

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ



دانشکده برق - رباتیک

عنوان :

طراحی کنترل کننده فازی رباست
برای
کنترل موقعیت بازوی ماهر لولایی

نگارش:

لیلا فلاح عراقی

اساتید راهنما :

آقای دکتر قره ویسی، آقای دکتر فاتح

پایان نامه جهت اخذ درجه کارشناسی ارشد

در رشته برق-کنترل

جز نقش تو در نظر نیامد ما را

جز کوی تو رهگذر نیامد ما را

خداوند مهربان را شکر میگویم او که

یاور همیشگی ام در زندگی است.

سلام من به روح بزرگ پدرم

او که اسوهٔ ایمان و صبر در زندگی ام گشت.

پدرم! درود من به روح پاکت

سلام من به روح آسمانی مادرم

ای سرفی شقایق، لطافت شبنم، شمیم یاس!

زندگی ام لحظه ای غالی از دعای یاد تو مباد.

با تشکر از زحمات کلیه معلمان، دبیران و استادان محترم

مشعلهای فروزندهٔ طریق علم

درود رهروان علم تقدیم فداکاری هایتان

در اینجا از جناب آقای دکتر قره ویسی و جناب آقای دکتر فاتح که بعنوان

اساتید راهنما، زحمات فراوانی را متحمل شدند صمیمانه تشکر میکنم.

و در پایان برای تمامی اساتید گرامی در گروه برق- کنترل دانشگاه صنعتی

شاهرود آرزوی موفقیت و پیروزی دارم.

عنوان	صفحه
مقدمه	۱
فصل اول: روشهای کنترل ربات	۴
۱-۱- کنترل PID	۷
۱-۲- کنترل موقعیت با روش گام به گام	۹
۱-۳- کنترل کننده پیشرو	۹
۱-۴- کنترل امپدانس	۱۰
۱-۵- کنترل با گشتاور محاسبه شده	۱۲
۱-۶- کنترل تطبیقی	۱۴
۱-۷- کنترل ساختار متغیر	۱۶
۱-۸- کنترل فازی	۱۸
فصل دوم: دینامیک بازوی لولایی	۲۰
۱-۲- رباتهای حرکت مستقیم	۲۱
۱-۲-۱- کنترل رباتهای حرکت مستقیم	۲۳
۱-۲-۲- کنترل ربات سریع العمل	۲۴
۱-۲-۳- اهمیت رباتهای حرکت مستقیم	۲۵
۲-۲- دینامیک بازوی ماهر	۲۶

۲۷	۱-۲-۲- دینامیک بازوی تک مفصلی
۳۱	۲-۲-۲- بازوی موتوردار.....
۳۷	۳-۲- دینامیک بازوی دو رابط
۳۹	۴-۲- دینامیک بازوی سه رابط
۴۰	۵-۲- گشتاور محاسبه شده برای بازوی ماهر حرکت مستقیم
۴۱	۶-۲- نتیجه گیری
۴۲	فصل سوم: بررسی سیستمهای کنترل فازی و روش جبران سازی موازی
۴۴	۱-۳- مجموعه های فازی و منطق فازی
۴۵	۳-۱-۱- توابع عضویت.....
۴۸	۳-۱-۲- پایگاه قوانین و استنتاج.....
۴۹	۳-۲- مدلسازی و کنترل فازی.....
۵۱	۳-۲-۱- یادگیری در سیستمهای منطق فازی
۵۱	• دانش افراد خبره
۵۱	• داده نمونه گیری شده.....
۵۲	• خوشه سازی.....
۵۴	• تقریب کمترین مربع خطا از داده ورودی - خروجی.....
۵۵	۳-۳- وضعیتهای چندمرحله ای.....

۵۸.....	۳-۳-۱- کنترل بر اساس عملکرد
۵۹.....	۳-۳-۲- کنترل ترتیبی
۶۱.....	۳-۴-۴- مدل فازى تاكاگى- سوگنو
۶۲.....	۳-۴-۱- ساختار مدل فازى
۶۴.....	۳-۴-۲- غيرخطى ناحیه ای
۷۰.....	۳-۴-۳- تقريب محلی درفضاهای مجزاء فازى
۷۲.....	۳-۵- جبران توزيعی موازى
۷۷.....	۳-۶- طراحی کنترل <i>LMI</i>
۷۸.....	۳-۶-۱- شرایط پایداری
۷۹.....	۳-۶-۲- شرایط پایداری آرام شده
۸۰.....	۳-۶-۳- طراحی کنترل کننده پایدار
۸۲.....	۳-۶-۴- طراحی کنترل کننده فازى با استفاده از شرایط پایداری آرام شده
۸۲.....	۳-۷- سرعت نزولى
۸۲.....	۳-۷-۱- طراحی کنترل کننده فازى <i>Decay Rate</i>
۸۳.....	۳-۷-۲- شرایط پایداری آرام شده با استفاده از کنترل کننده <i>Decay Rate</i>
۸۴.....	۳-۸- محدودیت روى ورودى و خروجى کنترل
۸۴.....	۳-۸-۱- محدودیت روى ورودى کنترل

۸۴.....	۳-۸-۲- محدودیت روی خروجی.....
۸۵.....	۳-۸-۳- شرط مستقل حالت اولیه.....
۸۵.....	۳-۹- حذف اغتشاش.....
۸۶.....	۳-۱۰- کنترل فازی <i>Robust</i>
۸۷.....	۳-۱۰-۱- مدل فازی شامل عدم قطعیت.....
۸۸.....	۳-۱۰-۲- شرط پایداری ریاست.....
۹۰.....	کاربردها.....
۹۰.....	۳-۱۱- نتیجه گیری.....
۹۲.....	فصل چهارم: طراحی کنترل کننده <i>PDC</i> فازی ریاست برای ربات تک مفصلی.....
۹۴.....	۴-۱- کنترل دینامیکهای ربات و موتورهای مفاصل.....
۹۷.....	۴-۲- طراحی کنترل کننده فازی برای بازوی تک رابط.....
۱۰۰.....	۴-۲-۱- خطی سازی سیستم غیرخطی با روش تاکاگی-سوگنو.....
۱۰۲.....	۴-۲-۲- طراحی کنترل کننده فازی <i>PDC</i> برای بازوی ماهر تک رابط.....
۱۰۵.....	۴-۳- نتایج حاصل از شبیه سازی بازوی ماهر تک رابط.....
۱۰۵.....	۴-۳-۱- نتایج حاصل از شبیه سازی بدون اعمال اغتشاش.....
۱۰۷.....	۴-۳-۲- نتایج حاصل از شبیه سازی با حضور اغتشاش.....
۱۰۹.....	فصل پنجم: تعمیم کنترل کننده <i>PDC</i> فازی ریاست برای بازوهای ماهر n رابط.....

- ۱-۵- مدل سازی مدل دینامیکی یک بازوی ماهر دو رابط ۱۱۰
- ۱-۱-۵- فرمول سازی مسئله ۱۱۱
- ۲-۱-۵- خطی سازی سیستم غیر خطی با روش تاکاگی - سوگنو ۱۱۴
- ۳-۱-۵- طراحی کنترل کننده فازی *PDC* برای بازوی ماهر دو رابط ۱۲۰
- ۲-۵- نتایج حاصل از شبیه سازی بازوی ماهر دو رابط ۱۲۳
- فصل ششم: نتیجه گیری و پیشنهادات** ۱۲۷
- ۱-۶- نتیجه گیری ۱۲۸
- ۲-۶- پیشنهادات ۱۲۹

منابع

چکیده

در این پایان نامه، طراحی کنترل کننده فازی پایدار برای یک سیستم غیرخطی با استفاده از روش PDC فازی مورد بررسی قرار میگیرد و سپس روش ارائه شده برای کنترل بازوی تک رابط و دو رابط راه اندازی شده توسط موتور DC ، مورد استفاده قرار میگیرد.

روشهای مختلفی برای کنترل بازوی ماهر استفاده شده اند که از این جمله میتوان به کنترل PID ، کنترل گام به گام، کنترل پیشرو، کنترل امپدانس، ... و کنترل فازی اشاره نمود.

کنترل فازی یکی از روشهایی است که در دهه های اخیر برای کنترل سیستمهای غیرخطی مورد استفاده قرار گرفته است.

برای طراحی یک کنترل کننده فازی ابتدا لازم است ماتریس فضای حالت از یک مدل ترکیبی دینامیک بازو و موتور محرک با استفاده از روش تاکاگی سوگنو حاصل شود و سپس یک کنترل کننده فیدبک حالت بفرم جبرانساز توزیع موازی (PDC) بر اساس کنترل کننده فازی طراحی شده بمنظور کاهش خطای تعقیب و پایداری سیستم فازی استخراج شود و با استفاده از تئوری لیاپانف شرایط کافی برای تضمین پایداری ریاست برای سیستمهای مورد نظر توسط یک مجموعه از نامساویهای ماتریس خطی حاصل گردد. استفاده از سیستمهای چند مرحله ای یکی از شیوه هایی است که کاهش تعداد قوانین فازی منجر میگردد و با استفاده از این تکنیک یک سیستم پیچیده قابل تبدیل به چندین سیستم ساده تر و مجزا از هم خواهد بود.

نتایج بدست آمده در این مقاله گویای این هستند که روش حاصل یک راه حل مناسب برای پایداری سیستمهای پیچیده و دارای عدم قطعیت فراهم می آورد.

مقدمه

سیستمهای رباتیکی نقش مهمی را در جوامع ما ایفاء میکنند و حضورشان و وابستگی ما به آنها در حال رشد است. آنها در مواردی که حضور انسان ممکن نبوده و یا خطرناک باشد بهترین جایگزین برای انسانها بشمار می آیند تکنیکهای بسیاری برای کنترل رباتها وجود دارد نظیر شبکه عصبی، الگوریتم تکاملی و منطق فازی.

کنترل فیدبک به رباتها افزوده شد تا استانداردهای کیفی با اطمینان از تعقیب دقیق تغییرات با پارامترهای ربات، افزایش یابد.

کنترل فیدبک خطا مشتقی - تناسبی (PD) و انتگرالی - مشتقی - تناسبی (PID) در ۳۰ سال اخیر، این کارها را بنحو احسن انجام داده است، اما با افزایش نیاز بیشتر برای سرعت و انعطاف پذیری در ساخت، رباتها با توانایی بیشتری در توان و دقت مجهز شده اند و نیز تکنیکهای کنترل جدیدتری نظیر منطق فازی رباست و فیدبک غیرخطی، برای افزایش نیازمندیهای عملکرد استفاده شده اند، این تلاشها نتایج رضایت بخش اما مدلهای سیستم پیچیده ای را بدنبال داشته اند [18].

تکنیکهای کنترل چند مرحله ای، نمایش یک کنترل کننده پیچیده به بخشهای ساده تر را توسعه داده اند. نتیجه، یک کنترل کننده است با محاسبات کمتر برای رسیدن به نتایج مشابه با همان کنترل کننده پیچیده که حافظه کمتری را استفاده میکند [10].

کنترل و مدلسازی چند مرحله ای بعنوان یک کاندیدای ابتدایی در کنترل رباتهای جدید، سرعت بالا و حرکت مستقیم دیده میشود [10].

منطق فازی یک ابزار کلی برای کاربردهای کنترلی ارائه کرده است. دو نوع مدل فازی وجود دارد: مدل فازی ممدانی و مدل فازی تاکاگی-سوگنو. کاربرد ویژه سیستمهای فازی تاکاگی-سوگنو بدلیل توانایی شان در تقریب رفتارهای غیرخطی است [5]. از سال ۱۹۹۲ پایداری سیستم فیدبک حالت شامل یک کنترل کننده فازی بطور گسترده مطرح شده است. تاناکا و سوگنو یک قضیه بر اساس تحلیل پایداری مدل فازی $T-S$ ارائه دادند. وانگ ات ال روش جبران توزیعی موازی را بعنوان یک زمینه طراحی پیشنهاد داد و همینطور نتایج تحلیلی را بکمک یک الگوریتم کنترل بمنظور بررسی پایداری با استفاده از بهینه سازی قیود نامساوی ماتریس خطی اصلاح نمود. زاوات ال یک طرح کنترل سیستماتیک برای پایدار نمودن کنترل کننده فازی بر اساس بهینه سازی جملات LMI پیشنهاد داد. استفاده از منطق فازی برای نمایش عدم قطعیهائی که با تئوری احتمال قابل حل نیستند، مزیت دیگری برای آن می باشد بدین ترتیب کنترل PDC فازی یکی از مفیدترین تکنیکهای کنترل ربات بشمار می آید. برای توصیف یک سیستم پیچیده، طراحی با کنترل فازی چندمرحله ای به کاهش تعداد متغیرها و نیز کاهش تعداد قوانین کنترل کننده می انجامد [10].

در این مقاله CFS و DFS بترتیب بیانگر سیستمهای کنترل فازی پیوسته و گسسته هستند.

$S > 0$ به این معنی است که S یک ماتریس معین مثبت است، $S > T$ یعنی $S - T > 0$ و $W = 0$ یعنی W یک ماتریس صفر است که همه المانهای آن صفرند.

در چندین مورد ویژه از اختصارات زیر استفاده کرده ایم:

$$i < j \quad s.t. h_i \cap h_j \neq \phi,$$

$$l \leq j \quad s.t. h_l \cap h_j \neq \phi,$$

برای مثال ، شرط $s.t. h_i \cap h_j \neq \emptyset$, $i < j \leq r$ به مفهوم آن است که شرط باید برای تمامی

$i < j$ هایی که $h_i \cap h_j \neq \emptyset$ (برای همه $z(t)$ ها، $(h_i(z(t)) \times h_j(z(t))) = 0$ برقرار باشد.

$h_i(z(t))$ یعنی وزن i امی قانون محاسبه شده از توابع عضویت در مراحل مقدماتی و r تعداد

قوانین اگر- آنگاه است . توجه داشته باشید که $h_i \cap h_j \neq \emptyset$ است اگر و فقط اگر i امین و j امین

قانون هیچگونه اشتراکی نداشته باشند.

فصل اوّل :

مروشهای کنترل مربّات

بمنظور استفاده از یک ربات لازم است بتوانیم بازوهای آنرا بطور دلخواه بحرکت درآوریم.

رسیدن به این هدف در مبحث کنترل ربات بررسی میشود. برای انتقال یک جسم از یک نقطه به نقطه دیگر توسط ربات دو روش وجود دارد، یکی رساندن جسم از نقطه مبدأ به نقطه نهایی از هر مسیر دلخواه ممکن (حرکت نقطه به نقطه) و دیگری حرکت جسم از مبدأ به مقصد تحت مسیر کاملاً مشخص و از پیش طراحی شده (کنترل مسیر یا تعقیب مسیر⁽¹⁾).

در حرکت نقطه به نقطه، کنترل موقعیت ربات بدون توجه به مسیر حرکت انجام می شود و آن برای عملیاتی مفید است که از نظر طبیعت مجزا هستند، مثلاً جوش نقطه ای، کاربردی از حرکت نقطه به نقطه ابزار می باشد. کنترل مسیر در حالتی مورد استفاده قرار میگیرد که بدلیل وجود قیود کاری، لازم است ربات مسیر خاصی را در فضا دنبال کند، در اینحالت نیاز داریم اطلاعات مربوط به حرکت مطلوب ربات، بصورت مجموعه نقاط معین و پشت سرهم، برای متغیرهای مفصل معین شوند. این اطلاعات، مقادیر مطلوب متغیرهای مفاصل را در تمام زمانها مشخص میکند. طراحی میتواند از ابتدا با استفاده از چندجمله ایهای پیوسته و مشتق پذیر در مختصات

1-Trajectory control or Trajectory Tracking

متغیرهای مفاصل حل شود و یا در مرحله اول، حرکت مطلوب ربات در فضای سه بعدی مشخص شده و سپس با حل مسئله سینماتیک معکوس، این متغیرها را بدست آورد.

مثالهای کاربردی رباتهایی با کنترل حرکت پیوسته شامل پاشش رنگ، جوشکاری قوس و کارهای چسباندن یا آب بندی کردن است. مسلم است که مسئله تعقیب مسیر از حرکت نقطه به نقطه، به مراتب مشکل تر است.

منظور از کنترل بازوی ربات، بدست آوردن قانون کنترلی است که با در نظر داشتن دینامیک بازو و مسیر مطلوب حرکت بتواند سیگنال کنترلی را تولید کند که موجب حرکت مفاصل ربات برای تعقیب آن مسیر گردد. در مسئله کنترل با مشکلات زیر روبرو هستیم:

- مدل دینامیکی بازوی ربات، کاملاً غیرخطی است.
- تعیین دقیق پارامترهای مدل در مواردی که انعطاف در مفاصل داریم و ربات کاملاً صلب نیست میسر نمی باشد و با توجه به وجود تغییرات بار و ممان اینرسی و اصطکاک، با اغتشاشات روبرو هستیم [6].

با توجه به مسائل عنوان شده ، روشهای مختلفی برای کنترل بازوی ماهر ارائه گردیده است که برخی از آنها عبارتند از : کنترل PID ، کنترل گام به گام ⁽¹⁾ ، کنترل پیشرو ⁽²⁾ ، کنترل امپدانس ⁽³⁾ ، کنترل گشتاور محاسبه شده ⁽⁴⁾ ، کنترل تطبیقی ⁽⁵⁾ ، کنترل ساختار متغیر ⁽⁶⁾ و کنترل فازی ⁽⁷⁾ .

در ادامه در مورد هر کدام از این روشها مختصری شرح داده و محاسن و معایب آنها را بطور خلاصه ذکر میکنیم.

۱-۱- کنترل PID

یکی از روشهای معمول کنترل بازوی ماهر که در بسیاری از رباتهای صنعتی مورد استفاده قرار میگیرد کنترل PID است. برای تشریح ریاضی PID فرض کنید $r(t) \in R^n$ نمایش ورودی مرجع مورد نظر بوده و هدف ، دنبال کردن متغیرهای مفصل است. $e(t)$ بیانگر خطای ردیابی در زمان t است. اگر $e(t)$ را اختلاف بین ورودی مرجع و متغیر مرجع بنامیم ، در اینحالت با استفاده از خطا، مشتق خطا و انتگرال آن ، به محاسبه گشتاور مورد نیاز می پردازیم.

$$e(t) \cong r(t) - q(t) \quad (1-1)$$

1-step by step control

5- adaptive control

2- Feed forward control

6- variable structure control

3- Impedance control

7- fuzzy control

4- Computed torque control

۱-۲- کنترل موقعیت با روش گام به گام

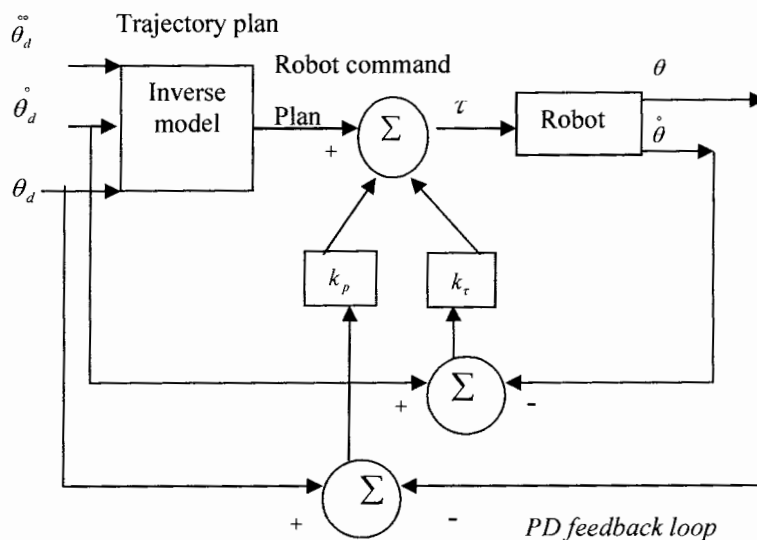
یک روش کنترل جدید مورد قبول در صنعت رباتهای حرکت مستقیم بدون چرخ دنده ، روش یادگیری مسیر است که یادگیری از طریق تکرار نامیده میشود [10]. در این روش ، ربات برای ادامهٔ صحیح مسیر نقشه ، فرمان را بصورت مرحله به مرحله یاد میگیرد . این کنترل کننده ، مسیر فرمان ربات را در زمان اجرای مسیر اصلاح میکند. این روش به مدل دقیق ربات نیاز ندارد و برای سیستمهای پیچیده قابل استفاده است. در این روش بدلیل اینکه هر تغییری در نقشه و یا در ربات باید آموزش داده شود ، بنابراین صحت و دقت عمل تعقیب برای یک تعداد از سیکلها کاهش می یابد .

۱-۳- کنترل کنندهٔ پیشرو

انواع دیگری نیز از کنترل کننده ها برای کنترل سرعت رباتهای حرکت مستقیم بدون چرخ دنده ، استفاده میشوند که یکی از اینها کنترل کنندهٔ پیشرو است [10]. کنترل پیشرو ، یک مدل معکوس از ربات را استفاده میکند تا مسیر طرح را به یک طرح معمولی ربات برگرداند . همانطور که در شکل ۱-۲ آمده است، اگر مدل معکوس ربات صحیح باشد ، فرمانهای صادر شده ، بطور دقیق ربات را به مسیر خواسته شده هدایت می کند.

بدلیل سادگی و دقت و تواناییهای استفاده در سرعتهای بالا، این روش کنترل، بطور گسترده در تحقیقات رباتیکی استفاده میشود. اشکال عمده این روش ، نیاز آن به یک مدل صحیح از دینامیک

وارون ربات ، برای مسیریابی است . در مجموع، این روش هنگامی که حلقه فیدبک، شامل کنترل کننده معکوس نباشد، نسبت به ورودیهای اغتشاش به ربات ، خیلی مقاوم نیست.



شکل ۱-۲- کنترل کننده پیشرو

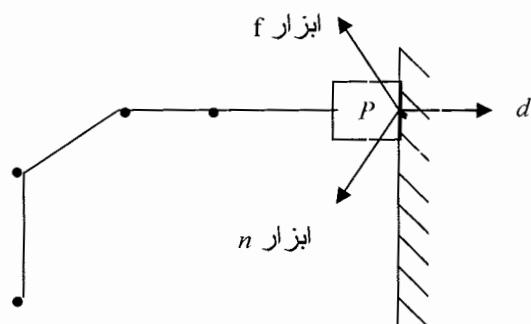
۴-۱- کنترل امپدانس

سالیزبوری (۱۹۸۰)، هوگان (۱۹۸۵)، اسراوسلوتین (۱۹۸۶)، وسین (۱۹۸۷) و آل ات آل (۱۹۸۸) روشی را برای طراحی کنترل کننده هایی ارائه دادند که هنگام تماس ابزار با محیط کار میکنند. این روش یک حوزه فعال تحقیقاتی است و عباراتی از قبیل کنترل سازگاری ، کنترل نیرو ، کنترل دورگه و کنترل امپدانس به آن اطلاق میگردد.

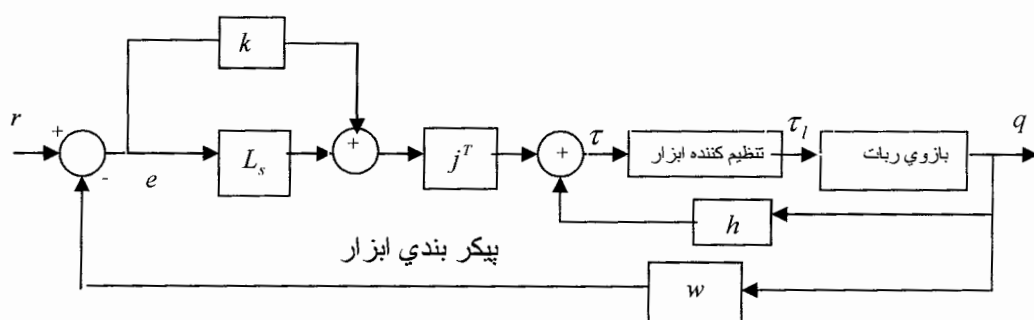
امپدانس مجری نهایی، رابطه بین نیرو (F) و سرعت (v) در آن نقطه است [19].

$$F(s) = Z(s).v(s) \quad (۳-۱)$$

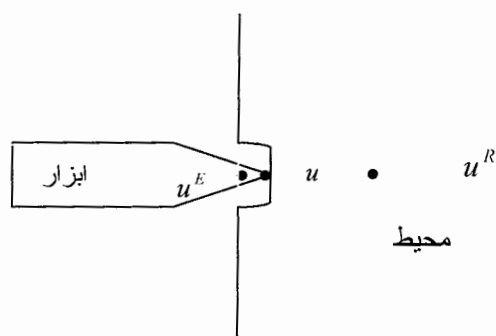
مجموعه ای از محدودیتهای طبیعی که توسط محیط به ربات اعمال میگردد مربوط به جابجایی یا دستکاری کردن است. برای مثال فرض کنید نوک ابزار با یک سطح سخت G تماس پیدا میکند، در اینجا محیط یک محدودیت طبیعی موقعیت تحمیل مینماید که از نفوذ نوک ابزار به داخل سطح G جلوگیری میکند. وقتی ابزار در تماس با سطح G است برایش معنی ندارد که برای کنترل موقعیت در جهت d ، عمود بر G اقدامی کند بلکه در جهت d نیازمند به کنترل نیرو میباشد. بهر حال ما میتوانیم نیروی نوک ابزار را در طول d کنترل کنیم، بعلاوه میتوانیم همزمان موقعیت نوک ابزار در طول جهات مماس بر G را کنترل نماییم. در این روش، ابزار میتواند وادار شود که در امتداد سطح بلغزد، این را یک حرکت سازگار در طول سطح G مینامند. نمای شماتیکی کنترل کننده امپدانس در شکل ۴-۱ آمده است.



شکل ۴-۱- محدودیت طبیعی اعمال شده به محیط



شکل ۱-۴- کنترل امپدانس یک بازوی رباتیکی [19]



شکل ۱-۵- تغییر شکل محیط توسط ربات

۱-۵- کنترل با گشتاور محاسبه شده

تکنیک دیگری که مشابه با کنترل پیشرو است، کنترل کننده گشتاور محاسبه شده یا بطور کلی کنترل کننده فیدبک خطی نامیده میشود. این روش برای حل مسئله سروربات در سال ۱۹۷۴ توسط *Bejczy* مطرح شد و آن بر مبنای استفاده از معادله دینامیکی ربات، به محاسبه گشتاور مورد نیاز برای اعمال به مفصل می پردازد، این کنترل کننده، گشتاور لازم را برای نگهداری سرعت و موقعیت جاری ربات معین میکند [10].

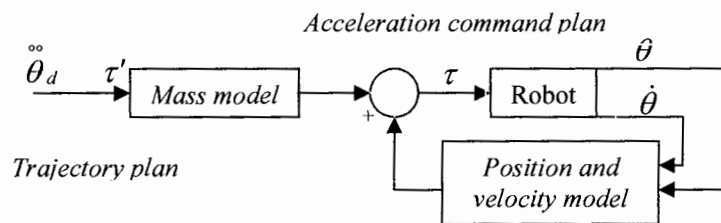
در این روش با استفاده مستقیم از معادله دینامیکی کامل بازو نه فقط اثرات جاذبه بلکه نیروی کوریولیس و گریز از مرکز، اصطکاک و تانسور اینرسی بازو را حذف می نماید و پیاده سازی کنترل گشتاور محاسبه شده باید متکی بر تخمین پارامترهای بازوی رباتیکی باشد. تغییر در موقعیت و سرعت با فرمانهای شتاب داده میشود که از طرح مسیر نشان داده شده در شکل ۱-۶ نتیجه میشود ، این روش یک مدل خطی ساده را بفرم $\frac{\theta}{\tau} = \frac{1}{s^2}$ ایجاد می کند.

روش گشتاور محاسبه شده مزیتی که دارد این است که به اغتشاشهای محیط حساسیت کمتری نشان میدهد و بیشترین مزیت آن سادگی نسبی پیاده سازی آن است.

اولین مسئله، تخمین پارامترهای ربات است که ناشی از خطاهای اندازه گیری است و دیگر آنکه مدل ربات کامل نمی باشد، بعنوان مثال انعطاف بازوهای ربات در این مدلها در نظر گرفته نمیشود. همچنین اگر ورودی مرجع، تغییرات شدیدی داشته باشد ، گشتاور محاسبه شده می تواند مقادیر بزرگ و خارج از محدوده ای را نتیجه دهد.

در واقع مشکل اصلی آن است که این کنترل کننده از نوع حلقه باز می باشد [10].

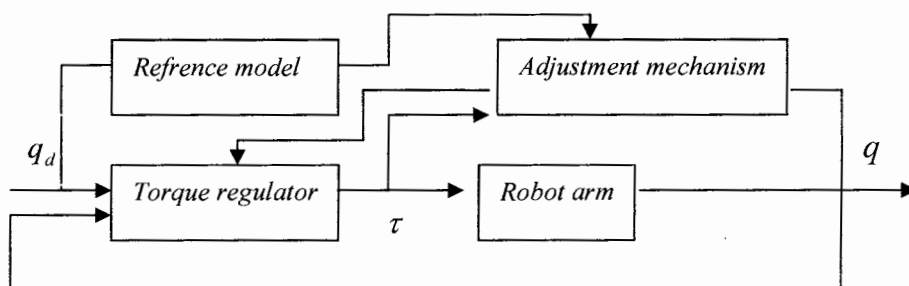
و بدین ترتیب از معایب آن عدم حساسیت به مدلسازی صحیح و نیز نیاز به محاسبه زمان حقیقی برای فیدبک و پیشرو را میتوان اشاره نمود.



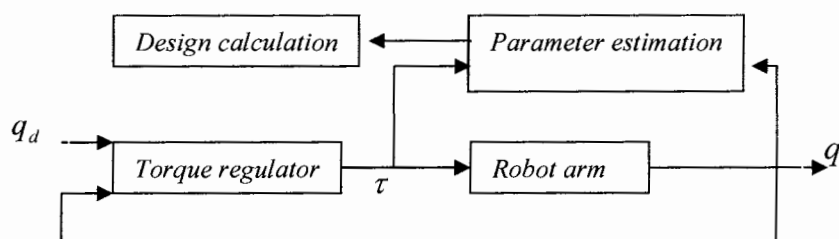
شکل ۱-۶- کنترل گشتاور محاسبه شده

۱-۶- کنترل تطبیقی

مسئله طراحی یک کنترل کننده که از مشخصات یک سیستم مستقل است، از جمله آمال و آرزوهای مهندسين کنترل است و از آنجایی که روشهای گفته شده در قسمتهای قبلی به تغییرات پارامترهای سیستم حساس هستند لذا بدنبال کنترل کننده ای هستیم که تا سر حد امکان از مدل سیستم، مستقل باشد. اولین ایده ای که به ذهن خطور میکند، استفاده از کنترل کننده تطبیقی است. در این روش، کنترل کننده خود را با سیستم تنظیم می کند و تغییرات بوجود آمده در مدل سیستم بروی آن اثر گذارده و کنترل کننده خود را با این تغییرات هماهنگ میکند. یکی از کاربردهای اخیر، استعمال یک کنترل کننده تطبیقی برای محاسبه نوسانهای فرکانس بالا در سیگنالهای کنترلی بزرگ است [10].



شکل ۱-۷- کنترل کننده تطبیقی مدل مرجع



شکل ۱-۸- کنترل کننده تطبیقی خودتنظیم

کنترل کننده تطبیقی به دو نوع عمده تقسیم میشوند:

کنترل کننده های مدل مرجع ^(۱) و کنترل کننده های خود تنظیم ^(۲) .

در بالا شمای این دو کنترل کننده آمده است.

1- Model Reference Adaptive Control

2- Self Tuning Adaptive control

در کنترل کننده های تطبیقی مدل مرجع ، پاسخ مطلوب به سیگنال ورودی را بصورت یک مدل در اختیار سیستم قرار می دهیم و بر اساس خطای میان خروجی سیستم و خروجی مدل مرجع و با استفاده از یک مکانیزم تصحیح، پارامترهای کنترل کننده را اصلاح می نماییم [10].

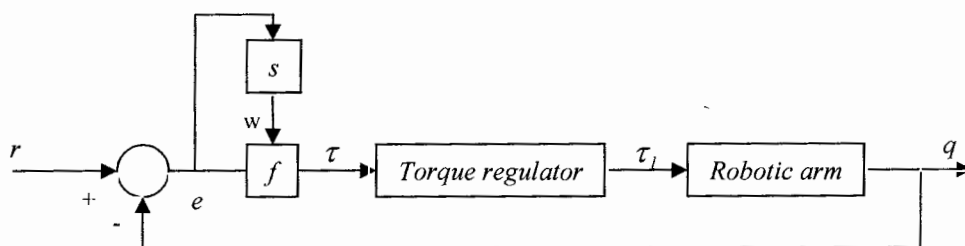
در نوع دیگر ابتدا با یک روش شناسایی پارامترهای سیستم را تخمین زده و سپس بکمک آن کنترل کننده را طراحی می کنیم. از جمله معایب این کنترل کننده ها پیچیدگی الگوریتم و طولانی بودن زمان محاسباتی ، بمنظور پیاده سازی آن می باشد [10].

