Factored Particle Filtering for Data Fusion and Situation Assessments in Urban Environments

Subrata Das and David Lawless
Charles River Analytics, Inc.
625 Mount Auburn Street
Cambridge, MA 02138, USA
sdas@cra.com

Brenda Ng and Avi Pfeffer
Division of Engineering and Applied Science
Harvard University
Cambridge, MA 02138, USA
avi@eecs.harvard.edu

Abstract - *We present and demonstrate a particle filtering approach to data fusion and situation assessment for military operations in urban environments. Our approach views such an environment as a physical system whose state vector is composed of a large number of both discrete and continuous variables representing properties of tracked entities. Inferencing on such vector-based models exploits both causal dependencies among variables in the state vector via its dynamic Bayesian belief network representation and vector decomposition into weakly interacting subcomponents. To effectively leverage the decomposition, instead of straightforward particle filtering based inferencing, the proposed algorithm maintains factored particles over clusters of state variables, thus resulting in smaller variance. The algorithm samples discrete modes and approximates the continuous variables by a multi-normal distribution updated at each time step by an unscented Kalman filter. The approach is demonstrated using a Marine Corps operational scenario involving a potential ambush on city streets.*

Keywords: data fusion, situation assessment, particle filtering, dynamic belief networks, urban warfare.

1 Introduction

Decision support for Marine Corps operations in an urban environment has always been an area of concern for planners, as the environment can be populated with a mixture of unknown numbers and types of adversarial and neutral entities. Adversarial entities in the environment need to be identified and tracked as part of the fusion process, while minimizing false alarms, to recognize higher-level situations (e.g. ambush, interdiction) required in the effective application of force and to achieve mission objectives. Our approach to building an inference system for decision support, via Data Fusion (DF) and Situation Assessment (SA), views an urban environment as a physical system whose behavior is defined by varying numbers and types of entities.

Traditionally, for the purpose of modeling and determining the states of a physical system, the system is represented as a Hidden Markov Model (HMM) or a State-Space Model (SSM). The hidden state in an HMM is represented by a single discrete random variable that takes on a fixed number of possible values. The hidden state in an SSM is represented by a single vector-valued random variable. The SSM approach is usually preferred due to its finer granularity in representing states via vector attribute coefficients. Given the SSM and all its unknown parameters, one can use standard filtering techniques to estimate the unobservable coefficients in the state vector. However, there are serious drawbacks in this approach:

- The flat vector representation in a conventional SSM approach neither fully exploits the conditional/marginal independences that may exist among random variables representing entities within the environment, nor effectively portrays any dependencies that might exist among variables.
- Vector attribute coefficients are usually drawn from continuous valued domains, and thus fail to integrate reasoning in "hybrid" numeric and symbolic domains involving both continuous and discrete variables.
- State transition via tracking becomes virtually impossible in the presence of a large number of adversarial units, due to the sheer size of the mode space incorporating the behavior of each unit.

Our approach overcomes the first of the above limitations by transforming SSM vectors to temporal or Dynamic Belief Networks (DBNs) (Nicholson and Brady, 1993; Kanazawa et al., 1995; Murphy 2002) in which the hidden state is represented by a set of random variables, each of which is either discrete or continuous. Belief Network technology offers several advantages, including its graphical approach to situation modeling via causal relationships among battlefield concepts and its handling of uncertainty consistent with established probability semantics. Inferencing on such networks exploits causal dependencies among variables and allows smoothing, filtering, and prediction, in the sense of a traditional physical system, to provide SA via the states of the hidden variables.

An efficient exact inferencing algorithm on hybrid DBNs containing both discrete and continuous variables is not available to date. Lener et al. (2001) developed an inference algorithm for static hybrid belief networks, which are Conditional Linear Gaussian models, where the conditional distribution of the continuous variables given an assignment to the discrete variables is a multivariate Gaussian. Cob and Shenoy (2004) developed an inference algorithm in hybrid belief networks using Mixtures of Truncated Potentials. Our approach is a sampling-based approximate inference algorithm called particle filtering (PF) (Djuric et al., 2003; Arulampalam et al., 2002; Doucet et al., 2001; Israd and Blake, 1998; Gordon et al., 1993) to deal with hybrid DBNs containing both discrete and continuous variables. A particle filter algorithm approximates conditional densities for state transitions by a finite set of weighted sums of samples, referred to as particles. For applications of particle filtering in single and multi-target tracking for data fusion, see (Schulz et al., 2003; Karlsson, et al., 2003; Hue et al., 2002; Magee and Boyle, 2002; Avitzour, 1995). Specifically, we make use of an enhancement of PF, known as the Rao-Blackwellised Particle Filter (RBPF) (Doucet et al., 2000), which exploits the structure of a DBN to decrease the dimension of the sampling distribution, thus requiring fewer particles to achieve the same degree of accuracy offered by standard PF technology.

To efficiently deal with high-dimensional hybrid state vectors representing an urban environment as a whole, our approach exploits the fact that multiple tracked adversarial units in the environment work autonomously with intermittent communications amongst themselves. Therefore, instead of applying straightforward particle filtering based inferencing, the proposed algorithm, called Hybrid-FP (Ng, Pfeffer, et. al., 2004), maintains factored particles over clusters of state variables. Because the clusters have far fewer variables than the entire state space, the variance resulting from maintaining cluster distributions is much smaller than that from maintaining the belief state as a whole. Based on the DBN structure, each factored particle, representing the state of a tracked unit, samples discrete modes and approximates the continuous variables by a multi-normal distribution updated at each time step by the unscented Kalman filter (Wan and van der Merwe, 2000). Such look-ahead prediction in the factored state space improves tracking accuracy and avoids combinatorial explosions in mode enumeration.

