Joint Integrated Track Splitting for Multi-sensor Multi-target Tracking in Clutter

Yifan Xie[1], Haeho Lee[2], Myonghwan Ahn[2], Bum Jik Lee[3] and Taek Lyul Song[1]
[1]*Department of Electronic Systems Engineering, Hanyang University, Ansan, Republic of Korea*
[2]*Combat System R&D Lab, LIG Nex1 Co. Ltd., Seongnam, Republic of Korea*
[3]*Submarine Combat System Part, Daewoo Shipbuilding & Marine Engineering, Seoul, Republic of Korea*

Keywords: Multitarget, Multisensor, Target Existence, Joint Integrated Track Splitting, S-D Assignment.

Abstract: Automatic object tracking for track initialization, confirmation and termination can be realized by using the probability of target existence, which is a track quality measure for false track discrimination. In this paper, we present a multi-sensor multi-target tracking application based on the probability of target existence and multi-sensor joint integrated track splitting (MS-JITS), which is an extension of JITS framework to multi-sensor systems. For fair comparison, incorporation of the target existence paradigm and the S-D assignment is implemented. This work also consummates the S-D assignment based estimators for track management by the probability of target existence.

1 INTRODUCTION

In many target tracking surveillance systems, such as air traffic control or ground target tracking for multiple targets, measurements can be received from multiple sensors. Because of the limited sensing region of sensors, a single sensor can only partially access information of the environment. Information from multiple sensors can be combined using data fusion algorithms to achieve cooperative observation effects (Xiong and Svensson, 2002). However the target tracker has no prior knowledge on the origin of each measurement, which terms multi-dimensional assignment (MDA) problem (Gilbertand and Hofstra, 1988) in multi-sensor systems. Pattipatti et al. proposed the S-D assignment algorithm to solve this non-deterministic polynomial-time hard (NP-hard) problem. By using the Lagrangian(dual) relaxation approach, the problem is then solved as a series of 2-D subproblems (Pattipati et al., 1992; Deb et al., 1997).

In multi-target scenario, a mechanism is required to decide the source (clutter or particular track) of received measurements, and the objective is to associate measurement-to-track. This problem turns more critical in multi-sensor multi-target situations, where the sensor fusion problem has to be overcomed as well. The conventional solutions for multi-target tracking (MTT) include joint probabilistic data association (JPDA) (Blackman and Popoli, 1999) and multi-hypothesis tracking (MHT)(Reid, 1979; Blackman, 2004), etc. For multi-sensor scenarios, multi-sensor JPDA (Frei and Pao, 1998; Pao and Frei, 1995) and multi-sensor MHT (Kirubarajan et al., 2001) are proposed. These algorithms enumerate all possible measurement to target allocations, and use likelihood to evaluate each hypothesis. The number of hypotheses grows exponentially in the number of tracks and the number of measurements involved. The computational cost in generating the possibilities to data association is usually excessive when the number of tracks and number of measurements are large. Therefore, suboptimal solutions with computational feasibility are required.

Integrated track splitting (ITS) (Mušicki et al., 2007; Shi et al., 2015b) is a suboptimal multi-scan single target estimator, that is capable of false track discrimination (FTD) using the probability of target existence as a track quality measure. The false track discrimination identifies the true tracks and eliminates false alarms, which is of the essential functionality in track maintenance (Song et al., 2013; Song et al., 2015c; Song et al., 2015a). The probability of target existence for false track discrimination is obtained by a simple equation which utilizes the parameters involved in the associate probability calculations. ITS has been extended for dealing with multi-target as joint integrated track splitting (JITS) (Mušicki and

Xie, Y., Lee, H., Ahn, M., Lee, B. and Song, T.
Joint Integrated Track Splitting for Multi-sensor Multi-target Tracking in Clutter.
DOI: 10.5220/0005960102990307
In *Proceedings of the 13th International Conference on Informatics in Control, Automation and Robotics (ICINCO 2016) - Volume 1*, pages 299-307
ISBN: 978-989-758-198-4

Evans, 2009; Shi et al., 2015a). Similar to MHT, each component of JITS forms the history of measurement hypotheses involved in the track for multiple scans. The JITS algorithm is also able to distinguish between the true tracks and false alarms by recursively calculating the probability of target existence.

In this paper, we present a sub-optimal multi-sensor multi-scan multi-target tracking algorithm based on the JITS algorithm, which we call multi-sensor joint integrated track splitting (MS-JITS). A comparison between the S-D assignment algorithm and the MS-JITS algorithm is also conducted.

Problem formulation is detailed in Section II. Section III presents the general MS-JITS algorithm. An application of target existence paradigm to S-D assignment is described in Section IV. A simulation study which demonstrates the performance of MS-JITS is given in Section V, followed by the concluding remarks in Section VI.

2 PROBLEM FORMULATION

2.1 Measurement Model

A number of measurements of sensors are received at each sampling time k synchronously. Due to the uncertainty of measurement origins, the received measurements may originate from the target or clutter. Let P_{Ds} denote the target detection probability of sensor s. Since the targets are not always detectable, $P_{Ds} < 1$.

The gating process selects measurements from sensor s and forms a validated measurement set $y_s(k)$ with cardinality m_k^s at time k, $y_{s,i}(k)$ indicates the ith element of $y_s(k)$. P_G is the probability that the true measurement falls in the gate if target exists. Denote $Y_s^k = \{y_s(k), Y_s^{k-1}\}$ as the set of validated measurements up to time k. And a collection of all sensor measurement is $Y^k = \{Y_1^k, Y_2^k, ..., Y_S^k\}$.

