



دانشگاه خواجہ نصیرالدین طوسی
دانشکده مهندسی برق - گروه کنترل

پایان نامه دوره کارشناسی ارشد

مهندسی برق

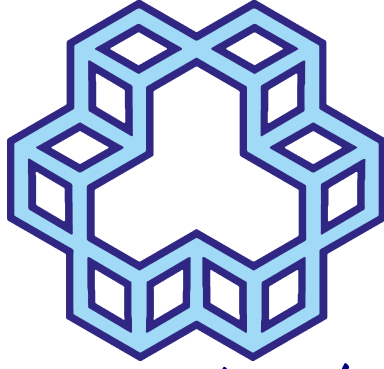
ارائه فرمول بندی یکپارچه بهینه سازی مبتنی بر گراف به منظور کالیبراسیون و تخمین حالت ربات

محمد رضا دیندارلو

استاد راهنما

دکتر حمیدرضا تقی راد

تابستان ۱۴۰۳



دانشگاه خواجہ نصیرالدین طوسی
دانشکده مهندسی برق - گروه کنترل

پایان نامه دوره کارشناسی ارشد
مهندسی برق

عنوان

ارائه فرمول بندی یکپارچه بهینه سازی مبتنی بر
گراف به منظور کالیبراسیون و تخمین حالت ربات

نگارش

محمد رضا دیندارلو

استاد راهنما

دکتر حمید رضا تقی راد

استادان مشاور

دکتر فلیپ کاردو و دکتر سید احمد خلیل پور

تابستان ۱۴۰۳



تقدیم به:

به آنان که با علم خود زندگی آزاد می سازند



تأییدیه هیئت داوران جلسه‌ی دفاع از پایان‌نامه کارشناسی ارشد

هیأت داوران پس از مطالعه‌ی پایان‌نامه و شرکت در جلسه‌ی دفاع از پایان‌نامه تهیه‌شده با عنوان «ارائه فرمول بندی یکپارچه بهینه سازی مبتنی بر گراف به منظور کالیبراسیون و تخمین حالت ربات» توسط آقای / خانم محمدرضا دیندارلو صحت و کفایت تحقیق انجام شده را برای اخذ درجه‌ی کارشناسی ارشد در رشته‌ی مهندسی برق در تاریخ تابستان ۱۴۰۳ مورد تأیید قرار دادند.

۱. استاد راهنما: دکتر حمیدرضا تقی راد..... امضا

۲. استاد مشاور: دکتر فلیپ کاردو..... امضا

۳. استاد مشاور: دکتر سید احمد خلیل پور..... امضا

۴. استاد داور داخلی: دکتر داور داخلی..... امضا

۵. استاد مدعو: دکتر داور خارجی..... امضا

۶. نماینده تحصیلات تکمیلی دانشکده: دکتر نماینده..... امضا



دانشگاه صنعتی خواجه نصیرالدین طوسی

اظهارنامه دانشجو

اینجانب محمدرضا دیندارلو به شماره دانشجویی ۴۰۰۳۰۸۲۴ دانشجوی کارشناسی ارشد رشته‌ی مهندسی برق دانشکده برق دانشگاه خواجه نصیرالدین طوسی گواهی می‌نمایم که تحقیقات ارائه شده در این پایان‌نامه با عنوان:

ارائه فرمول بندی یکپارچه بهینه سازی مبتنی بر گراف به منظور کالیراسیون و تخمین حالت ربات

توسط اینجانب انجام و بدون هرگونه دخل و تصرف است و موارد نسخه برداری شده از آثار دیگران را با ذکر کامل مشخصات منبع ذکر کرده‌ام. در صورت اثبات خلاف مندرجات فوق، به تشخیص دانشگاه مطابق با ضوابط و مقررات حاکم (قانون حمایت از حقوق مؤلفان و مصنفان و قانون ترجمه و تکثیر کتب و نشریات و آثار صوتی، ضوابط و مقررات آموزشی، پژوهشی و انضباطی و غیره) با اینجانب رفتار خواهد شد. در ضمن، مسئولیت هرگونه پاسخگویی به اشخاص اعم از حقیقی و حقوقی و مراجع ذی صلاح (اعم از اداری و قضایی) به عهده‌ی اینجانب خواهد بود و دانشگاه هیچ گونه مسئولیتی در این خصوص نخواهد داشت.

نام و نام خانوادگی دانشجو: محمدرضا دیندارلو

تاریخ و امضای دانشجو:



حق طبع، نشر و مالکیت نتایج

حق چاپ و تکثیر این پایان نامه متعلق به نویسندگان آن می باشد. بهره برداری از این پایان نامه در چهارچوب مقررات کتابخانه و با توجه به محدودیتی که توسط استاد راهنما به شرح زیر تعیین می گردد، بلامانع است:

☐ بهره برداری از این پایان نامه برای همگان و با ذکر منبع، بلامانع است.

☐ بهره برداری از این پایان نامه با اخذ مجوز از استاد راهنما و با ذکر منبع، بلامانع است.

☐ بهره برداری از این پایان نامه تا تاریخ _____ ممنوع است.

استاد راهنما: دکتر حمیدرضا تقی راد امضا

استاد مشاور: دکتر فلیپ کاردو امضا

استاد مشاور: دکتر سید احمد خلیل پور امضا

قدردانی

اکنون که به یاری پروردگار و یاری و راهنمایی اساتید بزرگ موفق به پایان این رساله شده‌ام وظیفه خود دانسته که نهایت سپاسگزاری را از تمامی عزیزانی که در این راه به من کمک کرده‌اند را به عمل آورم: در آغاز از استاد بزرگ و دانشمند جناب آقای/سرکار خانم که راهنمایی این پایانامه را به عهده داشته‌اند کمال تشکر را دارم. از جناب آقایان/ خانم‌ها که اساتید مشاور این پایانامه بوده‌اند نیز قدردانی می‌نمایم. از داوران گرامی ... که زحمت داوری و تصحیح این پایانامه را به عهده داشتند کمال سپاس را دارم. خالصانه از تمامی اساتید و معلمان و مدرسانی که در مقاطع مختلف تحصیلی به من علم آموخته و مرا از سرچشمه دانایی سیراب کرده‌اند متشکرم. از کلیه هم دانشجویان و همراهان عزیز، دوستان خوبم خانم‌ها و آقایان نهایت سپاس را دارم. و در پایان این پایان‌نامه را تقدیم می‌کنم به که با حضورش و همراهی اش همیشه راه را به من نشان داده و مرا در این راه استوار و ثابت قدم نموده است.

محمد رضا دیندارلو

تابستان ۱۴۰۳

چکیده

واژگان کلیدی

فهرست مطالب

ت	فهرست تصاویر
ث	فهرست جداول
ج	فهرست الگوریتم‌ها
ح	فهرست برنامه‌ها
۱	فصل ۱: مقدمه
۳	فصل ۲: مروری بر مطالعات انجام شده
۵	فصل ۳: رویکرد گراف مبنا موقعیت‌یابی و کالیبراسیون به صورت همزمان
۵	۱.۳ مقدمه
۶	۲.۳ روش‌های مرسوم مسئله کالیبراسیون
۷	۱.۲.۳ ترکیب حسگرها
۸	۲.۲.۳ ترکیب شبه اندازه‌گیری‌ها
۸	۳.۳ روش‌های مرسوم مسئله موقعیت‌یابی
۱۰	۴.۳ رویکرد گراف مبنا برای حل مسئله کالیبراسیون و موقعیت‌یابی به صورت همزمان
۱۱	۱.۴.۳ بیان الگوریتم گراف عامل
۱۲	۲.۴.۳ گراف عامل پیشنهادی برای کالیبراسیون و موقعیت‌یابی به صورت همزمان
۱۷	۵.۳ نتیجه‌گیری

فصل ۴: پیاده سازی رویکرد گراف مبنا جهت موقعیت یابی و کالیبراسیون همزمان برای ربات

کابلی صلب	۱۹
انتخاب ربات مناسب جهت توسعه الگوریتم	۱۹
توسعه گراف عامل برای یک ربات چهار کابلی با فرض کابل صلب	۲۰
معرفی ربات کابلی ARASCam	۱۰۲.۴
بیان فرمول بندی مسئله و فرضیات	۲۱
رویت پذیری	۳۰۲.۴
بهینه سازی پارامترها با استفاده از گراف های عاملی	۲۴
نتایج پیاده سازی	۴.۴
راه اندازی سیستم و فرضیات	۱.۴.۴
کالیبراسیون خودکار بدون پارامترهای اولیه	۲.۴.۴
ترکیب بینایی-سینماتیک و استفاده از گراف عامل معرفی شده	۳.۴.۴
انتشار عدم قطعیت	۴.۴.۴
بحث و گفت وگو	۵.۴.۴

فصل ۵: پیاده سازی رویکرد گراف مبنا جهت موقعیت یابی و کالیبراسیون همزمان برای ربات

کابلی خم شده	۳۳
نمادها و تعاریف	۱.۵
معادلات مدل کابل خم شده	۲.۵
سینماتیک ربات	۳.۵
گراف عامل کالیبراسیون و مکان یابی همزمان سینماتیک-ایستا	۴.۵
عامل طول کابل خم شده	۱.۴.۵
عامل مکان اتصال کابل به پولی	۲.۴.۵
عامل اندازه گیری نیرو	۳.۴.۵
عامل پیشین مکان	۴.۴.۵
نتایج شبیه سازی	۵.۵

۴۱	۱.۵.۵	صحت‌سنجی مدل
۴۳	۲.۵.۵	نتایج نهایی کالیبراسیون با گراف عامل توسعه داده‌شده
۴۵	۶.۵	بحث
۴۵	۱.۶.۵	اهمیت در نظر گرفتن اثر آویزان شدن کابل
۴۵	۲.۶.۵	نکات مربوط به روش مقداردهی اولیه
۴۷	فصل ۶:	نتیجه‌گیری و پیشنهادات برای آینده
۴۹	مراجع	
۵۳	پیوست آ:	آشنایی سریع با برخی دستورات لاتک
۵۵	پیوست ب:	جدول، نمودار و الگوریتم در لاتک
۵۷	پیوست پ:	مراجع، واژه‌نامه و حاشیه‌نویسی

فهرست تصاویر

۶	۱.۳
۱۳	۲.۳
۱۴	۳.۳
۱۵	۴.۳
۱۶	۵.۳
۱۶	۶.۳
۲۱ ARAS-CAM معلق	۱.۴
۲۴ صلب	۲.۴
۳.۴	پالایش متوالی یک باور یکنواخت بر روی پارامترهای سینماتیکی از طریق الگوریتم مونت-کارلو
۲۷	۴.۴
۴.۴	راست: توزیع خطای پنجه قبل و بعد از بهبود پارامترها، چپ: خطای مکان‌یابی در دستگاه
۲۸	کارتزین
۲۹	۵.۴
۳۴	۱.۵
۳۸	۲.۵
۴۱	۳.۵
۴۲	۴.۵

فهرست جداول

۱.۴	خطای مکان‌یابی با روش‌های مختلف (واحد متر)	۲۸
۲.۴	سازگاری آماری عدم قطعیت‌های تخمین زده شده	۳۰
۱.۵	صحت‌سنجی مدل	۴۲
۲.۵	نتایج میانگین خطای مطلق کالیبراسیون با استفاده از گراف عامل توسعه داده شده	۴۴

فهرست الگوریتم‌ها

فهرست برنامه‌ها

فصل ۱

مقدمه

در این بخش، به مرور کارهای انجام شده پیشین در این موضوع پرداخته‌ایم. سپس، به بررسی ضعف‌ها و خلأهای موجود در این حوزه می‌پردازیم. پس از آن، فرضیات مطرح شده در تحقیق حاضر توضیح داده می‌شوند. در ادامه، به معرفی دقیق تحقیقات انجام شده توسط خودمان می‌پردازیم، که شامل بخش‌هایی است که به عنوان novelty در نظر گرفته شده‌اند. باید توجه داشت که موارد مطرح شده در این فصل باید مختصر و کلی باشند، و جزئیات بیشتر به فصل‌های آتی اختصاص داده شود.

فصل ۲

مروری بر مطالعات انجام شده

این بخش باید حداکثر شامل ۲۵ صفحه باشد و به پیشینه تحقیق و استناد به کارهای پیشین در این حوزه بپردازد. این بخش متشکل از سه قسمت است که به ترتیب به کالیراسیون، موقعیت یابی، و فکتور گراف می پردازند.

فصل ۳

رویکرد گراف مبنا موقعیت‌یابی و کالیبراسیون به صورت همزمان

۱.۳ مقدمه

همانطور که در فصل قبل ذکر شد، اگرچه حسگرهای فضای مفصل سریع و ارزان هستند، اما زمانی که از آنها برای اندازه‌گیری مقادیر مجری نهایی استفاده می‌شود، دقت مدل سینماتیکی برای تعیین دقت قابل دستیابی بسیار مهم است. علاوه بر این، در زمینه همجوشی و ترکیب اندازه‌گیری‌ها، هم‌ثبت کردن داده‌ها [۱] اولین گام اساسی است. به عبارت دیگر، حسگرها باید اندازه‌گیری‌های خود را در یک مختصات یکپارچه ارائه دهند. اهمیت هم‌ثبت به دلیل فرض اساسی نویز گاوسی با میانگین صفر در الگوریتم‌های ترکیب داده‌ها است.

نکته قابل توجه دیگر برای ربات‌های آسان نصب، لزوم بی‌نیازی الگوریتم کالیبراسیون پیشنهادی به حسگرهای گران‌قیمت و یا حسگرهایی که نیاز به تعمیر و نگهداری سطح بالایی دارند، می‌باشد. علاوه بر این، فرآیند کالیبراسیون باید به اندازه‌ای ساده باشد که اجرای آن در مکان‌های مختلف آسان و سریع باشد. با اینکه کالیبراسیون موضوعی است که بسیاری از پژوهشگران به آن علاقه‌مند هستند، اما مفهوم بهره‌گیری از چندین حسگر برای بهبود نتایج کمتر مورد توجه قرار گرفته است.

از طرفی دیگر، افزون بر مفهوم و ضرورت کالیبراسیون در حوزه ربات‌ها، موقعیت‌یابی آنها نیز مورد توجه بسیاری قرار گرفته است. همانطور که پیش‌تر بیان شد، الگوریتم‌های بسیاری در راستای ترکیب حسگرها و همچنین کاهش زمان پردازش برای موقعیت‌یابی ربات به صورت زمان-واقعی در انواع دیگر ربات‌ها همچون



شکل ۱.۳

ربات‌های خودران مورد استفاده قرار گرفته است.

در این فصل مروری بر روش‌های مرسوم کالیبراسیون و موقعیت‌یابی ربات‌ها خواهیم داشت. سپس نگاهی به معایب این روش‌ها خواهیم داشت و برای حل آن‌ها رویکردی را ارائه خواهیم داد که معایب این روش‌ها را برطرف کند. در نهایت با استفاده از این رویکرد، یک ربات کابلی را در نظر خواهیم گرفت و با اعمال رویکرد مطرح شده، نتایج کالیبراسیون و موقعیت‌یابی را به صورت همزمان ارائه خواهیم داد.

۲.۳ روش‌های مرسوم مسئله کالیبراسیون

به صورت کلی، انتظار می‌رود چنانچه به یک ربات در دنیای واقع یک ورودی مشخص اعمال شود، با اعمال همان ورودی به مدل پاسخی یکسان دریافت شود. با این حال همواره وجود نامعینی‌ها و عدم دقیق بودن پارامترهای مدل در واقعیت ما را از رسیدن به چنین پاسخی ایده‌آل باز می‌دارد. این نامعینی‌ها می‌تواند ناشی از تقریب‌هایی باشد که در مدل داریم و یا پدیده‌هایی که در مدل‌سازی مورد توجه کامل قرار نگرفته‌اند. جنس این نامعینی‌ها می‌تواند ریشه در سینماتیک ربات و یا دینامیک آن باشد. فرآیند کالیبراسیون می‌تواند این نامعینی‌ها را در جهتی کاهش دهد که پاسخ‌هایی که از مدل و ربات در پیاده‌سازی واقعی دریافت می‌کنیم، کاهش پیدا کند. آنچه در این کار مورد بررسی قرار گرفته است کالیبراسیون سینماتیکی می‌باشد. شکل ۱.۳ نمایش بلوکی از یک فرآیند کالیبراسیون سینماتیکی بنا بر تعریف بیان شده می‌باشد. همانطور که در این شکل مشاهده می‌شود آنچه به عنوان خطا در نظر گرفته می‌شود تفاوت موقعیت فضایی ربات است که ناشی از مدل سینماتیکی ربات (در اینجا سینماتیک مستقیم) و ربات واقعی در فضای کاری ربات، با یک ورودی مشترک در فضای مفصلی آن می‌باشد.

با نگاهی به آخرین تحقیقات بر روی مسئله کالیبراسیون ربات‌ها، ایجاد یک مسئله بهینه‌سازی غیرخطی و حل آن برای یافتن مقادیر دقیق این پارامترهای سینماتیکی و دینامیکی ربات مرسوم می‌باشد [۲، ۳، ۴، ۵].