۷-۱- کنترل ساختار متغیر

همانگونه که اشاره شد، منطقی که کنترل تطبیقی بر آن استوار است، تغییرات تدریجی ضرایب کنترل کننده بر اساس ارزیابی سیستم است. بر این اساس می توان الگوی دیگری را نیز پیشنهاد کرد و آن تغییرات سریع سیگنال کنترل با توجه به حالت سیستم تحت کنترل می باشد. به این دسته از سیگنالهای کنترلی ، سیستمهای ساختار متغیر گویند [10] ^(۱).

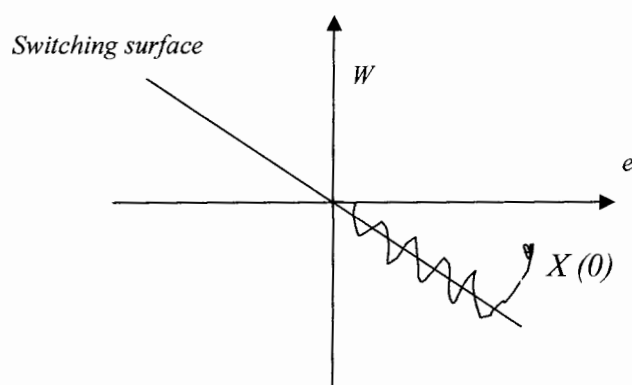
بلوک دیاگرام یک سیستم ساختار متغیر در شکل ۱-۹ آمده است.

در اینجا تنها داشتن اطلاعاتی در مورد محدوده تغییرات پارامترهای سیستم کفایت می کند و بهمین دلیل نسبت به خطای تخمین می تواند غیر حساس شود.



شکل ۹-۱- بلوک دیاگرام کنترل کننده ساختار متغیر یک بازوی رباتیک

در کنترل کننده های ساختار متغیر، با تعریف یک سطح سوئیچینگ^(۱) که در فضای حالت تعریف می شود، نوع ساختار کنترل کننده با توجه با اینکه حالت سیستم نسبت به این سطح چه موقعیتی را دارد، تغییر میکند. اگر یک سیستم درجه دوم را در نظر بگیریم و فرض کنیم که سطح سوئیچینگ آن یک خط از مبدأ باشد، شکل ۱۰-۱ وضعیت خطا نسبت به این سطح را نشان میدهد.



شکل ۱۰-۱- مد لغزشی برای یک سیستم مرتبه دوم

1-switching surface

همانطور که ملاحظه میشود، پس از رسیدن متغیرهای حالت (خطا و مشتق آن) به این سطح، سیگنال کنترل موجب حرکت زیگزاگ در اطراف سطح و هدایت آن به سمت صفر می شود، به این حالت مد لغزشی گویند.

در تئوری، فرکانس سوئیچینگ بی نهایت است اما در عمل اینطور نیست. کنترل کننده ساختار متغیر دارای دو مشکل عمده است. اول آنکه برای بدست آوردن قانون کنترلی مناسب روش تضمین شده واحدی ندارد و دیگری به این حقیقت مربوط میشود که گشتاور مورد نیاز می تواند مقداری بزرگ و غیر قابل دسترس را انتخاب نموده و ضمناً لازم است تا قابلیت تغییرات سریع را دارا باشد.

۸-۱- کنترل فازی

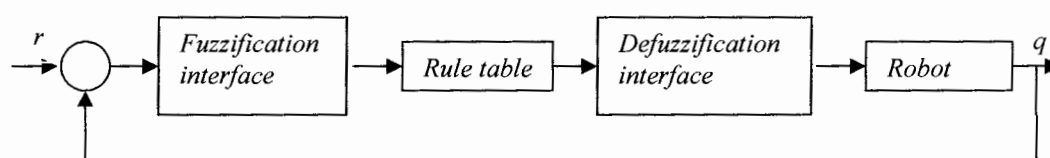
یکی از جدیدترین روشهای مطرح شده بمنظور کنترل ربات، استفاده از کنترل کننده های فازی است.

بلوک دیاگرام ساده شده این کنترل کننده در شکل ۱-۱۱ آمده است، همانطور که مشاهده میشود ابتدا خطای میان ورودی مرجع و حالات سیستم به واحد فازی کننده^(۱) منتقل میشود که وظیفه اصلی آن تبدیل مقادیر به متغیرهای زبانی^(۲) است، سپس در بلوک بعدی، با استفاده

1-Fuzzification Interface

2-Linguistic value

از قوانین فازی مربوطه که با استفاده از اطلاعات مختلف در مورد سیستم و رابطه سیگنال ورودی-خروجی بدست آمده است ، خروجی که می بایست به سیستم اعمال گردد ، بصورت یک متغیر فازی مشخص شده و در نهایت متغیر مذکور توسط واحد غیرفازی کننده⁽¹⁾ بصورت یک مقدار صحیح به بازوی ربات اعمال خواهد شد.



شکل ۱-۱۱- کنترل کننده فازی

فصل دوّم :

ديناميك بانزوي لوكايي

در این فصل ابتدا به شرح رباتهای حرکت مستقیم و کنترل آنها پرداخته ، سپس مزایای استفاده از آنها را بیان میکنیم. در بخش ۲-۲ به تفصیل به بسط مدل بازوی یک - رابط می پردازیم و پس از حصول دینامیک بازوی ماهر، شبیه سازی آنرا بدون کنترل کننده می آوریم و بعد از آن بازوی تک رابط متصل شده به موتور کنترل شونده با روتور را شرح داده و این بازوی موتوردار را توسط یک کنترل کننده PD پایدار میکنیم. بعد از آن به بسط مدل دینامیکی بازوهای دو و سه رابط می پردازیم.

۲-۱- رباتهای حرکت مستقیم^(۱)

رباتهای صنعتی معمولاً شکلی از یک سیستم مبنا با رابط های صلب هستند که بصورت سری یا موازی برای تنظیم موقعیت مجری نهایی، عمل میکنند. مجری نهایی میتواند نوعی نگهدارنده یا گیره باشد که برای جابجایی مواد یا اشیاء از محلی به محل دیگر استفاده شود و نیز این مجری نهایی میتواند یک اتصال (رابط) باشد که در موقعیتی قرار گیرد و اعمال نیروهای ممکن یک وسیله، نظیر یک جوشکار یا تفنگ نقاشی را ممکن سازد. به هر کدام از رابط ها، یک سیستم

محركه متصل است که می تواند یک رابط را نسبت به رابط دیگر بچرخاند (*revolute*) یا در امتداد آن حرکت دهد (مفاصل کشویی^(۱)).

دو نوع عمده از پیکربندی ربات وجود دارد، اولی پیکربندی باز یا پیکربندی رشته ای^(۲) نامیده میشود. پیکربندی باز از سرعت مطلوبی برخوردار است اما انعطاف پذیری آن مشکل است. بازوی انسان مثال خوبی از یک پیکر بندی باز است.

پیکربندی دیگر بازوی رباتیک، رابط بسته است^(۳) است، در این روش طراحی، یک یا چند رابط در دو نقطه یا نقاط بیشتری در طول رابط متصل میشوند. این اتصالات بیشتر، به ربات پایداری بیشتری میدهند اما سرعت کلی آنها کم میکنند. یک شکل کلی و پیچیده از ربات رابط بسته، مکانیزم استوارت^(۴) است که میتواند حرکاتی در هر سه جهت ایجاد کند و برای شبیه سازی عملکرد هواپیما بسیار مناسب است. شکل ۱-۲ مثالهایی از این رباتها را نشان میدهد.



شکل ۱-۲- نمایی از رباتهای رابط باز - رابط بسته و مکانیزم استوارت

1- Prismatic joints

2- The open link or String

3- The closed link

4- Stewart platform

سیستم محرکه مستقیم یکی از انواع روشهای راه اندازی مورد استفاده در بازوهای رباتیکی است. اولین محرکه ها بسته به توان و قابلیت اعتمادشان ، روغنی و بادی بودند. محرکهای روغنی هم اکنون نیز در ابزارهای ساختمانی و در نگهدارنده های مصالح بزرگ استفاده میشوند. محرکهای بادی از قدرت کمتری برخوردارند اما نزد انسان ایمن تر هستند. اکنون محرکهای روغنی و بادی در جاهایی که به کنترل دقیقتر نیاز است یا جاهایی که نشتی روغن سبب مشکل میشود، با موتورهای الکتریکی چرخ دنده ای جایگزین شده اند. اغلب کارهای رباتیک با موتورهای الکتریکی چرخ دنده ای انجام میشوند و سرانجام از موتورهای الکتریکی حرکت مستقیم برای حذف خلاصی چرخ دنده ⁽¹⁾ یا ناحیه مرده ایجاد شده در موتورهای الکتریکی دنده ای بمنظور افزایش ماکزیمم سرعت موتورها استفاده میشود.

۲-۱-۱- کنترل رباتهای حرکت مستقیم

برای کنترل بازوی ماهر حرکت مستقیم ، روشهای فراوانی وجود دارد که از فیدبک رابط مستقل با محاسبات زیاد تا کنترل مدل بهینه تقسیم بندی می شوند. اغلب سیستمهای صنعتی ، از فیدبک رابط مستقل یا کنترل یادگیری بصورت تکرار ILC ⁽²⁾ استفاده میکنند، به این دلیل که آنها در طراحی ساده بوده و با میکروکنترلرهای در دسترس قابل انجام هستند [10]. محققان به مطالعات بیشتری در زمینه الگوریتمهای فیدبک پیشرفته ادامه دادند و به دقت بیشتر حتی در سرعتهای بالا دست یافته اند.

1-Iterative Learning Control

2-back lash

متداولترین روش بکاررفته، کنترل مستقل هر رابط از بازوی ماهر است، که ساده ترین و راحتترین روش انجام است. کنترل کننده میتواند فیدبک *PID* کلاسیک با یک الگوریتم پیشرفته تر نظیر منطق فازی، شبکه های فازی، مدل های بهینه، طراحی رباست یا ترکیبی از این نوع کنترل کننده ها را استفاده کند. این روش، داده خطا از رابط کنترل شده را بمنظور معین کردن فیدبک برای آن رابط استفاده میکند و این محاسبات را در حد مینیمم نگه داشته و اجازه تشخیص سریع فرمان فیدبک را میدهد. بمنظور نگهداشتن معقول تعقیب مسیر، کنترل کننده های بازوی ماهر حرکت مستقیم، باید با استفاده از کنترل رابط مستقل، سرعت و شتابهای مجاز رابط های قبلی را برای طراحی مسیر محدود کند، بنابراین یکی از مهمترین مزیت های بازوهای حرکت مستقیم، سرعت محدود شده آن است.

۲-۱-۲- کنترل ربات سریع العمل

رشته های رباتیک و کنترل ربات، در سالهای اخیر شاهد پیشرفت عظیمی بوده اند. رباتها هم بزرگتر و هم بصورت کوچکتر ساخته شده اند و کاربردهای جدیدتری را عهده دار شده اند. امروزه از رباتها برای لمس کردن و گرفتن اشیاء و مواد بزرگ و مضر برای انسان و نیز جراحیهای میکروسکوپی در جوامع پیشرفته و نیز در کارهای نقاشی استفاده می شود. یکی از بیشترین کمکهای چالش برانگیز اخیر، استفاده از رباتها برای ساخت رباتها و در نتیجه مطالعه استنتاج مشخص برای اصلاح عملکرد رباتها است. یکی از این نتایج دراندازه و کاربرد، استفاده از موتورهای حرکت مستقیم است که حجم کوچک و سرعت زیاد را نتیجه میدهند. زمینه های جراحی ظریف و استفاده از رباتهای کوچک بجای انسان از دیگر تحقیقات در زمینه ربات هستند [10].

۲-۱-۳- اهمیت رباتهای حرکت مستقیم

رباتها بارهای سنگین را حمل میکنند و کارهای تکراری را بدون هیچگونه تغییری پذیری نسبت به انسانها انجام میدهند و در محیطهای سخت و دشوار نظیر اطاقهای نظافت و غرفه های نقاشی کار میکنند.

دو نمونه از متداولترین رباتهای صنعتی ، رباتهای جابجا کننده اجسام و دیگری رباتهای جوشکار/نقاش هستند که در صنعت خودروسازی استفاده حیاتی دارند. هردو نوع این رباتها ، اغلب با موتورهای حرکت مستقیم ساخته میشوند. یک موتور حرکت مستقیم ، دارای محور خروجی است که مستقیماً به رابط محرک متصل میشود و هیچ چرخ دنده و کمر بند و تسمه ای هم وجود ندارد. همچنین یک موتور حرکت مستقیم، نوعی مغناطیس عظیم و نیز سیم بندیهای مارپیچی فراوانی دارد که اجازه میدهد موتور به گشتاورهای بالا در سرعتهای پایین برسد و این دقت با ایجاد تعداد زیادی از دندانه های استاتور و روتور ایجاد میشود که باعث ایجاد سیکلهای مغناطیسی فراوانی در هر چرخش میشوند. موتورهای حرکت مستقیم در اندازه های مختلف هستند و اغلب در سیستمهای کنترل موقعیت و یا در سیستمهایی که نیازمند سرعتهای دورانی دقیق هستند، استفاده میشوند.

مثالهایی از کاربردهای عمومی : ضبط صدا، زمینه های تست ارتعاش، دیسک درایوها (مغناطیسی، CD ، DVD) و وسایل نقلیه الکتریکی هستند.

رباتهای حرکت مستقیم [18] بدلیل دقت و تکرار پذیر بودنشان بسیار مورد نیاز هستند، در اینها هیچگونه واکنش سریع و یا نواحی مرده وجود ندارد، در عوض خروجی محور موتور حرکت

مستقیم ، وابسته به رابطی است که آنها را روشن میکند. بیشترین کاربرد رباتهای حرکت مستقیم، گروه جابجا کننده های اجسام و نضافتچی ها هستند [10].

۲-۲- دینامیک بازوی ماهر

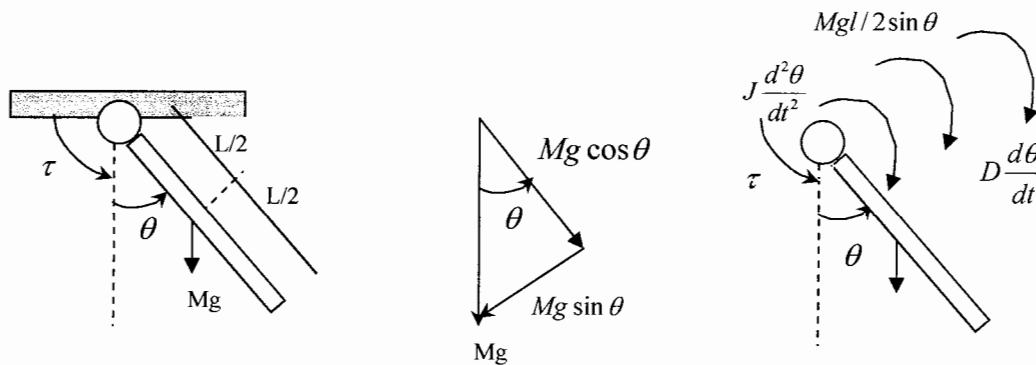
در این بخش به تفصیل به بسط مدل های بازوی ماهر یک ، دو و سه رابط می پردازیم. هنگام مدلسازی یک سیستم ربات با موتورهای حرکت مستقیم، فرض میشود که گشتاور تولید شده توسط موتور، متناسب با جریان ورودی است. بنابراین مدل رباتهای حرکت مستقیم، شامل یک مدل موتور نیستند و فقط گشتاور موتور بعنوان متغیر ورودی نشان داده میشود. دینامیکهای یک ربات حرکت مستقیم n مفصلی اغلب در فضای حالت به شکل نشان داده شده در معادله زیر نوشته میشود [10]:

$$M(\theta)\ddot{\theta} + V_{cent}(\theta)[\dot{\theta}^2] + V_{cor}(\theta)[\dot{\theta}_i\dot{\theta}_j] + G(\theta) = \tau \quad (۱-۲)$$

که در این معادله $\theta = [\theta_1, \theta_2, \dots, \theta_n]^T \in R^n$ بردار موقعیت مفصل، $[\dot{\theta}^2] = [\dot{\theta}_1^2, \dot{\theta}_2^2, \dots, \dot{\theta}_n^2]^T \in R^n$ بردار مربع سرعت مفاصل است. $[\dot{\theta}_i\dot{\theta}_j] = [\dot{\theta}_1\dot{\theta}_2, \dot{\theta}_1\dot{\theta}_3, \dots, \dot{\theta}_{n-1}\dot{\theta}_n] \in R^{(n-1) \times n/2}$ بردار حاصلضرب سرعت مفاصل است که $M(\theta) \in R^{n \times n}$ ، ماتریس لختی است، $V_{cent}(\theta) \in R^{n \times n}$ ماتریس جانب مرکز و $V_{cor}(\theta) \in R^{n \times (n-1)n/2}$ ماتریس کوریولیس است. $G(\theta) \in R^n$ بردار ثقل و $\tau \in R^n$ بردار نیرو/گشتاور مفصل است.

۲-۱-۲- دینامیک بازوی تک مفصلی

پاسخ دینامیکی یک مدل سیستم مکانیکی مانند دست انسان را در نظر بگیرید. تابع تبدیل، موقعیت زاویه ای خروجی مفصل بازو را به گشتاور ورودی تامین شده توسط ماهیچه بازو مربوط میکند. در این مدل یک گشتاور ورودی $\tau(t)$ ، ویسکوز میرایی D در مفصل بازو و لختی J در اطراف بازو را داریم و همچنین یک جزء از وزن بازو $w=Mg$ که جرم بازو M و شتاب ثقل g است، باعث ایجاد یک گشتاور غیرخطی میشود. فرض کنید بازو دارای چگالی یکنواخت است بطوریکه وزن به مرکز $l/2$ اعمال شده باشد (l : طول بازو).



شکل ۲-۲- (a) بازوی ربات (b) اجزاء بردار وزن (c) گشتاور جسم آزاد

در یک رابط صلب، حرکت چرخشی توسط گشتاور اعمالی به رابط و نیز نشان دادن جرم در مرکز رابط، طبیعی است.

پاندول یک مدل خوب برای یک بازوی ربات با یک درجه آزادی را بما میدهد. دیگر تغییرات فیزیکی به طرحهای مختلف بازوی ماهر بستگی دارد، برای مثال، اگر ما رابط صلب را بطور افقی

نصب کنیم و سپس آن را مفصل دار کنیم، تاثیر وزن در حرکت را کاهش داده ایم. از این مسئله در طراحیهای ربات اسکارا^(۱)، مونتاژکار دکارتی^(۲) و استوانه ای استفاده میشود. رباتهای صنعتی مفصل بندی شده عمودی و کروی، رابط صلب رو به بالا دارند. یک گیره و نیز یک ربات صنعتی مفصل بندی شده عمودی، نظیر پاندول ساده، اتصال رابطی بسمت پایین دارند.

دیاگرام جسم آزاد شکل ۲-۲ را در نظر بگیرید. شکل نشان میدهد که حرکت با سه گشتاور، مهار میشود: جزء وزن $(Mgl/2) \times \sin(\theta)$ ، گشتاور میرایی $D.d\theta/dt$ و گشتاور سکون $J.d^2\theta/dt^2$. برای یک جرم منگوله ای در انتهای هر بازو، سکون $J = MI^2$ است حال آنکه، برای یک بازوی جرم توزیع شده، سکون فقط $MI^2/12$ است.

معادلات دیفرانسیل جسم آزاد را بفرم زیر می نویسیم این معادله پاسخ دینامیکی سیستم را توصیف میکند.

$$J \frac{d^2\theta}{dt^2} + D \frac{d\theta}{dt} + \frac{Mgl}{2} \sin \theta = \tau(t) \quad (2-2)$$

اگر زاویه θ را کوچک انتخاب نماییم، با استفاده از تقریب زاویه ای کوچک چنین حاصل میشود:

$$\sin \theta \cong \theta \quad (3-2)$$

$$J \frac{d^2\theta}{dt^2} + D \frac{d\theta}{dt} + \frac{Mgl}{2} \theta = \tau(t)$$

1- Selective Compliant Articulated Robot for

2- Assembly Cartesia

که با این تقریب به یک معادله خطی رسیده و با استفاده از تبدیل لاپلاس، تابع تبدیل بین ورودی (گشتاور) و خروجی (موقعیت زاویه ای) چنین حاصل میشود:

$$\frac{\theta(s)}{\tau(s)} = \frac{1}{Js^2 + Ds + \frac{Mgl}{2}} = \frac{1/J}{s^2 + \frac{D}{J}s + \frac{Mgl}{2J}} \quad (4-2)$$

برای یک پاندول مفصل دار شده افقی بدلیل اینکه اثر وزن حذف میگردد، معادلات چنین میشوند:

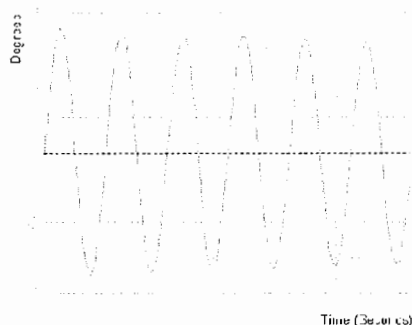
$$\tau(t) = J \frac{d^2\theta}{dt^2} + D \frac{d\theta}{dt} \quad (5-2)$$

در شکل ۳-۲ پاسخ ضربه برای یک ربات تک مفصلی با نتیجه خطی آمده است و در شکل ۴-۲ بلوک دیاگرام ربات تک مفصلی غیرخطی و پاسخ سیستم در شکل ۵-۲ نشان داده شده است.

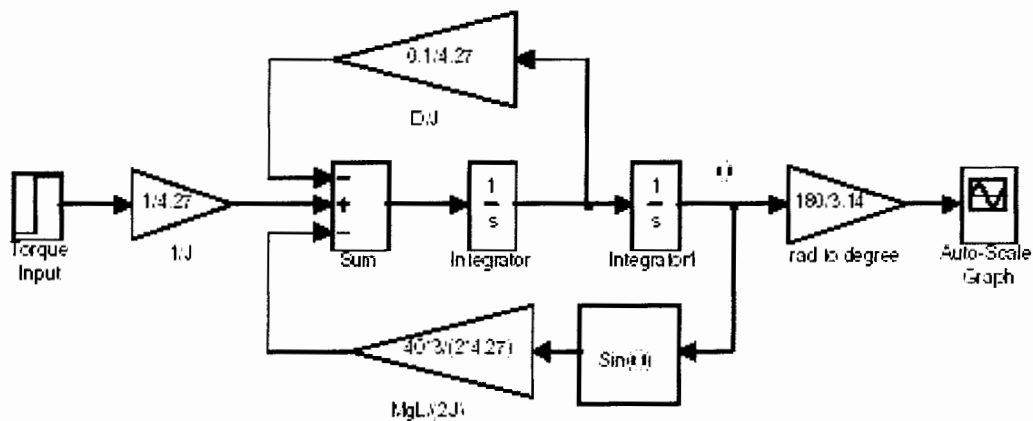
در این شبیه سازیها مقادیر پارامترها را بقرار زیر تعریف کرده ایم:

$l=3 \text{ feet}$ ، $w=Mg=40 \text{ pounds}$ ، $J=4.27 \text{ ft-lb. s}^2$ ، $D=0.01 \text{ lbs.}$ و نیز یک دامنه گشتاور

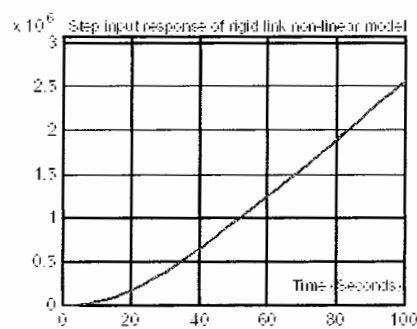
$\tau(t) = 75 \text{ ft.lb.s}$ را اعمال کرده ایم.



شکل ۳-۲- پاسخ ضربه بازوی ربات تک رابط با تقریب خطی



شکل ۲-۴- بلوک دیاگرام سیستم بازوی تک رابط غیرخطی با گشتاور



شکل ۲-۵- پاسخ بازوی تک مفصل غیرخطی هنگامی که ورودی پله $T(0)=75 \text{ ft-lb}$

۲-۲-۲- بازوی موتوردار

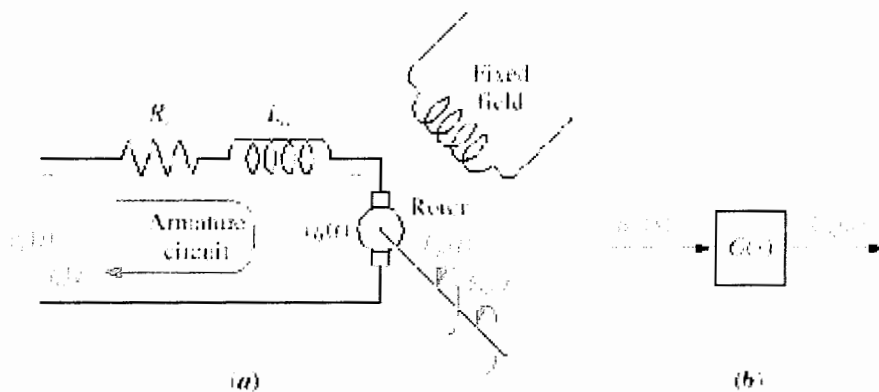
همانطور که گفتیم مدل رابط صلب در واقع ساختار اصلی یک بازوی ربات با یک درجه آزادی است. حال در این بخش می‌خواهیم به این بازوی رباتیکی، موتور را نیز اضافه کنیم. یک مدل موتور DC با کنترل آرمیچر و یک تحریک ثابت را فرض کنید. مدل الکتریکی چنین موتوری در زیر آمده است. ولتاژ آرمیچر $e_a(t)$ ، ولتاژ ایجاد شده توسط یک تقویت کننده، موتور را کنترل میکند.

موتور دارای مقاومت R_a ، اندوکتانس L_a و ثابت نیروی محرکه با برق برگشتی k_b است. ولتاژ emf

برگشتی توسط حرکت سیم بندیهای آرمیچر در ناحیه مغناطیسی ثابت، ایجاد میگردد.

ولتاژ محرکه القایی متناسب با سرعت موتور است، بدین صورت:

$$v_b(t) = k_b \frac{d\theta}{dt} \quad (۶-۲)$$



شکل ۶-۲- موتور DC با تحریک ثابت (a) دیاگرام مدار (b) بلوک دیاگرام

با اتخاذ تبدیل لاپلاس داریم:

$$v_b(s) = s k_b \theta(s)$$

معادله بخش الکتریکی مدار چنین است:

$$E_a(s) = R_a I_a(s) + L_a s I_a(s) + v_b(s) \quad (۷-۲)$$

که بصورت زیربازنویسی میشود:

$$I_a(s) = \frac{E_a(s) - k_b s \theta(s)}{L_a s + R_a} \quad (۸-۲)$$

گشتاور ایجاد شده توسط موتور، متناسب با جریان آرمیچر است،

$$\tau_m(s) = k_t I_a(s) \quad (۹-۲)$$

این گشتاور، آرمیچر و بار را تغییر میدهد. تعادل گشتاور در محور موتور، ارتباط گشتاور با زاویه را

میدهد که بصورت زیر بیان میشود:

$$\tau_m(s) = J \frac{d^2 \theta_m}{dt^2} + D \frac{d\theta_m}{dt} \quad (10-2)$$

θ_m : موقعیت زاویه ای محور موتور

J : کلیه جرمهای متصل به محور موتور

D : ضریب اصطکاک ویسکوز

با اعمال تبدیل لاپلاس داریم :

$$\tau_m(s) = Js^2 \theta_m(s) + Ds \theta_m(s) \quad (11-2)$$

با حل معادله فوق برای زاویه محور داریم:

$$\theta_m(s) = \frac{\tau_m(s)}{Js^2 + Ds} \quad (12-2)$$

اگر بین موتور و بار یک چرخ دنده موجود باشد، آنگاه زاویه تغییر داده شده توسط بار، از زاویه

تغییر داده شده توسط موتور، متفاوت خواهد بود.

روابط، وابسته به نسبت تبدیل چرخ دنده هستند .

$$s = R_m \theta_m = R_l \theta_l \quad (13-2)$$

محیط چرخ دنده موتور $2\pi.R_m$ است که N_m دندانه و محیط چرخ دنده بار $2\pi.R_l$ بوده که N_l

چرخ دنده دارد. بدین ترتیب داریم:

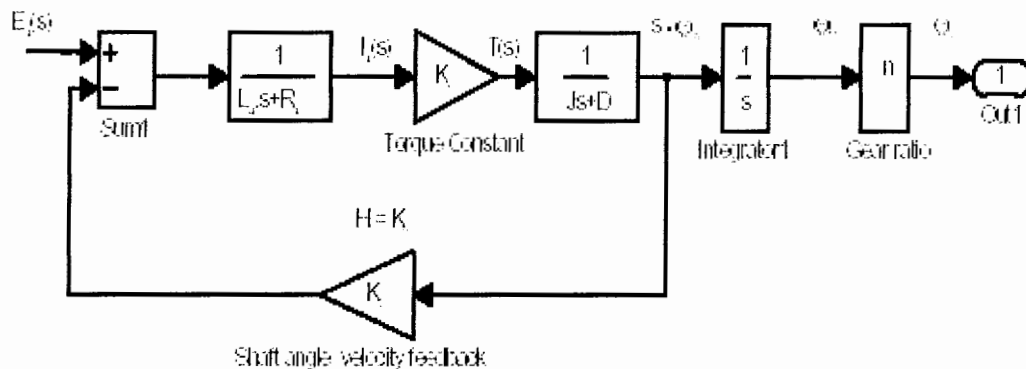
$$N_l \theta_l = N_m \theta_m$$

$$\frac{\theta_l}{\theta_m} = \frac{N_m}{N_l} = n$$

چرخ دنده همچنین ممکن است برای انعکاس مقادیر طرف بار یک چرخ دنده به طرف موتور به منظور تعادل گشتاور در سمت موتور ، استفاده شود.

$$J_{ml} = \frac{J_l}{n^2}, D_{ml} = \frac{D_l}{n^2}, K_{ml} = \frac{K_l}{n^2} \quad (۱۴-۲)$$

با توجه به روابط بالا، بلوک دیاگرام موتور DC کنترل شده آرمیچر را نظیر نمای شماتیک نشان داده شده در شکل ۷-۲ داریم:



شکل ۷-۲- بلوک دیاگرام موتور DC کنترل شده با آرمیچر

با ساده نمودن بلوک دیاگرام موتور DC کنترل شده با آرمیچر شکل ۷-۲ ، توابع تبدیل موتور بدین ترتیب بدست می آیند:

$$G(s) = \frac{\theta_l(s)}{E(s)} = \frac{K_l n}{s[(Js + D)(L_a s + R_a) + K_b K_t]} \quad (۱۵-۲)$$

$$G(s) = \frac{K_l n}{s[JL_a s^2 + (JR_a + DL_a)s + DR_a + K_b K_t]} \quad (۱۶-۲)$$

این مدل از مرتبه سوم است. حال آنکه در سروموتورها ، اغلب اندوکتانس L_a قابل چشم پوشی است و بدین ترتیب این مدل به مدلی از مرتبه دوم کاهش می یابد.

اکنون میتوان این مدل را به یک بازوی تک مفصلی اعمال نمود.

فرض کنید یک موتور DC برای راندن یک بازوی ربات بطور افقی مانند نشان داده شده در شکل

۸-۲ استفاده میشود. بازو دارای جرم $M=1kg$ و طول $L=5m$ است، فاکتور ویسکوزیته میرایی را

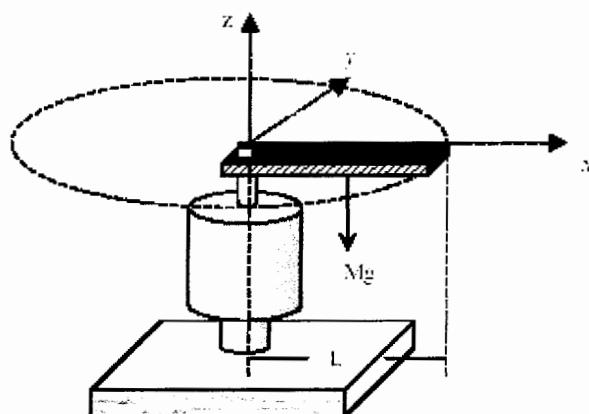
$D=0.1$ فرض میکنیم و همچنین فرض میکنیم ورودی سیستم، یک سیگنال ولتاژ با رنج $0-10v$

است. این سیگنال برای فراهم نمودن ولتاژ و جریان کنترل به موتور استفاده میشود.

پارامترهای موتور در زیر آمده اند. حال میخواهیم در اینجا یک جبران ساز طراحی کنیم بطوریکه

ولتاژ $0-10v$ بطور خطی زاویه ای تا زاویه 90 درجه پاسخ خواسته شده را با اورشوتی کمتر از 10%

و زمان نشست زیر 0.2 ثانیه و خطای حالت پایدار صفر داشته باشد.



