Architecture Diagram to Track an Object via Machine Learning for Bound Error of Particle Filter

Qui Vo –Phu[1], Tuan Nguyen-Hoang[2], Nga Ly-Tu[3]
[1,2,3]*School of Computer Science and Engineering, International University-VNUHCM, Vietnam*

Abstract- In this paper, we propose an approach to find bound error of Kullback-Leibler Distance (KLD)-particle filter (PF) based machine learnings to solve the nonlinear target tracking in Wireless sensor network (WSN). It is essentially to estimate the target state from the effect of the received signal strength (RSS) variations or observation model undergoes nonlinearities. Here, the value bound error of KLD resampling plays an important role in estimation accuracy and convergence rate of declining number of particle used. To ameliorate the effect of the RSS variations by generating a sample set near the high-likelihood region, our proposal considers how to find the bound error of KLD PF for each iteration. The first iteration, using the observation information via KLD resampling with optimal bound error to conduct a resampling on the basis of the initial bound error. From the second to the end iteration, we propose the KNN technique to search the predicted bound error value that fulfills the minimum of mean number of particle used between at the current and the next iteration. Simulation results show that the proposed learnings can obtain better estimation accuracy with shorter computational time and improve the efficient number of particles compared to traditional methods.
Keywords – SIR, bound error, KLD-resampling, RSS, KNN, LDA

I. INTRODUCTION

Using weighted particle set with assigned primary weights serves as the basic idea of a particle filter (PF) is one of these methods to improve the estimation of target location in space, called a recursive Bayesian filter in [1]. Monte Carlo method also use a set of weighted particle to realize the recursive Bayesian filter for an effective nonlinear non-Gaussian system, called suboptimal prediction method, applied in the field of target tracking in [2].

The resampling step is a critical procedure for PF to avoid a degenerate set of particles (sample impoverishment) leading to the estimation inaccuracy. There are many methods are introduced such as initially employed to combat degeneracy in [3], generally replicating high-weighted particles to replace low-weighted particles in [4] for reducing the probability that the filter loses tracking.

A number of authors have considered the effects of choosing metric and weight functional approach on PFs in [5-9]. The first approach, the PF based on Kullback-Leibler Distance (KLD)-sampling, determines the minimum number of particles needed to maintain the approximation quality in the sampling process. Meanwhile, the authors in [9] introduced adjusting standard deviation and then using gradient data for KLD-sampling to further improve the operation time and sample set size for target tracking thanks to the given upper bound error with fixed probability. In the KLD-sampling in [7], the predictive belief state is used as the estimate of the underlying posterior to improve the micro-ability and adaptability of particle set. The noise variance of the new information estimation system is determined based on reflect the relationship between the accuracy of the target prediction and the uncertainty of the system. It uses to determine the sampling of the proposed distribution. The authors in [10] enhanced the ability to predict the particle set via the new information of observation to control the number of particles double sampling. Currently, the authors in [11] applied the trained network for KLD sampling to generate the new bin size though space division by KD tree that helps balance between approximation error and runtime.

While KLD-resampling algorithms in [5, 8,14] also determine the number of particles to resample so that the KLD between the distribution of particles before resampling and after resampling does not exceed a pre-specified bound error. Our current works in [8,14] introduced an enhanced PF based on the finding optimal bound error for KLD-resampling. But the optimal bound error in here is maintained during the online training data, it is still open problem.

To overcome it, we propose architecture diagram to collect and predict the bound error values for online training data of the problem in [8,14]. Our first proposal is to extend for our method in [15] (K-Nearest Neighbor-KNN technique) to find the nearest bound errors for all classes that reach the minimal of mean number of particles used. Our second proposal is to transform the features (iterations of finding bound error) into a lower dimensional space, called Linear Discriminant Analysis-LDA technique, which maximizes the ratio of the between-class variance to the within-class variance.of noises or the changing of RSS measurements. Our experiments show that combining supervised machine learning in [13] to the predict bound error for KLD resampling based RSS measurements in WSN system in [8,9,12,14,15] not only enhances the estimation accuracy but also improves the efficient number of particles used when compared to traditional methods. Our methods are also the another latest technique in [11] to apply the trained data for KLD-resampling by generating the predicted bound error based supervised machine learnings that helps balance between approximation error (RMSE) and runtime.

