

بهبود موقعیت‌یابی و نقشه‌یابی هم‌زمان با استفاده از الگوریتم اجتماع ذرات و سیستم فازی-عصبی تطبیقی

رمضان هاونگی، استادیار

دانشکده مهندسی برق و کامپیوتر - دانشگاه بیرجند - بیرجند - ایران - Havangi@Birjand.ac.ir

چکیده: مسئله موقعیت‌یابی و نقشه‌یابی هم‌زمان (SLAM) یکی از نیازهای اساسی برای ربات‌های خودمختار متحرک است که در محیط‌های ناشناخته حرکت می‌کنند. الگوریتم UFastSLAM یک روش مؤثر برای این منظور است. این روش با به‌کاربردن تبدیل خنثی، الگوریتم FastSLAM را بهبود می‌دهد. با وجود این، فرآیند نمونه‌برداری مجدد و اطلاعات آماری نامعلوم نویز فرآیند و اندازه‌گیری منجر به ناسازگاری می‌شود. در این مقاله، برای بهبود UFastSLAM از حیث دقت و سازگاری، الگوریتم UFastSLAM بهبود یافته با استفاده از الگوریتم اجتماع ذرات و سیستم فازی-عصبی تطبیقی (ANFIS) ارائه شده است. در روش پیشنهادی ANFIS به‌طور تطبیقی مشخصات آماری نویزها را تخمین می‌زند و بر سازگاری نظارت دارد. درحالی که الگوریتم اجتماع ذرات برای اصلاح نمونه‌ها استفاده شده است. مهم‌ترین امتیاز الگوریتم پیشنهادی نسبت به سایر الگوریتم‌ها عملکرد بهتر آن از حیث دقت و سازگاری تحت شرایط مختلف است. به‌ویژه وقتی که مشخصات آماری نویزها نامعلوم است، عملکرد سایر الگوریتم‌ها کاهش می‌یابد درحالی که روش پیشنهادی از دقت بالایی برخوردار است. به‌علاوه، نسبت به سایر روش‌ها، روش پیشنهادی وابستگی کمتری به تعداد ذرات دارد و بنابراین با حجم محاسبات کمتر به دقت بیشتری می‌رسد.

واژه‌های کلیدی: موقعیت‌یابی و نقشه‌یابی هم‌زمان ربات (SLAM)، UFastSLAM، سیستم فازی-عصبی تطبیقی (ANFIS)، الگوریتم اجتماع ذرات

Improving of Simultaneous Localization and Mapping using Particle Swarm Optimization and Adaptive Neuro-Fuzzy Inference System

R. Havangi, Assistant Professor

Faculty of Electrical and Computer Engineering, University of Birjand, Birjand, Iran, Email: Havangi@Birjand.ac.ir

Abstract: The simultaneous localization and mapping (SLAM) problem is a fundamental requirement for autonomous robots that moves in unknown environment. The UFastSLAM is effective way for this purpose. This method improves the FastSLAM algorithm using unscented transform. However, the resampling process and unknown statistical information process and measurement noise lead to inconsistency. To improve UFastSLAM in terms of accuracy and consistency, in this article, the improved UFastSLAM using particle swarm optimization (PSO) and adaptive neuro-fuzzy inference system (ANFIS) is proposed. In this method, ANFIS estimates adaptively statistical characteristics of the noises and supervises the consistency. While PSO is used to modify samples. Especially when the statistical characteristics of the noises are unknown, the performance of other algorithms decreases while proposed method has high accuracy. In addition, compared to other methods, the proposed method less dependent on the number of particles and thus it provides greater accuracy with less computational cost.

Keywords: Simultaneous localization and mapping (SLAM), UFastSLAM, adaptive neuro-fuzzy inference system (ANFIS), particle swarm optimization.

تاریخ ارسال مقاله: ۱۳۹۵/۰۶/۲۳

تاریخ اصلاح مقاله: ۱۳۹۵/۰۸/۱۱

تاریخ پذیرش مقاله: ۱۳۹۵/۱۰/۱۶

نام نویسنده مسئول: رمضان هاونگی

نشانی نویسنده مسئول: ایران - بیرجند-انتهای بلوار شهید آوینی - دانشگاه بیرجند - دانشکده مهندسی برق و کامپیوتر

۱- مقدمه

مسئله موقعیت‌یابی و نقشه‌یابی هم‌زمان یکی از نیازهای ضروری ربات‌های خودمختار متحرکی است که در محیط‌های ناشناخته حرکت می‌کنند [۱-۲]. حل مسئله SLAM بر اساس فیلتر کالمن توسعه یافته (EKF-SLAM) یکی از قدیمی‌ترین و محبوب‌ترین روش‌ها است [۳-۴]. با وجود این، EKF-SLAM از دو مشکل عمده پیچیدگی محاسبات و تناظر داده رنج می‌برد [۵].

برای حل این مشکلات حل مسئله SLAM مبتنی بر فیلترهای ذره‌ای که به FastSLAM معروف است مطرح شده است [۶-۷]. این روش علی‌رغم موفقیت‌ها دارای مشکلاتی است که مهم‌ترین آن‌ها ناسازگاری و تباهدگی است [۸-۱۰].

در [۸] نشان داده شده است که الگوریتم FastSLAM به یک تخمین‌های ناسازگار منجر می‌شود. با توجه به این‌که روش FastSLAM هر ذره به‌طور ضمنی تاریخچه حالت را در نقشه مربوط به خود ثبت می‌کند از این‌رو هر زمان که یک ذره در فرآیند نمونه‌برداری مجدد از بین می‌رود کل نقشه مربوط به آن از دست می‌رود و سبب انحطاط در اطلاعات گذشته می‌شود. نتایج شبیه‌سازی‌ها در این مقاله نشان می‌دهند که صرف‌نظر از تعداد ذرات و همچنین چگالی نشانه‌ها در محیط، الگوریتم FastSLAM با گذشت زمان تباهیده خواهد شد و تخمین‌های خوش بینانه‌ای از نامعینی‌ها دارد.

تاکنون برای بهبود عملکرد FastSLAM تحقیقات زیادی انجام شده است. در [۱۱] به‌منظور بهبود FastSLAM یک رویه SLAM مبتنی بر ترکیب فیلتر ذره‌ای خنثی^۱ (UPF) و فیلترهای کالمن خنثی^۲ (UKFs) بنام UPF-UKF SLAM پیشنهاد دادند. در این روش از UPF برای تخمین حالت ربات و از UKF‌ها برای تخمین موقعیت نشانه‌ها استفاده شده است. در این مقاله نشان داده شده است که تخمین حالت ربات با استفاده از UPF دقیق‌تر و دارای سازگاری بهتری نسبت به تخمین حالت ربات با استفاده از فیلتر ذره‌ای است. در [۱۲] یک الگوریتم FastSLAM بر اساس فیلتر ذره‌ای کمکی^۳ (APF) با درون‌یابی Stirling معرفی کردند. در این مقاله فرآیند خطی‌سازی با محاسبه ژاکوبین به‌وسیله درون‌یابی Stirling حذف شده که شبیه تبدیل خنثی^۴ است. اولین امتیاز الگوریتم پیشنهادی این است که نیاز به ماتریس ژاکوبین ندارد و دوم اینکه تقریب‌های غیرخطی استفاده‌شده در این فیلتر بهتر از روش استفاده‌شده در EKF است. به‌علاوه وقتی از APF به‌عنوان توزیع پیشنهادی استفاده می‌شود، داده اندازه‌گیری به توزیع واقعی ذرات بیش‌تر نزدیک است. از نتایج مقاله مشاهده می‌شود که دقت تخمین موقعیت‌های ربات افزایش و سرعت تباهدگی ذرات به‌وسیله الگوریتم پیشنهادی کاهش یافته است.