To summarize, our approach to building a hybrid inference system consists of building two modules, Data Fusion (DF) and Situation Assessment (SA), based on a particle filtering based inferencing scheme. The DF module is concerned with tracking individual units in an urban environment represented as SSMs. It makes use of the Hybrid-FP algorithms to track multiple adversarial units. The SA module infers high-level situations based on the tracked individual units identified by the DF module. The SA module makes use of hybrid DBNs for representing temporal and causal relationships among various unit types. Particle filtering is also used for approximate inferencing in such hybrid DBNs. We have demonstrated the proposed inference system for multi-source intelligence analysis with an application to a Marine Corps operation in an urban environment.

2 Operational Overview

This section illustrates the function of the conceptual modules and their interactions in the context of a simple scenario (more complex large-scale scenarios will be presented in the next section). The scenario we present here is a Marine Corps operational scenario in an urban terrain involving a potential ambush at a city square in a region of conflict. Given adequate sensor data from a simulation platform, we will use the proposed hybrid inference system to perform unit tracking and a SA to determine whether an ambush of a Blue force performing a routine street patrol is imminent. Figure 1 below provides a computer visualization of our Urban Ambush Scenario. The scenario unfolds essentially as follows:

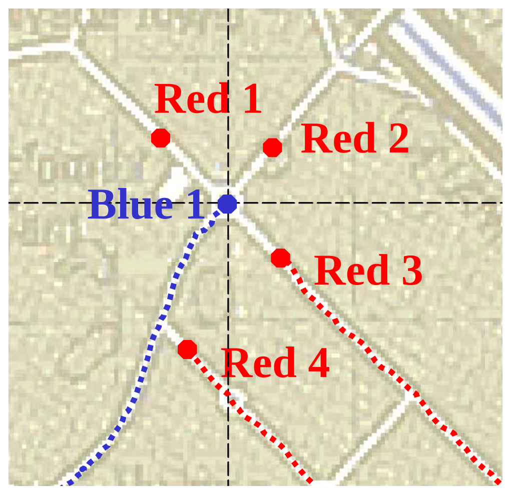

Figure 1: Urban Ambush Scenario Visualization

- Blue headquarters receives reports of Red forces of squad size at locations Red 1 and Red 2, on streets leading into a large, busy city square.
- Patrol force Blue 1 proceeds to the square from the southwest to investigate.
- Subsequently, reports of force Red 3 moving toward the square from the southeast are received; Red 3 blocks the alternate escape route for Blue 1.
- Soon after, a report of force Red 4 is received, which may be moving from the southeast to block the primary escape route for Blue 1.

As shown in the figure, Red forces are likely moving to surround the Blue force and cut off all escape routes. The ultimate practical utility of the situation assessment in this particular scenario is that it can automate – hence expedite - the fusion and processing of incoming field and intelligence reports, and provide warning to Blue 1 and standby backup units in a more timely fashion than was previously possible.

The DF module tracks the environment, which will ultimately comprise four enemy units of squad size. Other than possible intermittent communications among these units, each of these four units can be tracked individually; but concurrently the situation requires high-level assessment of a potential ambush situation based on the status of all four units. The DF module will employ the factored sampling technique in which each particle representing one possible state of the whole environment will be factored into four sub-particles, each representing the status of a squad. The SA module will assess the high-level ambush situation based on the presence of one or more squad units.

3 Enabling Technologies

Dynamic Bayesian Networks (DBNs) (Murphy, 2000; Ghahramani, 2001) are simply Bayesian networks (Pearl 1988) for modeling time series data. In DBNs, directed arcs flow forward in time, representing the assumption that an event can cause another event in the future, but non vice-versa. There are two key ideas in extending a BN to a DBN:

- All nodes of the BN are associated with particular time steps, simply by indexing the nodes with a time step value.
- Some BN nodes for a given time step may have causal dependencies on nodes from earlier time steps; such dependencies are called temporal dependencies.

In a DBN, the hidden state X_t as well as the observation Y_t is represented by a set of random variables, each of which can be discrete or continuous. Suppose $Y_{0:t}$ denotes the sequence $Y_1,...,Y_t$ of random variables and $y_{0:t}$ denotes one realization of $Y_{0:t}$.

- *Prediction* is to find the posterior distribution of the hidden state X_t given the past observations $y_{0:t-1}$.
- *Filtering* is to find the posterior distribution of the hidden state X_t given the current and past observations $y_{0:t}$.
- *Smoothing* is to find the posterior distribution of $X_t(1 \le t \le n)$ given the observed time series $y_{0:n}$

Various exact inferencing techniques exist (Boyen and Koller, 1998 ; Das et al., 2004) for dealing with DBNs with discrete variables, efficient algorithms for hybrid DBNs that we use here for SA modeling purpose do not currently exist. Particle filtering is an approximate but efficient inferencing technique for large DBNs.

The particle filter is a simple and effective algorithm for estimating the state of a dynamic system over time where the state cannot be measured directly, but may be inferred from a set of observations at each time. The algorithm is also known as sequential Monte Carlo introduced in (Handschin and Mayne, 1969), Condensation (CONditional DENSity propagATION) (Israd and Blake, 1998), Sampling Importance Resampling (SIR) (Doucet, 2001), the bootstrap filter (Gordon et al., 1993), the survival of the fittest (Kanazawa et al., 1995), etc. The particle filter is a non-parametric approach, and thus handles non-linearities and multi-modal distributions by approximating them via a finite weighted sum of *N* samples, named as particles. With sufficiently many particles, an approximate conditional distribution can be obtained that is arbitrarily closed to the true conditional distribution. For an overview of the state of the art in applications of particle filter, see (Doucet, 2001; Djuric, et al., 2003).

