2017 International Conference on Control, Automation and Information Sciences (ICCAIS)
October 31-November 3, 2017, Chiang Mai, Thailand

OSPA$^{(2)}$: Using the OSPA Metric to Evaluate Multi-target Tracking Performance

Michael Beard
Department of Electrical and Computer Engineering
Curtin University
Bentley, WA 6102
michael.beard@curtin.edu.au

Ba Tuong Vo
Department of Electrical and Computer Engineering
Curtin University
Bentley, WA 6102
ba-tuong.vo@curtin.edu.au

Ba-Ngu Vo
Department of Electrical and Computer Engineering
Curtin University
Bentley, WA 6102
ba-ngu.vo@curtin.edu.au

***Abstract*—The optimal sub-pattern assignment (OSPA) metric is a distance between two sets of points that jointly accounts for the dissimilarity in the number of points and the values of the points in the respective sets. The OSPA metric is often used for measuring the distance between two sets of points in Euclidean space. A common example is in multi-target filtering, where the aim is to estimate the set of current target states, all of which have the same dimension. In multi-target tracking (MTT), the aim is to estimate the set of target tracks over a period of time, rather than the set of target states at each time step. In this case, it is not sufficient to analyse the multi-target filtering error at each time step in isolation. It is important that a metric for evaluating MTT performance accounts for the dissimilarity between the overall target tracks, which are generally of different dimensions. In this paper, we demonstrate that MTT error can be captured using the OSPA metric to define a distance between two sets of tracks.**

***Index Terms*—OSPA Metric, Multi-target Tracking, Performance Evaluation**

I. Introduction

The optimal sub-pattern assignment (OSPA) distance [1] is a metric that has been widely adopted as a method to evaluate multi-target filtering performance. Most commonly, it is applied by computing the distance between two multi-target states, usually a set of true target states and a set of estimated target states. The traditional method of applying OSPA is to evaluate it at each time instant, and to interpret the result as an instantaneous "per-target error", i.e. smaller OSPA distances indicate better overall filtering performance.

The OSPA distance has also been used in other applications, such as clustering of point patterns [2], [3] and multiple instance learning [4]. In [2] the OSPA distance was used to compute Bary-centers for grouping point pattern data into a given number of clusters. In [3], the OSPA distance was used in the affinity propagation algorithm for clustering point pattern data with unknown number of clusters. This approach was extended to other multiple instance learning tasks such as classification and anomaly detection in [4].

Many multi-target filtering algorithms also output a measure of their confidence in the estimated multi-target states, usually as covariance matrices. A method for incorporating this information in the OSPA distance was proposed in [5], where the authors use the Hellinger distance between two points, each augmented with corresponding covariance matrices, as the base distance for the OSPA.

In applications such as simultaneous localisation and mapping (SLAM), the OSPA metric can saturate to a limiting value irrespective of the cardinality errors, and it penalises missed detections and false alarms in an unequal manner. In [6], [7] the authors relaxed the "per-target error" concept and introduced a modification of the OSPA metric called the cardinalised optimal linear assignment (COLA) metric, as a complement to the OSPA metric for feature map evaluation. It was also demonstrated in [7] that the OSPA and COLA metric plots together provide an intuitive and robust evaluation of map estimation errors.

In contrast to multi-target *filtering*, which estimates the multi-target state at each time step, the aim of multi-target *tracking* (MTT) is to estimate the number of targets and their tracks, i.e. the set of target tracks. The OSPA distance between the estimated and true multi-target states at each time instant can provide some indication of tracking performance, and has found widespread use in the literature, however, it does not account for errors between the estimated and true sets of tracks. The main drawback of using the distance between multi-target states is that phenomena such as track switching and fragmentation are not penalised in a consistent manner.

The issue of measuring tracking (rather than filtering) performance using a mathematical metric was first discussed in [8], in which a metric based on the OSPA, called the OSPA-T metric, was proposed. Since the exact OSPA-T metric is computationally intractable in general, an approximation was also proposed in [8]. However, this approximation leads to a result which no longer satisfies the axioms of a true metric. In [9], the authors discussed a number of drawbacks of the approximate OSPA-T, and define another metric for sets of tracks. While the metric in [9] alleviates the drawbacks of the approximate OSPA-T, it is also intractable.

