

## TRACKING OF MULTIPLE MERGING AND SPLITTING TARGETS: A STATISTICAL PERSPECTIVE

Curtis B. Storlie<sup>1</sup>, Thomas C. M. Lee<sup>2,3</sup>, Jan Hannig<sup>3</sup> and Douglas Nychka<sup>4</sup>

<sup>1</sup>*University of New Mexico*, <sup>2</sup>*The Chinese University of Hong Kong*,  
<sup>3</sup>*Colorado State University* and <sup>4</sup>*National Center for Atmospheric Research*

*Abstract:* This article considers the important problem of tracking multiple moving targets captured in image sequences. It has two primary objectives. The first is to serve as an introduction of the target tracking problem to the statistical community. It achieves this by providing a common definition of the tracking problem, a survey of important existing work, and a discussion of the relative advantages and shortcomings of such work. The second objective is to propose a statistical method for solving a wide class of tracking problems, namely, when the system of interest contains birth, death, merging and splitting of targets. The stochastic model behind this method is continuous time in nature and is equipped with a realistic mechanism for handling merging and splitting. Its finite sample properties are assessed via numerical experiments. Finally, the method is applied to two scientific problems for which it was originally designed: the tracking of (i) storms captured in radar reflectivity image data, and (ii) vortices from a high-resolution simulated vorticity field.

*Key words and phrases:* Convective systems, merging, multiple hypothesis tracking, multiple target tracking, splitting, track estimation, turbulence.

### 1. Introduction

Multiple target tracking is an important problem arising in many scientific and engineering investigations. It has importance in radar and signal processing, air traffic control, robot vision, GPS-based navigation, biomedical engineering, and video surveillance, to name a few. In this article we provide an introduction to, and propose a new statistical method for, multiple target tracking. Our work is motivated by the scientific need for storm tracking from radar reflectivity data and vortex tracking in turbulence fields. In these applications, the splitting and merging of targets are quite common, and such events are effectively and realistically accommodated by the proposed method. It is also our hope that this article will stimulate more statistical work in the interesting and exciting area of target tracking.

### 1.1. Problem Definition

Typically a complete tracking application is composed of two parts. The first is to extract the locations and/or other attributes of the targets from each image frame. There is no unified solution for this, as different targets need different methods for extraction. For example, human faces and missiles require very different target recognition methods to detect their appearances in an image. Once the target coordinates are located, the second part of the tracking application is to link these coordinates together so that coordinates of the same target detected at different image frames are connected to form a reconstruction of the path that this target traveled. In the tracking literature this second part of coordinate linking is commonly referred as the *data association problem*.

For the rest of this article we assume that the target coordinates and/or other useful attributes have already been extracted from the image sequence and focus on the second step, the data association. However, we do **not** assume that these targets are perfectly extracted. That is, we allow some of the real targets to remain undetected in some image frames (i.e., “*missing*” targets), we allow the presence of false alarms, and we also allow the target coordinates to be recorded with measurement errors. Furthermore, we also assume the occurrences of the following four events. First, we allow targets to appear for the first time or disappear permanently at any times during the image sequence; these events are called *birth* and *death*, respectively. Second, we permit situations under which two targets combine together to form a larger target; this is called *merging*. Lastly, targets are also allowed to break into smaller pieces; this is called *splitting*.

In summary, the tracking problem that this article considers is to, given the target information (coordinates and/or other attributes) acquired by the first extraction phase, recover the path, also known as the *track*, of each target traveled. In doing so, we allow birth, death, merging and splitting of targets. We also allow missing targets and false alarms. Our approach to this tracking problem is to first fit a stochastic model that incorporates all variables of interest, including times of birth, death, merging and splitting events, as well as target locations. We then estimate their conditional distribution given the data. Finally, the solution that corresponds to the mode of this estimated distribution is taken as the tracking estimate.

### 1.2. Review of the tracking problem

We first briefly review some popular methods for tackling the above tracking problem. The reader is also referred to the books by Bar-Shalom, Li and Kirubarajan (2001), Blackman and Popoli (1999), and Stone, Barlow and Corwin (1999) for a comprehensive description of modern tracking techniques.

The target tracking problem has been studied extensively in the engineering literature over the past thirty years. The approaches to this problem can

be loosely classified into two groups: non-statistical and statistical. The non-statistical group mainly use either image differencing techniques to detect target movements for consecutive images, as in Pece (2002), or heuristically minimize various objective functions that penalize the smoothness of the track estimates (e.g., Sethi and Jain (1987) and Salari and Sethi (1990)). Typically these non-statistical methods are fast and simple, and have been used with some success in the area of storm tracking (e.g., Johnson, Mackeen, Witt, Mitchell, Stumpf, Eilts and Thomas (1998), Wolfson, Forman, Hallowell and Moore (1999), Dixon (1994), Tuttle and Gall (1999), Lakshmanan, Rabin and DeBrunner (2003) and Hodges (1994, 1999)). However, they also possess a serious drawback: their inability to adequately handle birth, death, missing targets and false alarms. Moreover, these methods are only able to provide track estimates of the target movements and do not offer any mechanisms for assessing the associated uncertainties.

Hence, for the rest of this article, we focus our attention on the statistical approaches. The main idea is to employ a statistical model to describe the movements of the targets. Once a target model is proposed, the traditional approach to the data association problem is to then find the collection of tracks that maximizes the likelihood of the model. In the tracking literature such a collection of tracks is often called a *hypothesis*. To achieve such a likelihood maximization, two major issues are involved: (i) fast calculation of the likelihood value for any given hypothesis, and (ii) an effective search algorithm for locating the hypothesis that maximizes the likelihood function.

In the literature, typically a linear Gaussian state space model is applied to address the first issue, as it allows for efficient likelihood calculation of a given data association hypothesis via the Kalman filter. However, methods have also been developed for efficient likelihood calculation for more general cases. For examples, the extended Kalman filter (Anderson and Moore (1979)) is based on local linear approximations to a nonlinear system, and the unscented Kalman filter (Julier and Uhlmann (2004)) uses higher order approximation to calculate likelihood for nonlinear systems.

Most recently, research in this area has focused on a new class of filtering methods based on particle filtering or sequential Monte Carlo (Gordon, Salmond and Smith (1993), Kitagawa (1996), Liu and Chen (1998), Doucet, Godsill and Andrieu (2000) and Doucet, de Freitas and Gordon (2001)). This approach represents the target distribution with a set of samples, called particles, each with its own importance weight. These particles are then propagated through time to provide an approximation to the distribution at subsequent time steps. Liu and Chen (1998) develop a sequential importance sampling framework under which most of these procedures can be unified. Additionally Chen and Liu (2000) consider a class of models that are conditionally linear and Gaussian. Still other

methods such as the probability hypothesis density method (Mahler (2003) and Vo, Singh and Doucet (2005)) approximate the likelihood by propagating only the posterior expectation instead of the entire distribution through to subsequent times.

Assuming a method for fast likelihood calculation is available, the next step is to develop a search algorithm to find the data association hypothesis that maximizes the likelihood of the model. The most widely used heuristic search algorithm for this purpose is the Multiple Hypothesis Tracking (MHT) algorithm of Reid (1979). A brief description is given in Section 4.3. We have implemented a variant of this MHT algorithm for our likelihood maximization; see also Section 4.3.

There is also the Bayesian approach to the data association problem where a prior distribution is given to the possible data associations. Notice that in this setup the prior is imposed on the data associations but not on any model parameters. Usually this prior is an uninformative point uniform distribution over all of the possible associations. This is akin to assuming that the observations of targets/false alarms at each time are recorded in a random order. The solution is then to calculate the posterior distribution of the data association hypotheses given the observations. Since this is a very computationally intensive task, standard MCMC methods are generally too time consuming for many applications. To overcome this issue, several particle filter approaches have been suggested to take advantage of the sequential nature of the problem (Särkkä, Vehtari and Lampinen (2007), Kreucher, Kastella and Hero (2005) and Vermaak, Godsill and Perez (2005)). In Section 4 we have adopted an MHT approach to approximating the posterior distribution; this is similar in spirit to the ideas presented in Obermeyer and Poore (2004).

As mentioned earlier, this article is motivated by the need for tracking merging and splitting targets, such as storms or vortexes. Merging and splitting of targets can be common in radar applications as well, though in a slightly different context. That is, when two targets are close together, resolution limits may prevent them from being simultaneously detected. The detection method will then return only one (or even no) observation for these two targets. This certainly poses additional difficulties and challenges. Although this is perceived as a very important issue by Daum (1994) and Blackman (2004), we are unaware of any satisfactory solution to it. Most existing methods for tracking merging targets are not well defined in terms of an overall probabilistic model (Trunk and Wilson (1981), Chang and Bar-Shalom (1984), Koch and van Keuk (1997) and Genovesio and Olivo-Marin (2004)).

Up to now the most complete model for merging and splitting targets seems to be the one presented in Khan, Balch and Dellaert (2005, 2006). It is appropriate for the situation of unresolved radar measurements discussed above.

However, for the following reasons, this model is inadequate for representing the physical processes that we are studying in Section 2. First, it does not allow for the birth and death of targets. Second, no dependence structure is specified to describe the relationship between the merging events and the corresponding parent target locations. In other words, the model would allow for merging of targets that are distant apart. Lastly, its merging and splitting events are very temporary, in the sense that newly merged targets are as likely as any other targets to not be merged in the next image frame. This is unrealistic for many real problems where when targets merge, they tend to stay merged for an extended period of time. There is a clear need for a tracking method that can handle the above issues satisfactorily.

### 1.3. The proposed method

Our method for meeting these needs is based on a new continuous-time stochastic model that incorporates birth, death, merging and splitting of targets into the likelihood, as well as missing targets and false alarms. It also allows us to accommodate other target attributes (such as size or intensity) to improve the estimation results. Utilizing attribute information has been shown to increase tracking performance in other applications (e.g., Bal and Alam (2005), Roh, Kang and Lee (2000), Salmond and Parr (2003), and Angelova and Mihaylova (2006)). To the best of our knowledge, this is the first time all the above events are explicitly built into a stochastic model for tracking. We note that, although this model is fairly complex and contains different components, it was designed so that the resulting likelihood function can be expressed in closed form, and hence be computed efficiently.

The proposed model is continuous in time, which provides two additional advantages. First, it can be easily applied to irregularly sampled image sequences, and second, it allows the asymptotic properties of the tracking estimates to be studied when the sampling time converges to zero. Although it is beyond the scope of this article to examine these large sample properties, a theoretical justification for our tracking method is provided in Storlie, Hannig and Lee (2008).

When compared to existing methods, another benefit of our work is the way that we handle the static model parameters (e.g. rates of birth and death, noise variances, etc.). There are many approaches to particle filtering that allow for estimation of static model parameters along with the state variables (e.g., Andrieu and Doucet (2003), Doucet, de Freitas and Gordon (2003) and Doucet et al. (2001)). However, these rarely apply to multiple target tracking problems; most existing methods assume that many of the key model parameters are known. We provide consistent estimates of these model parameters and we calculate the distribution of the data associations given these estimates and the observations. Our approach might therefore be considered empirical Bayes where, just as in

the Bayesian approach to tracking, a uniform prior is imposed on the data associations but not on any of the model parameters.

The rest of this article is organized as follows. In Section 2 we describe two problems that motivate our work. The proposed stochastic model is presented in Section 3. Section 4 demonstrates how this model can be applied to provide track estimates of the targets. Finite sample performance of the methodology is then illustrated on simulated data in Section 5. Sections 6 and 7 report the tracking results obtained by applying the method to the original problems. Lastly, concluding remarks and possible future work are given in Section 8. We note that there is also an online supplementary document to this article, available at <http://www.stat.colostate.edu/~tlee/tracking/>. The supplement provides additional information that we refer to; references to it are always preceded by the letter S (e.g., equation (S.1) refers to the first equation in the supplementary document).

## 2. Two Motivating Problems

This section describes two problems for which the tracking of multiple targets is an important step to their solutions. Note that these examples are nonstandard tracking problems meant to serve as a clear illustration of our methodology, even though this results in difficulty levels that are not necessarily representative of current state-of-the-art tracking challenges.

### 2.1. Convective systems

Figure 1 shows radar reflectivity images evolving over time on July 14, 1996 from 1:00am to 3:30am. Radar reflectivity is correlated to rainfall intensity so that we can roughly attribute the variation of color in these images to different rainfall activities. In these images blue indicates 0 inches/hour of rainfall increasing, on a log scale, to bright yellow indicating  $\geq 2$  inches/hour. The images are separated by 30 minutes.

The merger of two systems that are located at the corner of South Dakota, Wyoming, and Nebraska can be seen in the first few images of Figure 1. These are clearly separate systems until the third image, where they are now one. There is also a splitting event in the panhandle of Florida as a fairly large system breaks apart into two smaller systems.

The targets that we wish to track are the larger convective systems. For our purposes, a convective system is defined to be a rainfall system that is larger than 100 km in length (approximately  $1^\circ$  of latitude or longitude). This problem has been studied previously by Davis, Manning, Carbone, Trier and Tuttle (2003). The very short term behavior (less than 1 hour) of such systems is reasonably well known, but the moderately short term (1 to 6 hours) and long term (1 to 2 days)

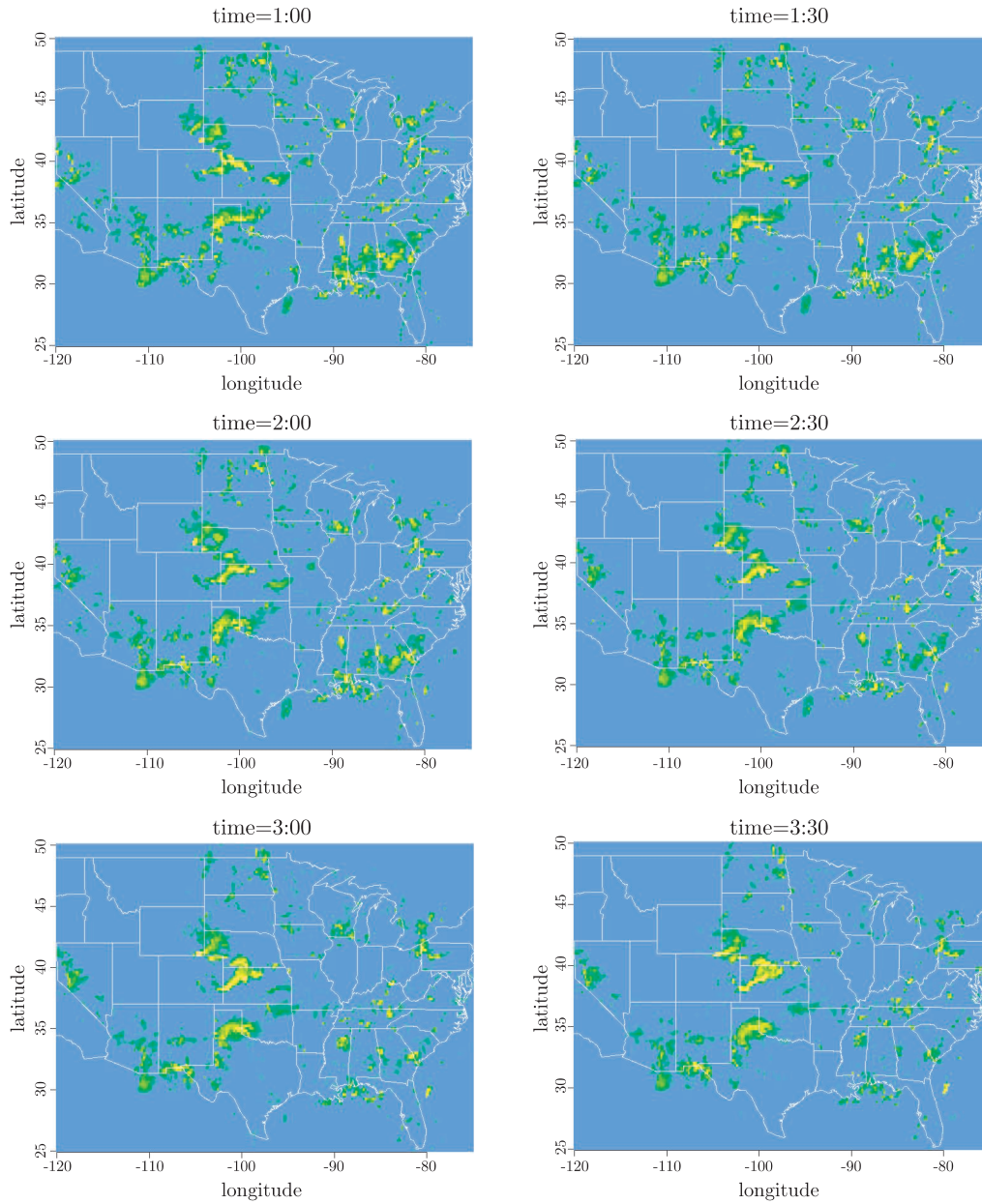


Figure 1. Radar Reflectivity Images Evolving During July 14, 1996.

behaviors are still largely unknown. It is certainly desirable for the purposes of climate modeling to gain a better understanding of the longer term behavior.