The approach here is to exploit causal structure amongst domain variables in order to parameterize the belief state as a product of local belief states. This method (Boyen and Koller, 1998) exploits the idea of weak interaction between different model components to artificially impose independencies between weakly interacting subsystems. Unfortunately, the method, which performs exact belief update via junction tree propagation, breaks down under load, as the dynamic propagation becomes too expensive. Moreover, the method cannot handle networks containing discrete variables with continuous parents. Therefore, we adopt here a particle filtering (PF) approach. But standard PF is an expensive solution if the size of state space is too high, requiring a large number of particles to achieve decent accuracy. To deal with this problem, we introduce the idea of *factored particles* (Ng et al., 2002). Instead of maintaining particles over the entire state of the system, we maintain particles over clusters of state variables. Because the clusters have far fewer variables than the entire state space, the variance resulting from maintaining cluster distributions is much smaller than that from maintaining the belief state as a whole. Here we extend the idea of factored sampling to efficiently track multiple adversarial units in urban environments.

Formally, the proposed approximation scheme, called hybrid factored sampling (Hybrid-FP), assumes an environment is factored into a set of clusters $C = \{c_1,...,c_K\}$. For each cluster c_i, it maintains a set of hybrid factored particles $\{s_t^{(c,i)}\}_{i=1}^{N_c}$, where N_c is the number of particles in the cluster. Each hybrid factored particle $s_t^{(c,i)}$ consists of an instantiation of the factored mode vector $Z_t^{(c,i)}$ and the Gaussian mean and covariance matrix $\mu_t^{(c,i)}$ and $\Sigma_t^{(c,i)}$ that characterize the distribution of the continuous variables $X_t^{(c,i)}$ associated with the instantiated mode.

Hybrid-FP adopts the Rao-Blackwellised Particle Filter (RBPF) framework to exploit the structure of a Dynamic Bayesian Network (DBN) to decrease the size of the state space, thus requiring fewer particles. The framework presented in look-ahead RBPF to factor the filtering distribution into

$$p(Z_{0:t}, X_{0:t} \mid Y_{1:t}) = p(X_{0:t} \mid Z_{0:t}, Y_{1:t})\, p(Z_{0:t} \mid Y_{1:t})$$

Since the Gaussian distribution $p(X_{0:t} | Z_{0:t}, Y_{1:t})$ can be computed analytically given $p(Z_{0:t} | Y_{1:t})$, we can reduce our sampling space to the lower dimensional $p(Z_{0:t} | Y_{1:t})$. Instead of sampling entire mode trajectories $Z_{0:t}$, it can be improved based on

$$p(Z_{0:t} | Y_{1:t}) = p(Z_t | Z_{0:t-1}, Y_{1:t}) p(Z_{0:t-1} | Y_{1:t})$$

If the dependence on Y_t in the second term above is taken away and we use the probability distribution

$$p(Z_{0:t} | Y_{1:t}) = p(Z_t | Z_{0:t-1}, Y_{1:t}) p(Z_{0:t-1} | Y_{1:t-1})$$

then Z_t can be recursively sampled from the optimal distribution $p(Z_t | Z_{0:t-1}, Y_{1:t})$. The importance weight in this is as follows:

$$w_t = \sum_{z_t} p(Y_t | Z_{0:t}, Y_{1:t-1}) p(Z_t | Z_{0:t-1}, Y_{1:t-1})$$

Figure 2 shows a schematic of the Hybrid-FP algorithm. The algorithm has four main phases: look-ahead prediction, join, global dynamics propagation, and projection. A resampling step is placed before the global dynamics propagation step. We briefly describe each of these four steps in the following.