The paper is organized as follows. Our proposed algorithm is presented in Section II such as architecture diagram and an example to explain to our model by applied KNN and LDA methods for KLD PF. All experimental results based on PYTHON for target tracking are shown in Section III. Finally, we conclude the paper in Section IV.

II. PROPOSED ALGORITHM

2.1 Architecture diagram

We propose the architecture diagram model to train the bound error of KLD PF via Machine learning in WSN systems.

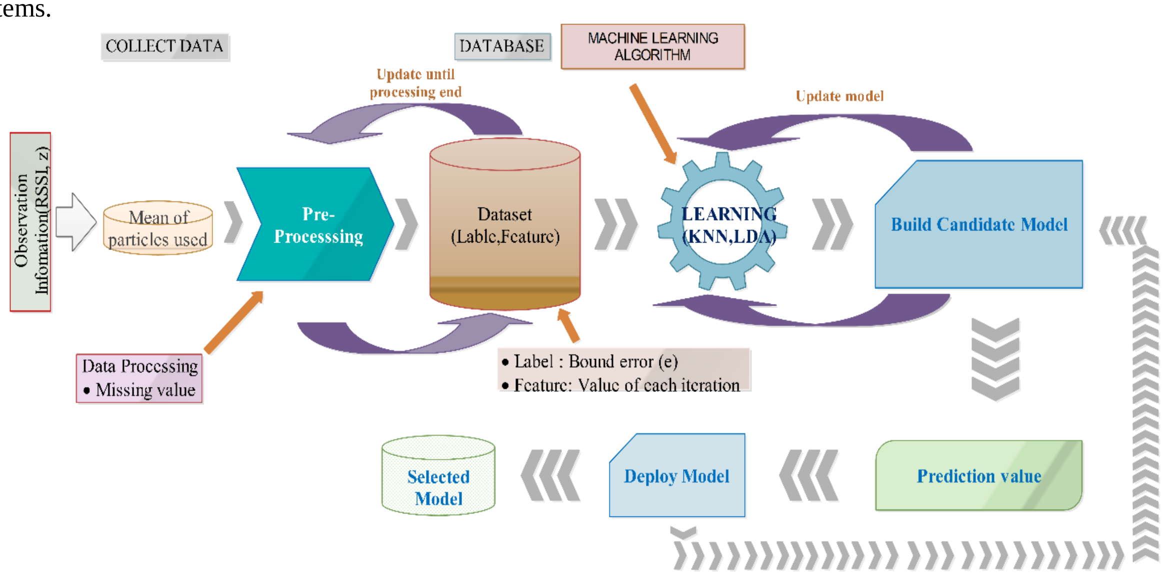


Figure 1. Architecture diagram to track a target based supervised machine learning-KLD in WSN

First, our diagram collects the observation information based the system state model for the mobile wireless sensor in [12] is defined as follows

$$x_k = x_{k-1} + V_k \Delta t + Q w_k, \tag{1}$$

$$z_k = Pref + K log\left(x_k\right) + R v_k, \tag{2}$$

where x_k is the position of a mobile node from the anchor, Δt is the time segment, z_k is the RSS measurement; V_k is the current velocity which consists of determined velocity and random velocity; the w_k and v_k denote the system state noise and measurement noises which obey Gauss distributions whose mean are 0 and variances are Q and R, respectively; Pref is reference value of RSS, and K is the factor in path loss.

To estimate the target location, weighted particle set with assigned primary weights serves as the basic idea of a particle filter (PF) is called a recursive Bayesian filter in [1]. This is a suboptimal prediction method applied in the field of target tracking in [2]. In the sampling process, as these individuals in the population are sorted by non-domination, the use of fast KLD-sampling technique in [7], called an adaptive PF at each iteration of the PF, determine the number of samples such that, with probability $1-\delta$, the error between the true posterior and the sample-based approximation is less than ε. KLD is used to show how to determine the number of samples so that the distance between the sample-based maximum likelihood estimate and the true posterior does not exceed a pre-specified threshold ε. The KLD between the proposal (q) and (p) distributions can be defined in discrete form as follows

$$d_{KL}\left(p \| q\right) = \sum_{x} p\left(x\right) log\left(\frac{p\left(x\right)}{q\left(x\right)}\right). \tag{3}$$