در [۱۳-۱۶] یک الگوریتم بر اساس تبدیل خنثی مقیاس‌شده به‌نام UFastSLAM برای بهبود دقت تخمین‌ها و سازگاری FastSLAM پیشنهاد شده است. در این الگوریتم به‌جای خطی‌سازی مدل‌های

غیرخطی از طریق بسط مرتبه اول سری تیلور در میانگین حالت‌های وسیله، میانگین و کواریانس را با به‌کارگیری تکنیک فیلتر ذره‌ای خنثی (UPF) محاسبه می‌کند. به‌علاوه به‌روزرسانی میانگین و کواریانس نشانه‌ها با استفاده از فیلتر خنثی از خطاهای خطی‌سازی و محاسبه ژاکوبین در تخمین نشانه‌ها جلوگیری می‌کند. نشانه‌های جدید با استفاده از تبدیل خنثی به نقشه اضافه می‌شوند. یکی از تفاوت‌های کار انجام‌شده در [۱۴-۱۶] با کارهای مشابه انجام‌شده در این زمینه در آن است که برخلاف کارهای قبلی که تنها در قسمتی از الگوریتم FastSLAM از تبدیل خنثی استفاده شده، در این الگوریتم برای طراحی تابع توزیع پیشنهادی، پیشگویی و به‌روزرسانی نشانه‌ها، مقداردهی اولیه نشانه‌ها و اضافه‌کردن نشانه‌ها از تبدیل خنثی استفاده شده است. نتایج این مقاله حاکی از آن است که دقت تخمین‌ها نسبت به روش‌های FastSLAM بهبود داده شده است. این بدان معناست که الگوریتم پیشنهادی امتیاز افزوده شده کاهش تعداد ذرات را دارد درحالی که دقت تخمین‌ها حفظ می‌شود. از نقطه‌نظر سازگاری، از آنجایی که حالت‌های وسیله و نشانه‌ها بدون خطاهای خطی‌سازی جمع‌شونده در UFastSLAM تخمین زده می‌شوند، بسیاری از ذرات اطلاعات دقیقی دارند. از این‌رو علی‌رغم این‌که در این روش تنوع ذرات مانند روش‌های قبلی است، تخمین‌های دقیق‌تر سبب می‌شوند که سازگاری فیلتر در حد چند ثانیه افزایش یابد.

در [۱۷] یک الگوریتم توزیع‌شده SLAM برای حل مسئله موقعیت‌یابی و نقشه‌یابی هم‌زمان ارائه شده است. در این روش از الگوریتم اجتماع ذرات برای بهبود فیلتر ذره‌ای توزیع‌شده استفاده در سیستم SLAM توزیع‌شده و از روش وزن اینرسی بهینه‌سازی شده برای بهبود عملکرد الگوریتم اجتماع ذرات استفاده شده است. در [۱۸] الگوریتم STSRCDFastSLAM برای بهبود UFastSLAM پیشنهاد شده است. در این روش به‌جای تبدیل خنثی از تبدیلات تفاضل مرکزی استفاده شده است که سبب بهبود سازگاری و دقت می‌شود. این روش برای تخمین نشانه‌ها و در تشکیل تابع توزیع پیشنهادی از STSRCDF استفاده می‌کند.

با وجود این، فرآیند نمونه‌برداری مجدد الگوریتم UFastSLAM با دورریختن برخی ذرات و چند بار کپی کردن ذرات دیگر سبب کاهش تنوع میان ذرات و در نتیجه منجر به ناسازگاری می‌شود. علاوه‌براین UFastSLAM تحت این فرض که اطلاعات آماری نویز فرآیند و اندازه‌گیری معلوم باشند به تخمین‌های سازگاری از موقعیت ربات و نقشه محیط منجر می‌شود. در کاربردهای واقعی اطلاعات آماری فرآیند و اندازه‌گیری نامعلوم هستند. از طرف دیگر تخمین‌های ضعیف از مشخصات آماری نویزها ممکن است به‌طور جدی سبب پایین آمدن عملکرد UFastSLAM شود

برای حل این مشکل، در این مقاله الگوریتم UFastSLAM بهبودیافته با استفاده از PSO و ANFIS ارائه شده است. در این روش ANFIS بر سازگاری نظارت می‌کند و PSO برای ایجاد تنوع میان

با استفاده از متغیرهای تعریف‌شده در بالا، مسئله SLAM به دست آوردن بهترین تخمین حالت ربات S_t و نقشه Θ با استفاده از اندازه‌گیری‌های نویزی z^t و کنترل‌های u^t است. از دیدگاه احتمالاتی، این مسئله به‌وسیله پسین زیر قابل بیان است [۱۹]:

$$p(S_t, \Theta | z^t, u^t) \quad (۴)$$

اگر تناظر داده‌ها معلوم باشد، پسین SLAM به‌صورت زیر نوشته می‌شود [۱۹]:

$$p(S_t, \Theta | z^t, u^t, n^t) \quad (۵)$$

که n^t بیانگر مجموعه همه تناظر داده‌ها و به‌صورت زیر است:

$$n^t = \{n_1 \quad n_2 \quad \dots \quad n_t\} \quad (۶)$$

به‌طور کلی دو روش عمده برای حل مسئله SLAM عبارتند از روش‌های SLAM مبتنی بر فیلتر کالمن توسعه یافته (EKF-SLAM) و روش‌های SLAM مبتنی بر فیلتر ذره‌ای (FastSLAM).

روش FastSLAM برخلاف EKF-SLAM به فرم خاصی از پسین وابسته نیستند. این روش برخلاف روش‌های EKF-SLAM، به دلایل محاسباتی پسین بر روی مسیر ربات و نقشه به‌صورت زیر در نظر گرفته می‌شود:

$$p(S^t, \Theta | z^t, u^t, n^t) \quad (۷)$$

این اختلاف باعث می‌شود که بتوان پسین SLAM را به‌صورت حاصل ضرب عبارت‌های ساده‌تر زیر تجزیه نمود:

$$p(S^t, \Theta | z^t, u^t, n^t) = \underbrace{p(S^t | z^t, u^t, n^t)}_{\text{path posterior}} \underbrace{\prod_{n=1}^N p(\theta_n | S^t, z^t, u^t, n^t)}_{\text{landmark estimators}} \quad (۸)$$

این فاکتورسازی بیان می‌کند که پسین مسئله SLAM می‌تواند به‌صورت حاصل ضرب پسین مسیر ربات و حاصل ضرب پسین‌های N نشانه در نظر گرفت. الگوریتم UFastSLAM پسین بر روی مسیر ربات $p(S^t | z^t, u^t, n^t)$ را با فیلتر ذره‌ای خنثی و N پسین باقیمانده (پسین نشانه‌ها) $p(\theta_n | S^t, z^t, u^t, n^t)$ با استفاده از فیلتر خنثی تخمین می‌زند. تخمین‌گرهای نشانه‌ها مشروط بر روی مسیر ربات هستند که بدین معنا است که یک نسخه مجزا از هر نشانه بر روی هر ذره وجود دارد. با توجه به معادله (۸)، هر ذره در الگوریتم UFastSLAM به‌فرم زیر است [۱۹]:

$$S_t^{[m]} = \langle S^{t,[m]}, \mu_{1,t}^{[m]}, \Sigma_{1,t}^{[m]}, \dots, \mu_{N,t}^{[m]}, \Sigma_{N,t}^{[m]} \rangle$$

که $[m]$ شاخص ذرات، $S^{t,[m]}$ تخمین مسیر ذره m ام، $\mu_{N,t}^{[m]}$ و $\Sigma_{N,t}^{[m]}$ میانگین و کواریانس نشانه N ام است. الگوریتم UFastSLAM در حالت کلی دارای چهار گام اساسی به‌صورت زیر است [۱۹-۲۰]:

نمونه‌ها بعد از نمونه‌برداری مجدد استفاده شده است. ساختار بقیه مقاله به‌شرح زیر است.

