Retrodiction of Data Association Probabilities via Convex Optimization

Selim Özgen, Florian Rosenthal, Jana Mayer, Benjamin Noack, Uwe D. Hanebeck
Intelligent Sensor-Actuator-Systems, KIT, Germany
{selim.oezgen,firstname.lastname}@kit.edu, uwe.hanebeck@ieee.org

Marco F. Huber
USU Software AG, Karlsruhe, Germany
Karlsruhe Institute of Technology (KIT), Germany
marco.huber@ieee.org

***Abstract*—In a surveillance environment with high clutter, finding the correct measurement to track associations becomes extremely important for efficient target tracking. This study offers a novel algorithm to retrodict the data association probabilities at any past time instant, when the batch set of measurements is kept in memory. For the retrodiction procedure, the batch association cost is first written explicitly as a binary integer optimization problem with a quadratic cost function and it is shown that the relaxed form of the problem is convex. From the relaxed problem, a lower bound for the optimal association cost is derived, and this lower bound is used as the data association probabilities pertaining to that selected time instant in the past. Due to its consideration of the batch set of data in a retrospective manner, we will call this algorithm as Retrodictive Probabilistic Data Association, RPDA. For simplification of the mathematical analysis, a single point target with no missing measurements, i.e. $P_D = 1$, is taken into account.**

***Index Terms*—target tracking, measurement-to-track association, convex optimization.**

I. Introduction

A fundamental problem of target tracking is the measurement-to-track association [1]. This study introduces a novel approach for tracking targets accurately under the presence of sensor noise and heavy clutter. There is a vast literature on the subject and there has been a number of unconventional solutions offered such as sampling based methods [5], incorporation of machine learning ideas [6] or random finite set approaches where the association problem is solved using set-valued observations [2]–[4]. Yet still, conventional solutions to the data association problem can be characterized under recursive and batch tracking methods [7].

Recursive trackers associate the measurements with the track considering only the instantaneous measurement set. The simplest approach is the Nearest Neighbour Algorithm (NN), where only one measurement is considered for the measurement update and this measurement is selected using the maximum likelihood principle [8]. Robust tracking results can be achieved by using probabilistic approaches, where the track is updated considering all viable measurements with different weights. The Probabilistic Data Association (PDA) based on the minimum mean square theory is the most frequently used association algorithm of this kind [9]. Many different versions of PDA can be found in literature [10], [11].

Batch estimation methods take measurement sets from consecutive time steps into account. A decision given at any point depends not only on the elapsed measurements, but also on the future measurements, which tends to a delay in the decision. The advantage is that future measurements are also used to resolve the data association ambiguities. The optimal solution here would be the enumeration and evaluation of all association decision sequences which is known as Multiple Hypothesis Tracking (MHT) [12], [13]. Yet, due to the computational complexity of the problem, sub-optimal solutions have been offered such as relaxing the single measurement-to-track association constraint [14], [15], considering two consecutive scans together [16] or using a smoothing procedure to find the data association probabilities [17], [18].

This paper presents a novel approach to this batch smoothing procedure. As an alternative to traversing a decision tree, first the batch association cost function is written explicitly in terms of the decision variables for the whole time horizon. This cost function is then used to form a binary integer optimization problem with a quadratic cost function. As the solution method, continuous relaxation of the decision variables will be considered, which is a well-known technique used for planning problems such as data association [15], [16], [19], [20] and sensor management [21], [22]. It will be formally proven that the relaxed optimization problem is convex.

In a former paper [23], the same batch association cost function was minimized by using rollout algorithms [24]. There, the first branches of the decision tree, which correspond to the the near-future costs, were traversed and the far-future costs from the further branches were calculated by using a heuristic policy. This heuristic policy provides an upper bound for the association cost which was then used to decide on the past decision variable. In this novel formulation, the aim is to retrodict the data association probabilities. For this, similar to the former approach, the first branches of the decision tree are traversed. The cost calculated from the first branches are directly proportional to the data association probabilities used by the PDA algorithm. Then, a relaxed binary integer problem is formulated for a respective horizon to find a lower bound for the association costs. Finally, for the calculation of the data association probabilities, this overall cost will be used instead of the cost of the first branches. For this reason, we saw it fit to call this novel approach as Retrodictive Probabilistic Data Association. As the formulated relaxed problem is convex, there are polynomial time algorithms for the calculation of

the optimal solution [25], [26].

This paper is formulated as follows. In Section II, the measurement-to-track association problem in case of batch measurements and the respective cost function will be introduced. In Section III, the convex optimization problem will be formulated and the aforementioned solution to the data association problem will be explained in detail. In Section IV, the simulation results will be given. Section V will conclude the paper.

II. Problem Formulation

The measurement-to-track association problem described in this study is analyzed for discrete-time linear Gaussian models. The target extension is not taken into account and it is assumed that there is at most one measurement coming from each target at each time instant. For simplification of the mathematical formulation, potential missed detections are omitted, but this can also be incorporated into the problem. The system model of the target is given by

$$\underline{\boldsymbol{x}}_k = \mathbf{A}_k \underline{\boldsymbol{x}}_{k-1} + \underline{\boldsymbol{w}}_{k-1}. \tag{1}$$

The measurement model is given as below:

$$\underline{\boldsymbol{z}}_k = \mathbf{H}_k \underline{\boldsymbol{x}}_k + \underline{\boldsymbol{v}}_k, \tag{2}$$

where $k = 0, 1, \dots$ is the discrete time index, both $\mathbf{A}_k$ and $\mathbf{H}_k$ are time-variant matrices, and $\underline{\boldsymbol{w}}_k \sim \mathcal{N}(0, \mathbf{C}_k^w)$. $\underline{\boldsymbol{v}}_k \sim \mathcal{N}(0, \mathbf{C}_k^v)$ denote the process and measurement noises, respectively, where $\mathcal{N}(m, P)$ denotes normal probability distribution with mean m and covariance P. The initial system state is also given by $\underline{\boldsymbol{x}}_0 \sim \mathcal{N}(\underline{\hat{x}}_0, \mathbf{C}_0)$.