Keeping in mind that the OSPA metric is a distance between two general finite sets of points, in this paper, we show that it can be readily applied to measure MTT performance. If a satisfactory base distance can be defined between two tracks

978-1-5386-3114-0/17/$31.00 ©2017 IEEE

(of different lengths), then the OSPA distance between two sets of tracks can be directly applied, without any modifications. Such a base distance can be defined by making use of the OSPA distance for sets of points in Euclidean space. Accordingly, we refer to the OSPA distance between two sets of tracks using this base distance as the OSPA$^{(2)}$ (OSPA-on-OSPA) distance.

In Section II, we define the base distance between tracks as a weighted time-averaged OSPA between the target states arising from each track, which is then used directly in the context of the original OSPA formulation, thereby yielding a true metric on the space of finite sets of tracks. In Section III, we illustrate the behavior of the proposed performance evaluation technique on some synthetic test scenarios that include some common errors made by real-world tracking algorithms. These demonstrate that the proposed method is capable of capturing various types of tracking errors, which would go unnoticed if the OSPA metric were applied in the traditional manner.

II. OSPA$^{(2)}$: Optimal Sub-Pattern Assignment Distance for Sets of Tracks

A. Mathematical Formulation

We begin by letting $\mathbb{T} = \{1, 2, \ldots, K\}$ be a finite space of time indices, which includes all time indices from the beginning to the end of a scenario, and we let $\mathbb{X}$ be the single-object state space. Let $\mathbb{U}$ be the space of all functions mapping time indices in $\mathbb{T}$ to state vectors in $\mathbb{X}$, i.e. $\mathbb{U} = \{f : \mathbb{T} \mapsto \mathbb{X}\}$. Let $f_T \in \mathbb{U}$ be a function with domain $T \subseteq \mathbb{T}$, which we shall refer to herein as a *track*. The domain T of a track represents the set of all times at which the object exists, i.e. the object only exists at all time indices $t \in T$.

Remark 1. The true track of a target, referred to as a trajectory, always has support on a consecutive range of time indices. However, multi-target trackers (especially online trackers), do not necessarily give continuous track estimates, and may produce fragmented tracks due to disappearance/reappearance when the trackers revert to earlier decisions when new data arrives. Hence, our definition accommodates both (continuous) trajectories and (fragmented) tracks.

Let $\mathcal{F}(\mathbb{X})$ be the space of finite subsets of $\mathbb{X}$, and as defined in [1], let $d_p^{(c)}(\phi, \psi)$ be the OSPA distance between $\phi, \psi \in \mathcal{F}(\mathbb{X})$ with order p and cutoff c. That is, for $\phi = \left\{\phi^{(1)}, \phi^{(2)}, \ldots, \phi^{(m)}\right\}$ and $\psi = \left\{\psi^{(1)}, \psi^{(2)}, \ldots, \psi^{(n)}\right\}$, with $m \leq n$

$$d_p^{(c)}(\phi, \psi) = \left(\frac{1}{n}\left(\min_{\pi \in \Pi_n} \sum_{i=1}^{m} \bar{d}^{(c)}\left(\phi^{(i)}, \psi^{(\pi(i))}\right)^p + c^p (n - m)\right)\right)^{1/p} \tag{1}$$

where $\bar{d}^{(c)}\left(\phi^{(i)}, \psi^{(i)}\right) = \min\left(c, d\left(\phi^{(i)}, \psi^{(i)}\right)\right)$, in which $d(\cdot, \cdot)$ is a metric on the single-object state space $\mathbb{X}$. If $m > n$, then $d_p^{(c)}(\phi, \psi) \triangleq d_p^{(c)}(\psi, \phi)$.

1) Base Distance Between Tracks: We shall now make use of this to define a metric on the space of tracks $\mathbb{U}$, which shall serve as the base distance for the OSPA-on-OSPA, or OSPA$^{(2)}$, metric on the space of finite sets of tracks $\mathcal{F}(\mathbb{U})$. Let us define the distance $\tilde{d}_q^{(c)}(f_S, f_T; w)$ between two tracks $f_S, f_T \in \mathbb{U}$ as a convex combination of the OSPA distances between the set of states defined by f_S and f_T, at all times $t \in \{1, \ldots, K\}$, i.e.

$$\tilde{d}_q^{(c)}(f_S, f_T; w) = \left(\sum_{t=1}^{K}\left[w(t)\, d^{(c)}\left(\{f_S(t)\}, \{f_T(t)\}\right)\right]^q\right)^{1/q} \tag{2}$$

where $w(\cdot) > 0$ is a positive weighting function defined for $t \in \{1, \ldots, K\}$, such that $\sum_{t \in \mathbb{T}} w(t) = 1$, and q is a positive integer. Note that in equation (2), $\{f_S(t)\}$ is a singleton if $t \in S$, and $\{f_S(t)\} = \emptyset$ if $t \notin S$. In this case, the OSPA distance defined in equation (1) reduces to the following

$$d^{(c)}(\phi, \psi) = \begin{cases} 0, & |\phi| = |\psi| = 0 \\ c, & |\phi| \neq |\psi| \\ \min\left(c, d(\phi, \psi)\right), & |\phi| = |\psi| = 1 \end{cases}, \tag{3}$$

noting that the order parameter p is no longer required, and is consequently omitted from (3).

