

## An Improved Particle Filter based on Soft Computing with Application in Target Tracking

Ramazan Havangi  
University of Birjand, Birjand, Iran.  
Havangi@Birjand.ac.ir

Corresponding author address: Ramazan Havangi, Faculty of Electrical and Computer Engineering, University of Birjand, Birjand, Iran, Post Code: 9717434765.

**Abstract-** Particle filter is one of the most important filters for estimating nonlinear/non-Gaussian systems that is used in many applications. In a standard particle filter, since the common state density function of the state is approximated by recursive importance sampling, the dimension of the joint posterior grows with each time step. This causes the algorithm to be rapidly degenerated. Therefore, the use of a resampling strategy is required for guaranteeing a logical approximation of the density function on the entire path. However, in practice, resampling step is performed on marginal space. Since the system may not exhibit an exponential forgetting behavior from its past errors, it will produce an incomplete estimate with a small number of finite sampling processes on the marginal space. To solve this problem, an improved particle filter based soft computing is proposed in this paper. Unlike a particle filter, this filter is applied to the marginal distribution, and the sampling dimensions do not increase with time. In addition, sampling has been improved using an evolutionary differential algorithm. The proposed method is evaluated using computer simulations. The results show that the proposed method has a better performance than standard particle filter

**Keywords-** Particle filter, Resampling step, Differential evolution algorithm (DE).



## فیلتر ذره‌ای بهبود یافته مبتنی بر محاسبات نرم با کاربرد در ردیابی هدف

رمضان هاونگی

دانشکده مهندسی برق و کامپیوتر، دانشگاه بیرجند، بیرجند، ایران.

Havangi@Birjand.ac.ir

\* نشانی نویسنده مسئول: رمضان هاونگی، بیرجند، دانشگاه بیرجند، دانشکده مهندسی برق و کامپیوتر، کد پستی: ۹۷۱۷۴۳۴۷۶۵

چکیده- فیلتر ذره‌ای یکی از مهمترین فیلترها در تخمین سیستم‌های غیر خطی/غیر گوسی که در کاربردهای زیادی استفاده شده است. در فیلتر ذره‌ای استاندارد، از آنجایی که تابع چگالی پسین مشترک حالت با استفاده از نمونه برداری پراهمیت بازگشتی تقریب زده می‌شود، ابعاد تابع چگالی پسین مشترک در هر لحظه از زمان رشد می‌کند. این موجب می‌شود که الگوریتم سریعاً "تباهیده شود. بنابراین استفاده از استراتژی نمونه برداری مجدد بمنظور تضمین یک تقریب منطقی از تابع چگالی احتمال پسین روی کل مسیر لازم می‌شود. با وجود این، در پیاده‌سازی فیلتر ذره‌ای، نمونه برداری مجدد روی فضای حاشیه‌ای انجام می‌شود. از آنجایی که سیستم ممکن است دارای رفتار فراموشی نمایی از خطاهای گذشته‌اش نباشد، با تعداد ذره محدود فرآیند نمونه برداری مجدد روی فضای حاشیه‌ای یک تخمین ناسازگار بوجود می‌آورد. برای رفع این مشکل، در این مقاله فیلتر ذره‌ای بهبود یافته مبتنی بر محاسبات نرم پیشنهاد شده است. در این فیلتر برخلاف فیلتر ذره‌ای، نمونه برداری بر روی توزیع حاشیه‌ای انجام می‌شود و ابعاد نمونه برداری با زمان افزایش نمی‌یابد. علاوه، نمونه برداری با استفاده از الگوریتم تکامل تفاضلی (DE) بهبود داده شده است. روش پیشنهادی با استفاده از شبیه‌سازی‌های کامپیوتری مورد ارزیابی قرار گرفته است. نتایج نشان می‌دهند روش پیشنهادی دارای عمل کرد بهتری نسبت به فیلتر ذره‌ای استاندارد است.

واژه‌های کلیدی: فیلتر ذره‌ای، نمونه برداری مجدد، الگوریتم تفاضل تکاملی

### ۱- مقدمه

تخمین تابع چگالی احتمال پسین<sup>۱</sup> است. با دانستن چگالی احتمال پسین می‌توان تخمین بهینه حالت‌ها را نسبت به هر تابع معیاری محاسبه نمود.

بسته به مدل فرآیند و اندازه‌گیری، روش‌های مختلفی برای پیاده‌سازی عملی فیلتر بیز وجود دارد. فیلتر کالمن عمومی‌ترین روش برای تخمین سیستم‌های خطی با نویز سفید است [۳-۴]. برای سیستم‌های غیرخطی و دارای نویز سفید گوسی از روش‌های زیر بهینه مانند فیلتر کالمن توسعه یافته، فیلتر کالمن بی‌رد می‌توان استفاده کرد [۵،۲]. در بسیاری از کاربردهای عملی با

مسئله تخمین حالت موضوع مهمی در بسیاری از کاربردها از جمله ردیابی اشیاء، موقعیت‌یابی و نقشه‌یابی، ناوبری و ... می‌باشد. مطالعه در زمینه تخمین سیستم‌های غیرخطی از حدود چهل سال قبل شروع شده است. هدف از مسئله تخمین پیدا کردن مقدار واقعی حالت‌های سیستم دینامیکی با استفاده از دنباله‌ای از اندازه‌گیری‌های نویزی است [۱-۲].

تخمین بازگشتی بیزین یک پایه برای بیشتر روش‌های فیلترینگ است. از دیدگاه تئوری بیزین، مسئله تخمین عبارت از

## Archive of SID

ذره‌ای جلوگیری شده است و بدین ترتیب سبب بزرگ شدن ناحیه جستجوی ذرات می‌شود [۱۴].

در حالت کلی، فرم‌های پایه فیلترهای ذره‌ای با یک نمونه از تابع پسین در زمان  $t-1$  شروع و پیش‌گویی حالت‌ها در لحظه  $t$  انجام می‌دهند. سپس بروز رسانی وزن‌های پراهمیت بر اساس اندازه‌گیری  $y_t$  صورت می‌گیرد. این نمونه‌ها یک تقریب از توزیع مشترک  $p(x_{1:t}/y_{1:t})$  در لحظه  $t$  تشکیل می‌دهند [۱۵ و ۱۶].

در همه نسخه‌های فیلتر ذره‌ای، از آنجایی که تابع چگالی پسین مشترک<sup>۸</sup> حالت با استفاده از نمونه‌برداری پراهمیت بازگشتی تقریب زده می‌شود، در فرم رسمی، فرآیند نمونه برداری بر اهمیت باید بر روی کل مسیر اجرا شود. با وجود این، در پیاده‌سازی فیلتر ذره‌ای، نمونه‌برداری مجدد روی فضای حاشیه‌ای انجام می‌شود که اگر سیستم نسبت خطاهای گذشته‌اش فراموشی نمایی نداشته باشد با تعداد ذره محدود یک تخمین ناسازگار بوجود می‌آورد.

در این مقاله یک فیلتر ذره‌ای حاشیه‌ای بهبود یافته ارائه شده است که سبب افزایش دقت، ایجاد تنوع میان ذرات و سازگاری می‌شود. فیلتر پیشنهادی بر روی توزیع حاشیه‌ای انجام می‌شود و ابعاد نمونه‌برداری با زمان افزایش نمی‌یابد. بعلاوه، نمونه‌برداری با استفاده از الگوریتم تفاضل تکاملی بهبود داده شده است. برتری استفاده از فیلتر ذره‌ای حاشیه‌ای تفاضلی نسبت به فیلتر ذره‌ای این است که عمل‌کرد آن در اصطلاح واریانس وزن پراهمیت، دقت تخمین و سازگاری برتری دارد.

ساختار بقیه مقاله به شرح زیر است. در بخش ۲ پیش‌زمینه‌های لازم ارائه می‌شوند. در بخش ۳ فیلتر ذره‌ای بهبود یافته معرفی شده است. در بخش ۴ روش پیشنهادی تحت شرایط مختلف مورد ارزیابی قرار گرفته است.

## ۲- پیش‌زمینه

## ۲-۱- فیلتر ذره‌ای

فیلتر ذره‌ای یک روش موثر برای حل مسئله تخمین در مواجهه با سیستم‌های غیرخطی با نویز غیرگوسی می‌باشد. این روش جزء روش‌های آماری مونت کارلو است. برای توصیف فیلتر ذره‌ای استاندارد، سیستم غیر خطی زیر را در نظر بگیرید:

$$x_t = f(x_{t-1}) + w_t \quad (1)$$

$$y_t = g(x_t) + v_t \quad (2)$$

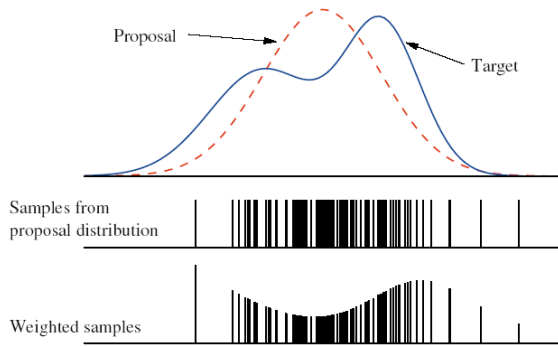
که در آن  $x_t$  و  $y_t$  به ترتیب متغیرهای حالت و اندازه‌گیری سیستم و  $f$  و  $g$  توابع غیرخطی می‌باشند. همچنین فرض می‌شود که  $w_t$  و  $v_t$  به ترتیب نویز پروسه و نویز اندازه‌گیری می‌باشند. فیلتر

سیستم‌های غیرخطی- غیرگوسی سرکار داریم که در آن فیلتر کالمن توسعه یافته بصورت بهینه عمل نمی‌کند [۷-۶، ۱]. برای تخمین چنین سیستم‌هایی از فیلتر ذره‌ای<sup>۲</sup> استفاده می‌شود. فیلتر ذره‌ای یک روش بازگشتی مونت-کارلو است که به‌طور وسیعی در تخمین سیستم‌های غیرخطی- غیرگوسی کاربرد دارد. در فیلتر ذره‌ای تابع چگالی احتمال پسین با مجموعه‌ای از ذرات وزن داده شده تخمین زده می‌شود [۶، ۱].

