

به نام خداوند جان و خرد
کزین برتر اندیشه بر نگذرد



دانشگاه صنعتی شاهرود
دانشکده برق و رباتیک

پایان نامه کارشناسی ارشد
برق - کنترل

ملاحظه حالات تکین در کنترل ربات

استاد راهنما:

آقای دکتر محمد مهدی فاتح

تهیه و تنظیم:

محمد رضا حسینی معصوم

اسفند ۸۶

پیشکش :

پدر شکيبا و دلسوزم که به من خوب زيستن را آموخت.

برادرانم که با آنها زندگي کرده ام.

محمد رضا شجریان خسرو آواز ایران که آوازش نوای زندگي است.

به ويژه مادرم که هر چه دارم از اوست.

و نامش زنده باد.

با سپاس از :

جناب آقای دکتر محمد مهدی فاتح ، استاد راهنمای اینجانب ، که این پایان نامه جز با راهنمایی های ارزشمند ایشان شکل نمی گرفت.

جناب آقای دکتر حداد ظریف ، مدیر گروه کنترل دانشکده برق ، که از هیچ کمکی دریغ نکردند.

همه اساتید و آموزگاران که از آنها آموخته ام.

آقایان مزدک تیمور تاشلو، حمید رضا مدرس و اسماعیل بنی سعید دوستانی که مرا در انجام پایان نامه یاری رساندند.

چکیده

در این پایان نامه روش جدیدی برای کنترل ربات در نقاط تکین ارائه میشود. در این روش جدید با بررسی موقعیت ربات از ناحیه تکین با اطلاع شده و سپس کنترل ربات در همسایگی نقاط تکین از وضعیت کنترل دکارتی به وضعیت کنترل مفصلی منتقل می شود. این پایان نامه به کنترل ربات در نقاط تکین می پردازد و روشهای کنترلی وارون ژاکوبین، معکوس مجازی ژاکوبین و ترانهاده ژاکوبین برای عبور از نقاط تکین بررسی می شوند.

کنترل ربات به دو صورت کنترل فضای مفصلی و کنترل فضای دکارتی قابل انجام است. مزیت کنترل در فضای کار، استفاده از اطلاعات موقعیت مجری نهایی است که موجب افزایش کارایی می شود ولی با مشکل نقاط تکین روبرو هستیم. در مقابل، کنترل در فضای مفصلی، مشکل نقاط تکین را ندارد و اجرای کنترل آسان است. اما به دلیل عدم اطلاع از وضعیت مجری نهایی ممکن است کارایی سیستم کنترل پایین بیاید. برای کنترل موقعیت ربات در فضای کار لازم است که از قبل مسیر مطلوب در فضای کار و سپس در فضای مفصلی را تعیین نماییم. بررسی نقاط تکین در تعیین مسیر ربات بسیار مهم است. زیرا کنترل ربات در این نقاط با مشکل مواجه می شود. در نقاط تکین ممکن است که سیستم بعضی از درجات آزادی خود را از دست بدهد. در این صورت اعمال کنترل ربات در فضای کار در این درجات آزادی میسر نبوده و موجب بالا رفتن ولتاژ موتورها می شود. در نتیجه ممکن است به موتورها و پس از آن به ساختار مکانیکی ربات آسیب برسد. بنابراین سعی بر آن است که در کنترل ربات در فضای دکارتی، ربات از نقاط تکین عبور نکند. قابل توجه آن که کنترل ربات در فضای مفصلی عاری از نقاط تکین است.

کلمات کلیدی: وضعیت تکین ، کنترل فضای دکارتی ، ژاکوبین ربات ، ماتریس معکوس مجازی ژاکوبین

فهرست

صفحه

عنوان

فصل اول مقدمه ۱

فصل دوم حرکت ربات

- ۱-۲- مطالعه ژاکوبین ۶
- ۲-۲- سینماتیک مستقیم ۱۲
- ۳-۲- معادله دینامیکی ربات ۱۶

فصل سوم کنترل ربات

- ۱-۳- روش خطی سازی فیدبکی ۲۶
- ۲-۳- روش دینامیک وارون در فضای مفصلی ۲۷
- ۳-۳- روش دینامیک وارون در فضای دکارتی ۳۴

فصل چهارم حالات تکین

- ۱-۴- حالات تکین در ربات آرنجی صفحه ای دو رابط ۴۴
- ۲-۴- نتایج روش دینامیک وارون مفصلی برای مسیر تکین ۴۷
- ۳-۴- نتایج روش دینامیک وارون دکارتی با J^T برای مسیر تکین ۵۰
- ۴-۴- ماتریس معکوس مجازی ۵۴
- ۵-۴- نتایج روش دینامیک وارون دکارتی با J^+ برای ردگیری مسیر تکین ۵۶

فصل پنجم روش کنترل سازگار با نقاط تکین

- نتایج روش سازگار با نقاط تکین برای مسیر تکین ۷۲

نتیجه گیری ۷۹

منابع ۸۰

فهرست شکل ها

فصل دوم

- شکل ۱-۲ بردار جابجایی مفاصل ۸
- شکل ۲-۲ مختصات دو مفصل مجاور ۹
- شکل ۳-۲ شمای ربات دو رابط لولایی ۱۳
- شکل ۴-۲ شمای کلی ربات صفحه ای دو رابط ۱۵
- شکل ۵-۲ شمای کلی ربات صفحه ای دو رابط از دید دینامیکی ۲۱

فصل سوم

- شکل ۱-۳ شمای کلی سیستم کنترل ربات با روش دینامیک وارون در فضای مفصلی ۲۹
- شکل ۲-۳ مسیری غیر تکین در فضای دکارتی ۳۰
- شکل ۳-۳ مسیر مطلوب مفصلی غیر تکین ۳۰
- شکل ۴-۳ خطای روش دینامیک وارون در فضای مفصلی برای مسیر غیر تکین ۳۱
- شکل ۵-۳ سرعت مفاصل در روش دینامیک وارون در فضای مفصلی برای مسیر غیر تکین ۳۲
- شکل ۶-۳ ولتاژ مفاصل در روش دینامیک وارون در فضای مفصلی برای مسیر غیر تکین ۳۲
- شکل ۷-۳ گشتاور مفاصل در روش دینامیک وارون در فضای مفصلی برای مسیر غیر تکین ۳۳
- شکل ۸-۳ شمای کلی سیستم کنترل دینامیک وارون در فضای دکارتی ۳۵
- شکل ۹-۳ مسیری غیر تکین در فضای دکارتی ۳۵
- شکل ۱۰-۳ مسیر مطلوب غیر تکین در فضای دکارتی ۳۶
- شکل ۱۱-۳ خطای ردگیری مسیری غیر تکین در روش دینامیک وارون دکارتی ۳۶
- شکل ۱۲-۳ سرعت مجری نهایی در روش دینامیک وارون دکارتی ۳۷
- شکل ۱۳-۳ سرعت مفاصل برای مسیری غیر تکین در روش دینامیک وارون در فضای دکارتی ۳۸
- شکل ۱۴-۳ ولتاژ مفاصل برای مسیری غیر تکین در روش دینامیک وارون در فضای دکارتی ۳۸
- شکل ۱۵-۳ گشتاور مفاصل برای مسیری غیر تکین در روش دینامیک وارون در فضای دکارتی ۳۹
- شکل ۱۶-۳ دترمینان ماتریس ژاکوبین برای مسیر غیر تکین در روش دینامیک وارون در فضای دکارتی ۳۹

فصل چهارم

شکل ۴-۱ شمای پیکربندی های تکین ۴۵

شکل ۴-۲ مسیرهای تکین گسترده و تا شده ۴۵

شکل ۴-۳ مسیر تکین که از دو نقطه تکین مرزی و مرکزی عبور می کند ۴۶

شکل ۴-۴ مسیر تکین مطلوب مفصلی به روش دینامیک وارون مفصلی ۴۷

شکل ۴-۵ خطای ردگیری زوایای مفاصل برای مسیر تکین در روش دینامیک وارون مفصلی ۴۸

شکل ۴-۶ سرعت مفاصل در طی مسیر تکین در روش دینامیک وارون مفصلی ۴۸

شکل ۴-۷ ولتاژ مفاصل در طی مسیر تکین در روش دینامیک وارون مفصلی ۴۹

شکل ۴-۸ گشتاور مفاصل در طی مسیر تکین در روش دینامیک وارون مفصلی ۴۹

شکل ۴-۹ مسیر مطلوب تکین در فضای دکارتی ۵۰

شکل ۴-۱۰ دترمینان ماتریس ژاکوبین طی مسیر تکین در روش دینامیک وارون فضای دکارتی با استفاده از J^T ۵۱

شکل ۴-۱۱ خطای ردگیری دکارتی مسیر تکین در روش دینامیک وارون دکارتی با استفاده از J^T ۵۱

شکل ۴-۱۲ سرعت مفاصل طی مسیر تکین در روش دینامیک وارون دکارتی با استفاده از J^T ۵۲

شکل ۴-۱۳ سرعت مجری نهایی طی مسیر تکین در روش دینامیک وارون دکارتی با استفاده از J^T ۵۲

شکل ۴-۱۴ ولتاژ مفاصل طی مسیر تکین در روش دینامیک وارون دکارتی با استفاده از J^T ۵۳

شکل ۴-۱۵ گشتاور مفاصل در طی مسیر تکین در روش دینامیک وارون دکارتی با استفاده از J^T ۵۳

شکل ۴-۱۶ مسیر مطلوب تکین در فضای دکارتی ۵۷

شکل ۴-۱۷ خطای ردگیری در طی مسیر تکین در روش دینامیک وارون دکارتی با استفاده از J^+ ۵۷

شکل ۴-۱۸ خطای ردگیری در نقطه تکین اول در روش دینامیک وارون دکارتی با استفاده از J^+ ۵۸

شکل ۴-۱۹ خطای ردگیری در نقطه تکین دوم در روش دینامیک وارون دکارتی با استفاده از J^+ ۵۸

شکل ۴-۲۰ افزایش بی رویه سرعت مفصل اول در نقطه تکین اول در روش دینامیک وارون دکارتی با استفاده از J^+ ۵۹

شکل ۴-۲۱ افزایش بی رویه سرعت مفصل اول در نقطه تکین دوم در روش دینامیک وارون دکارتی با استفاده از J^+ ۵۹

شکل ۴-۲۲ سرعت مجری نهایی در نقطه تکین اول در روش دینامیک وارون دکارتی با استفاده از J^+ ۶۰

شکل ۴-۲۳ سرعت مجری نهایی در نقطه تکین دوم در روش دینامیک وارون دکارتی با استفاده از J^+ ۶۰

شکل ۴-۲۴ ولتاژ مفاصل در نقطه تکین اول در روش دینامیک وارون دکارتی با استفاده از J^+ ۶۱

شکل ۴-۲۵ ولتاژ مفاصل در نقطه تکین دوم در روش دینامیک وارون دکارتی با استفاده از J^+ ۶۱

شکل ۴-۲۶ گشتاور مفاصل در نقطه تکین اول در روش دینامیک وارون دکارتی با استفاده از J^+ ۶۲

شکل ۴-۲۷ گشتاور مفاصل در نقطه تکین دوم در روش دینامیک وارون دکارتی با استفاده از J^+ ۶۲

شکل ۴-۲۸ ولتاژ مفاصل در نقطه تکین دوم در روش دینامیک وارون دکارتی با استفاده از J^+ ۶۳

شکل ۴-۲۹ دترمینان ماتریس ژاکوبین در طی مسیر تکین در روش دینامیک وارون دکارتی
 با استفاده از J^+ برای مسیر تکین با نقطه شروع مفصلی به صورت $\begin{bmatrix} 0 \\ 0.1 \end{bmatrix} rad$ ۶۴

- شکل ۴-۳۰ خطای ردگیری در طی مسیر تکین در روش دینامیک وارون دکارتی
۶۵. $\begin{bmatrix} 0 \\ 0.1 \end{bmatrix}$ rad استفاده از J^+ برای مسیر تکین با نقطه شروع مفصلی به صورت
- شکل ۴-۳۱ سرعت مجری نهایی در طی مسیر تکین در روش دینامیک وارون دکارتی
۶۵. $\begin{bmatrix} 0 \\ 0.1 \end{bmatrix}$ rad استفاده از J^+ برای مسیر تکین با نقطه شروع مفصلی به صورت
- شکل ۴-۳۲ سرعت مفاصل در آغاز مسیر در روش دینامیک وارون دکارتی
۶۶. $\begin{bmatrix} 0 \\ 0.1 \end{bmatrix}$ rad استفاده از J^+ برای مسیر تکین با نقطه شروع مفصلی به صورت
- شکل ۴-۳۳ سرعت مفاصل در نقطه تکین دوم با روش دینامیک وارون دکارتی
۶۶. $\begin{bmatrix} 0 \\ 0.1 \end{bmatrix}$ rad استفاده از J^+ برای مسیر تکین با نقطه شروع مفصلی به صورت
- شکل ۴-۳۴ ولتاژ مفاصل در آغاز مسیر با روش دینامیک وارون دکارتی
۶۷. $\begin{bmatrix} 0 \\ 0.1 \end{bmatrix}$ rad استفاده از J^+ برای مسیر تکین با نقطه شروع مفصلی به صورت
- شکل ۴-۳۵ ولتاژ مفاصل در نقطه تکین مرکزی با روش دینامیک وارون دکارتی
۶۷. $\begin{bmatrix} 0 \\ 0.1 \end{bmatrix}$ rad استفاده از J^+ برای مسیر تکین با نقطه شروع مفصلی به صورت
- شکل ۴-۳۶ گشتاور مفاصل در آغاز مسیر با روش دینامیک وارون دکارتی
۶۸. $\begin{bmatrix} 0 \\ 0.1 \end{bmatrix}$ rad استفاده از J^+ برای مسیر تکین با نقطه شروع مفصلی به صورت
- شکل ۴-۳۷ گشتاور مفاصل در نقطه تکین مرکزی با روش دینامیک وارون دکارتی
۶۹. $\begin{bmatrix} 0 \\ 0.1 \end{bmatrix}$ rad استفاده از J^+ برای مسیر تکین با نقطه شروع مفصلی به صورت

فصل پنجم

- شکل ۵-۱ شمای کلی سیستم کنترل سازگار با نقاط تکین ۷۱
- شکل ۵-۲ دترمینان ماتریس ژاکوبین ربات در طی مسیر تکین در روش سازگار با نقاط تکین ۷۲
- شکل ۵-۳ مسیر دکارتی مطلوب تکین ۷۳
- شکل ۵-۴ خطای ردگیری مجری نهایی در مسیر تکین با روش سازگار با حالات تکین ۷۳
- شکل ۵-۵ سرعت مجری نهایی در طی مسیر تکین با روش سازگار با حالات تکین ۷۴
- شکل ۵-۶ سرعت مفاصل در طی مسیر تکین در روش سازگار با حالات تکین ۷۴
- شکل ۵-۷ ولتاژ مفاصل در طی مسیر تکین با روش سازگار با حالات تکین ۷۵
- شکل ۵-۸ گشتاور مفاصل در طی مسیر تکین با روش سازگار با حالات تکین ۷۶

فصل اول

مقدمه

مقدمه

کارهای ربات در فضای کار انجام می شود در حالی که یک ربات در مختصات مفصلی کنترل می شود. هنگامی که یک ربات برای اجرای یک حرکت یا اعمال یک نیروی مشخص فرمان داده می شود، فرمان باید در مختصات مفصلی به محرکه ها منتقل شود.

در فضای کاری ربات جهاتی از حرکت وجود دارند که حرکت کوچکی را در فضای مفصلی می طلبد، همچنین جهاتی از نیروهای اعمالی هستند که کمترین گشتاور را در فضای مفصلی تولید می کنند. همچنین در مقابل، جهاتی از حرکت و نیرو در فضای کاری وجود دارند که حرکت و گشتاور زیادی را در فضای مفصلی تولید می کنند. از این رو ما در کنترل ربات باید به این نکته توجه کنیم که حرکت ربات در فضای کاری تا جایی که امکان دارد کمترین حرکت را در فضای مفصلی تولید کند. البته از دید دیگر هم می توان به این موضوع نگرست و آن این که ما اگر مجاز هستیم مسیر هایی را انتخاب کنیم که کمترین انرژی جنبشی یا همان حرکت را در فضای مفصلی تولید کند و البته همان کار مورد انتظار را انجام دهد.

از مواردی که در کنترل ربات به مشکل افزایش بی رویه حرکت در فضای مفصلی بر می خوریم نقاط تکین در ربات است. نقاط تکین بر حسب جهت حرکت مجری نهایی (نقطه انتهایی) ربات و نیز محدوده فضای کاری ربات انواع مختلفی دارند و هر کدام اثرات خود را بر مفاصلی از ربات دارند. [۱] و [۲] و [۳] در فصل ۴ این پایان نامه نقاط تکین معرفی می شوند و به تفصیل درباره آنها بحث می شود.

مشکل ردگیری مسیر دکارتی در نزدیکی نقاط تکین اولین بار ۳۶ سال پیش مطرح و بررسی شد [۴]. از آن زمان تلاش های زیادی برای طراحی مکانیکی ربات به طوری که در طول مسیر از نقطه تکین عبور نکند انجام شد [۵]. محدودیت گستره حرکت ربات به طوری که نقاط تکین از گستره حرکتی ربات حذف شوند تا جایی قابل قبول است، اما باید توجه داشت که این کار باعث کاهش فضای کاری ربات و کاهش مانور آن می شود. از طرفی به تازگی مکانیزم های جدیدی از قبیل ربات های موازی معرفی شده اند که مجموعه ای از نقاط تکین به نسبت پیچیده ای دارند [۶] و [۷]

و [۸]. از این رو به الگوریتم های کنترلی ای نیاز داریم که بتواند پایداری ردگیری را در نزدیکی نقطه تکین و یا حتی هنگام عبور از نقطه تکین حفظ کند .

روش های گوناگونی تا کنون برای حل مشکل ردگیری مسیر ربات در فضای دکارتی و در نقاط تکین توسط محققین پیشنهاد شده است که برخی از آنها هنوز در عمل اجرا نشده اند و در حد نظریات محکم علمی هستند.

در برخی از این روش ها معیاری برای ربات تعریف می شود که این معیار بستگی مستقیم به سرعت و انرژی جنبشی مفاصل ربات دارد و این معیار نباید از حد معینی در مسیر حرکت ربات بیشتر شود. هرگاه به نقطه تکینی در طول مسیر برسیم از آن جایی که سرعت مفاصل در این نقطه بیشتر می شود، معیار مذکور نیز از حد مجاز تخطی کرده و ناگزیر برای کم شدن این معیار تعریف شده ربات مسیر دیگری را در پیش می گیرد. در این روش از نقطه تکین عبور نمی کنیم بلکه آن را در مسیر ربات دور می زنیم [۹].

یکی از روش ها معروف به روش فضای تهی است که در ربات هایی که ماتریس ژاکوبین آن ها دارای کاهش رتبه هستند یا به عبارتی ماتریس ژاکوبین آنها مربعی نیست و در اصطلاح به آنها ربات های زاید می گویند به کار می رود و از آن جایی که ربات های غیر زاید نیز در نقاط تکین دچار کاهش رتبه می شوند می توان از روش فضای تهی برای حل مشکل نقاط تکین آنها استفاده کرد. خلاصه این روش این است که جزء رتبه کامل ماتریس ژاکوبین ربات را از جزء زاید آن برای استفاده در کنترلر جدا می کنند [۱۰] و [۱۱].

از آن جا که در ترمینان ماتریس ژاکوبین ربات در نقاط تکین صفر است و ماتریس ژاکوبین ربات معکوس ناپذیر می شود لذا روش هایی پیشنهاد شد که از این معکوس ناپذیری را حل کند. از جمله این روش ها استفاده از ترانهاده ماتریس ژاکوبین به جای معکوس ماتریس ژاکوبین ربات است [۱۲-۱۴]. همچنین از ماتریس معکوس مجازی ژاکوبین ربات به جای معکوس ماتریس ژاکوبین استفاده شده است [۱۵، ۱].

روش دیگر افزودن یک مفصل به ربات دارای درجه اضافی حرکت است. که البته می توان این روش را برای ربات های بدون درجه اضافی حرکت نیز به کار برد تا مشکل ردگیری این ربات ها در نقاط تکین حل شود. البته این روش ساختار ربات را از لحاظ پیکربندی مفاصل تغییر می دهد [۱۶].

روش دیگر استفاده از روش نرم حد اقل وزنی و نرم حداقل مربعات برای سرعت مفاصل است. به این صورت که در بردار سرعت مفاصل و نیز ماتریس ژاکوبین ربات ماتریس های وزنی ای ضرب می کنیم تا به هنگام رسیدن به نقاط تکین سرعت مفاصل به کمترین حد خود برسد [۱۷] و [۱۸].

روش هایی موسوم به بهینه سازی محلی و کلی نیز در کنترل ربات برای رهایی از مشکل نقاط تکین مورد استفاده قرار گرفته اند. در روش بهینه سازی محلی ماتریس اینرسی ربات را با ضرب در ماتریس ژاکوبین معکوس مجازی ربات وزن دار می کنیم زیرا در نقاط تکین بسته به نوع نقطه تکین سرعت برخی از مفاصل و در نتیجه مقدار برخی از درایه های ماتریس ژاکوبین افزایش می یابد و اثر خود را در ماتریس وزنی اینرسی توسط ماتریس ژاکوبین نشان می دهد و از این رو از

روش های کنترل بهینه می توان به کنترل ربات پرداخت. روش بهینه سازی کلی نیاز به سینماتیک وارون دقیق دارد که لازمه آن به دست آوردن مشتقات زمانی چندین مرتبه متغیر ها و همچنین ماتریس ژاکوبین دقیق است. از این رو محاسبات این روش طولانی تر و نیز پیچیده تر می باشد هر چند که بهینه سازی کلی دارای پاسخی بهتر و با پایداری بیشتر در تمام طول مسیر حرکت ربات می باشد [۱۹].

روش دیگر شناسایی موارد تکین ماتریس ژاکوبین ربات به روش تجزیه مقادیر تکین (SVD) و استفاده از آن در کنترل چند متغیره است [۱] و [۲۰].

به تازگی از روش های ژنتیک الگوریتم و کنترل فازی نیز در کنترل ربات بهره گرفته شده است که در آنها نیز راه حل هایی برای مشکل نقاط تکین پیشنهاد شده است [۲۱] و [۲۲].

هدف ما معرفی روش های کنترلی ای است که توانایی ردگیری در نزدیکی نقاط تکین را با کمترین خطا داشته باشند .

در فصل دوم ما حرکت ربات و روابط حاکم بر آن را بررسی می کنیم . در فصل سوم به بحث کنترل ربات می پردازیم . در بخش چهارم حالات تکین را تعریف می کنیم و نتایج روش های کنترلی رایج ربات را در این حالات بررسی می کنیم. در فصل پنجم کنترلر سازگار با حالات تکین ربات را بررسی می کنیم و در انتها به برآیند مطالب می پردازیم .

فصل دوم

معادلات حرکت ربات

۲

معادلات حرکت ربات

۲-۱- ژاکوبین

ژاکوبین موارد زیر را تعیین می کند [۲۴] و [۲۵].

- (۱) رابطه سرعت خطی و زاویه ای مجری نهایی با سرعت مفاصل را تعیین می کند .
- (۲) تعیین مسیرهای مختصاتی در فضای کاری و فضای مفصلی
- (۳) رابطه نیروها و گشتاورهای مجری نهایی با گشتاورهای متغیرهای مفصلی را ارائه می دهد .
- (۴) تعیین حالات *sin gular* (تکین) ربات
- (۵) تعیین (طراحی) مسیرهای زمانی

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

که در آن J ماتریس ژاکوبین ربات، \dot{q} بردار متغیرهای مفصلی و \dot{x} بردار سرعت های مجری نهایی است. معادله (۱) به صورت زیر بیان می شود.

$$\begin{bmatrix} v_0^n \\ \omega_0^n \end{bmatrix} = \begin{bmatrix} J_v \\ J_\omega \end{bmatrix} [\dot{q}] \quad (2-2)$$

که در آن q بردار متغیرهای مفصلی است که در مفاصل دورانی θ و در مفاصل کشویی d است . روش محاسبه ژاکوبین در ادامه می آید :

سرعت زاویه ای [۲۳] و [۲۴] :

$$\omega_0^n = \dot{q}_1 k + R_0^1 \dot{q}_2 k + R_0^2 \dot{q}_3 k + \dots + R_0^{i-1} \dot{q}_i + \dots + R_0^{n-1} \dot{q}_n k \quad (3-2)$$

که در آن ω_0^n سرعت زاویه ای مفصل n ام نسبت به پایه است .

اما برای سرعت زاویه ای و ماتریس دوران روابطی به صورت زیر برقرار است [۲۳] و [۲۴]:

$$\omega_{i-1}^i = \dot{q}_i k \quad z_{i-1} = R_0^{i-1} k \quad (۴-۲)$$

که در آن ω_{i-1}^i سرعت زاویه ای مفصل i ام نسبت به مفصل $i-1$ ، \dot{q}_i مشتق متغیر مفصلی ،
 R_0^{i-1} ماتریس دوران مفصل $i-1$ نسبت به پایه است.

از معادله بالا سرعت زاویه ای مفصل n نسبت به پایه به صورت زیر به دست می آید .

$$\omega_0^n = [z_0 z_1 \cdots z_{i-1} \cdots z_{n-1}] \begin{bmatrix} \dot{q}_1 \\ \vdots \\ \dot{q}_n \end{bmatrix} \quad (۵-۲)$$

که در آن \dot{q}_i مشتق متغیر مفصلی و z_n بردار z مفصل n ام است .

آنگاه آن را به فرم زیر می توان نوشت

$$\omega_0^n = J_\omega \dot{q} \Rightarrow J_\omega = [z_0 z_1 \cdots z_{n-1}] \quad (۶-۲)$$

که در آن z_n راستای z در مختصات مفصل n و J_ω بخش چرخشی ژاکوبین است .

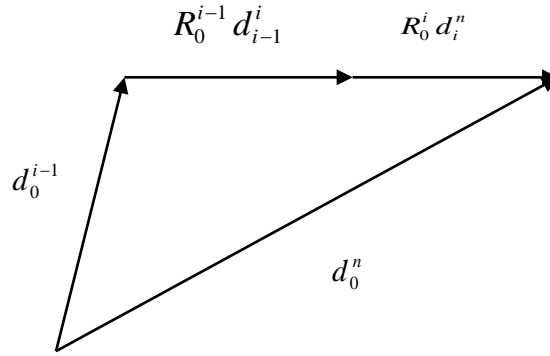
به این ترتیب در مفصل لولایی $\omega_{i-1}^i = \dot{q}_i k$ و در مفصل کشویی $\omega_{i-1}^i = 0$ است . از این رو می توان
 ژاکوبین چرخشی را به صورت کلی به صورت زیر نوشت:

$$J_\omega = [\rho_1 z_0 \quad \rho_2 z_1 \quad \cdots \quad \rho_n z_{n-1}] \quad (۷-۲)$$

که در مفصل لولایی $\rho_i = 1$ و در مفصل کشویی : $\rho_i = 0$ است .

سرعت خطی :

به شکل ۱-۲ توجه کنید :



شکل ۱-۲ بردار جابجایی مفاصل

در این شکل d نشانگر میزان جابجایی هر مفصل نسبت به مفاصل دیگر می باشد. که شماره مفاصل با اندیس مشخص می شود. از شکل خواهیم داشت:

$$d_0^n = d_0^{i-1} + R_0^{i-1} d_{i-1}^i + R_0^i d_i^n \quad (۸-۲)$$

که در آن d_0^n فاصله خطی مفصل n تا پایه است.

از طرفی سرعت خطی مفصل n نسبت به پایه از رابطه زیر به دست می آید

$$v_0^n = \dot{d}_0^n \quad (۹-۲)$$

که در آن v_0^n سرعت خطی مفصل n نسبت به پایه است.

اگر مفصل i کشویی باشد و فقط بخواهیم i را حرکت دهیم و اثر آن را بر سرعت مجری نهایی بیابیم در حالی که بقیه مفاصل قفل هستند، باید d_0^n را به صورت زیر محاسبه کرد [۲۴].

$$d_0^n = d_0^n(q) \Rightarrow \dot{d}_0^n = \sum_{i=1}^n \frac{\partial d_0^n}{\partial q_i} \dot{q}_i = \begin{bmatrix} \frac{\partial d_0^n}{\partial q_1} & \dots & \frac{\partial d_0^n}{\partial q_n} \end{bmatrix} \begin{bmatrix} \dot{q}_1 \\ \vdots \\ \dot{q}_n \end{bmatrix} \quad (۱۰-۲)$$

از این رو با توجه به معادله ۲-۲ بخش خطی ژاکوبین به صورت زیر است.

$$J_v = \begin{bmatrix} \frac{\partial d_0^n}{\partial q_1} \\ \vdots \\ \frac{\partial d_0^n}{\partial q_n} \end{bmatrix}^T \quad (۱۱-۲)$$

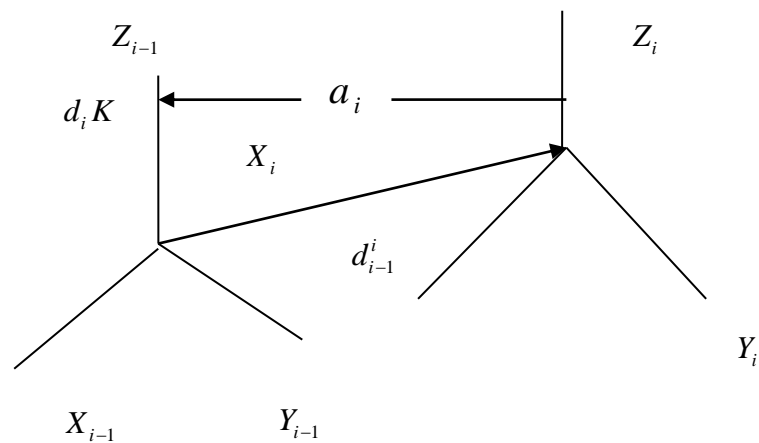
برای محاسبه $\frac{\partial d_0^n}{\partial q_i}$ فقط q_i متغیر و بقیه ثابتند. لذا می نویسیم [۲۴]:

$$d_0^n = d_0^{i-1} + R_0^{i-1} d_{i-1}^i + R_0^i d_i^n \quad (۱۲-۲)$$

که در آن d_{i-1}^i جایجایی مفصل i نسبت به مفصل $i-1$ و R_0^i ماتریس دوران مفصل i نسبت به پایه است. رابطه بالا را می توان به صورت زیر بیان کرد.

$$d_0^n = d_0^1 + R_0^1 d_1^2 + \dots + R_0^{i-1} d_{i-1}^i + \dots + R_0^{n-1} d_{n-1}^n \quad (۱۳-۲)$$

به شکل ۲-۲ توجه کنید :



شکل ۲-۲ مختصات دو مفصل مجاور

این شکل مختصات دو مفصل مجاور را نشان می دهد. که از آن خواهیم داشت [۲۴]:

$$d_{i-1}^i = d_i k + a_i R_{i-1}^i i \quad (۱۴-۲)$$

که در آن a_i فاصله z_i تا z_{i-1} و $d_i k$ فاصله x_i تا x_{i-1} است.

اگر فقط مفصل کشویی i حرکت کند :

برای مفصل n امین مفصل لولایی خواهیم داشت [۲۴]:

$$d_0^n = d_0^{i-1} + R_0^{i-1} d_{i-1}^n \quad (۱۵-۲)$$

که در آن d_0^{i-1} جابجایی مفصل $i-1$ نسبت به پایه ، R_0^{i-1} ماتریس دوران مفصل $i-1$ نسبت به پایه است.

وقتی از مفصل $i-1$ تا صفر قفل هستند R_0^{i-1} و d_0^{i-1} ثابت هستند و رابطه زیر را خواهیم داشت:

$$\dot{d}_0^n = R_0^{i-1} \dot{d}_{i-1}^n \quad (۱۶-۲)$$

همچنین مشتق جابجایی مفصل n نسبت به مفصل $i-1$ از رابطه زیر به دست می آید [۲۳]:

$$\dot{d}_{i-1}^n = \omega_{i-1}^i \times d_{i-1}^n = R_0^{i-1} \omega_{i-1}^i \times R_0^{i-1} d_{i-1}^n \quad (۱۷-۲)$$

$i-1$ اولین رابطی است که بعد از آن چرخش داریم .
از قانون $\dot{a} = \omega \times a$ در رابطه بالا استفاده شده است .
از طرفی داریم:

$$R_0^{i-1} d_{i-1}^n = O_n - O_{i-1} \quad (۱۸-۲)$$

که در آن O_n مبدأ دستگاه n در دستگاه صفر است . O_{i-1} مبدأ دستگاه $i-1$ در دستگاه صفر است .