شکل ۸-۲- بازوی تک مفصلی حرکت داده شده توسط یک موتور DC با کنترل آرمیچر

پارامترهای موتور بدین صورت هستند:

$$J_a = 0.001kg - m^2 / s^2$$

$$D_a = 0.01 N - ms / rad$$

$$R_a = 1 Ohm$$

$$L_a = 0 H$$

$$K_b = 1 V - s / rad$$

ابتدا با در نظر گرفتن سیستم بدون چرخ دنده ، لختی رابط صلب بصورت زیر محاسبه میشود:

$$J_l = \frac{ML^2}{12} = \frac{1 \times 5^2}{12} = 0.4167 \text{ kg.m}^2 \quad (17-2)$$

بر طبق انعکاس امپدانس ، کل لختی J و فاکتور میرایی کل D می باشد.

$$J = J_a + J_l = 0.001 + 0.4167 = 0.4177 \text{ kg.m}^2$$

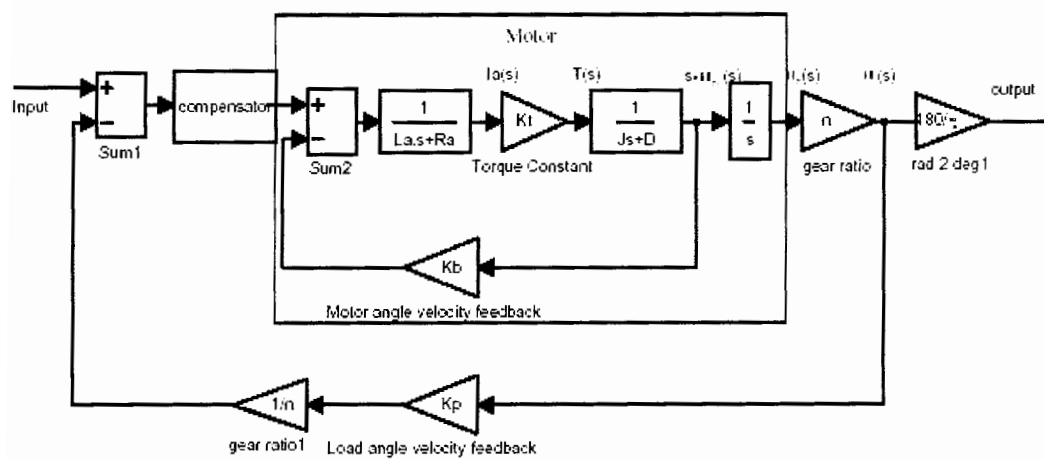
$$D = D_a + D_l = 0.01 + 0.1 = 0.11$$

با قرار دادن مقادیر معلوم در معادله (۱۶-۲) بدست می آوریم:

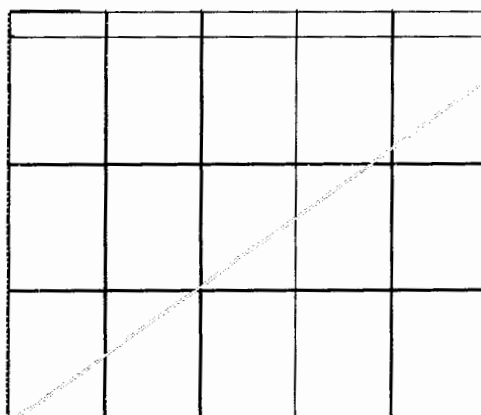
$$G(s) = \frac{\theta_l(s)}{E(s)} = \frac{1}{s[(0.4177s + 0.11)(1 + 0 \times s) + 1 \times 1]}$$

$$G(s) = \frac{1}{s(0.4177s + 1.11)} \quad (18-2)$$

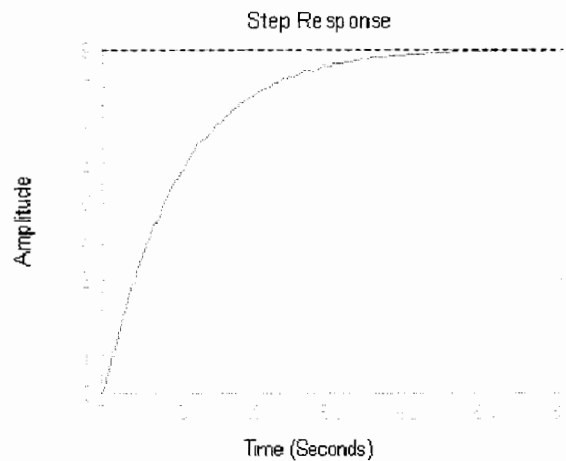
پاسخ پله این سیستم در شکل زیر آمده است.



شکل ۲-۹- مدل فیدبک سرعت و موقعیت مفصل متصل شده به موتور



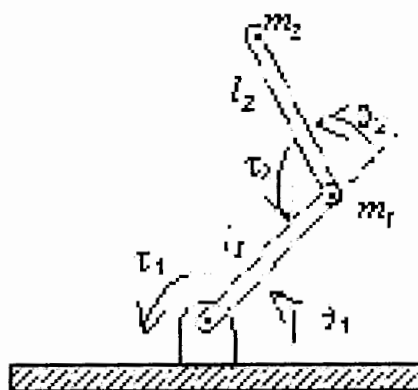
شکل ۲-۱۰- پاسخ پله بازوی ماهر تک رابط موتوردار با فیدبک سرعت



شکل ۲-۱۱- پاسخ پله سیستم با فیدبک موقعیت

در مثال فوق بازوی موتوردار را با استفاده از فیدبک موقعیت و سرعت (PD control) کنترل نمودیم و همانطور که در فصل اول اشاره شد این یکی از روشهای کنترل بازوی ماهر است. در فصل ۴ کنترل بازوی ماهر و نتایج شبیه سازی آنرا برای بازوی فوق با استفاده از روش فازی شرح میدهیم.

۲-۳- دینامیک بازوی دو رابط



شکل ۲-۱۲- شماتیک بازوی دو رابط

با فرض اینکه این مدل با نیروی وزن عمودی تحت تاثیر قرار میگیرد، با استفاده از روش فرمول سازی نیوتن-اوایلر، در هر رابط، نیروهای خارجی سرعت و شتاب معین میشوند که اینها نتیجه شده از نیروهای درونی برای معین کردن نیرو و گشتاور در هر مفصل ربات هستند و در نهایت مجموعه ای از معادلات دینامیکی حاصل میشود که گشتاور مورد نیاز در هر مفصل را با مقادیر موقعیت (θ) ، سرعت $(\dot{\theta})$ و شتاب $(\ddot{\theta})$ ، برای یک ربات با پارامترهای جرم (m_i) و طول (l_i) برای هر رابط تعیین میکنند.

معادلات (۲-۱۹) و (۲-۲۰) گشتاور تولید شده در اولین و دومین مفصل را هنگامی که جرم بطور ایده آل در انتهای هر مفصل قرار داشته باشد، تعریف میکنند [10].

$$\tau_1 = [(m_1 + m_2)l_1^2 + m_2l_2^2 + 2m_2l_1l_2 \cos \theta_2] \ddot{\theta}_1 + (m_2l_2^2 + m_2l_1l_2 \cos \theta_2) \ddot{\theta}_2 \quad (2-19)$$

$$- m_2l_1l_2 \sin \theta_2 (2\dot{\theta}_1\dot{\theta}_2 + \dot{\theta}_2^2) + (m_1 + m_2)gl_1 \cos \theta_1 + m_2gl_2 \cos(\theta_1 + \theta_2)$$

$$\tau_2 = m_2l_2(l_2 + l_1 \cos(\theta_2)) \ddot{\theta}_1 + l_2 \ddot{\theta}_2 + l_1 \sin(\theta_2) \dot{\theta}_1^2 + g \cos(\theta_1 + \theta_2) \quad (2-20)$$

ماتریسهای معادله دینامیک بازوی دو رابط عمود به مرکز ثقل چنین هستند :

$$M(\theta) = \begin{bmatrix} (m_1 + m_2)l_1^2 + m_2l_2^2 + 2m_2l_1l_2 \cos \theta_2 & m_2l_2^2 + m_2l_1l_2 \cos \theta_2 \\ m_2l_2^2 + m_2l_1l_2 \cos \theta_2 & m_2l_2^2 \end{bmatrix} \quad (2-21)$$

$$V_{cent}(\theta) = \begin{bmatrix} 0 & -m_2l_1l_2 \sin \theta_2 \\ 0 & m_2l_1l_2 \sin \theta_2 \end{bmatrix} \quad (2-22)$$

$$V_{cor}(\theta) = \begin{bmatrix} -2m_2 l_1 l_2 \sin \theta_2 \\ 0 \end{bmatrix} \quad (23-2)$$

$$G(\theta) = \begin{bmatrix} (m_1 + m_2) g l_1 \cos \theta_1 + m_2 g l_2 \cos(\theta_1 + \theta_2) \\ m_2 g l_2 \cos(\theta_1 + \theta_2) \end{bmatrix} \quad (24-2)$$

۲-۴- دینامیک بازوی سه رابط

ماتریسهای بازوی دو وجهی سه رابط، عمود بر مرکز ثقل چنین بدست می آیند [10]:

$$M(\theta) = \begin{bmatrix} m_3 l_3^2 + 2m_3 l_1 l_3 C_{23} + 2m_3 l_2 l_3 c_3 & m_3 l_3^2 + m_3 l_1 l_3 C_{23} + 2m_3 l_2 l_3 c_3 & m_3 l_3^2 + m_3 l_1 l_3 c_{23} \\ + (m_2 + m_3) l_2^2 + 2(m_2 + m_3) l_1 l_2 c_2 & + (m_2 + m_3) l_2^2 + (m_2 + m_3) l_1 l_2 c_2 & + m_3 l_2 l_3 c_3 \\ + (m_1 + m_2 + m_3) l_1^2 & & \\ m_3 l_3^2 + m_3 l_1 l_3 C_{23} + 2m_3 l_2 l_3 c_3 & m_3 l_3^2 + 2m_3 l_2 l_3 c_3 + (m_2 + m_3) l_2^2 & m_3 l_3^2 + m_3 l_2 l_3 c_3 \\ + (m_2 + m_3) l_2^2 + (m_2 + m_3) l_1 l_2 c_2 & & \\ m_3 l_3^2 + m_3 l_1 l_3 c_{23} + m_3 l_2 l_3 c_3 & m_3 l_3^2 + m_3 l_2 l_3 c_3 & m_3 l_3^2 \end{bmatrix} \quad (25-2)$$

$$V_{cent}(\theta) = \begin{bmatrix} 0 & -m_3 l_1 l_3 s_{23} & -m_3 l_1 l_3 s_{23} - (m_2 + m_3) l_1 l_2 s_2 \\ m_3 l_1 l_3 s_{23} + (m_2 + m_3) l_1 l_2 s_2 & 0 & -m_3 l_2 l_3 s_3 \\ m_3 l_1 l_3 s_{23} + m_3 l_2 l_3 s_3 & m_3 l_2 l_3 s_3 & 0 \end{bmatrix} \quad (26-2)$$

$$V_{cor}(\theta) = \begin{bmatrix} -2m_3 l_1 l_3 s_{23} - 2(m_2 + m_3) l_1 l_2 s_2 & -2m_3 l_1 l_3 s_{23} - 2m_3 l_1 l_2 s_2 & -2m_3 l_1 l_3 s_{23} - 2m_3 l_1 l_2 s_2 \\ 0 & -2m_3 l_2 l_3 s_3 & -2m_3 l_2 l_3 s_3 \\ 2m_3 l_2 l_3 s_3 & 0 & 0 \end{bmatrix} \quad (27-2)$$

$$G(\theta) = \begin{bmatrix} m_3 l_3 c_{123} + (m_2 + m_3) l_2 c_{12} + (m_1 + m_2 + m_3) l_1 c_1 \\ m_3 l_3 c_{123} + (m_2 + m_3) l_2 c_{12} \\ m_3 l_3 c_{123} \end{bmatrix} \quad (2-28)$$

۲-۵- روش گشتاور محاسبه شده برای بازوی ماهر حرکت مستقیم

گشتاور محاسبه شده یک روش معمول در کنترل بازوی ربات است. اگر مقادیر معادله (۲-۱۹)

بخوبی معلوم باشند، آنگاه قانون کنترل گشتاور محاسبه شده قابل استفاده است [10]. با جایگذاری:

$$\alpha = M(\theta)$$

$$\tau = \alpha \tau' + \beta, \beta = V_{cent}(\theta)[\dot{\theta}^2] + V_{cor}(\theta)[\dot{\theta}_i \dot{\theta}_j] + G(\theta)$$

نتیجه میشود:

$$\ddot{\theta} = \tau \quad (2-29)$$

اکنون کنترل PD ، بمنظور تعقیب مسیر میتواند بکار رود، بگونه ای که :

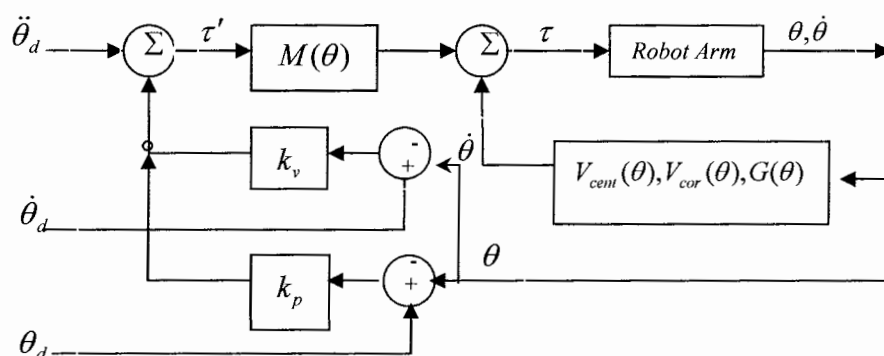
$$\tau' = \ddot{\theta}_d + k_v \dot{e} + k_p e$$

$$e = \theta_d - \theta$$

و این منجر به یک سیستم حلقه بسته با $\ddot{e} + k_v \dot{e} + k_p e = 0$ میشود. بهره های کنترل در

ماتریسهای شکل ۲-۱۳ یک سیستم گشتاور محاسبه شده کامل را با فیدبک تعقیب مسیر نشان

میدهند.



شکل ۲-۱۳- بلوک دیاگرام کنترل گشتاور محاسبه شده

۲-۶- نتیجه گیری

یکی از راههای افزایش سرعت عمل در بازوهای ماهر، استفاده از موتورهای الکتریکی حرکت مستقیم بدون چرخ دنده بجای استفاده از موتورهای الکتریکی چرخ دنده ای میباشد. موتور حرکت مستقیم با حذف خلاصی چرخ دنده ها و نیز دارا بودن حجم کوچک ، ماکزیمم سرعت موتور را افزایش میدهد. مهمترین مزیت بازوی رباتی که از این موتور استفاده میکند، ایجاد محدودیت در سرعت آن است.

در این فصل مدل دینامیکی بازوهای ماهر یک، دو و سه رابط را بیان نموده و با استفاده از یک مدل موتور کنترل شونده با جریان آرمیچر ، کنترل کننده ای طراحی نمودیم که با دریافت سیگنال ورودی و با استفاده از فیدبک موقعیت و سرعت پاسخ مطلوب را بدهد.

فصل سوّم :

بررسي سيستمهاي كنترل فائري

و

مروش جبران سانري موانري

مفهوم مدلسازی با مجموعه ها ، اصلاح کننده ها و قوانین ، نسبت به معادلات ، ایده جدیدی است که مجموعه های فازی و منطق فازی نامیده میشوند. منطق فازی در ابتدا برای کنترل استفاده میشد و به دو صورت عمده ممدانی و سوگنو نشان داده می شود.

در این فصل ابتدا مروری بر تاریخ مجموعه های فازی و منطق فازی ، فرمهای اصلی کنترل منطق فازی ، روشهای استفاده شده بمنظور گرفتن اطلاعات برای یک سیستم منطق فازی خواهیم داشت، سپس روشهای طراحی مدل فازی تاکاگی- سوگنو (مدل فازی T-S) و ساختار آن و نیز مدلهای فازی چند مرحله ای را بررسی میکنیم و یک مدل اساسی برای طراحی کنترل کننده فازی با استفاده از مفهوم جبران توزیعی موازی (PDC) ⁽¹⁾ را شرح میدهم. بعلاوه در این فصل تحلیل پایداری و مسائل طراحی کنترل را با یک نامساوی ماتریس خطی (LMI) ⁽²⁾ بیان کرده و سپس شرایط پایداری رباست را برای سیستمهای فازی دارای عدم قطعیت شرح میدهم.

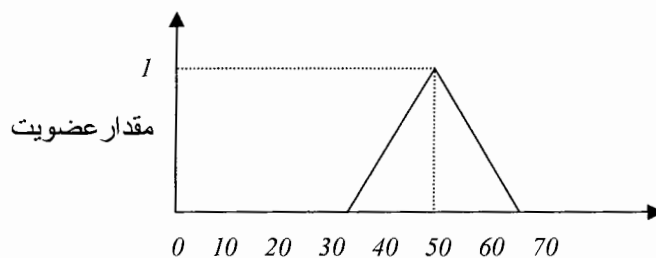
(1) *Parallel distributed compensation*

(2) *Linear matrix inequality*

۳-۱- مجموعه های فازی و منطق فازی

تئوری مجموعه فازی و استفاده از آن، نسبت به رشته های منطق، ریاضیات و مهندسی، جدید است. دکتر لطفی زاده ابتدا نظریه اش را با عنوان 'unsharp boundaries' در سال ۱۹۶۵ ارائه داد [10].

تعریف این روش و تشخیص اینکه کجا یک مجموعه به انتها میرسد و دیگری شروع میشود مشکل مینمود و به این دلیل نام فازی برای توصیف محدوده های نامعلوم مجموعه ها ابداع گردید. تئوری مجموعه فازی با تئوری مجموعه کلاسیک متفاوت است و آن با تعریف محدوده های نامعلوم مقادیر فازی با توابع عضویت مشخص میشود در حالیکه، مجموعه کلاسیک فقط یک تابع عضویت از پیش تعریف شده دارد که مقدار ۱ برای درستی و ۰ را برای نادرستی در نظر میگیرد. مجموعه های فازی، شامل عناصری با مقادیر عضویتی در محدوده $[0, 1]$ هستند، مثلاً توصیف "اعداد حدود ۵۰" یک مجموعه فازی را شامل میشود که اعداد نزدیک ۵۰ دارای بیشترین مقدار عضویت هستند و با دور شدن از ۵۰، مقادیر عضویت اعداد به صفر گرایش پیدا میکند.



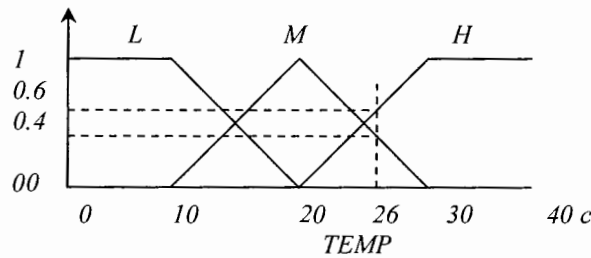
شکل ۳-۱- تابع عضویت برای اعداد حدود ۵۰

و بدین ترتیب مجموعه فازی با المانهای در فضا تعریف میشود که این المانها به مقادیر عضویتی در رنج $[0,1]$ تعلق دارند. چنین اسکلت بندی یک روش طبیعی توصیف مسائل در مواقعی است که منبع، مبهم بوده و عاری از ضوابط تعریف شده معین توابع کلاسیک باشد. لطفی زاده، منطق فازی را بمنظور شرح چگونگی استنتاج انسانها از مشاهدات واقعی شان، ارائه داد.

برای مثال استفاده از منطق فازی "هنگامیکه خانه گرم است، سرعت فن را زیاد کن" یک استنتاج فازی را معین میکند که بیان میدارد تا چه اندازه سرعت فن باید زیاد شود. توجه به این نکته جالب است که منطق فازی قطعی است و با داشتن ارتباط منطقی بین ورودی و خروجی فازی، مقادیر فازی ورودی و خروجی، روش استنتاج با مقادیر فازی، خروجی یک سیستم فازی دقیقاً معین میشود.

۳-۱-۱- توابع عضویت

در کنترل دمای اتاق، بمنظور تعریف قوانین لازم است که، اندازه های دمای اتاق و نیز زمینه های سرعت فن به مقادیر فازی و بالعکس تبدیل شوند. اینگونه تبدیلهای فازی سازی و غیر فازی سازی می نامند. فازی سازی در مرحله تبدیل، مقادیری نظیر دمای اتاق را به عناصر فازی با مقادیر عضویت تبدیل میکند و بالعکس، غیرفازی سازی در مرحله تبدیل، یک مجموعه از عناصر فازی با مقادیر عضویت را به تعدادی اعداد نظیر سرعت فن تبدیل میکند. در شکل ۳-۲ توابع عضویت برای سنجش دما و کنترل سرعت فن آمده است.



شکل ۳-۲- توابع عضویت برای سنجش دما و کنترل

در این مثال، دمای $26^{\circ}C$ معادل با مقادیر فازی زیر است:

$[Cold (0.0), warm (0.4), hot (0.6)]$.

توابع عضویت نشان داده شده در شکل بالا بصورت خطی و مثلثی آمده اند، اینها شکلهایی از توابع عضویت هستند که محاسبات را ساده میکنند.

غیرفازی سازی پیچیده تر است و روشهای مختلفی بمنظور غیرفازی سازی استفاده میشوند .

ساده ترین آنها روش مرکز سطح ⁽¹⁾ است که در شکل ۳-۳ آمده است. یک فرم کلی تر غیر فازی سازی ، روش میانگین مرکز ثقل یا غیر فازی سازی میانی ⁽²⁾ خوانده می شود. این روش ، میانگین مراکز وزنی توابع عضویت فازی خروجی را همانطور که در شکل ۳-۳ آمده است ، نشان میدهد. روش مرکز سطح اغلب از روش مرکز میانگین ساده تر است.

فرمهای دیگری نیز از غیر فازی سازی وجود دارند که وابسته به ساختار قوانین و روش استنتاج هستند.

(1) Center-of-area

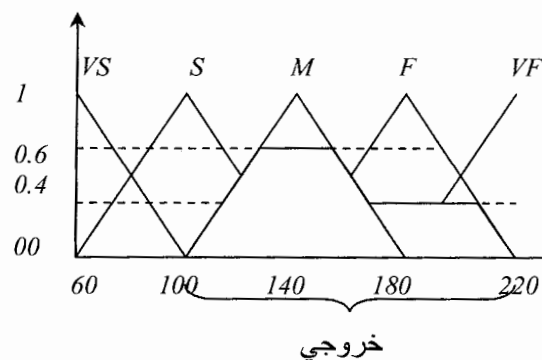
(2) weighted centroid average or centriod defuzzification

تنها مرحله باقیمانده در یک سیستم منطق فازی ، معین کردن مقادیر فازی خروجی است که

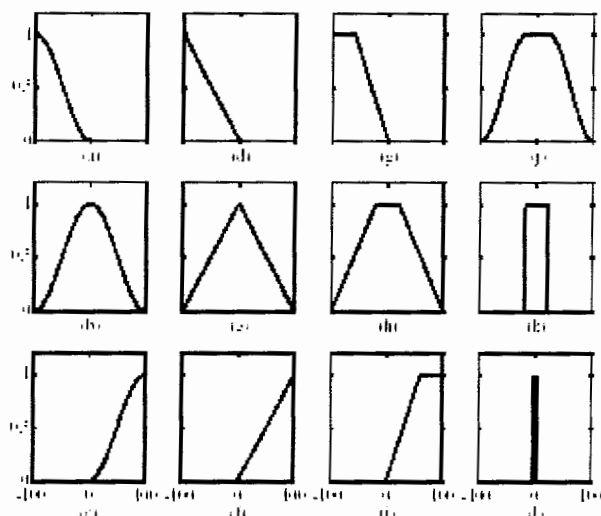
بصورت زیر حاصل میشود:

$$output = \frac{\sum_{i=1}^m \mu_i^o c_i^o}{\sum_{i=1}^m \mu_i^o} \quad (1-3)$$

که μ_i^o مقدار عضویت i امین المان فازی خروجی ، c_i^o مقدار مرکزی تابع عضویت i امین خروجی است.



شکل ۳-۳- غیر فازی سازی سرعت فن



شکل ۳-۴- مثالهایی از توابع عضویت : از بالا به پایین سمت چپ: (a) s-function (b) π _ functoin

(c) Z-function (d-f) فرم مثلثی (g-i) فرم دوزنقه ای (j) π _ functoin (k) مستطیل (l) واحد

۳-۱-۲- پایگاه قوانین و استنتاج

اصل کار با منطق فازی ، استفاده از قوانین است. این قوانین با جملات اگر- آنگاه فازی بیان میشوند و میتوانند چندین ورودی را شامل باشند، مثلا قانون "اگر گرم و مرطوب است آنگاه سرعت فن را زیاد کن". برای این قانون "اگر گرم" و "مرطوب" آنگاه "سرعت فن را بالا ببر" ، مقادیر عضویت نتیجه "بالا ببر" ، ترکیبی از مقادیر عضویت "گرم" و "مرطوب" است و این ترکیب بعنوان مینیمم مقادیر عضویت ورودی یا بعنوان حاصلضرب مقادیر عضویت ورودی تفسیر می شود. استفاده از مقدار مینیمم فقط مقادیری را میدهد که در هر دو وجود داشته باشند و با استفاده از نرم افزار کامپیوتری *MATLAB* حصول مقدار مینیمم آسان میشود.

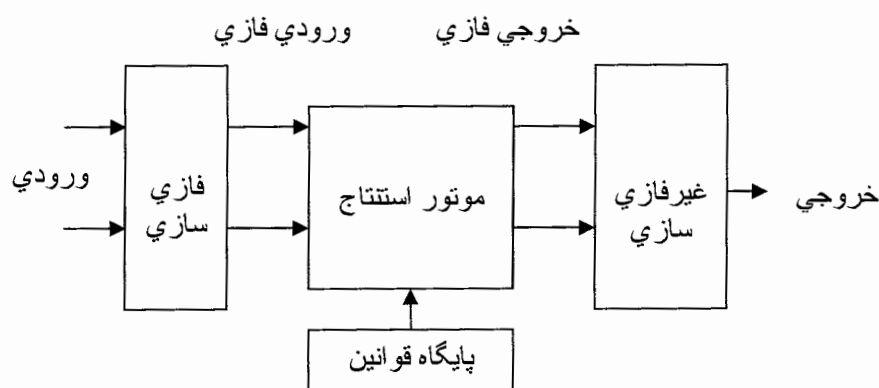
همچنین تفسیر جمله "یا" ، اشتراک کلیه مقادیر است. در این روش بر اساس اقتضای استفاده ، در اجرای کامپیوتری ، اجتماع مقادیر خروجی برای عناصر یکسان بعنوان مجموع مقادیر عضویت هر قانون گرفته میشود. مرحله نتیجه ترکیبی می تواند بصورت یک *min-max* یا بصورت یک روش ضرب - جمع نوشته شود، همانطور که در معادله (۲-۳) آمده است:

$$\mu_i^o = \max_{l=1}^M [\min(\mu_l^1, \dots, \mu_l^n)], \mu_i^o = \sum_{l=1}^M \prod (\mu_l^1, \dots, \mu_l^n) \quad (2-3)$$

μ_i^o ، عضویت i امین مقدار زبانی خروجی است و $(1, \dots, M)$ قوانین با i امین مقدار زبانی خروجی و $\mu_l^1, \dots, \mu_l^n$ مقادیر عضویت قوانین مربوطه هستند.

۳-۲- مدلسازی و کنترل فازی

از بین بسیاری از کاربردهای مجموعه ها و منطق فازی ، شاید کنترل فازی معمولترین آنها باشد. دو نوع ساختار منطق فازی وجود دارد که در کنترل و مدلسازی استفاده می شود. ساختار اول توسط ممدانی و دومی توسط سوگنو (تاکاگی - سوگنو ، تاکاگی - سوگنو - کانگ) ارائه شدند. روش ممدانی یک اجرای خیلی مستقیم از منطق فازی لطفی زاده است و روش سوگنو یک روش جهت دار عددی با قوانین اگر - آنگاه فازی است که قسمت‌های منتجه ، معادلات خطی یا اعداد ثابت هستند. مدل استفاده شده توسط ممدانی در شکل ۳-۵ آمده است و شامل چهار عنصر اصلی است: فازی سازی ، استنتاج ، پایگاه قوانین و غیرفازی سازی. این روش برای کنترل بسیاری از سیستمها از جمله شوینده های ساده، خشک کننده ها ، دوربینها و نیز سیستمهای پیچیده دیگری نظیر دیگهای بخار و طرحهای شیمیایی استفاده میشود.



شکل ۳-۵- بلوک دیاگرام منطق فازی ممدانی

روش سوگنو، در مرحله فازی سازی برای ایجاد المانهای فازی با مقادیر عضویت و در استفاده از پایگاه قانون با استنتاج، بسیار مشابه با روش ممدانی است.

اختلاف روش سوگنو با روش ممدانی در داشتن قوانین خروجی مستقل از متغیرهای ورودی بخصوص معادلات خطی مرتبه اول یا صفر، است.

تاکاگی و سوگنو روشی را برای ترکیب در یک روش فازی با تقریبهای خطی محلی از یک کنترل کننده غیرخطی بصورت ذیل ارائه دادند.

IF x_1 is C_1^l and...and x_n is C_n^l , THEN $y^l = c_0^l + c_1^l x_1 + \dots + c_n^l x_n$

$$f(x) = \frac{\sum_{l=1}^M y^l \mu^l}{\sum_{l=1}^M \mu^l}$$

که x_i مقدار i امین ورودی، C_i^l مقدار زبانی i امین ورودی است که برای l امین قانون بکار میرود.

y^l معادله نتیجه شده از قانون l ام، c_i^l ، $i=0, 1, \dots, n$ ضرایب معادله نتیجه شده l ام هستند.

$f(x)$ خروجی سیستم فازی، M تعداد قوانین و μ^l مقدار عضویت l امین برآیند است.

روش سوگنو بسادگی روش ممدانی نمی باشد بلکه برای ابزارهای محاسباتی پیچیده تر و

قدرتمندتر است. مقدار خروجی طبقه طبقه، مستقیماً از مدلهای نتیجه شده قوانین که از یک

مرحله غیرفازی سازی بدست می آید، محاسبه میشود.

در مجموع، اگر معادلات خروجی، خطی و مستقل باشند، باید ضرایب معادلاتشان از داده ورودی-

خروجی با استفاده از تقریب خطای کمترین مربعات $(LSE)^{(1)}$ بهینه شود.

(1) Least Squares Error

۳-۲-۱- یادگیری در سیستمهای منطق فازی

بعد از ایجاد یک ساختار کنترل کننده باید توابع عضویت ، قوانین فازی ، روش استنتاج و روش غیرفازی سازی را مشخص نمود ، این مقادیر ”هوشیاری“ را به سیستم می افزایند.

فقط ساختاری نظیر انسان میتواند فکر کند و مهارتها را در حین عمل و تکرار بیاموزد و هنگام طرح کنترل کننده فازی، توجه به این اصل لازم است.

پنج روش معمول بمنظور ایجاد یک سیستم فازی برای مدل سازی یا طرح کنترل کننده وجود دارد و هر کدام مزیتها و مضرتهایی دارند.

• دانش افراد خبره

از ابتدا یکی از جنبه های مهم منطق فازی ، توانایی گرفتن اطلاعات بطور مستقیم از عملگرهای انسانی، طراحها یا مدیران بود. متخصصان سیستم و افراد خبره در این زمینه برای گرفتن و تبادل اطلاعاتشان از عملکردهای کنترل سیستم ، مذاکره میکردند.

بسیاری از کارهای موفقیت آمیز منطق فازی توسط متخصصان حاصل شده است. مثالهایی در این زمینه شامل عملکرد یک دستگاه کاغذ و روزنامه ، یک طرح بازیابی آب تلف شده و کوره سیمان هستند. از مزیتهای این سلسله روشها میتوان به سادگی و سرعت آنها اشاره نمود.

• داده نمونه گیری شده

اولین روش عددی بکار گرفته شده، داده نمونه برداری شده از سیستم را برای گسترش قوانین و توابع عضویت کنترل کننده مدل فازی استفاده میکرد.

در این روش، گروه کوچکی از اطلاعات برای معین کردن قوانین و توابع عضویت تقریبی بکار میروند. روشهای مستعمل برای تعریف قوانین، منطق فازی با استفاده از ماکزیمم $FLM^{(1)}$ و منطق فازی با استفاده از میانگین وزن شده (2) نامیده میشود. روش داده نمونه برداری شده برای استفاده در سیستمهایی مطرح میشود که در آنها بدست آوردن داده مشکل است، نظیر یک طرح واکنش شیمیایی یا وقتی که داده برای کلیه شرایط عمل در دسترس نباشد.

روش داده نمونه برداری شده نسبتاً ساده است و هنگام کار با این روش، نیاز به هیچ تخصصی نیست.

