بسم‌الله الرحمن الرحیم

دانشگاه صنعتی اصفهان

دانشکده مهندسی مکانیک

توسعه‌ی روشFast-SLAM در مکان‌یابی و نقشه‌یابی همزمان ربات سیار

پایان‌نامه کارشناسی ارشد مهندسی مکانیک طراحی کاربردی

سهیل قراتپه

اساتید راهنما

دکتر مهدی کشمیری دکتر حمیدرضا تقی‌راد

۱۳۹۲

دانشگاه صنعتی اصفهان

دانشکده مهندسی مکانیک

پایان‌نامه کارشناسی ارشد مهندسی مکانیک – طراحی کاربردی آقای سهیل قراتپه

تحت عنوان

توسعه‌ی روشFast-SLAM در مکان‌یابی و نقشه‌یابی همزمان ربات سیار

در تاریخ ..... توسط کمیته تخصصی زیر مورد بررسی و تصویب نهایی قرار گرفت.

۱. استاد راهنمای پایان‌نامه دکتر .....

۲. استاد راهنمای پایان‌نامه دکتر .....

۳. استاد مشاور پایان‌نامه دکتر .....

۴. استاد داور دکتر .....

۵. استاد داور دکتر .....

سرپرست تحصیلات تکمیلی دانشکده دکتر .....

کلیه حقوق مادی مترتب بر نتایج مطالعات، ابتکارات

و نوآوری‌های ناشی از تحقیق موضوع این پایان-

نامه متعلق به دانشگاه صنعتی اصفهان است.

تقدیم به پدر و مادر عزیز و همسر مهربانم

تشکر و قدردانی

در هر دوره‌ای از زندگی انسان، عده‌ای به زندگی او می‌آیند و عده‌ای از زندگی او، خواهند رفت. اما در این میان کسانی نیز هستند که نه به روزمره‌گی‌های او، بلکه به عمق وجود رخنه می‌کنند. این کسان، شاید روزی از روزمره‌گی‌های ما رخت بندند، اما جای حضورشان در عمق وجود باقیست. اینان پیام آوران حضور، و نه حتی وجود خداوند، برروی زمین‌اند.

قدردانی از این انسانها کاریست بی‌معنا.آنها وجودشان با عشق سرشته شده و تنها تاثیرشان در دنیای خارج، لاجرم عشق ورزیدن است.

به هر حال، بردن نام این عزیزان، نه آنکه به منزله‌ی جبران و تشکر از زحماتشان باشد، بلکه آرامیست بر دل این حقیر.

در ابتدا بوسه‌ای خواهم زد بر دستان مهربان پدر و مادرم که در تمام دوران تحصیل، همواره مشوق و حامی من بودند.

سر تعظیم فرود می‌آورم بر عشق بی‌کران همسرم که نگاهش ترجمان محبت است و وجودش شور زندگی.

تشکر می‌کنم ازاساتید بزرگوارم، دکتر کشمیری و دکتر تقی‌راد، که همواره راهنمایی‌هایشان، چراغ راهی بود در تاریکی‌های جهل این حقیر.

قدردانی می‌کنم ازمهربان برادری که حضور چند ساله‌ام در کلاس‌های درس، نتیجه‌ای جز شرمساری دوری از او، برایم نداشت.

تشکر می‌کنم از دوستان عزیزی که در طول این دوره‌ی پر تلاطم، کمک‌های بی‌منتشان، آرامشی بود بر دریای وجود. دوستان عزیزم آقایان؛ کسری خصوصی، امیرحسین تمجیدی، هم‌اتاقی‌های مهربانم آقایان بردیا نظام زاده، حامد پورآزاد و فرید علوی و برادران کوچکم محمد قربانیان، امیر حیدری.

چکیده

مکان‌یابی و نقشه‌یابی همزمان ربات سیار، بخشی بسیار مهم و ضروری از مسئله‌ی کلی تماماً خودمختار کردن این ربات‌ها است. مکان‌یابی دقیق به نقشه‌ای از محیط نیاز دارد و برای تولید نقشه نیز به مکان‌سنجی دقیق نیاز است.