We assume that the measurements start arriving at time $k = 1$. A measurement value $\underline{z}_k$ is a realization of the random vector $\underline{\boldsymbol{z}}_k$. Yet at every time step, a measurement set $\mathcal{Z}_k = \{\underline{z}_k^1, \dots, \underline{z}_k^{m_k}\}$ is received by the sensor where m_k is the number of measurements available at time k. Note that, the measurements either arrive from clutter or the target itself. Therefore, in a single target scenario with probability of detection $P_D = 1$, the association decision at each time k becomes

$$\theta_k = i \implies (\underline{z}_k^i \text{ is from target}) \wedge (\mathcal{Z}_k \setminus \underline{z}_k^i \text{ is clutter}) \tag{3}$$

where $i \in \{1, \dots, m_k\}$. Now, define $\mathcal{Z}^{k-1} = \{\mathcal{Z}_1, \dots, \mathcal{Z}_{k-1}\}$ as the sets of measurements and $\Theta^{k-1} = (\theta_1, \dots, \theta_{k-1})$ as the association decision sequence up to $k-1$. Then we can write

$$p(\underline{\boldsymbol{x}}_k | \mathcal{Z}^{k-1}, \Theta^{k-1}) = \mathcal{N}(\underline{\boldsymbol{x}}_k; \underline{\hat{x}}_k^p(\Theta^{k-1}), \mathbf{C}_k^p) \tag{4}$$

and $\underline{\hat{x}}_k^p(\Theta^{k-1})$ and $\mathbf{C}_k^p$ refer to the predicted mean and covariance of the state conditioned on the measurements up to $k-1$, respectively. When a linear system with Gaussian noise with $P_D = 1$ is considered, the Kalman filter $\mathbf{C}_k^p$ can be calculated independent of the value of the measurements, thus $\mathbf{C}_k^p$ is not function of Θ^{k-1}. After the instant measurement set $\mathcal{Z}_k$ is taken and the association decision θ_k is given, the updated probability distribution

$$p(\underline{\boldsymbol{x}}_k | \mathcal{Z}^k, \Theta^k) = \mathcal{N}(\underline{\boldsymbol{x}}_k; \underline{\hat{x}}_k^e(\Theta^k), \mathbf{C}_k^e), \tag{5}$$

can be found where $\underline{\hat{x}}_k^e(\Theta^k)$ and $\mathbf{C}_k^e$ refer to the estimated mean and covariance of the state conditioned on the measurements up to k. Similar to $\mathbf{C}_k^p$, $\mathbf{C}_k^e$ can be calculated independent of Θ^k. When the sequence of measurement-to-target associations for all times are known, the optimal estimator for the system state in maximum likelihood and linear minimum mean square sense is given by (4) and (5).

When the association decisions are not given, the problem changes fundamentally. Assume that all measurements up to time N are collected. As it is not possible to know which Θ^N is correct, we need to calculate how probable each hypothesis is. Using Bayes' rule, the posterior probability distribution for any association decision sequence can be written as follows;

$$p(\Theta^N | \mathcal{Z}^N) = \frac{p(\mathcal{Z}^N | \Theta^N) \cdot P(\Theta^N)}{P(\mathcal{Z}^N)} \tag{6}$$

$$\propto p(\mathcal{Z}^N | \Theta^N), \tag{7}$$

assuming that there is no prior information about any of the Θ^N. To calculate $p(\mathcal{Z}^N | \Theta^N)$, we need to also take into account the distribution of the clutter as can be seen from from (3). To simplify the calculations, we will assume that the clutter is uniformly and independently distributed in the measurement space V. The number of clutter detections, m_c, within a region of volume V is assumed to have a Poisson distribution:

$$p(\underline{z}_k^i | \underline{z}_k^i \text{ is clutter}) = \frac{1}{V} \tag{8}$$

$$m_c \sim \text{Poisson}(\beta_{FA} V), \tag{9}$$

where β_{FA} is the clutter density per unit volume. Now define $\underline{\boldsymbol{x}}_{0:N} = \{\underline{\boldsymbol{x}}_0, \dots, \underline{\boldsymbol{x}}_N\}$ as the augmented set of states, we can write,

$$p(\Theta^N | \mathcal{Z}^N) \propto \int p(\mathcal{Z}^N, \underline{\boldsymbol{x}}_{0:N} | \Theta^N) \, d\underline{\boldsymbol{x}}_{0:N} \tag{10}$$

$$= \int \Bigg(\prod_{k=1}^{N} p(\mathcal{Z}_k | \underline{\boldsymbol{x}}_k, \mathcal{Z}^{k-1}, \Theta^N) \, p(\underline{\boldsymbol{x}}_k | \underline{\boldsymbol{x}}_{0:k-1}, \mathcal{Z}^{k-1}, \Theta^N) \Bigg) p(\underline{\boldsymbol{x}}_0) \, d\underline{\boldsymbol{x}}_{0:N} \tag{11}$$

$$= \int \cdots \int \Bigg(\prod_{k=1}^{N} p(\underline{z}_k^{\theta_k} | \underline{\boldsymbol{x}}_k) p(\underline{\boldsymbol{x}}_k | \underline{\boldsymbol{x}}_{k-1}) \Bigg) p(\underline{\boldsymbol{x}}_0) \, d\underline{\boldsymbol{x}}_{0:N} \tag{12}$$

$$= \prod_{k=1}^{N} \mathcal{N}(\underline{z}_k^{\theta_k}; \mathbf{H}_k \underline{\hat{x}}_k^p(\Theta^{k-1}), \mathbf{S}_k) \tag{13}$$

where $\mathbf{S}_k = \mathbf{H}_k \mathbf{C}_k^p \mathbf{H}_k' + \mathbf{C}_k^v$ is the innovation covariance. This likelihood function measures how good the data is fitting to the innovation (residual) process. In (12), we have used the uniform and independent distribution assumption for the clutter. For the single target case, the term $p(\mathcal{Z}_k | \underline{\boldsymbol{x}}_k, \mathcal{Z}^{k-1}, \Theta^N)$, can be written as a product of likelihoods where $\underline{z}_k^{\theta_k}$ is coming from the target and from (8), the measurements coming from clutter can be cancelled out.

