Distributed Gaussian Particle Filter for Target Tracking in Wireless Sensor Networks

Xuguang Yang¹, Yuechuan Zhang¹⁺, Xukang Wu¹, Lianhai Shan¹, Yunzhou Qiu¹, and Chunlei

Zheng¹

¹ Shanghai Internet of Things Co., LTD.

Abstract. In this paper, we present a distributed Gaussian particle filter based on Mahalanobis distance (DGPF-MD) for target tracking in wireless sensor networks. The proposed algorithm consists of three major steps. First, a sensor selection scheme is performed to reduce the cost of transmission among sensors with high accuracy. Second, a distributed Gaussian particle filter is adopted for each selected sensor to estimate the local statistics. Third, during weighted average fusion, the global estimate is based on the utility of the data provided by the member sensors, which is characterized as MD between the sensor and predicted target position. Compared with the centralized particle filters (CPFs), our experimental evaluations show that the DGPF-MD has more acceptable complexity, lower communication cost, and shorter tracking latency.

Keywords: Wireless sensor network, Gaussian particle filtering, distributed particle filter, target tracking, Mahalanobis distance

1. Introduction

Wireless sensor networks (WSNs) [1] contained a large number of scattered low-cost sensor nodes, which can provide rich and complex information. As a very important class of WSNs applications, target tracking [2] studies the dynamic state estimation by modeling the state space as a stochastic process evolving over time.

For dynamic state estimation such as target tracking, particle filter (PF) is one of the most widely used methods [3]. Distributed particle filters (DPFs) for collaborative tracking in WSNs have been received significant attention [4] recently. In [5], an average consensus filter is used to estimate global mean and covariance of posterior probability. Consensus-based distributed unscented particle filter (CD/UPF) in [6] computes global estimate from local estimates during the consensus step, which is based on an optimal fusion rule. In [7], the consensus algorithms are adopted to compute the global likelihood function rather than the posterior. Consensus-based DPFs are robust to changes in the network topology and to link failures. However, each node communicates with all its neighbors through multiple iterations to establish an agreement on certain global statistics. The speed of convergence (i.e. number of iterations) affects the communication requirements.

In this work, we propose a distributed PF that does not rely on consensus algorithm. To address the distributed collaborative target tracking in WSNs, a dynamic cluster around the target is formed, and a sensor selection scheme is also applied to select several informative sensors from the cluster. Each selected sensor performs local GPF and estimates local statistics in parallel. To achieve distributed information fusion, a weighted average fusion algorithm is employed to estimate global mean and covariance of posterior probability. The weights are based on the utility of the data provided by the member sensors, which is characterized as Mahalanobis distance (MD) between the sensor and predicted target position. In addition, an

⁺ Corresponding author. Tel.: +86 021 59523688-027.

E-mail address: zhangyuechuan@shiotc.com.

unscented transformation (UT) is used to propagate the mean and covariance to generate the predicted statistics, as UT incorporates partial high order information of mean and covariance into the estimates for non-linear models [8]. Our method can reduce complexity and communication requirements while preserve the tracking accuracy.

The rest of this paper is organized as follows. Section 2 presents the proposed distributed Gaussian particle filter based on MD (DGPF-MD). Section 3 analyzes the performance of DGPF-MD in target tracking application through simulations. Section 4 makes a conclusion of the paper.

2. Distributed Particle Filter for target tracking

For distributed collaborative target tracking in WSNs, it is necessary to form a dynamic cluster around the target and properly select the cluster head (CH) and the cluster members during the tracking process. The selection of cluster members is based on the information utility, which can be evaluated as MD. The selected member nodes receive the target state prediction, perform local PF, and transmit the local estimations to CH. The CH fuses the collected information and estimates global statistics.