اولین حل ارائه‌شده از این مسئله، حل توسط فیلتر کالمن توسعه‌یافته، مشکلاتی از جمله کند بودن و عدم توانایی اجرای به‌هنگام را داشت. حل بعدی که برای مسئله‌ی SLAM ارائه شد، حل توسط فیلترهای ذره‌ای بود که Fast-SLAM نام داشت. روش فیلتر ذره‌ای به علت استفاده از نمونه‌ها برای معرفی توابع چگالی احتمال، محدود به فرض گوسی بودن نویزهای محیطی نیست. بعلاوه در این روش فرض محدودکننده‌ی خطی بودن معادلات سیستم، که در فیلترهای کالمن در نظر گرفته می‌شد، وجود ندارد و می‌توان از مدل سیستم و مدل مشاهده‌ی غیرخطی در معادلات فیلتر، استفاده کرد. محدود نبودن این نوع از فیلترها به معادلات خطی، استفاده از فیلترهای ذره‌ای را برای حل مسئله‌ی SLAM که معادلاتی غیرخطی دارند، بسیار مناسب کرده است.

الگوریتم Fast-SLAM با وجود ویژگی‌های بسیار مثبت خود نسبت به روش‌های قبلی حل مسئله‌ی SLAM، به علت نقص در معادلات فیزیکی، در شیب‌های مثبت و منفی محیطی دچار خطا می‌شود. در این پژوهش خطای موجود در الگوریتم اصلاح‌شده و بر روی ربات پیاده‌سازی شده است. این پیاده‌سازی به صورت به‌هنگام صورت می‌گیرد به این معنی که الگوریتم همزمان با داده‌برداری ربات از محیط، اجراشده و نقشه‌ی محیط را تشکیل داده و مکان ربات را اصلاح می‌کند. اصلاح مدل حرکت و مدل مشاهده موجب برطرف شدن خطای موجود در الگوریتم پایه خواهد شد. در ادامه مکان‌یابی ربات از دو بعد به ۲/۵ بعد ارتقا یافته است. مکان‌یابی ربات که پیش‌تر به صورت صفحه‌ای (در صفحه‌ی x,y) انجام می‌شد، به مکان‌یابی در فضا (در سه بعد x,y,z) ارتقا یافت. برای انجام این کار، از داده‌های رسیده از حسگر اینرسی و قید به وجود آمده در حرکت ربات بر روی شیب استفاده شد. معادلات قید به‌وجودآمده و معادلات مشاهده‌ی اصلاح‌شده در یک فیلتر کالمن توسعه‌یافته جایگذاری شده و با استفاده از این فیلتر اصلاح موقعیت در راستای z انجام شده است.