2.2 Target Model

The target dynamics are modeled in Cartesian coordinates. Under the additive noise assumption, the τth target kinematic and measurement equations for track τ at k scan are defined by

$$x_k^\tau = F_k^\tau x_{k-1}^\tau + \omega_k^\tau, \quad (1)$$

$$z_{s,k}^\tau = H_s x_k^\tau + v_{s,k}. \quad (2)$$

where x_k^τ is the target τ state vector, $z_{s,k}^\tau$ is the target measurement. F_k^τ and H_s are the transition matrix and measurement matrix of sensor s respectively. The process noise ω_k^τ and measurement noise $v_{s,k}$ are assumed to be zero-mean, uncorrelated white Gaussian noise sequences with known covariance Q_k and $R_{s,k}$.

As for track existence state, ψ_k^τ is the event that target τ exists. On the contrary, $\bar{\psi}_k^\tau$ suggests that target τ does not exist at scan k.

2.3 Clutter Model

The number of clutter measurements in the surveillance space follows Poisson distribution. The clutter measurement density at point $y_{s,i}(k)$ is denoted by $\rho_{s,i}(k) \triangleq \rho(y_{s,i}(k))$. Spatial distribution of clutter measurements is assumed to be uniform in the surveillance space.

3 MS-JITS ALGORITHM

3.1 Single-sensor JITS

3.1.1 Track State

The track state probability density function (pdf) conditioned on the measurement set Y_s^k at time k is given by

$$p(x_k^\tau, \psi_k^\tau | Y_s^k) = p(x_k^\tau | \psi_k^\tau, Y_s^k) P(\psi_k^\tau | Y_s^k), \quad (3)$$

where $P(\psi_k^\tau | Y_s^k)$ represents the probability of target existence of track τ. It also suggests that the track state pdf is always calculated conditioned on the target existence event ψ_k^τ. Specifically, the track state pdf is approximated by a Gaussian mixture of mutually exclusive and exhaustive track components such as

$$p(x_k^\tau | \psi_k^\tau, Y_s^k) = \sum_{c_k^\tau} \xi_k^\tau (c_k^\tau | \psi_k^\tau, Y_s^k) p(x_k^\tau | c_k^\tau, \psi_k^\tau, Y_s^k). \quad (4)$$

The track component c_k^τ consists of the following compositions:

- given measurement history;
- trajectory state pdf $p(x_k^\tau | c_k^\tau, \psi_k^\tau, Y_s^k)$;
- component probability $\xi_k^\tau (c_k^\tau | \psi_k^\tau, Y_s^k)$, which is indexed by c_k^τ, subjects to the constraint of

$$\sum_{c_k^\tau} \xi_k^\tau (c_k^\tau | \psi_k^\tau, Y_s^k) = 1. \quad (5)$$

3.1.2 Track Prediction

Track state predicted pdf involved in propagating from time $k-1$ to k yields

$$p(x_k^\tau, \psi_k^\tau | Y_s^{k-1}) = p(x_k^\tau | \psi_k^\tau, Y_s^{k-1}) P(\psi_k^\tau | Y_s^{k-1}). \quad (6)$$

Assumed that the target is always observable whenever it exists, then the existence and observability can be modeled with Markov Chain 1 (Mušicki et al., 1994). The predicted target existence is given by

$$P(\psi_k^\tau|Y_s^{k-1}) = \Delta_{1,1}P(\psi_{k-1}^\tau|Y_s^{k-1}), \quad (7)$$

where $\Delta_{1,1}$ is the Markov transition probability.

The predicted track state pdf is denoted as

$$p(x_k^\tau|\psi_k^\tau, Y_s^{k-1}) = \sum_{c_{k-1}^\tau} \xi_k^\tau(c_{k-1}^\tau|\psi_k^\tau, Y_s^{k-1}) \times p(x_k^\tau|c_{k-1}^\tau, \psi_k^\tau, Y_s^{k-1}). \quad (8)$$

Relative track component probabilities do not change during the propagation. Each track component propagates individually based on the prediction step of the Kalman filter such that yields

$$p(x_k^\tau|c_{k-1}^\tau, \psi_k^\tau, Y_s^{k-1}) = N(x_k^\tau; \hat{x}_{k|k-1}^\tau(c_{k-1}^\tau), \hat{P}_{k|k-1}^\tau(c_{k-1}^\tau)). \quad (9)$$

3.1.3 Measurement Selection

Each track component selects measurements from sensor s separately, the validated measurements will be included in the set $y_s(k)$. And the relevant measurement likelihood becomes

$$\begin{aligned} p_{k,i}(c_{k-1}^\tau) &\triangleq p(y_{s,i}(k)|\hat{x}_{k|k-1}^\tau(c_{k-1}^\tau)) \\ &= \frac{1}{P_G} N(y_{s,i}(k); H\hat{x}_{k|k-1}^\tau(c_{k-1}^\tau), \hat{S}_k^\tau(c_{k-1}^\tau)), \end{aligned} \quad (10)$$

where $\hat{S}_k^\tau(c_{k-1}^\tau)$ is predicted measurement error covariance. The mixed likelihood for the common measurement shared by components in track τ becomes

$$p_{k,i} = \sum_{c_{k-1}^\tau} \xi_{k-1}^\tau(c_{k-1}^\tau) p_{k,i}(c_{k-1}^\tau). \quad (11)$$

3.1.4 Joint Multi-target Data Association

In order to reduce the computational complexity, tracks that share common validated measurements are termed a cluster. Joint multi-target data association is performed simultaneously for each cluster.

A joint event ε is an assignment among possible assignments of all measurements to all tracks in the cluster, and it must obey the principle that each track is assigned with zero or one measurement; each measurement is assigned to zero or one track. The joint events are mutually exclusive (Mušicki and Evans, 2009).