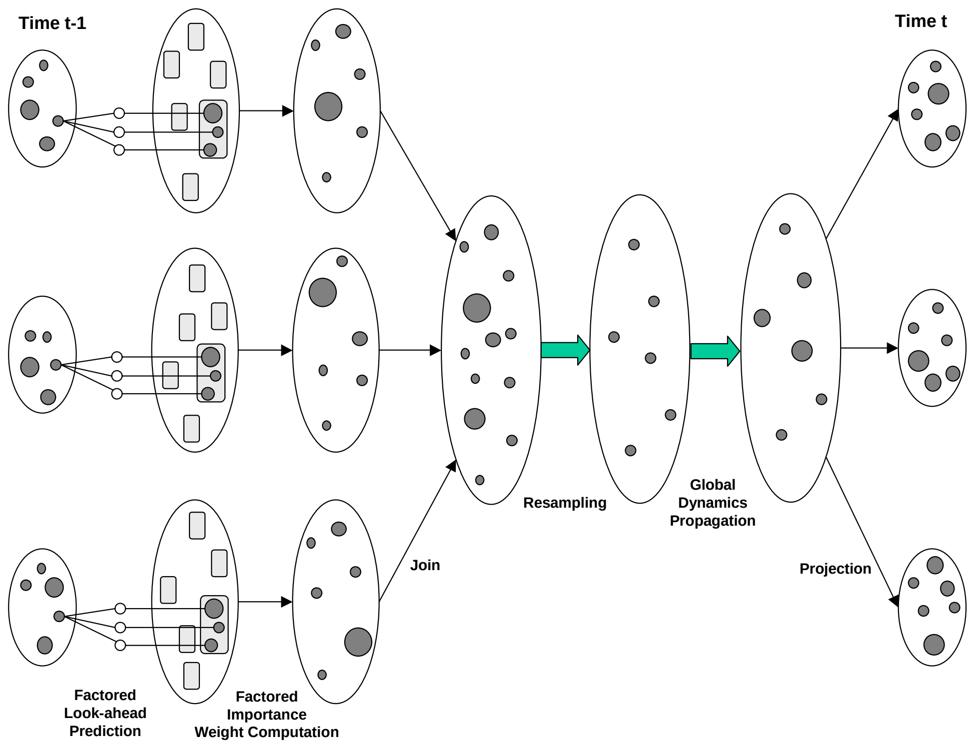

Figure 2: Schematic of the Hybrid-FP algorithm

In the look-ahead prediction step, for each cluster c_i, we begin with a set of factored particles $\{s_{t-1}^{(c,i)}\}_{i=1}^{N}$ for that cluster. The goal is to propagate each factored particle $s_{t-1}^{(c,i)} = (z_{t-1}^{(c,i)}, \mu_{t-1}^{(c,i)}, \Sigma_{t-1}^{(c,i)})$ forward in time and assess how well its future states or successor particles $\{s_t^{(c,j_i)}\}$ can predict the evidence at the next time slice. We calculate an importance weight for these successor states and sum them up to get a look-ahead importance weight for $s_{t-1}^{(c,i)}$ that incorporates the most recent information. In particular, for each factored particle $s_{t-1}^{(c,i)} = (z_{t-1}^{(c,i)}, \mu_{t-1}^{(c,i)}, \Sigma_{t-1}^{(c,i)})$ we first enumerate over the lower dimensional discrete factored mode space to obtain the set of successor mode states $\{Z_t^{(c,j_i)}\}$ reachable from the factored mode $Z_{t-1}^{(c,i)}$. For each factored mode $Z_{t-1}^{(c,j_i)}$, we propagate $X_t^{(c,j_i)}$ by applying an Unscented Kalman Filter (UKF) on $(\mu_{t-1}^{(c,i)}, \Sigma_{t-1}^{(c,i)})$ through the local dynamics model. The importance weight for each factored particle is computed by summing up the contribution from its successor states.

In the join step, the global dynamics is reintroduced back into the factored representation after each projection. This join step is important, as otherwise we would be treating the factored processes as completely independent from one another. The factored particles are joined to form a set of full particles by an equijoin operation that produces all the complete particles that are consistent with some factor in each cluster. It does not make sense to perform equijoin on real-valued continuous variables since they are represented by their sufficient statistics in RBPF. Therefore, we multiply the factored statistics together to form the full statistics for the continuous variables.

In the global dynamics propagation step, for each resampled particle $s_{t-1}^{(i)}$, we perform global mode propagation and sample $Z_t^{(i)}$ from its mode transition probability distribution. After having instantiated the mode $\{Z_{t-1}^{(i)}\}_{i=1}^N$, we propagate the continuous variables by running UKF on $\{\mu_{t-1}^{(i)}, \Sigma_{t-1}^{(i)}\}_{i=1}^N$ to get the time-*t* statistics $\{\mu_t^{(i)}, \Sigma_t^{(i)}\}_{i=1}^N$.

To project a hybrid particle $s_{t-1}^{(c,i)} = (z_{t-1}^{(c,i)}, \mu_{t-1}^{(c,i)}, \Sigma_{t-1}^{(c,i)})$ we project the mode over the factored mode domain and project the Gaussian statistics over the factored continuous domain.

4 Problem Modeling Methodology

We are interested in hybrid models of urban environments that can be represented in the form shown in Figure 3. We represent the state space as follows:

$$Z_t \sim p(Z_t \mid Z_{t-1}, X_{t-1}, G)$$
$$X_t = F(Z_t, X_{t-1}, V_t)$$
$$Y_t = H(Z_t, X_t, W_t)$$

where Z_t denotes the unknown discrete variables where each variable has a finite state space size; $X_t \in \mathbb{R}^{n_x}$ denotes the unknown continuous states; $Y_t \in \mathbb{R}^{n_y}$ denotes the measurements; $V_t \in \mathbb{R}^{n_v}$ and $W_t \in \mathbb{R}^{n_w}$ are independent white noises; and G denotes the guard function that defines the conditions that can trigger transition from one state to another.

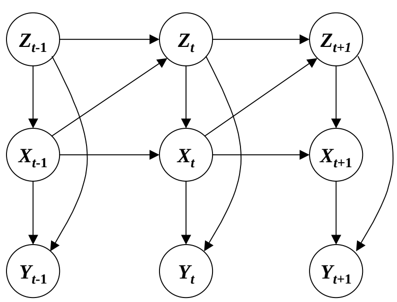

Figure 3: DBN representing a hybrid HMM

As an example (see Figure 4), suppose we are tracking a target "T" (e.g. enemy vehicle), over some period of time and whose current location is "L". Then the variables of the discrete state vector Z_t for Level 1 fusion include the terrain (*go*, *no go*, *slow go*) at L, weather (*sunny*, *rainy*, etc.) at L, area (*named area of interests* 1, 2, 3, etc.) of L: $Z_t = [terrain_t, weather_t, area_t]$; the variables of the continuous state vector X_t represents the coordinates of L and the velocities of U in the $x-y$ plane: $X_t = [px_t, py_t, vx_t, vy_t]$; and the observation vector Y_t represents radar measurements such as range, azimuth, and elevation: $Y_t = [r_t, a_t, e_t]$. An example dependency is that the weather at a particular area affects the mobility of that area, which, in turn, affects the speed of the unit being tracked. The state space $Z_t \sim p(Z_t \mid Z_{t-1}, X_{t-1}, G)$ and $X_t = F(Z_t, X_{t-1}, V_t)$ can be defined based on the dependency relationships among variables. The observation model $Y_t = H(Z_t, X_t, W_t)$ is constructed from the sensor model.