2.1. Sensor selection

Assume that the global estimation and covariance are calculated at time k-1. The CH predicts the state and its covariance by the scaled unscented transformation (UT) [8]. Calculate sigma points $S_{k-1}^{(i)} = \{W_m^{(i)}, W_c^{(i)}, \chi_{k-1}^{(i)}\}$ according to the following scheme.

$$\begin{cases} \chi_{k-1}^{(0)} = \hat{x}_{k-1} \\ \chi_{k-1}^{(i)} = \hat{x}_{k-1} + \left(\sqrt{(n_x + \lambda)P_{k-1}}\right)^{(i)}, \ i = 1, \dots, n_x \\ \chi_{k-1}^{(i)} = \hat{x}_{k-1} - \left(\sqrt{(n_x + \lambda)P_{k-1}}\right)^{(i)}, \ i = n_x + 1, \dots, 2n_x \\ \begin{cases} W_m^{(0)} = \lambda/(n_x + \lambda) \\ W_c^{(0)} = \lambda(n_x + \lambda) + (1 - \alpha^2 + \beta) \\ W_m^{(i)} = W_c^{(i)} = \frac{0.5}{n_x + \lambda}, \ i = 1, 2, \dots, 2n_x \end{cases}$$
(1)

Where $\lambda = \alpha^2 (n_x + \kappa) - n$, α, β, κ are the scaling parameters. α controls the "size" of the sigma point distribution and should be a small number $(10^{-4} \le \alpha \le 1)$. β is a non-negative weighting term which minimizes the effects from high order terms. For Gaussian distributions, $\beta = 2$ is optimal. κ determines the approximating precision (usually set to 0). $(\sqrt{(n_x + \lambda)P_{k-1}})^{(i)}$ is the *i*th row or column of the matrix square root of $(n_x + \lambda)P_{k-1}$. In this step, the Cholesky decomposition can be chosen as one of the possible solutions for square rooting of a matrix. $W^{(i)}$ is the weight associated with the *i*th point.

The transformed sigma points are obtained as follows

$$\chi_{k|k-1}^{(i)} = f(\chi_{k-1}^{(i)}) \tag{3}$$

The sigma points of the noise \mathbf{w}_k are $\{0, \pm (\sqrt{(n+\lambda)Q_{k-1}})^{(i)}\}$. The predicted mean and covariance at time k are given by

$$\overline{\boldsymbol{x}}_{k|k-1} = \sum_{i=0}^{2n} W_m^{(i)} \chi_{k|k-1}^{(i)} \tag{4}$$

$$\boldsymbol{P}_{k|k-1} = \boldsymbol{Q}_{k-1} + \sum_{i=0}^{2n} W_c^{(i)} \left(\chi_{k|k-1}^{(i)} - \overline{\boldsymbol{x}}_{k|k-1} \right) \left(\chi_{k|k-1}^{(i)} - \overline{\boldsymbol{x}}_{k|k-1} \right)^T$$
(5)

Since the predicted target state $\overline{x}_{k|k-1}$ at time step k is available, it would be utilized to select sensors. According to the statistics theory, the probability density function (PDF) of the target located at coordinate X = (x, y) will be

$$p(x,y) = \frac{1}{2\pi} (\det(\mathbf{P}_{k|k-1}))^{-1/2} \exp\left(-\frac{1}{2} \left(X - \overline{x}_{k|k-1}\right) \mathbf{P}_{k|k-1}^{-1} \left(X - \overline{x}_{k|k-1}\right)^T\right)$$
(6)

Assume that the location of sensors are known, and coordinate of the *n*th sensor $(1 \le n \le N_s)$ is $S_n = (sx_n, sy_n)$. Thus the probability that the *n*th sensor can detect the target is

$$p(S_n) = \frac{1}{2\pi} \left(\det(\mathbf{P}_{k|k-1}) \right)^{-\frac{1}{2}} \exp\left(-\frac{1}{2} (S_n - \overline{\mathbf{x}}_{k|k-1}) \mathbf{P}_{k|k-1}^{-1} (S_n - \overline{\mathbf{x}}_{k|k-1})^T \right)$$
(7)

Take logarithms on both sides of (4) to obtain

$$\log(\mathbf{p}(\mathbf{S}_n)) = C - \frac{1}{2} \left(S_n - \overline{\mathbf{x}}_{k|k-1} \right) \boldsymbol{P}_{k|k-1}^{-1} \left(S_n - \overline{\mathbf{x}}_{k|k-1} \right)^T$$
(8)

Wherein $C = -\log(2\pi) - \frac{1}{2}\log(\det(\mathbf{P}_{k|k-1}))$. C is a constant, dependent on the predicted covariance at time k. The second part is a form of MD, reflecting the distance between the sensor location and the predicted target position.

$$MD_n = \sqrt{(S_n - \overline{x}_{k|k-1}) \mathbf{P}_{k|k-1}^{-1} (S_n - \overline{x}_{k|k-1})^T}$$
(9)

A small number of sensors providing the most informative measurements should be chosen as the cluster members at each time step. Since the larger MD_n , the less utility of the corresponding sensor. We define the utility function as

 $\varphi_{utility}(S_n) = 1/MD_n \tag{10}$

According to the utility of sensors, the CH chooses M members. M can be changed according to different requirement of tracking accuracy.