در بخش ۲ مسئله SLAM و روش UFastSLAM به‌طور مختصر بررسی می‌شود. الگوریتم بهبودیافته UFastSLAM در بخش ۳ معرفی می‌گردد. در بخش ۴ الگوریتم پیشنهادی با استفاده از شبیه‌سازی‌های مختلف مورد ارزیابی قرار گرفته است. در بخش ۵ نتیجه‌گیری ارائه شده است.

۲- روش UFastSLAM

یک ربات متحرک را که در یک محیط ناشناخته استاتیک در حال حرکت است در نظر بگیرید. ربات فرامین کنترلی را اجرا و اندازه‌گیری‌ها را از نشانه‌هایی که در محیط هستند جمع‌آوری می‌کند. مسئله SLAM فرآیند به‌دست‌آوردن نقشه محیط و مسیر ربات از مجموعه فرامین کنترلی و اندازه‌گیری‌های آغشته به نویز است. از دیدگاه تئوری بیزین، برای این منظور باید تابع پسین SLAM در هر لحظه تخمین زده شود.

برای تعریف تابع پسین SLAM، فرض می‌شود S_t بردار حالت ربات در لحظه t باشد. بردار حالت برای ربات‌هایی که در یک محیط مسطح عمل می‌کنند، شامل موقعیت ربات در صفحه و سمت است. بنابراین کل مسیر ربات در هر لحظه S^t به‌صورت زیر قابل بیان است:

$$S^t = \{s_1 \quad s_2 \quad \dots \quad s_t\} \quad (۱)$$

همچنین فرض می‌شود که بتوان محیط ربات را به‌صورت N نشانه نقطه‌ای غیرمتحرک مدل نمود. نشانه‌های نقطه‌ای عموماً برای ارائه موقعیت نشانه‌های استخراج‌شده از اطلاعات سنسورها استفاده می‌شوند. مجموعه موقعیت‌های N نشانه به‌صورت $\Theta = \{\theta_1 \quad \dots \quad \theta_N\}$ نوشته می‌شود که بیان‌کننده نقشه محیط است. هنگامی که ربات در محیط حرکت می‌کند، اطلاعات نسبی درباره حرکت خودش جمع‌آوری می‌کند. این اطلاعات با استفاده از انکدرهای نصب‌شده بر روی چرخ‌ها و یا با استفاده از واحد اندازه‌گیری اینرسی (IMU) تولید می‌شوند.

صرف‌نظر از منشأ، هر اندازه‌گیری از حرکت ربات به‌طور کلی منتسب به یک کنترل است. اگر u_t کنترل در لحظه t باشد، مجموعه همه کنترل‌های اجراشده به‌وسیله ربات در هر لحظه را می‌توان به‌صورت زیر نوشت:

$$u^t = \{u_1 \quad u_2 \quad \dots \quad u_t\} \quad (۲)$$

هنگامی که ربات در محیط حرکت می‌کند، فاصله و سمت خود را از نشانه‌های نزدیک اندازه‌گیری می‌کند. اگر Z_t اندازه‌گیری در لحظه t باشد، مجموعه همه اندازه‌گیری‌های جمع‌آوری‌شده به‌وسیله ربات Z^t به‌صورت زیر است:

$$Z^t = \{z_1 \quad z_2 \quad \dots \quad z_t\} \quad (۳)$$

با استفاده از نقاط سیگمای انتقال یافته میانگین $S_{t|t-1}^{[m]}$ و کواریانس $P_{t|t-1}^{[m]}$ پیش بینی شده حالت ربات به صورت زیر محاسبه می شوند:

$$s_{t|t-1}^{[m]} = \sum_{i=0}^{2n} \omega_m^{[i]} \bar{s}_{t|t-1}^{[i][m]} \quad (۱۵)$$

$$P_{t|t-1}^{[m]} = \sum_{i=0}^{2n} \omega_c^{[i]} (\bar{s}_{t|t-1}^{[i][m]} - s_{t|t-1}^{[m]})(\bar{s}_{t|t-1}^{[i][m]} - s_{t|t-1}^{[m]})^T \quad (۱۶)$$

که وزن های $\omega_m^{[i]}$ و $\omega_c^{[i]}$ به صورت زیر محاسبه می شوند:

$$\omega_m^{[0]} = \frac{\lambda}{(n+\lambda)}, \omega_m^{[i]} = \frac{\lambda}{2(n+\lambda)}, i=1, \dots, 2n \quad (۱۷)$$

$$\omega_c^{[0]} = \frac{\lambda}{(n+\lambda)} + (1-\alpha^2 + \beta) \quad (۱۸)$$

$$\omega_c^{[i]} = \frac{\lambda}{2(n+\lambda)}, i=1, \dots, 2n$$

پارامتر β یک اسکالر غیر منفی است. هنگامی که بعضی نشانه ها مشاهده می شوند، تناظر داده ماهیت آن ها را مشخص می کند و در نتیجه اندازه گیری پیش بینی شده $\bar{z}_t^{[m]}$ به صورت زیر محاسبه می شود:

$$\zeta_t^{[i][m]} = h(\bar{s}_{t|t-1}^{[i][m]}, \mu_{n,t-1}^{[m]}) \quad (۱۹)$$

$$\bar{z}_t^{[m]} = \sum_{i=0}^{2n} \omega_m^{[i]} \zeta_t^{[i][m]}$$

ماتریس گین $K_t^{[m]}$ به صورت زیر محاسبه می شود:

$$P_{vv}^{[m]} = \sum_{i=0}^{2n} \omega_c^{[i]} (\zeta_t^{[i][m]} - \bar{z}_t^{[m]})(\zeta_t^{[i][m]} - \bar{z}_t^{[m]})^T + R_t \quad (۲۰)$$

$$P_{\delta v}^{[m]} = \sum_{i=0}^{2n} \omega_c^{[i]} (\bar{s}_{t|t-1}^{[i][m]} - s_{t|t-1}^{[m]})(\zeta_t^{[i][m]} - \bar{z}_t^{[m]})^T$$

$$K_t^{[m]} = P_{\delta v}^{[m]} (P_{vv}^{[m]})^{-1}$$

که $h(\cdot)$ مدل اندازه گیری است. اندازه گیری ها برای به روز رسانی میانگین حالت پیش بینی شده $s_{t|t-1}^{[m]}$ و کواریانس آن $P_{t|t-1}^{[m]}$ به صورت زیر به کار گرفته می شوند:

$$s_{t|t}^{[m]} = s_{t|t-1}^{[m]} + K_t^{[m]}(z_t - \bar{z}_t^{[m]}) \quad (۲۱)$$

$$P_{t|t}^{[m]} = P_{t|t-1}^{[m]} - K_t^{[m]} P_{\delta v}^{[m]} (K_t^{[m]})^T$$

- نمونه برداری
- محاسبه وزن های پراهمیت
- به روز رسانی موقعیت نشانه ها
- نمونه برداری مجدد^۶