In order for $\tilde{d}_q^{(c)}(\cdot, \cdot; w)$ to be a metric on $\mathbb{U}$, the following four properties must be satisfied:

- $\tilde{d}_q^{(c)}(f_S, f_T; w) \geq 0$ (non-negativity),
- $\tilde{d}_q^{(c)}(f_S, f_T; w) = 0 \iff f_S = f_T$ (identity),
- $\tilde{d}_q^{(c)}(f_S, f_T; w) = \tilde{d}_q^{(c)}(f_S, f_T; w)$ (symmetry),
- $\tilde{d}_q^{(c)}(f_S, f_T; w) \leq \tilde{d}_q^{(c)}(f_S, f_R; w) + \tilde{d}_q^{(c)}(f_R, f_T; w)$ (triangle inequality).

Firstly, since $d^{(c)}(\cdot, \cdot) \geq 0$ and $w(\cdot) > 0$, the non-negativity property is clearly satisfied. Second, for $\phi, \psi \in \mathbb{X}$, since $d^{(c)}(\phi, \psi) = 0$ if and only if $\phi = \psi$, then $\tilde{d}_q^{(c)}(f_S, f_T; w) = 0$ if and only if $\{f_S(t)\} = \{f_T(t)\}$ since $w(t) > 0$ for all $t \in \mathbb{T}$. Thus the identity property is satisfied. Third, since $d^{(c)}(\cdot, \cdot)$ is symmetric in its arguments, $\tilde{d}_q^{(c)}(\cdot, \cdot; w)$ is also symmetric in its arguments, thus satisfying the third property. Finally, it can be easily demonstrated that the triangle inequality holds as a direct consequence of the Minkowski inequality, and the fact that $d^{(c)}(\cdot, \cdot)$ satisfies the triangle inequality. We have thus shown that all four properties hold, making $\tilde{d}_q^{(c)}(\cdot, \cdot; w)$ a metric on the space $\mathbb{U}$.

Remark 2. Since $d^{(c)}(\cdot, \cdot) \leq c$ and $w(\cdot) \leq 1$, the distance between tracks saturates at the value c, i.e. $\tilde{d}_q^{(c)}(\cdot, \cdot; w) \leq c$.

Remark 3. If we relax $w(\cdot)$ to allow $w(t) = 0$ for some values of $t \in \mathbb{T}$, then $f_S = f_T \implies \tilde{d}^{(c)}(f_S, f_T; w) = 0$ but not vice versa, thus making $\tilde{d}_q^{(c)}(\cdot, \cdot; w)$ a pseudo-metric instead of a full metric.

Remark 4. For two tracks $f_T^{(1)}$ and $f_T^{(2)}$ with the same domain T, setting $q = 2$ and $w(t) = \frac{1}{|T|}$ in (2) yields the root mean square error (RMSE) between $f_T^{(1)}$ and $f_T^{(2)}$. The base distance for $q = 2$ can thus be interpreted as a generalisation of the RMSE for tracks of different lengths and/or domains.

2) OSPA$^{(2)}$ for Tracks: The distance between two tracks (as just defined) is both a metric on the space $\mathbb{U}$, and capped at the value c, thus it is suitable to serve as a base distance for the OSPA metric on the space of finite sets of tracks $\mathcal{F}(\mathbb{U})$. Let $X = \left\{f^{(1)}, f^{(2)}, \ldots, f^{(m)}\right\} \subseteq \mathcal{F}(\mathbb{U})$ and $Y = \left\{g^{(1)}, g^{(2)}, \ldots, g^{(n)}\right\} \subseteq \mathcal{F}(\mathbb{U})$ be two sets of tracks, where $m \leq n$. Note that for clarity of notation, we have omitted the domain of each track, using the abbreviations $f^{(i)} = f^{(i)}_{S^{(i)}}$ and $g^{(i)} = g^{(i)}_{T^{(i)}}$. We define the distance $\check{d}^{(c)}_{p,q}(X, Y; w)$ between X and Y as the OSPA with base distance $\tilde{d}^{(c)}_q$ (the time averaged OSPA given by equation (2)). That is