با وجود مزایای زیادی که فیلتر ذره‌ای در تخمین سیستم‌های غیرخطی و غیرگوسی دارد، دارای نقطه ضعف بزرگی است. در فیلتر ذره‌ای، حتی با انتخاب اولیه تعداد زیاد ذرات ممکن است هیچ ذره‌ای در نزدیکی حالت صحیح قرار نگیرد. این ضعف در متون مربوط به فیلتر ذره‌ای به مسئله تباهدگی<sup>۳</sup> معروف است. برای کاهش تباهدگی در فیلتر ذره‌ای استاندارد از نمونه برداری مجدد<sup>۴</sup> استفاده می‌شود [۶، ۱]. گام نمونه برداری مجدد ضمن اینکه برای فیلتر ذره‌ای حیاتی است، سبب پدیده دیگری بنام فقر نمونه‌ها می‌شود. در این حالت تنوع میان ذرات از بین می‌رود و در بدترین حالت، همه ذرات به یک نقطه از فضای حالت ریزش می‌کنند [۷].

تا کنون محققان نسخه‌های مختلفی را برای بهبود فیلتر ذره‌ای ارائه کرده‌اند که از جمله آنها می‌توان به فیلتر ذره‌ای کمکی (APF)<sup>۵</sup> [۸]، فیلتر ذره‌ای منظم شده (RPF)<sup>۶</sup> [۹]، فیلتر ذره‌ای بی‌رد [۱۰] اشاره نمود.

با توجه به خصوصیات مشترک روش‌های فرا ابتکاری با فیلتر ذره‌ای، این الگوریتم‌ها در بهبود فیلتر ذره‌ای در حوزه فیلترینگ از سابقه خوبی برخوردارند. به عنوان مثال با وارد کردن اپراتورهای الگوریتم ژنتیک در فیلتر ذره‌ای فیلتر ذره‌ای ژنتیک معرفی شده است [۱۱]. در [۱۲] برای حل مسئله تباهدگی در فیلتر ذره‌ای از استراتژی‌های تکاملی استفاده شده است. در [۱۳] برای حل مسئله فقر نمونه‌ها و وابستگی به تعداد نمونه‌ها با ترکیب الگوریتم بهینه‌سازی اجتماع ذرات (PSO)<sup>۷</sup> با فیلتر ذره‌ای، فیلتر ذره‌ای جدید بنام فیلتر ذره‌ای بهینه‌سازی اجتماع ذرات (PSOPF) ارائه شده است. در این الگوریتم، جدیدترین اندازه‌گیری‌ها در پروسه نمونه برداری شرکت کرده و سبب بهینه‌سازی این پروسه می‌شوند. از طریق الگوریتم اجتماع ذرات، ذرات به سمت ناحیه‌ای حرکت می‌کنند که دارای چگالی احتمال پسین بالاتری دارند. این الگوریتم سبب حل مشکل فقر ذرات و کاهش تعداد ذرات برای تخمین دقیق خواهد شد. پارک و همکاران در سال ۲۰۰۹ با استفاده از عملگرهای ژنتیک، از همگرایی زودرس ذرات فیلتر



شکل ۱: نمونه برداری بر اهمیت [۱۶]

بنابراین می‌توان وزن ذرات را به صورت رابطه بازگشتی زیر بازنویسی کرد [۲]:

$$w_t^{(i)} = w_{t-1}^{(i)} \frac{p(y_t | x_t^{(i)}) p(x_t^{(i)} | x_{t-1}^{(i)})}{q(x_t^{(i)} | x_{t-1}^{(i)}, y_t)} \quad (9)$$

شکل ۱ نمونه برداری بر اهمیت را بصورت گرافیکی نشان می‌دهد. این شیوه بدست آوردن تابع چگالی احتمال پسین به الگوریتم نمونه‌برداری بر اهمیت بازگشتی (SIS) [۱۱] معروف است [۱۹]. به منظور جلوگیری از پدیده تباهیدگی، نمونه‌برداری مجدد معرفی شده است [۱]. هر جا که تباهیدگی مشاهده شود نیاز به نمونه‌برداری مجدد است. نمونه برداری مجدد سبب می‌شود که ذرات با وزن کوچک حذف و ذرات با وزن بزرگتر تکثیر شوند [۱]. الگوریتم SIS با نمونه‌گیری مجدد به الگوریتم SIR [۱۲] معروف است که به‌طور خلاصه دارای گام‌های زیر است:

۱- نمونه‌برداری

ذرات جدید از جمعیت قدیم آنها و با نمونه‌برداری از توزیع پیشنهادی  $q(x_t | y_{1:t})$  بدست می‌آیند.

۲- محاسبه وزن نمونه‌ها

به هر ذره، یک وزن اختصاص داده می‌شود که از رابطه زیر به دست می‌آید:

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

۳- نرمال سازی وزن‌ها

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

۳-نمونه برداری مجدد

ذره‌ای تابع چگالی احتمال پسین  $p(x_{0:t} | y_{1:t})$  را بصورت مجموعه‌ای از ذرات وزن داده شده به صورت زیر بیان می‌کند:

$$S_t = \{(x_t^{(i)}, w_t^{(i)}) | i = 1, \dots, N\} \quad (3)$$

که  $N$  بیانگر تعداد ذرات و  $w_t^{(i)}$  وزن مربوط به  $x_t^{(i)}$  است. در این حالت  $p(x_{0:t} | y_{1:t})$  با مجموع وزن داده شده‌ای به صورت زیر تقریب زده می‌شود [۱-۲]:

$$p(x_{0:t} | y_{1:t}) \approx \sum_{i=1}^N w_t^{(i)} \delta(x_t - x_t^{(i)}) \quad (4)$$

که  $\delta(x)$  تابع دلتای دیراک است و  $w_t^{(i)} > 0$  وزن مربوط به  $x_t^{(i)}$  و  $\sum_{i=1}^n w_t^{(i)} = 1$  است. به دلیل اینکه نمونه برداری مستقیم از

تابع چگالی احتمال اصلی که به تابع توزیع هدف<sup>۹</sup> معروف است ممکن نیست، از روش نمونه‌برداری بر اهمیت استفاده می‌شود. در این روش به جای نمونه‌برداری از تابع هدف از یک تابع توزیع پیشنهادی<sup>۱۰</sup> نمونه‌برداری می‌شود. با در نظر گرفتن تابع چگالی پیشنهادی به صورت  $q(x_{0:t} | y_{1:t})$ ، وزن ذرات به صورت زیر محاسبه می‌شود [۱]:

$$w_t^{(i)} = \frac{p(x_{0:t}^{(i)} | y_{1:t})}{q(x_{0:t}^{(i)} | y_{1:t})} \quad (5)$$

با تجزیه تابع چگالی پیشنهادی به صورت زیر، وزن‌ها به صورت بازگشتی محاسبه خواهند شد [۱۶]:

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

در این صورت نمونه‌های  $x_{0:k}^i$  از تابع چگالی پیشنهادی با افزودن نمونه‌های  $x_{0:k-1}^i$  که از چگالی پیشنهادی  $q(x_{0:k-1} | y_{1:k-1})$  نمونه‌برداری شده‌اند با حالت جدید  $x_{0:k}^i$  نمونه‌برداری شده از  $q(x_{0:k} | x_{0:k-1}, y_{1:k-1})$  بدست می‌آید که وزن این نمونه‌ها به صورت زیر است [۱۷-۱۸]:

$$w_t = \frac{p(x_{1:t} | y_{1:t})}{q(x_{t:t-1} | y_{1:t-1}) q(x_t | y_t, x_{t-1})} \quad (7)$$

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

$$(8)$$

$$\begin{aligned} p(x_{0:k} | y_{1:k}) &= \frac{p(y_k | x_{0:k}, y_{1:k-1}) p(x_{0:k} | y_{1:k-1})}{p(y_k | y_{1:k-1})} \\ &= \frac{p(y_k | x_{0:k}, y_{1:k-1}) p(x_k | x_{0:k-1}, y_{1:k-1})}{p(y_k | y_{1:k-1})} p(x_{0:k-1} | y_{1:k-1}) \\ &= \frac{p(y_k | x_k) p(x_k | x_{k-1})}{p(y_k | y_{1:k-1})} p(x_{0:k-1} | y_{1:k-1}) \\ &\propto p(y_k | x_k, y_{1:k-1}) p(x_k | x_{k-1}) p(x_{0:k-1} | y_{1:k-1}) \end{aligned}$$

## Archive of SID

الگوریتم DE، سرعت بالا، سادگی و قدرتمند بودن آن است. الگوریتم DE نسخه‌های گوناگونی دارد که با  $DE/x/y/z$  مشخص می‌شوند. فرض کنید مسئله بهینه‌سازی زیر را داریم:

$$f(X), X \in S, S \subseteq R^D$$

که  $D$  ابعاد متغیرهای تصمیم است. الگوریتم DE یک جمعیت شامل NP بردار که کاندیدهای حل هستند با اپراتورهای جهش، برش و انتخاب نمو می‌دهد. هر بردار کاندید جواب به صورت زیر در نظر گرفته می‌شود:

$$X_{i,G} = [x_{1,i,G}, x_{2,i,G}, \dots, x_{D,i,G}] ; i=1,2,\dots, NP$$

که  $G$  شماره تکرار فعلی است. الگوریتم DE به‌طور خلاصه شامل مراحل زیر است [۲۱]:

۱- مقدار دهی اولیه جمعیت  $(C(0))$

در DE جمعیت اولیه بصورت تصادفی درون ناحیه جستجو که با باندهای ماکزیمم و مینیمم محدود شده است ایجاد می‌شود.  $J$  امین پارامتر از  $X_{i,G}$  به صورت زیر مقدار دهی اولیه می‌شود:

$$x_{j,i,G} = L_j + rand(0,1).(U_j - L_j)$$

که  $rand(0,1)$  اعداد تصادفی یکنواخت در فاصله  $[0,1]$  ایجاد می‌کند و  $L_j$  و  $U_j$  باند بالا و باند پایین برای  $J$  امین متغیر ارایه می‌کنند.

