Rauch-Tung-Striebel Smoothing Linear Multi-Target Tracking in Clutter

S. A. Memon*1, W. -G. Kim1, M. Park1, and M. Attique2

1Department of Defense Systems Engineering, Sejong University, 209 Neungdong-ro, Gwangjin-gu, Seoul, Republic of Korea
2Department of Software, Sejong University, 209 Neungdong-ro, Gwangjin-gu, Seoul, Republic of Korea

Corresponding author: S. A. Memon (e-mail: sufyana@sejong.ac.kr).

“This work was supported by the faculty research fund of Sejong University (2021-2022) and the basic science research program of the National Research Foundation of Korea funded by Ministry of Education under Grant 2020R1G1A1013221.”

ABSTRACT Most conventional data association based multi-target tracking (MTT) algorithms typically suffer from intractable computational complexities and could not perform in an environment where large number of closely spaced multiple targets move across each other in clutters. Unlike to the existing MTT systems, the linear multi-target (LM) algorithm modifies the measurement detection followed by neighbored tracks as a clutter, hence, it updates the track without the influence of other tracks. Thus, LM technique is a computationally efficient algorithm that allows the multi-target system to play like a single target tracking algorithm. Smoothing maximizes the state estimation accuracy and reduces estimation error based on future scan measurement. However, only few research paper focused on the LM algorithm without utilizing the benefits of the smoothing. This paper presents Rauch-Tung-Striebel Smoothing in the linear multi-target based on integrated probabilistic data association (RTS-LMIPDA). The RTS-LMIPDA algorithm fuses forward and backward LM track predictions to obtain the smoothing prediction which is required to calculate the smoothing multi-target state estimates in the forward track. Numerical analysis is presented to illustrate the estimation accuracy and false track discrimination (FTD) performances of RTS-LMIPDA in comparison to the existing MTT algorithms using the simulations.

INDEX TERMS data association, estimation, false-track discrimination (FTD), linear multi-target tracking, smoothing, target existence probability (TEP).

I. INTRODUCTION

A radar detector returns uncertain measurements from multi-targets and various random sources to a multi-target tracking system. The measurements that originate from unknown object sources (e.g., reflections from terrain, target thermal noise, and various weather conditions such as clouds) are referred to as clutter measurements. In a radar surveillance region, the trajectory behavior, existence, and number of measurements that may have occurred from one of several targets are unknown. In addition, each feasible target measurement has a low detection probability $P_d$, indicating that target tracking algorithms do not have prior information on the radar measurements. As a result, a target tracking algorithm initializes and updates the false (clutter) tracks as well as the true (target) tracks using the measurements received from a radar detector in each time scan. In this complex cluttered environment, the tracking algorithms employ false track discrimination (FTD) to identify and discriminate the false tracks [1]. FTD utilizes the target existence probability (TEP) as a track quality measure to confirm the true track and terminate the false track [2].

Conventional multi-target tracking (MTT) algorithms use the joint data association technique, which deals with target and clutter measurements by associating the measurements to the tracks in a probabilistic way. In a multi-target situation, the tracks may have joint (shared) measurements because the measurements may have originated from the potential target being followed by neighbor tracks. Therefore, the MTT algorithms such as joint probabilistic data association (JPDA) [3], joint integrated PDA (JIPDA) [4][5], joint create all feasible joint measurement-to-track association assignment hypotheses and recursively calculate their joint a-posteriori probabilities in each scan. In this situation, a cluster is formed that groups the shared tracks with similar measurements of detection. The number of feasible joint measurement events grows combinatorically in terms of the number of tracks and measurements.
involved in a cluster. Thus, the numerical complexities of the JPDA and JIPDA algorithms often exceed the available computational resources, especially when there is a large number of targets in the cluttered environment. The joint data association is also used for multi-scan, multi-target tracking algorithms, such as joint integrated track splitting [6][7]. These multi-scan MTT algorithms are even more complex filters, which form an exponentially growing number of track components. Similarly, a multi-scan multiple hypothesis tracker [8][9] obtains the exponentially increasing number of multiple hypotheses by enumerating all feasible track-to-measurement association hypothesis. In addition to this complexity, each MHT hypothesis is followed by a global association which considering all tracks and measurements over a span of scans. Thus, MHT consumes more computation time than that of joint data association algorithms. The linear multi-target based on integrated probabilistic data association (LM-IPDA) method was developed to overcome these intractable computational complexities [10][11]. LM-IPDA bypasses the enumeration of the feasible joint measurement events of the multi-targets using the modified clutter measurement density, which treats the neighbor tracks as a source of clutter. Thus, a track is updated by considering other target detection measurements as a modified clutter measurement being followed by neighbor tracks. Compared to joint data association algorithms [3][4][5][6][7], LM-IPDA is more computationally feasible and efficient for multi-target tracking in clutter. Subsequently, the linear multi-target algorithm was extended with an integrated track splitting filter [7]. The aforementioned algorithms use the probability of target existence as a track quality measurement for FTD, except for [3], [8], and [9].

Generally, a non-smoothing algorithm generates higher estimation errors due to the availability of the sensor measurement up to and including time scan k. This often limits the accuracy in the trajectory estimation and track quality measure of FTD. Therefore, a smoothing Rauch-Tung-Striebel (RTS) algorithm was developed, which evaluates the track state estimate in past scans based on the measurements available beyond time scan k [12]. In the recent development of single target tracking smoothing algorithms, researchers smoothed the track state estimate by fusing each individual forward track state prediction with multi-track backward predictions that were falling in the validation gate of a forward track [13][14][15]. The multi-track backward predictions were recursively estimated in each scan using the sensor measurements, starting from the last scan N and ending at the first scan k of the smoothing interval. This made the smoothing algorithm efficient for target tracking but increased the smoothing time delay due to the time consumed in the initialization and estimation of the backward multi-tracks. These algorithms were extended for smoothing MTT by utilizing a joint data association algorithm, such as fixed interval smoothing based on JIPDA (FlsJIPDA) [16]. Recently, fixed interval smoothing was utilized in multi-scan, multi-target joint integrated track splitting to calculate the smoothing a-posteriori probabilities of the multiple track components associated in a cluster for smoothing state estimation [17]. RTS smoothing equations in a JIPDA algorithm (sJIPDA) use the forward track validated measurements to obtain a backward track prediction [18]. Thus, each updated backward track corresponds to each existing forward track for fusion. This makes the RTS smoothing algorithms, such as [18] and [19], optimal and computationally cost efficient. All smoothing algorithms discussed above use the smoothing probability of target existence for the FTD evaluation. However, the former MTT smoothing algorithms, such as the smoothing JPDA [20], smoothing multiple hypothesis tracker [21], and smoothing probability hypothesis density filter [22][23] did not produce FTD results.

