Tracking Multiple Evolving Threats With Cluttered Surveillance Observations

Zachariah Sutton Peter Willett Yaakov Bar-Shalom

Department of ECE
University of Connecticut
Storrs, Connecticut 06269

***Abstract*—Many threats in the form of human actions (terrorist attacks, military actions, etc.) can be stochastically modeled by someone with relevant expert knowledge. In this work, a threat is taken to be a modeled *sequence* of actions that evolve over time and culminate at some ultimate goal. A model would be a hypothesis as to how a threat would develop, and what kind of observable evidence it would produce along the way. This modeling method allows us to attempt detection using the preliminary evidence of a threat. This would theoretically allow the user to take preemptive action; i.e., the user can intercede before its culmination. This work presents a method of stochastically modeling these types of processes using Hidden Markov Models (HMMs). We then present a detection scheme based on random finite set (RFS) filters (Bernoulli filters) that allows for detection of multiple threat processes using a single cluttered stream of observed data.**

I. Background

We begin with some collection of patterns of human activity that have been identified as suspicious or threatening. We wish to monitor a stream of routine surveillance data and detect if any of the threatening patterns exist. To be detectable, these patterns should have a "storyline" type of structure; they should evolve over time and produce a sequence of observable evidence. Also, the significant part of the threat – the part we wish to prevent or interfere with – should occur toward the end of the process so that we receive preliminary evidence. Threats that develop and culminate instantaneously will not be considered.

Hidden Markov models (HMMs) are a natural choice for modeling these process. A HMM is a Markov process – discrete in this case – whose states are not directly observable, but have some probabilistic relationship to what is observed. In this application the states would be stages in the threat process, which can't be seen but are related to the observations. General tutorials on HMMs can be found at [5], [6]. Some previous work has been done on modeling these particular types of threats with HMMs in [8].

We assume that the observation stream is mostly clutter observations that do not originate from any threat. If a threat exists, the observations originating from it are interspersed in the clutter stream. We further assume that observations from both clutter and threats take the form of transactions (communication, travel, financial transactions, etc.) between entities (people, places, objects). In [9] we showed that detection of a single threat process was improved by giving each entity a weight between 0 and 1 that indicates its probability of being involved in the threat and dynamically updating these weights as we receive observations. Then current observations involving entities that have been linked to suspicious activity in the past are taken more seriously.

Since a modeled threat may or may not be present, a Bernoulli filter lends itself well to the detection problem. The Bernoulli filter uses a random finite set (RFS) framework which maintains a target state estimate that is weighted by the target's probability of existence. The work in [3] and [9] uses Bernoulli filters to detect threat models similar to what we use here.

Supported by NPS via ONR contract N00244-16-1-0017.

II. Contributions

There are two main contributions in this paper. First, we have revised the structure of the HMM transition and emission matrices from previous work. We do this partly to make implementation more streamlined and partly out of necessity since we are now dealing with multiple target processes.

Second, we present a particle filter that borrows the random finite set framework from the Bernoulli filter. Since we are now considering multiple targets, the random finite set representing the joint state can have more than one element. Because of the single observation stream, we must perform data association between the filter elements and the observations.

III. Models

III-A Population Model

We define an array of all entity identities $\boldsymbol{\mathcal{E}} = \{e_1, e_2, ..., e_{N_e}\}$ of size N_e. It is implied that the entities are human actors, but they could just as easily be locations or objects. For each observation, a pair of these entities will be linked by a transaction. This assumes that all entities are uniquely identifiable and will be correctly identified when observed. This requirement could be relaxed in future work if a feature based observation model [8] is used.

With all entities identified, we assign to each an indicator Bernoulli random variable where a value of 1 means the entity is involved in the threat and 0 means it is not involved. Let us denote the distributions (probability of value 1) of these indicator random variables as $K = [k_1, k_2, ..., k_{N_e}]$. These are the entity weights that we wish to update based on observations and use as a clue in the detection. Since we are dealing with multiple targets and it is possible for an entity to be involved in one threat and not another, we must create and update one of these arrays for each filter element.