A real contribution then would be the validation and improvement of the storm activity in Regional Climate Models. These are complex computer models

for the state of the atmosphere; they are used for the prediction of weather and also to generate data to answer hypotheses. It is thus necessary for these models to maintain a high degree of reality in terms of the storms that they produce. Comparing the distribution of storm tracks from data to that of the Regional Climate Model is one way to verify this. Hence, a much needed tool is a procedure that can recover the movements and the interactions of all of the convective systems.

## 2.2. 2D turbulence

The second motivating problem is a 2D turbulence simulation of freely decaying vortexes; see Figure 2. These images are of a 2D vorticity field with random initial conditions as it develops over time with no energy loss to the overall system. The white objects are centers of vorticity rotating in a clockwise direction, whereas the black vortexes have the opposite rotation.

Vortexes of the same spin will coalesce as they move close to each other. There is a good example of a merger, between times 8 and 9, of two white vortexes that are left of center and below center in the images. Also, vortexes of opposite spin have a tendency to parallel each other for a while before moving off in different directions. Two vortexes exhibiting this behavior are called dipoles. An example of dipoling are the two vortexes that are left of center and above center in the images, they travel upwards and slightly left during the image sequence.

Recently such image sequences are a subject of much research (Bracco, McWilliams, Murante, Provenzale, and Weiss (2000), Pasquero, Provenzale and Weiss (2002), and Weiss and McWilliams (1993)), as it is a paradigm for anisotropic geophysical and astrophysical turbulence and, at the same time, it is also the most computationally accessible example of fluid turbulence. Turbulence remains a largely open area of research. Automatic tracking of turbulence structures in this simple example is an important first step toward achieving a better understanding of turbulence dynamics in more complex systems.

## 3. A Stochastic Model for Target Tracking

In this section we propose a stochastic model for solving multiple target tracking problems. Throughout the entire modeling process, we aim to achieve the following two important and somewhat conflicting goals: (i) we want to incorporate as much as possible of our physical understanding of the scientific problems into the model, and (ii) for computational feasibility, we want the resulting likelihood function to be quickly and accurately evaluated.



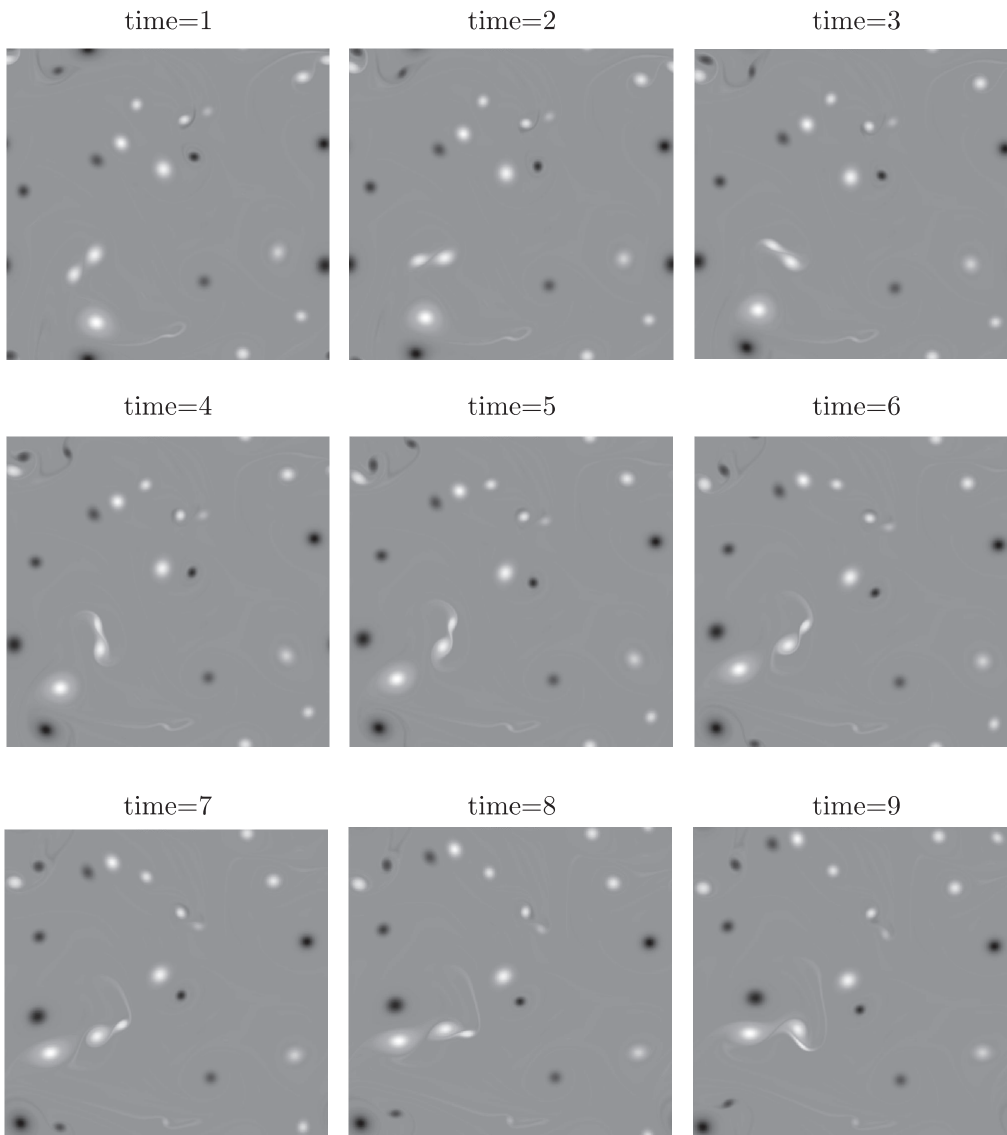


Figure 2. 2D turbulence simulation.

Define a path,  $(X(t), Y(t))$ , as the coordinates of the centroid of a target at time  $t$ . We assume that the image sequence is sampled from a continuous process at discrete times  $\mathbf{t} = (t_1, \dots, t_n)$ . We focus on 2D settings, but our approach can be easily generalized to higher-dimensional problems. We model the path of any target by a 2D random process. This is complicated however by the occurrences of birth, death, splitting and merging of targets, along with missing observations and false alarms.

The proposed model consists of the following five sub-models: (i) the *Event Model* that controls the birth, death, merging and splitting of targets; (ii) the *Observability Model* that determines when a real target is detected or missing; (iii) the *Location Model* that describes the movements of the targets; (iv) the *Attribute Model* that incorporates other characteristics of the targets (e.g., size, orientation, or intensity); and (v) the *False Alarm Model* that handles the occurrence of false alarms.

### 3.1. Target event model

The Event Model is a continuous-time Markov model that determines how and when birth, death, merging or splitting events occur. The Markov assumption implies that the times between successive events are independent exponentially distributed random variables. While the best distribution for modeling event waiting times is problem dependent, the memoryless property of the exponential distribution does seem to be a realistic assumption for the storm and vortex problems discussed above.

In the Event Model the rates at which birth, death, splitting and merging events happen are given by  $\lambda_b$ ,  $N(t)\lambda_d$ ,  $N(t)\lambda_s$ , and  $(N(t) - 1)\lambda_m$  respectively, where  $N(t)$  is the number of targets in existence at time  $t$ . It is assumed that the initial number of targets  $N_0$  follows a Poisson distribution:  $N_0 = N(t_1) \sim \text{Poisson}(\lambda_0)$ . Notice that the rates of death and splitting events are proportional to the number of targets in existence. This is because every target has an individual rate of dying (or splitting) which is independent of other targets. For merging events, it may seem more natural for the rate to be proportional to  $\binom{N(t)}{2}$ , the number of pairs of targets. However, we adopted  $(N(t) - 1)\lambda_m$  for the following two reasons. First, if we used  $\binom{N(t)}{2}$ , the problem would not be scale invariant in that if  $N(t)$  were scaled up by a factor of two,  $\binom{N(t)}{2}$  would be scaled up by a factor of 4 instead of 2. Second, it is also intuitive to use  $(N(t) - 1)\lambda_m$  as the rate. This is because each target should only be merged with its closest neighbor, and there are at most  $N(t) - 1$  such closest pairs to consider when there are  $N(t)$  targets.

The following notation will be used to describe the Event Model

$$\begin{aligned}
 U_{b,j} &= \text{number of births in the interval } [t_j, t_{j+1}) \\
 U_{d,j} &= \text{number of deaths in the interval } [t_j, t_{j+1}) \\
 U_{s,j} &= \text{number of splits in the interval } [t_j, t_{j+1}) \\
 U_{m,j} &= \text{number of mergers in the interval } [t_j, t_{j+1}).
 \end{aligned} \tag{3.1}$$

We write  $\mathbf{U}_b = (U_{b,1}, \dots, U_{b,n})$ , and similarly for  $\mathbf{U}_d$ ,  $\mathbf{U}_s$ , and  $\mathbf{U}_m$ . Also, denote the collection of  $N_0$  and the  $\mathbf{U}$ 's by  $\mathcal{U} = (N_0, \mathbf{U}_b, \mathbf{U}_d, \mathbf{U}_s, \mathbf{U}_m)$ .

Each target, regardless of its status (e.g., alive or dead), will be uniquely identified by a positive integer starting from 1. We call such integers *indices*. The initial targets alive at time  $t_1$  are arbitrarily labeled with indices 1 through  $N_0$ . The following actions are taken at the times when one of the four possible events happens. When there is a birth, the new target is given the next available index. For example, if there are already 10 targets in the model (some currently alive, some could be dead), these targets would have been labeled uniquely with indices from 1 to 10, and the new target will be given an index of 11. When there is a death, all targets that are still alive are equally likely to be selected as the one that dies. When there is a split, all of the living targets are equally likely to be the parent, and the children are given the next two available indices. Finally, for merging events all of the possible pairs of all living targets are equally likely to be the parents, and the child is given the next available index.

Notice that the assumptions that all pairs of events are equally likely appears to be in contradiction to the principle that only close targets are eligible to merge together. We rectify this issue in the Location Model to be described in Section 3.3. In short, locations of the parents of a merger are conditioned to be “close” to each other right before the merger. This shifts the burden of enforcing the property that “only close targets merge together” to the location model. This leads to an important simplification of the likelihood calculation since the location model depends on the Event Model but not vice-versa.

We specify which targets were involved in the events by

$$\begin{aligned}
 V_{b,j} &= \text{the collection of indices of targets that were born in the interval } [t_j, t_{j+1}) \\
 V_{d,j} &= \text{the collection of indices of targets that died in the interval } [t_j, t_{j+1}) \\
 V_{s,j} &= \text{the collection of triplets } (i_1, i_2, i_3) \text{ where } i_1 \text{ is the index of the parent and} \\
 &\quad i_2, i_3 \text{ are the children for every split in the interval } [t_j, t_{j+1}) \\
 V_{m,j} &= \text{the collection of triplets } (i_1, i_2, i_3) \text{ where } i_1, i_2 \text{ are the indices of the} \\
 &\quad \text{parents and } i_3 \text{ is the child for every merger in the interval } [t_j, t_{j+1}). \quad (3.2)
 \end{aligned}$$

Let  $\mathbf{V}_b = (V_{b,1}, \dots, V_{b,n})$ , and similarly for  $\mathbf{V}_d$ ,  $\mathbf{V}_s$ , and  $\mathbf{V}_m$ . The collection of all the  $\mathbf{V}$ 's will be denoted as  $\mathcal{V} = (\mathbf{V}_b, \mathbf{V}_d, \mathbf{V}_s, \mathbf{V}_m)$ .

Lastly, it should be noted that this is a hidden Markov model in that we do not actually observe the variables  $\mathcal{U}$  and  $\mathcal{V}$  from the data. Predicting these variables is part of the tracking problem. This will be described further in Section 4.

### 3.2. Observability model

Now we discuss our approach to modeling missing targets, that is, real targets that exist but were not detected in some image frames. It is certainly not ideal

to simply allow these targets to die when they have escaped from detection, and to start new paths when they appear again. Instead, for such a missing target, a good tracking method should be able to impute its existence and return one single path as its track estimate. In the tracking literature missing targets are usually modeled using *i.i.d.* Bernoulli random variables. We adopt this approach as well. That is, at any time  $t$ , if a target exists it has probability  $P_d$  of being detected and producing an observation. This is also assumed independent over time.

### 3.3. Target location model

When a target is determined to exist by the Event Model, we model the path  $(X_i(t), Y_i(t))$  of the  $i$ th target with a Gaussian process, which is commonly used in many tracking applications. We also assume the target paths are independent of other targets unless they are required to split or merge as determined by the Event Model. The dependency introduced by splitting and merging will be described later, and we first present the distribution of  $X_i(t)$  under generic conditions. The distribution of  $Y_i(t)$  is similar, with the obvious changes in notation and parameters, and independent of  $X_i(t)$ .

Let the  $x$  component of location and velocity of the  $i$ th target at time  $t$  be denoted by  $X_i(t)$  and  $X'_i(t)$ , respectively, and denote the time of initiation of the  $i$ th target by  $\xi_i$ . If the  $i$ th target exists at the first observation time  $t_1$ , it is assumed that  $\xi_i = t_1$ . Then

$$X_i(t) = X_i(\xi_i) + X'_i(\xi_i)(t - \xi_i) + \sigma_i G_i(t - \xi_i), \quad (3.3)$$

where  $G_i(t)$  is some continuous mean zero Gaussian process, for which we have chosen to use an integrated Brownian motion (IBM). This is sometimes referred to as a nearly constant velocity model, and is popular for target location within the tracking community. In addition, a comparison of IBM path realizations to those produced by storms and vortices revealed that the IBM model was flexible enough to represent these paths well.

The initial position  $X_i(\xi_i)$  and the initial velocity  $X'_i(\xi_i)$  depend on whether the target resulted from a birth, merging or splitting event. More detail for each of these cases is given below.

#### Initial Conditions for a Target Resulting from a Birth Event.

Suppose that the  $i$ th target is the result of a birth. It is assumed that the initial position and velocity are Gaussian,  $X_i(\xi_i) \sim \mathcal{N}(\mu_{X_0}, \sigma_{X_0}^2)$  and  $X'_i(\xi_i) \sim \mathcal{N}(\mu_{X'_0}, \sigma_{X'_0}^2)$ . For the two problems described above, it may also seem reasonable to use a uniform distribution to model the initial location  $X_i(\xi_i)$ . However, very often the likelihood of a uniform distribution can be satisfactorily mimicked

by sufficiently increasing the variance of a normal distribution. Thus we keep the original Gaussian assumption for mathematical convenience.

**Initial Conditions for a Target Resulting from a Merging Event.** Now suppose that the  $i$ th target is the child resulting from a merging event. Let  $p_i = (p_{i,1}, p_{i,2})$  be the vector containing the indices of the two parents. Suppose for now that size information is available and that it is realistic to model the expected size as a constant, as in Section 3.4. Use  $E(S_j)$  to denote the mean size of the  $j$ th target. Physically, the initial position or centroid of the child should be the average, weighted by size, of the positions of the parents at the time of merger,

$$X_i(\xi_i) = \left( \frac{E(S_{p_{i,1}})}{E(S_i)} X_{p_{i,1}}(\xi_i) + \frac{E(S_{p_{i,2}})}{E(S_i)} X_{p_{i,2}}(\xi_i) \right).$$

Also, by conservation of momentum, the initial velocity of the child should be the weighted average of the velocities of the parents at the time of merger,

$$X'_i(\xi_i) = \left( \frac{E(S_{p_{i,1}})}{E(S_i)} X'_{p_{i,1}}(\xi_i) + \frac{E(S_{p_{i,2}})}{E(S_i)} X'_{p_{i,2}}(\xi_i) \right).$$

If there is no size information available then we can let the initial position of the child be the simple average of the positions of the parents, plus perhaps a small amount of noise  $\psi_{m,i}$ . Figure 3 displays a physical representation of this. We can also mimic the conservation of momentum by taking the child's velocity to be the simple average of the parent velocities plus noise. This yields

$$X_i(\xi_i) = \frac{1}{2} (X_{p_{i,1}}(\xi_i) + X_{p_{i,2}}(\xi_i)) + \psi_{m,i} \quad (3.4)$$

$$X'_i(\xi_i) = \frac{1}{2} (X'_{p_{i,1}}(\xi_i) + X'_{p_{i,2}}(\xi_i)) + \psi'_{m,i} \quad (3.5)$$

where  $\psi_{m,i} \sim \mathcal{N}(0, \sigma_{X_m}^2)$  and  $\psi'_{m,i} \sim \mathcal{N}(0, \sigma_{X'_m}^2)$ . Presumably,  $\sigma_{X_m}^2$  and  $\sigma_{X'_m}^2$  are small so that the new target location and velocity are likely to be close to the averages of the parents. **Parent Locations at the Time of a Merging Event.** Notice that in our modeling so far, the parent targets are not required to be close to each other at the time of a merging event. To ensure that the parents move close to each other before merging, the difference between locations of the parents at the time of merger is conditioned to be small. This is done as follows.