This study utilizes RTS equations for smoothing a linear multi-target based on IPDA in a fixed-interval (RTS-LMIPDA). RTS formulas are used in the LM-IPDA to improve the estimation accuracy and FTD track quality measurement with a limited smoothing delay. Table 1 compares the contributions and motivations of the RTS-LMIPDA algorithm to the existing algorithms.

| # | RTS-LMIPDA | sJIPDA | FlsJIPDA |
|---|---|---|---|
| 1 | forward multi-tracks are estimated in each scan k (from scan k to N) | forward multi-tracks cluster are estimated in each scan k (from scan k to N) | backward multi-tracks cluster are estimated in each scan k (from scan N to k) |
| 2 | backward prediction is obtained using RTS in each scan k (from scan N to k) | backward joint predictions are obtained using RTS and cluster measurements in each scan k (from scan N to k) | forward joint predictions are initialized by sensor measurements in each scan k (from scan k to N) |
| 3 | forward and backward predictions are fused in each scan k (each individual backward-forward track fusion). | forward and backward joint predictions are fused in each scan k. | a forward track is fused with predicted multiple backward tracks to obtain a validation gate in k. |
| 4 | forward multi-tracks are smoothed in each scan using smoothing track measurements | forward multi-tracks are smoothed in each scan using smoothing cluster measurements | Track state output is smoothed to obtain forward estimates by using smoothing data association probabilities in k. |

RTS-LMIPDA uses overlapping fixed-intervals, which have a length of N – k + 1 scans, where N indicates the last
scan and \( k \) indicates the first scan of an interval. The forward tracks are initialized and updated using the Kalman Filter [24] recursively in each scan of the interval. When a forward track arrives at scan \( k = N \), a backward track is initialized using a forward track validated measurement. Similar to sJIPDA, the backward prediction is obtained from the forward and RTS estimates. Finally, the RTS-LMIPDA algorithm fuses forward and backward predictions to obtain the smoothing linear multi-target state estimations. The a-posteriori smoothing probability of target existence is also calculated in each scan for FTD evaluation.

Section II discusses the target and sensor measurement model. The RTS smoothing based on LM-IPDA is examined in Section III. The estimation statistic results of RTS-LMIPDA are verified using simulation in Section IV, followed by a conclusion in Section V.

II. TARGET PROPAGATION AND MEASUREMENT MODELS

It is assumed that a track has a hybrid state consisting of a target trajectory state \( x_k^T \) and target existence event \( \chi_k^T \), where \( T \) indicates an index of targets and tracks. Without lost generality, the target existence is a random event which generates one measurement in each scan and is always detectable with the probability \( P_D \). The target model obeys the Markov chain one model [2] which uses state transition probability to maintain and update the existing track. The state transition probability is the probability that the target exists in scan \( k \), provided that it existed in scan \( k-1 \), which is expressed as:

\[
\alpha_{t,1} = P\{\chi_k^T | \chi_{k-1}^T\} \equiv 1 - \frac{\Delta T_{k,k-1}}{T_s},
\]

where \( T_s \) denotes the average duration of the target existence and \( \Delta T_{k,k-1} \) indicates the time between scan \( k-1 \) and \( k \). It is assumed that each target state is a four-dimensional state vector, such as two-dimensional position and velocity vectors. Each target trajectory state \( x_k^T \) measured by a sensor in scan \( k-1 \) propagates to scan \( k \) using a state transition model \( F \) in the target dynamic equation, expressed as:

\[
x_k^T = F x_{k-1}^T + v_{k-1}^T,
\]

where the target trajectory state is correlated by a zero-mean white Gaussian process noise \( v_{k-1} \) that has a known covariance \( Q \). The \( Q \) and \( F \) expressions are derived in [1] and [12], respectively, and are expressed as:

\[
Q = q \begin{bmatrix} 0.24T_{12} & 0.57T_{12} \\ 0.57T_{12} & T^2_{12} \end{bmatrix}
\]

and

\[
F = \begin{bmatrix} I_{2x2} & T_{2x2} \\ O_{2x2} & I_{2x2} \end{bmatrix},
\]

where \( q \) represents the white Gaussian acceleration uncertainty received between two scan times \( T \), \( I_{2x2} \) is the \( 2 \times 2 \) identity matrix, and \( O_{2x2} \) is the \( 2 \times 2 \) zeros matrix.

The sensor observes each target in each scan \( k \) and if a \( a^{th} \) target exists and is detected with the probability \( P_D \), then it generates the position measurement \( y_k^T \) using:

\[
y_k^T = H_k x_k^T + w_k^T,
\]

where \( H_k = [I_{2x2}, O_{2x2}] \), which calculates the target state position measurement. \( w_k \) is a white Gaussian measurement noise with a zero-mean and known covariance matrix \( R_k \). In addition, a random number of clutter measurements are observed and detected using a sensor that follows a non-homogeneous Poisson distribution [25]. The density of clutter measurements is defined by the number of sensor measurements received from each scan at the \( x \) and \( y \) axes of the surveillance region, and it is assumed to be known in this study.

\( Y_k \) represents the cardinality of a set of \( M_k \) sensor measurements received in scan \( k \). The \( m^{th} \) measurement in \( Y_k \) is denoted by \( Y_{k,m} \) and has a constant clutter measurement density, \( \rho_k \equiv \rho(Y_{k,m}) \). \( \{\Phi\} = \Phi_1, \Phi_2, \ldots, \Phi_N \) denotes the sequence of the consecutive measurement times in a smoothing interval, which has a length of \( N \) that denotes the last scan index in a smoothing interval. For example, \( Y^{\Phi} = \{Y_{\Phi_1}, Y_{\Phi_2}, \ldots, Y_{\Phi_N}\} \) is the measurement interval that implies the sequence of consecutive sets of measurements.

