A Tracker Adjunct Processing System for Reconsideration of Firm Tracker Decisions

David J. Trawick, Benjamin J. Slocumb and Randy C. Paffenroth

Numerica Corporation, 4850 Hahns Peak Drive, Suite 200, Loveland, CO, USA

ABSTRACT

Most modern maximum likelihood multiple target tracking systems (e.g., Multiple Hypothesis Tracking (MHT) and Numerica's Multiple Frame Assignment (MFA)) need to determine how to separate their input measurements into subsets corresponding to the observations of individual targets. These observation sets form the tracks of the system, and the process of determining these sets is known as data association. Real-time constraints frequently force the use of only the maximum likelihood choice for data association (over some time window), although alternative data association choices may have been considered in the process of choosing the most likely.

This paper presents a Tracker Adjunct Processing (TAP) system that captures and manages the uncertainty encountered in making data association decisions. The TAP combines input observation data and the data association alternatives considered by the tracker into a dynamic Bayesian network (DBN). The network efficiently represents the combined alternative tracking hypotheses. Bayesian network evidence propagation methods are used to update the network in light of new evidence, which may consist of new observations, new alternative data associations, newly received late observations, hypothetical connections, or other flexible queries. The maximum likelihood tracking hypothesis can then be redetermined, which may result in changes to the best tracking hypothesis. The recommended changes can then be communicated back to the associated tracking system, which can then update its tracks. In this manner, the TAP's interpretation makes the firm, fixed (formerly maximum likelihood) decisions of the tracker "softer," i.e., less absolute. The TAP can also assess (and reassess) track purity regions by ambiguity level.

We illustrate the working of the TAP with several examples, one in particular showing the incorporation of critical, late or infrequent data. These data are *critical* in the sense that they are very valuable in resolving ambiguities in tracking and combat identification; thus, the motivation to use these data is high even though there are complexities in applying it. Some data may be *late* because of significant network delays, while other data may be *infrequently reported* because they come from "specialized" sensors that provide updates only every once in a while.

Keywords: Multiple Hypothesis Tracking, Uncertainty Management, Dynamic Bayesian Networks, Late Data Association

1. INTRODUCTION

One requirement of modern tracking systems is that they provide the best overall situational awareness given the data and the processing time available. In order to stay current, the analysis that can be performed must be real-time for the computing resources present in the system. Trackers in a networked environment are under even more time pressure, where they must coordinate with other remote trackers to maintain a Single Integrated Air Picture (SIAP).

These constraints imply that a lot of interesting, even critical, analysis cannot practically be performed by the tracker. For example, newly arrived data may imply that firm decisions made in the past (such as data association decisions) were incorrect and should be changed. Multiple frame tracking systems, such as Multiple Hypothesis Tracking (MHT),[1] address this data association issue for a small number of frames. Numerica's Multiple Frame Assignment (MFA) tracking system[2, 3] is a specific example of this type of state-of-the-art tracking system. These systems, however, can neither reconsider nor revise any firm decisions made prior to the current window of frames.

Further author information: (Send correspondence to D.J.T.)

D.J.T.: E-mail: david.trawick@numerica.us, Telephone: 1 970 461 2422 x223

B.J.S.: E-mail: benjamin.slocumb@numerica.us, Telephone: 1 970 461 2422 x212

R.C.P.: E-mail: randy.paffenroth@numerica.us, Telephone: 1 970 461 2422 x233

The key innovation in our design presented herein is to divide the responsibilities of a tracking system into those that must always be kept up-to-date and those that involve non-real-time decisions. This organization allows us to process certain computations outside the tracking system proper, either as time allows or as needed by the tracking system. We have identified this tracking system support process as the *Tracker Adjunct Processor*, or TAP.

Some examples of non-real-time tracking tasks suitable for the TAP include the following: (i) handling of critical, late or infrequently reported data (the focus of this effort); (ii) reconnection of broken tracks; (iii) termination of tracks no longer being observed by the sensor; (iv) identification of redundant or spurious tracks caused by not correctly associating local sensor data to system tracks; (v) detection of the "incompatible" fusion of multi-sensor data to system/network tracks based on incongruous kinematic or feature properties.

The critical, late, or infrequently reported data issue may arise as follows. The data are *critical* in the sense that they are very valuable in resolving ambiguities in tracking and combat identification; thus, the motivation to use these data is high even though there are complexities in applying it. Some data may be *late* because of significant network delays, while other data may be *infrequently reported* because they come from "specialized" sensors that provide updates only every once in a while. This scenario is illustrated in Figure 1.

