Deterministic Gibbs Sampling for Data Association in Multi-Object Tracking

Laura M. Wolf¹ and Marcus Baum¹

Abstract—In multi-object tracking multiple objects generate multiple sensor measurements, which are used to estimate the objects' state simultaneously. Since it is unknown from which object a measurement originates, a data association problem arises. Considering all possible associations is computationally infeasible for large numbers of objects and measurements. Hence, approximation methods are applied to compute the most relevant associations. Here, we focus on deterministic methods, since multi object tracking is often applied in safety-critical areas. In this work we show that Herded Gibbs sampling, a deterministic version of Gibbs sampling, applied in the Labeled Multi Bernoulli filter, yields results of the same quality as randomized Gibbs sampling while having the same computational complexity. We conclude that it is a suitable deterministic alternative to randomized Gibbs sampling and could be a promising approach for other data association problems.

I. INTRODUCTION

The tracking of multiple moving objects based on measurements, Multi-Object Tracking (MOT), is a highly investigated field with safety relevance in application areas such as aerial and maritime surveillance as well as driver assistance systems and autonomous vehicles in the automotive area.

The measurements are obtained from miscellaneous sensors, e.g., radar, lidar or sonar sensors. Due to the respective sensor technology and environmental factors, the measurements are in general noise corrupted. Hence, existing objects are not always detected and false alarms can occur caused by clutter detection. Furthermore, in most scenarios the number of objects is unknown and dynamic, and it is unknown from which object a measurement originated, see Fig. 1, since there could be multiple measurements close to one object (e.g., object 1 and measurements 1 and 2) or multiple predicted objects close to the same measurement (e.g., objects 3 and 4 and measurement 4). These unknown quantities lead to a data association problem, which has exponential complexity [1].

In this work, we consider classical MOT, where each object is assumed to generate at most one measurement per scan.

There exist numerous approaches to MOT in literature. Most MOT algorithms are based on Joint Probabilistic Data Association (JPDA) [2], Multi Hypothesis Tracking (MHT) [3], Probabilistic Multi Hypothesis Tracking (PMHT) [4] or Random Finite Sets (RFSs), where RFS based filters such

Fig. 1. MOT scenario with four cars (boxes) and four measurements (crosses).

as the Probability Hypothesis Density (PHD) filter [5] are well suited to handle object appearances and disappearances. A prime example of RFS trackers is the Labeled Multi Bernoulli (LMB) filter [6]. It considers all possible track-to-measurement associations as well as all possibilities of an object disappearing or persisting in the update step. The number of possible associations grows exponentially [6], which makes it computationally infeasible to exactly compute the posterior for growing numbers of objects or measurements.

So far, two approaches to approximate the LMB posterior are common. Firstly, Murty's Algorithm, which calculates the T best associations, and secondly Gibbs sampling, which generates associations based on their conditional likelihood [7]. Both algorithms are suitable approximations, but the Gibbs Sampler clearly outperforms Murty's Algorithm regarding computation time, due to lower complexity [7]. Moreover, Gibbs sampling relies on the generation of random samples, thus producing nondeterministic results.

With MOT algorithms employed for safety critical functionalities such as navigation, nondeterminism can lead to other problems regarding repeatability and validation [8]. Drawbacks of randomized approaches are, for example, that corner test cases might be hard to generate and that it is not sure whether a method will pass a test again, even if it has already passed it. Higher test capacities are required to calculate failure probabilities. Especially in the automotive area these concerns and the testability of systems pose a critical challenge due to the vast amount of scenarios in the real world [8]. Furthermore, due to the application of pseudonumber generators, randomized algorithms are not truly random and can have unforeseen side effects.

To avoid these issues, this paper focuses on deterministic methods to approximate the LMB posterior. Besides Murty's

*This work was supported by Zentrum für digitale Innovationen Niedersachsen (ZDIN), Zukunftslabor Mobilität.

¹Data Fusion Group, Institute of Computer Science, University of Göttingen, Germany laura.wolf@cs.uni-goettingen.de, marcus.baum@cs.uni-goettingen.de

Algorithm, we will investigate Roecker's Algorithm [9], [10], a heuristic to generate joint associations for the JPDA filter, and Herded Gibbs sampling [11], where a deterministic sampling method known as "herding" [12] has been applied to Gibbs sampling. The deterministic Herded Gibbs has already proven to outperform randomized Gibbs sampling in image denoising and named entity recognition [11].

