Monte Carlo Mean for Non-Gaussian Autonomous Object Tracking

L. Marata[a], J. Chuma[a], I. Ngebani[b], A. Yahya[a] and O.L.A. López[c]

maratal@biust.ac.bw, chumaj@biust.ac.bw,iboz55@gmail.com, yahyaa@biust.ac.bw, onel.alcarazlopez@oulu.fi

[a]*Botswana International University of Science and Technology*
[b]*University of Botswana*
[c]*University of Oulu*

Abstract

Object tracking is highly applicable in present day technologies such as electronic warfare, robotics and air surveillance. It is normally done using the measurements from sensors such as RADAR. Unfortunately due to the presence of deleterious noise, measurements are inaccurate. To address this problem, different estimation methods intended at reducing the effects of this noise have been developed. In spite of this, most methods were developed mainly for Gaussian noise, leaving non-Gaussian noise scenarios unresolved. Particle filters were introduced to address a more general noise scenario. Even so, particle filters are mathematically complex especially when used in high dimensional systems. To circumvent these problems, the present work proposes the Separate Monte Carlo Mean (SMC-MEAN) which is formulated on the Bayesian particle filtering framework. The proposed method is applied to an autonomous object tracking problem in both Gaussian and non-Gaussian distributed noise. Results of the proposed method are compared to the Kalman filter and Maximum A Posteriori (MAP) in Exponential and Logistic distributed noise using the mean squared error (MSE) and Cramér Rao lower bound (CRLB). The proposed method outperforms the other methods by an average of 17% yet maintaining low mathematical complexity.

Keywords: Autonomous tracking, Kalman Filter, Maximum A Posteriori, Particle Filtering, Cramer Rao lower bound.

1. Introduction

In almost all electronics and communication systems, e.g., telecommunications, computer vision and RADAR, there is a great need for application of signal processing techniques in order to enhance performance [1]. One of the commonly known problems associated with these systems is the deleterious noise that makes it hard to fully benefit from such electronics systems. Surgical robots mentioned in [2] are an example of an application which needs very accurate signal estimation. RADAR systems also need signal estimation for time delays [3]. Unmanned vehicles such as autonomous tractors are among applications which have not yet fully benefited from automation despite their potential use in precision farming [4, 5]. Autonomous objects have gained a lot of interest from researchers due to their widespread use in modern days. Some key areas which are still open to research are the visual object tracking for video traffic controls, human machine interactions and diagnostics machines among others [6]. These autonomous objects need trackers which can clearly initialize target as well positioning. One general problem reported in the literature is the additive noise which has challenged researchers to come up with robust motion estimators in presence of noise. A comprehensive survey of these issues is presented in [7] and [8].

To deal with the problems mentioned above, deterministic [9, 10] and Bayesian [11, 12] signal estimation methods have been used. Even though the deterministic approaches are important because of their ease of modeling, Bayesian methods are being highly recommended due to their ability to incorporate some information relating to the measurement under estimation [13–15]. Some of the classical estimators include the mean, median, maximum likelihood, Kalman filter, Maximum A Posteriori and Pitman estimators [16, 17]. Despite the plethora of the estimators, the estimation problem has not been fully solved because of the effects of the equipment and noise nature as reported in [2, 18], respectively. One other problem which always arises is the hard decision to choose one estimator over the other, mainly because of the different performances proved by the comparison between the mean and the median [19]. The difference in performance by different estimators is attributed to the way in which the noise is modeled. Owing to the limitations of the central limit theorem (CLT) and the law of large numbers (LLN) most of the noise is assumed to follow a Gaussian distribution and therefore good performance is only attained in scenarios where noise is truly Gaussian [18].

Motion estimation and tracking are key activities in many computer vision applications, e.g., video tracking, surveillance, automotive security [20]. It mainly concentrates on locating a changing state over time such as the speed of a moving vehicle over an interval of time. Gandhi *et al* in [21] derived a robust Kalman filter for Gaussian noise in the presence of observation outliers. Their work used batch mode Kalman filter based on the generalized maximum likelihood type estimator (GM-Kalman Filter). Applied in tracking problem, results showed that when two observations with noise variances of 10 m^2 and 100 m^2 were used, the GM-Kalman filter achieved an MSE of

$17,9m^2$ as compared to 73.5 m^2 achieved when a single observation of variance 100 m^2 was used. Furthermore, the GM-Kalman filter outperformed the classical Kalman filter in efficiency by 85%. In spite of the good effects of using many observations, on top of being limited to Gaussian noise scenario, it is a computationally expensive method. On the other hand, Barnerjee and Burlina in [22] proposed the use of the Support Vector Data Description (SVDD) to alleviate computational burden imposed by particle filters. Their experimentation was based on non-linear non-Gaussian dynamic system tracking and color based tracking. Their findings proved that the SVDD reduced the computational time by 50% as compared to the normal particle filtering framework. Even so, their work is mathematically complex for multivariate distributions.