Let  $d = (d_1, d_2, d_3)$  be a vector containing the indices of the three targets involved in a merging event where  $d_1$  and  $d_2$  are the parents while  $d_3$  is the index of the child. Let  $D$  be the difference in location between the two parents at the time of merger plus a noise term,

$$D = X_{d_1}(\xi_{d_3}) - X_{d_2}(\xi_{d_3}) + \psi_d, \quad (3.6)$$

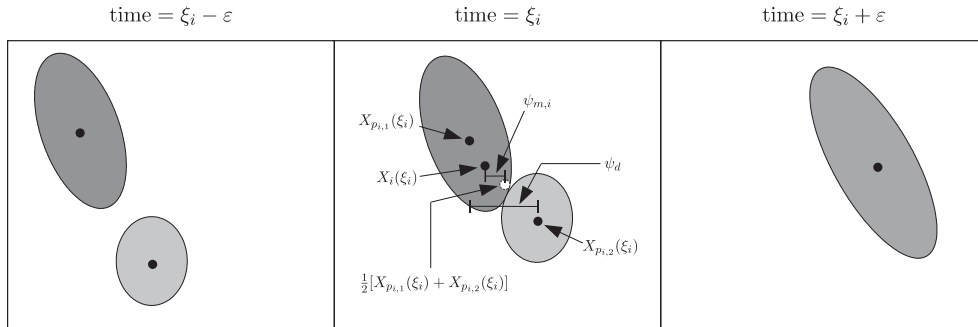


Figure 3. Physical description of a merger.

where  $\psi_d \sim \mathcal{N}(0, \sigma_{X_d}^2)$  and independent of the targets. If  $\sigma_{X_d}$  is small, then it is likely that  $\psi_d$  is small in absolute value. If we then condition the model for  $X_{d_1}$  and  $X_{d_2}$  on the event  $D = 0$ , this will ensure that the parents are only a small distance  $\psi_d$  apart at the time of the merging event. In Figure 3 once again, we see a merging event with a possible realization of  $\psi_d$ .

In general, there will be  $N_m = \sum_{j=1}^n U_{m,j}$  merging events during the time window  $[t_1, t_n]$ . We condition the target paths on all of these mergers in a manner similar to that above. This is described more precisely as follows. Let  $D_i$  be the  $D$  from (3.6) and  $\psi_{d,i}$  be the corresponding  $\psi_d$  for the  $i$ th merging event,  $i = 1, \dots, N_m$ . We then condition the model for  $(X_1, \dots, X_M)$  on the event  $\{(D_1, \dots, D_{N_m}) = (0, \dots, 0)\}$ , where  $M$  is the total number of targets that existed before time  $t_n$ .

Here we remark that our approach for handling merging, as illustrated by Figure 3, is a very realistic physical description of merging targets for many real problems, including the two problems described earlier.

**Initial Conditions for a Target Resulting from a Splitting Event.** Suppose that the  $i$ th target is initiated by a splitting event. That is, the  $i$ th target is one of two children resulting from a splitting event. To keep notation consistent, let  $p_{i,1}$  be the index of the parent. The location of a child resulting from a split is modeled by

$$X_i(t) = X_{p_{i,1}}(\xi_i) + \psi_{s,i} + \left[ X'_{p_{i,1}}(\xi_i) + \psi'_{s,i} \right] (t - \xi_i) + \sigma_i G_i(t - \xi_i), \quad (3.7)$$

where  $\psi_{s,i} \sim \mathcal{N}(0, \sigma_{X_s}^2)$  and  $\psi'_{s,i} \sim \mathcal{N}(0, \sigma_{X'_s}^2)$ . Similar to that for merging events, the initial position and velocity of a new child from a splitting event is the same as that of the parent plus a random error. It is assumed that  $\sigma_{X_s}^2$  is small so that the new targets are likely to appear close to where the parent split. Similarly,  $\sigma_{X'_s}^2$  should be small so that the new targets have a velocity similar to that of their parent.

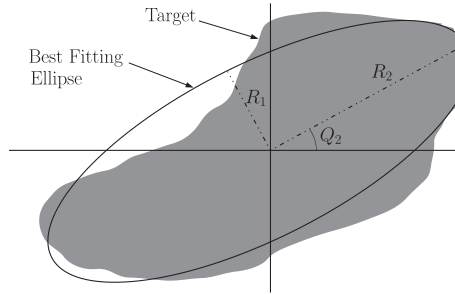


Figure 4.  $R_1$ ,  $R_2$ , and  $Q_2$ .

If size information is available then the conservation of momentum assumption can be imposed on the two new paths from the children splitting off from the parent. This can be achieved by conditioning in a similar manner as for parent locations at the time of a merging event.

**Measurement Error.** Due to imperfect detection or some other reasons, we do not always observe the exact locations of the targets. Rather, instead of  $X_i(t)$ , we observe a noisy version of it. We assume that the measurement errors are additive, Gaussian and independent over time. That is, for  $j = 1, \dots, n$ , we observe

$$X_i^*(t_j) = X_i(t_j) + \varepsilon_{i,j} \quad \text{with} \quad \varepsilon_{i,j} \stackrel{i.i.d.}{\sim} \mathcal{N}(0, \sigma_{X_e}^2).$$

### 3.4. Target attribute models

In this section we describe several models that can be applied in conjunction with the Event and Location Models to improve the tracking results when some auxiliary information about target attributes is available. We present a few special cases of attributes that are commonly available: size, orientation and intensity. Other attributes may be handled in a similar manner.

**Size.** In the 2D case, the size of a target is the area and, similar to the approach in Angelova and Mihaylova (2006), we use the length of the minor and major axes,  $R_1(t)$  and  $R_2(t)$ , of the best fitting ellipse of the target to characterize it (see Figure 4). We assume that a best fitting ellipse for each target has already been obtained, e.g., by standard imaging techniques as described in Rosenfeld and Kak (1982).

Let  $R_{1,i}(t)$  and  $R_{2,i}(t)$  be the minor and major radii respectively for the  $i$ th target at time  $t$ . The quantities  $R_{1,i}(t)$  and  $R_{2,i}(t)$  are modeled as log-normal random variables with parameters  $(\mu_{R_{1,i}}, \sigma_{R_{1,i}}^2)$  and  $(\mu_{R_{2,i}}, \sigma_{R_{2,i}}^2)$ , respectively. The log-normal distribution is commonly used to describe the distribution of

lengths (Wang and Lee (2006)). These observations are also assumed to be independent over time. One could certainly make other more realistic assumptions, such as allowing the size of a target to change as a positive continuous process. However, this assumption leads to complications in the likelihood calculation when there are splitting and merging events. This in turn drastically increases the computational load, and we do not consider this option further.

**Intensity.** For many tracking problems, the intensity  $I(t)$  of a target can be defined in various meaningful manners. For example, for the storm application, it can be defined as the maximum rainfall rate, the average of rainfall rates, or even some combination of the two. Such intensities can be modeled in a manner similar to size. We can assume some appropriate random process for the observations. For the storm application in Section 6, the intensity did not seem to change much over time, and we found that treating the observations as *i.i.d.* from a lognormal distribution was adequate.

**Orientation.** The orientation of a target can be measured by the angle,  $Q_2$  of the major axis corresponding to  $R_2$ , as shown in Figure 4. We assume that  $Q_2$  follows a von Mises distribution on  $[0, \pi]$  with parameters  $\alpha_i$  and  $\beta_i$  (Fisher (1995)). As with intensity and radii, we assume that  $Q_2(t)$  is *i.i.d.* over time in the storm application of Section 6.

### 3.5. False alarm model

The modeling of false alarms, also known as clutter, is divided into three parts (event, location and attributes) in a manner similar to that for targets.

**False Alarm Event and Location.** In most existing tracking methods, the number of false alarms in each image is assumed to be Poisson while the locations are typically assumed uniform throughout the image. This is equivalent to a homogeneous spatial Poisson process (Diggle (2003)). We also assume that the distribution of false alarms is a spatial Poisson process (possibly heterogeneous) that is independent over time, with intensity function  $\rho(x, y)$ . In addition, we recommend using  $\rho(x, y) = \lambda_f [X_i(\xi)](x) [Y_i(\xi)](y)$ , where  $[X_i(\xi)]$  is the density of the initial  $x$ -location of a target resulting from a birth, and similarly for  $[Y_i(\xi)]$  with respect to the  $y$ -location. This ensures that the contribution of the initial location to the likelihood does not influence the probability that an observation is a target versus a false alarm; see Section 4. Notice that  $\lambda_f$  is then the expected number of false alarms at each time point. Unlike the case for real targets, there is no need for an Observability Model for false alarms, as “missing false alarms” will never exist.

**False Alarm Attributes.** If any attributes are used to model the real targets, they must also be used to model the false alarms. This is to guarantee that the



likelihood computed by assuming a group of sequential observations originated from a real target would be comparable to the likelihood computed by assuming the same observations were false alarms. We also propose that a false alarm attribute be *i.i.d.* over each of the false alarm occurrences. The conceptual reason for the identical distribution is that all of these false observations are artifacts of the same device. Hence it is reasonable to assume the parameters are the same.

#### 4. The Tracking Estimate

In this section we formally define the estimand of our tracking problem and propose a method for estimating it. The setup for this problem is as follows. We collect data at times  $t_1, \dots, t_n$ . At the  $j$ th time step,  $m_j$  observations are detected from the  $j$ th image frame. Each of these  $m_j$  observations is either a real target or a false alarm. Let  $Z_i(t_j)$  be the  $i$ th observation at time  $t_j$ ,  $i = 1, \dots, m_j$ . Each  $Z_i(t_j)$  is a vector of the location values for either a target or a false alarm. Depending on the problem, it may also include the values of attribute variables.

Let  $\mathcal{Z} = \{Z_i(t_j) : j = 1, \dots, n; i = 1, \dots, m_j\}$  be the collection of observations at all times. From our data,  $\mathcal{Z}$ , we need to decide whether each observation,  $Z_i(t_j)$ , originated from a real target or a false alarm. In addition, if it was from a real target, we also need to decide which target it should be assigned to. Note that each observation can be assigned to only one target and each target can only have one observation assigned to it. We let  $p_i(t_j)$  be the index of the target that observation  $Z_i(t_j)$  originated from. We can define the index for a false alarm to be 0.

Write  $\mathcal{P} = \{p_i(t_j) : j = 1, \dots, n; i = 1, \dots, m_j\}$ . So for a given  $\mathcal{Z}$ ,  $\mathcal{P}$  will specify the tracks of each target. On top of that we must also specify the events (births, deaths, splitting and merging) that occurred with the variables  $\mathcal{U}$  and  $\mathcal{V}$  defined in (3.1) and (3.2), respectively. The variables  $\mathcal{U}$  and  $\mathcal{V}$ , together with  $\mathcal{P}$ , will denote a solution to the tracking problem. Let our estimate of the tracking solution  $(\mathcal{U}, \mathcal{V}, \mathcal{P})$  be denoted  $(\hat{\mathcal{U}}, \hat{\mathcal{V}}, \hat{\mathcal{P}})$ . Notice that in the usual statistical terminology this is actually called a prediction problem, since  $(\mathcal{U}, \mathcal{V}, \mathcal{P})$  is a random variable in our framework. We chose to call it the *tracking estimate* instead of the tracking prediction to avoid the potential confusion with track prediction (of future target locations).

##### 4.1. Calculating $(\hat{\mathcal{U}}, \hat{\mathcal{V}}, \hat{\mathcal{P}})$

First we delay the issue of parameter estimation to Section 4.4, and assume for the moment that all the parameters in the model, such as  $\lambda_0, \lambda_b, \dots, P_d$ , the  $\sigma_i$ 's etc., described in Section 3, are known; see the first paragraph in Section S.6 for a complete listing of the model parameters. We use the notation  $[X]$  to denote

the probability density function of the random variable  $X$ ,  $[X](x)$  to denote  $[X]$  evaluated at  $x$  and  $[X | Y]$  to denote the conditional density of  $X$  given  $Y$ .

To achieve our tracking estimate, we compute the conditional density of  $(\mathcal{U}, \mathcal{V}, \mathcal{P})$  given  $\mathcal{Z}$ ,

$$[\mathcal{U}, \mathcal{V}, \mathcal{P} | \mathcal{Z} = z](u, v, p). \quad (4.1)$$

Note that this is a probability mass function since the variables  $(\mathcal{U}, \mathcal{V}, \mathcal{P})$  are discrete. This can be computed fairly efficiently because the likelihood of our model conveniently factorizes into several conditional densities. See Section S3 for details.

From this it is natural to define our tracking estimate as

$$(\hat{\mathcal{U}}, \hat{\mathcal{V}}, \hat{\mathcal{P}}) = \arg \max_{u, v, p} [\mathcal{U}, \mathcal{V}, \mathcal{P} | \mathcal{Z} = z](u, v, p). \quad (4.2)$$

Even more, we can calculate the probability that  $(\hat{\mathcal{U}}, \hat{\mathcal{V}}, \hat{\mathcal{P}})$  is the correct solution given the data  $\mathcal{Z}$  as  $[\mathcal{U}, \mathcal{V}, \mathcal{P} | \mathcal{Z} = z](\hat{\mathcal{U}}, \hat{\mathcal{V}}, \hat{\mathcal{P}})$ .

A common but major difficulty in most problems is that it is not computationally feasible to enumerate all possible tracking solutions and calculate their likelihood values. To overcome this issue, we have developed a variant of the Multiple Hypothesis Tracking (MHT) algorithm of Reid (1979), the details of which are described in Section 4.3.

Our MHT algorithm will locate an approximation to the solution that maximizes (4.1), and hence it provides a point estimate  $(\hat{\mathcal{U}}, \hat{\mathcal{V}}, \hat{\mathcal{P}})$  to our tracking problem. Furthermore, by providing an estimate of the density function (4.1), this MHT algorithm also allows us to make probability statements about the solution. This is achieved as follows. Upon completion, the MHT algorithm provides, approximately, the  $K$  best solutions; i.e., those with highest likelihood values. We discuss the choice of  $K$  in the next subsection. Label these  $K$  solutions  $(u_i, v_i, p_i)$  for  $i = 1, \dots, K$  and let  $\mathcal{K}$  denote the set of these  $K$  solutions. If we assume that the correct solution is in  $\mathcal{K}$ , then we can calculate the conditional density of  $(\mathcal{U}, \mathcal{V}, \mathcal{P})$  given  $\mathcal{Z} = z$  and the event  $(\mathcal{U}, \mathcal{V}, \mathcal{P}) \in \mathcal{K}$ ,

$$[\mathcal{U}, \mathcal{V}, \mathcal{P} | \mathcal{Z}, (\mathcal{U}, \mathcal{V}, \mathcal{P}) \in \mathcal{K}](u, v, p | z). \quad (4.3)$$

In practice we can then compute the distribution given in (4.3) to approximate that in (4.1).

## 4.2. Confidence sets

Equation (4.1) provides the distribution of all possible tracking solutions given the data  $\mathcal{Z}$ , which is much more informative than just the estimate  $(\hat{\mathcal{U}}, \hat{\mathcal{V}}, \hat{\mathcal{P}})$ . For example, by summing over the relevant probabilities, we could calculate the

probability that a given observation is a target or a false alarm, or the probability that two targets merged, etc. Another innovative use of (4.1) is confidence set construction. That is, we could construct a confidence set  $C$  in the sense that the probability that  $C$  contains the correct solution is  $\geq (1 - \alpha)100\%$ . A major challenge for this is how to summarize and display all elements (i.e., different solutions of paths) in  $C$  in a meaningful and informative manner. We will report our work on this topic in a future paper.

### 4.3. A modified MHT algorithm

We have modified the MHT algorithm of Reid (1979) to approximate our tracking solution (4.2). Major steps of this modification are described next, while a more detailed description is given in Section S.4. We begin with a brief description of the original MHT algorithm of Reid (1979).

First we note that in the tracking literature a possible solution to a tracking problem is called a *hypothesis*, and hence the name of the algorithm. The original MHT algorithm assumes that an objective function (e.g., likelihood) has been constructed for evaluating the score of any solution at any time step. It processes the data sequentially; at each time step  $t$  it keeps a set of the  $K$  hypotheses with highest likelihood, where  $K$  is pre-specified by the user; at time step  $t + 1$  it forms the collection of all feasible hypotheses given the new data for each of the hypotheses from time  $t$ , and prunes the list of the new hypotheses back to  $K$  of them, by eliminating those that have low scores; the algorithm continues in a similar fashion until the data sequence finishes. In this way  $K$  sets of tracking estimates are obtained and the one that corresponds to the best score value is taken as the final tracking estimate.