The required number Nr of samples can be determined as follows $N_r = \chi^2_{k-1,1-\delta} / 2\varepsilon$, where k is the number of bins with support, the quantizes of Chi-square distribution can be computed as follows $P\left(\chi^2_{k-1} \le \chi^2_{k-1,1-\delta}\right) = 1-\delta.$
Second, to collect data based the mean particle used criterion in Figure 1 as follows

$$N_r = \frac{k-1}{2\varepsilon}\left(1-\frac{2}{9(k-1)}+\sqrt{\frac{2}{9(k-1)}}z_{1-\delta}\right)^3, \quad (4)$$

where $z_{1-\delta}$ is the upper quartile of the standard normal distribution.
Our current work, Ly-Tu et. al. in [8,9], introduced that the collected bound error range of KLD–resampling in [8] (See in Algorithm 6) from 0.7 to 0.975, the value of variances R and Q in (1) and (2) from 0.1 to 0.9, in all cases of number particles N (N=100, 200, 600) to track the target in WSN system. Based on this model, the mean number of particle used, bound error, and runtime are stored in one file excel. An example of values R=Q=0.5, N=100 for 9 classes of Label (called bound error in Figure 1, namely Epsilon in Figure 2) is shown in Figure 2.

	Epsilon	Iter1	Iter2	Iter3	...	Iter37	Iter38	Iter39	Iter40
0	0.700	20.95	10.00	6.00	...	6.90	6.85	6.95	6.95
1	0.700	20.30	7.95	5.35	...	6.65	6.80	6.75	6.80
2	0.700	21.10	8.20	5.35	...	6.85	6.90	6.95	7.00
3	0.700	20.45	8.60	6.15	...	6.85	6.90	6.85	6.75
4	0.700	20.65	8.15	5.50	...	6.90	6.95	6.95	6.85
5	0.700	20.40	8.30	5.65	...	7.00	6.95	6.95	7.00
6	0.700	20.20	8.25	4.90	...	7.00	7.00	7.00	7.00
7	0.700	20.95	8.55	5.45	...	6.90	7.00	6.95	6.90
8	0.700	20.65	8.95	6.00	...	7.00	7.00	6.95	7.00
9	0.700	19.90	7.85	5.50	...	6.95	6.90	6.95	6.95
..	...	...	...	...	...	...	...	...	...
453	0.975	14.70	4.65	3.35	...	2.00	2.00	2.00	2.00
454	0.975	15.00	5.40	3.45	...	2.00	2.00	2.00	2.00
455	0.975	14.55	5.50	3.55	...	2.00	2.00	2.00	2.00
456	0.975	14.75	5.30	3.75	...	2.00	2.00	2.00	2.00
457	0.975	14.30	5.10	3.75	...	2.05	2.05	2.05	2.05
458	0.975	14.45	5.70	3.75	...	2.05	2.05	2.05	2.05

[459 rows x 41 columns]

Figure 2. Dataset of 9 classes of Epsilon in case of N=100

Each class has 51 rows and 41 columns. The first column is described the label of class (Epsilon) and the 40 next columns are assigned as 40 features. In order to overcome the process of these missing values (Pre-processing in Figure 1), we follow the first method of four ones in [13] to remove these missing data.
The objective of our proposal is to find the bound error (Epsilon) to reach the minimal of mean number of particle used based KLD-resampling adjusted bound error in [8] (see in Algorithm 6), therefore our works introduce the bound error algorithm with the initial bound error in [8] applied the first iteration and the predicted bound errors based KNN or LDA as shown in Figure 3.