Set $T_0(\varepsilon)$ is composed by tracks with no allocated measurement; Set $T_1(\varepsilon)$ is the collection of tracks allocated one measurement; $i(\varepsilon,\tau)$ indicates that measurement i in ε is allocated to track τ. And the corresponding joint event probability is

$$P(\varepsilon|Y_s^k) = C_k^{-1} \prod_{\varepsilon\in T_0(\varepsilon)} [1 - P_{Ds}P_G P(\psi_k^\tau|Y_s^{k-1})] \times \prod_{\varepsilon\in T_1(\varepsilon)} [P_{Ds}P_G P(\psi_k^\tau|Y_s^{k-1}) \frac{p_k(i(\varepsilon,\tau))}{\rho_{s,i}(k)}], \quad (12)$$

where C_k is the normalization constant, $p_k(i(\varepsilon,\tau))$ is the likelihood that measurement i originates from target τ. Since the joint events are mutually exclusive, they form an exhaustive set

$$\sum_\varepsilon P(\varepsilon|Y_s^k) = 1. \quad (13)$$

A set of all joint events that measurement i is assigned to track τ is denoted by $\Xi(i,\tau)$. Hypothesis $\theta_k^\tau(i)$ denotes that measurement i is the detection of target τ at time k, and the probability of $\theta_k^\tau(i)$ satisfies

$$p(\theta_k^\tau(0)|Y_s^k) = \sum_{\varepsilon\in\Xi(0,\tau)} P(\varepsilon|Y_s^k), \quad (14)$$

$$p(\psi_k^\tau, \theta_k^\tau(0)|Y_s^k) = \frac{(1 - P_{Ds}P_G)P(\psi_k^\tau|Y_s^{k-1})}{1 - P_{Ds}P_G P(\psi_k^\tau|Y_s^{k-1})} \times p(\theta_k^\tau(0)|Y_s^k) \quad (15)$$

$$p(\psi_k^\tau, \theta_k^\tau(i)|Y_s^k) = \sum_{\varepsilon\in\Xi(i,\tau)} P(\varepsilon|Y_s^k). \quad (16)$$

The data association probabilities based on the assumption of target existence are then given by

$$\begin{aligned} \beta_{k,i}^\tau &\triangleq p(\theta_k^\tau(i)|\psi_k^\tau, Y_s^k) \\ &= \frac{p(\psi_k^\tau, \theta_k^\tau(i)|Y_s^k)}{P(\psi_k^\tau|Y_s^k)}. \end{aligned} \quad (17)$$

3.1.5 Track Update

The track state pdf is updated by the measurements in $y_s(k)$. The update track state can be obtained from the Bayes rule

$$p(x_k^\tau, \psi_k^\tau|Y_s^k) = p(x_k^\tau|\psi_k^\tau, Y_s^k)P(\psi_k^\tau|Y_s^k) \quad (18)$$

where the posterior probability of target existence is given by

$$P(\psi_k^\tau|Y_s^k) = \sum_{i=0}^{m_k^s} p(\psi_k^\tau, \theta_k^\tau(i)|Y_s^k) \quad (19)$$

The track state a posterior pdf $p(x_k^\tau|\psi_k^\tau, Y_s^k)$ is a mixture of all track components pdf such that

$$p(x_k^\tau|\psi_k^\tau, Y_s^k) = \sum_{c_k^\tau} \xi_k^\tau(c_k^\tau|\psi_k^\tau, Y_s^k) p(x_k^\tau|c_k^\tau, \psi_k^\tau, Y_s^k) \quad (20)$$

where track component posterior probability $\xi_k^\tau(c_k^\tau|\psi_k^\tau, Y_s^k)$ is given by

$$\xi_k^\tau(c_k^\tau|\psi_k^\tau, Y_s^k) = \begin{cases} \xi_k^\tau(c_{k-1}^\tau|\psi_k^\tau, Y_s^{k-1})\beta_{k,i}^\tau, i = 0 \\ \xi_k^\tau(c_{k-1}^\tau|\psi_k^\tau, Y_s^{k-1})\beta_{k,i}^\tau \frac{p_{k,i}(c_{k-1}^\tau)}{p_{k,i}}, i \neq 0 \end{cases} \quad (21)$$

where $i = 0$ represents the hypothesis that none of the validated measurements originate from the target.

Then updated by using Kalman filter with measurement $y_{s,i}(k)$,

$$p(x_k^\tau|c_k^\tau, \psi_k^\tau, Y_s^k) = N(x_k^\tau; \hat{x}_{k|k}^\tau(c_k^\tau), \hat{P}_{k|k}^\tau(c_k^\tau)), \quad (22)$$

$$[\hat{x}_{k|k}^\tau(c_k^\tau), \hat{P}_{k|k}^\tau(c_k^\tau)] = KF_U[y_{s,i}(k), \hat{x}_{k|k-1}^\tau(c_{k-1}^\tau), \hat{P}_{k|k-1}^\tau(c_{k-1}^\tau), H_s, R_{s,k}], \quad (23)$$

where the KF_U stands for the Kalman filter update.

3.1.6 Component Management

Without component control, the number of track components increase exponentially in time, thus efficient management is crucial. The component management in JITS involves two parts: merging and pruning.

Component merging deals with track components with similar features, for which we applied a relatively short retained track component history as similarity measure. Component pruning is to remove those track components with low probability, so that track is always composed by more significant components (Challa et al., 2011).

