Distributed Data Association for Multitarget Tracking-A Mathematical Perspective

R. Sangeetha, B. Kalpana

Abstract

Target tracking is the technique of maintaining state estimates of one or more targets over a period of time. Multitarget tracking is concerned with the state estimation of an unknown number of moving targets. The objective of multitarget tracking is to enable the sensor subsystem to identify and track multiple targets given atmospheric disturbances and clutter environment, which obscure the target. In this paper, we present certain mathematical models with Kalman consensus filter, data association concepts and maximum mutual information based sensor selection related to our work and propose track to track fusion which is a well-organized technique for multitarget tracking. We prove theoretically that mathematical models can improve the distributed data association technique with mutual information based sensor selection and it results in an optimal feasible solution for multitarget tracking.

1. Introduction

Target tracking is extremely important and challenging problem in military and civilian applications, since it deals with data association, intricate signal processing, real world decision making, multi modal sensing and so on. Military applications include ballistic defense, air traffic control, missile guidance, ocean and battlefield surveillance [3]. Multitarget tracking is entirely different from single target tracking and it has two significant challenges [4] [5]: Data Association and Estimation. Multitarget tracking requires a complex distributed data association technique to partition the detected measurements to each individual data source and establish their correspondence with the maintained trackers. If a target is detected then the target parameters are verified before it creates a track for that target. The major elements of a track file are target position, target velocity and target acceleration.

* RSangeetha. Tel.: +91-9367176785
E-mail address: sangeethadj@gmail.com.
The paper is organized as follows: Section 2 deals with the mathematical models for the target state estimation. Section 3 describes the issues and proposed technique for Distributed Data Association in Multitarget tracking. Section 4 discusses the maximum mutual information based sensor selection proposed in [1]. Section 5 is a conclusion of the theoretical and mathematical proposition work with an outline of the future scope.


In this section, we describe the process model for the target state estimation. The acquaintance of the target’s current state and its transition matrix determine the next state of the target. The observation model helps to simulate the sensor behavior when target’s existence is known. The observable state of the target depends on the sensing mode employed by the sensor. In addition, sensor selection strategy is used. We assume a range that comes under surveillance with a total of $N_T$ sensors and $S_m(t)$ are the set of nodes of that are neighbors with $S_m^i$ (an arbitrary sensor node) at time $t$ for $i^{th}$ target.

2.1. Process Model

The process model [4][8] finds the state of the target at time instant $t$ given the state of the target at time instant $t-1$. Let us consider a target state vector based on piecewise constant acceleration model with $M$ number of targets. The target process state vector is defined

$$ X = [x \ y \ z \ x \ y \ z \ x \ y \ z]^T, \quad (1) $$

and it evolves in time according to

$$ X^i_t = F X^i_{t-1} + V^i_t, \quad \forall \ i = 1,2,...,M \quad (2) $$

where

$$ F = \begin{bmatrix} I & TI & (T^2 / 2)I \\ 0 & I & TI \\ 0 & 0 & I \end{bmatrix}, \quad V^i = [V^i_x \ V^i_y \ V^i_z] $$

In the above equations, the subscript $t$ denotes time index, superscript $T$ represents the transpose, $X$ is the real target state vector, $F$ is the target state transition matrix, $V$ is the Gaussian-distributed process noise vector assumed to be zero mean in the $x$, $y$, $z$ directions respectively. $I$ is the identity matrix of order 3x3, 0 is the null matrix of order 3x3 and $T$ is the sampling time. The maneuver variance of $i^{th}$ target $V^i$ can be measured by $(Q\delta)^i$, where $\delta$ is Dirac delta function and $Q$ is the process noise covariance

$$ Q = 2\alpha\sigma_m^2 \begin{bmatrix} (T^6 / 20)I & (T^4 / 8)I & (T^3 / 6)I \\ (T^4 / 8)I & (T^3 / 3)I & (T^2 / 2)I \\ (T^3 / 6)I & (T^2 / 2)I & TI \end{bmatrix}, \quad (3) $$

where $\alpha$ is the reciprocal of the constant time and $\sigma_m^2$ is the variance assumed equal along the three axes,

$$ \sigma_m^2 = A^2_{max} [1 + 4P_{max} - P_0] / 3 \quad (4) $$
equation (4) represents the target acceleration with uniform distribution, between the limits $A_{\text{max}}$ to $A_{\text{max}}$ and probability varies from $P_0$ to $P_{\text{max}}$.