۲-۱- نمونه برداری

در این روش، حالت های ربات از تابع توزیع پیشنهادی به صورت زیر نمونه برداری می شوند:

$$s_t \sim q(s_t | s^{t-1,[m]}, z^t, u^t, n^t) \quad (۹)$$

همچنین، هر ذره در زمان اندازه گیری با استفاده از UKF بروز می شود. با فرض اینکه نویز فرآیند صفر باشد، میانگین و کواریانس بردار حالت افزوده شده به ترتیب زیر است:

$$s_{t-1|t-1}^{a[m]} = \begin{bmatrix} s_{t-1|t-1}^{[m]} \\ 0 \end{bmatrix}, \quad P_{t-1|t-1}^{a[m]} = \begin{bmatrix} P_{t-1|t-1}^{[m]} & 0 \\ 0 & Q_t \end{bmatrix} \quad (۱۰)$$

که $s_{t-1|t-1}^{a[m]}$ بردار افزوده شده، Q_t کواریانس نویز فرآیند، و $s_{t-1|t-1}^{[m]}$ و $P_{t-1|t-1}^{[m]}$ به ترتیب میانگین و کواریانس پیشین حالت ربات است. بنابراین فاکتورگیری چلوسکی^۷ $P_{t-1|t-1}^{[m]}$ به صورت زیر است:

$$S_{t-1|t-1}^{a[m]} = \text{chol}(P_{t-1|t-1}^{a[m]}) \quad (۱۱)$$

برای پیش بینی یک مجموعه متقارن از $2n+1$ نقاط سیگما برای بردار حالت افزوده شده به صورت زیر تشکیل می شوند:

$$\delta_{t-1|t-1}^{a[0][m]} = s_{t-1|t-1}^{a[m]} \quad (۱۲)$$

$$\delta_{t-1|t-1}^{a[i][m]} = s_{t-1|t-1}^{a[m]} + \gamma (S_{t-1|t-1}^{a[m]})_i, i=1, \dots, n$$

$$\delta_{t-1|t-1}^{a[i][m]} = s_{t-1|t-1}^{a[m]} - \gamma (S_{t-1|t-1}^{a[m]})_i, i=n+1, \dots, 2n$$

که پارامترهای γ و λ به صورت $\lambda = n(\alpha^2 - 1)$ و $\gamma = \sqrt{n+\lambda}$ و ثابت α معمولاً یک مقدار مثبت کوچک است. عبارت $(S_{t-1|t-1}^{a[m]})_i$ است از i امین ستون $S_{t-1|t-1}^{a[m]}$ است. هر نقطه سیگما $\delta_{t-1|t-1}^{a[i][m]}$ شامل مولفه های حالت و نویز کنترل به صورت زیر است:

$$\delta_{t-1|t-1}^{a[i][m]} = \begin{bmatrix} \delta_{t-1|t-1}^{[i][m]} \\ \delta_t^{u[i][m]} \end{bmatrix}^T \quad (۱۳)$$

مجموعه نقاط سیگمای $\delta_{t-1|t-1}^{[i][m]}$ به صورت زیر از مدل حرکت ربات انتقال داده می شوند:

$$\bar{s}_{t|t-1}^{[i][m]} = f_v(\delta_{t-1|t-1}^{[i][m]}, u_t + \delta_t^{u[i][m]}) \quad (۱۴)$$

که Z_t اندازه‌گیری واقعی است. از تابع توزیع گوسین تولیدشده به‌وسیله میانگین و کواریانس وسیله، حالت‌های هر ذره به‌صورت زیر نمونه‌برداری می‌شوند:

۲-۲-۲- بروز رسانی نشانه‌ها

$$\begin{aligned}\chi^{[0][m]} &= \mu_{n_t, t-1}^{[m]} a \\ \chi^{[i][m]} &= \mu_{n_t, t-1}^{[m]} + (\sqrt{(n+\lambda)\Sigma_{n_t, t-1}^{[m]}})_i, \quad i=1, \dots, n \\ \chi^{[i][m]} &= \mu_{n_t, t-1}^{[m]} - (\sqrt{(n+\lambda)\Sigma_{n_t, t-1}^{[m]}})_i, \quad i=n+1, \dots, 2n\end{aligned}$$
$$\begin{aligned}\bar{Z}_t^{[i||m]} &= h(\chi^{[i||m]}, s_{t|t}^{[m]}), i=0, \dots, 2n \\ \hat{z}_t^{[m]} &= \sum_{i=0}^{2n} \omega_g^{[i]} \bar{Z}_t^{[i||m]} \\ \bar{S}_t^{[m]} &= \sum_{i=0}^{2n} \omega_c^{[i]} (\bar{Z}_t^{[i||m]} - \hat{z}_t^{[m]})(\bar{Z}_t^{[i||m]} - \hat{z}_t^{[m]})^T + R_t \\ \bar{\Sigma}_t^{[m]} &= \sum_{i=0}^{2n} \omega_c^{[i]} (\chi^{[i||m]} - \mu_{n_t, t-1}^{[m]})(\bar{Z}_t^{[i||m]} - \hat{z}_t^{[m]})^T \\ \bar{K}_t^{[m]} &= \bar{\Sigma}_t^{[m]} (\bar{S}_t^{[m]})^{-1}\end{aligned}$$
$$\begin{aligned}\mu_{n_t,t}^{[m]} &= \mu_{n_t,t-1}^{[m]} + \bar{K}_t^{[m]}(z_t - \hat{z}_t^{[m]}) \\ \Sigma_{n_t,t}^{[m]} &= \Sigma_{n_t,t-1}^{[m]} - \bar{K}_t^{[m]} \bar{S}_t^{[m]} (\bar{K}_t^{[m]})^T\end{aligned}$$

وزن پر اهمیت ذرات با توجه به آخرین اندازه‌گیری جاری به‌صورت زیر محاسبه می‌شود:

$$w_t^{[m]} = w_{t-1}^{[m]} \frac{p(z_t | s_t^{[m]})p(s_t^{[m]} | s_{t-1}^{[m]}, u_t)}{N(s_{t:t}^*; \mathbf{s}_{t:t}^{[m]}, \mathbf{P}_{t:t}^{[m]})} \quad (23)$$

تفاوت روش بهبود یافته با UFastSLAM در تطبیق مشخصات آماری نویزها و گام اصلاح نمونه‌ها است که در ادامه توضیح داده می‌شود.