که در این صورت رابطه را به صورت زیر می توان نوشت:

$$\dot{d}_{i-1}^n = R_0^{i-1} \dot{q}_i k \times (O_n - O_{i-1}) = z_{i-1} \dot{q}_i \times (O_n - O_{i-1}) = z_{i-1} \times (O_n - O_{i-1}) \dot{q}_i \quad (۱۹-۲)$$

که در آن O_n مبدأ دستگاه n در دستگاه صفر ، O_{i-1} مبدأ دستگاه $i-1$ در دستگاه صفر و z_{i-1} بردار z مفصل $i-1$ است.

رابطه اخیر نتیجه می دهد که بخش خطی ژاکوبین برای مفصل لولایی برابر با $z_{i-1} \times (O_n - O_{i-1})$ می باشد .

خلاصه :

اگر مفصل i کشویی باشد $J_i = \begin{bmatrix} z_{i-1} \\ 0 \end{bmatrix}$ و اگر مفصل i لولایی باشد $J_i = \begin{bmatrix} z_{i-1} \times (O_n - O_{i-1}) \\ z_{i-1} \end{bmatrix}$ است.

همچنین $O_n = d_0^n$ و $z_n = R_0^n k$ است .

اگر T_0^n (ماتریس تبدیل) را داشته باشیم از آن می توان R_0^n, d_0^n را به دست آورد . برای این کار می توان معادلات سینماتیک را نوشت و پارامترها را به دست آورد و بعد جاگذاری کرد . که در ادامه به آن می پردازیم .

۲-۲- سینماتیک مستقیم:

در سینماتیک مستقیم با توجه به مقدار متغیرهای مفصلی موقعیت و جهت مجری نهایی را به دست می آوریم. این کار با تعیین T_0^n انجام می گیرد [۲۴] و [۲۸].

$$T_0^n = \begin{bmatrix} R_0^n & d_0^n \\ 0 & 1 \end{bmatrix} T_0^n = T_0^1 T_1^2 \dots T_{n-1}^n \quad (20-2)$$

در تعیین T_{i-1}^i از اصول دناویت هارتنبرگ استفاده می شود.

اصول دناویت هارتنبرگ:

- (۱) محور X_i عمود بر محور Z_{i-1} باشد.
- (۲) محور X_i با محور Z_{i-1} متقاطع باشد [۲۴].

متغیرهای رابط :

- زاویه θ_i : زاویه از X_{i-1} تا X_i در جهت Z_{i-1}
- انحراف d_i : فاصله از X_{i-1} تا X_i در جهت Z_{i-1}
- طول a_i : فاصله از Z_{i-1} تا Z_i در جهت X_i
- پیچش α_i : زاویه از Z_{i-1} تا Z_i در جهت X_i

برای به دست آوردن T_{i-1}^i چهار مرحله کار انجام می شود :

ابتدا دوران حول محور z به اندازه θ_i ، سپس انتقال در راستای محور z به اندازه d_i ، سپس انتقال در راستای محور x به اندازه a_i و در نهایت دوران حول محور x به اندازه α_i

$$T_{i-1}^i = R(Z, \theta_i) T(Z, d_i) T(X, a_i) R(X, \alpha_i) \quad (21-2)$$

R ماتریس دوران و T ماتریس انتقال است.

با قرار دادن ماتریس های انتقال و دوران مربوطه ، ماتریس کلی تبدیل به صورت ۲-۲۲ نوشته می شود:

$$T_{i-1}^i = \begin{bmatrix} c\theta_i & -s\theta_i c\alpha_i & s\theta_i s\alpha_i & a_i c\theta_i \\ s\theta_i & c\theta_i c\alpha_i & -c\theta_i s\alpha_i & a_i s\theta_i \\ 0 & s\alpha_i & c\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (22-2)$$

که در آن s نشانگر سینوس و c نشانگر کسینوس است.

از ماتریس تبدیل ماتریس های دوران و انتقال را می توان به صورت زیر محاسبه کرد [۱].

$$T_{i-1}^i = [R_{i-1}^i \quad d_{i-1}^i] \quad (23-2)$$

پیاده سازی سینماتیک مستقیم :

برای پیاده سازی سینماتیک مستقیم مراحل زیر را طی کنیم [۲۴] و [۲۵].

(۱) رابط ها را از ۰ تا n شماره گذاری می کنیم . رابط ۰ همان پایه است .

(۲) مفاصل را از ۱ تا n شماره گذاری می کنیم .

(۳) به هر رابط یک دستگاه مختصات وصل می کنیم به طوری که محور X_i با محور Z_{i-1} عمود و متقاطع باشد .

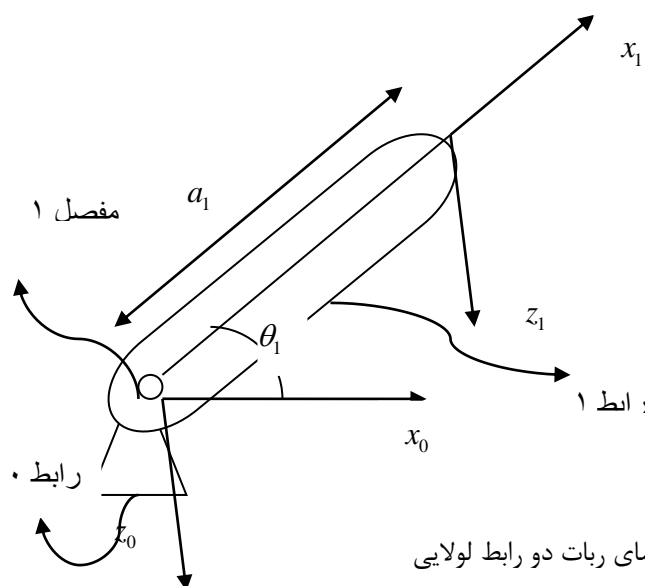
(۴) جهت محور Z در امتداد محور مفصل و جهت محور Y در جهت راستگرد (در حالی که شست ما در جهت محور Z است از X به سمت Y بیایم) باشد .

(۵) پارامتر های رابط را معین می کنیم و T_{i-1}^i را محاسبه می کنیم .

(۶) تبدیل T_0^n را محاسبه می کنیم .

نمونه اول ، ژاکوبین ربات یک رابط لولایی:

رابط	d	a	α	θ
۱	۰	a_1	۰	θ_1



شکل ۲-۳ شمای ربات دو رابط لولایی

ابتدا T_0^1 را محاسبه می کنیم .

$$T_0^1 = \begin{bmatrix} c\theta_1 & -s\theta_1 & 0 & a_1 c\theta_1 \\ s\theta_1 & c\theta_1 & 0 & a_1 s\theta_1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (24-2)$$

آنگاه می توان گفت :

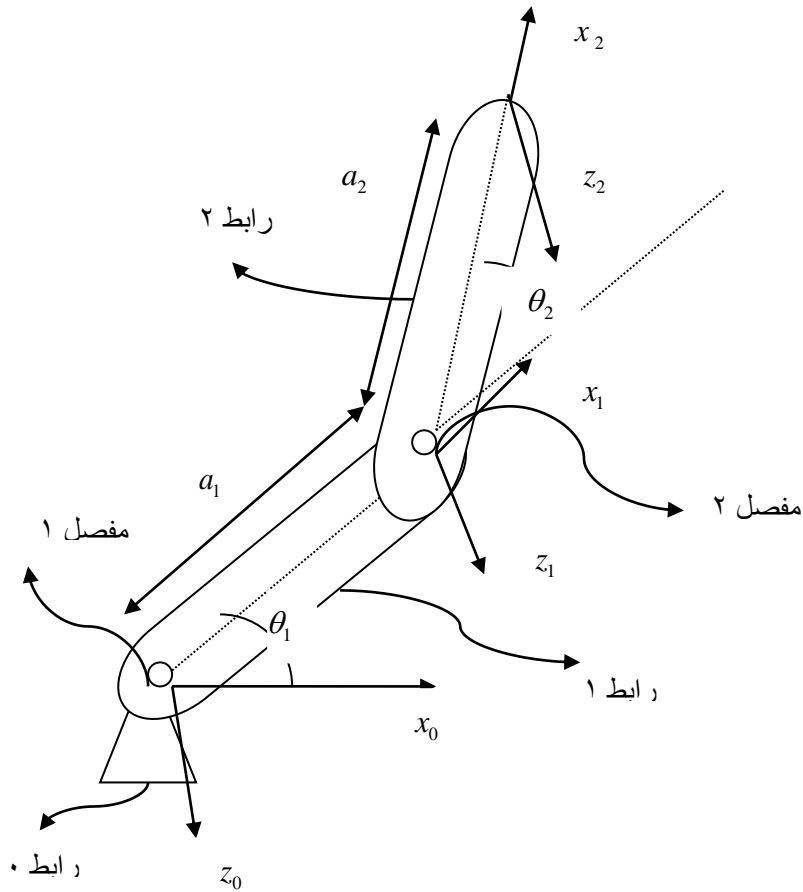
$$d_0^1 = O_1 = \begin{bmatrix} a_1 c\theta_1 \\ a_1 s\theta_1 \end{bmatrix} R_0^1 = \begin{bmatrix} c\theta_1 & -s\theta_1 \\ s\theta_1 & c\theta_1 \end{bmatrix} \quad (25-2)$$

و در نهایت ژاکوبین این ربات به صورت زیر به دست می آید.

$$J = \begin{bmatrix} Z_0 \times (O_1 - O_0) \\ Z_0 \end{bmatrix} = \begin{bmatrix} K \times (a_1 c\theta_1, a_1 s\theta_1, 0) \\ K \end{bmatrix} = ? \quad (26-2)$$

رابط	a_i	α_i	d_i	θ_i
1	a_1	0	0	θ_1^*
2	a_2	0	0	θ_2^*

نمونه دوم ، ژاکوبین بازوی ماهر صفحه ای دو رابط :



شکل ۲-۴ شمای کلی ربات صفحه ای دو رابط

ماتریس تبدیل مفصل اول نسبت به پایه:

$$T_0^1 = \begin{bmatrix} c_1 & -s_1 & 0 & a_1 c_1 \\ s_1 & c_1 & 0 & a_1 s_1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \Rightarrow O_1 = d_0^1 = \begin{bmatrix} a_1 c_1 \\ a_1 s_1 \\ 0 \end{bmatrix} \quad (27-2)$$

s نشانگر \sin و c نشانگر \cos ، مختصات مفصل اول در مختصات پایه است.

ماتریس تبدیل مفصل دوم نسبت به مفصل اول:

$$T_1^2 = \begin{bmatrix} c_2 & -s_2 & 0 & a_2 c_2 \\ s_2 & c_2 & 0 & a_2 s_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (28-2)$$

ماتریس تبدیل مفصل دوم نسبت به پایه:

$$T_0^2 = T_0^1 T_1^2 = \begin{bmatrix} c_{12} - s_{12} & -c_1 s_2 - c_2 s_1 & 0 & a_2 c_{12} - a_2 s_{12} + a_1 c_1 \\ c_2 s_1 + c_1 s_2 & -s_{12} + c_{12} & 0 & a_2 c_2 s_1 + a_2 c_1 s_2 + a_1 s_1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (29-2)$$

$$\Rightarrow O_2 = d_0^2 = \begin{bmatrix} a_2 c_{12} - a_2 s_{12} + a_1 c_1 \\ a_2 c_2 s_1 + a_2 c_1 s_2 + a_1 s_1 \\ 0 \end{bmatrix} \quad (30-2)$$

و در نهایت ماتریس ژاکوبین به صورت زیر محاسبه می شود.

$$J = \begin{bmatrix} Z_0 \times (O_2 - O_0) & Z_1 \times (O_2 - O_1) \\ Z_0 & Z_1 \end{bmatrix} = \begin{bmatrix} -a_1 s_1 - a_2 s_{12} & -a_2 s_{12} \\ a_1 c_1 + a_2 c_{12} & a_2 c_{12} \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 1 & 1 \end{bmatrix} \quad (31-2)$$

ژاکوبین را برای انواع ربات ها با هر تعداد رابط و مفصل می توان به همین شکل محاسبه کرد .
و بدین گونه است که می توان بین سرعت های مفصلی و دکارتی رابطه برقرار کرد .

$$\dot{x} = J\dot{q}$$

۲-۳- معادله دینامیکی ربات:

با استفاده از اصل " دالامبر " در خصوص ذرات ، معادله حرکت اویلر لاگرانژ به فرم زیر به دست می آید [۲۴] و [۲۵]:

$$\frac{d}{dt} \frac{\partial L}{\partial \dot{q}_j} - \frac{\partial L}{\partial q_j} = \tau_j \quad (32-2)$$

که در آن L لاگرانژین سیستم می باشد . و به فرم ۲-۳۳ است :

$$L = K - V \quad (۳۳-۲)$$

$V(q)$ انرژی پتانسیل سیستم است. τ_j گشتاور ناشی از نیروهای محرکه وارد بر ذره j و یا در ربات، مفصل j است. q_j متغیر مفصلی است که در مفاصل لولایی زاویه چرخش و در مفاصل کشویی طول جابجایی مفصل می باشد.

اما برای به دست آوردن معادلات انرژی های جنبشی و پتانسیل به صورت زیر عمل می کنیم: اگر در نظر بگیریم که دستگاه مختصاتی را به صورت صلب به طوریکه مرکز دستگاه مختصات در مرکز جرم باشد به جسم (در ربات به یک رابط آن) متصل کنیم، آنگاه سرعت یک نقطه روی جسم (رابط) به وسیله رابطه زیر به دست می آید [۲۴] و [۲۵].

$$v = v_c + \omega \times r \quad (۳۴-۲)$$

معادله بالا سرعت ذره را نسبت به یک دستگاه مبنا بیان می کند. اما ممکن است این رابطه را برای محاسبه همان بردار اما در دستگاه مختصات در حال حرکت به کار ببریم. تبدیل با ضرب بردار در ماتریس دوران اجرا می شود. اگر R ماتریس دورانی را نشان دهد که بردارهای آزاد در حال حرکت را به دستگاه مبنا تبدیل می کند. پس سرعت یک ذره که در r قرار دارد نسبت به دستگاه در حال حرکت برابر است با:

$$v = v_c + S(\omega)r \quad (۳۵-۲)$$

که ماتریس پاد متقارن $S(\omega)$ در بخش قبل تعریف شده است. اگر جسمی داشته باشیم که از ذرات پیوسته تشکیل می شود و ρ چگالی جرم جسم باشد و نیز B ناحیه ای سه بعدی از فضا باشد که توسط جسم اشغال شده است یعنی به عبارتی:

$$\int_B \rho(x, y, z) dx dy dz = m \quad (۳۶-۲)$$

آنگاه انرژی جنبشی جسم به صورت زیر است:

$$K = \frac{1}{2} \int v^T(x, y, z) v(x, y, z) \rho(x, y, z) dx dy dz = \frac{1}{2} \int_B v^T(x, y, z) v(x, y, z) dm \quad (۳۷-۲)$$

با جاگذاری رابطه ۳۵-۲ در رابطه ۳۷-۲، انرژی جنبشی به صورت زیر نوشته می شود:

$$K = \frac{1}{2} \int_B [v_c + S(\omega)r]^T [v_c + S(\omega)r] dm \quad (38-2)$$

که بعد از محاسبه و ساده سازی ، کل انرژی جنبشی جسم به صورت زیر می باشد :

$$K = \frac{1}{2} m v_c^T v_c + \frac{1}{2} \omega^T I \omega \quad (39-2)$$

v_c سرعت خطی جسم ، ω سرعت زاویه ای جسم می باشد.

که بخش اول آن همان انرژی جنبشی است . بخش دوم یک تصحیح است و در حالاتی پدیدار می

گردد که جسم همانطور که انتقال می یابد می چرخد .

I ماتریس لختی 3×3 است که به صورت زیر تعریف می شود:

$$I = \begin{bmatrix} \int (y^2 + z^2) dm & - \int xy dm & - \int xz dm \\ - \int xy dm & \int (x^2 + z^2) dm & - \int yz dm \\ - \int xz dm & - \int yz dm & \int (x^2 + y^2) dm \end{bmatrix} \quad (40-2)$$

x و y و z مختصات جسم (رابط) در فضا است .

اکنون این بحث را روی ربات با n رابط می بریم :

در بخش قبل دیدیم که سرعت های زاویه ای و خطی هر نقطه روی هر رابط به سادگی می تواند بر

حسب ماتریس ژاکوبین و مشتق متغیرهای مفاصل بیان شود و می توان نوشت :

$$v_{ci} = J_{v_{ci}}(q) \dot{q} \quad , \quad \omega_i = R_i^T(q) J_{\omega_i}(q) \dot{q} \quad (41-2)$$

که در آن $J_{v_{ci}}$, J_{ω_i} به ترتیب ، بخش دورانی و خطی ژاکوبین هستند .

ماتریس دوران $R_i^T(q)$ بر این نکته تأکید می کند که محاسبات باید در دستگاه متصل به رابط بیان شوند .

اکنون در نظر می گیریم که جرم رابط i برابر با m_i است و I_i ماتریس اینرسی رابط i در

مختصاتی موازی با دستگاه i که مبدأ آن در مرکز جرم است ، باشد . آنگاه از رابطه نتیجه می

شود که کل انرژی جنبشی ربات برابر است با :

$$K = \frac{1}{2} \dot{q}^T \sum [m_i J_{v_{ci}}(q)^T J_{v_{ci}}(q) + J_{\omega_i}(q)^T R_i(q) I_i R_i(q)^T J_{\omega_i}(q)] \dot{q} \quad (42-2)$$

به عبارت دیگر انرژی جنبشی ربات به فرم زیر است :

$$K = \frac{1}{2} \dot{q}^T D(q) \dot{q} \quad (43-2)$$

$D(q)$ یک ماتریس معین مثبت متقارن است که به طور کلی وابسته به پیکر بندی ربات است .
ماتریس D ماتریس اینرسی نامیده می شود .

اکنون روی انرژی پتانسیل بحث می کنیم . در حالت دینامیک صلب تنها منبع انرژی پتانسیل نیروی جاذبه است . بنابراین کل انرژی پتانسیل برابر است با [۲۵،۲۴]:

$$V = \int g^T r dm = g^T \int_B r dm = g^T r_c m \quad (44-2)$$

به عبارت دیگر انرژی پتانسیل جسم همان است که اگر جرم جسم به صورت فشرده در مرکز جرم آن بود . پس انرژی پتانسیل ربات فقط بستگی به بردار q دارد و به \dot{q} بستگی ندارد .
اکنون معادلات اوپلر لاگرانژ را برای ربات تشکیل می دهیم . انرژی جنبشی ربات را از رابطه ۴۳-۲ داریم که به شکل زیر است.

$$K = \frac{1}{2} \sum_{i,j}^n d_{ij}(q) \dot{q}_i \dot{q}_j = \frac{1}{2} \dot{q}^T D(q) \dot{q} \quad (45-2)$$

لذا از معادله لاگرانژین می توانیم بنویسیم:

$$L = K - V = \frac{1}{2} \sum d_{ij}(q) \dot{q}_i \dot{q}_j - V(q) \quad (46-2)$$

اگر رابطه بالا را در نظر بگیریم . باید دو مؤلفه سمت راست آن را به دست آوریم .

$$\frac{\partial L}{\partial \dot{q}_k} = \sum d_{kj}(q) \dot{q}_j \quad (47-2)$$

$$\frac{d}{dt} \frac{\partial L}{\partial \dot{q}_k} = \sum_j d_{kj}(q) \ddot{q}_j + \sum_j \frac{d}{dt} d_{kj}(q) \dot{q}_j = \sum d_{kj}(q) \ddot{q}_j + \sum_{i,j} \frac{\partial d_{kj}}{\partial q_i} \dot{q}_i \dot{q}_j \quad (۴۸-۲)$$

همچنین

$$\frac{\partial L}{\partial q_k} = \frac{1}{2} \sum_{i,j} \frac{\partial d_{ij}}{\partial q_k} \dot{q}_i \dot{q}_j - \frac{\partial V}{\partial q_k} \quad (۴۹-۲)$$

که در آن V انرژی پتانسیل، q_i و q_j متغیرهای مفصل در ربات هستند. پس معادله حرکت اویلر لاگرانژ (۳۲-۲) را می توان چنین نوشت:

$$\sum_j d_{kj}(q) \ddot{q}_j + \sum_{i,j} \left\{ \frac{\partial d_{kj}}{\partial q_i} - \frac{1}{2} \frac{\partial d_{ij}}{\partial q_k} \right\} \dot{q}_i \dot{q}_j - \frac{\partial V}{\partial q_k} = \tau_k \quad (۵۰-۲)$$

$$k = 1, \dots, n$$

معادله بالا از سه بخش تشکیل شده است، نخست بخشی که شامل ضرب های از نوع \ddot{q}_i است که جانب مرکز نامیده می شود. دوم بخشی که شامل ضرب های از نوع $\dot{q}_i \dot{q}_j$ برای $i \neq j$ است که بخش کوریولیس نامیده می شود. بخش سوم آن عباراتی هستند که فقط شامل q هستند و مشتقات آن را ندارند. واضح است که آخرین مورد از مشتق انرژی پتانسیل پدید می آید و بخش گرانشی نامیده می شود. رابطه اخیر به طور معمول به صورت ماتریسی زیر نوشته می شود.

$$D(q)\ddot{q} + C(q, \dot{q}) + G(q) = \tau \quad (۵۱-۲)$$

که در آن D ماتریس اینرسی ربات، C ماتریس کوریولیس و G ماتریس انرژی پتانسیل است.

اکنون معادله دینامیکی ربات آرنجی صفحه ای را به دست می آوریم:
این ربات دو مفصل لولایی دارد که شمای کلی آن را در شکل ۵-۲ مشاهده می کنید. برای q_i برای $i=1,2$ زاویه مفصل را نشان می دهد. m_i جرم رابط، l_i طول رابط i ، l_{ci} فاصله بین مفصل قبلی و مرکز جرم رابط i و I_i اینرسی رابط i حول محوری که از صفحه خارج می شود و از مرکز جرم رابط i عبور می کند، می باشند.

از روابط ژاکوبین برای محاسبه انرژی جنبشی استفاده می کنیم. در بخش ۲-۱ داشتیم:

$$J_{\omega_i} = 0 \quad \text{و} \quad J_{v_i} = Z_{i-1}$$

$$\text{اگر مفصل } i \text{ کشویی باشد آنکاه: } J_{\omega_i} = 0 \quad \text{و} \quad J_{v_i} = Z_{i-1}$$

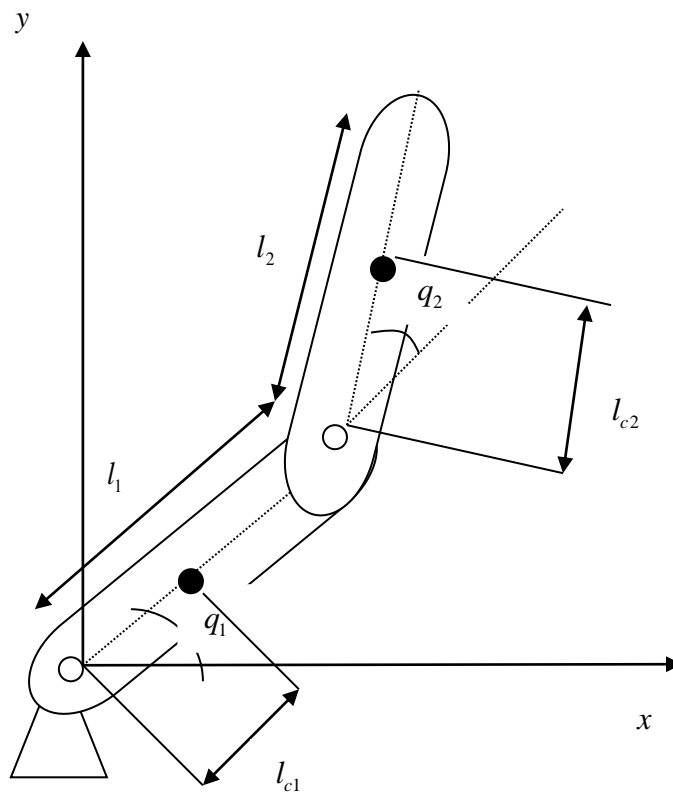
$$\text{اگر مفصل } i \text{ لولایی باشد آنکاه: } J_{\omega_i} = Z_{i-1} \times (O_n - O_{i-1}) \quad \text{و} \quad J_{v_i} = Z_{i-1}$$

از آن جایی که در ربات آرنجی صفحه ای دو مفصل لولایی داریم لذا هر دو ترم خطی و دورانی ژاکوبین را برای هر مفصل ، به طور جداگانه باید محاسبه کنیم .
 از معادله ۲-۳۱ دو بخش خطی و دورانی ژاکوبین را می توان تفکیک کرد.
 ابتدا بخش خطی ژاکوبین :

$$J_{v_{c1}} = \begin{bmatrix} -l_{c1} \sin q_1 & 0 \\ l_{c1} \cos q_1 & 0 \\ 0 & 0 \end{bmatrix} \quad (11) \quad (52-2)$$

9

$$J_{v_{c2}} = \begin{bmatrix} -l_1 \sin q_1 - l_{c2} \sin(q_1 + q_2) & -l_{c2} \sin(q_1 + q_2) \\ l_1 \cos q_1 + l_{c2} \cos(q_1 + q_2) & l_{c2} \cos(q_1 + q_2) \\ 0 & 0 \end{bmatrix} \quad (53-2)$$



شکل ۲-۵ شمای کلی ربات صفحه ای دو رابط از دید دینامیکی

بنابراین بخش انتقال انرژی جنبشی از رابطه ۲-۳۹ به دست می آید.

$$\frac{1}{2}m_1v_{c1}^T v_{c1} + \frac{1}{2}m_2v_{c2}^T v_{c2} = \frac{1}{2}\dot{q}^T \{m_1J_{v_{c1}}^T J_{v_{c1}} + m_2J_{v_{c2}}^T J_{v_{c2}}\}\dot{q} \quad (۵۴-۲)$$

اکنون به بخش مربوط به سرعت زاویه ای از ژاکوبین می پردازیم :
اول روشن است که :

$$\omega_1 = \dot{q}_1 k \quad , \quad \omega_2 = (\dot{q}_1 + \dot{q}_2)k \quad (۵۵-۲)$$

اما برای به دست آوردن J_{ω_2} , J_{ω_1} به بخش ژاکوبین مراجعه می کنیم از آنجا خواهیم داشت :

$$J_{\omega_1} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \end{bmatrix} \quad , \quad J_{\omega_2} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 1 \end{bmatrix} \quad (۵۶-۲)$$

واز رابطه ۲-۳۹ زیر بخش دورانی انرژی جنبشی به دست می آید :

$$\frac{1}{2}\dot{q}^T (J_{\omega_1}^T R_1 I_1 R_1^T J_{\omega_1} + J_{\omega_2}^T R_2 I_2 R_2^T J_{\omega_2})\dot{q} \quad (۵۷-۲)$$

و در نهایت با جمع دو رابطه ۲-۵۴ و ۲-۵۷ انرژی جنبشی کل را خواهیم داشت و از آنجا ماتریس $D(q)$ به صورت زیر به دست می آید:

$$D(q) = \{m_1J_{v_{c1}}^T J_{v_{c1}} + m_2J_{v_{c2}}^T J_{v_{c2}}\} + \begin{bmatrix} I_1 + I_2 & I_2 \\ I_2 & I_2 \end{bmatrix} \quad (۵۸-۲)$$

که در آن I_1 اینرسی رابط اول و I_2 اینرسی رابط دوم است.
که نتیجه می دهد :

$$D(q) = \begin{bmatrix} m_1l_{c1}^2 + m_2(l_1^2 + l_{c2}^2 + 2l_1l_{c2} \cos q_2) + I_1 + I_2 & m_2(l_{c2}^2 + l_1l_{c2} \cos q_2) + I_2 \\ m_2(l_{c2}^2 + l_1l_{c2} \cos q_2) + I_2 & m_2l_{c2}^2 + I_2 \end{bmatrix} \quad (۵۹-۲)$$

اکنون ماتریس C را محاسبه می کنیم . برای این منظور داریم :

$$\dot{D}(q) - \dot{q}^T \frac{\partial D(q)}{\partial q} = C(q, \dot{q}) \quad (۶۰-۲)$$

$$\Rightarrow C(q, \dot{q}) = \begin{bmatrix} -m_2 l_1 l_{c2} \dot{q}_2 \sin q_2 & -(\dot{q}_1 + \dot{q}_2)(m_2 l_1 l_{c2} \sin q_2) \\ m_2 l_1 l_{c2} \sin q_2 & 0 \end{bmatrix} \quad (۶۱-۲)$$

و در نهایت باید ماتریس گرانش را محاسبه کنیم . برای هر رابط انرژی پتانسیل ، حاصل ضرب جرم آن در شتاب جاذبه و ارتفاع مرکز جرم آن می باشد . بر این اساس خواهیم داشت:

$$\begin{aligned} V_1 &= m_1 g l_{c1} \sin q_1 \\ V_2 &= m_2 g (l_1 \sin q_1 + l_{c2} \sin(q_1 + q_2)) \\ V &= V_1 + V_2 = (m_1 l_{c1} + m_2 l_1) g \sin q_1 + m_2 l_{c2} g \sin(q_1 + q_2) \end{aligned} \quad (۶۲-۲)$$

اما ماتریس گرانش به فرم زیر محاسبه می شد :

$$G = \begin{bmatrix} \frac{\partial V}{\partial q_1} \\ \frac{\partial V}{\partial q_2} \end{bmatrix}$$

پس داریم :

$$G = \begin{bmatrix} (m_1 l_{c1} + m_2 l_1) g \cos q_1 + m_2 l_{c2} g \cos(q_1 + q_2) \\ m_2 l_{c2} g \cos(q_1 + q_2) \end{bmatrix} \quad (۶۳-۲)$$

بدین گونه است که ما تمام مؤلفه های معادله دینامیکی ربات آرنجی صفحه ای را به دست آوردیم .

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

فصل سوم کنترل ربات

۳

کنترل ربات

در بخش قبل معادله دینامیکی ربات را به دست آوردیم . اگر در نظر بگیریم که جرم هر رابط در یک سر آن متمرکز شود ، یعنی $(l_{c1} = l_1, l_{c2} = l_2)$ و از اینرسی هر رابط صرفنظر کنیم آنگاه معادلات دینامیکی به صورت در می آید [۲۳] و [۲۴] و [۲۵] :

$$D(q)\ddot{q} + C(q, \dot{q}) + G(q) = \tau \quad (1-3)$$

$$D(q) = \begin{bmatrix} l_2^2 m_2 + 2l_1 l_2 m_2 \cos q_2 + l_1^2 (m_1 + m_2) & l_2^2 m_2 + l_1 l_2 m_2 \cos q_2 \\ l_2^2 m_2 + l_1 l_2 m_2 \cos q_2 & l_2^2 m_2 \end{bmatrix}$$

$$C(q, \dot{q}) = \begin{bmatrix} -m_2 l_1 l_2 \dot{q}_2^2 \sin q_2 - 2m_2 l_1 l_2 \dot{q}_1 \dot{q}_2 \sin q_2 \\ m_2 l_1 l_2 \dot{q}_1^2 \sin q_2 \end{bmatrix} \quad (2-3)$$

$$G(q) = \begin{bmatrix} m_2 l_2 g \cos(q_1 + q_2) + (m_1 + m_2) l_1 g \cos q_1 \\ m_2 l_2 g \cos(q_1 + q_2) \end{bmatrix}$$

ما می دانیم که مفاصل ربات توسط ولتاژ ورودی موتور آنها به حرکت در می آیند . لذا در عمل ورودی سیستم باید ولتاژ مفاصل باشد . برای این منظور معادله حرکت موتورها را نیز به صورت زیر لحاظ می کنیم :

$$R_a j \ddot{\theta} + (R_a B + k_b k_m) \dot{\theta} = k_m V(t) + r R_a \tau_L \quad (3-3)$$

که در آن $V(t)$ بردار ولتاژ آرمیچر، j ماتریس لختی، R_a ماتریس مقاومت آرمیچر، θ بردار موقعیت روتور (rad) یا همان متغیر مفصل (q)، τ_L بردار گشتاور بار، $(N-m)$ ، B ماتریس ثابت میرایی، k_m ماتریس ثابت گشتاور، r ماتریس ضریب کاهش چرخ دنده ها، k_b ماتریس ثابت ولتاژالقایی می باشند.