2.2. Observation Model [4] [8]

To model the sensor observation uncertainties, noise is added to the real target state vector. Sensor nodes can only observe the two dimensions of the target state, velocity and acceleration of the target state are not observable. But the measurement of the target state location is available in spherical coordinates. Moreover, sensor nodes collect range, bearing and elevation data, since they do not observe the target coordinates directly.

The $i$th targets range $r_{i,m,d}$, bearing $\theta_{i,m,d}$ and elevation $\phi_{i,m,d}$ measured by $S_m$ are defined with the respect to the true range $r_{i,m}$, bearing $\theta_{i,m}$ and elevation $\phi_{i,m}$ as

$$ r_{i,m,d} = r_{i,m} + n_r, \quad \theta_{i,m,d} = \theta_{i,m} + n_\theta, \quad \phi_{i,m,d} = \phi_{i,m} + n_\phi $$

where range $r_{i,m}$, bearing $\theta_{i,m}$ and elevation $\phi_{i,m}$ denote the radar measurements. The original values $r$, $b$ and $e$ can be expressed as

$$ r = \sqrt{x^2 + y^2 + z^2}, \quad b = \tan^{-1}(y/z), \quad e = \tan^{-1}(z/\sqrt{x^2 + y^2}) $$

The errors in range $n_r$, bearing $n_\theta$ and elevation $n_\phi$ are assumed to be independent and moments of Gaussian distribution:

$$ \text{E} [n_{r,m}] = \text{E} [n_{\theta,m}] = \text{E} [n_{\phi,m}] = 0 $$

$$ \text{E} [(n_{r,m})^2] = \sigma_{r,m}^2, \quad \text{E} [(n_{\theta,m})^2] = \sigma_{\theta,m}^2, \quad \text{E} [(n_{\phi,m})^2] = \sigma_{\phi,m}^2 $$

where the $i$th targets time dependence is implicit. The mean target state vector observed after the unbiased spherical to cartesian coordinates is given as follows,

$$ \varphi_m = \begin{bmatrix} x_{i,m,d} \\ y_{i,m,d} \\ z_{i,m,d} \end{bmatrix} = \begin{bmatrix} r_{i,m,d} \cos \theta_{i,m,d} \cos \phi_{i,m,d} \\ r_{i,m,d} \sin \theta_{i,m,d} \cos \phi_{i,m,d} \\ r_{i,m,d} \sin \phi_{i,m,d} \end{bmatrix} - \mu_m, $$

where $\mu_m$ is average true bias and $\mu_m = \begin{bmatrix} r_{i,m,d} \cos \theta_{i,m,d} \cos \phi_{i,m,d} (e^{-\sigma_{\theta,m}^2} - e^{\sigma_{\theta,m}^2} - e^{-\sigma_{\phi,m}^2/2} - e^{\sigma_{\phi,m}^2/2}) \\ r_{i,m,d} \sin \theta_{i,m,d} \cos \phi_{i,m,d} (e^{-\sigma_{\theta,m}^2} - e^{\sigma_{\theta,m}^2} - e^{-\sigma_{\phi,m}^2/2} - e^{\sigma_{\phi,m}^2/2}) \\ r_{i,m,d} \sin \phi_{i,m,d} (e^{-\sigma_{\theta,m}^2} - e^{\sigma_{\phi,m}^2/2}) \end{bmatrix}$. 


The covariance matrix $R^i_m$, of the observation errors in $\phi_m$ is

$$
R^i_m = \begin{bmatrix}
R^i_{m11} & R^i_{m12} & R^i_{m13} \\
R^i_{m21} & R^i_{m22} & R^i_{m23} \\
R^i_{m31} & R^i_{m32} & R^i_{m33}
\end{bmatrix}
$$

(10)

where $R^i_{m11}, R^i_{m12}, R^i_{m13}, R^i_{m22}, R^i_{m23}, R^i_{m33}$ are given in Appendix as in [2,3].