Our paper is structured as follows: Section II defines the problem, Section III presents the different association methods, followed by an evaluation on simulated data in Section IV.

II. PROBLEM FORMULATION

In MOT algorithms multiple hypotheses regarding which measurement originated from which object are considered. A joint association θ maps objects to measurements. Since each object can generate at most one measurement and each measurement can only stem from one object, this is a one on one mapping. Since an object can disappear or be undetected even if it exists, the possibilities of object disappearance and missed detection are added to the joint association, however these states are not exclusive, since more than one object can disappear or be undetected in one time step. With $\{1, \dots, P\}$ the set of predicted objects and $\{1, \dots, M\}$ the set of measurements, a joint association $\theta : \{1, \dots, P\} \mapsto \{-1, \dots, M\}$ can be defined as follows:

$$\theta = (\theta_1, \dots, \theta_P), \quad (1)$$

where θ_i has value -1 if object i does not exist, 0 if it exists, but is undetected, and value $j \in \{1, \dots, M\}$, if it is associated with measurement j . Measurements to which no objects are assigned are considered to be clutter.

Some joint associations are more likely than others, e.g., $\theta = (1, 0, 0, 4)$ and $\theta' = (2, -1, 4, 0)$ are more likely than $\theta'' = (4, 2, 3, 0)$. Due to the exponentially increasing number of possible joint associations, it could be advantageous to only consider likely associations in order to decrease complexity. However, depending on noise and situation, unlikely associations may still occur, leading to a tradeoff between accuracy and complexity.

The likelihood of a joint association θ depends on the likelihood η_{i, θ_i} for a single association θ_i . For the joint prediction and update scheme for the LMB filter [7], which estimates object disappearances and missed detections jointly with measurement associations in the LMB filter, η is defined as:

$$\eta_{i, j} = \begin{cases} 1 - r_i p_S, & j = -1, \\ (1 - p_D) r_i p_S, & j = 0, \\ l_{i, j} r_i p_S, & j > 0, \end{cases} \quad (2)$$

for $1 \leq i \leq P$ and $-1 \leq j \leq M$, where r_i is the existence probability of object i , p_S is the survival probability, p_D the detection probability and $l_{i, j}$ the likelihood that measurement j was generated by object i .

Applying (2) the weight of association θ can be computed as

$$w^{(\theta)} = \prod_{i \in P} \eta_{i, \theta_i}. \quad (3)$$

Given a set Θ of joint associations, the update step can be computed considering only these instead of all possible associations. The LMB update, computing the updated existence probability r_i and spatial distribution $p_i(x)$ of object i , would then be given by the following:

$$r_i = \sum_{\theta \in \Theta, \theta_i \geq 0} \bar{w}^{(\theta)}, \quad (4)$$

$$p_i(x) = \frac{1}{r_i} \sum_{\theta \in \Theta, \theta_i \geq 0} \bar{w}^{(\theta)} p_i^{(\theta)}(x), \quad (5)$$

where $\bar{w}^{(\theta)}$ are the normalized weights of the associations and $p_i^{(\theta)}(x)$ is the posterior distribution of object i , given association θ . Note that for updating object i , only joint associations where it exists are considered.

Now, the complexity of the LMB update is determined by the size of Θ , which contains different associations depending on the algorithm applied.

III. DETERMINISTIC ASSOCIATION METHODS

In this section, three deterministic algorithms to compute joint assignments will be presented, namely Murty's Algorithm, Roecker's Algorithm and Herded Gibbs sampling.

A. Murty's Algorithm

Murty's Algorithm computes the T best solutions to the linear assignment problem, which can be solved optimally by the Hungarian method [13]. In order to apply Murty's Algorithm on the problem of finding the T best joint assignments, η is converted to an extended cost matrix as described in [14]. It has the shape $P \times 2P + M$ and consists of two square matrices, containing the likelihoods of object disappearance and missed detection on the diagonals, respectively. The off-diagonal entries are set to ∞ . The rest of the cost matrix contains the likelihoods of the measurements being generated by the objects (shape $P \times M$). By taking the negative logarithm of all likelihoods, the problem of maximizing the LMB weights $w^{(\theta)}$, which are products of the likelihood η , see (3), is converted to the problem of minimizing a sum. To compute the T best assignments, Murty's Algorithm divides the set of possible assignments into at most $P - 1$ disjoint subsets excluding the best assignment and computing the best assignment in each subset. Now the process is repeated with the next best assignment until T assignments are found. The Hungarian method can be implemented in $\mathcal{O}((2P + M)^3)$ [15], which yields a total complexity for Murty's Algorithm of $\mathcal{O}(T \cdot (2P + M)^4)$ [14]. Other ranked assignment algorithms such as [16] are able to compute the T best assignments even with cubic complexity.

