A Kalman-Particle Filter for Estimating the Number and State of Multiple Targets

W. Lu[§], H. Wei[§], and S. Ferrari[§]

[§]Laboratory for Intelligent Systems and Control (LISC), Department of Mechanical Engineering and Materials Science, Duke University, Durham, NC, USA;

ABSTRACT

The problem of estimating the number and state of multiple targets using a sensor with limited sensing ability is raised in a variety of applications, including monitoring of endangered species, civilian security, and military surveillance. The particle filter is widely used to solve this problem since Kalman filter's disadvantage on estimating non-Gaussian distribution. However, the problem becomes intractable when the number of total targets are unknown and one measurement is associated serval targets. This paper presents a novel filter technique which combines Kalman filter and particle filter for estimating the number and state of total targets based on the measurement obtained online. The estimation is represented by a set of weighted particles, different from classical particle filter, where each particle is a gaussian instead of a point mass in the system state.

Keywords: Kalman filter, particle filter, kalman-particle filter, multiple target tracking

1. INTRODUCTION

The problem of tracking and monitoring targets using one position-fixed sensor is relevant to a variety of applications, including monitoring of urban environments,[1] tracking anomalies in manufacturing plants,[2] and tracking of endangered species.[3] The position-fixed sensor is deployed to measure targets based on limited information that only becomes available when the target enters the sensor's field-of-view (FOV) or visibility region. The sensor's FOV is defined as a compact subset of the region of interest, in which the sensor can obtain measurements from the targets.

Then, under proper assumptions that include additive random noise with a Gaussian distribution, the target state can be estimated from frequent observations of its measurable output, using a Kalman filter.[4] This approach is well suited to long-range high-accuracy sensors, such as radars, and to moving targets with a known dynamical model and initial conditions. However, most of these underlying assumptions are violated in modern applications of sensors, because the targets' motion models are unknown, and, possible, random and nonlinear. Also, due to the use of low-cost passive sensors, measurement errors and noise may be non-additive and non-Gaussian. An extended Kalman filter (EKF) can be used when the system dynamics are nonlinear, but can be linearized about nominal operating conditions.[5] An unscented Kalman filter (UKF) method, based on the unscented transformation (UT) method, can be applied to compute the mean and covariance of a function up to the second order of the Taylor expansion.[6,7] However, the efficiency of these filters decreases significantly when the system dynamics are highly nonlinear, and when the random effects are non-Gaussian. Recently, a non-parametric method based on condensation and Monte Carlo simulation, known as a particle filter, has been proposed for tracking multiple targets exhibiting nonlinear dynamics and non-Gaussian random effects.[8] Particle filters are well suited to modern surveillance systems because they can be applied to Bayesian models in which the hidden variables are connected by a Markov chain in discrete time, but the target state is continuous, as in Markov motion models.

In the classical particle filter method, a weighted set of particles or point masses are used to represent the probability density function (PDF) of the target state by means of a superposition of weighted Dirac delta functions.[9] At each iteration of the particle filter, particles representing possible target state values are sampled

Further author information: (Send correspondence to W.L.)
W.L., H.W., and S.F.: E-mail: {wenjie.lu, hongchuan.wei, sferrari}@duke.edu, Telephone: 1 919 660 5305

from an importance density function.[10] The weight associated with each particle is then obtained from the target-state likelihood function, and from the prior estimation of the target state PDF. When the effective particle size is smaller than a predefined threshold, a re-sampling technique can be implemented.[11] One disadvantage of classical particle-filtering techniques is that the target-state transition function is used as the importance density function to sample particles, without taking new observations into account.[12] As a result, when the target state transition function is much broader than the likelihood function, few sampled particles have proper locations and weights. An improved particle filter, the unscented particle filter (UPF) has been proposed in,[12] to overcome this difficulty, by combining UKF and the particle-filtering technique. The UKF generates a proposed distribution in which the current measurements are considered, and then the distribution is used as the importance density to sample particles. Another disadvantage of existing particle filters is that the point-mass representation provides limited information about the estimated PDF of the target state, and does not account for the targets' dynamic equations. Another particle filter was proposed recently,[13] where the particles are sampled based on the supporting intervals of the target-state likelihood function and the prior estimation function of the target state. In this case, the supporting interval of a distribution is defined as the 90% confidence interval.[14] The weight for each particle is obtained by considering the likelihood function and the transition function simultaneously. Then, the weighted expectation maximization (EM) algorithm is implemented to use the sampled weighted particles to generate a normal mixture model of the distribution.