اکنون از ترکیب دو رابطه ۳-۱ و ۳-۳ داریم:

$$k_m^{-1} R_a r^{-1} [j + r^2 D(q)] \ddot{q} + k_m^{-1} R_a r^{-1} [(k_m R_a^{-1} k_b + B) \dot{q} - r^2 h(q, \dot{q})] = V(t) \quad (4-3)$$

که در آن $h(q, \dot{q}) = C(q, \dot{q}) + G(q)$ می باشد.

ورودی سیستم فوق می تواند ولتاژ V و خروجی های آن q ، \dot{q} باشند. در کنترل ربات باید دو نکته مهم را مد نظر قرار داد:

اول اینکه معادله دینامیکی ربات یک معادله چند متغیره است.

دوم اینکه معادله دینامیکی ربات غیر خطی است.

برای مورد اول می توان مفاصل را جداگانه بررسی کرده و برای هر مفصل قانون کنترل را به کار برد و یا اینکه معادلات را به همان فرم ماتریسی در نظر گرفت و قانون کنترل را برقرار کرد. اما برای مورد دوم باید از ابزار و روش های کنترل غیر خطی بهره گرفت. ما روش خطی سازی فیدبکی را پی می گیریم.

۳-۱- روش خطی سازی فیدبکی

ایده خطی سازی فیدبکی، یعنی حذف غیر خطی ها و اعمال یک دینامیک خطی دلخواه، را می توان برای دسته ای از سیستم های غیر خطی که فرم متعارفی کنترل پذیری دارند به کار برد. سیستمی به فرم کنترل پذیر است که دینامیک های آن به صورت زیر نمایش داده شوند [۲۶]:

$$\dot{x}^{(n)} = f(x) + b(x)u \quad (5-3)$$

که در آن u ورودی کنترل اسکالر، x خروجی اسکالر مورد نظر، $X = [x, \dot{x}, \dots, x^{(n-1)}]^T$ بردار حالت، و $f(x)$ و $b(x)$ توابع غیر خطی از حالت هستند [۲۶].

در سیستم هایی که قابل نمایش به فرم متعارفی کنترل پذیرند، با فرض اینکه $b(x)$ مخالف صفر است، با استفاده از ورودی کنترل زیر

$$u = \frac{1}{b}[v - f] \quad (6-3)$$

می توانیم غیر خطی ها را حذف کنیم و رابطه ورودی - خروجی زیر را به دست آوریم :

$$x^{(n)} = v \quad (7-3)$$

آنگاه قانون کنترل زیر:

$$v = -k_0 - k_1 \dot{x} - \dots - k_{n-1} x^{(n-1)} \quad (9-3)$$

که در آن k_i ها طوری انتخاب شده اند که ریشه های چند جمله ای مشخصه $p^n + k_{n-1}p^{n-1} + \dots + k_0$ همگی در نیمه چپ صفحه مختلط قرار گیرند ، منجر به دینامیک پایدار مجانبی زیر می شود :

$$x^{(n)} + k_{n-1}x^{(n-1)} + \dots + k_0x = 0 \quad (10-3)$$

که ایجاب می کند تا $x(t) \rightarrow 0$ و همچنین اگر بحث ردگیری خروجی دلخواه $x_d(t)$ مطرح باشد آنگاه قانون کنترل زیر

$$v = x_d^{(n)} - k_0 e - k_1 \dot{e} - \dots - k_{n-1} e^{(n-1)} \quad (11-3)$$

که در آن $e(t) = x(t) - x_d(t)$ خطای ردگیری است به پایداری مجانبی منجر خواهد شد ، اگر که k_i ها به گونه ای انتخاب شوند که ریشه های چند جمله ای مشخصه همگی در سمت چپ صفحه مختلط قرار گیرند . و با توجه به اینکه معادله فوق خطی است می توان گفت که سیستم دارای پایداری مجانبی سراسری است . و در نتیجه

$$t \rightarrow \infty \Rightarrow e \rightarrow 0$$

همچنین با توجه به اینکه معادله دینامیکی حاصل یک سیستم نا متغیر با زمان است و تمام مقادیر ویژه ماتریس ضریب حالت در سمت چپ صفحه مختلط است ، سیستم دارای پایداری به صورت *uniform* نیز می باشد .

از آن جایی که سیستم پایدار مجانبی و نیز *uniform* است ، پس سیستم پایداری کلی دارد . اکنون روش خطی سازی فیدبکی را برای ربات آرنجی صفحه ای به کار می گیریم . این روش کنترلی در ربات ها به روش " دینامیک وارون " یا " گشتاور محاسبه شده " معروف است .

۳-۲- کنترل ربات با روش دینامیک وارون در فضای مفصلی

می خواهیم زاویه دو مفصل را با ورودی ولتاژ اعمالی به موتور هر مفصل کنترل کنیم . رابطه ۳-۴ را که معادله دینامیکی ربات است را می توان به این شکل نوشت [۲۴] و [۲۵] :

$$V = av' + b \quad (۱۲-۳)$$

که در آن

$$a = k_m^{-1} R_a r^{-1} [j + r^2 D(q)] = D \quad (۱۳-۳)$$

و همچنین

$$b = [(k_m R_a^{-1} k_b + B)\dot{q} + r^2 h(q, \dot{q})] = H \quad (۱۴-۳)$$

آنگاه

$$v' = \ddot{q} \quad (۱۵-۳)$$

اکنون قانون کنترلی را که در روش خطی سازی فیدبکی استفاده می شد به کار می گیریم :

$$v' = \ddot{q}_d + k_v(\dot{q}_d - \dot{q}) + k_p(q_d - q) \quad (۱۶-۳)$$

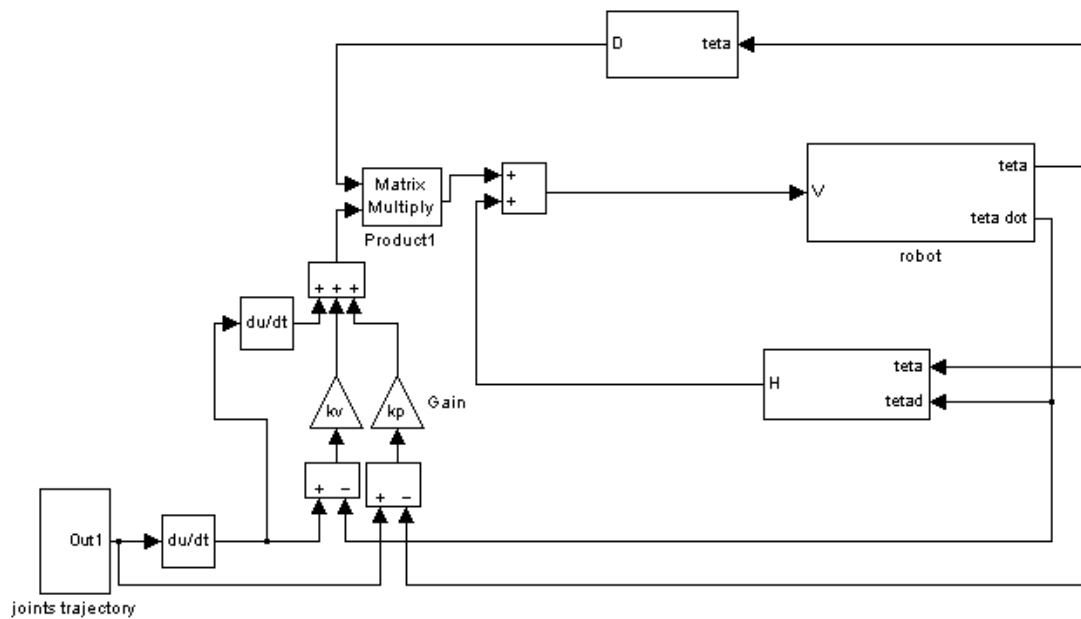
که در این صورت خواهیم داشت :

$$\ddot{q}_d - \ddot{q} + k_v(\dot{q}_d - \dot{q}) + k_p(q_d - q) = 0 \quad (۱۷-۳)$$

یا

$$\ddot{e} + k_v \dot{e} + k_p e = 0 \quad (۱۸-۳)$$

که در آن k_p و k_v ماتریس های قطری گین سیستم می باشند .
شمای کلی سیستم کنترل ربات با روش دینامیک وارون در شکل ۱-۳ آمده است :



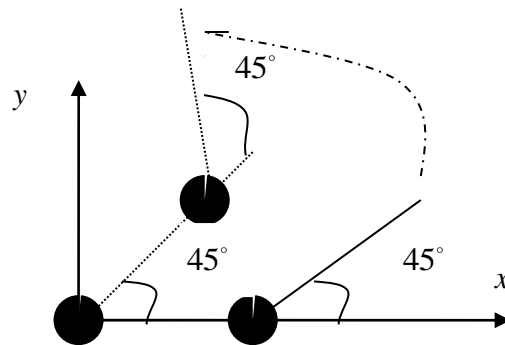
شکل ۳-۱ شمای کلی سیستم کنترل ربات با روش دینامیک وارون در فضای مفصلی

اکنون با مقادیر زیر کنترل ردگیری را به روش دینامیک وارون برای مقادیر زاویه های گوناگون انجام می دهیم :

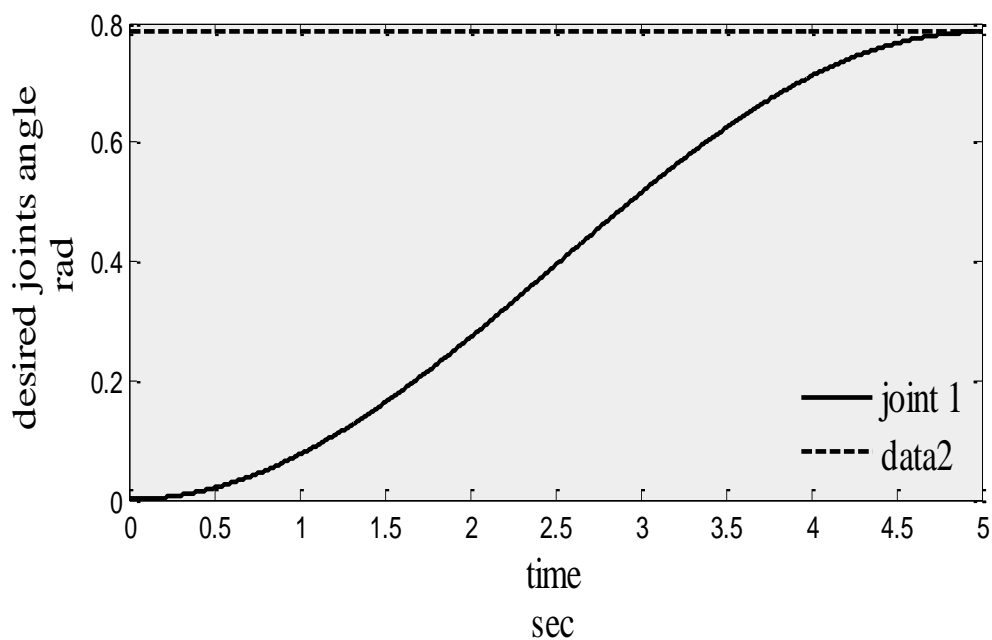
$$k_b = \begin{bmatrix} 0.03 & 0 \\ 0 & 0.03 \end{bmatrix} \quad k_m = \begin{bmatrix} 0.03 & 0 \\ 0 & 0.03 \end{bmatrix} \quad j = \begin{bmatrix} 10^{-6} & 0 \\ 0 & 10^{-6} \end{bmatrix}$$

$$B = \begin{bmatrix} 10^{-4} & 0 \\ 0 & 10^{-4} \end{bmatrix} \quad R_a = \begin{bmatrix} 0.1 & 0 \\ 0 & 0.1 \end{bmatrix} \quad r = \begin{bmatrix} 0.01 & 0 \\ 0 & 0.01 \end{bmatrix}$$

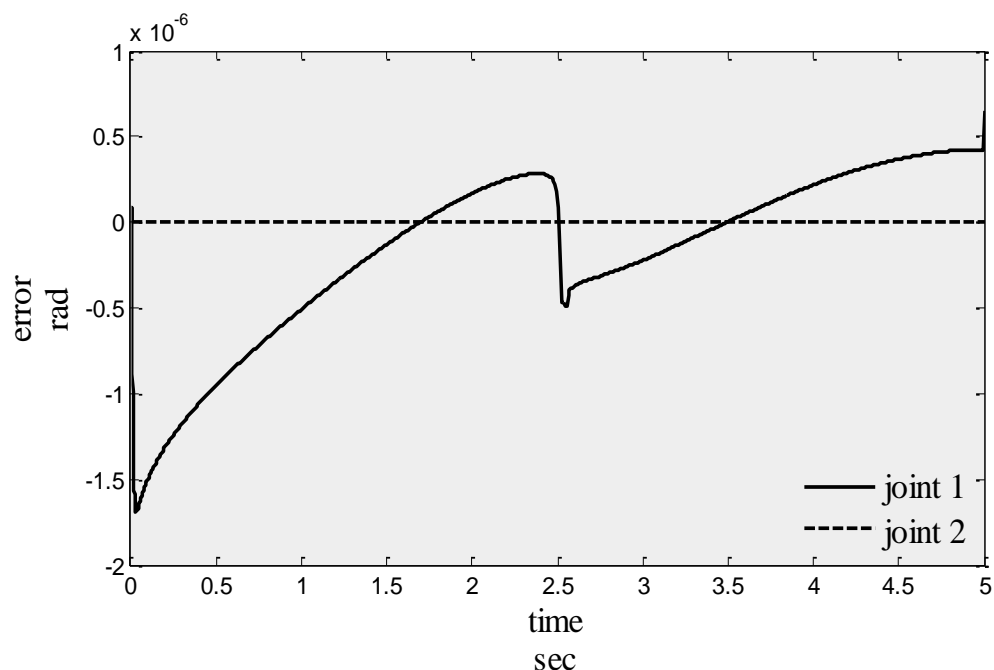
نتایج شبیه سازی به روش کنترل دینامیک وارون در ربات آرنجی صفحه ای نمونه ای که برای این روش در نظر گرفته ایم به این صورت است که زاویه مفصل اول در ابتدا صفر است و زاویه مفصل دوم 45° است و ربات از این حالت شروع به حرکت می کند . مسیر دلخواهی که برای ربات در نظر گرفته شده به این صورت است که زاویه مفصل اول به 45° درجه یا 0.78 رادیان برسد و زاویه مفصل دوم 45° بماند . نتایج شبیه سازی به صورت زیر است :



شکل ۲-۳ مسیری غیر تکین در فضای دکارتی



شکل ۳-۳ مسیر مطلوب مفصلی غیر تکین

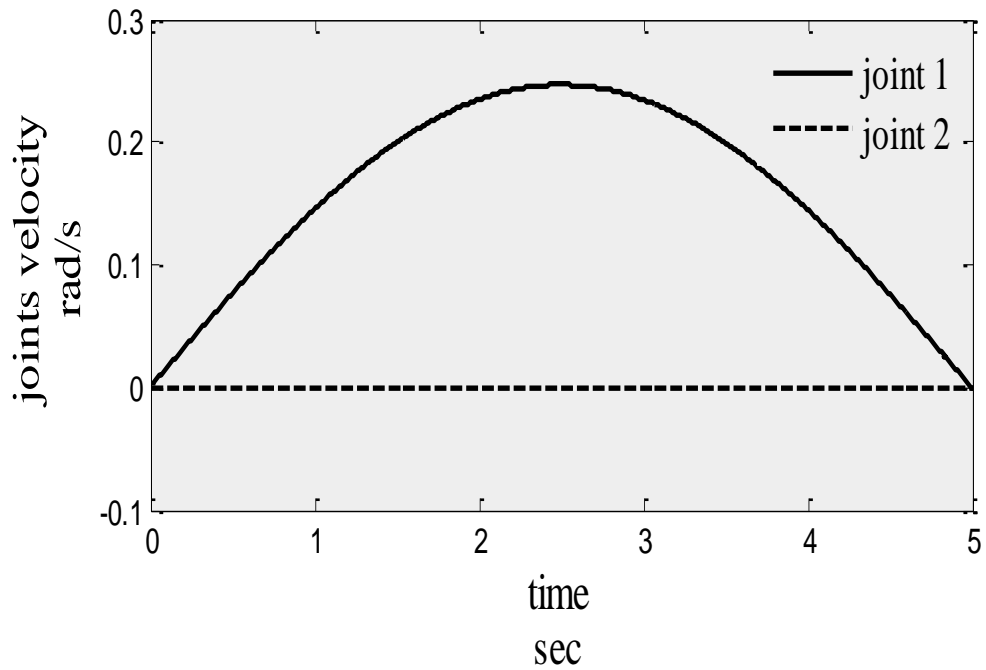


شکل ۳-۴ خطای روش دینامیک وارون در فضای مفصلی برای مسیر غیر تکین

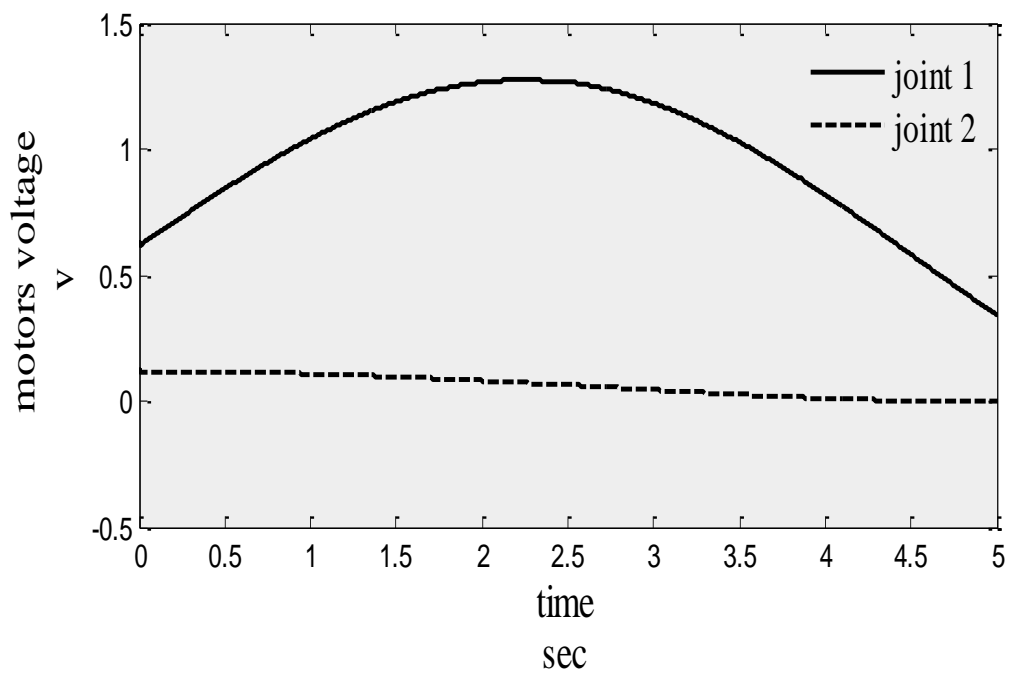
همین طور که مشاهده می شود خطا بسیار کم است. شکل های ۳-۵ و ۳-۶ به ترتیب سرعت و ولتاژ مفاصل را در روش دینامیک وارون در فضای مفصلی برای مسیر غیر تکین نشان می دهند که سرعت مفصل اول در بیشترین مقدار خود به 0.25 rad/s می رسد و سرعت مفصل دوم از آن جا که زاویه مفصل دوم در طی مسیر غیر تکین تغییری نمی کند و در 45° ثابت می ماند صفر است. همچنین ولتاژ مفصل اول در بیشینه خود به 1.25 V می رسد و برای مفصل دوم ولتاژ از 0.2 V به صفر می رسد.

شکل ۳-۷ نیز گشتاور مفاصل را نشان می دهد که مشاهده می شود گشتاور مفصل اول در بیشینه خود به 0.05 Nm می رسد و گشتاور مفصل دوم صفر است به این خاطر که زاویه مفصل دوم در طول مسیر ثابت است.

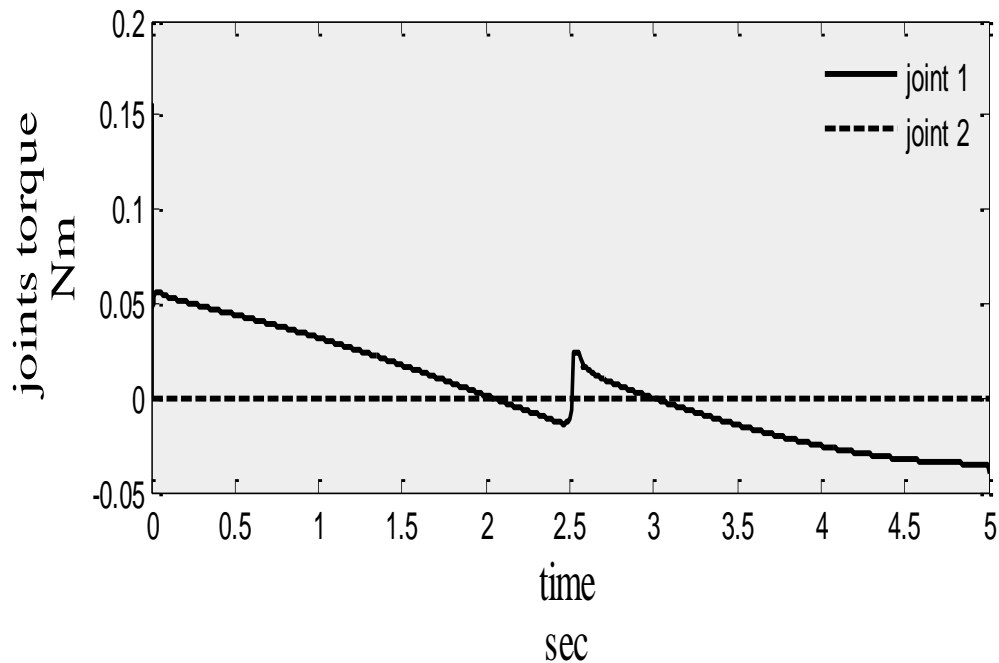
از آن جا که روش دینامیک وارون در فضای مفصلی تنها زاویه مفاصل را کنترل می کند و کنترلی روی فضای دکارتی ندارد ، لذا مسیر ربات در فضای دکارتی در این مورد اهمیت ندارد و نمایش داده نمی شود. همچنین سرعت مجری نهایی نیز در این مورد مطرح نیست. جلوتر در روش دینامیک وارون در فضای دکارتی ، خطای مسیر دکارتی و نیز سرعت مجری نهایی را مد نظر قرار خواهیم داد.



شکل ۳-۵ سرعت مفاصل در روش دینامیک وارون در فضای مفصلی برای مسیر غیر تکین



شکل ۳-۶ ولتاژ مفاصل در روش دینامیک وارون در فضای مفصلی برای مسیر غیر تکین



شکل ۳-۷ گشتاور مفاصل در روش دینامیک وارون در فضای مفصلی برای مسیر غیر تکین

همان طور که مشاهده می شود خطای ردگیری مسیر در حد $10^{-6} m$ است ، سرعت ، ولتاژ و گشتاور مفاصل محدود است و در کل مشکل خاصی در ردگیری مسیر غیر تکین با روش کنترل دینامیک وارون مفصلی مشاهده نمی شود. نتایج روش دینامیک وارون مفصلی برای مسیر تکین در فصل بعد خواهد آمد .

کنترل ربات در فضای دکارتی

از آنجا که مشخصات کار بازوی ماهر ، مانند گرفتن یک جسم یا دنبال کردن یک مسیر و ... اغلب نسبت به مجری نهایی داده می شود طبیعی است که باید الگوریتم کنترل به طور مستقیم در فضای کار به دست آید ، تا اینکه در فضای مفصل داده شود [۲۴] و [۲۵] . از بخش ژاکوبین ، معادله سینماتیک مستقیم از مرتبه یک زیر را داریم :

$$\dot{x} = J\dot{q} \quad (۳-۲۱)$$

۳-۳- کنترل دینامیک وارون در فضای دکارتی

داشتیم : $\dot{x} = J\dot{q}$ آنگاه :

$$\ddot{q} = \dot{J}^{-1}\dot{x} + J^{-1}\ddot{x} \quad \text{و} \quad \dot{q} = J^{-1}\dot{x} \quad (۲۲-۳)$$

آنگاه معادله دینامیکی ۳-۴ گفته شده در بخش قبل را می توان به فرم زیر نوشت :

$$J^{-1}k_m^{-1}R_a r^{-1}[j+r^2D(q)]\ddot{q} + k_m^{-1}R_a r^{-1}[BJ^{-1}\dot{x} + k_m R_a^{-1}k_b J^{-1}\dot{x} + r^2h(q, \dot{q})] - DJ^{-1}J\dot{q} = V(t) \quad (۲۳-۳)$$

که در آن $h(q, \dot{q}) = C(q, \dot{q}) + G(q)$ ، $D(q)$ ماتریس اینرسی ، J ماتریس ژاکوبین ربات ، و q متغیر مفصلی است.
پس می توانیم بنویسیم

$$D_x(q) = J^{-1}D(q) \quad (۲۴-۳)$$

$$H_x(q, \dot{q}) = k_m^{-1}R_a r^{-1}[BJ^{-1}\dot{x} + k_m R_a^{-1}k_b J^{-1}\dot{x} + r^2h(q, \dot{q})] - DJ^{-1}J\dot{q} \quad (۲۵-۳)$$

که به همان شیوه گفته شده در بخش دینامیک وارون در فضای مفصلی می توان کنترل دینامیک وارون را در فضای دکارتی به گونه ای به کار برد که ورودی سیستم ولتاژ باشد [۲۴].

$$V = av' + b \quad (۲۶-۳)$$

که در آن

$$a = D_x \quad (۲۷-۳)$$

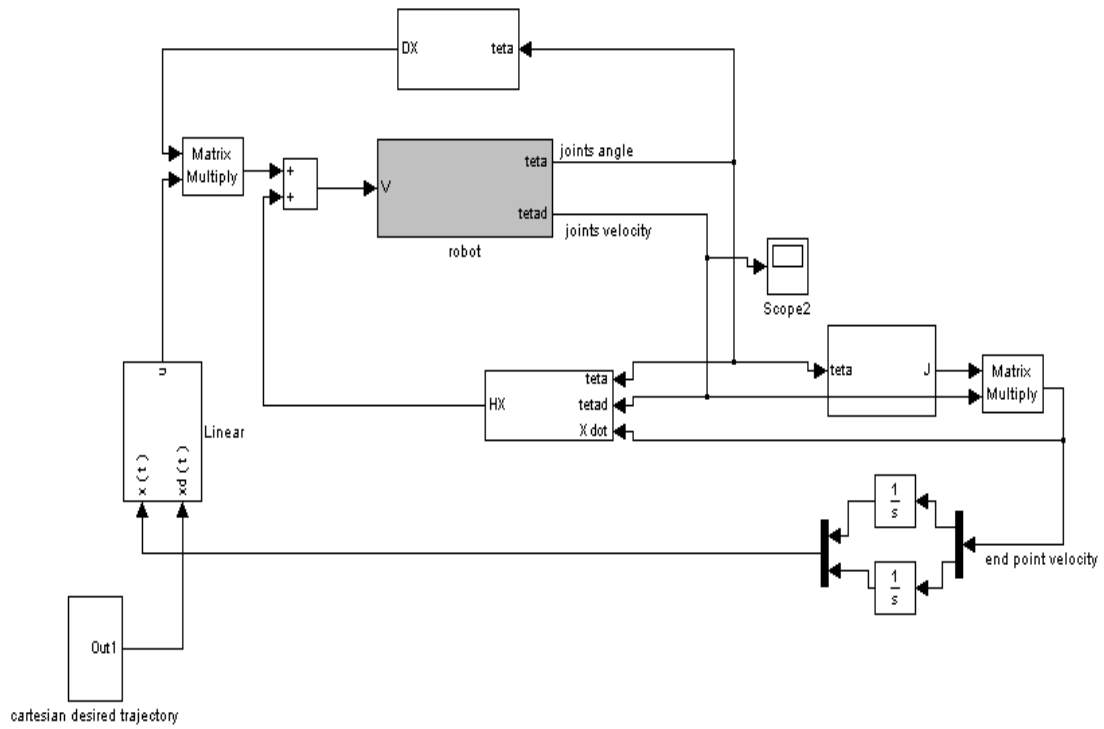
و همچنین

$$b = H_x \quad (۲۸-۳)$$

$$v' = \ddot{x}$$

(۲۹-۳)

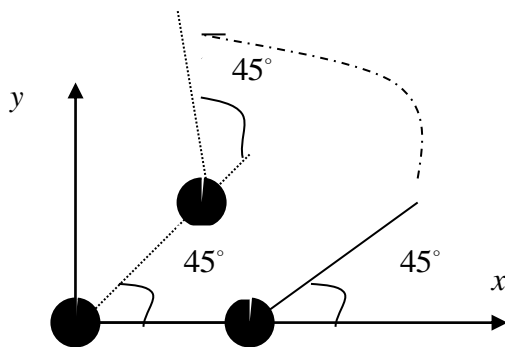
شمای کلی سیستم کنترل دینامیک وارون ربات در فضای دکارتی در شکل ۳-۳ آمده است.