3.2 Multi-sensor JITS

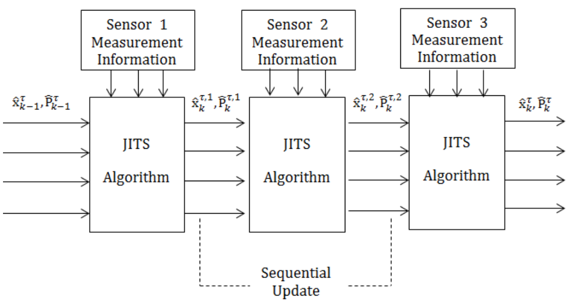

Figure 1: Sequential implementation of JITS algorithm.

Difficulties for generalizing a single sensor JITS filter to a multi-sensor case are complicated. Since the sensors are synchronized and the measurement noises across the sensors are uncorrelated, the two update schemes are available: parallel processing and sequential update. In the parallel processing, the sensor measurements are transmitted to a fusion center and converted with respect to a reference coordinate system. The target state is updated by applying the JITS algorithm to the measurements in the fusion center. However, it requires a solution for multiple detection target tracking as the measurements contain multiple detections of a target gathered from different sensors(Habtemariam et al., 2013). To realize the parallel processing, sensors need to be synchronized and heavy data traffic to the fusion center is expected. The rigorous formulas are too complicated for practical usage (Smith and Singh, 2006)(Bar-Shalom et al., 2011).

In the sequential update, the target state is updated with the data from one sensor at a time. The essence of sequential update for multi-sensor JITS algorithm is operating single sensor JITS algorithm iteratively, so that the computational complexity is significantly decreased. Sensors do not have to be synchronized. In (Pao and Frei, 1995), it suggests that the sequential update gives superior tracking performance than the parallel process when data association is considered. This is primarily due to fact that better filtered estimates are available after processing each sensor's data. Measurement selection procedure filters out many measurements that are not statistically significant. And as a consequence, the sequential update scheme, which is computationally tractable and with better tracking performance, is applied for solving multi-sensor problem in this paper.

The way of implementing the sequential update for the MS-JITS algorithm is to process the measurements from each sensor in succession (Bar-Shalom and Li, 1995), using single-sensor JITS algorithm as shown in Figure 1. The outputs of the sth ($s \leq S-1$) sensor are regarded as intermediate track stated pdf of target τ, state vector $\hat{x}_k^{\tau,s}$ and the covariance $\hat{P}_k^{\tau,s}$. Using intermediate result in place of the predicted state, then proceed to the next sensor for updating, and finally cumulates the ultimate results $\hat{x}_k^\tau$ and $\hat{P}_k^\tau$ in the last sensor.

$$[\hat{x}_{k|k}^{\tau,s}, \hat{P}_{k|k}^{\tau,s}] = JITS[\hat{x}_{k|k}^{\tau,s-1}, \hat{P}_{k|k}^{\tau,s-1}, y_s(k)], \quad s = 1, 2, \ldots S-1 \quad (24)$$

$$[\hat{x}_{k|k}^\tau, \hat{P}_{k|k}^\tau] = JITS[\hat{x}_{k|k}^{\tau,s-1}, \hat{P}_{k|k}^{\tau,s-1}, y_s(k)], s = S \quad (25)$$

where $JITS$ denotes the single-sensor JITS algorithm described in Section III-A.

The posterior probability of target existence in multi-sensor target tracking application should give due consideration to validated measurements in every sensor. And it can be calculated by updating target existence probability iteratively using the joint multi-target data association method (19) described in Section III.

4 TARGET EXISTENCE PROBABILITY IN S-D ASSIGNMENT

Unlike the S-D assignment, all the tracks in MS-JITS algorithm are automatically managed (confirmation or termination) by probability of target existence. For a fair result comparison, it stands to reason that this method should be applied to S-D assignment.

Every element (including dummy measurement) in a composite measurement contributes to the posterior probability of target existence. We use the measurement from the first sensor to derive an intermediate probability of target existence, using it in place of the predicted probability for the next sensor. The posterior probability of target existence is given by

$$P(\psi_k^\tau|Y_s^k) = \frac{(\prod_{s=1}^{S}\Lambda_k^s)P(\psi_k^\tau|Y_s^{k-1})}{1-(1-\prod_{s=1}^{S}\Lambda_k^s)P(\psi_k^\tau|Y_s^{k-1})}, \quad (26)$$

where Λ_k^s is the measurement likelihood ratio in sensor s, denoted by

$$\Lambda_k^s = \begin{cases} 1-P_{Ds}P_G + P_{Ds}P_G\frac{p_{k,i}^s}{\rho_{s,i}(k)} & \text{if } \kappa_{s,i}^\tau = 1 \\ 1-P_{Ds}P_G & \text{otherwise} \end{cases}. \quad (27)$$

The binary variable $\kappa_{s,i}^\tau = 1$ indicates that $y_{s,i}(k)$ is the measurement of target τ. The measurement likelihood $p_{k,i}^s$ (indexed by sensor s) is given by

$$p_{k,i}^s = \begin{cases} N(y_{s,i}(k);\hat{x}_{k|k-1}^\tau, S_k^\tau), s=1 \\ N(y_{s,i}(k);\hat{x}_{k|k}^{\tau,s-1}, S_k^\tau), s=2,3,\ldots,S \end{cases} \quad (28)$$

where S_k^τ is the predicted measurement covariance of the standard Kalman filter. The posterior probability of target existence is given in (26), for the sake of brevity, the superscript τ is omitted for Λ_k^s, $p_{k,i}^s$. The derivation is described in appendix.

5 SIMULATION

This section provides a comparison between the MS-JITS algorithm and the S-D assignment algorithm $(S=3)$ by simulating the scenarios with varying clutter measurement densities and target detection probabilities respectively.