$$\check{d}^{(c)}_{p,q}(X, Y; w) \tag{4}$$
$$= \left(\frac{1}{n}\left(\min_{\pi\in\Pi_n}\sum_{i=1}^{m}\tilde{d}^{(c)}_q\left(f^{(i)}, g^{(\pi(i))}; w\right)^p + c^p(n-m)\right)\right)^{1/p},$$

where c is the cutoff, p is the order, q is the order of the base distance, and w is a collection of convex weights. Note that if $m > n$, then $\check{d}^{(c)}_{p,q}(X, Y; w) \triangleq \check{d}^{(c)}_{p,q}(Y, X; w)$. We refer to this as the OSPA-on-OSPA, or OSPA$^{(2)}$ distance. Evaluating (4) involves the following four steps:

1) Choose an appropriate weighting function $w(\cdot) \geq 0$ such that $\sum_{t\in\mathbb{T}} w(t) = 1$.
2) Compute an $m \times n$ cost matrix C, where the j-th column of the i-th row is given by $C_{i,j} = \tilde{d}^{(c)}_q\left(f^{(i)}, g^{(j)}; w\right)$, according to (2).
3) Use a 2-D optimal assignment algorithm (such as Hungarian or auction) on the matrix C, to find the minimum cost assignment π of columns to rows in C.
4) Use the result of step 3 to evaluate $\check{d}^{(c)}_{p,q}(X, Y; w)$ according to (4).

The distance $\check{d}^{(c)}_{p,q}(X, Y; w)$ can be interpreted as the time-averaged per-track error.

B. Visualisation Methods

In practice, it is desirable to examine the tracking performance as a function of time, so that trends in the behaviour of tracking algorithms can be analysed in response to changing scenario conditions. This can be achieved by plotting

$$\alpha_k(X, Y) = \check{d}^{(c)}_{p,q}(X_k, Y_k; w_k) \tag{5}$$

as a function of k, where w_k is a weighting function at time k, and

$$X_k = \{f_T \in X | T \cap \text{supp}(w_k) \neq \emptyset\}, \tag{6}$$
$$Y_k = \{f_T \in Y | T \cap \text{supp}(w_k) \neq \emptyset\}. \tag{7}$$

Note that $X_k \subseteq X$ and $Y_k \subseteq Y$ are the subsets of tracks whose domain overlaps with the support of w_k. Two possibilities for the choice of weighting function are as follows.

1) *Expanding window*: At each time step k, the support of the weighting function includes all time indices from the beginning of the scenario up to k. For example
$$w_k(t) = 1_{\{1:k\}}(t)\frac{t^r}{\sum_{t=1}^{k} t^r} \tag{8}$$
where larger values of the parameter $r \geq 0$ place higher emphasis on more recent localisation errors, and less emphasis on localisation errors committed in the past. In this case, the base distance is a weighted time-averaged OSPA distance over the entire scenario up to the current time. Such a weighting function causes the metric to never forget past errors committed by the tracker. However, if the performance improves over time, the relative influence of earlier errors reduces due to the decreasing weight as we look further back in time.
2) *Sliding window*: Suppose at each time step k, we are only interested in the tracking performance over the previous N time steps, and we wish to progressively discount past errors as we move forward in time. This can be achieved by limiting the support of the weighting function as follows
$$w_k(t) = 1_{\{k-N+1:k\}}(t)\frac{(t+N-k)^r}{\sum_{t=k-N+1}^{k}(t+N-k)^r}, \tag{9}$$
which results in the base distance becoming a weighted time-averaged OSPA over the previous N time steps. As with the expanding window, larger values for $r \geq 0$ yield functions that place more emphasis on recent errors, and less on past errors. In this case, the metric forgets about any localisation and cardinality errors committed before time $k-N+1$. Although this method has the advantage that past cardinality errors can be forgotten, it may result in windowing artifacts that manifest themselves as discontinuities in the resulting OSPA$^{(2)}$ curve.

Remark 5. Note that computing the OSPA$^{(2)}$ (for sets of tracks) on a sliding window, as described in item 2 above, converges to the traditional OSPA (for sets of points) as the duration N becomes smaller. For $N = 1$ the OSPA$^{(2)}$ becomes identical to the traditional OSPA.

Other weighting functions may be chosen depending on the desired method of performance analysis. For example, it may be desirable to use uniform weights, such that past errors are given the same precedence as current errors (this corresponds to the case of $r = 0$ in equations (8) and (9)). The most appropriate type of weighting function depends on the situation at hand, and different choices can be used to demonstrate different aspects of the tracking performance.