III. RAUCH-TUNG-STRIEBEL SMOOTHING LINEAR MULTI-TARGET BASED ON IPDA (RTS-LMIPDA)

This paper utilizes RTS equations [12] in the LMIPDA algorithm to improve the estimation accuracy and FTD performance in comparison to other existing tracking algorithms for tracking closely moving multi-targets in clutter. Multiple forward tracks are initialized using a sensor measurement set \( Y_k \) received in each scan \( k \), assuming an overlapped fixed interval smoothing based on \( Y_k \), where \( N \geq k \). In addition, forward LMIPDA (f-LMIPDA) is used to estimate all existing tracks. One backward track is initialized and updated using the forward track validated measurement \( y_{1,k} \in Y_k \) for each corresponding forward track if the existing (updated) tracks survived (as determined by the updated target existence probability) and they arrived at a predetermined interval length at time \( n \). The backward LMIPDA (b-LMIPDA) track is propagated using RTS and forward estimates until the first scan \( k \) of the smoothing interval. Each corresponding forward and backward prediction are fused to obtain the smoothing prediction and innovation in each scan \( k \) moving forward in time. Subsequently, the RTS-LMIPDA track is estimated by updating the smoothing predictions based on the smoothing validation measurements \( \tilde{y}_{1,k} \in Y_k \) in each scan \( k \). The probability of target existence is smoothed using \( Y_k^\Phi \), which is utilized for track confirmation and termination in each scan \( k \).
A. GENERALIZED LMIPDA (g-LMIPDA)

Tracks are initialized based on the sensor measurements \( Y^{(t)} \) using a well-known approach referred to as two-point measurement initialization [1]. The LMIPDA (g-LMIPDA) equations were generalized with respect to the measurement time sequence to avoid the repeated use of the forward and backward filter equations. For t-LMIPDA, \( \{ \Phi \} \) was replaced by \( k \), which implies that \( \{ \Phi \} = 1: k \). For b-LMIPDA, \( \{ \Phi \} \) was replaced by \( k^* \), which implies that \( \{ \Phi \} = k: N \). For state prediction fusion, \( \{ \Phi \} \) was replaced by \( \{ \Phi \}_k \), which implies that \( k \) was excluded from \( \{ \Phi \} \), for example, \( N_k = \{1: k-1, k+1: N\} \). Lastly, for RTS-LMIPDA, \( \{ \Phi \} = 1: N, N \geq k \). The subscript \( \{ \Phi \} \) (e.g., \( \hat{x}_{k|\Phi}\bar{k} \)) denotes conditioning on the corresponding measurement set in \( Y^{(k)} \).

The LMIPDA track obtains the probability density function (PDF) of the target trajectory state \( x_k \) at scan \( k \) based on \( Y^{(k)} \), expressed as:

\[
p(x_k | Y^{(k)}) = \mathbb{N}(x_k; \hat{x}_{k|\Phi}\bar{k}, \hat{P}_{k|\Phi}\bar{k}),
\]

where \( \mathbb{N}(x; m, P) \) denotes the normalized Gaussian distribution of the trajectory state \( x_k \) with mean \( m \) and covariance \( P \). If \( y_k \) is removed from the measurement interval \( Y^{(k)} \), then \( k \not\in \{ \Phi \} \) and \( p(x_k | Y^{(k)}) \) become a state prediction PDF. For example, \( p(x_k | Y^{k-1}) \), \( p(x_k | Y^{k+1}) \), and \( p(x_k | Y^{N_k}) \) imply forward, backward, and smoothing predictions, respectively [24].

The LMIPDA track recursion starts with the hybrid state prediction, expressed as:

\[
p\left( \hat{x}_{k|\Phi}\bar{k}, x_k | Y^{k-1} \right) = p\left( \hat{x}_{k|\Phi}\bar{k} | Y^{k-1} \right) \cdot p\left( x_k | \hat{x}_{k|\Phi}\bar{k}, Y^{k-1} \right).
\]

The hybrid state prediction, with respect to the conditioning measurement sequence, is then expressed as:

\[
p\left( \hat{x}_{k|\Phi}\bar{k}, x_k | Y^{(k-1)} \right) = p\left( x_k | \hat{x}_{k|\Phi}\bar{k} \right) \cdot p\left( \hat{x}_{k|\Phi}\bar{k} | Y^{(k-1)} \right),
\]

where the \( r \)th state predicted PDF is expressed as:

\[
p\left( x_k | \hat{x}_{k|\Phi}\bar{k} \right) = \mathbb{N}(x_k; \hat{x}_{k|\Phi}\bar{k}, \hat{P}_{k|\Phi}\bar{k}),
\]

and the \( r \)th state predicted target existence probability is expressed as:

\[
p\left( \hat{x}_{k|\Phi}\bar{k} | Y^{(k-1)} \right) = \alpha_{k,i} p\left( \hat{x}_{k|\Phi}\bar{k} \right).\] (10)

Each track state prediction selects the validation measurement \( y_{k,i} \) from \( Y_k \) by applying the measurement selection condition provided in (11). If the \( r \)th target is detected with the detection probability \( P_{G(r)} \), then the \( r \)th validated measurement \( y_{k,i} \) is derived from the \( r \)th target (\( y_{k,i} \in Y_{k,i} \)). Otherwise, it is a clutter measurement, expressed as:

\[
\begin{pmatrix}
Y_{k,i} - H_k \hat{x}_{k|\Phi}\bar{k} \\
Y_{k,i} - H_k \hat{x}_{k|\Phi}\bar{k}
\end{pmatrix} \leq \delta,
\]

where superscript (·)\(^T\) denotes the transpose and \( \delta \) denotes the maximum threshold for validation measurement selection, which is calculated using the gating probability, \( P_{G(r)} \) [1]. \( P_{G(r)} = 1 - e^{-0.5\delta^2} \). Equation (11) builds an elliptical gate which surrounds the cardinality of measurements \( y_k \) with a number of selected measurements \( m_l \) within the gate volume \( V_{(\Phi)\bar{k}} \) [1]. This is expressed as:

\[
V_{(\Phi)\bar{k}} = \frac{\delta^d \pi^{d/2}}{\Gamma(1 + d/2)} \sqrt{|A|},
\]

where \( |A| \) represents the determinant of the surveillance area \( A, d \) is the dimensionality of the surveillance space, and \( \Gamma(\cdot) \) represents the Gamma function.