فهرست

۱-فصل اول مقدمه ۲

۱-۱ پیشگفتار ۲

۱-۲ مکان‌یابی و تولید نقشه به صورت همزمان (SLAM) ۳

۱-۳ رباتیک احتمالی ۴

۱-۴ مروری بر کارهای انجام‌شده ۵

۱-۵ تعریف مسئله ۸

۱-۶ نوآوری پروژه و اهمیت آن ۱۰

۱-۷ فصل‌بندی‌های پایاننامه ۱۰

۲-فصل دوم نظریه‌ی فیلترسازی ۱۱

۲-۱ مفاهیم پایه در احتمالات ۱۱

۲-۲ فیلتر بیز ۱۴

۲-۲-۱ فرضیه مارکوف ۱۴

۲-۲-۲ الگوریتم فیلتر بیز ۱۴

۲-۳ فیلتر کالمن ۱۶

۲-۴ فیلتر کالمن توسعه‌یافته ۱۷

۲-۵ فیلتر ذره‌ای ۱۸

۲-۵-۱ وزن‌دهی به ذرات ۲۲

۲-۵-۲ انحطاط الگوریتم ۲۵

۲-۵-۳ باز نمونه‌برداری ۲۶

۲-۶ جمع‌بندی ۲۷

۳-فصل سوم حل مسئله‌ی SLAM با استفاده از فیلتر ذره‌ای به صورت به‌هنگام ۲۹

۳-۱ مقدمه ۲۹

۳-۲ شیوه‌های مختلف توصیف محیط ۳۰

۳-۲-۱ نقشه‌ی ویژگی مبنا ۳۰

۳-۲-۲ نقشه‌های موقعیت مبنا ۳۱

۳-۲-۳ نقشه‌های توپولوژیکی ۳۱
۳-۳ مدل حرکت ۳۲
۳-۳-۱ مدل حرکتی جابه‌جایی مبنا ۳۳
۳-۳-۲ مدل حرکتی سرعت مبنا ۳۳
۳-۴ مدل مشاهده ۳۴
۳-۵ بردار حالت سیستم در مسئله SLAM ۳۵
۳-۶ مراحل حل مسئله مکان‌یابی و تولید نقشه‌ی ویژگی مبنا ۳۷
۳-۷ حل مسئله‌ی SLAM با استفاده از فیلترهای ذره‌ای برای نقشه‌ی ویژگی مبنا ۳۸
۳-۷-۱ استخراج وابستگی بین مؤلفه‌های بردار حالت یا رائوبلکولایز کردن ۳۹
۳-۸ الگوریتم FAST-SLAM با نقشه‌ی شبکه‌ای یا موقعیت مبنا ۴۵
۳-۹ مزایا و معایب FAST-SLAM ۵۰
۳-۱۰ جمع‌بندی ۵۳
۴-فصل چهارم پیاده‌سازی عملی ۵۴
۴-۱ سخت‌افزار ربات ۵۴
۴-۱-۲ طرح بدنه ۵۵
۴-۱-۳ جا نمایی بردها و حسگرها ۵۶
۴-۱-۴ اتصالات ۵۷
۴-۲ نرم‌افزار ربات ۵۸
۴-۲-۱ معرفی ROS و جنبه‌های مختلف نرم‌افزاری آن ۵۸
۴-۲-۲ اجزای اصلی یک ساختار نرم‌افزاری در محیط ROS ۶۱
۴-۲-۳ بسته‌ی gmapping ۶۳
۴-۲-۴ بسته‌ی TF ۶۴
۴-۲-۵ کتابخانه‌ی فیلترهای بیزی ۶۵
۴-۳ راه‌اندازی حسگرهای ربات ۶۶
۴-۳-۱ حسگر پویشگر لیزری ۶۶

۴-۳-۲ راه‌اندازی حسگر IMU ۶۷
۴-۳-۳ راه‌اندازی انکودرها ۶۷
۴-۴ جمع‌بندی ۶۸
۵-فصل پنجم اصلاح نقشه‌ی دوبعدی و بسط مکان‌یابی به سه بعد
۷۰ERROR! BOOKMARK NOT DEFINED.
۵-۱ شرح مسئله **۷۰ERROR! BOOKMARK NOT DEFINED.**
۵-۲ اصلاح معادلات مدل حرکت برای حرکت سه بعدی **۷۱ERROR! BOOKMARK NOT DEFINED.**
۵-۳ نتایج یک آزمایش عملی **۷۳ERROR! BOOKMARK NOT DEFINED.**
۵-۴ مکان‌یابی سه بعدی **۷۷ERROR! BOOKMARK NOT DEFINED.**
۶-فصل ششم نتیجه‌گیری ۷۰
۶-۱ جمع‌بندی ۷۰
۶-۲ پیشنهادها ۷۱
مراجع ۸۵
پیوست الف. دستورات پرکاربرد در محیط نرم‌افزاری ROS ۷۷
پیوست ب. مشخصه‌های بسته GMAPPING ۸۱

