Research Note

مهندسی مکانیک شریف، (بهار ۱۳۹۳)
دوره‌ی ۳-۳۰، شماره‌ی ۱، ص. ۱۴۳-۱۵۰، (یادداشت فنّی)

بزرگ‌ترین دایره‌ی عاری از تکینگی در فضای کاری ربات‌های موازی صفحه‌یی، با استفاده از تحلیل بازه‌یی و جابه‌جایی مرزها

محمدهادی فرزانه کلورزی * **(کارشناس ارشد)**
مهدی طالع ماسوله * **(استادیار)**
بهنام مشهدی غلامعلی (کارشناس ارشد)
دانشکده‌ی علوم و فنون نوین، دانشگاه تهران

در این مطالعه الگوریتمی قاعده‌مند، براساس مفهوم تحلیل بازه‌یی، ارائه می‌شود تا به‌وسیله‌ی آن بتوان بزرگ‌ترین دایره‌ی عاری از تکینگی[۱] در فضای کاری ربات‌های موازی صفحه‌یی را تعیین کرد. این الگوریتم برای بهینه‌سازی فضای کاری ربات‌های سه درجه آزادی کارآمد است و به‌عنوان مطالعه‌ی موردی سازوکار موازی $3\text{-}R\underline{P}R$ بررسی شده است. دو روش ارائه شده بدین منظور مبتنی بر تحلیل بازه‌یی و روش جابه‌جایی هندسی هستند. در روش تحلیل بازه‌یی، الگوریتم اصلی نهایتاً منتهی به یافتن بزرگ‌ترین دایره‌ی عاری از تکینگی در فضای کاری کل ربات می‌شود. علاوه بر آن، روش جابه‌جایی هندسی ارائه شده می‌تواند در نرم‌افزارهای CAD و نیز در ابزارهای محاسباتی پیاده‌سازی شود. سهم اعظم این مقاله به یافتن فضای کاری عاری از تکینگی با در نظر گرفتن فضای کاری ناشی از محدودیت‌های مکانیکی در عملگرها اختصاص دارد ــ مطلبی که این مطالعه را از سایر مطالعات مشابه متمایز می‌کند.

واژگان کلیدی: ربات‌های موازی صفحه‌یی، فضای کاری، تکینگی، تحلیل بازه‌یی، جابه‌جایی هندسی.

mhfarzane@ut.ac.ir
m.t.masouleh@ut.ac.ir
b.mashhadi@ut.ac.ir

۱. مقدمه

سازوکارهای موازی، که معمولاً به اشتباه نسل نوین ربات‌ها خوانده می‌شوند، پیشینه‌یی بسیار قدیمی‌تر از سازوکارهای متداول سری دارند.[۱] علاوه بر این، سازوکارهای موازی به‌دلیل برخورداری از خصوصیات ارزشمندی همچون تحمل بارهای بیشتر و دقت بالاتر، امروزه در پاره‌یی از کاربردهای صنعتی بسیار مورد توجه قرارگرفته‌اند. به‌عنوان مثال می‌توان به سازوکار گاف ـاستوارت[۲] در کاربردهای شبیه‌سازی پرواز[۲و۳] و ربات‌های دلتا[۴] در خط تولید کارخانه‌ها اشاره کرد.[۵و۶] در عین حال، مسائل مهمی مانند پیچیدگی‌های ریاضی در محاسبات سینماتیکی، وجود گستره‌یی از موقعیت‌های غیر قابل کنترل به‌نام تکینگی[۷] در فضای کاری[۳] محدود آنها، از جمله معضلات آنهاست که نافی رشد چشمگیر آنها در صنعت است. چالش‌های بیان شده غالباً در تحلیل سینماتیکی سازوکارهای موازی رخ می‌دهد و این مسائل توجه بسیاری از محققین را به مطالعه‌ی خواص سینماتیکی به‌منظور پیشنهاد روش‌های کارآمدتر جلب کرده است که نهایتاً منجر به بازبینی خصوصیات سینماتیکی و به دست آوردن طرح بهینه برای ساخت ربات می‌شود.[۸]

از آنجا که فضای کاری محدود سازوکارهای موازی با تکینگی همراه است، در مقام طراحی سازوکارهای موازی، تحلیل فضای کاری و تکینگی از اهمیت فوق‌العاده زیادی برخوردار است. براین اساس، تحلیل این فضا باید به‌گونه‌یی انجام شود که منجر به ساخت سازوکارهای موازی عاری از تکینگی شود.[۹] «تکینگی» باعث از دست رفتن استحکام ذاتی سازوکار و به دنبال آن، افزایش یا کاهش درجه‌ی آزادی[۴] مجری نهایی[۵] می‌شود.[۷و۱۰] از دیدگاه ریاضی، تکینگی با ریشه‌های دترمینان ماتریس ژاکوبین مرتبط است.[۷و۱۱] در واقع ماتریس ژاکوبین نگاشتی است میان سرعت مفصل‌ها و سرعت فضای دکارتی سازوکار، که برگرفته از خصوصیات سینماتیکی مرتبه اول هستند. در عمل، این کار با گرفتن مشتق زمانی از معادلات سینماتیک مستقیم (FKW)[۶] صورت می‌پذیرد. در ادبیات مربوطه، تکینگی سازوکارهای موازی از دیدگاه‌های مختلفی دسته‌بندی شده است.[۱۲] در این نوشتار از دیدگاه پیشنهادی توسط محققین،[۷] که به منطق این نوشتار نزدیک‌تر است، بهره برده‌ایم. تکینگی در سازوکارهای موازی به سه نوع تفکیک می‌شود: ۱. تکینگی سینماتیک معکوس[۷]؛ ۲. تکینگی سینماتیک مستقیم؛ ۳. ترکیبی از تکینگی سینماتیک معکوس و مستقیم. در مطالعات پیشینِ تکینگیِ نوع ۲، سازوکارهای موازی بیشترین مورد توجه قرارگرفته است، بدان سبب که حضورشان در میان فضای کاری و وابستگی‌شان به بعضی عبارات غیر خطی است و مطالعه درمورد آنها مستلزم به‌کارگیری شیوه‌های ظریف

* نویسنده مسئول

تاریخ: دریافت ۱۳۹۲/۶/۱۷، اصلاحیه ۱۳۹۲/۱۰/۴، پذیرش ۱۳۹۲/۱۰/۱۴.

ریاضی ــ نظیر نظریه‌ی پیچه[۸] و جبر گرسمان ـ‌کیلی[۹] ــ است.[۶و۱۳]

در حالت کلی درمورد فضای کاری سازوکارهای موازی، معضل اصلی غیر ممکن بودن ارائه‌ی فضای کاری آنها در محیط شش‌بعدی یا ارائه‌ی دقیق فضای سه‌بعدی به‌همراه حرکات دورانی است.[۱۴] برای رفع این مشکل، نمایش برشی از دوران ثابت[۱۰] یا جابه‌جایی ثابت[۱۱] پیشنهاد می‌شود. تمرکز این نوشتار در نمایش برشی (مقطعی) از دوران است که به آن «فضای کاری دوران ثابت» اطلاق می‌شود. فضای کاری دوران ثابت شامل دسته‌یی از موقعیت‌های دکارتی در دسترس مجری نهایی برای آن دوران خاص است.[۱۴]

