با استفاده از روابط (۱۰-۳) تا (۱۲-۳) دینامیک حلقه بسته به صورت زیر درمی‌آید،

$$\dot{q} = \dot{q}_d + K_p(q_d - q) \quad (۱۱-۳)$$

بنابراین می‌توان نتیجه گرفت که،

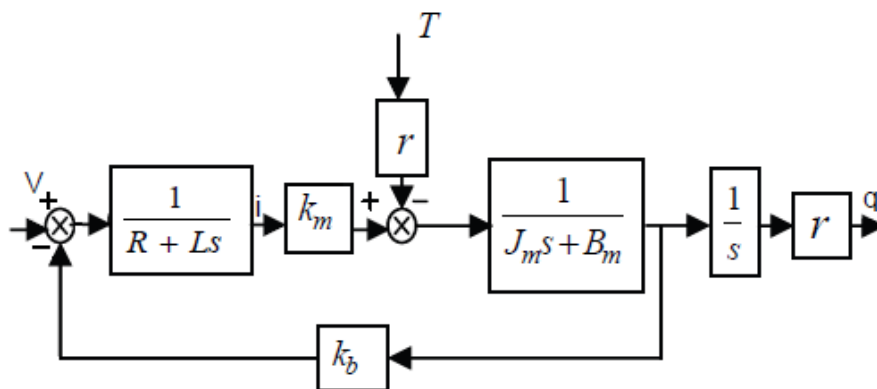
$$\dot{e} + K_p e = 0 \quad (۱۲-۳)$$

که در آن،

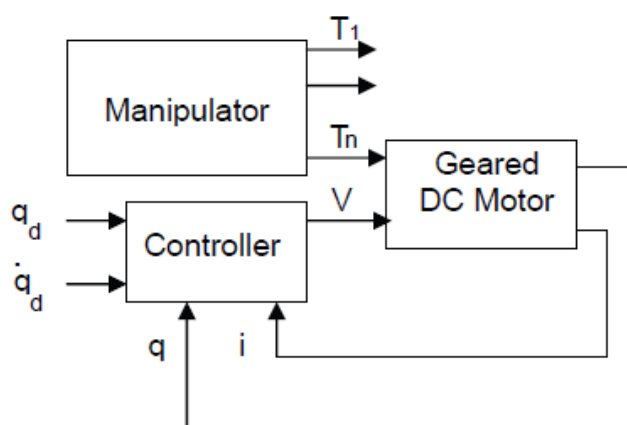
$$e = q_d - q \quad (۱۳-۳)$$

است. بنابراین، اگر K_p مثبت معین باشد، باگذشت زمان خطا به سمت صفر میل می‌کند. همان‌طور که ملاحظه شد، قانون کنترل (۱۰-۳) خطای ردگیری را به صفر می‌رساند.

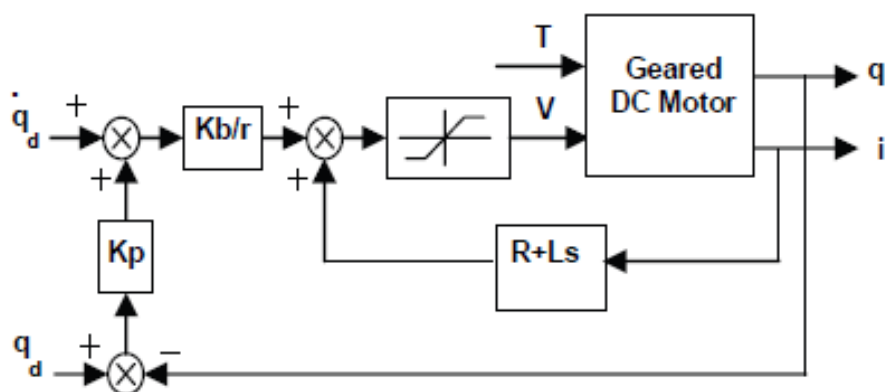
دیگرام موتور مغناطیس دائم DC در شکل (۲-۳) رسم شده است و شکل (۳-۳) سیستم کنترل ربات را نمایش می‌دهد. در شکل (۴-۳) سیستم کنترل موتور با محدودکننده ولتاژ آورده شده است.



شکل ۲-۳ دیاگرام موتور مغناطیس دائم DC [۵۲]



شکل ۳-۳ سیستم کنترل ربات بر مبنای راهبرد کنترل ولتاژ [۵۲]

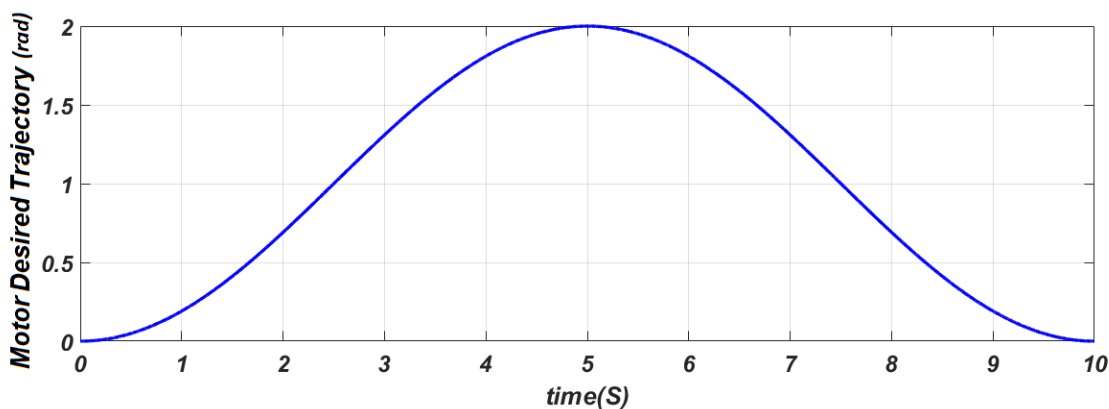


شکل ۴-۳ سیستم کنترل موتور [۵۲]

۳-۴- شبیه‌سازی سیستم کنترل

سیستم کنترل به صورت مستقل به مفصل‌های ربات اعمال می‌شود. بنابراین برای کنترل موتورهای هر مفصل از یک کنترل کننده جداگانه استفاده می‌شود. پارامترهای موتور در جدول (۳-۱)، و پارامترهای دینامیکی ربات طبق جدول (۲-۲) داده شده‌اند. مسیر مطلوب برای ردگیری هر مفصل مطابق تابع زیر انتخاب شده است:

$$q_{di} = 1 - \cos\left(\frac{\pi t}{5}\right), \quad i = 1, 2 \quad (3-15)$$



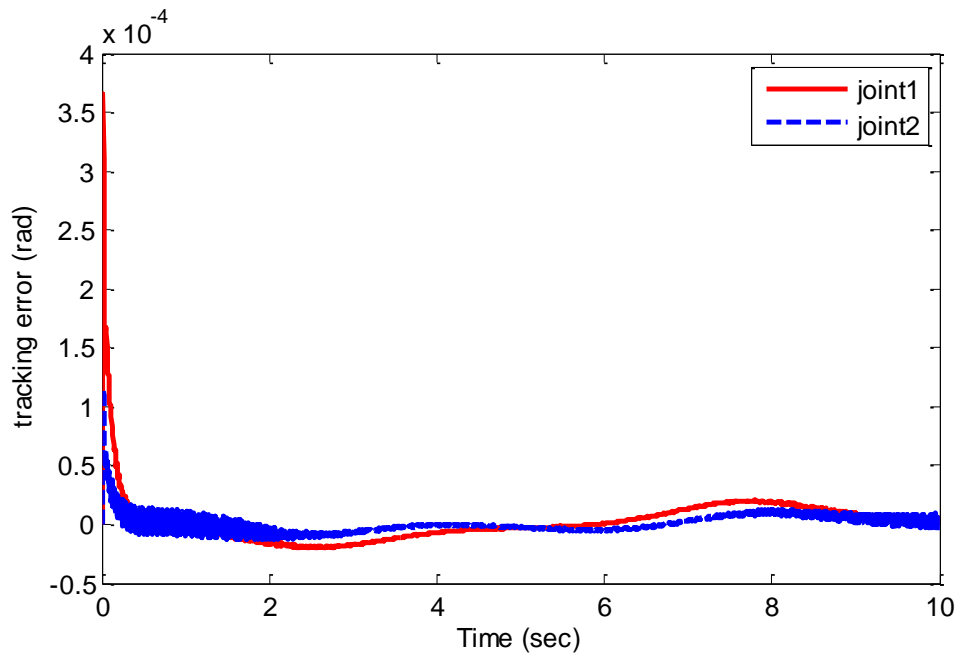
شکل ۳-۵ مسیر مطلوب ردگیری در شبیه‌سازی راهبرد کنترل ولتاژ

در پارامترهای موتور $K_m = K_b$ و در شبیه‌سازی $K_p = 10$ است.

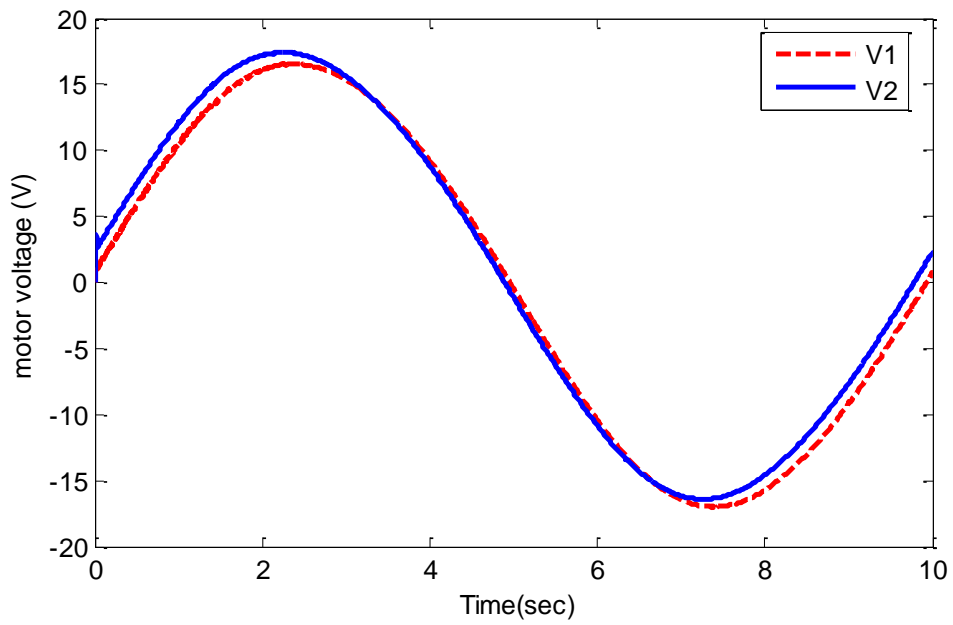
جدول ۳-۱ پارامترهای موتور جریان مستقیم مغناطیس دائم

Motor	$R(\Omega)$	k_m (V.s/rad)	J_m (Nms ² /rad)	B_m (Nms ² /rad)	r	$L(H)$	$u_{max}(V)$
1, 2	1.6	0.26	0.002	0.001	0.01	0.001	45

خطای ردگیری در شکل (۳-۵) رسم شده است. همان‌طور که مشاهده می‌گردد عملکرد ردگیری سیستم کنترل بسیار مناسب است. شکل (۳-۶) سیگنال ولتاژ موتورها را نشان می‌دهد. همان‌طور که ملاحظه می‌شود سیگنالی نرم است، نوسانات شدید ندارد و در محدوده کاری موتورها قرار دارد.



شکل ۳-۶ خطای ردگیری سیستم کنترل با راهبرد کنترل ولتاژ



شکل ۳-۷ ولتاژ موتورهای سیستم کنترل با راهبرد کنترل ولتاژ

فصل چهارم

بهینه‌سازی و الگوریتم‌های تکاملی هوشمند

توسعه علم بهینه‌سازی برخواسته از آرزوی انسان برای رسیدن به کمال است. انسان می‌خواهد بهترین را تجسم کرده و به آن دست یابد. اما از آنجایی که می‌داند نمی‌تواند تمام شرایط حاکم بر بهترین را به‌خوبی بشناسد، در بیشتر موارد به‌جای بهترین جواب یا بهینه مطلق، به یک جواب رضایت‌بخش بسنده می‌کند. در صنعت امروز این بحث مطرح است که اگر قرار است قطعه‌ای، سازه‌ای، فرآیندی و یا سیستمی طراحی شود، چگونه تصمیمات تکنولوژیکی و مدیریتی به‌طور بهینه گرفته شود. هدف نهایی از چنین تصمیماتی کمینه کردن تلاش لازم و یا بیشینه کردن سود مورد نظر است. تلاش لازم یا سود مورد نظر در هر وضعیت علمی را می‌توان به‌صورت تابعی از متغیرهای تصمیم مشخص بیان کرد. بنابراین بهینه‌سازی را می‌توان به‌عنوان فرآیند یافتن شرایطی که مقدار بیشینه یا کمینه یک تابع را به دست می‌دهد، تعریف کرد. واضح است که اگر نقطه x منطبق بر مقدار بیشینه تابع $f(x)$ باشد، این نقطه بر مقدار کمینه تابع $-f(x)$ هم منطبق است. پس بدون از دست دادن کلیت، می‌توان بهینه‌سازی را به معنای کمینه‌سازی در نظر گرفت، زیرا بیشینه یک تابع را می‌توان به جستجوی کمینه منفی آن تابع پیدا کرد [۶۴].

۴-۲- بهینه‌سازی

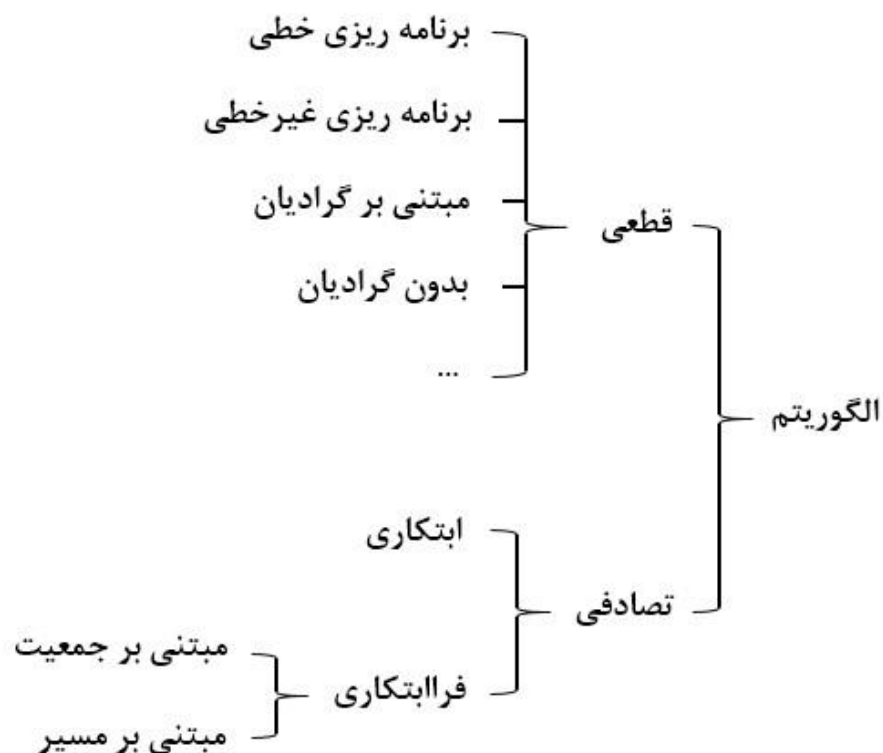
شکل (۴-۱) صورت کلی تقسیم‌بندی الگوریتم‌های بهینه‌سازی را نشان می‌دهد. در اینجا بهینه‌سازی هوشمند به دسته‌ی وسیعی از الگوریتم‌های بهینه‌سازی تصادفی مبتنی بر جمعیت اطلاق می‌شود که در این فصل به معرفی الگوریتم بهینه‌سازی ازدحام ذرات^۱، الگوریتم کرم شب‌تاب^۲ و الگوریتم خفاش^۳ پرداخته و در ادامه با استفاده از این الگوریتم‌ها عملیات بهینه‌سازی انجام داده می‌شود. اگر مسئله

^۱ Particle Swarm Optimization (PSO)

^۲ Genetic Algorithm (GA)

^۳ Bat Algorithm (BA)

بهینه‌سازی پیچیده، تصادفی و غیرخطی از درجه بالا با بهینه محلی چندگانه باشد، الگوریتم‌های بهینه سازی هوشمند در مقایسه با روش‌های بهینه‌سازی کلاسیک پر مزیت محسوب می‌شوند. از مزایای بهینه‌سازی هوشمند، توازی ذاتی‌شان، قابلیت حل کردن مسائل عظیم و پیچیده، کمترین نیاز به دانشی خاص در حیطه کار و غیره می‌توان اشاره کرد.



شکل ۴-۱ تقسیم‌بندی الگوریتم‌های بهینه‌سازی [۵۵]

روش‌های بهینه‌سازی و جستجو کلاسیک را می‌توان در دودسته؛ روش‌های جبری و روش‌های شمارشی طبقه‌بندی کرد. روش‌های جبری از جمله پرترفدارترین روش‌های بهینه‌سازی هستند. این روش‌ها را می‌توان در قالب روش‌های جستجو مستقیم^۱ و یا روش‌های جستجو غیرمستقیم^۲ طبقه‌بندی نمود. در روش‌های جستجو غیرمستقیم بهینه‌سازی تابع هدف با توجه به یک سری محدودیت‌های

¹ Direct Search

² Indirect Search

خاص (معادلات خطی و یا غیرخطی) انجام می‌شود. روش لاگرانژ یکی از روش‌های مرسوم در این راستا می‌باشد. این روش تعمیمی از روش‌های یافتن نقاط اکسترمم محلی در جبر مقدماتی می‌باشد. رویه یافتن اکسترمم‌های محلی بر مبنای مشتق مرتبه اول تابع هدف انجام می‌گیرد. درواقع با مساوی صفر قرار دادن مشتق مرتبه اول مکان این نقاط تعیین می‌گردد. تکنیک‌های بهینه‌سازی مستقیم به مقدار شیب محلی در راستای یافتن اکسترمم بستگی دارند. گرچه تلاش‌های زیادی برای ارتقا این روش‌ها صورت گرفته اما به‌رحال این روش‌ها دارای کمبودهایی می‌باشند. یکی از کمبودها این است که این روش‌ها در مقیاس محلی عمل می‌کنند و جستجو برای یافتن ماکسیمم و مینیمم در یک همسایگی محدود صورت می‌گیرد. به‌این‌ترتیب الگوریتم با یافتن اکسترمم‌های محلی از یافتن اکسترمم‌های سراسری باز می‌ماند. درواقع می‌توان این روش‌ها را یک روش ایده آل برای حل مسائلی با محدودیت‌های ایده آل و مشتقات پیوسته دانست. اما به‌رحال مسائل کاربردی دنیای واقعی پر از ناپیوستگی‌های متعدد و بی‌نظمی هستند که مدل‌سازی آن‌ها را بسیار دشوار می‌سازد. این محدودیت‌ها دانشمندان را به سمت یافتن یک راه‌حل جدید فارغ از محدودیت‌های جبری گسیل داشته است. تکنیک‌های شمارشی، بر مبنای جستجو در یک محدوده‌ی پیوسته نامتناهی یا یک محدوده با تعداد ناپیوستگی نامتناهی می‌باشد که به‌وسیله آن الگوریتم بهینه‌سازی هر یک از نقاط این محدوده را به‌صورت پی‌درپی مورد ارزیابی قرار می‌دهد. اگرچه الگوریتم شمارشی دارای طبیعت بسیار ساده‌ای می‌باشد و یک فرآیند منطقی به نظر می‌رسد اما بازدهی چندان مناسبی ندارد. زیرا در واقعیت دامنه جستجو بسیار گسترده و بزرگ می‌باشد و نمی‌توان بر روی کل دامنه فضای جستجو تمرکز کرد و تمام نقاط را مورد بررسی قرار داد. بنابراین می‌توان گفت عملکرد روش‌های شمارشی و جبری در ارتباط با مسائل دنیای واقعی چندان مناسب نیست [۶۵].

بیشتر روش‌های بهینه‌سازی کلاسیک، مخصوصاً روش‌های جستجوی مبتنی بر گرادیان، روش‌های جستجوی محلی هستند. جستجو معمولاً با یک حدس شروع شده و تلاش می‌کند تا کیفیت پاسخ‌ها را

بهبود بخشد و ممکن است در دام بهینه محلی بیفتد. برای خروج از بهینه محلی، از تغییرات معینی به وسیله اتفاق باید استفاده کرد. ثابت شده است که الگوریتم‌های بهینه‌سازی هوشمند در فرار از بهینه محلی و پیدا کردن بهینه سراسری در فضاهای جستجوی پیچیده و بزرگ مؤثر هستند و چنانچه زمان محاسبات نامحدود باشد دستیابی به بهینه سراسری را تضمین می‌کند [۶۴].