III-B Observation Model

We define a set $\boldsymbol{\mathcal{Z}} = \{z_1, z_2, ..., z_{N_z}\}$ of all possible transaction types. This should be an exhaustive set of the transaction types from clutter and threat processes. We model an observation O^i with the structure shown in Figure 1, where $z^i \in \boldsymbol{\mathcal{Z}}$ is some transaction linking entities $\{e^i_a, e^i_b\} \subset \boldsymbol{\mathcal{E}}$.

It is helpful to think of our observation model as an attempt to give structure to simple sentences where $\boldsymbol{\mathcal{Z}}$ is the set of all possible verbs and $\boldsymbol{\mathcal{E}}$ is the set of all possible nouns. In this work we assume that necessary preprocessing of data can be done to fit observations roughly into the framework of this model. This is relatively simple in some cases; e.g. Person A places a call to Person B, or Person X gets on a flight to City J. However, fitting data to this model can be harder in some cases; e.g. Person M posts suspicious social media content, or there is a large crowd of people at Location D.

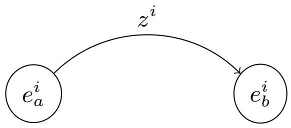

Fig. 1. Structure of an observation

III-C Threat Process Model

III-C1 Transition Model: For this work we have modified the model of the state transition structure of the HMMs to be as shown in Figure 2. Let us denote the set of all light colored states as $\boldsymbol{\mathcal{C}}$, and the set of all shaded states as $\boldsymbol{\mathcal{T}}$.

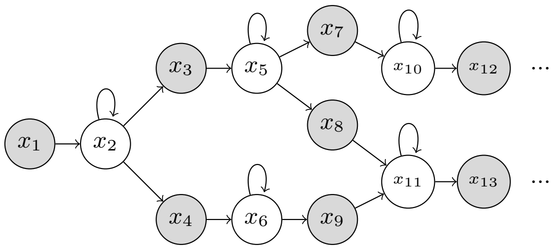

Fig. 2. Example Markov chain

All states in $\boldsymbol{\mathcal{C}}$ are considered "wait" states and are given a very high probability of self-transitioning. These states model the long periods where the threat is not emitting any observations. The states in $\boldsymbol{\mathcal{T}}$ are considered the target states and do not self transition so are visited only once, if at all. These states provide the sparse "true" observations that are produced by the threat. This structure allows us to treat the threat model as a typical HMM, eliminating the need to code a transition based model as in [8], [3].

The state transition matrix A defining the structure in Fig. 2 would be (partially) given by

$$A = \begin{bmatrix} 0 & 1 & 0 & \dots & & & & & & & \\ 0 & p_{\text{st}} & p' & p' & \dots & & & & & & \\ \vdots & 0 & 0 & 0 & 1 & 0 & \dots & & & & \\ & \vdots & 0 & 0 & 0 & 1 & 0 & \dots & & & \\ & & \vdots & 0 & p_{\text{st}} & 0 & p' & p' & 0 & \dots & \\ & & & \vdots & 0 & p_{\text{st}} & 0 & 0 & p' & 0 & \dots \\ & & & & \vdots & 0 & 0 & 0 & 0 & 1 & 0 \\ & & & & & & \vdots & 0 & 0 & 0 & 1 \\ & & & & & & & & \vdots & 0 & 1 \end{bmatrix}$$

where each row is a transition probability mass function for the corresponding state. In other words, the element $A^{i,j}$ is the probability of transitioning from state i to state j. The wait states self transition with probability p_{st} and the probabilities of the other transitions p' are not of particular importance and are mostly dictated by the constraint that all rows of A sum to 1. The self transition probabilities are not generally required to be the same for all wait states. The expected wait time between any two target states is given by the mean sojourn time out of the wait state between them [1]. With all self transition probabilities equal to p_{st}, the expected time between true observations is