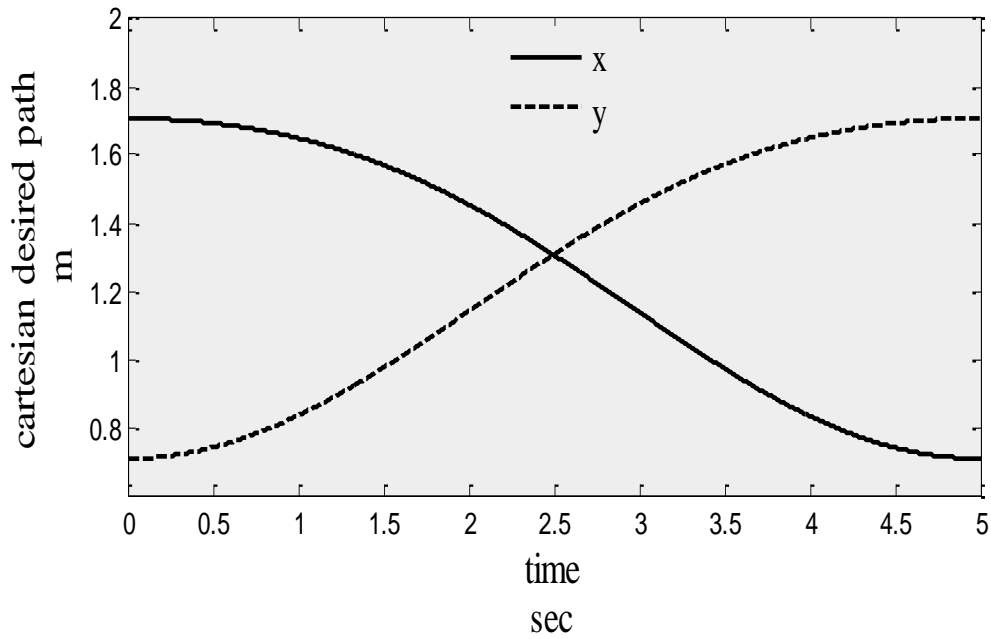
شکل ۳-۸ شمای کلی سیستم کنترل دینامیک وارون در فضای دکارتی

نتایج شبیه سازی سیستم فوق برای مسیری غیر تکین در صفحه بعد آمده است. (مسیره های تکین در فصل بعد توضیح داده می شود)

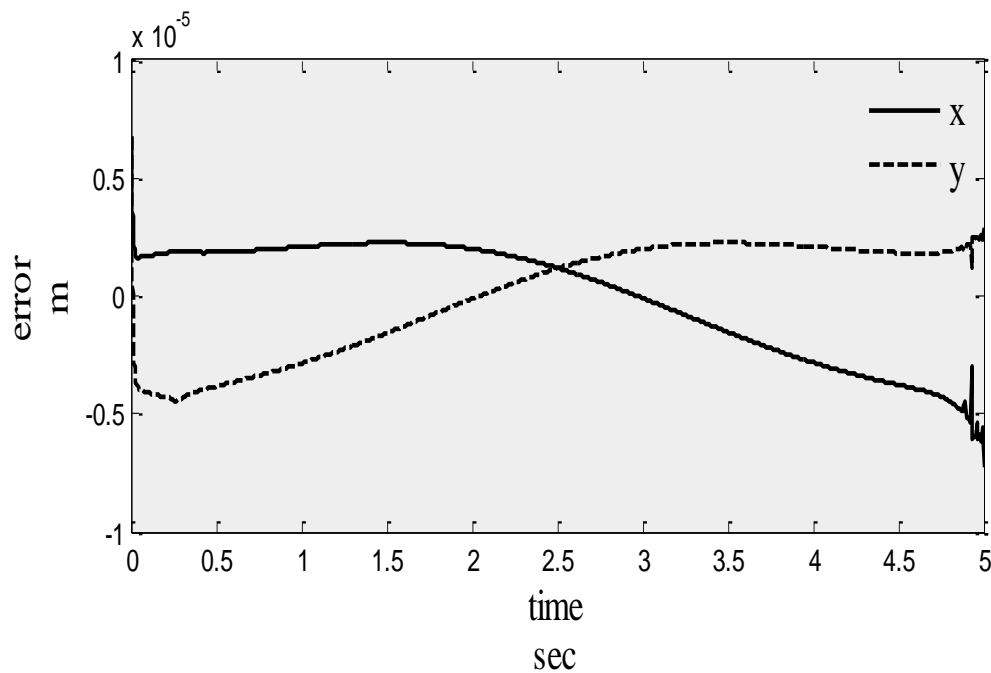
مسیر مورد نظر در فضای دکارتی به شکل زیر است :



شکل ۳-۹ مسیری غیر تکین در فضای دکارتی



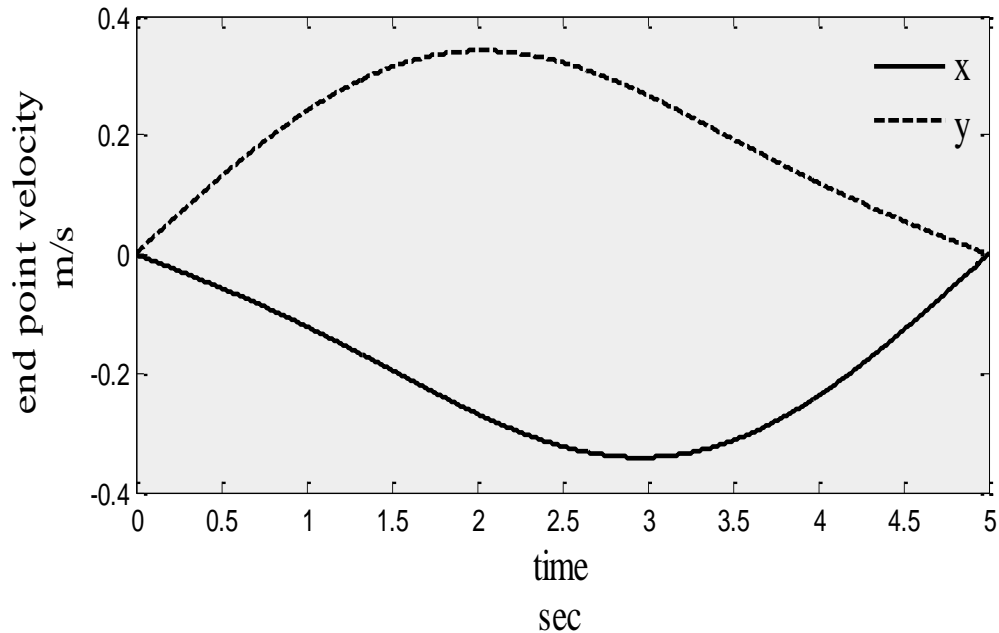
شکل ۳-۱۰ مسیر مطلوب غیر تکین در فضای دکارتی



شکل ۳-۱۱ خطای ردگیری مسیر دکارتی غیر تکین در روش دینامیک وارون دکارتی

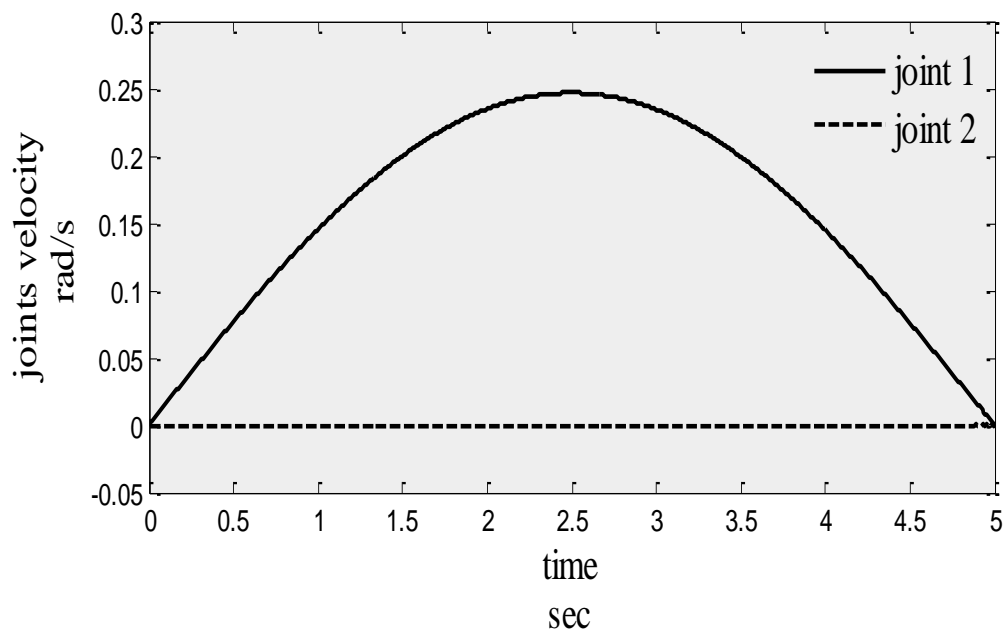
در روش دینامیک وارون در فضای دکارتی از آن جا که روی مسیر دکارتی کنترل داریم لذا می توانیم خطای ردگیری مسیر دکارتی را نشان دهیم همچنین سرعت مجری نهایی را نیز مشاهده و تحلیل نماییم.

در شکل ۱۱-۳ مشاهده می شود که خطای ردگیری در بیشترین مقدار خود به 0.5×10^{-5} می رسد که بسیار کم است.

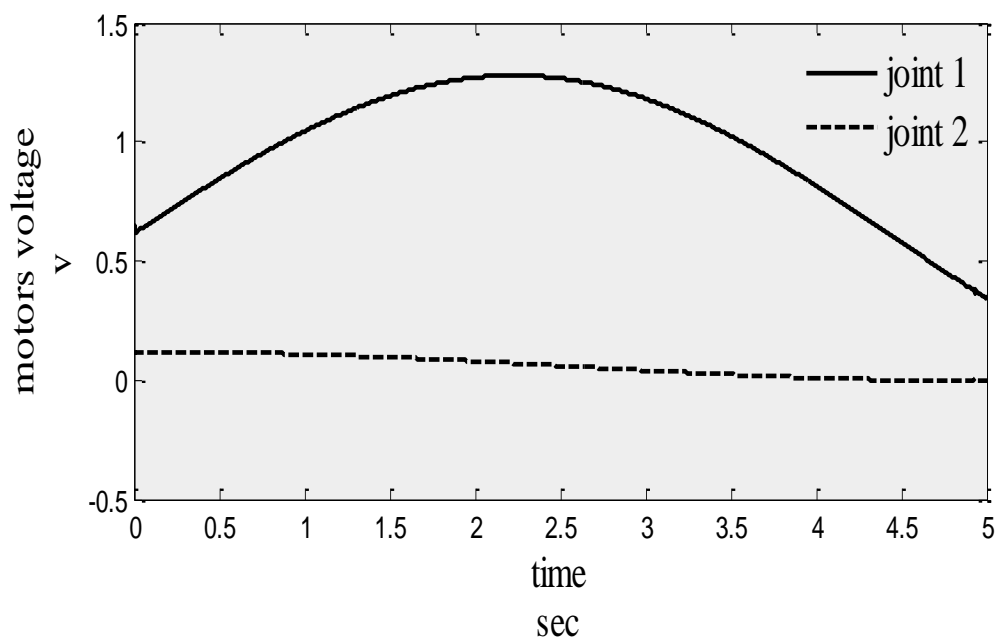


شکل ۱۲-۳ سرعت مجری نهایی در روش دینامیک وارون دکارتی

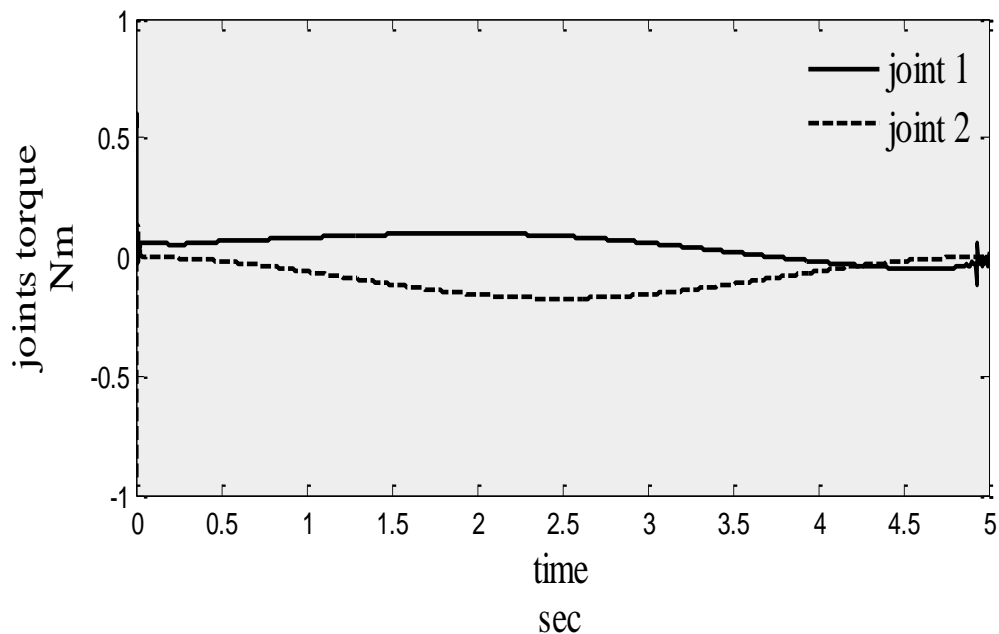
شکل های ۱۲-۳ و ۱۳-۳ به ترتیب سرعت مجری نهایی و سرعت مفاصل را در روش دینامیک وارون در فضای دکارتی برای مسیر غیر تکین نشان می دهند که سرعت مجری نهایی در بیشترین مقدار خود به 0.35 rad/s می رسد و سرعت مفصل اول همانند روش دینامیک وارون در بیشینه خود به 0.25 rad/s می رسد. از آن جا که زاویه مفصل دوم در طی مسیر غیر تکین تغییری نمی کند و در 45° ثابت می ماند، سرعت مفصل دوم صفر است. اگر شکل ۱۳-۳ را که سرعت مفاصل در روش دینامیک وارون در فضای مفصلی است را با شکل ۵-۳ که سرعت مفاصل در روش دینامیک وارون در فضای دکارتی است، مقایسه کنیم مشاهده می شود که یکسان هستند. از این رو می توان گفت سرعت مفاصل در هر دو روش مذکور برای مسیر های یکسان غیر تکین، یکسان است. در شکل ۱۴-۳ مشاهده می شود که ولتاژ مفصل اول در بیشینه خود به 1.25 V می رسد. ولتاژ مفصل دوم نیز از 0.2 V به صفر می رسد. در این مورد نیز با مقایسه شکل ۱۴-۳ با شکل ۶-۳ مشاهده می شود که تغییرات ولتاژ مفاصل در طی یک مسیر غیر تکین، در هر دو روش دینامیک وارون در فضای مفصلی و دینامیک وارون در فضای دکارتی یکسان است.



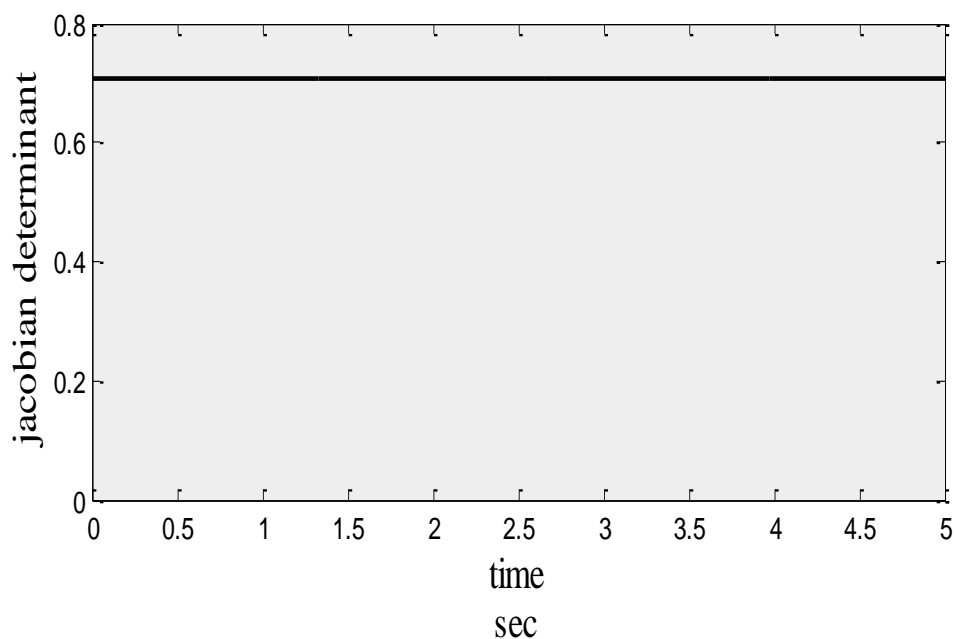
شکل ۳-۱۳ سرعت مفاصل برای مسیری غیر تکین در روش دینامیک وارون در فضای دکارتی



شکل ۳-۱۴ ولتاژ مفاصل برای مسیری غیر تکین در روش دینامیک وارون در فضای دکارتی



شکل ۳-۱۵ گشتاور مفاصل برای مسیری غیر تکین در روش دینامیک وارون در فضای دکارتی



شکل ۳-۱۶ دترمینان ماتریس ژاکوبین برای مسیر غیر تکین در روش دینامیک وارون در فضای دکارتی

شکل ۳-۱۵ تغییرات گشتاور را در طی مسیر غیر تکین به روش دینامیک وارون در فضای دکارتی نمایش می دهد. در شکل ۳-۱۶ دترمینان ماتریس ژاکوبین را در طی مسیر غیر تکین با روش کنترل اخیر مشاهده می کنید. در فصل بعد خواهیم گفت که در نقاط تکین ، دترمینان ماتریس

ژاکوبین ربات صفر است. از این رو، از آن جا که دترمینان مسیر فوق در شکل ۳-۱۶ روی 0.7 ثابت است لذا غیر تکین بودن این مسیر به این گونه نیز تأیید می شود. مشاهده شد که خطای مسیر در بیشترین مقدار خود به $0.5 \times 10^{-5} m$ می رسد. ولتاژ موتورها در بیشینه خود به 1.3v می رسد، گشتاور مفاصل در بیشینه خود به $0.2 Nm$ می رسد. سرعت مفاصل نیز در بیشینه خود به $0.25 rad / s$ می رسد که این مقادیر همه در محدوده عملکرد موتورها قرار دارند و لذا کنترلر دینامیک وارون در فضای مفصلی با مسیرهای غیر تکین مشکلی ندارد و آنها را به خوبی کنترل می کند. در فصل بعد این نتایج را با نتایج حاصل از ردگیری یک مسیر تکین مقایسه می کنیم.

فصل چہارم حالات تکین

حالات تکین

تا این جا روش های معمول در کنترل ربات را بررسی کردیم . اما در کنترل ربات در فضای دکارتی مشکلی اساسی به نام " حالات تکین " وجود دارد .
 به سادگی حالات تکین را می توان این گونه تعریف کرد که :
 هر پیکربندی از ربات که باعث دترمینان ژاکوبین برابر با صفر شود آن پیکربندی را پیکربندی تکین می گوئیم [۱] و [۲۴] .

$$\dot{x} = J\dot{q} \Rightarrow \dot{q} = J^{-1}\dot{x} \quad (1-4)$$

$$J^{-1} = \frac{1}{\det J} [adj J] \quad (2-4)$$

$$\det J = 0 \rightarrow q = \begin{bmatrix} q_1 \\ \vdots \\ q_n \end{bmatrix} \text{ پیکربندی تکین است .}$$

از آن جایی که دترمینان ژاکوبین ربات صفر است ، در این حالت ژاکوبین ربات معکوس ناپذیر است و نمی توان رابطه ای بین فضای دکارتی و فضای مفصلی برقرار کرد . و لذا کنترل ربات در فضای دکارتی مختل می شود [۱] و [۲] و [۳] .

همچنین از آن جا که $\dot{q} = J^{-1}\dot{x}$ و $J^{-1} = \frac{1}{\det J} [adj J]$ که در آن $adj J$ ، ماتریس الحاقی ژاکوبین است، می توان گفت که برای رسیدن به سرعت های محدود دکارتی در پیکربندی های تکین ، لازم است که سرعت مفصل ها به طور نامحدودی افزایش یابند .
 و نیز از آن جا که $F = J^{-T}\tau$ است ، فراهم کردن نیروی معین در مجری نهایی میسر نمی باشد زیرا این کار مستلزم وجود گشتاور نامحدود در مفاصل است .
 به موارد فوق در بخش مقدمه نیز اشاره شده بود ، همچنین گفتیم که سرعت ها و گشتاورهای نامحدود در مفاصل ممکن است موجب فروپاشی ربات شود .

یک ربات سریال با n متغیر مفصل $q = (q_1, q_2, \dots, q_n)$ را در نظر می گیریم ، q به یک پیکربندی اشاره می کند . همه پیکربندی های ربات فضای پیکربندی ربات ($C-space$) را

تشکیل می دهند. جهات و موقعیت های مجری نهایی ربات با مجموعه ای از m متغیر مستقل $x = (x_1, x_2, \dots, x_m)$ نشان داده می شوند. همه این جهات و موقعیت ها فضای کاری را تشکیل می دهند. در حالت کلی $m = 6$ است. رابطه بین پیکربندی های ربات و موقعیت و جهات مجری نهایی به شدت غیر خطی است و به صورت $k(q) = x$ تعریف می شود. که در آن $k: \mathcal{R}^n \rightarrow \mathcal{R}^m$ تابع سینماتیک نامیده می شود.

در یک ربات با درجه اضافی حرکت $m = n$ است. و برای یک ربات بدون درجه اضافی حرکت $m < n$ است.

مشتق مرتبه اول تابع سینماتیک روی یکپیکربندی تکین به صورت زیر است:

$$dx = J(q) dq \quad (3-4)$$

که در آن $J(q) = \frac{\partial k(q)}{\partial q}$ همان ژاکوبین ربات است.

هر پیکربندی تکین کاهش رتبه ای به اندازه d دارد که $d = m - r$ و r رتبه ماتریس ژاکوبین می باشد.

پیکربندی های تکین زیرمجموعه ای از فضای پیکربندی می باشند. تصویر پیکربندی های تکین تحت تابع سینماتیک k ، در فضای کاری، نقطه تکین نامیده می شوند [۱]. در سیستم های چند متغیره، عملی به نام تجزیه مقدار تکین به صورت زیر تعریف می شود [۱] و [۲۰]:

$$J(q) = U(q) \Sigma(q) V^T(q) \quad (4-4)$$

جایی که ستون های $u_i, i = 1, \dots, m$ از $U(q)$ و $v_i, i = 1, \dots, n$ از $V(q)$ به ترتیب از پایه های \mathcal{R}^n و \mathcal{R}^m هستند.

$\sum [diag(\sigma_1, \sigma_2, \dots, \sigma_m | 0_{m*(n-m)})]$ شامل مقادیر تکین $\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_m \geq 0$ است.

در یک پیکربندی تکین از دیمانسیون d همه $\sigma_i (i < r)$ ها صفر هستند [۱].

زیر ماتریس $U_s(q)$ را که $m \times d$ است و از d ستون آخر ماتریس $U(q)$ به دست می آید را تفکیک می کنیم. این ماتریس زیر فضای تکین از فضای کار را تشکیل می دهد. یک بردار از این زیر فضا یک جهت تکین از فضای کار را تشکیل می دهد.

با ضرب معادلات اخیر از طرف چپ در $U_s(q)$ برای هر dq به ترتیب به دست می آوریم

$$U_s^T(q) dx = 0 \quad (5-4)$$

این به این معنی است که در جهت تکین سرعت مجری نهایی صفر می شود یا به عبارتی مجری نهایی در یک پیکربندی تکین متوقف می شود .
اینک مشتق مرتبه دوم تابع سینماتیک را در نظر می گیریم

$$d^2(x) = dJ(q)dq + J(q)d^2q \quad (6-4)$$

دوباره یک پیکربندی تکین را در نظر می گیریم و هر دو طرف آن را در $U_s^T(q)$ ضرب می کنیم که خواهیم داشت :

$$U_s^T(q)(d^2x - dJ(q)dq) = 0 \quad (7-4)$$

این به این معنی است که مجری نهایی در زیر فضای تکین می تواند شتاب بگیرد اگر که حرکتی در مفاصل به صورت $dJ(q)dq \neq 0$ وجود داشته باشد .

۴-۱ حالات تکین در ربات آرنجی صفحه ای دو رابط:

پیش تر ژاکوبین ربات صفحه ای دو رابط را در فضای سه بعدی به فرم زیر به دست آوردیم :

$$J = \begin{bmatrix} -a_1 \sin(q_1) - a_2 \sin(q_1 + q_2) & -a_2 \sin(q_1 + q_2) \\ a_1 \cos(q_1) + a_2 \cos(q_1 + q_2) & a_2 \cos(q_1 + q_2) \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 1 & 1 \end{bmatrix} \quad (8-4)$$

که آن را در فضای دو بعدی صفحه x و y به این صورت می توان نوشت :

$$J = \begin{bmatrix} -a_1 \sin(q_1) - a_2 \sin(q_1 + q_2) & -a_2 \sin(q_1 + q_2) \\ a_1 \cos(q_1) + a_2 \cos(q_1 + q_2) & a_2 \cos(q_1 + q_2) \end{bmatrix} \quad (9-4)$$

اکنون دترمینان ماتریس فوق را محاسبه می کنیم :

$$\det J = a_1 a_2 \cos(q_1) \sin(q_1 + q_2) - a_1 a_2 \sin(q_1) \cos(q_1 + q_2) \quad (10-4)$$

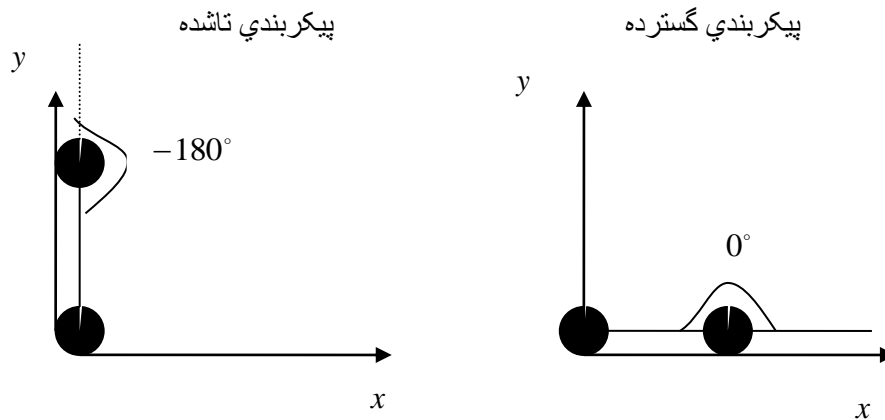
از برابر صفر قرار دادن ژاکوبین ، پیکربندی های تکین به صورت زیر به دست می آید :

$$q = \begin{bmatrix} q_1 \\ q_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} \frac{\pi}{2} \\ -\pi \end{bmatrix} \quad (11-4)$$

که جواب کلی آن به صورت زیر است:

$$q_2 = 0, -\pi \quad (12-4)$$

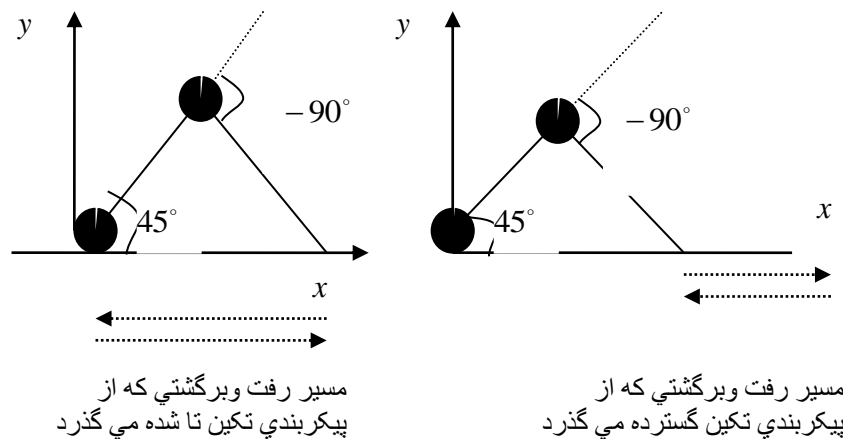
شمای این دو پیکربندی در صفحه مختصات دو بعدی به شکل زیر است :



شکل ۴-۱ شمای پیکربندی های تکین

مسیر تکین

مسیر تکین را مسیری تعریف می کنیم که در طی آن از یک حالت تکین عبور کنیم [۱]. دو مسیر تکین نمونه برای ربات آرئجی دو رابط در نظر می گیریم که یکی از پیکربندی گسترده و دیگری از پیکربندی تکین تاشده بگذرد . شمای این دو مسیر به شکل زیر است :

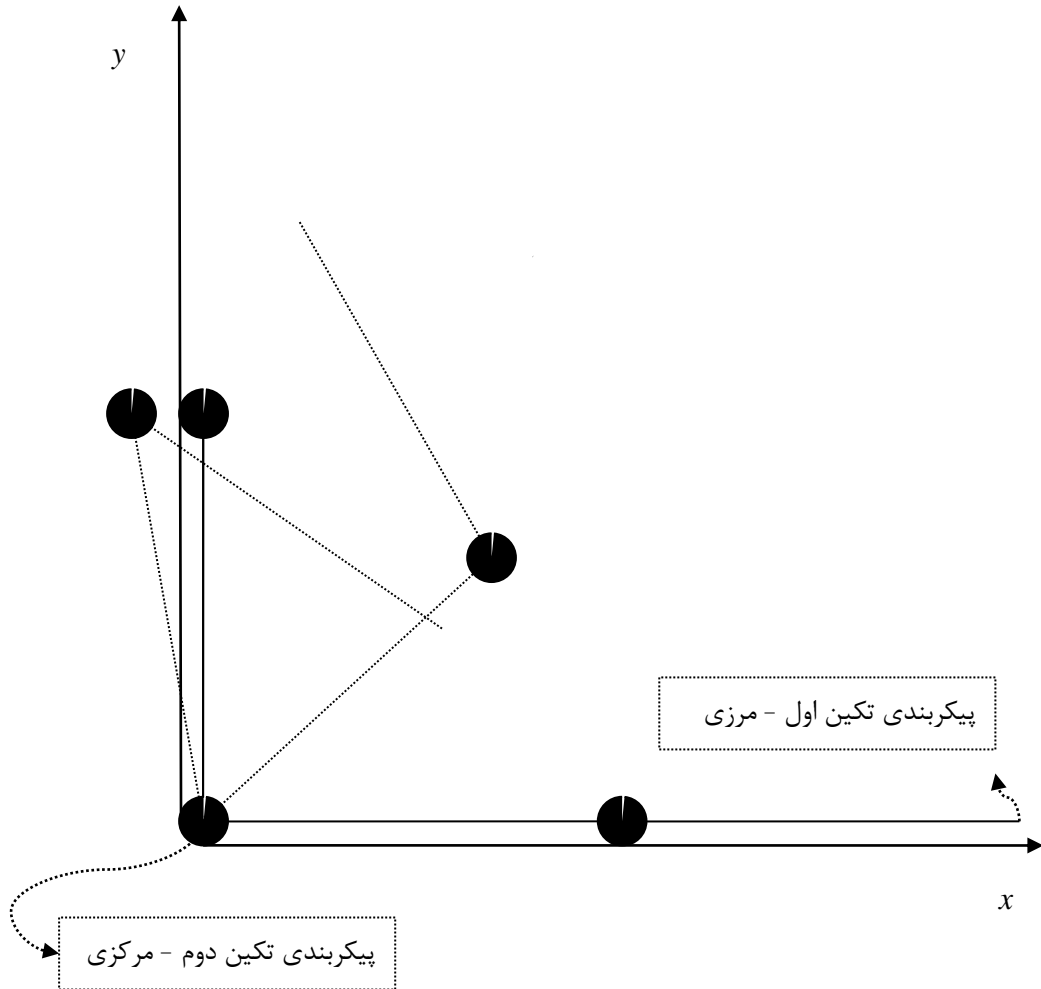


مسیر رفت و برگشتی که از پیکربندی تکین تا شده می گذرد

مسیر رفت و برگشتی که از پیکربندی تکین گسترده می گذرد

شکل ۴-۲

برای بررسی قابلیت روش های مختلف کنترل ربات در مسیر های تکین از مسیر تکین زیر استفاده می کنیم .



شکل ۴-۳ مسیر تکین که از دو نقطه تکین مرزی و مرکزی عبور می کند

همان طور که از شکل مسیر بر می آید ربات در لحظه شروع حرکت خود پیکر بندی تکین گسترده را دارد که به صورت کلی تر از آن جا که ربات در مرز فضای کاری خود قرار دارد این نقطه تکین را

نقطه تکین مرزی می نامیم.[۹] (فضای کاری ربات دایره ای به شعاع $1m$ و به مرکز نقطه $\begin{bmatrix} 0 \\ 0 \end{bmatrix}$

است.)

همچنین ربات در مسیر حرکت خود به نقطه تکین تا شده نیز می رسد که این نقطه تکین را نیز از آن جایی که در مرکز فضای کاری ربات قرار دارد ، نقطه تکین مرکزی می نامیم.

از دید زوایای مفاصل نیز ربات با زاویه مفاصل $\begin{bmatrix} q_1 \\ q_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$ شروع به حرکت می کند و بعد از طی

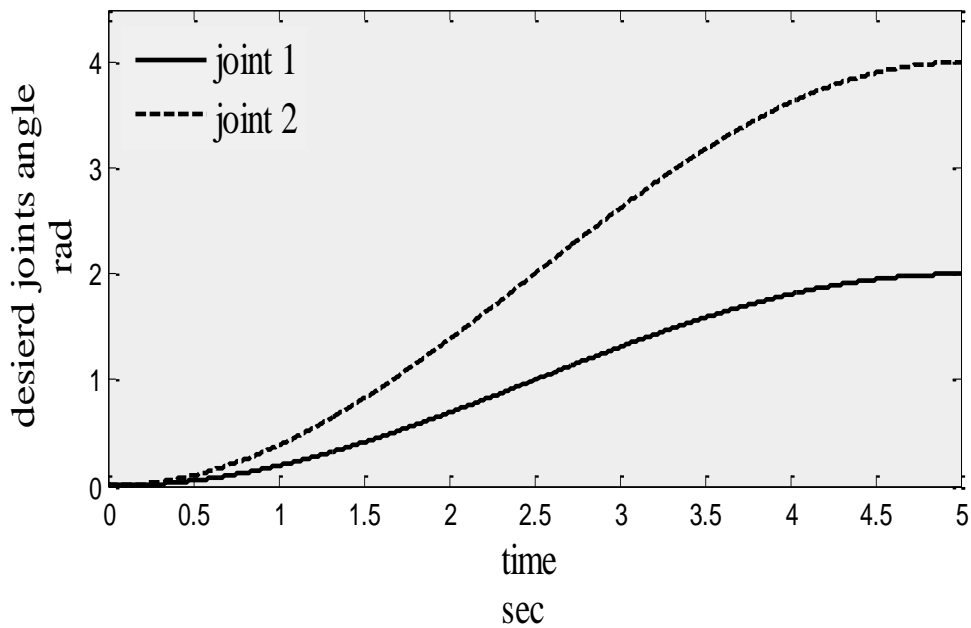
مسیر در نهایت به زاویه مفاصل $\begin{bmatrix} q_1 \\ q_2 \end{bmatrix} = \begin{bmatrix} 2 \\ 4 \end{bmatrix}$ رادین می رسد.