۳-۴- بیان ریاضی یک مسئله بهینه‌سازی

هر مسئله بهینه‌سازی برای اینکه مورد تحلیل قرار گیرد باید به صورت یک مسئله کمینه‌سازی ریاضی بیان گردد. به طور کلی یک مسئله کمینه‌سازی ریاضی را می‌توان به صورت زیر نوشت:

$$X = \begin{Bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{Bmatrix} \text{ بردار } \left. \begin{array}{l} \text{را به گونه‌ای بیابید که تابع } f(x) \text{ را به شرط قیدهای زیر کمینه کند:} \end{array} \right\}$$

$$\begin{aligned} \varphi_j(x) &\leq 0, & j &= 1, 2, \dots, j \\ \psi_k(x) &= 0, & k &= 1, 2, \dots, k \end{aligned} \quad (۱-۴)$$

که X یک بردار n بعدی است و بردار طراحی^۱ نامیده می‌شود. $f(x)$ تابع هدف^۲ و $\varphi_j(x)$ و $\psi_k(x)$ به ترتیب قیدهای نامساوی^۳ و قیدهای مساوی^۴ هستند. مسئله مطرح شده یک مسئله بهینه‌سازی مقید نامیده می‌شود. برخی از مسائل دارای قید نیستند و می‌توان آن‌ها را به صورت زیر بیان کرد:

$$X = \begin{Bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{Bmatrix} \text{ بردار } \left. \begin{array}{l} \text{را به گونه‌ای بیابید که تابع } f(x) \text{ را کمینه کند.} \end{array} \right\}$$

چنین مسائلی را مسائل بهینه‌سازی نامقید گویند.

¹ Design Vector

² Objective Function

³ Inequality Constraint

⁴ Equality Constraint

۴-۳-۱- بردار طراحی

هر سیستم با مجموعه‌ای از کمیت‌ها بیان می‌شود. همه کمیت‌هایی که به صورت متغیر بر رفتار سیستم تأثیر می‌گذارند، متغیر طراحی^۱ نامیده می‌شوند. این متغیرها را با x_i نشان می‌دهند که $i = 1, 2, \dots$

می‌باشد. مجموعه متغیرهای طراحی به صورت بردار طراحی $X = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix}$ ارائه می‌شوند.

۴-۳-۲- فضای طراحی^۲

یک فضای n بعدی مشخص را که هر محور مختصات آن بیانگر یک متغیر طراحی x_i که $i = 1, 2, \dots$ است، در نظر بگیرید. چنین فضایی را فضای طراحی گویند. هر نقطه در این فضای n بعدی، یک نقطه طراحی نامیده می‌شود. این نقطه یک جواب امکان‌پذیر یا امکان‌ناپذیر را برای مسئله طراحی ارائه می‌کند.

۴-۳-۳- قیدهای طراحی

در بسیاری از مسائل علمی نمی‌توان متغیرهای طراحی را به دلخواه انتخاب کرد، بلکه این متغیرها باید ویژگی‌های عملی مشخص و دیگر نیازمندی‌ها را برآورده نمایند. قیدهایی را که باید به منظور تهیه یک طرح مورد قبول آورده شوند، قیدهای طراحی گویند [۶۴].

۴-۴- پارامترهای کنترلی مشترک بین الگوریتم‌های بهینه‌سازی هوشمند

از جمله مهم‌ترین پارامترهای کنترلی الگوریتم‌های بهینه‌سازی مبتنی بر جمعیت که در این فصل سه مورد از آن‌ها مورد مطالعه قرار می‌گیرد، اندازه گروه^۳ و تعداد تکرارها^۴ می‌باشد. اندازه گروه همان تعداد اعضا در گروه است. هرچه تعداد بیشتر شود، محدوده اولیه گروه بزرگ‌تر می‌شود که برای

¹ Design Variable

² Design Space

³ Swarm size

⁴ Number of Iteration

مقداردهی اولیه اعضا استفاده می‌شود. هرچند، اعضای بیشتر پیچیدگی محاسبات در هر تکرار را افزایش می‌دهد اما گروه بزرگ، در هر تکرار قسمت‌های بزرگ‌تری را در فضای جستجو تحت پوشش می‌گیرد و این باعث می‌شود اعضای بیشتر بتوانند با تکرار کمتر به جواب خوب برسند. همچنین تعداد تکرارها برای رسیدن به یک جواب خوب، وابسته به مسئله است. تکرارهای خیلی کم ممکن است قبل از موعد، جستجو را خاتمه دهد و تکرارهای بیش‌ازحد، نتیجه‌ای جز اضافه شدن محاسبات پیچیده و غیرضروری را در پی ندارد.

۴-۴-۱- تابع هدف

روش‌های طراحی معمول، ما را دریافتن یک طرح قابل قبول یاری می‌دهند. این نوع طراحی تنها شرایط لازم برای جواب را برآورده می‌سازد. لیکن عموماً تنها قابل قبول بودن یک طرح موردنظر نیست، و هدف از بهینه‌سازی، انتخاب بهترین طرح از بین طرح‌های قابل قبول موجود می‌باشد. بنابراین باید معیاری برای مقایسه طرح‌های قابل قبول مختلف و انتخاب بهترین آن‌ها تعیین شود. چنین معیاری که طرح نسبت به آن بهینه می‌شود به صورت تابعی از متغیرهای طراحی بیان می‌شود که تابع معیار و یا تابع هدف نامیده می‌شود. انتخاب تابع هدف به طبیعت مسئله بستگی دارد.

در بعضی حالات لازم است که بیش از یک معیار به‌طور هم‌زمان برآورده شوند. یک مسئله بهینه‌سازی با توابع هدف چندگانه را با عنوان یک مسئله بهینه‌سازی با توابع هدف چندگانه^۱ می‌شناسند. باهدف‌های چندگانه پیچیدگی مسئله زیاد می‌شود. یک‌راه ساده برای حل چنین مسائلی این است که تابع هدف به‌عنوان ترکیب خطی از توابع هدف چندگانه در نظر گرفته شود. یعنی اگر $f_1(x)$ و $f_2(x)$ دو تابع هدف ممکن باشند، برای بهینه‌سازی، یک تابع هدف جدید به‌صورت زیر در نظر می‌گیریم:

^۱ Multiobjective Programming Problem

$$f(x) = w_1 n_1 f_1(x) + w_2 n_2 f_2(x) \quad (۲-۴)$$

n_1 و n_2 ضریب نرمال سازی توابع هدف هستند. معمولاً این مقادیر باتوجه به اندازه توابع هدف تنظیم می‌شوند و ارزش توابع را در حالت چندگانه یکسان می‌کنند. w_1 و w_2 مقادیر ثابتی هستند که بیانگر اهمیت یا وزن یک تابع هدف نسبت به دیگری در توابع هدف چندگانه است. تنظیم این ضرایب برحسب ارزشی که توابع برای طراح دارند تنظیم می‌گردد.

۴-۴-۲- معیار توقف^۱ و شاخص عملکرد الگوریتم‌های بهینه‌سازی هوشمند

آخرین مرحله الگوریتم مربوط به شرایط توقف است. به عبارت دیگر برای پایان دادن به تکرار جستجو از معیارهایی استفاده می‌شود. تعیین معیار توقف برای یک فرآیند بهینه‌سازی، یک مسئله مهم در الگوریتم بهینه‌سازی است. برای این منظور ممکن است معیارهای مختلفی مورد استفاده قرار بگیرد. شرط توقف را می‌توان به چند صورت تعریف کرد مثلاً بر روی حداکثر تعداد تکرار، یا اینکه ماکزیمم تغییر در بهترین شایستگی برای تعداد معینی از حرکت جمعیت، کمتر از یک مقدار کوچک تعریف شده باشد. تابع هدف مطرح شده مبنی بر معیار خطا است و عملکرد یک سیستم بهینه‌سازی برحسب معیار خطا مورد ارزیابی قرار می‌گیرد. تعدادی از این معیارها به شرح زیر می‌باشند [۵۶]:

- معیار انتگرال قدر مطلق خطا^۲:

$$I_{IAE} = \int_0^T |e(t)| dt \quad (۳-۴)$$

- معیار انتگرال مجذور خطا^۳:

^۱ Stopping criterion

^۲ Integral of Absolute Errors

^۳ Integral Square of Errors

$$I_{ISE} = \int_0^T e^2(t) dt \quad (4-4)$$

- معیار انتگرال حاصل ضرب زمان در قدر مطلق خطا^۱:

$$I_{ITAE} = \int_0^T t|e(t)| dt \quad (5-4)$$

- معیار میانگین مجذور خطا^۲:

$$MSE = \frac{1}{T} \int_0^T e^2(t) dt \quad (6-4)$$

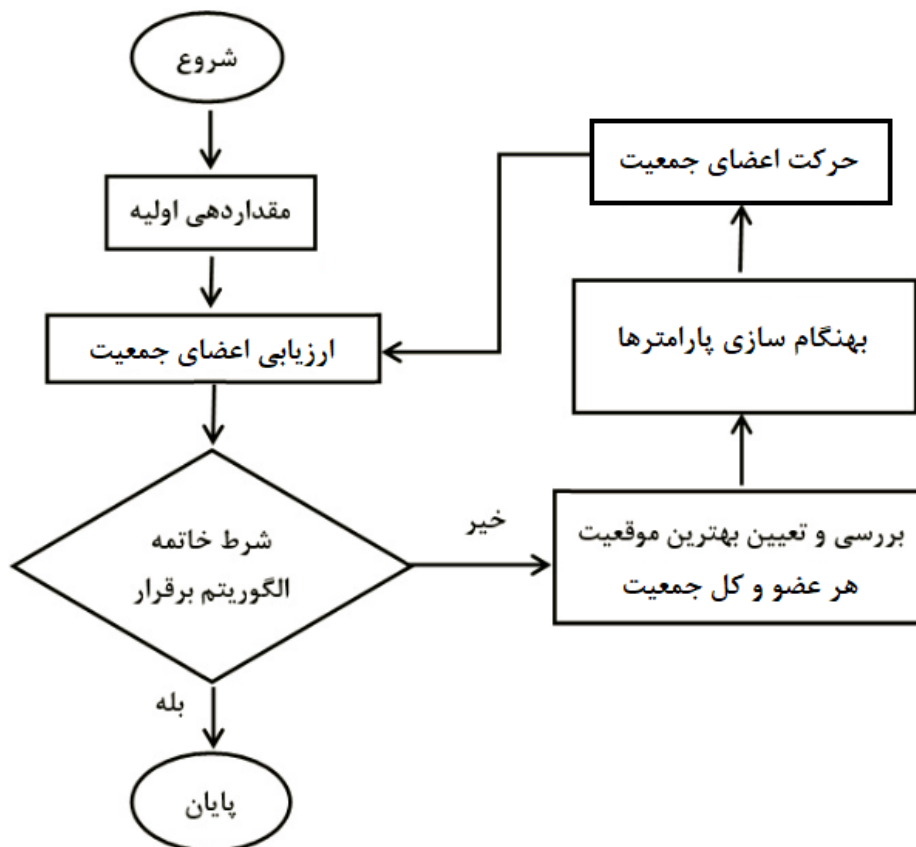
بین شاخص‌های فوق، انتگرال مجذور خطا و میانگین مجذور خطا متداول‌تر هستند.

فرآیند پیاده‌سازی الگوریتم‌های بهینه‌سازی مبتنی بر جمعیت به وسیله فلوجارت در شکل (۲-۴)

قابل مشاهده است.

¹ Integral of Time multiplied by Absolute Error

² Mean Square Errors



شکل ۴-۲ فلوجارت الگوریتم‌های بهینه‌سازی مبتنی بر جمعیت

۴-۵- الگوریتم بهینه‌سازی ازدحام ذرات

الگوریتم بهینه‌سازی ازدحام ذرات یا PSO یکی از تکنیک‌های بهینه‌سازی مبتنی بر قوانین احتمال است. ایده اولیه این الگوریتم در سال ۱۹۹۵ میلادی توسط کندی^۱ (روانشناس اجتماعی) و دکتر ابرهارت^۲ (دانشمند علوم کامپیوتر) و بر اساس رفتار گروهی مانند دسته‌های ماهی و پرندگان در طبیعت توسعه یافته است [۵۷]. الگوریتم بهینه‌سازی ازدحام ذرات یک الگوریتم جستجوی مستقیم (بدون گرادیان) است که یک جمعیت از ذرات به‌طور موازی برای مینیمم کردن یک تابع معلوم در فضای چندبعدی (n -بعدی)، بدون استفاده از اطلاعات گرادیان به جستجو می‌پردازد [۵۸]. آن‌ها ابتدا قصد

^۱ Kennedy

^۲ Eberhart

داشتند که با بهره‌گیری از مدل‌ها و روابط اجتماعی نوعی هوش محاسباتی را به وجود بیاورند که به توانایی‌های فردی ویژه نیازی نداشته باشد. فرض بر این است که گروهی از پرندگان به‌صورت تصادفی در یک منطقه به دنبال غذا می‌گردند درحالی‌که در یک منطقه از فضای جستجو^۱، غذا وجود دارد و پرندگان به‌این‌علت که از مکان غذا اطلاعی ندارند و فقط میزان فاصله خود تا آن مکان را می‌دانند به دنبال پرنده‌ای حرکت می‌کنند که نزدیک‌ترین فاصله تا غذا را دارد. در الگوریتم PSO هر پرنده، یک جواب ممکن در فضای مسئله است که ذره نامیده می‌شود. هر ذره دارای یک مقدار شایستگی^۲ است که توسط تابع هزینه مسئله محاسبه می‌شود و پرنده‌ای که به غذا نزدیک‌تر است شایستگی بیشتری دارد. به‌صورت نظری حداقل هر پرنده به‌عنوان یکی از اعضای گروه از تجربه خود و یافته‌های سایر اعضا برای یافتن غذا بهره می‌برد و با استفاده از این اطلاعات مسیر جهت حرکت خود را به سمت ذره بهینه هدایت می‌کند. در واقع الگوریتم بهینه‌سازی اجتماع ذرات که مقدار کمینه یک تابع را جستجو می‌کند، همانند دسته‌ای از پرندگان عمل می‌کنند که به دنبال غذا می‌گردند. هیچ‌کدام از ذرات به‌تنهایی قادر به حل مسئله نیستند بلکه وقتی می‌توان به حل مسئله امیدوار شد که ذرات با همدیگر در تعامل و ارتباط باشند و این مشارکت یک مزیت قطعی بر جستجوی رقابتی برای یافتن غذای توزیع شده است. پایه اصلی نظریه الگوریتم بهینه‌سازی اجتماع پرندگان همین تسهیم اطلاعات بین اعضای یک گروه است [۶۷].

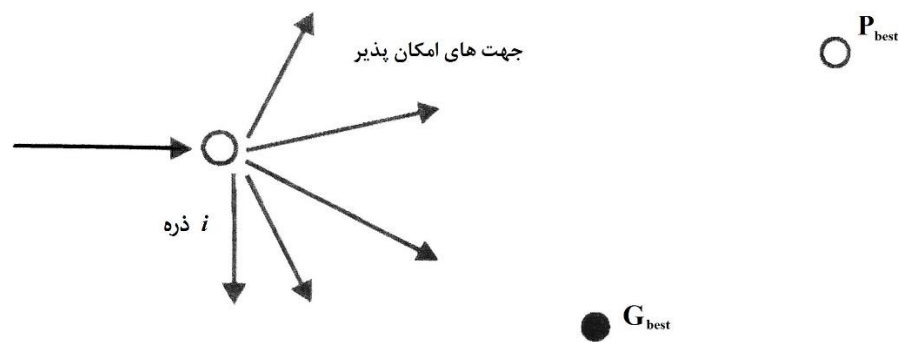
۴-۵-۱- عملکرد الگوریتم بهینه‌سازی ازدحام ذرات

این الگوریتم فضای تابع هدف را به‌وسیله تنظیم مدارهای عوامل منحصربه‌فرد که ذرات نامیده می‌شوند، به‌عنوان مسیرهای تکه‌ای تشکیل شده به‌وسیله بردارهای موقعیتی، به شیوه‌ای شبه اتفاقی جستجو می‌کند. حرکت ذره دارای دو جزء اصلی است: یک جزء اتفاقی و یک جزء قطعی. ذره درحالی‌که تمایل

¹ Search space

² Fitness

به حرکت اتفاقی دارد به سوی بهترین موقعیت موجود سراسری جذب می‌شود. هر ذره به صورت چندبعدی با دو مقدار V_{id} و X_{id} که به ترتیب معرف وضعیت مکانی و سرعت مربوط به بعد d ام از i امین ذره هستند تعریف می‌شود. وقتی ذره مکانی را می‌یابد که از مکان‌های یافته شده قبلی بهتر است، آن را به عنوان بهترین موقعیت موجود برای ذره i بهنگام می‌کند. این مقدار، بهترین جواب از لحاظ شایستگی است که P_{best} نامیده می‌شود. برای همه n ذره به طور جداگانه یک موقعیت به عنوان بهترین وجود دارد و هدف، یافتن بهترین در میان همه بهترین‌های موجود است. این مقدار بهترین کلی است و G_{best} نامیده می‌شود. تا زمانی که به شمار تکرارهای معینی برسیم یا به مرحله‌ای برسیم که تابع هدف دیگر بهبود نیابد این روند تکرار می‌شود [۵۹]. اگر ذره در یک همسایگی محلی خودش به عنوان ذره‌ای از جمعیت شرکت کند این مقدار تنها در همان همسایگی محاسبه می‌شود و بهترین محلی است که L_{best} نام دارد.



شکل ۳-۴ نمایش نموداری حرکت ذرات در روش ازدحام ذرات [۶۶]

حرکت ذرات به صورت شماتیک در شکل (۳-۴) نشان داده شده است. به طوری که P_{best} بهترین موقعیت موجود برای ذره i ام بوده و برای بهترین موقعیت سراسری داریم:

$$G_{best} \approx \min/\max\{f(x_i)\}, (i=1,2,\dots,n) \quad (۷-۴)$$

بعد از یافتن P_{best} و $G_{best}(L_{best})$ ، هر ذره سرعت و وضعیت مکانی خود را طبق روابط زیر بهنگام می‌کند:

$$V_i^{t+1} = wV_i^t + c_1r_1(P_{best_i}^t - x_i^t) + c_2r_2(G_{best}^t - x_i^t) \quad (۸-۴)$$

$$x_i^{t+1} = x_i^t + V_i^{t+1} \quad (۹-۴)$$

در این روابط:

x_i^t : وضعیت فعلی ذره i ام

x_i^{t+1} : وضعیت جدید ذره i ام

V_i^{t+1} : سرعتی که باعث تغییر موقعیت ذره از x_i^t به x_i^{t+1} می‌شود و بر اساس مقادیر رندمی از سرعت قبلی، بهترین ذره بین کل ذرات و بهترین موقعیت ذره i ام به دست می‌آید.

w : وزن اینرسی^۱ که وظیفه بسیار مهمی در همگرایی^۲ الگوریتم دارد. هرچه بزرگ‌تر باشد فضای بیشتری مورد بررسی قرار می‌گیرد. در شروع الگوریتم وزن اینرسی باید بزرگ باشد تا فضای گسترده‌ای مورد بررسی^۳ قرار گیرد و با افزایش تکرارها این مقدار کوچک شود تا موجب تمرکز^۴ جستجو در فضای کوچک در گام‌های پایانی گردد. ($0.2 < w < 0.9$)

r_1 و r_2 : اعداد تصادفی با توزیع یکنواخت که در بازه $(0, 1)$ تولید می‌شوند.

c_1 و c_2 : پارامترهای یادگیری یا ثابت‌های شتاب^۵ یا به ترتیب ضرایب وزنی برای بهترین تجربه شخصی هر ذره و بهترین تجربه عمومی تمام ذرات هستند که قابل تنظیم می‌باشند و به صورت معمول برابر با

¹ Inertia weight

² Convergence

³ Exploitation

⁴ Exploration

⁵ Learning factors

$c_1 \approx c_2 \approx 2$ در نظر گرفته می‌شود.