Kreucher proposed joint multitarget probability density (JMPD)[9] to estimate the number of total targets in AOI and their state, where targets are moving. By using JMPD, the data association problem is avoided, however, the JMPD results in a joint system state space, the dimension of which is the dimension of a target state times number of total targets. Since the number of total targets is unknown, the joint space size remain unavailable. To overcome this problem, it is assumed the number of total targets has a maximum value. Therefore, when the maximum number of targets is large, the joint state space becomes intractable.

This paper presents a novel filter technique which combines Kalman filter and particle filter for estimating the number and state of total targets based on the measurement obtained online. The estimation is represented by a set of weighted particles, different from classical particle filter, where each particle is a gaussian instead of a point mass. The weight of each particle represents the probability of existing a target, while its gaussian indicates the state distribution for this target. More importantly, the update of particles is different from classical particle filter. For each particle, the gaussian parameters are updated based using Kalman filter given a measurement. To overcome the data association problem, in this paper, when one particle is updated, the other particles are considered as the measurement condition, which will be explained in Section 4. The novel kalman-particle filter technique requires less particles than classical particle filters, and can solve multiple target estimation problem without increasing the state space dimensions.

The paper is organized as follows. Section 2 describes the multiple targets estimation problem formulation and assumptions. The background on the particle filter and Kalman filter is reviewed in Section 3. Section 4 presents the Kalman-Particle filter technique. The method is demonstrated through numerical simulations and results, presented in Section 5. Conclusions and future work are described in Section 6.

2. PROBLEM FORMULATION

A number of targets, stationary or moving, are populated in the two dimensional workspace. N denotes the number of total targets. The whole workspace is visible to a position fixed sensor (not shown). The goal of the sensor is to obtain the state estimation for all the targets, denoted as $\mathbf{X}_k$, and number estimation of total targets, denoted as T_k, at time step k. The target The states for total targets at k, $[\mathbf{x}_k^1, \mathbf{x}_k^2, \cdots, \mathbf{x}_k^N]$, is denoted as $\mathbf{X}_k$, which has N state vectors. The estimation of total target state at k, $[\mathbf{x}_k^1, \mathbf{x}_k^2, \cdots, \mathbf{x}_k^{T^k}]$, is denoted as $\mathbf{X}^k$. T^k is the estimation of total target number at t^k. The ith target is modeled as

$$\mathbf{x}_k^i = \mathbf{F}_k \mathbf{x}_{k-1}^i + \boldsymbol{v}_k \tag{1}$$

where,

$$\boldsymbol{v}_k \approx N(0, \mathbf{Q}_k) \tag{2}$$

$\mathbf{F}_k$ and $\mathbf{Q}_k$ are assumed known.

In standard estimation theory, a sensor that obtains a vector of measurements $\mathbf{z}^k \in \mathbb{R}^r$ in order to estimate an unknown state vector set $\mathbf{X}^k \in \mathbb{R}^n$ at time k is modeled as,

$$\mathbf{z}^k = \mathbf{h}(\mathbf{X}^k, \boldsymbol{\lambda}^k) \tag{3}$$

where $\mathbf{h} : \mathbb{R}^{n+\wp} \to \mathbb{R}^r$ is a deterministic vector function that is possibly nonlinear, the random vector $\boldsymbol{\lambda}^k \in \mathbb{R}^{\wp}$ represents the sensor characteristics, such as sensor action, sensor mode, environmental conditions, and sensor noise or measurement errors.

In this paper, the sensor is modeled as

$$\mathbf{z}_k = \mathbf{H}_k \mathbf{x}_i + \boldsymbol{\nu}_k \tag{4}$$

$$\mathbf{z}_k = \frac{1}{N} \sum_{i=1}^{N} \mathbf{x}_i + \boldsymbol{\nu}_k \tag{5}$$

where N is the target number, and for simplicity $\mathbf{H}_k$ is set as $\frac{1}{N}\mathbf{I}$. The white noise term is defined as

$$\boldsymbol{\nu}_k \sim (N)(\mathbf{0}, \mathbf{R}_k) \tag{6}$$

Since the whole workspace is visible to the position fixed sensor, all targets are measured at the same time. However, this assumption can be loosened by the definition of sensor FOV, where only the targets in the FOV can be measured.