$$\bar{t}_{\text{st}} = \frac{1}{1 - p_{\text{st}}} \tag{1}$$

III-C2 Emission Model: At any point in time, an active threat process will emit either a true observation or nothing. The states in $\boldsymbol{\mathcal{C}}$ always emit nothing. The possible sequences of actions corresponding to the threat are modeled with the states in $\boldsymbol{\mathcal{T}}$. Each state in $\boldsymbol{\mathcal{T}}$ has a unique emission distribution over $\boldsymbol{\mathcal{Z}}$, stored in the corresponding row of the emission matrix. Each of these distributions should be relatively specific, giving weight to only a small number of transaction types. We must also allow for missed detections of true observations. To model this effect, we assign each target state a probability of detection p_{d}. These states then output nothing with probability $1 - p_{\text{d}}$.

For implementation purposes, we append an extra element on to the right hand side of $\boldsymbol{\mathcal{Z}}$ that corresponds to "no observation". The emission matrix B models the relationship between the states and this augmented set of transactions. For the simplest case of one possible transaction type for each target state, the model shown in Figure 2 could have an emission matrix of the form

$$B = \begin{bmatrix} 0 & p_{\text{d}} & 0 & 0 & \dots & 0 & 0 & 1-p_{\text{d}} \\ 0 & 0 & 0 & 0 & \dots & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & \dots & 0 & p_{\text{d}} & 1-p_{\text{d}} \\ p_{\text{d}} & 0 & 0 & 0 & \dots & 0 & 0 & 1-p_{\text{d}} \\ 0 & 0 & 0 & 0 & \dots & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & \dots & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & \dots & p_{\text{d}} & 0 & 1-p_{\text{d}} \\ 0 & 0 & 0 & p_{\text{d}} & \dots & 0 & 0 & 1-p_{\text{d}} \\ 0 & 0 & p_{\text{d}} & 0 & \dots & 0 & 0 & 1-p_{\text{d}} \\ 0 & 0 & 0 & 0 & \dots & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & \dots & 0 & 0 & 1 \\ \vdots & \vdots & \vdots & \vdots & & \vdots & \vdots & \vdots \end{bmatrix}$$

where each row is the emission distribution over the augmented $\boldsymbol{\mathcal{Z}}$ for the corresponding state and must sum to 1.

The transactions emitted by target states always link two entities drawn independently but exclusively from some unknown population subset $\boldsymbol{I} \subset \boldsymbol{\mathcal{E}}$. Each entity in $\boldsymbol{I}$ is defined as being involved in the threat we are modeling. The algorithm requires an estimate of how many entities will be involved in a given threat N_{inv}, but can assume no prior information on the identities of the involved entities. It is assumed that N_{inv} is such that at least some of the involved entities appear more than once in true observations, since the real power of the algorithm comes from involved entities that are observed doing *multiple* suspicious activities.

III-D Clutter Process Model

A clutter process Λ_{cl} is modeled as a single state HMM that emits transaction types based on a clutter emission distribution B_{cl}. This distribution should give some weight to every transaction type in $\boldsymbol{\mathcal{Z}}$. In each time step (the discrete transition period of the threat process Markov chains), the clutter model is made to emit multiple transactions where the types of the transactions are drawn independently according to B_{cl}. The number of clutter observations at each step is assumed Poisson distributed with parameter λ_{cl}. The clutter process also samples two entities uniformly with replacement from the entire population $\boldsymbol{\mathcal{E}}$ to be linked by each of these transactions.

The overall set of observations at some time t is then the union of the current clutter observations and any true observations that may be present. This current set of observations will be denoted $\mathbf{O}_t$ and the set of all subsets of observations up to and including time t is denoted $\mathbf{O}^t$. Note that due to the sparseness of true observations, most observation sets will contain only clutter.

IV. Filter

From here on, we assume that we have a set of N_m threat models

$$\boldsymbol{\Lambda} = \{\Lambda^i\}_{i=1}^{N_\mathrm{m}} = \{(A^i, B^i, N_\mathrm{inv}^i)\}_{i=1}^{N_\mathrm{m}} \tag{2}$$

along with the clutter model $(B_\mathrm{cl}, \lambda_\mathrm{cl})$.