اکنون به کنترل این مسیر با روش های کنترلی گفته شده تا به این جا می پردازیم.

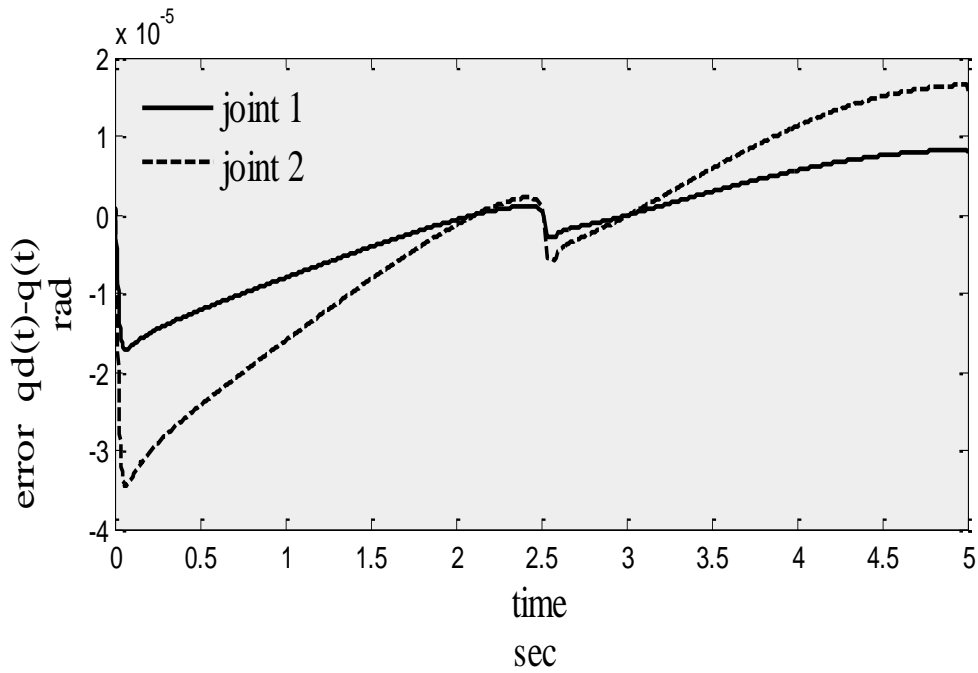
۲-۴ نتایج روش دینامیک وارون مفصلی برای مسیر تکین:

در این روش مسیر مفصلی را در نظر می گیریم . منظور از مسیر مفصلی زوایای مفاصل در هر زمان است . البته از آنجا که فضای کاری ربات دکارتی است ، این روش ، برای کنترل ربات ، ایده آل نیست زیرا این روش زوایای مفاصل را کنترل می کند و مختصات مجری نهایی ربات را نمی بیند و لحاظ نمی کند .

نتایج روش دینامیک وارون در فضای مفصلی به صورت زیر است:

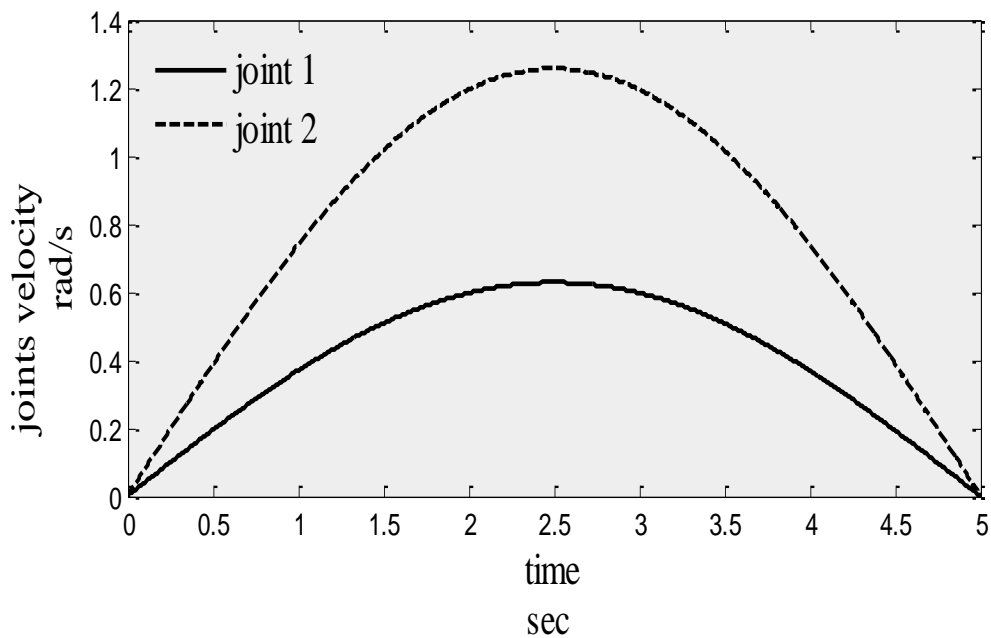


شکل ۴-۴ مسیر تکین مطلوب مفصلی به روش دینامیک وارون مفصلی



شکل ۴-۵ خطای ردگیری زوایای مفاصل برای مسیر تکین در روش دینامیک وارون مفصلی

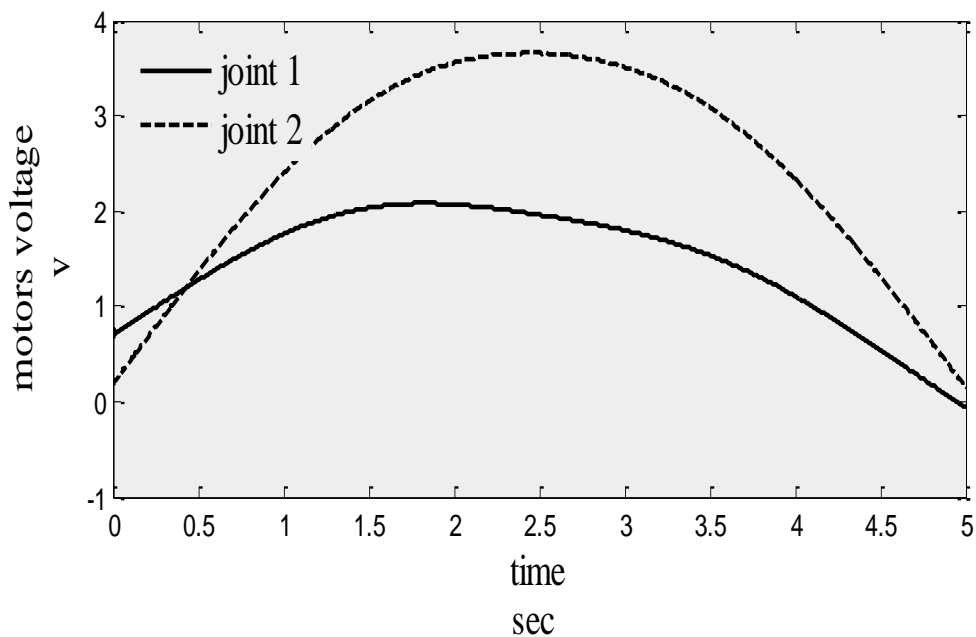
مشاهده می شود که خطای ردگیری در بیشترین مقدار خود به $3 \times 10^{-5} m$ می رسد که بسیار کم و در همان حد خطای روش دینامیک وارون مفصلی برای مسیر غیر تکین است.



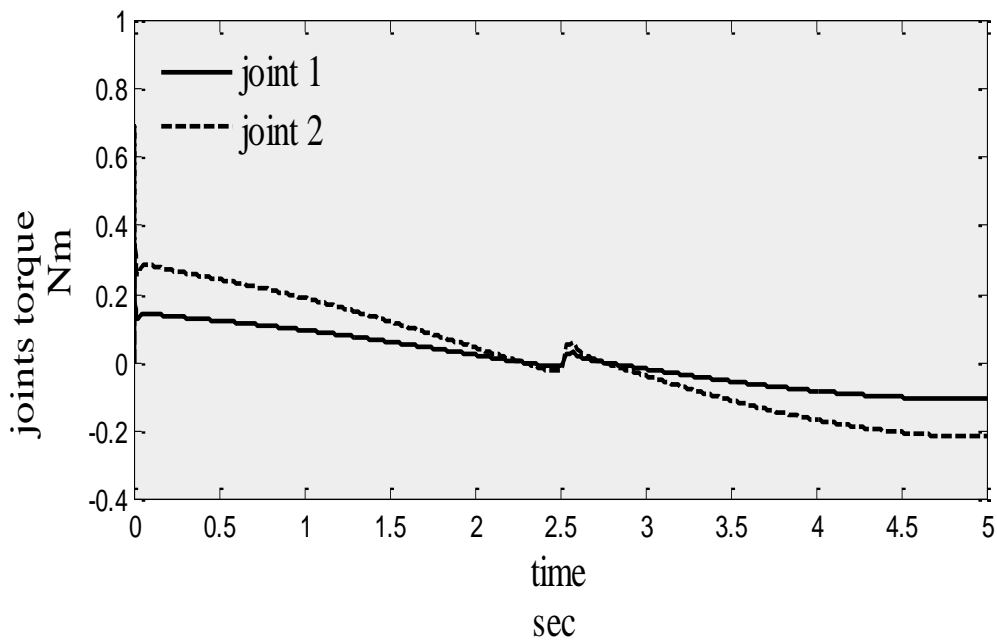
شکل ۴-۶ سرعت مفاصل در طی مسیر تکین در روش دینامیک وارون مفصلی

مشاهده می شود که سرعت مفاصل نیز مشابه سرعت مفاصل در روش دینامیک وارون مفصلی و دکارتی برای مسیر غیر تکین است. با این تفاوت که در این جا سرعت مفصل دوم صفر نیست زیرا

در طی مسیر ، زاویه مفصل دوم تغییر می کند. سرعت مفصل اول در بیشینه خود به 0.6 rad/s و سرعت مفصل دوم به 1.22 rad/s می رسد.



شکل ۴-۷ ولتاژ مفاصل در طی مسیر تکین در روش دینامیک وارون مفصلی



شکل ۴-۸ گشتاور مفاصل در طی مسیر تکین در روش دینامیک وارون مفصلی

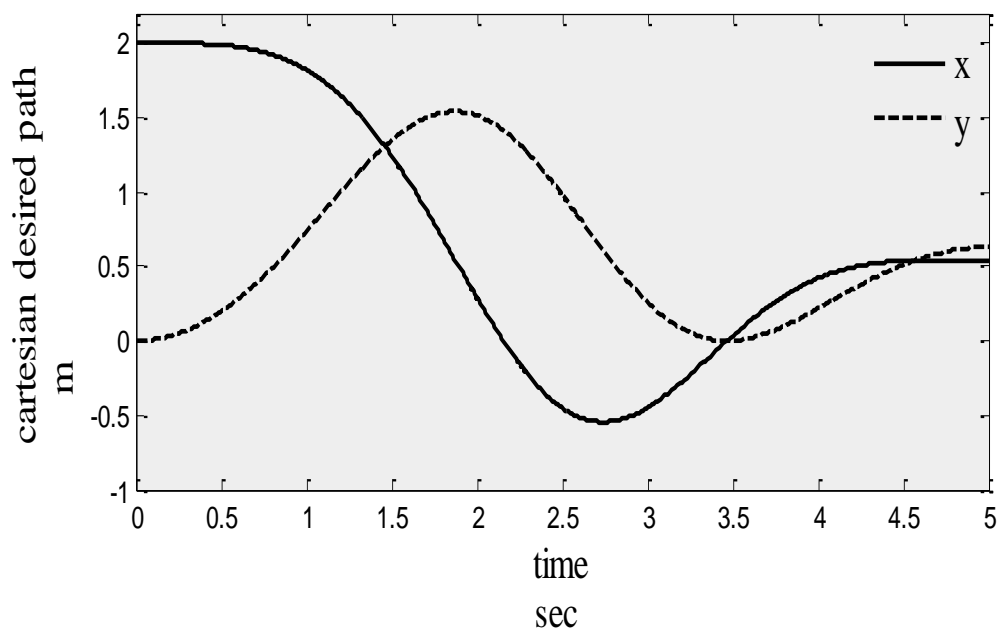
از شکل های ۴-۷ و ۴-۸ بر می آید که ولتاژ و گشتاور مفاصل نیز همان روند روش های دینامیک وارون مفصلی و دکارتی را برای ردگیری مسیر غیر تکین دارد. با این تفاوت که مقدار ولتاژ و

گشتاور مفصل دوم صفر نمی شود که البته این هم ربطی به تکین بودن مسیر ندارد بلکه به این خاطر است که در این مسیر ، مفصل دوم نیز حرکت می کند. همان طور که از شکل های در نتایج مشاهده می شود که بیشینه خطای مفصلی $3.5 \times 10^{-5} \text{ rad}$ می باشد . همچنین k_p و k_v که ضرایب کنترلر دینامیک وارون در فضای مفصلی هستند ، به ترتیب 200 و 10000 بوده است . سرعت ، ولتاژ و گشتاور مفاصل همگی محدود هستند و در محدوده ولتاژ موتورها است. مشاهده می شود که حالات تکین در روش کنترل دینامیک وارون در فضای مفصلی خللی وارد نمی کنند . ولتاژ موتورها محدود است و خطای رد گیری حد مطلوبی دارد.

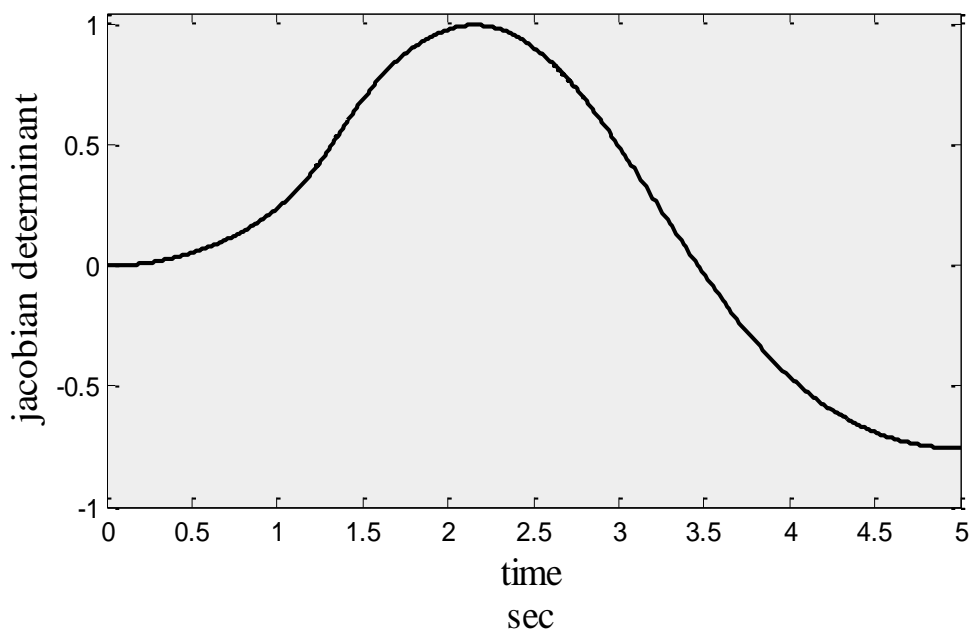
۳-۴ نتایج روش دینامیک وارون دکارتی با J^T برای مسیر تکین:

ابتدا روش دینامیک وارون دکارتی را به شیوه معمول آن ، یعنی با استفاده از ماتریس معکوس ژاکوبین (J^{-1}) ، به کار می گیریم . اما از آن جایی که در شروع مسیر تکین ، نقطه تکین از نوع مرزی را داریم و دترمینان ماتریس ژاکوبین در نقاط تکین صفر است و در نتیجه ماتریس ژاکوبین معکوس پذیر نمی باشد لذا مشاهده می شود که سیستم کنترل دینامیک وارون دکارتی در نقطه شروع با چالشی اساسی به سبب وجود نقطه تکین مرزی روبرو است و توانایی ردگیری مسیر را ندارد.

در گام بعد ، از ترانهاده ماتریس ژاکوبین (J^T) به جای ماتریس ژاکوبین استفاده می کنیم [۱۲] و [۱۴، ۱۳، ۱۲]. که با استفاده از مشتق گیری عددی نتایج آن به این صورت است:

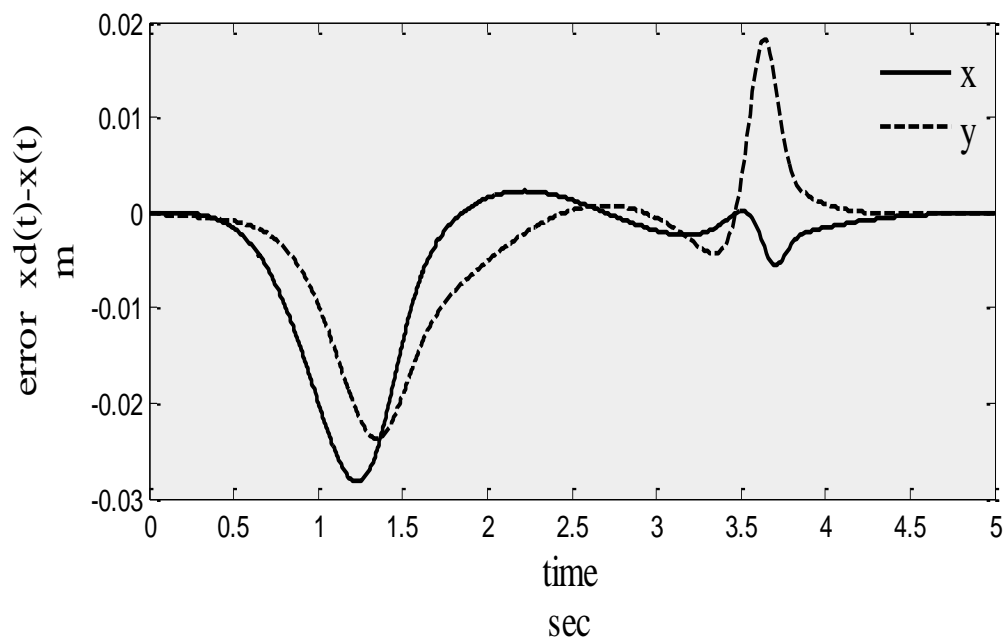


شکل ۴-۹ مسیر مطلوب تکین در فضای دکارتی



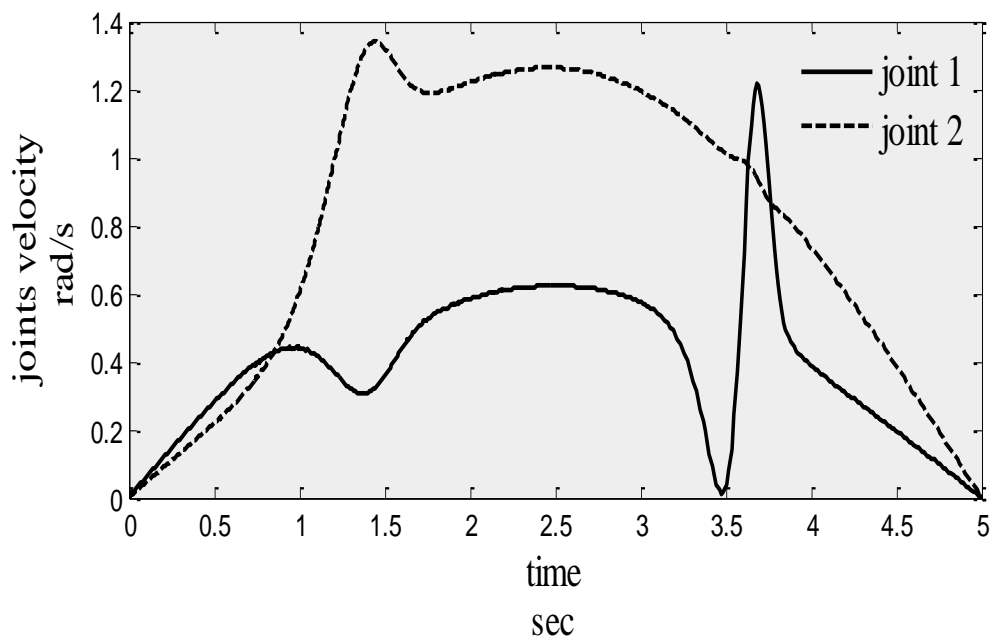
شکل ۴-۱۰ دترمینان ماتریس ژاکوبین در طی مسیر تکین در روش دینامیک وارون فضای دکارتی با استفاده از J^T

همان طور که در شکل ۴-۱۰ مشاهده می شود دترمینان ماتریس ژاکوبین ربات در دو نقطه طی مسیر صفر می شود. یکی در ابتدای مسیر است که همان نقطه تکین مرزی است، دیگری در زمان 3.5sec است که همان نقطه تکین مرکزی است. در شکل ۴-۱۱ مشاهده می شود که در نقاط تکین خطای ردگیری به خاطر استفاده از ماتریس ترانهاده ژاکوبین به جای معکوس ماتریس ژاکوبین ربات تقریباً صفر است اما خطای سیستم به همین دلیل استفاده از ماتریس ترانهاده ژاکوبین در بقیه مسیر زیاد است به طوری که در بیشترین حد خود به حدود 3cm می رسد.

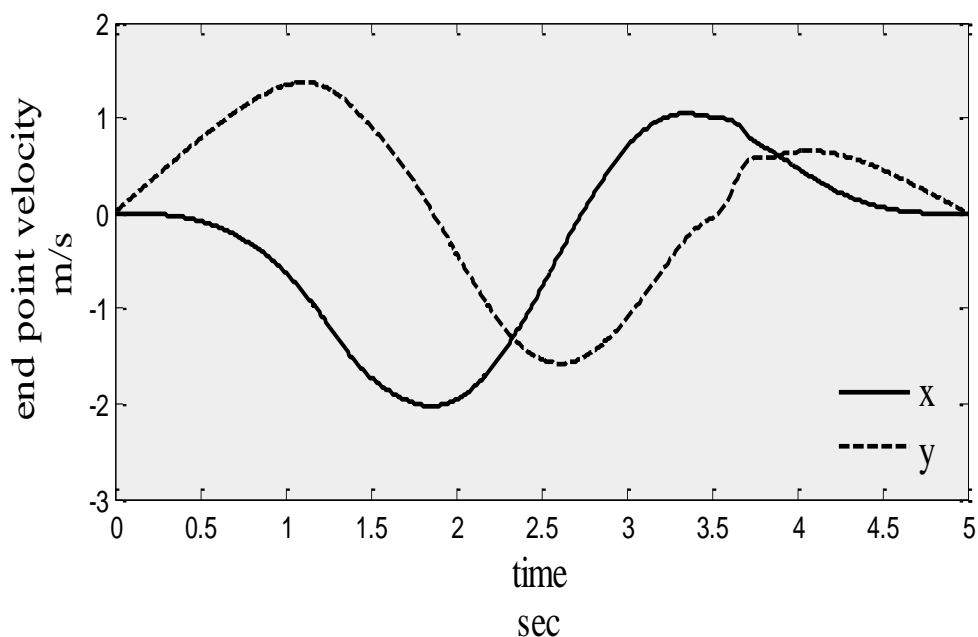


شکل ۴-۱۱ خطای ردگیری دکارتی مسیر تکین در روش دینامیک وارون دکارتی با استفاده از J^T

شکل ۴-۱۲ سرعت مفاصل را در طی مسیر تکین در روش دینامیک وارون دکارتی با استفاده از J^T نشان می دهد. همان طور که مشاهده می شود سرعت مفاصل در نقاط تکین صفر است اما اندکی پس از نقاط تکین تغییر زیاد در سرعت مفاصل داریم که این خوشایند نیست. خطای ردگیری و تغییرات سرعت مفاصل این روش را در فصل بعد با خطای ردگیری و سرعت مفاصل حاصل از روش کنترل سازگار با نقاط تکین که ارائه خواهیم کرد ، مقایسه می کنیم.

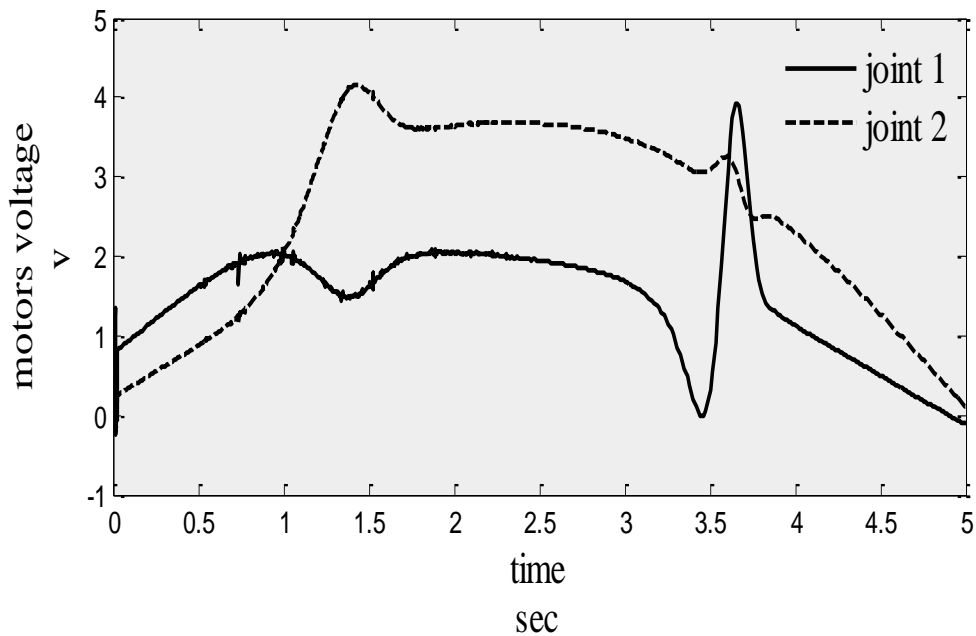


شکل ۴-۱۲ سرعت مفاصل طی مسیر تکین در روش دینامیک وارون دکارتی با استفاده از J^T



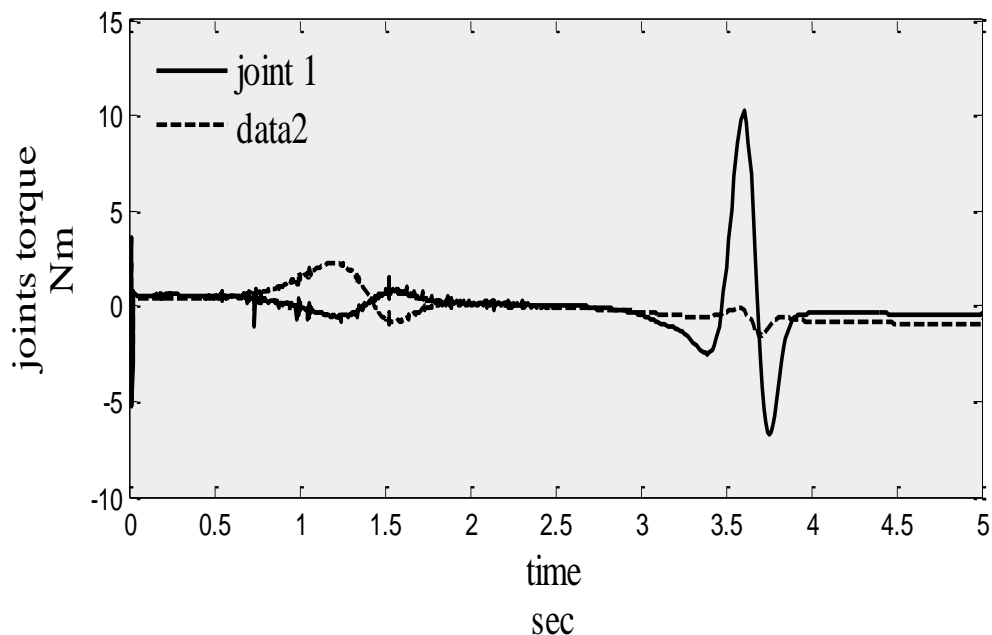
شکل ۴-۱۳ سرعت مجری نهایی طی مسیر تکین در روش دینامیک وارون دکارتی با استفاده از J^T

سرعت مجری نهایی را نشان می دهد که روند خوبی دارد و در بیشینه خود به 2 m/s می رسد.



شکل ۴-۱۴ ولتاژ مفاصل طی مسیر تکین در روش دینامیک وارون دکارتی با استفاده از J^T

همان طور که مشاهده می شود بعد از نقاط تکین تغییر ولتاژی در حد 4v را داریم. شکل ۳-۱۰ گشتاور مفاصل را طی مسیر تکین در روش دینامیک وارون دکارتی با استفاده از J^T نشان می دهد که در نقاط تکین بیشترین گشتاور را داریم.



شکل ۴-۱۵ گشتاور مفاصل در طی مسیر تکین در روش دینامیک وارون دکارتی با استفاده از J^T

مشاهده می شود که با به کار بردن ترانهاده ژاکوبین می توانیم از نقطه تکین مرزی که در ابتدای مسیر قرار دارد ، با ولتاژ $1.5v$ و خطای صفر بگذریم . و این بدیهی است چون ماتریس ترانهاده ژاکوبین در هنگام صفر شدن دترمینان ژاکوبین به مشکل معکوس ناپذیری بر نمی خورد . همچنین در نقطه تکین مرکزی ولتاژ به حدود $4v$ ، گشتاور به $10 Nm$ ، خطای رد گیری مسیر به حدود $0.02m$ و سرعت مفصل به $1.2 rad / s$ می رسد . اما این روش (استفاده از J^t در دینامیک وارون دکارتی) خطای زیادی را علاوه بر نقطه تکین مرکزی در سایر نقاط غیر تکین مسیر دارد . در گام بعد از ماتریس معکوس مجازی ژاکوبین (J^+) ، به جای ماتریس ژاکوبین استفاده می کنیم. ابتدا به ماتریس معکوس مجازی می پردازیم.

۴-۴- ماتریس معکوس مجازی

یک روش پایه برای محاسبه ماتریس معکوس مجازی را تعریف می کنیم . ماتریس معکوس مجازی برای تعیین کنترل بهینه یا الگوریتم های فیلترینگ استفاده می شود [27].
اگر A یک ماتریس $m \times n$ باشد و رتبه آن q باشد ، $m \leq n$ ، (اگر $m > n$ باشد ، ماتریس معکوس مجازی A^T را می توان با توجه به این اصل که $(A^T)^+ = (A^+)^T$ است به دست آورد .)
اگر $q = m$ آنگاه ماتریس معکوس مجازی به صورت زیر به دست می آید :

$$A^+ = A^T (AA^T)^{-1} \quad (13-4)$$

الگوریتمی را که در زیر می آوریم کلی سازی ساده ای از (34) است .
۱) به وسیله عملیات سطری (با پیش ضرب کردن ماتریس P ، ماتریسی $m \times m$ را به دست می آوریم .) ماتریس A را به ماتریس زیر کاهش می دهیم :

$$\begin{bmatrix} A_1 \\ 0 \end{bmatrix} \quad (14-4)$$

جایی که A_1 ماتریسی $q \times n$ است.

$$PA = \begin{bmatrix} A_1 \\ 0 \end{bmatrix} \quad (15-4)$$

۲) ماتریس $q \times m$ را محاسبه می کنیم:

$$C = A_1 A^T \quad (16-4)$$

آنگاه :

$$(A^+)^T = C^+ A_1 \quad (17-4)$$

جایی که

$$C^+ = C^T (CC^T)^{-1} \quad (18-4)$$

و نیز

$$A^+ = A_1^T (CC^T)^{-1} C \quad (19-4)$$

اثبات :

با این ساختار ستون های A_1^T ، رتبه A^T را پوشش می دهند . همچنین $C^T = A_1^T A$ است و ماتریس A یک طرح یک به یک از رتبه A^T درون رتبه A دارد . بنابراین ستون های C^T ، رتبه A را پوشش می دهند ، که از دیمانسیون q است . بنابراین C نیز دارای ماکزیمم رتبه q است . بنابراین (CC^T) رتبه q دارد و معکوس پذیر است .