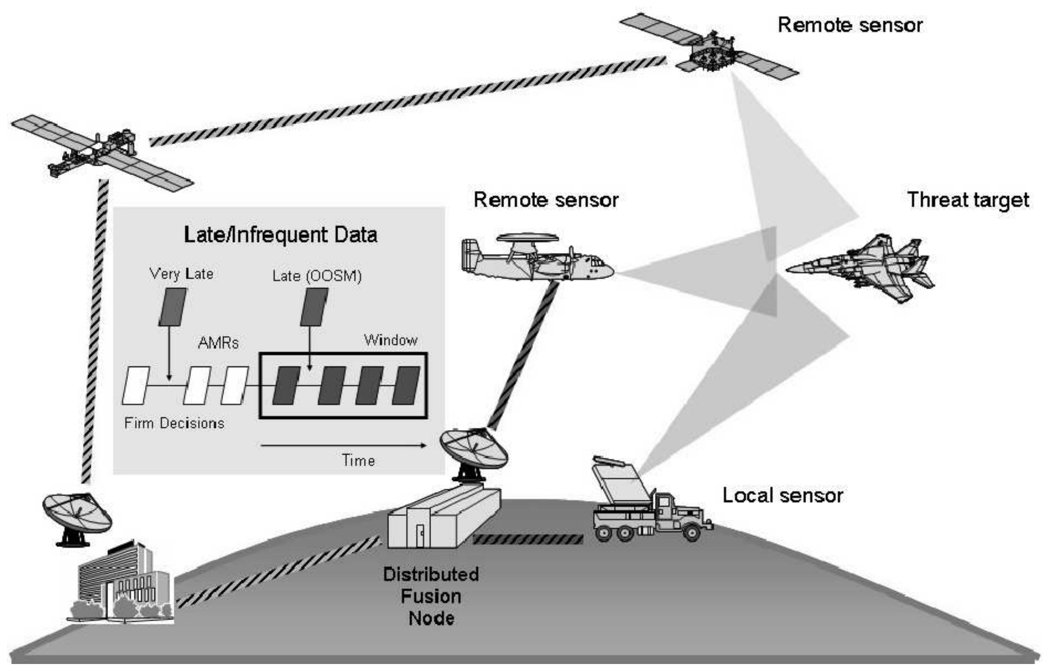

Figure 1. Diagram showing the arrival of late/infrequent data at a networked tracking/fusion node. Data arrive at the local node from the remote airborne surveillance radar late because it has a long scan period compared to the local electronically scanned array radar. Data arrive very late from the satellite-based sensor because they must be routed via satellite links, processed at ground stations, and then forwarded to the local node. Both remote platforms can provide critical information that can be used to resolve data association and combat ID ambiguities.

The TAP incorporates critical, late, or infrequently reported data by determining when (i.e., where in time) the data are to be added, predicting/retrodicting the ambiguous tracks to that time for the best re-estimates of the tracks at that time, solving an association problem to assign the new data to the existing tracks, and finally re-estimating the ambiguous tracks newly incorporating the late data. We give some examples of this process in Section 5. As appropriate, a re-estimate can be made of the best track hypothesis. The TAP then coordinates with the associated MHT tracking system to determine if any changes to the MHT's track hypotheses are desired as a consequence of the new information. This may in turn result in messages to be sent to other network peers in order to maintain SIAP.

The TAP has as its core component Numerica's *Bayesian Network Tracking Database* (BNTD).[4,5] The BNTD uses analysis from the associated MHT tracker to form a Bayesian Network[6–8] representing the input observation data and the ambiguous multiple data association hypotheses from the tracker. It can then use Bayesian network methods to propagate the impact of new evidence throughout the network, and report on the consequences of such new data or hypothesized data to the track picture. This forms a very flexible and general approach for addressing the non-real-time issues described above. In particular, the BNTD can reconsider, undo, and redo the "firmly established" decisions of the MHT tracker in light of critical, late/infrequently reported data. The TAP can then send these recommended changes back to the MHT tracker.

Of special importance is the BNTD's processing of attribute data. Although the kinematic data are of fundamental importance, and late kinematic data can change the interpretation of the track picture, the information provided by kinematic data loses relevance quickly. Attribute data, however, change less frequently over time and may drastically change our interpretation of the current track picture. The details of the BNTD's processing of attribute data are discussed below in Section 3.

2. BAYESIAN ESTIMATION BASICS

Here we formulate a very general approach to state estimation. This is designed to be suitable for the problems we desire to address and more. Specifically, we want a method that allows for reconsidering our estimates at any time given newly arrived input evidence from any time. This model is meant to be general, but may not always be appropriate as a continually updated part of a real-time system.

A Bayesian approach to tracking and smoothing yields a factoring of the probability calculations involved that may be implemented in a graphical network. Ambiguous data associations among observations are modeled by multiple connections among observations in the network, yielding ambiguous tracks. New observation data are added to the network with connections representing possible data associations to observations already in the network.

2.1 Bayesian Tracking and Smoothing

Suppose we want to consider a sequence of observations $z_{1:N}$ as all associating with each other, i.e., as forming one track. Then we can calculate the state PDF $p(x_i|z_{1:N})$ for the estimated state x_i corresponding to the time of the observation z_i by first factoring this expression into four terms. The following derivation uses repeated applications of the conditional probability definition, Bayes' Rule, and the assumption that the observations are made independently of each other.

$$\begin{aligned}
p(x_i|z_{1:N}) &= p(x_i|z_1, ..., z_i, ..., z_N) \\
&= p(x_i|z_{1:i-1}, z_i, z_{i+1:N}) \\
&= p(x_i, z_{1:i-1}, z_i, z_{i+1:N})/p(z_{1:i-1}, z_i, z_{i+1:N}) \\
&= \frac{p(x_i|z_{1:i-1})p(x_i|z_i)p(x_i|z_{i+1:N})}{p(x_i)^2}.
\end{aligned}$$

We then define the four parts of this final factoring as follows:

$$\begin{aligned}
p_L(x_i) &= p(x_i|z_{1:i-1}), \\
p_C(x_i) &= p(x_i|z_i), \\
p_R(x_i) &= p(x_i|z_{i+1:N}), \text{ and} \\
p_0(x_i) &= p(x_i).
\end{aligned}$$

Then we have:

$$p(x_i|z_{1:N}) = \frac{p(x_i|z_{1:i-1})p(x_i|z_i)p(x_i|z_{i+1:N})}{p(x_i)^2} = \frac{p_L(x_i)p_C(x_i)p_R(x_i)}{p_0(x_i)^2}.$$

The factoring of the term $p(x_i|z_{1:N})$ into the left, center, right, and prior terms can be loosely thought of as follows. The state distribution at time i corresponding to some observation z_i in an established track is the product of the distribution given only the observation z_i, the distribution of all data before z_i, and the distribution of all data after z_i. All of these terms depend on some prior distribution $p(x_i)$, so to keep from counting this prior multiple times it is appropriately normalized.