We now present a method of detecting and tracking the modeled threats with a particle based RFS filter while simultaneously updating the entity involvement probabilities in each K.

The filter treats the joint target state as a random finite set $\mathbf{S}$. Since for now we are not considering multiple simultaneous instances of the same target model, the RFS has N_m elements, each corresponding to a particular target model, so $\mathbf{S} = \{\mathrm{S}^i\}_{i=1}^{N_\mathrm{m}}$. Each element is either empty $\{\emptyset\}$ if the corresponding target doesn't exist, or it takes on the value of the current state of the corresponding HMM. More formally, we can think of the RFS elements as taking on values in the augmented state space $\mathrm{S}^i \in \{X^i\} \sqcup \{\emptyset\}$, where $\{X^i\}$ is the state space of the ith model. The filter outputs an estimated pmf $f^i(\mathrm{S})$ over this augmented state space for each filter element. The output $f^i(\mathrm{S})$ can also be thought of as a pmf over the target states $\{X^i\}$ that is weighted with a probability of target existence q, where $(1-q)$ is the probability that the corresponding RFS element is empty.

IV-A Dynamics

The dynamics of each RFS element are modeled as a Markov process that governs transitions in the augmented state space $\{X^i\} \sqcup \{\emptyset\}$. This process is completely parameterized by a probability of target birth p_b, a probability of target survival p_s, the birth distribution $f_\mathrm{b}(x)$, and the corresponding target state transition matrix A. The transition matrix for the ith element S^i is

$$\boldsymbol{\Pi}_i = \left[\begin{array}{c|c} p_\mathrm{s} \cdot A^i & \vec{1} \cdot (1-p_\mathrm{s}) \\ \hline p_\mathrm{b} \cdot f_\mathrm{b}(x) & (1-p_\mathrm{b}) \end{array}\right] \tag{3}$$

where $\vec{1}$ is a column vector of ones that matches the vertical dimension of A^i. Note that the added row on the bottom and column on the right correspond to the appended "does not exist" state $\{\emptyset\}$.

We will now describe one cycle of the filter in four main stages: particle resample, data association, update, and prediction.

IV-B Resample Step

Say at time t we are given the predicted pmfs for all filter elements based on all previous observations and data associations $\{f_{t|t-1}^i(\mathrm{S})\}_{i=1}^{N_\mathrm{m}}$. We first sample a set of N_p *unique* particles

$$\boldsymbol{\Phi} = \{\phi^j\}_{j=1}^{N_\mathrm{p}} \tag{4}$$

Each particle is a unique proposal of the current joint state of the filter elements. That is, any one particle is itself a set given by

$$\phi^j = \left\{\tilde{\mathrm{S}}^{i,j}\right\}_{i=1}^{N_\mathrm{m}} \tag{5}$$

where $\tilde{\mathrm{S}}^{i,j}$ is an element of $\{X^i\} \sqcup \{\emptyset\}$ sampled for the jth particle according to $f_{t|t-1}^i(\mathrm{S}^i)$. Since our models have discrete states, they lend themselves nicely to this sampling.

We then assign each particle a prior weight based on the predicted pmfs we sampled from.

$$\mathcal{W}_{t|t-1}^j = \prod_{i=1}^{N_\mathrm{m}} f_{t|t-1}^i(\tilde{\mathrm{S}}^{i,j}), \qquad j = 1, 2, ..., N_\mathrm{p} \tag{6}$$

IV-C Data Association Step

Now we look at the current observation set

$$\mathbf{O}_t = \{\mathrm{O}^n\}_{n=1}^W = \{(e_a^n, e_b^n, z^n)\}_{n=1}^W \tag{7}$$

Note that the size of the set W can change from one time to the next since the number of clutter observations is a Poisson random variable.