۲- اپراتور جهش

بعد ایجاد جمعیت اولیه، الگوریتم DE یک بردار جهش داده شده  $V_{i,G}$  برای هر عضو  $X_{i,G}$  که بردار هدف نامیده می‌شود تولید می‌کند. نماد عمومی برای استراتژی‌های جهش "DE/x/y" است. استراتژی‌های متنوعی در متون است که استراتژی‌های معروف آن به شرح زیر می‌باشد:

• **DE/rand/1**

$$V_{i,G} = X_{i_1,G} + F(X_{i_2,G} - X_{i_3,G})$$

• **DE/rand/2**

$$V_{i,G} = X_{i_1,G} + F(X_{i_2,G} - X_{i_3,G}) + F(X_{i_4,G} - X_{i_5,G})$$

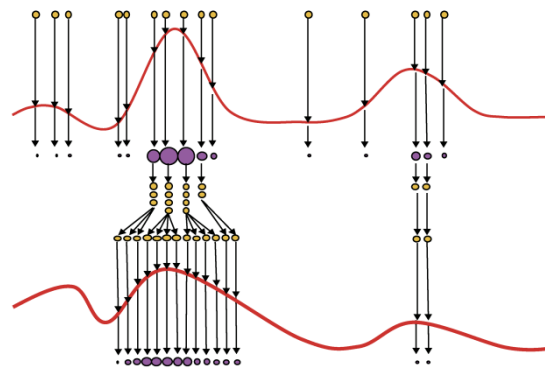
• **DE/current-to-best/1**

$$V_{i,G} = X_{i,G} + F(X_{best,G} - X_{i,G}) + F(X_{i_1,G} - X_{i_2,G})$$

$i_1, i_2, i_3$  به صورت تصادفی به گونه‌ای انتخاب می‌شوند که دو به دو متفاوت باشند.

۳- اپراتور برش

اپراتور برش به هر جفت بردار هدف  $X_{i,G}$  و بردار جهش داده متناظر آن  $V_{i,G}$  برای تولید بردار آزمایشی  $U_{i,G}$  بکار می‌رود.



شکل ۲: الگوریتم SIR [۱۸]

در نمونه‌برداری مجدد ذرات با وزن کمتر با ذرات با وزن بالاتر جایگزین می‌شوند. شکل ۲ عمل کرد الگوریتم SIR را به صورت گرافیکی نشان می‌دهد. پروسه نمونه‌برداری مجدد در فیلتر SIR اگر چه سبب کاهش مسئله تابهدگی می‌شود اما از طرف دیگر کاهش تنوع میان ذرات و ایجاد پدیده‌ای بنام فقر نمونه را در بر دارد [۱]. این پدیده وقتی نويز پروسه پایین باشد شدید و سبب ریزش همه ذرات به یک نقطه ریزش می‌شود. در نتیجه نمی‌توانند تقریب درستی از تابع چگالی احتمال پسین بزنند و دقت تخمین‌ها پایین خواهد آمد.

## ۲-۲- الگوریتم تفاضل تکاملی

الگوریتم تکاملی تفاضلی (DE) نخستین بار در سال ۱۹۹۵ توسط استورن و پرایس معرفی شد [۲۰]. مهمترین تمایز الگوریتم تفاضل تکاملی با سایر روش‌های تکاملی در این است که این الگوریتم از اطلاعات فاصله و جهت جمعیت فعلی برای هدایت پروسه جستجو استفاده می‌کند. این الگوریتم با ایجاد یک جمعیت اولیه شروع به کار می‌کند. سپس با اعمال اپراتورهایی شامل جهش و برش نسل نوزاد تشکیل شده و در مرحله بعد که مرحله انتخاب نام دارد، نسل نوزاد با نسل والد برای میزان شایستگی که توسط تابع هدف سنجیده می‌شود، مقایسه می‌گردد. سپس بهترین اعضا به عنوان نسل بعدی وارد مرحله بعد می‌گردند. این عمل تا رسیدن به نتایج مطلوب ادامه می‌یابد. یکی از مزایای این الگوریتم داشتن حافظه می‌باشد که اطلاعات جواب‌های مناسب را در جمعیت فعلی حفظ می‌کند. دیگر مزیت این الگوریتم مربوط به عملکرد انتخاب آن می‌باشد. در این الگوریتم، همه اعضای یک جمعیت شانس مساوی برای انتخاب شدن به عنوان یکی از والدین را دارند به این صورت که نسل نوزاد با نسل والد از نظر میزان شایستگی که توسط تابع هدف سنجیده می‌شود، مقایسه می‌گردد. سپس بهترین اعضا به عنوان نسل بعدی وارد مرحله بعد می‌گردند. مهمترین ویژگی‌های

Archive of SID

$$p(x_t | y_{1:t}) \propto p(y_t | x_t) p(x_t | y_{1:t-1})$$

$$= p(z_t | x_t) \int p(x_t | x_{t-1}) p(x_{t-1} | z_{1:t-1}) dx_{t-1} \quad (14)$$

انتگرال رابطه (۱۲) در حالت کلی بصورت تحلیلی حل آن مشکل است. اما از آنجایی که  $p(x_{t-1} | y_{1:t-1})$  بوسیله مجموع ذرات  $\{x_{t-1}^{(i)}, w_{t-1}^{(i)}\}$  تخمین زده شده است تقریب  $p(x_t | y_{1:t-1})$  به صورت زیر است:

$$p(x_t | y_{1:t-1}) = \sum_{i=1}^N w_{t-1}^{(i)} p(x_t | x_{t-1}^{(i)}) \quad (15)$$

آنگاه تابع چگالی احتمال پسین به صورت زیر تقریب زده می‌شود:

$$p(x_t | y_{1:t}) \propto p(y_t | x_t) \sum_{i=1}^N w_{t-1}^{(i)} p(x_t | x_{t-1}^{(i)}) \quad (16)$$

با توجه به اینکه که نمونه‌برداری از پسین واقعی  $p(x_t | y_{1:t})$  معمولا غیرممکن است، از یک توزیع پیشنهادی باید نمونه‌برداری شود. وزن پر اهمیت ذرات به صورت زیر است:

$$w_t = \frac{p(x_t | y_{1:t})}{q(x_t | y_{1:t})} \quad (17)$$

که با جایگذاری  $p(x_t | y_{1:t})$  وزن ذرات به صورت زیر محاسبه می‌شود:

$$w_t^{(i)} = \frac{p(y_t | x_t^{(i)}) \sum_{j=1}^M w_{t-1}^{(j)} p(x_t^{(i)} | x_{t-1}^{(j)})}{q(x_t | y_{1:t})} \quad (18)$$

شبه کد فیلتر ذره‌ای حاشیه‌ای در شکل ۳ به‌طور خلاصه نشان داده شده است.

شکل(۳): شبه کد فیلتر ذره‌ای حاشیه‌ای

۱. برای  $i=1, \dots, N$  نمونه برداری از تابع توزیع پیشنهادی  $q(x_t | y_{1:t})$

۲. برای  $i=1, \dots, N$  محاسبه وزن پراهمیت

$$w_t^{(i)} = \frac{p(y_t | x_t^{(i)}) \sum_{j=1}^M w_{t-1}^{(j)} p(x_t^{(i)} | x_{t-1}^{(j)})}{q(x_t | y_{1:t})}$$

۳. نرمال سازی وزن ها

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

۳-۱- بهبود نمونه‌برداری

یک ویژگی فیلتر ذره‌ای حاشیه‌ای این است که واریانس وزن ذرات در طول زمان افزایش می‌یابد. در نتیجه تلاش محاسباتی

$$u_{j,i,G} = \begin{cases} v_{j,i,G} & j \in J \\ x_{j,i,G} & j \notin J \end{cases}$$

که  $J$  نشان دهنده ژن‌های کروموزوم  $j$  ام است که تحت اپراتور برش قرار می‌گیرند.

۴- اپراتور انتخاب

اپراتور انتخاب برای انتخاب بهتر از بین  $V_{i,G}$  و  $X_{i,G}$  برای جمعیت بعدی استفاده می‌شود. در صورتی که تابع معیار  $V_{i,G}$  از تابع معیار  $X_{i,G}$  بهتر باشد  $V_{i,G}$  به جمعیت بعد منتقل خواهد شد و در غیر این صورت خود  $X_{i,G}$  به جمعیت بعد منتقل خواهد شد.