The LM-IPDA computes the likelihood \( l_{k,i} \) of the measurement \( y_{k,i} \) (\( i > 0 \)) with respect to a track \( r \) in:

\[
l_{k,i} = \mathcal{N} \left( y_{k,i}; H_k \hat{x}_{k|\Phi}\bar{k}, H_k \hat{P}_{k|\Phi}\bar{k} H_k^T + R_k \right) / P_{G(r)}. \]

The selected measurement \( y_{k,i} \) consists of a \( r \)th target measurement with the a-priori probability \( P_{k,i} \) [15], expressed as:

\[
P_k = P_{D(r)} P_{G(r)} P \left( \hat{x}_k | Y^{(k)} \right) \cdot \frac{l_{k,i}}{\rho_{k,i}} \sum_{i=1}^{m_l} \frac{l_{k,i}}{\rho_{k,i}},
\]

where \( i = 0 \) and \( P_{r,0} = l_{r,0} = 0 \). These a-priori probabilities of the detected measurements are mutually exclusive and exhaustive, which ensures that only one measurement is a target detection. The LMIPDA modifies the clutter measurements density being observed by a \( n \)th track in the coordinates of \( y_{k,i} \), expressed as:

\[
\mu_{k,i} = \rho_{k,i} + \sum_{\sigma \neq \pi} \frac{l_{k,i} n P_{r,\pi}}{1 - P_{r,\pi}},
\]

where \( \mu_{k,i} \) represents the modified clutter measurements density, \( \sigma \) represents the neighbored tracks (other than \( r \)), and \( t_i \) indicates the total number of tracks.

The LMIPDA algorithm uses (15) to update the \( r \)th track in scan \( k \) by calculating the track likelihood ratio and data association probabilities of the measurements \( y_{k,i} \), respectively expressed as:

\[
\lambda_{k,i} = 1 - R_{D(r)} P_{G(r)} + P_{D(r)} P_{G(r)} \sum_{i=0}^{l_{k,i}} \frac{l_{k,i}}{\mu_{k,i}}
\]

and
\[
\beta_{kj}^i = \frac{1}{\lambda_k} \left[ 1 - P_{D(\tau)} P_{G(r)}^i \right] \quad i = 0
\]

\[
P\left( \chi_k \mid Y^{(j)} \right) = \frac{\lambda_k^i P\left( \chi_k \mid Y^{(j)} \right)}{1 - \left(1 - \lambda_k^i\right) P\left( \chi_k \mid Y^{(j)} \right)}.
\]

Equation (16) estimates the predicted probability of the \(t\)th target existence in:

The LMIPDA estimates the state prediction based on \(y_{k,i}\) by utilizing the Kalman filter estimation \((KF)\) [24], expressed as:

\[
\begin{bmatrix}
\tilde{x}_{k|i}^r \n \tilde{P}_{k|i}^r
\end{bmatrix} = KF_E \left( y_{k,i}, R_k, \chi_{k|i}^r, \tilde{P}_{k|i}^r \right).
\]

Equation (17) approximates the \(t\)th state estimates obtained in (19) using one Gaussian PDF mean and covariance, respectively expressed as:

\[
\tilde{x}_{k|i}^r = \sum_{j=0}^{m_k} \beta_{k,i}^j \tilde{x}_{k|i,j}^r
\]

and

\[
\tilde{P}_{k|i}^r = \sum_{j=0}^{m_k} \beta_{k,i}^j \left( \tilde{P}_{k|i,j}^r + \tilde{x}_{k|i,j}^r \tilde{x}_{k|i,j}^r^T \right) - \tilde{x}_{k|i}^r \tilde{x}_{k|i}^r^T.
\]

Similarly, the LMIPDA obtains the recursive multi-track estimation in each scan of a smoothing interval by using (8) to (20).

B. Forward LMIPDA (f-LMIPDA)

The forward LMIPDA (f-LMIPDA) uses feasible measurement pairs from two consecutive scans (two-point difference initialization) [1] to initialize a new track in the \(k\)th scan. Each initialized track has a hybrid state and each track recursion starts with an updated state PDF \(\tilde{x}_{k-1|k-1}^r\), state covariance \(\tilde{P}_{k-1|k-1}^r\), and initial probability of target existence \(P(\chi_{k-1} \mid Y^{k-1})\) at scan \(k - 1\). The updated track state PDF propagates from scan \(k - 1\) to scan \(k\) to obtain the state prediction for scan \(k\) using the Kalman Filter propagation equation, expressed as:

\[
\begin{bmatrix}
\tilde{x}_{k|k-1}^r \n \tilde{P}_{k|k-1}^r
\end{bmatrix} = KF \left( \tilde{x}_{k-1|k-1}^r, \tilde{P}_{k-1|k-1}^r, F, Q \right).
\]

An initial target existence probability is propagated from scan \(k - 1\) to scan \(k\) using (10) with \(|\Phi| = k - 1\). The f-LMIPDA applies (8) - (21) with \(|\Phi| = k\) and \(|\Phi| = k\): \(k - 1\) to update the state prediction and propagate the state estimate using \(Y^k\) in scan \(k\). The track update obtains its state estimate \(\tilde{x}_{k|k}^r\), state covariance \(\tilde{P}_{k|k}^r\), and target existence probability \(P(\chi_{k} \mid Y^{k})\) at scan \(k\), conditioning on \(Y^k\) using (20) and (18). The f-LMIPDA algorithm propagates the updated tracks in each scan \(k\) using (21), followed by (8) - (20) to recursively calculate the multi-track estimation in each scan \(k\) until arrival at the last scan index \(n\) of a smoothing interval. The values of state prediction and state estimation must be stored in each scan \(k\) required for the use of the b-LMIPDA and RTS-LMIPDA algorithms.