For each particle, we find the single most likely data association σ^j between the particle states and the set of observations. We use a modified form of the auction algorithm [2] which assumes a clutter process that can be assigned multiple observations. The likelihoods used in the algorithm are

$$\varphi(\mathrm{O}^n|\tilde{\mathrm{S}}^{i,j}) = \begin{cases} p(z^n|\Lambda_\mathrm{cl})(\frac{1}{N_e})^2, & \text{if} \quad \tilde{\mathrm{S}}^{i,j} \in \{\boldsymbol{\mathcal{C}}^i, \emptyset\} \\ p(z^n|\tilde{\mathrm{S}}^{i,j})(k_a^{i,n})(k_b^{i,n})(\frac{1}{N_\mathrm{inv}^i})^2, & \text{if} \quad \tilde{\mathrm{S}}^{i,j} \in \{\boldsymbol{\mathcal{T}}^i\} \end{cases} \tag{8}$$

where $p(z^n|\Lambda_\mathrm{cl})$ is available from B_cl, $p(z^n|\tilde{\mathrm{S}}^{i,j})$ is available from B^i, and $k_a^{i,n}$ is a (clunky) notation for the current probability that the first entity in the nth observation is involved in threat process i. This value along with $k_b^{i,n}$ is available from the current predicted distributions in K^i which are discussed in more detail in Section V. To be more precise, the auction algorithm takes as input the logarithm of these likelihoods and returns the valid assignment such that the sum of the log likelihoods is maximized. It also returns the value of the maximized sum which is the total log likelihood and can easily be converted to the likelihood of the the particle and the association based on the observation. We will denote this value $\varphi(\mathbf{O}_t|\phi^j, \sigma^j)$.

We now have a data association for each particle $\{\sigma^j\}_{j=1}^{N_\mathrm{p}}$ where each association can be expressed as a vector of length W where the nth element indicates the process to which the nth observation has been assigned. That is, $\sigma^{j,n} = i$ means that for the jth particle, the nth observation has been assigned to the ith process. If $\sigma^{j,n} = 0$, the nth observation has been assigned to the clutter process.

Using the associations, we can calculate the overall likelihood of each particle/association pair taking into account the Poisson clutter model and any missed detections.

$$\mathcal{L}(\mathbf{O}_t|\phi^j, \sigma^j) = \mathrm{e}^{-\lambda_\mathrm{cl}}\, \varphi(\mathbf{O}_t|\phi^j, \sigma^j) \prod_{n:\sigma^{j,n}=0} \lambda_\mathrm{cl} \prod_{\substack{i:\tilde{\mathrm{S}}^{i,j} \in \boldsymbol{\mathcal{T}}^i, \\ \nexists n:\sigma^{j,n}=i}} (1-p_\mathrm{d}) \tag{9}$$

We will refer to this value as $\mathcal{L}^j$ for short hand notation.

IV-D Update Step

Now we can update the weight of each particle with the prior weights from (6) and the likelihoods from (9) with

$$\mathcal{W}_{t|t}^j = \frac{\mathcal{L}^j\ \mathcal{W}_{t|t-1}^j}{\sum_{j=1}^{N_\mathrm{p}} \mathcal{L}^j\ \mathcal{W}_{t|t-1}^j} \tag{10}$$

The updated pmf for each filter element is then approximated by

$$f_{t|t}^i(\mathrm{S}) = \sum_{j=1}^{N_\mathrm{p}} \mathcal{W}_{t|t}^j \mathbb{1}(\mathrm{S} = \tilde{\mathrm{S}}^{i,j}) \tag{11}$$

where $\mathbb{1}(\cdot)$ is the indicator function. In words, the updated pmf for process i evaluated at state S is the sum of the updated weights of all the particles that propose that process i is in state S. It is possible that a state will have an updated probability mass of 0 – if there are no particles that propose that state. But if we sample sufficiently many particles, this will only happen for states whose prior masses were negligible.