B. Roecker's Algorithm

Roecker proposed a heuristic to compute joint associations for a near optimal JPDA [9], [10]. The algorithm operates on a sorted list of single associations L , in each step deleting one single association (see Algorithm 3). A joint assignment is generated using the remaining associations, this can be done in a greedy fashion, but it is also possible to compute the best assignment using the Hungarian method [13]. Note that

the removed single association can be considered again in the next step. This process can be repeated recursively, thus excluding pairs, triplets or higher tuples for each assignment, see Algorithms 1 and 2.

Algorithm 1 Roecker

Input: N_r, N_b, L
 Compute joint assignment of L
 Compute weight of assignment
for $i = 1$ **to** N_r **do**
 Call Remove_Multiple($0, i, N_b, L$)
end for

N_r is the number of single associations (e.g., single associations, tuples, triples) to be removed for each assignment, N_a states the index, after which single association should be removed from L , N_b states the number of removed associations, after which the loop should break. We added this constant, since the single association likelihoods in L decrease very fast and due to the greedy approach, the computed joint assignments do not differ, if the likelihoods of the excluded associations are too low. Therefore, complexity can be reduced by ending the loops earlier without considerable changes in approximation quality.

Algorithm 2 Remove_Multiple

Input: N_a, N_r, N_b, L
if $N_r = 1$ **then**
 Call Remove_One(N_a, N_b, L)
else
 for $j = N_a$ **to** $N_a + N_b$ **do**
 $\bar{L} = L$
 Flag j^{th} single association in $\bar{L}$ as removed
 if $N_r > 2$ **then**
 Call Remove_Multiple($j, N_r - 1, N_b, \bar{L}$)
 else
 Call Remove_One($j, N_b, \bar{L}$)
 end if
 end for
end if

Computing one greedy assignment has a complexity of $\mathcal{O}(P(M+2))$, thus, Roecker's Algorithm has a complexity of $\mathcal{O}(N_b^{N_r} P(M+2))$ for removing all tuples up to length N_r .

Algorithm 3 Remove_One

Input: N_a, N_b, L
for $i = N_a + 1$ **to** $N_a + 1 + N_b$ **do**
 $\bar{L} = L$
 Flag i^{th} single association in $\bar{L}$ as removed
 Compute joint assignment of $\bar{L}$
 Compute weight of assignment, if it is not a duplicate
end for

C. Herded Gibbs Sampling

By sampling joint assignments from their conditional probabilities, which can be inferred from η , Gibbs sampling has been successfully applied to decrease the complexity of the update steps of the LMB and Generalized Labeled Multi Bernoulli (GLMB) filters, but it is randomized. The work [11] applied the process of "Herding" to Gibbs sampling, thus creating a deterministic version of it. Herding [12] is a procedure to compute a sequence of pseudo-samples matching the moments μ of given data points. The t^{th} sample $\mathbf{x}^{(t)} \in \mathcal{X} \subseteq \mathbb{R}^n$ is computed as follows:

$$\mathbf{x}^{(t)} = \arg \max_{\mathbf{x} \in \mathcal{X}} \langle \mathbf{w}^{(t-1)}, \phi(\mathbf{x}) \rangle, \quad (6)$$

$$\mathbf{w}^{(t)} = \mathbf{w}^{(t-1)} + \mu - \phi(\mathbf{x}^{(t)}), \quad (7)$$

with a feature map $\phi : \mathcal{X} \rightarrow \mathcal{H}$ from the state space $\mathcal{X}$ to a Hilbert space $\mathcal{H}$ with inner product $\langle \cdot, \cdot \rangle$, the moment $\mu \in \mathcal{H}$ and the weight vector $\mathbf{w} \in \mathcal{H}$.