Under the appropriate conditions[5] (a linear dynamic system, with Gaussian density states and errors throughout) this factoring can be interpreted as the Kalman Filter: p_L provides the state propagation, p_C provides the measurement or observation function, and the multiplication provides the update step (p_R is not involved in strictly forward filtering). Under these same conditions, this factoring can be interpreted as the forward-backward variety[9,10] of Kalman Smoothing, where p_R is the backward (retrodicted) propagation. Future work will assess the non-Gaussian case.

The prior term p_0 is interpreted as diffuse or uninformative[11] in both the filtering and smoothing cases for kinematic state estimation. This represents what we know about the kinematic state of a target before observing it: essentially nothing. This term is normalized away during the estimation process.

2.2 Tracking/Smoothing Using a Dynamic Bayesian Network

The factoring of the state estimate above gives pieces suitable for recursive computation, as reflected in the Kalman filter/smoother interpretation. The parts of this factoring also provide an organization of the filtering/smoothing problem into a graphical network model. With the correct linear/Gaussian assumptions this can be cast directly into a dynamic Bayesian network.[8] The multiplications of Gaussian PDFs represented by this network are exactly the mean and covariance calculations given by the Kalman smoothing equations. Figure 2 shows an expanded example of the dependencies represented in this network.

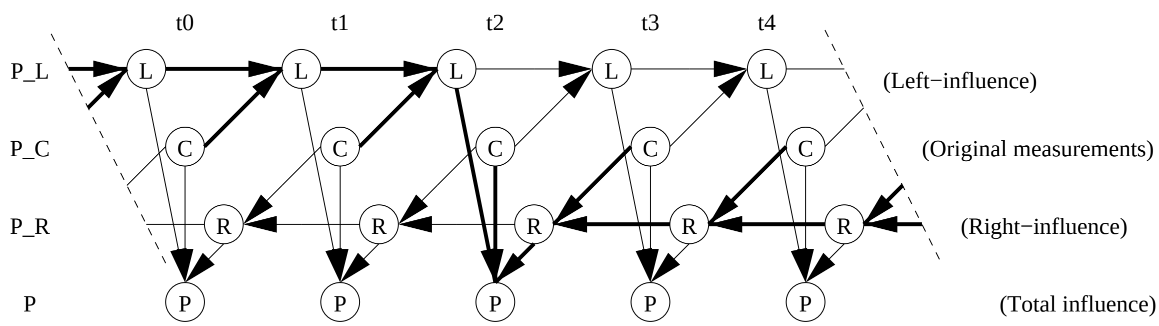

Figure 2. **Markov Network produced by the factoring of the state estimate at a given time.** The bold arcs in the network show the propagation of evidence/influence on the state estimate at time $t2$. A state estimate from all the observations earlier in time (from the "left") is propagated to the time $t2$, as is an estimate from all the observations later in time (from the "right"). These estimates are combined with the observation at time $t2$ to form the smoothed state estimate at that time. Note that although the dependencies appear complicated, there are no loops in this network; all estimation ends in a "P" state, thereby showing that the graph is Markovian.

Even without the linear/Gaussian assumptions, this Markovian organization remains useful since it specifies all the dependencies of state estimates on other input data and state estimates. This allows us to use alternative motion models and filtering approaches, which can be chosen to suit the application domain. For example, we may use an Interacting Multiple Model (IMM) smoother[12] in place of the Kalman smoother. The network structure will remain the same, only the underlying filters and models in use will change.

2.3 Modeling Ambiguous Data Association

Thus far we have only considered tracking/smoothing for a single observation sequence. The Bayesian network approach allows for a natural method of modeling ambiguous data association, by keeping ambiguous connections among observation nodes in the network graph.

While the branching of possible alternative data association paths is a common technique in MHT systems, the BNTD takes this a step further to model also the convergence of possible alternatives where the observation sequence no longer differs, i.e., is no longer ambiguous. This convergence of state estimates corresponds to an addition of PDFs, weighted by the relative strength of the data association connections involved.

In order to determine the relative strength of the data association connections, we begin by finding the k-best local data association hypotheses among the frames involved. For this we use the input data association problems from the associated tracker. The k-best hypotheses are found using Murty's method.[13,14] The k-best hypotheses are then "averaged" by building a graph that is a combination of all the k-best hypotheses, where the weight on an arc of the graph is the sum of the probabilities of the hypotheses to which it belongs. The "averaging" process is illustrated in Figure 3. The details of this process are given in a previous report.[5]

