This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the WCNC 2008 proceedings.

# Using Swarm Intelligence and Bayesian Inference for Aircraft Interrogation

**Ganapathi Kamath, Xiang Ye and Lisa Ann Osadciw**
Department of Electrical Engineering and Computer Science
Syracuse University, Syracuse, NY 13244-1240
(315)443-3366(office)/(315)-443-2583(fax)
gkamathh/xye/laosadci@syr.edu

**Abstract:** Sensor management deals with the efficient resource allocation to meet mission objectives of the application, air traffic control. A schedule for the sensors is constructed, which simultaneously meets the measurement accuracy and update rate, while minimizing the transmissions from the sensors. Bayesian inference is used to determine management requirements for individual aircraft. Particle swarm optimization, a technique modeled after swarming insects to solve multi-objective optimization problems efficiently, is used to form, test, and search potential sensor schedules for a final schedule solution that maximizes the fitness function. The fitness function is composed of weighted performance estimates enabling quantitative comparisons among schedules. A Bayesian network optimizes the values of the weights as well as the performance requirements to get the best overall performance possible from the entire sensor network system.

**Keywords:** Sensor management, Bayesian networks, particle swarm optimization, multi-objective optimization.

## I. Introduction

Sensor network management integrates three technology areas: data fusion, communications, and signal processing. Data fusion has been a growing research area for some time, which strives to increase the information gathered, and decrease the amount of data transferred [1, 2]. A sensor management system improves the efficiency of the sensors and communications, simultaneously. It accurately predicts the impact of new operating parameters on global performance to select the appropriate sensor settings. Also, the system can receive feedback from the network in the form of actual global performance measurements, which will verify the algorithm as well as the model's accuracy.

This paper describes the design of an automatic scheduling algorithm for an air traffic control (ATC) sensor network, which consists of a Bayesian network (BN) integrated with a particle swarm optimization (PSO) algorithm. This sensor management system collects information from aircraft through wireless communications established between sensors and aircraft. Based on the information, the BN updates the sensor's real-time operating parameters by off-line processing of monitored system performance, dynamically changing environment, and current sensor capabilities. The PSO performs its multi-objective optimization using these parameters searching for the best schedule, which meets performance requirements, such as communicating with higher probabilities, updating the aircraft information at a specified rate, and minimizing the number of communications needed.

In section II, the aircraft traffic control problem is discribed. We present our generalized sensor management system for this problem in Section III. We evaluate our technique's performance through experiments and analyze the results in Section IV and Section V. Finally, concluding remarks are drawn in Section VI.

## II. Problem Description

Aircraft in the coverage area need to be interrogated. The area has smaller regions, which have tailored the requirements to be met for a given aircraft. The aircraft requirements include meeting a probability of detection within a certain amount of time. The requirements are derived using an indirect inference mechanism such as a BN. In this paper the emphasis being on sensor management, a direct mapping from region to requirement is used.

Sensors are deployed in the coverage area to detect the moving aircraft. The surrounding regions can change the performance requirement of an aircraft and its area. Some regions may have more stringent detection requirements than others. The remaining areas, which aren't specified as part of any particular region, are referred to as the surrounding area, which have a relaxed requirement imposed on the aircraft present in them. Sensor placement is a research area, which has been dealt with extensively elsewhere. In this paper, it is assumed that sensors are stationary.

The sensors first detect the aircraft using broadcaster messages from the aircraft or sensor network. The sensors are active during the time of interrogation. All sensors have the ability to receive, but only some have the ability to send interrogation messages. Sensor can either transmit and receive or only receive. Of course, a sensor can't transmit and receive at the same time.

