

موقعیت‌یابی ربات بر اساس فیلتر ذره‌ای بهبودیافته با فیلتر کالمن گروهی هوشمند و گام MCMC

رمضان هاونگی، استادیار

دانشکده مهندسی برق و کامپیوتر - دانشگاه بیرجند - بیرجند - ایران - havangi@birjand.ac.ir

چکیده: مسئله موقعیت‌یابی یکی از نیازهای ضروری برای ربات‌های خودمختار است. روش‌های مختلفی برای موقعیت‌یابی ارائه شده است که موقعیت‌یابی بر اساس فیلتر ذره‌ای یکی از مؤثرترین روش‌ها است. باوجوداین، این روش دارای مشکلاتی که مهم‌ترین آن‌ها عدم سازگاری، تباهیدگی و وابستگی به مشخصات آماری نویزها است. برای حل این مشکلات، در این مقاله، یک الگوریتم موقعیت‌یابی مبتنی بر فیلتر ذره‌ای بهبودیافته با فیلتر کالمن گروهی هوشمند (IEnKF) و گام مارکوف چابین مونت‌کارلو (MCMC) پیشنهاد شده است. در این روش، تابع توزیع پیشنهادی با استفاده از IEnKF ایجاد می‌شود که یک سیستم فازی-عصبی تطبیقی بر عملکرد آن نظارت دارد. به‌علاوه بعد از نمونه برداری مجدد، از گام MCMC برای افزایش تنوع میان ذرات استفاده شده است.

واژه‌های کلیدی: موقعیت‌یابی ربات، فیلتر ذره‌ای، فیلتر ذره‌ای گروهی، گام MCMC

Robot Localization based on Modified Particle Filter with Intelligent Ensemble Kalman Filter and MCMC Step

R. Havangi, Associate Professor

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

Abstract: The localization problem is a fundamental requirement for autonomous robots. Different methods are proposed for localization that particle filter based localization is most effective methods. However, this method has problems of inconsistency, degeneracy and dependence on statistics of noises. To solve these problems, in this paper, the modified particle filter based on localization with intelligent ensemble Kalman filter (IEnKF) and MCMC step is proposed. In this approach, the proposal distribution is created using IEnKF that an adaptive neuro-fuzzy supervised its performance. In addition, to increase diversity of particles after resampling, MCMC step is used.

Keywords: Robot localization, particle filter, ensemble kalman filter, MCMC step.

تاریخ ارسال مقاله: ۹۴/۹/۸

تاریخ اصلاح مقاله: ۹۴/۱۱/۳

تاریخ پذیرش مقاله: ۹۵/۱/۲۹

نام نویسنده مسئول: رمضان هاونگی

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

۱- مقدمه

علی‌رغم بهبودهای انجام‌شده بر روی موقعیت‌یابی مبتنی بر PF و موفقیت‌های آن‌ها، نتایج تحقیقات نشان می‌دهند که این الگوریتم هنوز مشکلات ناسازگاری و تباهی‌دگی را دارد [۱۶-۱۷].

در حالت کلی، یک خاصیت موقعیت‌یابی مبتنی بر PF این است که واریانس وزن نمونه‌ها در طول زمان افزایش می‌یابد و فیلتر تباهی‌ده می‌شود [۱۸-۱۹]. برای کاهش این مشکل، نمونه‌برداری مجدد^۱ بر اساس وزن ذرات انجام می‌شود [۱۸]. اگر چه نمونه‌برداری مجدد مسئله تباهی‌دگی را کاهش می‌دهد، اما مشکل دیگری را به وجود می‌آورد و سبب از بین رفتن تنوع میان ذرات می‌شود. با گذشت زمان تعداد ذرات مجزا و در نتیجه کواریانس نمونه‌ها کاهش می‌یابد و ناسازگاری زیاد می‌شود [۱۸-۱۹]. بنابراین به بهبود تنوع میان نمونه‌ها بعد از نمونه‌برداری مجدد بدون اینکه روی اعتبار تقریب تأثیر بگذارد نیاز است [۱۹-۲۰].

به‌علاوه موقعیت‌یابی مبتنی بر PF با این فرض است که اطلاعات آماری نوین فرآیند و اندازه‌گیری (ماتریس کواریانس‌های نوین فرآیند و اندازه‌گیری) معلوم باشند. تحت این شرایط فیلتر ذره‌ای تخمین‌های سازگاری را به دست می‌آورد [۲۶-۲۱]. با وجود این، در کاربردهای واقعی موقعیت‌یابی ربات این اطلاعات نامعلوم است و یا تخمین نادقیقی از آن‌ها در دسترس است. از طرف دیگر تخمین‌های ضعیف از مشخصات آماری نویزها ممکن است به‌طور جدی سبب پایین آمدن عملکرد موقعیت‌یابی و حتی ناپایداری آن شود.

برای بهبود این مشکلات، در این مقاله، یک الگوریتم موقعیت‌یابی مبتنی بر فیلتر ذره‌ای بهبودیافته (MPF)^۲ بر پایه فیلتر کالمن گروهی هوشمند و گام MCMC^۳ پیشنهاد شده است. در این روش، تابع توزیع پیشنهادی با استفاده از فیلتر کالمن گروهی (EnKF)^۴ ایجاد می‌شود که یک سیستم عصبی-فازی تطبیقی (ANFIS)^۵ بر عملکرد آن نظارت دارد. برای افزایش تنوع میان نمونه‌ها، از گام MCMC بعد از نمونه‌برداری مجدد استفاده شده است. به‌طور خلاصه مهم‌ترین نوآوری‌های این مقاله به شرح زیر است:

- استفاده از EnKF در طراحی تابع توزیع پیشنهادی
- بهبود سازگاری فیلتر ذره‌ای با استفاده سیستم عصبی-فازی تطبیقی
- استفاده از یک گام MCMC برای افزایش تنوع میان ذرات

ساختار بقیه مقاله به این شرح است: در بخش ۲ موقعیت‌یابی مبتنی بر فیلتر ذره‌ای به‌طور مختصر بررسی می‌شود. موقعیت‌یابی بر اساس فیلتر ذره‌ای بهبودیافته در بخش ۳ ارائه شده است. در بخش ۴ الگوریتم موقعیت‌یابی پیشنهادی تحت شرایط مختلف مورد ارزیابی قرار گرفته است. در بخش ۵ نتیجه‌گیری ارائه شده است.

کاربردهای زیادی وجود دارد که یک ربات متحرک خودمختار به جای انسان مأموریت‌هایی را انجام می‌دهد. برای اجرای چنین اهدافی به‌صورت اتوماتیک، مسئله موقعیت‌یابی ربات یک موضوع کلیدی است. در حقیقت، حل مسئله موقعیت‌یابی ربات پیش‌نیاز بسیاری از پیشرفت‌ها برای رسیدن به ربات‌های خودمختار و هوشمند است. از این‌رو در دهه اخیر، تلاش‌های زیادی برای ارائه راه‌حل‌های کارآمد برای حل این مسئله در بین محققان صورت گرفته و روش‌های مختلفی ارائه شده است.

مسئله موقعیت‌یابی ربات، تعیین موقعیت و جهت ربات با استفاده از اطلاعات اندازه‌گیری‌شده توسط سنسورهای نصب‌شده بر روی آن و نقشه محیط است [۲-۱]. دو روش کلی برای موقعیت‌یابی ربات وجود دارد: موقعیت‌یابی نسبی و مطلق. موقعیت‌یابی نسبی، تعیین موقعیت ربات با استفاده از سنسورهای داخلی (مانند انکدرها، شتاب‌سنج‌ها وژیروسکوپ‌ها) است. مهم‌ترین ضعف این روش افزایش خطای موقعیت‌یابی در طول زمان است. موقعیت‌یابی مطلق، تعیین موقعیت و وضعیت ربات با استفاده از سنسورهای خارجی نصب‌شده بر روی آن (مانند فاصله‌یاب‌لیزری، دوربین) و نقشه محیط است. به‌منظور بهره‌مندی از مزایای هر دو روش، معمولاً از ترکیب آن‌ها جهت موقعیت‌یابی ربات استفاده می‌شود [۲-۴]. نظر به اینکه مدل حرکت ربات (فرآیند) و اندازه‌گیری در مسئله موقعیت‌یابی ربات غیرخطی است دو روش موقعیت‌یابی وجود دارد: موقعیت‌یابی بر اساس فیلتر کالمن توسعه‌یافته (EKF)^۱ و موقعیت‌یابی بر اساس فیلتر ذره‌ای (PF)^۲. موقعیت‌یابی بر اساس EKF یکی از قدیمی‌ترین روش‌ها است که تاکنون نویسندگان زیادی بر روی آن تحقیق کرده‌اند [۷-۵]. علی‌رغم این، با توجه به خطی‌سازی معادلات غیرخطی حرکت ربات، موقعیت‌یابی بر اساس EKF از دقت خوبی برخوردار نبوده و امکان ناسازگاری^۳ و ناپایداری آن وجود دارد [۶-۷].