از آنجایی که نشانه‌ها ثابت هستند، کواریانس نویز فرآیند Q_t در دینامیک نشانه‌ها کاملاً معلوم است. بنابراین الگوریتم تطبیقی برای ماتریس کواریانس اندازه‌گیری R_t را در هنگام به‌روزرسانی نشانه‌ها استخراج کرد. برای این منظور الگوریتم از تکنیک تطبیق کواریانس^{۱۰} است. ایده اصلی این تکنیک این است که مقدار واقعی کواریانس باقی‌مانده^{۱۱} یا توالی تغییر با مقدار تئوری آن سازگار باشد. اگر اندازه‌گیری Z_t و اندازه‌گیری پیش‌بینی شده $\hat{Z}_{n_t,t}$ باشد، باقی‌مانده به‌صورت زیر تعریف می‌شود:

که $\hat{z}_i$ به صورت زیر است:

توالی تغییر اطلاعات اضافی قابل دسترس برای فیلتر به عنوان یک توالی از اندازه گیری جدید z_t ارائه می کند. توالی وزن داده شده، به عنوان یک تصحیح برای تخمین پیش بینی شده $\mu_{t,t-1}^{[m]}$ برای تشکیل تخمین $\mu_{t,t}^{[m]}$ عمل می کند. کواریانس ثنوری توالی تغییر S_t به صورت زیر است:

که $\bar{Z}_t^{[i]}$ به صورت زیر است:

کواریانس واقعی باقی‌مانده $\hat{C}_k$ به‌وسیله کواریانس نمونه‌هایش از طریق متوسط‌گیری درون یک پنجره تخمین متحرک با اندازه N_w به‌صورت زیر تخمین زده می‌شود:

$$\hat{C}_t = \frac{1}{N} \sum_{i=k-N+1}^t (r_i^T r_i) \quad (28)$$

$$W_{t+1} = W_t - \eta \frac{\partial E}{\partial W_t} \quad (34)$$

که η و W به ترتیب نرخ آموزش و پارامتر تنظیم ANFIS را ارائه می کنند. با فرض اینکه $W = [m, \sigma, w]^T$ بردار وزن ANFIS باشد، گرادیان E نسبت به یک بردار وزن دلخواه W به صورت زیر است:

$$\frac{\partial E}{\partial W_t} = -e_t \frac{\partial \nabla R_t}{\partial W_t} \quad (35)$$

با کاربرد بازگشتی قانون زنجیری، ترم خطا برای هر لایه ابتدا محاسبه می شود و سپس پارامترهای متناظر با لایه ها تنظیم می شوند.

۳-۲-۳- اصلاح نمونه ها

در حالت کلی، یک خاصیت SLAM مبتنی بر فیلتر ذره ای این است که واریانس وزن نمونه ها در طول زمان افزایش می یابد. برای حل این مشکل نمونه برداری مجدد انجام می شود. مشهورترین الگوریتم های نمونه برداری مجدد عبارت از چند جمله ای^{۱۳}، طبقه طبقه شده^{۱۴} سیستماتیک^{۱۵} باقی مانده (PR)^{۱۶} هستند [۲۶]. این الگوریتم ها با توجه به وزن ذرات تصمیم می گیرند که چه تعداد نسخه از هر ذره بسازد. این تکنیک در بسیاری از تحقیقات استفاده شده است [۲۶]. اگرچه این تکنیک ها از عملکرد خوبی برخوردار هستند، آن ها موجب از دست دادن تنوع میان ذرات می شوند.

در این مقاله به جای حذف و تکثیر نمونه ها، ذرات اصلاح می شوند و در موقعیت های مناسبی ذرات قرار می گیرند. برای این منظور، ذرات به گونه ای حرکت داده می شوند که تابع چگالی احتمال پسین ماکزیمم شود. برای رسیدن به این هدف، یک تابع هزینه به صورت زیر در نظر گرفته می شود:

$$\arg \max_s p(s^t | z^t, u^t, n^t) \quad (36)$$

با به کار بردن قانون بیز، پسین روی مسیر ربات به صورت زیر قابل تجزیه است:

$$p(s^t | z^t, u^t, n^t) = \eta p(z_t | s^t, z^{t-1}, u^t, n^t) p(s_t | s^{t-1}, z^{t-1}, u^t, n^t) p(s^{t-1} | z^{t-1}, u^{t-1}, n^{t-1}) \quad (37)$$

که η ثابت نرمال سازی است. در نتیجه، تابع هزینه به صورت زیر است:

$$\arg \max_s p(z_t | s^t, z^{t-1}, u^t, n^t) p(s_t | s^{t-1}, z^{t-1}, u^t, n^t) p_0 \quad (38)$$

با تعریف p_0 به صورت زیر:

$$p_0 = p(s^{t-1} | z^{t-1}, u^{t-1}, n^{t-1}) \quad (39)$$

تابع هزینه معادل است با:

که $i_0 = k - N + 1$ اولین نمونه درون پنجره تخمین است. اندازه پنجره N_w به گونه ای انتخاب می شود تا مشخصات آماری نرمی بدهد. برای پردازش تفاوت بین S_t و مقدار واقعی آن $\hat{C}_k$ ، متغیر درجه تطابق (DOM)^{۱۲} به صورت زیر تعریف می شود:

$$DOM_t = S_t - \hat{C}_t \quad (29)$$

اگر مقدار واقعی کواریانس $\hat{C}_t$ با مقدار تئوری آن اختلاف S_t داشته باشد، آنگاه المان های R_t بر اساس اندازه این اختلاف تنظیم می شوند. بنابراین می توان از تغییر R_t مطابق با مقدار DOM_t ، به منظور کاهش اختلاف بین S_t و $\hat{C}_t$ استفاده نمود. تطبیق المان (i, i) از R_t ، مطابق المان (i, i) از DOM_t است و قوانین کلی تطبیق آن به صورت زیر است:

۱. اگر $DOM_t(i, i) \cong 0$ ، آنگاه R_t بدون تغییر باقی می ماند.

۲. اگر $DOM_t(i, i) > 0$ ، آنگاه R_t باید کاهش یابد.

۳. اگر $DOM_t(i, i) < 0$ ، آنگاه R_t باید افزایش یابد.

در اینجا ANFIS برای تنظیم R_t ارائه شده است. سیستم فازی تطبیقی یک بانک متشکل از دو سیستم است که هر زیرسیستم یک سیستم فازی تطبیقی دو ورودی یک خروجی به کار می برد.

سیستم های دو ورودی- تک خروجی ANFIS برای تنظیم هر المان قطری R_t استفاده شده است. ورودی های ANFIS عبارت از DOM_t و ∇DOM_t هستند که ∇DOM_t به صورت زیر تعریف می شود:

$$\nabla DOM_t = DOM_t - DOM_{t-1} \quad (30)$$

تطبیق R_t با استفاده از رابطه زیر انجام می شود:

$$R_t = R_t + \nabla R_t \quad (31)$$

که ∇R_t خروجی های ANFIS است. سیستم فازی تطبیقی یک شبکه پنج لایه است. برای فراهم کردن یک فهم روشن از ساختار ANFIS استفاده شده، فرض کنید $u_i^{(\ell)}$ و $f_i^{(\ell)}$ بیان کننده ورودی به خروجی از i امین نود ℓ امین لایه به ترتیب باشد. الگوریتم آموزش برای تنظیم وزن های شبکه از طریق مینیمم کردن تابع هزینه زیر است:

$$E = \frac{1}{2} e_t^2 \quad (32)$$

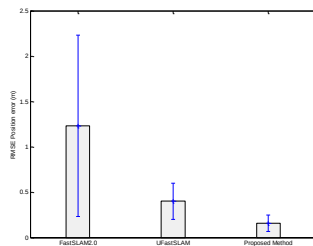
که

$$e_t = S_t - \hat{C}_t \quad (33)$$

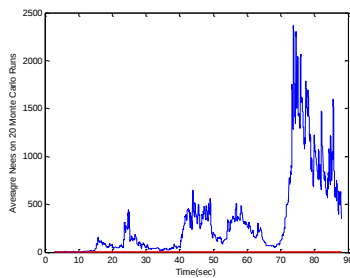
با استفاده از الگوریتم پس انتشار خطا (BP)، بردار وزن ANFIS به گونه ای تنظیم می شود که خطای تعریف شده در (۳۸) کمتر از مقدار آستانه تعریف شده بعد از چند سیکل آموزش داده شده باشد. الگوریتم BP به صورت زیر داده شده است:

Serial no. 83

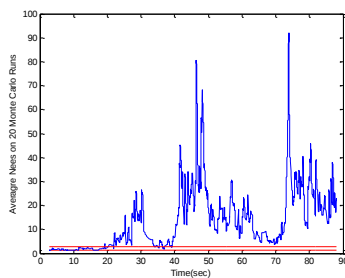
بهبود پیدا می‌کند جدید موجب افزایش تنوع میان ذرات و در نتیجه سازگاری و دقت می‌شود.