• خوشه سازی (3)

خوشه سازی یک روش عددی است که دسته های اطلاعات را گروه بندی میکند. یک شکل معمول از خوشه بندی، c -means نامیده میشود که برای پیدا کردن بهترین گروه بندیهای اطلاعات از توابع عضویت، استفاده میشود.

معادله زیر یک تابع ارزشی را که با استفاده از مراکز خوشه ها مینیمم شده است، نشان میدهد.

$$J = \sum_{i=1}^M \sum_{j=1}^R (\mu_{ij})^m (x^i - v^j)^2 \quad (3-3)$$

J ارزشی است که باید مینیمم شود، m تعداد نقاط داده، R تعداد قوانین، μ عضویت i امین نقطه

(1) Fuzzy Logic Maximum

(2) Fuzzy Logic Weighted

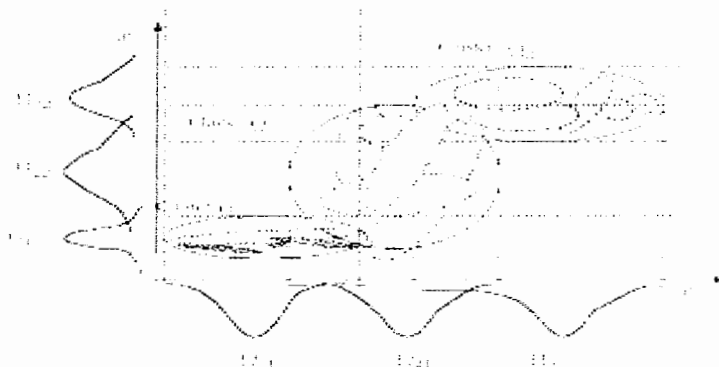
(3) Clustering

داده در σ امین خوشه، m یک پارامتر طراحی، x یک نقطه داده و v مجموعه مراکز خوشه هاست. روش دیگر خوشه بندی، نزدیکترین همسایگی نامیده میشود. این روش بسته ترین برآیند قانون را برای جفت مجموعه داده جاری پیاده میکند و اگر هیچ قانونی به اندازه کافی بسته نباشد، یک قانون جدید ایجاد میشود.

معادله (۴-۳) برای خوشه بندی نزدیکترین همسایگی استفاده میشود. هر دو روش بالا یعنی نزدیکترین همسایگی و c -means تا زمانی که قابل درک بوده و براحتی توسط الگوریتمهای کامپیوتری قابل اجرا باشند، استفاده میشوند.

$$f(x|\theta) = \frac{\sum_{i=1}^R A_i \prod_{j=1}^n \exp\left(-\left(\frac{x_j - v_j^i}{2\sigma}\right)^2\right)}{\sum_{i=1}^R B_i \prod_{j=1}^n \exp\left(-\left(\frac{x_j - v_j^i}{2\sigma}\right)^2\right)} \quad (4-3)$$

بعنوان روش عددی، اینها تکرارهای بسیاری را برای جایی، بمنظور رسیدن به حل بهینه نیاز دارند و به پارامترهای خوشه حساس هستند. θ پارامترهای توابع عضویت، x, R, v نظیر معادله خوشه سازی بالا هستند. σ پهنای توابع عضویت و A, B وزنهاى نزدیکترین الگوریتم همسایه هستند.



شکل ۴-۳ - دسته بندی فازی با خوشه سازی

• تقریب کمترین مربع خطا از داده ورودی - خروجی

یکی دیگر از معمولترین روشهای خوشه بندی برای مدل‌های منطق فازی ، روش کمترین خطای مربع نامیده میشود. LSE یک مجموعه بزرگ از داده های ورودی تطبیق شده (هماهنگ) با خروجی خواسته شده در هر مجموعه از اطلاعات است.

معادلات برآیند قانون ، کمترین مجموع را برای مربع مقادیر داده خواسته شده منهای مقادیر داده واقعی میدهند. با ترکیب یک دسته از معادلات خطی، حل LSE ، مقادیر داده خواسته شده را با

معکوس کردن ضرایب قانون برای مقدار واقعی ضرب میکند.

$$e = y_d - y_a$$

برای مجموعه داده های بزرگ، معکوس کردن، مقدار زیادی از حافظه را نیاز دارد و ممکن است بسیار سخت حل شود. یک روش افزایشی با استفاده از ماتریس کاواریانس برای کاهش نیازمندیهای حافظه و پیدا کردن معکوس ساختگی، گسترش یافته است. فرمولهای پیاپی در رابطه (۵-۳) آمده اند.

$$S_{i+1} = S_i - \frac{S_i a_{i+1}^T S_i}{1 + a_{i+1}^T S_i a_{i+1}} \quad (5-3)$$

$$X_{i+1} = X_i + S_{i+1} a_{i+1} (b_{i+1}^T - a_{i+1}^T X_i)$$

S ماتریس کاواریانس، $i=0, 1, \dots, p-1, p$ تعداد نقاط دنباله، X_i برآورد مربعات کمترین تکرار، a_i^T یک سطر از ماتریس عضویت ورودی است و b_i^T مقدار خروجی مربوط به a_i^T است.

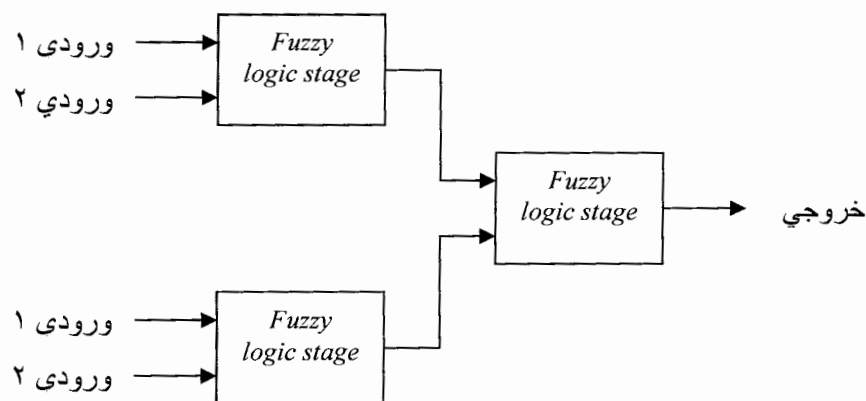
شرایط اولیه ، $X_0 = 0, S_0 = \gamma.I$ است که γ یک عدد مثبت بزرگ و I یک ماتریس شناسایی با اندازه مساوی با تعداد قوانین است.

LSE، یک مدل خیلی صحیح از اطلاعات را میدهد. سه اشکال *LSE* این است که آن به یک سیستم خطی از معادلات نیاز دارد (محدودیت در کاربرد)، و دیگر آنکه *LSE* بطور ناگهانی نمیتواند در یک سیستم قرار گیرد و آن فقط پارامترهای برآیند و سیستم فازی را هماهنگ میکند.

۳-۳- وضعیتهای چندمرحله ای

هرچند مدل‌های منطق فازی موفقیت‌های بسیاری هم در زمینه عملی و هم تئوری داشته اند، ولی موفقیت‌ها وابسته به سیستم‌های ساده یا اجزاء ساده سیستم‌های بزرگتر بوده اند. برای پاسخگو بودن به اجزای پیچیده تر سیستم هدف، مدل منطق فازی بطور توصیفی گسترش می یابد و هر پارامتر دیگری برای مدل شدن نیاز به بعد دیگری در پایگاه قوانین دارد و این، تعداد قوانین را با تعداد توابع عضویت در بعد جدید ضرب میکند که باعث ایجاد مشکل ابعادی میشود و مدل منطق فازی را تحت تاثیر قرار میدهد. از نتایج سوء این مسئله، افزایش نیازمندیهای حافظه و نیز افزایش زمان محاسبه است.

برای غلبه کردن بر این مسئله، مدل‌های چندمرحله ای بررسی میشوند. یک مدل چند مرحله ای شامل چندین مرحله سری و موازی است که خروجی برخی مراحل، ورودی دیگر مراحل است. شکل ۳-۷ یک ساختار چند مرحله ای شاخص را نشان میدهد.



شکل ۳-۷- بلوک دیاگرام چند مرحله ای

همینکه بعدهای جدید افزوده میشوند ، آنها با سیستم فاز ی مرتبه پایین ترکیب میشوند. بنابراین، بعدهای اضافه شده یک رشد خطی نسبت به رشد توصیفی در اندازه مدل ایجاد میکنند.

این راه حل ، تعداد قوانین را کم میکند. برای مثال یک کنترل کننده *PID* فاز ی تک مرحله ای با پنج مجموعه فاز ی، در هر کدام سه بعد ورودی ، با ۳۴۳ قانون طراحی میشود درحالیکه یک مدل چندمرحله ای با پایگاه قانون دو بعدی با تعداد ورودیهای یکسان از مجموعه های فاز ی فقط ۹۸ قانون دارد. پژوهشهای اخیر نشان میدهد انسانها تمایل به انتخاب برخی از اطلاعات، قبل از ترکیب آن با برخی یا همه داده های ورودی به عنوان پیش پردازش دارند. بنابراین انسانها یک روش چندمرحله ای طبیعی را برای فهم و کنترل سیستمهای پیچیده استفاده میکنند.

مدل قوانین یک طرح تک مرحله ای از یک سیستم چند ورودی (>2) بصورت معادله (۳-۶) خواهد بود، در حالیکه معادله چندمرحله ای آن بصورت معادله (۳-۷) آمده است.

$$IF \tilde{u}_1 \text{ is } \tilde{A}_1^l \text{ and } \tilde{u}_2 \text{ is } \tilde{A}_2^k \text{ and}, \dots, \text{and } \tilde{u}_n \text{ is } \tilde{A}_n^l \text{ THEN } \tilde{y}_q \text{ is } \tilde{B}_q^p \quad (۳-۶)$$

که $\tilde{u}_i$ یک متغیر زبانی است.

با دوباره نویسی سیستم تک مرحله ای فوق بصورت یک سیستم چند مرحله ای ، قوانین فازی بفرم معادله زیر خواهند بود . این ساختار منطقی در بلوک دیاگرام شکل (۷-۳) مشهود است.

IF $\tilde{u}_1$ is $\tilde{A}_1^j$ and $\tilde{u}_2$ is $\tilde{A}_2^k$ THEN $\tilde{c}_1$ is $\tilde{C}_1^l$

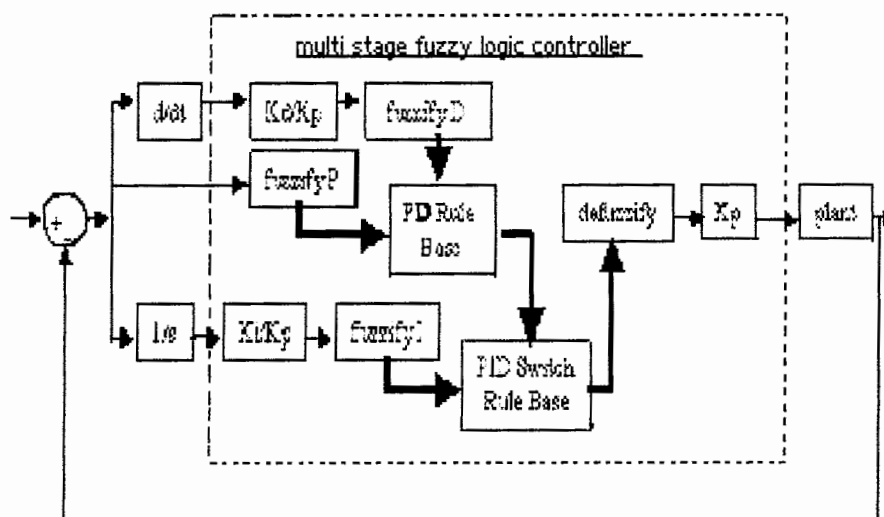
IF $\tilde{u}_3$ is $\tilde{A}_3^j$ and $\tilde{u}_4$ is $\tilde{A}_4^k$ THEN $\tilde{c}_2$ is $\tilde{C}_2^m$ (۷-۳)

IF $\tilde{c}_1$ is $\tilde{C}_1^l$ and $\tilde{c}_2$ is $\tilde{C}_2^m$ THEN $\tilde{y}_q$ is $\tilde{B}_q^p$

این ساختار چند مرحله ای درک بهتری از کنترل کننده میدهد.

در زیر یک ساختار چند مرحله ای از یک سیستم کنترل PID آورده ایم.

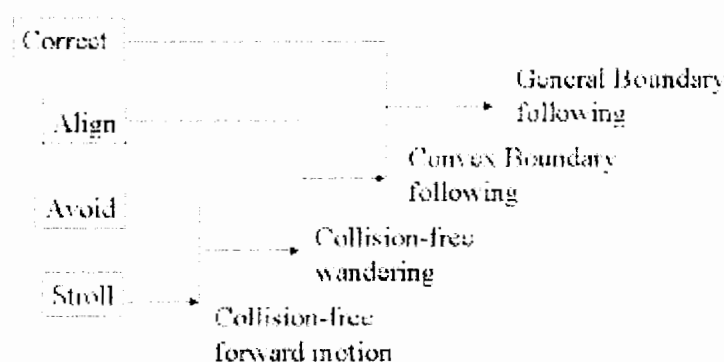
البته باید یاد آوری شود که یک ساختار چندمرحله ای ممکن است دقت یک مدل تک مرحله ای با بعد بالا را نداشته باشد.



شکل ۸-۳- کنترل کننده PID فازی چند مرحله ای

۳-۳-۱- کنترل بر اساس عملکرد

مدلسازی چندمرحله ای بسیار مشابه با کنترل بر اساس عملکرد است. کنترل بر اساس عملکرد با موفقیت برای عملکرد رباتها در محیطهای پیچیده با کنترل کننده های نسبتا ساده مورد استفاده قرار گرفته است. شکل ۳-۹ مثالی از یک کنترل کننده حرکتی بر اساس عملکرد را میدهد که شباهت آن با کنترل کننده چندمرحله ای بخوبی دیده میشود.



شکل ۳-۹- بلوک دیاگرام کنترل بر اساس عملکرد

در این سیستم مثال ، *stroll* کمترین حد کنترل هست و تا حدیکه هیچ سنسور دیگری فعال نباشد ادامه می یابد. اگر یک مانع شناسایی شود ، سطح *Avoid* به ربات فرمان میدهد و کنترل را به سطح پایینتر برمیگرداند تا هنگامیکه مانع دیگری شناسایی نشود. در این روش بسیاری از رفتارها میتوانند ایجاد شوند و به کنترل کننده ربات افزوده شوند بدون اینکه تغییری در کنترل کننده طراحی شده قبلی ایجاد شود.

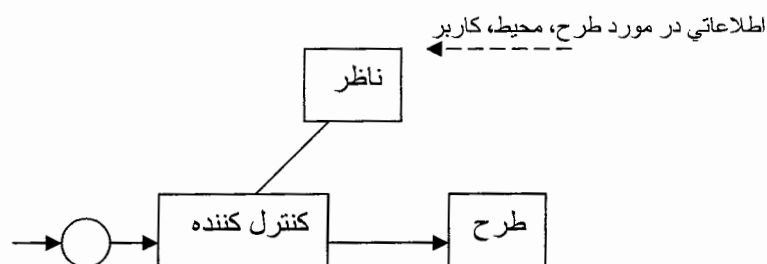
یک کنترل کننده PID فازی چند مرحله ای همانطور که در شکل ۳-۹ آمده است میتواند بعنوان کنترل کننده بر اساس عملکرد در نظر گرفته شود. این سیستم یک کنترل کننده انتگرالی بعنوان کمترین مقدار برای نگهداشتن خطای کنترل حالت پایدار صفر دارد.

کنترل کننده های بر اساس عملکرد عموماً ضروری نیستند، یک مثال ساده از این کنترل کننده را میتوان حرکت ربات به سمت چپ، مثال زد که در هر مانعی اگر مسیر صحیح بسمت راست باشد به همان مسیر میرود اما این روش بسیار طولانی تر از حل بهینه است. بعبارت دیگر کنترل کننده های براساس عملکرد، هم رسیدن به هدف و هم سادگی طراحی را تحت تاثیر قرار میدهند.

۳-۳-۲- کنترل ترتیبی

کنترل کننده های ترتیبی معمولاً کنترل کننده ناظر^(۱) نامیده میشوند.

بلوک دیاگرام کنترل ترتیبی در زیر آمده است.

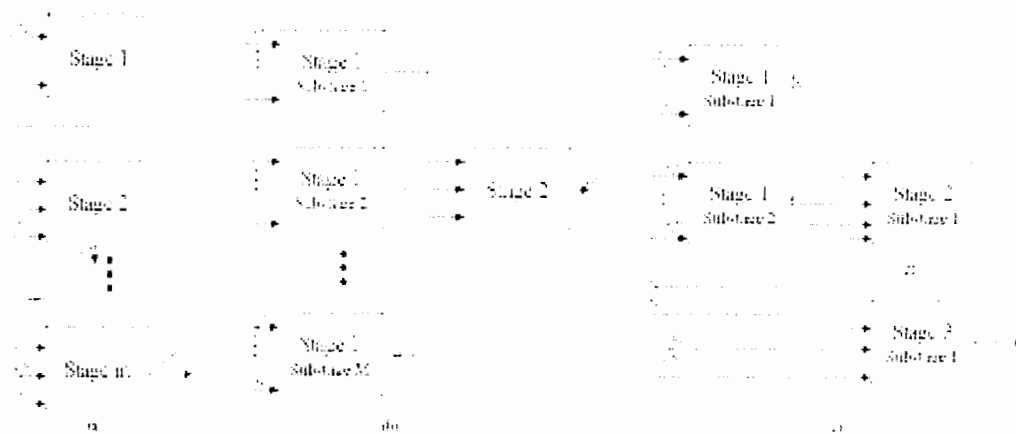


شکل ۳-۱۰- کنترل کننده ترتیبی

1- Supervisory controllers.

دو نوع اصلی از ساخت چند مرحله ایها وجود دارد، افزایشی^(۱) و مجموع^(۲).

شکل ۳-۱۱ دو نوع اصلی را با یک ساختار هیبرید نشان میدهد.

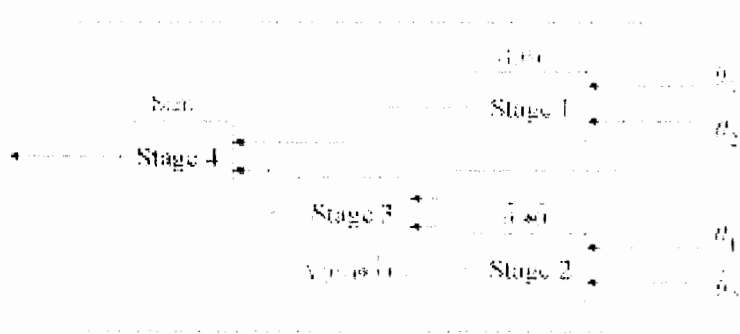


شکل ۳-۱۱- ساختار چند مرحله ایها (a) افزایشی (b) مجموع (c) هیبرید

و در آخر نیز یک نما از مدل منطق فازی چند مرحله ای ساده، برای گشتاور محاسبه شده بفرم

$$G(\theta) + v_{cor}(\theta) * \dot{\theta}_1 \dot{\theta}_2$$

می آوریم.



شکل ۳-۱۲- مدل چند مرحله ای ساده برای ساختار گشتاور محاسبه شده

1- Incremental

2- Aggregated

۳-۴- مدل فازی تاکاگی - سوگنو

مدل فازی تاکاگی - سوگنو ($T-S$) توسط مجموعه ای از قوانین اگر- آنگاه فازی توصیف میشود که روابط ورودی- خروجی خطی محلی یک سیستم غیرخطی را نشان میدهد. بسیاری از سیستمهای دینامیک غیرخطی میتوانند با مدل فازی تاکاگی - سوگنو نمایش یابند. قانون i ام مدل فازی $T-S$ بصورت زیر است:

$\langle \text{Plant Rule } i \rangle$

IF $z_1(t)$ is M_{i1} And...And $z_p(t)$ is M_{ip}

$$\text{THEN } \begin{cases} \dot{x}(t) = A_i x(t) + B_i u(t), & i = 1, 2, \dots, r \\ y(t) = C_i x(t), \end{cases} \quad (۸-۳)$$

که $\dot{x}(t) = x(t)$ برای سیستمهای CFS و $\dot{x}(t) = x(t+1)$ برای سیستمهای DFS میباشد. M_{ij} مجموعه فازی و r تعداد قوانین اگر- آنگاه فازی است. $x(t) \in R^n$ بردار حالت، $u(t) \in R^m$ بردار ورودی، $y(t) \in R^q$ بردار خروجی، $A_i \in R^{n \times n}$ و $B_i \in R^{n \times m}$. $z_1(t), \dots, z_p(t)$ متغیرهای از پیش فرض شده اند. هر معادله پیوسته خطی، که با $A_i x(t) + B_i u(t)$ نمایش داده شود، یک زیرسیستم نامیده میشود.

خروجی نهایی سیستمهای فازی در همه زمانها، برای هر جفت $(x(t), u(t))$ بصورت زیر خواهد بود.

$$\dot{x}(t) = \sum_{i=1}^r h_i(z(t)) [A_i x(t) + B_i u(t)] \quad (۹-۳)$$

$$z(t) = [z_1(t) \ z_2(t) \ \dots \ z_p(t)] \quad (۱۰-۳)$$

$$y(t) = \frac{\sum_{i=1}^r w_i(z(t)) C_i x(t)}{\sum_{i=1}^r w_i(z(t))} = \sum_{i=1}^r h_i(z(t)) C_i x(t) \quad (۱۱-۳)$$

$$w_i(z(t)) = \prod_{j=1}^p M_{ij}(z_j(t)), \quad (12-3)$$

$M_{ij}(z_j(t))$ درجه عضویت $z_j(t)$ در M_{ij} است.

مثال ۱- در یک DFS فرض کنید، $p = n$,

$$z_1(t) = x(t), z_2(t) = x(t-1), \dots, z_n(t) = x(t-n+1)$$

آنگاه، قوانین فازی بصورت زیر خواهند بود،

$\langle \text{Plant Rule } i \rangle$

IF $x(t)$ is M_{i1} And...And $x(t-n+1)$ is M_{in} ,

$$\text{THEN} \begin{cases} x(t+1) = A_i x(t) + B_i u(t), \\ y(t) = C_i x(t). \end{cases} \quad i = 1, 2, \dots, r$$

$$x(t) = [x(t) \ x(t-1) \ \dots x(t-n+1)]^T$$

۳-۴-۱- ساختار مدل فازی

شکل ۳-۱۳ طریقه طراحی کنترل کننده فازی را نشان میدهد. بمنظور طراحی یک کنترل کننده

فازی برای یک سیستم غیرخطی، به یک مدل فازی تاکاگی-سوگنو نیاز داریم. در این بخش به

ساخت چنین مدل فازی می پردازیم.

در کل دوروش برای ساخت مدلهای فازی وجود دارد:

۱- شناسایی با استفاده از داده ورودی-خروجی (مدلسازی فازی)

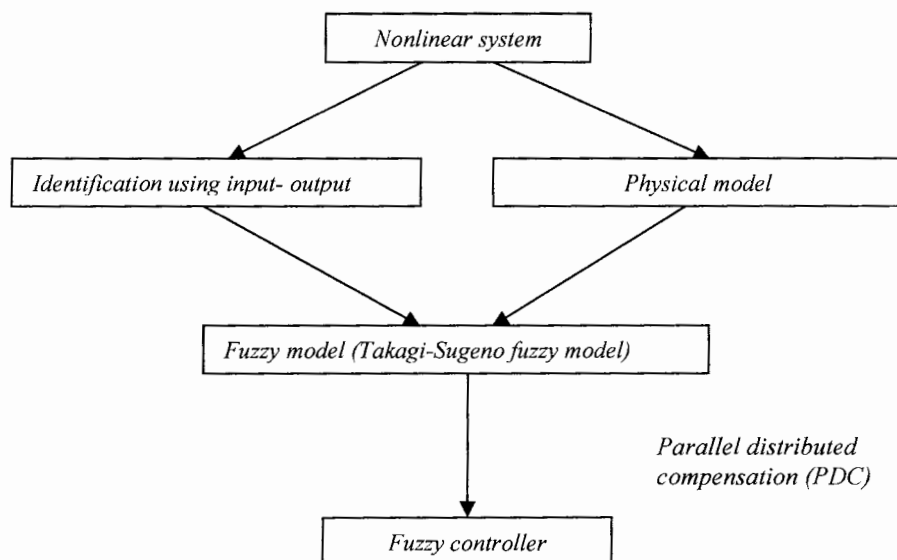
۲- نتیجه گیری از معادلات سیستم غیرخطی مفروض.

مقاله بسیار با ارزشی در زمینه مدلسازی سیستمهای فازی با استفاده از روش داده ورودی - خروجی توسط تاکاگی - سوگنو و بدنبال آن توسط کانگ ارائه شده است [20]. این روش اساساً شامل دو قسمت است:

شناسایی ساختار و شناسایی پارامتر.

شناسایی با استفاده از مدل فازی، در طرحهایی مناسب است که نمایش آنها با مدلهای تحلیلی یا مدلهای فیزیکی، مشکل بوده و یا امکان پذیر نباشد.

در برخی موارد، روش دوم که مدلهای دینامیک غیرخطی را به مدل فازی تبدیل میکند، مناسبتر است. در این بخش روی روش دوم تمرکز میکنیم. این روش، از تقریب محلی غیرخطی ناحیه ای یا ترکیبی از آنها برای ساخت مدلهای فازی استفاده میکند.



شکل ۳-۱۳- اصول طراحی کنترل فازی

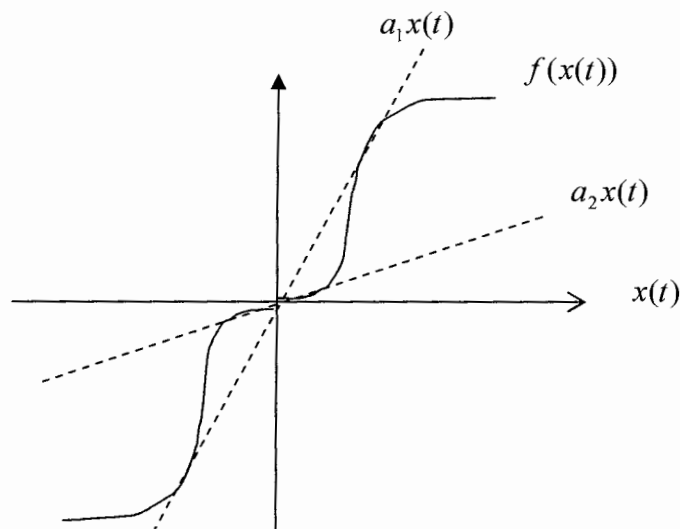
۳-۴-۲- غیر خطی ناحیه ای

غیر خطی ناحیه ای بر ایده زیر بنا شده است :

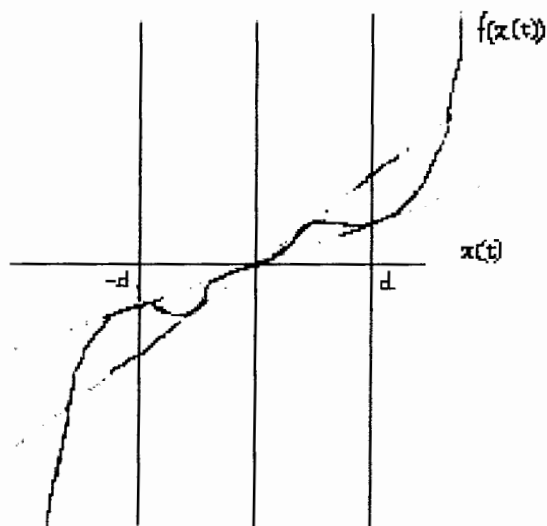
سیستم غیر خطی ساده $\dot{x}(t) = f(x(t))$ را که در آن $f(0) = 0$ ، را در نظر بگیرید . هدف یافتن ناحیه سراسری (global sector) است که $\dot{x}(t) = f(x(t)) \in [a_1 \ a_2]x(t)$ باشد.

شکل ۳-۱۴، روش غیرخطی ناحیه ای سراسری را شرح میدهد . این روش ، یک ساختار مدل فازی دقیق را تضمین میکند .

گاهی یافتن نواحی سراسری برای سیستمهای غیرخطی ، مشکل است . در اینگونه موارد میتوان غیرخطی را بصورت جزء محلی در نظر گرفت ، بدلیل این که متغیرهای سیستمهای فیزیکی اغلب محدودند .



شکل ۳-۱۴- غیر خطی ناحیه ای گلوبال



شکل ۳-۱۵- غیر خطی ناحیه ای محلی

شکل ۳-۱۵، غیر خطی ناحیه ای محلی را شرح میدهد، دو خط نواحی محلی بین $-d < x(t) < d$ را نشان میدهند. مدل فازی، دقیقاً سیستم غیر خطی را در ناحیه محلی که $-d < x(t) < d$ است، نشان میدهد.

مثال ۲، مراحل ساخت مدل‌های فازی را بیان میکند.

مثال ۲- سیستم غیر خطی زیر را در نظر بگیرید :

$$\begin{pmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \end{pmatrix} = \begin{pmatrix} -x_1(t) + x_1(t)x_2^3(t) \\ -x_2(t) + (3+x_2(t))x_1^3(t) \end{pmatrix}. \quad (۱۳-۳)$$

برای سادگی فرض میکنیم، $x_1(t) \in [-1,1]$ و $x_2(t) \in [-1,1]$.

در واقع هربازه ای را میتوان برای $x_1(t)$ و $x_2(t)$ ، بمنظور ساخت یک مدل فازی در نظر گرفت.

معادله (۱۳-۳) بصورت زیر نوشته میشود:

$$\dot{x}(t) = \begin{bmatrix} -1 & x_1(t)x_2^2(t) \\ (3+x_2(t))x_1^2(t) & -1 \end{bmatrix} x(t),$$

$x(t) = [x_1(t) \ x_2(t)]^T$ و $x_1(t)x_2^2(t)$ و $(3+x_2(t))x_1^2(t)$ جملات غیر خطی هستند.

جملات غیرخطی را به اینصورت تعریف میکنیم: $z_1(t) \equiv x_1(t)x_2^2(t)$ و

$$z_2(t) \equiv (3+x_2(t))x_1^2(t).$$

بنابراین خواهیم داشت:

$$\dot{x}(t) = \begin{bmatrix} -1 & z_1(t) \\ z_2(t) & -1 \end{bmatrix} x(t).$$

مینیمم و ماکزیمم مقادیر $z_1(t)$ و $z_2(t)$ را در بازه $x_1(t) \in [-1,1]$ و $x_2(t) \in [-1,1]$ بدست

می آوریم :

$$\max_{x_1(t), x_2(t)} z_1(t) = 1, \quad \min_{x_1(t), x_2(t)} z_1(t) = -1,$$

$$\max_{x_1(t), x_2(t)} z_2(t) = 4, \quad \min_{x_1(t), x_2(t)} z_2(t) = 0.$$

از مقادیر ماکزیمم و مینیمم ، $z_1(t)$ و $z_2(t)$ به اینصورت نوشته میشوند :

$$z_1(t) = x_1(t)x_2^2(t) = M_1(z_1(t)).1 + M_2(z_1(t)).(-1)$$

$$z_2(t) = (3+x_2(t))x_1^2(t) = N_1(z_2(t)).4 + N_2(z_2(t)).0,$$

$$M_1(z_1(t)) + M_2(z_1(t)) = 1, \quad \text{که:}$$

$$N_1(z_2(t)) + N_2(z_2(t)) = 1.$$

بدین ترتیب با استفاده از روابط فوق توابع عضویت بصورت زیر محاسبه میشوند:

$$\begin{aligned} M_1(z_1(t)) &= \frac{z_1(t)+1}{2}, & M_2(z_1(t)) &= \frac{1-z_1(t)}{2}, \\ N_1(z_2(t)) &= \frac{z_2(t)}{4}, & N_2(z_2(t)) &= \frac{4-z_2(t)}{4}. \end{aligned}$$

توابع عضویت را بترتیب "مثبت"، "منفی"، "بزرگ"، "کوچک" تعریف میکنیم.

بدین ترتیب سیستم غیرخطی (۳-۱۳) با مدل فازی زیر نمایش می یابد.

Model Rule 1:

IF $z_1(t)$ *is "Positive" and* $z_2(t)$ *is "Big"*

THEN $\dot{x}(t) = A_1 x(t)$.

Model Rule 2:

IF $z_1(t)$ *is "Positive" and* $z_2(t)$ *is "Small"*

THEN $\dot{x}(t) = A_2 x(t)$.

Model Rule 3:

IF $z_1(t)$ *is "Negative" and* $z_2(t)$ *is "Big"*

THEN $\dot{x}(t) = A_3 x(t)$.