مطابق این رویکرد های مرسوم، برای ایجاد فرمول بندی مناسب مسئله مطرح شده در شکل ۱.۳ خواهیم داشت:

$$\tilde{\phi} = \arg \min_{\phi} \sum_{n=1}^N \text{error}_i[n] = \arg \min_{\phi} \sum_{n=1}^N \|F_i(\mathbf{q}[n], \phi) - X_i[n]\|_{\Sigma}^2 \quad (1.3)$$

در این معادله، ϕ بردار پارامتر های سینماتیکی و $\tilde{\phi}$ تخمین آن است. علاوه بر این، $X_i[n]$ مقدار اندازه گیری شده توسط حسگر فضای کار ربات، و $\mathbf{q}[.]$ مقادیر اندازه گیری های متقابل حسگری در مفاصل می باشد. تابع مدل ربات $F[.]$ بیانگر مدل سینماتیک مستقیم ربات می باشد. تابع هزینه هدف، بر روی مجموعی از N نمونه داده جمع آوری شده در فرآیند کالیبراسیون می باشد. افزون بر این، Σ نیز بیانگر ماتریس کوواریانس اندازه گیری می باشد که به عنوان عامل نرمال سازی برای محاسبه هزینه عمل می کند. هر چه مقدار کوواریانس بیشتر باشد، میزان تاثیر گذاری خطای متقابل آن بر روی تابع هزینه کمتر خواهد بود. همچنین برای محاسبه نرم روش های زیادی ارائه شده است که آنچه بیشتر مورد استفاده قرار می گیرد نرم های هابر^۱ می باشد [۶]. معادله بهینه سازی غیر خطی بیان شده در ۱.۳ می تواند با روش های بازگشتی الگوریتم های غیر خطی حداقل مربعات^۲ همچون لونبرگ-مارکوارت^۳ و یا روش های گاوس-نیوتون^۴ می باشد [۷].

با نگاهی دیگر به دیاگرام مطرح شده در ۱.۳ و همچنین معادله ۱.۳، مشاهده می شود که افزایش دقت اندازه گیری و همچنین برآورده کردن تمامی قیود مدل می تواند منجر به بهبود نتیجه کالیبراسیون شود. به منظور دستیابی به این هدف، رویکردهایی همچون ترکیب چندین حسگر و یا افزودن قیود جدید که از ساختار هندسی ربات استخراج می شود، معرفی می شوند. ترکیب این حسگرها باید به گونه ای باشد که علاوه بر کاهش خطای نهایی کالیبراسیون، خروج هر کدام از حسگرها منجر به توقف فرآیند کالیبراسیون نشود. همچنین واضح است که افزودن این قیود می تواند منجر به حل پیچیده تری از مسئله شود. در ادامه، نگاهی به فرمول بندی مسئله کالیبراسیون با در نظر گرفتن این ترکیب ها خواهیم داشت.

۱.۲.۳ ترکیب حسگرها

در معادله ۱.۳، زیر نویس i بیانگر وجود یک حسگر و خطایی که از مقادیر اندازه گیری حسگر در هر نمونه بوده می باشد. فرمول بندی ساختاری که به صورت همزمان از چندین حسگر در راستای ایجاد تابع هزینه استفاده

نماید می‌تواند به صورت زیر تعریف شود:

$$\tilde{\phi} = \arg \min_{\phi} \sum_{n=1}^N \sum_{i=1}^M W_i \text{error}_i[n] \quad (2.3)$$

در این معادله W_i ها یک پارامتر وزن برای ترکیب چندین منبع اطلاعاتی با توجه به میزان کیفیت و اهمیت آنها می‌باشد.

۲.۲.۳ ترکیب شبه اندازه‌گیری‌ها

اندازه‌گیری‌های حسگری تنها منبع اطلاعات برای حل مسئله نیستند. ساختار ربات و توجه به هندسه آن برای حل مسئله تعریف شده می‌تواند مفید واقع شود. برای مثال فاصله‌های بین برخی از نقاط می‌تواند با توجه به ساختار ربات می‌تواند ویژگی‌هایی نسبی و یا مطلق داشته باشند. این اطلاعات با عنوان داده‌های شبه اندازه‌گیری شناخته می‌شوند. این دسته از اطلاعات که از قبل مشخص هستند، می‌توانند به صورت قیود به مسئله اضافه شوند. بنابراین مسئله کلی بهینه‌سازی کالیبراسیون ۲.۳ در حضور این قیود به صورت زیر باز نویسی می‌شود.

$$\tilde{\phi} = \arg \min_{\phi} \sum_{n=1}^N \sum_{i=1}^M W_i \text{error}_i[n] \quad (3.3)$$

$$g_j(\phi) = 0 \quad \text{where} \quad j = 1, \dots, K$$

که در اینجا $g_j(\phi) = 0$ قیود هندسی معلوم بر روی پارامترهای سینماتیکی ربات می‌باشند.

۳.۳ روش‌های مرسوم مسئله موقعیت‌یابی

موقعیت‌یابی^۱ ربات فرآیند تعیین مکان ربات نسبت به محیط اطراف آن می‌باشد. دانستن موقعیت دقیق ربات در محیط، پیش‌نیازی اساسی برای اتخاذ تصمیمات صحیح و حرکت‌های بعدی مؤثر است. بدون اطلاعات موقعیتی دقیق، ربات نمی‌تواند مسیریابی^۲ و یا ردیابی^۳ مناسبی را داشته باشد و ممکن است با موانع برخورد کند

یا مسیر بهینه‌ای را طی نکند [۸]. علاوه بر این، سیستم‌های کنترلی ربات‌ها نیازمند اطلاعات دقیق و لحظه‌ای از موقعیت و جهت‌گیری ربات هستند تا بتوانند فرمان‌های مناسب را صادر کنند. بدون داده‌های دقیق موقعیتی، کنترلرها نمی‌توانند حرکات دقیقی را تولید کنند که منجر به عملکرد نامناسب و ناپایداری ربات می‌شود [۹]. فرآیند کالیبراسیون ربات که در بخش ۲.۳ مورد بررسی قرار گرفت نیز یازمند داشتن اطلاعات دقیق از موقعیت ربات است. با داشتن داده‌های موقعیتی دقیق، می‌توان خطاهای سیستماتیک را شناسایی و تصحیح کرد و به این ترتیب دقت و کارایی ربات را بهبود بخشید. این امر به ویژه در ربات‌هایی که نیاز به انجام وظایف حساس و دقیق دارند، حیاتی است.

موقعیت‌یابی دقیق ربات باعث کاهش عدم قطعیت در تصمیم‌گیری‌ها و عملیات ربات می‌شود. این امر نه تنها به افزایش اعتمادپذیری ربات در انجام وظایف محوله منجر می‌شود، بلکه احتمال بروز خطاها و حوادث ناشی از اشتباهات موقعیتی را نیز کاهش می‌دهد. همچنین در سیستم‌هایی که شامل چندین ربات هستند، اطلاعات دقیق موقعیتی هر ربات برای هماهنگی و همکاری بین ربات‌ها ضروری است. این اطلاعات به ربات‌ها کمک می‌کند تا از موقعیت یکدیگر آگاه باشند و بتوانند به صورت هماهنگ وظایف مشترک را انجام دهند. در این راستا توسعه و بهبود تکنیک‌های موقعیت‌یابی به منظور افزایش دقت و کارایی ربات‌ها، از اهمیت ویژه‌ای برخوردار است [۱۰].

روش‌های ارائه شده برای موقعیت‌یابی ربات را می‌توان به سه دسته اصلی مسافت پیمایی^۱، موقعیت‌یابی جهانی^۲ و مکان‌یابی و نقشه‌یابی به صورت همزمان^۳ تقسیم کرد. این روش‌ها با توجه به نوع حسگرهای تعبیه شده بر روی ربات می‌تواند مورد استفاده قرار گیرد. ترکیب داده‌ها برای همانند آنچه در بخش ۲.۳ مورد توجه قرار گرفت، در موقعیت‌یابی و تخمین حالت ربات نیز می‌تواند نقش موثری را ایفا کند. حسگرهای استفاده شده از نظر جنس داده‌ها و فرکانس داده برداری نیز می‌تواند متفاوت باشد که در ترکیب داده‌ها خصوصاً زمانی که اجرای الگوریتم به صورت زمان واقعی می‌باشد، چالش برانگیز خواهد بود. طیف وسیعی از روش‌های مرسوم ارائه شده برای ترکیب داده‌ها در راستای تخمین حالت، رویکردهای بر مبنای فیلتر هستند. این روش‌ها که به رویکردهای آماری^۴ نیز شناخته می‌شوند، در دو دهه اخیر فعالیت‌های زیادی را به خود اختصاص داده‌اند. پایه این روش‌ها بر قانون بیز^۵ بنا نهاده شده است. مقاله [۱۱] دسته‌بندی جامعی از روش‌های فیلتر مبنا برای تخمین موقعیت ارائه کرده است. از میان روش‌های بیان شده، کالمن فیلتر و فیلترهای ذرات^۶ به عنوان فراگیرترین رویکرد مورد استفاده قرار گرفته شده است. این فیلترها با استفاده از فرض مارکوفی برای حالت‌ها و به کارگیری اطلاعات پیشین، تخمینی از حالت جدید ارائه می‌کنند.

۴.۳ رویکرد گراف مبنا برای حل مسئله کالیبراسیون و موقعیت‌یابی به

صورت همزمان

روش‌های مرسوم کالیبراسیون و موقعیت‌یابی ربات‌ها که تا به اینجا معرفی شده‌اند، فرمول‌بندی‌های مشخصی برای حل این دو مسئله ارائه داده‌اند. همانطور که در فصل قبل مشاهده شد، سادگی و سرعت بالای این روش‌ها باعث پیاده‌سازی گسترده آن‌ها گردیده است. با این حال، این روش‌ها دارای معایبی نیز هستند. اول اینکه برای هر مسئله کالیبراسیون و ربات، فرمول‌بندی مسئله باید از ابتدا توسعه داده شود. دوم اینکه این روش‌ها از تنگ بودن ذاتی مسئله‌ها برای سرعت بخشیدن به محاسبات استفاده نمی‌کنند. بزرگ شدن و پیچیده شدن فرمول‌بندی این مسائل باعث می‌شود که از حل آن‌ها به صورت زمان واقعی فاصله گرفته شود. همچنین، این روش‌ها برای حل مسائل غیرخطی نیاز به خطی‌سازی دارند که نه تنها به پیچیدگی‌های محاسباتی می‌افزاید، بلکه باعث کاهش دقت نیز می‌شود. سومین موضوع، روش‌های مرسوم فیلتر مبنا از داده‌های جاری و لحظه‌ای استفاده می‌کنند که باعث می‌شود علاوه بر مشکلات در مدیریت داده‌هایی که با تأخیر به سیستم می‌رسند، نتوانند داده‌های تاریخی را به صورت کامل در یک مسئله بهینه‌سازی دسته‌ای وارد کنند. این مشکل در مسائل موقعیت‌یابی باعث ایجاد مشکلات جدی همچون لغزش^۱ و کاهش دقت تا حد قابل توجهی می‌شود. چهارمین عیب این روش‌ها، عدم انعطاف‌پذیری آن‌ها برای بسط دادن مسئله با افزودن قیود به سیستم یا داده‌های حسگری به آن است. با توجه به این معایب، روش‌های مرسوم ممکن است در برخی کاربردهای پیشرفته رباتیک کارایی لازم را نداشته باشند.

در این فصل، رویکردی گراف مبنا برای حل مسئله کالیبراسیون و موقعیت‌یابی ربات بیان می‌گردد که با فرمول‌بندی یکپارچه، هر دو مسئله را به صورت همزمان در یک مسئله بهینه‌سازی حل می‌کند. ویژگی‌های ذاتی این رویکرد در حل این مسئله واحد به تمامی معایب مطرح شده در روش‌های مرسوم پاسخ می‌دهد و باعث ایجاد حلی کامل و قابل بسط می‌شود. این رویکرد گراف مبنا به دلیل استفاده از ساختارهای گرافی، قادر به مدیریت بهینه‌تر داده‌های مختلف است. با ادغام تمامی داده‌های تاریخی و جاری در یک مسئله بهینه‌سازی دسته‌ای، این روش از داده‌های ورودی به صورت کامل استفاده کرده و به مشکلات مدیریت داده‌های تأخیر دار و لحظه‌ای غلبه می‌کند. همچنین، به دلیل عدم نیاز به خطی‌سازی مکرر، دقت محاسبات افزایش یافته و پیچیدگی‌های محاسباتی کاهش می‌یابد. علاوه بر این، انعطاف‌پذیری بالای این رویکرد امکان افزودن قیود و داده‌های حسگری جدید را فراهم می‌کند، بدون آن که نیاز به بازتعریف کلی فرمول‌بندی باشد. این ویژگی‌ها، در کنار توانایی بهره‌گیری از تنگ بودن ذاتی مسئله‌ها برای بهینه‌سازی محاسبات، این رویکرد گراف مبنا را به یک ابزار قدرتمند برای کالیبراسیون و موقعیت‌یابی ربات‌ها تبدیل می‌کند.

الگوریتم گراف مبنا استفاده شده برای این فرمول‌بندی در این پایان‌نامه، الگوریتم گراف عامل^۱ می‌باشد. در ادامه، ابتدا نگاهی بر ریاضیات مرتبط با الگوریتم گراف عامل خواهیم داشت و سپس گراف عامل یکپارچه‌ای را برای حل مسئله مطرح شده معرفی خواهیم کرد. گراف عاملی که در ادامه پیشنهاد خواهد شد، حلی سیستماتیک برای کنار هم قرار دادن بلوک‌های ساختاری (عامل‌ها) در راستای تعریف یک مسئله کالیبراسیون در کنار مسئله موقعیت‌یابی به صورت یکپارچه است، که قابلیت گسترش به حسگرهای بیشتر و قیود اضافی را نیز دارا می‌باشد. علاوه بر این، از آنجایی که پیاده‌سازی‌های منابع باز و بهینه‌سازی شده برای این روش وجود دارد، مسئله کالیبراسیون و موقعیت‌یابی همزمان مطرح شده می‌تواند بر روی سیستم‌های نهفته بر روی ربات نیز پیاده‌سازی گردد. در این پایان‌نامه برای پیاده‌سازی گراف عامل پیشنهاد داده شده، از کتابخانه GTSAM استفاده شده است.

۱.۴.۳ بیان الگوریتم گراف عامل

مسئله غیرخطی تعریف شده در ۳.۳ می‌تواند به صورت یک مدل گرافی که متشکل از گره‌هایی است که بیانگر متغیرهای تصمیم‌گیری هستند و یال‌هایی که ارتباط بین قیود و این متغیرها را نشان می‌دهند، بیان شود. در جامعه رباتیک، گراف عامل نمونه‌ای از این بیان است. این گراف‌ها یک چارچوب قوی و قابل انطباق برای بیان مسائل متنوع از تخمین حالت^۲ تا برنامه‌ریزی حرکت^۳، ارائه می‌دهند. این الگوریتم برای حل مسائل بهینه‌سازی که شامل متغیرهای مختلف و قیود پیچیده است، مناسب می‌باشد. یکی از روش‌های بیان این مدل، استفاده از نمودارهای دو بخشی $F = (\mathcal{U}, \mathcal{V}, \mathcal{E})$ است که عامل‌ها (\mathcal{U}) با استفاده از یال‌ها (\mathcal{E}) روابط و قیودی را بین گره‌ها (\mathcal{V}) ایجاد می‌کنند. بدین ترتیب بخش اول، یعنی گره‌ها، نمایانگر متغیرهای ناشناخته یا پارامترهای مدل هستند که آنها را با ϕ_i نشان می‌دهیم. به عنوان مثال، در مسئله کالیبراسیون و موقعیت‌یابی، گره‌ها می‌توانند نمایانگر موقعیت‌های مختلف ربات یا پارامترهای کالیبراسیون باشند. همچنین بخش دوم، یعنی عامل‌ها، نشان‌دهنده قیود یا روابط بین متغیرها می‌باشد که آنها را با ψ_i نشان می‌دهیم. این قیود می‌توانند شامل معادلات غیرخطی یا روابط پیچیده‌ای باشند که باید در فرآیند بهینه‌سازی در نظر گرفته شوند. بدین ترتیب یک گراف عامل F بیان‌کننده نحوه ایجاد یک تابع انرژی کلی ψ از تک تک اجزای سیستم می‌باشد:

$$\psi(\phi) = \prod_i \psi_i(\phi_i) \quad (4.3)$$

کمینه‌سازی لگاریتم تابع $\psi(\phi)$ منجر به ایجاد یک مسئله غیرخطی معادل می‌شود که مسئله بهینه‌سازی مورد نظر را مشخص می‌کند.