مطالعات بسیار فراوانی درخصوص فضای کاری عاری از تکینگی سازوکارهای موازی صورت پذیرفته است که بیشترشان مبتنی بر روش‌های محاسبات عددی پیچیده‌اند و با محدودیت‌هایی نیز مواجه‌اند. مطالعه‌ی فضای کاری عاری از تکینگی سازوکارها ــ به‌ویژه قبل از شروع به ساخت‌شان ــ از اهمیت به‌سزایی برخوردار است و این امر با عنایت به تعداد مقالات چاپ‌شده در این زمینه کاملاً مشهود است. در زمینه‌ی مکان تکینگی ربات‌های موازی سه درجه آزادی صفحه‌یی، مطالعات گسترده‌یی با استفاده از نظریه‌ی پیچه انجام شده است.[۱۳] همچنین روشی براساس ویژگی‌های هندسی یک ربات موازی ارائه شده[۱۵] که برای یک ربات سه‌پایه‌ی موازی فضای کاری عاری از تکینگی به دست آمده است. در مطالعه‌یی دیگر، محققین با استفاده از این حقیقت که می‌توان مسئله‌ی دایره‌ی عاری از تکینگی در ربات‌های سه و شش درجه آزادی را به‌لحاظ ریاضی یک مسئله‌ی بهینه‌سازی با قید ضریب لاگرانژ در نظر گرفت،[۱۶و۱۷] به حل این مسئله پراخته‌اند. در این مقالات مرکز دایره را نقطه‌یی ثابت و معلوم در نظر گرفته‌اند و فقط شعاع دایره مورد مطالعه قرار می‌گیرد. برخی از محققین در مطالعات خود دستورالعمل‌هایی عددی برای یافتن فضای کاری عاری از تکینگی ربات‌های سه و شش درجه آزادی پیشنهاد کرده‌اند[۱۸-۲۰] و اخیراً نیز با استفاده از بهینه‌سازی گروهی عملی[۱۲] دایره‌ی عاری از تکینگی یک ربات سه درجه آزادی به دست آمده است.[۲۱] علاوه بر این، نزدیکی به تکینگی با فرمول‌بندی مسئله به شکل بهینه‌سازی مقید[۱۳] و ضریب لاگرانژ مطرح شده است.[۲۲]

در این نوشتار هدف ارائه‌ی روشی منسجم بر پایه‌ی تحلیل بازه‌یی[۱۴] [۲۳] است که به‌وسیله‌ی آن بتوان بزرگ‌ترین دایره‌ی عاری از تکینگی سازوکار موازی سه درجه آزادی صفحه‌یی را به دست آورد. این در حالی است که مرکز دایره نامشخص است و به‌صورت هوشمندانه در روال الگوریتم به دست خواهد آمد. سپس با بیان مفهوم هندسی یافتن مرکز دایره‌ی محاطی، نتایج این دو روش با یکدیگر مقایسه می‌شود. در نظر گرفتن مرزهای فضای کاری، وجه تمایز این تحلیل با دیگر مطالعات انجام شده در این خصوص[۹،۱۵-۱۷و۲۱] است.

استفاده از تحلیل بازه‌یی به‌عنوان روش عددی جایگزینِ روش‌های مبتنی بر الگوریتم‌های تکاملی[۱۵] ــ مانند الگوریتم ژنتیکی[۱۶] ــ به‌منظور به دست آوردن نتایج مطمئن عملی، که کاربردهای فراوانی در یافتن فضای کاری ربات‌های موازی و دیگر خصوصیات سینماتیکی[۲۴] آنها دارد، فواید بسیار زیادی دارد. این فواید به‌اختصار عبارت‌اند از:[۲۵] ۱. در قیاس با بسیاری از ابزارهای ریاضی هوشمند که ممکن است منجر به پردازش‌های زمان‌بر و همگرایی به یک بهینه‌ی محلی شود، تحلیل بازه‌یی همانند یک جعبه‌ی سیاه نیست، چون ممکن است نیاز به ترکیب کردن خلاقیت و مفاهیم عددی باشد تا بتوان آن را کارآمدتر کرد؛ ۲. تحلیل بازه‌یی امکان یافتن تمام جواب‌های مربوط به یک سیستم نامعادله را در یک فضای جست‌وجو فراهم می‌کند؛[۲۶و۲۷] ۳. برای مسائل دو و سه‌بعدی منتج به پدید آمدن نمایشی از سیر تکاملی پاسخ‌ها و به تصویر کشیدن روند حل به‌منظور داشتن آگاهی بهتر از مسئله می‌شود؛ ۴. اجازه می‌دهد موارد غیر قطعی در مدل ربات، همانند عدم قطعیت در اندازه‌ی طول بازوها و دقت عملگرها، در نظر گرفته شود. شایان ذکر است کدهای نوشته شده برای تحلیل بازه‌یی در مطالعه‌ی حاضر تحت کتابخانه‌ی اینتلب[۱۷] در نرم‌افزار متلب[۱۸] حاصل شده است.

در ادامه‌ی این نوشتار، ابتدا مدل سینماتیکی سازوکار تحت بررسی، به‌طور مشخص ۳-R$\underline{\text{P}}$R، به‌تفصیل بیان شده است. سپس مفهوم کلی تحلیل بازه‌یی در الگوریتم پیشنهادی به‌طورکامل توضیح داده شده (بخش ۳) و چهار شبه‌کد با هدف فهم بهتر مسئله تهیه شده است. سپس ضمن ارائه‌ی نتایج به دست آمده از روش تحلیل بازه‌یی برای رباتِ تحت مطالعه، به بیان مفاهیم هندسی و روش جابه‌جایی هندسی[۱۹] پرداخته خواهد شد و نتایج حاصل از دو روش با یکدیگر مقایسه می‌شود.

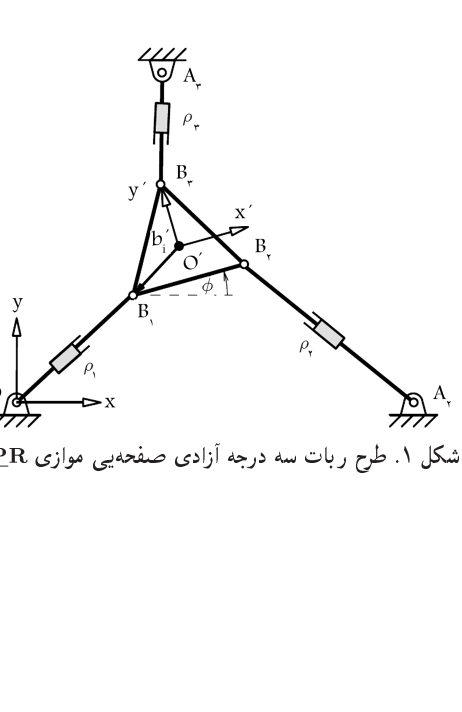
۲. توصیف معماری و مروری بر سینماتیک مرتبه اول ربات موازی ۳-R$\underline{\text{P}}$R

در شکل ۱ شمایی از سازوکار موازی ۳-R$\underline{\text{P}}$R نشان داده شده است. در این نوشتار R و $\underline{\text{P}}$ به‌ترتیب نمایانگر مفصل لولایی[۲۰] و عملگر کشویی[۲۱] هستند. چنان که مشاهده می‌شود، سازوکار ۳-R$\underline{\text{P}}$R صفحه‌یی دارای یک مثلث ثابت $\Delta_{A_1A_2A_3}$ در بستر است که به‌وسیله‌ی سه عملگر کشویی با طول متغیر ρ_i، $i = 1, 2, 3$، به مجری نهایی آن که یک مثلث متحرک ($\Delta_{B_1B_2B_3}$) است متصل می‌شود. مفاصل لولایی فعال[۲۲]اند و درنقاط A_i و B_i واقع شده‌اند. حل معادلات سینماتیک معکوس به‌سهولت انجام می‌پذیرد و شامل یافتن فاصله‌ی میان نقاط A_i و B_i است:

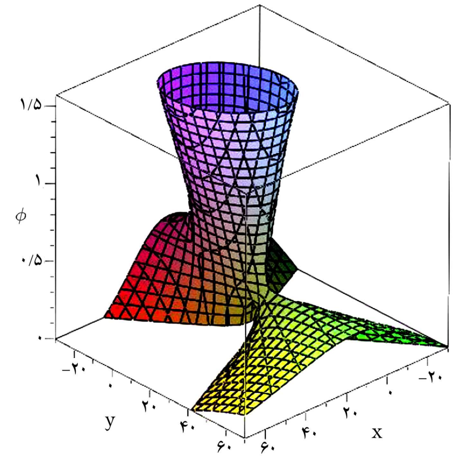
$$(x_{Bi} - x_{Ai})^2 + (y_{Bi} - y_{Ai})^2 = \rho_i^2, \qquad i = 1, 2, 3 \tag{1}$$

از رابطه‌ی ۱ می‌توان به‌عنوان معادله‌ی اصلی برای حل سینماتیک مستقیم و فضای کاری ۳-R$\underline{\text{P}}$R که فراتر از بحث این مقاله است، استفاده کرد.[۱۰،۲۸و۲۹] از آنجا که موضوع اصلی این مقاله در ارتباط با تکینگی است، نگاهی دقیق‌تر به ماتریس ژاکوبین خالی از لطف نیست. صرف نظر از جزئیات ریاضی، روابط سینماتیک مرتبه اول، حاصل مشتق‌گیری از معادلات سینماتیک مستقیم نسبت به زمان است. ماتریس ژاکوبین برای سازوکار تحت بررسی عبارت خواهد بود از:

$$\mathbf{J} = \begin{bmatrix} n_{1x} & n_{1y} & (\mathbf{b}_1 \times \mathbf{n}_1) \times \mathbf{k} \\ n_{2x} & n_{2y} & (\mathbf{b}_2 \times \mathbf{n}_2) \times \mathbf{k} \\ n_{3x} & n_{3y} & (\mathbf{b}_3 \times \mathbf{n}_3) \times \mathbf{k} \end{bmatrix} \tag{2}$$



شکل ۱. طرح ربات سه درجه آزادی صفحه‌یی موازی ۳-R$\underline{\text{P}}$R.[۸]



شکل ۲. رویه‌ی تکینگی ربات ۳-R$\underline{\text{P}}$R که معادله درجه ۲ است.

که در آن $\mathbf{b}_i$، $i = ۱, ۲, ۳$، بردار موقعیت نقطه‌ی B_i در مختصات ثابت، و بردار یکه در راستای i امین عملگر کشویی به شکل $\mathbf{n_i} = [n_{ix} \quad n_{iy} \quad ۰]^T$ است. شایان ذکر است که در این مقاله بالانویس $'(\cdot)$ در نمایش یک بردار نشان‌گر ارائه‌ی آن در مختصات متحرک $O'_{x'y'}$ است. تکینگی نوع دو[۷] در حالت‌هایی از مجری نهایی رخ می‌دهد که دترمینان ماتریس ژاکوبین صفر می‌شود:

$$D(x, y, \varphi) = \det(\mathbf{J}) = ۰ \qquad (۳)$$

که در آن، x و y نشان‌دهنده‌ی مکان مجری نهایی در صفحه‌ی دکارتی و φ جهت‌گیری (دوران) آن حول محور عمود به صفحه است. در شکل ۲ نمایش سه‌بعدی رویه‌ی مکان تکینگی ربات ۳-RPR نشان داده شده سات. منحنی تکینگی ۳-RPR به سه شکل کلی برای دوران‌های مختلف مجری نهایی ظاهر می‌شود:[۲۰و۲۹] هذلولی‌گون، سهمی‌گون و بیضی‌گون (که تحت شرایط خاص به شکل دایره هم ظاهر شود)؛ این امر به‌وضوح در شکل ۲ نمایان است.

۳. تحلیل بازه‌یی و ناحیه‌ی عاری از تکینگی

۱.۳. تحلیل بازه‌یی و چهارچوب ریاضی

پس از معرفی تحلیل بازه‌یی توسط محققین،[۲۲] با توجه به مقالات متعدد در این زمینه، این روش امروزه به ابزار عددی قدرتمندی در حل گستره‌ی وسیعی از مسائل بدل شده است. اخیراً با آشکار شدن خصوصیات چشم‌گیر تحلیل بازه‌یی ــ مثل یافتن پاسخ‌های مسئله در بازه‌ی مورد نظر ــ و نیز با توجه به خطاهای محاسبات رایانه‌یی، گردکردن موجب جذب علاقه‌ی بسیاری از محققین در جامعه‌ی رباتیک برای به‌کارگیری از این ابزار در حل مسائل پیچیده‌یی مانند سینماتیک مستقیم و معکوس، کالیبراسیون و فضای کاری عاری از تکینگی شده است. بازه‌ی متغیر x با عنوان X معرفی می‌شود که محدود به دو حد بالا و پایین است $\mathrm{X} = [\bar{\mathrm{x}}, \underline{\mathrm{x}}]$ و در آن $\bar{\mathrm{x}} \leq \mathrm{x} \leq \underline{\mathrm{x}}$. یک تابع را زمانی بازه‌یی قلمداد می‌کنند که بعضی از متغیرها و پارامترهای آن بازه‌یی باشد.

مفهوم کلی تحلیل بازه‌یی مبتنی است بر نصف‌کردن یک مربع در فضای دوبعدی (یا یک فرامربع در فضاهای بالاتر از دوبعدی) است، به‌شکلی که با اعمال شروط ریاضی مناسب مربع‌های نصف‌شده به جواب سوق پیدا کنند. در واقع، تأکید بیشتری خواهیم داشت بر روش پیدا کردن مربع اول ــ که به‌صورت اختیاری توسط کاربر انتخاب می‌شود ــ و سپس پیدا کردن مربع بعدی به‌شکلی که به جواب سوق پیدا کند. چنان که در بخش بعدی نیز توضیح داده خواهد شد، با تلفیق بعضی از مفاهیم سنتی حل عددی و تحلیل بازه‌یی، موانع موجود در راه فضای کاری عاری از تکینگی برطرف خواهد شد.

۲.۳. به دست آوردن فضای کاری، فضای کاری عاری از تکینگی، بزرگ‌ترین دایره برای مربع اولیه

در این بخش، چهار الگوریتم با عناوین B ،C ،D و A ارائه شده‌اند که تمامی آنها برای دوران داده‌شده‌ی مجری نهایی برای مربع اولیه‌ی تعیین‌شده توسط کاربر، به‌نام مربع جاری (Bc) است. در الگوریتم C الگوریتم‌های ذکر شده، مربع جاری تکامل نخواهد یافت، هیچ مربع جدیدی از این مربع ایجاد نخواهد شد، و تمامی نتایج تنها در حوزه‌ی Bc معتبر است. پس، تضمینی وجود ندارد که نتایج به دست آمده از الگوریتم آخر (C) برای کل فضای کاری اعتبار داشته باشد. لذا برای غلبه بر این مشکل، الگوریتم C بهبود بخشیده شده و با نام الگوریتم D در بخش بعد معرفی می‌شود؛ در این راستا با اعمال پاره‌یی از قوانین مربع‌های جدیدی ایجاد می‌شود که منجر به یافتن مرکز دایره‌ی بهینه‌ی معتبر برای تمام فضای کاری خواهد شد. می‌توان الگوریتم C را تا حدی معادل تحلیل بازه‌یی مطالعات پیشین[۱۶] در نظر گرفت، که در آنها مرکز دایره (در اینجا مربع اولیه) مشخص است و الگوریتم D به‌منظور رفع محدودیت‌های روش‌های گذشته اعمال می‌شود. در این نوشتار به‌منظور شفافیت بیشتر، فهرست با $\mathbf{L}$ و مربع با B نشان داده می‌شود.