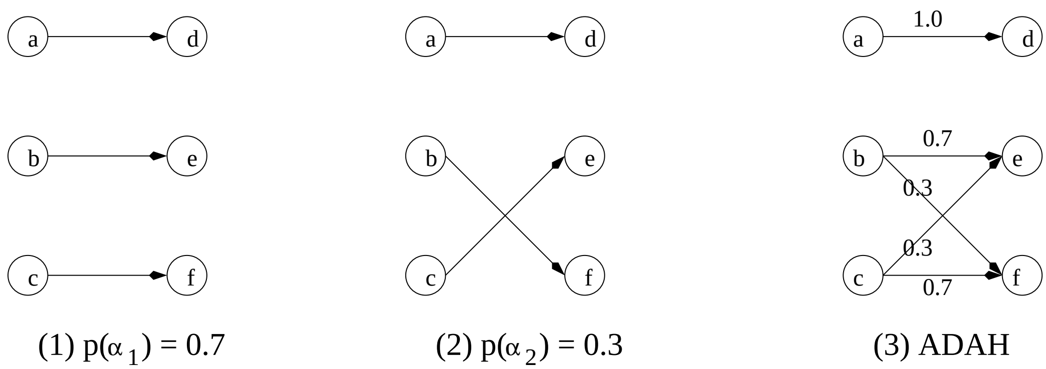

Figure 3. **Calculation of the averaged data association hypothesis (ADAH).** The k-best analysis of a data association problem yields two hypotheses. These two data association hypotheses, α_1 with $p(\alpha_1) = 0.7$ in (1) and α_2 with $p(\alpha_2) = 0.3$ in (2), are "averaged" to form the ADAH graph in (3). Note that the "sum" of the hypotheses forms a new, averaged data association hypothesis graph with all the connections of the original hypotheses. The averaging process gives a weight to all connections of the original hypotheses. (The 0 weight connections are eliminated from the graph representation. The observations a, b, and c are at some time t_1, and the observations d, e, and f are at a later time t_2. The arrows of the graphs indicate time going forward.) The data associations show path convergences at observations e and f.

Figure 3 also shows the convergence of possible alternative data association paths. The predicted state estimates at these times are formed using a weighted sum of the PDFs from the ambiguous sources, in effect saying the predicted state comes from an "or" of the possible contributing states. This sum of PDFs is formed using Gaussian moment matching,[11] in the same manner as the mixing of PDFs in IMMs, weighted by the data association connection weights calculated previously.

2.4 Incorporating Late Kinematic Data

Normally the BNTD receives observation data from the associated tracking system in roughly sequential order with occasional out-of-sequence-measurements (OOSMs). These reports come interspersed with initial, ambiguous, possible data associations among the recently received observations. For data that arrive outside of the tracking system's window there are no data associations available. It is for this case that the BNTD requires extended functionality in order to insert these data into the network with the appropriate possible data associations.

Data are sent to the BNTD in the form of Associated Measurement Reports (AMRs). Typically these will include an association of an observation to a specific track ID. If this is the case, the data associations for this observation are already determined; they are the observations in that track ID. The new observation is placed into the network and connected in chronological sequence to that track's observations.

However, the AMR may be of the form that does not give a track association to the observation. Moreover, we may want to reevaluate the track association as determined by the rest of the system. Accordingly, potential data associations must be determined for associating the new data with the existing data of the network.

Incoming frames of late data cannot be expected to fit neatly into the existing sequence of frames already received. For example, late data frames may extend in time through several of the existing data frames. In this case, the context of plus or minus two frames (for example) may not be enough for a data association analysis for the late data frame, and such a fixed length frame context is seen to be inappropriate in general.

Although we do not know which frames are before and after the late data, or even if the frames can be arranged sequentially, we do know which arcs of the network cross the times of the late data observations. These arcs are existing data associations among observations of the network and are pieces of the ambiguous tracks of the BNTD. These ambiguous tracks are what we want to attempt to associate with the late data. Thus, an assignment problem is setup between these tracks and the late data observations. Figure 4 shows the process of adding late kinematic data to the network.

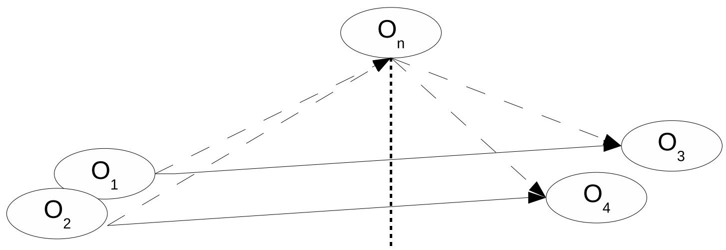

Figure 4. **Adding Late Kinematic Data to the Network.** Here the existing network has O_1 connecting to O_3 and O_2 to O_4. Then the new, late kinematic observation O_n arrives and needs to be incorporated into the network. A data association problem is built to consider the possible associations of the new O_n with the existing (ambiguous) tracks. O_n is at a time corresponding to the vertical dotted line, and thus could potentially be part of any of the two connections this line is shown to intersect. The data association problem then has $(O_1, 0, O_3), (O_2, 0, O_4)$ (the existing connections), and $(O_1, O_n, O_3), (O_2, O_n, O_4)$, the new potential connections.

The tracks chosen here may overlap in a way that complicates the assignment problem. In particular, some of these arcs may have the same starting observation (or ending observation, but not both). We setup the assignment problem to solve this arbitration among arc starts and ends by using a three-dimensional assignment problem. The first and third dimension are the start and end observations of the candidate arcs, and the middle dimension is composed of the late data. The assignment problem starts by including the track arcs with no association to the late data. Then the late data are added by forming potential associations with the track arcs whose time in which they are contained. The costs of these existing and potential associations are computed from the new observations and the state estimations of the network.

As described so far, the arcs of the assignment problem developed for the late data are of the form (i, j, k), where i and k select the observations at the beginning and ending of the existing track arc of the network, and j selects the observation of the late data frame. The assignment problem also includes $(i, 0, k)$, representing where the late data frame is skipped.

The costs for the assignment problem are determined by predicting the existing tracks to the time of the new observation and calculating the observation to track assignment cost. The k-best hypotheses are retrieved from the assignment problem's solutions, and an "average" solution is formed. Arcs in this solution with a low enough cost* are added to the network. Arcs with too high a cost are not added.

Two more details should be mentioned about how this assignment problem is built. Virtual observations (not corresponding to real data) are kept in the network at times positive and negative infinity to provide network connectivity to the start and end of tracks. In the network, the beginning of a track is represented by an arc from the virtual observation at the time of negative infinity to the first observation of the track. Similarly, the end of a track is represented by an arc from the last observation of the track to the virtual observation at the time of positive infinity. These (infinite) arcs may be among those that cross the time of the late data observations. When inserted into the assignment problem, these virtual observations should allow multiple associations; thus, they are given the zero index in the assignment problem. Newly arrived late data may then be associated with a track prior to its current beginning or after its current end.