۲.۴.۳ گراف عامل پیشنهادی برای کالیبراسیون و موقعیت یابی به صورت همزمان

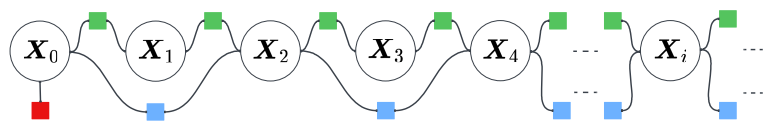
همانطور که بیان شد، استفاده از گراف های عامل در موقعیت یابی ربات ها مزایای متعددی دارد. یکی از مزایای اصلی این است که گراف عامل می تواند به طور مؤثری اطلاعات نامطمئن را مدیریت کرده و تخمین های دقیق تری ارائه دهد. همچنین، این روش به دلیل ساختار گرافی خود، قابلیت انعطاف پذیری و مقیاس پذیری بالایی دارد، به طوری که می توان به راحتی اندازه گیری های جدید را به گراف اضافه کرده و تخمین ها را به روزرسانی کرد. در مسئله ی موقعیت یابی ربات ها، هدف اصلی تخمین دقیق موقعیت و جهت ربات در محیط است. برای این منظور، از اطلاعات مختلفی مانند داده های حسگرها، اندازه گیری های فاصله و داده های اینرسی استفاده می شود. گراف عامل این اطلاعات را به شکلی یکپارچه و هماهنگ ترکیب می کند.

برای مدل سازی مسئله موقعیت یابی با گراف عامل، ابتدا باید متغیرهای حالت ربات و محدودیت های مرتبط با آنها تعریف شوند. متغیرهای حالت می توانند شامل موقعیت و جهت پنجه ربات در نقاط مختلف زمانی نسبت به یک چارچوب پایه مشخص باشند. برای تعریف این متغیرهای حالت در زمان i با کمک ماتریس انتقال پنجه ربات در فضای $SE(3)$ به صورت زیر خواهد بود:

$$X_i = \begin{bmatrix} \mathbf{R}_i & \mathbf{t}_i \\ \mathbf{0} & 1 \end{bmatrix} \quad (5.3)$$

که $X_i^{4 \times 4}$ بیانگر ماتریس انتقال شامل ماتریس چرخش $\mathbf{R}_i^{3 \times 3}$ و بردار انتقال $\mathbf{t}_i^{3 \times 1}$ نسبت به چارچوب پایه می باشد. علاوه بر تعیین چارچوب پایه در حل مسئله موقعیت یابی، محدودیت ها نیز می توانند شامل اندازه گیری های حسگرهای متفاوت باشد. ما در این فرمول بندی حسگر اندازه گیری های اینرسی^۱ و یک حسگر فاصله پیمایی چشمی^۲ را به عنوان داده های اندازه گیری سیستم با فرکانس های متفاوت در نظر می گیریم.

شکل ۲.۳ نشان دهنده گراف عاملی است که برای حل مسئله موقعیت یابی با فرضیات در نظر گرفته شده می توان ارائه کرد. دایره ها نشان دهنده متغیرهای مسئله هستند که در اینجا موقعیت ربات می باشند و آن ها را با ماتریس X_i در زمان i نمایش می دهد. مربع های رنگی نشان دهنده محدودیت ها و داده های حسگری هستند که با گذر زمان به سیستم وارد می شوند و زنجیره موقعیت ربات را نیز امتداد می دهند. چارچوب پایه تعیین شده که می تواند نقطه شروع یا نقطه استراحت ربات باشد، توسط یک عامل واحد^۳ (در اینجا مربع قرمز رنگ)، موقعیت یابی ربات را مقید می کند. همچنین داده های حسگری با توجه به فرکانس داده برداری آن ها می توانند به صورت عامل های دودویی که بین دو یا چند متغیر قیدی را ایجاد می کنند، وارد مسئله شوند. در این گراف،



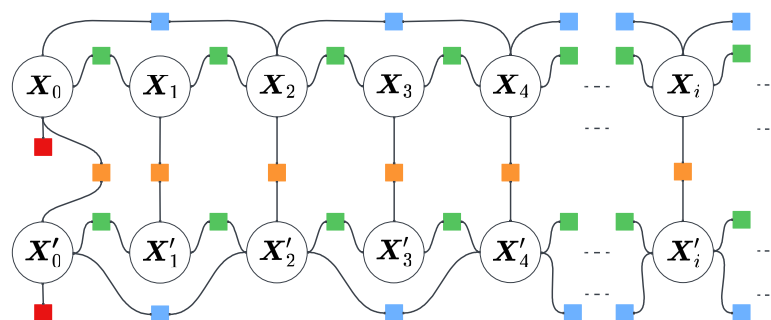
شکل ۲.۳

عامل‌های مشخص شده با رنگ سبز نشان‌دهنده اندازه‌گیری‌های حسگر اینرسی و همچنین اطلاعات حسگر فاصله‌پیمایی چشمی با عامل‌هایی به رنگ آبی وارد مسئله می‌شوند. با توجه به نرخ اضافه شدن این اطلاعات به سیستم، می‌توان دریافت که فرکانس داده‌برداری از حسگر اینرسی دو برابر حسگر فاصله‌پیمایی است.

در این گراف، آنچه باعث حل یکپارچه و استفاده بهینه از تمامی اطلاعات مسئله می‌شود، زنجیره‌ای است که میان متغیرهای مسئله ایجاد شده است. با دریافت هر داده جدید از حسگر، یک متغیر موقعیت جدید برای ربات ایجاد می‌شود که حل مسئله بهینه‌سازی برای پیدا کردن این متغیر منجر به به‌روز رسانی تمامی گره‌ها به صورت یکجا می‌شود و این باعث ایجاد یک راه‌حل ارزشمند برای یک مسئله تنگ می‌گردد. در این بخش، هدف تعریف پایه‌ای یک مسئله موقعیت‌یابی برای ربات است. گراف‌های متنوع و فرمول‌بندی‌شده برای اهداف خاص‌تر در [۱۲]، [۱۳] و [۱۴] قابل مطالعه هستند.

افزودن حسگرهای جدید با ایجاد گره‌های عامل با مدل نویزهای مناسب در فواصل زمانی منظم، می‌تواند نتیجه موقعیت‌یابی را بهبود بخشد. این ترکیب حسگرها می‌تواند در فواصل زمانی‌ای که حسگرها به دلیل شرایط محیطی از سیستم خارج می‌شوند، از توقف موقعیت‌یابی جلوگیری کند. به عنوان مثال، الگوریتم‌هایی که از داده‌های دوربین استفاده می‌کنند، زمانی که ویژگی‌های محیط برای پردازش تصاویر نامناسب است یا ربات وارد محدوده‌ای تاریک می‌شود، قادر به ارائه نتایج مناسبی نیستند. به عنوان نمونه‌ای دیگر، زمانی که سیستم موقعیت‌یابی جهانی^۱ مورد استفاده قرار می‌گیرد، مکان‌هایی همچون تونل‌ها می‌توانند این سیستم جمع‌آوری داده را با مشکل مواجه کنند. بدین ترتیب، با خروج برخی از حسگرها موقعیت‌یابی با استفاده از داده‌های دیگر انجام می‌شود و با ورود مجدد حسگرها، نتایج رو به بهبود خواهند رفت. استفاده از این رویکرد محدود به نوع ربات و یا حسگرهای آن نبوده است. از ربات‌های خودران [۱۵] تا ربات‌های پرنده [۱۶] و یا ربات‌های جراح مورد استفاده در اتاق‌های عمل می‌توانند از این روش استفاده کنند.

استفاده از گراف در مسئله مکان‌یابی برای ربات‌های چندعاملی می‌تواند کلیدی برای بهبود نتایج و فرمول‌بندی ساده‌تر باشد. سامانه آموزش جراحی چشم ARASH:ASiST در تیم آزمایشگاهی ارس توسعه یافته است [۱۷]. این سامانه از دو دستگاه ربات مجزا برای تسهیل فرآیند آموزش جراحی تشکیل شده است. در این سامانه، ربات دوم باید حرکات ربات اول را دنبال کند تا آموزش جراحی با استفاده از این سامانه انجام پذیرد. پیدا کردن چارچوب این ربات‌ها در یک دستگاه مختصات می‌تواند یکی از ابزارهایی باشد که در پیاده‌سازی الگوریتم‌های متنوع کنترلی



شکل ۳.۳

در فرآیند یادگیری مهارت مورد استفاده قرار گیرد. یکی از روش های پیشنهادی برای این هدف می تواند استفاده از گراف شکل ۳.۳ باشد. در این گراف، مکان یابی برای هر کدام از این ربات ها با به روز رسانی X_i و X'_i برای هر یک از ربات ها به صورت مجزا، انجام می شود. هم چارچوب سازی این دو ربات می تواند توسط قیود عاملی که در اینجا با رنگ نارنجی نشان داده شده است، انجام شود. دیگر عامل ها با رنگ های مشخص شده همانند تعاریف بیان شده در شکل ۲.۳ می باشند.

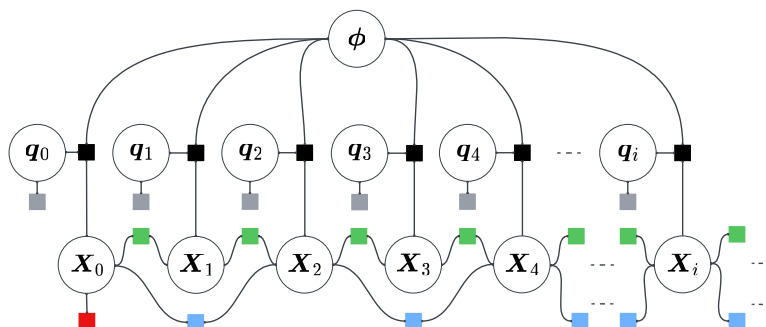
آنچه تا بدین جای حل مسئله مشاهده شده است، توانایی بالای الگوریتم گراف عامل در ایجاد مسئله ای انعطاف پذیر بوده است. مقیدسازی این مسئله می تواند از قیود فضای کار ربات فراتر رفته و مسئله را در فضای مفصلی و ساختار سینماتیک ربات نیز مقید کند. وجود پارامترهای سینماتیکی در فرمول بندی می تواند با در نظر گرفتن آنها به عنوان متغیرهایی که همواره در ربات ثابت هستند یا متغیرهایی که با گذر زمان تغییر می کنند، به عنوان پارامترهای کالیبراسیون، مقادیر بهینه آنها را به دست آورد. به عبارتی، در حالی که مسئله موقعیت یابی در حال انجام است، مسئله کالیبراسیون نیز حل شود. همچنین اضافه شدن این قیود سینماتیکی می تواند موقعیت یابی ربات را بهبود بخشد.

مطابق فرمول بندی های مرسوم ارائه شده، معادلات سینماتیک نگاشت های غیرخطی بین فضای مفصل و فضای کار ربات ایجاد می شوند. بدین ترتیب:

$$X = FK(q, \phi), \quad q = IK(X, \phi) \quad (۶.۳)$$

که X موقعیت ربات در فضای کار و q بردار مقدار زاویه ای/طولی مفصل های ربات هستند. در این معادله ϕ بردار پارامترهای سینماتیکی ربات است که به هندسه ساختاری ربات مربوط می شود.

در ادامه، با فرض آنکه ربات مسیری تصادفی را در فضای کاری خود طی می کند و همچنین داده های سنسوری در هر دو فضای معرفی شده در حال جمع آوری هستند، قصد داریم گراف ۲.۳ را به گونه ای بسط



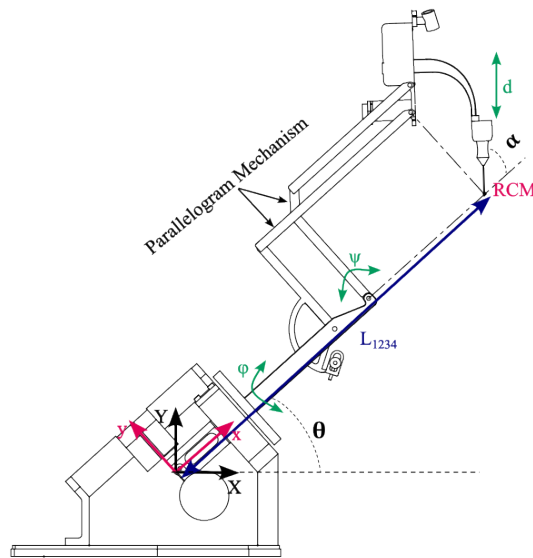
شکل ۴.۳

دهیم که کالیبراسیون سینماتیکی ربات در کنار موقعیتیابی در یک گراف یکپارچه حل شود. برای این کار از قیود سینماتیکی ۶.۳ استفاده کرده و آنها را به صورت عامل به مسئله می افزاییم. ابتدا فرض کالیبراسیون را بر ثابت بودن پارامترهای سینماتیکی و عدم تغییر آنها در طول زمان می گذاریم. شکل ۴.۳ بیانگر یک گراف برای حل این مسئله با این فرض بیان شده می باشد. در این گراف، قسمت موقعیتیابی همانند آنچه پیشتر بیان شد می باشد. همچنین عامل های مشخص شده با مربع های سیاه رنگ بیانگر روابط سینماتیکی ربات هستند که تابعی از مقادیر متغیرهای مفصل، موقعیت ربات در فضای کار و بردار پارامترهای سینماتیکی ربات ϕ می باشند. همچنین عامل های خاکستری رنگ بیانگر مقادیر اندازه گیری شده در فضای مفصل ربات از حسگرهای آن می باشند.

فرمول بندی سینماتیک بیان شده در ۶.۳ می تواند طیف وسیعی از ربات ها را در بر داشته باشد. به عنوان نمونه، برای روشن سازی بهتر مسئله، نمونه موردی ربات ARASH:ASiST را باری دیگر در نظر می گیریم. شکل ۵.۳ شمایی از هندسه با ساختاری متوازی الاضلاع از این ربات را نشان می دهد. پارامترهای سینماتیک این ربات با بردار $\phi = \{\alpha, \theta, l_{1234}\}$ مطابق اندازه های بیان شده بر روی این شکل تعریف می شود. همچنین با توجه به مقادیری که بردار مفاصل ربات $q = \{\phi, \psi, d\}$ دارند، مکان پنجه ربات X در نقطه دوران از راه دور مشخص شده است. با استفاده از این بیان، ساختار سینماتیکی این ربات می تواند به فرمول بندی سینماتیک مستقیم زیر مطابق آنچه در [۱۷] استخراج شده است، منجر شود:

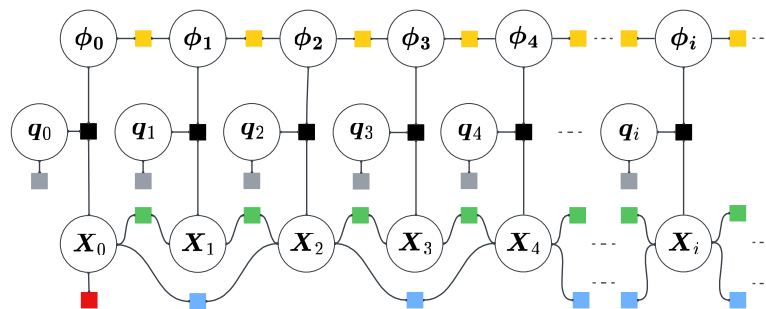
$$\begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} c_{\theta}(l_{1234} + dc_{\alpha+\psi}) - ds_{\theta}c_{\phi}s_{\alpha+\psi} \\ s_{\theta}(l_{1234} + dc_{\alpha+\psi}) + dc_{\theta}c_{\phi}s_{\alpha+\psi} \\ ds_{\phi}s_{\alpha+\psi} \end{pmatrix} \quad (۷.۳)$$

که s و c به ترتیب بیانگر توابع $\sin(\cdot)$ و $\cos(\cdot)$ هستند. تفسیر مکان یابی و کالیبراسیون سینماتیکی بیان شده در گراف ۴.۳، استفاده از معادله ۷.۳ در عامل های بیان شده به عنوان قیود سینماتیکی ربات (عامل های سیاه) است.



شکل ۵.۳

شکل ۶.۳ نشان دهنده گراف قابل استفاده دیگری می باشد که عاری از فرض ثابت بودن پارامترهای کالیبراسیون سینماتیکی بوده و این پارامترها با گذر زمان تغییر می کند و در مسائل زمان واقعی چالش برانگیز هستند. عامل های زرد ایجاد شده در بین پارامترهای کالیبراسیون در زمان های متوالی قیودی هستند که از نوسانات و تغییرات ناگهانی و زیاد این پارامترهای جلوگیری کنند. وجود این قید از آنجایی است که در دنیای واقع انتظار بر تغییر آهسته و منطقی این پارامترهای هندسی می باشد. با این بیان صورت گرفته، افزودن قیود متفاوت به مسئله بدون نیاز به تغییر فرمول بندی دیگر بخش ها قابل انجام خواهد بود. در نهایت، این گراف های عامل می توانند با استفاده از حل کننده های افزایشی که کتابخانه GTSAM در اختیار ما قرار می دهد، به صورت زمان واقعی حل شوند.