۱.۲.۳. به دست آوردن فضای کاری دوران ثابت از طریق تحلیل بازه‌یی (الگوریتم A)

فضای کاری دوران ثابت یک سازوکار، چنان که قبلا بیان شد، دسته‌یی از نقاط فضای دکارتی است که به ازای تمامی آنها برای یک دوران خاص، مجری نهایی توانایی دسترسی به آن نقاط را دارد.

روش‌های بسیاری در این رابطه در ادبیات مربوطه بیان شده است[۱۰و۳۰] که از الگوریتم‌های نقطه‌گذاری تا روش‌های هندسی را شامل می‌شود (مانند استفاده از نرم‌افزارهای CAD).[۱۴] شایان ذکر است روش‌های متعددی بر پایه‌ی تحلیل بازه‌یی برای نشان دادن فضای کاری دوران ثابت پیشنهاد شده است. در این نوشتار نیز، الگوریتمی در شکل ۳ به‌نام الگوریتم A پیشنهاد می‌شود برای مسئله‌ی یافتن فضای کاری دوران ثابت شامل حل سینماتیک معکوس (معادله‌ی ۱) برای $\rho_{\min}$ و $\rho_{\max}$ که نشان‌گر محدودیت‌های مکانیکی عملگر طی یک دوران خاص است. به‌همین دلیل مسئله به حل شش نامعادله‌ی برای $i = ۱, ۲, ۳$ بدل خواهد شد که منتج به یافتن دسته نقاط ممکن خواهد شد:

$$\begin{cases} (x_{Ai} - x_{Bi})^۲ + (y_{Ai} - y_{Bi})^۲ \leq \rho^۲_{\max} \\ (x_{Ai} - x_{Bi})^۲ + (y_{Ai} - y_{Bi})^۲ \geq \rho^۲_{\min} \end{cases} \qquad (۴)$$

در شکل ۳ الگوریتم A، که به حل سیستم نامعادلات بالا به‌منظور یافتن فضای کاری دوران ثابت می‌پردازد، ارائه شده است. یادآور می‌شود که الگوریتم A منجر به ایجاد سه دسته جواب می‌شود که هرکدام از این دسته‌ها شامل لیستی از مربع‌های مختلف است: ۱. نواحی داخلی فضای کاری ($\mathrm{W_{in}}$)؛ ۲. نواحی خارج از فضای کاری ($\mathrm{W_{out}}$)؛ ۳. مرزهای فضای کاری $\mathrm{W_b}$. اگر مربع $\mathrm{B_c}$ نامعادلات ۴ را برای همه‌ی پایه‌ها به‌طور هم‌زمان ارضا کند، پس $\mathrm{B_c}$ داخل فضای کاری است و به لیست $\mathrm{W_{in}}$ اضافه می‌شود؛ اما اگر تمامی نامعادلات را برای همه‌ی پایه‌ها ارضا نکند پس

- **Initialize** empty lists $\mathcal{W}_{in}$, $\mathcal{W}_{out}$ and $\mathcal{W}_b$
- **Initialize** list $\mathcal{W}$ containing initial task space intervals (boxes) to be analyzed
- **Initialize** B_C as the current box
- **While** ($\mathcal{W}$ not empty)
  - **Extract** B_C from first column of matrix $\mathcal{W}$
  - **Empty** first column of $\mathcal{W}$
  - **Subs** B_C into Eq. (4)
  - **Case** result is inner, outer or boundary box
    * **Case 1:** (The solution is inside the workspace) add B_C to list $\mathcal{W}_{in}$
    * **Case 2:** (The solution is outside the workspace) add B_C to list $\mathcal{W}_{out}$
    * **Case 3:** (The solution lies within boundary)
      If (dimension of $B_C > \epsilon$)
        Bisect B_C by the biggest edge
        Add two new boxes into the list $\mathcal{W}$
      Else If (threshold dimension ϵ has been reached)
        Add B_C into list $\mathcal{W}_b$
      End If
  End Case
- **End While**
- **Return** $\mathcal{W}_b$ as the workspace boundaries

شکل ۳. شبه‌کد الگوریتم **A**، برای یافتن فضای کاری دوران ثابت به فرم بازه‌یی.

کاملاً خارج از فضای کاری است و به لیست (W_{out}) اضافه می‌شود؛ و اگر در بعضی از نامعادلات صدق کند و در برخی صدق نکند آنگاه از طول بزرگ‌تر به دو مربع تقسیم می‌شود و به انتهای لیست W_b اضافه می‌شود. سپس الگوریتم این مراحل را در مربع دیگری از W_b تکرار می‌کند. چنان که در شکل ۳ مشاهده می‌شود، الگوریتم تا جایی ادامه می‌یابد که طول مربع تحت بررسی B_C از مقدار ε تعیین‌شده بیشتر باشد؛ ε نشان‌گر دقت نهایی الگوریتم است.

۲.۲.۳. به دست آوردن نقاط تکینگی برای دوران ثابت از طریق تحلیل بازه‌یی (الگوریتم B)

مفهوم به‌کار رفته در الگوریتم B برای یافتن نواحی داخلی (S_{in})، نواحی خارجی (S_{out}) و مرزهای تکینگی (S_b) در محتوا کاملاً شبیه است به آنچه در الگوریتم A استفاده شده است. در این مورد، معادله‌ی ۳ ارائه‌دهنده‌ی مکان تکینگی است و باید به‌عنوان معادله‌ی اصلی برای یافتن نواحی داخل، خارج و مرز مورد استفاده قرار گیرد.

شکل ۴ شبه کد الگوریتم B را نشان می‌دهد که مانند الگوریتم A عمل می‌کند، با این تفاوت که به‌جای درنظر گرفتن ۶ نامعادله برای کل سازوکار، از یک معادله (معادله‌ی ۳) استفاده خواهد شد. اگر مربع تحت بررسی (B_C) تماماً بزرگ‌تر از صفر باشد آنگاه به لیست نواحی داخل (S_{in}) اضافه خواهد شد. اگر B_C تماماً کوچک‌تر از صفر باشد، به لیست S_{out} افزوده خواهد شد. و در نهایت، اگر B_C صفر را شامل شود، از ضلع بزرگ‌تر به دو بخش تقسیم خواهد شد و به انتهای لیست S_b اضافه می‌شود. مانند قبل، این الگوریتم هم تا جایی ادامه پیدا می‌کند که دقت لازم (ε) به دست بیاید.