Three targets are observed in clutter on a two dimensional surveillance region with 1000 m in length and 400 m in width. The clutter measurements are assumed to be uniformly distributed. Figure 2 shows a representative trial over the entire Monte Carlo trials. The illustrated arrows and crosses are the trajectories for each target and clutter measurements respectively.

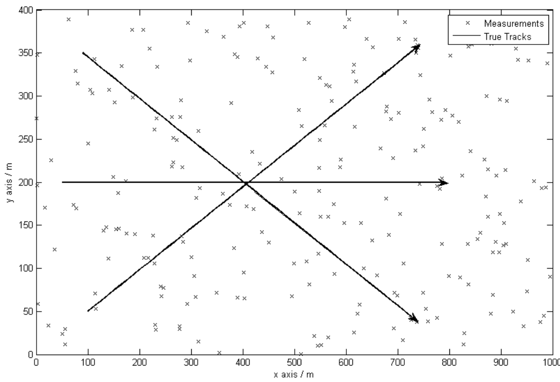

Figure 2: Surveillance region.

The selection window size is given with gating probability $P_G = 0.99$. The measurements noise is assumed to be Gaussian distributed with zero-mean and standard deviation $\sigma_s = 5m$. The target state vector consists of

$$x_k^\tau = [x, y, \dot{x}, \dot{y}]^T \quad (29)$$

with transition matrix F_k^τ

$$F_k^\tau = \begin{bmatrix} 1 & 0 & T & 0 \\ 0 & 1 & 0 & T \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \quad (30)$$

The velocity vectors of the targets are $[13m/s, -6.25m/s]^T$, $[15m/s, 0m/s]^T$, $[13m/s, 6.25m/s]^T$ respectively. Sampling time is T = 1s. Target maximum velocity is $v_{max} = 25m/s$. The process noise is zero-mean with covariance

$$Q_k = q\begin{bmatrix} \frac{T^4}{4} & 0 & \frac{T^3}{2} & 0 \\ 0 & \frac{T^4}{4} & 0 & \frac{T^3}{2} \\ \frac{T^3}{2} & 0 & T^2 & 0 \\ 0 & \frac{T^3}{2} & 0 & T^2 \end{bmatrix}, \quad (31)$$

where $q = 0.75\ m^2/s^4$.

Tracks are initialized automatically using two-point differencing (Challa et al., 2011), and a constant value of probability of target existence is assigned to each newly initialized track. The probability of target existence is recursively updated in subsequent scans. If the probability exceeds the confirmation threshold, the corresponding track is confirmed. And the track will be terminated if the probability falls below the termination threshold. The Markov transitional probability is $\triangle_{1,1} = 0.98$.

Three scenarios of this experiment are considered, each with a different target detection probability and clutter measurement density: A. clutter measurement density $\rho_{s,i}(k) = 1\times10^{-4}$ and low target detection probability $P_{Ds} = 0.6$; B. clutter measurement density $\rho_{s,i}(k) = 2\times10^{-4}$ and high target detection probability $P_{Ds} = 0.9$; C. clutter measurement density $\rho_{s,i}(k) = 2 \times 10^{-4}$ and low target detection probability $P_{Ds} = 0.6$. The Monte Carlo simulations for each scenario consist of 200 trails. MS-JITS is updated by a Gaussian mixture of component state pdf such that most of the measurements in the validation window contributes to the update. The estimator based on S-D assignment chooses the composite measurement by evaluating maximum likelihood and the global nearest neighbor using 2-D dynamic assignment algorithm (Popp et al., 2001).

5.1 False Track Discrimination

The false track discrimination results are shown in Figures 3, 4 and 5. The graphs show the percentage of confirmed true tracks of MS-JITS and S-D assignment over time for each scenario.

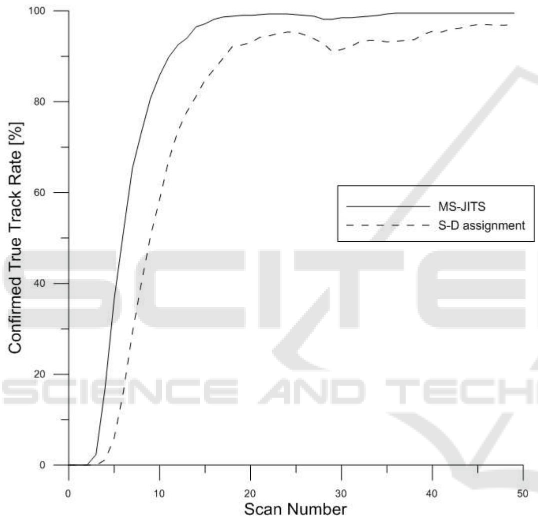

Figure 3: Confirmed true track rate-Scenario A.

MS-JITS always confirms track faster than S-D assignment. These figures show that MS-JITS has superior advantage of true track confirmation, which is apparent in dense clutter density and low detection probability environment.