A multiple object tracking on a Bayesian framework was introduced in [23] based on the Monte Carlo simulation particles. Authors proposed to track objects simultaneously by assuming the independence of the individual object tracks. Their work made different posteriors for the individual tracks even for targets maneuvering very close to each other. In this approach little mathematical complexity is maintained for multiple object tracking. Despite the promising results of their findings, their experimentation was only limited to the case where the random fluctuations of the velocity were Gaussian distributed. Moreover their work was limited to linear model of the motion in multiple target tracking.

The present work makes use of the idea in [23] but extending it to a single object tracking case in non-Gaussian noise. It computes single posteriors rather than the evaluation of multivariate distribution presented by [22]. Moreover the present work also makes a threshold ratio, say ξ of the highest particle weight before computing the mean. A threshold search is empirically done in the results section. This stems from the fact that weight of each particle w is such that $w \sim U[0,1]$. Mathematically it then follows from the Uniform distribution and the law of large numbers that

$$P_r[w \geqslant \xi w_{max}] \xrightarrow{a.s} 1 \tag{1}$$

Theoretically this will make all samples of weights to be within the upper 60^{th} percentile hence taking weights which are highly significant and more importantly improving the reliability of the estimate. The effective sample size maintained at each time step is

$$N_{p_{eff}} = \frac{1}{\sum_{i=1}^{N_p} (w_{i,N_p})^2} \tag{2}$$

where N_p is the total particle size. It is also worth noting that in contrast with works [21, 23, 24], which are limited to Gaussian noise tracks, the present work uses different noise distributions. Moreover a simpler particle filtering framework is used, by evaluating a series of single variate distributions as compared to [22] and [25], which evaluate multivariate distributions. It is shown that the proposed method outperforms the MAP and the Kalman filter as well as maintaining a small deviation from the CRLB in the linear case. It is also shown that the proposed method outperforms the MAP and the Extended Kalman filter (EKF) in the non-linear estimation.

Section 2 presents the motion model of an autonomous object in different noise scenarios, Section 3 presents the proposed algorithm, Section 4 presents the simulations results. Finally Section 5 concludes the paper and defines the future works.

2. State space model for the autonomous object

The autonomous object state parameters, which are to be estimated at each time step n where $n = 1, 2, ...N$, are the different evolving states which here we choose to represent with a variable χ. In linear case they are x or y positions while for the non-linear case are the range and bearing angle denoted by γ and θ, respectively. The corresponding measurement of individual state is denoted by ψ. Both the state and the measurements are assumed to have a noise of a given variance with a certain probability distribution function. The model of the generic dynamic system is given by (3) and (4),

$$\mathbf{s}_n = f(\mathbf{s}_{n-1}, \omega_{n-1}), \tag{3}$$
$$\mathbf{z}_n = g(\mathbf{s}_n, \mathbf{v}_n), \tag{4}$$

where functions f and g transform the Markov state process to the present state and measurement, respectively. $\mathbf{s}$ and $\mathbf{z}$ are the vectors of the state and the measurement at the time instant n respectively. In addition, system noises ω_{n-1} and $\mathbf{v}_n$ are modeled using the Exponential and Logistic distributions in function of χ as shown in (5) and (6), respectively.

$$p(\chi, \lambda) = \lambda e^{-\lambda\chi}, \quad \chi \in (0, \infty) \tag{5}$$
$$p(\chi, \mu) = \frac{1}{4s}\text{sech}^2\left(\frac{\chi - \mu}{2\alpha}\right), \quad \chi \in (-\infty, \infty) \tag{6}$$

In (5), λ represents the scale parameter, while in (6), α and μ represent the positive scale and the location parameter, respectively. It is worth noting that under ideal situations the state and measurement processes will be functions of the undisturbed state and measurement values e.g. $\mathbf{s}_n = f(\mathbf{s}_{n-1})$ and $\mathbf{z}_n = f(\mathbf{s}_n)$ respectively. Unfortunately there are always random fluctuations as reflected in (3) and (4). The present work uses (3) and (4) to track an autonomous moving object in non-Gaussian noisy measurement. This is done by points mimicking the true path and later computing a point estimate of the mimic points. The mimic points mean will in turn bring an estimate which approximates the undisturbed model (3).