۳-فیلتر ذره‌ای بهبود یافته

در فیلتر ذره‌ای استاندارد، از آنجایی که تابع چگالی پسین مشترک<sup>۱۳</sup> حالت  $p(x_{0:t} | y_{1:t})$  با استفاده از نمونه‌برداری پراهمیت بازگشتی تقریب زده می‌شود، ابعاد تابع چگالی پسین مشترک در هر لحظه از زمان رشد می‌کند. این موجب می‌شود که الگوریتم سریعاً تباهیده شود و بنابراین استفاده از استراتژی نمونه‌برداری مجدد به منظور گارانتی یک تقریب منطقی از تابع چگالی احتمال پسین لازم می‌شود. با توجه به ساختار فیلتر ذره‌ای استاندارد، لازم است فرآیند نمونه‌برداری پراهمیت بر روی کل مسیر  $(x_{0:t}, \omega_{0:t})$  اجرا شود. در نتیجه باید تعداد ذرات برای حفظ کردن سطح مطلوبی از دقت با گذشت زمان افزایش یابد.

با وجود این، اکثر تحقیقات در عمل نمونه‌برداری مجدد را روی فضای حاشیه‌ای  $(x_t, \omega_t)$  انجام داده‌اند. در حالی که، تخمین وابسته به کل مسیر حالت است و لزوماً رفتار فراموشی نمایی از خطای گذشته‌اش ندارد و نمونه‌برداری مجدد این مسیرها در فضای مشترک برای حذف گذشته در زمان محدود گارانتی شده است. به‌علاوه از آنجایی که ممکن است برخی سیستم‌ها دارای رفتار فراموشی نمایی از خطاهای گذشته نداشته باشند، با تعداد ذره محدود فرآیند نمونه‌برداری مجدد روی فضای حاشیه‌ای یک تخمین ناسازگار بوجود می‌آورد.

برای رفع این مشکل در این مقاله از فیلتر ذره‌ای حاشیه‌ای استفاده می‌شود. فیلتر ذره‌ای حاشیه‌ای برخلاف فیلتر ذره‌ای بر روی توزیع حاشیه‌ای  $p(x_t | y_{1:t})$  انجام می‌شود و ابعاد نمونه‌برداری با زمان افزایش نمی‌یابد. گام‌های بروز رسانی و پیش بینی در فیلتر ذره‌ای حاشیه‌ای به صورت زیر است:

$$p(x_t | y_{1:t-1}) = \int p(x_t | x_{t-1}) p(x_{t-1} | y_{1:t-1}) dx_{t-1} \quad (12)$$

$$p(x_t | y_{1:t}) = \frac{p(y_t | x_t) P(x_t | y_{1:t-1})}{P(y_t | y_{1:t-1})} \quad (13)$$

با جایگذاری (۱۲) در (۱۳) گام بروز رسانی به صورت زیر می‌شود:

Archive of SID

گام نمونه برداری پراهمیت وارد شده است. برای این منظور تابع توزیع پیشنهادی به صورت ساده زیر در نظر شده است:

$$q(x_t | y_{1:t}) = \sum_{i=1}^N w_{t-1}^{(i)} p(x_t | x_{t-1}^{(i)}) \quad (24)$$

با وجود این، قبل از نمونه برداری ذرات با استفاده از الگوریتم تفاضل تکاملی به گونه‌ای جابجا می‌شوند که در مکان‌هایی قرار گیرند که تابع درست‌نمایی  $p(y_t | x_t)$  بیشینه شود. روش نمونه برداری ارائه شده می‌تواند تنوع میان ذرات را حفظ کند و بعلاوه ذرات نمونه برداری مجدد شده را وادار کند که به‌طور مجانبی نمونه‌ها را از تابع چگالی احتمال پسین حالت‌های واقعی تقریب بزنند.

مسئله جابجایی ذرات، را می‌توان بصورت مسئله ماکزیمم کردن تابع درست‌نمایی زیر در نظر گرفته شده است:

$$\arg \max_{x_t} p(y_t | x_t) \quad (25)$$

شکل ۴: الگوریتم تشکیل بردار  $J$

$$J \sim U(1, n_x)$$

$$j \sim U(0, n_x - 1)$$

Repeat

$$J \leftarrow J \cup \{j+1\}$$

$$J = (j+1) \bmod n_x$$

Until  $U(0,1) \geq p_r$  or  $|J| = n_x$

شکل ۵: بهبود نمونه برداری بر اساس تکامل تفاضلی

۱. مقدار دهی جمعیت اولیه  $C_0$

۲. تشکیل بردار جهش یافته با اعمال اپراتور جهش

$$v_{t,i,G} = x_{t,i,G} + F(x_{t,best,G} - x_{t,i,G}) + F(x_{t,i_1,G} - x_{t,i_2,G})$$

۳. ایجاد بردار آزمایشی با بکارگیری اپراتور برش

$$u_{t,j,i,G} = \begin{cases} v_{t,j,i,G} & j \in J \\ x_{t,j,i,G} & o.w. \end{cases}$$

۴. محاسبه تابع هدف

۵. انتخاب بر اساس تابع هدف

$$x_{t,i,G+1} = \begin{cases} u_{t,i,G} & \text{if } f(u_{t,i,G}) \geq f(x_{t,i,G}) \\ x_{t,i,G} & o.w. \end{cases}$$

در این مقاله از الگوریتم‌های تکاملی برای حل مسئله بهینه سازی فوق استفاده شده است. الگوریتم‌های تکاملی مختلفی تاکنون ارائه

زیادی برای بروز رسانی ذرات با وزن ناچیز می‌شود. یک راه حل برای این مشکل بهبود نمونه برداری و در نتیجه انتخاب مناسب تابع توزیع پیشنهادی است. عمومی‌ترین تابع توزیع پیشنهادی، توزیع پیشین است:

$$q(x_t) = p(x_t | y_{1:t-1}) \quad (19)$$

که در این صورت وزن ذرات از روابط زیر محاسبه می‌شود:

$$w_t^{(i)} = \frac{p(x_t^{(i)} | y_{1:t})}{q(x_t^{(i)})} = \frac{1}{\hat{c}_t} p(y_t | x_t^{(i)}) \frac{p(x_t^{(i)} | y_{1:t-1})}{q(x_t^{(i)})} \quad (20)$$

که  $\hat{c}_t$  به صورت زیر است:

$$\hat{c}_t = \sum_{i=1}^N p(y_t | x_t^{(i)}) \frac{p(x_t^{(i)} | y_{1:t-1})}{q(x_t^{(i)})}$$

و در نتیجه وزن ذرات به صورت زیر است:

$$w_t^{(i)} \propto p(y_t | x_t^{(i)}) \quad (21)$$

در این استراتژی از آنجایی که آخرین اطلاعات اندازه‌گیری در فرآیند نمونه برداری مورد توجه قرار نمی‌گیرد، دقت پایین بوده و لازم است که تعداد ذرات زیاد باشند. این مسئله بویژه در سیستم‌های چند بعدی سبب افزایش حجم محاسبات و پیچیدگی محاسبات خواهد شد. انتخاب بهینه برای تابع توزیع پیشنهادی در فیلتر ذره‌ای حاشیه‌ای، تابع پسین است:

$$q(x_t | y_{1:t}) = p(x_t | y_{1:t}) = p(y_t | x_t) \sum_{i=1}^N w_{t-1}^{(i)} p(x_t | x_{t-1}^{(i)}) \quad (22)$$

$$= \sum_{j=1}^N \omega_{t-1}^{(j)} p(y_t | x_t) p(x_t | y_t, x_{t-1}^{(j)})$$

در این صورت وزن ذرات به صورت زیر محاسبه می‌شود:

$$w_t^{(i)} = \frac{p(y_t | x_t^{(i)}) \sum_{j=1}^N w_{t-1}^{(j)} p(x_t^{(i)} | x_{t-1}^{(j)})}{\sum_{j=1}^N \omega_{t-1}^{(j)} p(y_t | x_t) p(x_t | y_t, x_{t-1}^{(j)})} \quad (23)$$

با وجود این، در حالت کلی نمونه برداری از این تابع توزیع پیشنهادی در اغلب موارد مشکل است و حجم محاسبات را افزایش می‌دهد.

برای اجتناب از این مشکل و در عین حال دقت بالا، در این مقاله، با استفاده از الگوریتم تکامل تفاضلی اندازه‌گیری‌ها به داخل

Archive of SID

جمعیت جدید، اپراتور برش به بردار هدف و بردار جهش یافته متناظر آن اعمال و بردار آزمایشی  $U_{i,G}$  تشکیل می‌شود:

$$u_{t,j,i,G} = \begin{cases} v_{t,j,i,G} & j \in J \\ x_{t,j,i,G} & j \notin J \end{cases} \quad (27)$$

$j = 1 \dots D$

که  $x_{t,i,j,G}$  اشاره به  $j$ -th المان بردار  $x_{t,i,G}$  دارد و  $J$  مجموعه نقاط برش است. روش‌های مختلفی برای ایجاد  $J$  وجود دارد که در این مقاله از الگوریتم داده شده در شکل (۴) استفاده شده است.