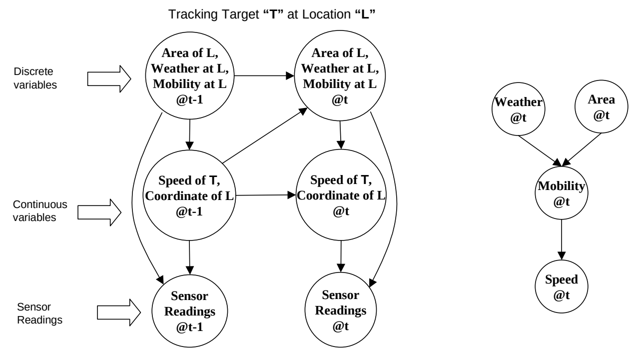

Figure 4: Variables and Dependencies

We applied this methodology to an urban ambush scenario involving a Blue unit and two opposing Red units; the intentions and threat posed to Blue by the Red units was to be inferred in the simulation. Figure 5 presents a block diagram of the DBN developed for the scenario; note that:

- Each block represents several DBN variables.
- The arrows in this case represent one or more causal links between variables in the blocks.
- To illustrate temporal effects, we show two copies of the DBN (for time=t and time=t+1); the causal links in this case are mostly for state persistence over time.

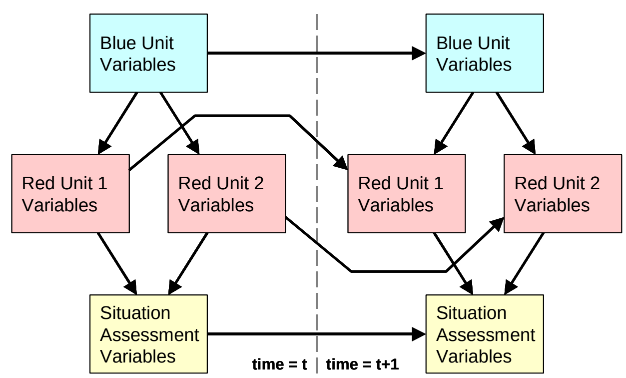

Figure 5 : Scenario DBN Block Diagram

The block organization in the diagram is particularly significant, as it is suggestive of the factorization that will take place during particle filtering. Specifically, the two Red units and the Blue unit operate almost independently, and thus (their variables) are ideal candidates for treatment using separate factored groups of particles during the inferencing operation.

To provide a little more detail on the scenario DBN, we focus on the block of variables for Red Unit 1; these variables are shown in Figure 6, which shows all the variables for the unit, but only intra-unit causal links are shown (i.e. no temporal links, no links with Blue, no situation assessment node links). Points to note about the figure include:

- A mix of discrete and continuous variables is present, with all possible combinations of causality between them, i.e. discrete-to-discrete, discrete-to-continuous, continuous-to-discrete, and continuous-to-continuous. (Only in the first case is causality quantified by conditional probability tables; various functions are employed for the others.)
- The variables track unit location, kinematics, strength, goals, and threat assessments.
- Both Level 1 fusion and some of the Level 2 fusion (situation assessment) for the scenario is handled by the unit variables.

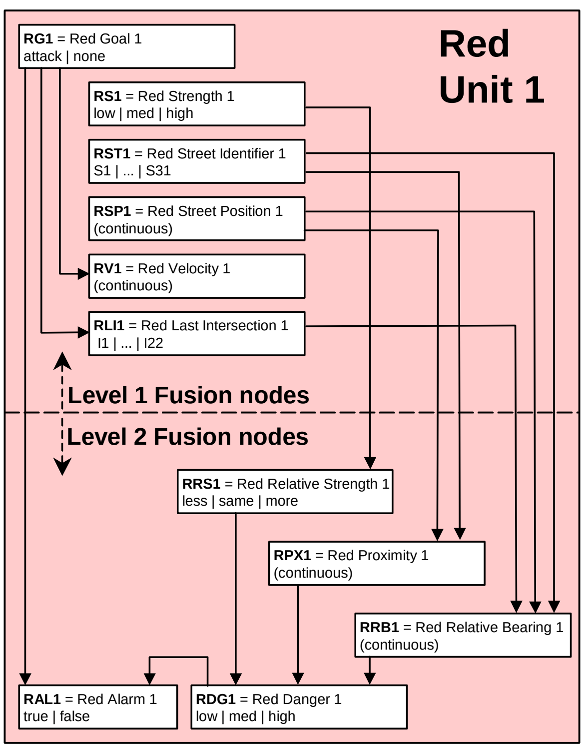

Figure 6 : Scenario DBN Partial Detail

The block of variables for Red Unit 2 is essentially the same as for Red1. The block of Blue unit variables is smaller and simpler, as we don't perform goal or threat assessment or for Blue. In the complete DBN, we aggregate the goal and threat assessments in the Situation Assessment block of variables; the final result is an alarm variable that indicates the degree of concern Blue should have over the (possibly converging) Red forces, for any point in time during the simulation.

5 Prototype Demonstration

Our objective in the context of the chosen scenario was to predict the movements of Red units along the chosen city streets, using factored particle filtering, and then assess the overall situation and threat to the Blue unit from both individual and aggregate Red forces. We started by positioning the scenario in a selected topographical environment, which we then abstracted into a graph that represents roads and positions in the city. Roads and streets are reduced to straight lines connecting to intersection points.

Next we simulated the scenario dynamics, i.e. the movements of the Red and Blue units, in a manner reasonably consistent with ambush operations, as we understood them from our subject matter expert. Similar to the scenario dynamics diagram of Figure 1, we have used 2 Red units and 1 Blue target unit moving along city streets. The attributes considered for each unit includes its position (i.e. what road it is on and its position along the road), velocity, and strength. Red units have an additional goal attribute of either attacking or not attacking the Blue unit.