برای حل این مشکلات موقعیت‌یابی ربات بر اساس PF مطرح شده است [۸-۱۰]. در این روش چگالی احتمال پسین (PDF)^۴ با مجموعه‌ای از ذرات وزن داده‌شده تخمین زده می‌شود. تاکنون محققان زیادی بر روی این روش تحقیق و بهینه‌سازی‌هایی را بر روی آن انجام داده‌اند [۱۱-۱۵]. در [۱۳] از فیلتر ذره‌ای تطبیقی (APF)^۵ برای موقعیت‌یابی ربات استفاده شده است. در این روش تعداد ذرات به‌صورت تطبیقی تنظیم می‌شوند. در [۱۴] یک فیلتر ذره‌ای بهبودیافته برای موقعیت‌یابی ربات معرفی شده است. در این روش برای کاهش حجم محاسبات، گام نمونه‌برداری مجدد اصلاح شده است. برای این منظور، در روش نمونه‌برداری مجدد ارائه‌شده، ذراتی که دارای تابع چگالی احتمال کوچکی هستند حذف می‌شوند و بقیه ذرات حفظ می‌شوند. در [۱۶] از فیلتر ذره‌ای هوشمند برای موقعیت‌یابی ربات استفاده شده است. در این روش از یک سیستم فازی برای جلوگیری از انحراف و ناپایداری الگوریتم موقعیت‌یابی استفاده شده است.

۲- موقعیت‌یابی ربات بر اساس فیلتر ذره‌ای

هدف از موقعیت‌یابی ربات تعیین موقعیت و جهت آن در یک محیط با نقشه معلوم است [۱]. در واقع ربات با استفاده از اطلاعات مسافت پیمایی، فاصله یاب لیزری و نقشه، موقعیت و جهت خود را در هر لحظه تعیین می‌کند [۱-۲].

برای حل مسئله موقعیت‌یابی فرض می‌شود که محیط استاتیکی اطراف ربات را بتوان با یک بردار θ که توصیف‌کننده المان‌های نقشه است نشان داد. با این فرض، مدل استاندارد توصیف‌کننده مسئله موقعیت‌یابی ربات به صورت زیر است [۱-۲]:

$$\begin{aligned} x_t &= f(x_{t-1}, u_t, \omega_{t-1}) \leftrightarrow p(x_t | x_{t-1}, u_t) \\ y_t &= h(x_t, \theta) + v_t \leftrightarrow p(y_t | x_t) \end{aligned} \quad (1)$$

که x_t بردار حالت ربات (موقعیت و جهت ربات)، u_t بردار ورودی کنترل، y_t بردار اندازه‌گیری، f ، h توابع غیرخطی، v_t ، ω_{t-1} به ترتیب نویز فرآیند و اندازه‌گیری می‌باشند. از دیدگاه تئوری بیزین مسئله موقعیت‌یابی ربات، تخمین تابع چگالی احتمال پسین $p(x_{0:t} | y_{1:t})$ است. به عبارت دیگر مسئله موقعیت‌یابی، تخمین توزیع پسین حالت‌های ربات به شرط همه اندازه‌گیری‌های قابل دسترس است. در موقعیت‌یابی با فیلتر ذره‌ای تابع چگالی احتمال پسین موقعیت‌یابی $p(x_{0:t} | y_{1:t})$ برحسب مجموعه‌ای از ذرات وزن داده شده به صورت زیر بیان می‌شود [۸-۱۰]:

$$S_t = \{(x_t^i, w_t^i) | i = 1, \dots, N\} \quad (2)$$

در این حالت $p(x_{0:t} | y_{1:t})$ با مجموع وزن داده شده‌ای به صورت زیر تقریب زده می‌شود:

$$p(x_{0:t} | y_{1:t}) \approx \sum_{i=1}^N w_t^i \delta(x_t - x_t^i) \quad (3)$$

که $\delta(\cdot)$ تابع دلتای دایرک، $w_t^i > 0$ وزن مربوط به x_t^i و $\sum_{i=1}^N w_t^i = 1$ است. از آنجایی که نمونه‌برداری مستقیم از تابع چگالی احتمال اصلی $p(x_{0:t} | y_{1:t})$ که به تابع توزیع هدف معروف است امکان‌پذیر نمی‌باشد، از این‌رو از روش نمونه‌برداری پراهمیت^{۱۱} استفاده می‌شود [۱۸]. در روش نمونه‌برداری پراهمیت به جای نمونه‌برداری از تابع هدف از یک تابع توزیع پیشنهادی نمونه‌برداری می‌شود [۱۸]. در صورتی که تابع توزیع پیشنهادی را به شکل $q(x_{0:t} | y_{1:t})$ در نظر بگیریم وزن ذرات به صورت زیر محاسبه می‌شود:

$$w_t^i = \frac{p(x_{0:t} | y_{1:t})}{q(x_{0:t} | y_{1:t})} \quad (4)$$

برای محاسبه وزن‌ها به صورت بازگشتی، تابع چگالی پیشنهادی به شکل زیر تجزیه می‌شود [۱۸]:

$$q(x_{0:t} | y_{1:t}) = q(x_t | x_{0:t-1}, y_{1:t}) q(x_{0:t-1} | y_{1:t-1}) \quad (5)$$

در این صورت نمونه‌های $x_{0:t}^i$ از تابع چگالی پیشنهادی با افزودن نمونه‌های $x_{0:t-1}^i$ که از چگالی پیشنهادی $q(x_{0:t-1} | y_{1:t-1})$ نمونه‌برداری شده‌اند با حالت جدید x_t^i نمونه‌برداری شده از $q(x_t | x_{0:t-1}, y_{1:t})$ به دست می‌آیند. وزن این نمونه‌ها به صورت زیر قابل محاسبه است [۱۰-۱۲]:

$$w_t^i = \frac{p(x_{0:t}^i | y_{1:t})}{q(x_t^i | x_{0:t-1}^i, y_{1:t-1}) q(x_{0:t-1}^i | y_{1:t-1})} \quad (6)$$

از طرفی با توجه به قانون بیز تابع چگالی احتمال پسین را می‌توان به صورت روابط زیر نوشت:

$$\begin{aligned} p(x_{0:t} | y_{1:t}) &= \frac{p(y_t | x_{0:t}, y_{1:t-1}) p(x_{0:t} | y_{1:t-1})}{p(y_t | y_{1:t-1})} \\ &= \frac{p(y_t | x_t) p(x_t | x_{t-1}, u_t)}{p(y_t | y_{1:t-1})} p(x_{0:t-1} | y_{1:t-1}) \\ &\propto p(y_t | x_t, y_{1:t-1}) p(x_t | x_{t-1}, u_t) p(x_{0:t-1} | y_{1:t-1}) \end{aligned} \quad (7)$$

بنابراین داریم:

$$w_t^i = w_{t-1}^i \frac{p(y_t | x_t^i) p(x_t^i | x_{t-1}^i, u_t)}{q(x_t^i | x_{0:t-1}^i, y_{1:t})} \quad (8)$$

با توجه به اینکه تابع توزیع پیشنهادی به صورت زیر است:

$$q(x_t^i | x_{0:t-1}^i, y_{1:t}) = q(x_t^i | x_{t-1}^i, y_t) \quad (9)$$

معادله (۸) به شکل زیر بازنویسی می‌شود [۱۴-۱۶]:

$$w_t^i = w_{t-1}^i \frac{p(y_t | x_t^i) p(x_t^i | x_{t-1}^i, u_t)}{q(x_t^i | x_{t-1}^i, y_t)} \quad (10)$$

به طور خلاصه موقعیت‌یابی بر اساس فیلتر ذره‌ای دارای گام‌های زیر می‌باشد:

۱- نمونه‌برداری

نمونه‌برداری عبارت از تولید ذرات از توزیع پیشنهادی $q(x_t^i | x_{t-1}^i, y_t)$ است.

۲- محاسبه وزن نمونه‌ها

به هر ذره، یک وزن اختصاص داده می‌شود که به صورت زیر است:

$$w_t^i = w_{t-1}^i \frac{p(y_t | x_t^i) p(x_t^i | x_{t-1}^i, u_t)}{q(x_t^i | x_{t-1}^i, y_t)}$$

۳- نرمال‌سازی وزن‌ها

$$w_t^i = \frac{w_t^i}{\sum_{i=1}^N w_t^i}$$

۴- نمونه‌برداری مجدد

نمونه‌برداری مجدد، حذف ذرات با وزن ناچیز و تکثیر ذرات با وزن بالا است.