فهرست شکل

شکل۱-۱ ربات کاوشگر که در محیط ناشناخته به جستجو برای حیات میپردازد ۳

شکل۱-۲ ربات ساخته‌شده برای اجرای الگوریتم FAST-SLAM ۹

شکل۱-۳ حسگر پویش گر لیزری مدل HOKOYO URG-04LX ۹

شکل۱-۴ حسگر اینرسی مدل XSENS ۹

شکل۱-۵ انکودر مدل AUTONICS ۹

شکل۱-۶ برد آردوییِنو مدل UNO ۹

شکل۲-۱ یک توزیع فرضی ۲۰

شکل۲-۲ معرفی توزیع فرضی با استفاده از نمونه و وزن نمونه ۲۰

شکل۲-۳ نمونه‌برداری از یک توزیع فرضی با استفاده از الگوریتم باز نمونه‌برداری ۲۱

شکل۲-۴ نمونه‌برداری از تابع توزیع دلخواه با استفاده از تابع توزیع گوسی
۲۳

شکل۲-۵ نمونه‌برداری اشتباه که باعث از دست رفتن اطلاعات می‌شود ۲۷

شکل۳-۱ نمونه‌هایی از نقشه‌ی ویژگی مبنا
۳۰

شکل۳-۲ نقشه‌ی موقعیت مبنا ۳۱

شکل۳-۳ مقایسه‌ی نقشه‌ی توپولوژیکی (سمت راست) و ویژگی مبنا (سمت چپ). ۳۲

شکل۳-۴ مدل حرکتی جابه‌جایی مبنا ۳۳

شکل۳-۵ مدل حرکتی سرعت مبنا ۳۴

شکل۳-۶ رباتی که در یک محیط ناشناخته قرار دارد و در حال تخمین موقعیت نسبی خود نسبت به نشانه‌هاست ۳۵

شکل۳-۷ مفهوم علائم ریاضی موجود در تابع توزیع چگالی هدف در مسئله‌ی SLAM ۴۰

شکل۳-۸ مدل سیستم به صورت کلی مکان قرارگیری تابع توزیع را محدود و مدل مشاهده به صورت محلی محدود می‌کند ۴۴

شکل۳-۹ نقشه‌ی شبکه‌ای و چگونگی در نظر گرفتن اشغال بودن و یا خالی بودن موقعیت متناظر با هر شبکه ۴۶

شکل۳-۱۰ خطای تجمعی در تخمین اولیه مکان ربات، باعث تشکیل نقشه‌ی محیط با خطای زیادی می‌شود ۴۸

شکل۳-۱۱ چند نقشه‌ی تولیدشده توسط الگوریتم FAST-SLAM 2.0 برای توصیف محیط از نقشه‌های موقعیت مبنا. ۴۹

شکل۳-۱۲ نقشه‌ی تولیدشده توسط الگوریتم FAST-SLAM
۵۰

شکل۳-۱۳ دقت الگوریتم FAST-SLAM در مقایسه با الگوریتم EKF
۵۱
شکل۳-۱۴ خطای الگوریتم FAST-SLAM در مقایسه با EKF ناشی شده از خطا در حرکت دورانی ربات ۵۲
شکل۳-۱۵ انطباق داده؛ الگوریتم به خاطر خطای پیش‌آمده در مورد زاویه توانایی تشخیص خود را از دست می‌دهد
۵۲