The general timing and most important events in the scenario are as follows. Blue starts traveling north from his headquarters; we first see Blue at time t=0 (seconds), at the southernmost point of the street grid. Also, Red1 starts traveling at t=0, without any intent to attack, heading generally west to east. Red2, however, starts at t=0 with definite intent to attack Blue (these goals are not directly observable by Blue, but are inferred via the situation analysis portion of the DBN), and heading southwest towards Blue. Later, at time t=90, Red2 realizes that it has missed its planned intercept and assault on Blue at the next intersection, and so initiates radio contact with Red2, which lasts for 10 seconds, and which is observable by Blue intelligence support. During the communication, Red2 enlists support of Red1, so that Red1 changes its goal to attack at t=98. Immediately afterwards, Red1 decelerates, reverses course, and advances upon the Blue position. Red2 continues to chase Blue and close from behind. The scenario concludes at t=160 with attack imminent, and with Blue surrounded, having Red1 closing from the north, and Red2 closing from the south.

The scenario outlined above was simulated in (Matlab) software, providing fairly realistic kinematics for the units. Additionally, the scenario simulation established 'ground truth' values for all system state variables of concern (i.e. those of the DBN in the next section). The simulation data was then used as input to the inferencing algorithms, and for computing metrics of the inference accuracy.

With the simulation (ground truth) and scenario DBN model established, inferencing via the Hybrid-FP algorithm (described above) was performed, with various sizes of 'particle clouds', i.e. different numbers of particles used for inferencing about the DBN variables.

Inferencing results were passed to a software component for visualization, and were also saved for follow-up analysis such as calculating performance metrics. Figure 7 illustrates the visualization component, showing an inference run in progress. Briefly, the various windows show:

- An overview map, located in the right middle of the frame. This gives a birds-eye view of the street map, and shows current positions for Red and Blue units.
- Detail maps for each unit, along the bottom, which show the unit's 'particle cloud' of estimated positions.
- A visualization of the DBN, shown in the upper left of the frame. The DBN visualization shows current state information for all variables in the simulation.
- Finally, a general information window is included, shown in the upper right of the frame. This provides essential background data, such as the current time step, the number of particles in use for the simulation, and various metrics of performance.

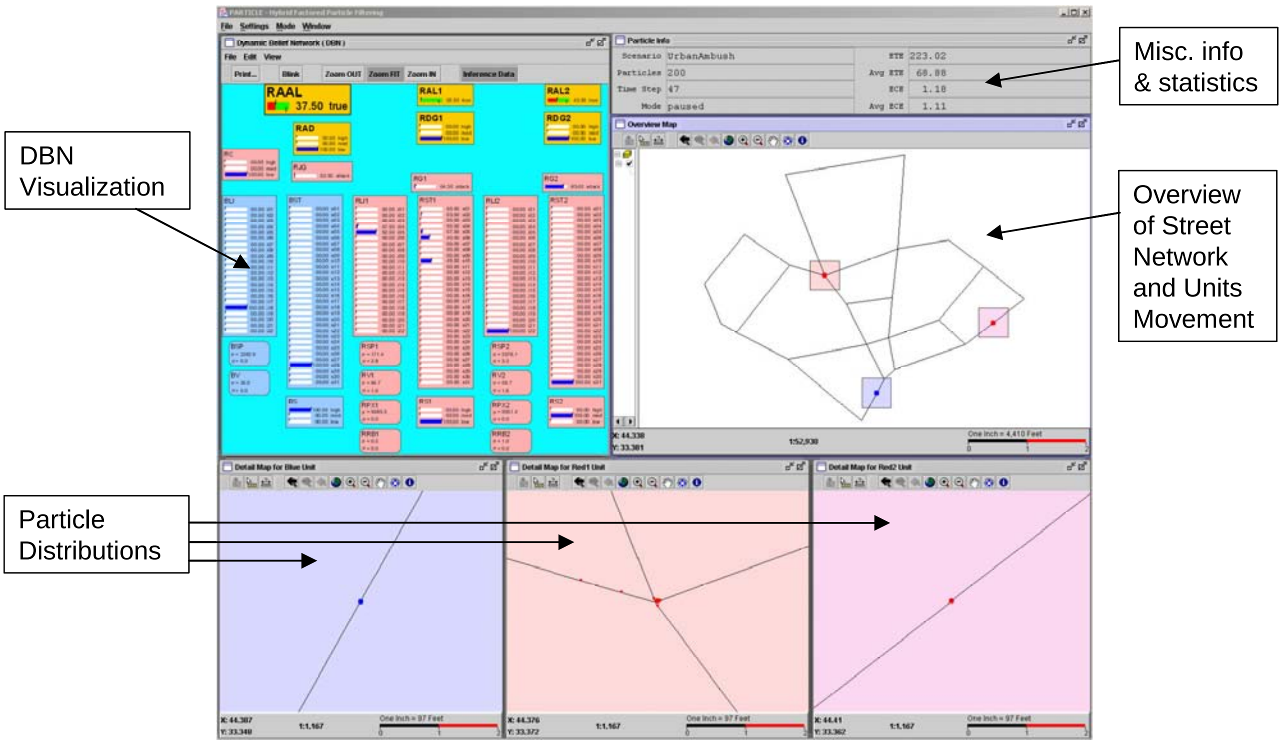

Figure 7 : Software User Interface

6 Performance Metrics

Choosing meaningful performance metrics for the prototype algorithms was difficult as the goals for the project were themselves rather general: we aimed to show that (1) with sufficient (simulated but realistic) input (e.g. intelligence information and real-time sensor feeds) and careful modeling of SME advice, we could successfully predict an impending ambush of Blue forces by Red in a difficult and cluttered urban combat environment. Moreover, we hoped to show that (2) the inferencing estimates (based on particle filtering) could be improved with more computational investment in a larger 'particle cloud'. Finally, we hoped to show that (3) our Hybrid-FP approach significantly eased the computational burden posed by the algorithms.