2.2. Local GPF (LGPF)

In distributed estimation, the local particles and their weights are only based on the local observations $z_{n,k}$, which results in inconsistent state estimates $p(x_k | z_{n,k})$ across the network. A global fusion (next subsection) is introduced to provide consistency in the local estimates. We use Gaussian approximations in the context of local UPF as the proposal distribution in which each sensor draws particles according to predicted state $\overline{x}_{k|k-1}$ and covariance $\overline{P}_{k|k-1}$ in (4) and (5).

(i) At time step k, the selected nodes are assumed to have received the global prediction of the target state $\overline{x}_{k|k-1}$ and $\overline{P}_{k|k-1}$. A new measurement $z_{n,k}$ is now available at the local nodes.

(ii) Sensor *n* generates N random particles from its proposal distribution $N(\mathbf{x}_{k}^{i}; \overline{\mathbf{x}}_{k|k-1}, \overline{P}_{k|k-1})$ and computes their correspondent weights $w_{n,k}^{i}$ based on Eq. (13). After this step, node *n* has a set of particles $\{\mathbf{x}_{n,k}^{i}, w_{n,k}^{i}\}_{i=1}^{N}$ that approximate the local posterior distribution $p(\mathbf{x}_{k}|\mathbf{z}_{n,k})$.

(iii) Sensor *n* computes the MMSE estimation:

$$\overline{\boldsymbol{x}}_{n,k} = \sum_{i=1}^{N} w_{n,k}^{i} x_{n,k}^{i}$$

$$\overline{\boldsymbol{P}}_{n,k} = \sum_{i=1}^{N} w_{n,k}^{i} (x_{n,k} - x_{n,k}^{i}) (x_{n,k} - x_{n,k}^{i})^{T}$$
(2)

(iv) The final step of the DPF algorithm is the fusion step. A global statistics $\bar{\mathbf{x}}_k$ and $\bar{\mathbf{P}}_k$ is computed from the local estimates, which is depicted in next subsection.

2.3. Estimation of Global Statistics via weighted average

When receiving the local estimates from the member sensors, CH calculates the global mean and covariance of the entire cluster by fusing the local means and covariance respectively.

$$\widehat{\boldsymbol{x}}_{k} = \sum_{n=1}^{N_{m}} w_{k}^{n} \overline{\boldsymbol{x}}_{n,k}$$

$$\boldsymbol{P}_{k} = \sum_{n=1}^{N_{m}} w_{k}^{n} (\overline{\boldsymbol{x}}_{n,k} - \widehat{\boldsymbol{x}}_{k}) (\overline{\boldsymbol{x}}_{n,k} - \widehat{\boldsymbol{x}}_{k})^{T}$$

$$\tag{4}$$

Where, the weights are set up based on utility function. The weights are suggested to be inversely proportional to the MD

$$w_k^n = \frac{\varphi_{utility}(\mathbf{S}_n)}{\sum_{n=1}^{N_m} \varphi_{utility}(\mathbf{S}_n)} = \frac{M D_n^{-1}}{\sum_{n=1}^{N_m} M D_n^{-1}}$$
(5)

Where $\{w_k^n\}$ are adaptive weights, and satisfy normalization metric $\sum_{n=1}^{N_c} w_k^i = 1$.

2.4. Distributed Gaussian Particle Filter based on MD

The algorithm is stated in the following.

In our proposed DGPF-MD scheme, the heavy task of CH is offloaded to small amount of sensors. Each selected sensor uses a LGPF to track the local Gaussian approximation $N(x_{n,k}; \bar{x}_{n,k}, \bar{P}_{n,k})$ of the posterior $p(x_k | z_{n,k})$. The estimate is based on the current local measurements and previous measurement reliability. The estimated statistics is transmitted between CH and its members.