شکل ۶: شبه کد فیلتر ذره‌ای بهبود یافته

۱. برای  $i=1, \dots, N$  نمونه برداری از تابع توزیع پیشنهادی

$$q(x_t | y_{1:t}) \sim \sum_{i=1}^N w_{t-1}^{(i)} p(x_t | x_{t-1}^{(i)})$$

۲. حرکت ذرات به ناحیه با ماکزیمم تابع درست‌نمایی با الگوریتم تقاضای تکاملی:

۲-۱- مقدار دهی جمعیت اولیه

۲-۲- اعمال اپراتور جهش

۲-۳- اعمال اپراتور برش

۲-۴- انتخاب بر اساس تابع هدف

۳. برای  $i=1, \dots, N$  محاسبه وزن پراهمیت

$$w_t^{(i)} = \frac{p(y_t | x_t^{(i)}) \sum_{j=1}^N w_{t-1}^{(j)} p(x_t^{(i)} | x_{t-1}^{(j)})}{\sum_{j=1}^N w_{t-1}^{(j)} p(x_t^{(i)} | x_{t-1}^{(j)})}$$

۴. نرمال سازی وزن‌ها

اپراتور انتخاب تعیین می‌کند که آیا بردار هدف یا بردار آزمایشی متناظر آن در جمعیت بعد  $(G+1)$  باقی بماند. برای این منظور، تابع معیار بردار آزمایشی با تابع معیار تابع بردار هدف مقایسه می‌شود:

$$x_{t,i,G+1} = \begin{cases} u_{t,i,G} & \text{if } f(u_{t,i,G}) \geq f(x_{t,i,G}) \\ x_{t,i,G} & \text{o.w.} \end{cases}$$

تابع معیار  $x_t$  به صورت زیر است:

$$f(\cdot) = p(y | x_t) \quad (18)$$

فرآیند برش، جهش و انتخاب روی جمعیت تا جایی تکرار می‌شود که ماکزیمم پیدا شود یا به ماکزیمم از قبل تعیین شده برسد. وقتی بهترین مقدار تابع برازندگی به یک آستانه مشخص رسید، تکرارها متوقف می‌شوند.

شده که در این مقاله از  $DE$  برای بهبود نمونه برداری استفاده شده است. استفاده از این الگوریتم بدان علت است که  $DE$  جواب را سریعتر از سایر الگوریتم پیدا می‌کند. به علاوه پیاده سازی آن راحت تر و تعداد پارامترهای کمتری برای تنظیم کردن دارد. نهایتاً وارد کردن  $DE$  در گام نمونه برداری سبب می‌شود که ذرات اطلاعات خود را با یکدیگر به اشتراک می‌گذارند و موقعیت بهترین ذره را دنبال کنند. همچنین  $DE$  دارای اپراتورهای پرش و جهش است که به کمک آنها می‌توان تنوع میان ذرات را حفظ کرد و از همگرایی زودرس آنها جلوگیری کرد.

در حالت کلی  $DE$  و فیلتر ذره‌ای حاشیه‌ای دارای مشخصاتی هستند که استفاده آنها با یکدیگر ترغیب کننده است.  $DE$  بهترین مقدار را از طریق بروز رسانی موقعیت ذرات بصورت بازگشتی و فیلتر ذره‌ای حاشیه‌ای بهترین تقریب توزیع تابع پسین را از طریق بروز رسانی بازگشتی موقعیت ذرات و وزنشان بدست می‌آورند. در  $DE$  یک ذره با کمترین تابع هزینه بیانگر نقطه بهینه در فضای جستجو است و در فیلتر ذره‌ای حاشیه‌ای ذره با بیشترین وزن به عنوان حالت محتمل سیستم در نظر گرفته می‌شود. فیلتر ذره‌ای حاشیه‌ای و  $DE$  هر دو ذرات را به سمت ذره با بهترین تابع هزینه سوق می‌دهند.

در حل مسئله مورد نظر با الگوریتم تقاضای تکاملی، ذرات  $\{x_t^i; i=1, \dots, N\}$  بعنوان بردارهای هدف جمعیت جاری و تابع درستمائی متناظرشان بعنوان توابع هدف بردارهای هدف نگریسته می‌شوند. الگوریتم تقاضای تکاملی ذرات را بوسیله یک فرآیند تکراری از جهش<sup>۱۴</sup> و برش<sup>۱۵</sup> و انتخاب ترکیب می‌کند و در نتیجه یک مجموعه ذرات متنوع ایجاد خواهد شد. فرض کنید  $C_{t,G} = \{x_{t,1,G}, \dots, x_{t,N,G}\}$  بیان کننده جمعیت جاری  $DE$  باشد که شامل  $N$  بردار هدف با ابعاد  $D$  است. الگوریتم  $DE$  از یک بردار  $x_{t,i,G}$  برای ارائه هر کاندید جواب  $i$  در تکرار  $G$  و گام زمانی داده شده  $t$  استفاده می‌کند. برای هر بردار هدف  $x_{t,i,G}$  الگوریتم  $DE$  یک بردار جهش یافته  $v_{t,i,G}$  را به صورت زیر تولید می‌کند:

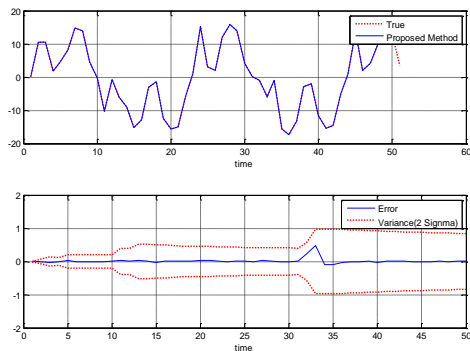
$$v_{t,i,G} = x_{t,i,G} + F(x_{t,best,G} - x_{t,i,G}) + F(x_{t,i_1,G} - x_{t,i_2,G}) \quad (26)$$

که  $x_{t,i,G}$  بردار هدف برای جهش دادن در تکرار  $G$  است.  $x_{t,i_2,G}$  و  $x_{t,i_1,G}$  دو عضو از جمعیت است که به صورت تصادفی طوری انتخاب شده‌اند که  $i_2$  و  $i_3$  با هم مساوی نباشند. پارامتر  $F$  ضریب مقیاس است که دامنه واریانس تفاضل  $(x_{t,i_2,G} - x_{t,i_3,G})$  را کنترل می‌کند. بعد از جهش بمنظور افزایش تنوع بردارهای



Archive of SID

این مقاله از آن برای بررسی و مقایسه عملکرد روش پیشنهادی در مقایسه با سایر روش‌ها تحت شرایط مختلف استفاده شده است.

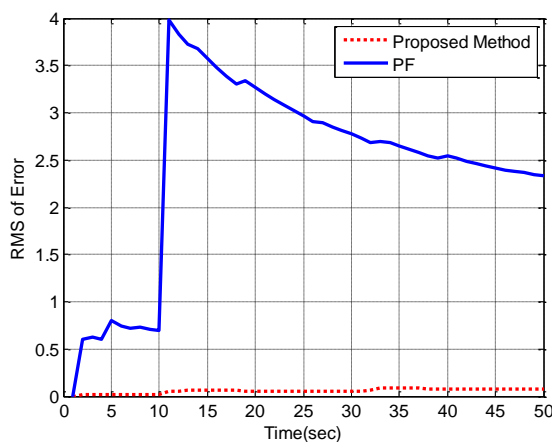


شکل ۸: تخمین بدست آمده از روش پیشنهادی  $R_t = 1^2, Q_t = 1^2$

برای ارزیابی عملکرد روش پیشنهادی، در اولین آزمایش نویز پروسه و اندازه‌گیری به صورت زیر در نظر گرفته شده است:

$$R_t = 1^2 \quad Q_t = 1^2 \quad (30)$$

همچنین تعداد ذرات  $N = 100$  در نظر گرفته شده است. شکل‌های ۷-۹ عملکرد فیلتر ذره‌ای استاندارد و فیلتر پیشنهادی را نشان می‌دهند. در شکل‌های ۷-۸ حالت واقعی، حالت تخمین زده شده و خطای تخمین و انحراف استاندارد خطا ( $2\sigma$ ) رسم شده است. انحراف استاندارد خطا از روش مونت کارلو با ۵۰ بار اجرا بدست آمده است. مقدار جذر میانگین مربع خطا ( $RMSE$ ) بر حسب زمان در شکل ۹ نشان داده شده است. نتایج نشان می‌دهند که عملکرد فیلتر پیشنهادی بهتر از فیلتر ذره‌ای است. این بدان دلیل است که مطابق شکل ۱۰ در روش پیشنهادی تنوع میان ذرات بیشتر از فیلتر ذره‌ای است. در نتیجه ذرات می‌توانند تقریب درست‌تری از تابع چگالی احتمال پسین بزنند و دقت تخمین‌ها بهبود می‌یابد.



شکل ۹: نسبت به زمان  $RMSE$  نسبت به زمان ( $R_t = 1^2, Q_t = 1^2$ )

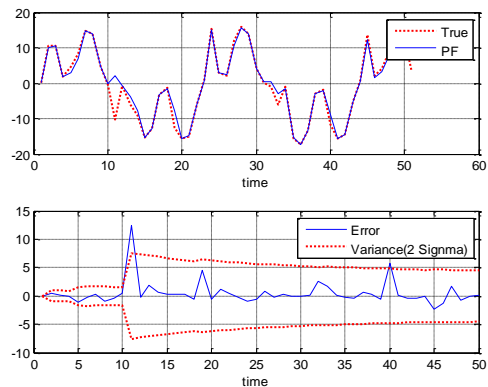
به طور خلاصه الگوریتم نمونه برداری بهبود یافته براساس تکامل تفاضلی بصورت شکل (۵) و شبه کد کلی فیلتر ذره‌ای حاشیه‌ای با الگوریتم تکامل تفاضلی به صورت شکل (۶) است.