In the simulation, the false alarm rates of the MS-JITS algorithm of the 3 scenarios are 3.3%, 0.86%, 2.5% compared with the S-D assignment algorithm of 25%, 11.67%, 32.5%. The inequality of the result is due to the fact that the essences of each algorithm are distinct. In the target state update part, the MS-JITS algorithm processes all the validated measurement, whereas the S-D assignment algorithm processes only one composite measurement that marginal target information is utilized. If efforts were made to minimize the gap of false alarm rates, the time duration that confirmation procedure required would decrease and the total confirmed true track rate would

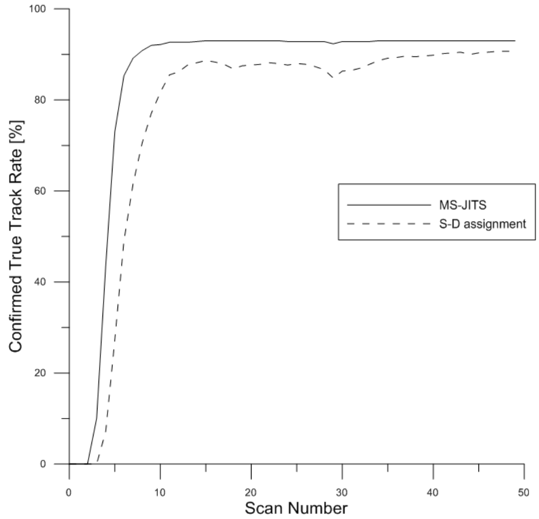

Figure 4: Confirmed true track rate-Scenario B.

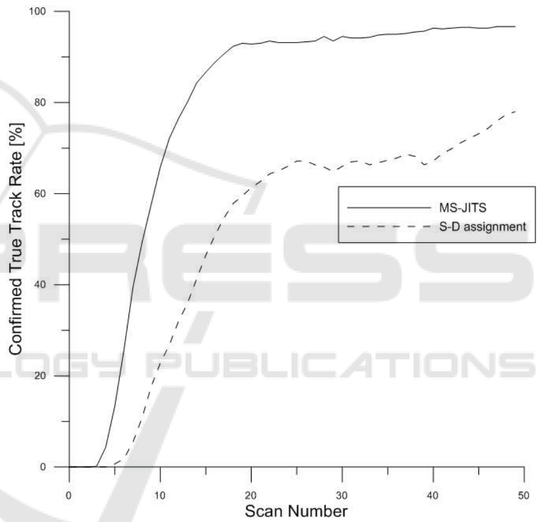

Figure 5: Confirmed true track rate-Scenario C.

increase for the MS-JITS algorithm.

5.2 Retention Test

Retention test is designed for observing the destination of every confirmed true track. Then we can evaluate the efficiency of an algorithm. The following information is obtained by tagging the confirmed true tracks at scan 15 and checking them again at scan 40 (Mušicki, 2006):

- nCases: Total number of tracks that following a target at scan 15;
- nOK: Total number of tracks that still following the original target at scan 40;
- nLost: Total number of tracks that not following any target at scan 40;

- nSwitched: Total number of tracks that end up following a different target at scan 40;
- nMerged: Total number of tracks disappeared due to merging between scan 15 and 40.

The target retention tests are listed in Table 1-3.

Table 1: Retention Statistics-Scenario A.

	MS-JITS	S-D
nCases	583	509
nOK	559	392
nLost	5	42
nSwitched	16	72
nMerged	3	3

Table 2: Retention Statistics-Scenario B.

	MS-JITS	S-D
nCases	561	541
nOK	544	377
nLost	1	85
nSwitched	16	70
nMerged	0	9

Table 3: Retention Statistics-Scenario C

	MS-JITS	S-D
nCases	520	279
nOK	481	137
nLost	11	99
nSwitched	23	42
nMerged	5	1

The results show that track retention capabilities of MS-JITS, with success rates (nOK / nCases) of 96%, 97%, 93%, is better than S-D assignment, with success rates of 77%, 70%, 49%. The percentage of false alarms is apparently decreased. In addition, the nLost and nSwitched number is much reduced in MS-JITS. That displays the track trajectory maintains well during propagation.

5.3 RMS error

Position root mean square (RMS) error is a criterion for evaluating track trajectory accuracy, and for simplicity each scenario shows the representative result only. The results are presented in Figures 6, 7 and 8, which show the RMS errors of MS-JITS and S-D assignment for different scenarios. The salients in every figure are shown around scan 25, which is the time that the close encounter between the tracks occurred. Due to the incorrect measurement-to-track association, some measurements are allocated to the unrelated track. Therefore, the track trajectory estimation accuracy degenerates. Obviously, MS-JITS always provides better trajectory estimate than S-D assignment.

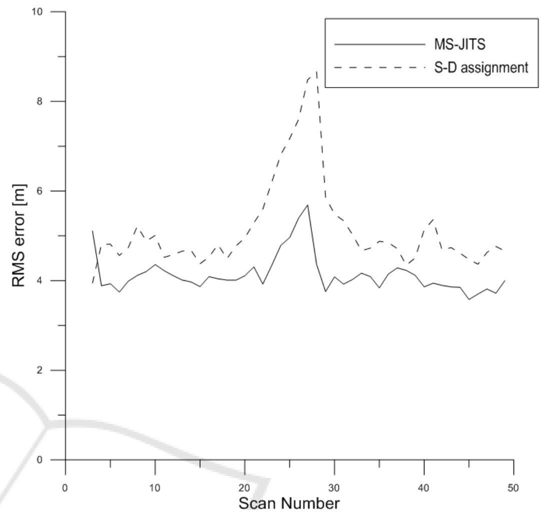

Figure 6: RMS error in position-Scenario A.

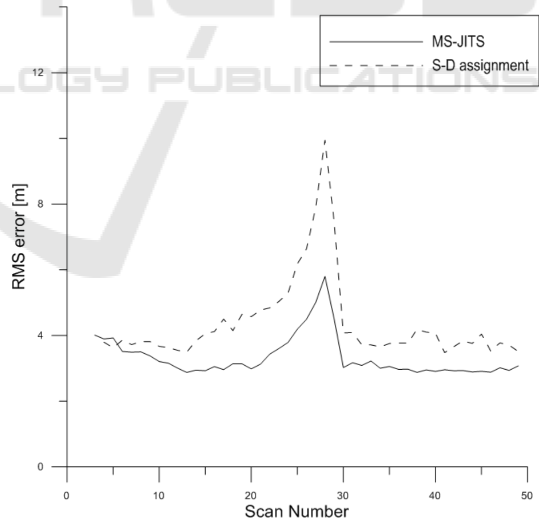

Figure 7: RMS error in position-Scenario B.