2.3. Multitarget Tracking with Kalman Consensus Filter [9]

As in [9], state estimation can be calculated using Kalman-Consensus Filter using distributed average-consensus on estimates obtained by local kalman filters. Node $m$ starts iteration $k$ by taking a set of measurements

$$
Z_m[t] = \{ z_{m1}[t], z_{m2}[t], \ldots \},
$$

(11)

where each measurement $z_{mj}[t]$ corresponds to a target. The next step is for node $m$ to assign its measurement set $Z_m[t]$ to its set of tracks $\chi = \{ \hat{x}_{1}, \hat{x}_{2}, \ldots \}$, The result of any filter is the optimal matching $f_m: Z_m[t] \rightarrow \chi$. The matching weights $w^i_{mj}$ for assigning measurement $z_{mj}$ to track $i$ are

$$
w^i_{mj} = (z_{mj} - H_m \bar{x}^i_m)^T (H_m P^i_m H_m^T + R_m)^{-1} (z_{mj} - H_m \bar{x}^i_m),
$$

(12)

and the number of standard deviations between $z_{mj}$ and the expected measurement. Let $\phi^i_m[t]$ be the measurement assigned to target $i$ so that $f_m(\phi^i_m[t]) = x_i$. The information vector and matrix for the track of target $i$ on node $m$ are

$$
u^i_m[t] = H_m^T R_m^{-1} \phi^i_m[t],
$$

$$
U^i_m[t] = H_m^T R_m^{-1} H_m,
$$

(13)

with $H_m$ being the observation matrix of $S_m$ that relates the target state estimate $x_m$ to the sensor measurement $\phi^i_m[t]$. An assignment algorithm is used to form a set of optimal matchings $g^{ml}: \chi \rightarrow \chi$, where $g^{ml}$ matches the tracks of node $i$ with the tracks of node $l$. The cost of assigning track $i_1$ of node $m$ to track $i_2$ of node $l$ is defined [8] as

$$
(x_m^{i_1} - x_l^{i_2})^T (P_m^{i_1} + P_l^{i_2})^{-1} (x_m^{i_1} - x_l^{i_2}),
$$

where $P^i_m$ is the covariance of the estimation and $x_m^{i}$ denote the prior state of the target $i$ by node $m$. 

Now, the assignment functions $g_{ml}$ allow the Kalman-Consensus Filter updates to be completed. Information vectors and matrices for track $i$ on node $m$ updates its own belief as give below

$$y^i_m[t] = \sum_{l \in J_m} u^l_{g_{ml}(i)}[t] Y^i_m = \sum_{l \in J_m} U^l_{g_{ml}(i)}[t] \quad (14)$$

### 3. Distributed Data Association for Multitarget Tracking using Kalman Consensus Filter

In multi-target tracking, Data Association [6,7] is a challenging task with measurement-to-measurement association, measurement-to-track association, track-to-track association, and track-to-sensor association. Measurements-to-track association [8] gives poor performance under a dense clutter scenario, as the possibility of a false assignment is not considered in the estimation. In this paper we discuss the track to track and measurement to track data association.

#### 3.1. Measurement to Track associations

Joint Probabilistic Data Association (JPDA) [4,7] is an effective method in handling clutter and missed detections. Joint Probabilistic Data Association is used to perform local measurement to track association [8] and the measurements assigned to track $i$ of node $m$ takes the form of the equation given below

$$z^i_m = \sum_{j=1}^{n_m} \beta^i_{mj} z_{mj} \quad (15)$$

The most complex approach to data association involves considering all possible measurement-to-track associations from the initialization of the system to the current time [10]. The JPDA state update equation is given by

$$\hat{x}^i_m = x^i_m + \left((P^i_m)^{-1} + U^i_m\right)^{-1} \left(u^i_m - (1 - \beta^i_{m0}) U^i_m x^i_m \right) \quad (16)$$

where $\beta^i_{mo} = \frac{\exp\left(-\left(\sqrt{\hat{z}^i_{mj}/2}\right)\right)}{b + \sum_{j=1}^{n_m} \exp\left(-\left(\sqrt{\hat{z}^i_{mj}/2}\right)\right)}$, $b = (2\pi)^{d/2} \lambda_0 |\Lambda|^{1/2} (1 - P_D P_G) / P_D$.