اگرچه سرعت می‌تواند هر مقداری داشته باشد، اما برای جلوگیری از واگرایی الگوریتم، مقدار نهایی سرعت هر ذره به یک بازه محدود می‌شود [۶۷]:

$$V_{id} \in [0, V_{max}] \quad (۱۰-۴)$$

۴-۵-۲- پیاده‌سازی الگوریتم بهینه‌سازی ازدحام ذرات

طبق شکل (۲-۴) مراحل اساسی پیاده‌سازی الگوریتم بهینه‌سازی ازدحام ذرات به صورت زیر قابل

بیان است:

- ۱) تعیین پارامترهای کنترلی
- ۲) مقداردهی اولیه: در طی این مرحله جمعیت به‌طور تصادفی مقداردهی اولیه می‌شوند و برای تمام ذرات موقعیت و سرعت به‌طور تصادفی ایجاد می‌شود.
- ۳) ارزیابی جمعیت اولیه: مقدار تابع هزینه به ازای هر ذره ارزیابی می‌گردد و G_{best} به‌عنوان بهترین ذره با بیشترین شایستگی انتخاب می‌شود.
- ۴) حرکت ذرات به سمت موقعیت جدید: جهت سرعت و حرکت هر ذره، بر اساس بهترین موقعیت خود ذره و بهترین موقعیت تمام ذرات طبق روابط (۴-۸) و (۴-۹) به‌روز می‌شود.
- ۵) چک کردن موقعیت جدید به محدوده‌های جستجو
- ۶) محاسبه تابع هدف برای موقعیت جدید تمامی ذرات و به‌روز کردن بهترین موقعیت هر ذره و بهترین موقعیت ذرات همسایه
- ۷) بررسی شرط همگرایی: اگر معیار توقف برآورده نشده باشد به مرحله ۴ برمی‌گردد و تکرار می‌شود. در غیر این صورت فرآیند بهینه‌سازی پایان می‌یابد.
- ۸) نتایج خروجی: بهترین پاسخ به‌دست‌آمده توسط این الگوریتم، خروجی است.

پیاده‌سازی الگوریتم بهینه‌سازی ازدحام ذرات بسیار ساده است و این سادگی موجب بالا رفتن سرعت

در محاسبات و رسیدن سریع به جواب دلخواه با حجم کم حافظه موردنیاز می‌باشد. در این الگوریتم تنها

G_{best} و P_{best} اطلاعات را به سایرین می‌دهند و در سیر تکامل، ذره‌ها تحت تأثیر بهترین جواب‌ها در اغلب موارد همگرا می‌شوند. همچنین این الگوریتم با توجه به پارامترهای کم آن، از لحاظ محاسباتی کم‌بار است.

۴-۶- الگوریتم بهینه‌سازی کرم شب‌تاب

الگوریتم کرم شب‌تاب یا به اختصار FA در اواخر سال ۲۰۰۷ توسط زین شی یانگ^۱ در دانشگاه کمبریج توسعه یافته است، که ایده اصلی آن از ارتباط نوری میان کرم‌های شب‌تاب که در مجموعه‌ها بزرگ در کنار هم زندگی می‌کنند الهام گرفته شده است [۶۰]. این الگوریتم را می‌توان از مظاهر هوش ازدحامی^۲ دانست، که در آن از همکاری (و احتمالاً رقابت) اعضای ساده و کم‌هوش، مرتبه بالاتری از هوشمندی ایجاد می‌شود که قطعاً توسط هیچ‌یک از اجزا قابل حصول نیست.

۴-۶-۱- عملکرد الگوریتم کرم شب‌تاب

کرم شب‌تاب موجودی است که از طریق حسگرها قادر به درک پیرامون خود بوده و از طریق تأثیرگذارنده‌ها می‌تواند روی محیط تأثیر بگذارد. الگوریتم کرم شب‌تاب الهام گرفته شده از مطالعات و مشاهدات روی کرم‌های شب‌تاب است. این مطالعات نشان داده که کرم‌های شب‌تاب حشراتی اجتماعی هستند که در کلونی‌ها زندگی می‌کنند و رفتار آن‌ها بیشتر در جهت بقاء کلونی است تا درجات بقاء یک جزء از آن. یکی از مهم‌ترین و جالب‌ترین رفتار کرم‌های شب‌تاب، رفتار آن‌ها برای یافتن غذا است و به‌ویژه چگونگی پیدا کردن کوتاه‌ترین مسیر میان منابع غذایی و آشپخانه. این نوع رفتار کرم‌های شب‌تاب دارای نوعی هوشمندی توده‌ای است که اخیراً مورد توجه دانشمندان قرار گرفته است. در هوشمندی اجتماعی عناصر میزانی از هوشمندی را دارا هستند درحالی‌که هوشمندی توده‌ای عناصر

¹ Xin-She Yang

² Swarm Intelligence

رفتاری تصادفی دارند و بین آنها هیچ نوع ارتباط مستقیمی وجود ندارد و آنها تنها به صورت غیرمستقیم و با استفاده از نشانه‌ها با یکدیگر در تماس هستند.

کرم‌های شب‌تاب با تولید نور چشمک‌زن کوتاه و موزون، باهم ارتباط برقرار می‌کنند. الگوی تابیدن معمولاً برای یک نوع خاص از کرم‌ها منحصر به فرد است. هر الگوی تابش برای هدفی خاص است و معیار سنجش تابیدن‌ها، میزان شدت و جذابیت نور و فاصله می‌باشد. یعنی کرم شب‌تابی که پرنورتر باشد جذاب‌تر و در نتیجه ثابت‌تر است. یعنی بقیه کرم‌های شب‌تاب به سمت آن جذب می‌شوند و این جذابیت بافاصله نسبت عکس دارد.

شدت نور^۱، در a یک فاصله خاص (r) از منبع نور، از قانون مربع معکوس پیروی می‌کند. گفته می‌شود که شدت نور (I) هنگامی که فاصله افزایش یابد، کاهش می‌یابد یا به عبارتی،

$$I \propto I/r^2 \quad (۱۱-۴)$$

است. علاوه بر این هوا نور را جذب می‌کند و نور با افزایش فاصله ضعیف و ضعیف‌تر می‌شود. این دو عامل ترکیبی، شتاب‌ها را در یک فاصله محدود بسیار دیدنی می‌کنند. معمولاً چند صد متر فاصله در شب برای ارتباطات شب‌تاب‌ها بسیار کافی است.

در الگوریتم کرم شب‌تاب دو پیامد مهم وجود دارد: نوسان شدت نور و فرمول‌بندی قابلیت جذب. برای سادگی همیشه می‌توان فرض کرد که قابلیت جذب یک کرم شب‌تاب به وسیله درخشندگی آن تعریف می‌شود که مجتمع با تابع هدفی که می‌خواهد بهینه شود، فرموله شده است. هر چند که قابلیت جذب نسبی است اما باید توسط دیگر کرم‌های شب‌تاب تشخیص داده شود. بنابراین بافاصله r_{ij} بین شب‌تاب i و شب‌تاب j تغییر خواهد کرد.

^۱ Intensity

علاوه بر این شدت نور با فاصله از منبع کاهش می‌یابد و نور توسط واسطه‌هایی مانند هوا جذب می‌شود.

شدت جذب $I(r)$ ، بر اساس قانون مربع معکوس زیر تغییر می‌کند،

$$I(r) = \frac{I_s}{r^2} \quad (12-4)$$

به طوری که I_s شدت نور در منبع می‌باشد. برای یک واسطه داده شده با ضریب جذب نور ثابت (γ) ، شدت نور (I) با فاصله (r) تغییر می‌کند، یعنی،

$$I = I_0 e^{-\gamma r} \quad (13-4)$$

I_0 شدت نور اولیه می‌باشد. برای اجتناب از تکینگی^۱ در $r = 0$ در رابطه (۱۲-۴)، اجرای مرکب قانون مربع معکوس و جذب می‌تواند به فرم گوسین زیر تخمین زده شود،

$$I(r) = I_0 e^{-\gamma r^2} \quad (14-4)$$

از آنجایی که قابلیت جذب یک کرم شب‌تاب با شدت نور دیده شده توسط کرم‌های شب‌تاب مجاور متناسب است، می‌توانیم قابلیت جذب را به صورت زیر تعریف کرد،

$$\beta = \beta_0 e^{-\gamma r^2} \quad (15-4)$$

برای راحتی در محاسبات می‌توان رابطه (۱۵-۴) را با استفاده از بسط تیلور به صورت زیر بازنویسی کرد،

$$\beta = \frac{\beta_0}{1 + \gamma r^2} \quad (16-4)$$

که β_0 جذابیت در $r = 0$ است. در واقع پارامتر β میزان کشش کرم i به سمت کرم j که در فاصله r از هم هستند را کنترل می‌کند.

¹ Singularity

روابط بالا، یک فاصله مشخصه $\Gamma = 1/\sqrt{\gamma}$ را تعریف می کنند که قابلیت جذب را به طور معنی دار از β_0 به $\beta_0 e^{-1}$ برای معادله (۴-۱۵) و $\beta_0/2$ برای رابطه (۴-۱۶) تغییر می دهد.

در پیاده سازی عملی، تابع قابلیت جذب کنندگی $\beta(r)$ ، می تواند هر تابع نزولی یکنواخت مانند تابع با فرم عمومی زیر باشد،

$$\beta(r) = \beta_0 e^{-\gamma r^m}, \quad (m \geq 1) \quad (17-4)$$

برای مقدار ثابت γ ، طول مشخصه به صورت زیر خواهد بود،

$$\Gamma = \gamma^{-1/m}, \quad m \rightarrow \infty \quad (18-4)$$

بالعکس، برای طول مقیاس Γ داده شده در یک مسئله بهینه سازی، پارامتر γ می تواند به عنوان یک مقدار اولیه معمول زیر استفاده شود،

$$\gamma = \frac{1}{\Gamma^m} \quad (19-4)$$

m همواره باید عددی مثبت باشد. اگر $m=0$ باشد یعنی قابلیت جذب در کل فضا ثابت است. هرچه این مقدار بزرگ تر شود، قابلیت جذب به سرعت ضعیف می شود. اگر مقدار m به صفر نزدیک شود، قابلیت جذب افت پیدا نمی کند.

فاصله بین دو شب تاب i و j واقع در مکان های x_i و x_j ، فاصله کارتیزین به صورت زیر می باشد،

$$r_{ij} = \|x_i - x_j\| = \sqrt{\sum_{k=1}^d (x_{i,k} - x_{j,k})^2} \quad (20-4)$$

به طوری که $x_{i,k}$ ، k امین عنصر مختصات فضایی x_i ، برای شتاب نام است. فاصله در حالت دوبعدی به صورت زیر به دست می آید،

$$r_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} \quad (21-4)$$

حرکت شب‌تاب i به سمت شب‌تاب بیش جذب‌کننده (درخشان‌تر) j با استفاده از رابطه زیر تعریف می‌شود،

$$x_i = x_i + \beta_0 e^{-\gamma r_{ij}^2} (x_j - x_i) + \alpha \varepsilon_i \quad (۲۲-۴)$$

عبارت دوم به علت جذب می‌باشد، عبارت سوم تصادفی سازی بوده و α پارامتر تصادفی سازی است. همچنین ε_i یک بردار اتفاقی با توزیع نرمال یا یکنواخت می‌باشد. در بیشتر پیاده‌سازی‌ها، $\beta_0 = 1$ و $\alpha \in [0,1]$ در نظر گرفته می‌شود.

پارامتر γ تغییرات قابلیت جذب‌کنندگی را مشخص کرده و مقدار آن به شدت برای تعیین کردن سرعت همگرایی و رفتار الگوریتم کرم شب‌تاب مهم است. در تئوری $\gamma \in [0, \infty]$ است، اما در عمل به وسیله طول مشخصه Γ برای سیستمی که بهینه می‌گردد، تعریف می‌شود. بنابراین در بسیاری از کاربردها مقدار آن از 0.1 تا 10 متغیر است.

فاصله r تعریف شده در بالا، محدود به فاصله اقلیدسی نمی‌شود بلکه می‌توان بر مبنای نوع مسئله، فاصله را در یک فضای n -بعدی تعریف کرد. در واقع، هر مشخصه‌ای که بتواند میزان بهره را به طور مؤثری مشخص نماید، می‌تواند به عنوان رابطه r در نظر گرفته شود.

مقیاس متداول Γ باید با مقیاس مربوط به مسئله بهینه‌سازی پیوند داشته باشد. اگر Γ ، برای هر تعداد زیاد کرم‌های شب‌تاب $m \gg n$ (که m تعداد بهینه محلی است) مقیاسی متداول برای مسئله بهینه‌سازی داده شده باشد، آنگاه مکان‌های اولیه این n کرم شب‌تاب، می‌باید نسبتاً به طور یکنواخت بر روی همه فضای جستجو توزیع شود. همچنان که تکرارها ادامه می‌یابند، کرم‌های شب‌تاب تمایل به همگرایی در نقاط بهینه محلی خواهند داشت (که شامل نقطه بهینه مطلق نیز هست). با مقایسه بهترین پاسخ‌ها در میان همه این مقادیر بهینه، بهینه مطلق به راحتی به دست خواهد آمد. تحقیقات نشان می‌دهد هنگامی

که $n \rightarrow \infty$ و $t \gg 1$ باشد، الگوریتم کرم شب‌تاب به جواب بهینه مطلق دست می‌یابد و بسیار سریع همگرا می‌شود.

اگر ضریب جذب نور خیلی کوچک باشد ($\gamma \rightarrow 0$)، قابلیت جذب‌کنندگی مقدار ثابت $\beta = \beta_0$ بوده و $\Gamma \rightarrow \infty$ است. یعنی یک شب‌تاب چشم‌ک‌زن در هر کجای قلمرو قابل‌دیدن می‌باشد. پس یک بهینه منفرد (معمولاً مطلق) به راحتی می‌تواند به دست آید. در این صورت الگوریتم کرم شب‌تاب به نوع خاصی از الگوریتم ازدحام ذرات شتابدار تبدیل می‌شود. اگر ضریب جذب نور خیلی بزرگ باشد ($\gamma \rightarrow \infty$)، منجر به $\Gamma \rightarrow 0$ و $\beta(r) = \delta(r)$ شده که تابع دلتای دیراک بوده و به این معنی است که قابلیت جذب‌کنندگی از دید دیگر شب‌تاب‌ها تقریباً صفر است. این معادل باحالتی است که یک کرم شب‌تاب به صورت اتفاقی در یک ناحیه با مه غلیظ پرسه می‌زند. در این حالت، دیگر شب‌تاب‌ها نمی‌توانند ببینند و هر شب‌تاب در یک مسیر کاملاً تصادفی پرسه می‌زنند. در این صورت الگوریتم کرم شب‌تاب برابر با یک جستجوی اتفاقی کامل می‌باشد.

از آنجایی که الگوریتم کرم شب‌تاب، معمولاً در بین این دو کرانه واقع می‌شود، بنابراین امکان تنظیم α و γ به گونه‌ای که بهتر از الگوریتم‌های جستجو اتفاقی و الگوریتم ازدحام ذرات عمل کند، وجود دارد. با تغییر پارامتر تصادفی سازی α ، می‌توان همگرایی الگوریتم را بهبود بخشید. بهبود زمانی اتفاق می‌افتد که با نزدیک شدن به پاسخ بهینه این پارامتر کاهش یابد. برای این منظور می‌توان از تابع زیر استفاده کرد،

$$\alpha = \alpha_{\infty} + (\alpha_0 - \alpha_{\infty})e^{-t} \quad (23-4)$$

$t \in [0, t_{max}]$ زمان تقریبی برای شبیه‌سازی و t_{max} بیشترین تعداد نسل‌هاست. α_0 پارامتر اولیه و α_{∞} پارامتر نهایی تصادفی‌سازی است. همچنین می‌توان از تابعی ساده به صورت زیر استفاده کرد،

$$\alpha = \alpha_0 \theta^t \quad (24-4)$$

$\theta \in (0,1]$ ثابت کاهش تصادفی است.

در الگوریتم کرم شب تاب، به طور صریح از بهترین مقدار مطلق (g_*) استفاده نشده است، بلکه تنها از آن برای کد کردن بهترین پاسخهای نهایی استفاده شده است. مطالعات نشان می دهد که اگر عبارت اضافی $\lambda \varepsilon_i (x_i - g_*)$ را به رابطه (۴-۲۲) اضافه کنیم، کارایی به صورت معنی دار افزایش می یابد که λ پارامتری مشابه با α و β بوده و ε_i مجموعه ای از اعداد تصادفی است [۶۴].

۴-۶-۲- پیاده سازی الگوریتم کرم شب تاب

نور چشمک زن می تواند به صورت مجتمع با تابع هدفی که می خواهد بهینه شود، فرموله گردد که خود باعث ایجاد یک الگوریتم بهینه سازی می شود. به منظور توسعه الگوریتم الهامی از کرم شب تاب، خصوصیات تشعشع کرم های شب تاب دلخواه سازی می شود. از سه قانون دلخواه سازی زیر استفاده می شود:

- همه کرم های شب تاب متحدالجنس^۱ هستند به طوری که یک شب تاب، به سمت دیگر شب تابها صرف نظر از جنسشان، جذب خواهد شد.
- قابلیت جذب کنندگی^۲ متناسب با درخشندگی^۳ آنها است. بنابراین برای هر دو شب تاب چشمک زن، شب تاب با درخشندگی کم به سوی شب تاب پرنورتر حرکت خواهد کرد. جذب کنندگی متناسب با درخشندگی بوده و هر دوی آنها هنگامی که فاصله افزایش یابد، کاهش می یابند.
- درخشندگی یک کرم شب تاب، به وسیله دورنمای تابع هدف تعیین می شود.

برای یک مسئله پیشینه سازی، درخشندگی به سادگی می تواند متناسب با مقدار تابع هدف تعیین شود.

¹ Unisex

² Attractiveness

³ Bioluminescent

بر اساس این سه قانون، مبنای مراحل الگوریتم کرم شب‌تاب طبق فلوجارت شکل (۴-۲)، به صورت زیر خلاصه می‌شود:

۱. تولید جمعیت اولیه کرم‌های شب‌تاب با ویژگی‌های زیر:
 - موقعیت: که بین حد پایین و حد بالای تعیین شده می‌باشد.
 - شدت نور: برای اعضای جمعیت اولیه به صورت رندم از بازه $[0,1]$ انتخاب می‌شود.
 - کیفیت: میزان برازندگی هر کرم شب‌تاب که از روی تابع هدف محاسبه می‌شود.
۲. به ازای $i=1, \dots, n$ شدت نور I_i در x_i و قابلیت جذب β ایجاد می‌شوند.
۳. موقعیت مقایسه شده و جواب بهینه به دست می‌آید.
۴. طبق رابطه ۴-۲۲ موقعیت‌های جدید موقتی با تنظیم قابلیت جذب و به روز کردن شدت نور همه کرم‌های شب‌تاب ایجاد می‌شود.
۵. به ازای $i=1, \dots, n$ و $j=1, \dots, n$ اگر $I_j > I_i$ برقرار باشد، کرم شب‌تاب i به سمت کرم شب‌تاب j حرکت می‌کند.
۶. یک جواب جدید با حرکت تصادفی ایجاد می‌شود.
۷. اگر شروط $rand < I_i$ و $f(x_i) < f(x^*)$ برقرار باشند، آنگاه جواب‌های جدید را می‌پذیرد و طبق روابط ۴-۱۳ و ۴-۱۵، شدت نور و قابلیت جذب بهنگام می‌شوند.
۸. کرم‌های شب‌تاب مرتب‌شده و بهترین جواب x^* پیدا می‌شود.
۹. اگر t به ماکزیمم مقدار خود رسید الگوریتم متوقف می‌شود در غیر این صورت $t=t+1$ قرار می‌گیرد و الگوریتم به مرحله ۴ بازمی‌گردد.

۴-۷- الگوریتم بهینه‌سازی خفاش

بسیاری از الگوریتم‌های ابتکاری و فراابتکاری، از رفتار سیستم‌های بیولوژیک یا سیستم‌های فیزیکی در طبیعت مشتق شده‌اند. الگوریتم بهینه‌سازی خفاش (BA)، در سال ۲۰۱۰ توسط پروفیسور یانگ معرفی گردید [۶۱]. این الگوریتم الهام گرفته شده از زندگی خفاش‌ها و بر اساس رفتار مکان‌یابی اشیا با استفاده از محاسبه زمان بازگشت اکو^۱ در خفاش‌ها است. خفاش‌ها می‌توانند شکار خود و انواع مختلف حشرات را حتی در تاریکی کامل نیز بیابند. ارسال پالس بلند و دریافت آن توسط خفاش‌ها، به آن‌ها آگاهی از جزئیات پیرامون و توانایی برای شکار و عبور از موانع را می‌دهد. ریز خفاش‌ها از نوعی رادار صوتی برای شناسایی شکار، اجتناب از برخورد با موانع و یافتن درز محل زندگی استفاده می‌کنند که مکان‌یابی با اکو نامیده می‌شود. خفاش‌ها پالسی متغیر (متناسب با استراتژی شکار - پالس بیشتر در نزدیکی طعمه) با صدای متغیر (بلندترین در هنگام جستجوی طعمه تا مقداری آرام‌تر هنگام نزدیک شدن به طعمه) را مخابره کرده و به اکویی که از برخورد آن با اشیای اطراف ایجاد می‌شود گوش می‌دهند.