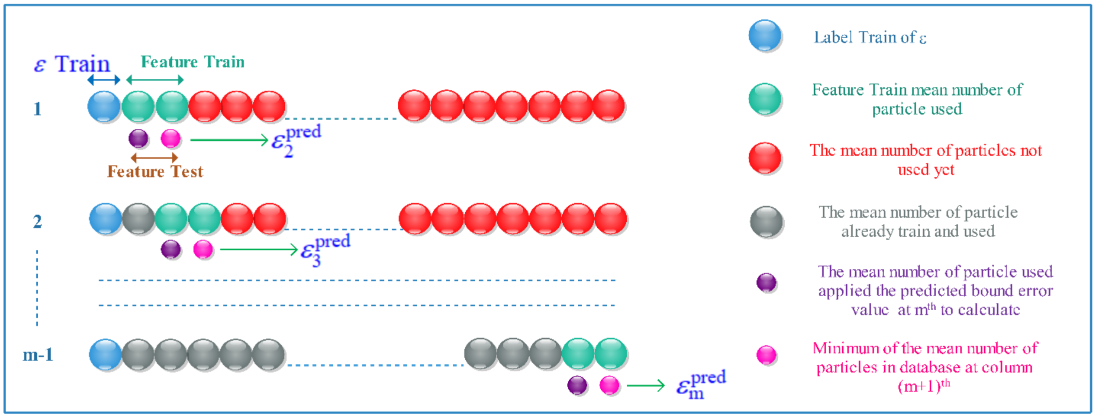


Figure 3. Predict the bound error for our dataset model

To more analysis this dataset in Figure 2, three classes are selected as the first class (0.7), the middle (0.875) and the last ones (0.975) for evaluating the overlap them during the first four iterations vs. the effect of the variance noise Q in (1) or the fluctuation of RSS measurements in (2) as shown in Figure 4, we can select the candidate model to deploy in reality. In here, if the predicted bound error is not satisfied conditions the performance of RMSE criterion in [8], it is removed out of the selected list. The output of our architecture diagram in Figure 1 is the predicted bound error which fulfills both the mean number of particle used and RMSE.

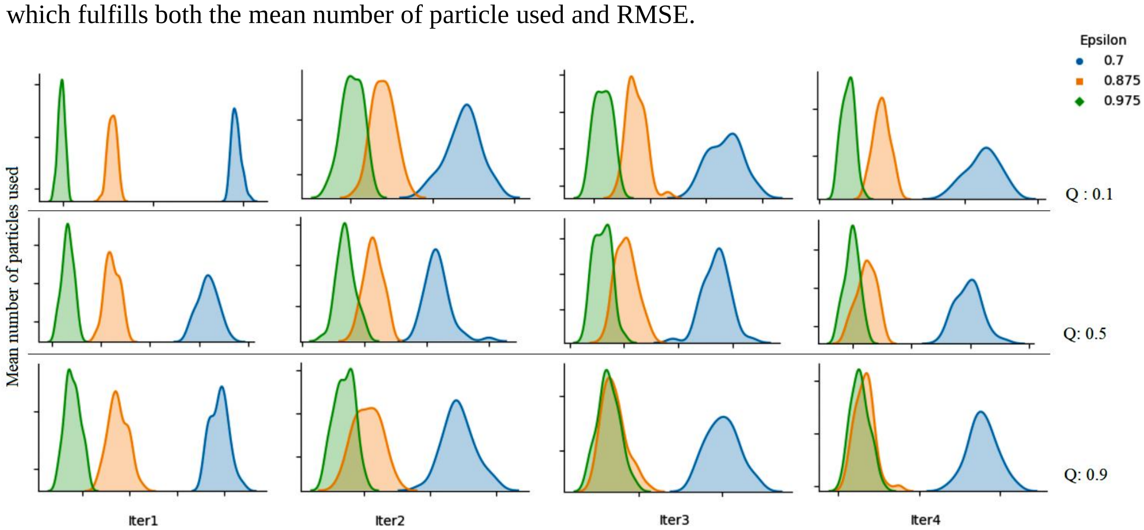
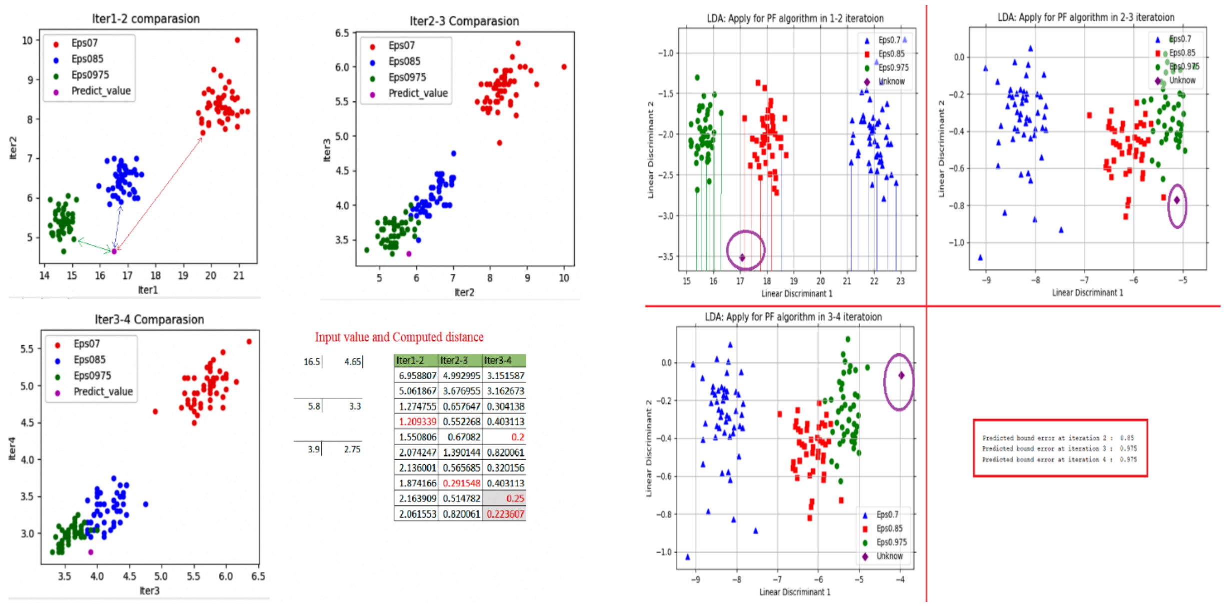