شکل ۶.۳

۵.۳ نتیجه‌گیری

در این فصل، رویکرد گراف مبنا برای حل همزمان مسئله کالیبراسیون و موقعیت‌یابی ربات‌ها مورد بررسی قرار گرفت. این روش با استفاده از گراف‌های عامل، امکان مدیریت بهینه داده‌ها و افزایش دقت محاسبات را فراهم می‌آورد و به راحتی قابل گسترش با قیود و حسگرهای جدید است.

ابتدا مروری بر روش‌های مرسوم موقعیت‌یابی و کالیبراسیون انجام شد. این روش‌ها، اگرچه ساده و سریع هستند، اما نیاز به بازتعریف مکرر فرمول‌بندی‌ها و خطی‌سازی‌های پیچیده دارند که باعث کاهش دقت و افزایش پیچیدگی محاسبات می‌شود. سپس، رویکردی جدید، بر مبنای گراف‌ها، برای حل این مسئله با توانایی رفع معایب روش‌های مرسوم، مطرح شد.

در معرفی این رویکرد گراف مبنا، مفاهیم پایه‌ای گراف‌های عامل و کاربرد آنها در مدل‌سازی روابط پیچیده بین متغیرها و قیود بیان شد. سپس، با استفاده از معادلات سینماتیکی ربات و داده‌های حسگری، گراف‌های عامل یکپارچه‌ای برای حل مسئله طراحی گردید. این گراف‌ها با توانایی فرمول‌بندی همزمان کالیبراسیون و موقعیت‌یابی نتایج مطلوبی در بهبود دقت و کارایی ربات را به دست می‌آورند.

فصل ۴

پیاده سازی رویکرد گراف مبنا جهت موقعیت یابی و کالیبراسیون همزمان برای ربات کابلی صلب

در فصل قبل رویکردی بر مبنای ساختار گراف برای انجام مسئله کالیبراسیون و موقعیت یابی ربات به صورت زمان واقعی ارائه شد. ویژگی هایی برای این رویکرد ارائه شد که حل این مسئله پیچیده و پرکاربرد رباتیکی را تسهیل کرده و باعث بهبود نتایج بهینه سازی شده است. در این فصل پیاده سازی کاملی از گراف های معرفی شده بر روی یک ربات خواهیم داشت. سعی بر آن بوده که ربات انتخاب شده برای این پیاده سازی از نظر سینماتیک و دینامیک دارای معادلاتی نسبتاً پیچیده باشد تا قدرت و سرعت این الگوریتم را مورد بررسی کافی قرار دهیم.

۱.۴ انتخاب ربات مناسب جهت توسعه الگوریتم

ربات نمونه انتخاب شده برای ارائه این فرمول بندی، یک ربات چهار کابلی فروتحریک آسان نصب می باشد. علت انتخاب این نوع ربات آسان نصب به عنوان موضوع مورد بررسی، قابلیت استفاده زیاد آنها در کارکردهای متنوع رباتیکی می باشد، به شرطی که هر بار پس از نصب در هر محیط دلخواه فرآیند کالیبراسیون بدون زمان بر و با کمترین زحمت انجام شود. فرمول بندی انجام شده برای این ربات به نحوی است که منجر به یک کالیبراسیون خودکار در کنار موقعیت یابی تنها با همان حسگرهایی که در ربات برای موقعیت یابی تعبیه شده است، بدون زحمت اضافی برای کاربر انجام می شود. نتیجه این رویکرد علاوه بر افزایش دقت نهایی این فرآیندها، مفهومی حقیقی تر به آسان نصب بودن به این دسته از ربات های کابلی می بخشد.

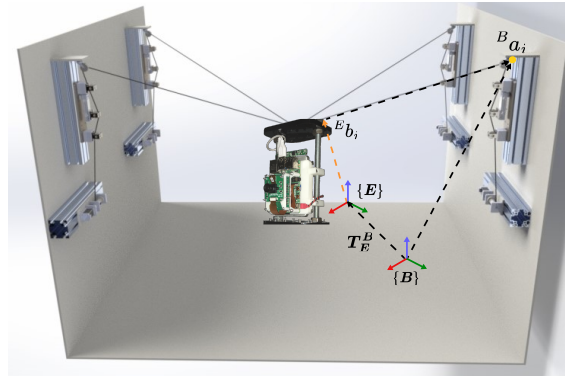
آنچه تا کنون بیشتر مورد بررسی قرار گرفته حل مسئله کالیبراسیون برای ربات هایی است که کابل آنها به عنوان جسمی صلب در نظر گرفته می شود. این فرض اساسی حل مسئله را آسان تر می کند. در ادامه ی این فصل ما با استفاده از رویکرد بیان شده، کالیبراسیون و موقعیت یابی را برای یک ربات کابلی با فرض یاد شده انجام می دهیم و نتایج را مورد بررسی قرار می دهیم. علی رغم اینکه این فرض حل مسئله را ساده تر می کند، زمانی که نسبت جرم پنجه ربات به جرم کابل ها زیاد شود، این فرض قابل قبول نخواهد بود. نشان خواهیم داد که با افزایش این میزان خم شدگی که معمولاً در ربات هایی با ابعاد بزرگتر دیده می شود، کالیبراسیون و موقعیت یابی با مشکل مواجه خواهد شد. راه حل ارائه شده برای حل این مشکل، ایجاد مسئله ای شامل قیدهای دینامیکی پیچیده کابل ها هستند. در این فصل سعی بر ایجاد گرافی خواهیم داشت که مسئله تعریف شده برای ربات کابلی صلب را حل کند. در فصل بعد، با مقیدسازی این گراف، مسئله ای کامل تر را فرمول بندی خواهیم کرد که مشکل مطرح شده را نیز آدرس دهی کند.

۲.۴ توسعه گراف عامل برای یک ربات چهار کابلی با فرض کابل صلب

در این بخش، انجام فرآیند کالیبراسیون و موقعیت یابی به صورت همزمان برای یک ربات کابلی با در نظر گرفتن فرض صلب بودن کابل ها با رویکرد گراف-مبنا انجام می شود. برای این فرآیند، ابتدا ربات چهار کابلی ARASCam معرفی می گردد. سپس فرضیاتی بر روی ساختار ربات ایجاد شده و با در نظر گرفتن این فرضیات، فرمول بندی جامعی برای ربات ارائه می شود. در نهایت، از این فرمول بندی برای پیاده سازی گراف عامل معرفی شده استفاده می شود و نتایج مورد بررسی قرار می گیرد.

۱.۲.۴ معرفی ربات کابلی ARASCam

ربات ARASCam که در گروه تحقیقاتی رباتیک ARAS توسعه یافته است، یک ربات معلق فروتنحریک موازی با کابل های محرک و با شش درجه آزادی می باشد. شکل ۱.۴ نسخه اولیه این ربات که برای یک فضای کاری نسبتاً کوچک آزمایشگاهی طراحی شده است را نشان می دهد. همانطور که در تصویر مشاهده می شود، پنجه ربات که مجهز به یک دوربین تعبیه شده بر روی پردازنده Raspberry PI می باشد، توسط چهار کابل در فضا معلق شده است. B بیانگر دستگاه مختصات پایه و همچنین E نشانه دهنده دستگاه مختصات پنجه ربات می باشد. i امین کابل در نقطه ${}^B b_i$ در مختصات پنجه به ربات و در نقطه ${}^E a_i$ در مختصات پایه به پولی متناظر آن متصل می شود. ماتریس انتقال از دستگاه مختصات پایه به پنجه با T_E^B نشان داده شده است.



شکل ۱.۴: نمایی از ربات چهار کابلی فروتحریک معلق ARAS-CAM

در این ربات، کابل ها با استفاده از یک سیستم مکانیکی درام جمع و باز می شوند. هر یک از کابل ها پس از خروج از درام، توسط مکانیزم مکانیکی از روی سنسور نیرو عبور کرده و سپس به پولی منتقل می شود. در نهایت، از نقطه مشخصی در پولی به ربات متصل می شود. همچنین از آنجایی که در این ربات نسبت جرم پنجه به جرم کابل های ربات مقدار زیادی است، می توان از خم شدگی کابل های آن صرف نظر کرد و کابل های آن را به عنوان اجسام صلب مدل کرد.

۲.۲.۴ بیان فرمول بندی مسئله و فرضیات

برای فرمول بندی مسئله از هندسه بیان شده در شکل ۱.۴ استفاده می کنیم. برای یک ربات با کابل های صلب، مدل اندازه گیری طول کابل \hat{z}_i با استفاده از حسگر انکودر روی ربات برای نمونه k ، به صورت زیر می تواند فرمول بندی شود:

$$l_i^*[k] \triangleq \| \mathbf{R}_E^B \mathbf{b}_i + \mathbf{t}_E^B - \mathbf{a}_i \|^2 \quad (۱.۴)$$

$$\hat{z}_i[k] = l_i^*[k] + l_i^o + w_{\text{enc}}[k]$$

که در آن $(\mathbf{R}_E^B, \mathbf{t}_E^B)$ نشان دهنده ماتریس جهت گیری و بردار انتقال پنجه در دستگاه مختصات پایه، l_i^o نشان دهنده مقدار جابجایی اولیه انکودر، و $w_{\text{enc}}[k]$ نشان دهنده نویز اندازه گیری طول است. اگر انعطاف پذیری کابل نیز در نظر گرفته شود، نیروی T_i در کابل، طول اندازه گیری شده کابل را به صورت زیر تغییر می دهد:

$$\hat{z}_i[k] = \left(1 - \frac{T_i[k] + w_T[k]}{EA} \right) l_i^*[k] + l_i^o + w_{\text{enc}}[k] \quad (۲.۴)$$

که در آن E مدول یانگ کابل، A سطح مقطع کابل، و $w_T[k]$ نويز اندازه گیری نیرو است. چنانچه نیروی کابل صفر باشد، $T_i[k] = 0$ ، در معادله ۲.۴ طول اندازه گیری شده توسط انکودر با فاصله واقعی بین a_i و b_i مطابقت دارد. با این حال، در پیکربندی ای که موتور در محل قفل شده و انکودر مقدار ثابتی را می خواند، جابجایی طول به دلیل کشیدگی توسط انکودر دیده نمی شود اما نسخه مقیاس شده وابسته به نیرو و از فاصله واقعی $l_i^*[k]$ اندازه گیری می شود. با افزایش E (کابل های سفت)، اهمیت این مقیاس بندی وابسته به نیرو به صفر کاهش می یابد و معادله ۲.۴ به معادله ۱.۴ ساده می شود. در مورد ربات با کابل های انعطاف پذیر، فرض می کنیم که حسگرهای نیروی کابل در نزدیکی عملگرها قرار گرفته اند. با این حال، برای بسیاری از کاربردها که به ربات های کابلی با اندازه متوسط نیاز دارند، انعطاف پذیری کابل ممکن است با انتخاب مناسب کابل ها قابل صرف نظر باشد.

علاوه بر فرمول بندی سینماتیک، مطابق شکل ۱.۴ ربات با یک دوربین روی انتهای ربات، متصل به پنجه، تجهیز شده است. در اینجا ما تصاویر دریافتی از این دوربین را با استفاده از الگوریتم SVO [۱۸] برای تولید اندازه گیری های موقعیت ربات در فضای $SE(3)$ به سیستم وارد می کنیم که بدین ترتیب برای موقعیت یابی ربات زنجیره ای از تغییرات مسافت پیمایی با فرمول بندی زیر به هم وصل می شوند:

$$\Delta T_{k-1}^k = \begin{bmatrix} R_{k-1}^k & t_{k-1}^k \\ 0 & s^{-1} \end{bmatrix} \quad (3.4)$$

در طول مرحله کالیبراسیون و موقعیت یابی با استفاده از این الگوریتم، فرض می کنیم محیط دارای نور خوب با بافت های غنی از ویژگی بوده و دوربین به آرامی حرکت می کند. مسئله کالیبراسیون و مکان یابی به تخمین مشترک موقعیت های ربات $T_E^B[k]$ ، موقعیت های نقاط اتصال کابل ها به پولی های متناظر a_i ، طول های اولیه کابل ها l_i^0 ، و مقیاس الگوریتم مسافت پیمایی s با استفاده از اندازه گیری های الگوریتم مسافت پیمایی ΔT_{k-1}^k ، انکودرهای طول کابل $z_i[k]$ ، و برای ربات ها با کابل های انعطاف پذیر، اندازه گیری های نیرو T_i کاهش می یابد. این تصویر از تخمین مشترک به حل یک مسئله بهینه سازی منجر می شود که مدل سینماتیکی را به اندازه گیری های انجام شده نزدیک تر کند. به عبارتی دیگر، فرمول بندی که در ۳.۳ ایجاد شد، در اینجا به فرمول بندی زیر بازنویسی می شود:

$$\min_{a_i, l_i^0, T_E^B[k], s} \sum_k \|h(a_i, l_i^0, T_E^B[k], s) - z_i[k]\|^2 \quad (4.4)$$

که مدل اندازه گیری، $h(\cdot)$ ، به صورت زیر تعریف می شود:

$$h(\mathbf{a}_i, l_i^*, \mathbf{T}_E^B[k], s) = s \left(1 - \frac{T_i[k] + w_T[k]}{EA} \right) l_i^*[k] + l_i^* + w_{\text{enc}}[k] \quad (5.4)$$

با این فرمول بندی، مقادیر بهینه نقاط پولی های ربات \mathbf{a}_i^* و موقعیت های پنجه ربات \mathbf{t}_E^{B*} با پارامتر بزرگنمایی s به دست می آید.

۳.۲.۴ ریت پذیری

برای تحلیل مشاهده پذیری، از روش ارائه شده در [۱۹] الهام گرفته ایم که از سیستم های مکان یابی UWB^۱ برای مقادیر اولیه الگوریتم بهینه سازی خود استفاده کرده است. با توجه به مدل اندازه گیری در معادله ۲.۴، می توان مسئله حداقل مربعات را به صورت خطی زیر بازنویسی کرد:

$$\begin{bmatrix} -2t_x[1]\alpha[1] & \cdots & -2t_x[k]\alpha[k] \\ -2t_y[1]\alpha[1] & \cdots & -2t_y[k]\alpha[k] \\ -2t_z[1]\alpha[1] & \cdots & -2t_z[k]\alpha[k] \\ d_t^*[1]\alpha[1] & \cdots & d_t^*[k]\alpha[k] \\ 2z_i[1] & \cdots & 2z_i[k] \\ \alpha[1] & \cdots & \alpha[k] \end{bmatrix}^T \begin{bmatrix} s^* a_x \\ s^* a_y \\ s^* a_z \\ s^* \\ l_{*i} \\ s^*(d_a^* - l_{*i}^*) \end{bmatrix} = \begin{bmatrix} z_i[1]^2 \\ z_i[2]^2 \\ \vdots \\ z_i[k]^2 \end{bmatrix} \quad (6.4)$$

که در آن d_a ، $d_t[k]$ و $\alpha[k]$ به صورت زیر تعریف می شوند:

$$\alpha[k] \triangleq \left(1 - \frac{\tau_i[k]}{EA} \right) \quad (7.4)$$

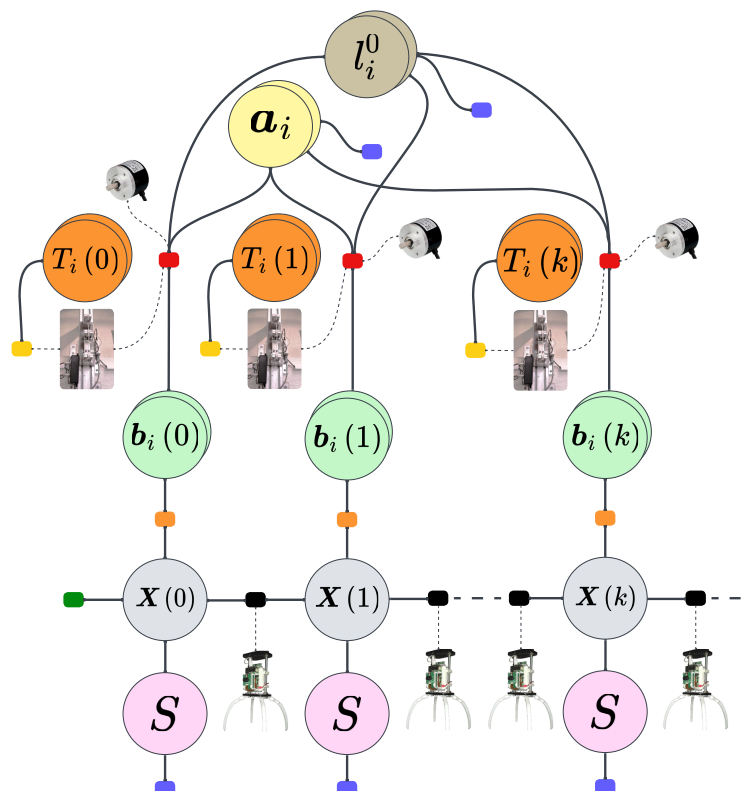
$$d_t^*[k] \triangleq t_x^*[k] + t_y^*[k] + t_z^*[k], \quad d_a^* \triangleq t_x^* + a_y^* + a_z^*$$

همان طور که در [۱۹] اشاره شده است، سیستم خطی در معادله ۶.۴ معمولاً در عمل خوش تعریف^۲ و

منجر به پاسخ های ضعیف خواهد شد. ما به جای این روش معرفی شده برای نقطه شروع، از چارچوب احتمالی استفاده می کنیم تا راه حل های با کیفیت بالاتری به دست آوریم. با این حال، وجود یک پاسخ برای این سیستم خطی، مشاهده پذیری پارامترهای سینماتیکی را با توجه به مجموعه اندازه گیری های تعریف شده در بخش قبلی اثبات می کند.