3. Separate Monte Carlo Mean (SMC-MEAN) for Tracking

The generic particle posterior estimation is centered on the idea of sampling from wrong distributions. This method is based on simulation of multiple number of support points with associated weights. The weights of these points are proportional to their (the points)'s contribution in the overall probability distribution. This is done recursively in the case of modeling dynamic systems and therefore the weights will be evaluated at each step [26]. To ensure that these weights are indeed taken from a distribution function, they are normalized to bring the sum of all weights to one. This approach does not need an assumption of the Gaussian distribution which is a common practice. Moreover to be able to combine the state dynamics with the measurement, each time step is characterized by a prediction step which is based on the idealized state. This idealized state is expected to maintain a high variation from the expected values and the correction stage will allow for incorporation of the measurement to reduce the amount of error. This idea is summerized in Algorithm 1 which is based on [27, 28].

Algorithm 1 Generic particle filtering framework

Require: $\mathbf{s}_0, \mathbf{z}_n, K, N_p$
Ensure: $w_{i,n}$
1: BEGIN
2: $p(\mathbf{s}_n|\mathbf{z}_n) \leftarrow f\{p(\mathbf{s}_n|\mathbf{s}_{n-1}, \mathbf{z}_n)\}$
3: **for** $i = 1 : N_p$ **do**
4: $\quad q(\mathbf{s}_{i,n}|\mathbf{z}_{i,n}) \leftarrow f\{q(\mathbf{s}_n|\mathbf{s}_{n-1}, \mathbf{z}_n)\}$
5: $\quad \hat{w}_{i,n} \leftarrow w_{i,n-1}\frac{p(\mathbf{z}_n|\mathbf{s}_n)p(\mathbf{s}_{i,n}|\mathbf{s}_{i,n-1})}{q(\mathbf{s}_{i,n}|\mathbf{s}_{i,0:n-1},\mathbf{z}_{0:n})}$
6: $\quad w_{i,n} \leftarrow \frac{w_{i,n}}{\sum_{j=1}^{N_p}\hat{w}_{j,n}}$
7: **end for**
8: END

Let $\mathbf{s}_0$, $\mathbf{z}_n$,K, N_p, q, $\hat{w}_{i,n}$, $\hat{w}_{j,n}$ be the initial states, measurement vector at time index n, number of observations made, number of particles used, the mimic distribution, the i$^{\text{th}}$ unnormalized weights at time instant n and the j$^{\text{th}}$ weight running the normalizing sum respectively for the Algorithm 1 and the rest of the paper. $\chi_{0:n}$ is the state from time 0 to n$^{\text{th}}$observation while $\psi_{0:n}$ is measurement from 0 to n. The loop between line 2 and 8 in Algorithm 1 computes the overall posterior distribution while the loop between line 3 and 7 generates N_p particles for each time step observation. Line 5 makes the different mimic points for each particle while line 6 computes the weight of each particle. It is worth noting that the posterior filtering distribution in Algorithm 1 is evaluated from a multidimensional distribution. One problem which can be associated with this approach is that a combination of a bad estimate in one state and a good estimate can still yield a high weight. The present work addresses this limitation by assuming separate distribution for each object motion state by deriving the SMC-MEAN. Mathematically $\mathbf{s}$ is a vector of state variables at time n but these can be made to be independent assuming that

$$\mathbf{s}_n = \chi_1, \chi_2 .....\chi_m \tag{7}$$

This can be computed as the joint distribution whose marginal results in individual state projection as shown below for the case of the m^{th} separate state.

$$p(\chi_m) = \int p(\chi_1)\prod_{i=1}^{m} p(\chi_i|\chi_{i-1})d\chi_{1:m-1} \tag{8}$$

Having separated the distribution of the states then it can be assumed that the evolving posterior distribution of such a state can be computed using

$$p(\chi_{0:n}|\psi_{1:n}) = f\{p(\chi_{0:n-1}|\psi_{0:n-1}), \psi_n\}, \tag{9}$$

which is the conditional probability of the state given the measurement. Exploiting the Bayesian rule in [29] for Markov process and independent measurement then the posterior distribution is given by

$$p(\psi_{0:n}|\chi_{0:n}) = \frac{p(\psi_n|\chi_n)p(\chi_n|\chi_{0:n-1})p(\chi_{0:n-1}|\psi_{0:n-1})}{p(\psi_n|\psi_{0:n-1})}. \tag{10}$$