Model Rule 4:

IF $z_1(t)$ *is "Negative" and* $z_2(t)$ *is "Small"*

THEN $\dot{x}(t) = A_4 x(t)$.

$$A_1 = \begin{bmatrix} -1 & 1 \\ 4 & -1 \end{bmatrix}, \quad A_2 = \begin{bmatrix} -1 & 1 \\ 0 & -1 \end{bmatrix},$$

که:

$$A_3 = \begin{bmatrix} -1 & -1 \\ 4 & -1 \end{bmatrix}, \quad A_4 = \begin{bmatrix} -1 & -1 \\ 0 & -1 \end{bmatrix}.$$

شکل ۳-۱۶ و ۳-۱۷ توابع عضویت را نشان می‌دهند. غیرفازی سازی به این صورت انجام میشود،

$$\dot{x}(t) = \sum_{i=1}^4 h_i(z(t)) A_i x(t),$$

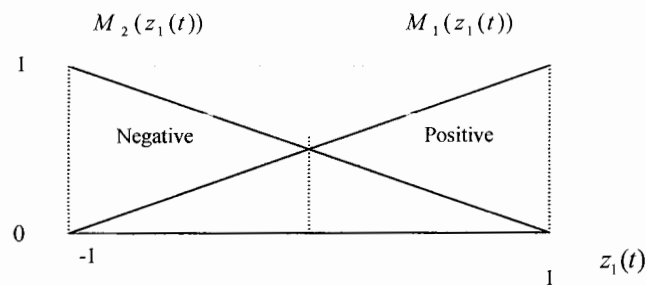
$$h_1(z(t)) = M_1(z_1(t)) \times N_1(z_2(t)),$$

$$h_2(z(t)) = M_1(z_1(t)) \times N_2(z_2(t)),$$

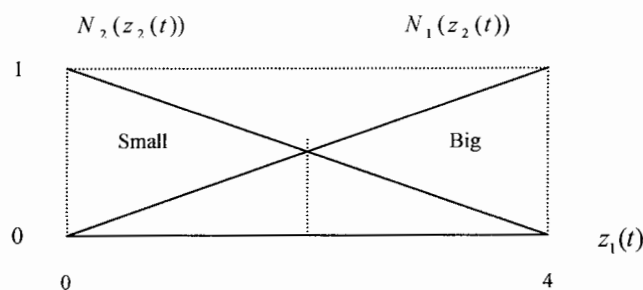
$$h_3(z(t)) = M_2(z_1(t)) \times N_1(z_2(t)),$$

$$h_4(z(t)) = M_2(z_1(t)) \times N_2(z_2(t)).$$

این مدل فازی، دقیقاً سیستم غیرخطی را در بازه $[-1,1] \times [-1,1]$ در فضای $x_1 - x_2$ نشان میدهد.



شکل ۳-۱۶- توابع عضویت $M_1(z_1(t))$ و $M_2(z_1(t))$



شکل ۳-۱۷- توابع عضویت $N_1(z_2(t))$ و $N_2(z_2(t))$

مثال ۳- معادلات حرکت پاندول معکوس چنین است :

$$\dot{x}_1(t) = x_2(t),$$

$$\dot{x}_2(t) = \frac{g \sin(x_1(t)) - amlx_2^2(t) \sin(2x_1(t)) / 2 - a \cos(x_1(t))u(t)}{4l/3 - aml \cos^2(x_1(t))} \quad (۱۴-۳)$$

$x_1(t)$ ، زاویه (برحسب رادیان) پاندول از خط عمود و $x_2(t)$ سرعت زاویه ای است ؛

$g = 9.8m/s^2$ ثابت مرکز ثقل ، m جرم پاندول ، M جرم وسیله (برحسب نیوتن) است.

معادله (۱۴-۳) را بصورت زیر بازنویسی میکنیم :

$$a = \frac{1}{(m+M)}$$

$$\dot{x}_2(t) = \frac{1}{4l/3 - aml \cos^2(x_1(t))} \times \left(g \sin(x_1(t)) - \frac{aml(x_2(t) \sin(2x_1(t))x_2(t))}{2} - a \cos(x_1(t))u(t) \right).$$

$$z_1(t) = \frac{1}{4l/3 - aml \cos^2(x_1(t))}, \quad \text{و تعریف میکنیم :}$$

$$z_2(t) = \sin(x_1(t)),$$

$$z_3(t) = x_2(t) \sin(2x_1(t)),$$

$$z_4(t) = \cos(x_1(t)).$$

$$\text{که } x_1(t) \in \left[-\frac{\pi}{2}, \frac{\pi}{2}\right], x_2(t) \in [-\alpha, \alpha]$$

بمنظور طراحی کنترل کننده فازی برای پاندول معکوس نیاز به $2^4 = 16$ قانون داریم.

تبصره ۵ - قبل از اعمال روش غیرخطی ناحیه ای ، اغلب روش بالا برای ساده کردن مدلهای

غیرخطی اصلی تا حد ممکن، روش خوبی است . این مرحله در کاربردهای عملی بسیار مهم است،

زیرا اغلب منجر به کاهش تعداد قوانین و سهولت در تجزیه و تحلیل سیستمهای کنترل میگردد.

۳-۴-۳- تقریب محلی در فضاهای مجزای فازی

راه دیگر بدست آوردن مدل‌های فازی $T-S$ ، اصطلاحاً تقریب محلی در فضاهای مجزای فازی گفته میشود. اساس این روش، تقریب جملات غیرخطی با جملات خطی تفکیک شده است. مثلاً مدل فازی پاندول معکوس اشاره شده در مثال قبل دارای ۱۶ قانون خواهد بود، اما در مثال زیر برای آن یک مدل ۲ قانونی با استفاده از روش تقریب محلی بدست می آوریم.

تعداد قوانین مدل، مستقیماً بستگی به تحلیل و طراحی شرایط LMI دارد به این دلیل که تعداد قوانین برای کل سیستم کنترل، ترکیب قوانین مدل و قوانین کنترل است.

نکته - همانطور که گفتیم تکنیک تقریب محلی به کاهش تعداد قوانین مدل فازی منجر میشود. اما طراحی قوانین کنترل مدل فازی بطور تقریبی، ممکن است پایداری سیستم‌های کنترل غیرخطی اصلی را تحت هر شرط کنترلی، تضمین نکنند. به همین خاطر بحث سیستم‌های فازی $Robust$ پیش خواهد آمد.

مثال ۴- مدل فازی ساخته شده پاندول معکوس در مثال قبل دارای ۱۶ قانون خواهد بود، اما در اینجا سعی میکنیم یک مدل فازی ۲ قانونی، با استفاده از تقریب محلی در فضاهای مجزای فازی بدست آوریم. در واقع، مدل نتیجه شده یک تقریب از سیستم اصلی است. یک کنترل کننده فازی که با استفاده از مدل فازی ۲ قانونی طراحی میشود همانند سیستم پاندول غیرخطی اصلی عمل میکند.

وقتی $x_1(t)$ نزدیک صفر باشد، معادلات غیرخطی بصورت زیر ساده میشوند،

$$\begin{aligned}\dot{x}_1(t) &= x_2(t), \\ \dot{x}_2(t) &= \frac{gx_1(t) - au(t)}{\frac{4l}{3} - aml}.\end{aligned}\tag{۱۹-۳}$$

و وقتی $x_1(t)$ نزدیک $\pm\pi/2$ باشد، معادلات غیر خطی به اینصورت ساده میشوند،

$$\begin{aligned}\dot{x}_1(t) &= x_2(t), \\ \dot{x}_2(t) &= \frac{2gx_1(t)/\pi - a\beta u(t)}{4l/3 - aml\beta^2}, \text{ where } \beta = \cos(88^\circ)\end{aligned}\quad (20-3)$$

توجه کنید که معادلات بالا اکنون سیستمهای خطی هستند و با استفاده از زیرسیستمهای خطی،

قواعد فازی زیر بدست می آیند،

Model Rule 1:

IF $x_1(t)$ is about 0

THEN $\dot{x}(t) = A_1X(t) + B_1u(t)$.

Model Rule 2:

IF $x_1(t)$ is about $\pm\pi/2$ ($|x_1| < \pi/2$),

THEN $\dot{x}(t) = A_2X(t) + B_2u(t)$.

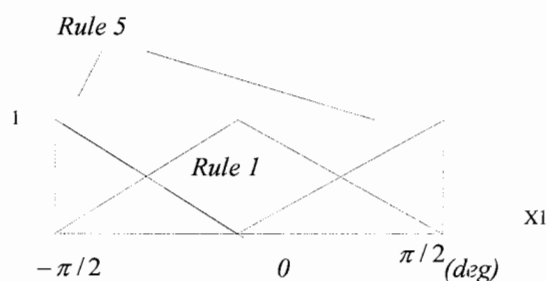
$$A_1 = \begin{bmatrix} 0 & 1 \\ \frac{g}{4l/3 - aml} & 0 \end{bmatrix}, \quad B_1 = \begin{bmatrix} 0 \\ -\frac{a}{4l/3 - aml} \end{bmatrix},$$

$$A_2 = \begin{bmatrix} 0 & 1 \\ \frac{2g}{\pi(4l/3 - aml\beta^2)} & 0 \end{bmatrix}, \quad B_2 = \begin{bmatrix} 0 \\ -\frac{a\beta}{4l/3 - aml\beta^2} \end{bmatrix}. \beta = \cos(88^\circ)$$

توابع عضویت قوانین ۱ و ۲ در شکل ۳-۲۳ آمده است. در مدل فازی مثال ۳، توابع عضویت با

همان دقت دینامیکهای غیرخطی بدست می آیند.

در مثال فوق توابع عضویت با استفاده از نوع مثلثی تعریف میشوند و همانطور که میدانیم مدل فازی، یک مدل تقریبی است.



شکل ۳-۱۸- توابع عضویت مدل ۲ قانونی

۳-۵- جبران توزیعی موازی

جبران توزیعی موازی (PDC)، توسط کانگ و سوگنو آغاز شد. اما در این روش طراحی، پایداری سیستمهای کنترل مطرح نمیشود. PDC روشی برای طراحی یک کنترل کننده فازی با استفاده از یک مدل فازی $T-S$ است، پس قبل از استفاده از روش PDC لازم است که موضوع کنترل (سیستم غیرخطی) با یک مدل فازی $T-S$ بیان شود. یاد آور میشویم که بسیاری از سیستمهای واقعی نظیر سیستمهای مکانیکی و سیستمهای آشوب با مدل فازی $T-S$ قابل بیان هستند. هر قانون کنترل در روش PDC از قانون مربوط به مدل فازی $T-S$ بدست می آید.

برای مدلهای فازی (۳-۸) کنترل کننده فازی زیر را با (PDC) می سازیم:

IF $z_1(t)$ is M_{i1} and...and $z_p(t)$ is M_{ip} ,

قانون کنترلی i ام:

THEN $u(t) = -F_i x(t)$, $i = 1, 2, \dots, r$

قوانین کنترل فازی بدست آمده، یک کنترل کننده خطی دارند، اما کنترل کننده های دیگری نیز می توان استفاده کرد ، مثلا کنترل کننده های فیدبک خروجی و کنترل کننده های فیدبک خروجی دینامیکی بجای کنترل کننده های فیدبک حالت .

درکل ، کنترل کننده فازی بصورت زیر نشان داده میشود :

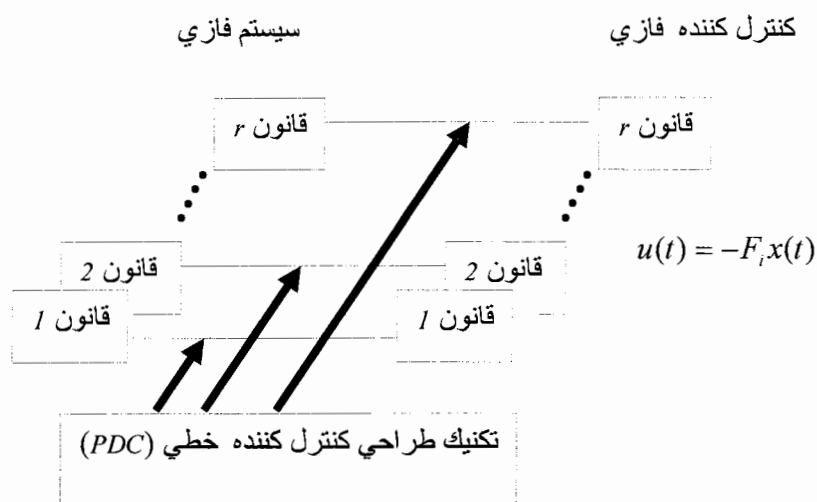
$$u(t) = - \frac{\sum_{i=1}^r w_i(z(t)) F_i x(t)}{\sum_{i=1}^r w_i(z(t))} = - \sum_{i=1}^r h_i(z(t)) F_i x(t). \quad (۲۱-۳)$$

طراحی کنترل کننده فازی یعنی ، معین کردن بهره های فیدبک محلی F_i .

PDC یک روش ساده و طبیعی، برای بکار گیری سیستمهای کنترل غیرخطی را میدهد.

بهره های فیدبک محلی باید با استفاده از شرایط طراحی جهانی معین میشوند و لازم آن این است

که عملکرد کنترل پذیری و پایداری جهانی تضمین شود.



شکل ۳-۱۹- طراحی جبران ساز توزیعی موازی PDC [2]

مثال ۵- برای موضوع کنترلی نظیر مدل قوانین مثال ۱ قوانین کنترل زیر با استفاده از PDC

ساخته میشوند،

قانون کنترلی i ام:

IF $x(t)$ is M_{i1} and...and $x(t-n+1)$ is M_{in} ,

THEN $u(t) = -F_i x(t)$, $i = 1, 2, \dots, r$.

سیستم حلقه باز زمان گسسته زیر را در نظر بگیرید،

$$x(t+1) = \sum_{i=1}^r h_i(z(t)) A_i x(t). \quad (22-3)$$

شرط کافی برای پایداری سیستم بصورت قضیه زیر بیان میشود.

قضیه ۱- تعادل سیستم فازی (۲۲-۳) پایدار مجانبی جهانی است اگر یک ماتریس مثبت معین

متعارفی p وجود داشته باشد، بطوریکه:

$$A_i^T p A_i - p < 0 \quad i = 1, 2, \dots, r \quad (23-3)$$

و یک p متعارفی باید برای همه زیرسیستمها وجود داشته باشد. این قضیه ، وقتی $r = 1$ باشد،

به قضیه پایداری لیاپانف برای سیستمهای خطی (زمان گسسته)، تقلیل می یابد. شرط پایداری

قضیه ۱ با استفاده از تابع مرتبه دوم $v(x(t)) = x(t)^T p x(t)$ اثبات میشود. اگر $p > 0$ وجود داشته

باشد ، سیستم (۲۲-۳) را پایدار مرتبه ۲ و $v(x(t))$ یک تابع لیاپانف درجه دوم نامیده میشود.

برای چک کردن پایداری ، نیاز به یافتن یک p متعارفی است (اغلب اوقات یک روش سعی و خطا

استفاده میشود) و یا باید ثابت شود که چنین p ی وجود ندارد و این یک مسئله LMI است.

مسائل LMI ، به عنوان ابزاری قدرتمند و در دسترس در مقاله های ریاضی ، قابل حل هستند.

سوآلی که پیش می آید این است که ، سیستم (۳-۲۲) اگر همه زیرسیستمهایش یعنی همه A_i ها پایدار باشند، پایدار است؟ جواب در کل ، خیر است ، همانطور که در مثال زیر توضیح داده میشود.

مثال ۶- سیستم فازی زیر را در نظر بگیرید:

Rule 1:

IF $x_2(t)$ is M_1 (e.g. Small),

THEN $x(t+1) = A_1 x(t)$.

Rule 2:

IF $x_2(t)$ is M_2 (e.g. Big),

THEN $x(t+1) = A_2 x(t)$.

$$A_1 = \begin{bmatrix} 1 & -0.5 \\ 1 & 0 \end{bmatrix}, A_2 = \begin{bmatrix} -1 & -0.5 \\ 1 & 0 \end{bmatrix}, x(t) = \begin{bmatrix} x_1(t) & x_2(t) \end{bmatrix}^T, \text{ در اینجا,}$$

شکل ۳-۲۵ توابع عضویت M_1 و M_2 را نشان میدهد . وقتی A_1 و A_2 پایدار باشند، زیرسیستمهای

خطی هم پایدارند اما به ازای برخی شرایط اولیه ، مثلاً در شکل ۳-۲۶ به ازای شرط اولیه

فازی $x = [0.90 \quad -0.7]^T$ سیستم فازی ناپایدار میگردد . یاد آوری میشود که خطی سازی سیستم

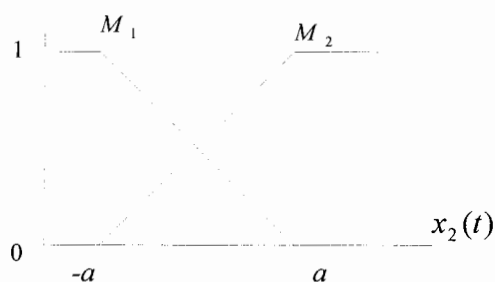
فازی حول نقطه 0 پایدار است . واضح است که وقتی سیستم ناپایدار باشد، $p > 0$ متعارفی وجود

نخواهد داشت .

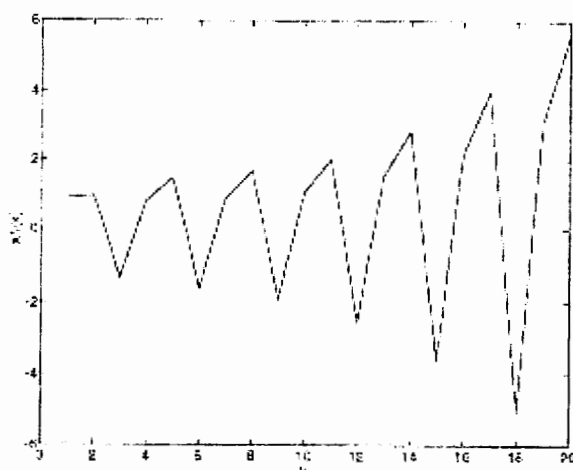
حال یک سوال مطرح میشود و آن اینکه، به ازای چه شرط اولیه ای سیستم فازی پایدار و

یا (ناپایدار) است؟

با مطالعه ناحیه جذب در مبدا میتوان به این سوال پاسخ داد.



شکل ۳-۲۰- توابع عضویت مثال 6



شکل ۳-۲۱- پاسخ مثال 6 ($a=1$)

شکل ۳-۲۲- نواحی جذب را برای $a=1$ نمایش میدهد. ناحیه مشکی، نواحی پایدار را نشان

میدهد. حال جالب است بدانیم چگونه نواحی جذب با تغییر توابع عضویت تغییر میکنند. مثلاً در

این شکل با تغییر a ناحیه جذب چگونه تغییر میکند؟

شکلهای b و c و d نواحی جذب را برای مقادیر مختلف a نشان میدهد. مشاهده میشود که با

کاهش (افزایش) a ، ناحیه جذب کوچکتر (بزرگتر) میشود. بدین ترتیب ناحیه جذب سیستم فازی

وابسته به توابع عضویت است.

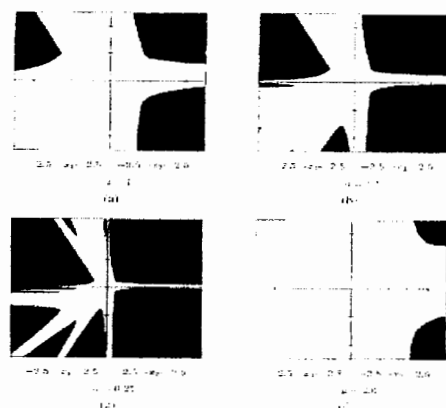
در مثال بالا وقتی $a = \infty$ ، سیستم فازی خواهد شد:

$$x(t+1) = \frac{A_1 + A_2}{2} x(t),$$

که پایدار مجانبی گلوبال و خطی است .

تعیین اصولی قوانین و توابع عضویتی که خواص پایداری را برآورده کنند مبحث جالبی است که

در ادامه با عنوان طراحی کنترل کننده با جبران توزیعی موازی شرح داده میشود.



شکل ۳-۲۲- نواحی جذب مثال 6

۳-۶- طراحی کنترل LMI

دربخش قبل ، روش و مفهوم اساسی جبران توزیعی موازی و اصول طراحی LMI را مطرح کردیم،

در این بخش به جزئیات تحلیل و طراحی با LMI ها میپردازیم . در اینجا نشان داده میشود که

بسیاری از مشخصات عملکردی کنترل نظیر شرایط پایداری، شرایط $Decay Rate$ و حذف اغتشاش

در سیستمهای فازی گسسته و پیوسته با جملات LMI قابل بیان هستند.

۳-۶-۱- شرایط پایداری

در سال ۱۹۹۰ موضوع پایداری سیستمهای کنترل فازی، در چهارچوب پایداری سیستمهای غیرخطی بررسی شد، در اینجا برخی نتایج مقدماتی در پایداری سیستمهای کنترل فازی مطرح میشود و سپس قضیه شرایط پایداری سیستمهای حلقه باز را بیان میکنند.

قضیه ۲- نقطه تعادل سیستم فازی پیوسته (۳-۸) با $u(t)=0$ ، پایدار مجانبی جهانی است، اگر

یک ماتریس معین مثبت متعارفی p وجود داشته باشد، بطوریکه:

$$A_i^T p + p A_i < 0, i = 1, 2, \dots, r \quad (۳-۲۴)$$

ماتریس متعارفی p باید برای همه زیر سیستمها وجود داشته باشد.

سپس قضیه زیر را برای پایداری سیستم حلقه بسته بدست می آوریم [20]. با جایگذاری

(۳-۸) در (۳-۲۱) به رابطه زیر میرسیم:

CFS:

$$\dot{x}(t) = \sum_{i=1}^r \sum_{j=1}^r h_i(z(t)) h_j(z(t)) \{A_i - B_i F_j\} x(t) \quad (۳-۲۵)$$

با تعریف، $G_{ij} = A_i - B_i F_j$ معادله (۳-۲۵) بصورت زیر نوشته میشود،

CFS:

$$\dot{x}(t) = \sum_{i=1}^r h_i(z(t)) h_j(z(t)) G_{ij} x(t) + 2 \sum_{i=1}^r \sum_{i < j} h_i(z(t)) h_j(z(t)) \left\{ \frac{G_{ij} + G_{ji}}{2} \right\} x(t) \quad (۳-۲۶)$$

قضیه ۳- نقطه تعادل سیستم کنترل فازی پیوسته (۳-۲۶) پایدار مجانبی جهانی است، اگر یک

ماتریس معین مثبت متعارفی p وجود داشته باشد، بطوریکه، $G_{ij} = A_i - B_i F_j$

$$G_{ii}^T p + p G_{ii} < 0$$

$$\left(\frac{G_{ij} + G_{ji}}{2}\right)^T p + p \left(\frac{G_{ij} + G_{ji}}{2}\right) \leq 0 \quad i < j \quad s.t. h_i \cap h_j \neq \emptyset \quad (27-3)$$

اثبات آن بطور مستقیم از قضیه ۲ نتیجه میشود.

برای تحلیل فازی سیستمهای فازی گسسته به [4] مراجعه شود.

طراحی کنترل فازی، یعنی معین کردن F_j ها ($j=1, 2, \dots, r$) بطوریکه شرایط قضیه ۳ توسط

یک ماتریس مثبت معین متعارفی p برآورده شود.

مسئله یافتن p متعارفی میتواند بطور عددی حل شود به این معنی که شرایط پایداری قضیه فوق

را می توان به صورت LMI بیان نمود، مثلا برای چک کردن شرایط پایداری قضیه ۳ ما به یک p

نیاز داریم که LMI های زیر را برآورده کند،

$$p > 0 \quad G_{ii}^T p + p G_{ii} < 0$$

$$\left(\frac{G_{ij} + G_{ji}}{2}\right)^T p + p \left(\frac{G_{ij} + G_{ji}}{2}\right) \leq 0 \quad , \quad i < j \quad s.t. h_i \cap h_j \neq \emptyset$$

یا اینکه باید ثابت شود که چنین p ای وجود ندارد. این یک مسئله غیرخطی است که می تواند

بطور عددی حل شود.

۳-۶-۲- شرایط پایداری آرام شده

تا کنون گفتیم که تحلیل پایداری سیستم کنترل فازی به مسئله یافتن p متعارفی، کاهش مییابد.

اما اگر r تعداد قوانین اگر- آنگاه بزرگ باشد، یافتن p متعارفی که شرایط قضیه ۳ را برآورده کند،

مشکل خواهد بود.

بدین منظور شرایط پایداری جدیدی را با استفاده از ساده سازی قضیه ۳ ارائه می نماییم.

قضیه ۴- فرض کنید تعداد قوانین فعال برای همه t ها کمترین مساوی s باشد ، $1 < s \leq r$ [20] .

نقطه تعادل سیستم کنترل فازی حلقه بسته پایدار مجانبی جهانی است ، اگر یک ماتریس معین

مثبت متعارفی p و یک ماتریس نیمه معین مثبت متعارفی Q وجود داشته باشد بطوری که،

$$G_{ii}^T p + p G_{ii} + (s-1)Q < 0 \quad (28-3)$$

$$\left(\frac{G_{ij} + G_{ji}}{2}\right)^T p + p \left(\frac{G_{ij} + G_{ji}}{2}\right) - Q \leq 0, \quad i < j \text{ s.t. } h_i \cap h_j \neq \emptyset, s > 1 \quad (29-3)$$

نکته : در نتایج قضایای ۳ و ۴ فرض میشود وزن $h_i(z(t))$ هر قانون به ازای همه t ها ، در کنترل

کننده فازی و مدل فازی یکسان هستند و در صورت عدم صدق این نکته بجای قضایای ۳ و ۴ از

$$G_{ij}^T p + p G_{ji} < 0$$

شرایط زیر استفاده میشود:

وقتی $Q=0$ باشد ، قضیه ۴ به قضیه ۳ کاهش می یابد.

۳-۶-۳- طراحی کنترل کننده پایدار

در این بخش به طراحی کنترل کننده های فازی پایدار برای سیستمهای پیوسته می پردازیم.

ابتدا یک مسئله طراحی را بررسی میکنیم که کنترل کننده فازی پایداری را با بهره های فیدبک

F_i برای CFS با استفاده از قضیه ۳ معین میکند.

با ضرب کردن طرفین (۲۷-۳) در p^{-1} و تعریف متغیر جدید $X = p^{-1}$ شرایط فوق را به این

صورت باز نویسی میکنیم،

$$-XA_i^T - A_i X + XF_i^T B_i^T + B_i F_i X > 0,$$

$$-XA_i^T - A_i X - XA_j^T - A_j X + XF_j^T B_i^T + B_i F_j X + XF_i^T B_j^T + B_j F_i X \geq 0.$$

با تعریف $M_i = F_i X$ برای $X > 0$ ، داریم ، $F_i = M_i X^{-1}$. و با جایگذاری در نامساویهای بالا داریم :

$$-XA_i^T - A_iX + M_i^T B_i^T + B_iM_i > 0,$$

$$-XA_i^T - A_iX - XA_j^T - A_jX + M_j^T B_i^T + B_iM_j + M_i^T B_j^T + B_jM_i \geq 0.$$

با استفاده از این شرایط LMI ، یک مسئله طراحی کنترل کننده فازی پایدار را تعریف میکنیم.

با یافتن $X > 0$ و M_i ($i=1,2,\dots,r$) هایی که شرایط زیر را برآورده کنند داریم:

$$-XA_i^T - A_iX + M_i^T B_i^T + B_iM_i > 0, \quad (30-3)$$

$$-XA_i^T - A_iX - XA_j^T - A_jX + M_j^T B_i^T + B_iM_j + M_i^T B_j^T + B_jM_i \geq 0,$$

$$X = p^{-1}, M_i = F_i X \quad i < j \quad st. h_i \cap h_j \neq \emptyset \quad (31-3)$$

شرایط بالا LMI هایی با نسبت متغیرهای X و M_i هستند. باید یک ماتریس معین مثبت X و

M_i که LMI ها را برآورده کند، بیابیم و یا ثابت کنیم که چنین X و M_i ی وجود ندارد.

بهره های فیدبک F_i و یک p متعارفی بصورت زیر بدست می آیند:

$$p = X^{-1}, F_i = M_i X^{-1} \quad (32-3)$$

با جایگذاری، این نامساویهای غیر خطی فوق (محدب) اکنون میتوانند با استفاده از مکمل $schur$

به LMI ها تبدیل شوند.

LMI های نتیجه شده در X و F_i چنین هستند:

$$\begin{bmatrix} X & XA_i^T - M_i^T B_i^T \\ A_iX - B_iM_i & X \end{bmatrix} > 0$$

$$\begin{bmatrix} X & \left\{ \frac{A_iX + A_jX - B_iM_j - B_jM_i}{2} \right\}^T \\ \left\{ \frac{A_iX + A_jX - B_iM_j - B_jM_i}{2} \right\} & X \end{bmatrix} \geq 0$$

۳-۶-۴- طراحی کنترل کننده فازی با استفاده از شرایط پایداری آرام شده

$X > 0$, $Y \geq 0$ و M_i ($i=1, 2 \dots r$) هایی را که شرایط زیر را برآورده کنند، می یابیم:

$$-XA_i^T - A_iX + M_i^T B_i^T + B_iM_i - (s-1)Y > 0, \quad (33-3)$$

$$2Y - XA_i^T - A_iX - XA_j^T - A_jX + M_j^T B_i^T + B_iM_j + M_i^T B_j^T + B_jM_i \geq 0,$$

$$X = p^{-1}, M_i = F_iX, Y = XQX, i < j \text{ s.t. } h_i \cap h_j \neq \phi,$$

شرایط بالا LMI هایی با نسبت متغیرهای Y, X و M_i هستند. باید یک ماتریس معین مثبت X و یک ماتریس نیمه معین مثبت Y و M_i که LMI ها را برآورده کند، بیابیم و یا ثابت کنیم که چنین M_i, X, Y وجود ندارد. بهره های فیدبک F_i ، یک p متعارفی و یک Q متعارفی بدین صورت از حل M_i و Y, X بدست می آیند: $p = X^{-1}, F_i = M_iX^{-1}, Q = pYp$.

۳-۷- سرعت نزولی

سرعت پاسخ وابسته به سرعت نزولی است، حال می خواهیم به طراحی کنترل کننده فازی با $Decay Rate$ پردازیم.

۳-۷-۱- طراحی کنترل کننده فازی $Decay Rate$

شرط $\dot{V}(x(t)) \leq -2\alpha V(x(t))$ برای همه مسیرها و کل i ها معادل است با:

$$G_{ii}^T P + PG_{ii} + 2\alpha P \leq 0 \quad (34-3)$$

$$\left(\frac{G_{ij} + G_{ji}}{2}\right)^T P + P\left(\frac{G_{ij} + G_{ji}}{2}\right) + 2\alpha P \leq 0 \quad (35-3)$$

برای $i < j$ ، بغیر از (i, j) هایی که $h_i(z(t)), h_j(z(t)) = 0, \forall t, \alpha > 0$.

بنابراین بزرگترین باند پایینی که در *Decay Rate* میتوان با استفاده از یک تابع مرتبه دوم لیاپانف

پیدا کرد ، از مسئله مینیمم سازی مقدار ویژه کلی در X و α بدین صورت حاصل میشود:

maximize α

$$X, M_1, \dots, M_r$$

$$-XA_i^T - A_iX + M_i^T B_i^T + B_iM_i - 2\alpha X > 0, \quad X > 0 \quad (36-3)$$

$$-XA_i^T - A_iX - XA_j^T - A_jX + M_j^T B_i^T + B_iM_j + M_i^T B_j^T + B_jM_i - 4\alpha X \geq 0,$$

$$i < j \text{ s.t. } h_i \cap h_j \neq \emptyset, \quad X = P^{-1}, M_i = F_i X.$$

۳-۷-۲- شرایط پایداری آرام شده با استفاده از کنترل کننده *Decay Rate*

شرط $\dot{V}(x(t)) \leq -2\alpha V(x(t))$ برای همه مسیرها معادل است با :

$$G_{ii}^T P + PG_{ii} + (s-1)Q + 2\alpha P < 0,$$

$$\left(\frac{G_{ij} + G_{ji}}{2}\right)^T P + P\left(\frac{G_{ij} + G_{ji}}{2}\right) - Q + 2\alpha P \leq 0, \quad \text{و}$$

$$i < j \text{ s.t. } h_i \cap h_j \neq \emptyset, \quad \alpha > 0$$

بنابراین بزرگترین باند پایینی که در *Decay Rate* میتوان با استفاده از یک تابع مرتبه دوم لیاپانف

پیدا کرد ، با X و α بدین صورت بدست می آید:

maximize α

$$X, Y, M_1, \dots, M_r$$

$$X > 0, Y \geq 0$$

مربوط به:

$$-XA_i^T - A_iX + M_i^T B_i^T + B_iM_i - (s-1)Y - 2\alpha X > 0, \quad (37-3)$$

$$2Y - XA_i^T - A_iX - XA_j^T - A_jX + M_j^T B_i^T + B_iM_j + M_i^T B_j^T + B_jM_i - 4\alpha X \geq 0,$$

$$i < j \text{ s.t. } h_i \cap h_j \neq \phi, X = P^{-1}, M_i = F_i X, Y = X Q X.$$

۳-۸- محدودیت روی ورودی و خروجی کنترل

۳-۸-۱- محدودیت روی ورودی کنترل

قضیه ۵- فرض کنید شرط اولیه $x(0)$ معلوم است. قید $\|u(t)\|_2 \leq \mu$ در همه زمانهای $t > 0$

قابل اجرا است اگر LMI های زیر برقرار باشند، ($X = P^{-1}$ and $M_i = F_i X$) [20].

$$\begin{bmatrix} 1 & x(0)^T \\ x(0) & X \end{bmatrix} \geq 0, \quad (38-3)$$

$$\begin{bmatrix} X & M_i^T \\ M_i & \mu^2 I \end{bmatrix} \geq 0. \quad (39-3)$$

۳-۸-۲- محدودیت روی خروجی

قضیه ۶- فرض کنید شرط اولیه $x(0)$ معلوم است. قید $\|y(t)\|_2 \leq \lambda$ در همه زمانهای $t > 0$

قابل اجرا است اگر LMI های زیر برقرار باشند، ($X = P^{-1}$)

$$\begin{bmatrix} 1 & x(0)^T \\ x(0) & X \end{bmatrix} \geq 0, \quad (40-3)$$

$$\begin{bmatrix} X & X C_i^T \\ C_i X & \lambda^2 I \end{bmatrix} \geq 0. \quad (41-3)$$

۳-۸-۳- شرط مستقل حالت اولیه

شرایط طراحی LMI های فوق برای محدودیتهای ورودی و خروجی به حالت‌های اولیه سیستم بستگی دارد. به این معنی که بهره‌های فیدبک F_i اگر حالت‌های اولیه $x(0)$ تغییر کند، باید مجدداً با استفاده از LMI های فوق تعیین شوند و این یک استفاده غیر مفید از LMI هاست.