For simple metrics on how well we accomplished these goals, we produced time-series plots of the most important DBN variables, for different sized particle clouds. The graphs compared the ground truth value and the average (over all particles) inference value for comparison, at each time step. Due to space limitations we cannot present a convincing amount of plot information here, however, the inference values of the critical 'alarm' variables do tend to estimate the ground truth values fairly well; moreover, visual inspection shows that the estimates clearly improve as the 'particle cloud' size increases. Hence we were convinced of success at the first two goals.

As for the final goal, time limitations did not permit specific measurements of Hybrid-FP performance versus that of non-factored particle filtering in the context of the Urban Ambush scenario. However, there is enough specific prior research (Ng et al, 2004) on this topic that we are confident the computational savings are significant enough to continue research and development of the factored approach to particle filtering.

Overall, we feel that the metrics gathered firmly (albeit not precisely) support our approach. In addition, we are considering enhancements to further improve accuracy, performance, and applicability. Figure 8 shows a sample plot of the goal (RG1, RG2 and the joint goal RJG) and communications variables (RC), for 100 particles, compared against the ground truth.

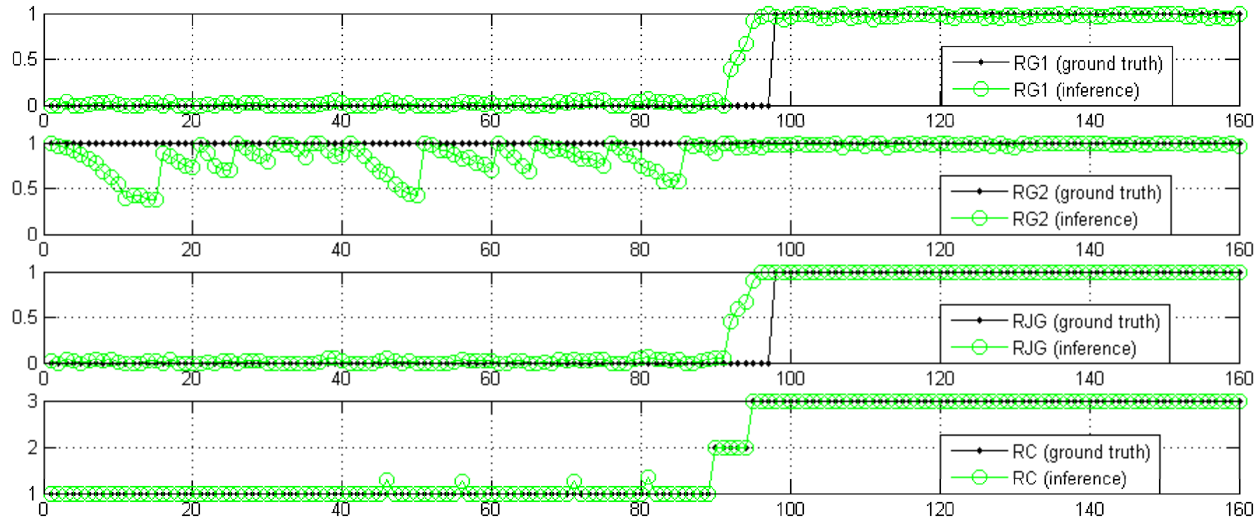

Figure 8 : Sample Performance Metric Plot (Goal variables, 100 particles)

7 Conclusions

We have presented here a factored particle filtering approach to data fusion (DF) and situation assessments (SA) in urban environments. The proposed hybrid approach to problem modeling via DBN by making use of both discrete and continuous variables (without discretization) reduces modeling effort, and enhances modeling accuracy, thereby enhances the accuracy of DF and SA results. The Dynamic Bayesian Network (DBN) formalism, based on probabilistic semantics, allows the representation of uncertain relationships among battlefield concepts as opposed to the logical formalism, which allows only Boolean relationships. The particle filtering approach to inferencing is simple but a powerful approximate DBN inferencing in the presence of both discrete and continuous variables. The hybrid factored sampling (Hybrid-FP) technique is an efficient way of tracking almost independent units in urban environments as illustrated in the ambush scenario. The Hybrid-FP technique solves the well-known 'curse of dimensionality' problem by factoring the state space, and thereby avoids maintaining a large number of particles. It can track high-dimensional hybrid system with reasonable accuracy because it is able to reduce the variance in the sampling process and to enrich the accuracy of the particle by an efficient look-ahead prediction. In the future, we plan to enhance the functionality of DF for tracking units via PF by incorporating richer dynamics and movement models of the units. Note that we depend on processed Level 1 fusion tracking results, wherever available, from the underlying fusion engine. Low-level tracking of every individual unit is beyond the scope of the proposed system. We are exploring the option of asynchronous sampling as a feasible solution to hybrid monitoring to enhance efficiency. Since multiple units naturally evolve at different pace, we want to ideally sample only at the onset of interesting events such as node transitions and not waste too many particles when the unit is in its default or nominal state. Finally, DF and SA in urban environments are particularly challenging due to the enemy's lack of conformity to established tactical doctrine, and we plan to explore a library based DBN construction technique.

References

[1] Arulampalam, S., Maskell, S., Gordon, N., and Clapp, T. (2002). A Tutorial on Particle Filters for On-line Non-linear/Non-Gaussian Bayesian Tracking. IEEE Transactions on Signal Processing. Vol. 50, pp. 174-188.