3. BACKGROUND

3.1 Kalman Filter Methods

Kalman filter[4] is a recursive method to have a statistically estimation of s system state based on a measurement sequence, minimizing the estimation uncertainty. The sensor provides the measurements of the system state, namely, position and velocity, with an additive white noise. Generally, in each iteration with the new measurement, the Kalman filter has two steps: a) first, it predicts the system state and their uncertainties; b) then it updates the system state and uncertainties with the latest measurement. The Kalman filter has following assumptions on system dynamics and sensor model. The system dynamics is defined as

$$\mathbf{x}_k = \mathbf{F}_k \mathbf{x}_{k-1} + \mathbf{B}_k \mathbf{u}_k + \boldsymbol{v}_k \tag{7}$$

where subscript k and $k-1$ denote the current and previous time index, while $\mathbf{F}_k$ is the system discrete transition matrix, and $\mathbf{B}_k$ and $\mathbf{u}_k$ are the control matrix and control input. $\boldsymbol{\nu}_k$ is the white noise, defined as

$$\boldsymbol{v}_k \sim N(\mathbf{0}, \mathbf{Q}_k) \tag{8}$$

where $\mathbf{Q}_k$ is the covariance. At kth time step, a measurement of the system true state $\mathbf{x}_k$ is made by a sensor, is given by

$$\mathbf{z}_k = \mathbf{H}_k \mathbf{x}_k + \boldsymbol{\nu}_k \tag{9}$$

where $\mathbf{H}_k$ is a mapping from system state space to measurement space, and the white noise $\mathbf{Q}_k$ is defined as

$$\boldsymbol{\nu}_k \approx N(0, R_k) \tag{10}$$

It is assumed that the noise $\boldsymbol{\nu}_k$ and $\boldsymbol{v}$ at each time step are independent.

To further introduce the classical Kalman filter method, let $\mathbf{x}_k$ denote the true state value, and let $\tilde{\mathbf{x}}_k$ denote the predicted state estimation given $\hat{\mathbf{x}}_{k-1}$ without $\mathbf{z}_k$, where $\hat{\mathbf{x}}_{k-1}$ is the updated estimation of system state at $k-1$ time step with $\mathbf{z}_{k-1}$ and $\hat{\mathbf{x}}_{k-2}$. Furthermore, let $\tilde{\boldsymbol{\Omega}}_k$ denote the predicted covariance given $\hat{\boldsymbol{\Omega}}_{k-1}$, where $\hat{\boldsymbol{\Omega}}_{k-1}$ is the updated estimation covariance at $k-1$. First, in the predicting step,

$$\tilde{\mathbf{x}}_k = \mathbf{F}_k \hat{\mathbf{x}}_{k-1} + \mathbf{B}_k \mathbf{u}_k \tag{11}$$

$$\tilde{\boldsymbol{\Omega}}_k = \mathbf{F}_k \hat{\boldsymbol{\Omega}}_{k-1} \mathbf{F}_k^T + \mathbf{Q}_k \tag{12}$$

In the updating step, the measurement $\mathbf{z}_k$ is used, together with above predicted state and covariance, to update the state and covariance. The residual, $\mathbf{y}_k$ between measurement and predicted state is given by

$$\mathbf{y}_k = \mathbf{z}_k - \mathbf{H}_k\tilde{\mathbf{x}}_k \tag{13}$$

The innovation covariance $\mathbf{S}_k$ is given by

$$\mathbf{S}_k = \mathbf{H}_k\tilde{\mathbf{\Omega}}_k\mathbf{H}_k^T + \mathbf{R}_k \tag{14}$$

Then, the optimal Kalman gain is calculated as

$$\mathbf{K}_k = \tilde{\mathbf{\Omega}}_k\mathbf{H}_k^T\mathbf{S}_k^{-1} \tag{15}$$

Therefor, the state and covariance can be updated by

$$\hat{\mathbf{x}}_k = \tilde{\mathbf{x}}_k + \mathbf{K}_k\mathbf{y}_k \tag{16}$$

$$\hat{\mathbf{\Omega}}_k = (\mathbf{I} - \mathbf{K}_k\mathbf{H}_k)\tilde{\mathbf{\Omega}}_k \tag{17}$$