B. Backward LMIPDA (b-LMIPDA)

When a forward track arrives at scan \(k = N\) (last scan index of the smoothing interval), a backward track state PDF propagates from scan \(k + 1\) to scan \(k\) using the validated measurements \(y_k\) selected by a corresponding forward track. Unlike the FisJIPDA algorithm, the backward state predictions are obtained by fusing the forward and smoothing estimates, followed by the RTS equations expressed in (22). First, the RTS equations (22) are utilized recursively to calculate the smoothing estimate \(\tilde{x}_{k|N}^{r}\) and its covariance \(\tilde{P}_{k|N}^{r}\) in each scan starting from scan \(N - 1\) and ending at the first scan \(k\) of the smoothing interval. Note that the smoothing estimate at scan \(N\) is equal to the forward estimate at scan \(k\), for example, \(\tilde{x}_{k|N}^{r} = \tilde{x}_{k=N|k=N}^{r}\).

\[
\tilde{x}_{k|N}^{r} = \tilde{x}_{k|k}^{r} + C_r^k \left( \tilde{x}_{k+l|N}^{r} - \tilde{x}_{k+l|k}^{r} \right)
\]

and

\[
\tilde{P}_{k|N}^{r} = \tilde{P}_{k|k}^{r} + C_r^k \left( \tilde{P}_{k+l|N}^{r} - \tilde{P}_{k+l|k}^{r} \right) \left( C_r^k \right)^T.
\]

where \(\tilde{x}_{k+l|k}^{r}\) and \(\tilde{P}_{k+l|k}^{r}\) denote the mean and covariance forward state propagations at scan \(k + 1\) conditioning on scan \(k\), respectively. \(\tilde{x}_{k+l|N}^{r}\) and \(\tilde{P}_{k+l|N}^{r}\) denote the mean and covariance smoothing state propagations at \(k + 1\) conditioning on the scan \(N\), respectively. \(C_r^k\) is the RTS filter gain at scan \(k\), expressed as:

\[
C_r^k = \tilde{P}_{k+l|k}^{r} \left( \tilde{P}_{k+l|k}^{r} \right)^{-1} F_k^T.
\]

The backward track state covariance \(\tilde{P}_{k+l|k}^{r}\) and mean \(\tilde{x}_{k+l|k}^{r}\) are calculated using the fusion of forward and smoothing estimates in each scan \(k\) (moving backward in time), respectively expressed as:

\[
\tilde{P}_{k+l|k}^{r} = \tilde{P}_{k+l|k}^{r} - \tilde{P}_{k+l|k}^{r} \left( \tilde{P}_{k+l|k}^{r} \right)^{-1} \tilde{x}_{k+l|k}^{r}.
\]

The b-LMIPDA state propagation begins with (23) and uses (13) - (20) with \(|\Phi| = k\); \(N\) to maintain and update the
backward probability of the target existence \( P(\mathbf{x}_k^r | Y^{k+1}) \) in each scan of the interval \([k, N]\).

\[ P(\mathbf{x}_k^r | Y^{k+1}) = P(\mathbf{x}_k^r | Y^k) + P(\mathbf{x}_k^r | Y^{k+1}) - P(\mathbf{x}_k^r | Y^k) P(\mathbf{x}_k^r | Y^{k+1}), \]

\[ \mathbf{x}^{r_\Phi}_{\mathbf{k}|\mathbf{\Phi}|} = \frac{\mathbf{P}^{r_\Phi}_{\mathbf{k}}}{\mathbf{P}^{r_{\Phi|-1}}_{\mathbf{k}}} \mathbf{x}^{r_{\Phi|-1}}_{\mathbf{k}} + \mathbf{P}^{r_{\Phi|-1}}_{\mathbf{k}} \mathbf{P}^{r_{\Phi|-1}}_{\mathbf{k}} \mathbf{P}^{r_{\Phi|-1}}_{\mathbf{k}} \mathbf{x}^{r_{\Phi|-1}}_{\mathbf{k}}. \]

\[ \mathbf{x}^{r_{\Phi|-1}}_{\mathbf{k}} = \frac{\mathbf{P}^{r_{\Phi|-1}}_{\mathbf{k}}}{\mathbf{P}^{r_{\Phi|-1}}_{\mathbf{k}}} \mathbf{x}^{r_{\Phi|-1}}_{\mathbf{k}} + \mathbf{P}^{r_{\Phi|-1}}_{\mathbf{k}} \mathbf{P}^{r_{\Phi|-1}}_{\mathbf{k}} \mathbf{P}^{r_{\Phi|-1}}_{\mathbf{k}} \mathbf{x}^{r_{\Phi|-1}}_{\mathbf{k}}. \]

\[ \mathbf{P}^{r_{\Phi|-1}}_{\mathbf{k}} = (\mathbf{P}^{r_{\Phi|-1}}_{\mathbf{k}})^{-1} \left[ (\mathbf{P}^{r_{\Phi|-1}}_{\mathbf{k}})^{-1} \mathbf{x}^{r_{\Phi|-1}}_{\mathbf{k}} + (\mathbf{P}^{r_{\Phi|-1}}_{\mathbf{k}})^{-1} \mathbf{x}^{r_{\Phi|-1}}_{\mathbf{k}} \right]. \]