An interrogation involves selecting the transmitter and a set of receivers for interrogating an aircraft. The transmitter power controls the range of an aircraft from the transmitter for interrogation. The transmitter has a maximum power, and the interrogation ranges of different transmitters must not be overlapped to prevent interference between communications. The further the distance from the transmitter, the lower the signal strength received by the aircraft. The probability of reception by the aircraft can be modeled using a distance dependent function. Since the transmitter power is controllable, it is adjusted so that the aircraft receives the signal with 97% probability of communication. The aircraft receives the transmitted signal and broadcasts its response, which are detected by the receivers. The interrogation is repeated until a successful detection occurs. This series of interrogate repetitions, Q is referred to a compound interrogation in this paper. However, the interrogation density, the total number of compound interrogates performed in a region, must be kept low to prevent interference with other communication systems. The overall probability of detection is the product of transmission and reception probabilities.

$$P = P_t \cdot (1-(1-P_{rec} \cdot P_{mofn})^Q) \quad \textbf{(1)}$$

A schedule is generated over a period of the longest update rate for any region. The schedule determines the transmitter, a set of receivers and transmit power needed for each interrogation in any region at a given time. Aircraft in a higher priority region have shorter update periods, while aircraft in a lower priority region, such as surrounding area, have longer update periods. There are as many compound interrogations as the number of update periods in a scheduling interval.

The objective is to generate the schedule, which minimizes interrogation density while maintaining update interval constraints and detecting aircraft with a detection probability as high as possible.

## III. System Design

### A. Sensor Management System

Sensor management, which includes the resource allocation problem addressed in this paper, may optimize the operation of any number and variety of sensors for a given mission. Thus, management is composed of different sensor control tasks for each application and sensor suite composition. This paper looks at efficiently allocating sensors so that their fused report of the aircraft's position on the ground meets both timing and successful communication probability, between sensor and aircraft, requirements. The algorithm differs significantly from other sensor management algorithms that may use information theory [ 3 ] to allocate a sensor's tracking and surveillance energy on a target by its use of swarm intelligence and artificial intelligence, Bayesian networks, which give the algorithm great scalability as well as the ability to balance many performance needs. The algorithm inherently handles sensor degradation by ignoring sensors that have poor status or are off-line. The algorithm scales to larger problems easily and, even more importantly, has the flexibility to redefine the mission based on its performance requirement needs based on geographical location [ 4 ]. Thus, a very flexible and scalable algorithm, consisting of both the Bayesian network for defining the requirements and swarm intelligence for optimizing management decisions, can be designed for all types of sensor networks as pointed out in [ 1 ]. As new sensor networks emerge connected through new communication hardware, the traditional sensor management systems using fusion trackers, fuzzy logic, artificial intelligence, or Dempster-Shafer theory [ 5 ] can not be easily transitioned to managers of the new sensor networks. The sensor management system designed for the ATC problem is shown in Figure 1 .

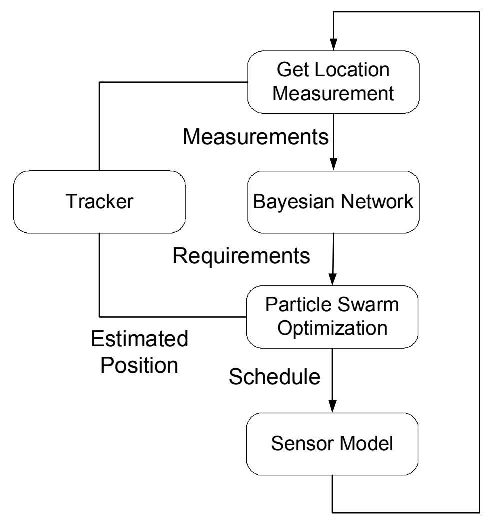


Fig. 1. Sensor Management System

### B. Mission Manager

#### 1) Bayesian Network Algorithm

The BN maintains the desired performance values and the importance of meeting parameter requirements for the mission. This algorithm has recently emerged as a tool for reasoning problems by modeling elements of uncertainty and using complex probability distributions in terms of joint and conditional probabilities of the key performance variables.