Figure 4. The diagonal chart of analysis for three selected classes.

2.2. An example to find the predicted bound error in case of R=Q=0.5, N=100

To apply and evaluate our dataset model, we propose KNN in [17] or LDA in [18] techniques to find the predicted bound error value for online training [13]. Our method is to apply bound error value in [6] as the new observation information for only the first iteration, called the initial bound error. Then it uses to conduct for finding others bound error based on KNN search or LDA, called the predicted bound errors, that fulfills the minimum of mean number of particle used between at the current and next iterations as shown in Figure 4.



(a) (b)

Figure 5. Describe how to search bound error for the first four iterations via (a) KNN and (b) LDA

III. EXPERIMENT AND RESULT

Setting up to track an object of our systems in [8,9] as follows R=0.5, Nmax=N; Vmax=5; Vmin=1; Vinit=5; Pref =-23; K=-45; a number of samples N =100, a range of variances Q (0.1,0.3,0.5,0.7,0.9), length time is 40 for sample size variation in 20 trials as shown in Table 1.

Table 1. Mean number of particle used vs RMSE and Runtime for all methods in case of N=100

	Mean number of particles used				RMSE				Runtime (ms)			
Q	KLD	KLDE	KNN-KLD	LDA-KLD	KLD	KLDE	KNN-KLD	LDA-KLD	KLD	KLDE	KNN-KLD	LDA-KLD
0.1	16.07	6.26	5.58	5.43	11.72	18.20	5.57	2.96	3.67	0.75	0.68	0.63
0.3	10.8	3.26	3.36	3.04	41.59	31.12	31.03	21.46	2.19	0.75	0.80	0.34
0.5	8.6	2.55	2.43	2.61	24.38	20.60	16.12	7.86	1.17	0.39	0.57	0.39
0.7	6.53	2.34	2.33	2.36	28.11	24.05	14.51	31.00	1.08	0.47	0.31	0.62
0.9	4.61	2.21	2.26	2.2	42.18	19.71	15.89	16.96	1.69	0.78	0.52	0.63

This Table shows that the variance of Q is large (0.5, 0.7, 0.9) our proposal, KNN-KLD is better than the others. However, when it is small, the LDA-KLD overtook the other methods. Besides, the RMSE value of our proposal is more improved than others. Moreover, thanks to pre-processing in [13] these unnecessary missing data are removed leading to the computational time of our proposal is significant enhancement when compared to KLD in [6] and KLDE in [8].

To verify the changing of bound error that effects to mean number of particles used in more detail, the variance of Q=0.5 in case of N=100 is considered to evaluate as shown in Figure 6.It confirms that the mean number of particles used of KNN-KLD method (red color) decreases slightly lower than that of KLDE (blue color) in [8] and LDA-KLD (green color). Our second proposal, namely LDA-KLD, is hard to construct the new dimensional space because the distance between all classes is small and the rate of overlap number is large (see in Figure 4).