محدودیتهای LMI روی کنترل ورودی و خروجی را اصلاح میکنیم، $x(0)$ نامشخص است، اما باند بالایی φ از $\|x(0)\|$ معلوم است بدین معنی که $\|x(0)\| \leq \varphi$ است. برای پوشش دادن مجموعه وسیعی از شرایط اولیه، مجموعه φ باید یک مقدار بزرگ را به خود اختصاص دهد.

LMI ها بصورت زیر اصلاح میشوند،

قضیه ۷- فرض کنید، $\|x(0)\| \leq \varphi$ ، که $x(0)$ نامعلوم است اما حد بالای φ معلوم است.

آنگاه $x^T(0)X^{-1}x(0) \leq 1$ است اگر $\varphi^2 I \leq X$ باشد، که $X = P^{-1}$.

۳-۹- حذف اغتشاش

در این بخش به طراحی کنترل کننده فازی همراه با اغتشاش برای مدل‌های فازی تاکاگی-سوگنو میپردازیم.

سیستم پیوسته زیر را با اغتشاش در نظر بگیرید:

$$\dot{x}(t) = \sum_{i=1}^r h_i(z(t)) \{A_i x(t) + B_i u(t) + E_i v(t)\}, \quad (42-3)$$

$$y(t) = \sum_{i=1}^r h_i(z(t)) C_i x(t), \quad (43-3)$$

که $v(t)$ اغتشاش است.

حذف اغتشاش را میتوان با مینیمم کردن γ مربوط به :

$$\sup_{\|v(t)\|_2 \neq 0} \frac{\|y(t)\|_2}{\|v(t)\|_2} \leq \gamma \quad (44-3)$$

انجام داد.

قضیه ۸- بهره های فیدبک F_i پایدار کننده مدل فازی و مینیمم کننده γ در رابطه (۴۴-۳) با

حل مسئله مینیمم سازی زیر بر اساس LMI ها بدست می آید.

$$\minimize_{x, M_1, \dots, M_r} \gamma^2$$

$$\begin{bmatrix} \left(\begin{array}{c} -\frac{1}{2}\{XA_i^T - M_j^T B_i^T + A_i X - B_i M_j \\ + XA_j^T - M_i^T B_j^T + A_j X - B_j M_i\} \end{array} \right) & -\frac{1}{2}(E_i + E_j) & \frac{1}{2}X(C_i + C_j)^T \\ -\frac{1}{2}(E_i + E_j)^T & \gamma^2 I & 0 \\ \frac{1}{2}(C_i + C_j)X & 0 & I \end{bmatrix} \geq 0, x > 0$$

$$i \leq j.s.t. h_i(z(t)) \cap h_j(z(t)) \neq \phi, M_i = F_i X.$$

۳-۱۰-کترل فازی Robust

حال به طراحی کترل فازی $Robust$ میپردازیم. در کل اگر طرح پایدار پذیر باشد، تعداد محدودی

کترل کننده پایدار وجود دارد. انتخاب یک کترل کننده ویژه از بین این گروه از کترل کننده -

های موجود، اغلب با مشخصات معینی از عملکرد کترل، صورت میگیرد. تکنیکهای مبنی بر LMI

نه تنها پایداری بلکه، برای مثال، سرعت خوب پاسخ، جلوگیری از اشباع محرک و محدودیت

خطای خروجی را نیز تضمین میکنند.

پایداری در کاربردهای کنترل عملی که اغلب همراه عدم قطعیت است ، مثلاً در طراحی محرکها و سنسورها در یک سیستم کنترل ، مطرح میشود. در اینجا یک کلاس از سیستمهای فازی تاکاگی- سوگنوی همراه با عدم قطعیت را تعریف میکنیم . همچنین در این بخش یک روش برای طراحی کنترل کننده فازی ارائه میدهیم بطوریکه نرم بلوکهای نامعین را خارج از کلاس پایداری کنترل کننده های PDC ، ماکزیمم کند.

۳-۱۰-۱ - مدل فازی شامل عدم قطعیت

در اینجا روی کنترل فازی $Robust$ در سیستمهای CFS تمرکز میکنیم ، برای طراحی کنترل فازی $Robust$ در سیستمهای DFS به مراجع [۴] ، [۵] مراجعه کنید.

برای دنبال کردن پایداری سیستمهای کنترل فازی ، اولین و لازم ترین مرحله در نظر گرفتن یک کلاس از سیستمهای فازی با عدم قطعیت است . برای این منظور ، ما بلوکهای نامشخص را به مدل فازی تاکاگی- سوگنو تبدیل میکنیم تا به مدل فازی دارای عدم قطعیت زیر برسیم،

plant Rule i

IF $z_1(t)$ is M_{i1} and...and $z_p(t)$ is M_{ip} (۴۵-۳)

THEN $\dot{x}(t) = (A_i + D_{ai}\Delta_{ai}(t)E_{ai})x(t) + (B_i + D_{bi}\Delta_{bi}(t)E_{bi})u(t)$, $i=1,2,\dots,r$

بلوکهای نامشخص ، در روابط زیر برای همه i ها صدق میکنند.

$$\|\Delta_{ai}(t)\| \leq \frac{1}{\gamma_{ai}}, \quad (۴۶-۳)$$

$$\Delta_{ai}(t) = \Delta_{ai}^T(t), \quad (۴۷-۳)$$

$$\|\Delta_{bi}(t)\| \leq \frac{1}{\gamma_{bi}}, \quad (۴۸-۳)$$

$$\Delta_{bi}(t) = \Delta_{bi}^T(t), \quad (49-3)$$

مدل فازی به اینصورت نمایش می یابد:

$$\dot{x}(t) = \sum_{i=1}^r h_i(z(t)) \{ (A_i + D_{ai} \Delta_{ai}(t) E_{ai}) x(t) + (B_i + D_{bi} \Delta_{bi}(t) E_{bi}) u(t) \}. \quad (50-3)$$

مدل فازی (45-3) یا (50-3) عدم قطعیتها را در بر دارند.

۳-۱۰-۲- شرط پایداری رباست

حال میخواهیم شرط پایداری را برای مدل فازی دارای عدم قطعیت (45-3) یا (50-3) نشان

دهیم. با جایگزین کردن کنترل کننده PDC در رابطه (50-3):

$$\begin{aligned} \dot{x}(t) &= \sum_{i=1}^r \sum_{j=1}^r h_i(z(t)) h_j(z(t)) \times \left\{ A_i - B_i F_j + \begin{bmatrix} D_{ai} & D_{bi} \end{bmatrix} \begin{bmatrix} \Delta_{ai} & 0 \\ 0 & \Delta_{bi} \end{bmatrix} \begin{bmatrix} E_{ai} \\ -E_{bi} F_j \end{bmatrix} \right\} x(t) \\ &= \sum_{i=1}^r h_i^2(z(t)) \left\{ A_i - B_i F_i + \begin{bmatrix} D_{ai} & D_{bi} \end{bmatrix} \begin{bmatrix} \Delta_{ai} & 0 \\ 0 & \Delta_{bi} \end{bmatrix} \begin{bmatrix} E_{ai} \\ -E_{bi} F_i \end{bmatrix} \right\} x(t) + \sum_{i=1}^r \sum_{i < j} h_i(z(t)) h_j(z(t)) \\ &\quad \times \left\{ A_i - B_i F_j + A_j - B_j F_i + \begin{bmatrix} D_{ai} & D_{bi} \end{bmatrix} \begin{bmatrix} \Delta_{ai} & 0 \\ 0 & \Delta_{bi} \end{bmatrix} \begin{bmatrix} E_{ai} \\ -E_{bi} F_j \end{bmatrix} \right\} + \\ &\quad \left[\begin{bmatrix} D_{ai} & D_{bi} \end{bmatrix} \begin{bmatrix} \Delta_{ai} & 0 \\ 0 & \Delta_{bi} \end{bmatrix} \begin{bmatrix} E_{ai} \\ -E_{bi} F_i \end{bmatrix} \right. \\ &\quad \left. + \begin{bmatrix} D_{aj} & D_{bj} \end{bmatrix} \begin{bmatrix} \Delta_{aj} & 0 \\ 0 & \Delta_{bj} \end{bmatrix} \begin{bmatrix} E_{aj} \\ -E_{bi} F_i \end{bmatrix} \right] x(t). \quad (51-3) \end{aligned}$$

به قضیه زیر میرسیم، که در واقع این قضیه شرایط پایداری $Robust$ را برای مدل فازی (45-3) یا

(50-3)، با کنترل کننده فازی PDC (3-21) میدهد.

قضیه ۹- سیستم فازی (۳-۴۵) یا [(۳-۵۰)]، با کنترل کننده PDC (۳-۲۱) پایدار میشود، اگر

یک ماتریس مثبت معین p و یک ماتریس متعارفی مثبت نیمه معین Q_0 وجود داشته باشد،

بطوریکه شرایط زیر را برآورده کند [20]،

$$S_{ii} + (s-1)Q_1 < 0, \quad \text{where } s > 1, \quad (52-3)$$

$$T_{ij} - 2Q_2 < 0, \quad i < j \quad \text{s.t. } h_i \cap h_j \neq \phi, \quad (53-3)$$

$$S_{ii} = \begin{bmatrix} (A_i - B_i F_i)^T P + P(A_i - B_i F_i) & PD_{ai} & PD_{bi} & E_{ai}^T & -F_i^T E_{bi}^T \\ D_{ai}^T P & -I & 0 & 0 & 0 \\ D_{bi}^T P & 0 & -I & 0 & 0 \\ E_{ai} & 0 & 0 & -\gamma_{ai}^2 I & 0 \\ -E_{bi} F_i & 0 & 0 & 0 & -\gamma_{bi}^2 I \end{bmatrix},$$

$$T_{ij} = \begin{bmatrix} \begin{pmatrix} (A_i - B_i F_j)^T P \\ + P(A_i - B_i F_j) \\ + (A_j - B_j F_i)^T P \\ + P(A_j - B_j F_i) \end{pmatrix} & PD_{ai} & PD_{bi} & PD_{aj} & PD_{bj} & E_{ai}^T & -F_j^T E_{bi}^T & E_{aj}^T & -F_i^T E_{bj}^T \\ D_{ai}^T P & -I & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ D_{bi}^T P & 0 & -I & 0 & 0 & 0 & 0 & 0 & 0 \\ D_{aj}^T P & 0 & 0 & -I & 0 & 0 & 0 & 0 & 0 \\ D_{bj}^T P & 0 & 0 & 0 & -I & 0 & 0 & 0 & 0 \\ E_{ai} & 0 & 0 & 0 & 0 & -\gamma_{ai}^2 I & 0 & 0 & 0 \\ -E_{bi} F_j & 0 & 0 & 0 & 0 & 0 & -\gamma_{bi}^2 I & 0 & 0 \\ E_{aj} & 0 & 0 & 0 & 0 & 0 & 0 & -\gamma_{aj}^2 I & 0 \\ -E_{bj} F_i & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -\gamma_{bj}^2 I \end{bmatrix}$$

$$Q_1 = \text{block-diag}(Q_0 \quad 0 \quad 0 \quad 0 \quad 0),$$

$$Q_2 = \text{block-diag}(Q_0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0).$$

اثبات آن با انتخاب یک کاندیدا از توابع لیپانف بصورت $x(t)^T p x(t)$ و بدنبال آن مشتق گیری و سپس جایگذاری بدست می آید. وقتی $Q_2 = 0, Q_1 = 0$ شود، یعنی، $Q_0 = 0$ ، شرایط پایداری *Robust* آرام شده بصورت زیر کاهش می یابد.

$$P > 0, \quad S_{ii} < 0, \quad T_{ij} < 0, \quad i < j \quad \text{s.t.} \quad h_i \cap h_j \neq \emptyset$$

کاربردها

با این بنیاد ساده، منطق فازی در جاهای زیادی از جامعه مدرن استفاده میشود. در علم پزشکی، منطق فازی برای تشخیص مقدماتی امراض ریه با استفاده از علائم اولیه بکار میرود. منطق فازی در بازاریابی برای کمک به توسعه روشهای فروش و نیز برای تخمین سهم فروش استفاده می شود. قیمت صنعت بانک سازی سنگاپور با استفاده از مجموعه های فازی و عملگرهای فازی توسط متخصصان تعیین میشود. این کاربردها تأثیر مفهوم ساده مجموعه ها و منطق فازی را نشان میدهد.

۳-۱۱- نتیجه گیری

بمنظور طراحی یک کنترل کننده فازی برای یک بازو، لازم است اطلاعات بصورت قوانین مربوط به مجموعه ها بیان شوند. *PDC* یک روش طراحی کنترل کننده فازی با استفاده از یک مدل

فازی $T - S$ است اما در این روش طراحی پایداری سیستمهای کنترل مطرح نمیشود. بکمک قوانین PDC فازی و LMI ها و بکمک توابع پایداری لیاپانف پایدارسازی سیستمهای فازی انجام میشود.

برای پاسخگو بودن منطق فازی به سیستمهای پیچیده تر، مدل منطق فازی بطور توصیفی گسترش می یابد و هر پارامتر دیگری برای مدل شدن نیاز به بعد دیگری در پایگاه قوانین دارد و این، تعداد قوانین را با تعداد توابع عضویت در بعد جدید ضرب میکند که باعث ایجاد مشکل ابعادی میشود و مدل منطق فازی تحت تاثیر قرار میدهد. از نتایج سوء این مسئله، افزایش نیازمندیهای حافظه و نیز افزایش زمان محاسبه است. برای غلبه کردن بر این مسئله، مدلهای چندمرحله ای بررسی میشوند. یک مدل چند مرحله ای شامل چندین مرحله سری و موازی است که خروجی برخی مراحل، ورودی دیگر مراحل است.

فصل چهارم :

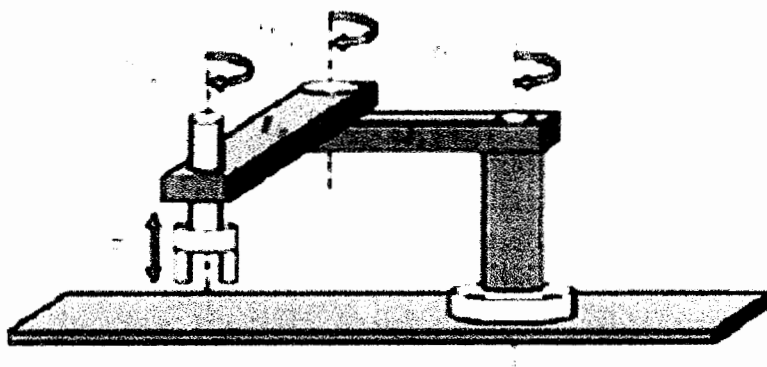
طراحی کنترل کننده PDC فازی

برای

بانروی ماهر تک رابط

در این فصل ابتدا مدل دینامیکی ترکیبی از بازوی ربات و موتورهای مفاصل را شرح می‌دهیم و سپس برای بازوی ماهر تک رابط، یک کنترل کننده فازی با استفاده از روش جبران سازی موازی و اعمال شرایط پایداری طراحی کنیم. نتایج شبیه سازی در آخر فصل خواهد آمد.

شماتیک کلی یک بازوی ماهر لولایی در شکل ۴-۱ نشان داده شده است، این نما، زاوایای مفاصل و حرکت آنها نسبت به دیگری نشان می‌دهد.



شکل ۴-۱- دیاگرام کلی یک بازوی ماهر لولایی

۴-۱- کنترل دینامیکهای ربات و موتورهای مفاصل

Beekman و lee [6] و [5] mahmoud ، یک مدل ترکیبی از بازوی ربات و موتور DC ارائه دادند.

معادلات حرکت یک بازو با مفاصل کشویی بدین ترتیب توصیف میشود:

$$M'(q)\ddot{q} + C(q, \dot{q})\dot{q} + G(q) = \tau + d' \quad (۱-۴)$$

که در این معادله:

$q, \dot{q}, \ddot{q} \in R^n$ بردار موقعیت، سرعت و شتاب مفاصل، $M'(q) \in R^{n \times n}$ ماتریس لختی،

$C(q, \dot{q}) \in R^n$ بردار نیروهای جانب مرکز و کوریولیس، $G(q) \in R^n$ بردار نیروی جاذبه، $\tau \in R^n$

برداز گشتاورهای ایجاد شده توسط مفاصل، $d' \in R^n$ اغتشاش خارجی وارد شده به سیستم

میباشد، ماتریس M' در معادله بالا، مثبت معین متقارن است.

ارتباط بین موقعیت مفصل q و موقعیت شفت موتور q_m توسط رابطه زیر داده میشود: $q_m = nq$

$n > 0, n \in R^n$ (ماتریس مثبت معین) یک ماتریس قطری از مقادیر چرخ دنده برای n مفصل

است.

مدل الکتریکی زامین موتور DC کنترل شده آرمیچر چنین است:

$$R_j i_j + L_j \frac{di_j}{dt} + k_{bj} \frac{dq_{mj}}{dt} = u_j, \text{ for } j = 1, 2, \dots, n. \quad (۲-۴)$$

R_j مقاومت مدار آرمیچر، L_j اندوکتانس مدار آرمیچر، ثابت k_{bj} نیروی الکتریکی مؤثر برگشتی

(EMF) موتور، i_j جریان آرمیچر، q_{mj} موقعیت شفت موتور و u_j ولتاژ ورودی آرمیچر است.

و داریم:

$$R = \text{diag}[R_j], L = \text{diag}[L_j], k_b = \text{diag}[k_{bj}]$$

[.] $diag$ نماد یک ماتریس قطری با بعد n میباشد.

$$u = \begin{bmatrix} u_1 \\ \vdots \\ u_n \end{bmatrix}, i = \begin{bmatrix} i_1 \\ \vdots \\ i_n \end{bmatrix}, q_m = \begin{bmatrix} q_{m1} \\ \vdots \\ q_{mn} \end{bmatrix} \quad (3-4)$$

بدین ترتیب مدل الکتریکی موتورهای DC کنترل آرمیچر در رابطه (۴-۲) بصورت ذیل بیان میشوند:

$$Ri + L \frac{di}{dt} + k_b \dot{q}_m = u \quad (4-4)$$

بنابراین گشتاور ایجاد شده در طرف مفصل با جریان آرمیچر مرتبط است.

$$\tau = n.k_t.i \quad (5-4)$$

$k_t \in R^{n \times n}$ ماتریس قطری ثابتهای گشتاور موتور و $k_t > 0$.

با جایگذاری (۵-۴) در (۴-۴) رابطه زیر حاصل میشود:

$$k_n \tau + L_n \dot{\tau} + k_{bn} \dot{q} = u \quad (6-4)$$

$$R_n = R(nk_t)^{-1}, L_n = L(nk_t)^{-1}, k_{bn} = k_b.n \quad (7-4)$$

روابط معکوس در (۷-۴) به طبیعت فیزیکی n و k_t بستگی دارند.

اکنون برای بدست آوردن مدل ربات ترکیبی، با جایگذاری (۴-۱) در (۶-۴) داریم:

$$L_n M'(q) \ddot{q}^{(3)} + (R_n M'(q) + L_n [M'(q) + C(q, \dot{q})]) \ddot{q} + (R_n C(q, \dot{q}) + L_n \dot{C}(q, \dot{q}) + k_{bn}) \dot{q} + R_n G(q) + L_n \dot{G}(q) = u + R_n d' + L_n \dot{d}' \quad (8-4)$$

$q^{(3)}$ بیانگر مشتق مرتبه سوم q است.

مدل دینامیکی (۸-۴) را بصورت کامل زیر می نویسیم:

$$M(q)q^{(3)} + D(q, \dot{q}, \ddot{q}) = u + R_n d' + L_n \dot{d}' \quad (9-4)$$

$$M(q) = L_n M'(q) \quad (10-4)$$

$$D(q, \dot{q}, \ddot{q}) = (R_n M'(q) + L_n [\dot{M}'(q) + C(q, \dot{q})]) \ddot{q} + \quad (11-4)$$

$$(R_n C(q, \dot{q}) + L_n \dot{C}(q, \dot{q}) + k_{bn}) \dot{q} + R_n G(q) + L_n \dot{G}(q).$$

$$x = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \triangleq \begin{bmatrix} q \\ \dot{q} \\ \ddot{q} \end{bmatrix} : \text{ با تشکیل بردار حالت :}$$

معادله (9-4) را بفرم استاندارد زیر مینویسیم:

$$\dot{x} = \begin{bmatrix} x_2 \\ x_3 \\ -M^{-1}(x_1)D(x) \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ M^{-1}(x_1) \end{bmatrix} u + \begin{bmatrix} 0 \\ 0 \\ M^{-1}(x_1)(R_n d' + L_n \dot{d}') \end{bmatrix}. \quad (12-4)$$

با جایگذاری $d = M^{-1}(x_1)(R_n d' + L_n \dot{d}')$ و با فرض اینکه $d \in L_2[0, t_f]$ معادله (12-4) را

بفرم زیر مینویسیم:

$$\dot{x} = \begin{bmatrix} x_2 \\ x_3 \\ -M^{-1}(x_1)D(x) \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ M^{-1}(x_1) \end{bmatrix} u + \begin{bmatrix} 0 \\ 0 \\ I \end{bmatrix} d. \quad (13-4)$$

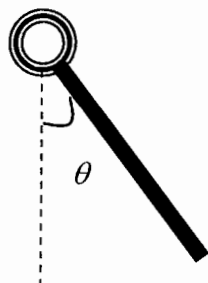
این معادله یک سیستم دینامیکی غیرخطی با یک تابع ورودی غیرخطی را نشان میدهد.

بردار حالت (13-4) ترکیبی است از موقعیتها، سرعتها و گشتاورهای مفاصل و بردار ورودی u

شامل ولتاژهای ورودی آرمیچر است.

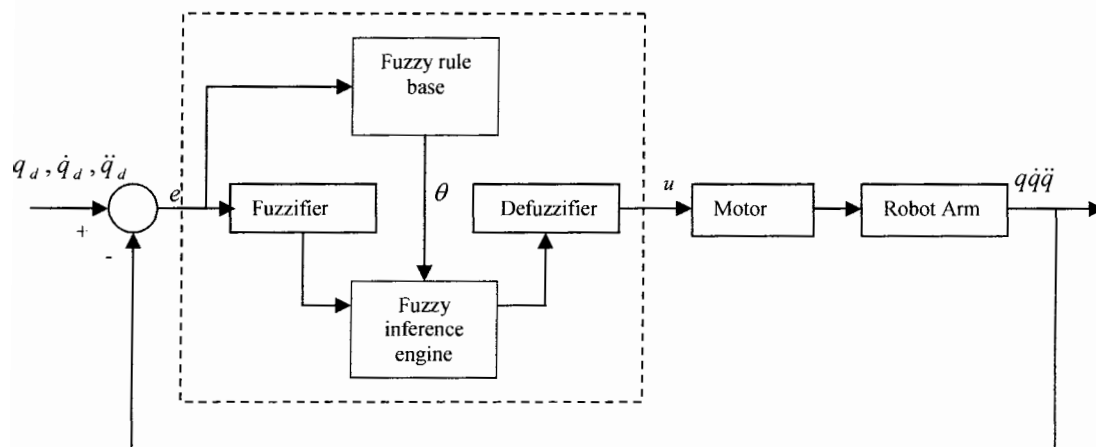
۲-۴- طراحی کنترل کننده فازی برای بازوی تک رابط

هدف ، کنترل جابجایی یک بازوی ماهر تک رابط در حضور اغتشاش و عدم قطعیتها است.

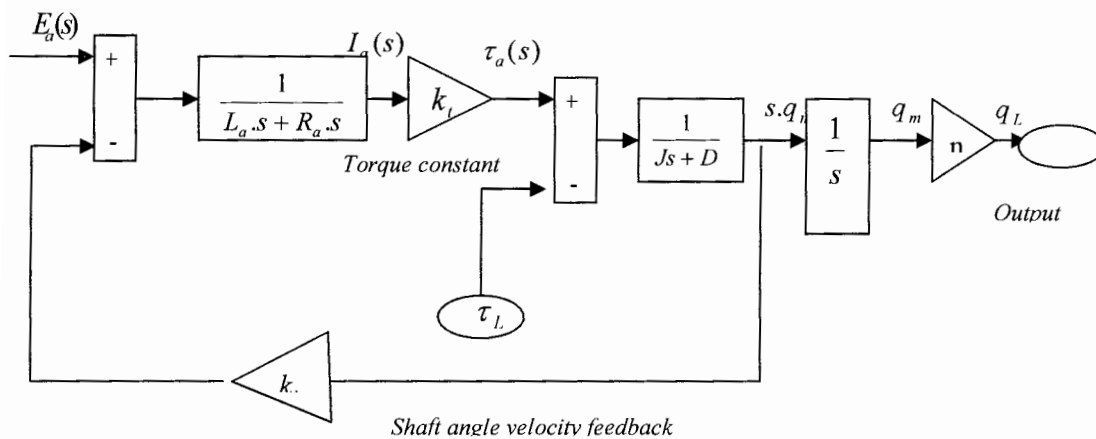


شکل ۲-۴- بازوی ماهر لولایی تک رابط

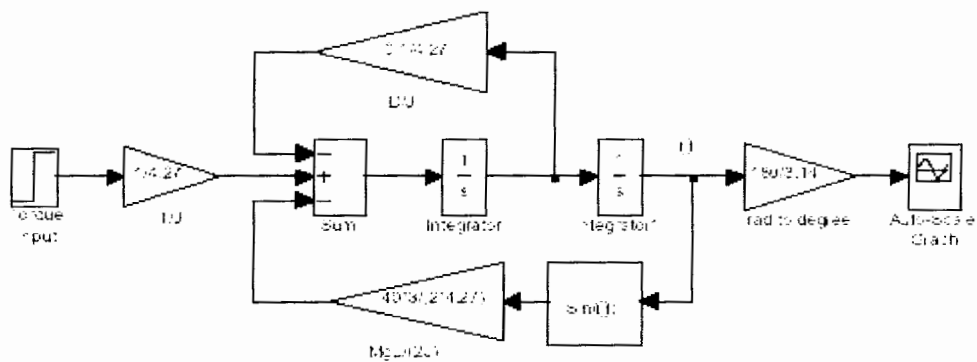
بلوک دیاگرام کلی طرح در شکل ۳-۴ آمده است [23] .



شکل ۳-۴- سیستم منطق فازی بمنظور کنترل مسیر



شکل ۴-۴- بلوک دیاگرام موتور DC کنترل شده با آرمیچر



شکل ۴-۵- بلوک دیاگرام سیستم رابط صلب غیر خطی

متغیرهای ورودی کنترل کننده فازی، خطای موقعیت زاویه ای $e(rad)$ و خطای سرعت زاویه ای $\dot{e}(rad/s)$ هستند که بصورت زیر تعریف میشوند:

موقعیت واقعی - فرمان موقعیت = e

$$\dot{e} = \frac{e(kt) - e((k-1)T)}{T} \quad (14-4)$$

T زمان نمونه گیری را نشان می‌دهد، متغیر خروجی کنترل کننده فازی، ورودی کنترل $u(n.m)$ به موتور راه انداز است.

با تعریف متغیرهای ذیل و جایگزینی آنها در معادله اصلی بازو، فرم فضای حالت را برای بازوی تک رابط بدست می آوریم.

معادله حرکت بازوی تک رابط چنین است:

$$J \frac{d^2 \theta}{dt^2} + D \frac{d\theta}{dt} + mgl \frac{\sin \theta}{2} = \tau(t) \quad (15-4)$$

θ زاویه بازو از خط عمود است.

$\tau(t)$ ، بعنوان ورودی به بازو اعمال می‌گردد که همان گشتاور تولید شده توسط موتور است.

خصوصیات موتور، انتخاب و مدل آنها را در فصل ۲ شرح دادیم.

مشخصات موتور بشرح ذیل میباشد:

$$R_t = 1.6\Omega, L = 0.048H, k_b = 0.19, k_t = 0.2613, n = 62.55.$$

با استفاده از رابطه (۷-۴)، داریم:

$$R_n = 1.6.(62.55 \times 0.2613)^{-1} = 98 \times 10^{-3} \Omega$$

$$L_n = 0.048.(62.55 \times 0.2613)^{-1} = 2.94 \times 10^{-3} H$$

$$k_{bn} = 0.19 \times 92.55 = 11.884$$

$$M' = 4.6kg, l = 0.5m, D = 0.1, g = 9.8m/s^2 \quad \text{و پارامترهای بازو چنین هستند:}$$

با استفاده از روابط (۱۰-۴) و (۱۱-۴)، M و $D(q, \dot{q}, \ddot{q})$ را محاسبه مینماییم.

$$M = 2.94 \times 10^{-3} \times 4.6 = 13.524 \times 10^{-3} \Rightarrow M^{-1} = 73.943$$

(۱۶-۴)

$$D(q, \dot{q}, \ddot{q}) = 0.451\ddot{q} + 11.89\dot{q} + 1.104 \times \sin(q) + 0.033 \times \cos(q)$$

و معادلات حالت بازو بدین ترتیب حاصل میگردند.

$$\dot{x} = \begin{bmatrix} x_2 \\ x_3 \\ -33.35 \times \ddot{q} - 879.18 \times \dot{q} - 81.63 \times \sin(q) - 2.44 \times \cos(q) \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 73.943 \end{bmatrix} u + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} d. \quad (۱۷-۴)$$

$$\dot{x} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -81.63 \times \frac{\sin(x_1)}{x_1} & -2.44 \times \frac{\cos(x_1)}{x_2} - 879.18 & -33.35 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 73.943 \end{bmatrix} u + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} d. \quad (۱۸-۴)$$

معادله فوق غیرخطی بوده و در ابتدا باید بصورت خطی بازنویسی شوند.

۴-۲-۱- خطی سازی سیستم غیر خطی با روش تاکاگی - سوگنو

ماتریس فضای حالت خطی برای معادله (۱۷-۴) بصورت ذیل بدست می آید.

جملات غیرخطی در نمایش فضای حالت فوق $z_1(t) = \frac{\sin x_1}{x_1}$, $z_2(t) = \frac{\cos x_1}{x_2}$ هستند.