It is important to understand that the OSPA-on-OSPA, or OSPA$^{(2)}$ distance, has a different interpretation to that of the traditional OSPA distance. Whereas the traditional OSPA distance captures the error between the true and estimated multi-target states at the current time step only, the OSPA$^{(2)}$ distance captures the error between the true and estimated sets of tracks over a range of time steps. The relative importance of the errors at each time are controlled via the weighting function w. Therefore, careful consideration must be given to the design of w, and the user must be mindful of this when interpreting the final result.

III. Example Scenarios

In this section, we demonstrate the behaviour of the OSPA$^{(2)}$ distance in analysing multi-target tracking performance, via some basic synthetic scenarios. In all cases, the ground truth consists of three trajectories, which occur over a period of 150 time steps. For the OSPA$^{(2)}$ calculations, the cutoff parameter is set to $c = 50$, the order parameters are $p = 1$ and $q = 2$, and we plot the results using a sliding window with weighting function as defined in equation (9) with $r = 3$. We compare the resulting OSPA$^{(2)}$ curves (for sets of tracks) with window lengths of $N = 75$, $N = 125$ and $N = 200$. We also compare the OSPA$^{(2)}$ with the traditional OSPA (for sets of points at each time step) to show that the proposed technique captures additional sources of error not considered by the traditional method.

A. Example Scenario 1

In this first example, we demonstrate the behaviour of the metric when the tracker experiences delays in track initiation and termination, which is common when tracking multiple targets in the presence of false alarms and misdetections. All three track estimates have a 10-step initiation delay, a 10-step termination delay, and a constant localisation error of 20m for the duration of the track. The scenario geometry is shown in Figure 1. Figure 2 shows the traditional OSPA for this scenario, and the OSPA$^{(2)}$ for the three different window lengths are shown in Figure 3.

The traditional OSPA shows increases corresponding to the track initiation and termination delays, and takes on the per-target localisation error of 20m at all other times. The OSPA$^{(2)}$ exhibits different behaviour, since each point in the curve captures the performance over a window of time instead of a single time step, which should be interpreted as the per-track per-unit-time error. The main features observable from plots in Figure 3 are explained by the following remarks.

- *Time 0 to 60:* There are three spikes of length 10, corresponding to the track initiation delays. The higher OSPA$^{(2)}$ during these spikes indicate cardinality errors, since the estimate set contains one less track than the ground truth over the window period. The second and third spikes exhibit a downward slope, because as time progresses, the weighting function increasingly discounts the errors due to the previous track initiation delays.
- *Time 60 to 110:* All three ground truth trajectories have corresponding estimates, thus the error steadily reduces. For the window length of 75, the OSPA$^{(2)}$ stabilises to the time-averaged per-track error of 20m, since the track initiation delays start to fall outside the window and are therefore forgotten. For the longer window lengths, the track initiation delays are not completely forgotten, hence the OSPA$^{(2)}$ does not reach the nominal localisation error of 20m. However, the value decreases steadily as the weighting function continues to discount the track initiation errors as time progresses.
- *Time 110 to 160:* There are three humps in the OSPA$^{(2)}$ curve, corresponding to the track termination delays. Notice that the height of these humps reduces with increasing window length. This is explained by the fact that the longer window length considers the cumulative tracking performance over a longer period of time. Since the base distance between trajectories is an average over the entire time window, if the window looks further into the past, the influence of the track termination delay is smaller relative to the errors committed over the rest of the time window.
- *Time 180 onward:* The first two OSPA$^{(2)}$ plots show three more spikes of length 10. These are due to the fact that the OSPA$^{(2)}$ is being computed over a sliding window, and they occur as the window moves past the end of the ground truth trajectories. As the window clears each ground truth trajectory, the cardinality error momentarily increases due to the track termination delay (since there is one more estimated trajectory than ground truth inside the window). Increasing the window length pushes these spikes further into the future, since the window takes longer to move past the end of each ground truth trajectory. For the window length of 200, these spikes will still be present, but they occur outside the plot range.

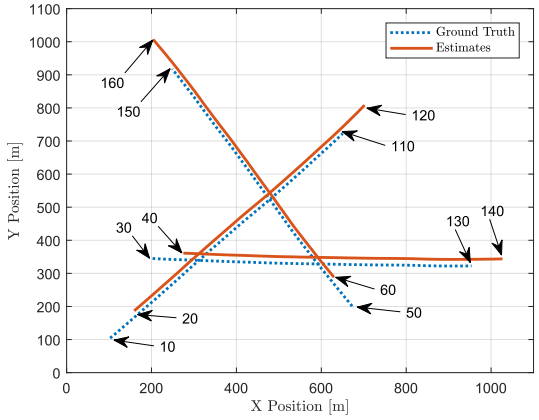

Fig. 1. Geometry for example scenario 1. The numbers indicate the start and end time indices of the ground truth trajectories and the estimated trajectories.