Here is an overview of our modified MHT algorithm. For all of the details including “gating” procedures and other computational efficient shortcuts, see Section S.4. At time  $t_1$ , we consider all combinations of each observation treated as a real target or a false alarm (thus if there are  $m_1$  initial observations, there are  $2^{m_1}$  possible combinations). Now imagine that we have a set of solutions for the observations through time  $t_{j-1}$ . We then take the new observations  $\mathbf{Z}_j$  at time  $t_j$  to form updated solutions based on all possible combinations of the following possibilities: we assume that each observation  $Z_i(t_j) \in \mathbf{Z}_j$  is either (i) an observation from an existing target track, (ii) the first observation from a target resulting from birth, (iii) the first observation from a target resulting from split, (iv) the first observation from a target resulting from merger, or (v) a false alarm. We also assume that any existing track that does not receive a new observation must either (i) become (or stay) missing, or (ii) terminate.

To reduce computational load, we only keep a subset of the new solutions (those with the highest likelihood) to form solutions for the next time step  $t_{j+1}$ .

The actual number of solutions  $K$  that we keep through to the next time step is determined as follows. Let  $L_j$  be the likelihood value of the best solution at time  $t_j$ . At each time step  $t_j$ , we keep only solutions that have likelihood values greater than  $cL_j$ , where  $c < 1$  is a user-defined parameter. In addition we limit the total number of hypotheses kept to be less than  $K_s$ . In our implementation we set  $c = e^{-10}$  and  $K_s = 200$ .

#### 4.4. Incorporating estimation of static model parameters

To this point we have assumed the values of all static model parameters to be known, the approach that is taken by most tracking methods. Here we consider the more realistic stance that these parameter values are unknown. First we collect all the static model parameters such as  $\lambda_0$ ,  $\lambda_b$ ,  $\sigma_i^2$ , etc. into the vector  $\theta$ , and we write the density in (4.1) as  $[\mathcal{U}, \mathcal{V}, \mathcal{P} \mid \mathcal{Z}]_\theta$  to explicitly represent the dependence on  $\theta$ .

To estimate  $\theta$ , one could use maximum likelihood, or any other suitable estimates; see Section S.5. Since these estimates depend on the variables  $(\mathcal{U}, \mathcal{V}, \mathcal{P})$ , we could allow each of the solutions  $(u, v, p)$  that we consider in the above modified MHT algorithm to have its own parameter estimate, which we denote  $\hat{\theta}(u, v, p)$ . Notice that this leads to an overly optimistic likelihood value

$$[\mathcal{U}, \mathcal{V}, \mathcal{P} \mid \mathcal{Z}]_{\theta=\hat{\theta}(u,v,p)}(u, v, p \mid z) \quad (4.4)$$

for each of the solutions. However, based on the work of Storlie et al. (2008), it can be shown under certain conditions that, as the time between observations tends to zero,

$$(\hat{\mathcal{U}}, \hat{\mathcal{V}}, \hat{\mathcal{P}}) = \arg \max_{u,v,p} [\mathcal{U}, \mathcal{V}, \mathcal{P} \mid \mathcal{Z}]_{\theta=\hat{\theta}(u,v,p)}(u, v, p \mid z) \quad (4.5)$$

is equal to  $(\mathcal{U}, \mathcal{V}, \mathcal{P})$  eventually, almost surely. Hence we use the distribution defined in (4.4) to calculate likelihood values within the MHT iterations, and to decide which solutions that we keep for propagation to the next time step.

Although the function in (4.4) is useful for calculating the point estimate  $(\hat{\mathcal{U}}, \hat{\mathcal{V}}, \hat{\mathcal{P}})$ , it is not a good approximation of the distribution in (4.1). This is because each argument,  $(u, v, p)$ , is given its own estimated value for  $\theta$  which introduces a bias that persists even asymptotically. In order to estimate (4.1), we can, however, use  $[\mathcal{U}, \mathcal{V}, \mathcal{P} \mid \mathcal{Z}]_{\theta=\tilde{\theta}}$  where  $\tilde{\theta}$  is a consistent estimator of  $\theta$ . This ensures that the probabilities given by  $[\mathcal{U}, \mathcal{V}, \mathcal{P} \mid \mathcal{Z}]_{\theta=\tilde{\theta}}$  have a frequentist interpretation, at least in an asymptotic sense. A natural candidate for  $\tilde{\theta}$  is given by  $\tilde{\theta} = \hat{\theta}(\hat{\mathcal{U}}, \hat{\mathcal{V}}, \hat{\mathcal{P}})$ . As long as the estimator  $\hat{\theta}(\mathcal{U}, \mathcal{V}, \mathcal{P})$  is consistent, then  $\hat{\theta}(\hat{\mathcal{U}}, \hat{\mathcal{V}}, \hat{\mathcal{P}})$  is also consistent by the result given in Storlie et al. (2008), mentioned above.

Thus, if the MHT algorithm finishes with  $K$  solutions  $\{(u_i, v_i, p_i)\}_{i=1}^K$  at the last step, we could use

$$\left\{ [\mathcal{U}, \mathcal{V}, \mathcal{P} \mid \mathcal{Z}, (\mathcal{U}, \mathcal{V}, \mathcal{P}) \in \mathcal{K}]_{\theta=\hat{\theta}(\hat{\mathcal{U}}, \hat{\mathcal{V}}, \hat{\mathcal{P}})}(u_i, v_i, p_i \mid z) \right\}_{i=1}^K \quad (4.6)$$

as an estimate of (4.1) to allow us to make various probability statements concerning our final tracking estimates. We used (4.6) to compute the probabilities to be reported in Figure 5 and Section S.6.

Lastly we remark that in practice the researcher can, based on physical consideration on the tracking problem at hand, impose various limits on the parameter estimates. This would serve to limit the amount of bias introduced into the approximation given in (4.6). For example, the researcher is usually familiar with the range for the number of targets and/or false alarms so that effective limits can be imposed for the corresponding model parameters. This is certainly an improvement over the need for specifying these parameter values exactly.

#### 4.5. Summary

Here we summarize the major steps of the proposed tracking method,

1. To initialize the algorithm, consider all combinations of each observation from  $\mathbf{Z}_1$  treated as a real target or a false alarm. Denote the set of these combinations as  $\mathcal{K}_1$ . Set  $t = 2$ .
2. With  $\mathbf{Z}_t$  and  $\mathcal{K}_{t-1}$ , obtain the set of possible solutions for time  $\{1, \dots, t\}$  by enumerating the possibilities listed in the third paragraph of Section 4.3.
3. Calculate the likelihood values for all these solutions using (4.4). Keep the  $K$  solutions that have the  $K$  largest likelihood values, where  $K$  is determined by the method described in the last paragraph of Section 4.3. Collect these  $K$  best solutions into  $\mathcal{K}_t$ .
4. Increment  $t$  and repeat Steps 2 and 3 until all  $\mathbf{Z}_t$  are processed. Denote the last set of tracking solutions as  $\mathcal{K}_n$ .
5. Take the solution from  $\mathcal{K}_n$  with the highest likelihood value as the final tracking estimate. In addition, using (4.6), one can use all members of  $\mathcal{K}_n$  to provide an estimate for the distribution (4.1).

#### 5. Simulated Data Results

In this section, we present the results of the above proposed tracking method on a simulated data set. For a full description and discussion of more extensive simulation results, see Section S.6 of the online supplementary document.

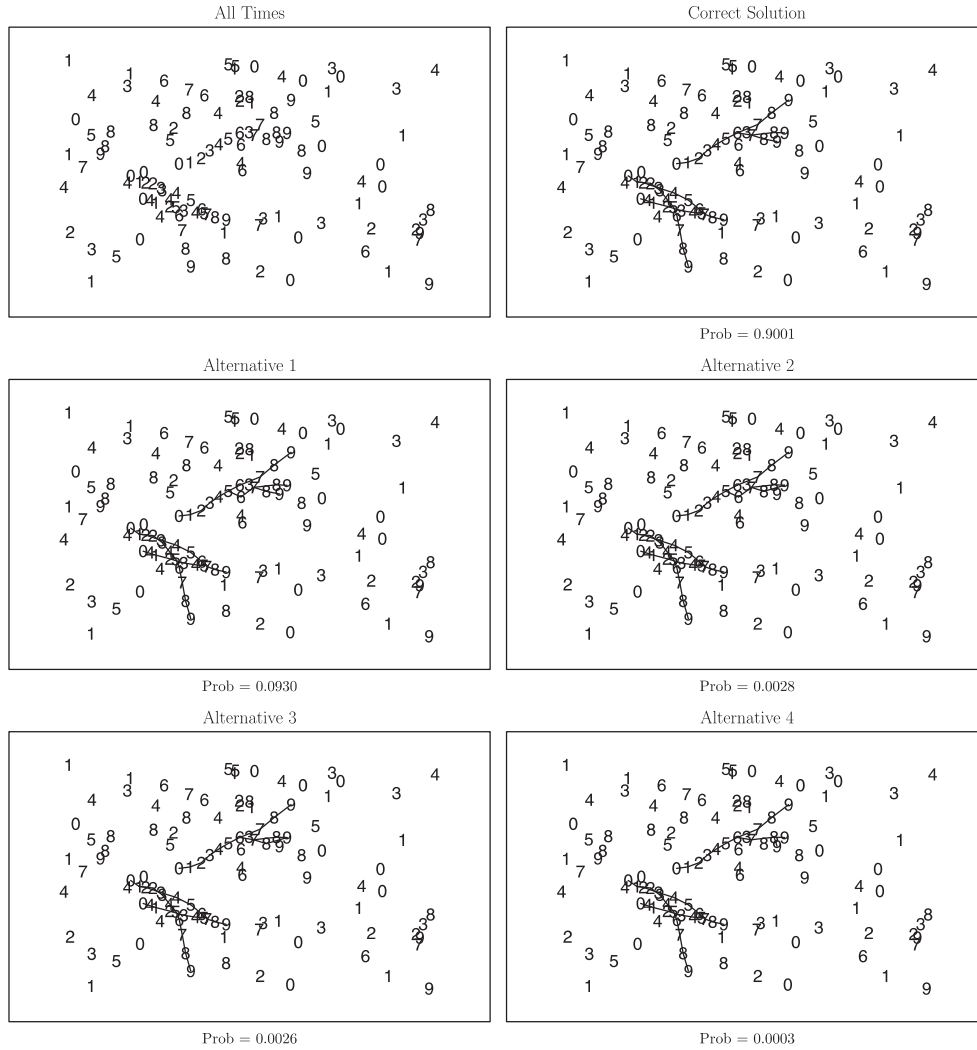


Figure 5. Tracking estimates of the proposed method on simulated data.

In this example, the data  $\mathcal{Z}$  are assumed to come from the model described in Section 3. The random motion component  $G_i(t)$  is an integrated Brownian Motion for all targets. The event parameters are set at  $\lambda_0 = 4$ ,  $\lambda_b = 0.1$ ,  $\lambda_d = 0.02$ ,  $\lambda_s = 0.06$ ,  $\lambda_m = 0.08$ , and  $\lambda_f = 8.0$ . All parameter values were chosen to mimic the rainfall data of Section 6. Values for other model parameters, along with the limits imposed on the parameters for the purposes of estimation, are given in Section S.6.

A realization from this model is shown in Figure 5. All the observations from all time steps are plotted together in one plot. Observations from time  $t_j$

are labeled ' $t_j$ '. The correct solution and the top four alternative solutions are provided along with their estimated probabilities approximated by (4.3). In this example we would be about 90% confident that our point tracking estimate is the correct solution. In fact, our estimate is the correct solution in this case.

## 6. Application to Rainfall Data

Here we consider tracking the storms that evolved during the morning of July 14, 1996, and shown in the images of Figure 1. We utilize available attribute data: size (radii of best fitting ellipse) and orientation. It can be concluded from the simulation results of Section S.6 that attribute information can be very beneficial for tracking estimation.

For the purposes of this problem, we are interested in tracking the mesoscale convective systems, which are usually defined to be storms with major axis longer than 100 km. This corresponds to roughly  $1^\circ$  latitude or longitude in Figure 1. Before we can track the storms, we must first identify them from the images and measure their location and attributes. An effective detection algorithm was used for this step and is described in Section S.7.

Recall that the images in Figure 1 are separated by 30 minutes. So this small illustration covers a time span of 2.5 hours. Figure 6 shows the same images as in Figure 1 after the initial processing of the detection algorithm. It is the ellipses in these images that we actually track.

Figure 7 shows the results of applying the proposed tracking method to the images in Figure 6. The limits for parameter estimation were set to the same as those in the simulations of Section S.6. The estimated tracks of the storms given by the tracking algorithm are displayed in black. Whenever there is a merging event an orange line is drawn connecting the parents to the child. Whenever there is a splitting event a magenta line is drawn connecting the parent to the children.

Notice that the estimate recovers the merging event that occurred between 1:00am and 1:30am, where the two storms in the south west corner of South Dakota merged together into one storm. Also, in this same time interval, the large system over Alabama and the panhandle of Florida split into two smaller systems. The reader is also referred to the same website that contains the online supplementary document (end of Section 1) to see a video of the raw data, the processed data, and the tracking solution with the path lines. The corresponding links from the above webpage are *Rain Fall Video*, *Cleaned Rainfall Images*, and *Paths Given by Tracking Algorithm*. These videos cover a longer time span as well, from 1:00am to 1:00pm on July 14, 1996.

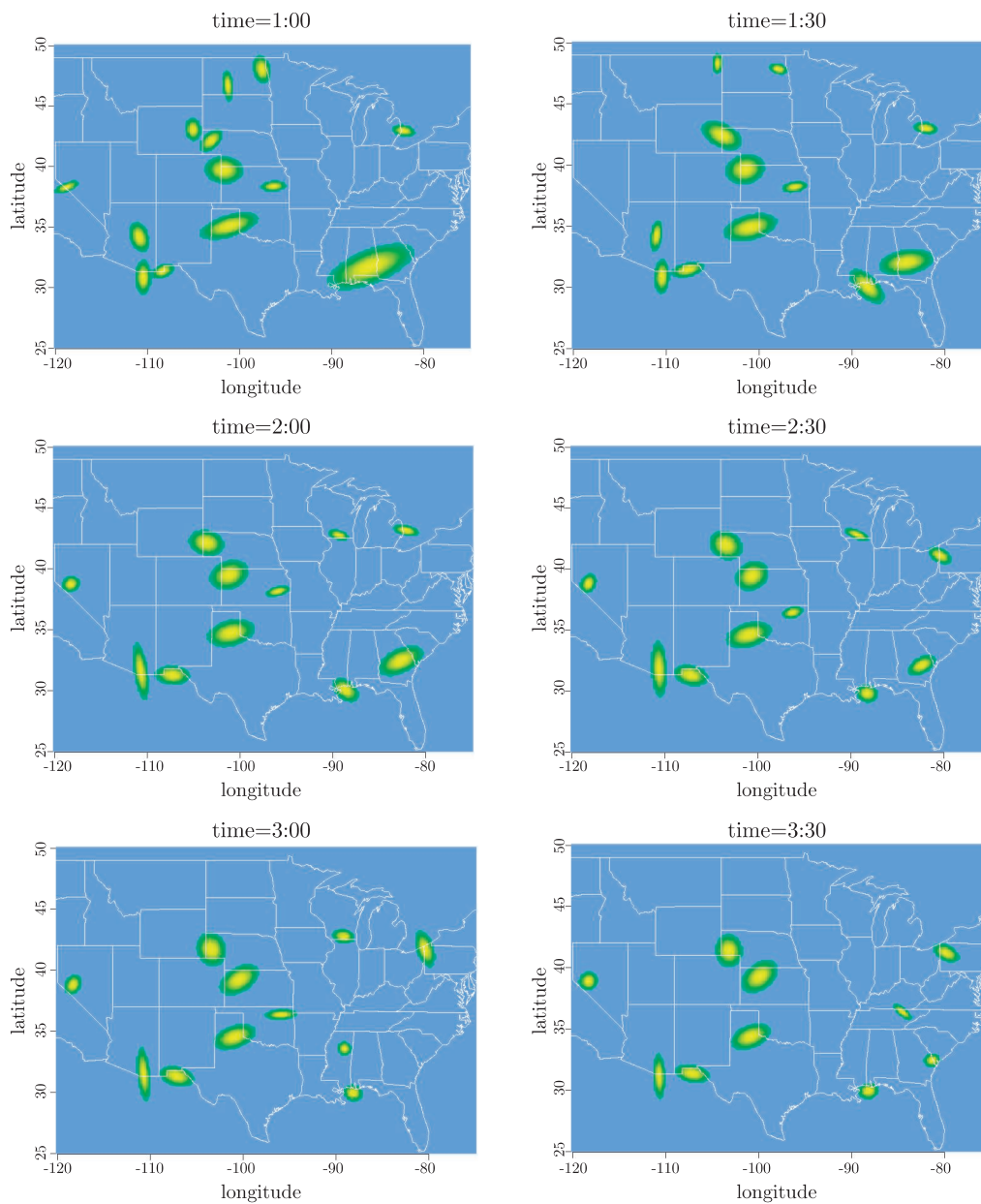


Figure 6. Best fitting ellipses of radar reflectivity images.