5.4 Computation Time

The computation time of both algorithms for each scenario are listed in Table 4.

It suggests that the MS-JITS algorithm requires more computation time, which provides a trade off between the computation time and tracking perfor-

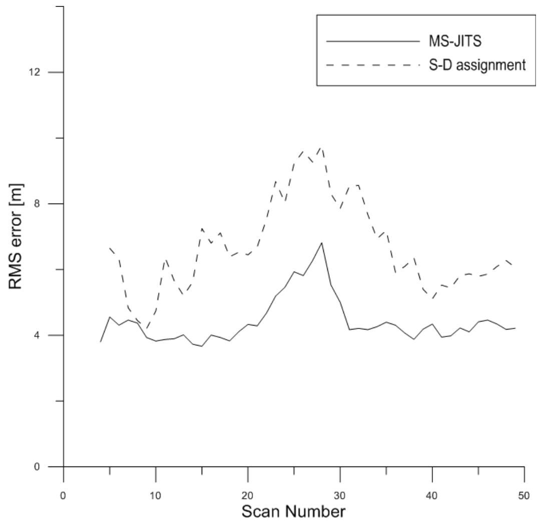

Figure 8: RMS error in position-Scenario C.

mance in practical applications. Though the computational load of MS-JITS is heavier, for some situations where track trajectories are located in near vicinity, MS-JITS is more preferable. A compromise of the computation time and estimation accuracy can be achieved by adopting an iterative implementation (Song et al., 2015b) for the JITS filter. The computational requirements and estimation accuracy can be classified into several levels by adjusting the number of iterations. The iterations start with single target tracking algorithm and each subsequent iteration improves the approximation towards the optimal multi-target solution within a finite number of iterations. The application of iterative JITS filter for multisensor system remains further exploration.

Table 4: Computation Time (sec.).

	MS-JITS	S-D
Scenario A	262	82
Scenario B	6691	415
Scenario C	4063	456

6 CONCLUSIONS

In this paper, we have presented a JITS based multisensor tracking algorithm, which is capable of false track discrimination by using the probability of target existence as a track quality measure. And the S-D assignment based estimator is enhanced by incorporating the probability of target existence for false track discrimination.

The S-D assignment algorithm may be effective in applications where computational efficiency is more important. But in situations where clutter measurement density is dense or target detection probability is low, the composite measurements become seriously contaminated or mostly consisted by dummy measurements. And as a consequence, the false alarm rate rises and track trajectory accuracy decreases. In contrast, the MS-JITS algorithm compensates the drawbacks at the cost of computational load. The MS-JITS algorithm processes measurement information more comprehensively with a more strict mechanism to distinguish the true tracks and false alarms. In addition, the retention test proves that the MS-JITS algorithm has stronger robustness. Thus, for applications where track trajectory accuracy and track maintenance are preferable, the MS-JITS algorithm offers an attractive alternative.

ACKNOWLEDGEMENTS

This paper was supported by the LIG-Nex1 Co., Ltd. under the contract LIGNEX1-2015-0108(00).

REFERENCES

Bar-Shalom, Y. and Li, X. (1995). *Multitarget-multisensor tracking: principles and techniques*. YBS Publishing.

Bar-Shalom, Y., Willett, P. K., and Tian, X. (2011). *Tracking and Data Fusion*. YBS Publishing.

Blackman, S. (2004). Multiple hypothesis tracking for multiple target tracking. *IEEE Aerospace and Electronic Systems Magazine*, 19(1):5–18.

Blackman, S. and Popoli, R. (1999). *Design and Analysis of Modern Tracking Systems*. Artech House.

Challa, S., Morelande, M. R., Mušicki, D., and Evans, R. J. (2011). *Fundamentals of object tracking*. Cambridge University Press.

Deb, S., Yeddanapudi, M., Pattipati, K., and Bar-Shalom, Y. (1997). A generalized s-d assignment algorithm for multisensor-multitarget state estimation. *IEEE Transactions on Aerospace and Electronic Systems*, 33(2):523–538.

Frei, C. W. and Pao, L. Y. (1998). Alternatives to monte-carlo simulation evaluations of two multisensor fusion algorithms. *Automatica*, 34(1):103–110.

Gilbertand, K. C. and Hofstra, R. B. (1988). Multidimensional assignment problems. *Decision Sciences*, 19(2):306–321.

Habtemariam, B., Tharmarasa, R., Thayaparan, T., Mallick, M., and Kirubarajan, T. (2013). A multiple-detection joint probabilistic data association filter. *IEEE Journal of Selected Topics in Signal Processing*, 7(3):461–471.

Kirubarajan, T., Wang, H., Bar-Shalom, Y., and Pattipati, K. R. (2001). Efficient multisensor fusion using multidimensional data association. *IEEE Transactions on Aerospace and Electronic Systems*, 37(2):386–400.

Mušicki, D. (2006). Track score and target existence. In *9th Int. Conf. on Information Fusion*, pages 1–7.

Mušicki, D., Evans, R., and Stanković, S. (1994). Integrated Probabilistic Data Association (IPDA). *IEEE Trans. Automatic Control*, 39(6):1237–1241.