3.2 Particle Filter Methods

The particle filter technique is a recursive model estimation method based on sequential Monte Carlo Simulations. It is applicable to nonlinear system dynamics with non-Gaussian random inputs. Moreover, because of their recursive nature, particle filters are easily applicable to online data processing and variable estimation. The main idea of particle filters is to represent the PDF functions with properly weighted and relocated point-mass, known as particles. These particles are sampled from an importance density which is crucial to the particle filter algorithm. Let $\{\mathbf{x}_{j,p}^{\kappa}, w_{j,p}^{\kappa}\}_{p=1}^{N}$ denote the weighted particles that are used to approximate the posterior PDF $f(\mathbf{x}_j^{\kappa} \mid Z_j^{\kappa})$ for the jth target at t_κ, where $Z_j^{\kappa} = \{\mathbf{z}_j^0, \ldots, \mathbf{z}_j^{\kappa}\}$ denotes the set of all measurements obtained by sensor i, from target j, up to t_κ. Then, the posterior probability density function of the target state, given the measurement at t_κ can be modeled as,

$$f(\mathbf{x}_j^{\kappa} \mid Z_j^{\kappa}) = \sum_{p=1}^{N} w_{j,p}^{\kappa}\delta(\mathbf{x}_{j,p}^{\kappa}), \quad \sum_{p=1}^{N} w_{j,p}^{\kappa} = 1 \tag{18}$$

where $w_{j,p}^{\kappa}$ is non-negative and δ is the Dirac delta function.[8] Although different particle filter techniques have been proposed,[10] the techniques always consist of the recursive propagation of the particles and the particle weights. In each iteration, the particles $\mathbf{x}_{j,p}^{\kappa}$ are sampled from the importance density $q(\mathbf{x})$. Then, weight $w_{j,p}^{k}$ is updated for each particle by

$$w_{j,p}^{\kappa} \propto \frac{p(\mathbf{x}_{j,p}^{\kappa})}{q(\mathbf{x}_{j,p}^{\kappa})} \tag{19}$$

where $p(\mathbf{x}_{j,p}^{\kappa}) \propto f(\mathbf{x}_{j,p}^{\kappa} \mid Z_j^{\kappa})$. Additionally, the weights are normalized at the end of each iteration.

Since the target state transition function is often used as the importance density function without considering the available new measurement, the sampled particles can not fully represent the target state estimation. Another common drawback of particle filters is the degeneracy phenomenon,[12] i.e., the variance of particle weights accumulates along iterations. This phenomenon indicates that a number of particles have low weights and no contributions in approximating the probability density function $f(\mathbf{x}_j^{\kappa} \mid \mathbf{z}_j^{\kappa})$ but put heavy computational burden to the algorithm. The number of particles with high weights is not sufficient to provide a good approximation. A common way to evaluate the degeneracy phenomenon is the effective sample size N_e,[11] obtained by,

$$N_e = \frac{1}{\sum_{p=1}^{N}(w_{j,p}^{\kappa})^2} \tag{20}$$

where $w_{j,p}^{\kappa},\ p = 1, 2, \ldots, N$ are the normalized weights. In general, a re-sampling procedure is taken when $N_e < N_s$, where N_s is a predefined threshold, and is usually set as $\frac{N}{2}$. Let $\{\mathbf{x}_{j,p}^{\kappa}, w_{j,p}^{\kappa}\}_{p=1}^{N}$ denote the particle set that needs to be re-sampled, and let $\{\mathbf{x}_{j,p}^{\kappa*}, w_{j,p}^{\kappa*}\}_{p=1}^{N}$ denote the particle set after re-sampling. The main idea of this re-sampling procedure is to eliminate the particles having low weights by re-sampling $\{\mathbf{x}_{j,p}^{\kappa*}, w_{j,p}^{\kappa*}\}_{p=1}^{N}$ from $\{\mathbf{x}_{j,p}^{\kappa}, w_{j,p}^{\kappa}\}_{p=1}^{N}$ with the probability of $p(\mathbf{x}_{j,p}^{\kappa*} = \mathbf{x}_{j,s}^{\kappa}) = w_{j,s}^{\kappa}$. At the end of the resampling procedure, $w_{j,p}^{\kappa*}, p = 1, 2, \ldots, N$ are set as $1/N$. However, the resampling procedure repeats the particles with high weights a number of times stochastically. This leads to diversity loss of particles.

In this paper, a novel Kalman-particle filter technique which combines Kalman filter and particle filter for estimating the number and state of total targets based on the measurement obtained online. The estimation is represented by a set of weighted particles, different from classical particle filter, where each particle is a gaussian instead of a point mass.