Our aim is to find the association decision sequence with the highest posterior probability. This is equivalent to finding the

maximum value of (13). Taking the negative logarithm of (13) and getting rid of the constant terms, the objective function that must be minimized can be written as:

$$J(\Theta^N; \underline{\hat{x}}_1^p, \mathcal{Z}^N) = \sum_{k=1}^{N} c(\theta_k; \underline{\hat{x}}_k^p(\Theta^{k-1}), \mathcal{Z}_k), \tag{14}$$

$$c(\theta_k; \underline{x}, \mathcal{Z}_k) = (\underline{z}_k^{\theta_k} - \mathbf{H}_k\underline{x})'\mathbf{S}_k^{-1}(\underline{z}_k^{\theta_k} - \mathbf{H}_k\underline{x}). \tag{15}$$

Note that the innovation covariances are not dependent on the association history due to the formulation of the problem. Note that when $k = 1$, we can write $\underline{\hat{x}}_k^p(\Theta^{k-1}) = \underline{\hat{x}}_1^p$ in (13); therefore the overall cost at left-hand side of (14) can be defined as a function of $\underline{\hat{x}}_1^p$. Finally, the immediate cost function in (15) is defined for the generic value of $\underline{x}$ while for every different selection of Θ^{k-1}, we arrive at a different $\underline{\hat{x}}_k^p$.

III. Solution Method

A. Formulation of the constrained optimization problem

To be able to use the matrix vector notation efficiently, we will first define the overall cost function given in (14) as a binary integer problem. Define the binary variables $\gamma_k^i \in \{0,1\}$, $1 \leq i \leq m_k$ and $\sum_{i=1}^{m_k} \gamma_k^i = 1$. For these binary variables, we can write

$$\theta_k = i \implies \gamma_k^i = 1 \wedge \gamma_k^j = 0, \ \forall j \neq i. \tag{16}$$

Now, define the binary variable vector $\underline{\Gamma}_k = [\gamma_k^1 \ \gamma_k^2 \ \dots \gamma_k^{m_k}]' \in \{0,1\}^{m_k}$ and the measurement matrix $\mathbf{Z}_k = [\underline{\hat{z}}_k^1 \ \underline{\hat{z}}_k^2 \ \dots \underline{\hat{z}}_k^{m_k}]$. Clearly, $\underline{z}_k^{\theta_k} = \mathbf{Z}_k\underline{\Gamma}_k$. Putting this into the immediate cost function at (15), we arrive at

$$c(\underline{\Gamma}_k; \underline{x}, \mathcal{Z}_k) = (\mathbf{Z}_k\underline{\Gamma}_k - \mathbf{H}_k\underline{x})'\mathbf{S}_k^{-1}(\mathbf{Z}_k\underline{\Gamma}_k - \mathbf{H}_k\underline{x}). \tag{17}$$

Using this quadratic cost in (17), we want to define a quadratic function for the overall cost. First, note that $\underline{\hat{x}}_{k+1}^p$, the mean of the predicted state estimate at time $k+1$, can be calculated when the initial state estimate $\underline{\hat{x}}_1^p$ and the association decisions up to and including time k are known. Therefore, if we define the column vector $\underline{\Gamma}^k = [\underline{\Gamma}_1' \ \underline{\Gamma}_2' \dots \underline{\Gamma}_k']'$, we can write

$$\underline{\hat{x}}_{k+1}^p = \mathbf{M}_x^k\underline{\hat{x}}_1^p + \mathbf{M}_z^k\underline{\Gamma}^k, \tag{18}$$

where the derivation of the matrices $\mathbf{M}_x^k$ and $\mathbf{M}_z^k$ in (18) are given in Appendix A. Using this result, the overall cost can be written as

$$J(\underline{\Gamma}^N; \underline{\hat{x}}_1^p, \mathcal{Z}^N) = (\underline{\Gamma}^N)'\mathbf{L}_{z,z}\underline{\Gamma}^N + (\underline{\hat{x}}_1^p)'\mathbf{L}_{x,z}\underline{\Gamma}^N + f(\underline{\hat{x}}_1^p), \tag{19}$$

where the derivation of the matrices $\mathbf{L}_{z,z}$ and $\mathbf{L}_{x,z}$ in (19) are given in Appendix B. The right hand side of (19) contains a term dependent only on $\underline{\hat{x}}_1^p$ which has no impact on the association decision sequence $\underline{\Gamma}^N$. Using (19), the problem setting can be formulated as the constrained optimization problem such as

$$\begin{aligned} \underline{\Gamma}_*^N &= \underset{\underline{\Gamma}^N}{\operatorname{argmin}} \, J(\underline{\Gamma}^N; \underline{\hat{x}}_1^p, \mathcal{Z}^N) \\ &\text{s.t. } \underline{1}' \cdot \underline{\Gamma}_k = 1, \ 1 \leq k \leq N \\ &\quad \underline{\Gamma}_k \in \{0,1\}^{m_k}, \ 1 \leq k \leq N \end{aligned} \tag{20}$$

where $\underline{1}$ is a vector of appropriate size composed of ones.

B. Convex relaxation

Notice that (20) is a binary integer problem with a quadratic cost function. The exact solution can only be found by an exhaustive search of all possible branches which is NP-hard. The convex relaxation of (20) can be achieved by

$$\begin{aligned} \tilde{\underline{\Gamma}}_*^N &= \underset{\underline{\Gamma}^N}{\operatorname{argmin}} \, J(\underline{\Gamma}^N; \underline{\hat{x}}_1^p, \mathcal{Z}^N) \\ &\text{s.t. } \underline{1}' \cdot \underline{\Gamma}_k = 1, \ 1 \leq k \leq N \\ &\quad \underline{\Gamma}_k \in [0,1]^{m_k}, \ 1 \leq k \leq N \end{aligned} \tag{21}$$

where the binary constraints are replaced with the affine ones. The following lemmas show that (21) is convex.