The BN is a graphical model that represents probabilistic relationships among variables of interest. Let $\boldsymbol{X}=\{X_1, X_2,..., X_n\}$ denote a set of interested variables, and $x_1, x_2,... x_n$ be the possible values for these variables. The BN consists of a directed acyclic graph (DAG) that corresponds to the conditional independence assertions about variables in $\boldsymbol{X}$ and the local probability distributions associated with each variable, which define the joint probability distribution for $\boldsymbol{X}$ together. The process for obtaining DAG can be implemented step by step. It starts by designating $X_1$ as a root cause, which means it is not influenced by any other variables, and assign it the local probability $P(x_1)$. Next, from the node $X_2$, if $X_1$ is a cause for $X_2$, establish a directed edge from $X_1$ to $X_2$ and quantify this link by the conditional probability $P(x_2|x_1)$. $X_1$ is called a parent node of $X_2$. If $X_1$ has no influence on $X_2$, they are independencies. At the $i^{\text{th}}$ stage, the node $X_i$ has parents $\{X_1, X_2,..., X_{i-1}\}$, represented by drawing the appropriate parent to child links and quantify this link group by the conditional probability $P(x_i|X_1, X_2,..., X_{i-1})$. This hierarchical process continues until all variables have a place in the DAG, and all causal relationships between parents and children are shown by direct links in the graph. The network formalism [9] then provides the joint probability distribution for $\boldsymbol{X}$, given by

$$P(x_1,...,x_n) = \prod_{i=1}^{n} P(x_i \mid \mathrm{X}_1, \mathrm{X}_2, \ldots, \mathrm{X}_{i-1}) \quad (2)$$

The system generates data that reflects its performance to the BN that uses it as feedback. This performance data is assumed to have certain probability distributions. Then BN converts uncertainty, such as the current operation environment of the sensors, into decisions.

*2) Bayesian Network Processor*

For this scheduling system, the BN assists in the final decision making choosing the priorities for meeting required performance parameters using both expert knowledge and evidence data. This BN receives performance data characterizing current aircraft motion and sensor status. Then the BN converts these to performance parameter priorities for various regions. These priorities are used by the PSO to select optimum sensor operating commands that best handle the current scenario conditions and sensor suite status.

Initially, the BN is organized based on expert knowledge. The graphical model is illustrated in Figure 2. All the variables are dependent on region so it is assumed that this graphical model pertains to one region only. When building the probabilistic model, the components of interest are identified as the operating parameter requirements for the system. In this case, they are aircraft detection probability requirements and update time requirements, which have causal relationship with the sensor performance and aircraft profiles at the current time.

The sensor status, the percentage of down sensors and sensor clumping are based on results collected from the sensors in a particular region. The performance miss rate is controlled by the sensor status in the network. Higher percentage of down sensors causes a higher probability of communications missing. Sensor clumping refers to the non-uniform distribution of sensors in a region. If an area within a region has a denser sensor concentration, the aircraft are at a much lower risk of receiving no observation. If the region has a uniform sensor concentration, all aircraft in that region have the same probability of being detected. Hence, as shown in Figure 2, the percentage of down sensors and sensor clumping are the parent nodes of communication miss allowed.

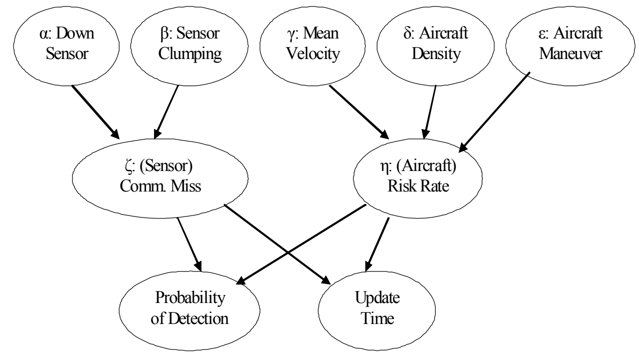


Fig. 2. Bayesian Network for Scheduling System

Moreover, the operating parameter requirements are also influenced by aircraft profiles, such as aircraft density, mean aircraft velocity and percentage of aircraft maneuvering. If the aircraft density is too high in one region, an aircraft may receive too many communications since the high aircraft density triggers a high interrogation rate for the entire region. On the other hand, the extremely low aircraft density may also cause too many interrogates as a result of attempting to maintain detection on new aircraft further away. Similar logic is applied to derive the relationships between the aircraft and sensor conditions and the performance requirements within regions. The requirements are sent to the PSO, which uses them to assign sensor resources to the aircraft.