شکل۳-۱۶ انطباق داده در فیلترهای ذره‌ای برای هر ذره به صورت جداگانه در نظر گرفته می‌شود ۵۳
شکل۴-۱ ربات ساخته‌شده برای پیاده‌سازی الگوریتم SLAM ۵۵
شکل۴-۲ نمای سه بعدی طرح گاری، طراحی‌شده توسط نرم‌افزار CATIA ۵۵
شکل۴-۳ نقشه‌ی چرخ‌های جلو
۵۶
شکل۴-۴ نقشه‌ی شافت ۵۷
شکل۴-۵ نقشه‌ی نگه‌دارنده بال ترنسفر ۵۸
شکل۴-۶ نحوه‌ی ارتباط ROS با سخت‌افزار کامپیوتر ۵۹
شکل۴-۷ شمای کلی پردازش‌های صورت گرفته در مرورگر اینترنتی ۶۰
شکل۴-۸ فضای کار در سیستم‌عامل ROS ۶۱
شکل۴-۹ فضای رابط کاربری در بسته‌ی RQT-GRAPH ۶۳
شکل۴-۱۰ رابط کاربری در بسته‌ی RQT-PLOT ۶۳
شکل۴-۱۱ نقشه‌های تشکیل‌شده با اجرای الگوریتم FAST-SLAM ۶۴
شکل۵-۱ ربات قرارگرفته در شیب
۷۱ERROR! BOOKMARK NOT DEFINED.
شکل۵-۲ دیاگرام کلی الگوریتم و نحوه‌ی اصلاح آن به وسیله‌ی IMU
۷۳ERROR! BOOKMARK NOT DEFINED.
شکل۵-۳ نقشه‌ی واقعی محیط
۷۳ERROR! BOOKMARK NOT DEFINED.
شکل۵-۴ نمای کناری شیب
۷۳ERROR! BOOKMARK NOT DEFINED.
شکل۵-۵ نقشه‌ی به دست آمده از اجرای الگوریتم اصلاح‌شده
۷۵ERROR! BOOKMARK NOT DEFINED.
شکل۵-۶ تغییرات S نسبت به زمان
۷۵ERROR! BOOKMARK NOT DEFINED.
شکل۵-۷ نمودار مشتق متغیر s نسبت به زمان.. ۷۶

شکل۵-۸ پیشروی جزئی ربات در شیب

۷۸ERROR! BOOKMARK NOT DEFINED.

شکل۵-۹ نتیجه‌ی اجرای الگوریتم تصحیح‌شده و تصحیح‌نشده

۷۹ERROR! BOOKMARK NOT DEFINED.

شکل ۵-۱۰ نقشه‌ی تشکیل‌شده توسط یاو به دست آمده از IMUو زاویه‌ی به دست آمده از اودومتری ۸۱

شکل الف-۱ فضای رابط کاربری در بسته‌ی RQT-GRAPH

۸۰

فهرست جداول

جدول۵-۱ پویش ربات در محیط به تفکیک زمانی

۷۴ERROR! BOOKMARK NOT DEFINED.

جدول ب-۱ TOPIC مورد نیاز برای اجرای بسته‌ی GMAPPING ۸۲

جدول ب-۲ TOPIC ارسال‌شده توسط بسته‌ی GMAPPING ۸۲

جدول ب-۳ جدول امکانات ارائه‌شده در بسته ۸۲

جدول ب-۴ متغیرهای بسته‌ی GMAPPING. ۸۳

جدول ب-۵ نحوه‌ی مقداردهی متغیرها ۸۵

فصل اول
مقدمه

۱-۱ پیشگفتار

در ابتدای ورود ربات‌ها به زندگی بشر، برنامه‌ریزی آن‌ها صرفاً برای محیط‌های خاص انجام می‌گرفت. در چنین شرایطی با فراهم کردن محیطی کنترل شده و شناخته‌شده، ربات به خوبی از پس فعالیت‌های محوله برمی‌آمد. با پیشرفت علم رباتیک و پیدایش تفکر ربات‌های سیار خودمختار[1] مشکلاتی جدی در توانمندی ربات‌ها احساس شد، چرا که در این موارد لازم بود ربات در شرایطی ناشناخته کار کند. ربات‌های متحرک خودمختار به حذف انسان از محیط‌های نامطلوب و خطرناک کمک می‌کند. جستجو و نجات افراد آسیب‌دیده، جابجایی افراد معلول، معدن کاوی، اکتشافات فضایی، کار در محیط‌هایی با آلودگی‌های مختلف شیمیایی، رادیواکتیوی و صوتی و کار در منازل و موزه‌ها مثال‌هایی از کاربردهای بی‌شمار ربات‌های متحرک خودمختار هستند.