فرض میکنیم: $x_1(t) \in \left[-\frac{\pi}{2}, \frac{\pi}{2}\right]$, $x_2(t) \in \left[-\frac{\pi}{2}, \frac{\pi}{2}\right] \text{ except } [-6, 6]$.

$$\dot{x}(t) = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -81.63 \times z_1 & -2.44 \times z_2 - 879.18 & -33.35 \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 73.943 \end{bmatrix} u(t) + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} d(t). \quad (۱۹-۴)$$

با محاسبه مقادیر ماکزیمم و مینیمم تابع غیرخطی، در بازه های مفروض، داریم:

$$z_1 = \frac{\sin x_1}{x_1} \Rightarrow \begin{cases} \max_{x_1, x_2} & z_1 = 1.000 \\ \min_{x_1, x_2} & z_1 = 0.6366 \end{cases} \quad (۲۰-۴)$$

$$z_2 = \frac{\cos x_1}{x_2} \Rightarrow \begin{cases} \max_{x_1, x_2} z_2 = 10 \\ \min_{x_1, x_2} z_2 = -0.6366 \end{cases} \quad (21-4)$$

و با استفاده از مقادیر مینیمم و ماکزیمم، $z_1(t)$ و $z_2(t)$ را بدست می آوریم:

$$z_1(t) = \frac{\sin x_1}{x_1} = M_1(z(t)) \times 0.6366 + M_2(z(t)) \times 1 \quad (22-4)$$

$$z_2(t) = \frac{\cos x_1}{x_2} = N_1(z(t)) \times (-0.6366) + N_2(z(t)) \times 10 \quad (23-4)$$

$$M_1(z(t)) + M_2(z(t)) = 1 \rightarrow M_1(z(t)) = 1 - M_2(z(t)) \quad (24-4)$$

$$N_1(z(t)) + N_2(z(t)) = 1 \rightarrow N_1(z(t)) = 1 - N_2(z(t)) \quad (25-4)$$

$$z_1(t) = 0.6366 + 0.3634 \times M_2(z(t)) \Rightarrow \begin{cases} M_2(z(t)) = \frac{z_1(t) - 0.6366}{0.3634} \text{ "big"} \\ M_1(z(t)) = \frac{1 - z_1(t)}{0.3634} \text{ "small"} \end{cases} \quad (26-4)$$

$$z_2(t) = -10.33 \times N_1(z(t)) + 10 \Rightarrow \begin{cases} N_2(z(t)) = \frac{z_2(t) - 10}{-10.33} \text{ "hi"} \\ N_1(z(t)) = \frac{0.6366 + z_2(t)}{10.33} \text{ "low"} \end{cases} \quad (27-4)$$

اکنون یک مدل چهار قانونی با استفاده از منطق فازی، بفرم زیر نوشته میشوند:

model rule 1:

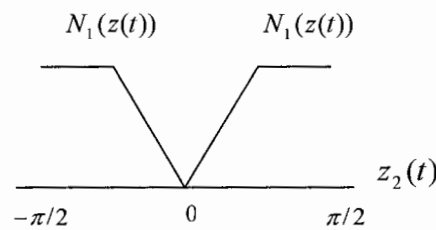
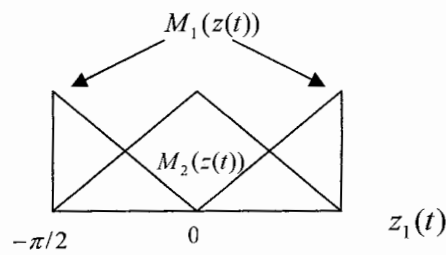
IF $z_1(t)$ is "big" and $z_2(t)$ is "hi",

then $\dot{x}(t) = A_1 x(t) + B_1 \tau(t) + d(t)$.

model rule 2:

IF $z_1(t)$ is "big" and $z_2(t)$ is "low",

then $\dot{x}(t) = A_2 x(t) + B_2 \tau(t) + d(t)$.



شکل ۴-۶- توابع عضویت بازوی ماهر تک رابط

۴-۲-۲- طراحی کنترل کننده فازی PDC برای بازوی ماهر تک رابط

در این مرحله به طراحی کنترل کننده PDC می پردازیم. PDC ، روشی برای طراحی یک کنترل کننده فازی با یک مدل فازی تاکاگی - سوگنو است بنابراین قوانین PDC برای کنترل کننده فازی فوق، چنین میگردد:

model rule 1:

IF $z_1(t)$ is "big" and $z_2(t)$ is "hi",

then $u_1(t) = F_1 x(t) + d(t)$.

model rule 2:

IF $z_1(t)$ is "big" and $z_2(t)$ is "low",

then $u_2(t) = F_2 x(t) + d(t)$.

model rule 3:

IF $z_1(t)$ is "small" and $z_2(t)$ is "hi",

then $u_3(t) = F_3 x(t) + d(t)$.

model rule 4:

IF $z_1(t)$ is "small" and $z_2(t)$ is "low",

then $u_4(t) = F_4 x(t) + d(t)$.

$$u(t) = -\sum_{i=1}^4 h_i(z(t)) F_i x(t) + d(t)$$

$$d = 0.1 \times \sin(2t)$$

ماتریس بهره F حاصل از کنترل کننده PDC ، با استفاده از جعبه ابزار LMI در مطلب چنین

حاصل می شود.

$$F = \begin{bmatrix} F_1 \\ F_2 \\ F_3 \\ F_4 \end{bmatrix} = \begin{bmatrix} 110.4 & 1220.64 & 44.81 & 0 \\ 110.4 & 1185.55 & 44.81 & 0 \\ 70.28 & 1220.64 & 44.81 & 0 \\ 70.28 & 1185.55 & 44.81 & 0 \end{bmatrix}$$

و ماتریس مثبت معین p بطریقی معین می شود که کلیه شرایط پایداری برقرار گردد.

حذف اغتشاش $d_i(t)$ با استفاده از رابطه (۴۴-۳) صورت میگیرد.

$Decay Rate$ و ماتریس مثبت معین متعارفی p حاصل، چنین هستند.

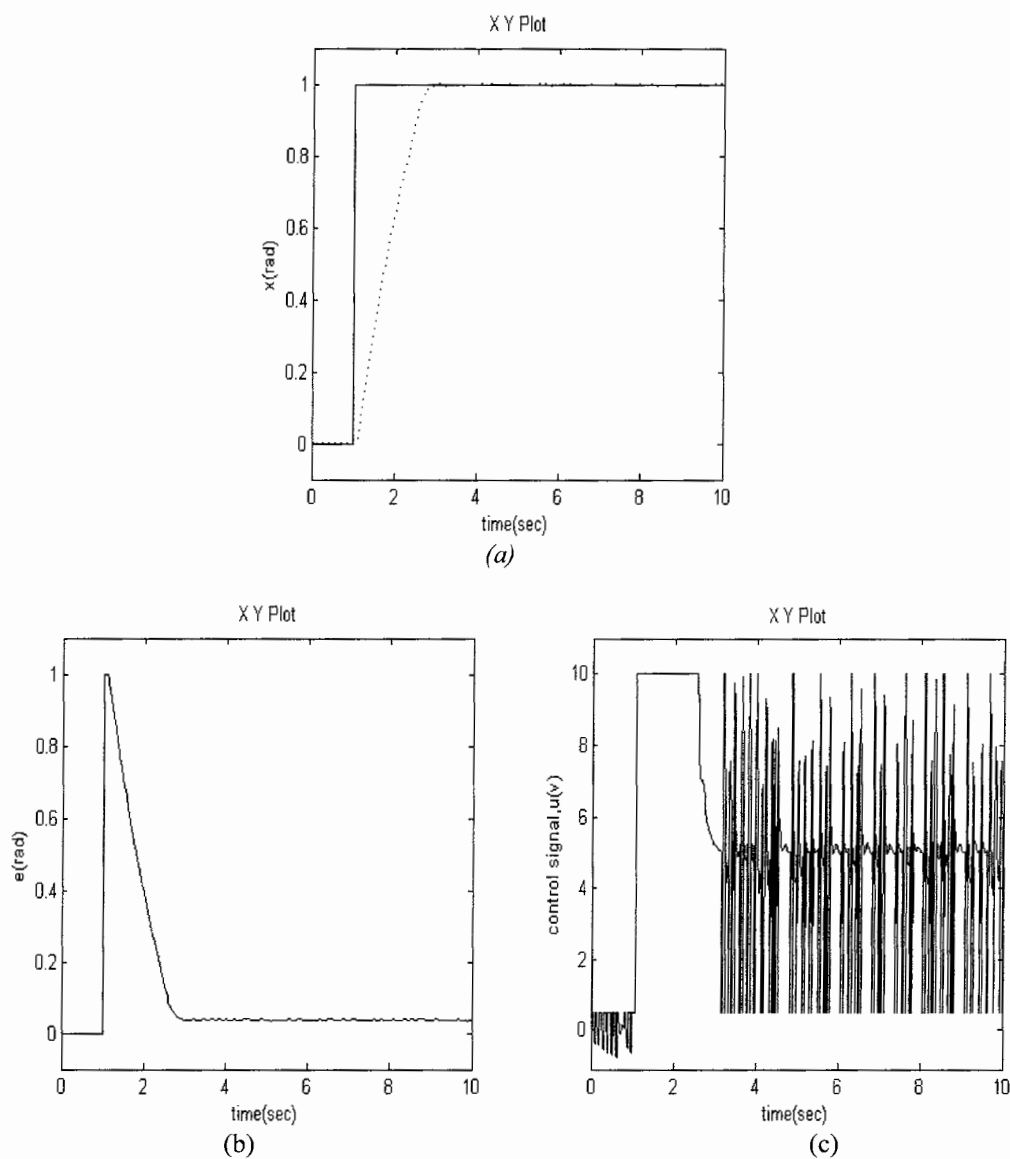
$$decay \ rate = 9.104 \times 10^{-5}$$

$$p = \begin{bmatrix} 280.5685 & 60.1814 & 280.5685 \\ 60.1814 & 220.4903 & 67.6261 \\ 280.5685 & 67.6261 & 489.0239 \end{bmatrix}$$

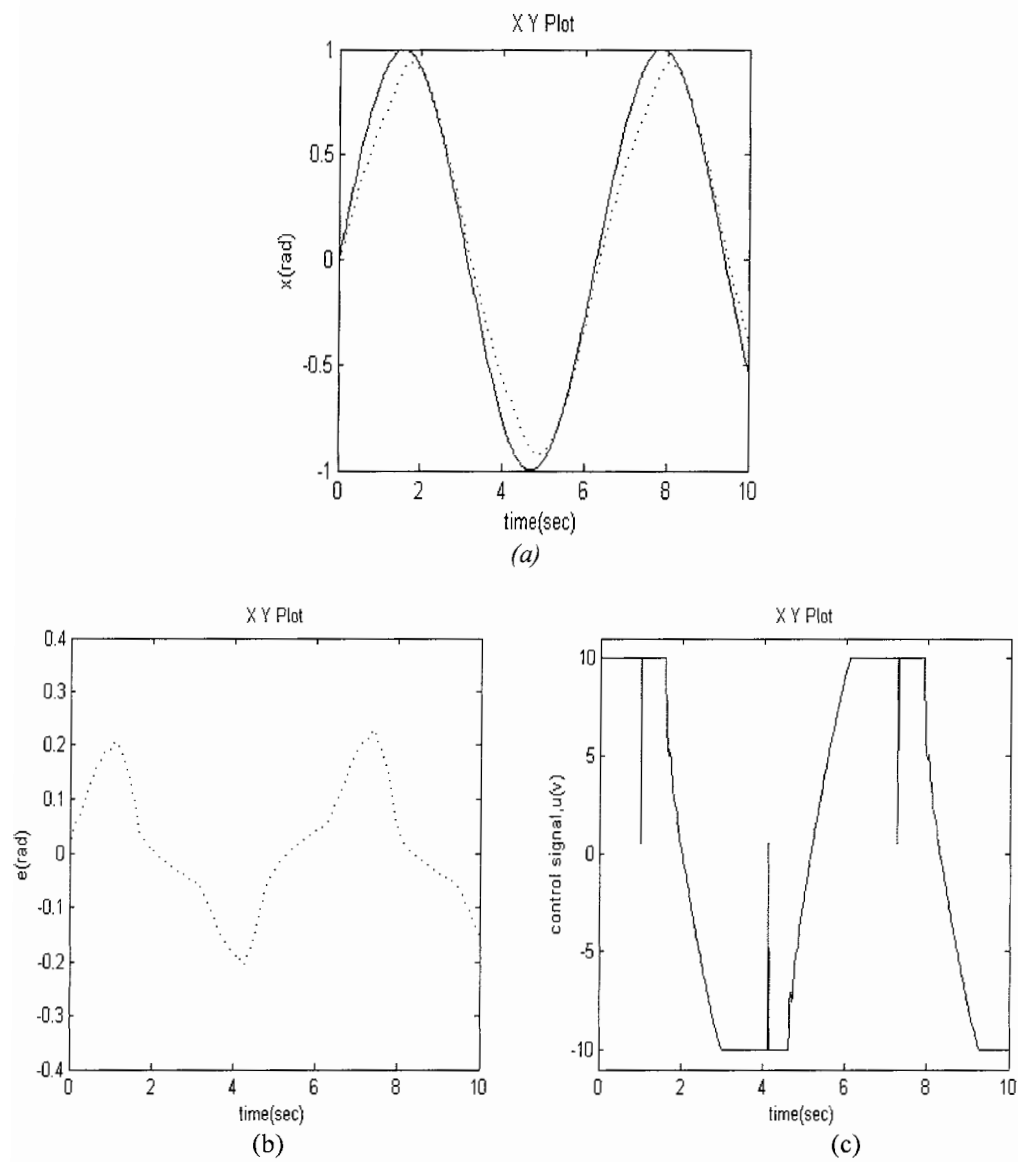
نتایج حاصل از شبیه سازی نشان میدهند که سیستم پایدار مجانبی جهانی است.

۴-۳- نتایج حاصل از شبیه سازی بازوی ماهر تک رابط

۴-۳-۱- نتایج حاصل از شبیه سازی بدون اعمال اغتشاش



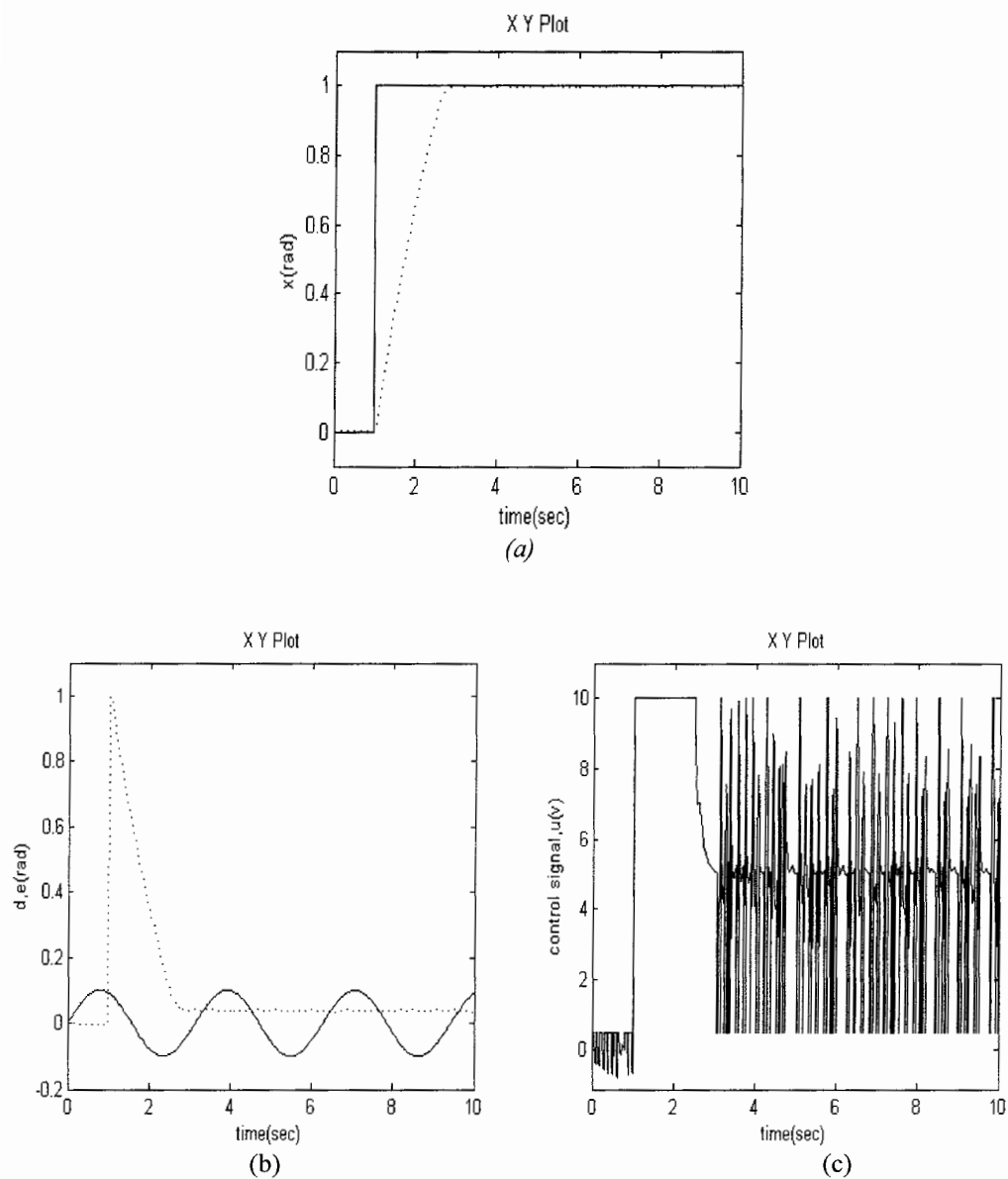
شکل ۴-۷- (a) مسیر مرجع و مسیر تعقیب شده (نقطه چین) بفرم پله (b) خطای حاصل از تعقیب مسیر (نقطه چین) سیگنال کنترل (c) (چین)



شکل ۴-۸- (a) مسیر مرجع و مسیر تعقیب شده (نقطه چین) بفرم سینوسی (b) خطای حاصل از تعقیب

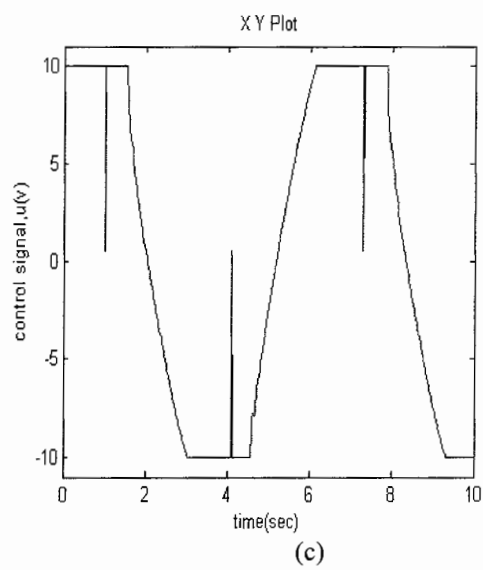
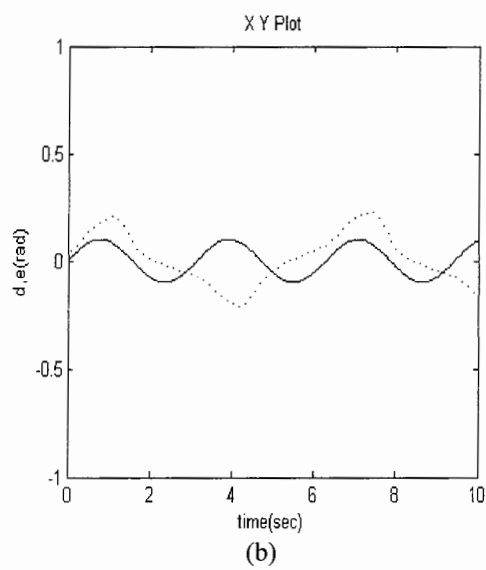
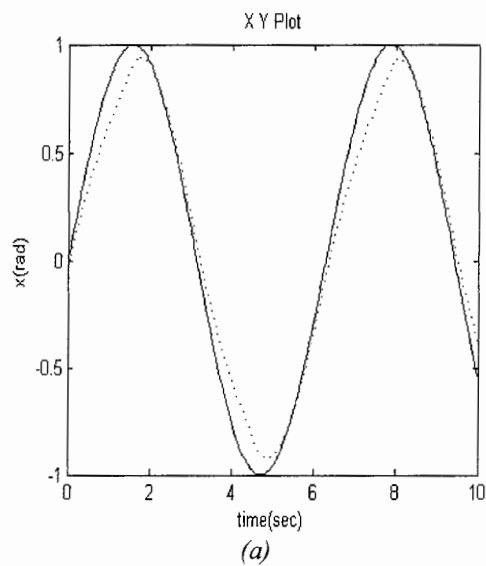
مسیر (نقطه چین) (c) سیگنال کنترل

۴-۳-۲- نتایج حاصل از شبیه سازی با حضور اغتشاش



شکل ۴-۹- (a) مسیر مرجع و مسیر تعقیب شده (نقطه چین) بفرم پله (b) اغتشاش وارد شده به سیستم و خطای

حاصل از تعقیب مسیر (نقطه چین) (c) سیگنال کنترل



شکل ۴-۱۰ (a) مسیر مرجع و مسیر تعقیب شده (نقطه چین) بفرم سینوسی (b) اغتشاش وارد شده به سیستم و

خطای حاصل از تعقیب مسیر (نقطه چین) (c) سیگنال کنترل

فصل پنجم :

تعمیم کنترل کننده PDC فازی

مرباست برای بانروهایی ماهر دو رابط

در این فصل یک کنترل کننده فازی پایدار با استفاده از روش جبران سازی موازی برای بازوی ماهر دو رابط لولایی، طراحی میکنیم و سپس به تعمیم بازوهای n رابط می پردازیم. معمولترین روش کنترل بازوی ماهر چند رابط در سیستمهای رباتیک، کنترل هر بازو بطور مستقل است یعنی میتوان هر بازو را بصورت مجزا کنترل نموده و خروجی یک رابط را بعنوان سیگنال اغتشاش به رابط دوم اعمال نمود.

۵-۱- مدلسازی مدل دینامیکی یک بازوی ماهر دو رابط

مدلسازی کلی سیستم با استفاده از قوانین فازی

یک بازوی لولایی دو رابط را در نظر بگیرید.

معادلات دینامیکی بصورت زیر نوشته میشوند:

$$\tau = M(\theta)\ddot{\theta} + \nu(\theta, \dot{\theta})\dot{\theta} + G(\theta) \quad (۱-۵)$$

$$\tau = \begin{bmatrix} \tau_1 \\ \tau_2 \end{bmatrix}, \theta = \begin{bmatrix} \theta_1 \\ \theta_2 \end{bmatrix} \quad (۲-۵)$$

$$M(\theta) = \begin{bmatrix} (m_1 + m_2)l_1^2 & m_2l_1l_2(s_1s_2 + c_1c_2) \\ m_2l_1l_2(s_1s_2 + c_1c_2) & m_2l_2^2 \end{bmatrix} \quad (۳-۵)$$

$$v(\theta, \dot{\theta}) = m_2 l_1 l_2 (c_1 s_2 - s_1 c_2) \begin{bmatrix} 0 & -\dot{\theta}_2 \\ -\dot{\theta}_1 & 0 \end{bmatrix} \quad (4-5)$$

$$G(\theta) = \begin{bmatrix} -(m_1 + m_2) l_1 g s_1 \\ -m_2 l_2 g s_2 \end{bmatrix} \quad (5-5)$$

که n تعداد مفاصل و $M(\theta)$ ماتریس $(n \times n)$ ممان اینرسی، $v(\theta, \dot{\theta})$ یک بردار $n \times 1$ شامل

جملات گریز از مرکز و کریولیس هستند و τ گشتاور اعمال شده به مفاصل است [21].

m_2, m_1 بترتیب جرم و l_2, l_1 طولهای بازوهای اول و دوم هستند، θ_1 زاویه ساخته شده توسط

بازوی اول با خط افق و θ_2 زاویه ساخته شده توسط بازوی اول و بازوی دوم است و نیز،

$$s_1 = \sin(q_1), s_2 = \sin(q_2), c_1 = \cos(q_1), c_2 = \cos(q_2)$$

۵-۱-۱- فرمولسازی مسئله

طراحی کنترل کننده فازی شامل مراحل زیر است:

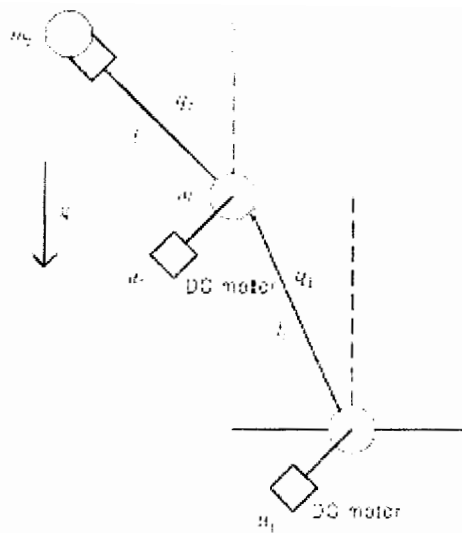
یک توصیف از ساختار کنترل کننده، بدست آوردن و نمایش دانسته ها یا پایگاه قوانین و طراحی

و تحلیل کنترل کننده. ورودیهای کنترل منطق فازی، خطا در موقعیت و خطا در سرعت هستند.

خروجیهای کنترل کننده، گشتاورهای محرک برای اعمال به مفاصل هستند.

دیگرام بازوی دو رابط راه اندازی شده توسط موتورهای DC در شکل ۵-۱ آمده است، همانگونه

که از شکل مشاهده میشود، هر بازو توسط یک موتور، هدایت میشود.



شکل ۵-۱- بازوی ماهر راه اندازی شده توسط موتورهای DC

برای مدل کردن فازی این سیستم ، مدل دینامیکی بازوی ماهر را بفرم فضای حالت مینویسیم.

$$\dot{x}_1 = x_2 + w_1 \quad (6-5)$$

$$\dot{x}_2 = f_1(x) + g_{11}(x)\tau_1 + g_{12}(x)\tau_2 + w_2 \quad (7-5)$$

$$\dot{x}_3 = x_4 + w_3 \quad (8-5)$$

$$\dot{x}_4 = f_2(x) + g_{21}(x)\tau_1 + g_{22}(x)\tau_2 + w_4 \quad (9-5)$$

$$y_1 = x_1 + v_1 \quad (10-5)$$

$$y_2 = x_3 + v_2 \quad (11-5)$$

$$f_1(x) = \frac{(s_1 c_2 - c_1 s_2)}{l_1 l_2 [(m_1 + m_2) - m_2 (s_1 s_2 + c_1 c_2)^2]} \times [m_2 l_1 l_2 (s_1 s_2 + c_1 c_2) x_2^2 - m_2 l_2^2 x_4^2] \\ + \frac{1}{l_1 l_2 [(m_1 + m_2) - m_2 (s_1 s_2 + c_1 c_2)^2]} \times [(m_1 + m_2) l_2 g s_1 - m_2 l_2 g s_2 (s_1 s_2 + c_1 c_2)] \quad (12-5)$$

$$f_2(x) = \frac{(s_1 c_2 - c_1 s_2)}{l_1 l_2 [(m_1 + m_2) - m_2 (s_1 s_2 + c_1 c_2)^2]} \times [-(m_1 + m_2) l_1^2 x_2^2 + m_2 l_1 l_2 (s_1 s_2 + c_1 c_2) x_4^2] \quad (13-5)$$

$$+ \frac{1}{l_1 l_2 [(m_1 + m_2) - m_2 (s_1 s_2 + c_1 c_2)^2]} \times [-(m_1 + m_2) l_1 g s_1 (s_1 s_2 + c_1 c_2) + (m_1 + m_2) l_1 g s_2]$$

$$g_{11}(x) = \frac{m_2 l_2^2}{m_2 l_1^2 l_2^2 [(m_1 + m_2) - m_2 (s_1 s_2 + c_1 c_2)^2]} \quad (14-5)$$

$$g_{12}(x) = \frac{-m_2 l_1 l_2 (s_1 s_2 + c_1 c_2)}{m_2 l_1^2 l_2^2 [(m_1 + m_2) - m_2 (s_1 s_2 + c_1 c_2)^2]} \quad (15-5)$$

$$g_{21}(x) = \frac{-m_2 l_1 l_2 (s_1 s_2 + c_1 c_2)}{m_2 l_1^2 l_2^2 [(m_1 + m_2) - m_2 (s_1 s_2 + c_1 c_2)^2]} \quad (16-5)$$

$$g_{22}(x) = \frac{(m_1 + m_2) l_1^2}{m_2 l_1^2 l_2^2 [(m_1 + m_2) - m_2 (s_1 s_2 + c_1 c_2)^2]} \quad (17-5)$$

مشخصات موتور DC بکار رفته در شبیه سازی بشرح ذیل میباشد:

$$R_t = 1.6\Omega, L = 0.048H, k_b = 0.19, k_t = 0.2613, n = 62.55.$$

$$R_n = 98 \times 10^{-3} \Omega, L_n = 2.94 \times 10^{-3} H, k_{bn} = 11.884$$

و پارامترهای بازو چنین هستند:

$$M'_1 = 4.6kg, M'_2 = 2.3kg, l_1 = 0.5m, l_2 = 0.2m, D_1 = 0.1, D_2 = 0.2, g = 9.8m/s^2$$

با استفاده از روابط (4-10) و (4-11)، M و $D(q, \dot{q}, \ddot{q})$ را محاسبه مینماییم.

$$M_1 = 2.94 \times 10^{-3} \times 4.6 = 13.524 \times 10^{-3} \Rightarrow M^{-1} = 73.943 \quad (18-5)$$

$$M_2 = 2.94 \times 10^{-3} \times 2.3 = 6.762 \times 10^{-3} \Rightarrow M^{-1} = 147.885 \quad (19-5)$$

$$D_1(q_1, \dot{q}_1, \ddot{q}_1) = 0.451\ddot{q}_1 + 11.89\dot{q}_1 + 1.104 \times \sin(q_1) + 0.033 \times \cos(q_1) \quad (20-5)$$

$$D_2(q_2, \dot{q}_2, \ddot{q}_2) = 0.226\ddot{q}_2 + 11.9\dot{q}_2 + 0.221 \times \sin(q_2) + 0.00663 \times \cos(q_2) \quad (21-5)$$

معادلات حالت بازوی اول بدین ترتیب حاصل میگردند.

$$\dot{x} = \begin{bmatrix} x_3 \\ x_5 \\ -33.35 \times \ddot{q}_1 - 87918 \times \dot{q}_1 - 81.63 \times \sin(q_1) - 2.44 \times \cos(q_1) \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 73.943 \end{bmatrix} u_1 + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} d_1. \quad (22-5)$$

$$\dot{x} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -81.63 \times \frac{\sin(x_1)}{x_1} & -2.44 \times \frac{\cos(x_1)}{x_3} & -87918 & -33.35 \end{bmatrix} \begin{bmatrix} x_1 \\ x_3 \\ x_5 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 73.943 \end{bmatrix} u_1 + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} d_1. \quad (23-5)$$

و برای بازوی دوم داریم:

$$\dot{x} = \begin{bmatrix} x_4 \\ x_6 \\ -33422 \times \ddot{q}_2 - 175983 \times \dot{q}_2 - 3268 \times \sin(q_2) - 0.98 \times \cos(q_2) \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 147885 \end{bmatrix} u_2 + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} d_2. \quad (24-5)$$

$$\dot{x} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -3268 \times \frac{\sin(x_2)}{x_2} & -0.98 \times \frac{\cos(x_2)}{x_4} & -175983 & -33422 \end{bmatrix} \begin{bmatrix} x_2 \\ x_4 \\ x_6 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 147885 \end{bmatrix} u_2 + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} d_2. \quad (25-5)$$

۵-۱-۲- خطی سازی سیستم غیر خطی با روش تاکاگی - سوگنو

جملات غیر خطی در نمایش فضای حالت بازوی اول، $z_1(t) = \frac{\sin x_1}{x_1}$ ، $z_2(t) = \frac{\cos x_1}{x_3}$ و در بازوی

دوم، $z_3(t) = \frac{\sin x_2}{x_2}$ ، $z_4(t) = \frac{\cos x_2}{x_4}$ هستند.