4. METHODOLOGY

In this paper, the system state consists of all the targets' state, and according to the sensor model (4), one measurement is associated with all the targets's state. Thus if only the Particle filter is used directly, the number of sampled particles is quite large due to the high dimensions of the system state. While if the Kalman filter only is applied, the measurement model does not satisfy the Kalman filter requirements. Therefor, these two filter techniques are combined to solve the high dimension problem. Different from classical Particle filter, the particles used to represent the estimation for number of total targets and their state is a set of weighted Gaussian. The ith particle at time k is denoted as

$$P_k^i = \{w_k^i, \mathcal{N}(\mathbf{x}_k^i | \boldsymbol{\mu}_k^i, \boldsymbol{\Omega}_k^i)\} \tag{21}$$

where w_k^i is the probability of existing a target having a state distribution as $\mathcal{N}(\mathbf{x}_k^i | \boldsymbol{\mu}_k^i, \boldsymbol{\Omega}_k^i)$. By this particle definition, the dimensions of the system state remain the same as the dimensions of each individual target $\mathbf{x}_k^i$, which is $[x, y, \dot{x}, \dot{y}]$ When these particles are calculated at k, the estimated number of total targets can be given as

$$T_k = \sum_{i=1}^{N_p} w_k^i \tag{22}$$

where N_p is the number of all particles and it is not necessary a constant. It is worth mentioning that $\sum_{i=1}^{N_p} w_k^i$ doesn't necessary equal 1.

Since the particle representation is different from classical particle filter, where each particle represents a possible value of system state, the updating of each particle and total weights are different. In the remainder of this paper, a modified Kalman filter is used to update each particle, the weight and the gaussian distribution. Different from Classical Kalman filter, where one measurement is associated with one target, the modified Kalman filter has to deal with the one measurement which is associated with all the targets in FOV at k.

The algorithm proposed in this paper is a recursive method. Without losing generality, it is assumed that at time k, the measurement $\boldsymbol{z}_k$ is available, and the estimation of the system, T_k and $\mathbf{X}_k$ at time step $k-1$, is represented by a particle set, denoted as $\mathcal{P}_{k-1} = \{P_{k-1}^1, P_{k-1}^2, \ldots, P_{k-1}^{N_p}\}$, where N_p is the number of all particles. By using the target dynamic function 1, $\mathcal{P}_{k-1}$ can be updated to $\tilde{\mathcal{P}}_k$ without using the $\boldsymbol{z}_k$. The parameters for each particle in $\tilde{\mathcal{P}}_k$ is also added by a $\tilde{}$. Due to limit of FOV, only a few particles may have contribution to the measurement. Let $\mathcal{P}_S$ denote set including the particles lie in the FOV, while let $\bar{\mathcal{P}} = \tilde{\mathcal{P}}_k / \mathcal{P}_S$ denote the complementary set. The particles in $\bar{\mathcal{P}}$ are only updated by applying (1), while the particles in $\mathcal{P}_S$ are updated by (1) and the modified Kalman filter using $\mathbf{z}_k$. Please Note the size of $\mathcal{P}_S$ is small, since the size of FOV is limited. Without loss of generality, it is assumed that $\mathcal{P}_S = \{\tilde{P}_k^1, \tilde{P}_k^2, \ldots, \tilde{P}_k^s\}$, where s is the number of particles.

According to the sensor model (4), $\mathbf{H}_k$ is a function based on number of total targets in the FOV, therefore, a combination of particles in $\mathcal{P}_S$ are needed to apply the (4) with the help of a boolean set $E_c = [e_1^c, e_1^c, \ldots, e_s^c]$, where $e_i^c \in \{0, 1\}$ and $c \in I_E$. I_E is the index set for possible combinations. Only the combination E_c, such that $\Pi(w_i)^{e_i^c}(1 - w_i)^{1-e_i^c} > \epsilon$, is considered, where ϵ is a predefined threshold. Then, the modified Kalman filter is used to give the updated gaussian parameters of all particles with $e_j^c = 1$. For updating jth particle with $e_j^c = 1$, the following procedures are executed. According to sensor model 4, the measurement is given by

$$\boldsymbol{z}_k = \frac{\sum_{i=1}^{s} \boldsymbol{\mu}_k^i e_i^c}{\sum_{i=1}^{s} e_i^c} + \frac{1}{(\sum_{i=1}^{s} e_i^c)^2} \sum_{i=1, i\neq j}^{s} e_i^c \boldsymbol{\Omega_i} + \boldsymbol{\nu}_k \tag{23}$$