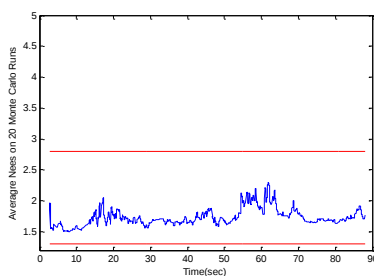
شکل ۴: RMSE



الف



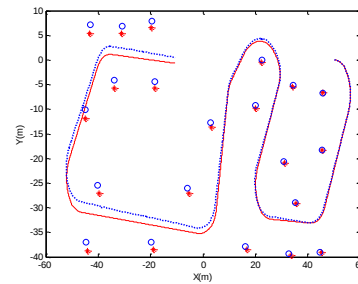
ب



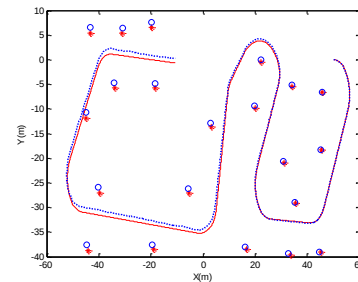
ج

شکل ۵: سازگاری (الف) FastSLAM2.0 (ب) UFastSLAM (ج) روش پیشنهادی

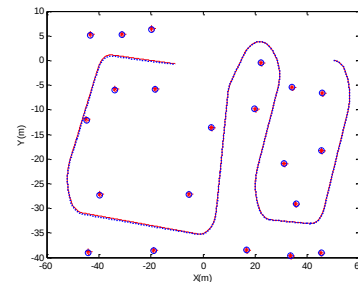
برای بررسی سازگاری، از متوسط نرمال‌شده مربع خطای تخمین (NEES) استفاده شده است. سازگاری به‌وسیله اجرای چندین بار مونت‌کارلو و محاسبه NEES ارزیابی می‌شود. شکل ۵ نشان می‌دهد که سازگاری روش پیشنهادی بیش‌تر از UFastSLAM و FastSLAM2.0



الف



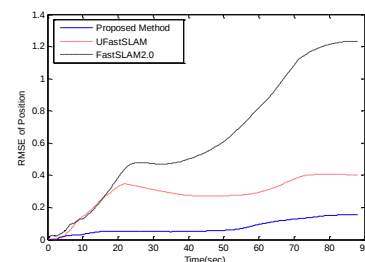
ب



ج

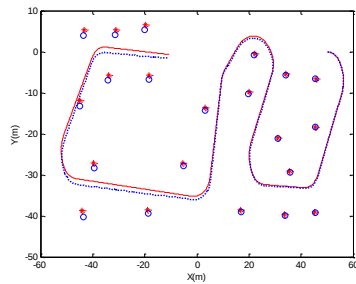
شکل ۲: مسیر و نقشه: (الف) UFastSLAM2.0 (ب) UFastSLAM

(ج) روش پیشنهادی

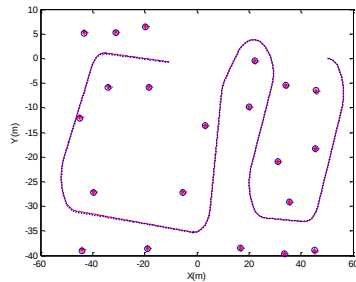


شکل ۳: جذر میانگین مربع خطا (RMSE) نسبت به زمان

مقدار RMSE موقعیت ربات روی زمان در شکل ۳ و میانگین و انحراف استاندارد آن در شکل ۴ نشان داده شده است. مشاهده می‌شود که RMSE روش پیشنهادی کم‌تر از FastSLAM2.0 و UFastSLAM است. این بدان دلیل است که اصلاح نمونه‌ها به‌وسیله PSO سبب بهبود توزیع نمونه‌ها و سرعت‌بخشیدن به همگرایی مجموعه ذرات می‌شود. بنابراین اثر هر ذره افزایش می‌یابد. در نتیجه تنوع میان ذرات



ب

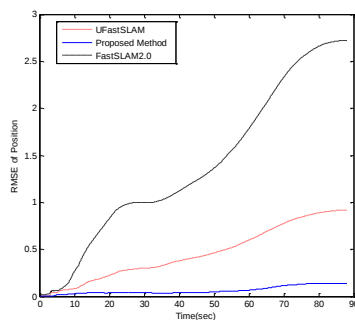


ج

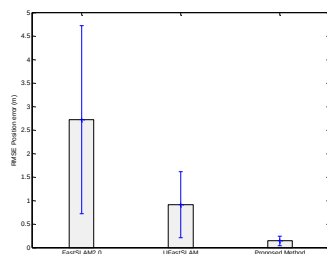
شکل ۷: مسیر و نقشه: الف) UFastSLAM2.0 ب) UFastSLAM

ج) روش پیشنهادی

همانند آزمایش قبلی سازگاری روش پیشنهادی بیشتر از UFastSLAM و FastSLAM2.0 است.

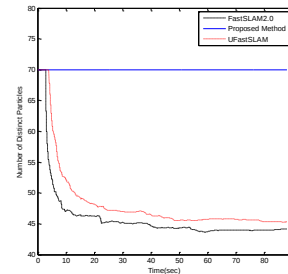


شکل ۸: RMSE نسبت به زمان



شکل ۹: RMSE

است. این بدان دلیل است که تنوع میان ذرات در روش پیشنهادی بیش‌تر از UFastSLAM و FastSLAM2.0 است. یک تخمین از نرخ از دست دادن تنوع میان ذرات با ثبت تعداد ذرات مجزا تخمین‌زده‌شده است.



شکل ۶: تعداد ذرات مجزا

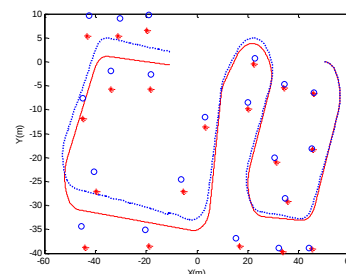
شکل ۶ تعداد ذرات مجزا که بعد از نمونه‌برداری مجدد شمارش شده‌اند برای الگوریتم‌ها نشان می‌دهد. ملاحظه می‌شود که تعداد ذرات مجزا در روش پیشنهادی بیش‌تر از UFastSLAM و FastSLAM2.0 است. در نتیجه روش پیشنهادی از حیث سازگاری نسبت به سایر الگوریتم‌ها برتری دارد.