- ۱- بهبود نمونه برداری
- ۲- محاسبه وزن پراهمیت
- ۳- تخمین تطبیقی مشخصات آماری نویزها
- ۴- نمونه‌برداری مجدد با گام MCMC

۱-۳- بهبود نمونه‌برداری

انتخاب تابع توزیع پیشنهادی یکی از گام‌های مهم در طراحی الگوریتم موقعیت‌یابی مبتنی بر فیلتر ذره‌ای است. استراتژی عمومی این است که از انتقال حرکت^{۱۳} نمونه‌برداری شود. اما این استراتژی وقتی توزیع پیشین^{۱۳} از تابع درست‌نمایی^{۱۴} پهن‌تر باشد یا وقتی اندازه‌گیری‌ها در دم تابع توزیع پیشنهادی ظاهر شوند با شکست روبرو می‌شود. برای غلبه بر این مشکل باید اندازه‌گیری‌ها به داخل تابع توزیع پیشنهادی وارد می‌شوند. از این‌رو در الگوریتم موقعیت‌یابی با MPF تابع توزیع پیشنهادی به‌صورت زیر در نظر گرفته شده است:

$$x_t \sim p(x_t^i | x_{t-1}^i, y_t) \quad (12)$$

در این مقاله، از فیلتر کالمن گروهی [۲۷-۲۹] برای ایجاد تابع توزیع پیشنهادی استفاده شده است. فرض کنید N گروه اولیه فیلتر کالمن گروهی $\{x_0^{i,a}\}_{i=1}^N$ وجود داشته باشد. ابتدا گروه‌ها مطابق معادله حرکت ربات منتشر می‌شوند و گروه‌های پیش‌بینی شده $\{x_t^{i,f}\}_{i=1}^N$ به‌صورت زیر به دست می‌آیند:

$$x_t^{i,f} \sim f(x_{t-1}^{i,a}, u_t) + \omega_{t-1}^i \quad (13)$$

در ادامه گین کالمن به‌صورت زیر محاسبه می‌شود:

$$K_t^i = P_t^{i-} H_t^T (H_t P_t^{i-} H_t^T + R_t)^{-1} \quad (14)$$

که $P_t H^T$ و $H P_t H^T$ به‌صورت زیر است:

$$\bar{x}_t^f = \frac{1}{N} \sum_{i=1}^N x_t^{i,f}$$

$$P_t H^T = \frac{1}{N-1} \sum_{i=1}^N (x_t^{i,f} - \bar{x}_t^f)(h(x_t^{i,f}) - h(\bar{x}_t^f))^T$$

$$H P_t H^T = \frac{1}{N-1} \sum_{i=1}^N (h(x_t^{i,f}) - h(\bar{x}_t^f))(h(x_t^{i,f}) - h(\bar{x}_t^f))^T$$

ذرات به‌صورت زیر به‌روزرسانی می‌شوند:

$$\bar{x}_t^{i,a} = x_t^{i,f} + K_t^i ((y_t - h(x_t^{i,f})) + v_{i,t})$$

$$\bar{x}_t^a = \frac{1}{N} \sum_{i=1}^N x_t^{i,a} \quad (15)$$

سپس تابع توزیع پیشنهادی به‌صورت زیر ایجاد می‌شود:

$$p(x_t^i | x_{t-1}^i, y_t) = N(\bar{m}_t^i, P_t) \quad (16)$$

میانگین \bar{m}_t^i و کواریانس P_t به‌صورت زیر می‌باشند:

در طول زمان واریانس وزن‌های پر اهمیت ذرات افزایش می‌یابد و منجر به ایجاد پدیده‌ای بنام تباهدگی می‌شود. تباهدگی پدیده‌ای غیرقابل اجتناب است و در همه الگوریتم‌های فیلترهای ذره‌ای وجود دارد. در صورت رخ دادن پدیده تباهدگی ذراتی به‌روزرسانی می‌شوند که سهم آن‌ها در تقریب تابع چگالی احتمال پسین موقعیت‌یابی $p(x_{0:t} | y_{1:t})$ ، ناچیز است. یک شاخص مناسب برای تشخیص تباهدگی موقعیت‌یابی بر اساس فیلتر ذره‌ای، اندازه مؤثر نمونه \hat{N}_{eff} می‌باشد که به‌صورت زیر است [۱۸-۱۹]:

$$\hat{N}_{eff} = \frac{1}{\sum_{i=1}^N (w_t^i)^2} \quad (11)$$

هر وقت \hat{N}_{eff} کوچک‌تر از یک مقدار از پیش تعیین‌شده باشد، به‌منظور جلوگیری از پدیده تباهدگی، پروسه نمونه‌برداری مجدد اجرا می‌شود. نمونه‌برداری مجدد سبب می‌شود که ذرات با وزن کوچک حذف و ذرات با وزن بزرگ‌تر تکثیر شوند. در حقیقت نمونه‌برداری مجدد یک نگاهت از مجموعه $\{x_t^i, w_t^i\}$ به مجموعه $\{x_t^{*i}, \frac{1}{N}\}$ است که ذرات تصادفی مجموعه جدید $\{x_t^{*i}\}_{i=1}^N$ به‌وسیله نمونه‌برداری مجدد تولید شده‌اند.

همان‌طور که قبلاً هم گفته شد، نمونه‌برداری مجدد اگرچه تباهدگی را کاهش می‌دهد، سبب حذف ذرات خوب از مجموعه ذرات می‌شود و در بدترین حالت ممکن است فیلتر واگرا شود. به‌علاوه، هر زمان که نمونه‌برداری مجدد در موقعیت‌یابی مبتنی بر فیلتر ذره‌ای انجام می‌شود، تنوع میان ذرات مجزا کاهش می‌یابد. در نتیجه کواریانس بین ذرات کم می‌شود و ناسازگاری فیلتر افزایش می‌یابد. در حقیقت در این حالت کواریانس محاسبه‌شده با فیلتر با کواریانس واقعی متفاوت است و فیلتر ممکن است واگرا شود [۱۷-۱۹].

به‌علاوه در موقعیت‌یابی مبتنی بر فیلتر ذره‌ای فرض می‌شود که اطلاعات آماری نویزها معلوم باشد. در حقیقت برای محاسبه وزن ذرات و همچنین تولید نمونه‌ها از تابع توزیع پیشنهادی نیاز به کواریانس نویز پروسه و اندازه‌گیری است. در بیشتر کاربردهای واقعی و از جمله موقعیت‌یابی این اطلاعات نامعلوم هستند. از طرف دیگر استفاده از تخمین‌های ضعیف مشخصه‌های آماری نویزها به‌طور جدی سبب پایین آمدن دقت موقعیت‌یابی مبتنی بر PF می‌شود [۲۴-۲۶].

۳- موقعیت‌یابی با فیلتر ذره‌ای بهبودیافته

در این بخش برای غلبه بر مشکلات موقعیت‌یابی بر اساس فیلتر ذره‌ای، موقعیت‌یابی با MPF ارائه شده است. در این روش از فیلتر کالمن گروهی هوشمند و گام MCMC برای بهبود موقعیت‌یابی مبتنی بر PF استفاده شده است. موقعیت‌یابی با MPF دارای گام‌های اساسی زیر است:

اندازه‌گیری پیش‌بینی شده \hat{y}_t به صورت زیر است:

$$\hat{y}_t = h(\hat{x}_t^i) \quad (24)$$

باقی‌مانده، اطلاعات اضافی قابل دسترس را برای فیلتر فراهم می‌کند. در حقیقت، در صورتی که فیلتر بهینه عمل کند باقی‌مانده نویز سفید با میانگین صفر خواهد بود. مقدار تئوری کواریانس باقی‌مانده S_t به صورت زیر است:

$$S_t = \frac{1}{N-1} \sum_{i=1}^N (h(x_t^i) - h(\bar{x}_t)) (h(x_t^i) - h(\bar{x}_t)) + R_t \quad (25)$$

کواریانس واقعی باقی‌مانده \hat{C}_t به وسیله کواریانس نمونه‌هایش از طریق متوسط‌گیری درون یک پنجره تخمین متحرک با اندازه m به صورت زیر تخمین زده می‌شود:

$$\hat{C}_t = \frac{1}{m} \sum_{i=1}^m (r_i r_i^T) \quad (26)$$

اندازه پنجره m به صورت تجربی انتخاب می‌شود تا مشخصات آماری نرمی بدهد. برای پردازش تفاوت بین S_t و مقدار واقعی آن \hat{C}_t ، یک متغیر جدید تعریف می‌شود که این متغیر درجه تطابق (DOM) نامیده می‌شود و به صورت زیر است:

$$DOM_t = \frac{\text{trace}(\hat{C}_t)}{\text{trace}(S_t)} \quad (27)$$

که $\text{trace}(\cdot)$ بیان‌کننده trace ماتریس داخل پراکنش است. اگر مقدار واقعی کواریانس \hat{C}_t با مقدار تئوری آن اختلاف S_t داشته باشد، آنگاه المان‌های R_t بر اساس اندازه این اختلاف تنظیم می‌شوند. بنابراین می‌توان از تغییر R_t مطابق با مقدار DOM_t ، به منظور کاهش اختلاف بین S_t و \hat{C}_t استفاده نمود:

$$R_t = \alpha R_{t-1} \quad (28)$$

تطبیق المان (i, i) از R_t ، مطابق المان (i, i) ام از DOM_t است و قوانین کلی تطبیق آن به صورت زیر است:

۱. اگر $DOM_t(i, i) \geq 1$ ، آنگاه $\alpha = 1$ است.
۲. اگر $DOM_t(i, i) > 0$ ، آنگاه α باید کاهش یابد.
۳. اگر $DOM_t(i, i) < 0$ ، آنگاه α باید افزایش یابد.