Lemma III.1. $\mathbf{L}_{z,z}$ *is a positive semi-definite matrix.*

Proof. In (17), we know that $c(\underline{\Gamma}_k; \underline{x}, \mathcal{Z}_k) \geq 0$ for all values of $\underline{\Gamma}_k$ and $\underline{x}$ as $\mathbf{S}_k > 0$. As (14) is the sum of incremental costs, we can also write $J(\underline{\Gamma}^N; \underline{x}, \mathcal{Z}^N) \geq 0$ for all values of $\underline{x}$ and $\underline{\Gamma}^N$. Fixing $\underline{x} = 0$, we arrive at $J(\underline{\Gamma}^N; 0, \mathcal{Z}^N) = (\underline{\Gamma}^N)'\mathbf{L}_{z,z}\underline{\Gamma}^N \geq 0$ for all values of $\underline{\Gamma}^N$. □

Note that $\underline{\hat{x}}_1^p$ is a constant value in (19). Then the addend $(\underline{\hat{x}}_1^p)'\mathbf{L}_{x,z}\underline{\Gamma}^N$ in (19) is a linear function of $\underline{\Gamma}^N$ and therefore it is convex. We will provide the following lemma without proof for the sake of completeness.

Lemma III.2. *Sum of two convex functions is a convex function [27].*

Lemma III.3. *By relaxing the problem as in* (21)*, the objective function becomes convex in terms of* $\underline{\Gamma}^N$.

Proof. The optimization problem in (21) is convex if the objective function and inequality constraints are convex and the equality constraints are affine. It is trivial to see that the constraints satisfy the requirements. As we know that $J(.)$ is a quadratic function and $\mathbf{L}_{z,z}$ is a positive semi-definite matrix, it is possible to deduce that the function $J(.)$ is also convex. □

Using these lemmas, we can assert that the relaxed version of the cost function is convex with a unique minimum value. There are efficient methods for solving the convex relaxation problems that typically require only a few tens of iterations for calculating the optimal solution even for large problem sizes [27]. However, the solution of the convex problem only approximates the optimal solution of the data association problem given in (20). That is to say, $\tilde{\underline{\Gamma}}_*^N$ is not composed of binary variables and the value of $J(\tilde{\underline{\Gamma}}_*^N; \underline{\hat{x}}_1^p, \mathcal{Z}^N)$ is only a lower bound for $J(\underline{\Gamma}_*^N; \underline{\hat{x}}_1^p, \mathcal{Z}^N)$. There are many methods like sampling or swapping that can be used to recover a binary solution for the problem and give an upper bound for the optimal solution. Moreover, to find the optimal solution to the constrained optimization, it is still necessary to utilize a further search technique, such as branch and bound, and visit the branches of the search tree [21]. However, it is possible to utilize the solution of the relaxed problem to retrodict the data association probabilities in a former time step as will be explained in the following section.

C. Approximate Stochastic Optimization

To emphasise the dependency of the optimal value function to the binary decision variable vector $\underline{\Gamma}_1$, we can write the overall cost function in the following form:

$$J(\underline{\Gamma}^N; \underline{\hat{x}}_1^p, \mathcal{Z}^N) = c(\underline{\Gamma}_1; \underline{\hat{x}}_1^p, \mathcal{Z}_1) + J(\underline{\Gamma}^{2:N}; \underline{\hat{x}}_2^p(\underline{\Gamma}_1), \mathcal{Z}^{2:N}) \tag{22}$$

where $\underline{\Gamma}^{2:N} = [\underline{\Gamma}_2' \dots \underline{\Gamma}_N']'$ and $\mathcal{Z}^{2:N} = \{\mathcal{Z}_2, \dots, \mathcal{Z}_N\}$. Clearly, the solution of (22) can be achieved by fixing the decision variable vector $\underline{\Gamma}_1$ to any of the possible m_1 values, calculating the first addend on the right hand side of (22) and then repeating the calculations for $J(\underline{\Gamma}^{2:N}; \underline{\hat{x}}_2^p(\underline{\Gamma}_1), \mathcal{Z}^{2:N})$ in a similar manner. As calculating the optimal cost function in (22) is NP-hard, a computationally affordable approximation of the second addend in (22) is necessary. Instead, we will use the following approximation;

$$V(\underline{\Gamma}_1; \underline{\hat{x}}_1^p, \mathcal{Z}^N) = c(\underline{\Gamma}_1; \underline{\hat{x}}_1^p, \mathcal{Z}_1) + J(\tilde{\underline{\Gamma}}_*^{2:N}; \underline{\hat{x}}_2^p(\underline{\Gamma}_1), \mathcal{Z}^{2:N}), \tag{23}$$

where $\tilde{\underline{\Gamma}}_*^{2:N}$ is the solution of the relaxed optimization problem in (21) and has a unique value. This way, a lower bound for the overall cost (23) can be calculated for any discrete value of $\underline{\Gamma}_1$. This lower bound can then be used as the retrodicted data association probabilities as

$$p(\underline{\Gamma}_1|\mathcal{Z}^N) \propto \exp^{-V(\underline{\Gamma}_1; \underline{\hat{x}}_1^p, \mathcal{Z}^N)}, \tag{24}$$

where $\underline{\Gamma}_1$ is any of the feasible binary integer solutions. If (23) is calculated for m_1 different values of $\underline{\Gamma}_1$, they can be used as the posterior association decision probability after normalization. Recalling our initial notation, we can write

$$p(\Theta_1 = i|\mathcal{Z}^N) = p(\underline{\Gamma}_1 = \underline{\boldsymbol{e}}_i|\mathcal{Z}^N) \tag{25}$$

where $\underline{\boldsymbol{e}}_i$ is a unit vector with a 1 in the ith position.

D. Using receding horizons

Although it is possible to use the applied strategy for any given horizon N, the dependency of a former association decision to the future decisions decreases at every time step. Therefore, we will use the proposed algorithm with a receding horizon. Assume that the association decisions are given until time k, that is, $\underline{\boldsymbol{x}}_{k+1} \sim \mathcal{N}(\underline{\hat{x}}_{k+1}^p(\underline{\Gamma}^k), \mathbf{C}_{k+1}^p)$. Similar to (19), we can write the overall cost function as $J(\underline{\Gamma}^{k+1:k+N}; \underline{\hat{x}}_{k+1}^p, \mathcal{Z}^{k+1:k+N})$ where $\underline{\Gamma}^{k+1:k+N}$ and $\mathcal{Z}^{k+1:k+N}$ are the binary decision variables and the set of measurements from time $k+1$ to $k+N$, respectively. Therefore, the aim of the algorithm is to retrodict the data association probabilities at time $k+1$ by considering all of the measurement history up to $\mathcal{Z}^{k+N}$.