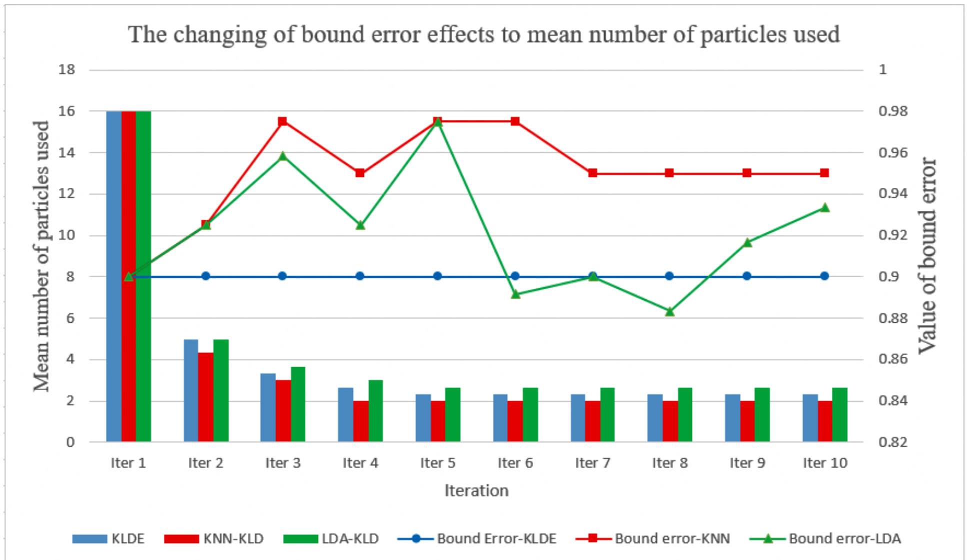


Figure 6. The predicted bound error vs. mean number of particles used (N =100,Q=0.5).

When the variance of noise Q declines between 0.1 and 0.3, our proposal, namely LDA-KLD, is better than the others. As Figure 7 in case of N=100 and Q=0.1, the mean number of particles used of our proposal (green color), namely LDA-KLD, falls down significant from the 4th iteration to the end about from 5 to 2. This is because the value of mean number of particles used in this case is too large (about 53 at the 1st iteration). Therefore, the LDA algorithm easily build the new dimensional spaceand as a result it is better than the other methods.

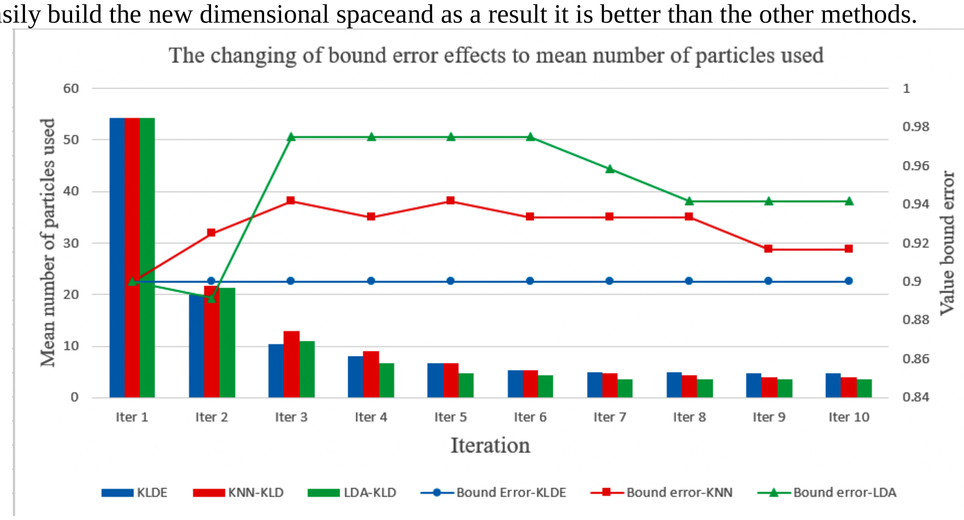


Figure 7. The changing of bound error vs. mean number of particles used (N=100, Q=0.1)

IV. CONCLUSION

This paper, we propose two new methods to find the value of bound error for online training data based KNN and LDA then applying KLD-resampling estimates the location of target in WSN to compare the performance of RMSE and runtime than others. Furthermore, our proposal also reduce the number of particles used via helped KLD-resampling when compared traditional methods. In the future, our group tends to apply deep learning in [16] to more improve our results.