از این رو می‌توان از یک سیستم فازی برای تنظیم R_t استفاده نمود. یکی از مشکلات سیستم فازی انتخاب بازه توابع عضویت است [۳۰]. در این مقاله برای رهایی از این مشکل، سیستم فازی-عصبی تطبیقی (ANFIS) برای تنظیم R_t استفاده شده است. مطابق شکل ۱، سیستم ANFIS انتخاب‌شده دارای دو ورودی و یک خروجی است. ورودی‌های ANFIS عبارت از DOM_t و ∇DOM_t هستند که ∇DOM_t به صورت زیر تعریف می‌شود:

$$\nabla DOM_t = DOM_t - DOM_{t-1} \quad (29)$$

$$\begin{aligned} \bar{m}_t^i &= f(x_{t-1}^{i,a}) + K_t^i (y_t - h(f(x_{t-1}^{i,a}))) \\ P_t &= \frac{1}{N-1} \sum_{i=1}^N (x_t^{i,a} - \bar{m}_t^i)(x_t^{i,a} - \bar{m}_t^i)^T \end{aligned} \quad (17)$$

در نهایت با استفاده از تابع توزیع گوسین تولیدشده، هر ذره به صورت زیر نمونه‌برداری می‌شود:

$$x_t \sim N(x_t; \bar{m}_t^i, P_t) \quad (18)$$

وقتی هیچ اندازه‌گیری وجود نداشته باشد، حالت‌های ربات بدون به‌روزرسانی با استفاده از (۱۲) پیش‌بینی خواهند شد.

۲-۳- محاسبه وزن پراهمیت ذرات

وزن پراهمیت ذرات با توجه به آخرین اندازه‌گیری جاری به صورت زیر محاسبه می‌شود:

$$w_t^i = w_{t-1}^i \frac{p(y_t | x_t^{i,a}) p(x_t^{i,a} | x_{t-1}^{i,a}, u_t)}{q(x_t^{i,a} | x_{t-1}^{i,a}, y_t)} \quad (19)$$

که تابع چگالی احتمال پیشین $p(x_t^{i,a} | x_{t-1}^{i,a})$ و تابع چگالی احتمال درست‌نمایی $p(y_t | x_t^{i,a})$ به صورت زیر محاسبه می‌شوند:

$$p(x_t^{i,a} | x_{t-1}^{i,a}, u_t) = N(f(x_{t-1}^{i,a}), Q_{t-1}) \quad (20)$$

$$p(y_t | x_t^{i,a}) = N(h(x_t^{i,a}), R_t) \quad (21)$$

نهایتاً وزن ذرات به صورت زیر محاسبه می‌شود:

$$w_t^i = w_{t-1}^i \frac{N(h(x_t^{i,a}), R_t) N(f(x_{t-1}^{i,a}), Q_{t-1})}{N(\bar{m}_t^i, P_t)} \quad (22)$$

۳-۳- تخمین تطبیقی مشخصات آماری نویزها

در موقعیت‌یابی مبتنی بر PF فرض بر این است که مشخصات آماری نویز فرآیند و اندازه‌گیری مشخص باشد. با وجود این، مشخصات آماری نویز پروسه و اندازه‌گیری در کاربردهای واقعی ربات نامشخص و حتی ممکن است با زمان تغییر کنند. اطلاعات آماری نادرست نویز فرآیند و اندازه‌گیری سبب کاهش عملکرد و حتی واگرایی می‌شوند [۲۳-۲۱]. یک راه مؤثر برای غلبه بر این ضعف استفاده از الگوریتم‌های تطبیقی است. مزیت استفاده از الگوریتم تطبیقی این است که موجب سازگار شدن کواریانس محاسبه‌شده فیلتر با کواریانس واقعی می‌شود [۲۴-۲۶]. تکنیک معروف برای این منظور، تطبیق کواریانس است که ایده اصلی آن این است که مقدار واقعی کواریانس باقی‌مانده (توالی تغییر) با مقدار تئوری آن سازگار باشد [۲۶-۲۵]. اگر اندازه‌گیری y_t و اندازه‌گیری پیش‌بینی شده \hat{y}_t باشد، باقی‌مانده به صورت زیر تعریف می‌شود:

$$r_t = y_t - \hat{y}_t \quad (23)$$

برای تنظیم وزن‌های شبکه w_i و پارامترهای توابع عضویت یعنی میانگین a و انحراف استاندارد c ، تابع هزینه زیر در نظر گرفته می‌شود:

$$E = \frac{1}{2} e_i^2 \quad (34)$$

که

$$e_i = S_i - \hat{C}_i$$

برای مینیمم کردن تابع هزینه، از الگوریتم پس‌انتشار (BP)^{۱۶} استفاده می‌شود. در حقیقت، بردار پارامترهای ANFIS (a, c, ω) به‌گونه‌ای تنظیم می‌شود که خطا از مقدار آستانه از قبل تعیین شده کوچک‌تر باشد. الگوریتم BP به‌روزرسانی پارامترها به‌صورت زیر است:

$$\begin{aligned} \omega_{i+1} &= \omega_i - \eta \frac{\partial E}{\partial \omega_i} \\ a_{i+1} &= a_i - \eta \frac{\partial E}{\partial a_i} \\ c_{i+1} &= c_i - \eta \frac{\partial E}{\partial c_i} \end{aligned} \quad (35)$$

که η نرخ آموزش و $\frac{\partial E}{\partial c_i}$ ، $\frac{\partial E}{\partial a_i}$ و $\frac{\partial E}{\partial \omega_i}$ به ترتیب گرادیان نسبت به ω_i ، a_i و c_i است.

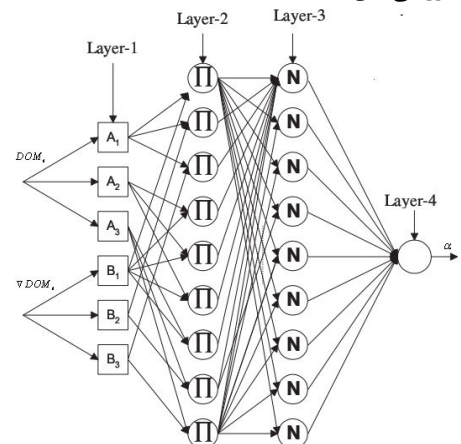
۳-۴- نمونه‌برداری مجدد با گام MCMC

در حوزه فیلتر ذره‌ای، الگوریتم‌های نمونه‌برداری مجدد مختلفی ارائه شده است که نمونه‌برداری مجدد چندجمله‌ای، باقی‌مانده (RR) و سیستماتیک باقی‌مانده (RSR) از جمله این الگوریتم‌ها هستند [۳۱-۳۲].

اگرچه نمونه‌برداری مجدد تباهیدگی موقعیت‌یابی بر اساس PF را کاهش می‌دهد، اما با گذشت زمان تعداد ذرات مجزا و در نتیجه واریانس نمونه‌ها کاهش می‌یابد و موجب ناسازگاری موقعیت‌یابی بر اساس PF می‌شود. از دست رفتن تنوع ذرات ناشی از این حقیقت است که در گام نمونه‌برداری مجدد، نمونه‌ها از یک توزیع گسسته به جای یک توزیع پیوسته تولید می‌شوند.

به منظور حفظ تنوع میان ذرات بعد از نمونه‌برداری مجدد، نیاز به ایجاد تنوع میان ذرات به گونه‌ای است که روی اعتبار تقریب تأثیر نگذارد. در این مقاله گام MCMC برای افزایش تنوع میان ذرات بعد از نمونه‌برداری مجدد استفاده شده است. برای توصیف MCMC، فرض کنید که ذرات مطابق پسین $p(x_{0,t} | y_{1:t})$ توزیع شده باشند. آنگاه با به کار بردن یک کرنل انتقال مارکوف چاین $\kappa(x_{0,t}^* | x_{0,t})$ با توزیع نامتغیر $p(x_{0,t} | y_{1:t})$ داریم:

تطبيق R_i با استفاده از رابطه (۲۸) انجام می‌شود که خروجی‌های ANFIS است.



شکل ۱: ساختار ANFIS با دو ورودی و نه قاعده

سیستم عصبی-فازی تطبیقی در نظر گرفته شده یک شبکه چهار لایه با ۹ قانون است. هر ورودی شبکه دارای سه تابع عضویت است که A_1 تا A_3 توابع عضویت ورودی اول و B_1 تا B_3 توابع عضویت ورودی دوم می‌باشند. با فرض اینکه $net_i^{(\ell)}$ و $O_i^{(\ell)}$ بیان‌کننده ورودی و خروجی i امین گره از ℓ امین لایه باشند، لایه‌های مختلف به‌صورت زیر است:

لایه اول: در این لایه، متغیر ورودی با استفاده از سه تابع عضویت گوسی فازی سازی می‌شود. خروجی تابع عضویت i ام به‌صورت زیر داده می‌شود:

$$o_i^{(1)} = \exp\left(-\left(\frac{net_i^{(1)} - c_i}{a_i}\right)^2\right) \quad (30)$$

که a_i میانگین و c_i انحراف استاندارد تابع عضویت گوسین است. لایه دوم: این لایه مجموعه قواعد فازی مقدمه را تشکیل می‌دهد. هر گره با یک مجموعه در فضای ورودی معادل است. یک گره قانون یک اپراتور AND فازی (استنتاج ضرب^{۱۰}) برای محاسبه قدرت آتش انجام می‌دهد. بنابراین توابع ورودی و خروجی گره قانون l ام به‌صورت زیر هستند:

$$o_l^{(2)} = \prod_i net_i^{(2)} \quad (31)$$

لایه سوم: گره‌های این لایه نرمال‌سازی قدرت آتش از لایه دوم را به‌صورت زیر انجام می‌دهند:

$$o_l^{(4)} = \frac{net_l^{(4)}}{\sum_{l=1}^9 net_l^{(4)}} \quad (32)$$

لایه چهارم: لایه چهارم، لایه خروجی است.