IV. Simulations

A. Problem definition

We consider the problem of tracking a single target under clutter in two dimensions $x, y \in [0, 3000]\, m$. The state vector is $\underline{\boldsymbol{x}} = [p_x\, v_x\, p_y\, v_y]'$. As we want to consider the effect of data association ambiguity, there is no model mismatch for this problem. Both the target and the tracker use a constant velocity model as defined below,

$$\underline{\boldsymbol{x}}_k = \begin{bmatrix} 1 & T & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & T \\ 0 & 0 & 0 & 1 \end{bmatrix} \underline{\boldsymbol{x}}_{k-1} + \begin{bmatrix} T^2/2 & 0 \\ T & 0 \\ 0 & T^2/2 \\ 0 & T \end{bmatrix} \underline{\boldsymbol{w}}_{k-1},$$

$$\underline{\boldsymbol{y}}_k = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \underline{\boldsymbol{x}}_k + \underline{\boldsymbol{v}}_k,$$

where $T = 1s$ is the sampling time. $\underline{\boldsymbol{w}}_k \sim \mathcal{N}\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, I_{2\times 2}\right)$ and $\underline{\boldsymbol{v}}_k \sim \mathcal{N}\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \sigma^2 I_{2\times 2}\right)$. The algorithm is tested for 100 Monte Carlo runs for different values of measurement noise standard deviation $\sigma \in \{20, 40\}m$, and the clutter density, $\beta_{FA} \in \{10^{-5}, 2\times 10^{-5}\}m^{-2}$. The volume of the surveillance region, V, in Equation (9) is $9 \times 10^6\, m^2$. The time lag for giving an association decision, N, is taken as 5 dwell times [28]. The simulation environment and the tracker codes were written in MATLAB environment using a standard desktop computer (Intel Core i7-6700 3.4GHz 16GB RAM).

B. Implementation issues

Although the model parameters are taken to be the same, there are some important issues to be pointed out about the experimental setup about the false track formation and measurement gating. Recall that the problem formulation in Section II was given for a single target with no missed detections. However, when a high clutter environment is taken into consideration, the formation of false tracks is inevitable. As we want to see the performance of the proposed algorithm for a generic scenario, the proposed idea will be used for multi-target tracking where all tracks are handled independent from each other with the proposed approach. To avoid false track formation, track initiation is handled by the M/N logic with high values in each phase of the algorithm ($9/4$ logic) [8]. Moreover, as there are no missed detections and model mismatch, we will delete the tracks after the first no measurement update. Therefore there is no prior information used about the track besides the fact that there are no missed detections.

The probability of gating is another important parameter of the experimental procedure. Recall that in the problem formulation, we have assumed that there are no missing measurements; $P_D = 1$. Gating probability is taken, $P_G \approx 1$, due to two reasons. First, when calculating the overall cost function in (14), we assume that the original measurement from the target is always available for the tracker to update the track. Second, this is also important for a fair comparison of the algorithms. For instance, for every association instance, PDA-based algorithms give some probability mass to no association by assuming all the measurements that pass the gating procedure are clutter. If $P_G \approx 1$, this probability mass will approach to zero, which will force all algorithms to use

any available measurement to perform the measurement update step. Yet under high clutter for a practical tracking algorithm, it is too costly to take into account all possible measurement-to-track associations due to both the number of computations and also the possible number of false tracks. Therefore, we have selected $P_G = 1 - 10^{-7}$ to make sure that the original measurement falls inside the measurement gate.

We would also like to discuss the gating procedure for the given algorithm. Through the calculation of $J(\underline{\Gamma}^{k+1:k+N}; \hat{\underline{x}}^p_{k+1}, \mathcal{Z}^{k+1:k+N})$ in (19), no explicit state estimates are calculated that can be used for the gating procedure throughout the horizon N. Hence, before starting the calculation of (19), a subset of the measurements are selected by,

$$\tilde{\mathcal{Z}}_{k+n} = \{\underline{z}^i_{k+n} |\ p(\underline{z}^i_{k+n} | \mathcal{Z}^k, \underline{\Gamma}^k) > p_G\},\ 1 \leq n \leq N \quad (26)$$

where p_G is a predefined threshold to satisfy the probability of gating value [29]. If there is any time instant n such that $\tilde{\mathcal{Z}}_{k+n} = \emptyset$, the tracker performs a no measurement update at time $k+1$.

C. Comparisons

The comparisons are performed between RPDA, recursive estimation algorithms NN, PDA and their rollout counterparts (NN$_R$, PDA$_R$) which were presented in [23]. The same sequential cost function given in (22) is computed using a different approach. The authors offer using a base policy for approximating $J(\underline{\Gamma}^{2:N}; \hat{\underline{x}}^p_2(\underline{\Gamma}_1), \mathcal{Z}^{2:N})$ for every different value of $\underline{\Gamma}_1$. This base policy can be any suboptimal data association algorithm and regarding the performance of the selected base policy, the quality of the final solution changes. The reader can refer to [23] for further details.

In this paper, there is a major difference compared to [23] considering the selection of the base policy for the rollout algorithms. In a high clutter environment, we have realized that the main computational burden of the algorithms was due to the consideration of all possible partitions of the measurement set. That is to say, when there is more than one target in the surveillance region and if these targets share some of their measurements, multi-target tracking algorithms such as GNN or JPDA need to account for how the shared measurements will be utilized mainly to avoid track coalescence. Yet we have observed that, when we utilize single target tracking algorithms (PDA and NN) as the rollout policy, it is possible to reduce the computational burden of the algorithm drastically without a major degradation at the performance. For a fair comparison, batch estimation algorithms such as K-best MHT should also be included in the comparison. However, in [23], it was discussed that the rollout versions for NN and PDA outperform the K-best version of the MHT algorithm. Therefore, it was seen sufficient to do the comparison with the aforementioned algorithms.