۴- نتایج

برای ارزیابی عملکرد و دقت روش پیشنهادی با الگوریتم‌های PF، EPF و UPF مقایسه شده است. برای این منظور از دو مثال استفاده شده است. اولین مثال یک مدل غیرخطی است. این مساله یک مثال معروف و مرجع در حوزه فیلترهای غیرخطی می باشد که در این مقاله برای بررسی و مقایسه عملکرد روش پیشنهادی در مقایسه با سایر روش‌ها استفاده شده است. در مثال دوم عملکرد روش پیشنهادی در مقایسه با سایر الگوریتم‌ها در یک کاربرد بررسی شده است. در این مثال، ردیابی یک هلیکوپتر روی دنباله-ای از تصاویر ویدئو بررسی شده است.

۴-۱- سیستم غیر خطی

برای ارزیابی عملکرد فیلتر ذره‌ای بهبود یافته مبتنی بر محاسبات نرم، سیستم غیرخطی زیر در نظر گرفته شده است. این سیستم از میزان غیرخطی بالایی برخوردار است به همین دلیل به عنوان یک مسئله بنچمارک جهت ارزیابی عملکرد فیلترها در بسیاری از مقالات مورد استفاده قرار گرفته است [۱-۲]:

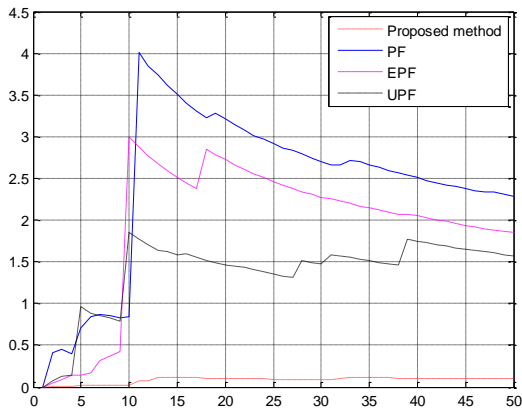


شکل ۷: تخمین بدست آمده از روش فیلتر ذره‌ای  $R_t = 1^2, Q_t = 1^2$

$$x_t = \frac{x_{t-1}}{2} + \frac{25x_{t-1}}{1+x_{t-1}^2} + 8\cos(1.2t) + w_t \quad (28)$$

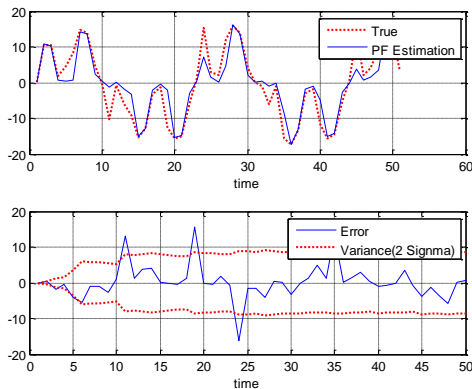
$$y_t = \frac{x_t^2}{20} + v_t \quad (29)$$

در این سیستم  $w_t$  و  $v_t$  بترتیب نویز اندازه‌گیری و فرآیند است که سفید و کواریانس آنها به ترتیب  $R_t$  و  $Q_t$  است. معادله ۲۸ مدل حرکت هدف و معادله ۲۹ مدل اندازه‌گیری‌های انجام شده از آن است. این سیستم از میزان غیرخطی بالایی برخوردار است و در



شکل ۱۲: RMSE الگوریتم‌های مختلف ( $R_t = 1^2, Q_t = 1^2$ )

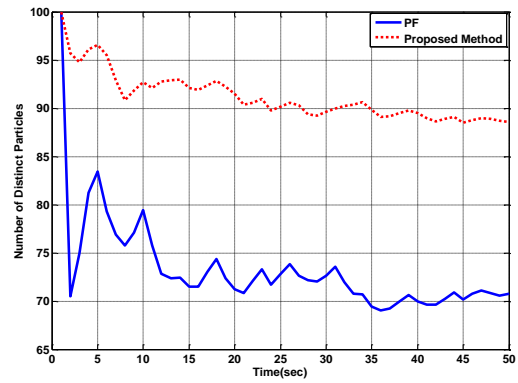
در جدول ۱ روش پیشنهادی در مقایسه با PF از نظر زمان محاسبات تحت ذرات مختلف با هم مقایسه شده است. همانطور که از جدول ۱ ملاحظه می‌شود زمان محاسبات الگوریتم پیشنهادی بیشتر از PF می‌باشد. با وجود این، با توجه به شکل ۱۱ و جدول ۱ مشاهده می‌شود که روش پیشنهادی برای بدست آوردن دقت تخمین یکسان با فیلتر ذره‌ای به مراتب تعداد ذرات کمتری لازم دارد. بنابراین یکی از مزیت‌های فیلتر پیشنهادی دقت بالاتر با تعداد ذرات کمتر و در نتیجه هزینه محاسباتی کمتر است.



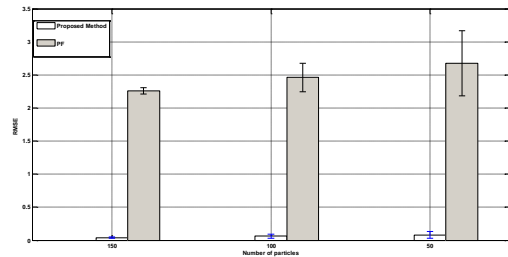
شکل ۱۳: تخمین بدست آمده از فیلتر ذره‌ای ( $R_t = 1^2, Q_t = 5^2$ )

جدول ۱: مقایسه الگوریتم‌ها از نظر زمان محاسبات

Number Particles	Methods	Processing Times
150	Proposed Method	0.91
	PF	0.21
100	Proposed Method	0.69
	PF	0.16
50	Proposed Method	0.29
	PF	0.12



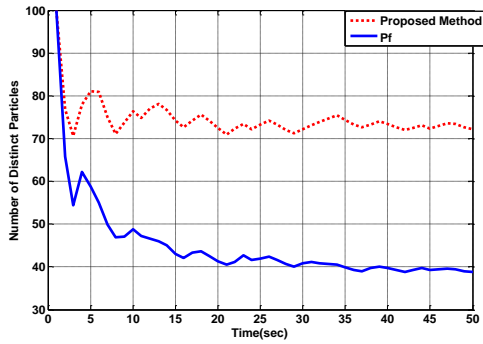
شکل ۱۰: تنوع میان ذرات ( $R_t = 1^2, Q_t = 1^2$ )



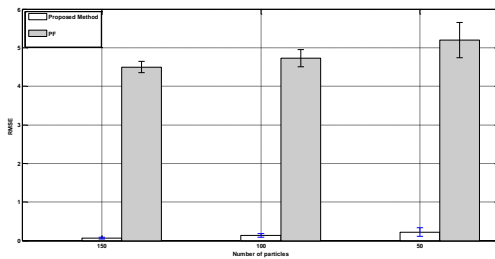
شکل ۱۱: عملکرد الگوریتم‌ها با تعداد ذرات مختلف ( $R_t = 1^2, Q_t = 1^2$ )

شکل ۱۱ عملکرد الگوریتم پیشنهادی را در مقایسه با PF با تعداد ذرات مختلف نشان می‌دهد. هر میله در این شکل نشان‌دهنده میانگین و انحراف استاندارد RMSE است. مشاهده می‌شود عملکرد روش پیشنهادی وابستگی زیادی به تعداد ذرات ندارد، در حالی که وابستگی عملکرد PF به تعداد ذرات بیشتر است. در الگوریتم پیشنهادی تنوع میان ذرات بیشتر حفظ می‌شود و باعث می‌شود که ذرات نمونه‌برداری مجدد شده به‌طور مجانبی نمونه‌ها را از تابع چگالی احتمال پسین حالت واقعی تقریب بزنند. به عبارت دیگر، الگوریتم پیشنهادی می‌تواند دقت بهتری با تعداد ذرات کمتر بدست آورد.

برای بررسی بیشتر، عملکرد روش پیشنهادی با EPF و UPF مقایسه شده است شکل ۱۲ مقایسه الگوریتم‌ها را برای حالتی که نویز پروسه و اندازه‌گیری  $R_t = 1^2, Q_t = 1^2$  در نظر گرفته شده است را نشان می‌دهد. همچنین تعداد ذرات  $N = 50$  در نظر گرفته شده است. همانطور که ملاحظه می‌شود RMSE روش پیشنهادی کمتر از سایر روش‌ها در طول زمان می‌باشد.



شکل ۱۶: تنوع میان ذرات  $R_t = 1^2, Q_t = 5^2$



شکل ۱۷: عملکرد الگوریتم‌ها با تعداد ذرات مختلف  $R_t = 1^2, Q_t = 5^2$

برای بررسی پایداری روش پیشنهادی نسبت به فیلتر ذره‌ای، سازگاری<sup>۱۷</sup> آن ارزیابی شده است. سازگاری فیلترها از نرمالیزه شده مربع خطا (NEES)<sup>۱۸</sup> بدست می‌آید که به صورت زیر است:

$$\varepsilon_t = (x_t - \hat{x}_t)^T \hat{P}_t^{-1} (x_t - \hat{x}_t) \quad (31)$$

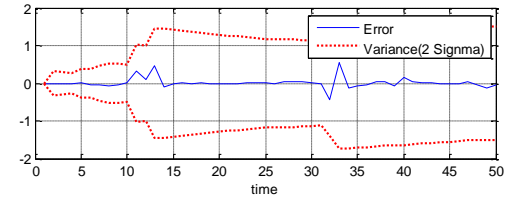
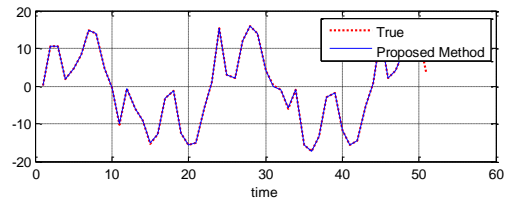
در رابطه بالا  $x_t$  مقدار واقعی حالت،  $\hat{x}_t$  حالت تخمین زده شده و  $\hat{P}_t$  کواریانس تخمین حالت است. سازگاری فیلتر با اجرای چندین اجرای مونت کارلو و محاسبه متوسط NEES ارزیابی می‌شود. برای  $M$  بار اجرای مونت کارلو متوسط NEES بصورت زیر محاسبه می‌شود:

$$\bar{\varepsilon}_t = \frac{1}{M} \sum_{i=1}^M \varepsilon_i \quad (32)$$

در مسئله مورد نظر سازگاری فیلتر پیشنهادی و فیلتر ذره‌ای برای حالتی بررسی می‌شود که ماتریس‌های کواریانس نویز پروسه و اندازه‌گیری به صورت زیر است:

$$R_k = 1^2 \quad Q_k = 5^2$$

بدلیل اینکه تعداد حالت‌های سیستم مورد نظر یکی است برای  $M = 50$ ، ناحیه احتمال ۹۵٪ برای  $\bar{\varepsilon}_t$  محدود به بازه [0.69 1.35] است.