## 7. Application to 2D Turbulence

Here we consider tracking the vortices in the 2D turbulence simulation displayed in Figure 2. We again utilize the size information but forgo the use of orientation, since all of the vortices are nearly circular. The detection algorithm



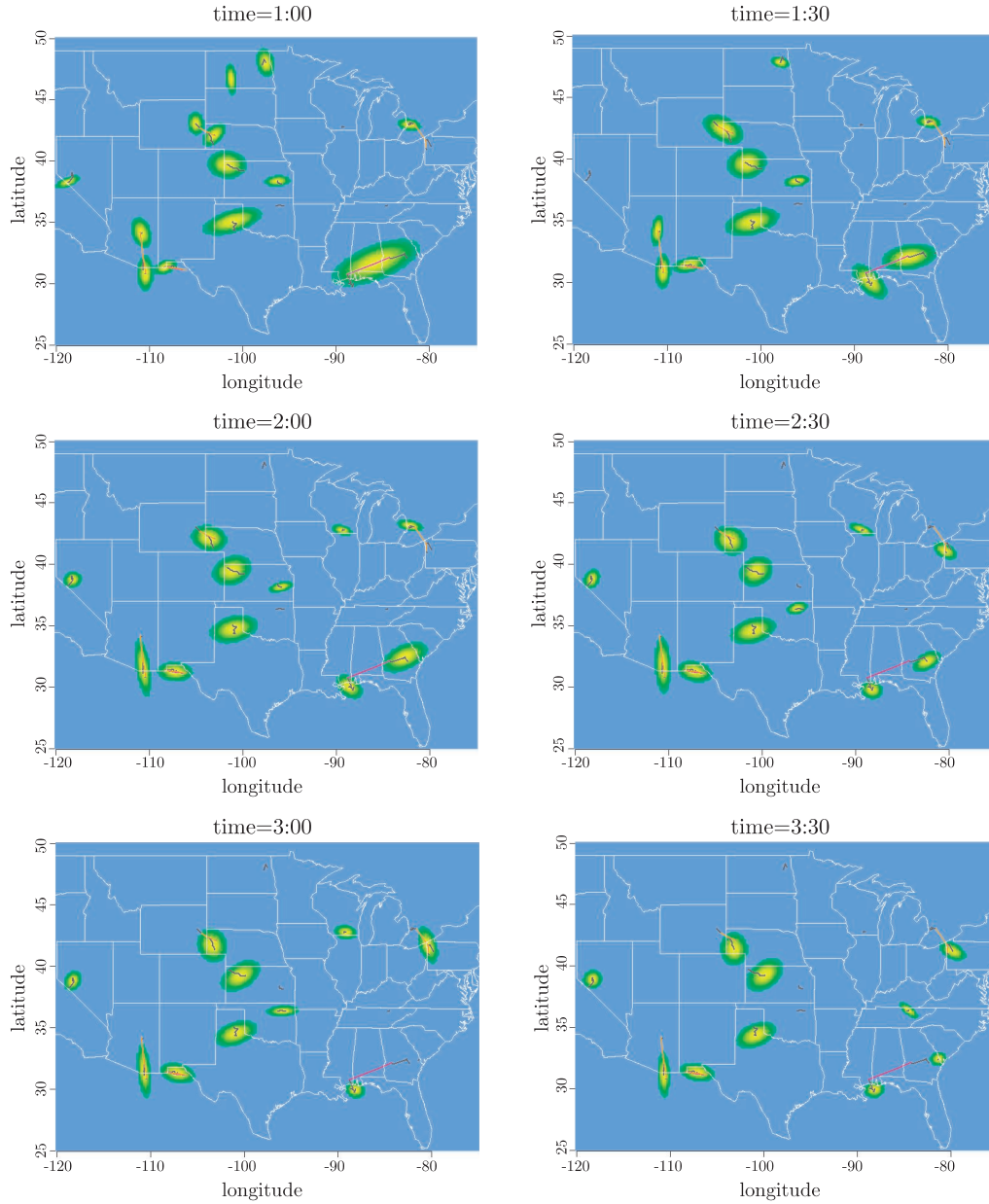


Figure 7. Best fitting ellipses with estimated paths superimposed

used to identify the vortices is similar to that used for the storms in Section 6.

Figure 8 shows the resulting  $(\hat{U}, \hat{V}, \hat{P})$  after applying the tracking algorithm to the processed data of Figure 2. Recall that the black vortices are spinning in a counterclockwise direction while the white vortices are spinning in a clockwise

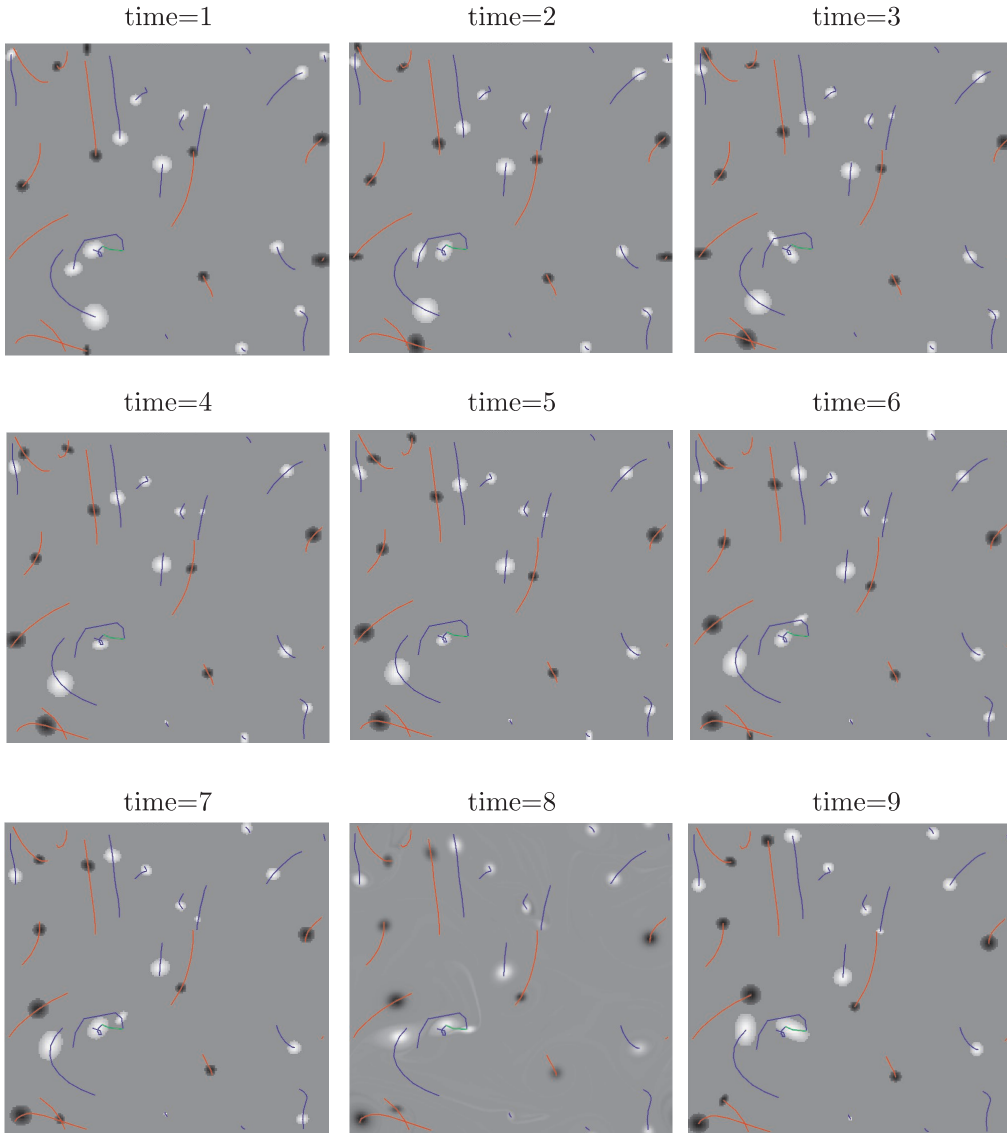


Figure 8. 2D turbulence simulation.

direction. The red lines indicate the paths of black vortices, while blue lines indicate the paths of white vortices. Merging events are indicated by orange lines for black vortices and green lines for white vortices. There is no splitting of targets in this problem.

A good example of a merging event can be found between frames 8 and 9. In addition, the tracking algorithm was able to capture the fact that one of these vortices was missing (not found by the detection algorithm) for two frames just

prior to the merging event. This is illustrated by a continuous path line where the vortex disappears for two frames.

The same website given at the end of Section 1 above also contains the corresponding video of the raw data given by the link *2D Turbulence Video*. Videos of the best estimate path lines superimposed on the cleaned images, as well as on the raw images, are given by the links *Vortex Paths on Cleaned Images* and *Vortex Paths on Raw Images*, respectively.

## 8. Conclusions and Further Work

In this article we have presented an introduction to multiple target tracking. We have also developed a new stochastic model that incorporates the events of birth, death, splitting and merging of targets into the likelihood, as well as missed detections and false alarms. Track estimation is then accomplished by considering the distribution of the relevant variables given the data.

The results obtained from simulated data provides strong evidence that this estimation procedure works very well even under the presence of false alarms and missing observations. The utility of the method was also demonstrated on two non-standard applications: radar reflectivity data collected over the United States and vortices in 2D turbulence images. A future project is to apply the method to other more difficult tracking applications, such as smaller scale convective systems (thunder storms) and/or missile target tracking.

For multiple target tracking estimation, there remain many challenging and interesting problems that are particularly suited for statisticians to tackle. As a first example, it would be useful to develop a procedure for sampling rate determination for tracking problems. That is, to automatically determine how frequently one should collect the data.

Secondly, what would be the best approach to incorporating and utilizing target attribute information when they are available? In this article we have used an ellipse to represent shape. Previously, Isard and Blake (1998) use curve segments to model target shape, allowing them to change over time, and Wang and Zhu (2004) decompose an image into basis functions, called tokens, and group these tokens together to form targets, again allowed to deform over time. However, all these methods are very much problem-dependent. It would be advantageous to develop a statistically sound principle to guide us in how to incorporate attribute information into the tracking solutions.

It would also seem very beneficial to combine the detection problem with the tracking problem. This is because the inherent spatio-temporal dependency provides additional information to help the detection algorithm predict where targets are more likely to exist in successive images. Some work has been done in this direction (e.g., Tonissen and Bar-Shalom (1998), Salmond and Birch (2001),

Kreucher et al. (2005) and Boers, Driessen, Torstensson, Trieb, Karlsson and Gustafsson (2006)), but the tracking models adopted tend to be relatively simple.

Lastly, virtually no rigorous attempts have been made to theoretically justify the use of any particular tracking methods. Although, as a first step, Storlie et al. (2008) investigate some theoretical properties of the estimates described in this article, there is still much theoretical work to be done to further the understanding of the multiple target tracking problem. Statisticians are well equipped to answer these questions (especially the last one), and it is our hope that this article will stimulate further statistical investigations in various aspects of the problem.

### Acknowledgements

The authors thank the National Science Foundation (Grants DMS-9815344 and 0504737) for partial support of this work, particularly for the funding for the conference presentation of the preliminary version, Storlie, Davis, Hoar, Lee, Nychka, Weiss and Whitcher (2004). The authors would also like to thank Tim Hoar for his help with data handling and computational resources, and Drs. Chris Davis and Jeff Weiss for their help in understanding the storm and vortex dynamics, respectively. Lastly the authors are grateful to the reviewers, an associate editor, and the editors for their most constructive comments, many of which are incorporated in the current version of this article.

### References

- Anderson, B. and Moore, J. (1979). *Optimal Filtering*. Prentice Hall, Englewood Cliffs, New Jersey.
- Andrieu, C. and Doucet, A. (2003). Online Expectation-Maximization type algorithms for parameter estimation in general state space models. *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing* **6**, 69-72.
- Angelova, D. and Mihaylova, L. (2006). A Monte Carlo algorithm for state and parameter estimation of extended targets. *Proceedings of the International Conference on Computational Science* **3994**, 624-631.
- Bal, B. and Alam, M. (2005). Automatic target tracking in FLIR image sequences using intensity variation function and template modeling. *IEEE Trans. Instrumentation and Measurement* **54**, 1846-1852.
- Bar-Shalom, Y., Li, X. and Kirubarajan, T. (2001). *Estimation with Applications to Tracking and Navigation*. Wiley, New York.
- Blackman, S. (2004). Multiple hypothesis tracking for multiple target tracking. *IEEE A&E Systems Magazine* **19**, 5-18.
- Blackman, S. and Popoli, R. (1999). *Design and Analysis of Modern Tracking Systems*. Artech House, Boston.

- Boers, Y., Driessen, H., Torstensson, J., Trieb, M., Karlsson, R. and Gustafsson, F. (2006). Track-before-detect algorithm for tracking extended targets. *IEE Proceedings of Radar, Sonar and Navigation* **153**, 345-351.
- Bracco, A., McWilliams, J., Murante, G., Provenzale, A. and Weiss, J. (2000). Revisiting freely decaying two-dimensional turbulence at millennial resolution. *Phys. Fluids* **12**, 2931-2941.
- Chang, K. and Bar-Shalom, Y. (1984). Joint probabilistic data association for multitarget tracking with possibly unresolved measurements and maneuvers. *IEEE Trans. Automat. Control* **29**, 585-594.
- Chen, R. and Liu, J. (2000). Mixture Kalman filters. *J. Roy. Statist. Soc. Ser. B* **62**, 493-508.
- Daum, F. E. (1994). The importance of resolution in multiple target tracking. *Signal and Data Processing of Small Targets 1994, SPIE Proceedings* **2235**, 329-338.
- Davis, C., Manning, K., Carbone, R., Trier, S. and Tuttle, J. (2003). Coherence of warm-season continental rainfall in numerical weather prediction models. *Monthly Weather Rev.* **131**, 2667-2679.
- Diggle, P. J. (2003). *Statistical Analysis of Spatial Point Patterns*. 2nd edition. Edward Arnold, New York.
- Dixon, M. (1994). Automated storm identification, tracking and forecasting - a radar based method. Ph.D. thesis, University of Colorado and National Center for Atmospheric Research.
- Doucet, A., de Freitas, N. and Gordon, N. (2001). *Sequential Monte Carlo Methods in Practice*. Springer Verlag, New York.
- Doucet, A., de Freitas, N. and Gordon, N. (2003). Parameter estimation in general state-space models using particle methods. *Ann. Inst. Statist. Math.* **55**, 409-422.
- Doucet, A., Godsill, S. and Andrieu, C. (2000). On sequential Monte Carlo sampling methods for bayesian methods. *Statist. Comput.* **10**, 197-208.
- Fisher, N. (1995). *Statistical Analysis of Circular Data*. Cambridge University Press.
- Genovesio, A. and Olivo-Marin, J. (2004). Split and merge data association filter for dense multi-target tracking. *Proceedings of the 17th International Conference on Pattern Recognition*, 677-680.
- Gordon, N., Salmond, D. and Smith, A. (1993). Novel approach to nonlinear/non-Gaussian Bayesian state estimation. *IEE Proceedings-F* **140**, 107-113.
- Hodges, K. (1994). A general method for tracking analysis and its application to meteorological data. *Monthly Weather Rev.* **122**, 2573-2586.
- Hodges, K. (1999). Adaptive constraints for feature tracking. *Monthly Weather Rev.* **127**, 1362-1373.
- Isard, M. and Blake, A. (1998). CONDENSATION - conditional density propagation for visual tracking. *Internat. J. Comput. Vision* **29**, 5-28.
- Johnson, J., Mackeen, P., Witt, A., Mitchell, E., Stumpf, G., Eilts, M. and Thomas, K. (1998). The storm cell identification and tracking algorithm: An enhanced WSR-88D algorithm. *Weather and Forecasting* **13**, 263-276.
- Julier, S. and Uhlmann, J. (2004). Unscented filtering and nonlinear estimation. *Proc. IEEE* **92**, 401-422.
- Khan, Z., Balch, T. and Dellaert, F. (2005). Multitarget tracking with split and merged measurements. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* **1**, 605-610.