For the FTD verification, it is necessary to identify the confirmed true track (CTT) and confirmed false track (CFT) using (26). A track \( r \) becomes a CTT if (26) is \( \leq 20 \); otherwise, it becomes a CFT. Furthermore, a CTT could become a CFT due to closely located clutter and targets if (26) exceeds the false track test threshold (\( \geq 40 \)). The 20 and 40 threshold values also depend on the target velocities and surveillance region. Therefore, the chi-squared test is calculated between each confirmed track and each target in each scan. In the multi-target situation, there could be many CTTs following the same target. However, there should be only one CTT for each corresponding target. Thus, the RTS-LMIPDA algorithm exploits the auction algorithm [27], which compares the minimum normalized distance squared value (left-hand-side term of (26)) corresponding to the CTTs. Subsequently, only one CTT is selected. However, if a track \( r \) is identified as a CFT for all targets, then it is counted as a CFT; otherwise, it is counted as the CTT. The procedure and implantation of the RTS-LMIPDA algorithm is illustrated using a flow-chart, as shown in Fig. 1.
(18) becomes lower than the initial TEP. Tracks are determined to be merged or not if they are following up the same target. If the tracks merge at scan \( k \), then the termination is determined by comparing their updated TEPs [1]. If the number of scans \( k \) is equal to the predefined interval length \( N \), then one backward is initialized for each forward track and the smoothing track data (estimate and its smoothing TEP) equals the forward track data (estimate and its updated TEP) at scan \( k = N \). For scan \( k' \geq \) first scan \( k \), the RTS is exploited using (22) for the backward prediction calculation expressed in (23). This is followed by the fusion of the forward and backward predictions using (24), and the RTS-LMIPDA track update using (11) - (20). Otherwise, if \( k' < k \), the algorithm proceeds to the next interval. The procedure is repeated until \( k' \) arrives at the end scan of the simulation run, as depicted in Fig. 1. The RTS-LMIPDA algorithm uses the fixed interval smoothing structure [1], that has a length of \( N - k + 1 \) scans. Figure 2 depicts first two overlapping smoothing interval structures.

Assume the current smoothing interval consisted of scans \( k = 1, 2, 3, \ldots, 8 \), and the next interval overlapped at scan \( N + 1 \) of the current interval, for example, \( k = 5, 6, 7, \ldots, 12 \). That is, the first four scans were discarded after RMSE and FTD statistics accumulation and four new scans were appended in the measurement interval. In the last interval, the smoothing statistics were accumulated for all scans. The RTS-LMIPDA follows four step smoothing procedure:

1. Forward track recursion starts with state pdf prediction (i.e., \( x_{T|2}^r \)) and obtains the state estimate from scan \( k \) to scan \( N \) conditioned on scan \( k - 1 \).
2. Smoothing and Backward tracks are simultaneously evaluated using (22) and (23), respectively, in each scan starting from scan \( N - 1 \) and ending to scan \( k \). Each backward prediction is conditioned on \( k|k+1 \), (e.g., \( x_{T|k}^s \)) and obtained using the fusion of the forward and smoothing estimate in each scan, except the scan \( N \). For example, fusion of \( \{ \hat{x}_{T|7}^r, \hat{x}_{T|7}^s \} = \hat{x}_{T|8}^r \).
3. Fuse the forward and backward predictions using (24) to obtain the smoothing prediction in each scan, conditioning on \( k|Nk \) (e.g., \( \hat{x}_{T|7}^s \)).
4. Finally, update the RTS track based on the sensor measurement up to the first half of interval, and then discard it after the statistic accumulation. Repeat step 1-4 in the next interval and so on.

**IV. EXPERIMENTAL ANALYSIS USING SIMULATIONS**

The simulation analysis compares the estimation accuracy, root-mean square errors (RMSEs), FTD track quality measure, and track retention of the RTS-LMIPDA, FIsJIPDA [16], FlsJITS [17], sJIPDA [18], sJITS [19], and LMIPDA [10] algorithms. The RTS-LMIPDA algorithm is designed to track six targets that are crossing near the position (387, 300) m and are closely moving in a two-dimensional surveillance region that is 700 m wide along the x- and y-axes, as shown in Fig. 3.

Figure 3 illustrates the RTS-LMIPDA multi-target trajectories estimated in one simulation run which clearly emphasize the application of the RTS in the LMIPDA algorithm. In the presence of clutter, each target moves uniformly at a different velocity and crosses one another approximately at scan \( k = 20 \). The total number of sensor scans per simulation run is 36 and the scan time \( T = 1 \) s. The average number of clutter per scan is 51, which satisfies a Poisson distribution with a measurement density of \( \rho_{k,i} = 1 \times 10^4 \) m\(^2\) in each scan. The target detection probability \( P_D = 0.9 \) and the target position measurement
noise reflected by a sensor has a covariance of $R = 25 I_2 \text{ m}^2$, where $I_2$ represents the two-dimensional identity matrix. The target model has process noise with a variance, $q = 0.75 \text{ m}^2/\text{s}^4$. The initial positions of the targets are listed in Table 2.

The simulation experiments of the algorithms are composed of 200 Monte Carlo runs. Each algorithm initialized roughly 117,337 forward tracks (587 per run) with an initial TEP of 0.01. Moreover, both FIsJIPDA and FIsJITS algorithms initialized approximately 351,613 (1758 per run) backward tracks, which consumed extra computational time for the estimation. These forward and backward tracks were maintained in each scan using the state transition probability of Markov Chain, one target existence model expressed in (1), which equals 0.97. The FIsJIPDA, FIsJITS, sJIPDA, and sJITS algorithms consumed a great deal of processing time compared to other algorithms due to the evaluation of joint data association and measurement probabilities of cluster tracks. More specifically, FIsJITS and sJITS that belongs to the computationally expensive ITS family, invested 23.38 secs and 16.4 secs, respectively, per run computation time for smoothing estimation of only six targets in clutters. In contrast, the proposed multi-target RTS-LMIPDA algorithm consumed only 1.45 secs per run computation time, thus the algorithm behaved as a single target tracker, similar to the LMIPDA at the cost of only 0.77 sec smoothing delay. It was also observed that simulation became halt when forcing the FIsJITS algorithm for tracking more than six targets in clutters due to excess of available resource’s memory. The simulation time comparison of the algorithms is shown in Table 3.

The same smoothing interval length was used for the RTS-LMIPDA, FIsJIPDA, FIsJITS, sJIPDA, and sJITS algorithms. Each algorithm is numerically analyzed using the same TEP, but its confirmation threshold was tuned to get almost the similar number of CFTs ($\approx 25$) to ensure a fair comparison analysis. The track retention accounting following useful statistics are accumulated before and the after crossing of targets:

- nCases: the number of CTTs following a target at scan 15
- nOk: the number of CTTs still pursuing the original target at scan 30
- nSwitched: the number of CTTs that end up original target but, now pursuing a different target at scan 30
- nLost: the number of nCases tracks not pursuing any target at scan 30 (because, they became false, merged or were terminated due to low TEP).