اگرچه هر پالس اکو کمتر از یک‌هزارم ثانیه به طول می‌انجامد، اما دارای فرکانس ثابت (به‌طور معمول بین 150-25 KHz است. قدرت شنوایی خفاش‌ها به‌طور معمول در حدود 300-400 μs است و پالس مخابره شده می‌تواند به بلندی 110 db باشد. از آنجایی که سرعت صوت در هوا حدود $v=340 m/s$ است، بنابراین طول موج λ برای انفجار مافوق صوت (خفاش‌ها ۱۰ تا ۲۰ انفجار صوتی را در هر ثانیه مخابره می‌کنند) با فرکانس ثابت f برابر است با:

$$\lambda = \frac{V}{f} \quad (11-4)$$

که برای فرکانس‌های بین 25-50 Hz دارای محدوده 2-14 mm است.

¹ Echolocation

الگوریتم خفاش برای بهینه‌سازی مسائل مختلف از قبیل مسائل دودویی و مسائل بهینه‌سازی چند هدف کاربرد داشته است [۶۲].

۴-۷-۱- عملکرد الگوریتم خفاش

خفاش‌ها همگی برای جستجوی طعمه، با سرعت v_i ، در مکان x_i ، با فرکانس ثابت f_{min} ، طول موج متغیر λ و بلندی صدای A_0 پرواز می‌کنند. آن‌ها می‌توانند به صورت اتوماتیک طول موج یا فرکانس پالس‌های انتشار یافته خود را تنظیم کنند و بر مبنای نزدیکی به هدف، سرعت انتشار پالس‌ها $r \in [0,1]$ را تعدیل کنند.

اگرچه بلندی صدا می‌تواند به شیوه‌های مختلف تغییر کند اما فرض می‌کنیم که بلندی صدا از یک مقدار بزرگ (مثبت) A_0 تا مقدار ثابت مینیمم A_{min} تغییر می‌کند،

$$A_{min} \leq \text{بلندی صدا} \leq A_0 \quad (۲۵-۴)$$

فرکانس f در محدوده $[f_{min}, f_{max}]$ برابر با محدوده طول موج $[\lambda_{min}, \lambda_{max}]$ است. سرعت پالس به سادگی می‌تواند در محدوده $[0,1]$ در نظر گرفته شود به طوری که 0 به معنای این است که اصلاً پالسی وجود ندارد و 1 به معنای بیشترین سرعت انتشار پالس است.

برای بهنگام شدن مکان x_i و سرعت v_i خفاش‌ها در فضای جستجو d -بعدی داریم:

$$\begin{cases} f_i = f_{min} + (f_{max} - f_{min})\beta \\ v_i^t = v_i^{t-1} + (x_i^t - x_*)f_i \\ x_i^t = x_i^{t-1} + v_i^t \end{cases} \quad (۲۶-۴)$$

به طوری که $\beta \in [0,1]$ یک بردار اتفاقی از یک توزیع یکنواخت است. x_* بهترین مکان (پاسخ) مطلق جاری بوده که بعد از مقایسه همه پاسخ‌ها در میان همه n خفاش واقع می‌شود. f_i اصولاً گام و محدوده

^۱ Loudness

حرکت ذرات را کنترل می‌کند (فرکانس تصادفی است). در ابتدای الگوریتم، هر خفاش یک فرکانس تصادفی از $[f_{min}, f_{max}]$ و مقادیر مختلفی از بلندی صدا (بلندی صدای اولیه A_i^0 به‌طور تصادفی در بازه $[0, 1]$) و سرعت انتشار پالس (به‌طور تصادفی در حدود صفر) را تعیین می‌کند.

برای بخش جستجوی محلی، یک‌بار یک پاسخ از میان بهترین پاسخ‌های موجود انتخاب می‌شود و سپس پاسخ جدید برای هر خفاش با استفاده از قدم تصادفی تولید می‌شود،

$$x_{new} = x_{old} + \varepsilon A^t \quad (27-4)$$

به‌گونه‌ای که $\varepsilon \in [-1, 1]$ یک عدد تصادفی بوده و $A^t = A_i^t$ متوسط بلندی صدا برای همه خفاش‌ها می‌باشد.

همچنان که تکرارها به‌پیش می‌روند، بلندی صدا A_i و نرخ انتشار پالس r_i باید به‌نگام شوند. معمولاً زمانی که خفاش طعمه خود را می‌یابد بلندی صدا کم می‌شود درحالی‌که نرخ انتشار پالس افزایش می‌یابد،

$$\begin{aligned} A_i^{t+1} &= \alpha A_i^t \\ r_i^{t+1} &= r_i^0 [1 - \exp(-\gamma t)] \end{aligned} \quad (28-4)$$

α نرخ کاهش A_i و γ نرخ افزایش r_i است. برای $0 < \alpha < 1$ و $\gamma > 0$ داریم،

$$A_i^t \rightarrow 0, \quad r_i^t \rightarrow r_i^0 \quad \text{as } t \rightarrow 0 \quad (29-4)$$

α و γ ثابت هستند. برای سادگی می‌توان $\gamma = \alpha$ در نظر گرفت.

۴-۷-۲- پیاده‌سازی الگوریتم خفاش

مراحل پیاده‌سازی الگوریتم خفاش طبق فلوچارت ارائه‌شده در شکل (۴-۲) به شرح زیر می‌باشد:

۱. تولید جمعیت اولیه خفاش‌ها با ویژگی‌های زیر:

- موقعیت: که بین حد پایین و حد بالای تعیین شده می‌باشد.

- سرعت: برای اعضای جمعیت اولیه سرعت صفر است.
 - کیفیت: میزان برازندگی هر خفاش که از روی تابع هدف محاسبه می‌شود.
۲. به ازای $i=1, \dots, n$ سرعت v_i ، فرکانس f_i در x_i ، نرخ پالس r_i و بلندی صدای A_i ایجاد می‌شوند.
۳. موقعیت مقایسه شده و جواب بهینه به دست می‌آید.
۴. طبق رابطه ۴-۲۶ موقعیت‌های جدید موقتی با تنظیم فرکانس و به‌روز کردن سرعت همه خفاش‌ها ایجاد می‌شود.
۵. اگر شرط $rand(0,1) > r_i$ برقرار باشد، آنگاه یک جواب در میان بهترین جواب‌ها با گام تصادفی انتخاب شده و با استفاده از رابطه (۴-۲۷) یک جواب محلی در اطراف بهترین جواب انتخاب شده ایجاد می‌شود.
۶. یک جواب جدید با پرواز تصادفی ایجاد می‌شود.
۷. اگر شروط $rand < A_i$ و $f(x_i) < f(x^*)$ برقرار باشند، آنگاه جواب‌های جدید را می‌پذیرد و طبق رابطه ۴-۱۵، r_i افزایش و A_i کاهش می‌یابد.
۸. خفاش‌ها مرتب‌شده و بهترین جواب x^* پیدا می‌شود.
۹. اگر t به ماکزیمم مقدار خود رسید الگوریتم متوقف می‌شود در غیر این صورت $t=t+1$ قرار می‌گیرد و الگوریتم به مرحله ۴ بازمی‌گردد.

فصل پنجم

کنترل کننده غیر خطی مقاوم

همواره مشکلات زیادی در پیاده‌سازی کنترل‌کننده‌های مبتنی بر مدل روی سیستم‌های حقیقی مانند ربات‌ها وجود دارد. یکی از مهم‌ترین این مشکلات، عدم توانایی در مدل‌سازی دقیق سیستم‌های حقیقی است. بعلاوه، حتی اگر این توانایی وجود داشته باشد، مدل حاصل آن قدر پیچیده است که طراحی کنترل‌کننده را دشوار می‌سازد. عدم دقت در مدل‌سازی، ناشی از دو مسئله است؛ یکی عدم قطعیت در پارامترهای مدل (عدم قطعیت پارامتری) و دیگری دینامیک‌های مدل نشده سیستم می‌باشد. روش‌های کنترلی بسیاری جهت مقابله با این عدم قطعیت‌ها در مقالات دیده می‌شود.

کنترل مقاوم یکی از اصلی‌ترین و پرکاربردترین روش‌هایی است که برای غلبه بر عدم قطعیت‌ها و اغتشاشات خارجی پیشنهاد شده است. طرح‌های قبلی کنترل ربات مبتنی بر راهبرد کنترل گشتاور می‌باشند. در این راهبرد گشتاور مفاصل ربات به‌عنوان فرمان کنترل یا ورودی سیستم است. اما در عمل، فرمان گشتاور نمی‌تواند به‌طور مستقیم به ورودی ربات اعمال گردد و در اجرا نیازمند وجود محرکه‌ها است. این راهبرد از دینامیک موتورها در قانون کنترل و اجرای فرمان چشم‌پوشی می‌کند. این عیب، کارآمدی ربات را برای انجام کارهای سریع و دقیق زیر سؤال می‌برد. بعلاوه، راهبرد کنترل گشتاور به خاطر دینامیک ربات، به‌شدت در معرض عدم قطعیت‌ها مخصوصاً اغتشاش بار مکانیکی می‌باشد که اغتشاش خارجی سیستم محسوب می‌شود. عدم قطعیت پارامتری و دینامیک‌های مدل نشده در سیستم تغییرات و تأثیرات ثابتی به وجود می‌آورند لذا در تکرارپذیر بودن سیستم و حلقه کنترلی خللی به وجود نمی‌آورند. ولی هنگامی که سیستم در معرض اغتشاشات خارجی باشد، درواقع با توجه به اینکه این اغتشاشات متغیر با زمان بوده و اطلاعاتی از آن‌ها در دسترس نیست، تکرارپذیری حلقه کنترلی با مشکل مواجه می‌شود.

بااینکه در رویکرد کنترل ولتاژ، دینامیک موتورها نیز منظور می‌گردد اما طرح کنترلی در مقایسه با رویکرد کنترل گشتاور ساده‌تر است. در راهبرد کنترل ولتاژ، ولتاژ موتورها به‌عنوان ورودی کنترل سیستم

رباتیک عمل می‌نماید. درحالی‌که در راهبرد مرسوم کنترل گشتاور، گشتاور مفاصل به‌عنوان ورودی کنترلی سیستم در نظر گرفته شده است. همچنین در رویکرد کنترل ولتاژ، کنترل ولتاژ موتورها به‌صورت مستقیم می‌باشد که باعث حذف یک حلقه کنترل جریان در کنترل گشتاور موتور می‌شود. از آنجایی که اغتشاش بار مکانیکی تأثیر خود را روی جریان نشان می‌دهند و در رویکرد کنترل ولتاژ، حلقه کنترل جریان حذف شده است، تا حد خیلی زیادی این رویکرد مصون از اغتشاشات خارجی می‌باشد. پس با توجه به کند بودن تغییرات پارامترهای داخلی موتور الکتریکی و تکراری بودن تغییرات و تأثیرات دینامیک‌های مدل نشده و عدم قطعیت‌های پارامتری، می‌توان با پیشنهاد یک قانون کنترل قوی و مقاوم در راهبرد کنترل ولتاژ، تکرارپذیر بودن سیستم حلقه بسته کنترل را تضمین نمود.

۵-۲- کنترل کننده غیرخطی مقاوم

همان‌طور که در فصل ۳ اشاره شد روابط حاکم بر کنترل ربات با راهبرد کنترل ولتاژ به‌قرار زیر است:

معادله کلی ربات به شکل رابطه (۱-۵) است،

$$D(q)\ddot{q} + C(q, \dot{q})\dot{q} + G(q) = \tau_l \quad (۱-۵)$$

که ماتریس $D(q)$ ماتریس $n \times n$ اینرسی، $C(q, \dot{q})\dot{q}$ بردار $n \times 1$ نیروی جانب مرکز و کوریولیس و $G(q)$ بردار $n \times 1$ نیروی گرانشی و τ_l بردار $n \times 1$ گشتاور مفاصل برای بازوی ربات می‌باشند.

موتور الکتریکی، گشتاور مفاصل را با رابطه زیر تأمین می‌کند.

$$J_m \ddot{\theta}_m + B_m \dot{\theta}_m + r \tau_l = \tau_m \quad (۲-۵)$$

که در آن J_m ماتریس قطری $n \times n$ ممان اینرسی، B_m ماتریس قطری $n \times n$ ضریب میرایی، θ_m بردار $n \times 1$ موقعیت زاویه‌ای، τ_m بردار گشتاور موتورها و r ماتریس قطری $n \times n$ ضرایب چرخ‌دنده‌های موتورها می‌باشند. رابطه بین موقعیت موتور و موقعیت ربات را می‌توان به‌صورت زیر نوشت:

$$q = r\theta_m \quad (3-5)$$

همان‌طور که در فصل ۳ اشاره شد فرم فضای حالت معادلات ربات با خطی سازی فیدبکی به دست می‌آیند. روابط (۷-۳) تا (۹-۳) که نشان می‌دهند سیستم ربات چند متغیره بسیار غیرخطی با تزویج زیاد و ماتریس حالت به فرم غیرهمراه است که چالش‌های زیادی در کنترل ربات به وجود آورده است. همچنین در روابط فضای حالت، ماتریس‌های مدل ربات (۱-۵) ظاهر می‌شوند که از لحاظ عملی به‌طور دقیق در دسترس نیستند. برای همین بر اساس راهبرد کنترل ولتاژ از مدل موتور در ساده‌سازی معادلات استفاده می‌شود و روش کنترلی بر اساس راهبرد کنترل ولتاژ که ربات را به سیستم تک ورودی-تک خروجی تبدیل کرده است و کنترل هر مفصل را جداگانه انجام می‌دهد ارائه می‌شود. بنابراین معادلات برای موتور و موقعیت یک رابط برای یک مفصل نوشته می‌شوند و در نهایت قابل تعمیم به بردار $n \times 1$ برای n رابط هستند. در واقع تمام روابط به‌صورت اسکالر و برای مفصل i ام نوشته شده‌اند و برای سادگی از آوردن اندیس i خودداری شده است. همچنین تمام معادلات دینامیکی ربات را به‌صورت یک‌بار روی مدل موتور الکتریکی در نظر می‌گیریم. در این صورت تأثیر همه‌ی دینامیک‌های مدل شده و نشده ربات روی جریان موتور دیده می‌شوند و برای کنترل ربات کافی است معادله‌ی الکتریکی موتور کنترل شود. ورودی سیستم در راهبرد کنترل ولتاژ، ولتاژ موتور است. معادله ولتاژ موتور را در نظر می‌گیریم،

$$RI_a + LI_a + \alpha \dot{q} = v \quad , \quad \alpha = k_b r^{-1} \quad (4-5)$$

که در آن R ماتریس قطری $n \times n$ مقاومت آرمیچر، L ماتریس قطری $n \times n$ اندوکتانس، k_b ماتریس قطری $n \times n$ ضریب عکس‌العمل آرمیچر، و I_a بردار $n \times 1$ جریان و v بردار $n \times 1$ ولتاژ موتورها می‌باشند. پارامترهای R و L و α مقادیر نامعلومی دارند. رابطه (۴-۵) به شکل زیر بازنویسی می‌کنیم،

$$\hat{R}I_a + \hat{\alpha}\dot{q} + d = v \quad (5-5)$$

\hat{R} و $\hat{\alpha}$ مقادیر معلوم از پارامترهای موتور هستند که توسط طراح مشخص می‌شود. با استفاده از روابط (۴-۵) و (۵-۵)، d به دست می‌آید،

$$d = L\dot{I}_a + (R - \hat{R})I_a + (\alpha - \hat{\alpha})\dot{q} \quad (6-5)$$

d شامل دینامیک‌های مدل نشده سیستم است و پارامترهای آن مقادیر ثابتی دارند. $L\dot{I}_a$ نویز فرکانس بالا تولید می‌کند و در بخش بعدی ثابت می‌شود که I_a و \dot{I}_a محدود هستند.

با توجه به روابط بالا سیگنال کنترل را پیشنهاد می‌دهیم،

$$v = \hat{R}I_a + \hat{\alpha}(\dot{q}_d + k_p e) + \rho \operatorname{sgn}(s) \quad (7-5)$$

که k_p ضریب تناسبی خطا، $e = q_d - q$ خطای ردگیری و $\rho \operatorname{sgn}(s)$ برای جبران عدم قطعیت و مقاوم‌سازی سیگنال کنترل است. با جایگذاری رابطه (7-5) در (5-5) و جایگذاری (6-5) در آن، سیستم حلقه بسته به صورت زیر بدست می‌آید،

$$\hat{\alpha}(\dot{q}_d + k_p e) = -\rho \operatorname{sgn}(s) + L\dot{I}_a + (R - \hat{R})I_a + (\alpha - \hat{\alpha})\dot{q} \quad (8-5)$$

تمام پارامترهای سیستم حلقه بسته تغییرات ثابتی دارند. بدین ترتیب سیستم حلقه بسته تکرارپذیر خواهد بود. ρ مقدار ثابتی ندارد و با استفاده از تابع شبه لیاپانوف به دست می‌آید. تعریف می‌کنیم،

$$s = \int \hat{\alpha}(\dot{e} + k_p e) dt \quad , \quad \dot{s} = \hat{\alpha}(\dot{e} + k_p e) \quad , \quad e = q_d - q \quad , \quad \dot{e} = \dot{q}_d - \dot{q} \quad (9-5)$$

تابع مثبت معین (شبه لیاپانوف) را به صورت زیر تعریف می‌کنیم،

$$V = \frac{1}{2}s^2 \quad , \quad \dot{V} = s\dot{s} \quad (10-5)$$

اگر $\dot{V} \leq 0$ باشد تضمین می‌کند $s \rightarrow 0$ ، در نتیجه $e \rightarrow 0$ خواهد رفت و سیستم پایدار خواهد بود. با پیشنهاد $\dot{V} \leq -\gamma|s|$ که γ یک مقدار مثبت است، ثابت می‌شود که $\dot{V} \leq 0$ و داریم،

$$s\dot{s} \leq -\gamma|s| \quad \Rightarrow \quad \operatorname{sgn}(s)\dot{s} \leq -\gamma \quad (11-5)$$

از رابطه (9-5) و (11-5) داریم،

$$\operatorname{sgn}(s)[\hat{\alpha}(\dot{e} + ke)] \leq -\gamma \quad (12-5)$$

با استفاده از روابط (8-5) و (12-5)، ρ به شکل زیر به دست می‌آید،

$$\text{sgn}(s)[- \rho \text{sgn}(s) + d] \leq -\gamma$$

$$-\rho + d \text{sgn}(s) \leq -\gamma \quad (13-5)$$

$$\rho \geq \gamma + d \text{sgn}(s)$$

$$\rho = \gamma + |d|$$

$|d|$ از رابطه (5-6) به شکل زیر به دست می‌آید،

$$|d| = L|\dot{I}_a| + |R - \hat{R}| \cdot |I_a| + |\alpha - \hat{\alpha}| \cdot |\dot{q}| \quad (14-5)$$

با توجه به اینکه $L\dot{I}_a$ نویز فرکانس بالا تولید می‌کند و اندازه‌گیری آن پیچیده است و برای سادگی و کم کردن حجم محاسبات می‌توان $|d|$ را از رابطه (5-5) به شکل زیر به دست آورد،

$$d = v - \hat{R}I_a - \hat{\alpha}\dot{q} \quad (15-5)$$

$$|d| = v_{max} + \hat{R}|I_a| + \hat{\alpha}|\dot{q}|$$

از طرفی، هرچه اختلاف $|d|$ و ρ کمتر باشد، پایداری سیستم به پایداری مجانبی نزدیک‌تر می‌شود. برای این منظور و با توجه به تغییرات نرم سیگنال ولتاژ، برای کوچک‌تر کردن مقدار $|d|$ در رابطه (5-15) به جای بیشینه ولتاژ، از ولتاژ لحظه قبل استفاده می‌کنیم،

$$|d| = |v(t - \delta)| + \hat{R}|I_a| + \hat{\alpha}|\dot{q}| \quad , \quad \delta \rightarrow 0 \quad (16-5)$$

با جایگذاری رابطه (5-16) در (5-13)، در نهایت قانون کنترل به شکل زیر به دست می‌آید،

$$v = \hat{R}I_a + \hat{\alpha}(\dot{q}_d + k_p e) + [\gamma + |v(t - \delta)| + \hat{R}|I_a| + \hat{\alpha}|\dot{q}|] \text{sgn}(s) \quad , \quad \delta \rightarrow 0 \quad (17-5)$$

با توجه به $[\lambda]$ و با فرض اینکه q_d (مسیر مطلوب) باید دارای منحنی صاف، نرم و تا مرتبه موردنیاز مشتق‌پذیر و کران‌دار باشد، در رابطه ولتاژ موتور (4-15) اگر ولتاژ موتور $|v| < v_{max}$ باشد می‌توان نتیجه گرفت \dot{q} ، I محدود هستند. ρ مقدار محدودی دارد و $\dot{V} \leq 0$ است و در نتیجه $s \rightarrow 0$ و $e \rightarrow 0$ می‌رود و از آنجاکه $e = q_d - q$ و q_d معلوم و محدود است پس q نیز محدود است یعنی بردار حالت

$[q, \dot{q}, I_a]^T$ محدود است و این به این معنی است که متغیرهای حالت سیستم محدود هستند و پایداری ثابت می‌شود.