۴-۲- عملکرد تحت نویزهای با مشخصه آماری نامعلوم

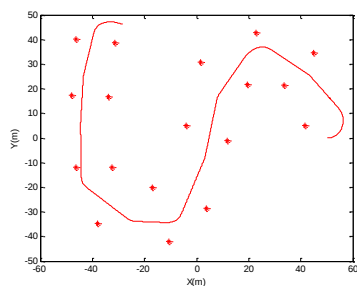
حال عملکرد الگوریتم‌ها را برای حالتی که نویز اندازه‌گیری به‌اشتباه به‌صورت $(\sigma_\theta = 2 \sigma_r = 0.05)$ تغییر داده شده، و نویز کنترل نیز ثابت $(\sigma_v = 0.2 \text{ m/s } \sigma_\gamma = 2^\circ)$ است مقایسه شده‌اند.

الگوریتم پیشنهادی R_t را از طریق ANFIS طوری تطبیق می‌کند که عدم‌تطابق بین مقدار تئوری و واقعی توالی تغییر را مینیمم کند. پارامترهای آزاد ANFIS به‌صورت اتوماتیک به‌وسیله گرایان نزولی آموزش داده می‌شود. درحالی‌که در UFastSLAM و FastSLAM2.0 ماتریس کواریانس نویز اندازه‌گیری در طول آزمایش ثابت است. از شکل‌های ۷ تا ۹ دیده می‌شود که مشابه حالت قبل عملکرد روش پیشنهادی بهتر از سایر الگوریتم‌ها است.

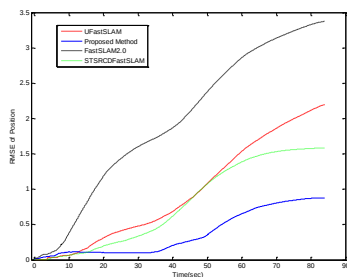
شکل ۱۰ سازگاری الگوریتم‌ها را در این وضعیت نشان می‌دهد.



الف

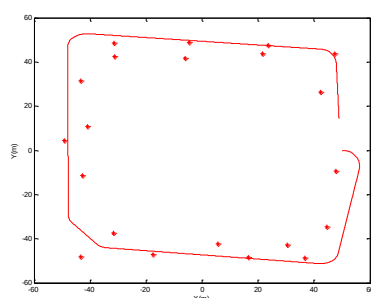


الف

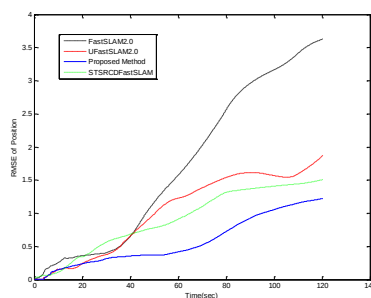


ب

شکل ۱۱: سناریوی اول الف) مسیر و نقشه ب) RMSE

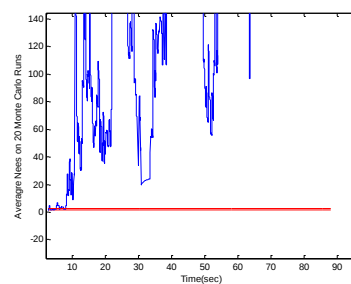


الف

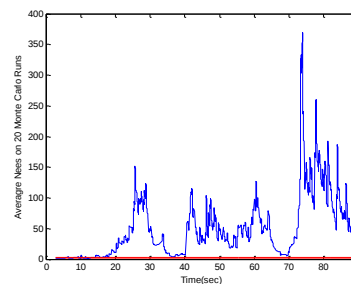


ب

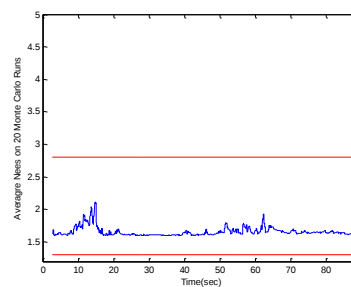
شکل ۱۲: سناریوی دوم الف) مسیر و نقشه ب) RMSE



الف



ب



ج

شکل ۱۰: سازگاری الف) UFastSLAM2.0 ب) UFastSLAM

ج) روش پیشنهادی

۴-۳- ارزیابی عملکرد تحت سناریوهای مختلف

برای ارزیابی بیشتر روش پیشنهادی، عملکرد آن با عملکرد الگوریتم‌های FastSLAM2.0 و UFastSLAM و STSRCDFastSLAM [۱۸] تحت دو سناریوهای مختلف دیگر مقایسه شده است. همان‌طور که از شکل‌های ۱۱ و ۱۲ ملاحظه می‌شود عملکرد الگوریتم پیشنهادی بر روی هر دو مسیر نسبت به سایر از الگوریتم‌ها برتری دارد. این نشان‌دهنده این است که عملکرد روش پیشنهادی تحت تأثیر مسیر نیست و همواره عملکرد آن از نقطه نظر دقت بهتر از سایر الگوریتم‌ها است.

۴-۳- هزینه محاسباتی

هزینه محاسباتی الگوریتم پیشنهادی با استفاده از شبیه‌سازی‌ها در Matlab تحلیل شده است. همان‌طور که در جدول ۱ نشان داده شده است زمان محاسبات الگوریتم پیشنهادی تقریباً نزدیک UFastSLAM است. مهم‌ترین امتیاز روش پیشنهادی از دیدگاه هزینه محاسباتی این است که برای به‌دست آوردن دقت تخمین یکسان با UFastSLAM تعداد ذرات کم‌تری لازم است. جدول ۱ نشان می‌دهد که وقتی روش پیشنهادی از ۳۰ استفاده می‌کند، خطای تخمین آن کم‌تر از UFastSLAM است که با ۷۰ ذره کار می‌کند. بنابراین روش پیشنهادی دقت بیش‌تر را با هزینه محاسباتی کم‌تر به‌دست می‌آورد.

جدول ۱: هزینه محاسباتی

Number of Particles	RMSE		Processing time	
	Proposed Method	UFastSLAM	Proposed Method	UFastSLAM
۷۰	۰.۱۵۱	۰.۴	۳۰۱	۲۲۳
۵۰	۰.۱۴۵	۱.۲	۲۱۸	۱۶۱
۳۰	۰.۱۳۵	۱.۹	۱۳۷	۱۰۱

۵- نتیجه‌گیری

الگوریتم UFastSLAM یک روش مؤثر برای موقعیت‌یابی و نقشه‌یابی هم‌زمان ربات است. این الگوریتم با به‌کاربردن تبدیل‌خنثی در الگوریتم FastSLAM به‌دست می‌آید. با وجود این، فرآیند نمونه‌برداری مجدد تنوع میان ذرات را کاهش می‌دهد و در نتیجه منجر به ناسازگاری می‌شود. به‌علاوه اطلاعات نامعلوم آماری نویز فرآیند و اندازه‌گیری به تخمین‌های ناسازگاری از موقعیت ربات و نقشه محیط در UFastSLAM منجر می‌شود. در این مقاله الگوریتم UFastSLAM بهبودیافته با استفاده از الگوریتم اجتماع ذرات و سیستم فازی-عصبی تطبیقی ارائه شده است. در این روش سیستم ANFIS به‌طور تطبیقی مشخصات آماری نویزها را تخمین می‌زند و الگوریتم اجتماع ذرات برای اصلاح نمونه‌ها استفاده شده است. الگوریتم بهبودیافته UFastSLAM تحت شرایط مختلف مورد ارزیابی قرار گرفته است. نتایج نشان می‌دهد که الگوریتم پیشنهادی از عملکرد بهتری نسبت به سایر الگوریتم‌ها از حیث دقت و سازگاری برخوردار است.