[1] Autonomous Mobile Robots

۱-۲ مکان‌یابی و تولید نقشه به صورت همزمان (SLAM[1])

بررسی نحوه درک محیط و نمایش آن، حرکت و کاوش در محیط ناشناخته و ایجاد تغییر در آن‌ها از طریق عوامل مصنوعی‌ای که توسط رایانه کنترل می‌شوند از جمله موضوعاتی هستند که با موضوع ربات‌های متحرک خودمختار پیوندی عمیق دارند. به طور کلی یک ربات متحرک برای انجام مأموریتی خاص طراحی و برنامه‌ریزی می‌شود. برای انجام این مأموریت، ربات باید در زمانی معین به هدف تعیین‌شده برسد. بنابراین لازم است ربات در هر گام مکان خود را نسبت به هدف تعیین کرده و با توجه به آن سیگنال کنترلی لازم را برای رسیدن به آن صادر نماید. به مسئله‌ی تعیین مکان ربات، مکان‌یابی[2] گفته می‌شود. علاوه بر این، کسب اطلاعات در مورد محیط نیز برای دوری از موانع، طراحی مسیر بهینه و اکتشاف محیط ضروری است. این اطلاعات به طور کلی توصیف محیط[3] نامیده می‌شود. به مسئله به دست آوردن توصیفی از محیط، ایجاد نقشه[4] اطلاق می‌شود. ربات موجود در شکل۱-۱ که برای کاوش به سطح مریخ فرستاده شده بود، برای مکان‌یابی و نقشه‌یابی از الگوریتم SLAM استفاده می‌کند.

شکل۱-۱ ربات کاوشگر که در محیط ناشناخته به جستجو برای حیات می‌پردازد[۱]

مسئله SLAM به بررسی کاوش ربات در محیط ناشناخته یا مکان‌یابی و تولید هم‌زمان نقشه و تعیین موقعیت ربات در آن نقشه می‌پردازد. دشواری این مسئله به وابستگی مکان‌یابی و تولید نقشه به یکدیگر برمی‌گردد. برای حل مسئله مکان‌یابی، دسترسی به نقشه‌ای از محیط تا بتوان با تطبیق مشاهدات در هر لحظه با آن تخمینی از مکان ربات به دست آورد، لازم به نظر می‌رسد. از سوی دیگر برای تولید نقشه نیز ربات باید در هر لحظه از مکان خود اطلاع داشته باشد تا بتواند با کنار هم قرار دادن اطلاعات به دست آمده نقشه را تکمیل کند.

---

[1] Simultaneous Localization And Mapping
[2] Localization
[3] Representation
[4] Mapping

برای حل مسئله SLAM چهارچوب‌های متفاوتی در نظر گرفته شده است. به عنوان نمونه می‌توان به پیاده‌سازی SLAM در محیط‌های داخلی[1]، محیط‌های خارجی[2]، داخل آب و عملیات هوایی اشاره کرد. توصیفی که از محیط در مسئله SLAM ارائه می‌شود، به دو دسته‌ی ۳ بعدی و ۲ بعدی تقسیم می‌شود.

۱-۳ رباتیک احتمالی

گفته شد که ناشناخته بودن محیط برای یک ربات سیار می‌تواند موجب بروز اختلالاتی در عملکرد مطلوب ربات شود. منشأ این مشکلات عدم قطعیت‌های مربوط به ربات و محیط اطراف است. به طور کلی در حل مسائل مربوط به ربات‌های سیار عدم قطعیت‌هایی وجود دارد که به پنج دسته کلی تقسیم می‌شوند: محیط، حسگرها، عملگرها، مدل کنترلی و محاسبات[۲].

الف) محیط[3]: ناشناخته بودن محیط به معنی عدم توانایی ما در به دست آوردن مدل ریاضی آن است و این مسئله خود به تنهایی بزرگ‌ترین منبع برای ایجاد خطا در عملکرد ربات سیار است. پستی و بلندی‌های زمین، تفاوت در اصطکاک قسمت‌های مختلف مسیر حرکت و... از عواملی هستند که در محیط باعث ایجاد عدم قطعیت شده تا حدی که این عدم قطعیت می‌تواند عملکرد ربات را در مدت زمان بسیار کوتاهی مخدوش کند.