The track retention statistics of each algorithm are accumulated from 200 simulation runs listed in Table 4. The RTS-LMIPDA provides highest nOk, and lowest nSwitched and nLost as compared with exiting algorithms.

### Table 2
**INITIAL POSITION OF TARGET (M)**

| Target | Initial Position | Target | Initial Position |
|--------|------------------|--------|------------------|
| 1      | [100; 300; 15; 0]| 4      | [100; 400; 15; -5.2] |
| 2      | [387; 100; 0; 13]| 5      | [200; 100; 10.4; 11] |
| 3      | [100; 200; 15; 5.2] | 6      | [600; 100; -12.6; 12] |

### Table 4
**SIMULATION PARAMETERS AND TRACK RETENTION STATISTICS**

|             | nCases | nOk  | nSwitched | nLost |
|-------------|--------|------|-----------|-------|
| RTS-LMIPDA  | 1187   | 996  | 161       | 13    |
| FIsJIPDA    | 957    | 587  | 370       | 243   |
| sJIPDA      | 1057   | 790  | 267       | 143   |
| FIsJITS     | 1151   | 611  | 540       | 49    |
| sJITS       | 1125   | 652  | 473       | 75    |
| LMIPDA      | 1094   | 933  | 191       | 106   |

The number of CTTs corresponding to all targets is shown in Fig. 4. The proposed RTS-LMIPDA algorithm confirmed a corresponding target track in scan $k = 3$ and maintained tracking of the potential target. The LMIPDA algorithm took complete advantage of the RTS smoothing algorithm to obtain the highest number of CTTs compared to existing algorithms. Only a few of the CTTs were dropped by the RTS-LMIPDA due to the cross-section of the targets in scan $k = 20$.

In the FIsJIPDA, FIsJITS, sJIPDA, and sJITS algorithms, the joint data association probabilities were evaluated only for the confirmed tracks to avoid using an excessive amount of computational resources. Both FIsJITS and sJITS performed well against FIsJIPDA and sJIPDA. However, their tracks were confirmed after scan 5 and have slower growth of confirmation rate as compared to the RTS-LMIPDA algorithm. Similarly, sJIPDA confirms its track in
scan 5 but occasionally tracks the corresponding target and eventually misses the target due to nSwitch and nLost statistics. In both FlsJIPDA and sJIPDA algorithms, their CTTs were often distracted due to the association of the shared target measurements observed in scans $k = 16–25$. Because, in these scans, the targets are closely moving in clutters. Thus, the number of CTTs for the FlsJIPDA and sJIPDA algorithms drops nearly 25% of the total number of CTTs as depicted in Fig. 4. Thus, the joint data association based smoothing methods were not proved to be advantageous for tracking a large number of targets in a cluttered environment mainly due to excessive processing time compared to other algorithms (Table 3) and the regular distraction of the confirmed tracks due to the sharing of the joint measurements. Thus, the RTS-LMIPDA algorithm provides improved estimation accuracy and FTD as compared to existing smoothing algorithms as well as LMIPDA. The RMSEs of the target’s position ($r = 1, 2, 4, 6$) were calculated using the position estimation error statistics of the confirmed true tracks as shown in Fig. 5.

![Fig. 5. RMSEs of the target’s position estimation ($r = 1, 2, 4, 6$).](image)

The LMIPDA being a non-smoothing algorithm produced high estimation errors. It was observed that the joint data association based algorithms such as FlsJIPDA, FlsJITS, sJIPDA, and sJITS often fluctuating estimation errors due to the distraction of shared tracks within vicinity of the neighbor measurements. However, sJIPDA have a similar RMSE trend as compared to RTS-LIPDA, but it was only observed for targets 2 and 6. It can also be seen that the FlsJIPDA algorithm lost target 1 in scans $k = 22–25$ and $k = 27$, and confirmed the target 4 only in scans $k = 21–23$ and $k = 27$. Similarly, the FlsJITS algorithm lost target 1 in scan $k = 26$, and target 6 in scans $k = 20–21$ and $k = 24–26$. In addition, FlsJITS algorithm confirmed the targets 4 and 6 quite late in the simulation for example, it detected the target 4 in scan $k = 13$ and the target 6 in scan $k = 17$. In the situation of targets 2 and 6, sJITS algorithm consistently produced the high RMSEs as compared to other smoothing algorithms due to the weak growth of smoothing target existence probability in each scan $k$. In comparison, the RTS-LMIPDA algorithm maintained the smoothed target existence probability well in each scan $k$ and obtain a significant reduction in the estimation errors, as depicted in Figs. 5(a)–(d). Thus, the proposed RTS-LMIPDA algorithm outperformed the existing algorithms and demonstrated superior performance in terms of multi-target state estimation accuracy, FTD track quality measurement, and RMSEs as shown in Figs. 4 and 5.

V. Conclusion

This study extended the LMIPDA algorithm using the RTS smoothing algorithm to improve its estimation accuracy and FTD for tracking multiple cross-over targets in a cluttered environment. Unlike the existing joint data association based multi-target algorithms (discussed in section IV), the LMIPDA did not form a cluster of tracks, thus ignoring entire evaluation steps of the joint multi-data association probabilities required for multi-target estimations. This saved a great deal of processing time in the multi-target estimation compared to existing algorithms (Table 3). Similar to the sJIPDA algorithm, the RTS-LMIPDA algorithm calculated the backward prediction using forward and RTS estimates corresponding to each forward track. Consequently, smoothing estimates were calculated using the fusion of forward and backward predictions. The integrated RTS-LMIPDA algorithm was experimentally proven to provide a significant smoothing estimation and FTD performance. The RTS-LMIPDA algorithm is verified to be an efficient asset for practical implementation and demonstrated the track retention statistics with the highest nOk and lowest nSwitched and nLost at the cost of smoothing time delay.

ACKNOWLEDGMENT

This research work was supported by the research fund of Sejong University, Seoul, Korea (2021-2022) and the basic science research program of the National Research Foundation of Korea funded by Ministry of Education under Grant 2020R1G1A1013221.