The Monte Carlo particle filter bases its approach on the fact that it is difficult to sample from the distribution $p(\chi_{0:n}|\psi_{0:n})$. The alternative would be to sample from a mimic distribution say $q(\chi)$, which can also be factorized as shown next,

$$q(\chi_{0:n}|\psi_{1:n}) = q(\chi_n, \chi_{0:n-1}|\psi_{1:n}). \tag{11}$$

Mathematically it can be shown that (8) reduces to (9) [27].

$$q(\chi_n, \chi_{0:n-1}|\psi_{1:n}) = q(\chi_n|\chi_{0:n-1}, \psi_{1:n})q(\chi_{0:n}|\psi_{0:n-1}). \tag{12}$$

This formulation shows that the mimic distribution resembles the true posterior distribution since they are defined by the same process. To compute the overall posterior distribution up to time instant n, then weights can be computed. These weights are the contribution of the Monte Carlo mimic points to the true points as the state is evolving. It is worth noting that the true posterior distribution is known only by means of computation of the likelihood and the prior. This therefore means that the weights are proportional to the distribution as shown below

$$w_n \propto \frac{p(\psi_n|\chi_n)p(\chi_n|\chi_{n-1})p(\chi_{0:n-1}|\psi_{0:n-1})}{q(\chi_n|\chi_{0:n-1}, \psi_{1:n})q(\chi_{0:n-1}|\psi_{0:n-1})}. \tag{13}$$

The complete discrete posterior distribution is then given by

$$p(\chi_{0:n}|\psi_{1:n}) \approx \sum_{i=1}^{N_p} w_{i,n}\delta(\chi_{0:n} - \chi_{i,0:n}), \tag{14}$$

which represents the overall posterior distribution of the whole state. This posterior distribution can be simplified for the state at a time index n by computing the marginal distribution, which according to [30] yields

$$p(\chi_n|\psi_{1:n}) \approx \sum_{i=1}^{N_p} w_{i,n}\delta(\chi_n - \chi_{i,n}), \tag{15}$$

With the availability of the posterior, any other computation can be made on this posterior including estimator analysis such as CRLB, MAP as well as the closed form expression, just to mention a few. The value of N_p is chosen arbitrarily such a way that it gives a good estimate, also maintaining good computation time. Hence, to avoid the particle degeneracy problem, which is outlined in [27], this work makes a threshold particle weight of ξ of the highest weight and anything less than the threshold is replaced by the highest weight particle. Schematically the

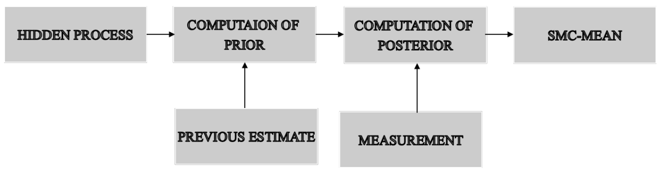

Figure 1: SMC-MEAN block diagram

block diagram in Figure 1 is used to represent how the SMC-MEAN is applied. The hidden process block contains the idealized evolving state, known mathematically. Computation of the prior is the block that focuses on finding the predicted values based on the previous estimate contained in the previous estimate block. The correction of the prediction is done in the computation of the posterior making use of the measurement. Algorithm 2 is used to estimate each state at a given time.

Algorithm 2 Separate Monte Carlo mean (SMC-MEAN)

Require: $\chi_0, \psi_n, K, N_s, v_x[0], v_y[0]$
Ensure: $\hat{\chi}$
1: BEGIN
2: **for** $n = 1 : K$ **do**
3: $\quad p(\chi_n|\psi_n) \leftarrow f\{p(\chi_n|\chi_{n-1}, \psi_n)\}$
4: $\quad$ **for** $i = 1 : N_s$ **do**
5: $\quad\quad q(\chi_{i,n}|\psi_{i,n}) \leftarrow f\{q(\chi_n|\chi_{n-1}, \psi_n)\}$
6: $\quad\quad w_{i,n} \leftarrow \frac{p(\chi_n|\psi_n)}{q(\chi_{i,n}|\psi_{i,n})}$
7: $\quad$ **end for**
8: $\quad$ **for** $i : N_p$ **do**
9: $\quad\quad TR = max(w_n)$
10: $\quad\quad$ **if** $w_i < \xi * TR$ **then**
11: $\quad\quad\quad \chi, w_i \leftarrow \chi, max(w_n)$
12: $\quad\quad$ **end if**
13: $\quad\quad \hat{\chi} \leftarrow \frac{1}{N_p} \sum_{i=1}^{N_p} \chi_i$
14: $\quad$ **end for**
15: **end for**
16: END