This procedure can be applied to deterministically generate samples from a normalized discrete probability distribution by setting μ equal to the probability distribution and the feature vector ϕ as the indicator vector, which consists of zeros and a single one at the position indicated by $\mathbf{x}$ [11].

In [11] herding is applied to Gibbs sampling [17] by generating deterministic samples from the conditional probability distributions. Herded Gibbs sampling requires to maintain a weight vector for each conditional distribution. Since there are P objects to sample associations for and for each of the other objects exist $M+2$ association possibilities, there are $P(M+2)^{(P-1)}$ conditional distributions to sample from. This resembles the worst case exponential space complexity of Herded Gibbs sampling. But as one object can only be assigned to one measurement, this number reduces drastically. Furthermore, the number of conditional distributions for which weight vectors have to be saved is limited by the number of samples drawn.

Algorithm 4 Herded Gibbs sampling

Input: η, T
 Initialize $\mathbf{x}^{(0)}$ with zeros.
for $t = 1$ **to** T **do**
 for $i = 0$ **to** P **do**
 Find $\mathbf{w}^{i, \bar{\mathbf{x}}}$, where
 $\bar{\mathbf{x}} := [\mathbf{x}_1^{(t)}, \dots, \mathbf{x}_{i-1}^{(t)}, \mathbf{x}_{i+1}^{(t-1)}, \dots, \mathbf{x}_P^{(t-1)}]$ is the current assignment of the other objects.
 if $\mathbf{w}^{i, \bar{\mathbf{x}}}$ has not been used **then**
 Initialize $\mathbf{w}^{i, \bar{\mathbf{x}}}$ with $\bar{\eta}_i / \sum_j \bar{\eta}_{i,j}$
 end if
 $\mathbf{x}_i^{(t)} = \arg \max_{\mathbf{x}} \mathbf{w}^{i, \bar{\mathbf{x}}}$
 $\mathbf{w}^{i, \bar{\mathbf{x}}} = \mathbf{w}^{i, \bar{\mathbf{x}}} + \bar{\eta}_i / \sum_j \bar{\eta}_{i,j}$ {with $\bar{\eta}$ defined in (8)}
 $\mathbf{w}_i^{i, \bar{\mathbf{x}}} = \mathbf{w}_i^{i, \bar{\mathbf{x}}} - 1$
 Save $\mathbf{w}^{i, \bar{\mathbf{x}}}$ to Hash Table.
 end for
end for
Output: $\mathbf{x}^{(1)}, \dots, \mathbf{x}^{(T)}$

The conditional probabilities $\bar{\eta}$ [14] of the joint distribution (3) are given by

$$\bar{\eta}_{i,j} \propto \begin{cases} 0, & \text{if } j > 0 \text{ and } j \text{ has been assigned,} \\ \eta_{i,j}, & \text{else.} \end{cases} \quad (8)$$

The probability that a measurement j is assigned to an object i is zero, if it is already associated with another object. Values -1 and 0 are excluded because they resemble object disappearance and missed detection, to which multiple objects can be assigned.

Looking up the weight vector $\mathbf{w}^{i,\bar{x}}$, see Algorithm 4, can be implemented in constant time using a Hash Table. If keeping track of assigned measurements, we can compute the conditional probabilities $\bar{\eta}$ in $\mathcal{O}(M+2)$, thus yielding a total complexity of $\mathcal{O}(TP(M+2))$ for computing T joint assignments. In contrast to Herded Gibbs sampling, Randomized Gibbs sampling does not maximize a weight vector, but samples directly from the conditional distribution $\bar{\eta}$, which can also be done in linear time. This results in the same complexity for both Gibbs Samplers.

IV. EVALUATION

In order to compare the different data association methods, we combined them with the LMB filter, which was implemented in Python, adapted from the MATLAB implementation¹ of [18] with Gaussian Mixture states and a constant velocity model. It was combined with the joint prediction and update described in [7]. We used fixed birth locations for simplicity, although an adaptive birth strategy would also be possible. The first test scenario consists of two objects crossing at time step 50. The second more complex example is defined as in [19], [18] and includes crossing and appearing/disappearing objects as well as objects from the same origin. Each association method was evaluated on 100 Monte Carlo runs of the filter. In order to compare the approximation accuracy of the different data association methods, the mean OSPA [20] metric with parameters $p = 1$ and $c = 100$ over all runs as well as the mean number of estimated objects over all runs were evaluated.