۳.۲.۳. بزرگ‌ترین دایره‌ی عاری از تکینگی دوران ثابت برای یک مربع اولیه‌ی معلوم (الگوریتم C)

با استفاده از نتایج الگوریتم A و B، که به‌ترتیب مرزهای فضای کاری و تکینگی به شکل بازه‌یی هستند، می‌توان به‌کمک الگوریتم C (ارائه شده در شکل ۵)، مرکز دایره و شعاع بزرگ‌ترین دایره‌ی عاری از تکینگی را برای یک مربع به‌نام مربع اولیه (B_i) محاسبه کرد. چنان که پیش‌تر گفته شد، الگوریتم C فقط روی همان مربع اولیه‌یی که کاربر تعریف کرده محاسبات را انجام می‌دهد، و از آنجا که این مربع به‌صورت اختیاری انتخاب شده است جواب به دست آمده تنها در همین مربع معتبر

- **Initialize** empty lists $\mathcal{S}_{in}$, $\mathcal{S}_{out}$ and $\mathcal{S}_b$
- **Initialize** list $\mathcal{S}$ containing initial task space intervals (boxes) to be analyzed
- **Initialize** B_C as the current box
- **While** ($\mathcal{S}$ not empty)
  - **Extract** B_C from first column of matrix $\mathcal{S}$
  - **Empty** first column of $\mathcal{S}$
  - **Subs** B_C into Eq. (3)
  - **Case** result is inner, outer or boundary box
    * **Case 1:** (The solution is inside the boundaries) add B_C to list $\mathcal{S}_{in}$
    * **Case 2:** (The solution is outside the boundaries) add B_C to list $\mathcal{S}_{out}$
    * **Case 3:** (The solution lies within boundary)
      If (dimension of $B_C > \epsilon$)
        Bisect B_C by the biggest edge
        Add two new boxes into the list $\mathcal{S}$
      Else If (threshold dimension ϵ has been reached)
        Add B_C into list $\mathcal{S}_b$
      End If
  End Case
- **End While**
- **Return** $\mathcal{S}_b$ as the singularity boundaries

شکل ۴. شبه‌کد الگوریتم **B**، برای یافتن مرزهای تکینگی دوران ثابت به شکل بازه‌یی.

- **Initialize** list $\mathcal{B}$ containing boundaries such as workspace borders ($\mathcal{W}_b$ taken from Alg. A, Table (I)) and singularity borders ($\mathcal{S}_b$ taken from Alg. B, Table (II))
- **Initialize** box B_i as initial guess, defined by the user
- **Add** B_i into B_C
- **While** (dimension $B_C > \epsilon$)
  - **Bisect** B_C by the biggest edge into B_{C1} and B_{C2}
  - **Calculate** distance from B_{C1} and then B_{C2} to $\mathcal{B}$, with respect to Eq. (distance) results $\mathbf{D_{C1}}$ and $\mathbf{D_{C2}}$
  - **If** ($\min(\mathbf{D_{C1}}) > \min(\mathbf{D_{C2}})$)
      Subs B_{C1} into the box B_C
      $\mathbf{D} \leftarrow \mathbf{D_{C1}}$
    Else
      Subs B_{C2} into the box B_C
      $\mathbf{D} \leftarrow \mathbf{D_{C2}}$
    End If

  End While
- **Return** $C_0 \leftarrow$ **Center**(B_C), as MSFC center
- **Return** $R_0 \leftarrow$ **min**(D), as MSFC radius

شکل ۵. شبه‌کد الگوریتم **C**، برای یافتن بزرگ‌ترین دایره‌ی عاری از تکینگی در فضای‌کاری دوران ثابت، برای یک مربع اولیه‌ی معلوم.

است. شایان ذکر است که ورودی الگوریتم C همان بازه‌های به دست آمده توسط الگوریتم‌های A و B برای مرزهای فضای کاری و نقاط تکینه است. روند الگوریتم چنین است: ابتدا مربع اولیه‌ی B_i به‌عنوان مربع تحت بررسی قلمداد می‌شود؛ سپس B_C به دو زیر مربع تقسیم می‌شود؛ آنگاه فاصله‌ی هریک از B_{C1} و B_{C2} تا همه‌ی مربع‌های فهرست B محاسبه می‌شود ــ B شامل مجموعه‌ی مرزهای فضای کاری (W_b)، و تکینگی (S_b) است. نتیجه‌ی محاسبات قسمت قبل برای هر زیرمربع یک لیست از فواصل به فرم بازه‌یی است، D_{C1} و D_{C2}.

در این مرحله زیرمربعی که حد پائین بزرگ‌تری دارد، به دیگر سخن، کمینه بزرگ‌تری دارد و نسبت به زیرمربع دیگر فاصله‌ی بیشتری تا مرزها دارد، نگاه داشته می‌شود و به‌عنوان B_C برای مرحله‌ی بعدی به‌کار برده می‌شود، زیرمربع دیگر هم در ادامه‌ی الگوریتم تأثیری ندارد و نادیده گرفته می‌شود. این روند نصف‌کردن تا جایی ادامه پیدا می‌کند که دقت مورد نظر (ε) ارضا شود. زمانی که B_C به‌اندازه‌ی کافی کوچک باشد، با دقت خوبی مرکز B_C به‌عنوان مرکز دایره و حد پائین فهرست جواب (D) شعاع بزرگ‌ترین دایره‌ی عاری از تکینگی است. در الگوریتم C اگر مربع اولیه‌ی

- For (all orientations)
- Run algorithm C (Table III) Consider initial guess B_i and point C_0
- $i \leftarrow 0$
- While (C_i is close to the edges of B_i)
  - Create box B_i with C_0 as center point and dimension(B_0)
  - Run algorithm C (Table III)
  - $i \leftarrow i+1$

  End While
- Return C_0 as MSFC center
- Return R_0 as MSFC radius

  End For
- Return profile of optimum circles

شکل ۶. شبه کد الگوریتم D، برای یافتن بزرگ‌ترین دایره‌ی عاری از تکینگی در فضای‌کاری، برای تمام دوران‌ها.

انتخاب شده توسط کاربر شامل مرکز بهینه‌ی کلی نباشد، آنگاه زمانی که فرایند به ε مورد نظر برسد، نقطه‌ی مرکز به لبه‌های B_i نزدیک خواهد بود. به‌عبارت دقیق‌تر، فاصله‌ی نقطه‌ی مرکز آخرین ε به لبه‌های B_i کم‌تر از ε است و این نشانگر آن است که نقطه‌ی به دست آمده متمایل به پیشروی در فضای کاری است. در بخش بعدی این الگوریتم گسترش داده خواهد شد، با این هدف که با بروز این حالت فرایند به‌صورت هوشمند ادامه یابد و مرکز دایره و شعاع دایره عاری از تکینگی را برای کل فضای کاری تضمین کند ــ خواه B_i شامل این مرکز باشد خواه نباشد.

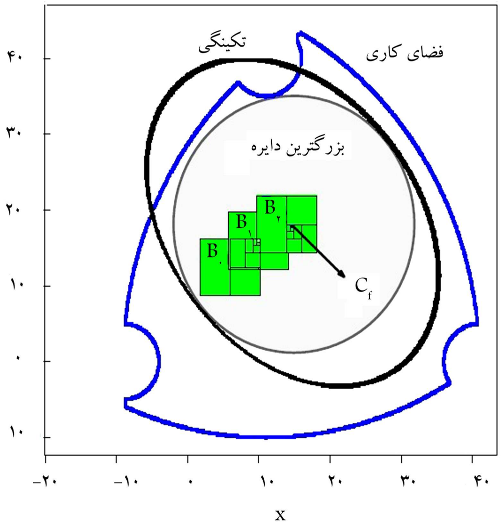
۴.۲.۳. یافتن دایره‌ی عاری از تکینگی تضمین شده برای تمام فضای کاری و برای همه‌ی دوارن‌ها (الگوریتم D)

در اینجا، برای تعمیم الگوریتم C و کارهای انجام شده‌ی قبلی،[۱۶] و به‌منظور یافتن مرکز دایره‌ی عاری از تکینگی تضمین شده برای تمام فضاه کاری، خصوصیات جدیدی به الگوریتم C اضافه خواهد شد. این الگوریتم با عنوان D (شکل ۶) قابلیت این را دارد که: ۱. مربع‌های دیگری با درنظرگرفتن B_i ایجاد کند تا بزرگ‌ترین دایره‌ی عاری از تکینگی را برای تمام فضای کاری پیدا کند؛ ۲. بزرگ‌ترین دایره را برای همه‌ی دوران‌های مجری نهایی پیدا کند.