بنابراین A^+ تعریف شده در رابطه ۴-۷ وجود دارد و ویژگی های زیر را دارد :

(۱) برای همه x های عضو فضای تهی ماتریس A^T داریم : $A^+ Ax = x$
اگر x عضو رتبه A^T باشد ، آنگاه برای $\alpha \in E_q$ خواهیم داشت : $x = A_1^T \alpha$
بنابراین :

$$Ax = AA_1^T \alpha \quad (20-4)$$

و

$$A^+ Ax = A_1^T (CC^T)^{-1} CAA_1^T \alpha \quad (21-4)$$

با استفاده از روابط ۴-۵ و ۴-۶

$$= A_1^T (CC^T)^{-1} CC^T \alpha \quad (22-4)$$

با استفاده از رابطه ۴-۴

$$= A_1^T \alpha \quad (23-4)$$

$$= x$$

(۲) به ازاء همه z های عضو فضای تهی A^T خواهیم داشت:

$$A^+ z = 0 \quad (24-4)$$

اگر z عضو فضای تهی A^T باشد، آنگاه

$$A^T z = 0 \quad (25-4)$$

و

$$A^+ z = A_1^T (CC^T)^{-1} Cz$$

$$= A_1^T (CC^T) A_1 A^T z \quad (26-4)$$

با استفاده از رابطه ۴-۴

$$= 0$$

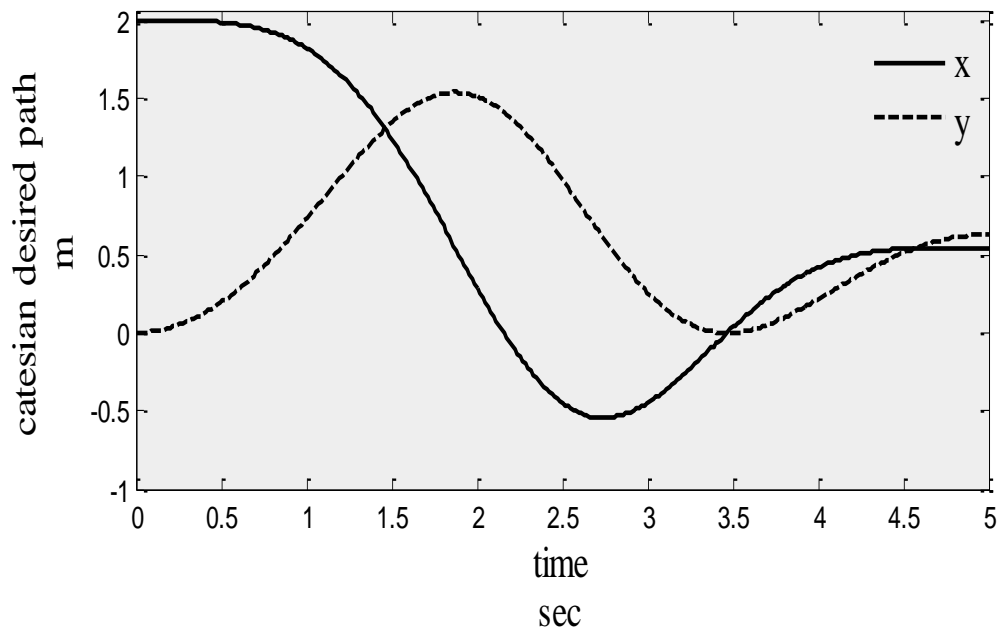
(۳) برای همه y های عضو رتبه A و همه z های عضو فضای تهی A^T خواهیم داشت:

$$A^+(y+z) = A^+y + A^+z$$

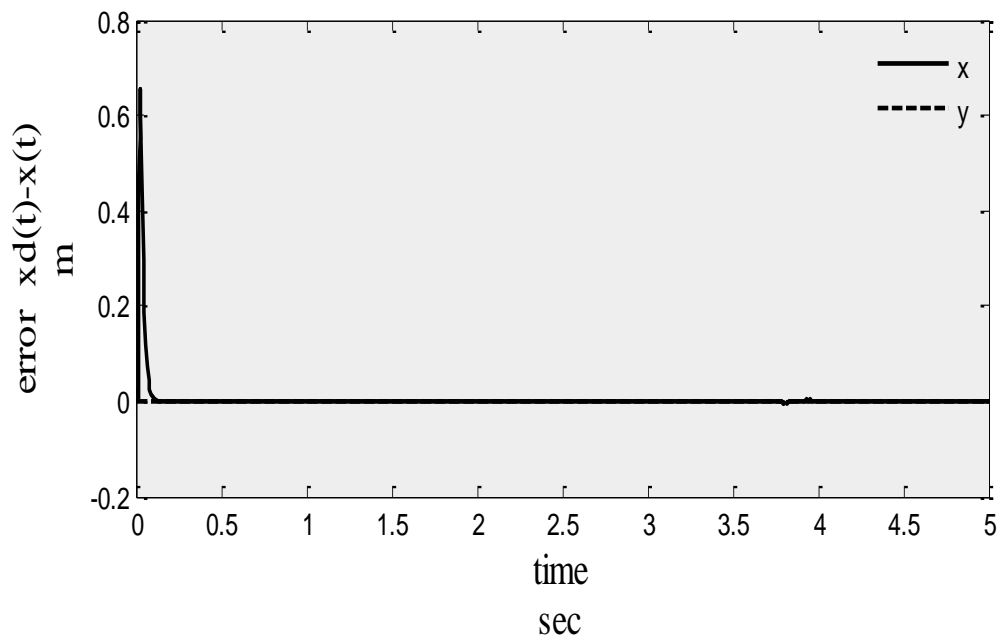
که بدیهی است. بنابراین A^+ تعریف شده با رابطه ۴-۷ معکوس مجازی ماتریس A است.

۴-۵ نتایج روش دینامیک وارون دکارتی با J^+ برای ردگیری مسیر تکین:

همان گونه که در تعریف ماتریس معکوس کاذب مشاهده شد، ماتریس معکوس کاذب در نقطه تکین که دترمینان ماتریس ژاکوبین صفر است و در نتیجه ماتریس ژاکوبین معکوس پذیر نیست، معکوس پذیر می باشد در عین حال عملکرد ماتریس معکوس ژاکوبین را تا حد زیادی دارد [۱] و [۱۵]. اکنون نتایج روش دینامیک وارون دکارتی با J^+ را بررسی می کنیم. نتایج ردگیری مسیر تکین با این روش به صورت زیر است:



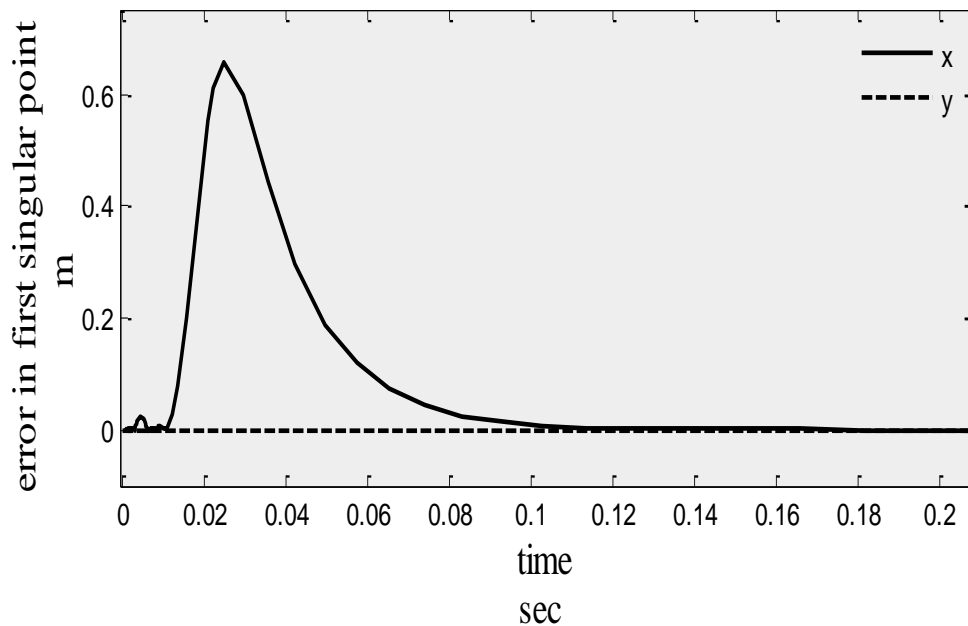
شکل ۴-۱۶ مسیر مطلوب تکین در فضای دکارتی



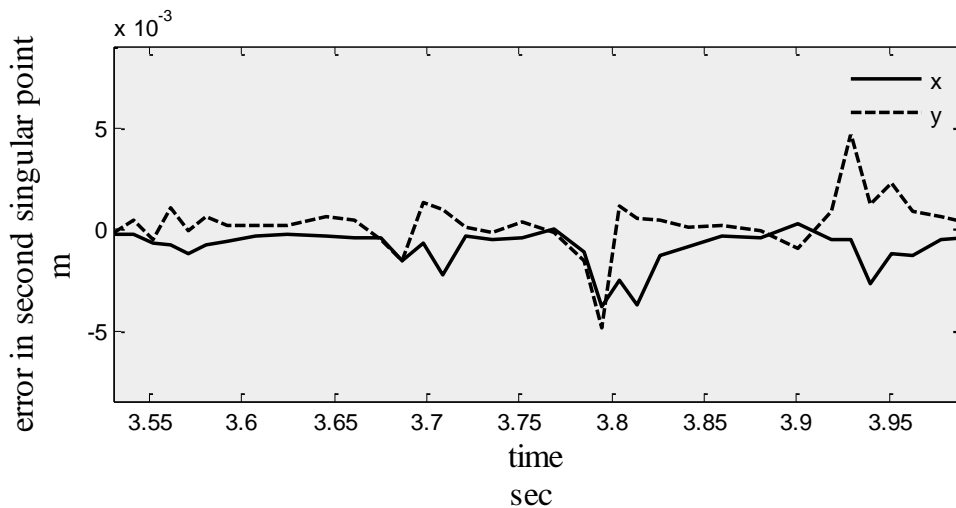
شکل ۴-۱۷ خطای ردگیری در طی مسیر تکین در روش دینامیک وارون دکارتی با استفاده از J^+

مشاهده می شود که خطای ردگیری مسیر در نقاط تکین افزایش می یابد. این افزایش برای نقطه تکین اول (تکین مرزی) بسیار زیادتر است زیرا ربات در این نقطه از حال سکون شروع به حرکت می کند در حالی که در نقطه تکین دوم (تکین مرزی) ربات در حال حرکت دارای ممان اینرسی می باشد. این ممان اینرسی به ربات اجازه می دهد تا با سرعت مفصل زیاد از نقطه تکین دوم

بگذرد. خطای ردگیری مسیر تکین در روش دینامیک وارون دکارتی با استفاده از J^+ و در نقاط تکین را در شکل های ۱۸-۴ و ۱۹-۴ دقیق تر مشاهده می کنید.

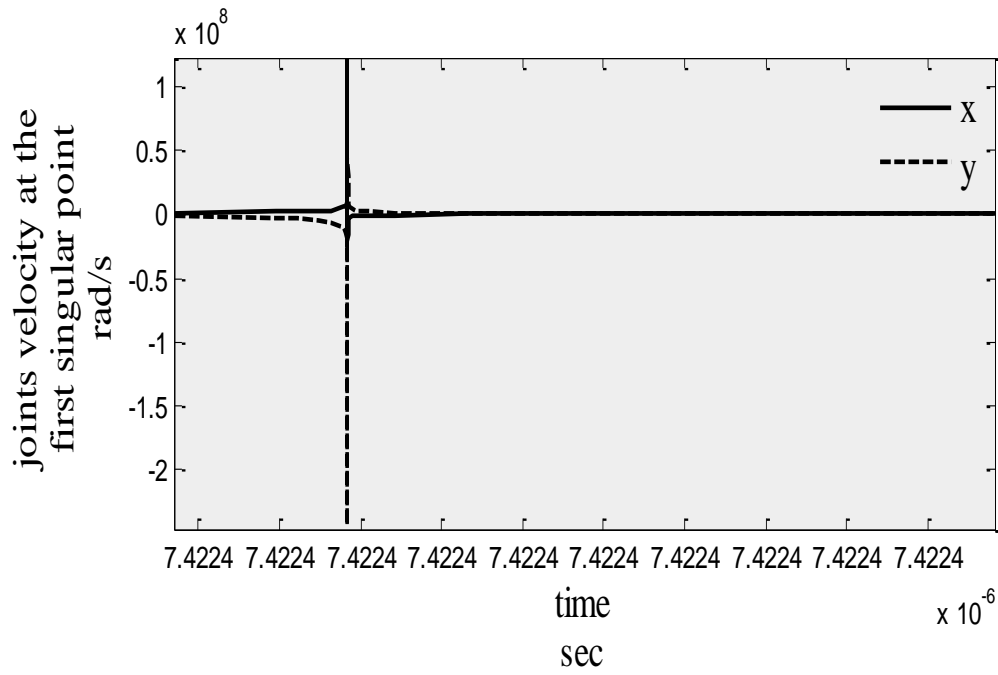


شکل ۱۸-۴ خطای ردگیری در نقطه تکین اول در روش دینامیک وارون دکارتی با استفاده از J^+

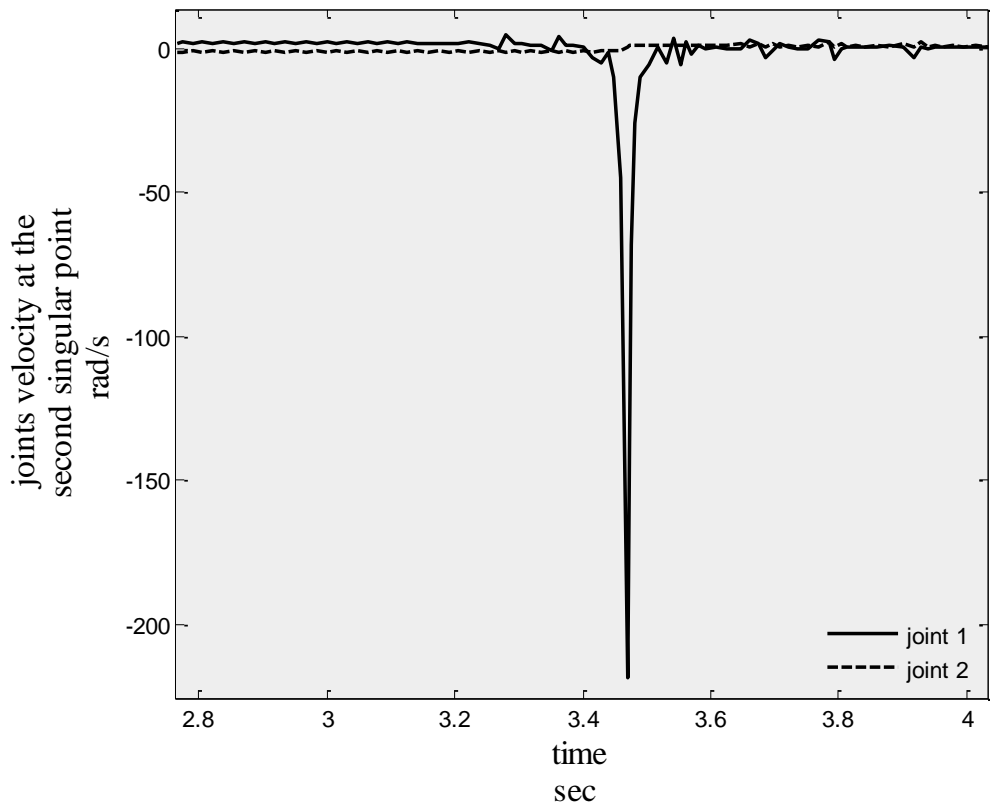


شکل ۱۹-۴ خطای ردگیری در نقطه تکین دوم در روش دینامیک وارون دکارتی با استفاده از J^+

افزایش بی رویه سرعت مفصل اول را به خصوص در نقطه تکین اول، در شکل های ۲۰-۴ و ۲۱-۴ مشاهده می کنید. هر چند سرعت مفاصل در سایر نقاط مسیر مناسب است اما بدیهی است که سرعت های بی رویه در نقاط تکین قابل قبول نیست و باید اصلاح شود. در فصل ۵ به اصلاح آن ها می پردازیم.

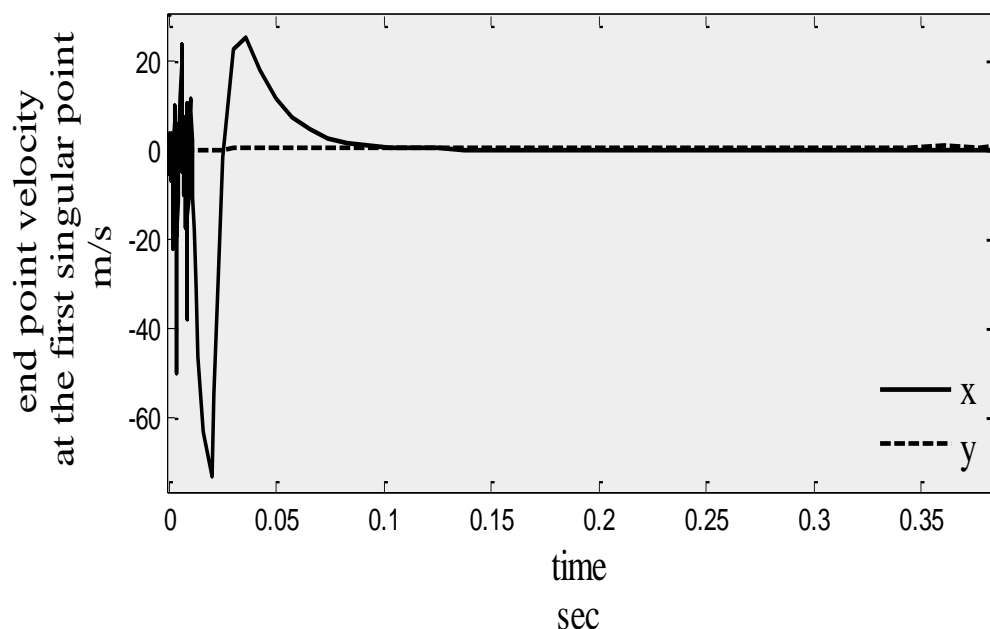


شکل ۴-۲۰ افزایش بی رویه سرعت مفصل اول در نقطه تکین اول در روش دینامیک وارون دکارتی با استفاده از J^+



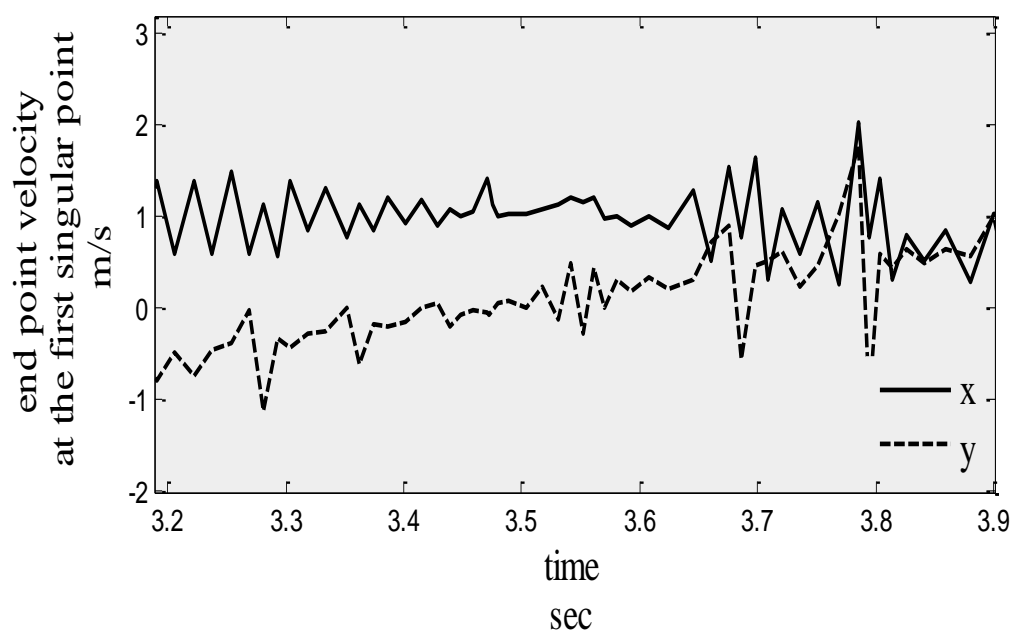
شکل ۴-۲۱ افزایش بی رویه سرعت مفصل اول در نقطه تکین دوم در روش دینامیک وارون دکارتی با استفاده از J^+

سرعت مجری نهایی را در نقاط تکین اول و دوم به ترتیب در شکل های ۲۲-۴ و ۲۳-۴ مشاهده می کنید. سرعت مجری نهایی در نقطه تکین اول به 74 m/s هم می رسد. البته به دلیل ممان اینرسی ربات در حال حرکت سرعت مجری نهایی در نقطه تکین دوم به مراتب کمتر است ، البته باید توجه داشت که این سرعت مجری نهایی با سرعت زیاد مفصل اول تأمین می شود.



شکل ۲۲-۴ سرعت مجری نهایی در نقطه تکین اول در روش دینامیک وارون دکارتی با استفاده از

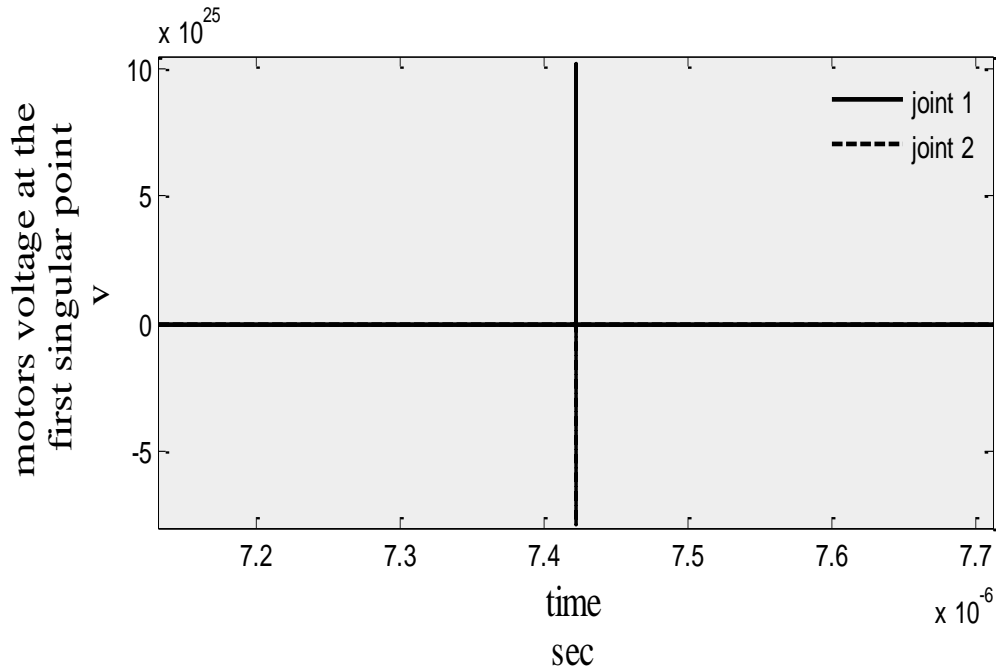
$$J^+$$



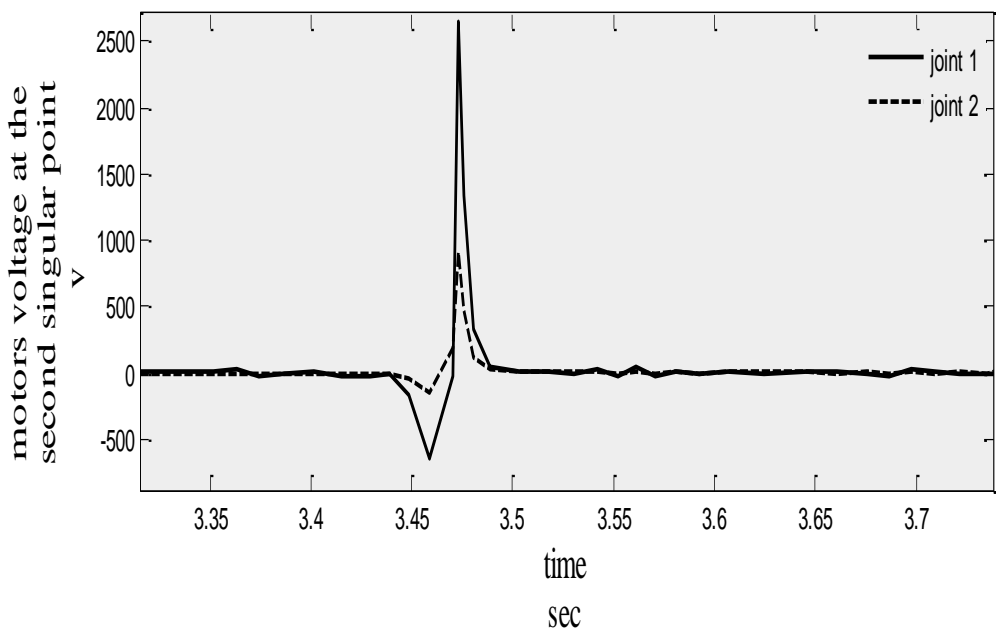
شکل ۲۳-۴ سرعت مجری نهایی در نقطه تکین دوم در روش دینامیک وارون دکارتی با استفاده از

$$J^+$$

شکل های ۲۴-۴ و ۲۵-۴ نیز افزایش بی رویه ولتاژ موتورها را در نقاط تکین نشان می دهند که بدیهی است در محدوده ولتاژ 24v موتورها قرار نمی گیرند و باید اصلاح شوند در فصل ۵ به اصلاح آنها می پردازیم.

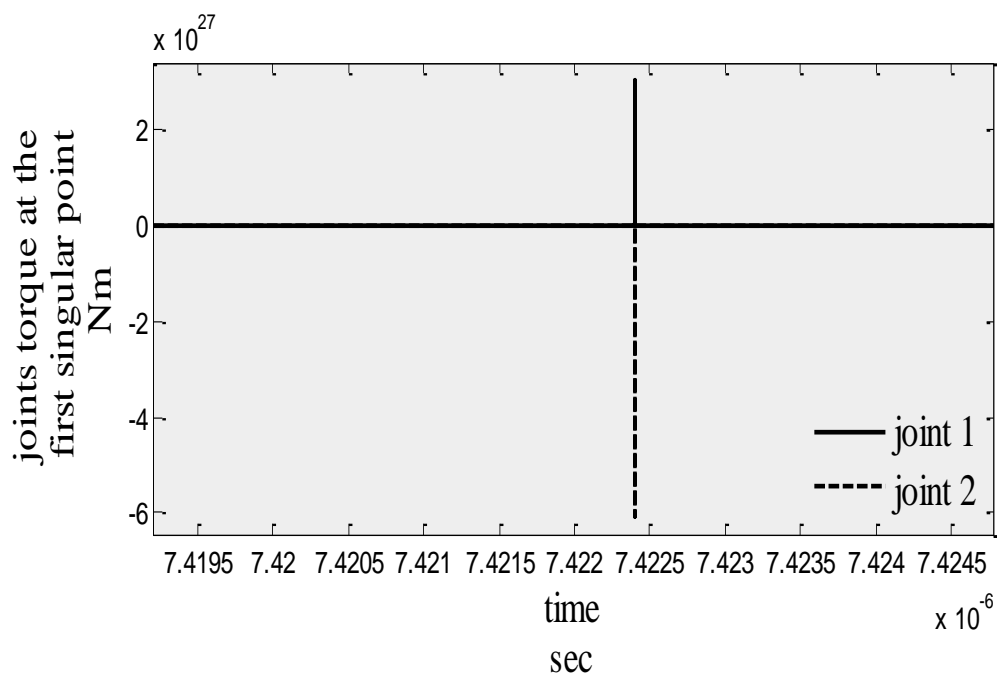


شکل ۲۴-۴ ولتاژ مفاصل در نقطه تکین اول در روش دینامیک وارون دکارتی با استفاده از J^+

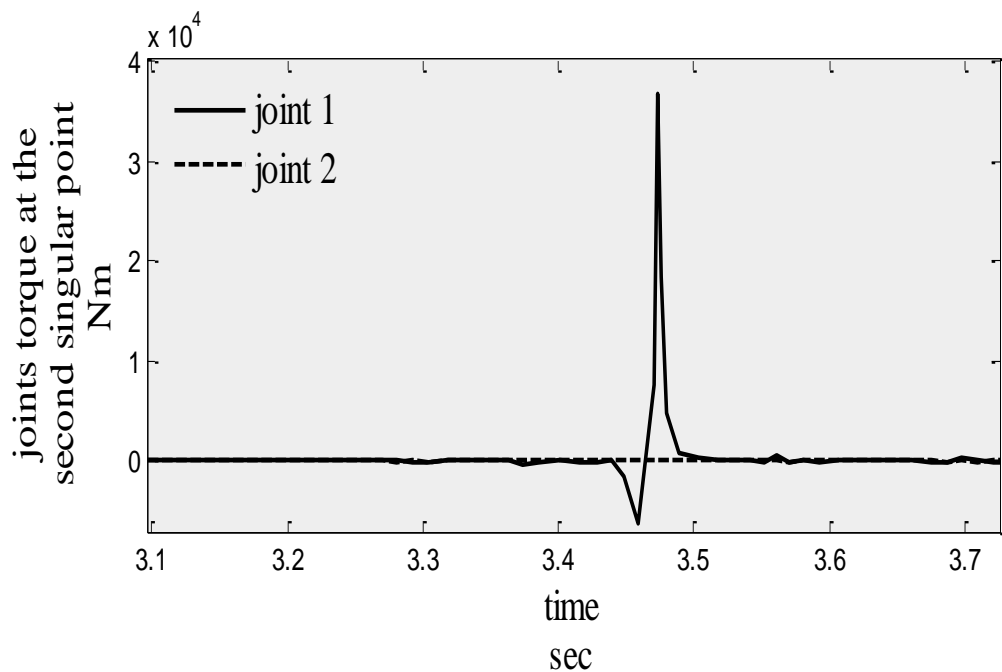


شکل ۲۵-۴ ولتاژ مفاصل در نقطه تکین دوم در روش دینامیک وارون دکارتی با استفاده از J^+

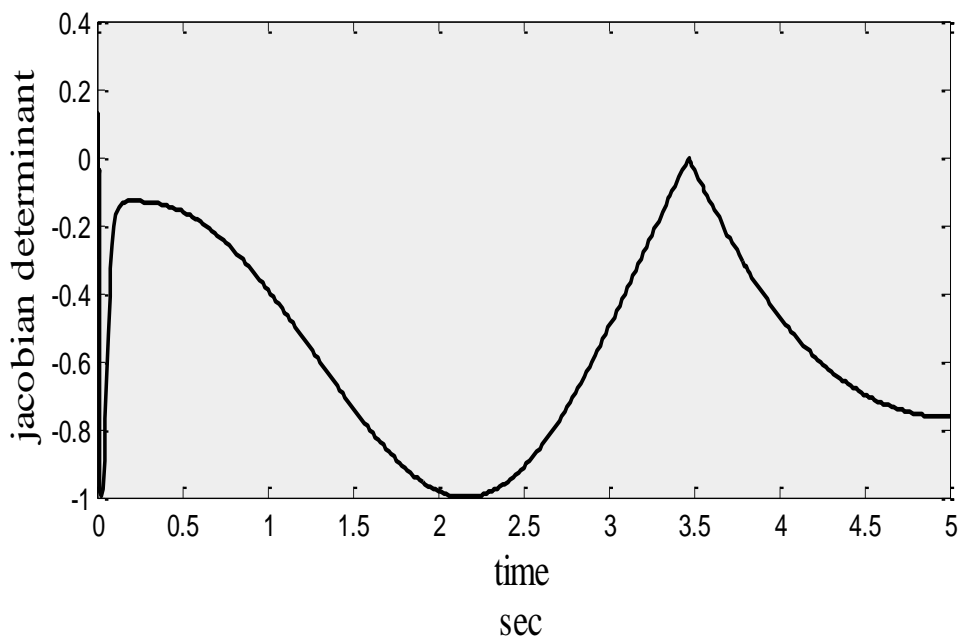
شکل های ۴-۲۶ و ۴-۲۷ نیز افزایش بی رویه گشتاور موتورها را در نقاط تکین نشان می دهند که بدیهی است در محدوده گشتاور موتور DC ۲۴v قرار نمی گیرند و باید اصلاح شوند در فصل ۵ به اصلاح آنها می پردازیم.