Mušicki, D. and Evans, R. J. (2009). Multiscan multitarget tracking in clutter with integrated track splitting filter. *IEEE Transactions on Aerospace and Electronic Systems*, 45(4):1432–1447.

Mušicki, D., Scala, B. F., and Evans, R. J. (2007). The integrated track splitting filter efficient multi-scan single target tracking in clutter. *IEEE Transactions on Aerospace and Electronic Systems*, 43(4):1409–1425.

Pao, L. Y. and Frei, C. W. (1995). A comparison of parallel and sequential implementations of a multisensor multitarget tracking algorithm. In *American Control Conference, Proceedings*, volume 3, pages 1683–1687.

Pattipati, K. R., Deb, S., Bar-Shalom, Y., and Washburn, R. B. (1992). A new relaxation algorithm and passive sensor data association. *IEEE Transactions on Aerospace and Electronic Systems*, 37(2):198–213.

Popp, R. L., Pattipati, K. R., and Bar-Shalom, Y. (2001). m-best s-d assignment algorithm with application to multitarget tracking. *IEEE Transactions on Aerospace and Electronic Systems*, 37(1):22–39.

Reid, D. (1979). An algorithm for tracking multiple targets. *IEEE Transactions Automatic Control*, 24(6):843–854.

Shi, Y. F., Song, T. L., and Lee, J. H. (2015a). Multitarget tracking in clutter using a high pulse repetition frequency radar. *IET, Radar, Sonar and Navigation*, 9(8):1047–1054.

Shi, Y. F., Song, T. L., and Mušcki, D. (2015b). Target tracking in clutter using a high pulse repetition frequency radar. *IET, Radar, Sonar and Navigation*, 9(3):299–307.

Smith, D. and Singh, S. (2006). Approaches to multisensor data fusion in target tracking: a survey. *IEEE Transactions on Knowledge and Data Engineering*, 18(12):1696–1710.

Song, T. L., Kim, H., and Mušicki, D. (2015a). Distributed (nonlinear) target tracking in clutter. *IEEE Transactions on Aerospace and Electronic Systems*, 51(1):654–668.

Song, T. L., Kim, H. W., and Musicki, D. (2015b). Iterative joint integrated probabilistic data association for multitarget tracking. *IEEE Transactions on Aerospace and Electronic Systems*, 51(1):642–653.

Song, T. L., Mušcki, D., and Kim, Y. (2015c). Multitarget tracking with state dependent detection. *IET, Radar, Sonar and Navigation*, 9(1):10–18.

Song, T. L., Mušicki, D., Kim, H. W., and Govaers, F. (2013). Gaussian mixture tracking: Mht and its comparison. In *Sensor Data Fusion: Trends, Solutions, Applications (SDF)*, pages 1–6.

Xiong, N. and Svensson, P. (2002). Multi-sensor management for information fusion: issues and approaches. In *5th Int. Conf. on Information Fusion*, volume 3, pages 163–186.

APPENDIX

Sequential Likelihood Ratio

This section is used to describe the posterior probability of target existence in MS-JITS algorithm. The measurements from the first sensor are used to compute an intermediate posterior probability of target existence $P(\psi_k^{\tau,1}|Y_s^k)$, with corresponding measurement likelihood ratio Λ_k^1 denoted by

$$P(\psi_k^{\tau,1}|Y_s^k) = \frac{\Lambda_k^1 P(\psi_k^{\tau}|Y_s^{k-1})}{1-(1-\Lambda_k^1)P(\psi_k^{\tau}|Y_s^{k-1})}. \qquad (32)$$

After gating the measurements in next sensor, target existence probability is updated,

$$P(\psi_k^{\tau,2}|Y_s^k) = \frac{\Lambda_k^2 P(\psi_k^{\tau,1}|Y_s^{k-1})}{1-(1-\Lambda_k^2)P(\psi_k^{\tau,1}|Y_s^{k-1})}. \qquad (33)$$

Substituting the $P(\psi_k^{\tau,1}|Y_s^{k-1})$ by equation (32) yields,

$$\begin{aligned} P(\psi_k^{\tau,2}|Y_s^k) &= \frac{\Lambda_k^2 \frac{\Lambda_k^1 P(\psi_k^{\tau}|Y_s^{k-1})}{1-(1-\Lambda_k^1)P(\psi_k^{\tau}|Y_s^{k-1})}}{1-(1-\Lambda_k^2)\frac{\Lambda_k^1 P(\psi_k^{\tau}|Y_s^{k-1})}{1-(1-\Lambda_k^1)P(\psi_k^{\tau}|Y_s^{k-1})}} \\ &= \frac{\Lambda_k^1\Lambda_k^2 P(\psi_k^{\tau}|Y_s^{k-1})}{1-(1-\Lambda_k^1\Lambda_k^2)P(\psi_k^{\tau}|Y_s^{k-1})}. \qquad (34) \end{aligned}$$

Similarly, we can obtain

$$P(\psi_k^{\tau,3}|Y_s^k) = \frac{\Lambda_k^1\Lambda_k^2\Lambda_k^3 P(\psi_k^{\tau}|Y_s^{k-1})}{1-(1-\Lambda_k^1\Lambda_k^2\Lambda_k^3)P(\psi_k^{\tau}|Y_s^{k-1})}. \qquad (35)$$

Finally, up to sensor S yields

$$P(\psi_k^{\tau}|Y_s^k) = \frac{(\prod_{s=1}^S \Lambda_k^s)P(\psi_k^{\tau}|Y_s^{k-1})}{1-(1-\prod_{s=1}^S \Lambda_k^s)P(\psi_k^{\tau}|Y_s^{k-1})}. \qquad (36)$$