- Khan, Z., Balch, T. and Dellaert, F. (2006). MCMC data association and sparse factorization updating for real time multitarget tracking with merged and multiple measurements. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **28**, 1960-1972.
- Kitagawa, G. (1996). Monte Carlo filter and smoother for non-Gaussian nonlinear state models. *J. Comput. Graph. Statist.* **5**, 1-25.
- Koch, W. and van Keuk, G. (1997). Multiple hypothesis track maintenance with possibly unresolved measurements. *IEEE Trans. Aerospace and Electronic Systems* **33**, 883-892.
- Kreucher, C., Kastella, K. and Hero, A. (2005). Multitarget tracking using the joint multitarget probability density. *IEEE Trans. Aerospace and Electronic Systems* **41**, 1396-1414.
- Lakshmanan, V., Rabin, R. and DeBrunner, V. (2003). Multiscale storm identification and forecast. *J. Atmospheric Research* **67**, 367-380.
- Liu, J. and Chen, R. (1998). Sequential monte carlo methods for dynamic systems. *J. Amer. Statist. Assoc.* **93**, 1032-1044.
- Mahler, R. (2003). Multitarget Bayes filtering via first-order multitarget moments. *IEEE Trans. Aerospace and Electronic Systems* **39**, 1152-1178.
- Obermeyer, F. and Poore, A. (2004). A Bayesian network tracking database. *Proceedings of the SPIE, ser. Conference on Signal and Data Processing of Small Targets* **5428**, 400-418.
- Pasquero, C., Provenazale, A. and Weiss, J. (2002). Vortex statistics from Eulerian and Lagrangian time series. *Physical Review Letters* **89**, 284501.
- Pece, A. (2002). Generative-model-based tracking by cluster analysis of image differences. *Robotics and Autonomous Systems* **39**, 181-194.
- Reid, D. (1979). An algorithm for tracking multiple targets. *IEEE Trans. Automat. Control* **24**, 843-854.
- Roh, H., Kang, S. and Lee, S. (2000). Multiple people tracking using an appearance model based on temporal color. *Proceedings. 15th International Conference on Pattern Recognition* **4**, 643-646.
- Rosenfeld, A. and Kak, A. (1982). *Digital Picture Processing*. 2nd edition. Academic Press.
- Salari, V. and Sethi, I. (1990). Feature point correspondence in the presence of occlusion. *IEEE Trans. Pattern Anal. Mach. Intell.* **12**, 87-91.
- Salmond, D. and Birch, H. (2001). A particle filter for track-before-detect. *Proc. Amer. Control Conference* **5**, 3755-3760.
- Salmond, D. and Parr, M. (2003). Track maintenance using measurements of target extent. *IEE Proceedings on Radar, Sonar, and Navigation* **150**, 389-395.
- Särkkä, S., Vehtari, A. and Lampinen, J. (2007). Rao-Blackwellized particle filter for multiple target tracking. *Information Fusion*. To appear.
- Sethi, I. and Jain, R. (1987). Finding trajectories of feature points in a monocular image sequence. *IEEE Trans. Pattern Anal. Mach. Intell.* **9**, 56-72.
- Stone, L., Barlow, C. and Corwin, T. (1999). *Bayesian Multiple Target Tracking*. Artech House, Boston.
- Storlie, C., Davis, C., Hoar, T., Lee, T., Nychka, D., Weiss, J. and Whitcher, B. (2004). Identifying and tracking turbulence structures. *Proceedings of the 38th Asilomar Conference on Signals, Systems, and Computers* **2**, 1700-1704.
- Storlie, C., Hannig, J. and Lee, T. (2008). The asymptotic properties of a multiple target tracking model. *Stochastic Process. Appl.* Submitted for publication.

- Tonissen, S. and Bar-Shalom, Y. (1998). Maximum likelihood track-before-detect with fluctuating target amplitude. *IEEE Trans. Aerospace and Electronic Systems* **34**, 796-809.
- Trunk, G. and Wilson, J. (1981). Track initiation of occasionally unresolved radar targets. *IEEE Trans. Aerospace and Electronic Systems* **17**, 122-130.
- Tuttle, J. and Gall, R. (1999). A single-radar technique for estimating the winds in tropical cyclones. *Bulletin of the American Meteorological Society* **80**, 653-668.
- Vermaak, J., Godsill, S. and Perez, P. (2005). Monte Carlo filtering for multitarget tracking and data association. *IEEE Trans. Aerospace and Electronic Systems* **41**, 309-332.
- Vo, B., Singh, S. and Doucet, A. (2005). Sequential Monte Carlo methods for Bayesian multitarget filtering with random finite sets. *IEEE Trans. Aerospace and Electronic Systems* **41**, 1224-1245.
- Wang, H. and Lee, T. (2006). Automatic parameter selection for a k-segments algorithm for computing principal curves. *Pattern Recognition Letters* **27**, 1142-1150.
- Wang, Y. and Zhu, S. (2004). Analysis and synthesis of textured motion: Particles and waves. *IEEE Trans. Pattern Anal. Mach. Intell.* **26**, 1348-1363.
- Weiss, J. and McWilliams, J. (1993). Temporal scaling behavior of decaying two-dimensional turbulence. *Physics of Fluids A: Fluid Dynamics* **5**, 608-621.
- Wolfson, M., Forman, B., Hollowell, R. and Moore, M. (1999). The growth and decay storm tracker. *8th Conference on Aviation, American Meteorological Society* 58-62.
- Department of Mathematics and Statistics, University of New Mexico. MSC03 2150, 1 University of New Mexico, Albuquerque, New Mexico 87131-0001, U.S.A.  
E-mail: storlie@stat.unm.edu
- Department of Statistics, The Chinese University of Hong Kong, and Department of Statistics, Colorado State University, U.S.A.  
E-mail: tlee@sta.cuhk.edu.hk
- Department of Statistics, Colorado State University, Fort Collins, Colorado 80523, U.S.A.  
E-mail: hannig@stat.colostate.edu
- Geophysical Statistics Project, National Center for Atmospheric Research, P.O. Box 3000, Boulder, CO 80307-3000, U.S.A.  
E-mail: nychka@ucar.edu

(Received June 2006; accepted November 2007)

## COMMENTARY

Jason Adaska

*BAE Systems, Advanced Information Technologies*

Storlie, Lee, Hannig, and Nychka approach the problem of multi-target tracking with merging and splitting events from a statistical point of view. The target tracking problem is well known in the engineering literature, and the authors

provide a great review of the history and successes in that field, particularly Multiple Hypothesis Tracking. They also set out to tackle a one of the toughest current problems in the field, that of tracking targets which interact. They apply this work to the practical problem of tracking storms.

Since my own background comes from an applied discipline, I can not help but approach this work with an eye toward some of the issues considered important for tracking systems used in the engineering community. Two broad criteria will be used for this purpose:

- Modeling – Does the statistical model accurately and succinctly capture the relevant behavior?
- Implementation – Will the procedure scale to realistic versions of the problem considered?

### Modeling Comments

The first criticism with regards to modeling has to do with the use of the size attributes (two radii and an orientation)  $R_1$ ,  $R_2$ ,  $Q$ . Blackman and Popoli (1999) remark that the “primary purpose of attribute fusion is to ascertain the ID of the target”, that is, they provide a mechanism to separate out one target from potential “confusers”. In traditional tracking contexts, one could consider something like color to be a useful attribute. If you want to track a red truck, then receiving a “red” measurement at two different times, would help discern the path of the target in the intervening period. This is usually made precise with a statistical model for the color sensor, and a prior probability on red trucks. What is clear from this simplified example is that an attribute must be correlated in time for it to be useful for ascertaining the identity of a target. If a truck changed color from one instant to the next, it would have no utility for resolving confusion. The size attributes  $R_1$ ,  $R_2$ ,  $Q$  are modeled as independent in time, so they suffer from a similar problem. They are not attributes in this usual sense, and thus not useful for resolving confusion in the multi-target tracking problem. Their use appears to be mainly in mitigating false alarms, and if this is the intent, then I would argue they should be treated from this perspective. For physical reasons, sizes of storms are correlated with time, and this could be an important feature to model. Additional utility for size could be injected by using it to inform merging events. For instance, the noise term for D (distance between two parent storms at the time of a merging event) could utilize a covariance that scales with the size of the object, thus allowing two large storms to merge even if their centroids are distantly separated.

The second comment pertains to the implicit coupling of the sensors to the underlying process. In short, the sensors should not be driving merging and splitting events, but “(p.22) *Existing tracks that do not receive a new observation*



... *must either 1) go missing or 2) terminate*". Setting aside the awkwardness of incorrect causality, there is a practical danger to this choice as a simple example illustrates. Suppose two storms are observed separated at  $t_0$ , merge unobserved at  $t_1$ , and are observed merged at  $t_2$ . The correct hypothesis is not available for this situation. One could imagine "recovering" by choosing the hypothesis "separated at  $t_0$ ,  $t_1$  and merged at  $t_2$ " which is available, and in a sense close to the true event history. The problem here is that this hypothesis requires two simultaneous misses at  $t_1$ . The likelihood could be very low in this instance, and in danger of being pruned by the MHT algorithm.

As a final modeling comment, the treatment of false alarm versus new track hypotheses seems to result a poor engineering design. Particularly, the false alarm intensity is set to the new track intensity. The authors state this was done "to ensure that the contribution of the initial location to the likelihood will not influence the probability that an observation is a target versus a false alarm". In practical terms, if location provides a mechanism for discriminating false alarms versus new targets, then this information should be put to use. In many cases, false alarms are a function of sensor errors which are typically decoupled from the physics of creating new objects. For instance, the formation of new storms would likely be a function of regional atmospheric conditions, while false alarms would often be related to thermal noise at the sensor. These stochastic processes are almost certainly uncorrelated.

### Implementation Comments

Perhaps the most important issue in a practical multi-target tracking system is the way in which the combinatorial explosion of hypotheses is managed. Clearly, one could always consider additional behavior in a stochastic system, and merging/splitting events are an important (and particularly challenging) example. However, additional branching of potential hypotheses only increases the rate at which hypotheses grow (and must be pruned), so attention to this issue is paramount. For instance, the number of individual merging events that could be hypothesized grows like the square of the number of tracks. The authors navigate this particular issue by setting the rate of merging events to be proportional to  $N(t)$  rather than  $N(t)^2$  and then using the location model to evaluate the pairwise distance between merging tracks, but, fundamentally, a procedure is still needed to efficiently prune away hypotheses that are unlikely without discarding those hypotheses which may be end up becoming the most likely at a later time. I would be very interested in seeing the authors extend their analysis to address these issues in combinatorial growth.

From a computational perspective, the choice made in computing  $\Sigma^*$  and  $\mu^*$  (parameters in the position distribution after a merger) is particularly troubling.

These quantities require a matrix inversion whose size depends on the length of the track (Supplement pp.5-6). Thus the computational cost increases with time as the tracks become larger. This is a chief reason why recursive estimation filters are pursued for multi-target tracking solutions in other fields. This is possibly the largest single problem for incorporating this technique in a real-world tracking application.

## Conclusion

I was pleased to read the authors contribution to the problem of multi-target tracking when targets split and merge, since it is simultaneously difficult and of great practical utility for the tracking community. In that literature, the problem of tracking objects that split and merge is sometimes referred to as the “group tracking problem”, where clusters of tracks may come together in formation, move as a coordinate unit, and split apart at a later time. Some recent work done under this heading is in Huali et al. (2001) and in Van Keuk (2002).

## References

- Blackman, S. and Popoli, R. (1999). *Design and Analysis of Modern Tracking Systems*, Artech House, Boston.
- Huali, W., Wang, D., et. al (2001). A New Algorithm for Group Tracking. *CIE International Conference on Radar*, 1159-1163.
- Van Keuk, G. (2002). MHT extraction and track maintenance of a target formation. *IEEE Transactions on Aerospace and Electronic Systems*, 288-295.

BAE Systems, Advanced Information Technologies 6 New England Executive Park Burlington, MA 01803, U.S.A.

E-mail:

(Received October 2008; accepted November 2008)

## COMMENTARY

David S. Choi and Patrick J. Wolfe

*Harvard University*

*Abstract:* Target tracking is a ubiquitous and important application in engineering and the sciences, with several problems still open for advancement from both an algorithmic and a methodological perspective. In this discussion we highlight two key issues that arise in practice: target covariance estimation and data association.

*Key words and phrases:* Data association, multi-target tracking, posterior covariance estimation

## 1. The Target Tracking Formulation of Storlie et al.

Storlie et al. describe a continuous-time stochastic model for target tracking in which a random number  $N$  of targets, each with random birth and death times, is observed in total over a finite time interval  $t \in [0, T]$ . Two-dimensional target paths  $(X_k(t), Y_k(t))$ , for  $k \in \{1, \dots, N\}$ , evolve as standard linear dynamical systems with Gaussian process noise; however, with positive probability, their initial states are constrained to be near the final states of other paths—indicating a splitting or merging of target “tracks.” In the sequel we denote by  $\mathcal{P}$  the set of  $N$  target paths  $\{(X_k(t), Y_k(t))\}$  and their associated birth and death times.

For each observation time  $t_j$ , a random number  $m_j$  of measurements is assumed; these measurements comprise either noisy observations of the targets or spurious “false alarms.” To indicate the association of measurements to targets, a latent categorical variable  $p_i(t_j)$  is defined for every  $i \in \{1, \dots, m_j\}$  and time  $t_j$ , such that  $p_i(t_j) = 0$  denotes a spurious measurement and  $p_i(t_j) = k$  a measurement of the  $k$ th target, with a specified prior odds ratio governing the false alarm rate. Computing the posterior distribution of the target paths, each with associated birth and death times, is made difficult by the fact that one must estimate or marginalize over a number of unknown nuisance parameters, including the set of latent target assignment variables  $\mathcal{A} := \{p_i(t_j)\}$ .

## 2. The Nagging Issue of Covariance Misunderestimation

In the absence of splits and merges, the celebrated Kalman filter provides an analytical means of estimating the set of target paths  $\mathcal{P}$  and associated uncertainty due to Gaussian measurement noise and linear, Gaussian state dynamics, conditional upon the set  $\mathcal{A}$  of latent target assignment variables. However, in cases where  $\mathcal{A}$  must be estimated from measurements, leading to the so-called data association problem (Blackman and Popoli (1999)), much work remains to be done in efficiently and accurately estimating the uncertainty associated with posterior estimates of  $\mathcal{P}$  as a function of the uncertainty associated with  $\mathcal{A}$ .

If we assume an average of  $n$  active targets at each measurement time, then the effective cardinality of  $\mathcal{A}$  is seen to be  $\mathcal{O}(n^{\sum m_j})$ , which quickly renders an exact marginalization infeasible. For this reason, the tracking community has long focused on efficiently approximating the  $K \ll |\mathcal{A}|$  most likely posterior estimates of  $\mathcal{A}$ . Storlie et al. adopt the same approach; however, it is clear that in cases where posterior mass fails to be well concentrated, this approximation can severely underestimate the uncertainty associated with estimates of  $\mathcal{P}$ .

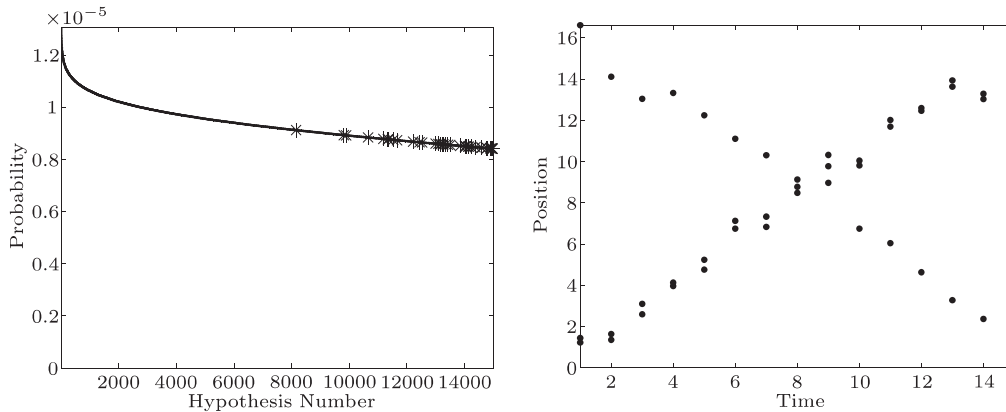


Figure 2.1. Left: Simulated measurements of three target paths, displaying either a single path crossing a pair of paths, or two paths that diverge as a third path approaches. Right: Ranked posterior probabilities of the target association set  $\mathcal{A}$  for its 15,000 most likely values, with those corresponding to the former possibility distinguished by asterisks.

Figure 2.1 provides a typical example in which this “ $K$ -best” heuristic fails unless  $K$  is roughly exponential in the number of measurements. Noiseless measurements are taken from three targets whose dynamics are modeled as a one-dimensional Brownian motion, with zero probability of false alarm or missed detection at each measurement time. For this scenario, the  $K$  most probable choices of  $\mathcal{A}$  can be found without resorting to approximation. Measurements could correspond either to two similar paths that are intersected by a third path, or to two paths that closely follow each other but then diverge as a third path approaches. Under typical assumptions, both possibilities are comparable in posterior probability, but the largest values of the target association set  $\mathcal{A}$  are dominated by the second possibility alone. Unless  $K$  is very large, it is clear that the resultant posterior estimates of the target paths  $\mathcal{P}$  may not only be extremely erroneous, but will also exhibit drastically underestimated covariance terms.

Moreover, in applications, these covariances are typically visualized separately for each target trajectory, amounting to a summarization of the joint posterior based on various marginals of the posterior of  $\mathcal{P}$ . However, symmetries resulting from exchangeability of the target parameters with respect to their labels arise when considering the entire posterior—an issue that can escape attention when conditioning on the  $K$  best estimates of  $\mathcal{A}$ . Hence, while the many modern-day successes of Monte Carlo methods would seem to suggest an approach based on sampling from the posterior of  $\mathcal{A}$  to more accurately capture true posterior covariances, it remains an open problem as to how best to employ

such samples in the context of posterior summarization for multi-target tracking.