The average performance measures obtained in the MC runs are shown in Tables. Tables I and II show the effect of increasing clutter density, while Table I and III show the impact of increasing the measurement noise under moderate clutter. The final column of each table shows the results when there is no clutter in the scenario, which can be seen as a lower bound for the RMSE results. The performance measures used in this study are taken from [30] with a few exceptions. The used measures can be given as follows;

- Number of Valid Tracks (NVT): A track is validated if it is assigned to the target. Track assignment is made if $(\hat{\underline{x}}^e_k - \underline{x}_k)'(\mathbf{C}^e_k)^{-1}(\hat{\underline{x}}^e_k - \underline{x}_k) \leq \gamma_G$ for $\gamma_G = \text{chi2inv}(P_G, 4)$ where $\underline{x}_k$ is the real state of the target. In ideal case, there should be only one track for each target.
- Number of False Tracks (NFT): A track is detected as a false one if it is not assigned to the target. A smaller quantity is better.
- Root Mean Squared Error (RMSE): If for any time instant a target is associated to the track, the l_2 distance between them are calculated. A smaller quantity is better.
- Target Coverage (TCvr): Target coverage shows in what percentage of the scenario, a track is assigned to the true target. This is not defined in [30]. A bigger quantity is better.
- Total Execution Time (TET): The runtime of different algorithms are measured for the same set of measurements. A smaller quantity is better.
- Rate of False Alarm (RFA): The average number of false tracks at each time instant of the scenario. A smaller quantity is better.

TABLE I: Performance measures for 100 trials with $\sigma = 20\,m$, $\beta_{FA} = 10^{-5}m^{-2}$ and $N = 5$

	NN	PDA	NN$_R$	PDA$_R$	RPDA	$\beta_{FA} = 0$
NVT	1	1	1	1	1	1
NFT	2.167	2.167	2.167	2.167	2.200	0
RMSE	14.45	14.21	13.81	13.87	**13.61**	**13.97**
TCvr	0.98	0.98	0.98	0.98	0.98	0.98
TET	1.49	1.47	1.83	1.84	1.97	0.04
RFA	0.108	0.108	0.091	0.091	0.087	0

TABLE II: Performance measures for 100 trials with $\sigma = 20\,m$, $\beta_{FA} = 2 \times 10^{-5}m^{-2}$ and $N = 5$

	NN	PDA	NN$_R$	PDA$_R$	RPDA	$\beta_{FA} = 0$
NVT	1.360	1.080	1.080	1.140	1.120	1.000
NFT	89.7	87.9	87.1	87.2	95.1	0
RMSE	16.63	17.81	15.24	15.45	**14.71**	**13.23**
TCvr	0.82	0.87	0.88	0.88	0.87	1
TET	5.68	5.75	8.14	8.07	6.91	0.04
RFA	5.78	6.00	6.23	6.27	4.07	0

TABLE III: Performance measures for 100 trials with $\sigma = 40\,m$, $\beta_{FA} = 10^{-5}m^{-2}$ and $N = 5$

	NN	PDA	NN$_R$	PDA$_R$	RPDA	$\beta_{FA} = 0$
NVT	1	1.03	1.06	1.06	1.03	1
NFT	93.35	84.3	84.23	84.06	100.51	0
RMSE	23.28	22.69	22.41	22.04	**21.65**	**22.39**
TCvr	0.95	0.99	0.98	0.98	0.98	0.98
TET	3.02	3.05	12.91	14.44	10.39	0.042
RFA	8.47	9.08	10.33	10.36	6.75	0

It can be seen from Table I that under moderate clutter, the performance of different algorithms are similar. There is always a track associated to the target. The number of

false tracks created by each scenario and the rate of false alarms are almost the same. This can be explained in the following way; due to low clutter density, there is almost no data ambiguity. Therefore each algorithm is basically able to find the correct measurement and perform the measurement update. This is also visible from the total execution time of the algorithms. When the data ambiguity is low, the update steps are performing the same operations, resulting is similar execution time results. One careful observation here would be about the RMSE performance of the proposed algorithm. Note that, the final result is even better in average compared to the error results when no clutter exists. This is due to the following fact; the real lower bound in this case should be the error of the Kalman smoother. However, as the data association ambiguity is not resolved for the time steps $k+1$ to $k+N$, it is not possible to directly use the Kalman smoother results. We are using the proposed algorithm to smooth the data association probabilities at time k. Therefore, we find it fair to provide the results of the Kalman filter as the lower bound.

When β_{FA} is increased as given in Table II, it can be seen that the a huge number of false tracks are created and at each time step there is an average of almost 6 false tracks. It should also be noted that the target coverage decreases for all of the scenarios due to the number of high clutter that should be processed at every time step. The difference between the RMSE results for different algorithms become evident for this case. While algorithms using recursive estimation, namely NN and PDA, give the worst results, the rollout algorithms also suffer from the hard decision process involved in their calculation. The novel algorithm RPDA can result in a lower RMSE with a similar target coverage rate due to the soft decision process. Moreover, the computational advantage of convex optimization to the rollout strategy is evident from the total execution time results. Computation of the matrices for the overall cost given in (19) is the computational drawback of the RPDA algorithm but then the computation of the overall cost function is straightforward. For the rollout based algorithms, it is necessary to use the base policy at each time step to give an association decision which turns out to have a higher cost compared to solving the relaxed convex optimization problem.

Finally, the effect of increasing the measurement noise can be seen from Table III. Even when the clutter level is moderate, the number of false tracks is higher compared to Table II as the tracker also takes into account a higher measurement noise. The target is totally covered almost for every scenario. It should be noticed that the algorithms using the current information for inference of the past data association probabilities become more susceptible to false tracks. This is an expected but unwanted trade-off of the proposed algorithm. Not only more false tracks are created but also the created tracks become resilient as the false alarm rates are investigated. The RMSE results have increased in general due to increasing measurement noise, yet the proposed algorithm is able to attain the best results.

V. Conclusions

In this paper, we have dealt with the single target tracking problem under clutter with no missed detections. To avoid track loss in such a scenario, a common approach is to wait for a predefined horizon to give the association decisions. Yet, number of possible association decision sequences increase exponentially with time and finding the optimal solution is an NP-hard problem.

The main contribution of this paper its formulation of the data association problem in case of batch measurements. It is shown that finding the optimal data association sequence is equivalent to solving a constrained optimization problem with a quadratic cost function. As this is a costly operation, the convex relaxation of this optimization problem is solved which gives a lower bound for the data association costs. This lower bound is then used for the retrodiction of the data association probabilities in a past time step. It was shown by the experiments that the computational requirements are almost equivalent to recursive estimation methods with a remarkable gain in the RMSE.