Algorithm I Distributed Gaussian Particle Filter based on MD (DGPF-MD)

- 1. Initialize random samples $\{x_0^i\}_{i=1}^N$ drawn from $p(x_0) \sim N(x_0; \mu_0, \Sigma_0)$, and $w_0^i = 1/N$. 2. At time k-1, assumed that the global estimation \hat{x}_k and P_k are obtained. CH generates the sigma points $\left\{S_{k-1}^{(i)}\right\}_{i=0}^{2n_{\chi}}$ and computes the statistics of prediction distribution $\overline{x}_{k|k-1}$ and $P_{k|k-1}$ according to Eq. (1)-(2).
- 3. Selected sensor *n* draw particles $\{x_{n,k}^i\}_{i=1}^N$ from its local importance distribution $q(x_{n,k}|z_{0:k}) \sim N(x_{n,k}; \overline{x}_{k|k-1}, P_{k|k-1})$ and update their correspondent weights according to Eq.(13). Normalize the weights as $w_{n,k}^i = w_{n,k}^i / \sum_{i=1}^N w_{n,k}^i$. 4. Sensor *n* estimate the local statistics $\overline{x}_{n,k}$ and $\overline{P}_{n,k}$ according to Eq.(11)-(12).
- 5. CH receive the local estimations and calculates the global statistics \overline{x}_k and \overline{P}_k according to Eq.(13)-(14), and $w_k^n = MD_n^{-1} / (\sum_{n=1}^{N_m} MD_n^{-1}).$

Compared with the CGPF, DGPF-MD decreases the number of particles apparently. Moreover, it lowers communication costs and reduces tracking latency in contrast to other DPFs, which will be detailed in the next section.

3. Simulations

3.1. Simulation environment

A distributed target tracking scenario in WSNs is simulated to evaluate the proposed DPF algorithm. The monitoring system consists of 200 sensors randomly scattered over a field of $800 \times 600m^2$. The sensors are assumed synchronized, and have the prior knowledge of their positions and the motion model. The sensors can detect the target within the sensing range $R_s = 80$ m and communicate with each other within communication radius $R_c = 160$ m.

We described a nearly constant velocity (CV) model [9] in the target state dynamics and an energy-based acoustic measurement model [10, 11], in the following form:

$$\begin{aligned} \boldsymbol{x}_{k} &= F_{k} \boldsymbol{x}_{k-1} + \Gamma \boldsymbol{w}_{k} \\ \boldsymbol{z}_{k} &= g \cdot \frac{s}{||\boldsymbol{r}_{t} - \boldsymbol{r}||^{\alpha}} + \boldsymbol{v}_{k} \end{aligned} \tag{6}$$

Where $\mathbf{x}_{k} = \begin{bmatrix} \xi_{k} & \dot{\xi}_{k} & \eta_{k} & \dot{\eta}_{k} \end{bmatrix}^{T}$ is the state vector, containing the target's location coordinates (ξ_{k}, η_{k}) and velocities $(\dot{\xi}_k, \dot{\eta}_k)$ along x-axis and y-axis. \mathbf{z}_k is the observed acoustic intensity. The matrices F_k and Γ are defined as follows:

$$F_{k} = \begin{bmatrix} 1 & T & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & T \\ 0 & 0 & 0 & 1 \end{bmatrix} , \qquad \Gamma = \begin{bmatrix} T^{2}/2 & 0 \\ T & 0 \\ 0 & T^{2}/2 \\ 0 & T \end{bmatrix}$$
(8)

In addition, *T* is the sample period. The variance of \boldsymbol{w}_k and \boldsymbol{v}_k are $\boldsymbol{Q}_k = \begin{bmatrix} \sigma_x^2 & 0\\ 0 & \sigma_y^2 \end{bmatrix}$ and σ_n^2 , respectively.

s is the acoustic energy emitted by the target. Since r_t and r respectively denote the coordinates of the target and the sensor at time k, $||r_t - r||$ is the Euclidean distance between them. g is the gain factor of the sensor, which can be measured during sensor calibration. α is the energy decay factor. In [22], it was shown that the effective decay α , is approximately 2 and this value is assumed for this paper.

The detailed parameters configurations are as follows. The variances of noises are $\sigma_x^2 = 0.2^2$, $\sigma_x^2 = 0.3^2$, $\sigma_n^2 = 0.005$. Also, s = 1000, $g_n = 1$. The simulation includes 50 time steps, and the time interval is T = 2. Target starts its track at time k=0 with initial state $\begin{bmatrix} 0 & 5 & 20 & 4 \end{bmatrix}^T$.