REFERENCES

[1] S. Challa, R. Evans, M. Morelande, and D. Mušicki, “Fundamentals of Object Tracking,” New York, USA, Cambridge Univ. Press, 2011

[2] D. Mušicki, R. Evans, and S. Stankovic: “Integrated probabilistic data association,” IEEE trans. Automat. Contr., vol. 39, no. 6, pp. 1237–1241, 1994.

[3] S. He, H. Shin and A. Tsourdos, “Multi-Sensor Multi-Target Tracking Using Domain Knowledge and Clustering,” IEEE Sensors J., vol. 18, no. 19, pp. 8074–8084, Oct. 2018
[4] D. Mušicki and R. Evans, “JIPDA: Automatic target tracking avoiding track coalescence,” IEEE Trans. Aerosp. Electron. Syst., vol. 51, no. 2, pp. 962 – 974, 2015.

[5] E. H. Lee, Q. Zhang, and T. L. Song, “Markov Chain Realization of Joint Integrated Probabilistic Data Association,” MDPI, vol. 17, no. 12, pp. 1 – 12, 2017.

[6] D. Mušicki and R.J. Evans, “Multiscan Multitarget Tracking in Clutter with Integrated Track Splitting Filter,” IEEE Trans. Aerosp. Electron. Syst., vol. 45, no. 4, pp. 1432–1447, Oct 2009.

[7] T. L. Song, D. Mušicki, and K. Yong, “Multi-target tracking with state dependent detection,” IET Radar Sonar Navig., vol. 9, no. 1, pp. 10–18, 2015.

[8] T. Sathyan, T. Chin, S. Arulampalam and D. Suter, “A Multiple Hypothesis Tracker for Multitarget Tracking with Multiple Simultaneous Measurements,” in IEEE Journal of Selected Topics in Signal Processing, vol. 7, no. 3, pp. 448-460, June 2013, doi: 10.1109/JSTSP.2013.2258322.

[9] J. Sun, Z. Wang, Q. Li, “A New Multiple Hypothesis Tracker Using Validation Gate with Motion Direction Constraint,” Sensors (Basel), vol. 20, no. 17, pp. 4816, 2020.

[10] D. Mušicki and B.L. Scala, “Multi-target tracking in clutter without measurement assignment,” IEEE Trans. Aerosp. Electron. Syst., vol. 44, no. 3, pp. 877–896, 2008.

[11] D. Mušicki, T. L. Song, and H. Lee, “Linear multitarget finite resolution tracking in clutter”, IEEE Trans. Aerosp. Electron. Syst., vol. 50, no. 3, pp. 1798–1812, 2014.

[12] Y. Bar-Shalom, X. R. Li, and T. Kirubarajan, “Estimation with Applications to Tracking and Navigation: Theory Algorithms and Software,” USA: Wiley & Sons, Inc. 2004.

[13] S. Memon, T. L. Song, and T. H. Kim, “Smoothing Data Association for Target Trajectory Estimation in Cluttered Environments,” Eurasip J. Adv. Signal Process., vol. 2016, no. 1, pp. 1–21, 2016.

[14] S. Memon, T. L. Song, K. H. Memon, I. Uliah, and U. Khan, “Modified Smoothing Data Association for Target Tracking in Clutter,” Expert Syst. Appl., vol. 141, no.1, Mar. 2020.

[15] K. Myungun, S. A. Memon, M. Shin, and H. Son, “Dynamic based trajectory estimation and tracking in an uncertain environment,” Expert Systems with Applications, vol. 177, 2021.

[16] S. A. Memon, M. Kim, M. Shin, J. Daudpoto, D. M. Pathan, and H. Son, “Extended Smoothing Joint Data Association for Multi-target Tracking in Cluttered Environments,” IET Radar, Sonar Navig. DOI: 10.1049/iet-rsn.2019.0075, Aug. 2019.

[17] S. A. Memon, M. Kim, and H. Son, “Tracking and Estimation of Multiple Cross-over Targets in Clutter,” Sensors, vol. 19, no. 3, pp. 741, 2019.

[18] T.H. Kim, D. Musicki, T.L. Song, and C.M. Lee, “Smoothing joint integrated probabilistic data association”, IET Radar Sonar Navig., vol. 9, no. 1, pp. 62-68, 2015.

[19] T.H. Kim and T.L. Song, “Multi-target multi-scan smoothing in clutter,” IET Radar Sonar Navig., vol. 10, no. 7, pp. 1270–1276, 2016.

[20] S. Jason, B. Travis, R. Mark, B. Jason, J. Moriba, and H. Keric, “Joint Probabilistic Data Association and Smoothing Applied to Multiple Space Object Tracking,” Journal of Guidance Control and Dynamics, vol. 41, no. 1, pp. 1–15, 2017.

[21] W. Koch, “Fixed-interval retrodiction approach to Bayesian IMM-MHT for maneuvering multiple targets,” IEEE Trans. Aerosp. Electron. Syst., vol. 36, pp. 2–14, 2000.

[22] R. P. S. Mahler, Ba-T. Vo and Ba-N. Vo, “Forward-Backward Probability Hypothesis Density Smoothing,” IEEE Trans. Aerosp. Electron. Syst., vol. 48, no. 1, pp. 707-728, Jan. 2012.

[23] S. Nagappa, E. D. Delande, D. E. Clark, and J. Houssineau, “A Tractable Forward–Backward CPHD Smoother,” IEEE Trans. Aerosp. Electron. Syst., vol. 53, no. 1, pp. 201–217, Jan. 2017.

[24] F. Farahi and H. S. Yazdi, “Probabilistic Kalman filter for moving object tracking,” Signal Process: Image Comm., vol. 82, 2020.

[25] Song, T.L., Mušicki, D.: ‘Adaptive clutter measurement density estimation for improved target tracking’, IEEE Trans. Aerosp. Electron. Syst., vol. 47, no. 2, pp. 1457–1466, 2011.

[26] D. Musicki and T.L. Song, “Smoothing innovations and data association with IPDA”, Automatica, vol. 48, no. 7, pp. 1324–1329, 2012.