Finally, we would like the assignment problem to allow for the possibility of late data connecting currently broken tracks. This is modeled by adding arcs to the assignment problem of the form (i, j, k) whenever arcs of the form $(i, j, 0)$ and $(0, j, k)$ have already been added; that is, whenever the late data can associate with both the end of some track and the beginning of another.

*Costs in our formulation are negative log-likelihoods, thus smaller values are "better."

Thus, the incorporation of late data into the network is only slightly more complicated than the normal process of adding new data. The late data may require the BNTD to produce and solve a data association problem, something it normally receives from the associated tracker. Once added to the network, the network is updated by propagating the consequences of the new evidence throughout. We note that this insertion of late data right when/where it belongs is exactly what most out of sequence measurement (OOSM) techniques[15,16] are attempting to approximate.

2.5 Updating the Associated Tracking System

If the BNTD has discovered anything interesting or important from its analysis of the data, it needs to communicate this back to the associated tracking system. This process is accomplished in two steps.

First, the BNTD must determine if it has any changes to the tracking picture. It does this by finding its best tracking hypothesis from the network. An MHT-like algorithm sweeps through the network to find this best hypothesis. This algorithm is smaller and faster than the typical MHT, since it does not have to generate possible data associations among the input observations; it can read these from the Bayesian network.

Then, the BNTD compares its best hypothesis with that of the tracker. If there are significant differences, a "perturbation message" is prepared to send this information to the tracker. The message is prepared so as to require the fewest changes to the tracker's track picture to bring it into agreement with the best hypothesis of the BNTD.

3. INCORPORATING ATTRIBUTE DATA

We follow here the distinction between feature and attribute data made by Drummond,[17] where attributes refer to discrete space characteristics of a target, as opposed to features, which refer to continuous characteristics. An attribute, such as a target classification or ID, is what one would expect from extended external analysis producing critical, late data.

The modeling of attribute data in the TAP/BNTD follows closely the methods for kinematic data above. A Bayesian approach to attribute state estimation is used, and this analysis is combined with the kinematic estimation in the network. The propagation of attribute evidence may then be applied to greatly reduce the ambiguity of the network.

3.1 Bayesian Attribute State Estimation

Following our earlier analysis, only this time estimating a discrete attribute state from a sequence of associated input observations or measurements, and using our previous notation we have:

$$p(x_i|z_{1:N}) = \frac{p(x_i|z_{1:i-1})p(x_i|z_i)p(x_i|z_{i+1:N})}{p(x_i)^2} = \frac{p_L(x_i)p_C(x_i)p_R(x_i)}{p_0(x_i)^2}.$$

Now we analyze these separate terms in more detail specific to the attribute state estimation problem.

3.1.1 The Prior Term $P_0(x_i)$

Although we argued that the kinematic prior was uninformative and therefore ignorable, for attributes this is not typically true. For example, an attribute classifying targets into friend, foe, and neutral, would have prior expectations highly dependent on the location of the sensor. We should expect a higher proportion of friends over Kansas than we would in an active war zone. Thus, we specify the prior for any attributes we want to model as an adjustable, scenario-dependent input parameter. Finally, we also assume that the prior is constant for the entire scenario, i.e., $p_0(x_i) = p_0(x)$ for all i.

3.1.2 The Central Term $P_C(x_i)$

The central term $p_C(x_i) = p(x_i|z_i)$ describes the characteristics of the measurement process and may be rewritten as $\frac{p(z_i|x_i)p(x_i)}{p(z_i)}$. The term $p(z_i|x_i)$ captures the sensor's abilities and should be an input parameter for the sensor in use. The $p(x_i)$ term is the prior, which may cancel out in the expression for $p(x_i|z_{1:N})$ above. The $p(z_i)$ term is calculated using the law of total probability:

$$p(z_i) = \sum_{i=1}^{N} p(z_i|x_i)p(x_i).$$

This is generally easy to compute in the (discrete) attribute case. Conceptually, the $p(z_i)$ term provides a suitable weighting for the measurement probability $p(z_i|x_i)$.

3.1.3 The Left and Right Terms, $P_L(x_i)$ and $P_R(x_i)$

We begin by considering the left term, knowing that the right term will be symmetric. The attributes we choose to model generally do not change over time; friends stay friends and a target's ID stays the same. From this we can conclude that:

$$\begin{aligned} p_L(x_i) &= p(x_i|z_{1:i-1}) \\ &= p(x_{i-1}|z_{1:i-1}) \\ &= p(x_{i-1}|z_{i-1})p(x_{i-1}|z_{1:i-2})/p(x_{i-1}) \\ &= p_C(x_{i-1})p_L(x_{i-1})/p_0(x_{i-1}). \end{aligned}$$

This recursion starts with the prior, so that $p_L(x_0) = p_0(x)$.

If the network has multiple paths into x_i, showing track ambiguity, then the several $p_L(x_i)$ terms from different previous paths must be weighted and summed to form the total $p_L(x_i)$. The weights are determined from the average data association hypothesis given in Section 2.3. Note that these weights represent the existing, previously computed ambiguous data association probabilities (typically beginning with kinematic only data).

3.1.4 Putting it all together

By way of summary, if we were to put together all of the previous analysis for a sequence $z_{1:N}$ of N singly associated observations of some attribute, we get the product for any $i \in \{1, \ldots, N\}$:

$$p(x_i|z_{1:N}) = p_0(x)\frac{p(z_1|x_1)}{p(z_1)} \cdots \frac{p(z_i|x_i)}{p(z_i)} \cdots \frac{p(z_N|x_N)}{p(z_N)}.$$

We note that although this is similar to the kinematic formulation, what is represented is very different; the probability distributions are all discrete, and the priors all play a significant part.