Then based on the evidence collected, the directed edge is quantified by the conditional probability to specify the strengths of the causal influence. Since the sensor status and aircraft profiles are conditionally independent given the performance requirements, the conditional probability distributions for the outputs, which are aircraft detection probability (Pd) and update time (Ut), are

$$P(Pd \mid \alpha,\beta,\gamma,\delta,\varepsilon,\xi,\eta) = P(Pd \mid \xi,\eta) \quad (3)$$

and

$$P(Ut \mid \alpha,\beta,\gamma,\delta,\varepsilon,\xi,\eta) = P(Ut \mid \xi,\eta) \quad (4)$$

Following the Bayes' theorem,

$$P(Pd \mid \xi,\eta) = \frac{P(\xi,\eta \mid Pd)P(Pd)}{P(\xi,\eta)} \quad (5)$$

and

$$P(Ut \mid \xi,\eta) = \frac{P(\xi,\eta \mid Ut)P(Ut)}{P(\xi,\eta)} \quad (6)$$

The assumed prior beliefs, $P(Pd)$ and $P(Ut)$, are updated to the posterior beliefs, $P(Pd\,|\,\xi,\eta)$ and $P(Ut\,|\,\xi,\eta)$, which are used to make decision, based on the supporting evidences. The supporting evidences, $P(\xi,\eta\,|\,Pd)$ and $P(\xi,\eta\,|\,Ut)$, are collected from system simulations without BN.

### C. Particle Swarm Optimization

The PSO is an optimization technique for multi-objective optimization problems that searches for an optimal solution in a fitness landscape. The fitness landscape is the range of the multi-objective that provides a metric for evaluating how optimal a solution is. The PSO algorithm is used when their aren't any suitable heuristics or algorithms that one can arrive at an optimal solution directly.

*1) The basic PSO algorithm for continuous variables*

Eberhart, R. C. and Kennedy J in 1995 lay the framework for the PSO using technique in [6]. The PSO operates iteratively on several particles, collectively known as the swarm. Each particle in the swarm represents a solution in the search space, which may not be optimal. A particle's solution is represented by using dimensions for each variable parameter needed for the sensor networks. A fitness function is the metric to determining the solution's optimality. The fitness of the particles is evaluated at every iteration. The position of the particle with the best fitness, called the global best, is preserved in memory. Each particle also preserves in memory the best position and its best fitness, called the particle best. These best fitness positions influence how the solutions change or particles move in the search space.

When a dimension $d$ of the $i^{\text{th}}$ particle's position is updated to a new value in the $t^{\text{th}}$ iteration, the velocity $v$ of the particle decides the new the value $x$ of each dimension $d$. The dimension's position update equation is given by

$$x_{id_{t+1}} = x_{id_t} + v_{id_t} \tag{7}$$

When the velocity is updated in the $t^{\text{th}}$ iteration, the velocity $v$ of the dimension $d$ of the particle decides where the particle position moves next. It is influenced by the particle best "pbest' and the global best "gbest". The "U"s are random uniform values between zero and one, which introduce a random scale factor to force all the particles to have some random movement. The dimension's velocity equation is given by

$$\begin{aligned} v_{id_{t+1}} &= c_0 \cdot v_{id_t} \\ &+ c_1 \cdot U \cdot (x_{id_t} - x_{id_{pbest}}) \\ &+ c_2 \cdot U \cdot (x_{id_t} - x_{id_{gbest}}) \end{aligned} \tag{8}$$

The particle swarm learns the characteristics of the fitness landscape and captures the essence in the particle and global best. This is accomplished through maintaining a local best solution for each particle and the global best solution is maintained after the particles share their own local best. After several iterations, the PSO converges to a solution.