Let χ_0 ,ψ_n,K,N_s, $v_x[0]$,$v_y[0]$ be the separate state, separate measurement at each time index n, number of observations made, number of particles used, initial velocities in x and y axis respectively. Loop between line 2 and 15 of algorithm 2 produces the time step point estimate of the true track position. Lines 4-12 compute the particle weights and compares particle weights to the ξ of the maximum weight. Similar to Algorithm 1, lines 4 to 7 in Algorithm 2 generate particles and computes the weight. One major difference between the two approaches is that the Algorithm 2 performs the particle generation for each state. After the weights have been computed, there will be those weights for mimics which are not close to the true path. Taking these weights into consideration might produce a bad estimate. Moreover work [27] and [31] mentions the degeneracy problem which might arise from taking weak mimic points which is why in this work they are eliminated. One key advantage of the SMC-MEAN is that it is capable of accounting for non-Gaussian noise scenarios which are generally more realistic since prior knowledge of the posterior is normally unavailable.

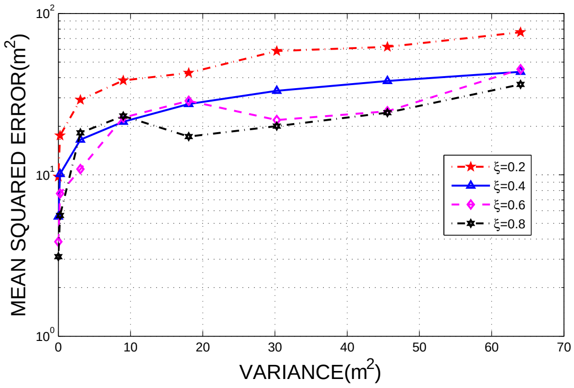

Figure 2: MSE of the estimators using different thresholds

4. Results and Analysis

The complete model used in this work for linear model is given by equation (16) and equation (17) while for non-linear it is given by equation (16) and equation (18).

$$\begin{bmatrix} x \\ y \\ \dot{x} \\ \dot{y} \end{bmatrix} = \begin{bmatrix} 1 & 0 & \Delta T & 0 \\ 0 & 1 & 0 & \Delta T \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x[n-1] \\ y[n-1] \\ \dot{x}[n-1] \\ \dot{y}[n-1] \end{bmatrix} + \begin{bmatrix} w_x \\ w_y \\ w_{\dot{x}} \\ w_{\dot{y}} \end{bmatrix} \tag{16}$$

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} x[n-1] \\ y[n-1] \\ \dot{x}[n-1] \\ \dot{y}[n-1] \end{bmatrix} + \begin{bmatrix} w_x \\ w_y \end{bmatrix} \tag{17}$$

$$\begin{bmatrix} \gamma \\ \theta \end{bmatrix} = \begin{bmatrix} \sqrt{x^2[n] + y^2[n]} + v_\gamma \\ \arctan\left(\frac{y[n]}{x[n]}\right) + v_\theta \end{bmatrix} \tag{18}$$

where w_x, w_y,$w_{\dot{x}}$,$w_{\dot{y}}$,v_γ and v_θ are the noise terms corresponding to the state in the subscript and they were initialized at a value of 0.5 while the velocities $\dot{x}$ and$\dot{y}$ were set at 10 m/s and 15 m/s respectively and ΔT is the sampling time set at 0.1s. The proposed method was compared to other estimation methods in three different scenarios being, linear non-Gaussian, non-linear Gaussian and non-linear non-Gaussian tracks. In order to improve the accuracy of the Mean Squared Error (MSE), 5000 Monte Carlo simulations were done on HP Intel(R) Celeron(R) CPUN2830. For the linear case the MSE was analyzed when the measurement is reported as a linear function of the x and y-axis. Linear results were also compared to the CRLB. For the non-linear case the performance analysis was done when the measurement was reported as a function of the bearing and the range.

Figure 2 shows the search for ξ which gives performance of SMC-MEAN using different threshold values, which ultimately influenced the threshold (ξ) for this work to be set at 0.4 due to consistency. Furthermore to choose the number of particles used, an experiment was run for MSE against particle size as reported in Figure 3. According to the results any mimic particle