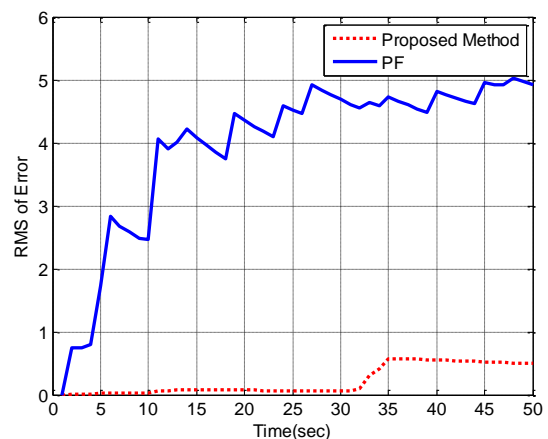


شکل ۱۴: تخمین بدست آمده از روش پیشنهادی  $R_t = 1^2, Q_t = 5^2$

شکل‌های ۱۶-۱۳ عملکرد فیلتر ذره‌ای استاندارد با فیلتر پیشنهادی برای حالتی نشان می‌دهد که نویز پروسه و اندازه‌گیری به صورت زیر در نظر گرفته شده است:

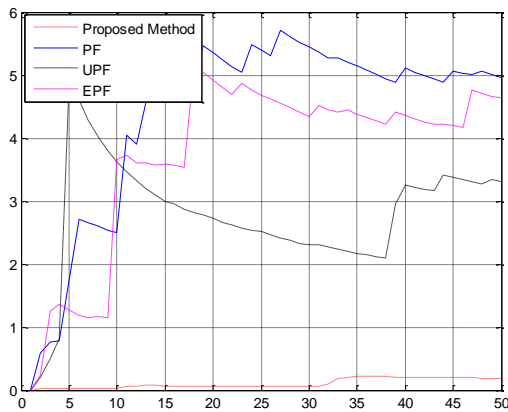
$$R_t = 1^2 \quad Q_t = 5^2 \quad (30)$$

همانطور که انتظار می‌رفت در این حالت عملکرد فیلتر ذره‌ای بدتر از حالت قبل است در حالی که عملکرد روش پیشنهادی تقریباً با حالت قبل یکسان می‌باشد. وقتی نویز پروسه نسبت به نویز اندازه‌گیری بیشتر باشد، به دلیل اینکه فقر نمونه بیشتر در فیلتر ذره‌ای اتفاق می‌افتد دقت تخمین‌ها و تنوع میان ذرات کمتر می‌شود. از شکل ۱۶ مشاهده می‌شود تنوع میان ذرات در فیلتر ذره‌ای نسبت به حالت قبل به طور قابل ملاحظه‌ای کاهش یافته است.

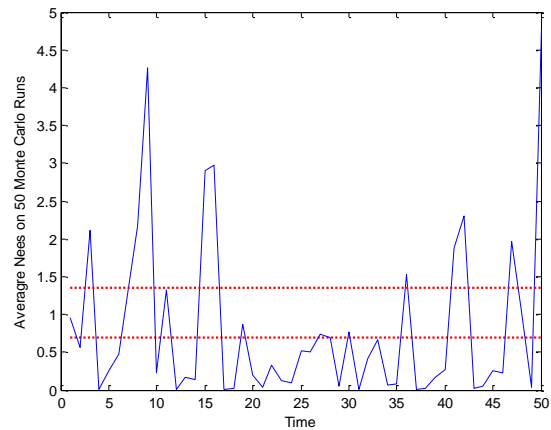


شکل ۱۵: RMSE نسبت به زمان  $R_t = 1^2, Q_t = 5^2$

شکل ۱۷ عملکرد الگوریتم پیشنهادی با PF با تعداد ذرات مختلف برای حالتی که  $R_t = 1^2, Q_t = 5^2$  است را نشان می‌دهد. ملاحظه می‌شود که مانند، حالت قبل، فیلتر پیشنهادی وابستگی کمتری به تعداد ذرات دارد.



شکل ۲۰: RMSE الگوریتم های مختلف ( $R_t = 1^2, Q_t = 5^2$ )



شکل ۱۸: سازگاری فیلتر ذره ای

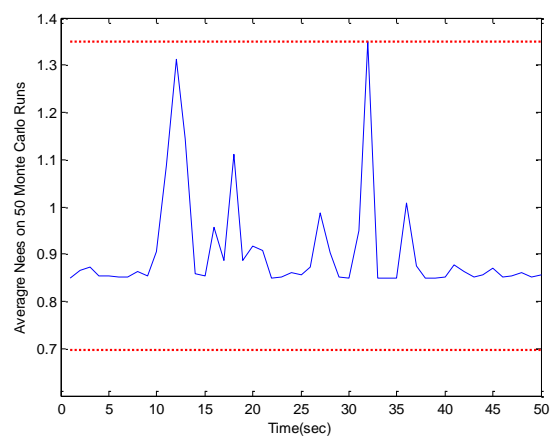
#### ۴-۳- ردیابی هلیکوپتر

به منظور ارزیابی عمل کرد روش پیشنهادی در مقایسه با فیلتر ذره‌ای در یک مثال کاربردی، عمل کرد آن در ردیابی یک هلیکوپتر در دنباله‌ای از تصاویر ویدئو بررسی شده است. هدف دنبال کردن یک هلیکوپتر سبز است که به سمت بالا و پایین کنترل می شود. در این آزمایش هلیکوپتر در حال پرواز به سوی درختان است. سپس آن را به سمت پایین حرکت می دهد. در هر دو الگوریتم تعداد ۱۸۰ ذره در نظر گرفته شده است.

نتایج در شکل‌های ۲۱ تا ۲۲ دیده می‌شود. همانطور که مشاهده می‌شود که عمل کرد روش پیشنهادی در مقایسه با فیلتر ذره‌ای بهتر است. وقتی هلیکوپتر به سمت درختان حرکت می‌کند، فیلتر ذره‌ای ردیابی را از دست می‌دهد. در حالی که روش پیشنهادی در سرتاسر ویدئو هلیکوپتر را بخوبی ردیابی می‌کند.

#### ۶- نتیجه گیری

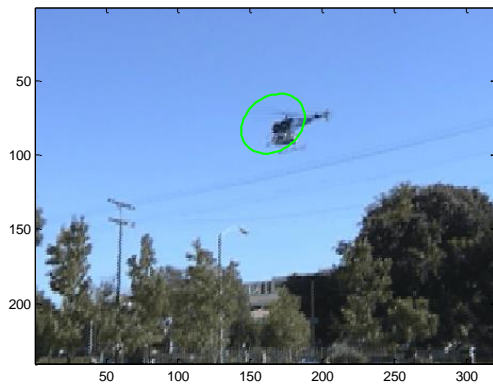
فیلتر ذره‌ای اهمیت ویژه‌ای در تخمین سیستم های غیر خطی- غیر گوسی است. در فیلتر ذره‌ای کلاسیک، پروسه نمونه‌برداری مجدد سبب کاهش تنوع میان ذرات و دقت تخمین‌ها می‌شود. برای رفع این مشکل و افزایش سازگاری، در این مقاله فیلتر ذره-ای بهبود یافته مبتنی بر محاسبات نرم پیشنهاد شده است. در این فیلتر نمونه برداری با استفاده از الگوریتم تفاضل تکاملی بهبود داده شده است در ادامه عمل کرد الگوریتم پیشنهادی با فیلتر ذره‌ای استاندارد برای یک سیستم که درجه غیرخطی آن بالاست مقایسه شده است. نتایج شبیه‌سازی‌ها نشان می‌دهند عملکرد فیلتر پیشنهادی بهتر از عملکرد فیلتر ذره‌ای استاندارد در شرایط مختلف است.