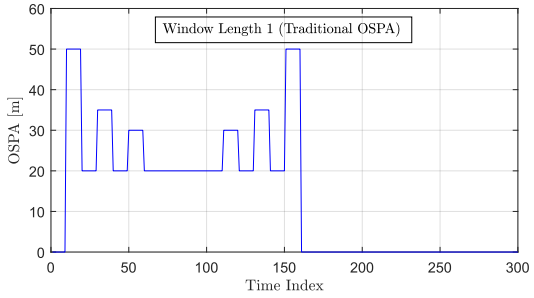

Fig. 2. Traditional OSPA distance for example scenario 1.

B. Example Scenario 2

This scenario illustrates the behaviour when the tracker temporarily drops the track estimates for two of the objects, but

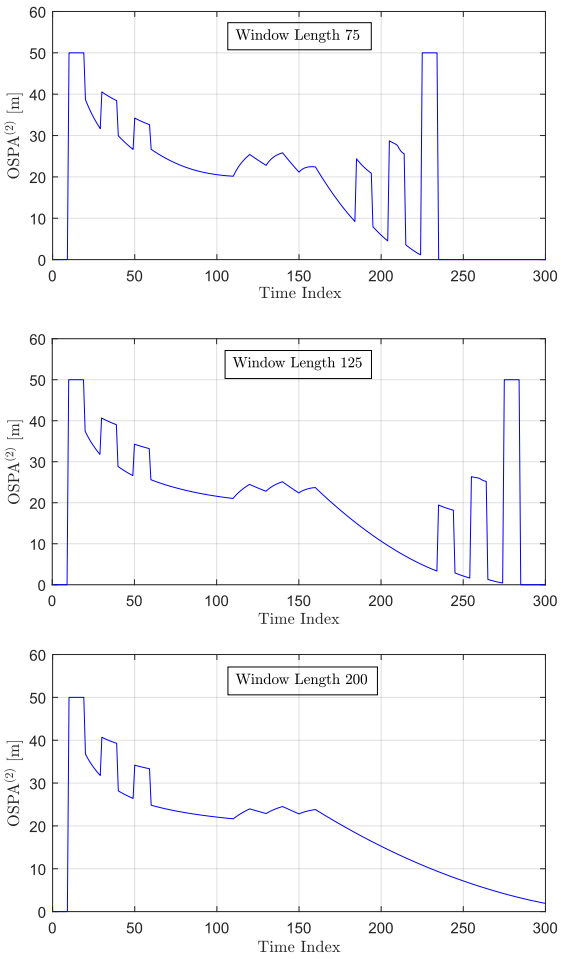

Fig. 3. OSPA$^{(2)}$ distance for example scenario 1.

recovers at a later time. One possible case is that the tracker continues the estimated trajectory with the same identity as originally assigned. Such a situation may arise when a tracker believes an object may have disappeared, but due to uncertainties in the observation process, it cannot not be certain about the disappearance. The tracker may thus retain the possibility that the object still exists, and if future observations confirm its existence, the track reappears with the same identity. This is shown in Figure 4, in which two of the tracks are dropped for a period of 10 steps each, one from time 70 to 80, and the other from time 90 to 100.

Another possibility is that instead of recovering the track with the same identity, the tracker assigns a new identity to the same object at a later time. This can arise when a tracker believes that an object may have disappeared, and due to limited processing or memory resources, quickly discards that track from its internal representation, foregoing any possibility that it still exists. Therefore, if evidence confirming the object's existence reappears, the tracker has no option but to establish a completely new track with a new identity. Intuitively, this behaviour should lead to a higher penalty than the case when the identity remains unchanged, since the tracker has failed to maintain a consistent estimate of object's identity over time. This is shown in Figure 5, in which two tracks are dropped and restarted with new identities 10 steps later.

The traditional OSPA is shown in Figure 6, and the OSPA$^{(2)}$ for the three different window lengths is shown in Figure 7. The traditional OSPA shows spikes for track initiation and termination, and also for the periods in which the two tracks are dropped. However, a significant drawback is that the case in which the track identities switch yields an identical result to the case in which the identities are correctly maintained. The OSPA$^{(2)}$ addresses this issue, and the following remarks explain the behaviour of the metric as observed in Figure 7.