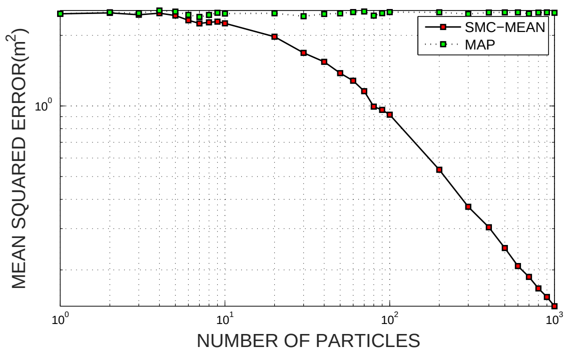

Figure 3: MSE for different Particle size in Logistic Noise at fixed variance

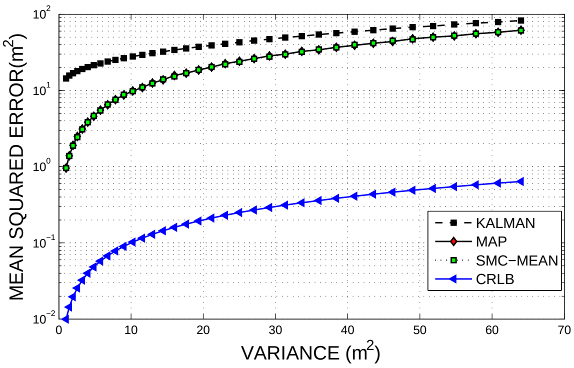

Figure 4: MSE of the estimators in Exponential noise for different variances

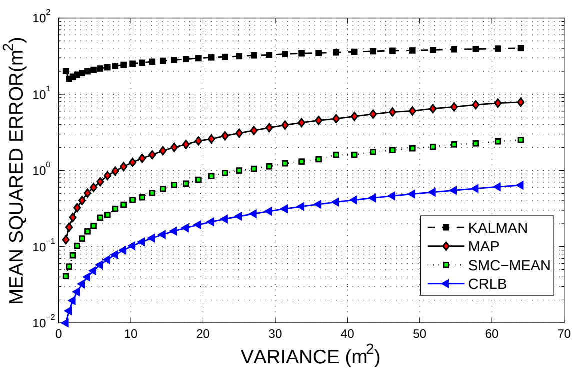

Figure 5: MSE of the estimators in Logistic noise for different variances

size value above 300 successfully drives the MSE to low values. Taking this into consideration and computational burden of many particles, 500 particles are used in this work.

Results in Figure 4 shows the performance in exponential noise scenario in linear track. The MSE results of the MAP and the SMC-MEAN are almost indistinguishable. One reason which could be attributed to this, is the fact that they are computed from the same posterior distribution. Moreover consistent with [32] this can be linked to the complexity of non-negative valued noise. On the other hand results in Figure 5 show that the SMC-MEAN performs very well in logistic distribution. Furthermore, this occurrence can be attributed to the fact that the Logistic distribution is heavy-tailed but has a centralized bell

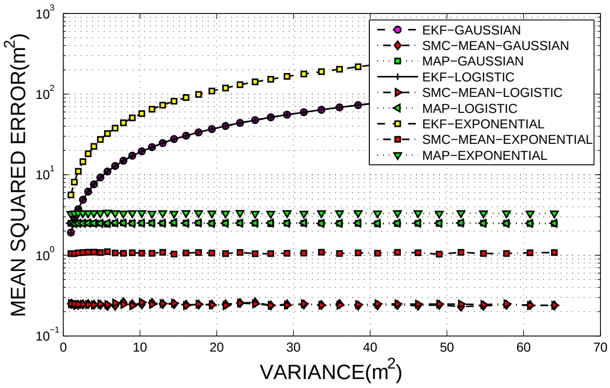

Figure 6: Comparison of different estimation methods in different distributions for Range-Bearing estimates

shaped distribution with central mean. This is an interesting result since the proposed SMC-MEAN is able to estimate even in heavy-tailed distributed noise. In both figure 5 and 4 show the performance of the Kalman filter to be poor in non-Gaussian noise.

Results in the range from Figure 6 show that the SMC-MEAN is both consistent and almost optimal as it attained an MSE less than or equal to 1 m^2, outperforming the MAP and the EKF. Even so, the MAP is also consistent due to its computation from the posterior distribution. EKF, as expected performs poorly in Exponential noise as it is mostly designed for Gaussian distributions. From the same Figure 6 it can be deduced that the SMC-MEAN is consistent especially in non-linear estimation problems. Even though the Logistic distribution is a heavier-tailed distribution than the Gaussian, the SMC-MEAN and the MAP portray similar behavior in Logistic and Gaussian noise. This can be attributed to the fact that both distributions are bell shaped taking positive and negative values. Furthermore the SMC-MEAN is consistent for both variances of $0,25\ m^2$ and $64\ m^2$ which proves its robustness to highly dispersed data. Another interesting results is that in non-linear object tracking in Exponential noise, the SMC-MEAN and MAP attained better results than those for linear case as shown by Figure 4. This shows that exponential noise requires a prior compensation which comes by default in non-linear functions used.