We evaluated Murty's Algorithm with 100 assignments, Herded Gibbs sampling as well as randomized Gibbs sampling with 1000 assignments and Roecker's Algorithm with $N_r = 2, 3$, $N_b = 10$ and greedy assignment computation. Using the Hungarian method instead of the greedy approach only slightly improved the results at the cost of a longer computation time and is therefore omitted here.

In order to compare the convergence behaviour of both Gibbs sampling methods, we evaluated the OSPA metric averaged over 100 Monte Carlo runs and all time steps of Herded Gibbs sampling and randomized Gibbs sampling on the complex example with a varying number of samples, $T \in \{10, 20, \dots, 100, 200, \dots, 1000\}$.

In both examples, Murty's Algorithm and both Gibbs samplers clearly outperform Roecker's Algorithm regarding the OSPA metric (Fig. 2a and 3a). Both tested variants of

Fig. 2. Comparison of different data association methods on cross example. Mean OSPA metric (a) and mean estimated cardinality (b) of 100 Monte Carlo runs of the LMB filter with different data association methods in a scenario where two objects cross at time step 50.

Roecker's Algorithm estimate too many tracks (Fig. 2b and 3b). This is caused by newly generated tracks in birth areas being assigned to clutter measurements, as can be seen in Fig. 4a. However, the true objects were correctly tracked. Increasing the length of removed tuples and therefore the number of generated assignments, does not necessarily yield better results, since Roecker2 estimated less clutter tracks in the cross example (Fig. 2a and 2b) and in the complex example before time step 20 (Fig. 3a and 3b).

There are almost no visible differences between Murty's Algorithm, randomized Gibbs sampling and Herded Gibbs sampling in both examples. All three of them show a very low OSPA error, as can be seen in Fig. 2a and 3a. Furthermore, the number of false tracks is very small in both examples (Fig. 2b and 3b). An example trace of Herded Gibbs sampling is given in Fig. 4b, proving that in this

¹<http://ba-tuong.vo-au.com/codes.html>

(a)

(b)

Fig. 3. Comparison of different data association methods on the complex example. Mean OSPA metric (a) and mean estimated cardinality (b) of 100 Monte Carlo runs of the LMB filter with different data association methods in a scenario with crossing, appearing and disappearing objects as well as objects emerging from the same origin.

simulation Herded Gibbs sampling estimated the objects correctly. Moreover, there are only a few false tracks, which are not established over a longer time period.

The OSPA error of Herded Gibbs sampling converges with a smaller number of samples than randomized Gibbs sampling (Fig. 5). Although the difference is very small, it also shows a slightly lower OSPA error in general. Therefore, Herded Gibbs sampling might be applied with a smaller number of samples than randomized Gibbs sampling, decreasing complexity. However, effective computation time heavily depends on the implementation.

V. CONCLUSION

The data association problem in MOT is a highly complex task with no easy solution. Conceptually simple heuristics

(a) Roecker2 greedy on cross example.

(b) Herded Gibbs on complex example.

Fig. 4. Exemplary single runs of LMB filter with Roecker2 greedy (a) and Herded Gibbs (b) on the cross and complex example, respectively. Clutter and object generated measurements of all time steps are shown at the same time. Estimated object trajectories are shown as colored lines.

Fig. 5. Comparison of both Gibbs sampling variants with varying numbers of samples T . OSPA averaged over all time steps and 100 Monte Carlo runs of the LMB filter on the complex example.

such as the greedy version of Roecker's Algorithm do not suffice to generate an adequate set of joint associations. Furthermore, the parameterization of Roecker's Algorithm seems to be of importance, since Roecker2 is the better choice in the Cross example and Roecker3 in the other. It could be investigated whether tuning N_r and N_b improves the results, although this requires extra effort and knowledge of the scenario. Moreover, the theoretical complexity is similar to the complexity of Herded Gibbs sampling, which clearly outperforms Roecker's Algorithm.

Herded Gibbs sampling achieves the same approximation quality as Murty's Algorithm and randomized Gibbs sampling while having the same complexity as randomized Gibbs sampling and furthermore being deterministic. The comparison of both Gibbs Samplers with varying sampling sized even suggests, that Herded Gibbs sampling could be applied with a smaller sampling size while yielding good approximation results.