$$\alpha = \sum_{l=1}^9 o_l^{(4)} \omega_l \quad (33)$$

که w_i وزن‌های اتصال در این لایه است.

$$x_t^{*i} = \hat{x}_t^i + K_t^i (y_t - h(\hat{x}_t^i)) + v_t$$

$$\bar{u}_t = \frac{1}{N} \sum_{i=1}^N x_t^{*i}$$

$$V_t = \frac{1}{N-1} \sum_{i=1}^N (x_t^{*i} - \bar{u}_t)(x_t^{*i} - \bar{u}_t)^T$$

۴. نمونه‌برداری از تابع توزیع پیشنهادی

$$x_t \sim N(x_t; u_t, V_t)$$

۵. محاسبه احتمال قبولی

$$\gamma = \min \left\{ 1, \frac{p(y_t | x_t^{*i}) p(x_t^{*i} | \tilde{x}_{t-1}^i, u_t) q(\tilde{x}_{t-1}^i | \tilde{x}_{t-1}^i, y_t)}{p(y_t | \tilde{x}_t^i) p(\tilde{x}_t^i | \tilde{x}_{t-1}^i, u_t) q(x_t^{*i} | \tilde{x}_{t-1}^i, y_t)} \right\}$$

۶. قبول یا رد حرکت نمونه

If $u \leq \gamma$

قبول حرکت با احتمال γ به صورت زیر

$$x^{t,[m]} = (\tilde{x}^{t-1,[m]}, x_t^{*[m]})$$

$$P^{t,[m]} = (\tilde{P}^{t-1,[m]}, P_t^{*[m]})$$

Else

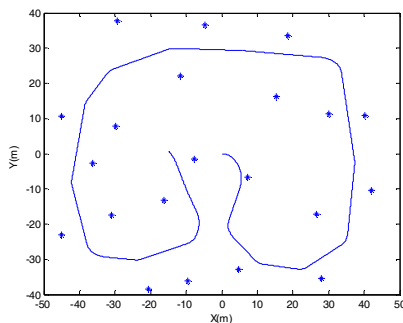
رد حرکت با احتمال $1-\gamma$ به صورت زیر:

$$x^{t,(i)} = \tilde{x}^{t,(i)}$$

$$P^{t,(i)} = \tilde{P}^{t,(i)}$$

end if

هنگامی که احتمال ذره جدید افزایش می‌یابد، احتمال قبولی نیز افزایش می‌یابد. در حقیقت اگر ذره قدیمی PDF پایین‌تری داشته باشد، ذره جدید جایگزین آن می‌شود.



شکل ۲: مسیر نامی ربات و نقشه محیط

۴- نتایج

حرکت يك ربات دوچرخه در نظر بگیرید. فرض می‌کنیم ربات روی مسیر شکل ۲ که در آن ربات تغییر مسیرهای زیادی داده است حرکت می‌کند. در این شکل مسیر واقعی ربات و نقشه محیط نشان داده شده است. نقشه‌هایی که برای بیان محیط به کار می‌روند دارای انواع مختلفی هستند که در اینجا از نقشه نشانه‌ای استفاده شده است [۱]. نقشه‌های نشانه‌ای محیط را با موقعیت‌های نشانه‌ها نشان می‌دهند [۱]. در شکل ۲ نقشه (موقعیت نشانه‌ها) با علامت * مشخص شده است.

$$\int \kappa(x_{0:t}^* | x_{0:t}) p(x_{0:t} | y_{0:t}) = p(x_{0:t}^* | y_{0:t}) \quad (36)$$

بعد از فرآیند انتقال از این کرنل، ذرات جدید هنوز مطابق $p(x_{0:t} | y_{1:t})$ توزیع شده‌اند. اما ذرات جدید مذکور ممکن است به سمت ناحیه‌ای با درست‌نمایی بالا حرکت کنند و کل واریانس توزیع جاری نسبت به توزیع نامتغیر تنها می‌تواند کاهش یابد. روش MH^{۱۷} یکی از عمومی‌ترین روش‌های پیاده‌سازی گام MCMC است [۳۱-۳۴]. الگوریتم MH يك توزیع شرطی را به‌عنوان تابع توزیع پیشنهادی برای ایجاد ماکوف چاین با توزیع نامتغیر به کار می‌برد. الگوریتم استاندارد MH به‌صورت زیر است:

۱. نمونه‌برداری $u \sim U_{[0,1]}$ که $U_{[0,1]}$ يك توزیع یکنواخت در بازه $[0,1]$ است.
۲. نمونه‌برداری از تابع توزیع پیشنهادی $x_t^{*i} \sim p(x_t | x_{t-1}^i, u_t)$
۳. اگر $u \leq \min \left\{ 1, \frac{p(y_t | x_t^{*i})}{p(y_t | \tilde{x}_t^i)} \right\}$

$$x_{0:t}^i = (\tilde{x}_{0:t-1}^i, x_t^{*i})$$

وگرنه

$$x_{0:t}^i = \tilde{x}_{0:t}^i$$

۴. پایان شرط.

در روش پیشنهادی، از الگوریتم MH روی هر ذره نمونه‌برداری مجدد شده $\tilde{x}^{t,i}$ با به کار بردن يك کرنل انتقال مارکوف با توزیع نامتغیر داده شده به وسیله $p(x_t^i | x_{t-1}^i, y_t)$ استفاده شده است. این الگوریتم در گام‌های زیر آورده شده است:

۱. نمونه‌برداری $u \sim U_{[0,1]}$
۲. پیش‌بینی ذرات با EnKF
- میانگین و کواریانس ذرات با استفاده از مدل حرکت ربات به‌صورت زیر پیش‌بینی می‌شوند:

$$x_t^{*i} \sim f(\tilde{x}_{t-1}^i, u_t) + \omega_{t-1}^i$$

$$\bar{x}_t^* = \frac{1}{N} \sum_{i=1}^N x_t^{*i}$$

سپس گین کالمن به‌صورت زیر محاسبه می‌شود:

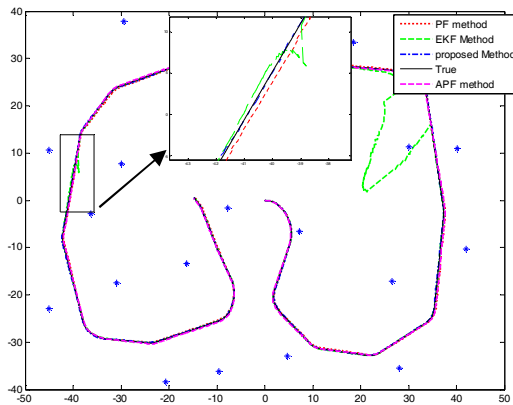
$$K_t^i = P_t^{*i-} H_t^{*T} (H_t^* P_t^{*i-} H_t^{*T} + R_t)^{-1}$$

$$P_t^* H^{*T} = \frac{1}{N-1} \sum_{i=1}^N (x_t^{*i} - \bar{x}_t^*) (h(x_t^{*i}) - h(\bar{x}_t^*))^T$$

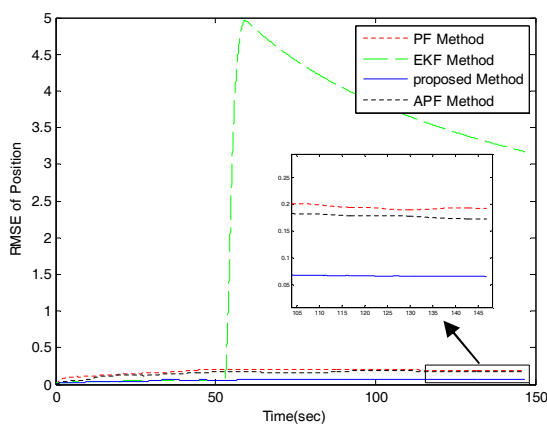
$$H^* P_t^* H^{*T} = \frac{1}{N-1} \sum_{i=1}^N (h(x_t^{*i}) - h(\bar{x}_t^*)) (h(x_t^{*i}) - h(\bar{x}_t^*))^T$$

۳. به‌روزرسانی ذرات با EnKF

پسین از حالت گوسی بودن منحرف شود. در نتیجه دقت تخمین‌های به‌دست‌آمده از EKF کاهش می‌یابد. همچنین ملاحظه می‌شود که عملکرد موقعیت‌یابی بر اساس APF اندکی بهتر از موقعیت‌یابی بر اساس PF است.