To quantitatively evaluate the performance of the proposed algorithm, we define the following root mean squared error (RMSE) of the estimated target position and velocity

$$\operatorname{RMSE}_{k} = \begin{cases} \sqrt{\frac{1}{N_{mc}} \sum_{j=1}^{N_{mc}} \left(\left(\hat{\xi}_{k}^{j} - \xi_{k}^{true} \right)^{2} + \left(\hat{\eta}_{k}^{j} - \eta_{k}^{true} \right)^{2} \right)} & , \text{ for position} \\ \sqrt{\frac{1}{N_{mc}} \sum_{j=1}^{N_{mc}} \left(\left(\hat{\xi}_{k}^{j} - \dot{\xi}_{k}^{true} \right)^{2} + \left(\hat{\eta}_{k}^{j} - \eta_{k}^{true} \right)^{2} \right)} & , \text{ for velocity} \end{cases}$$
(9)

3.2. Experimental Results

We compare the DGPF-MD with CGPF and the distributed GPF (DGPF) in [7]. It is assumed that the number of sensors in the active cluster is N_active_k and the CH selects M (M=4) sensors to perform LGPF at k. The number of particles at the fusion center of the CGPF is N=200, whereas the DGPF-MD and DGPF schemes respectively employ N/M and N/N_active_k particles. It is noted that $N_active_k > M$.

Figure 1 shows a practical trajectory and the tracking results obtained by three different filtering algorithms. Figure 2 demonstrates the temporal evolution of RMSE for k = 1, ..., 50.





Fig. 1: Tracking trajectory using three algorithms

Fig. 2: The Position RMSE and Velocity RMSE

According to both figures, the performance of DGPF-MD is almost as good as that of CGPF and DGPF. CGPF has a best tracking performance because of using all the candidate sensor data, but this high accuracy comes from a complete set of sensor data and the heavy computation burden in CH. DGPF outperforms slightly better than DGPF-MD. In the DGPF, the local information is diffused throughout the whole network, and the resulting approximate nearly contains the global information. This diffusion of the local information cost much time and energy. The performance degradation of DGPF-MD relative to CGPF and DGPF is caused by introducing minimum sensors and the strategy of LGPF based on the local measurement. Nevertheless, the performance degradation is acceptable, since DGPF-MD activates much fewer sensors and disperses the computation task to several activated sensors.

The dependence of the accuracy on the number of selected sensors M has also been studied. We compute the average RMSE (ARMSE) by averaging RMSE_k^2 over all simulated time steps. Table 1 presents the comparison of ARMSE of position and velocity, respectively. The results are obtained by averaging 10 independent experiments over the same network topology. The ARMSE of position decreases as M increases. The tracking accuracy of DGPF-MD (M=5) is almost identical to the DGPF. What's more, if M=6 or M=7 is used, the DGPF-MD achieves higher accuracy than DGPF and approaches the CGPF.

		-			•	
Comparison	CGPF	DGPF	DGPF-MD	DGPF-MD	DGPF-MD	DGPF-MD
			(<i>M</i> =4)	(<i>M</i> =5)	(<i>M</i> =6)	(<i>M</i> =7)
ARMSE of	3.298	4.231	4.634	4.304	3.956	3.716
position						
ARMSE of	0.801	0.896	0.872	0.885	0.824	0.837
velocity						

Table I: Comparison of ARMSE of Three Different Algorithms

3.3. Complexity and communication requirements

It is assumed that the number of cluster sensors is *n* at time *k*. The complexity of the GPF algorithm depends linearly on the number of particles. Thereby, the CGPF has a highest complexity. The DGPF cuts down the complexity by a factor of n as the number of particles at each sensor reduces from N to N/n. Similarly, the complexity of DGPF-MD is reduced by a factor of M (M < n) relative to CGPF.

The communication cost of the CGPF algorithm increases with the number of sensors since the fusion center requires all the candidate measurements. In DGPF, we assume the best case that the cluster members can communication with each other for $Rc \ge 2Rs$ and the global consensus statistics can be established through one iteration. The communication cost of DGPF is the highest, since the additional consensus algorithm calls for much more information exchange among sensors. The DGPF-MD reduces its communication load far below the level of the DGPF. It is because that the small amount but informative sensors are employed to perform LGPF without local measurements exchange. Then, the communication costs for these algorithms are listed in Table 2.