V. ACKNOWLEDGMENT

This research is funded by Vietnam National University HoChiMinh City (VNU-HCM) under grant number T2017-04-IT.

VI. REFERENCE

[1] Schön, T.B., "Solving nonlinear state estimation problems using particle filters-an engineering perspective". Division of Automatic Control, Linköping University, Linköping, Sweden, 2010.

[2] Arulampalam, M.S., et al., "A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking". IEEE Transactions on signal processing, 2002. 50(2): p. 174-188.

[3] Gordon, N.J., D.J. Salmond, and A.F. Smith. "Novel approach to nonlinear/non-Gaussian Bayesian state estimation". in IEE Proceedings F-Radar and Signal Processing. 1993. IET.

[4] Li, T., T.P. Sattar, and S. Sun, "Deterministic resampling: Unbiased sampling to avoid sample impoverishment in particle filters". Signal Processing, 2012. 92(7): p. 1637-1645.

[5] Li, T., M. Bolic, and P. Djuric, "Resampling methods for particle filtering". IEEE Signal Processing Magazine.

[6] Li, T., S. Sun, and T.P. Sattar, "Adapting sample size in particle filters through KLD-resampling". Electronics Letters, 2013. 49(12): p. 740-742.

[7] Fox, D., "Adapting the sample size in particle filters through KLD-sampling". The international Journal of robotics research, 2003. 22(12): p. 985-1003.

[8] N.Ly-Tu, T. Le-Tien, and L. Mai, "Performance of sampling/resampling-based particle filters applied to non-linear problems". REV Journal on Electronics and Communications, 2016. 4(3-4).

[9] Park, S. H., Kim, Y. J., Lee, H. C., & Lim, M. T. "Improved adaptive particle filter using adjusted variance and gradient data". in Multisensor Fusion and Integration for Intelligent Systems, 2008. MFI 2008. IEEE International Conference on. 2008. IEEE.

[10] Zhao, Q., Wei, C., Qi, L., & Yuan, W."Adaptive Double-Resampling Particle Filter Algorithm for Target Tracking". in International Conference on Frontier Computing. 2016. Springer.

[11] Dihua, S., Hao, Q., Min, Z., Senlin, C., & Liangyi, Y."Adaptive KLD sampling based Monte Carlo localization". in 2018 Chinese Control And Decision Conference (CCDC). 2018. IEEE.

[12] Wang, Z., X. Zhao, and X. Qian. "The analysis of localization algorithm of unscented particle filter based on RSS for linear wireless sensor networks". in Proceedings of the 32nd Chinese Control Conference. 2013. IEEE.

[13] Swamynathan, M., "Mastering Machine Learning with Python in Six Steps: A Practical Implementation Guide to Predictive Data Analytics Using Python". 2017: Apress.

[14] N.Ly-Tu, T. Le-Tien, and L. Mai. “A New Resampling Parameter Algorithm for Kullback-Leibler Distance with Adjusted Variance and Gradient Data Based on Particle Filter”, in International Conference on Industrial Networks and Intelligent Systems. 2017. Springer.

[15] N. Ly-Tu, T. Le-Tien, Q. Vo-Phu, and T. Huynh-Kha, A New Bound Error based K-Nearest Neighbor for Kullback-Leibler Distance Particle Filter in Tracking, in Proceedings National Conference on @, Thai Binh province,28-29/6/2019, p.1-6, ISBN:978-604-67-1287-9.

[16] Gao, C., Yan, J., Zhou, S., Chen, B., & Liu, H. “Long short-term memory-based recurrent neural networks for nonlinear target tracking”. Signal Processing, 2019.

[17] https://www.saedsayad.com/k_nearest_neighbors.htm

[18] Tharwat, A., Gaber, T., Ibrahim, A., & Hassanien, A. E."Linear discriminant analysis: A detailed tutorial." AI communications 30.2 (2017): 169-190.