ب) حسگرها[4]: حسگرها تنها وسیله در اختیار ربات برای درک محیط هستند که به طور عمده با دو نوع کلی عدم قطعیت مواجه‌اند. اولاً حسگرها محدوده‌ی کارکرد دارند و ثانیاً داده‌های حسگرها آغشته به نویز هستند که این مسئله به طور مستقیم به دقت آن‌ها مربوط می‌شود.

ج) عملگرها[5]: لقی در موتور الکتریکی، ویسکوز در نظر گرفتن اصطکاک روی شافت موتور و لغزش همگی از مواردی هستند که ایجاد خطا می‌کنند و منبع ایجاد عدم قطعیت می‌شود.

د) مدل دینامیکی: به عنوان مثال زمانی که در یک مسئله برای راحت شدن حل اصطکاک موجود در نظر گرفته نمی‌شود. همین مسئله در گذر زمان ایجاد خطای انباشته کرده و مسئله را به سمت واگرایی پیش می‌برد.

---

[1] Indoor Environment
[2] Outdoor Environment
[3] Environment
[4] Sensors
[5] Robot Actuators

ه) محاسبات[1]: به هر حال در سیستم‌های رباتیکی مجبور هستیم داده‌های به دست آمده از حسگرها را در حافظه ذخیره و محاسباتی را بر روی آن‌ها انجام دهیم. عمل ذخیره کردن و انجام محاسبات به معنی گرد کردن عدد تا حدی از دقت است. این مسئله در گذر زمان ایجاد خطای انباشته کرده و مسئله را واگرا می‌کند.

در دهه‌های اخیر جهت افزایش توانمندی‌های ربات‌ها متخصصان کوشیدند تا روشی برای در نظر گرفتن عدم قطعیت‌ها پیداکرده و آن‌ها را نیز در محاسبات خود تأثیر دهند. از آنجا که علم عدم قطعیت‌ها، علم احتمال و متغیرهای تصادفی است، در برخورد این علم با رباتیک شاخه‌ی جدیدی از علم با نام رباتیک احتمالی[2]مطرح شد. رباتیک احتمالی به بررسی مسائلی از رباتیک می‌پردازد که با این عدم قطعیت‌ها درگیر است. در این علم با این مسائل به شکل الگوریتم‌های احتمالی و متغیرهای تصادفی برخورد می‌شود و کارکرد ربات نسبت به این عدم قطعیت‌ها مقاوم می‌شود.

در رباتیک احتمالی سعی بر آن است که ربات را از محیط ایزوله‌ی صنعتی یا آزمایشگاهی که پیش‌تر در آن کار می‌کرده، بیرون آورده و وارد محیط‌های جدید کنند. این حوزه از علم با مسئله‌ی مکان‌یابی و رسم نقشه به صورت همزمان برای ربات‌های سیار شروع شد و بخش اعظمی از رباتیک احتمالی پاسخ به جنبه‌های مختلف مسئله‌ی SLAM است.

۱-۴ مروری بر کارهای انجام‌شده

اولین تلاش‌ها برای حل مسئله SLAM در یک چهارچوب احتمالی توسط اسمیث[3] و همکارانش، لئونارد[4] و دورانت[5] ارائه شد[۳]،[۴]و[۵]. این پژوهشگران بردارهای حالت ربات و محیط را به صورت متغیرهای تصادفی در نظر گرفته‌اند. در این پژوهش‌ها هدف SLAM، تخمین تابع توزیع چگالی احتمال گوسی برای هر یک از متغیرهای تصادفی در هر گام زمانی با استفاده از مشاهدات و تخمین‌های قبلی، توصیف نقشه محیط با استفاده از نشانه‌های تخمینی بوده است.