۵-۳- شبیه‌سازی روش کنترل غیرخطی مقاوم بر روی ربات

شبیه‌سازی بر روی ربات دو-رابط با موتور جریان مستقیم مغناطیس دائم انجام گرفته است. پارامترهای ربات دو-رابط مطابق جدول (۲-۲) است. در جدول (۱-۵) مشخصات موتور جریان مستقیم مغناطیس دائم مورد استفاده آمده است در این نوع موتورها $k_b = k_m$ است.

جدول ۱-۵ پارامترهای موتور جریان مستقیم مغناطیس دائم

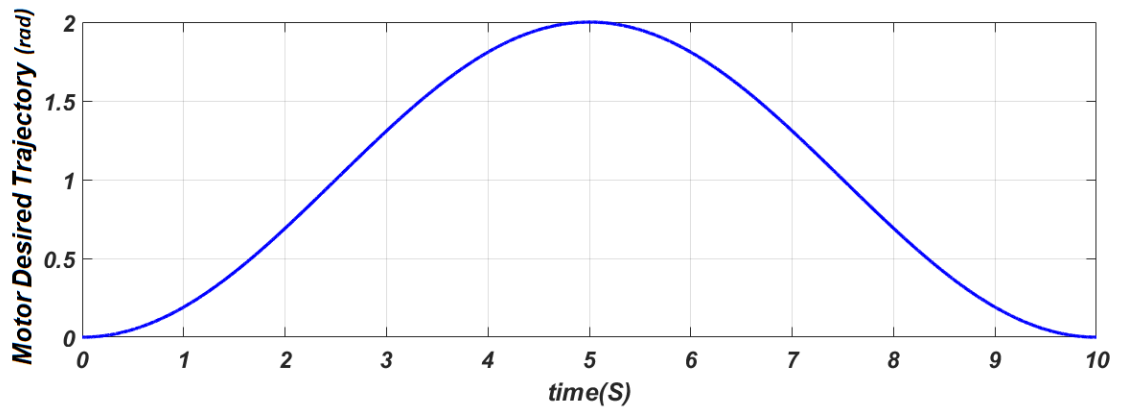
<i>Motor</i>	$R(\Omega)$	k_m <i>V.s/rad</i>	J_m <i>Nms²/rad</i>	B_m <i>Nms²/rad</i>	r	$L(H)$	$u_{max}(v)$
1, 2	1.26	0.26	0.0002	0.001	0.01	0.001	50

۵-۴- بررسی عملکرد کنترل کننده غیرخطی مقاوم در ردگیری مسیر مطلوب

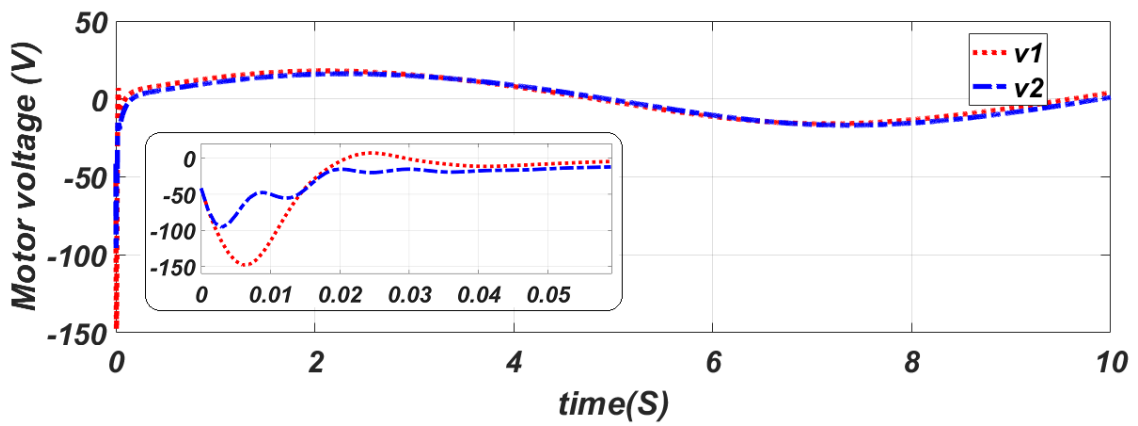
در این بخش عملکرد کنترل کننده پیشنهادی بر روی هر دو مفصل در ردگیری مسیر مطلوب آورده شده است. پارامترهای هر دو کنترل کننده مشابه است. مسیر مطلوب برای هر دو مفصل طبق شکل (۵-۱) با تابع زیر تعریف شده است :

$$q_d = 1 - \cos(\pi t/5) \quad (۱۸-۵)$$

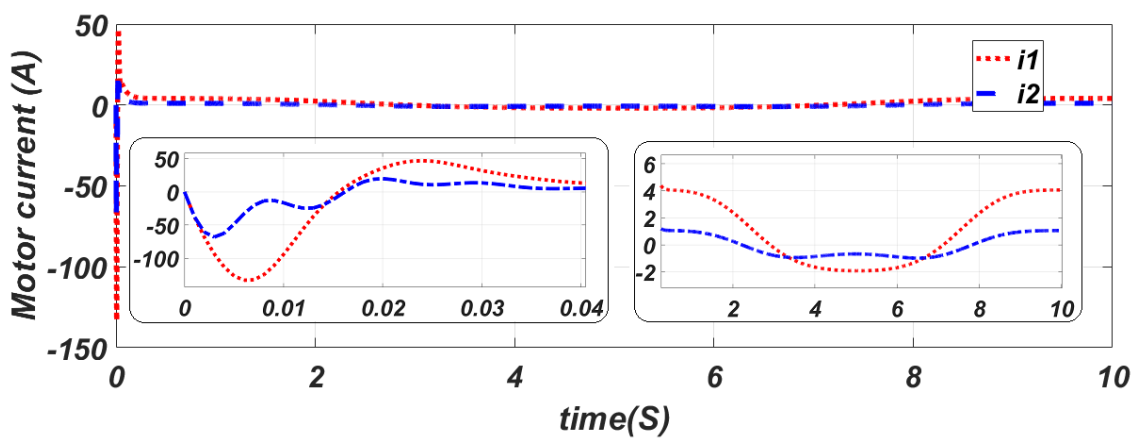
زمان اجرای شبیه‌سازی ۱۰ ثانیه انتخاب شده است. خطای اولیه برای هر دو مفصل ۰/۱ رادیان در نظر گرفته شده است. برای داشتن عدم قطعیت‌های پارامتری، \hat{R} و $\hat{\alpha}$ ، ۹۰ درصد مقدار واقعی‌شان در نظر گرفته شده‌اند. پارامترهای کنترل کننده عبارتند از $k_p = 20$ ، $\gamma = 10$. در شکل (۲-۵) ولتاژ ورودی موتور نشان داده می‌شود که محدود است. در شکل (۳-۵) جریان موتور نشان داده می‌شود. مشاهده می‌شود جریان موتور کران دار است.



شکل ۱-۵ مسیر مطلوب ردگیری

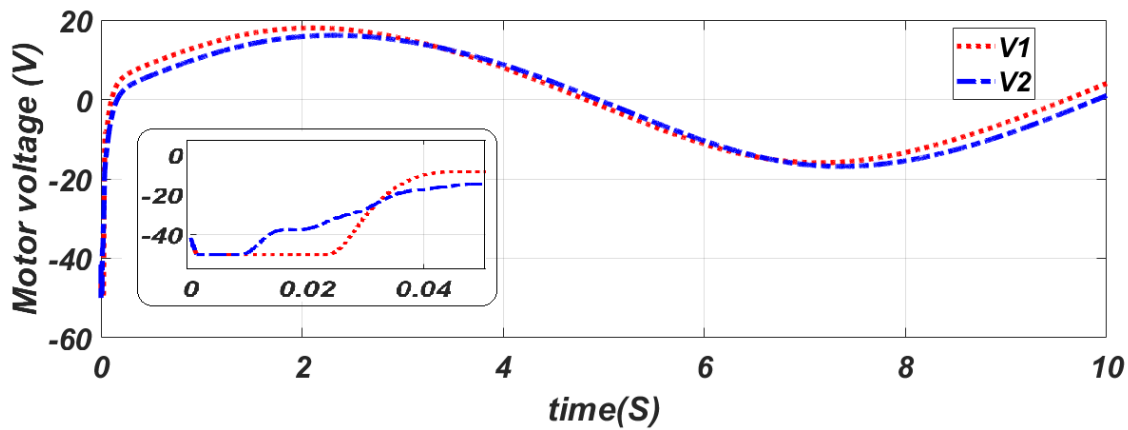


شکل ۲-۵ ولتاژ ورودی موتورها

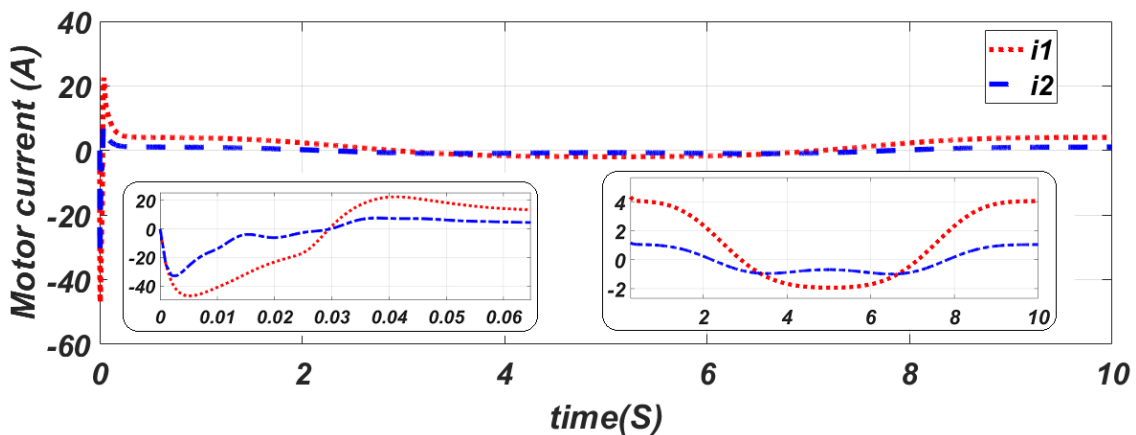


شکل ۳-۵ جریان موتورها

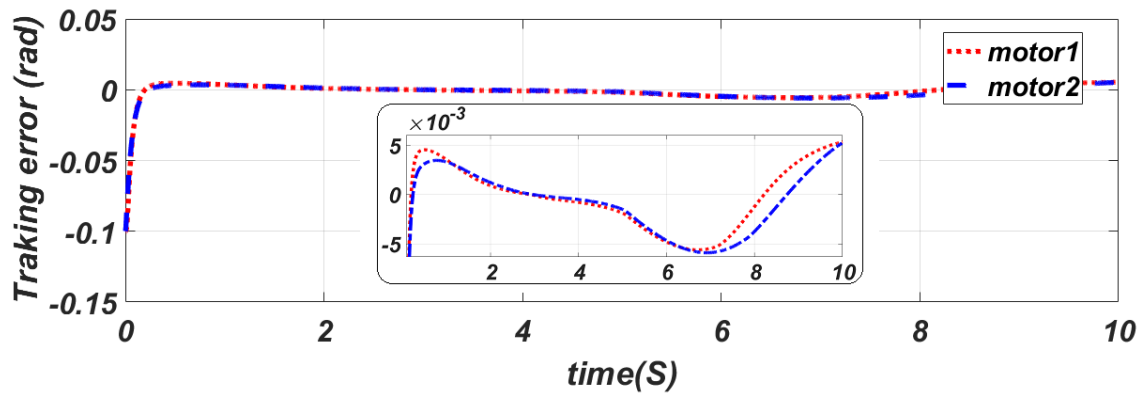
با توجه به مشخصات موتور الکتریکی که در جدول (۱-۵) آمده است، مقدار بیشینه ولتاژ موتور ۵۰ ولت می‌باشد. همان‌طور که شکل (۲-۵) و (۳-۵) نشان می‌دهد ولتاژ و جریان موتور در لحظه اول خارج از محدوده قابل تحمل موتور است. برای حفاظت الکتریکی موتور با بهره‌گیری از یک محدود کننده، ولتاژ ورودی به موتور را در بازه‌ی مجاز و قابل تحمل محدود می‌کنیم. شکل (۴-۵) و شکل (۵-۵) به ترتیب ولتاژ و جریان موتور بعد از اعمال محدود کننده به سیستم را نشان می‌دهند. در شکل (۶-۵) خطای ردگیری برای کنترل کننده پیشنهادی آمده است. خطا در انتهای ۱۰ ثانیه در محدوده 5.3×10^{-3} است. در شکل (۷-۵) نمودار تغییرات دینامیک‌های مدل نشده آورده شده است.



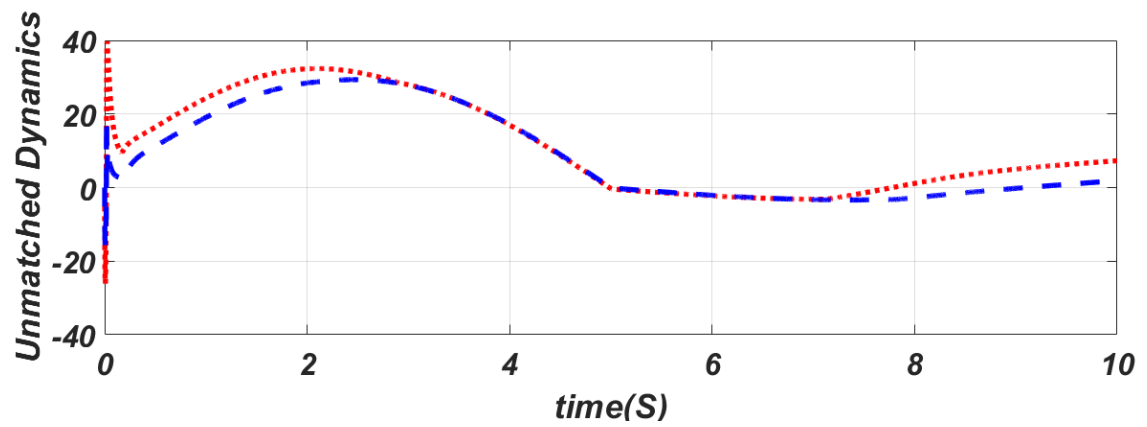
شکل ۴-۵ ولتاژ ورودی موتورها با اعمال محدود کننده



شکل ۵-۵ جریان ورودی موتورها با اعمال محدود کننده



شکل ۵-۶ خطای ردگیری کنترل کننده



شکل ۵-۷ نمودار تغییرات دینامیک‌های مدل نشده

فصل ششم

بهبودسازی کنترل کننده غیرخطی مقاوم

۶-۱- مقدمه

همان‌طور که در فصل قبل اشاره شد، عدم قطعیت پارامتری و دینامیک‌های مدل نشده در سیستم تغییرات و تأثیرات ثابتی به وجود می‌آورند لذا در تکرارپذیر بودن سیستم و حلقه کنترلی خللی به وجود نمی‌آورند. ولی هنگامی که سیستم در معرض اغتشاشات خارجی باشد، در واقع با توجه به اینکه این اغتشاشات متغیر با زمان بوده و اطلاعاتی از آن‌ها در دسترس نیست، تکرارپذیری حلقه کنترلی با مشکل مواجه می‌شود. در این صورت استفاده از الگوریتم‌های بهینه‌سازی هوشمند و یافتن جواب بهینه میسر نیست. الگوریتم‌های بهینه‌سازی خارج از حلقه کنترلی یا به اصطلاح "خارج از خط" عملیات بهینه‌سازی را انجام می‌دهند و از آنجایی که بهینه‌سازی وابسته به مسیر حرکت است، الگوریتم بهینه‌سازی باید برای هر مسیر اجرا شود. چنانچه مسیر حرکت تکراری باشد روش‌های بهینه‌سازی هوشمند قابل اجرا هستند در غیر این صورت نمی‌توان از الگوریتم‌های بهینه‌سازی استفاده کرد.

۶-۲- بهینه‌سازی سیستم کنترل غیر خطی مقاوم

ورودی سیستم در راهبرد کنترل ولتاژ، ولتاژ موتور است. معادله ولتاژ موتور به صورت زیر تعریف می‌شود،

$$RI_a + L\dot{I}_a + \alpha\dot{q} = v \quad , \quad \alpha = k_b r^{-1} \quad (۱-۶)$$

که صورت بازنویسی شده آن برابر است با،

$$\hat{R}I_a + \hat{\alpha}\dot{q} + d = v \quad (۲-۶)$$

و سیستم حلقه بسته به شکل زیر بدست آمد،

$$\hat{\alpha}(\dot{q}_a + k_p e) = -\rho \operatorname{sgn}(s) + L\dot{I}_a + (R - \hat{R})I_a + (\alpha - \hat{\alpha})\dot{q} \quad (۳-۶)$$

و در نهایت قانون کنترل طبق روابط (۴-۵) تا (۱۷-۵) به شکل زیر پیشنهاد می‌شود،

$$v = \hat{R}I_a + \hat{\alpha}(\dot{q}_a + k_p e) + \rho \operatorname{sgn}(s) \quad , \quad \rho = \gamma + |d| \quad (۳-۶)$$

$|d|$ اندازه دینامیک‌های مدل نشده سیستم است. شکل (۵-۵) نشان می‌دهد که مجموع این مقادیر در روش پیشنهادی محدوده مشخصی دارد و حلقه کنترلی به دلیل ماهیت راهبرد کنترل ولتاژ که در فصل قبل توضیح داده شد، مصون از اغتشاشات خارجی است. پس سیستم حلقه بسته به اصطلاح تکرارپذیر بوده و پارامترها در محدوده مشخصی ثابت هستند. حال با اثبات اینکه دینامیک‌های مدل نشده مقدار محدودی دارند، می‌توان گفت که اندازه دینامیک‌های مدل نشده از یک مقدار ثابتی کوچک‌تر است و می‌توان این مقدار را یک عدد ثابت در نظر گرفت. بدین وسیله حجم محاسبات برای محاسبه حدود دینامیک‌های مدل نشده بسیار کاهش می‌یابد و به سادگی می‌توان اندازه دینامیک‌های مدل نشده را یک پارامتر ثابت در نظر گرفت. همچنین γ در رابطه (۶-۳) مقدار مثبت ثابتی دارد. پس می‌توان ρ را یک مقدار مثبت ثابت در نظر گرفت. در این فصل با استفاده از الگوریتم‌های بهینه‌سازی هوشمند که در فصل ۴ به آن‌ها اشاره شد، بهینه‌ترین مقدار ثابت برای پارامترهای کنترل‌کننده شامل k_p و ρ را به دست می‌آوریم. بهینه‌سازی پارامترهای k_p و ρ از این جهت ارزشمند است که خطای ردگیری را کاهش و سرعت همگرایی را به شکل بهینه‌ای افزایش می‌دهد. در شبیه‌سازی‌های انجام‌شده، به جای تابع علامت در قانون کنترل از تابع اشباع استفاده شده است. زیرا تابع علامت باعث پدیده لرزش در سیگنال کنترل می‌گردد که از نظر فنی باعث فرسایش قطعات مکانیکی ربات شده و کنترل دقیق سیستم را با مشکل مواجه می‌کند. ضریب تابع اشباع (G) هرچه بزرگ‌تر باشد رفتار این تابع به تابع علامت نزدیک‌تر می‌شود و هرچه G کوچک‌تر باشد این تابع به خود تابع ورودی نزدیک‌تر می‌شود. کمترین مقدار برای گین تابع اشباع $G=1$ است که در این حالت تابع خروجی با تابع ورودی برابر است و فقط دامنه آن به اندازه دامنه تابع اشباع، محدود شده است. در اینجا دامنه تابع اشباع مانند تابع علامت در بازه $[-1, 1]$ محدود شده است. با توجه به اینکه معیار دقیقی برای تعیین G وجود ندارد، به عنوان یکی از پارامترهای بهینه‌سازی در نظر گرفته می‌شود.