IV-E Prediction Step

The predicted pmf of process i for time $t+1$ is based solely on the modeled dynamics of the RFS element i given in (3). In a programmed implementation where the updated pmf is a column vector, the predicted pmf is also a column vector given by multiplication with the transpose of the transition matrix in (3).

$$f^i_{t+1|t}(\mathrm{S}) = \mathbf{\Pi}_i^{\mathrm{T}} f^i_{t|t}(\mathrm{S}) \tag{12}$$

The predicted distributions are then used for particle sampling at the next time step.

V. Involvement Probabilities

The vector K^i contains the probability of involvement in the ith process (weight) for each entity. We model the underlying entity process as an array of independent indicators over the entire population where the expected number of "on" indicators at any time is N^i_{inv}, and the expected amount of time an indicator stays on is the expected target lifetime.

If we take

$$C^i = \frac{N^i_{inv}}{N_e} \tag{13}$$

to be the fraction of population expected to be involved, and D^i to be the expected target lifetime, the desired behavior is modeled by a two state Markov chain with transition matrix

$$\boldsymbol{H}^i = \begin{bmatrix} 1-\frac{1}{D^i} & \frac{C^i}{D^i(1-C^i)} \\ \frac{1}{D^i} & 1-\frac{C^i}{D^i(1-C^i)} \end{bmatrix} \tag{14}$$

where the first state is the involved or "on" state and the second is the uninvolved or "off" state. In simulations, this process does not govern the true entity involvements; instead an involved set is chosen at the time of target birth and remains the same throughout. But this model is used for prediction of the entity weights from one time step to the next.

We have devised a method to update the entity weights in each K^i based on the current updated estimate given by the particle filter. Let $\{e^m\}_{m=1}^{2W}$ be the set of currently observed entities. Let $(e^m \rightarrow_i \phi^j)$ denote the binary event "*the observation containing entity m has been associated with process i in particle j*" Once we have the updated particle weights from (10), the weights of each of the $2W$ currently observed entities are updated for each of the processes using the formula

$$K^i_{t|t}(e^m) = \frac{K^i_{t|t-1}(e^m)\left(\frac{1}{N^i_{inv}}\sum_{j=1}^{N_\mathrm{p}} \mathcal{W}^j_{t|t}\mathbb{1}(e^m \rightarrow_i \phi^j) + \frac{1}{N_e}\sum_{j=1}^{N_\mathrm{p}} \mathcal{W}^j_{t|t}\overline{\mathbb{1}(e^m \rightarrow_i \phi^j)}\right)}{\left(K^i_{t|t-1}(e^m)\frac{1}{N^i_{inv}}\sum_{j=1}^{N_\mathrm{p}} \mathcal{W}^j_{t|t}\mathbb{1}(e^m \rightarrow_i \phi^j)\right) + \left(\frac{1}{N_e}\sum_{j=1}^{N_\mathrm{p}} \mathcal{W}^j_{t|t}\overline{\mathbb{1}(e^m \rightarrow_i \phi^j)}\right)}, \qquad i = 1, 2, ..., N_\mathrm{m}, \qquad m = 1, 2, ..., 2W \tag{15}$$

where $\overline{\mathbb{1}(\cdot)}$ is the compliment of the indicator function.

All entity weights, including the current ones updated by (15), are predicted to the next time step using the dynamics modeled in 14. Since we are only interested in the weights and not their complements, this update amounts to

$$K^i_{t+1|t}(e^\ell) = \left(1 - \frac{1}{D^i}\right) K^i_{t|t}(e^\ell) + \frac{C^i}{D^i(1-C^i)}(1 - K^i_{t|t}(e^\ell)) \tag{16}$$

for

$$i = 1, 2..., N_\mathrm{m}, \qquad \ell = 1, 2, ..., N_\mathrm{e}$$

VI. Results

Performance is studied for a set of 3 synthetic threat models. A continuous simulation was run where the threat processes were randomly activated, allowed to complete, and forced to be inactive for a random "idle period". The simulation was allowed to run until each process had been activated at least 200 times. The statistic we use to declare detection is simply the probability of existence that the filter has for the process, or $1 - p(\emptyset)$. For a particular threshold between 0 and 1, a successful detection is declared if the probability of existence exceeds the threshold at any time while the threat is active. A false alarm is declared if the threshold is exceeded during an idle period. Plots of these results for a range of threshold values are given in Figure 3.