3.2 Incorporating Attribute Data in the Network

In order to know where the attribute data belong in the network, it is expected that attribute measurements will arrive along with corresponding kinematic measurements from the sensor. For example, an IFF sensor will typically return both kinematic data and several IFF codes indicating the identity and class(es) of the aircraft. With the kinematic data we can determine where to put the new observation into the network, as was done with the late kinematic data. This is only necessary if the observation is not already associated to a track by the tracker when sent to the TAP.

The attribute state estimation/smoothing steps take place along with the standard kinematic state estimation/smoothing. The kinematic and attribute update calculations are kept separate; at this stage they have no influence on each other, except that the attribute calculations use the existing data association probabilities, which may be kinematic-only based if all the attribute data are late. We show how everything is put together in the next Section.

4. UPDATING THE NETWORK DATA ASSOCIATION PROBABILITIES

Updating the full network includes updating all the estimated states, kinematic and attribute, as well as updating the data association probabilities that provide the relative strength of the ambiguous tracks of the network. All the states are re-estimated using the forward-backward smoothing techniques represented by the factored probability estimates as discussed in Section 3 for attributes and in the Section 2 for kinematic values. This smoothing relies on the existing data association probabilities; the next step is to update these association probabilities based on the updated kinematic and attribute states.

We note that this step implicitly puts loops into our Bayesian network. Such loops destroy the simple convergence properties of the Markov network, and several methods have been devised to address this situation.[6,7,18] We prefer to perform the updates of the states and data association likelihoods as separate steps of a version of the EM algorithm:[18,19] (i) calculate the expected value of the data association probabilities, and (ii) maximize the likelihood of the state PDF parameters. The initial results of step (i) come from the associated tracker's input to the TAP. The update of the Bayesian network previously described forms step (ii). Subsequent updates of step (i) are accomplished as described below.

Starting likelihoods for data association can be determined from the updated states. For the kinematic states, this likelihood is the usual data association likelihood[5] (basically, an estimate that the state means are the same when predicted to the same time, using a Gaussian distribution formed from the two same time state covariances). A match score $m(A, B)$ for the attribute states $A = p_A(x_i)$ and $B = p_B(x_i)$ of the attribute x_i for $i \in \{1, \dots, N\}$ is:

$$m(A, B) = \sum_{i=1}^{N} p_A(x_i) p_B(x_i).$$

The kinematic and attribute data association likelihoods between two states are then multiplied for use as the data association likelihood of the connection between these states. These likelihoods are used in an update pass through the data with the MHT-like algorithm of Section 2.5, only now we keep and use the multiple hypotheses that are found. These multiple hypotheses are treated like the k-best results of other data association problems. We derive individual track probabilities by summing the probabilities of the hypotheses that contain that track. Then the connections in the network are updated with the probabilities of the tracks to which they belong.

5. EXAMPLES

In this section we show several simple examples highlighting the application of typically late attribute data to the resolution of tracking picture ambiguity. These show how the TAP refines the understanding of the track picture, reinterpreting or possibly changing the firm decisions of the associated tracking system. We also show the output of the TAP for a more complex, realistic simulated scenario.

5.1 Simple Crossing Scenario

We start with a simple example of two targets. They approach each other and may or may not cross with equal likelihood. The targets' identities are known prior to the potential crossing from timely attribute data. Figure 5 shows the scenario prior to the arrival of late data. On the right side, the colors of the targets are unknown, since the targets may have switched positions with each other. The attribute data determined for the right side shows this indeterminacy; the probabilities for red and blue are equal for both targets.

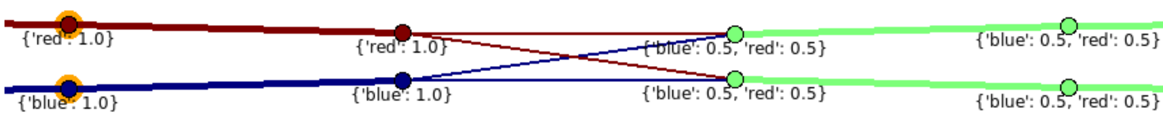

Figure 5. **Simple Crossing Scenario.** Two targets, identified as "red" and "blue" in the earlier (left) portion of the scenario, travel close enough to each other to yield an ambiguous crossing. Crossing or not crossing is equally likely in this case. Neither target's identity is known after the potential crossing. The highlighted observations show where the original attribute data were observed.

Late data then arrive and report that the upper target was "red." There are known to be only two targets in the system, one of which is red; thus, the other must be blue. Furthermore, the targets did not switch, because that would result in a change of colors. This resolves the ambiguity of the entire scenario; there is no potential crossing ambiguity remaining in Figure 6.

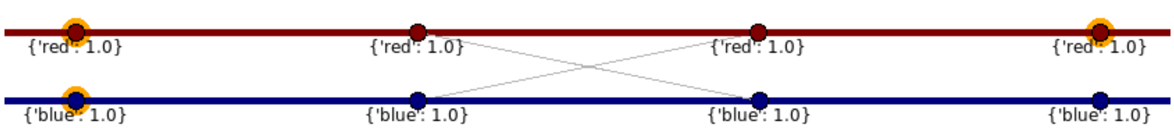

Figure 6. **Simple Crossing Resolved.** In the same scenario as in Figure 5, the targets' identities are now resolved after late data reported that the upper target was "red." The highlighted observation on the right shows where the late attribute information was applied. The thin crossing connections now show that the likelihood of crossing is negligible.