۳.۴ بهینه سازی پارامترها با استفاده از گراف های عاملی

برای حل مسئله مکان یابی و کالیبراسیون ربات کابلی ارائه شده در شکل ۱.۴ در ساختار گراف، ما یک گراف عامل با گره های متغیر $l_i^0, a_i, T_i(k), b_i(k), X(k), S$ و $T_i(k), S \in \mathbb{R}, l_i^0, X(k) \in SE(3)$ تعریف می کنیم که به ترتیب نمایانگر موقعیت دوربین، انحراف طول کابل (یا طول اولیه کابل)، نیروهای کابل، موقعیت نقاط پولی و نقاط اتصال کابل ها بر روی پنجه ربات هستند. شکل ۲.۴ نشان دهنده این گراف می باشد.



شکل ۲.۴: گراف عامل پیشنهادی برای حل مسئله کالیبراسیون و مکان یابی ربات کابلی صلب

همانطور که در شکل ۲ نشان داده شده است، گره های $X(k)$ با عامل های مسافت پیمایی^۱ مقیاس بندی شده (نشان داده شده با رنگ سیاه) که نشان دهنده موقعیت های نسبی و مقیاس دهی مسیر براساس مقدار گره مقیاس S است، به هم متصل می شوند. از سوی دیگر، هر موقعیت دوربین $X(k)$ به چهار نقطه اتصال کابل های در پنجه ربات $b_i(k)$ از طریق عامل های موقعیت نسبی (نشان داده شده با رنگ نارنجی) که با تبدیل های تعریف شده در مدل CAD پنجه ربات اولیه سازی شده اند، متصل می شود (که مشمول پارامترهای کالیبراسیون در این کار نیستند). هر یک از گره های $b_i(k)$ به نقاط اتصال پولی $a_i(k)$ از طریق عامل های سینماتیکی (نشان داده شده با رنگ قرمز) که طبق بخش ۲.۲.۴ و با استفاده از مدل اندازه گیری ۵.۴ تعریف شده اند، متصل می شوند.

همچنین، این عامل های سینماتیکی نیز به گره های نیروی کابل $T_i(k)$ و پارامتر متغیر طول اولیه کابل l_i متصل هستند. هر یک از این گره های نیرو کابل، توسط عامل های پیشین (نشان داده شده با رنگ زرد) با اندازه گیری های حسگر نیروسنج، محدود شده اند. همچنین، با استفاده از عامل پیشین متصل شده به اولین موقعیت پنجه ربات (نشان داده شده با رنگ سبز)، نقطه صفر مکان یابی ربات را می توان معین کرد. توجه داشته باشید که برای ربات های کابلی که کابل های با کیفیت بالا توسعه پیدا کرده اند، نیاز به اندازه گیری نیروی کابل ممکن است حذف شود و گره های $T_i(k)$ از گراف عاملی حذف شوند. در این حالت، عوامل سینماتیکی باید براساس معادله ۱.۴ تعریف شوند.

در نهایت، با انتخاب مقادیر مناسبی برای میانگین و ماتریس های گوسی مقادیر اولیه عامل های متصل به گره های مقیاس دهی (در بخش ۲.۴.۴ روشی برای این انتخاب مناسب ارائه خواهد شد)، محل پولی ها و مقادیر اولیه طول کابل ها (نشان داده شده با رنگ آبی) که به عنوان پارامترهای کالیبراسیون نقش بازی می کنند، گراف نهایی را تشکیل می دهیم. با استفاده از یک حل کننده مناسب می توان تابع بهینه احتمالاتی نهایی را بهینه کرد. مقادیر گره های سینماتیکی بهینه شده و عدم قطعیت های به حاشیه رفته مربوطه خروجی نهایی چارچوب ما هستند.

۴.۴ نتایج پیاده سازی

این بخش به ارزیابی کاربردی بودن گراف پیشنهادی در تخمین مشترک پارامترهای سینماتیک و موقعیت های پنجه ربات، مستقل از دستگاه های اندازه گیری خارجی یا مقادیر اولیه پارامترها می پردازد. علاوه بر این، مزایای ترکیب بینایی-سینماتیک^۲ نیز در این بخش مورد بررسی قرار می گیرد.

۱.۴.۴ راه اندازی سیستم و فرضیات

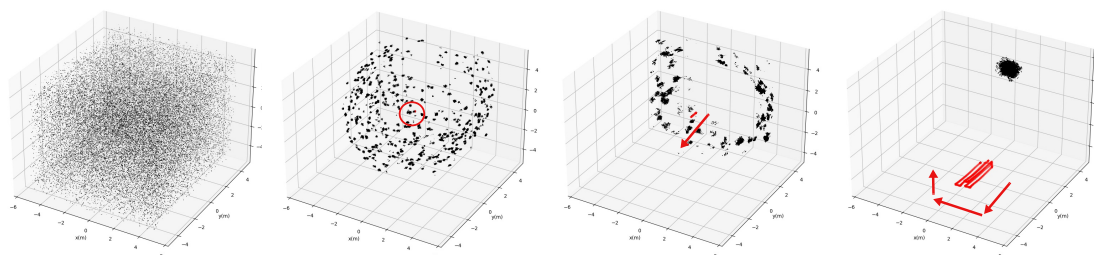
ما آزمایشات خود را با استفاده از ربات، ARAS-CAM یک ربات کابلی با اندازه متوسط و مساحت کاری $7 \times 3 \times 4$ مترمکعب که توسط گروه رباتیک ARAS در دانشگاه صنعتی خواجه نصیرالدین طوسی توسعه داده شده است، انجام می دهیم. این ربات، به یک مجموعه کامل از حسگرهای سینماتیک و بینایی-اینرسی^۱ مجهز شده است تا تحقیقات برآورد حالت برای ربات های کابلی را ممکن سازد. به طور خاص، هر واحد عملگر کابلی به یک حسگر نیرو با وضوح 25° نیوتن و یک انکودر اندازه گیری طول نسبی با دقت ۱ میلی متر مجهز شده است. از سوی دیگر، پنجه ربات دارای یک سیستم تعبیه شده با دوربین تک چشمی OV9281^۲ و واحد اندازه گیری اینرسی BMI088 است. برای پیش برد اهداف تحقیقات، ۱۲ رشته داده با اندازه گیری های داده مرجع سه بعدی با استفاده از سیستم ردیابی موقعیتی توسعه یافته در این آزمایشگاه، از این ربات با عنوان مجموعه داده ARAS-CAM ضبط کرده ایم. برای پیاده سازی های انجام شده، ما از رشته داده ۰۱ این مجموعه به عنوان نماینده حرکات پویاتر و از رشته داده ۱۲ به عنوان یک همتای هموارتر استفاده می کنیم.

به دلیل اندازه متوسط ARAS-CAM و انتخاب کابل های کولار بسیار سخت، تقریب کابل صلب که پیش تر مورد بحث قرار گرفت، در تمام آزمایشات ما حفظ می شود. علاوه بر این، فرض می کنیم که تقریب طول کابل اولیه در محدوده ± 50 سانتی متر مقدار واقعی قرار دارند که در عمل با آزاد کردن کابل ها از یک پیکربندی اولیه شناخته شده قابل دستیابی است.

برای تولید اندازه گیری های بینایی حرکتی، از الگوریتم SVO [۱۸] استفاده شده است که از نظر محاسباتی کارآمد بوده و به طور گسترده در کاربردهایی از جمله پرواز خودران تا واقعیت مجازی مورد استفاده قرار می گیرد. ما از اندازه گیری های مرجع برای اندازه گیری عدم قطعیت بینایی حرکتی الگوریتم SVO در پیاده سازی خود استفاده می کنیم.

۲.۴.۴ کالیبراسیون خودکار بدون پارامترهای اولیه

برای راه اندازی گراف فرمول بندی شده، باید حدس نسبتا مناسبی از مقادیر اولیه پارامترهای شناسایی و ماتریس های کوواریانس نویز در دست داشته باشیم. روشی که برای دستیابی به این موارد معرفی می کنیم، الگوریتمی مبتنی بر روش مونت-کارلو می باشند. این الگوریتم ونحوه کارکرد آن به صورت مفصل در [۲۰] توضیح داده شده است. در این بخش، ما از این الگوریتم ابتدایی مونت-کارلو برای شناسایی مکان های نقاط پولی، مقادیر اولیه طول کابل، و مقیاس بینایی حرکتی استفاده می کنیم. پس از شروع کارکرد، SVO کاربر یک چارچوب مختصات



شکل ۳.۴: پالایش متوالی یک باور یکنواخت بر روی پارامترهای سینماتیکی از طریق الگوریتم مونت-کارلو

صفر اختیاری با قرار دادن انتهای ربات در آن حالت و شروع الگوریتم انتخاب می‌کند. سپس، انتهای ربات بر اساس جهت عدم قطعیت بیشتر یا با دنبال کردن هنجار حرکت انتهای ربات در امتداد سه محور انتقالی و پوشش حجم بزرگ حرکت می‌کند.

همانطور که در شکل ۳.۴ نشان داده شده است، در ابتدا وقتی اطلاعاتی به فیلتر تزریق نمی‌شود، ذرات به طور یکنواخت در منطقه پخش می‌شوند (اولین نمودار از چپ). اما، همانطور که در شکل (نمودار دوم از چپ) نشان داده شده است، پس از کمی حرکت دادن دوربین در طول یک خط مستقیم، توزیع یکنواخت به یک پوسته کروی تبدیل می‌شود که تمامی مکان‌های ممکن برای مکان پولی را نشان می‌دهد و ضخامت آن عدم قطعیت بر طول اولیه کابل را نشان می‌دهد. به تدریج، با حرکت طولانی‌تر انتهای ربات و در امتداد جهت‌های دیگر، کره به یک حلقه تبدیل می‌شود همانطور که در شکل (نمودار سوم از چپ) نشان داده شده است. در نهایت به یک خوشه بیضوی از ذرات که مکان شناسایی شده نقطه پولی و عدم قطعیت مربوطه آن را نشان می‌دهد تبدیل می‌شود که در شکل (نمودار چهارم از چپ) نشان داده شده است.

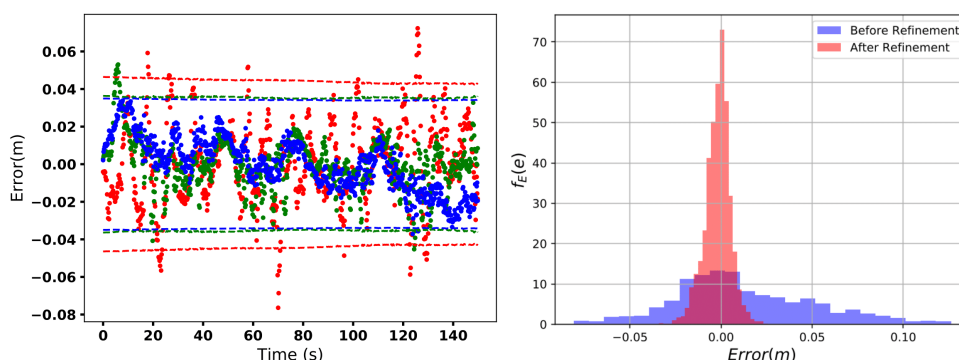
۳.۴.۴ ترکیب بینایی-سینماتیک و استفاده از گراف عامل معرفی شده

در این بخش، ما گراف عامل را که در بخش ۳.۴ بیان شده است، پیاده‌سازی می‌کنیم تا پارامترهای مراحل قبل را با در نظر گرفتن تأثیر متقابل تمام حالت‌ها بهبود دهیم. ما این گراف عامل را با استفاده از کتابخانه GT-SAM [۲۱] که یک حل‌کننده معروف در جامعه SLAM است و حالت‌های اجرای افق متحرک و افزایشی بسیار کارآمدی دارد، پیاده‌سازی می‌کنیم. گراف‌های عامل این گراف به صورت دستی در کلاس‌های C++ نوشته شده و به حل‌کننده این کتابخانه متصل شده‌است.

مزایای ترکیب سینماتیک و حسگر بینایی در بهبود نتایج نقش موثری داشته است. جدول ۱۰.۴ خلاصه‌ای از این نتیجه را در هر حالت ارائه می‌دهد. همانطور که انتظار می‌رود، دقت کلی با ترکیب این دو حالت بهبود می‌یابد. به ویژه برای رشته داده ۵۱، ترکیب بینایی-سینماتیک به بهبود دقت ۲۳ درصدی در مقایسه با سینماتیک مستقیم و

جدول ۱.۴: خطای مکان یابی با روش های مختلف (واحد متر)

رشته	خطای MSE سینماتیک مستقیم	خطای MSE بینایی حرکتی	خطای MSE روش پیشنهادی
۰۱	۰/۰۳۶	۰/۰۵	۰/۰۲۹
۱۲	۰/۰۳۵	۰/۰۳	۰/۰۲۸
میانگین	۰/۰۳۵۵	۰/۰۴	۰/۰۲۸۵



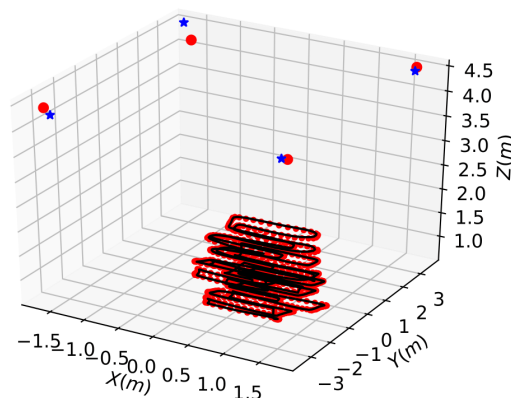
شکل ۴.۴: راست: توزیع خطای پنجه قبل و بعد از بهبود پارامترها، چپ: خطای مکان یابی در دستگاه کارترین

۷۲ درصدی در مقایسه با بینایی حرکتی منجر می شود. این اعداد به ترتیب برای رشته داده ۱۲ برابر با ۲۵ درصد و ۷ درصد است. بهبود عملکرد در رشته داده ۰۱ بیشتر است، که با این واقعیت همخوانی دارد که انتهای ربات در این رشته داده حرکات پویاتری دارد و در نتیجه بینایی حرکتی در معرض تخریب بیشتری قرار می گیرد. از سوی دیگر، این حرکات پویا نوسانات بزرگی ایجاد می کنند که از حالت های سینماتیک قابل مشاهده نیستند و منجر به خطاهای نسبتاً بزرگی می شوند که در [۲۲] نیز نشان داده شده است.

در ادامه، کاهش خطای پنجه ربات را پس از بهبود پارامترهای اولیه بررسی می کنیم. این خطا را به عنوان میانگین مربع خطا بین طول کابل محاسبه شده از معادله ۱.۴ و مقادیر داده های انکودر از ربات تعریف می کنیم. ما از این خطا به عنوان یک معیار جانشین برای مکان های نقاط پولی استفاده می کنیم زیرا در پیاده سازی ما، قرقره ها در خارج از میدان دید سیستم ردیابی ما (سیستم داده های مرجع^۱) قرار دارند و بنابراین مقادیر واقعی متناظر در نسخه فعلی مجموعه داده در دسترس نیستند.

شکل ۴.۴ (نمودار سمت راست) توزیع خطای پنجه ربات را قبل و بعد از بهبود پارامترها نشان می دهد. همان طور که دیده می شود، میانگین و واریانس این خطا به طور قابل توجهی پس از بهبود کاهش یافته است. به طور خاص، برای رشته داده ۰۱، مقدار خطا قبل از بهبود ۰/۰۴۳ متر و پس از بهبود ۰/۰۰۴۹ متر است که نشان دهنده یک کاهش مرتبه ای در خطا است. این مقادیر برای رشته داده ۱۲ به ترتیب ۰/۰۳ متر قبل و ۰/۰۰۵۳ متر

^۱truth ground



شکل ۵.۴: مسیر طی شده توسط ربات در کنار مکان اولیه و نهایی (بهیودیافته) پولی ها

متر بعد از بهبود هستند. همچنین شکل ۴.۴ (نمودار سمت چپ) نشان دهنده خطای مکان یابی پنجه ربات در دستگاه کارترین در واحد متر می باشد. این خطا بیانگر اختلاف مقادیر مکان پنجه در سه جهت از حل گراف عامل نسبت به مقادیر داده های مرجع می باشد.