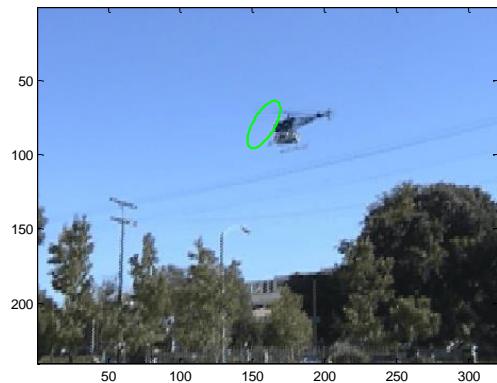
شکل ۱۹: سازگاری روش پیشنهادی

شکل‌های ۱۸ تا ۱۹ سازگاری فیلترها را نشان می‌دهند. در مقایسه با فیلتر ذره‌ای روش پیشنهادی از سازگاری بهتری برخوردار است. در فیلتر ذره‌ای بدلیل اینکه تابع پیشین پهن‌تر از تابع درست‌نمای می‌باشد به تعداد کمی از ذرات وزن بالایی اختصاص می‌یابد و بقیه ذرات وزن ناچیزی دارند. این مسئله سبب کم شدن تنوع میان ذرات و در نتیجه کاهش سازگاری فیلتر می‌شود. همانند قسمت قبل، برای ارزیابی بیشتر، عمل کرد روش پیشنهادی در شرایطی که نویز پروسه و اندازه‌گیری  $R_t = 1^2, Q_t = 5^2$  و تعداد ذرات  $N = 50$  با PF, UPF و EKF مقایسه شده است. نتایج نشان می‌دهد عمل کرد روش پیشنهادی بهتر از سایر روش‌ها است.

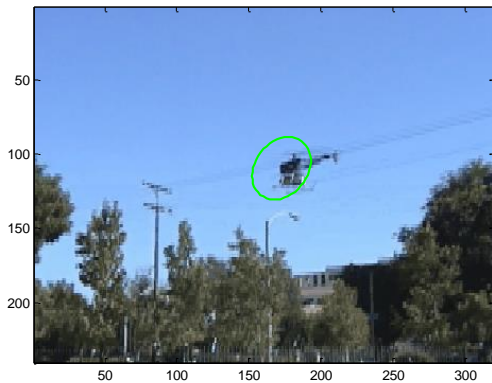
Archive of SID



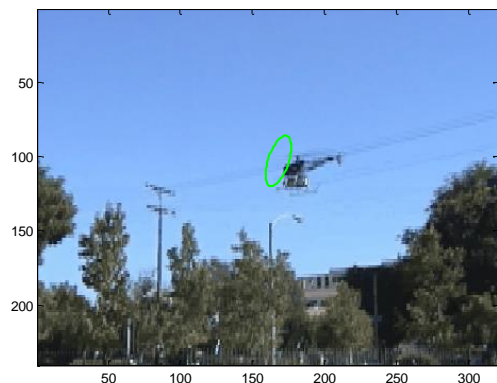
فرم ۱۶۲



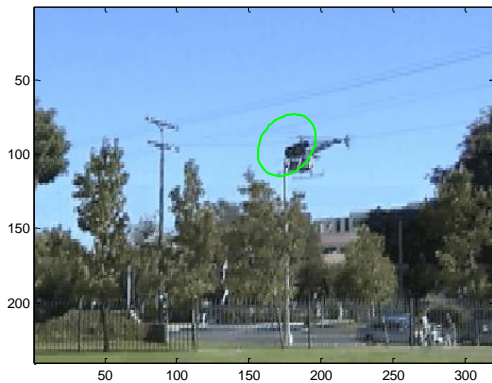
فرم ۱۶۲



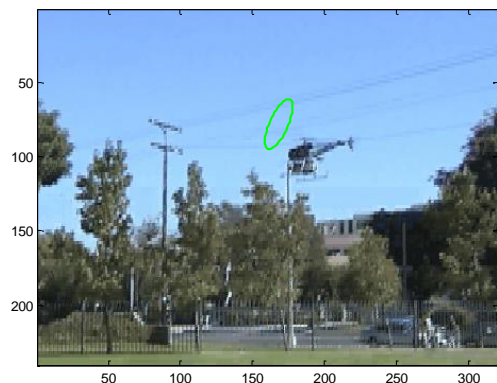
فرم ۲۲۴



فرم ۲۲۴



فرم ۲۴۱



فرم ۲۴۱

شکل ۲۲: ردیابی هلیکوپتر با فیلتر پیشنهادی

شکل ۲۱: ردیابی هلیکوپتر با فیلتر ذره ای

مراجع

- [1] S.Arulampalam, S.Maskell, N.Gordon, and T.Clapp, "A tutorial on particle filters for Online online/non-Gaussian Bayesian tracking", IEEE Trans. Signal Process., 2002, vol. 50, no. 2, pp. 174-188.
- [2] B.Ristic, S.Arulampalam, and N.Gordon, "in Beyond Kalman Filter: Particle filters tracking applications", 1st ed. Boston, 2004.
- [3] D.Simon, "Optimal State Estimation Kalman, H $\infty$  and Nonlinear Approaches", John Wiley and Sons, Inc, 2006.
- [4] R.Kalman, "A new approach to linear filtering and prediction problems", Trans. ASME J. Basic Eng., Ser. D, 1960, vol. 82, pp. 34-45.

زیر نویس ها:

- 1 Probability density function posterior
- 2 Particle filter
- 3 Degeneracy
- 4 Resampling
- 5 Auxiliary particle filter
- 6 Regularized particle filter
- 7 Particle swarm optimization
- 8 Joint posterior density
- 9 Target distribution
- 10 Proposal distribution
- 11 Sequence importance sampling
- 12 Sampling importance resampling
- 13 Joint posterior density
- 14 Mutation
- 15 Crossover
- 16 Root mean square error
- 17 Consistency
- 18 Normalized estimation error squared

- [5] S.C.Kramer, and H.W.Sorenson, "Recursive Bayesian estimation using piece-wise constant approximations", 1988, Automatica, vol. 24, no.6, pp.789-801.
- [6] N.Gordon, D.almond, and A.F.M.Smith, "Novel approach to nonlinear and non-Gaussian Bayesian state estimation", Proc. Inst. Elect. Eng., F, 1993, vol. 140, pp. 107-113.
- [7] A.Doucet, N.Gordon, and V.Krishnamurthy, "Particle filters for state estimation of jump Markov linear systems", IEEE Trans. Signal Processing, 2001, vol. 49, pp. 613-624.
- [8] M.S. Haque, S.Choi, and J.Baek, "Auxiliary Particle Filtering-Based Estimation of Remaining Useful Life of IGBT", IEEE Transactions on Industrial Electronics, 2018, vol.65, no.3, pp. 2693 - 2703.
- [9] A.Murangira, C.Musso, and K.Dahia, "A mixture regularized rao-blackwellized particle filter for terrain positioning, IEEE Transactions on Aerospace and Electronic Systems, 2016, vol. 52, no. 4, pp.1967-1985.
- [10] A.Murangira, M.Christian, C.Musso, and K.Dahia, "A mixture regularized rao-blackwellized particle filter for terrain positioning" IEEE Transactions on Aerospace and Electronic Systems, 2016, vol. 52, no. 4, pp.1967-1985.
- [11] G.Vezzani, U.Pattacini, G.Battistelli, L.Chisci, and L.Natale, "Memory Unscented Particle Filter for 6-DOF Tactile Localization", IEEE Transactions on Robotics, 2017, vol. 33, no. 5, pp.1139 - 1155.
- [12] Y.Zhang, S.Wang, and J.Li, "Improved particle filtering techniques based on generalized interactive genetic algorithm", Journal of Systems Engineering and Electronics, 2016, vol. 27, no. 1, pp.242 - 250
- [13] K.Uosaki, Y.Kimura, and T.Hatanaka, "Nonlinear State Estimation by Evolution Strategies Based Particle Filters", in: Proc. of the IEEE Congress on Evolutionary Computation, 2003.
- [14] G.Tong, Z.Fang, and X.Xu, "A Particle Swarm Optimized Particle Filter for Nonlinear System State Estimation", IEEE Congress on Evolutionary Computation Sheraton Vancouver Wall Centre Hotel, Vancouver, BC, Canada, July 16-21, 2006.
- [15] S.Park, J.Hwang, E. Kim, and H. Kang, "A New Evolutionary Particle Filter for the Prevention of Sample Impoverishment", IEEE Trans. On Evolutionary Computation, 2009, vol. 13, no. 4.
- [16] T.Li, M. Bolic, and P.M. Djuric, "Resampling Methods for Particle Filtering: Classification, implementation, and strategies", IEEE Signal Processing Magazine, 2015, vol. 32, no.3, pp. 70 - 86.
- [17] G.Choe, T. Wang, F.Liu, S. Hyon, J. Ha, "Particle filter with spline resampling and global transition model", IET Computer Vision, 2015, vol.1.9, no.2, pp.184-197.
- [18] F.Ruknudeen, and S.Asokan, "Application Particle Filter to On-Board Life Estimation of LED Lights", IEEE Photonics Journal, 2017, vol.9, no.3.
- [19] J. Liu, Z.Wang, and M.Xu, "A Kalman Estimation Based Rao-Blackwellized Particle Filtering for Radar Tracking", IEEE Access, 2017, vol.5, pp.8162-8174.
- [20] T.Li, T.P. Sattar, S.Sun, "Deterministic resampling: Unbiased sampling to avoid sample impoverishment in particle filters", Signal Processing, 2012, vol. 92, no.7, pp.1637-1645.
- [21] A.P.Engelbrecht, "Computational intelligence: An introduction, Second Edition", Johan Wiley & Sons, Ltd 2007.
- [22] H.Huang, F. Liu, X. Zhuo, and Z. Hao, "Differential Evolution Based on Self-Adaptive Fitness Function for Automated Test Case Generation", IEEE Computational Intelligence Magazine, 2017, vol.12, no.2, pp.46 - 55
- [23] Y.Yu, "Combining  $H_{\infty}$  filter and cost-reference particle filter for conditionally linear dynamic systems in unknown non-Gaussian noises", Signal Processing, 2013, vol.93, no.7, pp.1871-1878.