5.2 S-Shaped Ambiguous Scenario with Resolution

In the next scenario, three targets arrive from the left and are known to be targets “A,” “B,” and “C.” B and C approach and possibly cross. Then the now unknown middle target approaches A and possibly crosses. Crossing or not crossing is equally likely in both potential crossing cases. This leaves the three targets’ identities unknown at the end of the scenario, as in Figure 7.

Late data arrive and report that the middle target at the end of the scenario is target B. This information is propagated throughout the scenario, resolving all the ambiguity. There is only one of each target in the scenario, and the targets’ identities are constant; thus, an observation of target B means that neither of the other tracks at that time can be target B. This resolves the B/C ambiguous track to C, which in turn resolves the final track to target A. The result is three fully identified, non-interacting tracks. (This would be three horizontal straight lines, A on top, B in the middle, and C on the bottom, not shown.)

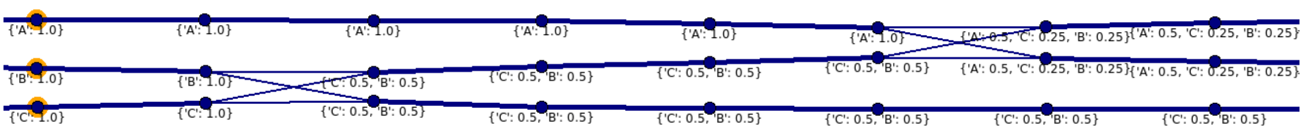

Figure 7. **“S” Scenario.** Three targets travel from left to right. Their identities are known at the beginning, as “A,” “B,” and “C.” These attribute data were observed together with the highlighted observations in the earlier (left) part of the scenario. The middle target possibly crosses with one of the outer targets. Then the unknown middle target possibly crosses with the other outer target. Crossing or not crossing is equally likely in both potential crossing cases. In this manner the interior target forms an “S” shaped curve. The target identities are unknown at the end of the scenario, as seen here. Late data indicating that the middle target is “B” resolve all the ambiguity of the scenario; there are no crossings, the top target is A, the middle is B, and the bottom is C.

5.3 Complex Multiple Sensor, Multiple Target Scenario

Here we show a more complex scenario, simulated within a realistic multitarget multisensor simulator. The scenario contains many sensors (six) observing many targets (about forty). The sensors are of various types, and the targets are of several different motion types. Slow and fast movers are included, as well as targets that are considered friendly and hostile. Figure 8 shows output from the TAP displayed with Google Earth for this scenario.

The challenge here comes from the closely spaced targets. Close targets yield ambiguous data associations that the tracker has sent to the TAP/BNTD. The result of TAP/BNTD processing (in Figure 8) shows the smoothed tracks and the resolution of all the input ambiguous associations.

This scenario shows that the TAP is capable of working with tasks of much greater complexity than simple toy problems. We have gone beyond the more theoretical examples of Sections 5.1 and 5.2, to show the ability of the TAP to analyze real problems.