در نهایت، نتایج کیفی اجرای الگوریتم ما بر روی رشته داده ۰۱ در شکل ۵.۴ نشان داده شده است. در این شکل، مسیر سیاه رنگ نمایانگر داده های مرجع و نقاط قرمز نشان دهنده موقعیت های اصلاح شده ربات توسط الگوریتم ما هستند. علاوه بر این، ستاره های آبی در شکل مکان های پولی قبل از بهبود و دایره های قرمز، آن ها را پس از بهینه سازی مشترک گراف عامل را نشان می دهند.

۴.۴.۴ انتشار عدم قطعیت

یکی از مزایای چارچوب پیشنهادی، توانایی آن در حفظ و در نظر گرفتن عدم قطعیت و نویز حسگر در طول فرآیند تخمین است. در پیاده سازی های ما، از مقادیر میانگین و ماتریس های کواریانس از برای راه اندازی الگوریتم با مقید کردن گراف عامل با استفاده از عامل های پیشین استفاده می کنیم. نتیجه این کار شباهت عدم قطعیت نهایی گراف عامل به توزیع خطای واقعی است. ما این سازگاری را با تعیین درصد مقادیر خطا در محدوده های σ ، 2σ و 3σ بررسی می کنیم.

همانطور که در جدول ۲.۴ نشان داده شده است، تقریباً ۶۰ درصد از خطاها در محدوده σ ، ۹۰ درصد در محدوده 2σ و تقریباً ۹۸ درصد در محدوده 3σ قرار دارند. نتایج جدول ۲.۴ نزدیک به محدوده های تئوریک σ هستند اما نشان دهنده یک پاسخ کمی بیش از حد مطمئن است. باید توجه داشت که سازگاری به شدت به مدل های حسگر مفروض و همچنین عدم قطعیت بینایی حرکتی تجربی برای الگوریتم SVO وابسته است. ما

جدول ۲.۴: سازگاری آماری عدم قطعیت های تخمین زده شده

محور	σ	2σ	3σ
x	۵۳٪	۸۲٪	۹۵٪
y	۷۱٪	۹۲٪	۹۷٪
z	۵۱٪	۸۸٪	۹۹٪

بررسی دقیق تر سازگاری عدم قطعیت را به کارهای آینده موکول می کنیم.

۵.۴.۴ بحث و گفت وگو

روش پیشنهادی در این فصل یک دیدگاه منعطف و یکپارچه در مورد کالیبراسیون و برآورد حالت برای ربات های کابلی صلب ارائه می دهد. این یکپارچگی در جامعه SLAM به شکل اجرای موازی تنظیم مجموعه ^۱ و ردیابی دوربین ^۲ به خوبی شناخته شده است. به ویژه برای عملکرد بلادرنگ ^۳، مرحله گراف عامل الگوریتم ما ممکن است به صورت افق متحرک حل شود و حالت های قدیمی به فاکتورهای پیشین حاشیه سازی شوند. از سوی دیگر، کالیبراسیون اولیه ممکن است بدون محدودیت بلادرنگ و با استفاده از اجرای دسته ای گراف روی رشته داده طولانی تری انجام شود.

پیاده سازی فعلی ما در Python روی یک لپ تاپ شخصی با پردازنده Core-i۷ Intel و ۱۶ گیگابایت RAM کمتر از یک دقیقه برای هر کابل زمان می برد تا روش ابتدایی مونت-کارلو را اجرا کند (برای ۷۵۰ موقعیت دوربین و ۵۰۰۰ ذره) و حدود ۳۰ ثانیه زمان برای بهبود آن ها توسط گراف عامل. حالت های اجرای افق متحرک حل کننده های بسیار کارآمد برای گراف های عامل مانند [۲۱] و [۲۳] از جامعه SLAM می توانند فرکانس برآورد موقعیت را به نرخ فریم دوربین افزایش دهند که برای بیشتر کنترل کننده های حلقه خارجی مناسب است. ما بررسی بیشتر عملکرد بلادرنگ الگوریتم خود را به کارهای آینده موکول می کنیم.

یکی دیگر از جنبه های مهم الگوریتم پیشنهادی، ماژولار بودن آن است. به ویژه، این فرمول بندی اجازه می دهد تا دیگر حالت ها از جمله اما نه محدود به، UWB IMU و حسگرهای سیستم به راحتی به گراف عامل اضافه شوند. علاوه بر این، اندازه گیری های شبه هندسی مانند هم سطحی نقاط پولی (به عنوان مثال دو پولی در همان دیوار) یا فاصله بین مکان های پولی (قابل اندازه گیری با استفاده از ابزارهای اندازه گیری کم هزینه) به راحتی در مسئله گنجانده می شوند که به طور قابل توجهی به دقت کلی تخمین کمک می کند.

در نهایت، سیستم پیشنهادی ممکن است به طور مستقیم برای کالیبراسیون سیستم های موقعیت یابی UWB استفاده شود. در این سیستم ها، مدل اندازه گیری فاصله بین تگ و انکرها دقیقاً با مدل کابل صلب در نظر گرفته شده در این فصل یکسان است. ما قصد داریم این موضوع را به عنوان کارهای آینده بیشتر بررسی کنیم.

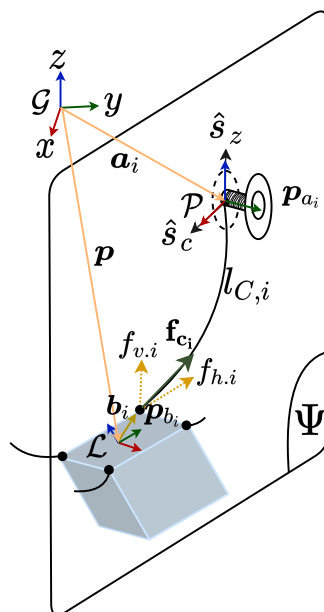
فصل ۵

پیاده سازی رویکرد گراف مبنا جهت موقعیت یابی و کالیبراسیون همزمان برای ربات کابلی خم شده

این پایان نامه به بررسی دقیق رویکردهای مختلف در حوزه رباتیک برای فرمول بندی ریاضی یک مسئله بهینه سازی مقید پرداخته است. از تحلیل مزایا و معایب روش های مرسوم تا رویکردهای جدید مبتنی بر گراف، روشی جامع برای فرمول بندی و حل مسئله بهینه سازی کالیبراسیون و مکان یابی همزمان ربات ها ارائه شده است. در فصل گذشته، عملکرد این روش بر روی یک ربات کابلی مقید که کابل های آن به صورت صلب در نظر گرفته شده بود، ارزیابی شد. پس از معرفی فرمول بندی سینماتیکی، گراف مربوط به آن ساخته و نتایج پیاده سازی بررسی شد. آنچه از ابتدای این پایان نامه به عنوان هدفی مهم معرفی گردید، ماژولاریتی و انعطاف پذیری روش، با قیدهای متفاوت و گسترده بود. انتخاب ربات کابلی به عنوان مورد مورد مطالعه، نیز به دلیل امکان پیاده سازی و ارزیابی همین هدف بوده است.

در این فصل، بدون تغییر در فرمول بندی های ارائه شده در فصل قبل، به مسئله قیدهای دینامیکی کابل افزوده خواهد شد. علیرغم پیچیدگی این مدل ها، حل کننده همچنان با دقت و سرعت بالا به نتایج مطلوب دست خواهد یافت. تاکنون تحقیقات بسیاری بر مدل سازی کابل های خم شده انجام شده است که نتایج دقیقی به دست داده اند. ما نیز برای حل مسئله کالیبراسیون و موقعیت یابی نیازمند افزودن این قیدها به مسئله هستیم. با این حال، پیچیدگی های این مدل ها باعث شده است که در برخی کارهای اخیر به جای حل مستقیم مسئله با این معادلات، از شبکه های عمیق استفاده شود که به دلیل مشکلات خاص خود، دقت و اطمینان کافی ندارند.

روش ما برای حل این چالش، استفاده از همان مقیدسازی هایی است که برای کابل های صلب انجام شده بود. در پایان، با حل این مسئله، مزایای این رویکرد را بار دیگر خواهیم دید؛ رویکردی که با دقت و قدرت بالا،



شکل ۱.۵: دیاگرام پنجه ربات متصل به یک کابل خم شده

مسئله کالیبراسیون و موقعیت یابی همزمان ربات ها را، حتی در شرایطی که کابل ها صلب نیستند، به نحوی که حل آنها در روش های مرسوم دشوار است، به سرانجام می رساند.

۱.۵ نمادها و تعاریف

این فصل یک ربات موازی کابلی معلق با شش درجه آزادی ($m = 6$) و چهار کابل فعال ($n = 4$) را مورد بررسی قرار می دهد. از آنجایی که $m > n$ ، این ربات فروتحریک^۱ است و یک ساختار نامقید^۲ تشکیل می دهد [۲۴]. شکل ۱.۵ ساختار این ربات را نشان می دهد که برای وضوح بیشتر تنها یک کابل در آن نمایش داده شده است. دستگاه مختصات \mathcal{L} به بدنه متحرک ربات متصل است، در حالی که دستگاه مختصات جهانی \mathcal{G} به طور ثابت به پایه ربات متصل شده است. مکان پنجه ربات نسبت به دستگاه مختصات جهانی با $(p, R) \in SE(3)$ نشان داده می شود، که در آن $p \in \mathbb{R}^3$ بردار انتقال از \mathcal{G} به \mathcal{L} است و $R \in SO(3)$ جهت گیری \mathcal{L} نسبت به \mathcal{G} است. کابل i از پایه ربات در نقطه p_{a_i} که در دستگاه مختصات جهانی تعریف شده است، جدا می شود و به پنجه ربات در نقطه p_{b_i} که در دستگاه مختصات بدنه محلی بیان شده است، متصل می شود.

ما تغییر شکل کابل را در یک صفحه عمودی دو بعدی Ψ مدل سازی می کنیم که پولی p_{a_i} و نقطه اتصال p_{b_i}

^۱ underactuated ^۲ under-constrained

در پنجه ربات را در بر می گیرد. دستگاه مختصات \mathcal{P} روی این صفحه در نقطه p_{a_i} قرار دارد و با بردارهای واحد \hat{s}_z که موازی با محور z جهانی است و \hat{s}_c که در جهت کابل بر روی صفحه $x_{\mathcal{P}} - y_{\mathcal{P}}$ دستگاه مختصات \mathcal{P} قرار دارد، تعریف می شود. به طور خاص، $\hat{s}_c = \frac{b_{xy_i} - a_{xy_i}}{\|b_{xy_i} - a_{xy_i}\|}$ ، که در آن b_{xy_i} و a_{xy_i} به ترتیب اجزای $x - y$ بردارهای b_i و a_i در دستگاه مختصات جهانی هستند.

۲.۵ معادلات مدل کابل خم شده

معادلات زنجیره ای اثر خم شدن کابل غیرقابل ارتجاع با جرم غیر قابل اغماض را همانطور که در [۲۵] توصیف شده است، به صورت زیر است:

$$z_i(x_c) = \frac{f_{h,i}}{g_c} \cdot \left(\cosh \left(\frac{g_c}{f_{h,i}} \cdot (x_c + C_{\backslash,i}) \right) - C_{\Upsilon,i} \right) \quad (۱.۵)$$

در این معادله، شکل خم شدن کابل با تابع $z_i(x_c)$ تعریف شده است. ثابت های زنجیره ای، $C_{\backslash,i}$ و $C_{\Upsilon,i}$ با توجه به شرایط مرزی نقطه انتهایی $(p_a)_z = z_i(\circ) = -\frac{f_v}{f_h}$ و $z'_i(L_i) = -\frac{f_v}{f_h}$ تعیین می شوند، که در آن $z'_i(L_i)$ شیب معادله (۱.۵) در $x_c = L_i = \|(b)_{xy_i} - (a)_{xy_i}\|$ است و به صورت زیر دارای حل بسته هستند:

$$C_{\backslash,i} = \frac{f_{h,i}}{g_c} \cdot \operatorname{asinh} \left(\frac{-f_{v,i}}{f_{h,i}} \right) - L_i \quad (۲.۵)$$

$$C_{\Upsilon,i} = \cosh \left(C_{\backslash,i} \cdot \frac{g_c}{f_{h,i}} \right) - \frac{g_c}{f_{h,i}} \cdot (p_a)_z \quad (۳.۵)$$

همانطور که در شکل ۱.۵ نشان داده شده است، $f_{h,i}$ و $f_{v,i}$ اجزای افقی و عمودی نیروی کابل f_{c_i} در دستگاه مختصات \mathcal{P} هستند. علاوه بر این، $g_c = g \cdot \rho_c$ است که در آن g و ρ_c به ترتیب شتاب گرانشی و جرم در واحد طول کابل هستند. در نهایت، طول منحنی معادله (۱.۵) به عنوان طول واقعی کابل $l_{C,i}$ تعریف می شود و به صورت زیر محاسبه می شود:

$$l_{C,i} = \int_0^{L_i} \sqrt{1 + \left(\frac{dz}{dx}\right)^2} dx \quad (4.5)$$

$$= \frac{f_{h,i}}{g_c} \cdot \left(\sinh \left(\frac{g_c}{f_{h,i}} (L_i + C_{v,i}) \right) - \sinh \left(\frac{g_c}{f_{h,i}} \cdot C_{v,i} \right) \right)$$

۳.۵ سینماتیک ربات

تحلیل سینماتیکی یک ربات موازی کابلی فروتحریک شامل هر دو معادلات هندسی و حالت ایستای آن است، که به طور معروف به تحلیل سینماتیک-ایستا معروف است [۲۶]. نیروی پیچشی^۱ پنجه ربات $\mathbf{w}_{ee} \in \mathbb{R}^6$ به نیروهای کابل $\mathbf{f}_c \in \mathbb{R}^4$ از طریق ماتریس ژاکوبی $\mathbf{J} \in \mathbb{R}^{4 \times 6}$ مرتبط می شود:

$$\mathbf{w}_{ee} = \mathbf{J}^T \mathbf{f}_c \quad (5.5)$$

در این اینجا، فرض می کنیم که \mathbf{w}_{ee} تنها توسط گرانش ایجاد شده است و به صورت زیر محاسبه می شود:

$$\mathbf{w}_{ee} = m_e g \begin{bmatrix} \hat{\mathbf{s}}_z \\ \mathbf{b}_{com} \times \hat{\mathbf{s}}_z \end{bmatrix} \quad (6.5)$$

که در آن m_e جرم پنجه ربات و \mathbf{b}_{com} جابجایی بین مبدا دستگاه مختصات \mathcal{L} و مرکز جرم (CoM) انتهای ربات است. برای پیوند دادن این معادلات حالت ایستا با مدل زنجیره ای در معادله (۴.۵)، هر جزء نیروی کابل به عنوان یک جفت افقی و عمودی نمایش داده می شود:

$$\mathbf{f}_{c_i} = \begin{bmatrix} f_{h,i} & f_{v,i} \end{bmatrix}^T \quad (7.5)$$

برای هر f_{c_i} ستون i مربوطه از ماتریس ژاکوبی J^T به صورت زیر بیان می شود:

$$J_i^T = \begin{bmatrix} -\hat{s}_{c,i} & \hat{s}_z \\ -Rb_i \times \hat{s}_{c,i} & Rb_i \times \hat{s}_z \end{bmatrix} \quad (۸.۵)$$

که در آن، ماتریس چرخش R ، بردارهای واحد $\hat{s}_{c,i}$ ، \hat{s}_z و بردار اتصال پنجه b_i در مختصات محلی، در بخش ۱.۵ تعریف شده اند. توجه داشته باشید که معادلات حالت ایستا در ۵.۵ یک مسئله نامعین^۱ است که در آن تعداد معادلات کمتر از تعداد متغیرها است. همانطور که در [۲۲] پیشنهاد شده است، ما تمام نیروهای کابل را بر اساس یک کابل مرجع بیان می کنیم. همانطور که در بخش ۴.۵ مشاهده خواهد شد، این انتخاب، تعداد حسگرهای نیرو مورد نیاز را به تنها یک عدد کاهش می دهد که برای ما نیز از اهمیت بالایی برخوردار است. همانطور که در [۲۲، ۲۷] ارائه شده است، با تقسیم ماتریس ژاکوبین و بردار نیرو، می توان ۵.۵ را به صورت زیر نوشت:

$$w_{ee} = \begin{bmatrix} J_{ref}^T & J_{res}^T \end{bmatrix} \cdot \begin{bmatrix} f_{c_{ref}} \\ f_{c_{res}} \end{bmatrix} \quad (۹.۵)$$

که نتیجه می دهد:

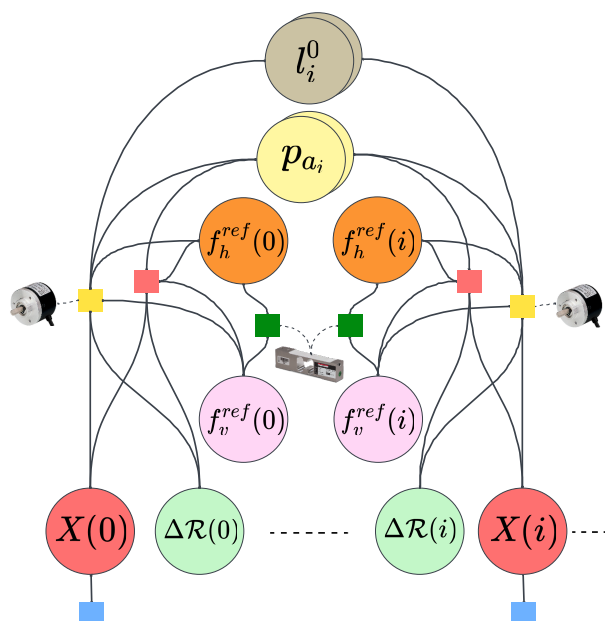
$$w_{ee} = J_{ref}^T \cdot f_{c_{ref}} + J_{res}^T \cdot f_{c_{res}} \quad (۱۰.۵)$$

که در آن، J_{ref}^T نمایانگر یک زیرماتریس ۶×۲ شامل دو ستون اول J^T است که مربوط به نیروی کابل مرجع $f_{c_{ref}}$ است. به دنبال آن، J_{res}^T به عنوان زیرماتریس باقی مانده ۶×۶ تعریف می شود و $f_{c_{res}}$ نمایانگر نیروهای کابل باقی مانده است. ما می توانیم $f_{c_{res}}$ را در معادله (۱۰.۵) به صورت زیر بازنویسی کنیم:

$$f_{c_{res}} = (J_{res}^T)^{-1} (w_{ee} - J_{ref}^T \cdot f_{c_{ref}}) \quad (۱۱.۵)$$

با این فرمول بندی، تعداد متغیرها از ۸ به ۲ کاهش می یابد، در حالی که معادلات حالت ایستا به طور ضمنی در معادله (۵.۵) نهفته می شود و نیاز افزودن یک قید جداگانه را حذف می کند [۲۷].