### 3. A Role for Efficient Batch Methods in Tracking?

The target tracking scenario described above provides a canonical example of the label-switching problem associated with estimating a mixture of distributions from data (Jasra, Holmes, and Stephens (2005)). The optimal procedure frequently being intractable, offline mixture estimation problems are typically treated using variants of the expectation-maximization approach. For online applications exhibiting an appropriate latency tolerance—potentially even the applications of Storlie et al.—batch estimation techniques may also hold promise for the target tracking problem. In this vein, one possible approach to batch clustering for target tracking would involve choosing an initial estimate for  $\mathcal{A}$ , and then iterating between the following steps.

1. Estimate  $\mathcal{P}$ , conditioned upon  $\mathcal{A}$ . Generally,  $\mathcal{P}$  is estimated using the Kalman filter; in the setting of Storlie et al., however, elements of  $\mathcal{P}$  fail to be conditionally independent given  $\mathcal{A}$ , and hence must be estimated jointly.
2. Given an estimate of  $\mathcal{P}$ , assume the target paths to be independent and choose  $\mathcal{A}$  to maximize the probability of the data. As is well known in the tracking community, such a target assignment can be efficiently computed.

To illustrate this latter point, let  $n_j$  be the number of active targets at measurement time  $t_j$ , and denote by  $k_1, k_2, \dots, k_{n_j}$  their indices with respect to the full index set of all targets  $\{1, \dots, N\}$ . Define a likelihood by letting  $z_i(t_j)$  denote the measurement value associated with the latent variable  $p_i(t_j)$ , assuming that targets are detected with probability  $p_d$  at each time  $t_j$ . We may then define for each  $t_j$  an  $(m_j + n_j) \times (m_j + n_j)$  matrix  $C_j$  with entries as follows:

$$C_j(i, l) = \begin{cases} \log p(z_i(t_j) | X_{k_i}(t_j), Y_{k_i}(t_j), p_i(t_j) = k_l) & i \in \{1, \dots, m_j\}, \\ & l \in \{1, \dots, n_j\}, \\ \log p(z_i(t_j) | p_i(t_j) = 0) & i \in \{1, \dots, m_j\}, \\ & l \in n_j + \{1, \dots, m_j\} \\ \log(1 - p_d) & i \in m_j + \{1, \dots, n_j\}, \\ & l \in \{1, \dots, n_j\} \\ 0 & \text{otherwise.} \end{cases}$$

The maximizing set of latent assignment variables  $\mathcal{A}$  can then be efficiently computed at each time step, for instance by applying the well-known Hungarian

algorithm (Kuhn (1955)) to solve the optimization problem  $\max_{\Pi} \text{tr}(C_j \Pi^T)$  over the set of all permutation matrices  $\Pi$ ; one then sets  $p_i(t_j) = k_l$  if the  $(i, l)$ th element of  $\Pi$  equals 1 for  $l \leq n_j$ , and  $p_i(t_j) = 0$  if  $\Pi(i, l) = 1$  for  $l > n_j$ .

This example serves to illustrate the connection between estimation of mixture distributions and the data association problem in tracking, and we conclude with a brief mention of more sophisticated approaches in the recent literature. In this vein, Markov chain Monte Carlo algorithms inspired by results of Jerrum, Sinclair and Vigoda (2001) have recently been adapted for the target tracking problem by Oh, Russell and Sastry (2004) and Fox, Choi, and Willsky (2006), and a growing body of work from the computer vision and robotics communities, for instance work by Montemerlo and Thrun (2003), employs related ideas by way of sequential importance sampling.

## References

- Blackman, S. and R. Popoli (1999). *Design and Analysis of Modern Tracking Systems*. Artech House Radar Library, Norwood, MA.
- Fox, E. B., D. S. Choi and A. S. Willsky (2006). Nonparametric Bayesian methods for large scale multi-target tracking. In *Proceedings of the 40th Asilomar Conference on Signals, Systems and Computers*, 2009-2013.
- Jasra, A., C. C. Holmes and D. A. Stephens (2005). Markov chain Monte Carlo methods and the label switching problem in Bayesian mixture modeling. *Statist.Sci.* **20**, 50.
- Jerrum, M., A. Sinclair and E. Vigoda (2001). A polynomial-time approximation algorithm for the permanent of a matrix with non-negative entries. In *Proceedings of the 33rd Annual ACM Symposium on Theory of Computing*, 712-721.
- Kuhn, H. (1955). The Hungarian method for the assignment problem. *Naval Research Logistics Quarterly* **2**, 83-97.
- Montemerlo, M. and S. Thrun (2003). Simultaneous localization and mapping with unknown data association using FastSLAM. In *Proceedings of the IEEE International Conference on Robotics and Automation*, 1985-1991.
- Oh, S., S. Russell and S. Sastry (2004). Markov chain Monte Carlo data association for general multiple-target tracking problems. In *Proceedings of the 43rd IEEE Conference on Decision and Control*, 735-742.

Statistics and Information Sciences Laboratory, Harvard University, Oxford Street, Cambridge, MA 02138, U.S.A.

E-mail: dchoi@seas.harvard.edu

Statistics and Information Sciences Laboratory, Harvard University, Oxford Street, Cambridge, MA 02138, U.S.A.

E-mail: wolfe@stat.harvard.edu

(Received October 2008; accepted November 2008)

**COMMENTARY**

Arnaud Doucet and Ba-Ngu Vo

*University of British Columbia and University of Melbourne*

We would like to congratulate the authors for this interesting and timely paper. We hope it will generate a lot of interest in the statistical community. Multitarget tracking problems appear in numerous applications in computer vision and signal processing, and the storms tracking example provided by the authors is an additional attractive application. We share the opinion of the authors that such problems require the development of principled statistical methods, and that statisticians have the potential to make valuable contributions to this field.

It should also be acknowledged that tracking is a field where statistical tools have been used for a very long time. In particular, tracking problems have long been formulated as optimal (Bayesian) filtering problems and a large number of approximation techniques, such as the MHT described in this paper, have been proposed to perform inference. Advanced computational methods such as particle filtering techniques (Gordon, Salmond and Smith (1993)) for non-linear non-Gaussian models resulted from a collaboration between tracking specialists and a statistician and first became popular in tracking/vision before being adopted by statisticians. This is a perfect illustration that there is much to be gained from interactions between the statistical and tracking/vision communities.

We want to comment here on a few recent approaches and open problems arising in tracking which should be of interest to statisticians.

Many tracking applications require us to track simultaneously potentially hundreds of targets using very noisy measurements. In such cases, although it is possible ‘theoretically’ to formulate the problem using a Bayesian model quite similar to the one presented in this paper, the current approximation techniques such as the MHT-based approaches typically do not scale gracefully. In the example provided by the authors the targets appear rather well separated, the clutter rate is quite low. Perhaps the authors could comment on the performance of their algorithm in more challenging settings, e.g. higher number of targets and denser clutter. Typically, the main problem is that more challenging scenarios require solving a difficult data association problem. This is why a variety of interesting alternatives have been proposed over the past few years.

In particular, we believe that the random set/point process approach to multitarget tracking originally proposed by Mahler and approximations such as the

PHD/CHPD type filters are very useful for these types of applications (Mahler (2007)), (Vo, Vo and Cantoni (2007)). For example, the PHD filter can accommodate splitting; merging is also permitted since the intensity function in the PHD filter is essentially the density of the average count of the number of targets. It is also possible to incorporate other features such as intensity, spin, size and shape into the state; see (Mahler (2007)) for a recent review. This emerging area should be of interest to statisticians especially those working on spatial point processes. A good source of information and papers can be found from the website <http://randomsets.eps.hw.ac.uk/index.html>. Other techniques have also been proposed to reduce the computational complexity of multitarget tracking; this includes the use of belief propagation techniques or the use of a modified likelihood (Gilholm, Godsill, Maskell and Salmond (2005)) that bypasses the data association problem.

We also share the opinion of the authors that tracking specialists have neglected important problems. It is indeed often assumed that the parameters of the tracking model are known, which is rarely the case. The procedure proposed by the authors in this paper is a valuable addition to the tracking literature. Although the authors did establish consistency of their procedure when the sampling rate goes to zero, we wonder whether it would be beneficial to try to approximate directly the marginal likelihood of the parameters in general scenarios. Ideally a Bayesian approach to recursive static parameter estimation would be the most appropriate approach to address this problem as we are typically able to elicit reasonable prior distributions. Unfortunately the current approximation schemes for Bayesian recursive parameter estimation (Andrieu, De Freitas and Doucet (1999)), (Fearnhead (2002)), (Storvik (2002)) suffer from serious pitfalls as discussed in (Andrieu, De Freitas and Doucet (1999)). Even if it were possible to find a sound computational method, it is unlikely that this would scale for high-dimensional tracking problems.

## References

- Andrieu, C., De Freitas, J. F. G. and Doucet, A. (1999). Sequential MCMC for Bayesian model selection. *Proceedings IEEE Workshop Higher Order Statistics*, 130-134.
- Fearnhead, P. (2002). MCMC, sufficient statistics and particle filter. *J. Comput. Graph. Statist.* **11**, 848-862.
- Gilholm, K., Godsill, S. J., Maskell, S. and Salmond, D. (2005). Poisson models for extended target and group tracking. *Proc. SPIE: Signal and Data Processing of Small Targets*.
- Gordon, N. J., Salmond, D. J. and Smith, A. F. M. (1993). Novel approach to nonlinear/non-Gaussian Bayesian state estimation. *IEE-Proceedings-F* **140**, 107-113.
- Mahler, R. (2007). *Statistical Multisource-Multitarget Information Fusion*. Artech House.



- Storvik, G. (2002). Particle filters in state space models with the presence of unknown static parameters. *IEEE Trans. Signal Process.* **50**, 281-289.
- Vo, B.-T., Vo, B.-N. and Cantoni, A. (2007). Analytic implementations of the cardinalized probability hypothesis density filter, *IEEE Trans. Signal Process.* **55**, 3553-35677.

Departments of Statistics & Computer Science, University of British Columbia, Vancouver, BC V6T 1Z4, Canada.

E-mail: arnaud@cs.ubc.ca

Department of Electrical Engineering, University of Melbourne, Victoria 3010, Australia.

E-mail: bv@ee.unimelb.edu.au

(Received October 2008; accepted November 2008)

## COMMENTARY

Sumeetpal S. Singh

*University of Cambridge*

The authors have attempted to address a number of genuinely important modeling, inference and computational issues in multi-target tracking. The authors have developed a more detailed statistical model that explicitly accounts for merging and splitting. I agree with the need to give merging a more careful consideration and models that explicitly penalize merging of distant targets is definitely needed. In fact the model for merging proposed is appealing as it does not complicate the computation of the likelihood. However it appears to be more a model for mathematical convenience and lacks a physical interpretation. Perhaps the authors could shed some light on this matter and several others detailed below.

Regarding the estimation of model parameters in multi-target tracking, the authors have not pointed to any previous work on the subject. The method they have proposed is intriguing and potentially quite useful as it appears to integrate nicely with the Multiple Hypothesis Tracking (MHT) algorithm. However, how accurate are the estimates of the static parameters themselves?

Modeling in continuous time naturally deals with irregularly spaced observations. However, as the supplement to the paper details, inference is more complex

and approximations are necessary. One of my concerns is about the accuracy of the approximations employed therein to define the sampling distribution of interest. How does one cope with the scenario when the frequency at which the observations are gathered cannot be increased to control the error induced by these specific approximations?

The introduction certainly makes the case for online inference and hence the use of an iterative batch inference technique like MCMC is not appropriate. However, I wish to alert the authors that this conclusion may not necessarily hold for their Convective Systems application where the observation records are spaced by thirty minute intervals. This problem is dealt with in Section 6 where results are given for a small illustration with six image sequences. The number of observations per post-processed image is about 13. My recent experience with a similar sized data set (but for a different application) Yoon and Singh (2008) that spanned 30 observation times and about the same number of observations per time instance suggests that the Convective Systems data could have been analyzed with a Bayesian approach and with MCMC as the computation tool. Below I make the case for a simplified model in discrete time that does not including merging or splitting targets.

Generally the model I considered is similar to that defined by the authors but in discrete time. Specifics are as follows. Linear Gaussian dynamics were assumed for the motion of individual targets and individual targets did not interact. Target evolutions were subjected to thinning (probability of death) and new targets were introduced at each time according to a spatial Poisson birth process. The observability model was the same as that of the authors but did not include any attributes. All hyperparameters were estimated, i.e., target survival and detection probabilities, intensities of the target birth and false alarms processes, covariance matrix of target measurement model. The statistical problem (and sampling procedure) was essentially an extension of that in Oh, Russell and Sastry (2004) to include, in addition to the estimation of the tracks, the estimation of hyperparameters as well as the birth and death time of targets. A simple Metropolis-Within-Gibbs sampler that alternated between sampling the tracks (association of observations and targets) and hyperparameters was employed. For about 20 to 30 observation times, the total computation time for the tracks and hyperparameters was a couple of hours (on an Intel Dual-Core CPU @ 3.00GHz). To extract the tracks, the (marginal) MAP estimate was chosen. The implementation was in Matlab and no attempt was made to code it efficiently. I would imagine that a C++ implementation with merging and splitting would run in comparable if not less time.

## References

- Yoon, J. W. and Singh, S. S. (2008). A Bayesian approach to tracking in Single Molecule Fluorescence Microscopy. CUED/F-INFENG/TR.612 , Engineering Department, Cambridge University, Cambridge.
- Oh, S., Russell, S. and Sastry, S (2004). Markov Chain Monte Carlo data association for general multiple-target tracking problems. *Proceedings of the IEEE Conference on Decision and Control* **1**, 735-742.

Department of Engineering, University of Cambridge, Trumpington Street Cambridge, CB2 1PZ, UK.

E-mail: sss40@cam.ac.uk

(Received October 2008; accepted November 2008)

## COMMENTARY

Ying Wu and Ying Nian Wu

*Northwestern University and University of California, Los Angeles*

### Model and Inference

The statistical model proposed in this paper is a generative model. It is very natural to use a top-down model for the birth/death and split/merge events. However, it can be quite difficult to fit this model. The proposed modified-MHT amounts to heuristics, and theoretical justification would be helpful.

The fundamental issue in video-based tracking is the matching of the image evidence between frames. This corresponds to the likelihood (or observation or measurement) model. If this model is weak, no matter how good the prior model (e.g., the event model, split/merge model in this paper) is, the tracking method cannot be expected to work well.

### Multiple Identical Targets

If the multiple targets have different visual appearances, tracking is a manageable task as we can track the targets individually, with a linear complexity. However, if the targets are more or less the same in their appearances and are interacting with each other, it becomes a very difficult problem, especially when we want to keep track of the identities of the targets.

If it is difficult to separate the collected observations or measurements so as to associate them to different targets, a conventional solution is to test joint hypotheses. This leads to exponential complexity, and heuristics are needed in practice to reduce the complexity. An interesting idea, which is quite different from the joint hypothesis approach, is to perform distributed inference. Each target is associated with an individual tracker, but these trackers are not independent. They exchange messages and interact with each other. When their interaction reaches equilibrium, the inference is obtained and the targets are tracked.

The sequence of papers by Yu and Wu (2004, 2005), Yu, Wu, Krahnstoeber and Tu (2008) and Yang, Yu and Wu (2007) aim to develop a distributed/collaborative approach to multiple target tracking. These papers seek to answer the following question: if we assign a single-target tracker to an individual target, how can they communicate or collaborate in a network to track the entire scene with a linear complexity, and in an optimal way.

### **Trackability**

Another interesting issue is "trackability," that is, whether the image frames of a motion sequence contain elements whose trajectories can be inferred unambiguously. In natural scenes, there are patterns that are not very trackable, such as the surfaces of sea, lake, and river, fire, smoke, tree leaves in wind viewed from far distance, etc. For such patterns, even though it is difficult to identify individual elements and their trajectories, we do perceive vivid motion patterns. It is still unclear how to represent and model such patterns. Doretto, Chiuso, Wu and Soatto (2003) call such patterns "dynamic textures," and propose an auto-regressive model coupled with singular value decomposition for dimension reduction. In this model, there are no explicit targets and trajectories.

There are also a lot of trackable patterns, such as the movements of human figures, animals, and vehicles. For such patterns, it is important to learn what is trackable, at least within a small period of time.

Recently, Wu, Si, Gong and Zhu (2008) proposed a method for learning what is trackable by assuming that there is an object moving at an unknown speed, and in addition to this overall movement, the object is also allowed to have local movements or deformations. Such an object is represented by a deformable template, modeled by an active basis model. An active basis is a composition of a small number (e.g., 60) of elongate and oriented wavelet elements, and these elements are allowed to locally perturb their locations and orientations when the template is matched to an image or a frame of a video sequence. The goodness of fit is measured by log-likelihood ratio or a correlation score.