Compare the above function to (9), we have following setting

$$\boldsymbol{H}_k = I \times \frac{1}{(\sum_{i=1}^{s} e_i^c)} \tag{24}$$

$$\boldsymbol{z}_k = \boldsymbol{z}_k - \frac{\sum_{i=1, i\neq j}^{s} \boldsymbol{\mu}_k^i e_i^c}{\sum_{i=1}^{s} e_i^c} \tag{25}$$

$$\boldsymbol{R}_k = \frac{1}{(\sum_{i=1}^{s} e_i^c)^2} \sum_{i=1, i\neq j}^{s} e_i^c \boldsymbol{\Omega_i} + \boldsymbol{\omega}_k \tag{26}$$

Then, by applying Kalman procedure

$$\mathbf{y}_k = \mathbf{z}_k - \mathbf{H}_k \tilde{\boldsymbol{\mu}}_k^j \tag{27}$$

$$\mathbf{S}_k = \mathbf{H}_k \tilde{\boldsymbol{\Omega}}_k^j \mathbf{H}_k^T + \mathbf{R}_k \tag{28}$$

$$\mathbf{K}_k = \tilde{\boldsymbol{\Omega}}_k^j \mathbf{H}_k^T \mathbf{S}_k^{-1} \tag{29}$$

$$\boldsymbol{\mu}_k^{j,c} = \tilde{\boldsymbol{\mu}}_k^j + \mathbf{K}_k \mathbf{y}_k \tag{30}$$

$$\boldsymbol{\Omega}_k^{j,c} = (\mathbf{I} - \mathbf{K}_k \mathbf{H}_k) \tilde{\boldsymbol{\Omega}}_k^j \tag{31}$$

The proof can be found in Appendix.

After all the particles appearing in combination E_c are updated, the weight w_c is for the particle combination E_c can be obtained by

$$w_c = \Pi_{i=1} s(w_i)^{e_i^c} (1 - w_i)^{1-e_i^c} \times \frac{1}{2\pi^2 \|\boldsymbol{\Omega}_c^{-1}\|} \times \exp\{-(\mathbf{z}_k - \boldsymbol{\mu}_c)^T \boldsymbol{\Omega}_c^{-1} (\mathbf{z}_k - \boldsymbol{\mu}_c)\} \tag{32}$$

where $c \in I_E$, and $\boldsymbol{\mu}_c$) and $\boldsymbol{\Omega}_c$ is given by

$$\boldsymbol{\mu}_c = \mathbf{H}_k \sum_i \boldsymbol{\mu}_k^{i,c} \tag{33}$$

$$\boldsymbol{\Omega}_c = \mathbf{H}_k \sum \boldsymbol{\Omega}_k^{i,c} \mathbf{H}_k^T + \mathbf{R}_k \tag{34}$$

Then, the particles appearing with $e_i^c = 1$ are put into a set G_c, and the set G_c is associated with the weight w_c.

After all G_c with E_c such that $\Pi(w_i)^{e_i^c}(1 - w_i)^{1-e_i^c} > \epsilon$ are updated, the weights for all set G_c are updated by

$$w_c = \frac{w_c}{\sum_{c \in I_E} w_c} \tag{35}$$

Then, in each group G_c, the weight of ith particle is updated as

$$w_c^i = \frac{w_k^i}{\Pi_{e_i=1} w_k^i} * w_c \tag{36}$$

After above calculation, for the particle $\tilde{P}_k^j$, it has some updated versions, namely, $\{w_c^i, \mathcal{N}(\boldsymbol{\mu}_k^{j,c}, \boldsymbol{\Omega}_k^{j,c}\}_c$ lying in different sets G_c such that $e_j^c = 1$. A K-mean algorithm is used to merge all $\{w_c^i, \mathcal{N}(\boldsymbol{\mu}_k^{j,c}, \boldsymbol{\Omega}_k^{j,c}\}_c$ if needed. If two particles, $\tilde{p}_k^{j,c}$ and $\tilde{p}_k^{j,d}$ are close enough, then they are combined as one particle, the weight of which is set as the summation of both weights, as

$$w_k^j = w_c^j + w_d^j \tag{37}$$

and its covariance is updated as

$$\boldsymbol{\Omega}_k^j = \text{mean}(\boldsymbol{\Omega}_k^{j,c}, \boldsymbol{\Omega}_k^{j,d}) \tag{38}$$

While, the distance between two particles is defined as

$$(\boldsymbol{\mu}_k^{j,c} - \boldsymbol{\mu}_k^{j,d} j)^T (\boldsymbol{\mu}_k^{j,c} - \boldsymbol{\mu}_k^{j,d} j) \tag{39}$$

If the particle $\{w_c^i, \mathcal{N}(\boldsymbol{\mu}_k^{j,c}, \boldsymbol{\Omega}_k^{j,c}\}_c$ is sufficiently far from any other particles, than this particle is added as one particle. By merging close particle and keeping the probability of have more particles give the method a advantage of better estimating the system state, namely, number of total targets and targets' state. The method is verified in a number of simulations in the following section.