شکل ۳: مسیر واقعی و تخمین‌زده شده توسط الگوریتم‌ها



شکل ۴: RMSE موقعیت ربات در الگوریتم‌های مختلف

۲-۴- نويزها با مشخصات آماری نامعلوم

در این قسمت، عملکرد موقعیت‌یابی بر اساس MPF درحالی‌که مشخصات آماری نويزها مشخص نیست، مورد بررسی قرار می‌گیرد. برای این منظور فرض می‌شود که مشخصات آماری نويز فرآیند کاملاً نامعلوم و مشخصات آماری نويز اندازه‌گیری نامعلوم و به صورت $(\sigma_r = 0.1, \sigma_\theta = 0.1)$ باشد. نتایج به‌دست‌آمده از الگوریتم‌ها در شکل ۵ نشان داده شده است. ملاحظه می‌شود در این حالت، مشابه حالت قبل، عملکرد موقعیت‌یابی مبتنی بر MPF بهتر از سایر روش‌ها است. درحالی‌که عملکرد سایر الگوریتم‌ها نسبت به حالت قبل بدتر شده است. در موقعیت‌یابی با MPF، مسیر تخمین‌زده شده به مسیر واقعی نزدیک‌تر می‌باشد. بنابراین روش موقعیت‌یابی پیشنهادی نسبت به مشخصات آماری نامعلوم نويزها عملکرد مقاوم‌تری دارد.

موقعیت‌یابی با استخراج نشانه‌ها از داده‌های اندازه‌گیری شده و متناظر ساختن آن‌ها با نشانه‌های موجود در نقشه انجام می‌شود. از تفاوت بین موقعیت نشانه‌های تخمین‌زده شده و اندازه‌گیری شده، برای محاسبه موقعیت و وضعیت ربات استفاده می‌شود.

همچنین فرض شده است که ربات مجهز به فاصله یاب لیزری است. فاصله‌یاب لیزری در جلوی ربات نصب شده است و فاصله و زاویه سمت از نشانه مشاهده شده را اندازه‌گیری می‌کند. سرعت ربات ۳ m/s است و ماکزیمم زاویه فرمان 18° آن 30° درجه در نظر گرفته می‌شود. در این ربات، فاصله بین چرخ جلو و عقب ۴ متر است و ماکزیمم دامنه فاصله‌یاب لیزری ۳۰ متر و میدان دید آن 180° درجه است. فرکانس کنترل ۴۰ Hz و انحراف استاندارد نويز کنترل به صورت زیر است:

$$\sigma_v = 0.3m/s, \quad \sigma_\gamma = 3^\circ$$

فرکانس مشاهدات ۵ Hz و انحراف استاندارد نويز اندازه‌گیری در

فاصله و سمت به صورت زیر است:

$$\sigma_r = 0.1m, \quad \sigma_\theta = 1^\circ$$

برای ارزیابی موقعیت‌یابی ربات بر اساس روش پیشنهادی MPF، عملکرد آن با موقعیت‌یابی بر اساس PF، EKF و APF [۱۷] تحت شرایط مختلف مقایسه شده است.

۴-۱- نويزها با مشخصات آماری معلوم

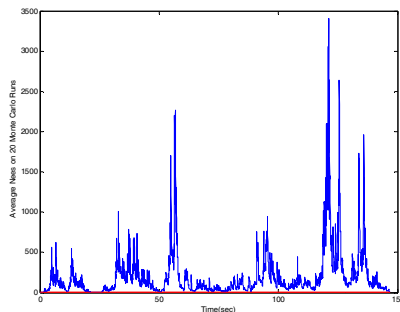
ابتدا عملکرد موقعیت‌یابی بر اساس روش پیشنهادی درحالی‌که مشخصات آماری نويز فرآیند و اندازه‌گیری با مقدار واقعی برابر است، مورد بررسی قرار می‌گیرد. شکل ۳ نتایج به‌دست‌آمده از موقعیت‌یابی بر اساس MPF، PF، APF [۱۷] و EKF را نشان می‌دهد. مشاهده می‌شود که نتایج عملکرد موقعیت‌یابی بر اساس MPF بهتر از سایر روش‌ها است. در موقعیت‌یابی با الگوریتم پیشنهادی، مسیر تخمین‌زده شده ربات تا حد امکان نزدیک به مسیر واقعی است. همان‌طور که از بزرگ‌شده بخشی از مسیر نشان داده شده در شکل ۳ ملاحظه می‌شود مسیر به‌دست‌آمده با الگوریتم پیشنهادی به مراتب بهتر از دیگر روش‌ها است.

برای ارزیابی دقیق‌تر عملکرد الگوریتم‌ها، جذر مربع خطا (RMSE) تخمین موقعیت به‌دست‌آمده از استفاده الگوریتم‌های موقعیت‌یابی مختلف در شکل ۴ نشان داده شده است. لازم به ذکر است که RMSE موقعیت از روش مونت‌کارلو با ۵۰ بار اجرا به دست آمده است. همچنین در موقعیت‌یابی بر اساس MPF، PF و APF تعداد ذرات $N=50$ انتخاب شده است. همان‌طور که از شکل مشاهده می‌شود عملکرد روش موقعیت‌یابی بر اساس MPF بر الگوریتم‌های دیگر برتری دارد. به دلیل غیرخطی بودن بالای معادله فرآیند و اندازه‌گیری، عملکرد موقعیت‌یابی بر اساس EKF نسبت به سایر الگوریتم‌های موقعیت‌یابی ضعیف‌تر است. غیرخطی بودن بالای معادلات حرکت ربات سبب می‌شود که تقریب خطی سیستم بد و در نتیجه تابع توزیع احتمال

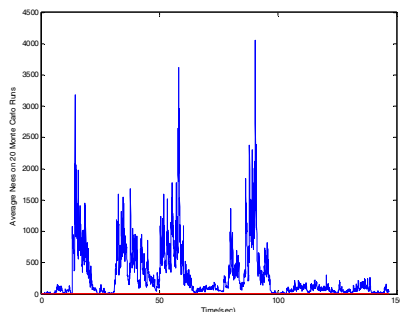
که x_t مقدار واقعی حالت ربات، \hat{x}_t مقدار تخمین‌زده شده حالت ربات و \hat{P}_t کواریانس خطای تخمین است. سازگاری فیلتر با چندین بار اجرا فیلتر و محاسبه متوسط NEES بررسی می‌شود. برای M بار اجرای مونت کارلو متوسط NEES به صورت زیر محاسبه می‌شود:

$$\bar{\varepsilon}_t = \frac{1}{M} \sum_{i=1}^M \varepsilon_{i,t} \quad (38)$$

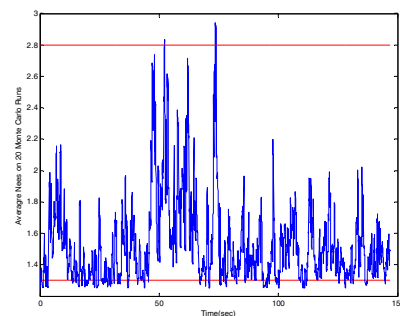
برای موقعیت دوبعدی وسیله، با ۲۰ شبیه‌سازی مونت کارلو، دو سمت احتمال ۹۵ درصد برای $\bar{\varepsilon}_t$ به بازه $[1/2, 3/79]$ محدود می‌شود. شکل‌های ۷ تا ۹ نشان می‌دهند که موقعیت‌یابی با MPF سازگار باقی می‌ماند، در حالی که NEES در موقعیت‌یابی با PF و APF ناسازگار است. این بدان دلیل است که تنوع ذرات در موقعیت‌یابی با MPF بیش‌تر است.



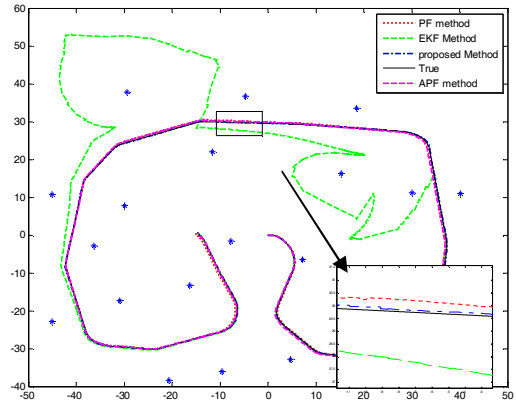
شکل ۷: سازگاری موقعیت‌یابی با PF



شکل ۸: سازگاری موقعیت‌یابی با APF

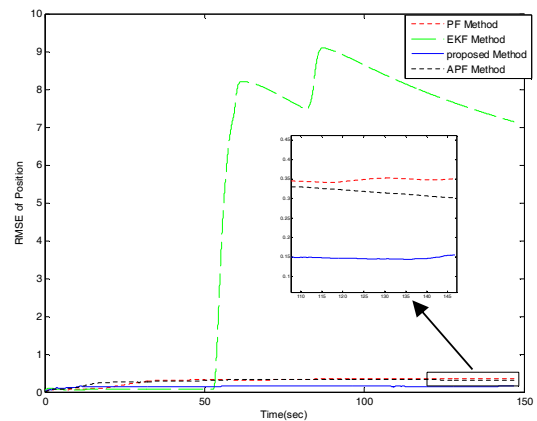


شکل ۹: سازگاری موقعیت‌یابی مبتنی بر MPF



شکل ۵: مسیر واقعی و تخمین‌زده شده توسط الگوریتم‌ها

در شکل ۶، RMSE تخمین موقعیت الگوریتم‌های مختلف نسبت به زمان نشان داده شده است. ملاحظه می‌شود عملکرد موقعیت‌یابی بر اساس سایر الگوریتم‌ها کاهش یافته در حالی که عملکرد موقعیت‌یابی بر اساس MPF به حالت قبلی آن نزدیک است. این بدان علت است که در موقعیت‌یابی با MPF مقدار کواریانس نویزها در طول زمان تخمین زده می‌شود در حالی که کواریانس نویزها در موقعیت‌یابی با EKF، PF و APF در طول زمان ثابت است.