مراجع

- [2] T. Bailey, H. Durrant, "Simultaneous localization and mapping (SLAM): part II," *IEEE Trans. on Robotics & Automation Magazine*, 13(3): 108-117, 2006.
- [3] G. Dissanayake, P. Newman, S. Clark, H. Durrant-Whyte, and M. Csorba, "A solution to the simultaneous localization and map building (SLAM) problem," *IEEE Trans. Robot. Automat.*, vol. 17, no. 3, pp. 229-241, Jun. 2001.
- [4] S. D. Huang and G. Dissanayake, "Convergence and Consistency Analysis for Extended Kalman Filter Based SLAM," *IEEE Trans. Robot.*, vol. 23, no. 5, pp. 1036-1049, Oct. 2007.
- [5] T. Bailey, J. Nieto, J. Guivant, M. Stevens and E. Nebot, "Consistency of the EKF-SLAM Algorithm," in *Proc. IEEE Int. Conf. IROS 2006*, October 9 - 15, 2006, pp. 3562-3568.
- [۶] عبیری، عقیل، محزون، محمدرضا، «ردیابی اهداف متحرک هوایی با استفاده از تخمین چگالی کرنل بر اساس الگوریتم فیلتر ذره»، *مجله مهندسی برق دانشگاه تبریز*، شماره ۳، صفحه ۹۷-۱۰۷، تبریز، پاییز ۹۴.
- [7] L. Buonocore, S.R. Barros dos Santos, A.A. Neto, C.L. Nascimento, "FastSLAM Filter Implementation for Indoor Autonomous Robot," *IEEE Intelligent Vehicles Symposium (IV) Gothenburg, Sweden*, June 19-22, 2016.
- [8] Z. Xu, L. Jun, G. Wenjing, "FastSLAM Algorithm Based on Weight Optimal Compensation Extended Kalman Filter," in *Proc. of the 33rd Chinese Control Conference* July 28-30, Nanjing, China, 2014.
- [9] N. Kwak, G. W. Kim, and B. H. Lee, "A new compensation technique based on analysis of resampling process in FastSLAM," *Robotica*, Mar. 2008.
- [10] I. K. Kim, N. Kwak, H. C. Lee, and B. H. Lee, "Improved particle fusing geometric relation between particles in FastSLAM," *Robotica*, doi: 10.1017/S0263574708005250 (Online), Jan. 2009.
- [11] X. Wang and H. Zhang, "A UPF-UKF framework for SLAM," in *Proc. of IEEE Intl. Conf. on Robotics and Automation*, pp. 1664-1669, 2007.
- [12] L. Zhang, X. Meng and Y. Chen, "A FastSLAM Algorithm Based on the Auxiliary Particle Filter with stirring interpolation," in *Proc. of the IEEE, Int. Conf. Information and Automation*, June 22 -25, 2009, Zhuhai/Macau, China.
- [13] S. Julier, "The scaled unscented transformation," in *Proceedings of the IEEE American Control Conference*, May 2002.
- [14] C. Kim, R. Sakthivel and W. K. Chung, "Unscented FastSLAM: A robust and efficient solution to the SLAM problem," *IEEE Trans. Robot.*, vol. 24, no. 4, 2008.
- [15] C. Kim, R. Sakthivel and W. K. Chung, "Unscented FastSLAM: A robust algorithm for the simultaneous localization and mapping problem," in *Proc. IEEE Int. Conf. Robot. Autom.*, pp. 2439-2445, 2007.
- [16] C. Kim, H. Kim, and W. K. Chung, "Exactly Rao-Blackwellized Unscented Particle Filters for Autonomous SLAM," in *Proc. IEEE Int. Conf. on Robotics and Automation*, May 9-13, 2011, Shanghai, China.
- [17] F.P.X. Wu, H. Yan, "Distributed SLAM System Using Particle Swarm Optimized Particle Filter for Mobile Robot Navigation," in *Proc. of IEEE International Conference on Mechatronics and Automation* August 7 - 10, Harbin, China, 2016.
- [18] D.J. min, L. Dan, Y. Hong-xiao and S. Hui, "An Improved FastSLAM Algorithm for Autonomous Vehicle Based on the Strong Tracking Square Root Central Difference Kalman Filter," in *Proc IEEE 18th International Conference on Intelligent Transportation Systems*, 2015.
- [19] H. Chang, H. Zhang, X. Yang, W. Yang, C. Y. Chen, G. Yang, "An Improved FastSLAM Algorithm Based on An Omni-directional Wheeled Mobile Robot," in *Proc. IEEE Int. Conf. on Advanced Intelligent Mechatronics (AIM) Banff*, Alberta, Canada, July 12-15, 2016.

- [1] H. Durrant, T. Bailey, "Simultaneous localization and mapping: part I," *IEEE Trans. on Robotics & Automation Magazine*, 13(2): 99-110, 2006.

- capability estimation approach for lithium- ion battery,” *Journal of Power Sources*, vol.289, pp.50-62,2015.
- [25] E.Cao,K.Jiang, “Adaptive unscented Kalman filter for input estimations in Diesel-engine selective catalytic reduction systems,” *Neurocomputing*, vol.205,pp.329-335,2016.
- [26] C.Fen, M.Wang and Q.B.Ji, “Analysis and comparison of resampling algorithms in particle filter,” *Journal of System Simulation*, vol. 21, no.4, pp.1101-1105,2009.
- [۲۷] اباذری،سعید، مرادی،امید. «بهبود میرایی نوسانات سیستم قدرت با به کارگیری UPFC و تنظیم پارامترهای کنترل کننده بر اساس یک الگوریتم جدید PSO»، *مجله مهندسی برق دانشگاه تبریز*، شماره ۱، صفحه ۱-۱۱-۱۱، بهار ۹۵.
- [20] J.Zhang,Y.J.K.Wang, “A Modified FastSLAM for an Autonomous Mobile Robot,” in *Proc. of IEEE International Conference on Mechatronics and Automation* August 7-10, Harbin, China,2016.
- [21] R.J.Fitzgerald, “Divergence of the Kalman filter,” *IEEE Trans. Autom. Control*, vol. AC-16, no. 6, pp.736–747, 1971.
- [22] R.K.Mehra, “On the identification of variances and adaptive Kalman filtering,” *IEEE Trans. Automat. Control*.15, pp.175–184, 1970.
- [23] H.X.Le, S.Matunaga, “A residual based adaptive unscented Kalman filter for fault recovery in attitude determination system of microsatellites,” *Acta Astronautica* , vol.105 ,pp.30–39,2014.
- [24] W.Zhang, W.Shi, Z. Ma, “Adaptive unscented Kalman filter based state of energy and power

زیر نویس ها

- ¹ Unscented particle filter
- ² Unscented Kalman filters
- ³ Auxiliary particle filter
- ⁴ Unscented transformation
- ⁵ Inertial measurement unit
- ⁶ Resampling
- ⁷ Cholesky factor
- ⁸ Innovation- based adaptive estimation
- ⁹ Innovation sequence
- ¹⁰ Covariance-matching
- ¹¹ Residual
- ¹² Degree of matching
- ¹³ Multinomial resampling
- ¹⁴ Stratified resampling
- ¹⁵ Systematic resampling
- ¹⁶ Residual resampling