برای بهینه‌سازی کنترل‌کننده پیشنهادی، تعداد جمعیت ۵ و تعداد تکرار ۱۰ در نظر گرفته شده

است. بردار طراحی برای انجام فرآیند بهینه‌سازی به صورت $X = \begin{Bmatrix} \rho \\ k_p \\ G \end{Bmatrix}$ در نظر گرفته می‌شود. فضای

طراحی برای هر مفصل یک فضای سه‌بعدی است و حدود پارامترها در بهینه‌سازی به صورت جدول (۱-۶) تنظیم شده است.

جدول ۱-۶ حدود پارامترها در بهینه‌سازی

مقدار	G	ρ	k_p
کمینه	۱	۰/۱	۱
بیشینه	۱۰	۵۰	۲۰۰

در پیاده‌سازی‌ها، پارامترهای هر دو کنترل‌کننده مشابه هستند. مسیر مطلوب برای هر دو مفصل طبق

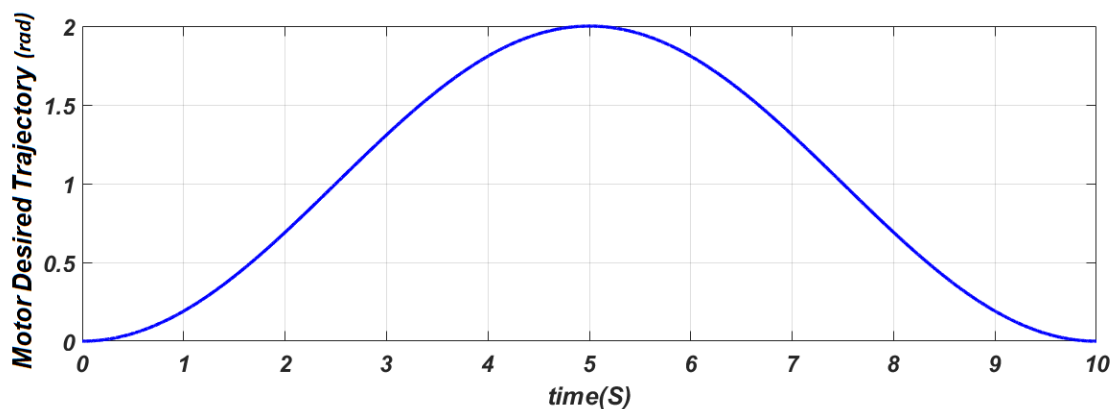
شکل (۱-۶) با تابع زیر تعریف شده است :

$$q_d = 1 - \cos(\pi t/5) \quad (۴-۶)$$

در تمام شبیه‌سازی‌ها، زمان اجرای شبیه‌سازی ۱۰ ثانیه انتخاب شده است. خطای اولیه برای هر دو

مفصل ۰/۱ رادیان در نظر گرفته شده است. برای داشتن عدم قطعیت‌های پارامتری، $\hat{\alpha}$ و \hat{R} ، ۹۰ درصد

مقدار واقعی‌شان در نظر گرفته شده‌اند.



شکل ۱-۶ مسیر مطلوب ردگیری در بهینه‌سازی

بهینه‌سازی با هدف کاهش دامنه سیگنال کنترل (کم شدن ولتاژ) و کمینه شدن خطای ردگیری انجام می‌شود. پس تابع هدف طبق رابطه (۴-۲)، به صورت،

$$f(x) = w_1(n_1 \cdot MSE) + w_2(n_2 \cdot MSV) \quad (۵-۶)$$

تعریف می‌شود که MSE مجذور مربعات خطای ردگیری^۱ و MSV مجذور مربعات ولتاژ ورودی^۲ می‌باشد. w_1 و w_2 وزن توابع هدف هستند که ارزش هر تابع در فرآیند بهینه‌سازی را مشخص می‌کنند به طوری که $w_1 + w_2 = 1$. یعنی هر تابعی که وزنش بیشتر باشد، فرآیند بهینه‌سازی تمرکز بیشتری روی آن تابع خواهد داشت. w_1 وزن تابع مربوط به خطای ردگیری و w_2 وزن تابع مربوط به سیگنال کنترل است. از آنجایی که تغییرات سیگنال کنترل در کنترل‌کننده پیشنهادی در بازه مجاز قرار دارد، فرآیند بهینه‌سازی روی کمینه کردن خطای ردگیری متمرکز می‌شود و وزن تابع هدف مربوط به خطای ردگیری بزرگ‌تر در نظر گرفته شده است. $w_1 = 0.8$ و $w_2 = 0.2$ در نظر گرفته شده است. با توجه به دامنه‌ی توابع هدف ابتدا با اعمال ضرایب n_1 و n_2 توابع نرمال‌سازی شده است. بدین ترتیب برای یکسان‌سازی ارزش توابع هدف، $n_1 = 1000$ و $n_2 = 0.001$ در نظر گرفته می‌شوند. شاخص عملکرد الگوریتم‌ها برای توابع $f_1(x)$ و $f_2(x)$ در رابطه (۴-۲) به ترتیب، تابع میانگین مجذور سیگنال کنترل و تابع میانگین مجذور خطای ردگیری در نظر گرفته شده است. معیار توقف الگوریتم‌ها، بر روی حداکثر تعداد تکرار تنظیم شده است. مشخصه‌های مورد مطالعه در فرآیند بهینه‌سازی الگوریتم‌ها، مدت زمان اجرای الگوریتم (t_A)، مجذور مربعات خطای ردگیری (MSE)، مجذور مربعات ولتاژ ورودی (MSV) و مقدار تابع هدف چندگانه^۳ ($MOFV$) می‌باشد.

۶-۲-۱- بهینه‌سازی با استفاده از الگوریتم بهینه‌سازی ازدحام ذرات

پارامترهای داخلی الگوریتم بهینه‌سازی ازدحام ذرات به صورت جدول (۶-۲) تنظیم شده است. با توجه

¹ Mean Square of Error

² Mean Square of Voltage

³ Multi Objective Function Value

به رابطه (۳-۵) در قانون کنترل پیشنهادی، پارامترهای ρ و k_p و G با استفاده از این الگوریتم به مقادیر موجود در جدول (۳-۶) همگرا شدند:

جدول ۲-۶ پارامترهای داخلی الگوریتم بهینه‌سازی ازدحام ذرات

W	W_{damp}	c_1	c_2
۰/۹	۰/۹۹	۲/۰۰	۲/۰۰

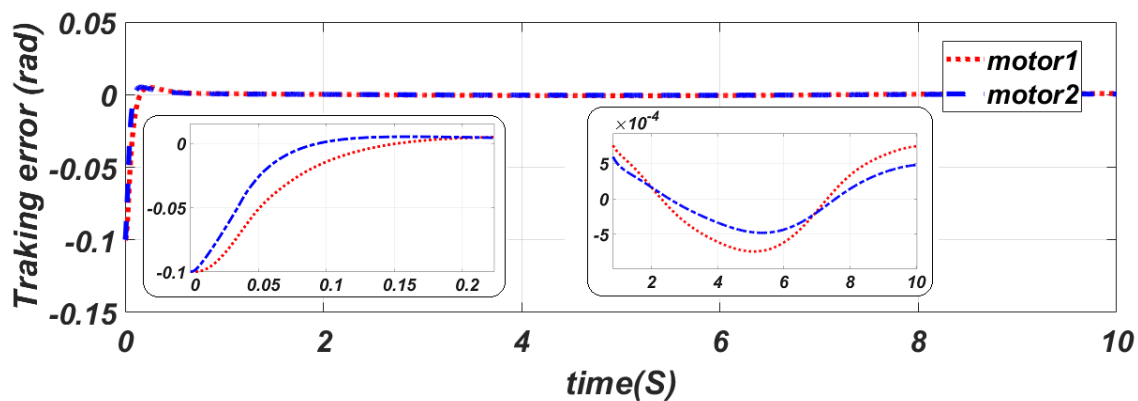
همان‌طور که در فصل ۴ به تفصیل بیان شد، وزن اینرسی یا ضریب همگرایی الگوریتم، W_{damp} نرخ کاهش وزن اینرسی، c_1 ضریب وزنی برای بهترین تجربه شخصی هر ذره و c_2 ضریب وزنی برای بهترین تجربه گروهی ذرات می‌باشد.

جدول ۳-۶ پارامترهای کنترل‌کننده بهینه و مشخصه‌های بهینه‌سازی الگوریتم ازدحام ذرات

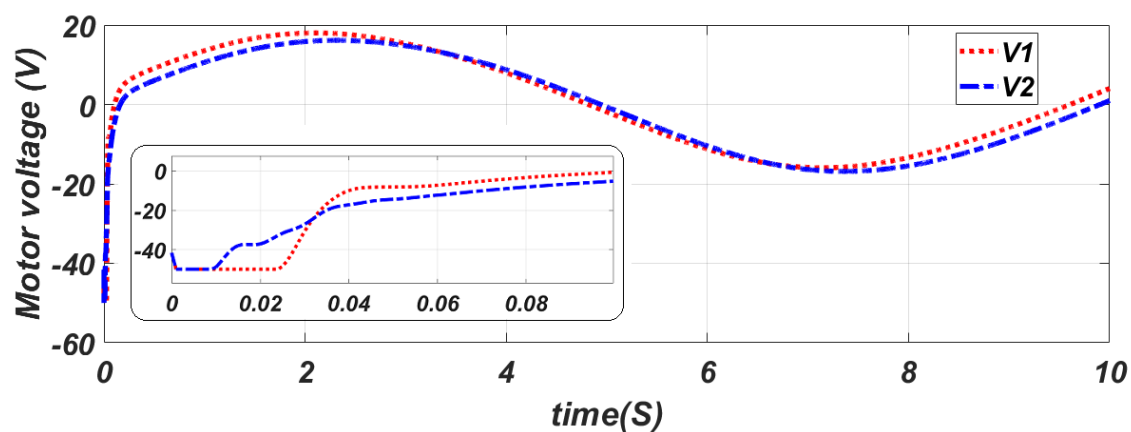
joint	k_p	ρ	G	MSV	MSE	MOFV	$t_A(s)$
1	199	13	8.45	171.34	3.76e-5	0.033	240.28
2	21	21.35	9.4	143.23	3.13e-5	0.017	

۲-۶-۱-۱-۱- بررسی عملکرد الگوریتم ازدحام ذرات در بهینه‌سازی کنترل‌کننده

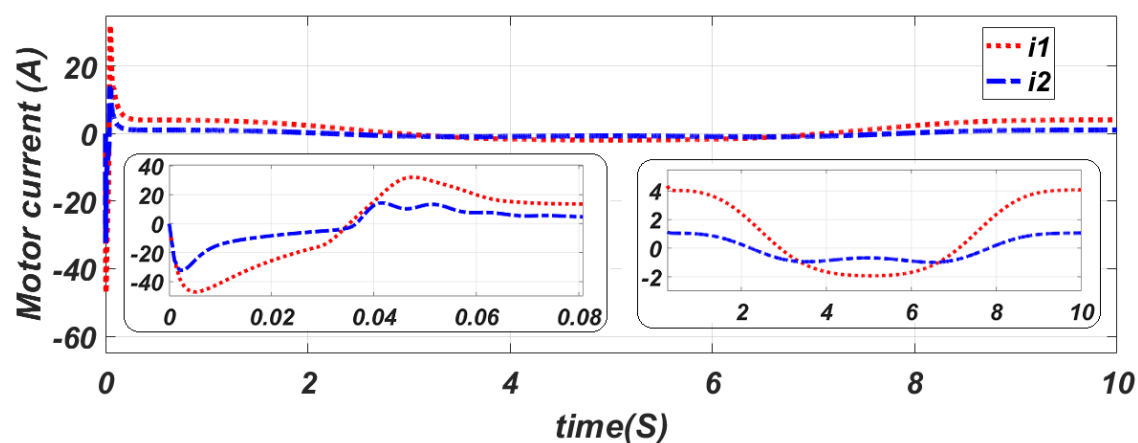
در این بخش عملکرد الگوریتم ازدحام ذرات در بهینه‌سازی کنترل‌کننده پیشنهادی بر روی هر دو مفصل در ردگیری مسیر مطلوب آورده شده است. در شکل (۲-۶) خطای ردگیری برای کنترل‌کننده پیشنهادی با بهینه‌سازی الگوریتم ازدحام ذرات آمده است. همان‌طور که در جدول ۳-۶ مشاهده می‌شود میانگین مجذور خطای ردگیری و میانگین مجذور ولتاژ ورودی مقادیر قابل قبولی دارند و خطا در انتهای ۱۰ ثانیه در محدوده 9×10^{-5} است. فرآیند بهینه‌سازی در مدت زمان تقریبی ۴ دقیقه انجام می‌شود. در شکل (۳-۶) ولتاژ ورودی موتور نشان داده می‌شود. در شکل (۴-۶) جریان موتور نشان داده می‌شود. در شکل (۵-۶) تنظیم پارامترها در روند الگوریتم ازدحام ذرات نشان داده می‌شود.



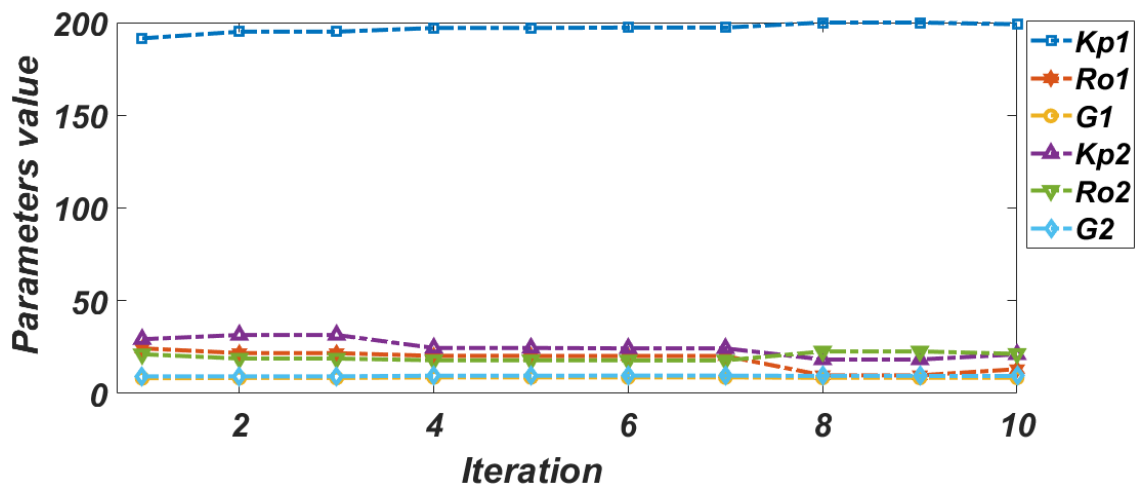
شکل ۶-۲ خطای ردگیری کنترل کننده بهینه با استفاده از الگوریتم بهینه‌سازی ازدحام ذرات



شکل ۶-۳ ولتاژ ورودی موتورها در حالت بهینه با استفاده از الگوریتم بهینه‌سازی ازدحام ذرات



شکل ۶-۴ جریان موتورها در حالت بهینه با استفاده از الگوریتم بهینه‌سازی ازدحام ذرات



شکل ۵-۶ نمودار تنظیم پارامترهای بهینه کنترل کننده با استفاده از الگوریتم بهینه‌سازی ازدحام ذرات

۶-۲-۲- بهینه‌سازی با استفاده از الگوریتم کرم شب‌تاب

پارامترهای داخلی الگوریتم بهینه‌سازی کرم شب‌تاب به صورت جدول (۶-۴) تنظیم شده است. با

توجه به رابطه (۵-۳) در قانون کنترل پیشنهادی، پارامترهای ρ و k_p و G با استفاده از این الگوریتم به

مقادیر موجود در جدول (۶-۵) همگرا شدند:

جدول ۶-۴ پارامترهای داخلی الگوریتم بهینه‌سازی کرم شب‌تاب

β	γ	α	α_{damp}	m
۲	۱	۰/۲	۰/۹۹	۲

طبق توضیحاتی که در فصل ۴ داده شد، β قابلیت جذب، γ ضریب جذب نور برای هر کرم شب‌تاب، α

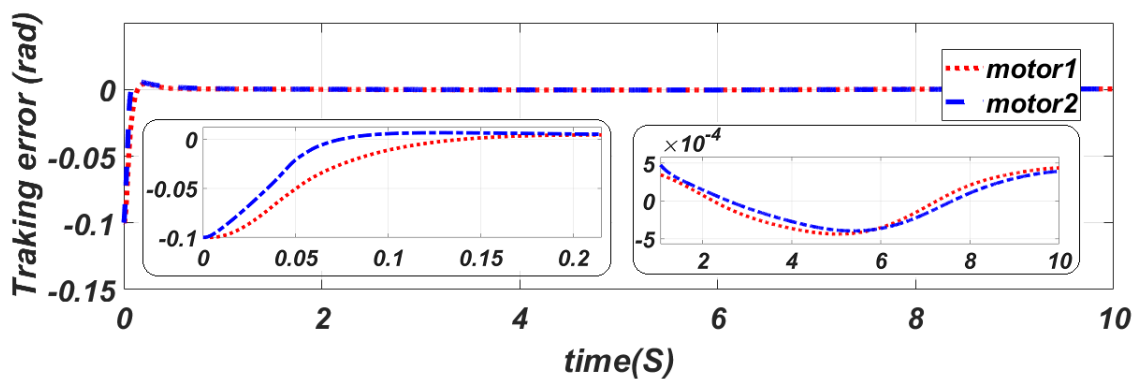
پارامتر تصادفی سازی حرکت کرم‌ها و m پارامتر تعیین کننده نرخ قابلیت جذب می‌باشد.

جدول ۶-۵ پارامترهای کنترل کننده بهینه و مشخصه‌های بهینه‌سازی الگوریتم کرم شب‌تاب

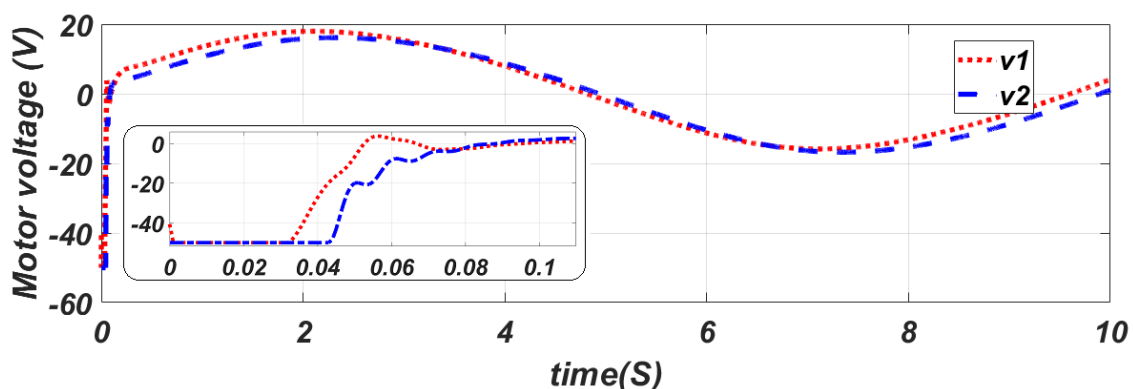
joint	k_p	ρ	G	MSV	MSE	MOFV	t_A (s)
1	137.34	12.76	6.34	165.81	3.89e-5	0.034	678.47
2	54.5	41.65	3.98	148.66	2.66e-5	0.017	

۶-۲-۲-۱- بررسی عملکرد الگوریتم کرم شب تاب در بهینه سازی کنترل کننده

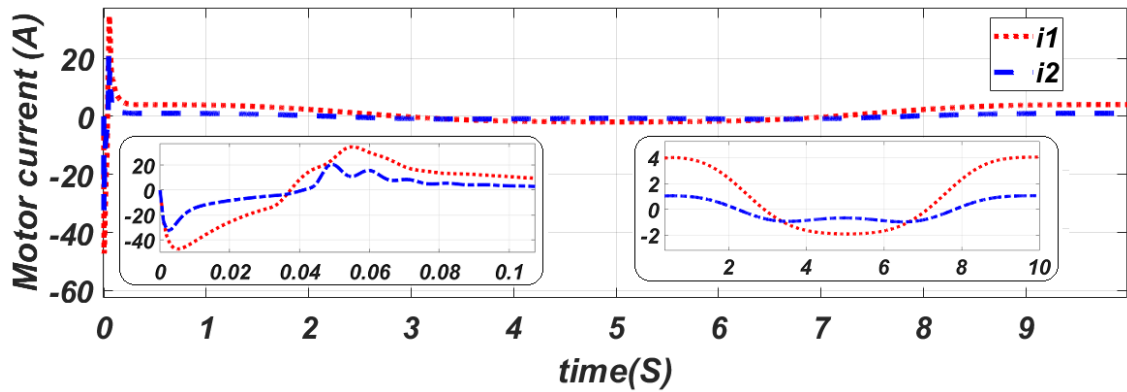
در این بخش عملکرد الگوریتم کرم شب تاب در بهینه سازی کنترل کننده پیشنهادی بر روی هر دو مفصل در ردگیری مسیر مطلوب آورده شده است. در شکل (۶-۶) خطای ردگیری برای کنترل کننده پیشنهادی با بهینه ساز الگوریتم کرم شب تاب آمده است. همانطور که در جدول ۶-۵ مشاهده می شود میانگین مجذور خطای ردگیری و میانگین مجذور ولتاژ ورودی مقادیر قابل قبولی دارند و خطا در انتهای ۱۰ ثانیه در محدوده 2×10^{-4} است. فرآیند بهینه سازی در مدت زمان تقریبی ۱۱ دقیقه انجام می شود. در شکل (۶-۷) ولتاژ ورودی موتور نشان داده می شود. در شکل (۶-۸) جریان موتور نشان داده می شود. در شکل (۶-۹) تنظیم پارامترها در روند الگوریتم ازدحام ذرات نشان داده می شود.