*2) Discrete variable adaptation*

A straight-forward approach for discrete variables adds rounding to find particle positions. This approach is subject to early convergence leading to non-optimal solution. The displacement equations for the special case of discrete binary PSO [7] are

$$S_{id} = \frac{1}{1+e^{-V_{id}}} \tag{9}$$

and

$$x_{id} = \begin{cases} 0 & \textit{if } \left(rand() <= S_{id}\right) \\ 1 & \textit{if } \left(rand() > S_{id}\right) \end{cases} \tag{10}$$

This discrete binary variable PSO has better convergence properties than the rounded version such as a reasonable convergence time and more likely to converge to global rather than local optima.

*3) Discrete multivalue adaptation*

From experimental simulations, it is apparent that the discrete binary PSO performs better than the adaptation of the continuous binary variable PSO using rounding. The probabilistic nature of particle displacement, which depends on a velocity term, is the single most important factor that causes the swarm efficiently and fully explore the search space [8]. The new algorithm incorporates the advantages of the discrete binary PSO to any range of discrete values from 1 to $M$. The equations for the multivalued case are

$$S_{id} = \frac{M}{1+e^{-V_{id}}} \tag{11}$$

and

$$x_{id} = round(N(S_{id}, \sigma \cdot (M-1)))\ . \tag{12}$$

and

$$x_{id} = \begin{cases} M-1 & \textit{if}\, x_{id} > M-1 \\ 0 & \textit{if}\, x_{id} < 0 \end{cases} \tag{13}$$

*4) Optimization Processor*

In this ATC application, the PSO receives its requirements from the BN and the estimated position of the aircraft from the tracker. The swarm consists of 25 particles. Each particle is made up of a vector of dimension sets. A dimension set correspond to the resources associated with an interrogate. They include dimensions that correspond to the time of interrogation, various indexes that correspond to the selection of receivers and transmitters that need to be allocated to the sensors. A binary random variable in each set decides if that particular interrogate is required.

Given the objectives, the setup of the fitness function in this problem is crucial for arriving close to the optimal solu-

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the WCNC 2008 proceedings.

tion. The problem is modeled as a maximizing problem, and the optimal solution has the highest fitness. Non optimal solutions are those that have been penalized for one or more violations of desirable objectives. In the current problem's context, the fitness function is designed to penalize the following: high interrogation density, interrogations close in time and space, transmitters used repeatedly, aircraft missed and interrogates less than required Pd.

## IV. Simulation

There are 5 defined regions in the given coverage area, region A, B, C, D and the surrounding area. The scenario is modeled over a 1000 second period. A complete schedule is required to be generated every 4 seconds. There are 40 sensors in the whole area. 15 of them function as RT, and the rest are RO. The sensor positions remain close to a hexagonal arrangement. Moreover, the sensors are dispersed uniformly among the area with the same rate of RT and RO. Figure 3 shows the sensor placement. The sensors are simulated to have a certain probability of failure, which affects the detection performance. Also, there exists another error from the joint measurement performed by the interrogation.

There are 30 aircraft moving around this area. The aircraft motion model returns the aircraft positions as a function of time. In this paper, the aircraft motion is simulated in the three-dimension space. This is done by designing actual flight paths for each aircraft by choosing key positions, which are somewhat randomness with the distance and phase to the center. Cubic spline interpolation is used to determine positions of aircraft as they travel between these positions. Periodically, each aircraft comes close to the center in region A. At this time, the aircraft descend closer to the ground as if landing, and ascend as they take off on the runway. Figure 4 shows a snapshot of the aircraft while in motion.