Appendix A
Derivation of (18)

It is possible to write,

$$\underline{\hat{x}}^p_{k+1} = \mathbf{A}_{k+1}\underline{\hat{x}}^e_k \tag{27}$$

$$= \mathbf{D}_k\underline{\hat{x}}^p_k + \mathbf{A}_{k+1}\mathbf{K}_k\mathbf{Z}_k\underline{\Gamma}_k \tag{28}$$

$$= \mathbf{M}^k_x\underline{\hat{x}}^p_1 + \sum_{i=1}^{k}\mathbf{M}^k_{z,i}\underline{\Gamma}_i \tag{29}$$

where

$$\mathbf{D}_i = \mathbf{A}_{i+1}(I - \mathbf{K}_i\mathbf{H}_i) \tag{30}$$

$$\mathbf{M}^k_{z,i} = \left(\prod_{j=1}^{k-i}\mathbf{D}_{k-j}\right)\mathbf{A}_{i+1}\mathbf{K}_i\mathbf{Z}_i \tag{31}$$

$$\mathbf{M}^k_x = \prod_{i=0}^{k-1}\mathbf{D}_{k-i} \tag{32}$$

Although not mentioned in the formulas, the means of the state estimates $\underline{\hat{x}}^p_{k+1}$ and $\underline{\hat{x}}^e_k$ are dependent on the association decision sequence Θ^k. Defining

$$\mathbf{M}^k_z = \begin{bmatrix}\mathbf{M}^k_{z,1} & \dots & \mathbf{M}^k_{z,k}\end{bmatrix},$$

we can write

$$\underline{\hat{x}}^p_{k+1} = \mathbf{M}^k_x\underline{\hat{x}}^p_1 + \mathbf{M}^k_z\underline{\Gamma}^k. \tag{33}$$

Appendix B
Derivation of (19)

Then instantaneous cost becomes,

$$\begin{aligned}
&c(\underline{\Gamma}_{k+1}; \underline{\hat{x}}^p_{k+1}, \mathcal{Z}_{k+1}) \\
&= (\mathbf{Z}_{k+1}\underline{\Gamma}_{k+1} - \mathbf{H}_{k+1}\underline{\hat{x}}^p_{k+1})'\mathbf{S}^{-1}_{k+1}(\mathbf{Z}_{k+1}\underline{\Gamma}_{k+1} - \mathbf{H}_{k+1}\underline{\hat{x}}^p_{k+1}) \\
&= (\mathbf{N}^{k+1}_x\underline{\hat{x}}^p_1 + \mathbf{N}^{k+1}_z\underline{\Gamma}^{k+1})'\mathbf{S}^{-1}_{k+1}(\mathbf{N}^{k+1}_x\underline{\hat{x}}^p_1 + \mathbf{N}^{k+1}_z\underline{\Gamma}^{k+1})
\end{aligned}$$

where $\mathbf{N}_x^{k+1} = -\mathbf{H}_k\mathbf{M}_x^k$ and $\mathbf{N}_z^{k+1} = \begin{bmatrix}-\mathbf{H}_k\mathbf{M}_z^k & \mathbf{Z}_{k+1}\end{bmatrix}$. Notice that $\underline{\hat{x}}_{k+1}^p$ is a function of $\underline{\hat{x}}_1^p$ and $\underline{\Gamma}^k$. Now defining

$$\mathbf{L}_{x,x}^{k+1} = (\mathbf{N}_x^{k+1})'\mathbf{S}_{k+1}^{-1}(\mathbf{N}_x^{k+1}) \tag{34}$$

$$\mathbf{L}_{z,z}^{k+1} = (\mathbf{N}_z^{k+1})'\mathbf{S}_{k+1}^{-1}(\mathbf{N}_z^{k+1}) \tag{35}$$

$$\mathbf{L}_{x,z}^{k+1} = (\mathbf{N}_x^{k+1})'\mathbf{S}_{k+1}^{-1}(\mathbf{N}_z^{k+1}) \tag{36}$$

we arrive at

$$\begin{aligned} c(\underline{\Gamma}_{k+1};\underline{\hat{x}}_{k+1}^p, \mathcal{Z}_{k+1}) = (\underline{\hat{x}}_1^p)'\mathbf{L}_{x,x}^{k+1}\underline{\hat{x}}_1^p \\ + 2(\underline{\hat{x}}_1^p)'\mathbf{L}_{x,z}^{k+1}\underline{\Gamma}^{k+1} + (\underline{\Gamma}^{k+1})'\mathbf{L}_{z,z}^{k+1}\underline{\Gamma}^{k+1} \end{aligned} \tag{37}$$

The matrices given in the overall cost function (19) can be found by using (37). Assuming that the state vector $\underline{\boldsymbol{x}}_k$ has a dimension of n_x and the measurement vector $\underline{\boldsymbol{z}}_k$ has a dimension of n_z, define matrices $\mathbf{L}_{z,z} \in \mathbb{R}^{Nn_z \times Nn_z}$ and $\mathbf{L}_{x,z} \in \mathbb{R}^{Nn_x \times Nn_z}$ and write,

$$(\underline{\Gamma}^N)'\mathbf{L}_{z,z}\underline{\Gamma}^N = \sum_{k=1}^{N}(\underline{\Gamma}^k)'\mathbf{L}_{z,z}^k\underline{\Gamma}^k$$

$$(\underline{\hat{x}}_1^p)'\mathbf{L}_{x,z}\underline{\Gamma}^N = \sum_{k=1}^{N}(\underline{\hat{x}}_1^p)'\mathbf{L}_{x,z}^k\underline{\Gamma}^k$$

and finally

$$J(\underline{\Gamma}^N;\underline{\hat{x}}_1^p, \mathcal{Z}^N) = (\underline{\Gamma}^N)'\mathbf{L}_{z,z}\underline{\Gamma}^N + (\underline{\hat{x}}_1^p)'\mathbf{L}_{x,z}\underline{\Gamma}^N + f(\underline{\hat{x}}_1^p) \tag{38}$$

REFERENCES

[1] G. W. Pulford, "Taxonomy of multiple target tracking methods," in *Radar, Sonar and Navigation, IEE Proceedings-*, vol. 152. IET, 2005, pp. 291–304.