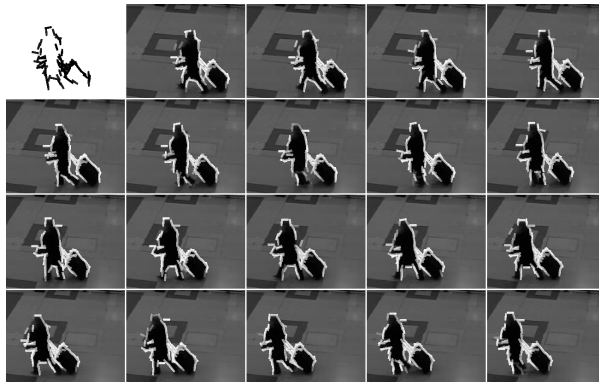


Figure 1. Learned template and superposed sketch for each frame at the estimated speed and direction.

Figure 1 illustrates an example. The first plot is the template of a man with a bag, where each wavelet element is represented by a small bar with the same length and at the same location and orientation as that wavelet element. The remaining plots show the 19 frames, and the superposed templates, which are deformed from the template in the first plot by perturbing the locations and orientations of the wavelet elements. Both the deformable template and the overall speed of this template can be learned automatically.

The above model is clearly too simple to describe more sophisticated motions, such as people dancing, horses running, birds flying while flapping their wings, etc. To model such motion patterns, we may need a sequence of animated templates, where each animated template can be modeled as a composition of a small number of moving wavelet elements.

It is still unclear how to define the concept of trackability. Apparently, it is related to coding complexity. The untrackable patterns appear more complex than the trackable ones, because the former do not have clear elements and trajectories. The concept of trackability seems also related to inferential uncertainty, that is, when there is too much uncertainty in estimating the targets and their trajectories, the pattern becomes untrackable.

Trackability also depends on spatial and temporal resolutions. When we reduce the resolutions of a trackable pattern in either the spatial or temporal domain or both, the pattern may become untrackable. It would be interesting to see whether one can develop a unified framework for modeling both regimes of motion patterns, as well as accounting for the transition between the two regimes. See Gong and Zhu (2008) for a preliminary attempt at understanding this issue.

## References

- Doretto, G., Chiuso, A., Wu, Y. N. and Soatto, S. (2003). Dynamic textures. *International Journal of Computer Vision*, 51, 91-109. <http://www.stat.ucla.edu/~ywu/research.html>.
- Gong, H. and Zhu, S. C. (2008). Trackability: information theoretical and empirical studies. Technical report. <http://www.stat.ucla.edu/hfgong/publications.html>.
- Wu, Y. N., Si, Z., Gong, H. and Zhu, S. C. (2008). Active basis for modeling, learning and recognizing deformable templates. Technical report. <http://www.stat.ucla.edu/~ywu/research.html>.
- Yang, M., Yu, T. and Wu, Y. (2007). Game-theoretic multiple target tracking. *IEEE Int'l Conf. on Computer Vision (ICCV)*. <http://www.ece.northwestern.edu/~yingwu/>.
- Yu, T. and Wu, Y. (2004). Collaborative tracking of multiple targets. *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*. <http://www.ece.northwestern.edu/~yingwu/>.
- Yu, T. and Wu, Y. (2005). Decentralized multiple target tracking using netted collaborative autonomous trackers. *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*. <http://www.ece.northwestern.edu/~yingwu/>.
- Yu, T., Wu, Y., Krahnstoeber, N. O. and Tu, P. H. (2008). Distributed data association and filtering for multiple target tracking. *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2008. <http://www.ece.northwestern.edu/~yingwu/>.

Department of Electrical Engineering & Computer Science, Northwestern University, Evanston, IL 60208, U.S.A.

E-mail: [yingwu@northwestern.edu](mailto:yingwu@northwestern.edu)

Department of Statistics, University of California, Los Angeles, CA 90095-1554, U.S.A.

E-mail: [ywu@stat.ucla.edu](mailto:ywu@stat.ucla.edu)

(Received October 2008; accepted November 2008)

## RESPONSE

Curtis B. Storlie<sup>1</sup>, Thomas C. M. Lee<sup>2,3</sup>, Jan Hannig<sup>3,4</sup> and Douglas Nychka<sup>5</sup>

<sup>1</sup>*University of New Mexico*, <sup>2</sup>*Chinese University of Hong Kong*,  
<sup>3</sup>*Colorado State University*, <sup>4</sup>*University of North Carolina at Chapel Hill*  
and <sup>5</sup>*National Center for Atmospheric Research*

First of all we would like to thank the co-editors of *Statistica Sinica* for organizing this discussion. We are also most grateful to all of the distinguished discussants for their insightful comments on our paper (henceforth denoted as SLHN). Many important points and suggestions are raised to improve our tracking procedure, along with some stimulating discussion of related topics. Space limitations prevent us from providing a comprehensive response to all of these

comments, but we shall attempt to respond to as many as possible. Our main goals are to clarify some of the important points raised, to provide a deeper understanding of our procedure, and to discuss how it can be improved. Before we start our individualized responses, we would like to mention that some theoretical justification to our tracking method is provided in Storlie, Hannig and Lee (2008).

*Adaska*: In this discussion, Adaska mentions several areas where he has questions about the practical application of our procedure. These criticisms are well thought out and will ultimately serve to improve the proposed method. Below, we use these criticisms to help further clarify some of the more subtle details of our tracking method.

**Size attributes**: Adaska comments that “If a truck changed color from one instant to the next, it would have no utility for resolving confusion. The size attributes  $R_1$ ,  $R_2$ ,  $Q$  are modeled as independent in time, so they suffer from a similar problem. They are not attributes in this usual sense, and thus not useful for resolving confusion in the multi-target tracking problem.” It is true that the size attributes  $R_1$ ,  $R_2$ ,  $Q$  are treated as independent observations over time. However, these variables are **not** identically distributed across targets in our model. Each target is allowed to have its own parameters (i.e., mean and variance) for  $R_1$ ,  $R_2$ ,  $Q$ . Thus, to use the analogy of the discussant, it would be as if the red truck was to appear red in color in each frame, but the exact color fluctuated slightly from one frame to another. The color *produced* by the truck (or, more precisely, captured and recorded by the sensor) would be independent from frame to frame, but the color observed in each frame comes from the same distribution that produces red observations on average.

As far as using size to help determine splitting and merging events, this is also built into the model by means of restricting the size parameters. That is, a storm resulting from a merger must have mean size equal to the sum of the mean sizes of the children. The comment about letting the noise term for  $D$  (distance between two parent storms at the time of a merging event) depend on the size of the storms is a good idea. This could be difficult to effectively implement however, since orientation of the two child storms would be an important factor in determining  $D$  as well.

**False Alarm Locations**: Adaska criticizes our use of the same initial location distribution for targets as that used for false alarms. He suggests that if location provides a mechanism for discriminating false alarms from new targets, then this information should be included. This is a valid point and our simplification in this case was admittedly made out of convenience. If there is a good reason to believe that false alarms and targets appear in substantially different locations, then this information should absolutely be incorporated into the initial

location distributions. As mentioned in SLHN, our model can easily allow for this if necessary. In future applications (e.g., recovering storm tracks) we will investigate its use.

**Implementation:** We made the choice in our implementation of searching over possible hypotheses to enforce the rule that “Existing tracks that do not receive a new observation ... must either 1) go missing or 2) terminate”. This choice is of course to make the search over hypotheses feasible in practice. If we allow for an arbitrary number of splits/mergers while targets are missing, then the number of hypotheses to examine increases too fast for even modest problems. Thus, such simplifications are necessary. As Adaska points out, this prevents us from identifying the correct hypothesis if two targets merge between frames; the new merged target is missing at the next frame. It is also true that this event can be approximated by the event that both targets are missing, then merge together in the next frame. The criticism is that this approximate event might have low likelihood. If the probability of missed detections is very small this could become a reasonable concern. However, in the 2D turbulence example, where the probability of missed detections is rather small, this exact situation can happen occasionally. That is, two vortexes start to merge and become distorted, thus causing the partially merged vortexes to show up missing. Even in cases such as this, we have had good success recovering the true event or the correct approximation to it. Also, it is hard to objectively define what is the truth in these cases; i.e., are there two missing targets or only one?

Lastly, there is concern about the likelihood calculation which requires the inversion of a matrix whose size depends on the length of the track. Because of the dependence introduced by splitting and merging, it is not possible to apply traditional filtering techniques to this model. Yes, this could be problematic for some applications if a simplification is not made. However, this issue can easily be handled in practice by simply ignoring the past beyond a certain point. The covariance of an integrated Brownian motion decays like  $t^3$  for example. There is very little extra information about a new observation in a track contained by an observation, say 20 time steps prior. That is, at time  $t_{21}$  the  $X(t_j)$ ,  $j = 2, \dots, 20$ , from the previous 19 time steps can tell us approximately all there is to know about the conditional distribution of  $X(t_{21})$ . In that case, we can safely ignore  $X(t_1)$  for practical purposes. Therefore, we can calculate likelihoods with a moving window when it is necessary to process data in real time, or if there are just a large number of frames. This can be accomplished by treating the “initial” locations in the first frame of the moving window as known according to the currently stored hypotheses. Also, as stated in the Supplement page 6, it may be possible to apply the innovations algorithm in a creative manner to this



problem to avoid the matrix inversion altogether. In either case, the method can be made feasible for sequential processing if necessary.

*Choi and Wolfe:* We are thankful for the interesting discussion by Choi and Wolfe of the challenges in the data association problem and the use of efficient batch algorithms to solve it. Specifically, they point out that the heuristic “ $K$ -best” hypotheses approach can fail under certain circumstances. They suggest that a Markov Chain Monte Carlo approach may capture the posterior probability more accurately, but the best way to utilize this approach is still an open problem.

We fully agree that the algorithm used in SLHN for finding likely hypotheses could be improved and this is a topic of further research. One practical issue with respect to the number of hypotheses, at least in the storm and vortex applications, is the number of time steps and size of the region involved in the processing of the data. For example, one might want to know the probability of a merger of two targets in a certain sub-region of time and space. Because of the number of targets and time points involved in the entire batch processing of the data, all of the  $K$  hypotheses kept may not contain enough information to distinguish them from this particular merging event. However, one could consider evaluating more alternative hypotheses by perturbing existing hypotheses “locally” in time and space to evaluate more precisely how likely a merger was in this particular case. This is still a heuristic approach to the posterior evaluation, but does provide a satisfactory solution to this particular question.

*Doucet and Vo:* We thank Doucet and Vo for their important discussion on the static parameter estimation issue and more challenging tracking problems. We discuss the static parameter estimation further as well in response to Singh. In very challenging tracking scenarios, where there are hundreds of targets and high clutter intensity, the random set/PHD approach has shown promises. Of course one can always use the approach we have taken in SLHN on more challenging problems. Whether we use MHT or Monte Carlo methods to approximate the posterior distribution, a major issue is the trade-off between computation time and approximation accuracy. When faced with any tracking problem, it is important to carefully consider all of the strengths and weaknesses of the available tracking algorithms/methods. More theoretical results considering algorithmic complexity and statistical convergence properties would certainly aid with this decision. However, there is no substitute for designing a simulation study that is similar (in terms of number of targets and clutter, etc) to the actual tracking problem at hand. One can then apply several of the viable algorithms (appropriately tuned to satisfy computation time constraints) and compare their performances. In many applications we suspect there will be a clear winner, thus making the implementation decision relatively simple.

*Singh:* We very much appreciate Singh's points. In particular, he provides excellent discussion about batch versus real-time processing, static parameter estimation, and a similar problem to our storms application for comparison. As Singh points out, we propose a model that explicitly allows for the merging of targets and automatically prohibits the merging of distant targets. In our view this is much more realistic than any of the existing approaches that attempt to allow for merging of targets. In Figure 3 of SLHN we make explicit the connection between our model and the physical reality of a merger. We feel that the description given there is actually a very good representation of the actual event.

It is true that when observations are coarsely spaced in time, some of the approximations made in the likelihood calculation could become less accurate. For example, suppose targets move quickly relative to the coarse time increment. Our approximation assumes that splitting events occur during the midpoint of the time increment. If, in reality, the split happens close to the beginning of the time increment, then it might allow the two children to get much further away from each other than our approximation would find likely. However, the use of a discrete model in place of our continuous model does not rectify this problem. It puts the burden of approximation in the modeling step, as opposed to the likelihood calculation. The reality is that storms, and most other targets in tracking applications, move and change state continuously in time. Any discrete time model is necessarily an approximation to the true dynamics. Often it is not possible to account for the error imposed by incorrectly modeling the process. Thus a major advantage to the continuous time model is that the process can be modeled more accurately and intuitively. Further, any approximations we have to make in the likelihood can be studied asymptotically to at least partially address the approximation issue. An asymptotic study of this model addressing these concepts in particular is provided in Storlie et al. (2008).

The consequences of incorrectly specified values of the static parameters is a largely ignored issue in the literature. How accurate the estimates of these static parameters are, and to what extent these estimates affect the track estimate, are important questions. These questions are at least partially addressed in Storlie et al. (2008), but for coarse time increments we are left without much knowledge about these estimates. Singh is promoting a fully Bayesian approach to this problem, which is certainly warranted in many cases. Practitioners are often familiar with the range for the number of targets and/or false alarms in their problems, etc. In these cases it is reasonable to provide boundaries for an uninformative prior, or even more informative prior distributions, for the values of the corresponding static parameters. The uncertainty in the values of these parameters can then be incorporated into the MCMC sampling scheme to build the posterior distribution of possible tracking solutions. When we wish

to batch process an image sequence, this is a viable alternative. In addition, sequential Monte Carlo techniques (e.g., particle filtering as described by Doucet, de Freitas and Gordon (2001)) could possibly be used when images must be processed sequentially. As, mentioned in the discussion of Doucet and Vo, this is still an open problem. In either case, however, adding more complexity to the estimation problem adds computation time. These approaches are the most appropriate when some prior information is known about static parameter values and the problem is small enough to make computation feasible.

We conclude by giving a computation time comparison to the example given in the discussion. The storm example as presented on the web (<http://www.stat.colostate.edu/~tlee/tracking/>) has 48 images over the course of 12 hours (only a subset was presented in SLHN), roughly twice the sample size of the example used by Singh. Also, allowing for the added complexity of splitting and merging roughly quadruples the computation time, at least with our implementation in this example. Our code is written in *R* and is also not optimized for efficiency. Still, on an Athalon 2,000 processor (1.67 GHz), this example took two hours to analyze. There are also a number of other control parameters that can make a difference here (burn-in and number of samples for MCMC, number of hypotheses kept for MHT, etc), so this is not necessarily an apples-to-apples comparison. Still, it seems that the fully Bayesian approach is substantially more computationally intensive. In this small, illustrative example, the fully Bayesian approach is certainly feasible. To analyze many months or even a year's worth of storm images, however, may not be feasible.

*Wu and Wu:* We thank Wu and Wu for their provocative discussion of *trackability* and *distributed inference*. First, we fully agree that theoretical justification is always helpful, and some is provided to our model choice in Storlie et al. (2008). In terms of the heuristic search, we also agree that there is room for improvement in this area. As pointed out by Choi and Wolfe, and by Doucet and Vo, this is an open problem. We appreciate the description of the interacting individual trackers approach as an alternative to the joint hypothesis approach. This does seem to have some promise in terms of alleviating algorithmic complexity in the optimization problem posed. However, it does seem that the optimization problem posed become less intuitive in order to gain this computational advantage.

We very much agree that the observation model (we call it the location model or motion model) is the most important part of the overall tracking model. If this model does not represent the actual physical motion well, then having a good event model for birth/death/split/merge will not help much. Thus, the beast of burden in our approach still remains the choice of the location model. We want to emphasize though, that our event model is completely flexible to allow for any location model that the modeler requires for his/her targets. Although, to be

completely candid, the location model should be a Gaussian process; otherwise, a difficult and/or computationally intensive likelihood evaluation would be necessary because of the conditioning on mergers. We would like to point out the flip side of this argument as well. If the motion of the targets is very well described by the location model, but the other dynamics (such as splitting and merging) are not modeled well or at all, then this can also have substantial adverse effects on the results as well. This would be particularly true in the applications described in SLHN.

## References

- Doucet, A., de Freitas, N. and Gordon, N. (2001), *Sequential Monte Carlo Methods in Practice*. Springer Verlag, New York.
- Storlie, C., Hannig, J. and Lee, T. C. M. (2008), The asymptotic properties of a multiple target tracking model. Submitted for publication.

Department of Mathematics and Statistics, University of New Mexico. MSC03 2150, 1 University of New Mexico, Albuquerque, New Mexico 87131-0001, U.S.A.

E-mail: storlie@stat.unm.edu

Department of Statistics, The Chinese University of Hong Kong, and Department of Statistics, Colorado State University, U.S.A.

E-mail: tlee@sta.cuhk.edu.hk

Department of Statistics, Colorado State University, Fort Collins, Colorado 80523, U.S.A.

E-mail: hannig@stat.colostate.edu

Geophysical Statistics Project, National Center for Atmospheric Research, P.O. Box 3000, Boulder, CO 80307-3000, U.S.A.

E-mail: nychka@ucar.edu

(Received October 2008; accepted November 2008)