^۱underdetermined



شکل ۲.۵: گراف عامل کالیبراسیون و مکان یابی همزمان ربات کابلی با کابل های خم شده

۴.۵ گراف عامل کالیبراسیون و مکان یابی همزمان سینماتیک-ایستا

در این بخش، با استفاده از روابط استخراج شده در قسمت قبل، و همچنین فرمول بندی سینماتیکی تعریف شده در فصل پیشین، رویکردی با یک فرمول بندی یکپارچه ایجاد می شود. رویکرد ما یک گراف عامل با گره های متغیر $(f_h^{ref}(k), f_v^{ref}(k) \in \mathbb{R}, \Delta R(k) \in SO(3), p_{a_i} \in \mathbb{R}^3, l_i^0, X(k) \in SE(3))$ تعریف می کند. این گره ها به ترتیب، نمایانگر موقعیت های پنجه ربات، مقادیر اولیه طول کابل ها، مکان های نقاط پولی، و تغییرات در جهت گیری ربات، و همچنین نیروی کابل مرجع در محورهای افقی و عمودی در حالت های ایستا ربات هستند. در ساختار فروتحریک ما، همه ترکیب های مکانی و جهت گیری قابل اجرا نیستند. در اینجا، $\Delta R(k)$ متغیری است که مقدار اولیه چرخش پنجه ربات را تغییر می دهد. علاوه بر این، کابل مرجع به عنوان کابلی که کاربر حسگر نیرو برای مقاصد کالیبراسیون بر روی آن تعبیه شده است، تعیین می شود.

شکل ۲.۵ ساختار این گراف عامل در راستای کالیبراسیون خودکار و همچنین مکان یابی همزمان برای ساختار تعریف شده را که برای دو نمونه از وضعیت های ایستا ۰ و i نمایش داده شده است، نشان می دهد. متغیرهای بهینه سازی در این گراف با دایره های برچسب خورده با نام های پارامترها در رنگ های مختلف نمایان شده اند. علاوه بر این، عامل ها با مربع های رنگی به تصویر کشیده شده اند، که عامل انکودر و یا همان طول کابل خم شده به رنگ زرد، عامل مکان اتصال کابل به پولی به رنگ قرمز، عامل اندازه گیری نیرو به رنگ سبز، و عامل پیشین موقعیت

به رنگ آبی است. هر عامل بر اساس معادلات سینماتیک و مدل ریاضی تعریف شده برای کابل توصیف شده در بخش ۲.۵ فرمول بندی شده و به شرح زیر تعریف می شوند:

۱.۴.۵ عامل طول کابل خم شده

این عامل رابطه ای بین اندازه گیری های انکودر و طول واقعی کابل از معادله ۴.۵ ایجاد می کند. این قید اندازه گیری برای کابل i به صورت زیر فرمول بندی می شود:

$$f(z_i^{enc}, \zeta)[k] = l_{C,i}[k] + l_i^* - z_i^{enc}[k] \quad (12.5)$$

که در آن $l_{C,i}$ نمایانگر طول واقعی کابل در اثر خم شدگی از نیروی وزن آن به طوریکه در معادله ۴.۵ تعریف شده است، می باشد. همچنین l_i^* نشان دهنده مقدار طول اولیه کابل است، و z_i^{enc} نمایانگر اندازه گیری نسبی انکودر مربوط به کابل i است. علاوه بر این، k بیان کننده شاخص نمونه داده های زمانی است.

۲.۴.۵ عامل مکان اتصال کابل به پولی

این عامل تضمین می کند که ارتفاع محاسبه شده نقطه اتصال کابل روی پنجه ربات، که از مکان پنجه ربات استنتاج شده، با ارتفاع کابل خم شده پیش بینی شده از پولی مربوطه مطابقت داشته باشد. تابع خطا برای پولی i به صورت زیر بیان می شود:

$$f(\zeta)[k] = (p_{b,i})_z[k] - z_i(L_i)[k] \quad (13.5)$$

که در آن، $(p_{b,i})_z$ به ارتفاع نقطه اتصال کابل i در مختصات جهانی اشاره دارد، و $z_i(L_i)$ نمایانگر شکل خم شدگی کابل در L_i به طوریکه در معادله ۱.۵ توصیف شده است.

۳.۴.۵ عامل اندازه گیری نیرو

این عامل نرم نیروی افقی و عمودی را به گونه ای محدود می کند که نزدیک به اندازه گیری نیرو از حسگر تعبیه شده روی کابل مرجع، در نزدیکی پنجه ربات باشد. توجه داشته باشید که این محدودیت تنها برای کابل

مرجع با یک تابع هزینه به صورت زیر مورد نیاز است:

$$f(f^m, \zeta)[k] = \|[f_h^{ref}[k] \ f_v^{ref}[k]]^T\| - f^m[k] \quad (14.5)$$

در اینجا، f_h^{ref} و f_v^{ref} به ترتیب نمایانگر نیروهای مرجع کابل در جهت های افقی و عمودی هستند، $\|\cdot\|$ نشان دهنده نرم اقلیدسی است، و f^m مقدار نیروی کابل مرجع است که توسط حسگر نیرو اندازه گیری شده است.

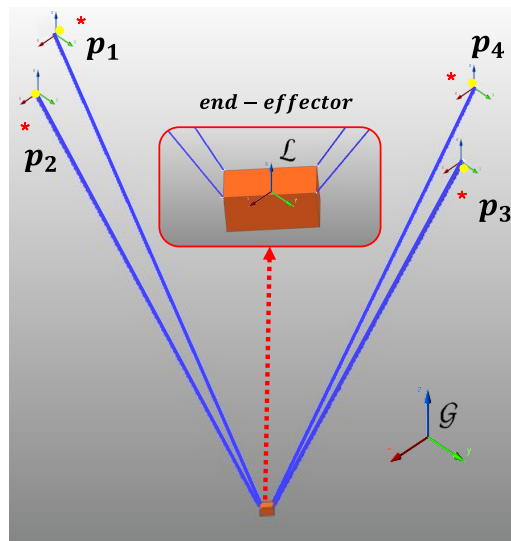
۴.۴.۵ عامل پیشین مکان

عوامل انکودر و اندازه گیری نیرو با گره های متغیر مکان $X(k)$ مرتبط هستند که نمایانگر وضعیت های ایستا ربات در فرآیند کالیبراسیون به طوریکه توسط یک سیستم محلی سازی مبتنی بر بینایی اندازه گیری شده است. هر موقعیت به حالات تعادلی مربوط می شود که در آن ربات از طریق چهار کابل خود ثابت است. نمونه هایی از وضعیت های ایستا در شکل ۲.۵ با نشانگرهای 0 و i برچسب گذاری شده اند. این عامل پیشین مکان نیز برای تعریف صفر ربات مورد استفاده قرار می گیرد.

۵.۵ نتایج شبیه سازی

این بخش به منظور اعتبارسنجی مدل و روش کالیبراسیون پیشنهادی از طریق شبیه سازی اجزای محدود^۱ سیستم ارائه شده است. ابتدا اعتبار فرمول بندی های سینماتیک-ایستا بررسی می شود و سپس نتایج کالیبراسیون برای دو ربات کابلی کوچک مقیاس و بزرگ مقیاس نشان داده می شود. برای شبیه سازی های اجزای محدود از نرم افزار RecurDyn [۲۸] استفاده خواهیم کرد، مدل گراف عامل خود را با استفاده از کتابخانه GTSAM [۲۱] پیاده سازی می کنیم و همچنین از SymForce [۲۹] برای استخراج مشتق عامل ها و ژاکوبین های مربوطه استفاده می کنیم.

^۱element finite



شکل ۳.۵: سناریوی ربات کوچک مقیاس در محیط شبیه ساز RecurDyn

۱.۵.۵ صحت سنجی مدل

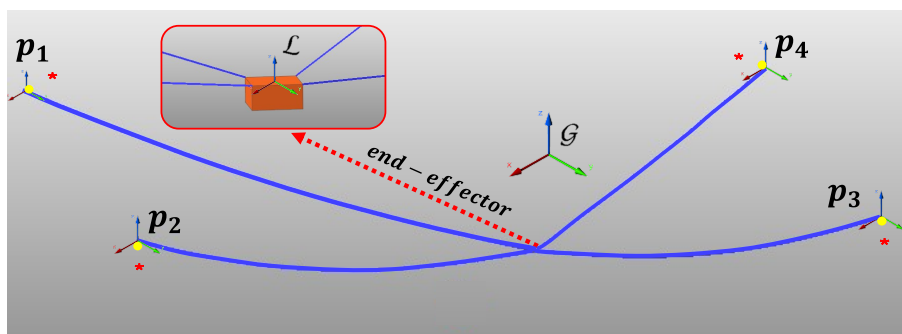
برای صحت سنجی دقت مدل سینماتیک-ایستا، همان طور که در بخش ۲.۵ توضیح داده شده است، گراف عامل سینماتیک-ایستا توسعه داده شده را با استفاده از عامل های پیشین بر روی متغیرهای استخراج شده از شبیه ساز مقید می کنیم. دو سناریوی ربات کابلی معلق کوچک و بزرگ مقیاس برای صحت سنجی این مدل انجام شده است. هدف از طراحی این دو سناریوی مجزا، بررسی دقت الگوریتم برای طیف وسیعی از پیاده سازی ها می باشد. شکل ۳.۵ و شکل ۴.۵ این سناریوهای ربات ها را در محیط شبیه ساز RecurDyn نشان می دهند. هر دو ربات چهار کابل را به چهار گوشه بالای یک جعبه مستطیلی متصل کرده اند. همچنین ابعاد ربات که از فواصل بین پولی ها که در شکل مشخص شده اند به دست آمده، (۲۸/۵، ۴/۵، ۱۲/۵) متر برای ربات کوچک و (۵۰، ۲۲۰، ۲۴۰) متر برای ربات بزرگ تر در نظر گرفته شده است. همان طور که در جدول ۱.۵ ذکر شده است، جرم پنجه برای ربات بزرگ ۳۴Kg و برای ربات کوچک ۴/۴Kg تنظیم شده است. چگالی طول کابل ها برای ربات کوچک ۱۰/۲g/m و برای ربات بزرگ تر ۷۲/۴g/m است، که به نسبت جرم پنجه ربات به کابل ۴/۱۴ برای ربات کوچک و ۰/۶۲ برای ربات بزرگ تر نتیجه می دهد. همان طور که در [۲۵] پیشنهاد شده است، این شرایط نشان دهنده تأثیر قابل توجه خم شدگی برای ربات بزرگ تر است.

دو ردیف اول جدول ۱.۵ درصد خطاهای میانگین در طول کابل پیش بینی شده (MPE-L) و همچنین این خطا برای نیرو (MPE-F) محاسبه شده در ۵ موقعیت ایستای تصادفی را ارائه می دهد. این مقادیر نشان دهنده تطابق نزدیک بین نیروی کابل و مقادیر طول کابل خم شده محاسبه شده از مدل ریاضی ارائه شده و مقادیر مربوطه

جدول ۱.۵: صحت سنجی مدل

کوچک مقیاس	بزرگ مقیاس	سناریو
$7/3595 \times 10^{-3}$	$8/8947 \times 10^{-3}$	MPE-L (%)
۰/۹۰۴۶	۰/۹۱۵۴	MPE-F (%)
$[10/8, 25/9]$	$[134/7, 200/7]$	$[l_{min}, l_{max}](m)$
$[12/4, 26/2]$	$[489/1, 954/3]$	$[f_{min}, f_{max}](N)$
$12/5 \times 4/5 \times 28/5$	$240 \times 220 \times 50$	اندازه ربات (m ^۳)
۴/۴	۳۴/۰	جرم پنجه ربات (Kg)
$\rho = 1/44 \text{ (g/cm}^3\text{)}$	$\rho = 1/44 \text{ (g/cm}^3\text{)}$	مشخصات کابل
شعاع = ۱/۵ (mm)	شعاع = ۴/۰ (mm)	

از شبیه ساز می باشد. به طور خاص، برای ربات بزرگ مقیاس، این خطاهای MPE-L و MPE-F به ترتیب برابر با ۰/۰۰۸۹٪ و ۰/۹۱۵۴٪ ارزیابی شده است، که بسیار کوچک تر از دامنه های طول کابل و نیروهای ایجاد شده برای هر کابل، مطابق مقادیر گزارش شده در جدول، می باشد. همان طور که به طور شهودی انتظار می رود، این تطابق برای ربات کابلی کوچک تر دقیق تر است. مقادیر مربوط به طول کابل و نیروهای پیش بینی شده در ستون سوم این جدول نشان دهنده این موضوع می باشند.



شکل ۴.۵: سناریوی ربات بزرگ مقیاس در محیط شبیه ساز RecurDyn

۲.۵.۵ نتایج نهایی کالیبراسیون با گراف عامل توسعه داده شده

در قسمت، روش کالیبراسیون را که در بخش ۴.۵ توضیح داده شده است، پیاده سازی می کنیم و اهمیت مدل سازی خم شدگی کابل را از طریق نتایج شبیه سازی نشان می دهیم. علاوه بر این، به طور خلاصه به مسئله مقداردهی اولیه کالیبراسیون پرداخته و یک راه حل احتمالی را همچون فصل قبل، پیشنهاد می کنیم.

هدف ما در فرآیند کالیبراسیون سینماتیکی، تعیین مکان های نقطه های پولی و طول اولیه کابل ها با استفاده از اندازه گیری های مجموعه ای از مکان های پنجه ربات، اندازه گیری های طول نسبی کابل و تنها مقادیر کشش کابل مرجع در نقطه اتصال پنجه است. طبق آزمایشات و نتایجی که از طیف وسیعی از داده ها استخراج شده است، توزیع نمونه ها بایستی به طور جامع فضای کاری ربات را پوشش دهد. تحلیل و توسعه مسیر مناسب برای جمع آوری کمترین تعداد داده در جهت انجام فرآیند کالیبراسیون موفق، به موضوع تحقیق آینده برای تکمیل این کار محول می شود.