The significant parameter settings for this simulation are: $\bar{t}_{\mathrm{st}} = 100$ (expected time between true observations), $N_\mathrm{e} = 2000$ (population size), $\lambda_{\mathrm{cl}} = 5$ (clutter Poisson parameter), and $N_z = 75$ (the total number of transaction types). Process 1 has 6 possible paths from start to finish, emits at most 11 observations, and involves 4 entities. Process 2 has 4 possible paths from start to finish, emits at most 14 observations, and involves 6 entities. Process 3 has 2 possible paths from start to finish, emits at most 16 observations, and involves 8 entities. The results suggest that the number of involved entities is the most significant factor in detectability of these particular models.

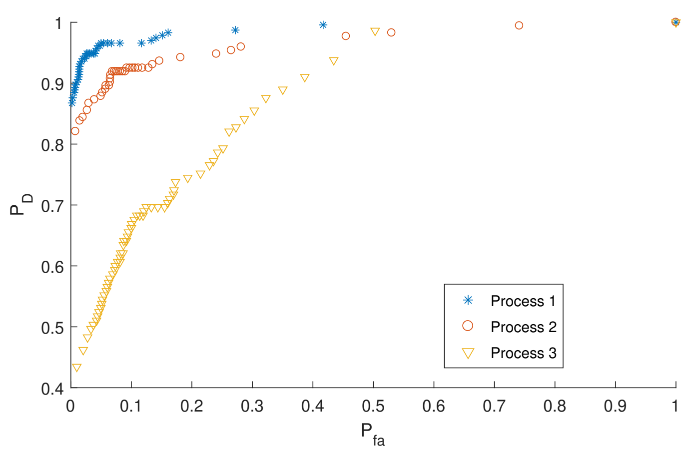

Fig. 3. Probability of detection versus probability of false alarm for the three threat processes. Each data point corresponds to a particular detection threshold.

REFERENCES

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

[2] D. Bertsekas, "The auction algorithm: A distributed relaxation method for the assignment problem, *Annals of Operations Research*, vol. 14, no. 1, pp. 105123, 1988.

[3] K. Granstrm, P. Willett, and Y. Bar-Shalom, "Asymmetric Threat Modeling Using HMMs: Bernoulli Filtering and Detectability Analysis", *IEEE Transactions on Signal Processing*, vol. 64, no. 10, pp. 2587-2601, May 2016.

[4] Y. Liu, S. D. Blostein, "Quickest detection of an abrupt change in a random sequence with finite change-time", *IEEE Transactions on Information Theory*, pp. 1985-1993, November 1994.

[5] L. R. Rabiner and B. H. Juang, "An introduction to hidden Markov models", *IEEE ASSP Mag.*, vol. 3, no. 1, pp. 416, Jan. 1986.

[6] L. R. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition," *Proc. IEEE*, vol. 77, no. 2, pp. 257-286, Feb. 1989.

[7] B. Ristic, B.-T. Vo, B.-N. Vo, and A. Farina, "A Tutorial on Bernoulli Filters: Theory, Implementation and Applications", *IEEE Transactions on Signal Processing*, vol. 61, no. 13, pp. 34063430, Jul. 2013.

[8] S. Singh, H. Tu, W. Donat, K. Pattipati, and P. Willett, "Anomaly detection via feature-aided tracking and hidden Markov models", *Transactions on Systems, Man, and CyberneticsPart A: Systems and Humans*, vol. 39, no. 1, pp. 144159, Jan. 2009.

[9] Z. Sutton, P. Willett, and Y. Bar-Shalom, "Modeling and Detection of Evolving Threats Using Random Finite Set Statistics", to appear in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, Apr. 2018.