[2] Avitzour, D. (1995). A stochastic simulation Bayesian approach to multitarget tracking. IEE Proceedings on Radar, Sonar, and Navigation. Vol. 142, pp. 41-44.

[3] Bar-Shalom, Y. and Fortmann, T. (1988). Tracking and Data Association. Academic Press.

[4] Boyen, X. and Koller, D. (1998). Tractable inference for complex stochastic processes. Proceedings of the 14th Conference on Uncertainty in Artificial Intelligence.

[5] Cobb, B. and Shenoy, P. (2003). Inference in Hybrid Bayesian Networks with Mixtures of Truncated Exponentials, Proc. of 6th Workshop on

Uncertainty Processing, Hejnice, Czech Republic, pp. 47-63,
[6] Das, S., Grey, R., and Gonsalves, P. (2002). Situation Assessment via Bayesian Belief Networks, Proceedings of the 5th International Conference on Information Fusion, Maryland, July.
[7] Das, S., Introne, J., Lawless, D., Hoyt, R. and Muza, S. (2004) "Probabilistic Unit Life Status Estimation (PULSE)," Proceedings of the 7th International Conference on Information Fusion, Stockholm, Sweden (June).
[8] Djuric, P., Kotecha, J., Zhang, J., Huang, Y., Ghirmai, T., Bugallo, M., and Miguez, J. (2003). Particle Filtering, IEE Signal Processing Magazine, September.
[9] Doucet, A., de Freitas, N., and Gordon, N. (eds.). (2001). Sequential Monte Carlo Methods in Practice, Springer-Verlag.
[10] Doucet, A. de Freitas, N., Murphy, K., and Russel, S. (2000). Rao-Blackwellised particle filtering for dynamic Bayesian networks. Proceedings of the Conference on Uncertainty in Artificial Intelligence (UAI).
[11] Ghahramani, Z. (2001). An Introduction to Hidden Markov Models and Bayesian Networks. International Journal of Pattern Recognition and Artificial Inteligence, Vol 15(1), pp. 9-42.
[12] Gilks, W., Richardson, S., and Spiegelhalter, D. (eds.) (1996). Markov Chain Monte Carlo in Practice, Chapman & Hall, London.
[13] Gordon, N., Salmond, D., and Smith, A. (1993). Novel approach to nonlinear/non-gausssian Bayesian state estimation. IEE Proceedings on Radar and Signal Processing, Vol. 140(2), pp. 107-113.
[14] Handschin, J. and Mayne, D. (1969). Monte Carlo technique to estimate the conditional expectation in multi-stage non-linear filtering. International Journal of Control, Vol. 9(5), pp. 547-559.
[15] Hue, C., Le Cadre, J-P, and Perez, P. (2002). Sequential Monte Carlo Methods for Multiple Target Tracking and Data Fusion. IEEE Transaction on Signal Processing, Vol. 50(2), pp. 309-325.
[16] Isard, M. and Blake, A. (1998). Condensation – conditional density propagation for visual tracking. International Journal of Computer Vision, Vol. 29, pp. 5-28.
[17] Kanazawa, K., Koller, D., and Russell, S. (1995). Stochastic simulation algorithms for dynamic probabilistic networks. Proceedings of the 11th Conference on Uncertainty in Artificial Intelligence, Morgan Kaufmann, pp. 346-351.
[18] Karlsson, R., Gustafsson, F., and Karlsson, T. (2003). Particle filtering and Cramer-Rao lower bound for underwater navigation. Proceedings of the IEEE Conference on Acoustics, Speech, and Signal Processing (ICASSP).
[19] Kong, A., Liu, J. S., and Wong W. H. (1994). Sequential imputation method and Bayesian missing data problems. Journal of the American Statistical Association, Vol. 89, pp. 278-288.
[20] Lerner, U., Segal, E., and Koller, D. (2001). Exact Inference in Networks with Discrete Children of Continuous Parents, Proc. of the 17th Annual Conference on Uncertainty in Artificial Intelligence (UAI), Seattle, Washington, August, pages 319--328.
[21] Magee, D. and Boyle, R. (2002). Detecting lameness using 'Re-sampling Condensation' and 'multi-stream cyclic hidden Markov Models'. Image and Vision Computing, Vol. 20, pp. 581-594.
[22] Murphy, K. (2002). "Modeling sequential data using graphical models". http://www.ai.mit.edu/~murphyk/papers.html.
[23] Nicholson E. and J. Brady. (1993). Dynamic belief networks for state monitoring. December 1993.
[24] Ng, B., Peshkin, L., and Pfeffer, A. (2002). Factored particles for scalable monitoring. Proceedings of the 18th Conference on Uncertainty in Artificial Intelligence.
[25] Ng, B., Pfeffer, A., Dearden, R., and Hutter, F. (2004). Factored sampling for monitoring nonlinear hybrid systems with autonomous transitions. Submitted to the Conference on Uncertainty in Artificial Intelligence.
[26] Pearl, J. (1988). Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference. San Mateo, CA, Morgan Kaufmann.
[27] Pitt, M. and Shephard, N. (1999). Filtering via simulation: auxiliary particle filters. Journal of the American Statistical Association, Vol. 94(446).
[28] Rabiner, L. (1989). A tutorial on hidden Markov models and selected applications in speech recognition. Proceedings of the IEEE 77, pp. 257-286
[29] Schulz, D., Fox, D., and Hightower, J. (2003). People tracking with anonymous and Id-sensors using Rao-Blackwellised particle filters. Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI).
[30] Wan, E. and van der Merwe, R. (2000). The unscented Kalman filter for nonlinear estimation. Proceedings of IEEE Symposium (AS-SPCC).