[2] R. P. Mahler, "Multitarget Bayes Filtering via First-Order Multitarget Moments," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 39, no. 4, pp. 1152–1178, 2003.

[3] B. N. Vo and W. K. Ma, "The Gaussian Mixture Probability Hypothesis Density Filter," *IEEE Transactions on Signal Processing*, vol. 54, no. 11, pp. 4091–4104, Nov. 2006.

[4] B. T. Vo, B. N. Vo, and A. Cantoni, "The cardinality balanced multi-target multi-Bernoulli filter and its implementations," *IEEE Transactions on Signal Processing*, vol. 57, no. 2, pp. 409–423, Feb. 2009.

[5] S. O. S. Oh, S. Russell, and S. Sastry, "Markov chain Monte Carlo data association for general multiple-target tracking problems," *Conference on Decision and Control (CDC)*, vol. 1, pp. 735–742, 2004.

[6] J. Sun, Q. Li, X. Zhang, and W. Sun, "An Efficient Implementation of Track-Oriented Multiple Hypothesis Tracker Using Graphical Model Approaches," *Mathematical Problems in Engineering*, vol. 2017, pp. 1–11, Sep. 2017.

[7] D. Musicki and B. La Scala, "Multi-target tracking in clutter without measurement assignment," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 44, no. 3, pp. 877–896, 2008.

[8] S. Blackman and R. Popoli, "Design and Analysis of Modern Tracking Systems," *Artech House, Norwood, MA*, pp. 967–1068, 1999.

[9] T. E. Fortmann, Y. Bar-Shalom, and M. Scheffe, "Multi-target tracking using joint probabilistic data association," *Decision and Control including the Symposium on Adaptive Processes, 1980 19th IEEE Conference on*, pp. 807–812, 1980.

[10] J. A. Roecker and G. I. Phillis, "Suboptimal joint probabilistic data association," *Aerospace and Electronic Systems, IEEE Transactions on*, vol. 29, no. 2, pp. 510–517, 1993.

[11] T. Kirubarajan and Y. Bar-Shalom, "Probabilistic data association techniques for target tracking in clutter," *Proceedings of the IEEE*, vol. 92, no. 3, pp. 536–556, 2004.

[12] D. Reid, "An algorithm for tracking multiple targets," *IEEE Transactions on Automatic Control*, vol. 24, no. 6, pp. 843–854, Dec. 1979.

[13] S. Blackman, "Multiple hypothesis tracking for multiple target tracking," *IEEE Aerospace and Electronic Systems Magazine*, vol. 19, no. 1, pp. 5–18, 2004.

[14] R. L. Streit and T. E. Luginbuhl, "Maximum likelihood method for probabilistic multihypothesis tracking," in *SPIE's International Symposium on Optical Engineering and Photonics in Aerospace Sensing*. International Society for Optics and Photonics, 1994, pp. 394–405.

[15] H. Gauvrit, J. P. L. Cadre, and C. Jauffret, "A formulation of multitarget tracking as an incomplete data problem," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 33, no. 4, pp. 1242–1257, Oct. 1997.

[16] L. Xinliang, L. Zhi-Quan, K. M. Wong, and E. Bosse, "An interior point linear programming approach to two-scan data association," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 35, no. 2, pp. 474–490, Apr. 1999.

[17] D. Musicki, T. L. Song, and T. H. Kim, "Smoothing multi-scan target tracking in clutter," *IEEE Transactions on Signal Processing*, vol. 61, no. 19, pp. 4740–4752, 2013.

[18] S. Memon, T. L. Song, and T. H. Kim, "Smoothing data association for target trajectory estimation in cluttered environments," *Eurasip Journal on Advances in Signal Processing*, vol. 2016, no. 1, pp. 1–13, 2016.

[19] K. R. Pattipati, S. Deb, Y. Bar-Shalom, and R. B. Washburn, "A new relaxation algorithm and passive sensor data association," *IEEE Transactions on Automatic Control*, vol. 37, no. 2, pp. 198–213, Feb. 1992.

[20] A. P. Poore and N. Rijavec, "A Lagrangian Relaxation Algorithm for Multidimensional Assignment Problems Arising from Multitarget Tracking," *SIAM Journal on Optimization*, vol. 3, no. 3, pp. 544–563, Aug. 1993.

[21] M. F. Huber, "On multi-step sensor scheduling via convex optimization," in *2010 2nd International Workshop on Cognitive Information Processing, CIP2010*. IEEE, Jun. 2010, pp. 376–381.

[22] S. Joshi and S. Boyd, "Sensor Selection via Convex Optimization," *IEEE Transactions on Signal Processing*, vol. 57, no. 2, pp. 451–462, Feb. 2009.

[23] S. Özgen, M. Demirekler, and U. Orguner, "On the Solution of Data Association Problem using Rollout Algorithms," in *Proc. FUSION-16*, 2016, p. 8.

[24] D. P. Bertsekas and J. N. Tsitsiklis, "Neuro-dynamic programming: an overview," in *Decision and Control, 1995., Proceedings of the 34th IEEE Conference on*, vol. 1. IEEE, 1995, pp. 560–564.

[25] Y. Ye, "On the complexity of approximating a KKT point of quadratic programming," *Mathematical Programming*, vol. 80, no. 2, pp. 195–211, Jan. 1998.

[26] Y. Ye and E. Tse, "An extension of Karmarkar's projective algorithm for convex quadratic programming," *Mathematical Programming*, vol. 44, no. 1-3, pp. 157–179, May 1989.

[27] S. P. Boyd and L. Vandenberghe, *Convex optimization*, 18th ed. Cambridge: Cambridge Univ. Press, 2015, oCLC: 767754283.

[28] G. Pulford and B. L. Scala, "Over-The-Horizon Radar Tracking Using The Viterbi Algorithm," DSTO, Tech. Rep., 1995.

[29] Y. Bar-Shalom, T. Kirubarajan, and C. Gokberk, "Tracking with classification-aided multiframe data association," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 41, no. 3, pp. 868–878, 2005.

[30] A. A. Gorji, R. Tharmarasa, and T. Kirubarajan, "Performance measures for multiple target tracking problems," in *14th International Conference on Information Fusion*, Jul. 2011, pp. 1–8.