5. SIMULATION AND RESULTS

we will add result tomorrow

6. CONCLUSION AND FUTURE WORK

This paper presents a novel filter technique which combines Kalman filter and particle filter for estimating the number and state of total targets based on the measurement obtained online. The estimation is represented by a set of weighted particles, different from classical particle filter, where each particle is a gaussian instead of a point mass. The weight of each particle represents the probability of existing a target, while its gaussian indicates the state distribution for this target. More importantly, the update of particles is different from classical particle filter. For each particle, the gaussian parameters are updated based using Kalman filter given a measurement.

7. APPENDIX

Without losing generality, $\mathcal{P}_S = \{\tilde{P}_k^1, \tilde{P}_k^2, \ldots, \tilde{P}_k^s\}$, $E = [e_1, e_2, \ldots, e_s]$, for any particle such that $e_j = 1$, its $\boldsymbol{\mu}_k^j$ and $\boldsymbol{\Omega}_k^j$, given $\boldsymbol{z}_k$ and $\mathcal{P}_S$.

$$y_k = \boldsymbol{z}_k - \frac{\sum_{i=1}^s \boldsymbol{\mu}_k^i e_i}{\sum_{i=1}^s e_i}) \tag{40}$$

$$\begin{aligned}
\boldsymbol{\Omega_k^j} &= \text{COV}(\boldsymbol{x}_k^j - \boldsymbol{\mu}_k^j) && (41)\\
&= \text{COV}(\boldsymbol{x}_k^j - (\tilde{\boldsymbol{\mu}}_k^j + \boldsymbol{K}_k^j y_k)) && (42)\\
&= \text{COV}(\boldsymbol{x}_k^j - (\tilde{\boldsymbol{\mu}}_k^j + \boldsymbol{K}_k^j (\frac{\sum_{i=1}^s \boldsymbol{x}_k^i e_i}{\sum_{i=1}^s e_i}) + \boldsymbol{\nu}_k - \frac{\sum_{i=1}^s \tilde{\boldsymbol{\mu}}_k^i e_i}{\sum_{i=1}^s e_i})) && (43)\\
&= \text{COV}((\boldsymbol{I} - \frac{1}{\sum_{i=1}^s e_i} \boldsymbol{I}\boldsymbol{K}_k^j)(\boldsymbol{x}_k^j - \tilde{\boldsymbol{\mu}}_k^j) - \boldsymbol{K}_k^j \boldsymbol{\nu}_k - \sum_{i=1, i\neq j}^s \boldsymbol{K}_k^j (\boldsymbol{x}_k^i - \tilde{\boldsymbol{\mu}}_k^i)) && (44)\\
&= (\boldsymbol{I} - \frac{1}{\sum_{i=1}^s e_i} \boldsymbol{I}\boldsymbol{K}_k^j) \tilde{\Omega}_k^j (\boldsymbol{I} - \frac{1}{\sum_{i=1}^s e_i} \boldsymbol{I}\boldsymbol{K}_k^j)^T + \boldsymbol{K}_k^j \boldsymbol{R}_k \boldsymbol{K}_k^j && (45)\\
&\quad + \frac{1}{\sum_{i=1}^s e_i} \boldsymbol{I}\boldsymbol{K}_k^j \sum_{i=1, i\neq j}^s \boldsymbol{\Omega}_k^i (\frac{1}{\sum_{i=1}^s e_i} \boldsymbol{I}\boldsymbol{K}_k^j)^T && (46)
\end{aligned}$$