شکل ۴-۲۶ گشتاور مفاصل در نقطه تکین اول در روش دینامیک وارون دکارتی با استفاده از J^+



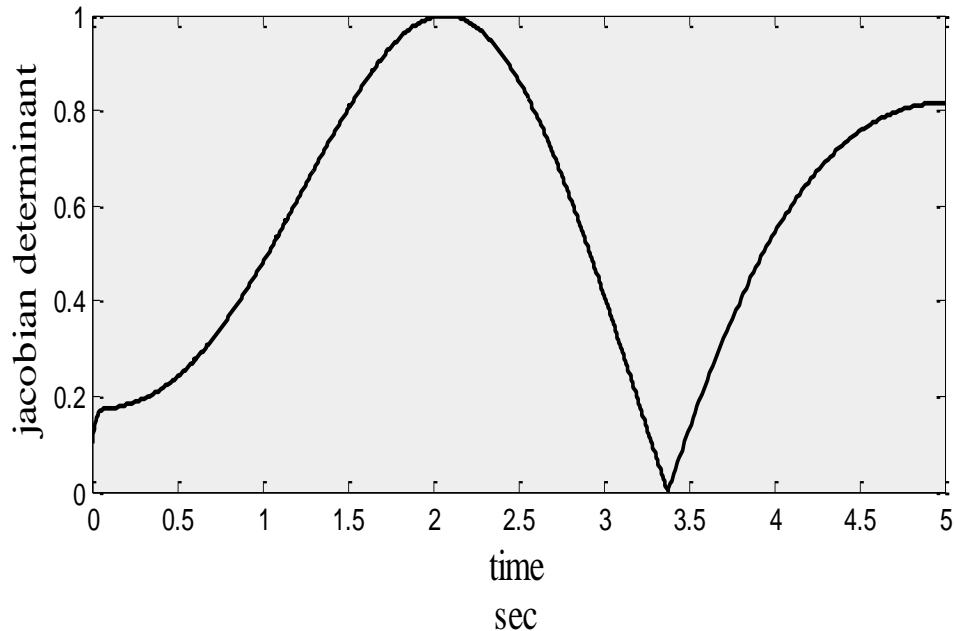
شکل ۴-۲۷ گشتاور مفاصل در نقطه تکین دوم در روش دینامیک وارون دکارتی با استفاده از J^+



شکل ۴-۲۸ ولتاژ مفاصل در نقطه تکین دوم در روش دینامیک وارون دکارتی با استفاده از J^+

شکل ۴-۲۸ دترمینان ماتریس ژاکوبین ربات را در طی مسیر تکین با روش دینامیک وارون دکارتی با استفاده از J^+ نشان می دهد که در آن دترمینان ماتریس ژاکوبین در نقاط تکین صفر است. مشاهده شد که به کار گیری ماتریس معکوس مجازی ژاکوبین به جای ماتریس معکوس ژاکوبین، اگر چه رفتار بهتری در نقاط غیر تکین را نتیجه می دهد و اگر چه باعث می شود تا ردگیری مسیر تکین انجام شود و از نقاط تکین عبور کنیم اما این کار به بهای خطای ردگیری، سرعت، ولتاژ و گشتاور فوق العاده بالای مفاصل در نقاط تکین صورت می گیرد. که این مقادیر به هیچ وجه در محدوده عملکرد موتورهای مفاصل قرار نمی گیرند. این مقادیر در نقطه تکین اول که نقطه تکین مرزی است بسیار بالا است به طوری که مقادیر سرعت، ولتاژ و گشتاور مفاصل در این نقطه در حدود 10^{20} است. اما این مقادیر به هنگام عبور ربات از نقطه تکین دوم که نقطه تکین مرکزی است، به مراتب کمتر است. هر چند که این مقادیر نیز در محدوده عملکرد موتورها قرار نمی گیرند و این کاهش مقادیر نیز به سبب ممان اینرسی ربات در لحظه عبور از نقطه تکین دوم است که به مجری نهایی این اجازه را می دهد که با مقادیر ولتاژ و گشتاور به مراتب کمتری نسبت به نقطه تکین مرزی از نقطه تکین مرکزی عبور کند. در نقطه تکین دوم ولتاژ موتورها به 2500 V ، گشتاور مفاصل به $3.5 \times 10^4\text{ Nm}$ و سرعت مفاصل به 240 rad/s می رسد.

البته می توان با تغییر اندکی در پیکربندی نقطه شروع مسیر نتایج را گرفت که بیشتر قابل مشاهده و بررسی باشد. برای این منظور زاویه ابتدایی مفصل دوم را 0.1rad در نظر می گیریم. این به این معنی است که دترمینان ماتریس ژاکوبین ربات در شروع مسیر دقیقاً صفر نیست. نتایج این تغییر در ادامه می آید.



شکل ۴-۲۹ دترمینان ماتریس ژاکوبین در طی مسیر تکین در روش دینامیک وارون دکارتی

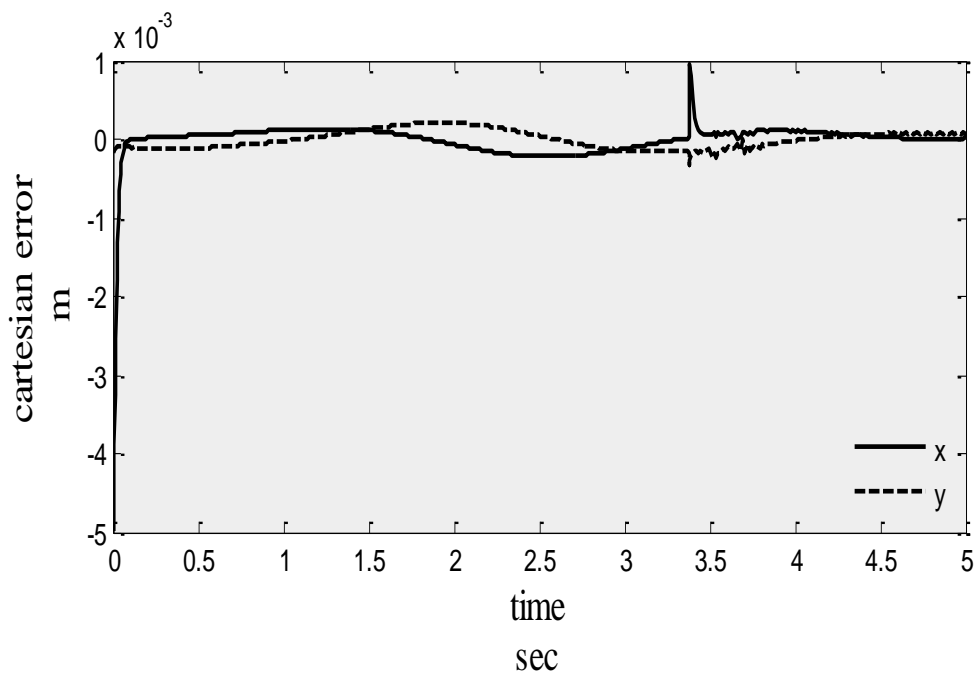
با استفاده از J^+ برای مسیر تکین با نقطه شروع مفصلی به صورت $\begin{bmatrix} 0 \\ 0.1 \end{bmatrix} \text{rad}$

همین طور که در شکل ۴-۲۹ مشاهده می شود، دترمینان ماتریس ژاکوبین ربات در لحظه شروع حرکت صفر نیست بلکه از 0.1 شروع می شود. و این باعث می شود که مشکلات ردگیری در ابتدای مسیر و در نقطه تکین مرزی کمتر شود.

از آن جایی که پیکربندی ربات را در لحظه شروع حرکت، غیر تکین کرده ایم، در شکل ۴-۳۰ مشاهده می شود که خطا در نقطه تکین اول به حد مطلوب 5mm می رسد و در نقطه تکین دوم خطا 1mm است. البته این خطا در نقطه تکین دوم با سرعت زیاد مفاصل نتیجه می شود که سرعت مفاصل در این نقطه را در شکل ۴-۳۳ مشاهده می کنید.

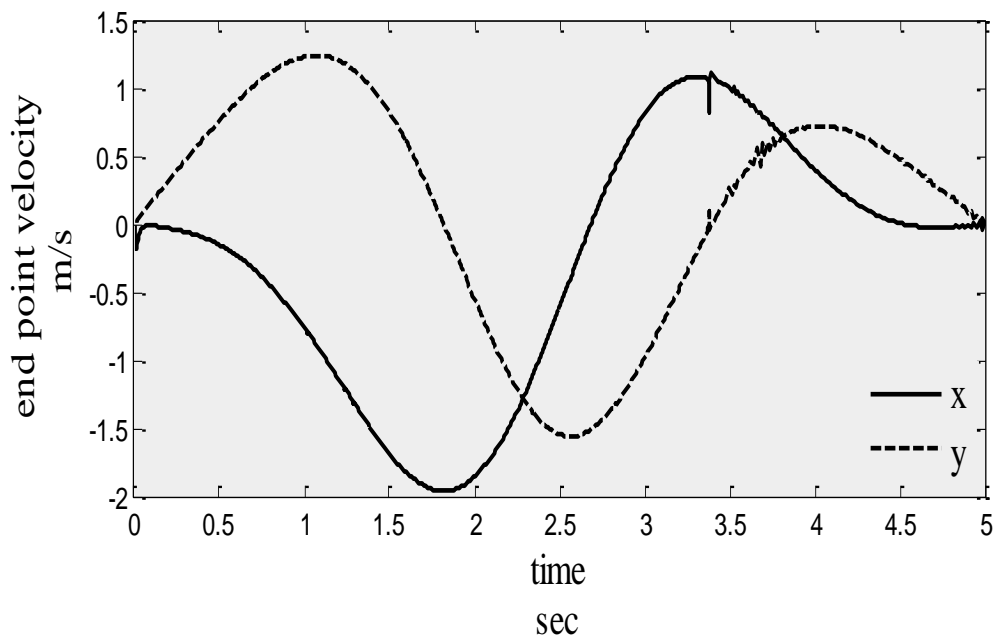
در شکل ۴-۳۱ سرعت مجری نهایی را در طی مسیر تکین با روش دینامیک وارون دکارتی با استفاده از J^+ برای مسیر تکین با نقطه شروع مفصلی به صورت $\begin{bmatrix} 0 \\ 0.1 \end{bmatrix} \text{rad}$ مشاهده می شود. که

روند مطلوبی دارد به ویژه در آغاز مسیر سرعت به صفر رسیده است همچنین سرعت در بیشینه خود به 1.58m/s می رسد.



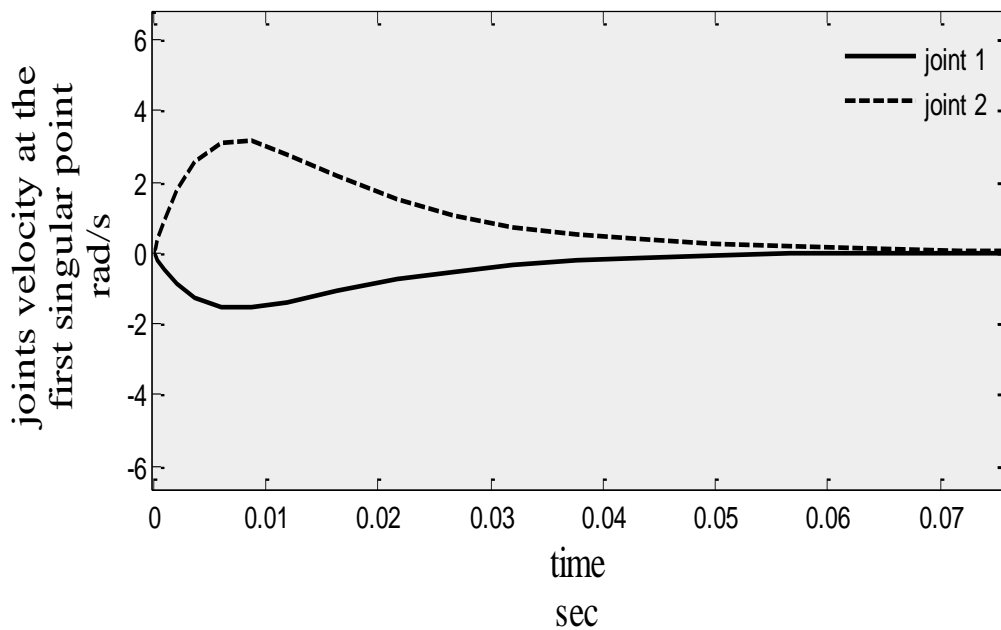
شکل ۴-۳۰ خطای ردگیری در طی مسیر تکین در روش دینامیک وارون دکارتی با استفاده از J^+

برای مسیر تکین با نقطه شروع مفصلی به صورت $\begin{bmatrix} 0 \\ 0.1 \end{bmatrix} rad$



شکل ۴-۳۱ سرعت مجری نهایی در طی مسیر تکین در روش دینامیک وارون دکارتی با استفاده از

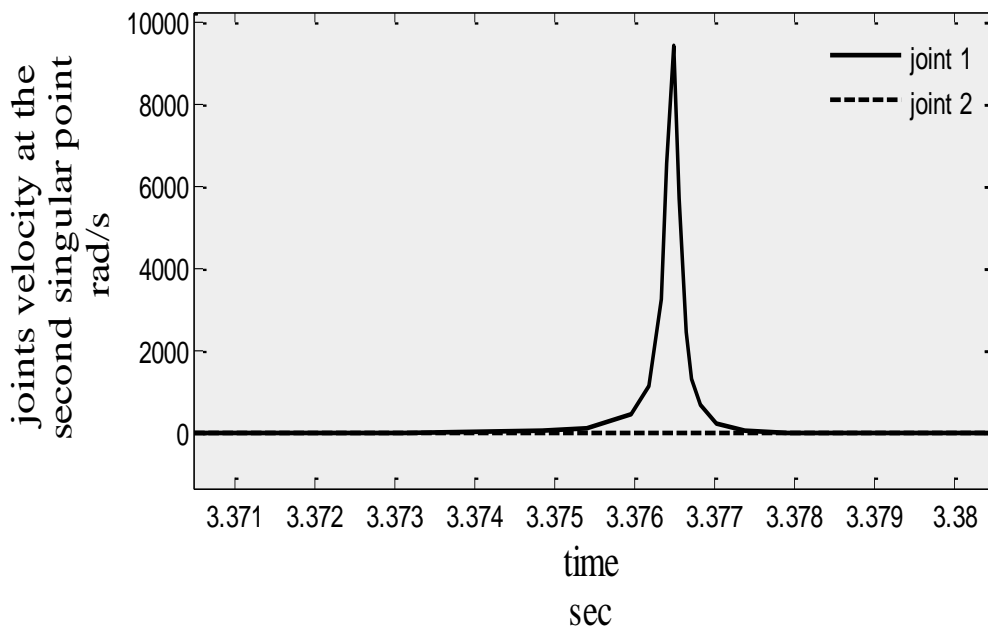
J^+ برای مسیر تکین با نقطه شروع مفصلی به صورت $\begin{bmatrix} 0 \\ 0.1 \end{bmatrix} rad$



شکل ۴-۳۲ سرعت مفاصل در آغاز مسیر در روش دینامیک وارون دکارتی با استفاده از J^+ برای

$$\begin{bmatrix} 0 \\ 0.1 \end{bmatrix} \text{rad}$$

مسیر تکین با نقطه شروع مفصلی به صورت



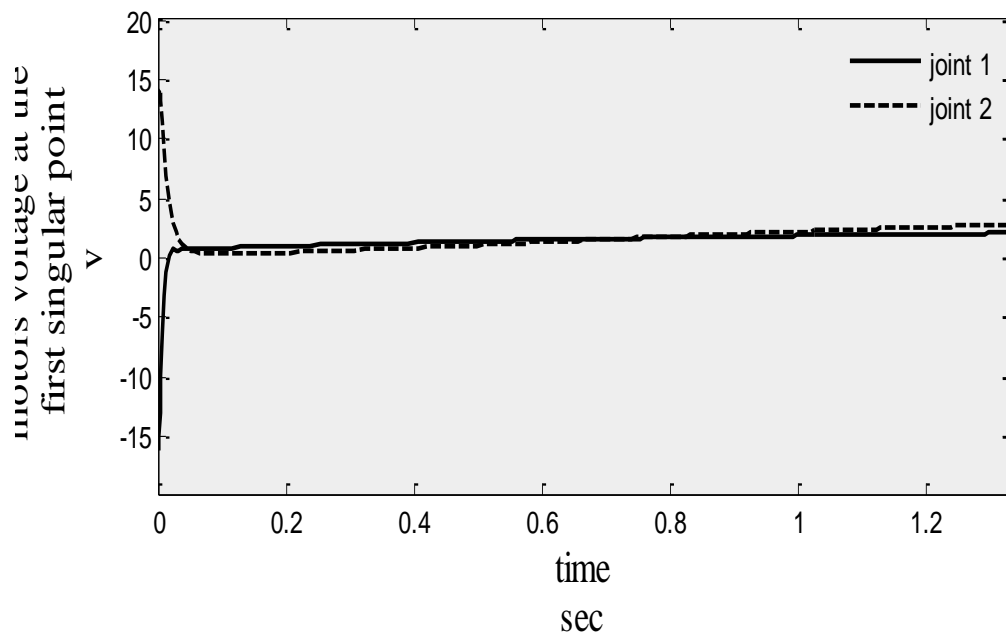
شکل ۴-۳۳ سرعت مفاصل در نقطه تکین دوم با روش دینامیک وارون دکارتی با استفاده از J^+

$$\begin{bmatrix} 0 \\ 0.1 \end{bmatrix} \text{rad}$$

برای مسیر تکین با نقطه شروع مفصلی به صورت

از آن جا که پیکربندی ربات را در لحظه شروع حرکت ، غیر تکین کرده ایم لذا در شکل ۴-۳۲ مشاهده می کنیم که سرعت مفاصل به مراتب کمتر شده و نیز از صفر آغار می شود. اما نقطه

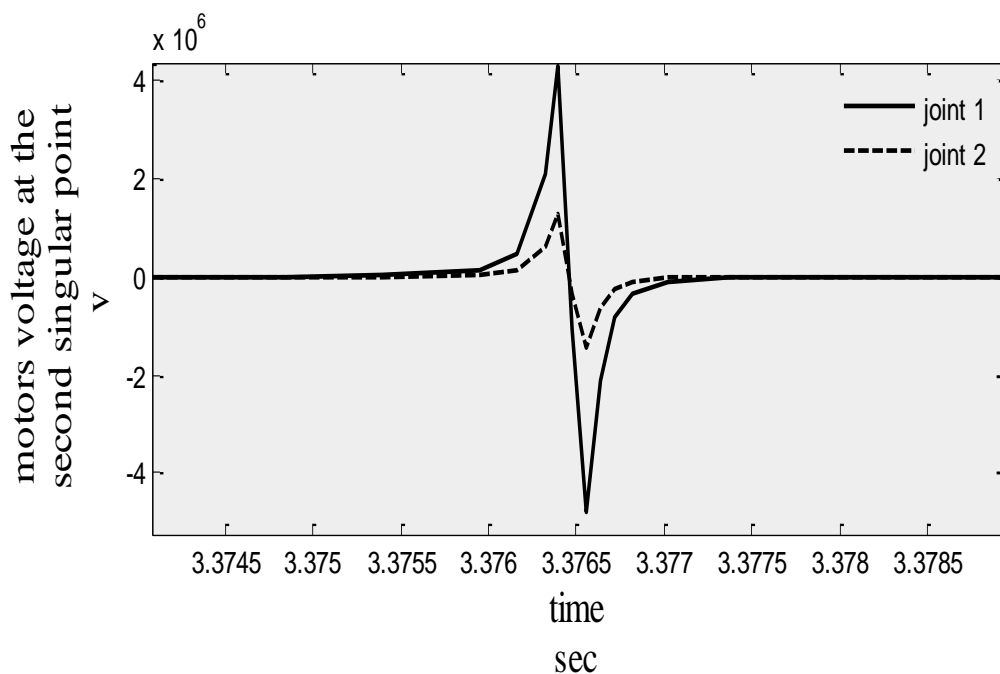
تکین دوم همچنان باقی است و در شکل ۳۳-۴ افزایش بی رویه سرعت مفصل اول در این نقطه مشاهده می شود که باید اصلاح شود. در فصل ۵ به اصلاح این سرعت ها می پردازیم.



شکل ۳۴-۴ ولتاژ مفاصل در آغاز مسیر با روش دینامیک وارون دکارتی با استفاده از J^+ برای مسیر

$$\begin{bmatrix} 0 \\ 0.1 \end{bmatrix} \text{ rad}$$

تکین با نقطه شروع مفصلی به صورت

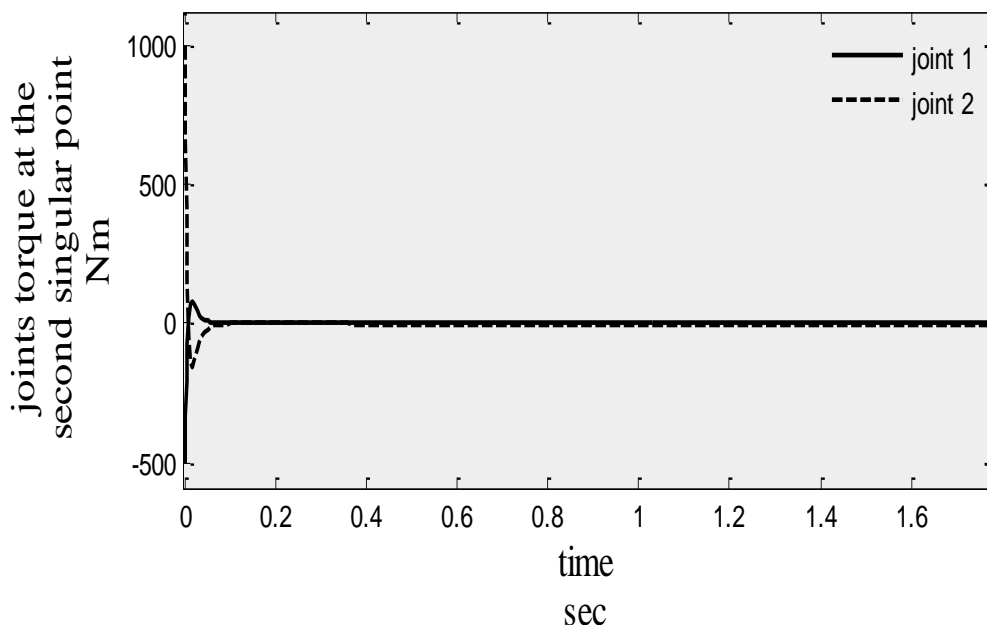


شکل ۳۵-۴ ولتاژ مفاصل در نقطه تکین مرکزی با روش دینامیک وارون دکارتی با استفاده از J^+

$$\begin{bmatrix} 0 \\ 0.1 \end{bmatrix} \text{ rad}$$

برای مسیر تکین با نقطه شروع مفصلی به صورت

از آن جا که پیکربندی ربات را در لحظه شروع حرکت ، غیر تکین کرده ایم لذا در شکل ۴-۳۴ مشاهده می کنیم که ولتاژ به مراتب کمتر شده و در آغاز ۱۵۷ است که در محدوده ۲۴۷ موتورها قرار دارد. اما نقطه تکین دوم همچنان باقی است و در شکل ۴-۳۵ افزایش بی رویه ولتاژ مفصل اول مشاهده در این نقطه مشاهده می شود که باید اصلاح شود. در فصل ۵ به اصلاح این ولتاژ می پردازیم.

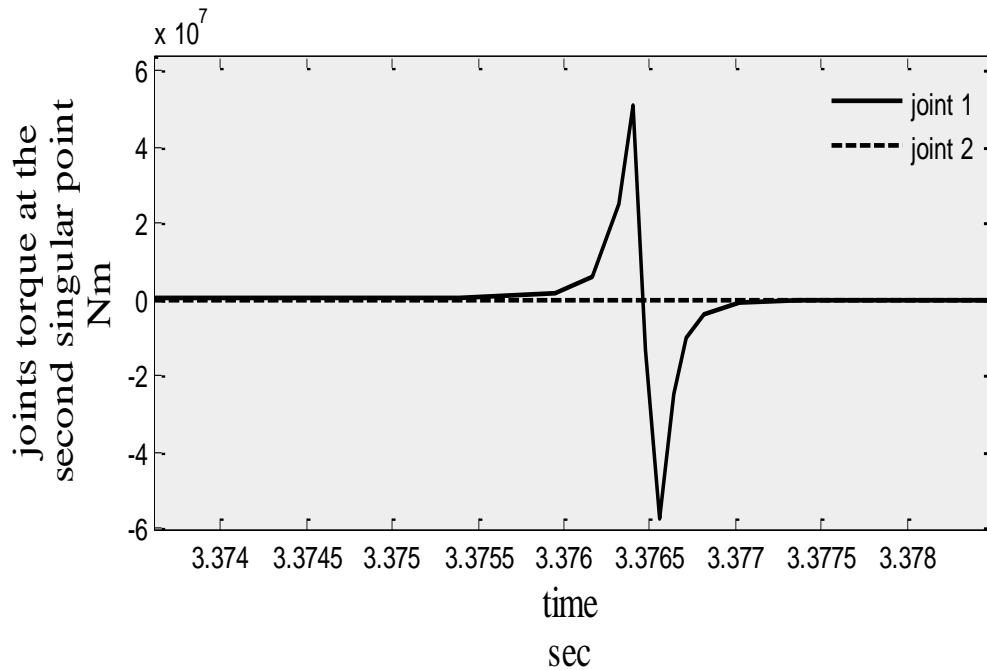


شکل ۴-۳۶ گشتاور مفاصل در آغاز مسیر با روش دینامیک وارون دکارتی با استفاده از J^+ برای

$$\begin{bmatrix} 0 \\ 0.1 \end{bmatrix} \text{rad}$$

مسیر تکین با نقطه شروع مفصلی به صورت

شکل های ۴-۳۶ و ۴-۳۷ گشتاور مفاصل را به ترتیب در آغاز مسیر و در نقطه تکین مرکزی نشان می دهند که مشاهده می شود در آغاز مسیر گشتاور مفاصل در بیشینه خود به دلیل ولتاژ اولیه ۱۵۷ موتورها به $1000 Nm$ نیز می رسد. این ولتاژ اولیه نیز به دلیل مقدار دترمینان ژاکوبین کم (0.1) در آغاز مسیر است. در فصل بعد و در روش کنترل سازگار با تکین که ارائه خواهیم کرد گشتاورها نیز در مقدار مطلوبی محدود می شوند.



شکل ۴-۳۷ گشتاور مفاصل در نقطه تکین مرکزی با روش دینامیک وارون دکارتی با استفاده از J^+

$$\begin{bmatrix} 0 \\ 0.1 \end{bmatrix} \text{rad}$$

برای مسیر تکین با نقطه شروع مفصلی به صورت

مشاهده می شود که با تغییر اندکی در زاویه مفصل دوم در شروع حرکت ربات و در واقع غیر صفر کردن دترمینان ماتریس ژاکوبین در ابتدای مسیر، سیستم کنترل دینامیک وارون در فضای دکارتی با J^+ در آغاز مسیر با خطای بیشینه 3 mm و سرعت بیشینه مفصل 3.2 rad/s عبور می کند. اما گشتاور بیشینه در این لحظه 1000 Nm است که همچنان مقدار زیادی برای موتورها می باشد.

در مورد نقطه تکین دوم نیز مشاهده می شود که همچنان مقادیر ولتاژ، گشتاور و سرعت مفاصل در این نقطه از محدوده عملکرد موتورها بیشتر است. هرچند که خطای دکارتی 1 mm را داشته باشیم. بدیهی است که در عمل محدوده عملکرد موتورها اجازه رسیدن به چنین خطایی را به ما نمی دهد.

در فصل بعد روش کنترلی سازگار با نقاط تکین را ارائه می کنیم.

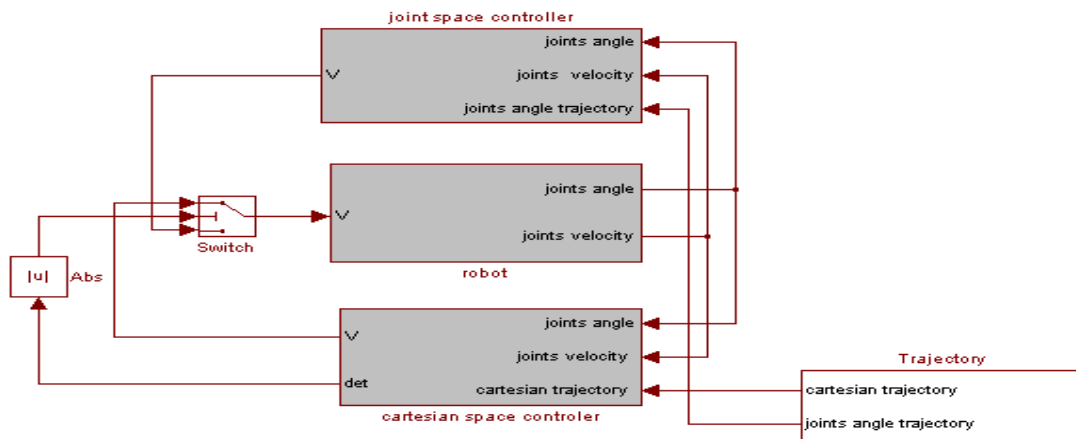
فصل ۵

روش کنترل سازگار با نقاط تکین

روش کنترل سازگار با نقاط تکین

همان طور که در فصل پیش دیدیم در حالی که ما راغب بودیم کنترلی در فضای دکارتی داشته باشیم تا مجری نهایی را به طور کامل در کنترل داشته باشیم. اما مشاهده کردیم که کنترل دینامیک وارون در فضای دکارتی با مشکل نقاط تکین روبرو است. به طوری که در نقاط تکین با افزایش بی رویه ولتاژ، گشتاور و سرعت مفاصل روبرو هستیم. از طرفی در روش کنترل وارون مفصلی کنترلی بر مجری نهایی نداریم.

از این رو مناسب دیدیم که دو روش دینامیک وارون در فضای مفصلی و دینامیک وارون در فضای دکارتی را با هم سوییچ کنیم. البته در سوییچینگ این دو کنترلر از آن جا که هر یک قانون کنترل جداگانه ای دارند، باید مواردی را در نظر گرفت. از جمله این که ورودی های کنترل در هر دو کنترل باید با هم هم نوایی داشته باشند و برای این منظور ضرایب k_p و k_v در دو سیستم باید به طور مناسبی تنظیم شود.