B_i را به‌عنوان مربع اولیه‌یی در نظر بگیرید که الگوریتم C بر آن اعمال شده است. حال اگر مرکز دایره‌ی به دست آمده از اعمال الگوریتم C بر B_i به لبه‌های B_i نزدیک، الگوریتم D مربع جدیدی با همان ابعاد B_i و حول مرکز دایره‌ی به دست آمده از C تشکیل می‌دهد، به‌نام B_1 و سپس الگوریتم C بر B_1 اعمال می‌شود. الگوریتم D تا جایی به ساختن مربع‌های جدید می‌پردازد که مرکز به دست آمده از لبه‌ی مربع B_C فاصله‌ی مطمئن را داشته باشد. معیار نزدیکی همان ε است. تمامی روش‌های بیان شده تا این زمان برای دوران ثابت است و برای دست‌یابی به بزرگ‌ترین دایره‌ی عاری از تکینگی برای همه‌ی دوران‌ها کافی است الگوریتم D برای گستره‌ی دوران‌های مورد نظر اعمال شود.

۴. نتایج

به‌منظور داشتن دید بهتر نسبت به الگوریتم‌های بیان‌شده تاکنون، در این بخش به بررسی جواب‌های حاصل از الگوریتم‌های A، B، C و D برای ربات موازی ۳-RPR با مشخصات ساختاری بیان شده در جدول ۱ خواهیم پرداخت. در



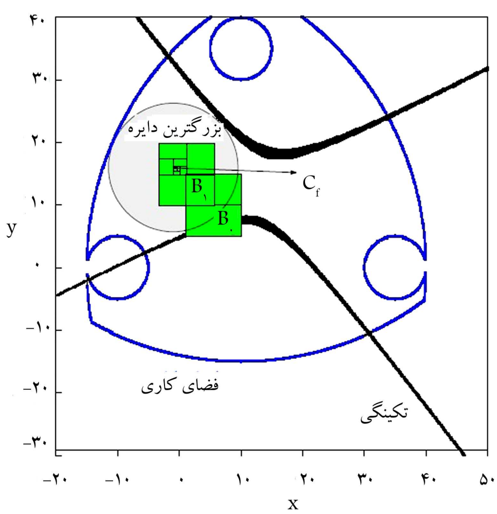
شکل ۷. نتایج به دست آمده برای فضای‌کاری، تکینگی و بزرگ‌ترین دایره‌ی عاری از تکینگی در دوران $\varphi = \frac{\pi}{36}$. دو مربع جدید به‌نام‌های B_1 و B_2 ایجاد شده، و به‌دلیل نزدیک نبودن C_0 به مرزهای B_2 نیاز به ایجاد مربع دیگری نیست.

جدول ۱. ابعاد ساختاری ربات تحت بررسی **۳-RPR**، همه‌ی اندازه‌ها به دسی‌متر است.

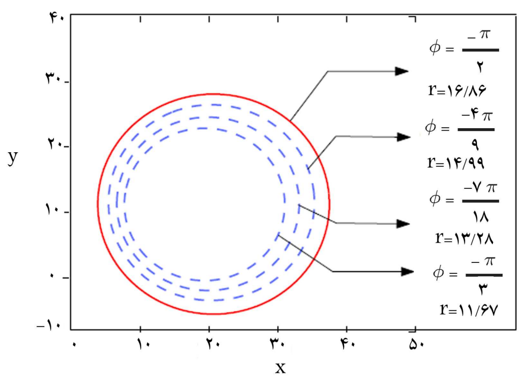
i	B_{ix}	B_{iy}	ρ_{min}	ρ_{max}	P'_{ix}	P'_{iy}
۱	۳٫۷۸	۴٫۳۴	۰٫۵	۵	−۴٫۸۳	−۳٫۱۹
۲	۳۴٫۴۷	−۳٫۷۸	۰٫۵	۵	۱۲٫۰۴	−۳٫۱۹
۳	۱۶٫۳۲	۳۴٫۷۶	۰٫۵	۵	۸٫۲۳	۱۲٫۰۹

شکل ۷ نتایج به دست آمده برای فضای کاری، تکینگی و بزرگ‌ترین دایره‌ی عاری از تکینگی در دوران $\varphi = \frac{\pi}{36}$ نمایش داده شده است. چنان که مشاهده می‌شود دو مربع جدید به نام‌های B_1 و B_2 ایجاد شده است و به‌دلیل این که C_0 به مرزهای B_2 نزدیک نیست، نیاز به ایجاد مربع دیگری نیست. فضای کاری دوران ثابت به دست آمده از الگوریتم A و تکینگی ــ در این حالت هذلولی‌گون ــ به دست آمده از الگوریتم B، برای دوران $\varphi = 0$ را به‌صورت همزمان نشان می‌دهد. سپس به‌منظور یافتن بزرگ‌ترین دایره‌ی عاری از تکینگی، مربع اولیه‌ی B_i یا B_0 انتخاب، و الگوریتم D بر روی آن پیاده شده است. یادآور می‌شود که الگوریتم C در بطن الگوریتم D وجود دارد و برای هر مربع تولید شده توسط الگوریتم D اجرا خواهد شد. پس برای B_i یا B_0 هم الگوریتم C مانند شکل ۵ اجرا خواهد شد و در ادامه الگوریتم D اجرا می‌شود. چنان که در شکل ۷ مشاهده می‌شود، الگوریتم D مربع جدیدی ایجاد کرده است، زیرا پیاده‌سازی الگوریتم C بر B_i یا B_0 مرکز دایره‌یی را نتیجه می‌دهد که نزدیک به مرزهای B_0 قرار دارد. در نهایت با پیاده‌سازی الگوریتم C روی B_1 مرکز دایره‌ی منتجه نزدیک به مرزهای B_1 نیست؛ لذا این نقطه مرکز دایره‌ی بهینه در فضای کاری و عاری از تکینگی است و شعاع دایره هم کم‌ترین فاصله‌ی این نقطه تا مرزهاست. شایان ذکر است نتایج این بخش از کدهای نوشته شده تحت کتابخانه‌ی اینتلب در نرم‌افزار متلب حاصله است.

در شکل ۸ حالتی از مجری نهایی مد نظر قرارگرفته که در آن دوران $\varphi = \frac{\pi}{36}$ و نمودار تکینگی به شکل بیضی است. به‌سادگی مشاهده می‌شود که برای مربع اولیه‌ی B_i در نظرگرفته شده در این مورد، الگوریتم D بعد از ساختن سه مربع جدید به جواب نهایی رسیده است که در آن B_2 شامل نقطه‌ی C_0 است که مرکز بزرگ‌ترین



شکل ۸. نتایج به دست آمده برای فضای کاری، تکینگی و بزرگ‌ترین دایره‌ی عاری از تکینگی در دوران $\varphi = ۰$. یک مربع جدید به‌نام $B_۱$ ایجاد شده است و به‌دلیل نزدیک‌نبودن $C_۰$ به مرزهای $B_۱$ نیاز به ایجاد مربع دیگری نیست.



شکل ۹. بزرگ‌ترین دایره‌ی عاری از تکینگی برای $\varphi = [-\frac{\pi}{۲}, -\frac{\pi}{۳}]$ در ربات $R\underline{P}R$-۳ با در نظر گرفتن ابعاد ارائه شده در جدول ۱.