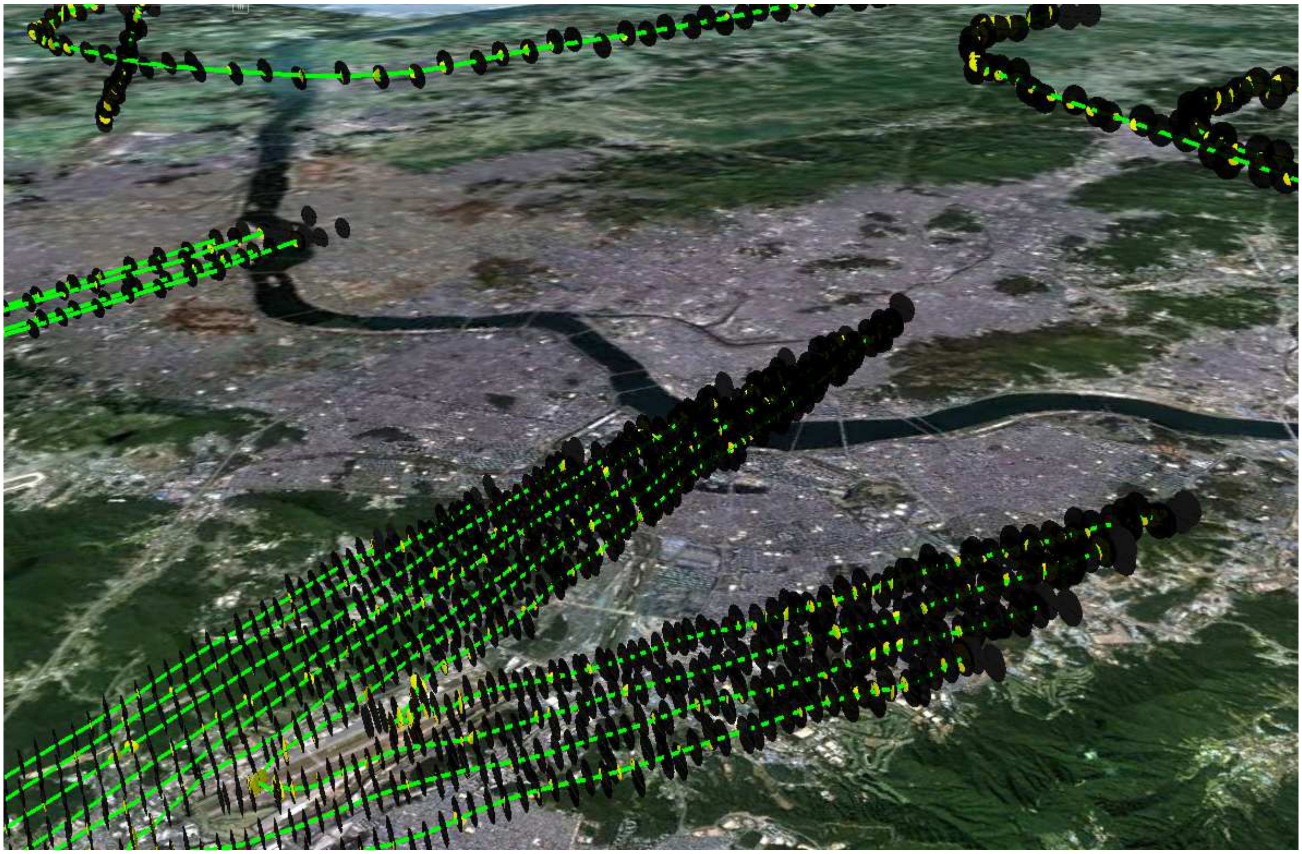

Figure 8. **Complex Multiple Sensor, Multiple Target Scenario.** Here we demonstrate TAP output from a complex, realistic simulated scenario, involving multiple sensor types and multiple target motion types. The dark ovals represent observations. The smaller light ovals are the track states. The connections show the resolution of the input ambiguous track associations.

6. SUMMARY AND CONCLUSIONS

We have shown how the TAP concept can be used to off-load important, possibly critical, time consuming analysis tasks from a tracker, so that the tracker can concentrate on the urgent, realtime tasks it must perform. We have also highlighted the functionality of the BNTD, a system designed for using a variety of evidence to update the total tracking picture. In particular, we have shown the BNTD's capabilities in updating the tracking picture using late data that may contain critical attribute information.

We have shown by example how the application of critical late data can resolve the ambiguity of entire scenarios. Other examples have shown how the propagation of attribute data can reveal the ambiguity remaining in the tracking picture which is also important to know. We conclude that the TAP provides a useful way to incorporate a variety of evidence from late data and other sources. This gives the overall tracking system a chance to reconsider its firm decisions in light of new evidence.

ACKNOWLEDGMENTS

This work was supported by the US Army Space and Missile Defense Command under contract W9113M-08-C-0083. We would also like to gratefully acknowledge the support of many others at Numerica.

REFERENCES

[1] Blackman, S. and Popoli, R., [*Design and Analysis of Modern Tracking Systems*], Artech House, Norwood, MA (1999).

[2] Poore, A. B., "Multidimensional assignment formulation of data association problems arising from multitarget tracking and multisensor data fusion," *Computational Optimization and Applications* **3**, 27–57 (1994).

[3] Poore, A. B. and Robertson III, A. J., "A new class of Lagrangian relaxation based algorithms for a class of multidimensional assignment problems," *Computational Optimization and Applications* **8**(2), 129–150 (1997).

[4] Obermeyer, F. H. and Poore, A. B., "A Bayesian network tracking database," *Proceedings of the SPIE, Conference on Signal and Data Processing of Small Targets* **5428**, 400–418 (August 2004).

[5] Slocumb, B. and Trawick, D., "Bayesian Network Tracking Database: Final Report," Missile Defense Agency SBIR Phase II Final Report Contract N00178-04-C-3007, Numerica Corporation (July 2006).

[6] Pearl, J., [*Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*], Morgan Kaufmann (1988).

[7] Cowell, R. G., Dawid, A. P., Lauritzen, S. L., and Spiegelhalter, D. J., [*Probabilistic Networks and Expert Systems*], Springer-Verlag, New York (1999).

[8] Ghahramani, Z., "An Introduction to Hidden Markov Models and Bayesian Networks," *International Journal of Pattern Recognition and Artificial Intelligence* **15**, 9–42 (2001).

[9] Fraser, D. C. and Potter, J. E., "The optimal linear smoother as a combination of two optimum linear filters," *IEEE Trans. on Automatic Control* **AC-14**, 387–390 (August 1969).

[10] Gelb, A., [*Applied Optimal Estimation*], The M.I.T. Press, Cambridge, Massachusetts, and London, England (1974).

[11] Bar-Shalom, Y., Li, X.-R., and Kirubarajan, T., [*Estimation with Applications to Tracking and Navigation: Theory, Algorithms, and Software*], J. Wiley (2001).

[12] Helmick, R. E., Blair, W. D., and Hoffman, S. A., "Fixed-interval smoothing for Markovian switching systems," *IEEE Trans. on Information Theory* **41**, 1845–1855 (November 1995).

[13] Murty, K., "An algorithm for ranking all the assignments in order of increasing cost," *Operations Research* **16**, 682–687 (1968).

[14] Miller, M. L., Stone, H. S., and Cox, I. J., "Optimizing murty's ranked assignment method," *IEEE Trans. Aerospace and Electronic Systems* **33**, 851–862 (July 1997).

[15] Zhang, K., Li, X.-R., and Zhu, Y., "Optimal update with out-of-sequence-measurements," *IEEE Trans. on Signal Processing* **53**, 1992–2004 (June 2005).

[16] Chan, S. and Paffenroth, R. C., "Out-of-sequence measurement updates for multi-hypothesis tracking algorithms," in [*Proceedings of the SPIE, Conference on Signal and Data Processing of Small Targets*], Drummond, O. E., ed., **6969** (2008).

[17] Drummond, O. E., "On features and attributes in multisensor, multitarget tracking," in [*Proceedings of the 2nd International Conference on Information Fusion*], (1999).

[18] Jordan, M. I., [*Learning in Graphical Models*], MIT Press (1999).

[19] Dempster, A., Laird, N., and Rubin, D., "Maximum-likelihood from incomplete data via the EM algorithm," *Journal of the Royal Statistical Society, B* **39**, 39 (1977).