- *Time 0 to 70:* All the OSPA$^{(2)}$ curves are the same as those in example scenario 1, since the scenarios are identical up until that time.
- *Time 70 to 100:* The OSPA$^{(2)}$ begins to increase at time 70 due to one of the estimated trajectories being dropped. In the case where the track identity stays the same, the OSPA$^{(2)}$ starts reducing at time 80 when the track is regained. The same behaviour is evident after time 90 when the second track is dropped and regained with the same identity. In the case where the track identity switches, the OSPA$^{(2)}$ exhibits a sudden jump at time 80 when the track is re-established with a new identity. This is due to a cardinality error, since there are now more distinct estimated tracks than ground truth trajectories. A similar jump is present at time 100 when the second dropped track is re-established with a new identity.
- *Time 110 to 160:* The three humps in the blue OSPA$^{(2)}$ curves are due to the track termination delays as described in example scenario 1. The orange curves also exhibit similar humps, however, the absolute value of the OSPA$^{(2)}$ is higher due to the cardinality errors associated with the track identity switches.
- For window length 75, the orange curve drops at time 145, and again at time 165. These are the times at which the initial track fragments for the dropped tracks begin to fall outside the time window. After this time, the metric forgets that the tracks were fragmented, and thus the OSPA$^{(2)}$ returns to the same value as if there were no identity switches. For the longer window lengths, these drops in the OSPA$^{(2)}$ curves are also evident, but they happen at later times. Thus, increasing the window length can be viewed as a means of increasing the penalty associated with track identity errors.
- *Time 180 onward:* The three spikes are due to windowing effects and the track termination delays as discussed in example scenario 1.

IV. Conclusion

We have discussed the use of the OSPA metric for multi-target tracking performance evaluation, which we have referred to as the OSPA-on-OSPA, or OSPA$^{(2)}$ distance. We illustrate that the OSPA$^{(2)}$ metric is capable of capturing tracking errors such as fragmentation and track switching. This is achieved by defining a base distance between two tracks as a weighted time-averaged OSPA distance between the target states arising

978-1-5386-3114-0/17/$31.00 ©2017 IEEE

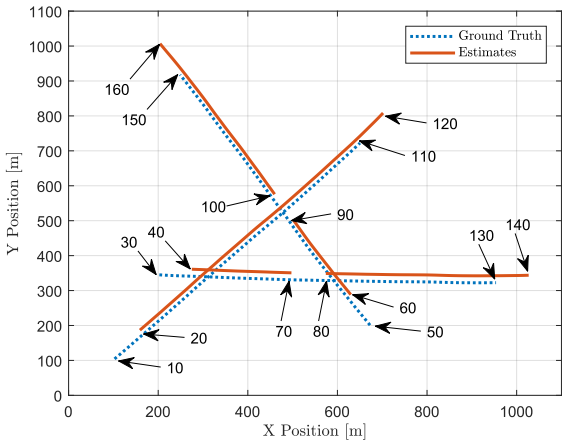

Fig. 4. Geometry for example scenario 2 with no track identity switches.

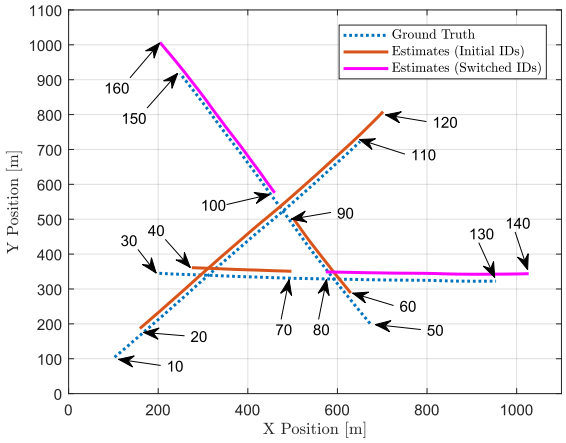

Fig. 5. Geometry for example scenario 2 with track identity switches.

from each track. The OSPA$^{(2)}$ distance between two sets of tracks follows naturally by substituting this base distance into the original OSPA formulation. The result is a metric in the true mathematical sense, which is simple to compute, and flexible enough to capture many important aspects of tracking performance which are not captured by the traditional OSPA distance between sets of points in Euclidean space.