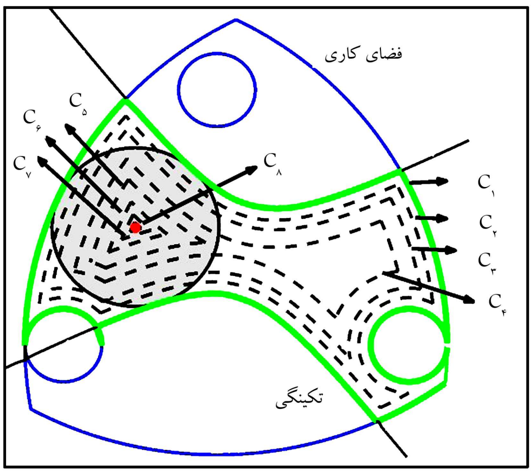
دایره‌ی عاری از تکینگی است ($C_۰$). نهایتاً از اجرای الگوریتم D برای گستره‌یی از دوران‌های مجری نهایی، $\varphi = [-\frac{\pi}{۲}, -\frac{\pi}{۳}]$، به دست آمده است (شکل ۹). چنان که مشاهده می‌شود، بزرگ‌ترین دایره‌ی عاری از تکینگی برای دوران‌های گفته شده دایره‌یی به شعاع $r = ۱۶/۸۶$ است که در $\varphi = -\frac{\pi}{۲}$ اتفاق می‌افتد.

۵. روش جابه‌جایی هندسی و یافتن نواحی عاری از تکینگی

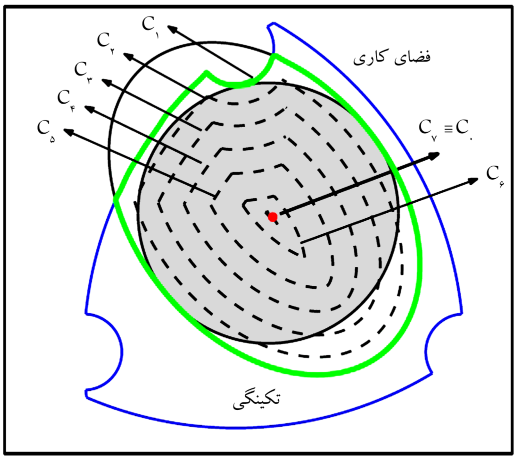
در این بخش یکی دیگر از روش‌های به دست آوردن بزرگ‌ترین دایره‌ی عاری از تکینگی به‌نام «جابه‌جایی هندسی» معرفی می‌شود که به‌نوعی از علم هندسه الهام گرفته شده است. تا آنجا که نویسنده می‌داند تاکنون برای یافتن بزرگ‌ترین دایره‌ی عاری از تکینگی از علم هندسه کمک گرفته نشده است. بیان این الگوریتم می‌تواند هم از طریق ابزارهای محاسباتی رایانه‌یی باشد و هم از طریق نرم‌افزارهای CAD صورت پذیرد. تقریباً همه‌ی نرم‌افزارهای CAD قابلیت جابه‌جایی هندسی پیچیده را دارند. تمامی مفهوم این روش هندسی مبتنی بر دو خاصیت هندسی دایره‌ی محاطی است: ۱. دایره‌ی محاط در یک منحنی در نقاط تماس با مرزها مماس است؛ ۲. فاصله‌ی مرکز این دایره تا همه‌ی نقاط تماس مساوی است. با عنایت به دو خاصیت بیان شده، مسئله‌ی یافتن بزرگ‌ترین دایره عاری از تکینگی به روش هندسی عبارت خواهد بود از: ۱. راستای مماس بر تمامی نقاط روی مرز $C_۱$ به دست می‌آید، سپس هر نقطه از منحنی $C_۱$ در راستای خط عمود بر مماس‌های به دست آمده به‌اندازه‌ی v جابه‌جا می‌شود. به این ترتیب از اجتماع نقاط جابه‌جا شده، منحنی جدیدی به‌نام $C_۲$ به دست می‌آید. با تکرار این مراحل می‌توان به منحنی $C_۳$ و به‌همین ترتیب $C_۴$ تا C_n دست یافت. این فرایند تا جایی ادامه می‌یابد که منحنی C_n به یک نقطه همگرا شود یا به‌عبارتی مساحت آن از دقت مطلوب کم‌تر شود که در این حالت با یک نقطه تقریب زده می‌شود. نقطه‌ی به دست آمده نشان‌دهنده‌ی مرکز بزرگ‌ترین دایره‌ی محاط شونده در منحنی تحت بررسی است. سپس با یافتن شعاع دایره از هر روشی دلخواه، بزرگ‌ترین دایره‌ی عاری از تکینگی به دست می‌آید.

قابل ذکر است که ممکن است در بعضی منحنی‌ها دو حالت خاص رخ دهد. نخست حالتی که یک گلوگاه در شکل وجود داشته باشد (شکل ۵). در چنین مواردی روش جابه‌جایی هندسی به تشکیل دو زیرناحیه می‌انجامد. با به‌کار بردن این الگوریتم در هر زیرناحیه می‌توان به نقطه‌ی مرکز بزرگ‌ترین دایره‌ی محاطی در همان ناحیه دست یافت. بدیهی است که بزرگ‌ترین دایره در میان تمام دوایر به دست آمده از همه‌ی زیرناحیه‌ها، بزرگ‌ترین دایره‌ی عاری از تکینگی است. دوم حالتی است که در آن نقاط شکستگی رو به داخل[۲۳] است. در این حالت روش جابه‌جایی هندسی به جواب مطلوب دست نمی‌یابد. برای حل این حالت کافی است به‌جای نقطه‌ی شکستگی یک خط بسیار کوچک قرار داده شود یا نقطه‌ی شکستگی را با دو نقطه در همسایگی‌اش تقریب زد. شایان ذکر است که این حالت در ربات‌های تحت بررسی این مقاله به‌هیچ عنوان رخ نخواهد داد.

در شکل ۱۰ و شکل ۱۱ نتایج به دست آمده برای یافتن بزرگ‌ترین دایره‌ی عاری از تکینگی با استفاده از روش جابه‌جایی هندسی در دوران‌های $\varphi = ۰$ و $\varphi = \frac{\pi}{۶}$ نشان داده شده است، که با نتایج به دست آمده از تحلیل بازه‌یی شکل ۸ و شکل ۹ تطابق کامل دارد. این نتایج دقت هر دو روش را نشان می‌دهد.



شکل ۱۰. نتایج به دست آمده برای فضای‌کاری، تکینگی و بزرگ‌ترین دایره‌ی عاری از تکینگی در دوران $\varphi = ۰$، به کمک روش جابه‌جایی هندسی.



شکل ۱۱. نتایج به دست آمده برای فضای‌کاری، تکینگی و بزرگ‌ترین دایره‌ی عاری از تکینگی در دوران $\varphi = \frac{\pi}{6}$، به کمک روش جابه‌جایی هندسی.

۶. نتیجه‌گیری

در این نوشتار دو روش برای یافتن بزرگ‌ترین دایره‌ی عاری از تکینگی در فضای کاری ربات‌های سه درجه آزادی صفحه‌یی، به‌نام‌های روش تحلیل بازه‌یی و روش هندسی جابه‌جایی هندسی ارائه شد. در روش تحلیل بازه‌یی چهار الگوریتم به‌نام‌های A، B، C و D ارائه شد تا با استفاده از آنها به‌ترتیب بتوان فضای کاری دوران ثابت، تکینگی، بزرگ‌ترین دایره‌ی عاری از تکینگی برای یک مربع خاص، و بزرگ‌ترین دایره‌ی عاری از تکینگی برای تمام فضای کاری یک ربات موازی سه درجه آزادی را پیدا کرد. به‌عنوان مطالعه‌ی موردی، این الگوریتم‌ها برای ربات ۳-RPR اجرا شد. به‌طور کلی هنر این الگوریتم در این است که برای یافتن بزرگ‌ترین دایره‌ی عاری از تکینگی در فضای کاری ربات‌های سه درجه آزادی، روشی قاعده مند[24] ارائه داده است.