The above information allows multiple independent measurement sources as sum of information with its update. The JPDA is a relatively simple recursive method, which does not require the storage of past observation data. JPDA is a good strategy to track highly maneuverable targets where process model, observation model present a strong nonlinearity, and the noises are not Gaussian.
3.2. Track to Track associations

Track-to-track fusion is an important issue in Multitarget data association. Unlike the measurement-to-track association, a particular challenge associated with this test is to account for the cross correlation between the tracks of the same target due to the common process noises [12].

Generally, there are two steps in a track-to-track association approach [9]
- Take a target feature, compute the similarity or distance between two tracks under that feature, then get the matrix of all distances.
- Choose an assignment algorithm to find the association.

The traditional approach for track-to-track association [11] with only absolute kinematic states uses a chi-squared distance to measure the similarity between two tracks. Given node m and track-to-track association [8] function $\mathcal{G}_{ml}$, the state update takes the form

$$\hat{x}_m^i = x_m^i + \left( (P_m^i)^{-1} + Y_m^i \right)^{-1} (y_m^i - Y_m^i x_m^i)$$  \hspace{1cm} (17)

where $y_m^i$ and $Y_m^i$ are given in the equation (14), $Y_m^{i_0} = (\beta_m^i Y_m^i)$, $J_m^i = N_m^i \cup m$, $J_m$ is formed using the estimated mutual information values of its neighbors and

$$\beta_m^i = \frac{b}{b + \sum_{j \neq i} \exp \left( - \frac{1}{2} \left( \frac{z_m^i}{w_{mj}} \right)^T \left( \frac{1}{w_{mj}} \right) \right)}.$$  \hspace{1cm} (18)

where $z_{mj}^i = z_{mj} - H_m x_m^i$, $d$ is the dimension of the space in which the target is moving, $P_D$ is the probability of detecting a target, $P_G$ is the probability of a target lying in a gate used to eliminate events with negligible probabilities, $\Lambda$ is the covariance of $z_{mj}$'s.

4. Maximum Mutual Information based Sensor Selection for Multitarget Tracking

Mutual information measures mutual dependence of two or more sensors. It is calculated to determine the current target state based on current observation. In [1], the mutual information gain based on the $m^{th}$ sensor is formulated as

$$J_m^i(t, \varphi(t)) = \frac{1}{2} \log \left[ \frac{U_m^i(t/t)}{U_m^i(t/t-1)} \right]$$  \hspace{1cm} (19)

where $U_m^i(t/t)$ is the information matrix at time instant $t$ after the $i^{th}$ target's state is observed, $m=1,2,\ldots,N_T$. The quantitative value of a sensor node is computed by the equation (19). It calculates the reduction in target state uncertainty with the new data. If the quantitative value obtained from the equation (19) is higher, then the sensor statistics about the target state is more informative.
5. Practical Example for Multitarget Tracking

Dynamic Modeling is an efficient statistical tool and thriving research area in Estimation theory, Control Systems and Time Series Analysis. In spite of its computational complexity, dynamic models play a vital role in multitarget tracking applications. Practical examples for multitarget tracking are aircraft tracking, sea animals or submarines tracking, ship navigation, ground vehicle tracking and many more.

From [13], aircraft and ship tracking examples explain multitarget tracking problem in real time environment. Example (i) an air traffic controller at a busy airport may incorrectly decide that a new return on his radar display corresponds to an aircraft already being tracked, rather than correctly recognizing the appearance of a new aircraft. Example (ii), a sonar operator may decide, hearing a number of echoes from a given sector that several ships are present, but be unable to accurately separate the echoes into individual ships tracks.

6. Conclusion

Sensor networks are localized and synchronized to track a target effectively. Also, it must be able to assess the jagged distance between the target and the sensor. In [1], distributed data fusion with mutual information based sensor selection for a single target was discussed. As an extension, we propose a distributed data association with mutual information based sensor selection strategies for multitarget tracking.

Informative subset of sensor nodes are selected using mutual information strategy to reduce the target localization error by improving the target state filtering performance. A distributed kalman-consensus filter is used to track multitrajects persistently. From the above models and strategies, we propose track to track association with Kalman-Consensus filter for multitarget tracking is better than measurement to track fusion association. In future, the stated theoretical strategies and mathematical models can be implemented to attain an optimal feasible solution.