5. Conclusions and Future Works

The SMC-MEAN was derived based on the Bayesian particle filtering framework. Results from the present work suggested that the separability of the individual tracks in the motion model allows for a simple mathematical estimation model. It has also been deduced that the proposed method is efficient in heavy-tailed distributions especially those with well defined moments as shown by an average MSE of approximately 3 at a variance of 60. The present work has in overall successfully tracked objects in different noise tracks.

Despite the good performance of the proposed methods, it still experiences difficulty in estimating linear path with exponential noise as shown by an MSE of approximately 50 at a variance of 60. This could be further investigated in future works. Furthermore, the proposed method is based on the particle filtering framework which is known to be a compromise between the number of particles and good estimates. It makes it important to improve on the proposed methods to be able to maintain high accuracy with little number of particle. This work was limited to additive noise and therefore future works can consider the same approach in non-additive noise such as multiplicative.

References

[1] F.-L. Luo, W. Williams, R. M. Rao, R. Narasimha, M.-J. Montpetit, Trends in signal processing applications and industry technology [in the spotlight], IEEE Signal Processing Magazine 29 (1) (2012) 184–174.

[2] M. O. Hawksford, E. S. Gower, Analysis of sensor impulse response effects on cramér rao lower bounds for signal parameter estimators, The Journal of Engineeringdoi:10.1049/joe.2014.0230.

[3] G. Y. Kulikov, M. V. Kulikova, The accurate continuous-discrete extended kalman filter for radar tracking, IEEE Transactions on Signal Processing 64 (4) (2016) 948–958. doi:10.1109/TSP.2015.2493985.
URL http://ieeexplore.ieee.org/document/7307216/

[4] T. Hague, J. Marchant, N. Tillett, Ground based sensing systems for autonomous agricultural vehicles, Computers and Electronics in Agriculture 25 (1) (2000) 11–28. doi:10.1016/S0168-1699(99)00053-8.

[5] R. Eaton, J. Katupitiya, K. W. Siew, B. Howarth, Autonomous Farming: Modeling and Control of Agricultural Machinery in a Unified Framework, in: 2008 15th International Conference on Mechatronics and Machine Vision in Practice, IEEE, 2008, pp. 499–504. doi:10.1109/MMVIP.2008.4749583.
URL http://ieeexplore.ieee.org/document/4749583/

[6] H. Yang, L. Shao, F. Zheng, L. Wang, Z. Song, Recent advances and trends in visual tracking: A review, Neurocomputing 74 (18) (2011) 3823–3831.

[7] M. Fiaz, A. Mahmood, S. K. Jung, Tracking noisy targets: A review of recent object tracking approaches, arXiv preprint arXiv:1802.03098.

[8] L. Leal-Taixé, A. Milan, K. Schindler, D. Cremers, I. Reid, S. Roth, Tracking the trackers: an analysis of the state of the art in multiple object tracking, arXiv preprint arXiv:1704.02781.

[9] A. Monakov, Maximum-likelihood estimation of parameters of an extended target in tracking monopulse radars, IEEE Transactions on Aerospace and Electronic Systems 48 (3) (2012) 2653–2665.

[10] M. Amini-Omam, F. Torkamani-Azar, S. A. Ghorashi, Maximum likelihood estimation for multiple camera target tracking on grassmann tangent subspace, IEEE Transactions on Cybernetics.

[11] L. Mihaylova, A. Y. Carmi, F. Septier, A. Gning, S. K. Pang, S. Godsill, Overview of bayesian sequential monte carlo methods for group and extended object tracking, Digital Signal Processing 25 (2014) 1–16.

[12] J. H. Kotecha, P. M. Djuric, Gaussian sum particle filtering, IEEE Transactions on signal processing 51 (10) (2003) 2602–2612.

[13] T. Gerkmann, Bayesian Estimation of Clean Speech Spectral Coefficients Given a Priori Knowledge of the Phase, IEEE Transactions on Signal Processing 62 (16) (2014) 4199–4208. doi:10.1109/TSP.2014.2336615.
URL http://ieeexplore.ieee.org/document/6850043/

[14] M. Han, E-bayesian estimation and hierarchical bayesian estimation of failure rate, Applied Mathematical Modelling 33 (4) (2009) 1915–1922.

[15] R. Jinan, T. Raveendran, Particle filters for multiple target tracking, Procedia Technology 24 (2016) 980–987.

[16] M. Pereyra, P. Schniter, E. Chouzenoux, J.-C. Pesquet, J.-Y. Tourneret, A. O. Hero, S. McLaughlin, A survey of stochastic simulation and optimization methods in signal processing, IEEE Journal of Selected Topics in Signal Processing 10 (2) (2016) 224–241.