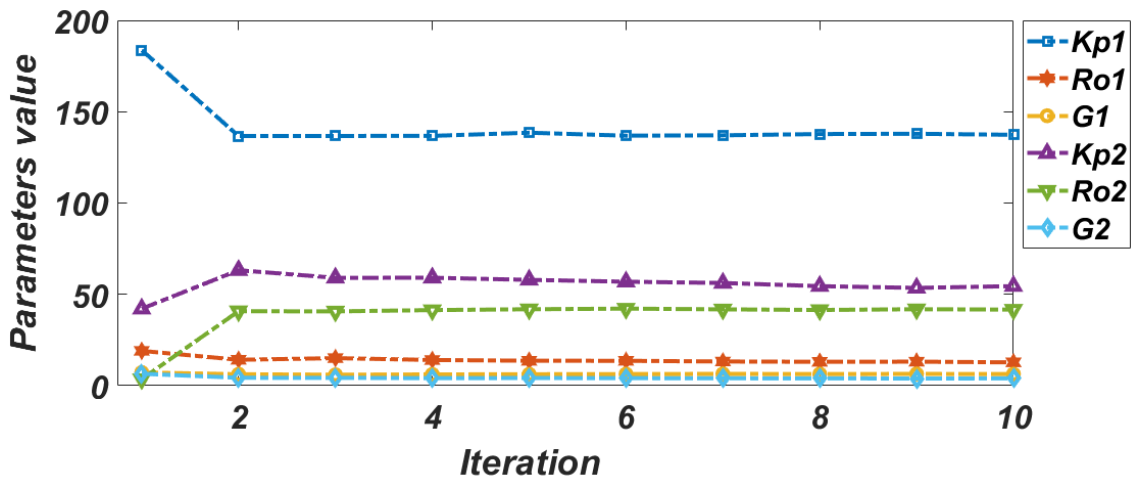
شکل ۶-۶ خطای ردگیری کنترل کننده بهینه با استفاده از الگوریتم بهینه سازی کرم شب تاب



شکل ۶-۷ ولتاژ ورودی موتورها در حالت بهینه با استفاده از الگوریتم بهینه سازی کرم شب تاب



شکل ۸-۶ جریان موتورها در حالت بهینه با استفاده از الگوریتم بهینه‌سازی کرم شب‌تاب



شکل ۹-۶ نمودار تنظیم پارامترهای بهینه کنترل‌کننده با استفاده از الگوریتم بهینه‌سازی کرم شب‌تاب

۶-۲-۳- بهینه‌سازی با استفاده از الگوریتم بهینه‌سازی خفاش

پارامترهای داخلی الگوریتم خفاش به صورت جدول (۶-۶) تنظیم شده است. با توجه به رابطه

(۶-۳) در قانون کنترل پیشنهادی، پارامترهای ρ و k_p و G با استفاده از الگوریتم خفاش به مقادیر موجود

در جدول (۶-۷) همگرا شدند:

جدول ۶-۶ پارامترهای داخلی الگوریتم بهینه‌سازی خفاش

A	r	α	γ	f_{min}	f_{max}
۰/۶	۰/۴	۰/۵	۰/۹	۰/۰۱	۰/۱

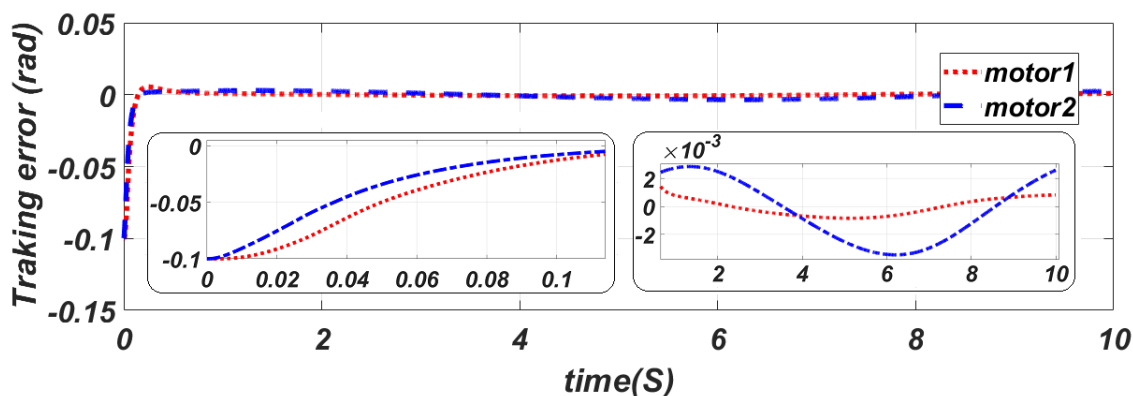
همان‌طور که در فصل ۴ اشاره شد، پارامتر A بلندی صدا، پارامتر r نرخ پالس، پارامتر α نرخ کاهش بلندی صدا، پارامتر γ نرخ کاهش نرخ پالس، f_{min} مقدار کمینه فرکانس و f_{max} مقدار بیشینه فرکانس می‌باشد.

جدول ۶-۷ پارامترهای کنترل‌کننده بهینه و مشخصه‌های بهینه‌سازی الگوریتم خفاش

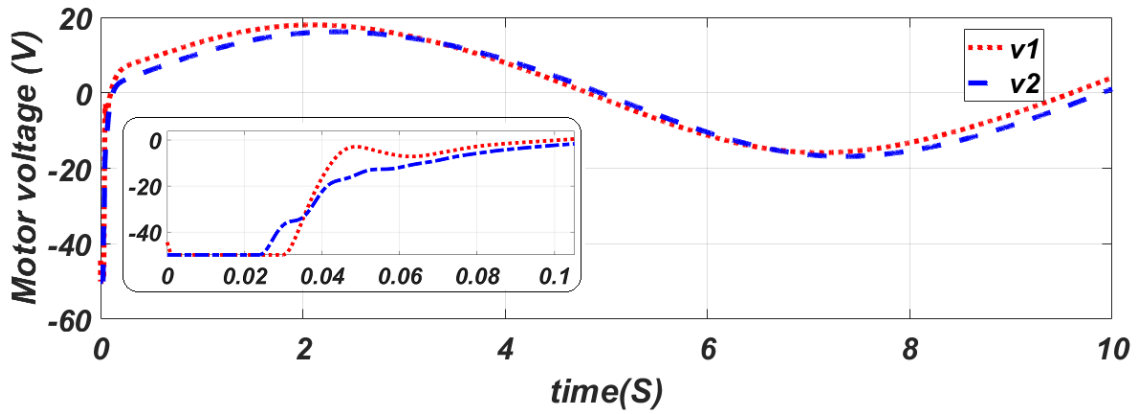
joint	k_p	ρ	G	MSV	MSE	MOFV	$t_A(s)$
1	71.89	26.54	3.3	160.23	3.95e-5	0.035	242.7
2	181.11	9	1	150	2.55e-5	0.017	

۶-۲-۳-۱- بررسی عملکرد الگوریتم خفاش در بهینه‌سازی کنترل‌کننده

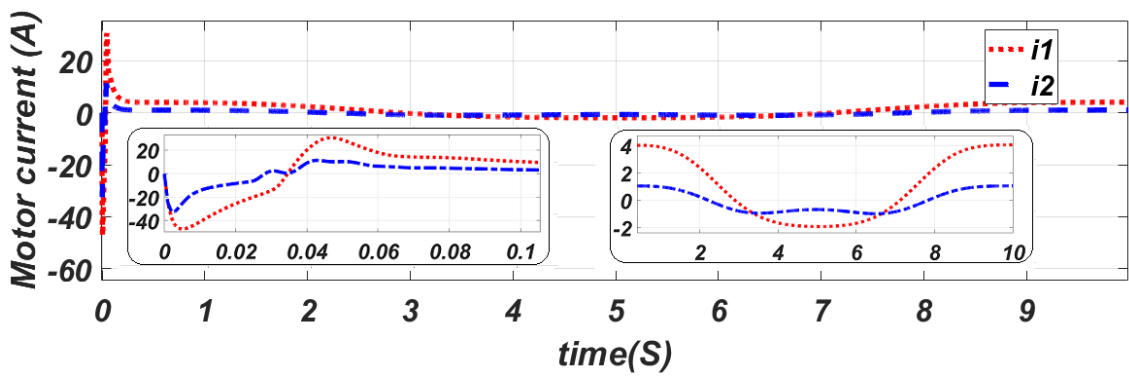
در این بخش عملکرد الگوریتم خفاش در بهینه‌سازی کنترل‌کننده پیشنهادی بر روی هر دو مفصل در ردگیری مسیر مطلوب آورده شده است. در شکل (۶-۱۰) خطای ردگیری برای کنترل‌کننده پیشنهادی با بهینه‌ساز الگوریتم خفاش آمده است. همان‌طور که در جدول (۶-۷) مشاهده می‌شود میانگین مجذور خطای ردگیری و میانگین مجذور ولتاژ ورودی مقادیر قابل قبولی دارند و خطا در انتهای ۱۰ ثانیه در محدوده 3×10^{-4} است. فرآیند بهینه‌سازی در مدت زمان تقریبی ۴ دقیقه انجام می‌شود. در شکل (۶-۱۱) ولتاژ ورودی موتور نشان داده می‌شود. در شکل (۶-۱۲) جریان موتور نشان داده می‌شود. در شکل (۶-۱۳) تنظیم پارامترها در روند الگوریتم ازدحام ذرات نشان داده می‌شود.



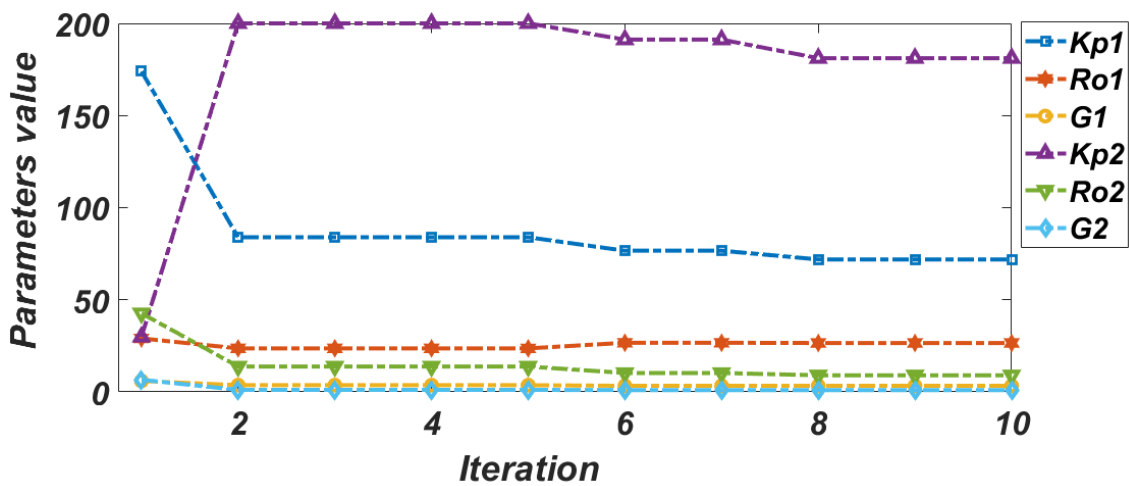
شکل ۶-۱۰ خطای ردگیری کنترل‌کننده بهینه با استفاده از الگوریتم خفاش



شکل ۶-۱۱ ولتاژ ورودی موتورها در حالت بهینه با استفاده از الگوریتم خفاش



شکل ۶-۱۲ جریان موتورها در حالت بهینه با استفاده از الگوریتم خفاش



شکل ۶-۱۳ نمودار تنظیم پارامترهای بهینه کنترل کننده با استفاده از الگوریتم خفاش

۳-۶- مقایسه عملکرد الگوریتم‌های تکاملی هوشمند در بهینه‌سازی سیستم کنترل

در این بخش ابتدا هر الگوریتم برای حالت‌های مختلفی از ضرایب وزنی ($w_1 + w_2 = 1$) اجرا می‌شود. در هر بار اجرای الگوریتم‌های بهینه‌سازی کنترل‌کننده غیرخطی مقاوم پیشنهادی برای بازوی اول ربات، مدت زمان اجرای الگوریتم (t_A)، مجذور مربعات خطای ردگیری (MSE)، مجذور مربعات ولتاژ ورودی (MSV)، مجذور مربعات جریان^۱ (MSC)، مقدار تابع هدف چندگانه ($MOFV$)، مجموع مطلق خطای ردگیری^۲ (SAE) و خطا در لحظه پایان الگوریتم (E_{end}) اندازه‌گیری شده و به همراه مقادیر پارامترهای بهینه‌سازی شده کنترل‌کننده (k_p, ρ, G) در جدول (۶-۸)، (۶-۹) و (۶-۱۰) که به ترتیب مربوط به الگوریتم‌های ازدحام ذرات، کرم شبتاب و خفاش می‌باشد، نمایش داده می‌شود.

جدول ۶-۸ عملکرد الگوریتم بهینه‌سازی ازدحام ذرات برای حالت‌های مختلفی از ضرایب وزنی تابع هدف چندگانه

Run (PSO)	$MOFV = w_1 \cdot (n_1 \cdot MSE) + w_2 \cdot (n_2 \cdot MSV)$					MSC	SAE	E_{end}	K_{p1}	ρ_1	G1	t_A (s)
	w_1	w_2	MOFV	MSE	MSV							
1	1	0	3.76e-2	3.77e-5	174.33	32.62	10.16	2.07e-4	200	0.1	1	112.6
2	0.9	0.1	4.98e-2	3.83e-5	153.7	13.96	8.47	3.12e-4	30.14	45.59	4.76	112.5
3	0.8	0.2	6.18e-2	3.9e-5	152.97	14.35	10.63	5.8e-4	28.94	21.5	5.63	113.09
4	0.7	0.3	7.33e-2	3.91e-5	153.31	14.66	11.21	6.53e-4	29.68	12.59	8.3	129.45
5	0.6	0.4	8.48e-2	4.02e-5	151.87	14.03	13.35	9.5e-4	25.79	43.89	1.87	115.8
6	0.5	0.5	9.81e-2	4.1e-5	155.21	16.11	21.57	2.06e-3	43.36	2.97	4.76	113.2
7	0.4	0.6	0.106	4.22e-5	149.33	12.61	10.73	5.93e-4	17.86	27.62	6.97	114.91
8	0.3	0.7	0.117	4.39e-5	148.88	12.55	13.20	8.87e-4	17.35	30.28	4.36	119.63
9	0.2	0.8	0.128	4.97e-5	148.75	12.43	25.5	2.62e-3	17.17	42.46	1	114.23
10	0.1	0.9	0.136	6.59e-5	144.59	11.1	15.65	6.78e-4	8.28	40.45	8.98	114.71
11	0	1	0.137	4.69e-4	137.04	8.35	108.74	3.17e-3	1	50	10	130.47

¹ Mean Square of Current

² Sum Absolute of Error

جدول ۶-۹ عملکرد الگوریتم بهینه‌سازی کرم شب‌تاب برای حالت‌های مختلفی از ضرایب وزنی تابع هدف چندگانه

Run (FA)	MOFV = $w_1 \cdot (n_1 \cdot \text{MSE}) + w_2 \cdot (n_2 \cdot \text{MSV})$					MSC	SAE	E _{end}	K _{p1}	ρ_1	G1	t _A (s)
	w ₁	w ₂	MOFV	MSE	MSV							
1	1	0	3.85e-2	3.85e-5	173.79	31.30	9.87	1.6e-4	196.6	9.46	6.67	370.7
2	0.9	0.1	5.01e-2	3.88e-5	155.56	16.59	11.98	7.17e-4	39.08	19.01	3.75	377.23
3	0.8	0.2	6.15e-2	3.87e-5	152.86	14.25	9.07	4.04e-4	26.26	36.17	5.32	384.37
4	0.7	0.3	7.5e-2	3.9e-5	159.13	18.05	9.95	2.1e-4	72.97	12.08	7.66	325.24
5	0.6	0.4	8.42e-2	3.96e-5	151.21	13.08	10.3	5.59e-4	23.34	30.94	5.05	346.23
6	0.5	0.5	0.102	3.88e-5	164.81	21.92	10.59	3.3e-4	124	22.20	2.14	338.5
7	0.4	0.6	0.106	4.05e-5	149.84	13.7	8.84	3.9e-4	18.39	32.18	8.87	352.7
8	0.3	0.7	0.117	4.48e-5	148.18	11.82	14.42	1.03e-3	16.9	11.63	9.94	349.14
9	0.2	0.8	0.129	4.99e-5	149.84	12.65	30.89	3.5e-3	20.78	13.97	1.61	333.6
10	0.1	0.9	0.136	6.02e-5	144.67	9.77	23.36	1.86e-3	10.42	11.92	8.69	338.23
11	0	1	0.137	4.55e-4	137.32	7.98	115.68	5.71e-3	1	37.04	7.44	343.45

جدول ۶-۱۰ عملکرد الگوریتم بهینه‌سازی خفاش برای حالت‌های مختلفی از ضرایب وزنی تابع هدف چندگانه

Run (BA)	MOFV = $w_1 \cdot (n_1 \cdot \text{MSE}) + w_2 \cdot (n_2 \cdot \text{MSV})$					MSC	SAE	E _{end}	K _{p1}	ρ_1	G1	t _A (s)
	w ₁	w ₂	MOFV	MSE	MSV							
1	1	0	3.8e-2	3.8e-5	165.76	24.63	12.9	6.01e-4	134.9	2.93	1.48	112.35
2	0.9	0.1	5.1e-2	3.92e-5	157.28	17.57	17.74	1.42e-3	60.87	2.7	5.92	112.15
3	0.8	0.2	6.15e-2	3.84e-5	153.64	13.94	12.73	8.38e-4	38.92	21.15	2.87	110.8
4	0.7	0.3	7.34e-2	3.9e-5	153.75	14.21	15.34	1.22e-3	40.61	14.81	2.57	113.51
5	0.6	0.4	8.41e-2	3.93e-5	151.31	13.23	9.78	4.96e-4	23.64	33.24	5.23	112.75
6	0.5	0.5	9.62e-2	4.14e-5	151.03	12.77	19.32	1.83e-3	27.73	14.40	2.57	112.65
7	0.4	0.6	0.106	4.34e-5	148.08	11.91	10.02	4.81e-4	15.55	27.78	9.83	112.15
8	0.3	0.7	0.117	4.48e-5	148.4	12.1	13.3	8.89e-4	16.47	19.03	7.3	113.85
9	0.2	0.8	0.128	4.75e-5	149.15	12.54	23.74	2.39e-3	18.58	11.2	3.84	113.49
10	0.1	0.9	0.136	5.62e-5	145.54	11.02	14.94	8.74e-4	10.58	34.48	6.39	114.36
11	0	1	0.137	4.68e-4	137.3	8.69	109.2	3.34e-3	1	50	9.5	112.9

همان‌طور که از داده‌های جداول بالا استنباط می‌شود، برای رسیدن به اهداف کنترلی موردنظر می‌توان ضرایب وزنی w_1 و w_2 را تغییر داد. طبق تابع هدف چندگانه‌ای که در رابطه (۴-۲) آمده است داریم :

$$MOFV = w_1 \cdot (n_1 \cdot MSE) + w_2 \cdot (n_2 \cdot MSV) \quad (5-7)$$

اگر هدف از کنترل ، فقط حداقل کردن خطای ردگیری باشد بهترین حالت برای ضرایب وزنی تابع هدف چندگانه $w_1 = 1, w_2 = 0$ می‌باشد. در این حالت خطای ردگیری به کمترین مقدار خود می‌رسد ولی ولتاژ و جریان در طی سیکل کنترل مقدار زیادی خواهند داشت. اگر هدف از کنترل، فقط حداقل کردن ولتاژ و جریان ورودی موتورها باشد بهترین بازدهی برای حالتی است که $w_1 = 0, w_2 = 1$ باشد. در این حالت بهینه‌سازی فقط روی کمینه کردن ولتاژ ورودی موتورها متمرکز می‌شود. برای دستیابی به حالت بهینه چندگانه که هم خطای ردگیری و هم ولتاژ ورودی موتورها کمینه گردد هر دو ضریب w_1 و w_2 باید مقدار داشته باشند به طوری که $w_1 + w_2 = 1$. از این رو تنظیم ضرایب وزنی با توجه به اهداف کنترلی توسط طراح انجام می‌شود.