در روش جابه‌جایی هندسی بیان شد که نتایج به دست آمده از هر دو روش مؤید یکدیگرند و این نشان می‌دهد که روش تحلیل بازه‌یی کاملاً استوار است و همچنین می‌توان برای تحلیل بهینه‌ی سازوکارهای دیگر از آن استفاده کرد. علاوه بر این، الگوریتم پیشنهادی در این نوشتار یکی از بزرگ‌ترین کاستی‌های موجود در ادبیات تحقیق، درمورد محدود کردن مرکز دایره به نقطه‌ی شروع جست‌وجو، را برطرف می‌سازد. مطالعات و کارهای آینده در راستای گسترش و توسعه‌ی الگوریتم پیشنهادی به سازوکارهای موازی پیچیده‌تر مانند گاف ـ استوارت است.

تقدیر و تشکر

نویسندگان این مقاله کمال تشکر و سپاسگذاری خود را از صندوق حمایت از پژوهشگران و فناوران نهاد ریاست جمهوری، بدلیل حمایت‌های مادی و معنوی، دارند.

پانوشت‌ها

1. singularity
2. Gough-Stewart
3. workspace
4. degree-of-freedom (DOF)
5. end-effector
6. forward kinematic equation (FKE)
7. inverse kinematic singularity
8. screw theory
9. grassmann-cayley algebra
10. constant-orientation
11. point displacement
12. practical swarm optimization
13. constrain optimization
14. interval analysis
15. evolutionary algorithm
16. genetic algorithm
17. Intlab
18. Matlab
19. offset approach
20. revolute
21. prismatic
22. passive
23. cuspidal point
24. systematic

منابع (References)

1. Davidson, J., Hunt, K. and Pennock, G. "Robots and screw theory", *Applications of Kinematics and Statics to Robotics*, (2004).
2. Dasgupta, B. and Mruthyunjaya, T. "The stewart platform manipulator: A Review", *Mechanism and Machine Theory*, **35**(1), pp. 15-40 (2000).
3. Stewart, D. "A platform with six degrees of freedom", *Sage Publications*, **180**(1), pp. 371-386 (1965).
4. Clavel, R. "Delta, a fast robot with parallel geometry", *in Proceedings of the 18th International Symposium on Industrial Robots*, pp. 91-100 (1988).
5. Parallelmic, http://www.parallemic.org/.
6. Kong, X. and Gosselin, C. "Type synthesis of parallel mechanisms", *Springer*, Heidelberg, **33**, p.2 (2007).
7. Gosselin, C. and Angeles, J. "Singularity analysis of closed-loop kinematic chains", *IEEE Transactions on Robotics and Automation*, **6**(3), pp. 281-290 (1990).

8. Saadatzi, M., Tale Masouleh, M., Taghirad, H., Gosselin, C. and Cardou, P. "On the optimum design of 3-RPR parallel mechanisms", in 19th Iranian Conference on Electrical Engineering (ICEE). IEEE, pp. 1-6 (2011).

9. Tale Masouleh, M. and Gosselin, C. "Determination of singularity-free zones in the workspace of planar 3-PRR parallel mechanisms", *Journal of Mechanical Design,* **129**, pp. 649 (2007).

10. Merlet, J.P. "Parallel robots", Springer, (2006).

11. Merlet, J. and Donelan, P. "On the regularity of the inverse jacobian of parallel robots", Advances in Robot Kinematics, pp. 41-48 (2006).

12. Conconi M. and Carricato, M. "A new assessment of singularities of parallel kinematic chains", *IEEE Transactions on Robotics*, **25**(4), pp. 757-770 (2009).

13. Bonev, I.A. Zlatanov, D. and Gosselin, C.M. "Singularity analysis of 3-DOF planar parallel mechanisms via screw theory", *Journal of Mechanical Design,* **125**, pp. 573 (2003).

14. Saadatzi, M. Tale Masouleh, M. and Taghirad, H. "Workspace analysis of 5-PRUR parallel mechanisms (3T2R)", Robotics and Computer-Integrated Manufacturing (2012).

15. Yang, Y. and O'Brien, J. "A case study of planar 3-RPR parallel robot singularity free workspace design", in International Conference on Mechatronics and Automation (ICMA). IEEE, pp. 1834-1838 (2007).

16. Li, H. Gosselin, C. and Richard, M. "Determination of maximal singularity-free zones in the workspace of planar three-degree-of-freedom parallel mechanisms", *Mechanism and machine theory,* **41**(10), pp. 1157-1167 (2006).

17. Li, H., Gosselin, C. and Richard, M.J. "Determination of the maximal singularity-free zones in the six-dimensional workspace of the general gough-stewart platform", *Mechanism and Machine Theory*, **42**(4), pp. 497-511 (2007).

18. Jiang, Q. and Gosselin, C.M. "Geometric synthesis of planar 3-RPR parallel mechanisms for singularity-free workspace", *Transactions of the Canadian Society for Mechanical Engineering*, **33**(4), pp. 667-678 (2009).

19. Jiang, Q. and Gosselin, C.M. "The maximal singularity-free workspace of planar 3-RPR parallel mechanisms", *in Proceedings of the 2006 International Conference on Mechatronics and Automation*, IEEE, pp. 142-146 (2006).

20. Jiang, Q. and Gosselin, C.M. "Geometric optimization of planar 3-RPR parallel mechanisms", *Transactions of the Canadian Society for Mechanical Engineering*, **31**(4), pp. 457-468 (2007).

21. Abbasnejad, G., Daniali, H. and Kazemi, S. "A new approach to determine the maximal singularity-free zone of 3-RPR planar parallel manipulator", *Robotica*, **30**(6), pp. 1005-1012 (2012).

22. Voglewede, P. and Ebert-Uphoff, I. "Overarching framework for measuring closeness to singularities of parallel manipulators", *IEEE Transactions on Robotics*, **21**(6), pp. 1037-1045 (2005).

23. Moore, R. and Bierbaum, F. "Methods and applications of interval analysis", *Society for Industrial Mathematics*, **2**, pp.3-4 (1979).

24. Hao, F. and Merlet, J. "Multi-criteria optimal design of parallel manipulators based on interval analysis", *Mechanism and Machine Theory*, **40**(2), pp. 157-171 (2005).

25. Merlet, J. "Interval analysis and robotics", Robotics Research, pp. 147-156 (2011).

26. Merlet, J.-P. "Solving the forward kinematics of a gough-type parallel manipulator with interval analysis", *The International Journal of Robotics Research,* **23**(3), pp. 221-235 (2004).

27. Oetomo, D., Daney, D., Shirinzadeh, B. and Merlet, J.-P. "Certified workspace analysis of 3RRR planar parallel flexure mechanism", in IEEE International Conference on Robotics and Automation (ICRA). IEEE, pp. 3838-3843 (2008).

28. Bonev, I.A. "Geometric analysis of parallel mechanisms", Ph.D. Dissertation, Laval University, Quebec, QC, Canada (October 2002).

29. Sefrioui, J. and Gosselin, C. "On the quadratic nature of the singularity curves of planar three-degree-of-freedom parallel manipulators", *Mechanism and Machine Theory*, **30**(4), pp. 553-551 (1995).

30. Gosselin, C. "Determination of the workspace of 6-DOF parallel manipulators", *ASME Journal of Mechanical Design,* **112**(3), pp. 331-336 (1990).