	Measurements		Estimation statistics	Total1
CGPF	n		/	п
	Per sensor	2(<i>n</i> -1)	2(<i>n</i> -1)(4+16)	
DGPF	Total	2 <i>n</i> (<i>n</i> -1)	2 <i>n</i> (<i>n</i> -1)(4+16)	42n(n-1)
DGPF-MD	/		2 <i>M</i> (4+16)	22 <i>M</i>

Table II: Comparison of Communications of CGPF, DGPF and DGPF-MD

Furthermore, tracking latency mainly results from data transmission and processing. In CGPF, convergecast communication brings about a long delay, which increases linearly with n, as the center unit has to receive messages in a sequential order. In the DGPF, an additional delay is caused by the consensus algorithms. Compared with the CGPF and DGPF, the latency of the DGPF-MD is reduced since transmission time is determined by M (M < n). As n increases, the advantage of DGPF-MD will become more apparent. Thus, the DGPF-MD may be more suitable for real-time applications

4. Conclusion

This paper proposes a MD-based, distributed Gaussian particle filter (DGPF-MD) for target tracking in WSNs, which employs a sensor selection scheme to cluster several informative sensors, while uses a weighted average fusion algorithm to estimate global mean and covariance of posterior probability. Also, UT is applied to propagate the mean and covariance to generate the predicted statistics. At each selected sensor, a LGPF is performed in parallel, and only local communications between the selected sensors and CH are demanded. The global statistics is estimated by weighted average fusion based on utility of the data provided by the member sensors, which is characterized as MD between the sensor and predicted target position.

Finally, the simulations evaluate the performance of the proposed DGPF-MD, CGPF and DGPF, which shows the DGPF-MD has more acceptable complexity, lower communications cost and shorter tracking latency. So we can see the DGPF-MD is more suitable for real-time applications.

5. Acknowledgements

The author would like to express his heartfelt gratitude to Dr. Lin Shang for her great help and this paper is also sponsored by Natural Science Foundation of Shanghai, with sponsor ID 15ZR1439800.

6. References

- Yick, J., Mukherjee, B. and Ghosal, D.: 'Wireless sensor network survey', Comput. Networks, 2008, 52, (12), pp. 2292-2330
- [2] Zhao, F. and Guibas, L. J.: 'Wireless sensor networks: an information processing approach', in (Ed.)^(Eds.):
 'Book Wireless sensor networks: an information processing approach' (Morgan Kaufmann, 2004, edn.), pp.
- [3] Jiang, B. and Ravindran, B.: 'Completely Distributed Particle Filters for Target Tracking in Sensor Networks', 2011, pp. 334-344

- [4] Hlinka, O., Hlawatsch, F. and Djuric, P. M.: 'Distributed particle filtering in agent networks: A survey, classification, and comparison', IEEE SIGNAL. PROC. MAG., 2013, 30, (1), pp. 61-81
- [5] Gu, D. B., Sun, J. X., Hu, Z. and Li, H. Z.: 'Consensus Based Distributed Particle Filter in Sensor Networks', 2008 International Conference on Information and Automation, Vols 1-4, 2008, pp. 302-307
- [6] Mohammadi, A. and Asif, A.: 'Consensus-Based Distributed Unscented Particle Filter', 2011 IEEE Statistical Signal Processing Workshop (Ssp), 2011, pp. 237-240
- [7] Hlinka, O., Sluciak, O., Hlawatsch, F., Djuric, P. M. and Rupp, M.: 'Distributed Gaussian Particle Filtering Using Likelihood Consensus', Int Conf Acoust Spee, 2011, pp. 3756-3759
- [8] Julier, S. J. and Uhlmann, J. K.: 'Unscented filtering and nonlinear estimation', Proceedings of the IEEE, 2004, 92, (3), pp. 401-422
- [9] Li, X. R. and Jilkov, V. P.: 'Survey of maneuvering target tracking. Part I. Dynamic models', Aerospace and Electronic Systems, IEEE Transactions on, 2003, 39, (4), pp. 1333-1364
- [10] Sheng, X. H. and Hu, Y. H.: 'Maximum likelihood multiple-source localization using acoustic energy measurements with wireless sensor networks', IEEE Transactions on Signal Processing, 2005, 53, (1), pp. 44-53
- [11] Meesookho, C., Mitra, U. and Narayanan, S.: 'On energy-based acoustic source localization for sensor networks', IEEE Transactions on Signal Processing, 2008, 56, (1), pp. 365-377