بهترین، بدترین و میانگین مقادیر مشخصه هر سه الگوریتم محاسبه شده و در جدول (۶-۱۱) آورده شده است. با مقایسه این مقادیر مشاهده می‌شود که در فرآیند بهینه‌سازی مشخصه‌ها، الگوریتم‌های PSO ، FA و BA عملکرد مشابهی دارند. با ثابت نگه‌داشتن پارامترهای داخلی الگوریتم‌ها مشاهده می‌شود که در بهینه‌سازی مشخصه‌های مربوط به خطای ردگیری الگوریتم PSO عملکرد بهتری داشته است. خطای ردگیری مقدار کوچکی دارد از این رو می‌توان نتیجه گرفت که PSO در بهینه‌سازی مقادیر با مقیاس کوچکتر توانایی بیشتری دارد. در بهینه‌سازی مشخصه‌های مربوط به ولتاژ ورودی الگوریتم BA عملکرد مناسب‌تری داشته است. با توجه به اینکه ولتاژ مقادیر خیلی بزرگتری نسبت به خطای ردگیری دارد می‌توان نتیجه گرفت که BA در بهینه‌سازی مقادیر با مقیاس بزرگتر توانایی بیشتری دارد. از نظر سرعت در تعداد اعضا و تعداد تکرار برابر، الگوریتم‌های PSO و BA وضعیت مناسب‌تری دارند و

الگوریتم FA علی‌رغم عملکرد مشابه، مدت زمان اجرای طولانی‌ای دارد (در حدود سه برابر دو الگوریتم دیگر). از نظر سرعت همگرایی پارامترها، الگوریتم FA سریعتر همگرا می‌شود. بدین ترتیب با توجه به مقایسه انجام شده و اهداف کنترلی مورد نظر، انتخاب الگوریتم مناسب، تنظیم پارامترهای داخلی آن و تنظیم ضرایب وزنی توابع هدف توسط طراح انجام می‌شود.

جدول ۶-۱۱ مقایسه عملکرد الگوریتم‌های PSO، FA و BA در بهینه‌سازی مشخصه‌ها

Algorithms		MOFV	MSE	MSV	MSC	SAE	E _{end}	t _A (s)
PSO	Best	3.76e-2	3.77e-5	137.04	8.35	8.47	2.07e-4	112.5
	Worst	0.137	4.69e-4	174.33	32.62	108.74	3.17e-3	130.47
	Mean	9.35e-2	4.01e-5	151.88	14.79	22.65	1.16e-3	117.32
FA	Best	3.85e-2	3.85e-5	137.32	7.98	8.84	1.6e-4	325.24
	Worst	0.137	4.55e-4	173.79	31.30	115.68	5.71e-3	384.37
	Mean	9.42e-2	8.03e-5	153.38	15.55	23.17	1.35e-3	350.85
BA	Best	3.8e-2	3.8e-5	137.3	8.69	9.78	4.81e-4	112.15
	Worst	0.137	4.68e-4	165.76	24.63	109.2	3.34e-3	114.36
	Mean	9.35e-2	8.14e-5	151.02	13.87	23.54	1.31e-3	112.81

فصل هفتم

نتیجه‌گیری

۷-۱- نتیجه گیری

در این پایان نامه با بهره گیری از مزایای راهبرد کنترل ولتاژ و با هدف بهینه سازی پارامترها، یک کنترل کننده تکرارپذیر طراحی شد که تا حد خیلی زیادی مصون از اغتشاشات خارجی است. عملکرد کنترل کننده با استفاده از الگوریتم های بهینه سازی ازدحام ذرات، کرم شبتاب و خفاش بهینه شد که علاوه بر تضمین پایداری، عملکرد بسیار مناسبی در کمینه کردن خطای ردگیری و دامنه سیگنال کنترل دارد. سیگنال کنترل همان ولتاژ ورودی موتورها است که کمینه کردن این مقدار به عنوان قسمتی از پروسه بهینه سازی مقرون به صرفه است. نتایج شبیه سازی بیانگر توانایی سیستم کنترل پیشنهادی در ردگیری مسیرهای مطلوب و مقاوم بودن آن در برابر انواع عدم قطعیت ها است.

همچنین، به منظور ارزیابی تأثیر بهینه سازی در عملکرد کنترل کننده، مقایسه ای بین کنترل کننده مقاوم طراحی شده و کنترل کننده های بهینه سازی شده با سه روش ذکر شده انجام شد. نتایج شبیه سازی، بیانگر برتری کنترل کننده بهینه با استفاده از الگوریتم های بهینه سازی هوشمند است. همچنین طبق مقایسات انجام شده مشاهده شد که عملکرد الگوریتم های مورد استفاده، از نظر دقت تقریباً یکسان بوده و از نظر سرعت در اجرا الگوریتم های PSO و BA وضعیت مناسب تری دارند و سرعت الگوریتم FA در همگرایی پارامترها بیشتر است.

۷-۲- پیشنهادها

برای کارهای بعدی می توان با توسعه راهبرد کنترل ولتاژ، از سایر الگوریتم های کنترلی برای طراحی سیستم کنترل تکرارپذیر استفاده کرد. همچنین می توان از سایر الگوریتم های بهینه سازی که به تازگی مطرح شده اند استفاده کرد. پیشنهاد دیگر اینکه با ترکیب الگوریتم های بهینه سازی هوشمند می توان از مزایای آن ها به طور هم زمان بهره گرفت. همچنین استفاده از روش های بهینه سازی به صورت برخط و یا ترکیبی از روش های برخط و خارج خط به صورتی که پایداری سیستم تضمین شود، می تواند روش مناسبی برای مقابله با عدم قطعیت های احتمالی باشد.

- [1] Albu-Schäffer, A., Ott, C., & Hirzinger, G. (2007), “**A unified passivity-based control framework for position, torque and impedance control of flexible joint robots**”. The international journal of robotics research, 26(1), 23-39.
- [2] Kelly, R., Davila, V. S., & Perez, J. A. L. (2006), “**Control of robot manipulators in joint space**”. Springer Science & Business Media.
- [3] Luh, J. Y. S. (1983), “**Conventional controller design for industrial robots—A tutorial**”. IEEE Transactions on Systems, Man, and Cybernetics, (3), 298-316.
- [4] Rohrs, C., Valavani, L., Athans, M., & Stein, G. (1985), “**Robustness of continuous-time adaptive control algorithms in the presence of unmodeled dynamics**”. IEEE Transactions on Automatic Control, 30(9), 881-889.
- [5] Isidori, A., (1989), “**Nonlinear Control Systems**”, Springer, Heidelberg
- [6] Craig, J. J., Hsu, P., & Sastry, S. S. (1987), “**Adaptive control of mechanical manipulators**”. The International Journal of Robotics Research, 6(2), 16-28.
- [7] Fateh, M. M. (2012), “**Nonlinear control of electrical flexible-joint robots**”. Nonlinear Dynamics, 67(4), 2549-2559.
- [8] Fateh, M. M. (2012), “**Robust control of flexible-joint robots using voltage control strategy**”. Nonlinear Dynamics, 67(2), 1525-1537.
- [9] Qu, Z., and Dawson, D. M. (1996), “**Robust tracking control of robot manipulators**”, New York: IEEE Press.
- [10] Cheah, C.C., Hirano, M., Kawamura, S., Arimoto, S., (2003), “**Approximate Jacobian control for robots with uncertain kinematics and dynamics**”, IEEE Trans. Robot. Autom, Vol.19, No.4, pp.692–702.
- [11] Fateh M.M., Soltanpour M.R., (2009), “**Robust task-space control of robot manipulators under imperfect transportation of control space**”, International Journal of Innovative Computing, Information and Control, Vol. 5, No. 11(A), pp. 3949-3960.
- [12] Abdallah, C., Dawson, D. M., Dorato, P., & Jamshidi, M. (1991), “**Survey of robust control for rigid robots**”. IEEE Control Systems, 11(2), 24-30.
- [13] Fateh, M. M. (2010), “**Proper uncertainty bound parameter to robust control of electrical manipulators using nominal model**”, Nonlinear Dynamics, Vol. 61, No. 4, pp. 655–666.

- [14] Fateh M. M., Azargoshasb S. and Khorashadizadeh S. (2014), “**Model-free discrete control for robot manipulators using a fuzzy estimator**”, The International Journal for Computation and Mathematics in Electrical and Electronic Engineering, Vol. 33, No. 3, pp. 1-18.
- [15] Fateh, M. M., Ahmadi, S. M., and Khorashadizadeh, S., (2014), “**Adaptive RBF network control for robot manipulators**”, Journal of AI and Data Mining, In Press.
- [16] Spong M., W., (1992), “**On the robust control of robot manipulators**”, IEEE Transaction on automatic control, Vol. 37, No. 11, pp. 1782-1786.
- [17] Spong M., W., Vidyasagar M., (1985), “**Robust nonlinear control of robot manipulators**”, Proceedings of the 24th conference on decision and control, pp. 1767-1772, Ft. Lauderdale, FL.
- [18] Spong M., W., Vidyasagar M., (1987), “**Robust linear compensator design for nonlinear robotic control**”, IEEE Journal of Robotics and Automation, Vol. 3, No. 4, pp. 345-351.
- [19] James T. I., Huang T. J-H. (1994), “**Robust nonlinear control of robot manipulators**”, Robotic and Automation, IEEE international conference, Vol. 3, pp.2083-2088.
- [20] Jafarov, E. M., Parlakci, M. A., & Istefanopulos, Y. (2005), “**A new variable structure PID-controller design for robot manipulators**”. IEEE Transactions on Control Systems Technology, 13(1), 122-130.
- [21] Slotine, J. J., & Sastry, S. S. (1983). “**Tracking control of non-linear systems using sliding surfaces, with application to robot manipulators**”. International journal of control, 38(2), 465-492.
- [22] Soltanpour, M. R., & Fateh, M. M. (2009), “**Adaptive robust tracking control of robot manipulators in the task-space under uncertainties**”. Australian Journal of Basic and Applied Sciences, 3(1), 308-322.
- [23] Su CY., Stepanenko Y., (1994), “**Adaptive sliding mode control of robot manipulators**”, general sliding manifold case, Automatica, Vol. 30, No. 9, pp. 1497-1500.
- [24] Fateh, M. M., & Fateh, S. (2012), “**Decentralized direct adaptive fuzzy control of robots using voltage control strategy**”. Nonlinear Dynamics, 70(3), 1919-1930.
- [25] Ying, H., (1998), “**The Takagi–Sugeno fuzzy controllers using the simplified linear control rules are nonlinear variable gain controllers**”, Automatica, Vol. 34, No. 2, pp. 157–167
- [26] Fateh M.M., (2010), “**Robust fuzzy control of electrical manipulators**”, Journal of Intelligent and Robotic Systems, Vol. 60, No. (3), pp. 415-434.

- [27] Wang, L. X., & Mendel, J. M. (1992), “**Fuzzy basis functions, universal approximation, and orthogonal least-squares learning**”, IEEE transactions on Neural Networks, Vol. 3, No. (5), pp. 807-814.
- [28] Wang, L. X., (1995), “**Design and analysis of fuzzy identifiers of nonlinear dynamic systems**”, IEEE transactions on automatic control, Vol. 40, No. (1), pp.11-23.
- [29] Giordano V., Topalov A.V., Kaynak O., Turchiano B., (2004), “**Sliding-mode approach for on-line neural identification of robotic manipulators**”, 5th Asian Control Conference., Vol.3, pp. 2060- 2065.
- [30] Sadati N., Ghadami R., Bagherpour M., (2005), “**An adaptive neural network sliding controller for robotic manipulators**”, IEEE International Conference on Industrial Technology, pp.1246-1251.
- [31] Xuemei R., Rad A.B., Lewis F.L., (2007), “**Neural Network-Based Compensation Control of Robot Manipulators with Unknown Dynamics**”, American Control Conference, pp.13-18.
- [32] Wai R.J., (2003), “**Tracking control based on neural network strategy for robot manipulators**”, Neurocomputing, Vol. 51, pp. 425-445.
- [33] Piltan, F., Sulaiman, N., Roosta, S., Marhaban, M. H., & Ramli, R. (2011), “**Design a new sliding mode adaptive hybrid fuzzy controller**”. Journal of Advanced Science & Engineering Research, 1(1), 115-123.
- [34] Li, T. H. S., & Huang, Y. C. (2010), “**MIMO adaptive fuzzy terminal sliding-mode controller for robotic manipulators**”. Information Sciences, 180(23), 4641-4660.
- [35] Kayacan, E., Kayacan, E., Ramon, H., & Saeys, W. (2013), “**Adaptive neuro-fuzzy control of a spherical rolling robot using sliding-mode-control-theory-based online learning algorithm**”. IEEE Transactions on Cybernetics, 43(1), 170-179.
- [36] Wai, R. J., & Muthusamy, R. (2013), “**Fuzzy-neural-network inherited sliding-mode control for robot manipulator including actuator dynamics**”. IEEE Transactions on Neural Networks and learning systems, 24(2), 274-287.
- [37] Sharkawy, A. B., & Salman, S. A. (2011), “**An adaptive fuzzy sliding mode control scheme for robotic systems**”. Intelligent control and automation, 2(04), 299.
- [38] Fateh, M. M., & Khorashadizadeh, S. (2012), “**Robust control of electrically driven robots by adaptive fuzzy estimation of uncertainty**”. Nonlinear Dynamics, 69(3), 1465-1477.
- [39] Malis, E., & Chaumette, F. (2002), “**Theoretical improvements in the stability analysis of a new class of model-free visual servoing methods**”. IEEE Transactions on robotics and automation, 18(2), 176-186.

- [40] Fard, M. B., & Khorashadizadeh, S. (2015), “**Model free robust impedance control of robot manipulators using fourier series expansion**”. In AI & Robotics (IRANOPEN), 2015 (pp. 1-7). IEEE.
- [41] Ahmadi, S. M., & Fateh, M. M. (2018), “**Task-space control of robots using an adaptive Taylor series uncertainty estimator**”. International Journal of Control, (just-accepted), 1-29
- [42] Fateh, M. M., & Khorashadizadeh, S. (2012), “**Optimal robust voltage control of electrically driven robot manipulators**”. Nonlinear Dynamics, 70(2), 1445-1458.
- [43] Vijay, M., & Jena, D. (2017), “**PSO based neuro fuzzy sliding mode control for a robot manipulator**”. Journal of Electrical Systems and Information Technology, 4(1), 243-256.
- [44] Fateh, M. M., & Baluchzadeh, M. (2016), “**Discrete-time repetitive optimal control: Robotic manipulators**”. Journal of AI and Data Mining, 4(1), 117-124.
- [45] Ni, J., Wu, L., Fan, X., & Yang, S. X. (2016), “**Bioinspired intelligent algorithm and its applications for mobile robot control: a survey**”. Computational intelligence and neuroscience, 2016, 1.
- [46] Brand, M., & Yu, X. H. (2013), “**Autonomous robot path optimization using firefly algorithm. In Machine Learning and Cybernetics (ICMLC)**”, 2013 International Conference on (Vol. 3, pp. 1028-1032). IEEE.
- [47] Nedic, N., Stojanovic, V., & Djordjevic, V. (2015), “**Optimal control of hydraulically driven parallel robot platform based on firefly algorithm**”. Nonlinear Dynamics, 82(3), 1457-1473.
- [48] Guo, J., Gao, Y., & Cui, G. (2015), “**The path planning for mobile robot based on bat algorithm**”. International Journal of Automation and Control, 9(1), 50-60.
- [49] Fister, D., Safarič, R., & Fister, I. (2016), “**Parameter tuning of PI-controller with Bat algorithm**”. Informatica, 40(1).
- [50] Li, G., Xu, H., & Lin, Y. (2018), “**Application of bat algorithm based time optimal control in multi-robots formation reconfiguration**”. Journal of Bionic Engineering, 15(1), 126-138.
- [51] Ortega R., Spong M., W.,(1988), “**Adaptive motion control of rigid robots: a tutorial**”, Proceedings of the 27th conference on decision and control, pp. 1575-1584, Austin, Texas.
- [52] Fateh M.M., (2008), “**On the voltage based control of electrical manipulators**”, International Journal of Control, Automation and System, Vol. 6, No.5, pp. 702–712.

- [53] Soltanpour M. R., Fateh M.M., (2009), “**Sliding mode robust control of robot manipulator in the task space by support of feedback linearization and backstepping control**”, World Applied Sciences Journal, Vol. 6, No. 1, pp. 70–76.
- [54] Spong M. W., Hutchinson, S., and Vidyasagar M. (2006), “**Robot modeling and control**”, Wiley, Hoboken.
- [55] Yang, X. S. (2010), “**Engineering optimization: an introduction with metaheuristic applications**”. John Wiley & Sons.
- [56] GirirajKumar, S. M., Jayaraj, D., & Kishan, A. R. (2010), “**PSO based tuning of a PID controller for a high performance drilling machine**”. International Journal of Computer Applications, 1(19), 12-18.
- [57] Eberhart, R., & Kennedy, J. (1995), “**A new optimizer using particle swarm theory**”. In Micro Machine and Human Science, 1995. MHS'95., Proceedings of the Sixth International Symposium on (pp. 39-43). IEEE.
- [58] Gazi, V., & Passino, K. M. (2011), “**Swarm stability and optimization**”. Springer Science & Business Media.
- [59] Modares, H., & Sistani, M. B. N. (2011), “**Solving nonlinear optimal control problems using a hybrid IPSO–SQP algorithm**”. Engineering Applications of Artificial Intelligence, 24(3), 476-484.
- [60] X.S. Yang, (2008), Firefly algorithm, Nature-Inspired Metaheuristic Algorithms 20 ,79–90.
- [61] Yang, X. S. (2010), “**A new metaheuristic bat-inspired algorithm**”. In Nature inspired cooperative strategies for optimization (NICSO 2010) (pp. 65-74). Springer, Berlin, Heidelberg.
- [62] Yang, X. S., & He, X. (2013), “**Bat algorithm: literature review and applications**”. International Journal of Bio-Inspired Computation, 5(3), 141-149.
- [63] Guo, J., Gao, Y., & Cui, G. (2015), “**The path planning for mobile robot based on bat algorithm**”. International Journal of Automation and Control, 9(1), 50-60.

[۶۴] فروزان، م، نیرومند، م، (۱۳۹۰)، روش‌های نوین بهینه‌سازی، اصفهان: جهاد دانشگاهی

[۶۵] سیدکیا، م، (۱۳۸۹)، الگوریتم ژنتیک در matlab، تهران: خدمات نشر کیان رایانه سبز

[۶۶] جلالی، غ، جعفری اسکندری، م، نوذری، ح، (۱۳۹۱)، بهینه‌سازی مهندسی با تکیه بر روش‌های فراابتکاری، تهران: موسسه فرهنگی هنری دیباگران

[۶۷] مطیع قادر، ح، لطفی، ش، سید اسفهلان، م، (۱۳۸۹)، مروری بر برخی از روش‌های بهینه‌سازی هوشمند، شبستر: دانشگاه آزاد اسلامی

Abstract

In this thesis, designing an optimal controller is presented by employing intelligent evolutionary algorithms to the position control of robotic arms based on voltage control strategy. As a case study, a robust control law based on voltage control strategy is proposed and applied to the position control of a two-link robotic arm in the presence of uncertainty. An advantage of the proposed control scheme is the repeatability of the closed loop system, which makes the use of optimization algorithms possible. Then Coefficients of the controller are calculated optimally to attain the control goals. The cost-function of optimization is evaluated using the mean square error index.

Three optimization algorithms namely particle swarm optimization, firefly algorithm, and bat algorithm are compared in terms of accuracy and speed of convergence. The proposed control law is free from robot dynamics due to the use of voltage control strategy. Robustness and stability of the proposed control system are achieved through optimization algorithms.

Keywords: Optimal Robot Control, Intelligent Robot Control, Voltage Control Strategy, Repeatability , Intelligent Evolutionary Algorithm, Robotic manipulator



دانشگاه صنعتی شاهرود

Shahroud University of Technology

Faculty of Electrical and Robotic Engineering

Optimization control of robot using intelligent algorithms

based on voltage control strategy

By: Yasser Zamandari

Supervisor:

Prof. Mohammad Mehdi Fateh

September 2018