شکل ۶: RMSE موقعیت ربات در الگوریتم‌های مختلف

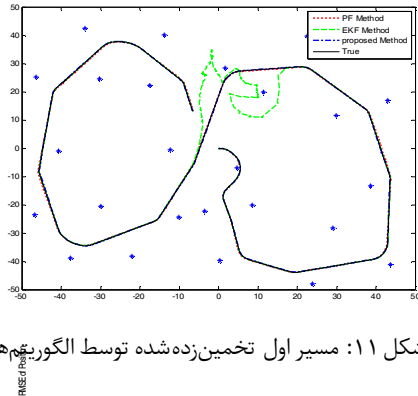
۳-۴- سازگاری

برای بررسی پایداری موقعیت‌یابی با MPF به بررسی سازگاری آن می‌پردازیم. مسئله سازگاری یکی از مسائل مهم در طراحی الگوریتم‌های موقعیت‌یابی است. انواع روش‌های تست سازگاری یک فیلتر در مرجع [۳۵] موجود است که برای مطالعه بیش‌تر می‌توان به آن‌ها مراجعه نمود.

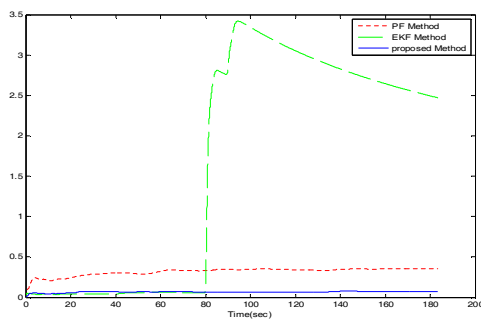
در این مقاله از تخمین نرمالیزه شده مربع خطا (NEES) برای تست سازگاری استفاده شده است. تخمین نرمالیزه شده مربع خطا به صورت زیر است [۳۵]:

$$\varepsilon_t = (x_t - \hat{x}_t)^T \hat{P}_t^{-1} (x_t - \hat{x}_t) \quad (39)$$

پیشنهادی در مقایسه با سایر الگوریتم‌ها نسبت به مسیرهای مختلف بررسی شود. برای این منظور ابتدا عملکرد الگوریتم‌ها روی یک محیط کوچک‌تر و سپس بر روی یک محیط بزرگ‌تر مورد بررسی قرار گرفته است. در هر دو مسیر فرض می‌شود که مشخصات آماری نویزها معلوم باشند.

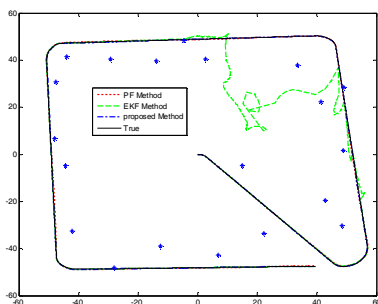


شکل ۱۱: مسیر اول تخمین‌زده شده توسط الگوریتم‌ها



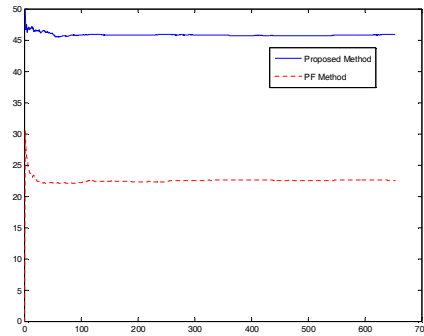
شکل ۱۲: RMSE موقعیت ربات در مسیر اول

شکل‌های ۱۱ و ۱۲ عملکرد موقعیت‌یابی مبتنی بر الگوریتم‌ها را بر روی مسیر اول و شکل‌های ۱۳ و ۱۴ عملکرد الگوریتم‌ها را بر روی مسیر دوم نشان می‌دهند. همان‌طور که ملاحظه می‌شود عملکرد موقعیت‌یابی مبتنی بر MPF در هر دو مسیر از نقطه نظر ردیابی و RMSE بهتر از سایر الگوریتم‌ها است. از آنجایی که مسیر دوم از مسیر اول بزرگ‌تر است عملکرد موقعیت‌یابی بر اساس EKF و PF روی مسیر دوم بدتر شده است در حالی که روش پیشنهادی در مسیر دوم هم به خوبی عمل کرده است.



شکل ۱۳: مسیر دوم تخمین‌زده شده توسط الگوریتم‌ها

یک تخمین از نرخ از دست رفتن تنوع میان ذرات با ثبت تعداد ذرات مجزا به دست می‌آید. همان‌طور که شکل ۱۰ نشان می‌دهد، تعداد ذرات مجزا در موقعیت‌یابی مبتنی بر MPF بیش‌تر از موقعیت‌یابی مبتنی بر PF است و بنابراین سازگاری روش پیشنهادی بیش‌تر است.



شکل ۱۰: تعداد ذرات مجزا

۴-۴- ذرات مختلف

برای ارزیابی تأثیر تعداد ذرات روی دقت الگوریتم موقعیت‌یابی بر اساس MPF و PF، عملکرد آن‌ها با تعداد ذرات مختلف با هم مقایسه شده است. در این حالت فرض شده است که مشخصات آماری نویز فرآیند کاملاً معلوم و مشخصات آماری نویز اندازه‌گیری نامعلوم و به صورت $(\sigma_r = 0.16, \sigma_\theta = 0.1)$ باشد.

جدول ۱: مقایسه الگوریتم‌ها با تعداد مختلف ذرات

Number of Particles	RMSE of Position			
	MPF		PF	
	Mean	Std	Mean	Std
۵۰	۰/۰۵۹	۰/۰۰۲۵	۰/۵۰	۰/۴۱
۳۰	۰/۰۶۱	۰/۰۰۴۱	۰/۶۹۱	۰/۶۴
۱۰	۰/۰۶۹	۰/۰۰۴۷	۰/۸۹۸	۰/۹۱

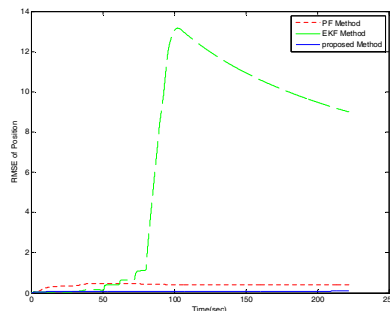
از جدول ۱ مشاهده می‌شود که RMSE موقعیت‌یابی مبتنی بر MPF وابستگی زیادی به تعداد ذرات ندارد. در حالی که RMSE موقعیت‌یابی با PF به تعداد ذرات وابستگی بیش‌تری دارد. این بدان علت است که در الگوریتم موقعیت‌یابی مبتنی بر MPF اولاً مشخصات آماری نویزها به‌طور تطبیقی توسط سیستم ANFIS تخمین زده می‌شوند. در نتیجه مشخصات آماری نویزها در طول زمان به سمت مشخصات واقعی میل می‌کنند. ثانیاً، تنوع ذرات بهبود یافته است و ذرات در مکان‌های مناسبی از تابع چگالی احتمال پسین قرار گرفته‌اند. بنابراین در روش پیشنهادی نیاز به ذرات زیادی نمی‌باشد.

۴-۵- محیط‌های مختلف

در این قسمت عملکرد الگوریتم‌ها روی محیط‌های مختلف مقایسه شده است. هدف از این آزمایش آن است که حساسیت عملکرد الگوریتم