[17] H. Fang, N. Tian, Y. Wang, M. Zhou, M. A. Haile, Nonlinear bayesian estimation: from kalman filtering to a broader horizon, IEEE/CAA Journal of Automatica Sinica 5 (2) (2018) 401–417.

[18] E. Gower, T. Tsalaile, M. Kgwadi, M. Hawksford, Robust coherent noise suppression by point estimation of the cauchy location parameter, World Academy of Science, Engineering and Technology, International Journal of Mathematical, Computational, Physical, Electrical and Computer Engineering 7 (2) (2013) 262–268.

[19] M. Sherman, Comparing the Sample Mean and the Sample Median: An Exploration in the Exponential Comparing the Sample Mean and the Sample Median: An Exploration in the Exponential Power Family, Source: The American Statistician 51 (1) (1997) 52–54.

[20] A. Yilmaz, O. Javed, M. Shah, Object Tracking: A Survey, ACM Comput. Surv. Article ACM Computing Surveys 38 (13). doi:10.1145/1177352.1177355.
URL http://doi.acm.org/10.1145/

[21] M. A. Gandhi, L. Mili, Robust Kalman Filter Based on a Generalized Maximum-Likelihood-Type Estimator, IEEE Transactions on Signal Processing 58 (5) (2010) 2509–2520. doi:10.1109/TSP.2009.2039731.
URL http://ieeexplore.ieee.org/document/5371933/

[22] A. Banerjee, P. Burlina, Efficient Particle Filtering via Sparse Kernel Density Estimation, IEEE Transactions on Image Processing 19 (9) (2010) 2480–2490. doi:10.1109/TIP.2010.2047667.
URL http://ieeexplore.ieee.org/document/5443441/

[23] W. Yi, M. R. Morelande, L. Kong, J. Yang, A Computationally Efficient Particle Filter for Multitarget Tracking Using an Independence Approximation, IEEE Transactions on Signal Processing 61 (4) (2013) 843–856. doi:10.1109/TSP.2012.2229999.
URL http://ieeexplore.ieee.org/document/6362269/

[24] X. Hu, M. Bao, X.-P. Zhang, L. Guan, Y.-H. Hu, Generalized Iterated Kalman Filter and its Performance Evaluation, IEEE Transactions on Signal Processing 63 (12) (2015) 3204–3217. doi:10.1109/TSP.2015.2423266.
URL http://ieeexplore.ieee.org/document/7086336/

[25] B. Ait-El-Fquih, I. Hoteit, A Variational Bayesian Multiple Particle Filtering Scheme for Large-Dimensional Systems, IEEE Transactions on Signal Processing 64 (20) (2016) 5409–5422. doi:10.1109/TSP.2016.2580524.
URL http://ieeexplore.ieee.org/document/7491275/

[26] N. J. Gordon, D. J. Salmond, A. F. Smith, Novel approach to nonlinear/non-gaussian bayesian state estimation, in: IEE Proceedings F (Radar and Signal Processing), Vol. 140, IET, 1993, pp. 107–113.

[27] M. Arulampalam, S. Maskell, N. Gordon, T. Clapp, A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking, IEEE Transactions on Signal Processing 50 (2) (2002) 174–188. doi:10.1109/78.978374.
URL http://ieeexplore.ieee.org/document/978374/

[28] A. De Freitas, L. Mihaylova, A. Gning, D. Angelova, V. Kadirkamanathan, Autonomous crowds tracking with box particle filtering and convolution particle filtering, Automatica 69 (2016) 380–394.

[29] A. J. Haug, Bayesian estimation and tracking: A practical guide, Wiley, 2012.

[30] J. Kotecha, P. Djuric, Gaussian particle filtering, IEEE Transactions on Signal Processing 51 (10) (2003) 2592–2601. doi:10.1109/TSP.2003.816758.
URL http://ieeexplore.ieee.org/document/1232326/

[31] T. Schon, F. Gustafsson, P.-J. Nordlund, Marginalized particle filters for mixed linear/nonlinear state-space models, IEEE Transactions on Signal Processing 53 (7) (2005) 2279–2289.

[32] D. N. Aloi, O. V. Korniyenko, Comparative Performance Analysis of a Kalman Filter and a Modified Double Exponential Filter for GPS-Only Position Estimation of Automotive Platforms in an Urban-Canyon Environment, IEEE Transactions on Vehicular Technology 56 (5) (2007) 2880–2892. doi:10.1109/TVT.2007.900396.
URL http://ieeexplore.ieee.org/document/4305475/