By setting

$$\frac{\partial \boldsymbol{\Omega_k^j}}{\partial \boldsymbol{K}_k^j} = 0 \tag{47}$$

therefore,

$$\begin{aligned}
\boldsymbol{K}_k^j &= \boldsymbol{\Omega}_k^j (\frac{1}{\sum_{i=1}^s e_i} \boldsymbol{I})^T (\boldsymbol{R}_k + \frac{1}{\sum_{i=1}^s e_i} \boldsymbol{I} \sum_{i=1}^s \boldsymbol{\Omega}_k^i (\frac{1}{\sum_{i=1}^s e_i} \boldsymbol{I})^T)^{-1} && (48)\\
&= \frac{1}{\sum_{i=1}^s e_i} \boldsymbol{\Omega}_k^j (\boldsymbol{R}_k + \frac{1}{(\sum_{i=1}^s e_i)^2} \sum_{i=1}^s \boldsymbol{\Omega}_k^i)^{-1} && (49)
\end{aligned}$$

then,

$$\boldsymbol{\mu}_k^j = \tilde{\boldsymbol{\mu}}_k^j + \boldsymbol{K}_k^j y_k \tag{50}$$

$$\boldsymbol{\Omega}_k^j = \tilde{\boldsymbol{\Omega}}_k^j - \frac{1}{\sum_{i=1}^s e_i}\tilde{\boldsymbol{\Omega}}_k^j(\boldsymbol{R}_k + \frac{1}{\sum_{i=1}^s e_i}\boldsymbol{I}\sum_{i=1}^s \boldsymbol{\Omega}_k^i(\frac{1}{\sum_{i=1}^s e_i}\boldsymbol{I})^T)^{-1}\tilde{\boldsymbol{\Omega}}_k^j \tag{51}$$

REFERENCES

[1] Ferrari, S., Cai, C., Fierro, R., and Perteet, B., "A multi-objective optimization approach to detecting and tracking dynamic targets in pursuit-evasion games," in [*Proc. of the 2007 American Control Conference*], 5316–5321 (2007).

[2] Culler, D., Estrin, D., and Srivastava, M., "Overview of sensor networks," *Computer* **37**(8), 41–49 (2004).

[3] Juang, P., Oki, H., Wang, Y., Martonosi, M., Peh, L., and Rubenstein, D., "Energy efficient computing for wildlife tracking: Design tradeoffs and early experiences with zebranet," in [*Proc. 10th International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS-X)*], 96–107 (2002).

[4] G. Welch, G. B., "An introduction to the kalman filter," tech. rep., Department of Computer Science, University of North Carolina at Chapel Hill.

[5] Julier, S. J. and Uhlmann, J. K., "A new extension of the kalman filter to nonlinear systems," *Proc. AeroSense: 11th Int. Symp. Aerospace/Defense Sensing, Simulation and Controls, pp. 182-197* (1997).

[6] Wan, E. and Van Der Merwe, R., "The unscented kalman filter for nonlinear estimation," in [*Proceedings of the IEEE 2000 Adaptive Systems for Signal Processing, Communications, and Control Symposium*], 153 –158 (2000).

[7] Julier, S., "The scaled unscented transformation," in [*American Control Conference, 2002. Proceedings of the 2002*], **6**, 4555–4559 (2002).

[8] Khan, Z., Balch, T., and Dellaert, F., "An mcmc-based particle filter for tracking multiple interacting targets," in [*Computer Vision - ECCV 2004*], Pajdla, T. and Matas, J., eds., 279–290 (2004).

[9] Kreucher, C., Kastella, K., and Hero, O., "Multitarget tracking using the joint multitarget probability density," *IEEE Transactions on Aerospace and Electronic Systems* **41**(4), 1396–1414 (2005).

[10] Arulampalam, M., Maskell, S., Gordon, N., and Clapp, T., "A tutorial on particle filters for online nonlinear/non-gaussian bayesian tracking," *Signal Processing, IEEE Transactions on* **50**(2), 174 –188 (2002).

[11] Carpenter, J., Clifford, P., and Fearnhead, P., "Improved particle filter for nonlinear problems," in [*Radar, Sonar and Navigation, IEEE Proceedings*], **146**(1), 2 –7 (1999).

[12] Rui, Y. and Chen, Y., "Better proposal distributions: Object tracking using unscented particle filter," *Computer Vision and Pattern Recognition, IEEE Computer Society Conference on* **2** (2001).

[13] W. Lu, G. Zhang, S. F. R. F. and Palunko, I., "An information potential approach for tracking and surveilling multiple moving targets using mobile sensor agents," *Proc. of SPIE* (2011).

[14] O'Gorman and W.", T., [*Applied adaptive statistical method: test of sigficance and condifence intervals*], Society for Industrial and Applied Mathematics,Philadelphia (2004).