ما داده های حسگر شبیه سازی شده خود را از نرم افزار RecurDyn دریافت کردیم و تغییرات نویز گاوسی میانگین صفر را برای وارد کردن نویز پیش بینی شده حسگر معرفی کردیم. به طور خاص، ما انحراف استاندارد^۱ ۱۰mm برای طول کابل، ۵N برای حسگر نیرو در ربات بزرگ مقیاس و ۱N برای حسگر ربات کوچک مقیاس اعمال کردیم. مکان های پنجه ربات با $\Delta T = \exp(\xi)$ که در آن $\xi \in \mathfrak{se}(3)$ عنصر جبر لی^۲ متناظر با بردار پیچش^۳ $\xi \in \mathbb{R}^6$ است، دچار تغییر شده اند. این بردارهای پیچش تغییرات، از توزیع گاوسی میانگین صفر $\xi \sim \mathcal{N}(0, \Sigma)$ با ماتریس کوواریانس Σ با انحراف ۰/۰۰۵ متر در درجات انتقالی و ۰/۰۱ رادیان در درجات آزادی چرخشی نمونه برداری شده اند. برای مقداردهی اولیه گراف عامل، مکان های نقطه های پولی از شبیه ساز به عنوان داده های مرجع را با دامنه ۱ متر برای ربات کوچک و ۱۰ متر برای ربات بزرگ دچار تغییر کردیم. علاوه بر این، این تغییرات برای مقادیر اولیه طول کابل ها، ۱۰ متر و ۸۰ متر برای ربات کوچک و بزرگ، به ترتیب، تنظیم شد. ما تخمین اولیه برای نیروی کابل مرجع در محور عمودی را به عنوان:

$$f_{v_s}^{ref} = \frac{m_e g}{4} \quad (15.5)$$

در نظر گرفتیم و برای محور افقی نیز به صورت زیر:

$$f_{h_s}^{ref} = \frac{f_{v_s}^{ref}}{\tan(\alpha)} \quad (16.5)$$

جدول ۲.۵: نتایج میانگین خطای مطلق کالیبراسیون با استفاده از گراف عامل توسعه داده شده

میانگین خطا (متر)	مقیاس بزرگ (۹ داده)	مقیاس بزرگ (۱۸ داده)	مقیاس بزرگ (۳۵ داده)	مقیاس کوچک (۸ داده)
پولی (متر)	۰/۳۸۷	۰/۲۲۶	۰/۱۹۱	۰/۱۴۹
طول اولیه کابل (متر)	۰/۳۸۰	۰/۲۱۷	۰/۱۷۳	۰/۱۱۵

که در آن α به صورت زیر تعریف می شود:

$$\alpha = \arccos \left(\frac{\hat{s}_{cref}^T \cdot [p_{bref} - p_{aref}]}{\|p_{bref} - p_{aref}\|} \right) \quad (17.5)$$

در این تعریف، \hat{s}_{cref} بردار واحدی است که در بخش ۲.۵ تعریف شده است و از تخمین های اولیه موقعیت پولی استفاده می کند. این نقاط باید به پولی های مربوط به کابل مرجع متقابل مربوط شوند.

نتایج کالیبراسیون برای مکان های پولی و طول اولیه کابل در جدول ۲.۵ ارائه شده است. این نتایج مربوط به سناریوهای نشان داده شده در شکل ۳.۵ و شکل ۴.۵ است، که در آن مکان های اولیه پولی با ستاره های قرمز و مکان های بهبود یافته پولی با دایره های زرد برای هر دو مورد نشان داده شده است. در این جدول، سه سناریوی مختلف کالیبراسیون برای ربات بزرگ مقیاس با تعداد مختلف نقاط نمونه برداری شده (یا همان تعداد داده ها برای کالیبراسیون) برای فرآیند بهینه سازی گزارش شده است. همان طور که انتظار می رود، دقت کالیبراسیون با افزایش تعداد داده های نمونه برداری بهبود می یابد. بنابراین جدول ۲.۵ نتایج دقیق تری را با ۳۵ نمونه داده برای ربات بزرگ و ۸ نمونه برای ربات کوچک ارائه می دهد. ما معتقدیم که برای ربات بزرگ تر، پارامترهای مربوط به خم شدگی کابل تأثیر عمیق تری بر دقت کالیبراسیون دارند. این موضوع به نوبه خود تعداد مؤثر پارامترهای مدل را افزایش می دهد و نیاز به نقاط نمونه برداری بیشتری برای شناسایی مناسب دارد.

حل شدن مسئله بهینه سازی کالیبراسیون در کنار زنجیره ای از پارامترهای مکان یابی که به مدل کابل، داده های حسگری پنجه ربات و وضعیت مفصل های ربات مقید شده است، حلی دقیق تر از مسئله را برای ما فراهم می کند. حرکت ربات در فضا می تواند منجر به ایجاد وضعیت های ایستا شود که در هر کدام از این وضعیت ها، قیدهای مربوط به سینماتیک به زنجیره مکان یابی متصل می شود. وجود تعدادی محدود از این قیود می تواند نقش مهمی در بهبود مکان ربات پس از مدتی حرکت در فضا را ایجاد کند و از خطاهای جمع شونده که از نویز حسگرها القا می شود، جلوگیری شود. عملکرد این قید همچون نشانگرهایی که در SLAM مورد استفاده قرار می گیرند، در اینجا نیز بسیار کارآمد هستند. از آنجایی که هدف ما در این فصل توسعه پایه و اساس الگوریتم مد نظر و

مقیدسازی آن با قیدهای پیچیده تر بود، از ایجاد پیچیدگی بیشتر در قسمت مکان یابی اجتناب گردید. افزودن قیود مکان یابی و توسعه بیشتر الگوریتم در این راستا، کار دشواری نخواهد بود.

موضوع دیگری که در فرمول بندی تعریف شده مورد توجه قرار دارد، صادق بودن معادلات طول خم شده کابل در حالت ایستای ربات است. به همین دلیل، داده هایی که برای کالیبراسیون مورد استفاده قرار داده شد، داده های حالت های ایستای ربات بودند. البته، حل مسئله در فضایی جامع تر، و نه محدود به داده های ایستا، نیازمند تغییر فرمول بندی به معادلات در فضای اجزای محدود می باشد. این توسعه و تحقیق به کارهای آینده سپرده شده است.

۶.۵ بحث

۱.۶.۵ اهمیت در نظر گرفتن اثر خم شدگی کابل

برای بررسی اهمیت در نظر گرفتن اثر خم شدگی کابل، فرآیند کالیبراسیون را برای ربات بزرگ تر با استفاده از گراف عامل ساده سازی شده که مدل کابل بی وزن صلب را که در فصل پیشین معرفی شد استفاده می شود. به عبارتی دیگر، گراف عامل مربوط به خم شدگی کابل را از گراف توسعه یافته در این فصل حذف می شود. در نتیجه، به دلیل این ساده سازی، خطای میانگین مطلق به طور قابل توجهی از 0.19 به $0.34/2$ متر افزایش یافته است که نشان دهنده کاهش قابل توجهی در دقت کالیبراسیون است. این تفاوت قابل توجه در کیفیت کالیبراسیون اهمیت حیاتی در نظر گرفتن خم شدگی کابل را تاکید می کند.

۲.۶.۵ نکات مربوط به روش مقداردهی اولیه

همان طور که در [۲۰] ذکر شده است، یکی از نگرانی های اصلی در حل مسئله بهینه سازی کالیبراسیون غیر مقعر و غیر خطی، مقداردهی اولیه صحیح آن است. اگر مقادیر اولیه به اندازه کافی نزدیک به راه حل جهانی مسئله نباشند، نتیجه ممکن است به شدت منحرف شود یا بهینه ساز حتی ممکن است واگرا شود. چارچوب ارائه شده در [۲۰] که در فصل پیشین نیز مورد استفاده قرار گرفت، تلاش می کند تا این مسئله مقداردهی اولیه را با استفاده از خروجی تقریبی یک الگوریتم بهینه سازی جهانی مونت کارلو حل کند. با این حال، [۲۰] مدل کابل صلب بدون اثرات خم شدگی را فرض می کند. ما معتقدیم که این الگوریتم به طور مستقیم می تواند برای مقداردهی اولیه مسئله کالیبراسیون توسعه یافته ارائه شده در این فصل به کار رود. همان طور که قبلاً ذکر شد، دقت کالیبراسیون ربات کابل بزرگ با مدل کابل ساده شده $0.34/2$ متر بود. این مقدار به طور قابل توجهی کوچکتر

از تغییرات ما در طول آزمایشات کالیبراسیون (۱۰ متر) است. این نشان می دهد که ما می توانیم با خیال راحت الگوریتم بهینه سازی خود را با خروجی های الگوریتم مشابهی که در [۲۰] ارائه شده است، مقداردهی اولیه کنیم. تأیید این فرضیه برای موارد خاص ارائه شده در این فصل به دلیل محدودیت های شبیه ساز مورد استفاده برای تولید تصاویر/داده های حسگر LiDAR مورد نیاز برای اجرای این الگوریتم امکان پذیر نبود. بررسی این ایده با استفاده از شبیه سازهای واقع گرایانه موضوع تحقیق آینده ی ما است.

فصل ۶

نتیجه‌گیری و پیشنهادات برای آینده

باید بیان کنیم به چه چیزایی رسیدیم و تلفیق‌های ما برای چه چیزی مناسب است. و چرا جذاب و خوب و مناسب و مفید است. اینها باید با زبان نتیجه‌گیری بیان شوند. نباید بازگویی کنیم مثل یک ربات.

مراجع

- [1] Hall, David L and Llinas, James. An introduction to multisensor data fusion. *Proceedings of the IEEE*, 85(1):6–23, 1997.
- [2] Elatta, AY, Gen, Li Pei, Zhi, Fan Liang, Daoyuan, Yu, and Fei, Luo. An overview of robot calibration. *Information Technology Journal*, 3(1):74–78, 2004.
- [3] Idá, Edoardo, Merlet, Jean-Pierre, and Carricato, Marco. Automatic self-calibration of suspended under-actuated cable-driven parallel robot using incremental measurements. in *Cable-Driven Parallel Robots: Proceedings of the 4th International Conference on Cable-Driven Parallel Robots 4*, pp. 333–344. Springer, 2019.
- [4] Idà, Edoardo, Briot, Sébastien, and Carricato, Marco. Identification of the inertial parameters of underactuated cable-driven parallel robots. *Mechanism and Machine Theory*, 167:104504, 2022.
- [5] Ida, Edoardo. Dynamics of undeactuated cable-driven parallel robots. 2021.
- [6] Chang, Lubin, Li, Kailong, and Hu, Baiqing. Huber’s m-estimation-based process uncertainty robust filter for integrated ins/gps. *IEEE Sensors Journal*, 15(6):3367–3374, 2015.
- [7] Dellaert, Frank, Kaess, Michael, et al. Factor graphs for robot perception. *Foundations and Trends® in Robotics*, 6(1-2):1–139, 2017.
- [8] Ahmad, Aamir, Tipaldi, Gian Diego, Lima, Pedro, and Burgard, Wolfram. Cooperative robot localization and target tracking based on least squares minimization. in *2013 IEEE International Conference on Robotics and Automation*, pp. 5696–5701. IEEE, 2013.
- [9] Guibas, Leonidas J, Motwani, Rajeev, and Raghavan, Prabhakar. The robot localization problem. *SIAM Journal on Computing*, 26(4):1120–1138, 1997.
- [10] Aragues, Rosario, Carlone, Luca, Calafiore, G, and Sagues, C. Multi-agent localization from noisy relative pose measurements. in *2011 IEEE International Conference on Robotics and Automation*, pp. 364–369. IEEE, 2011.

- [11] Panigrahi, Prabin Kumar and Bisoy, Sukant Kishoro. Localization strategies for autonomous mobile robots: A review. *Journal of King Saud University-Computer and Information Sciences*, 34(8):6019–6039, 2022.
- [12] Yang, Lyuxiao, Wu, Nan, Li, Bin, Yuan, Weijie, and Hanzo, Lajos. Indoor localization based on factor graphs: A unified framework. *IEEE Internet of Things Journal*, 10(5):4353–4366, 2022.
- [13] Song, Yang and Hsu, Li-Ta. Tightly coupled integrated navigation system via factor graph for uav indoor localization. *Aerospace Science and Technology*, 108:106370, 2021.
- [14] Leitinger, Erik, Meyer, Florian, Tufvesson, Fredrik, and Witrisal, Klaus. Factor graph based simultaneous localization and mapping using multipath channel information. in *2017 IEEE International Conference on Communications Workshops (ICC Workshops)*, pp. 652–658. IEEE, 2017.
- [15] Wilbers, Daniel, Merfels, Christian, and Stachniss, Cyrill. Localization with sliding window factor graphs on third-party maps for automated driving. in *2019 International conference on robotics and automation (ICRA)*, pp. 5951–5957. IEEE, 2019.
- [16] Dai, Jun, Liu, Songlin, Hao, Xiangyang, Ren, Zongbin, and Yang, Xiao. Uav localization algorithm based on factor graph optimization in complex scenes. *Sensors*, 22(15):5862, 2022.
- [17] Hassani, A, Dindarloo, MR, Khorambakht, R, Bataleblu, A, Sadeghi, H, Heidari, R, Iranfar, A, Hasani, P, Hojati, NS, Khorasani, A, et al. Kinematic and dynamic analysis of arash asist: Toward micro positioning. in *2021 9th RSI International Conference on Robotics and Mechatronics (ICRoM)*, pp. 59–65. IEEE, 2021.
- [18] Forster, Christian, Pizzoli, Matia, and Scaramuzza, Davide. SVO: Fast semi-direct monocular visual odometry. in *IEEE International Conference on Robotics and Automation (ICRA)*, 2014.
- [19] Blueml, Julian, Fornasier, Alessandro, and Weiss, Stephan. Bias compensated uwb anchor initialization using information-theoretic supported triangulation points. in *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 5490–5496. IEEE, 2021.
- [20] Khorrambakht, Rooholla, Damirchi, Hamed, Dindarloo, MR, Saki, A, Khalilpour, SA, Taghirad, Hamid D, and Weiss, Stephan. Graph-based visual-kinematic fusion and monte carlo initialization for fast-deployable cable-driven robots. in *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 1832–1839. IEEE, 2023.

- [21] Dellaert, Frank. Factor graphs and gtsam: A hands-on introduction. *Georgia Institute of Technology, Tech. Rep*, 2:4, 2012.
- [22] Allak, Eren, Khorrambakht, Rooholla, Brommer, Christian, and Weiss, Stephan. Kinematics-inertial fusion for localization of a 4-cable underactuated suspended robot considering cable sag. in *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 4989–4996. IEEE, 2022.
- [23] Martiros, Hayk, Miller, Aaron, Bucki, Nathan, Solliday, Bradley, Kennedy, Ryan, Zhu, Jack, Dang, Tung, Pattison, Dominic, Zheng, Harrison, Tomic, Teo, et al. Symforce: Symbolic computation and code generation for robotics. *arXiv preprint arXiv:2204.07889*, 2022.
- [24] Idà, Edoardo, Briot, Sébastien, and Carricato, Marco. Natural oscillations of underactuated cable-driven parallel robots. *IEEE Access*, 9:71660–71672, 2021.
- [25] Pott, Andreas and Bruckmann, Tobias. *Cable-driven parallel robots*. Springer, 2013.
- [26] Carricato, Marco. Direct geometrico-static problem of underconstrained cable-driven parallel robots with three cables. *Journal of Mechanisms and Robotics*, 5(3):031008, 2013.
- [27] Borgstrom, Per Henrik, Jordan, Brett L, Borgstrom, Bengt J, Stealey, Michael J, Sukhatme, Gaurav S, Batalin, Maxim A, and Kaiser, William J. Nims-pl: A cable-driven robot with self-calibration capabilities. *IEEE Transactions on Robotics*, 25(5):1005–1015, 2009.
- [28] FunctionBay, Inc. Recurdyn: Multi-body dynamics cae software. <https://functionbay.com/en/page/single/2/recurdyn-overview>, 2023.
- [29] Martiros, Hayk, Miller, Aaron, Bucki, Nathan, Solliday, Bradley, Kennedy, Ryan, Zhu, Jack, Dang, Tung, Pattison, Dominic, Zheng, Harrison, Tomic, Teo, Henry, Peter, Cross, Gareth, VanderMey, Josiah, Sun, Alvin, Wang, Samuel, and Holtz, Kristen. SymForce: Symbolic Computation and Code Generation for Robotics. in *Proceedings of Robotics: Science and Systems*, 2022.

پیوست آ

آشنایی سریع با برخی دستورات لاتک

پیوست ب

جدول، نمودار و الگوریتم در لاتک

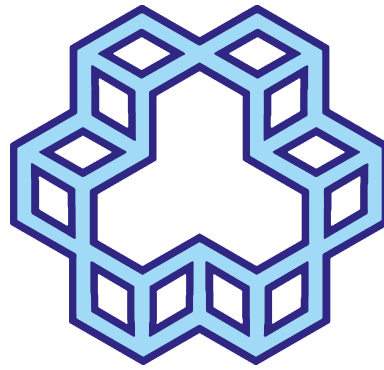
پیوست پ

مراجع، واژه‌نامه و حاشیه‌نویسی

Abstract

This thesis studies on writing projects, theses and dissertations using kntu-thesis class.
It ...

Keywords Writing Thesis, Template, \LaTeX , \Xy Persian



K. N. Toosi University of Technology
Faculty of ...- ... Group

Thesis Submitted in Partial Fulfillment of the
Requirements for the Degree of Master of Science (M.Sc.)
in ... Engineering

Prepared template for writing projects, theses, and dissertations of K. N. Toosi university of technology

By:
Mohammad Sina Allahkaram

Supervisors:
First Supervisor and Second Supervisor

Advisors:
First Advisor and Second Advisor

Winter 2023