شکل ۵-۱ شمای کلی سیستم کنترل سازگار با نقاط تکین

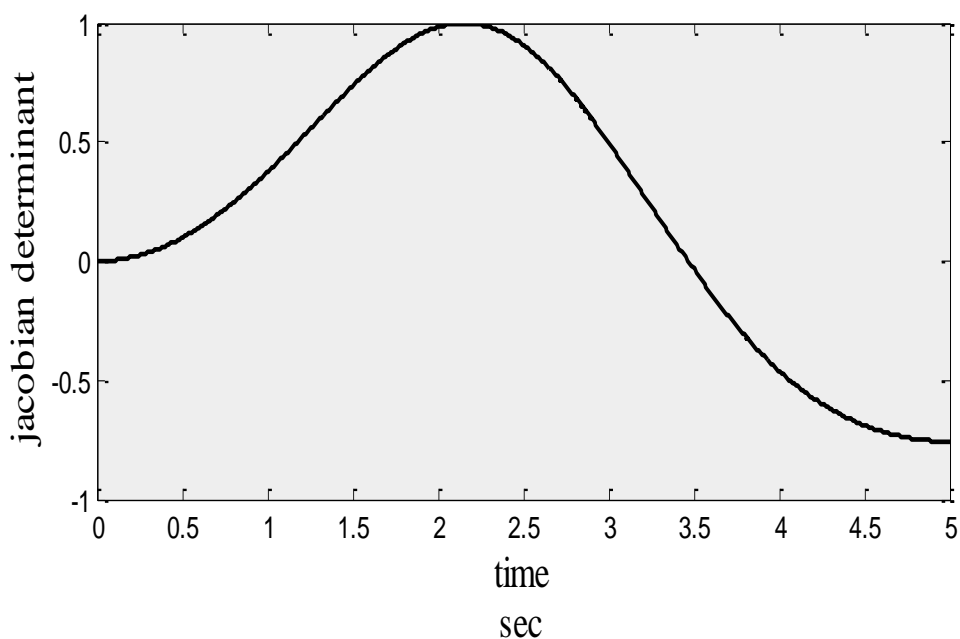
که بلوک های کنترل فضای مفصلی و فضای دکارتی در آن در فصل ۳ آمده است. در این روش شرط سوییچینگ را روی دترمینان ماتریس ژاکوبین ربات می گذاریم. به این گونه که هر گاه دترمینان ماتریس ژاکوبین به اندازه ای که اشاره خواهد شد، به صفر نزدیک شود، از کنترلر

دینامیک و ارون دکارتی به کنترلر دینامیک و ارون مفصلی سویچینگ صورت گیرد. زیرا این شرط به معنی معکوس ناپذیری ماتریس ژاکوبین ربات است و در کنترلر دینامیک و ارون دکارتی تأثیرات نامطلوبی را که قبلاً ذکر شد می گذارد.

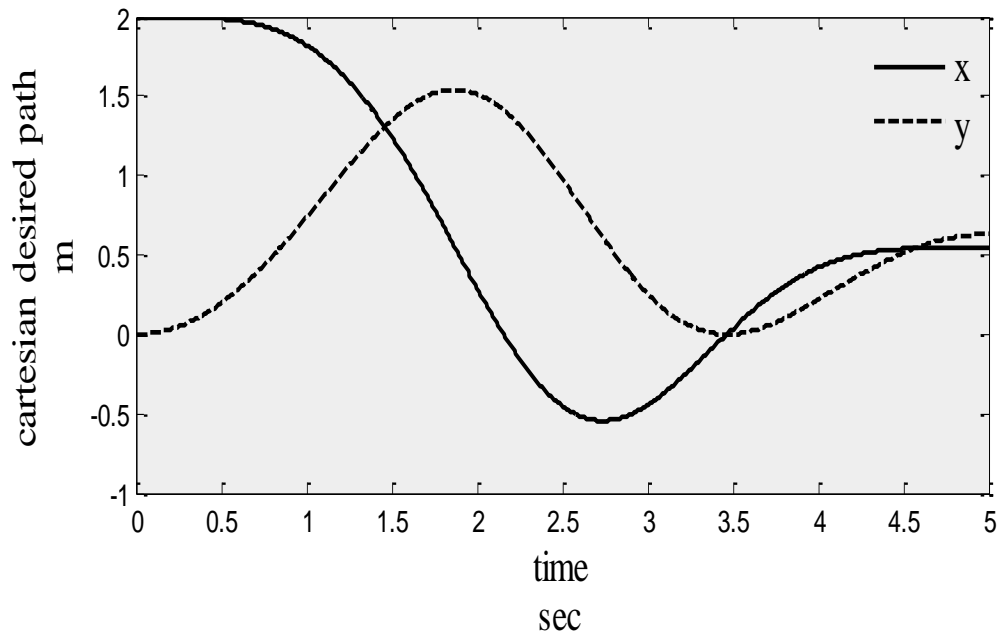
شمای کلی روش کنترل سازگار با نقاط تکین به صورت زیر است:

برای استفاده از این روش کنترل k_p و k_v را در کنترلر دینامیک و ارون مفصلی به ترتیب 10000 و 200 و در کنترلر دینامیک و ارون دکارتی 40000 و 400 قرار می دهیم. همچنین محدوده دترمینان ماتریس ژاکوبین را در شرط سویچینگ 0.1 قرار می دهیم.

نتایج روش سازگار با نقاط تکین برای مسیر تکین:

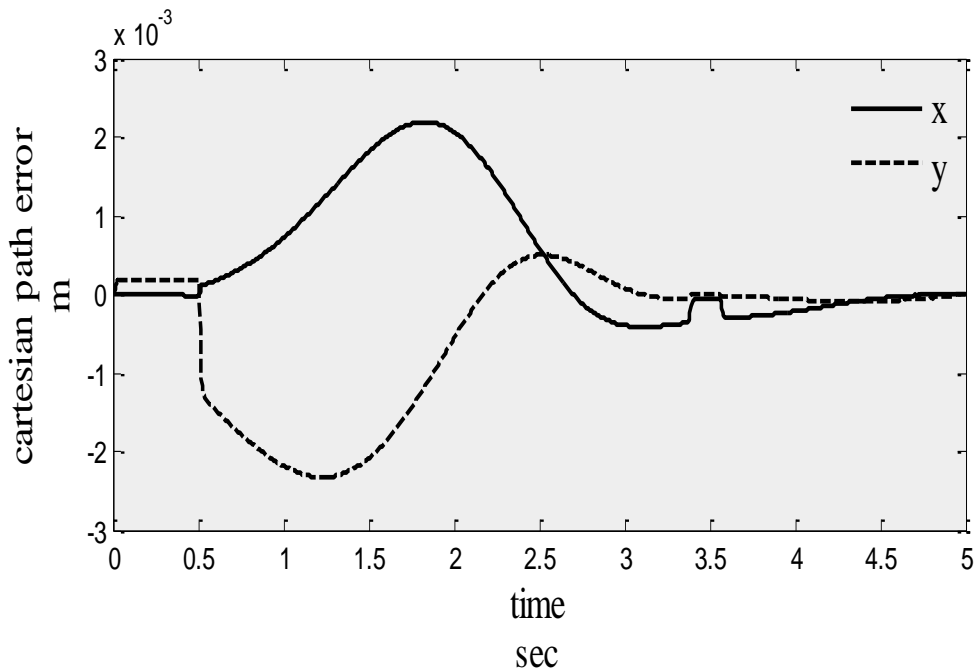


شکل ۲-۵ دترمینان ماتریس ژاکوبین ربات در طی مسیر تکین در روش سازگار با نقاط تکین همان طور که مشاهده می شود ، ربات دقیقاً از یک نقطه تکین مرزی شروع می شود.

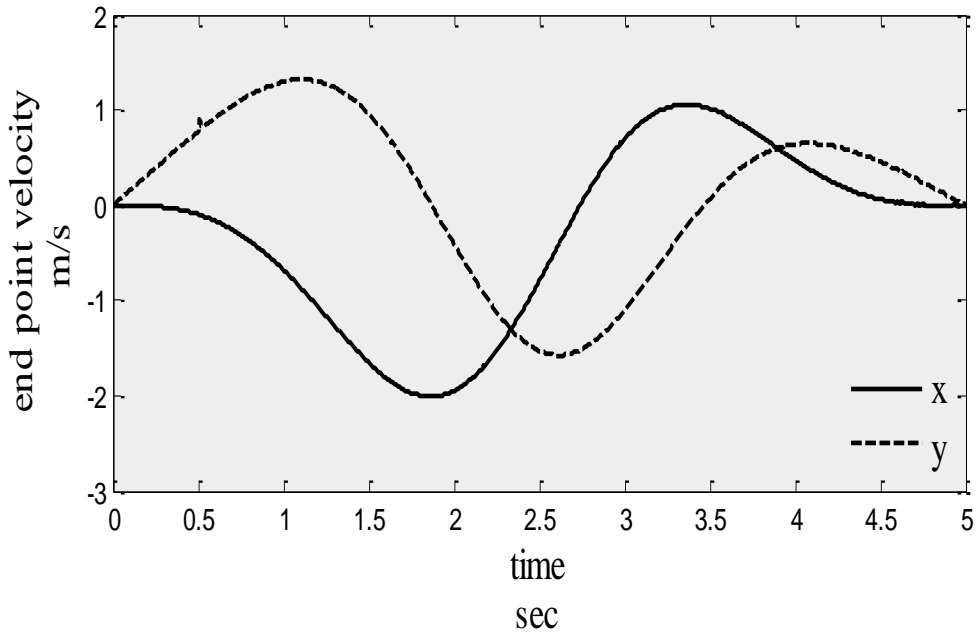


شکل ۳-۵ مسیر دکارتی مطلوب تکین

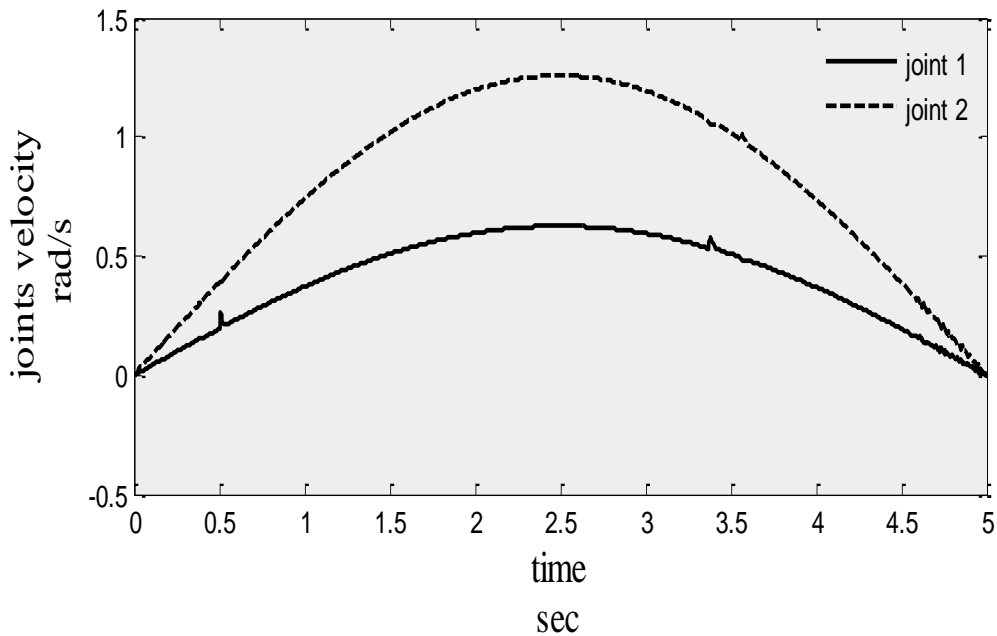
در شکل ۴-۵ خطای ردگیری مسیر تکین مشاهده می شود. مشخص است که خطای ردگیری مجری نهایی در این روش در نقاط تکین خیلی نزدیک به صفر است. همچنین در نقاط غیر تکین خطای ردگیری در بیشینه خود به حدود 2mm می رسد. که اگر این خطا را با شکل های ۴-۱۱ و ۴-۱۷ و ۴-۳۰ که خطاهای ردگیری در روش دینامیک وارون دکارتی در حالات مختلف است مقایسه کنیم به برتری قابل ملاحظه روش کنترل سازگار با نقاط تکین پی می بریم.



شکل ۴-۵ خطای ردگیری مجری نهایی در مسیر تکین با روش سازگار با حالات تکین



شکل ۵-۵ سرعت مجری نهایی در طی مسیر تکین با روش سازگار با حالات تکین در شکل ۳-۵ سرعت مجری نهایی را مشاهده می کنید که مشابه سرعت مجری نهایی در روش دینامیک وارون در فضای دکارتی برای مسیر تکین است با این تفاوت که این سرعت مطلوب این بار با سرعت مطلوب مفاصل، همچنین ولتاژ و گشتاوری در محدوده عملکرد موتورها حاصل می شود.

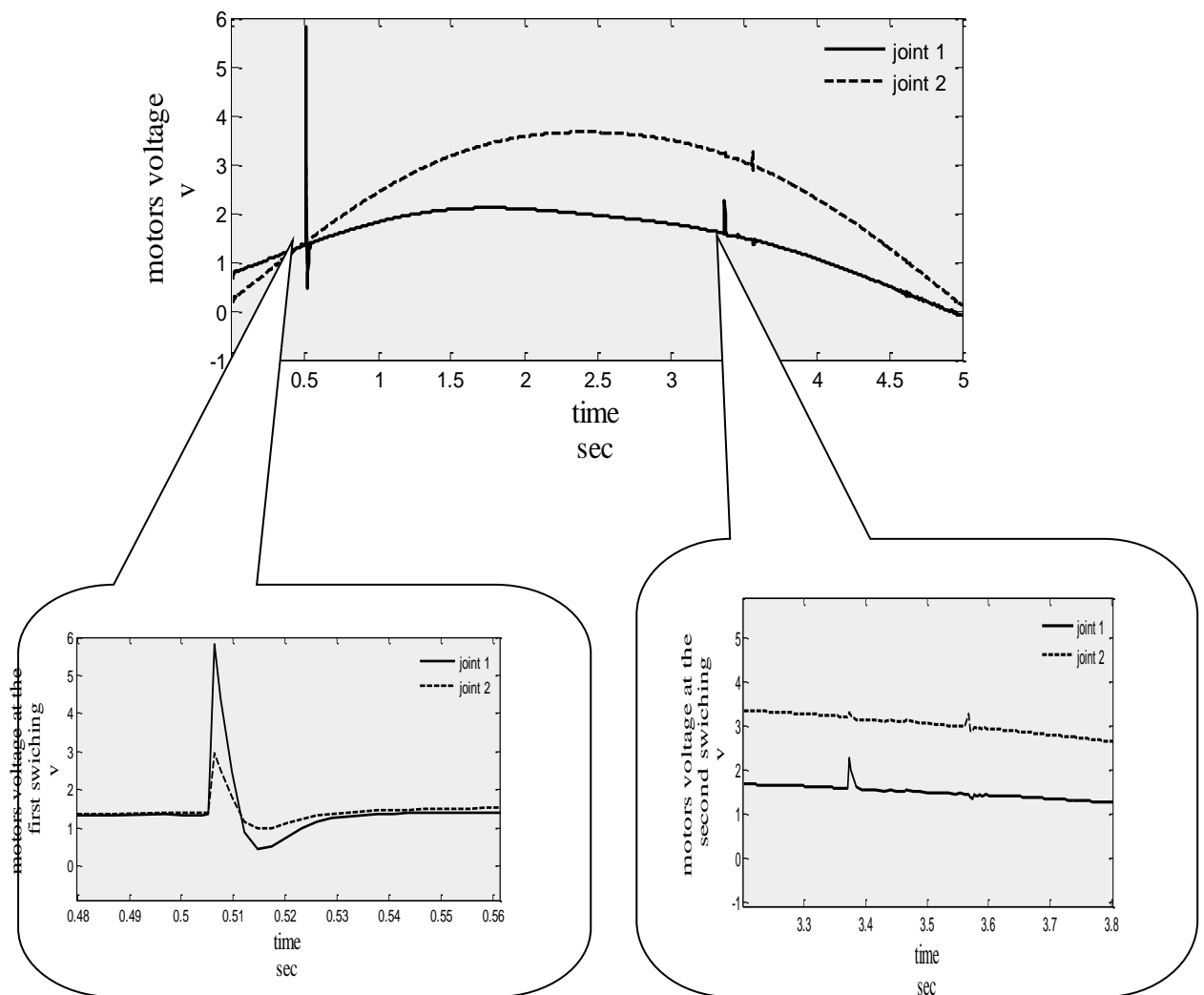


شکل ۶-۵ سرعت مفاصل در طی مسیر تکین در روش سازگار با حالات تکین

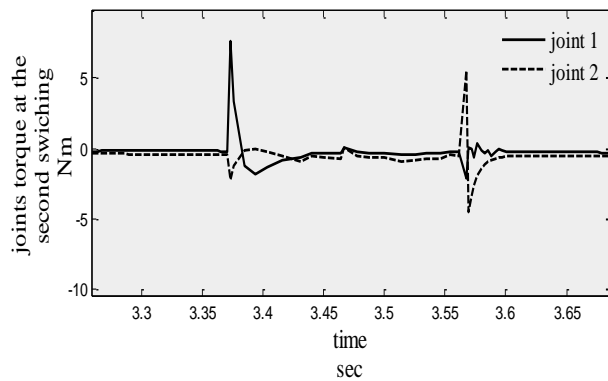
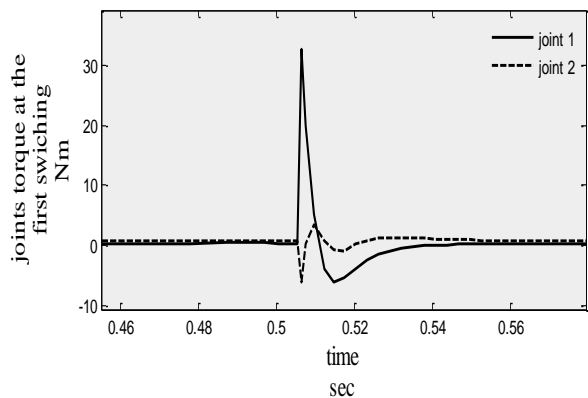
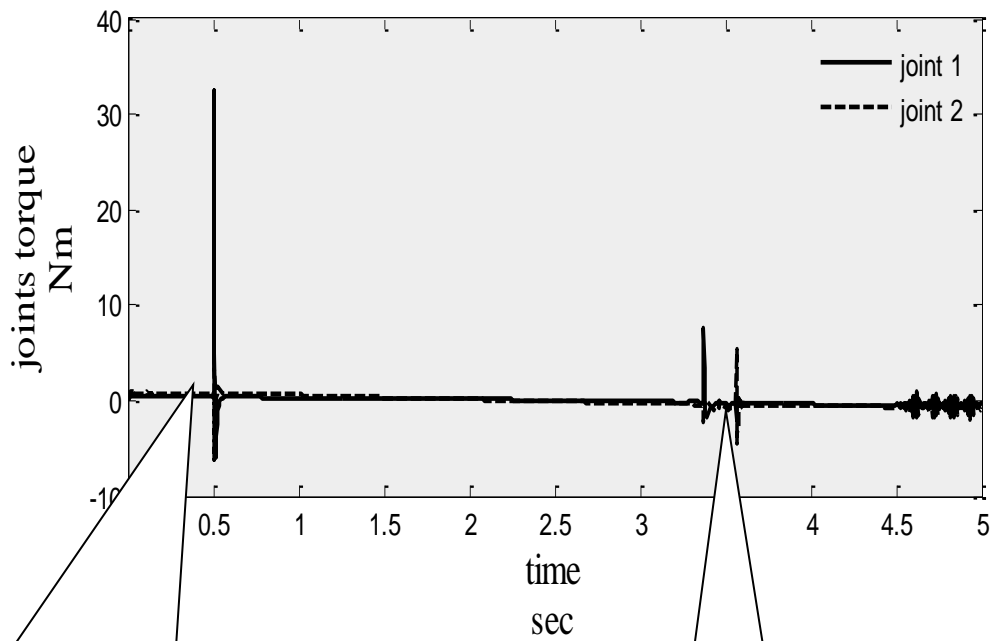
شکل ۶-۵ سرعت مفاصل را در روش سازگار با حالت تکین نشان می دهد. بامقایسه شکل ۵-۶ با شکل های ۴-۱۲، ۴-۲۰، ۴-۲۱ و ۴-۳۳ که تغییرات سرعت مفاصل در روش دینامیک وارون دکارتی در حالات مختلف هستند، می توان به برتری قابل ملاحظه روش سازگار با حالت تکین پی

برد به طوری که فقط افزایش ناچیزی در سرعت مفصل اول دیده می شود که در محدوده عملکرد موتورها است.

شکل ۵-۷ ولتاژ مفاصل را در طی مسیر تکین با روش سازگار با حالات تکین نمایش می دهد. مشاهده می شود که تا قبل از اولین سویچینگ مقدار ولتاژ مفاصل روند خوبی را تا 1.2V طی می کند. در لحظه سویچینگ اول است که در بیشترین مقدار، افزایش ولتاژی به اندازه 5V را در مفصل اول داریم. البته این افزایش ولتاژ بر خلاف حالات مختلف روش دینامیک وارون در فضای دکارتی که نتایج آن پیش تر در شکل های ۴-۲۴، ۴-۲۵، ۴-۳۴ و ۴-۳۵ آمد خیلی کمتر است و در محدوده موتور 24V DC قرار دارد. در نقطه تکین دوم نیز که سویچینگ دوم روی می دهد افزایش ولتاژ کمی به اندازه 1V را داریم.



شکل ۵-۷ ولتاژ مفاصل در طی مسیر تکین با روش سازگار با حالات تکین



شکل ۵-۸ گشتاور مفاصل در طی مسیر تکین با روش سازگار با حالات تکین

شکل ۵-۸ گشتاور مفاصل را در طی مسیر تکین به روش سازگار با نقاط تکین نمایش می دهد. مشاهده می شود که تا قبل از اولین سویچینگ مقدار گشتاور مفاصل روند خوبی را طی می کند. در لحظه سویچینگ اول است که در بیشترین مقدار، افزایش گشتاوری به اندازه $30 Nm$ را در مفصل اول داریم. البته این افزایش گشتاور بر خلاف حالات مختلف روش دینامیک وارون در فضای دکارتی که نتایج آن پیش تر در شکل های ۴-۲۶، ۴-۲۷، ۴-۳۶ و ۴-۳۷ آمد خیلی کمتر است و با ضرایب چرخ دنده $r = 0.01$ توسط موتور $24V DC$ می تواند تولید شود. در نقطه تکین دوم نیز که سویچینگ دوم روی می دهد افزایش گشتاور کمی به اندازه $5 Nm$ را داریم. مشاهده شد، با به کار گیری این روش کنترلی سازگار با نقاط تکین، خطای ردگیری مسیر تکین در محدوده نقاط تکین تقریباً صفر است و در همین حال در نقاط غیر تکین این مسیر، خطای ردگیری در بیشینه خود به $2.5mm$ می رسد. همچنین ولتاژ مفاصل نیز به خوبی در محدوده ولتاژ موتورها مهار می شود. به طوری که در هنگام سویچینگ اول بین کنترلر دینامیک وارون مفصلی و کنترلر دینامیک وارون دکارتی افزایش ولتاژی به اندازه $5.8V$ را داریم که در مقایسه با ولتاژ روش کنترل دینامیک وارون به تنهایی، در نقطه تکین اول بسیار ناچیز است. همچنین در نقطه تکین دوم (مرکزی) به دلیل وجود ممان اینرسی موجود در ربات در حال حرکت ولتاژ از این هم کمتر است و در هنگام سویچینگ دوم از کنترلر دینامیک وارون مفصلی به کنترلر دینامیک وارون دکارتی افزایش ولتاژی به اندازه $1V$ را داریم. بار دیگر به این نتیجه می رسیم که ردگیری مسیر تکین به هنگام عبور از نقطه تکین از نوع مرکزی ساده تر از ردگیری این مسیر در هنگام عبور از نقطه تکین از نوع مرزی است. لازم به ذکر است که با توجه به نمودار ۴-۳ در نقاط تکین افزایش بی رویه سرعت مفاصل را نیز نداریم و سرعت مفاصل خیلی خوب در نقاط تکین مهار شده است.

نتیجه گیری

نتیجه گیری

روش جدیدی برای کنترل ربات در نقاط تکین معرفی و بررسی شد. در این روش جدید با بررسی موقعیت ربات از ناحیه تکین با اطلاع شده و سپس کنترل ربات در همسایگی نقاط تکین از وضعیت کنترل دکارتی به وضعیت کنترل مفصلی منتقل می شود. نتایج نشان داد که این روش در مقایسه با روشهای ژاکوبین وارون، ترانهاده ژاکوبین، معکوس مجازی ژاکوبین و کنترل در فضای مفصلی مناسب تر است. نتایج این روش که سازگار با تکین است نشان داد که ولتاژ موتورها محدود و قابل قبول است. همچنین در نزدیکی نقاط تکین دارای خطای ردگیری نزدیک صفر است. همچنین می توان گفت که ردگیری مسیر تکین هنگام عبور از نقطه تکین مرکزی نتایج بهتری نسبت به هنگام عبور از نقطه تکین مرزی دارد.

همان گونه که مشاهده شد از آن جا که روش فضای مفصلی تنها با مختصات مفصلی کار می کند و مختصات دکارتی در آن دخالت ندارد در برابر تأثیرات نقاط تکین مقاوم است.

روش ژاکوبین وارون در فضای دکارتی هر چند دارای خطای ردگیری مناسبی است اما ولتاژ ورودی موتورها در نقاط تکین به طور نامعقولی زیاد می شود. بدیهی است موتورها نمی توانند چنین ولتاژی را داشته باشند و آن را تحمل کنند. مقدار این ولتاژها در بهترین حالت با استفاده از ماتریس معکوس مجازی ژاکوبین به جای ماتریس ژاکوبین ربات به $10^{25} v$ می رسد. با به کارگیری ماتریس ترانهاده ژاکوبین به جای ماتریس ژاکوبین ربات از آن جایی که معکوس ناپذیری ماتریس معکوس ژاکوبین مطرح نیست، ولتاژ وگشتاور موتورها محدود می شوند اما در عوض خطای بیشتری را داریم. به طوری که خطای سیستم در بهترین حالت در نقاط تکین به $3cm$ می رسد.

منابع:

- [1] Nenchev, N., Tsumaki, Y., Uchiyama, M., "Singularity consistent parametrization of robot motion and control," Department of Intelligent Machines and System Engineering, Hirosaki University, 2000.
- [2] Aboaf, E. W., Paul, R. P., "LIVING WITH THE SINGULARITY OF ROBOT WRISTS". Computer and Information Science Department, The University of Pennsylvania, IEEE1987 ,pp. 1713-1717
- [3] Oetomo, D., Marcelo, A., " Singularity Handling on Puma in Operational Space Formulation". National University of Singapore.
- [4] Lloyd, J. E., "Singularity-Robust Trajectory Generation", Computer Science Dept., University of British Columbia, IEEE International Conference on Robotics and Automation, September 2000, pp. 1-32.
- [5] Lloyd, J. E., "Singularity-Robust Trajectory Generation", Computer Science Dept., University of British Columbia, IEEE International Conference on Robotics and Automation, September 2000 , pp. 1-32.
- [6] Liu, G., Lou, Y., "Singularities of Parallel Manipulators: A Geometric Treatment". IEEE TRANSACTIONS ON ROBOTICS AND AUTOMATION, AUGUST 2003, pp. 579-593.
- [7] Simaan, M., Shoham, M., "Singularity Analysis of a Class of Composite Serial In-Parallel Robots". IEEE TRANSACTIONS ON ROBOTICS AND AUTOMATION, JUNE 2001, pp. 301-309.

- [8] Dafaoui, M., Pontnau, J., "Analysis and Design of a Six-DOF Parallel Manipulator, Modeling, Singular Configurations, and Workspace". IEEE TRANSACTIONS ON ROBOTICS AND AUTOMATION, FEBRUARY 1998, pp. 78-90.
- [9] Stephen, L., "Control of Redundant Manipulators for Task Compatibility" . Rockwell International Science Center Thousand Oaks, CA 91360. IEEE1987 , pp. 1718-1723.
- [10] J,M, Hollerbach. " Redundancy Resolution of Manipulators through Torque Optimization" . IEEE JOURNAL OF ROBOTICS AND AUTOMATION . AUGUST 1987, pp. 308-316.
- [11] Baillieul, J. " AVOIDING OBSTACLES AND RESOLVING KINEMATIC REDUNDANCY". Aerospace / Mechanical Engineering Boston University. IEEE1986, pp. 1698-1703.
- [12] Khatib, O. A., "Unified Approach for Motion and Force Control of Robot Manipulators : the Operational Space Formulation", IEEE Journal of Robotic and Automation, February 1987, pp. 43-52.
- [13] Chiaverini, S. Egeland, O. "A solution to the singularity problem for six-joint manipulators", Dipartimento di Meccanica e Sismotecnica Università degli Studi di Napoli., IEEE, 1990, pp. 644-649.
- [14] Samuel R. Buss., "Introduction to Inverse Kinematics with Jacobian Transpose, Pseudoinverse and Damped Least Squares methods". Department of Mathematics University of California, April 2004, pp. 1-19
- [15] Tesar, D., Kapoor, C., Fernandez, B "TASK-BASED DECISION MAKING AND CONTROL OF ROBOTIC MANIPULATORS", The University of Texas at Austin. December, 2004 ,pp. 33-36.
- [16] J. Y. S. Luh., Y. L. Gu. " INDUSTRIAL ROBOTS WITH SEVEN JOINTS". School of Electrical Engineering Purdue University West Lafayette, Indiana. IEEE 1985 , pp. 1010-1015.
- [17] Chan, T. F., Rajiv, V. "A Weighted Least-Norm Solution Based Scheme for Avoiding Joint Limits for Redundant Joint Manipulators" , IEEE TRANSACTIONS ON ROBOTICS AND AUTOMATION. April 1995, pp. 286-292.

- [18] "SPATIAL FREEDOM, SINGULARITY SURFACES AND WORKSPACES OF SERIAL KINEMATIC CHAINS". Doty, k.l., Schwartz, E.M., Department of Electrical Engineering University of Florida. , pp. 1-30.
- [19] Ki C. Suh., J.M, Hollerbach. "Local versus Global Torque Optimization of Redundant Manipulators". MIT Artificial Intelligence Laboratory Cambridge, IEEE 1987, pp. 619-624.
- [20] Manocha, D., Krishnan, S., "Solving Algebraic Systems using Matrix Computations". Department of Computer Science University of North Carolina.
- [21] Raju, G.V.S., Zhou, J., "Fuzzy rule based approach for robot motion control" IEEE 1992, pp. 1349-1355.
- [22] Alander, j.t., "On Findind the Optimal Genetic Algorithms for Robot Control Problems" IEEE/RSJ international workshop on intelligent robots and systems, 1991, pp. 1313-1317.
- [23] Wernholt, E., Ostring, M., "Modeling and Control of a Bending Backwards Industrial Robot". Division of Automatic Control Department of Electrical Engineering Linkopings university, May 2003.
- (٢٤) کتاب کنترول و دینامیک ربات ، اسپانگ و ویدیاساگار ، مترجم : جناب دکتر محمد مهدی فاتح
- (25) کتاب دینامیک و کنترول در رباتیک ، جان کریک ، مترجم : دکتر علی مقداری
- (26) کتاب کنترول غیر خطی ، اسلوتین و وایپینگ ، مترجم : محمد رضا هاشمی گلپایگانی
- [27] On the Calculation of Pseudo Inverses “. IEEE Transactions On Automatic Control . April, 1969 : pp. 204-205.
- [28] Schweinzer, H., "INTEGRATION OF A SENSOR IN A ROBOTER MOTION CONTROL WITH FAST REACTIONS PARALLEL PROCESSED IN REAL TIME" Technical University Vienna, IEEE 19 89 ,pp. 124-129

Abstract

In this thesis, a novel approach is presented for robot control. In the proposed approach, the control of robot is transformed from task space to joint space in the vicinity of singular points. The robot singularities are considered in the robot control approaches called the inverse Jacobian, the pseudo inverse Jacobian and transposed Jacobian for passing the singular points.

The robot control can be classified into joint space control and task space control. The advantage of control in the task space is sensing the end effector position which improves the performance. However, this kind of control is involved with the singular points. In contrast, control in the joint space, does not have the singularity problem and it is easy to implement. However, due to lack of sensing from the end effector position, the performance of control system may be degraded. It is required to plan a desired trajectory in the task space and joint space for controlling the position of the end effector. Considering the singularity is also important. Since, the control of robot is involved problems in the singular points. The control system may lose some degrees of freedom in the singular points. In this case, the control on the robot may be degraded in the task space and consequently the voltages of motors are increased. The motors and then the robot structure may be damaged. Therefore, the singular points are avoided to control robot in the task space. It is noted that the robot control in the joint space is not involve the singularity problem.

Keywords: manipulator Jacobian, singularity, task space control, pseudo inverse Jacobian



Shahrood University of Technology
Faculty of Electrical and Robotic Engineering

Master of Science Thesis
Electrical –Control Engineering

Consideration of Singular Points in Robot Control

Supervisor
Dr. Mohammad Mehdi Fateh

by
Mohammad Reza hosseini masoum

February 2008