References

Appendix

\[ R_{m11} = r_m^2 \left[ (\sin^2 \theta_m \sinh 2\sigma_\theta^2 + \cos^2 \theta_m \cosh 2\sigma_\theta^2)(\sin^2 \phi_m \sinh 2\sigma_\phi^2 + \cos^2 \phi_m \cosh 2\sigma_\phi^2) \right] \\
- [(\sin^2 \theta_m \sinh \sigma_\theta^2 + \cos^2 \theta_m \cosh \sigma_\theta^2)(\sin^2 \phi_m \sinh \sigma_\phi^2 + \cos^2 \phi_m \cosh \sigma_\phi^2)] \right) \\
+ \sigma_r^2 [2(\sin^2 \theta_m \sinh 2\sigma_\theta^2 + \cos^2 \theta_m \cosh 2\sigma_\theta^2)(\sin^2 \phi_m \sinh 2\sigma_\phi^2 + \cos^2 \phi_m \cosh 2\sigma_\phi^2)] \\
- [(\sin^2 \theta_m \sinh \sigma_\theta^2 + \cos^2 \theta_m \cosh \sigma_\theta^2)(\sin^2 \phi_m \sinh \sigma_\phi^2 + \cos^2 \phi_m \cosh \sigma_\phi^2)] \right] \cdot e^{-2\sigma_\theta^2} e^{-2\sigma_\phi^2} \\
R_{m12} = r_m^2 \left[ (\sin^2 \phi_m \sinh 2\sigma_\phi^2 + \cos^2 \phi_m \cosh 2\sigma_\phi^2) - (\sin^2 \phi_m \sinh \sigma_\phi^2 + \cos^2 \phi_m \cosh \sigma_\phi^2) \right] e^{\sigma_\phi^2} \\
R_{m22} = r_m^2 \left[ (\sin^2 \theta_m \cosh 2\sigma_\theta^2 + \cos^2 \theta_m \sinh 2\sigma_\theta^2)(\sin^2 \phi_m \sinh 2\sigma_\phi^2 + \cos^2 \phi_m \cosh 2\sigma_\phi^2) \right] \\
- [(\sin^2 \theta_m \cosh \sigma_\theta^2 + \cos^2 \theta_m \sinh \sigma_\theta^2)(\sin^2 \phi_m \sinh \sigma_\phi^2 + \cos^2 \phi_m \cosh \sigma_\phi^2)] \right) \\
+ \sigma_r^2 [2(\sin^2 \theta_m \cosh 2\sigma_\theta^2 + \cos^2 \theta_m \sinh 2\sigma_\theta^2)(\sin^2 \phi_m \sinh 2\sigma_\phi^2 + \cos^2 \phi_m \cosh 2\sigma_\phi^2)] \\
- [(\sin^2 \theta_m \cosh \sigma_\theta^2 + \cos^2 \theta_m \sinh \sigma_\theta^2)(\sin^2 \phi_m \sinh \sigma_\phi^2 + \cos^2 \phi_m \cosh \sigma_\phi^2)] \right] \cdot e^{-2\sigma_\theta^2} e^{-2\sigma_\phi^2} \\
R_{m23} = r_m^2 \left[ (1 - e^{-\sigma_\phi^2}) + \sigma_r^2 (2 - e^{-\sigma_\phi^2}) \right] \cdot \sin \theta_m \sin \phi_m \cos \phi_m . e^{-\sigma_\theta^2} e^{-4\sigma_\phi^2} \\
R_{m33} = r_m^2 \left[ (\sin^2 \phi_m \cosh 2\sigma_\phi^2 + \cos^2 \phi_m \sinh 2\sigma_\phi^2) - (\sin^2 \phi_m \cosh \sigma_\phi^2 + \cos^2 \phi_m \sinh \sigma_\phi^2) \right] \\
+ \sigma_r^2 [2(\sin^2 \phi_m \cosh 2\sigma_\phi^2 + \cos^2 \phi_m \sinh 2\sigma_\phi^2) - (\sin^2 \phi_m \cosh \sigma_\phi^2 + \cos^2 \phi_m \sinh \sigma_\phi^2)] \right] \cdot e^{-2\sigma_\phi^2}