با فرض $x_1(t), x_2(t) \in \left[-\frac{\pi}{2}, \frac{\pi}{2}\right]$ ، $x_3(t), x_4(t) \in \left[-\frac{\pi}{2}, \frac{\pi}{2}\right] \text{ except } [-6, 6]$ داریم:

$$\dot{x}_1(t) = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -81.63 \times z_1 & -2.44 \times z_2 & -87918 & -33.35 \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_3(t) \\ x_5(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 73.943 \end{bmatrix} u_1(t) + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} d_1(t). \quad (26-5)$$

$$\dot{x}_2(t) = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -3268 \times z_3 & -0.98 \times z_4 - 175983 & -33422 \end{bmatrix} \begin{bmatrix} x_2(t) \\ x_4(t) \\ x_6(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 73943 \end{bmatrix} u_2(t) + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} d_2(t). \quad (27-\delta)$$

با محاسبه مقادیر ماکزیمم و مینیمم توابع غیرخطی، در بازه های مفروض، داریم:

$$z_1 = \frac{\sin x_1}{x_1} \Rightarrow \begin{cases} \max_{x_1, x_2} z_1 = 1.000 \\ \min_{x_1, x_2} z_1 = 0.6366 \end{cases}$$

$$z_2 = \frac{\cos x_1}{x_3} \Rightarrow \begin{cases} \max_{x_1, x_3} z_2 = 10 \\ \min_{x_1, x_3} z_2 = -0.6366 \end{cases}$$

$$z_3 = \frac{\sin x_4}{x_4} \Rightarrow \begin{cases} \max_{x_4, x_5} z_3 = 1.000 \\ \min_{x_4, x_5} z_3 = 0.6366 \end{cases}$$

$$z_4 = \frac{\cos x_4}{x_6} \Rightarrow \begin{cases} \max_{x_4, x_5} z_4 = 10 \\ \min_{x_4, x_5} z_4 = -0.6366 \end{cases}$$

و با استفاده از مقادیر مینیمم و ماکزیمم، $z_1(t)$ ، $z_2(t)$ ، $z_3(t)$ ، $z_4(t)$ را بدست می آوریم:

$$z_1(t) = \frac{\sin x_1}{x_1} = M_1(z(t)) \times 0.6366 + M_2(z(t)) \times 1 \quad (28-\delta)$$

$$z_2(t) = \frac{\cos x_1}{x_3} = N_1(z(t)) \times (-0.6366) + N_2(z(t)) \times 10 \quad (29-\delta)$$

$$z_3(t) = \frac{\sin x_4}{x_4} = S_1(z(t)) \times 0.6366 + S_2(z(t)) \times 1 \quad (30-\delta)$$

$$z_4(t) = \frac{\cos x_4}{x_6} = Q_1(z(t)) \times (-0.6366) + Q_2(z(t)) \times 10 \quad (31-\delta)$$

$$M_1(z(t)) + M_2(z(t)) = 1 \rightarrow M_1(z(t)) = 1 - M_2(z(t)) \quad (32-\delta)$$

$$N_1(z(t)) + N_2(z(t)) = 1 \rightarrow N_1(z(t)) = 1 - N_2(z(t)) \quad (33-\delta)$$

$$S_1(z(t)) + S_2(z(t)) = 1 \rightarrow S_1(z(t)) = 1 - S_2(z(t)) \quad (34-\delta)$$

$$Q_1(z(t)) + Q_2(z(t)) = 1 \rightarrow Q_1(z(t)) = 1 - Q_2(z(t)) \quad (35-\Delta)$$

$$z_1(t) = 0.6366 + 0.3634 \times M_2(z(t)) \Rightarrow \begin{cases} M_2(z(t)) = \frac{z_1(t) - 0.6366}{0.3634} \text{ "big"} \\ M_1(z(t)) = \frac{1 - z_1(t)}{0.3634} \text{ "small"} \end{cases} \quad (36-\Delta)$$

$$z_2(t) = -10.33 \times N_1(z(t)) + 10 \Rightarrow \begin{cases} N_2(z(t)) = \frac{z_2(t) - 10}{-10.33} \text{ "hi"} \\ N_1(z(t)) = \frac{0.6366 + z_2(t)}{10.33} \text{ "low"} \end{cases} \quad (37-\Delta)$$

$$z_3(t) = 0.6366 + 0.3634 \times S_2(z(t)) \Rightarrow \begin{cases} S_2(z(t)) = \frac{z_3(t) - 0.6366}{0.3634} \text{ "big"} \\ S_1(z(t)) = \frac{1 - z_3(t)}{0.3634} \text{ "small"} \end{cases} \quad (38-\Delta)$$

$$z_4(t) = -10.33 \times Q_1(z(t)) + 10 \Rightarrow \begin{cases} Q_2(z(t)) = \frac{z_4(t) - 10}{-10.33} \text{ "hi"} \\ Q_1(z(t)) = \frac{0.6366 + z_4(t)}{10.33} \text{ "low"} \end{cases}$$

اکنون یک مدل هشت قانونی با استفاده از منطق فازی، بفرم زیر نوشته میشوند:

model rule 1:

IF $z_1(t)$ is "big" and $z_2(t)$ is "hi",

then $\dot{x}(t) = A_1 x(t) + B_1 \tau(t) + d_1(t)$.

model rule 2:

IF $z_1(t)$ is "big" and $z_2(t)$ is "low",

then $\dot{x}(t) = A_2 x(t) + B_2 \tau(t) + d_1(t)$.

model rule 3:

IF $z_1(t)$ is "small" and $z_2(t)$ is "hi",

then $\dot{x}(t) = A_3x(t) + B_3\tau(t) + d_1(t)$.

model rule 4:

IF $z_1(t)$ is "small" and $z_2(t)$ is "low",

then $\dot{x}(t) = A_4x(t) + B_4\tau(t) + d_1(t)$.

model rule 5:

IF $z_3(t)$ is "big" and $z_4(t)$ is "hi",

then $\dot{x}(t) = A_5x(t) + B_5\tau(t) + d_2(t)$.

model rule 6:

IF $z_3(t)$ is "big" and $z_4(t)$ is "low",

then $\dot{x}(t) = A_6x(t) + B_6\tau(t) + d_2(t)$.

model rule 7:

IF $z_3(t)$ is "small" and $z_4(t)$ is "hi",

then $\dot{x}(t) = A_7x(t) + B_7\tau(t) + d_2(t)$.

model rule 8:

IF $z_3(t)$ is "small" and $z_4(t)$ is "low",

then $\dot{x}(t) = A_8x(t) + B_8\tau(t) + d_2(t)$.

$$A_1 = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -81.63 & -903.58 & -33.35 \end{bmatrix}, B_1 = \begin{bmatrix} 0 \\ 0 \\ 73.943 \end{bmatrix}.$$

$$A_2 = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -81.63 & -877.63 & -33.35 \end{bmatrix}, B_2 = \begin{bmatrix} 0 \\ 0 \\ 73.943 \end{bmatrix}.$$

$$A_3 = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -51.966 & -903.58 & -33.35 \end{bmatrix}, B_3 = \begin{bmatrix} 0 \\ 0 \\ 73.943 \end{bmatrix}.$$

$$A_4 = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -51.966 & -877.63 & -33.35 \end{bmatrix}, B_4 = \begin{bmatrix} 0 \\ 0 \\ 73.943 \end{bmatrix}.$$

$$A_5 = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -32.68 & -1769.63 & -33.442 \end{bmatrix}, B_5 = \begin{bmatrix} 0 \\ 0 \\ 147.885 \end{bmatrix}.$$

$$A_6 = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -32.68 & -1759.21 & -33.442 \end{bmatrix}, B_6 = \begin{bmatrix} 0 \\ 0 \\ 147.885 \end{bmatrix}.$$

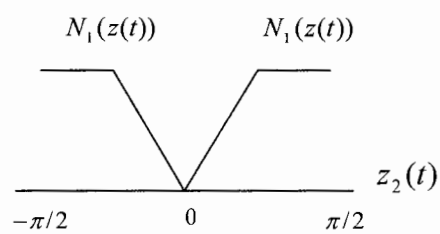
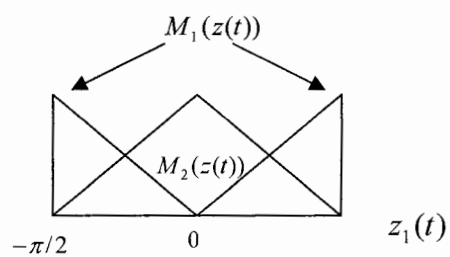
$$A_7 = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -20.8 & -1769.63 & -33.442 \end{bmatrix}, B_7 = \begin{bmatrix} 0 \\ 0 \\ 147.885 \end{bmatrix}.$$

$$A_8 = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -20.8 & -1759.21 & -33.442 \end{bmatrix}, B_8 = \begin{bmatrix} 0 \\ 0 \\ 147.885 \end{bmatrix}.$$

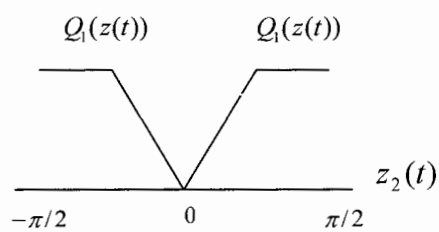
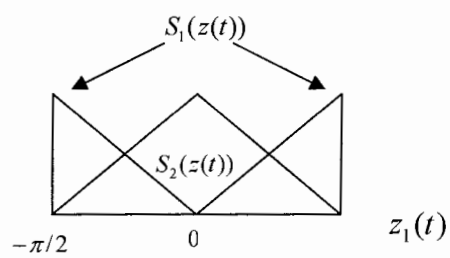
و غیرفازی سازی بدین ترتیب انجام میشود،

$$\dot{x}(t) = \sum_{i=1}^8 h_i(z(t)) [A_i x(t) + B_i \tau(t)] + d_i(t)$$

توابع عضویت بازوی اول و بازوی دوم در ذیل آمده است.



شکل ۵-۲- توابع عضویت بازوی اول



شکل ۵-۳- توابع عضویت بازوی دوم

۵-۱-۳- طراحی کنترل کننده فازي PDC برای بازوي ماهر دو رابط

قوانين PDC برای کنترل کننده فازي فوق، چنين ميگردد:

model rule 1:

IF $z_1(t)$ is "big" and $z_2(t)$ is "hi",

then $u_1(t) = F_1x(t) + d_1(t)$.

model rule 2:

IF $z_1(t)$ is "big" and $z_2(t)$ is "low",

then $u_2(t) = F_2x(t) + d_1(t)$.

model rule 3:

IF $z_1(t)$ is "small" and $z_2(t)$ is "hi",

then $u_3(t) = F_3x(t) + d_1(t)$.

model rule 4:

IF $z_1(t)$ is "small" and $z_2(t)$ is "low",

then $u_4(t) = F_4x(t) + d_1(t)$.

model rule 5:

IF $z_3(t)$ is "big" and $z_4(t)$ is "hi",

then $u_5(t) = F_5x(t) + d_2(t)$.

model rule 6:

IF $z_3(t)$ is "big" and $z_4(t)$ is "low",

then $u_6(t) = F_6x(t) + d_2(t)$.

model rule 7:

IF $z_3(t)$ is "small" and $z_4(t)$ is "hi",

then $u_7(t) = F_7 x(t) + d_2(t)$.

model rule 8:

IF $z_3(t)$ is "small" and $z_4(t)$ is "low",

then $u_8(t) = F_8 x(t) + d_2(t)$.

$$u(t) = -\sum_{i=1}^8 h_i(z(t)) F_i x(t) + d_i(t)$$

$$d_1 = 0.1 \times \sin(2t)$$

$$d_2 = 0.1 \times \cos(2t)$$

ماتریس بهره F_1, F_2 حاصل از کنترل کننده PDC ، با استفاده از جعبه ابزار LMI در مطلب چنین

حاصل می شود.

$$F_1 = \begin{bmatrix} k_1 \\ k_2 \\ k_3 \\ k_4 \end{bmatrix} = \begin{bmatrix} 110.4 & 1220.64 & 44.81 & 0 \\ 110.4 & 1185.55 & 44.81 & 0 \\ 70.28 & 1220.64 & 44.81 & 0 \\ 70.28 & 1185.55 & 44.81 & 0 \end{bmatrix}$$

$$F_2 = \begin{bmatrix} k_4 \\ k_5 \\ k_6 \\ k_7 \end{bmatrix} = \begin{bmatrix} 2.210 & 119.595 & 2.247 & 0 \\ 2.210 & 118.890 & 2.247 & 0 \\ 1.406 & 1185.55 & 2.247 & 0 \\ 1.406 & 118.890 & 2.247 & 0 \end{bmatrix}$$

و ماتریس مثبت معین p_1, p_2 بطریقی معین می شود که کلیه شرایط پایداری برقرار گردد.

$Decay Rate$ و ماتریس مثبت معین متعارفی p_1, p_2 ، چنین هستند.

$$decay \ rate1 = 9.1049 \times 10^{-5}$$

$$p_1 = \begin{bmatrix} 280.5685 & 60.1814 & 280.5685 \\ 60.1814 & 220.4903 & 67.6261 \\ 280.5685 & 67.6261 & 489.0239 \end{bmatrix}$$

$$decay \ rate2 = 9.104 \times 10^{-5}$$

$$p_2 = \begin{bmatrix} 261.4046 & 56.0706 & 261.4046 \\ 56.0706 & 609.837 & 84.69140 \\ 261.4046 & 84.6914 & 459.4787 \end{bmatrix}$$

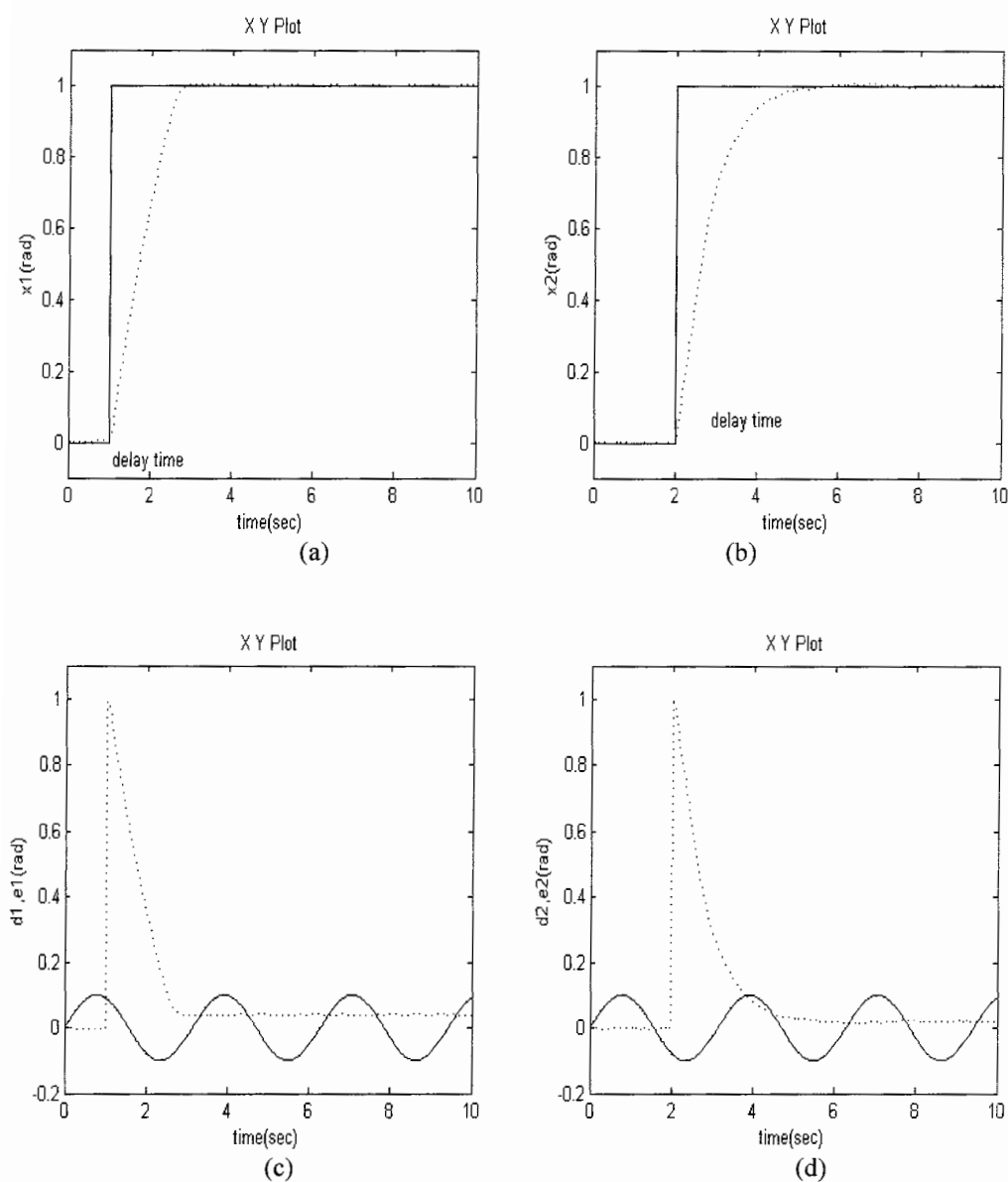
نتایج حاصل از شبیه سازی نشان میدهند که سیستم پایدار مجانبی جهانی است.

همین مراحل برای بازوهای n رابط قابل تعمیم بوده بدین ترتیب که هر سیستم را بطور مجزا و

مستقل از دیگر رابطها کنترل نموده و خروجی هر رابط برای رابط بعدی ایجاد سیگنال اغتشاش

می نماید.

۵-۲- نتایج حاصل از شبیه سازی بازوی ماهر دو رابط



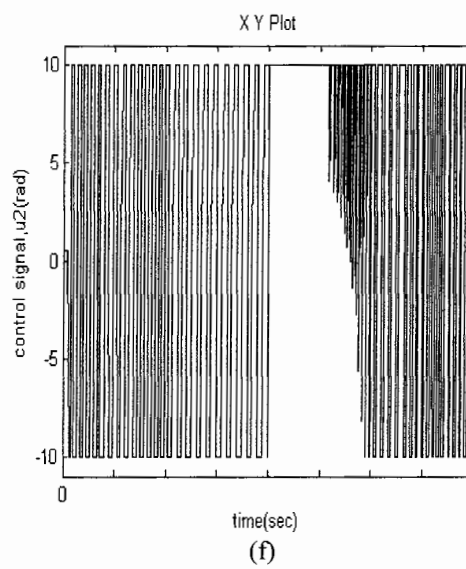
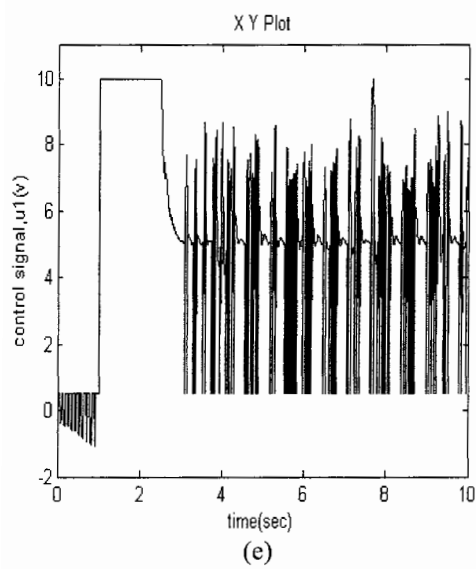
شکل ۵-۴-الف

(a) مسیر مرجع و مسیر تعقیب شده (نقطه چین) توسط بازوی اول

(b) مسیر مرجع و مسیر تعقیب شده (نقطه چین) توسط بازوی دوم

(c) اغتشاش وارد شده به سیستم و خطای تعقیب مسیر بازوی اول (نقطه چین)

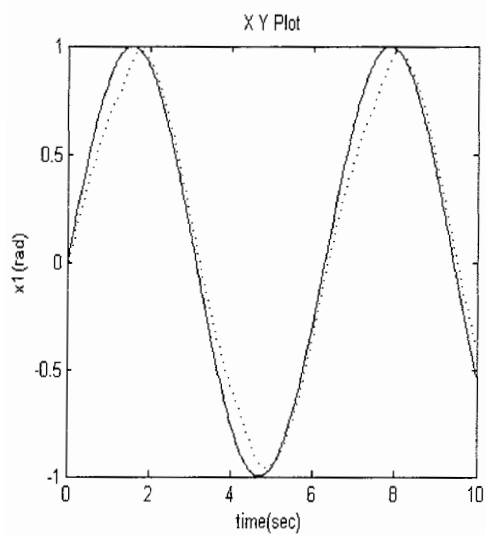
(d) اغتشاش وارد شده به سیستم و خطای تعقیب مسیر بازوی دوم (نقطه چین)



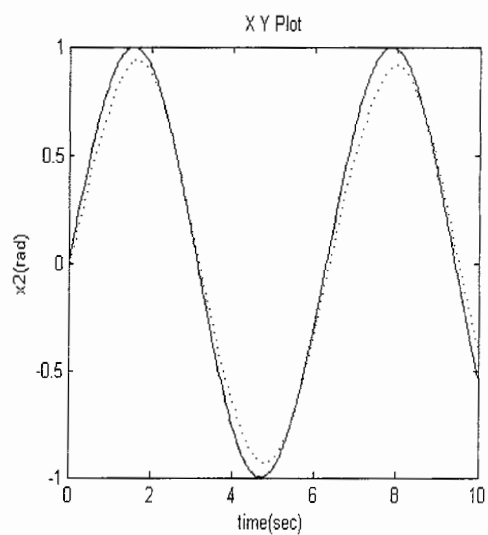
شکل ۴-۵ ب

(e) سیگنال کنترل بازوی اول

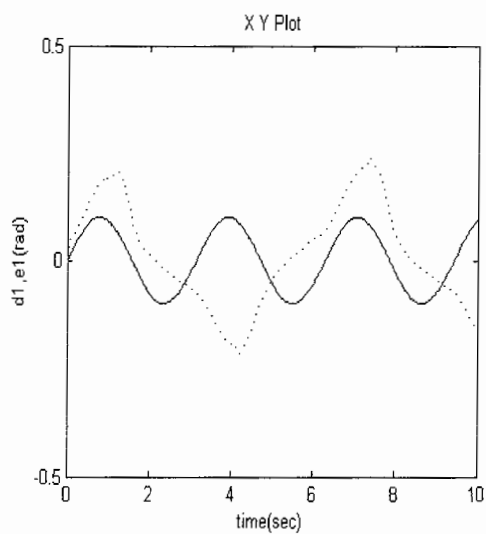
(f) سیگنال کنترل بازوی دوم



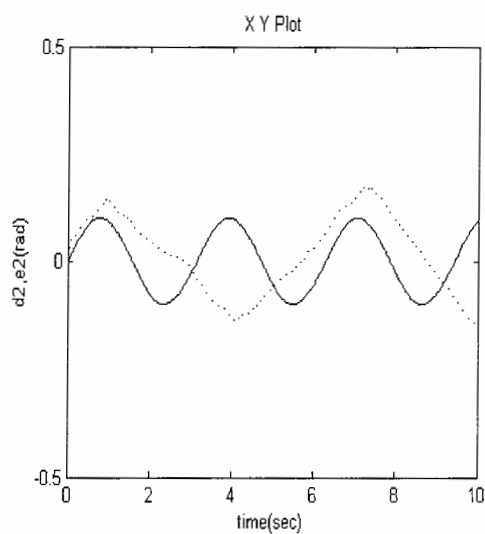
(a)



(b)



(c)



(d)

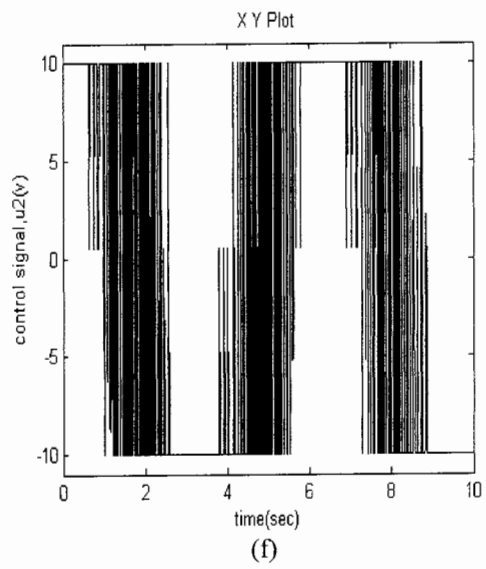
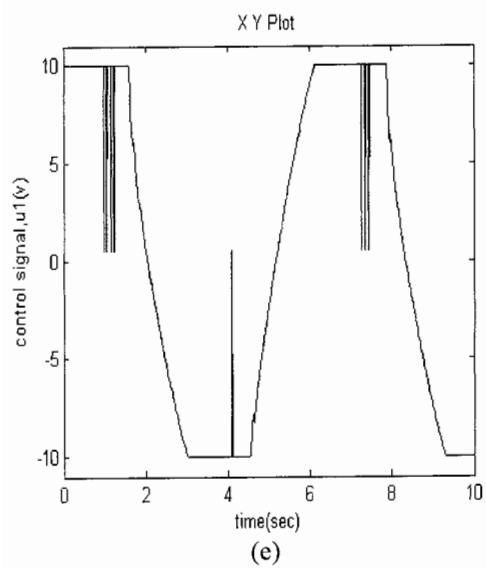
شکل ۵-۵-الف

(a) مسیر مرجع و مسیر تعقیب شده (نقطه چین) توسط بازوی اول

(b) مسیر مرجع و مسیر تعقیب شده (نقطه چین) توسط بازوی دوم

(c) اغتشاش وارد شده به سیستم و خطای تعقیب مسیر بازوی اول (نقطه چین)

(d) اغتشاش وارد شده به سیستم و خطای تعقیب مسیر بازوی دوم (نقطه چین)



شکل ۵-۵ ب

(e) سیگنال کنترل بازوی اول

(f) سیگنال کنترل بازوی دوم

فصل ششم:

نتیجه گیری و پیشنهادات

۶-۱- نتیجه گیری

در این پایان نامه، شرایط پایداری کنترل کننده فازی تاکاگی - سوگنو بررسی شده و سپس یک کنترل کننده فیدبک حالت با ساختار جبران توزیع موازی (PDC) برای پایدار سازی یک سیستم غیرخطی آورده شده است و بدنبال آن نا مساویهای ماتریس خطی (LMI) با استفاده از شرایط پایداری لیاپانف بدست آمده اند و نتایج حاصل از طراحی و شبیه سازی کنترل کننده فازی برای توضیح عملکرد سیستم رباتهای لولایی یک و دو رابط آورده شده است. نتایج حاصله با استفاده از PDC فازی و سیستمهای کنترل چند مرحله ای قابل تعمیم به بازوهای n رابط میباشد. مسایل نامساویهای ماتریس خطی با استفاده از جعبه ابزار بهینه سازی LMI در مطلب قابل حل هستند.

کنترل فازی روشی برای طراحی کنترل کننده رباست در حضور اغتشاش برای سیستمهای غیرخطی و نیز سیستمهای پیچیده است. دو نوع عمده کنترل کننده فازی وجود دارد: کنترل کننده ممدانی و دیگری کنترل کننده تاکاگی - سوگنو. اختلاف ایندو نوع کنترل کننده در خروجی آنها است بدین ترتیب که خروجی سیستمهای طراحی شده با مدل فازی تاکاگی سوگنو، بصورت مقادیر ثابت و یا مقایر خطی مدل میشوند که این مقادیر خطی روابط ورودی-خروجی محلی یک سیستم غیرخطی را نشان میدهند.

در مورد سیستمهای پیچیده تر میتوان از روش چند مرحله ایها استفاده نمود ، این روش سیستم را به صورت جزء های سری و موازی از کنترل کننده های کوچکتر تبدیل می نماید ، این عمل باعث کاهش تعداد قوانین مورد نیاز و نیز افزایش سرعت محاسبات خواهد شد.

نتایج حاصل از شبیه سازی سیستم بازوی دو رابط نشان میدهد که بازوی اول دارای تاخیر $t=2.45sec$ و بازوی دوم پس از $t=5.2sec$ به حالت پایدار مورد نظر میرسند، دلیل این تاخیر در سیستم دینامیک موتور و نسبت دور موتور می باشد.

۶-۲- پیشنهادات

استفاده از روش فازی تاکاگی- سوگنو و تحلیل سیستمهای چند مرحله ای برای سیستم بازوی لولایی یک و دو رابط، به نتایج رضایت بخشی منجر شده است، بدین ترتیب میتوان:

- روش طراحی فازی بیان شده را گسترش داده و آنرا با هر یک از روشهای طراحی مطرح شده در فصل اول نظیر کنترل امپدانس، شبکه عصبی و کنترل تطبیقی بکاربرد همینطور میتوان سیستم غیر خطی را با یک کنترل کننده PID یا PD پایدار نمود.
- روش کنترلی فوق را برای کلیه سیستمهای غیر خطی نظیر سیستمهای قدرت و ... استفاده نمود.

از این دسته میتوان به خطایابی روتور و ... اشاره کرد. خطایابی روتور با استفاده از شبکه های عصبی نتایج رضایت بخشی را در پی داشته اند با استفاده از روش فازی میتوان به نتایج مشابهی دست یافت.

- این روش برای خطایابی و اصلاح عملکرد دستگاه های پزشکی مثلا *ECG*, *EEG*, ... بسیار کارآمد خواهد بود.

- مقایسه ای میان نتایج حاصل از شبیه سازی سیستم با مدل فازی تاکاگی- سوگنو و ممدانی بعمل آورد، ایندو کنترل کننده قابل تبدیل به یکدیگر بوده و بدین ترتیب میتوان نتایج مشابهی از شبیه سازی بدست آورد.

- به ازای شرایط اولیه متفاوت ، کنترل کننده *PDC* فازی را طراحی نموده و پایدار نمود.

- در این پایان نامه، سیستم بازوی لولایی بصورت یک و دو رابط مستقل، کنترل شده است این در حالیکه میتوان همین سیستم را بطور یکپارچه در نظر گرفته و پس از بدست آوردن معادلات ترکیبی بازو و موتور ، سیستم فازی را بصورت چند متغیره طراحی نمود و نتایج حاصل را با سیستم رابط مستقل مقایسه نمود.

Abstract

In this proposal design of robust fuzzy controller for nonlinear systems was researched with PDC fuzzy and then gived method was used for control of one link and two link arm driven by DC motors.

Many procedures was used for control of skilled arms such as, PID control, step by step control, feed forward control, ...,and fuzzy control.

Fuzzy control is one of the methods that in last decay was used for control of nonlinear systems.

For design of a fuzzy controller first it is necessary to find combined dynamics of DC motor and arm equations with using of T-S model and then a state space feedback control in form of PDC was entracted based on designed fuzzy controller for decreasing of tracking error and stability of fuzzy system.

Using of lyapunov theory, sufficient conditions for assurance of robust stability for considered system was achieved by a Linear Matrix Inequality (LMI).

utilization of muli stage systems is one of procedures that causes decreasing of fuzzy rules and with practice of this technique, a complicated system can be converting to several simplified and segregated systems.

References

- [1] Soo Yeong Yi and Myung Jin Chung. "Robustness of Fuzzy Logic Control for an Uncertain Dynamic System".1998. IEEE .vol 6.no 2.
- [2] Hua O.Wang .1996. "An Approach to Fuzzy Control of Nonlinear Systems: Stability and Design". IEEE.vol 4.no 1. Kazuo Tanaka, Member, IEEE, and Michael F.Griffin, Member, IEEE.
- [3] Chin-Tzong Pang and Sy-Ming Guu. "Sufficient Conditions for the Stability of Linear Takagi–Sugeno Free Fuzzy Systems". 2003. IEEE transactions on fuzzy systems, vol. 11, NO. 5,
- [4] K.Tanaka, T.Hori, K.Yamafuji, and H.O.Wang, "An Integrated Algorithm of Fuzzy Modeling and Controller Design for Nonlinear Systems,"1999.IEEE International Conference on Fuzzy Systems, Vol. 2, Seoul, August 1999, PP.887-892.
- [5] K.Tanaka, T.Hori, K.Yamafuji, and H.O.Wang, "An Integrated Fuzzy control System Design for Nonlinear Systems," .Dec.1999. 38Th IEEE Conference on Decision and Control, Phoenix, PP.4349-4354.
- [6] Devendra P. Garg Manish Kumar, "Genetic algorithm based PD control and fuzzy logic control of a two link", 2002. ASME International Mechanical Engineering Congress & Exposition
- [7] Somwang Arisariyawong and Siam Charoenseang, "Reducing steady-state errors of a direct drive robot using neurofuzzy control". 2001. Second Asian Symposium on Industrial Automation and Robotics BITEC, Bangkok, Thailand.
- [8] Bernhard Sprenger, Ladislav Kucera, and Safer Mourad, "Balancing of an Inverted Pendulum with a SCARA Robot".1998. Institute of robotics.

- [9] *Robust Control Toolbox. For Use with MATLAB.*
- [10] James M. Adams. "Proposal for Dissertation Research of Optimized Multi-Stage Fuzzy Logic Control for a Multi-Link Robot". 2002 B.S.E.E (Michigan Technological University) 1986. M.S.E.E. (Air Force Institute of Technology) 1988.
- [11] Ming Co. "Motion Control Systems Development for Bearcat II". 2001. Department of ech .ind .and nuc. Eng.
- [12] Devendra P. Garg Manish Kumar, "Multiple robot control using force/torque and vision sensors". Department of Mechanical Engineering and Materials Science Duke University, USA.
- [13] "Inverted Pendulum"
Institute of Automation, University of Bremen Advanced Control Systems Laboratory, University of Bremen, Institute of Automation
- [14] M. L. Hadjili, V. Wertz, G. "Local and global state-feedback control design for Takagi-Sugeno fuzzy systems". Scorletti. University_e de Caen. 6, France.
- [15] Adams J.M., and Rattan, K.S. "Control of a Direct Drive Robot using Multi-Stage PID". 2001. Intelligent Proceedings of the 44th IEEE Midwest Symposium on Circuits and Systems, MWSCAS. pp 543-546.
- [16] Craig R. Carignan, David L. Akin. "Achieving impedance objectives in robot teleportation". University of Maryland
- [17] Joongseon Joh, Ye-Haw Chen, Member, IEEE, and Reza Langari, "On the Stability Issues of Linear". 1998. Takagi-Sugeno Fuzzy Models, IEEE, VOL. 6, NO.3.

- [18] Bor-Sen Chen, Huey-Jian Uang, and Chung-Shi Tseng, "Robust Tracking Enhancement of Robot Systems Including Motor Dynamics: A Fuzzy-Based Dynamic Game Approach".1998. *IEEE transactions on fuzzy systems*, vol. 6, NO. 4
- [19] Chung-Shi Tseng, Bor-Sen Chen, and Huey-Jian Uang. "Fuzzy Tracking Control Design for Nonlinear".2001. *IEEE transactions on fuzzy systems*, VOL. 9, NO. 3. *IEEE*
- [20] Kazuo Tanaka Hua O.Wang "FUZZY CONTROL SYSTEMS DESIGN and ANALYSIS".
- [21] Bor-sen chen,Hoey-jian Uang,chung-shi Tseng, " Robust Trackinkg Enhancement of Robot Systems Including Motor Dynamics: A Fuzzy Based Dynamic Game Approach".