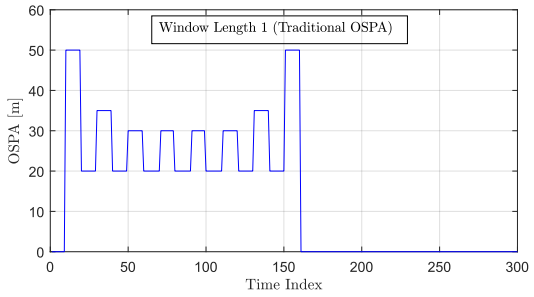

Fig. 6. Traditional OSPA distance for example scenario 2. Note that both cases (with and without identity switches) yield the same result.

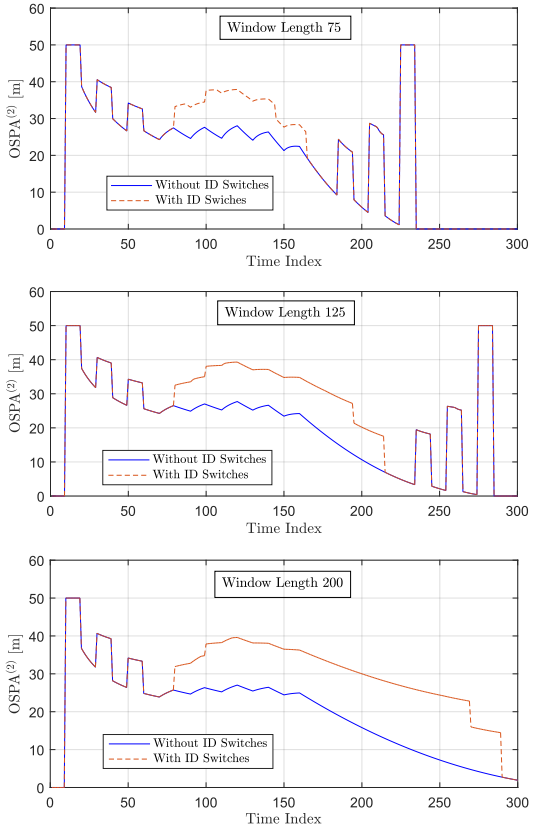

Fig. 7. OSPA$^{(2)}$ for scenario 2. The case without identity switches (solid blue lines) corresponds to the scenario in Figure 4, and the case with identity switches (dashed orange lines) corresponds to the scenario in Figure 5.

References

[1] D. Schuhmacher, B.-T. Vo, and B.-N. Vo, "A consistent metric for performance evaluation of multi-object filters," *IEEE Trans. Signal Processing*, vol. 56, no. 8, pp. 3447-3457, 2008.

[2] M. Baum, B. Balasingam, P. Willett and U. D. Hanebeck, "OSPA barycenters for clustering set-valued data," in *Proc. 18th Int. Conf. on Information Fusion*, pp. 1375-1381, Washington DC, USA, July 2015.

[3] Q. N. Tran, B.-N. Vo, D. Phung and B.-T. Vo, "Clustering for point pattern data," in *Proc. 23rd Int. Conf. on Pattern Recognition*, pp. 3174-3179, Dec. 2016.

[4] Q. N. Tran, B.-N. Vo, D. Phung, B.-T. Vo and T. Nguyen, "Multiple Instance Learning with the Optimal Sub-Pattern Assignment Metric," arXiv preprint arXiv:1703.08933, 2017.

[5] S. Nagappa, D. E. Clark and R. Mahler, "Incorporating track uncertainty into the OSPA metric," in *Proc. 14th Int. Conf. on Information Fusion*, Chicago, Illinois, USA, July 2011.

[6] P. Barrios, G. Naqvi, M. Adams, K. Leung and F. Inostroza, "The cardinalized optimal linear assignment (COLA) metric for multi-object error evaluation," in *Proc. 18th Int. Conf. on Information Fusion*, pp. 271-279), Washington DC, USA, July 2015.

[7] P. Barrios, M. Adams, K. Leung, F. Inostroza, G. Naqvi and M. E. Orchard, "Metrics for Evaluating Feature-Based Mapping Performance," *IEEE Trans. Robotics*, vol. 33, no.1, pp. 198-213, 2017.

[8] B. Ristic, B.-N. Vo, D. Clark, and B.-T. Vo, "A metric for performance evaluation of multi-target tracking algorithms," *IEEE Trans. Signal Processing*, vol. 59, no. 7, pp. 3452–3457, 2011.

[9] T. Vu and R. Evans, "A new performance metric for multiple target tracking based on optimal subpattern assignment," in *Proc.17th Int. Conf. on Information Fusion*, Salamanca, Spain, July 2014.