- [6] T. T. Hoang, P. M. Duong, N. T. T. Van, D. A. Viet, and T. Q. Vinh, "Development of an EKF-based localization algorithm using compass sensor and LRF," *12th International Conference on Control, Automation, Robotics and Vision, ICARCV*, Guangzhou - China, 2012.
- [7] A. Shendre, and V. Angale, "Localization of a mobile autonomous robot using extended Kalman filter," *Third International Conference on Advances in Computing and Communications*, 2013.
- [8] F. Gustafsson, F. Gunnarsson, N. Bergman, and U. Forsell, "Particle filters for positioning, navigation and tracking," *IEEE Transactions on Signal Processing*, vol. 50, no. 2, pp. 425-437, 2005.
- [9] F. Gustafsson, F. Gunnarsson, N. Bergman, U. Forsell, J. Jansson, R. Karlsson, and P. J. Nordlund, "Particle filters for positioning, navigation and tracking," *IEEE Transactions Signal Processing*, vol. 50, no. 2, pp. 425-436, 2002.
- [10] S. A. Hiremath, G. W. A. M. van der Heijden, F. K. van Evert, A. Stein, C. J. F. ter Braak, "Laser range finder model for autonomous navigation of a robot in a maize field using a particle filter," *Computers and Electronics in Agriculture, Computers and Electronics in Agriculture*, vol. 100, pp. 41-50, 2014.
- [11] J. Gonzalez, J. L. Blanco, C. Galindo, A. Ortiz-de Galisteo, J. A. Fernandez-Madrigo, F. A. Moreno, and J. L. Martinez, "Mobile robot localization based on ultra-wide-band ranging: A particle filter approach," *Robotics and Autonomous Systems*, vol. 57, no. 5, pp. 496-507, 2009.
- [12] G. R. Gerasimos, "Nonlinear Kalman filters and particle filters for integrated navigation of unmanned aerial vehicles," *Robotics and Autonomous Systems*, vol. 60, no. 6, pp. 978-995, 2012.
- [13] D. Fox "KLD-sampling: Adaptive particle filters," *Advances in Neural Information Processing Systems*, 2001.
- [14] A. A. Wardhana, E. Clearesta, A. Widyotriatmo, et al., "Mobile robot localization using modified particle filter," *3rd International Conference on Instrumentation Control and Automation (ICA)*, Bali, Indonesia, 2013.
- [15] B. F. Wu, and C. L. Jen, "Particle-filter-based radio localization for mobile robots in the environments With Low-Density WLAN APs," *IEEE Transactions on Industrial Electronics*, vol. 61, no. 12, pp. 6860-6870, 2014.
- [16] G. Siamantas, T. Stouraitis, and A. Tzes "Intelligent particle filter based robot localization," *19th Mediterranean Conference on Control and Automation*, Aquis Corfu Holiday Palace, Corfu, Greece, 2011.
- [17] Y. Xia, and Y. Yang, "Mobile robot localization method based on adaptive particle filter," *C. Xiong et al. (Eds.): ICIRA 2008, Part I*, LNAI 5314, pp. 963-972, Springer-Verlag, Berlin, Heidelberg, 2008.
- [18] M. Sanjeev Arulampalam, S. Maskell, N. Gordon, and Tim Clapp, "A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking," *IEEE Transactions on Signal Processing*, vol. 50, no. 2, pp. 174-188, 2002.

[۱۹] عقیل عبیری و محمدرضا محزون، «ردیابی اهداف متحرک هوایی با استفاده از تخمین چگالی کرنل بر اساس الگوریتم



شکل ۱۴: RMSE موقعیت ربات در مسیر دوم

۵- نتیجه‌گیری

برای حل مشکلات موقعیت‌یابی ربات مبتنی بر فیلتر ذره‌ای، در این مقاله موقعیت‌یابی ربات بر اساس فیلتر ذره‌ای بهبودیافته ارائه شده است. در این روش، از فیلتر کالمن گروهی هوشمند برای ایجاد تابع توزیع پیشنهادی استفاده می‌شود که در آن سیستم عصبی-فازی تطبیقی بر عملکرد فیلتر کالمن گروهی نظارت می‌کند و مشخصات آماری نویزها را به‌طور تطبیقی تنظیم می‌کند. همچنین از گام MCMC برای ایجاد تنوع میان نمونه‌ها بعد از نمونه‌برداری مجدد استفاده شده است. عملکرد موقعیت‌یابی مبتنی بر MPF با سایر روش‌ها تحت شرایط مختلف مقایسه شده است. نتایج نشان می‌دهند که در همه شرایط عملکرد روش پیشنهادی بهتر از سایر الگوریتم‌ها است. به‌علاوه تحت شرایطی که مشخصات آماری نویزها نامعلوم باشند، روش پیشنهادی از مقاومت و دقت خوبی برخوردار است، درحالی‌که عملکرد موقعیت‌یابی بر اساس EKF و PF بدتر می‌شود. همچنین روش پیشنهادی برای به‌دست‌آوردن دقت برابر با موقعیت‌یابی بر اساس PF نیاز به تعداد ذرات کمتری دارد که این یکی دیگر از ویژگی مفید روش پیشنهادی است. د نتیجه روش پیشنهادی با تعداد ذرات کمتر به دقت بالاتری می‌رسد که این مزیت منجر به کاهش حجم محاسبات می‌شود.

مراجع

- [1] S. Thrun, W. Burgard, and D. Fox, *Probabilistic Robotics*, MIT Press, 2005.
- [2] F. Kong, Y. Chen, and J. Xie, Gang, "Mobile robot localization based on extended Kalman filter," *Proceedings of the 6th World Congress on Intelligent Control and Automation*, 2006.
- [3] S. Y. Chen, "Kalman filter for robot vision: A survey," *IEEE Transactions on Industrial Electronics*, vol. 59, no. 11, pp. 4409-4419, 2012.
- [4] M. Pinto, A. P. Moreira, and A. Matos, "Localization of mobile robots using an extended Kalman filter in a LEGO NXT," *IEEE Transactions on Education*, vol. 55, no. 1, pp. 135-144, 2012.
- [5] S. Yousefi, X. W. Chang, and B. Champagne, "Mobile localization in Non-line-of-sight using constrained square-root unscented Kalman filter," *IEEE Transactions on Vehicular Technology*, vol. 64, no. 5, pp. 2071-2083, 2014.

- [35] Y. Bar-Shalom, X. R. Li, and T. Kirubarajan, *Estimation with Applications to Tracking and Navigation*, John Wiley and Sons, 2001.
- [36] D. S. Frankel, *Model Driven Architecture: Applying MDA to Enterprise Computing*, OMG Press, Wiley Publishing, 2003.
- فیلتر ذره، «مجله مهندسی برق دانشگاه تبریز، دوره ۴۵، شماره ۳، صفحه ۱۰۷-۹۷، ۱۳۹۴.
- [20] S. Park, J. Hwang, E. Kim, and H. Kang, "A new evolutionary particle filter for the prevention of sample impoverishment," *IEEE Transactions on Evolutionary Computation*, vol. 13, no. 4, pp. 801- 809, 2009.
- [21] D. Zhuo-Hua, F. Ming, C. Zi-Xing, and Y. U. Jin-Xia, "An adaptive particle filter for mobile robot fault diagnosis," *Journal of Central South University*, 2006.
- [22] G. Cen, N. Matsuhira, J. Hirokawa, H. Ogawa, and I. Hagiwara, "Mobile robot global localization using particle filters," *International Conference on Control, Automation and Systems*, 2008.
- [23] A. Rahimi, K. D. Kumar, and H. Alighanbari, "Enhanced adaptive unscented Kalman filter for reaction wheels," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 51, no. 2, 2015.
- [24] W. Li, D. Gong, M. Liu, J. Chen, and D. Duan, "Adaptive robust Kalman filter for relative navigation using global position system," *IET Radar, Sonar & Navigation*, vol. 7, no. 5, pp. 471-479, 2013.
- [25] G. Xiangdong, Y. Deyong, and S. Katayama, "Seam tracking monitoring based on adaptive Kalman filter embedded Elman neural network during high-power fiber laser welding," *IEEE Transactions on Industrial Electronics*, vol. 59, no. 11, pp. 4315-4325, 2012.
- [26] S. Roujol, B. D. de Senneville, S. Hey, C. Moonen, and M. Ries, "Robust adaptive extended Kalman filtering for real-time MR-thermometry guided HIFU interventions," *IEEE Transactions on Medical Imaging*, vol. 31, no. 3, pp. 533-542, 2012.
- [27] M. D. Butala, R. A. Frazin, Y. Chen, and F. Kamalabadi, "Tomographic imaging of dynamic objects with the ensemble Kalman filter," *IEEE Transactions on Image Processing*, vol. 18, no. 7, pp. 1573-1587, 2009.
- [28] G. Burgers, P. J. van Leeuwen, and G. Evensen, "Analysis scheme in the ensemble Kalman filter," *Mon. Weather Rev.*, vol. 126, no. 6, pp. 1719-1724, 1998.
- [29] G. Evensen, *Data Assimilation: The Ensemble Kalman Filter*, Berlin, Germany, Springer-Verlag, 2007.
- [۳۰] امیر سیدموسوی و محمد فرخی «طراحی کنترل کننده فازی- تطبیقی برای کنترل ربات‌های با بازوهای انعطاف‌پذیر،» مجله مهندسی برق دانشگاه تبریز، دوره ۳۹، شماره ۲، صفحه ۶۰-۴۹، ۱۳۸۸.
- [31] M. Bolic, P. M. Djuric, and S. Hong, "Resampling algorithms and architectures for distributed particle filters," *IEEE Transactions, Signal Processing*, vol. 53, no. 7, pp. 442-2450, 2005.
- [32] 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.
- [33] C. Andrieu, N. De Freitas, A. Doucet, and M. I. Jordan, "An introduction to MCMC for machine learning," *Journal Machine Learning*, vol. 50, pp. 5-43, 2003.
- [34] J. Huang, and W. Hu, "MCMC-Particle-based group tracking of space objects within Bayesian framework," *Advances in Space Research*, vol. 53, pp 280-294, 2014.

زیرنویس‌ها

- ¹ Extended Kalman filter
² Particle filter
³ Consistency
⁴ Probability density function
⁵ Adaptive particle filter
⁶ Resampling
⁷ Modified particle filter
⁸ Markov chain Monte Carlo
⁹ Ensemble Kalman filter
¹⁰ Adaptive Neuro-Fuzzy inference system
¹¹ Important sampling
¹² Transition motion
¹³ Prior distribution
¹⁴ Likelihood function
¹⁵ Product inference
¹⁶ Back propagation
¹⁷ Metropolis-Hastings
¹⁸ Steering angle
¹⁹ Root-mean-square error
²⁰ Normalized estimation error squared