در سال ۲۰۰۱ توسط دیسانیاک[6] و همکارانش راه‌حلی بر مبنای کالمن فیلتر توسعه‌یافته[1] برای حل مسئله SLAM ارائه شد[۶]. در این مقاله تخمین اولیه بر پایه‌ی معادلات حالتی که برای سینماتیک مسئله استخراج می‌شود

[1]Computations
[2]Probabilistic robotics
[3]Smith
[4]Leonard
[5]Durrant
[6]Dissanayake

انجام گرفت و در آن مسائلی چون همگرایی نقشه بررسی شد. به دلیل اینکه معادلات فضای حالت مدل حرکت و مدل مشاهده غیرخطی در نظر گرفته می‌شود، استفاده از فیلتر کالمن به صورت مستقیم امکان‌پذیر نیست و در هر مرحله لازم است مدل‌های خطی تقریب زده‌شده توسط جمله اول بسط تیلور حول نقطه تخمین را در معادلات قرار داد که این خطای ناشی از مدل‌سازی می‌تواند منجر به ناسازگاری و واگرا شدن حل شود.

یکی دیگر از مشکلات الگوریتم EKF-SLAM، بزرگی بردار حالت و افزایش پارامترهای حالت سیستم با افزایش مشاهدات ربات از محیط است که منجر به بزرگ شدن ماتریس کوواریانس حالت با درجه پیچیدگی $O(N^2)$ می‌شود. در مراحل انجام فرآیند SLAM توسط الگوریتم کالمن فیلتر توسعه‌یافته نیاز به به‌روزرسانی و معکوس کردن این ماتریس است. این هزینه محاسباتی زیاد مشکل‌زا بوده و موجب ناکارآمدی الگوریتم در محیط‌های خارجی می‌شود[۷]. برای حل این مشکل در دهه اخیر تلاش‌هایی صورت گرفته است. یکی از این راه‌حل‌ها در سال ۲۰۰۴ توسط ترون[۲] و همکارانش ارائه شد[۸]. در این مقاله روشی برای حل مسئله SLAM با استفاده از فیلتر اطلاعات توسعه‌یافته موسوم به SEIF ارائه شد. در این روش به جای استفاده از ماتریس کوواریانس از ماتریس اطلاعات که معکوس آن است استفاده می‌شود و معادلات فیلتر کالمن با این ماتریس بازنویسی شده و فیلتر اطلاعات به دست می‌آید. در این ماتریس خاصیت پراکندگی وجود دارد و می‌توان از وابستگی نشانه‌هایی که فاصله زیادی از هم دارند صرف‌نظر کرد، تا حجم محاسبات کاهش یابد. با این وجود با توجه به اینکه ماتریس کوواریانس به طور دقیق مشخص نیست، کیفیت تناظر داده در این روش کاهش می‌یابد.

بیلی[۳] و همکارانش در سال ۲۰۰۶ به بررسی همگرایی حل در روش EKF پرداختند[۹]. در همان سال توسط کاستالنوس[۴] و همکارانش نکاتی راجع به بهبود همگرایی و سرعت همگرایی در تخمین مسئله با استفاده از کالمن فیلتر توسعه‌یافته ارائه شد[۱۰]. آن‌ها نشان دادند که خطی‌سازی معادلات در حالتی که عدم قطعیت از یک حد بالایی عبور کند، حل را به شدت واگرا می‌کند. همچنین نشان دادند که همگرایی الگوریتم EKF-SLAM با محدود کردن درجه عدم قطعیت در محاسبات پیوسته نقشه تصادفی توسط ساختن نقشه‌های متوالی محلی غیر وابسته و استفاده از توصیف مرکز ربات برای هر نقشه محلی، بهبود می‌یابد.

در سال ۲۰۰۸ Paz و همکاران روش حل جدیدی بر پایه‌ی کالمن فیلتر توسعه‌یافته ارائه دادند که Divide & Conquer SLAM نامیده می‌شود[۱۱]. در این الگوریتم درجه پیچیدگی محاسبات در هر مرحله از $O(N^2)$ به

---

[1] Extended Kalman Filter (EKF)
[2] Thrun
[3] Bailey
[4] Castellanos