REFERENCES

- [1] J. Vermaak, S. Godsill, and P. Perez, "Monte Carlo filtering for multi-target tracking and data association," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 41, no. 1, pp. 309–332, Jan. 2005.
- [2] T. Fortmann, Y. Bar-Shalom, and M. Scheffe, "Sonar tracking of multiple targets using joint probabilistic data association," *IEEE Journal of Oceanic Engineering*, vol. 8, no. 3, pp. 173–184, Jul. 1983.
- [3] D. Reid, "An algorithm for tracking multiple targets," *IEEE Transactions on Automatic Control*, vol. 24, no. 6, pp. 843–854, Dec. 1979.
- [4] R. L. Streit and T. E. Luginbuhl, "Probabilistic Multi-Hypothesis Tracking," Naval Underwater Systems Center Newport RI, Tech. Rep. NUWC-NPT-TR-10428, Feb. 1995.
- [5] R. Mahler, "Multitarget bayes filtering via first-order multitarget moments," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 39, no. 4, pp. 1152–1178, Oct. 2003.
- [6] S. Reuter, "Multi-object tracking using random finite sets," Dissertation, Universität Ulm, Sep. 2014.
- [7] S. Reuter, A. Danzer, M. Stübler, A. Scheel, and K. Granström, "A Fast Implementation of the Labeled Multi-Bernoulli Filter Using Gibbs Sampling," in *2017 IEEE Intelligent Vehicles Symposium (IV)*, Jun. 2017, pp. 765–772.
- [8] P. Koopman and M. Wagner, "Challenges in Autonomous Vehicle Testing and Validation," *SAE International Journal of Transportation Safety*, vol. 4, no. 1, pp. 15–24, 2016.
- [9] J. A. Roecker, "A class of near optimal JPDA algorithms," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 30, no. 2, pp. 504–510, Apr. 1994.
- [10] K. Romeo, D. F. Crouse, Y. Bar-Shalom, and P. Willett, "The JPDAF in practical systems: Approximations," in *Signal and Data Processing of Small Targets 2010*, vol. 7698. International Society for Optics and Photonics, Apr. 2010, p. 76981I.
- [11] Y. Chen, L. Bornn, N. De Freitas, M. Eskelin, J. Fang, and M. Welling, "Herded Gibbs Sampling," *The Journal of Machine Learning Research*, vol. 17, no. 1, pp. 263–291, Jan. 2016.
- [12] M. Welling, "Herding dynamical weights to learn," in *Proceedings of the 26th Annual International Conference on Machine Learning*, ser. ICML '09. Montreal, Quebec, Canada: Association for Computing Machinery, Jun. 2009, pp. 1121–1128.
- [13] H. W. Kuhn, "The Hungarian method for the assignment problem," *Naval Research Logistics Quarterly*, vol. 2, no. 1-2, pp. 83–97, 1955.
- [14] B. Vo, B. Vo, and H. G. Hoang, "An Efficient Implementation of the Generalized Labeled Multi-Bernoulli Filter," *IEEE Transactions on Signal Processing*, vol. 65, no. 8, pp. 1975–1987, Apr. 2017.
- [15] R. Jonker and T. Volgenant, "Improving the Hungarian assignment algorithm," *Operations Research Letters*, vol. 5, no. 4, pp. 171–175, Oct. 1986.
- [16] C. R. Pedersen, L. Relund Nielsen, and K. A. Andersen, "An algorithm for ranking assignments using reoptimization," *Computers & Operations Research*, vol. 35, no. 11, pp. 3714–3726, Nov. 2008.
- [17] S. Geman and D. Geman, "Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images," in *Readings in Computer Vision*, M. A. Fischler and O. Firschein, Eds. San Francisco (CA): Morgan Kaufmann, Jan. 1987, pp. 564–584.
- [18] S. Reuter, B. Vo, B. Vo, and K. Dietmayer, "The Labeled Multi-Bernoulli Filter," *IEEE Transactions on Signal Processing*, vol. 62, no. 12, pp. 3246–3260, Jun. 2014.
- [19] Ba-Tuong Vo, Ba-Ngu 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.
- [20] D. Schuhmacher, B. Vo, and B. Vo, "A Consistent Metric for Performance Evaluation of Multi-Object Filters," *IEEE Transactions on Signal Processing*, vol. 56, no. 8, pp. 3447–3457, Aug. 2008.