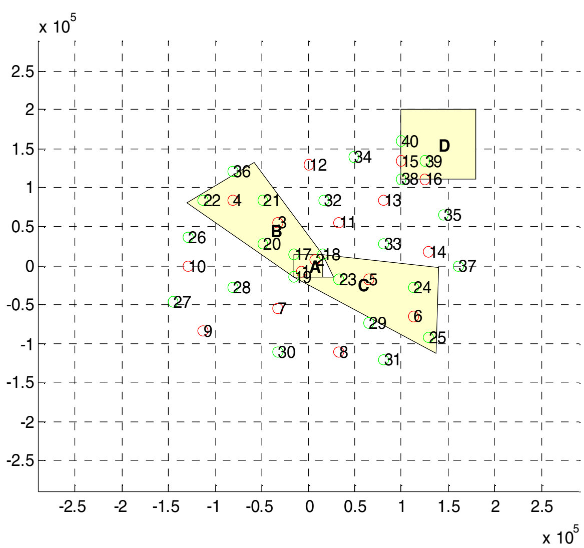

Fig. 3. Sensor Placement

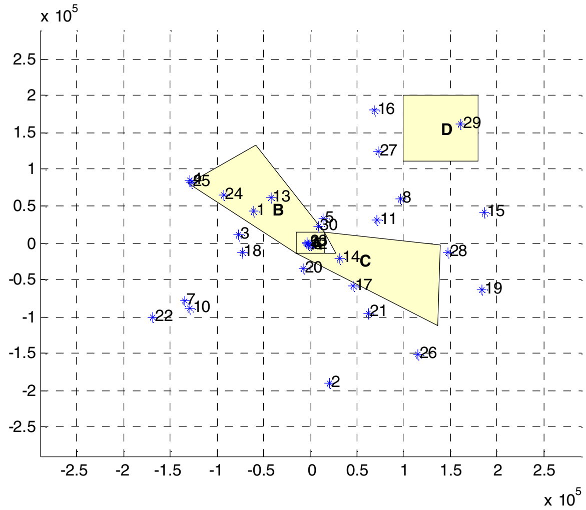

Fig. 4. Sampled Aircraft Positions

Prior to running the BN, the sensor status and aircraft profiles are collected for this scenario to initialize the BN. The probability of detection requirement is allowed to take 10 states varying from each region. The update time is one of three states: 1 second, 2 seconds and 4 seconds. In a given region, the data collection process is executed for each combination of detection probability and update time. Then the data are analyzed for joint probabilities, $P(\xi_k, \eta_l, Pd_i)$ and $P(\xi_k, \eta_l, Ut_j)$. The subscript presents the state number of each variable. These joint probabilities are then given as inputs to the BN during the simulation.

After the joint probability table is calculated, prior probabilities are set uniformly for aircraft detection probability and update time requirements: $P(Pd_i) = 1/10$ and $P(Ut_j) = 1/3$, $i = 1, 2, ..., 10$, $j = 1, 2, 3$. Then, the states of the communication miss percentage and aircraft risk rate, $(\xi_k, \eta_l)$ are chosen based on the system performance at current time. The conditional probabilities are obtained by marginalizing the joint probabilities of collected data. Next, the specific values for the current situation, $P(\xi_k, \eta_l \mid Pd_i)$ and $P(\xi_k, \eta_l \mid Ut_j)$ are selected corresponding the system performance. The interesting posterior conditional probabilities are found from

$$P(Pd_i \mid \xi_k, \eta_l) = \frac{P(\xi_k, \eta_l, Pd_i)}{P(\xi_k, \eta_l)} = \frac{P(\xi_k, \eta_l \mid Pd_i) P(Pd_i)}{P(\xi_k, \eta_l)} \quad (14)$$

and

$$P(Ut_j \mid \xi_k, \eta_l) = \frac{P(\xi_k, \eta_l, Ut_j)}{P(\xi_k, \eta_l)} = \frac{P(\xi_k, \eta_l \mid Ut_j) P(Ut_j)}{P(\xi_k, \eta_l)} \quad (15)$$

At last, the $Pd$ and $Ut$ values are returned to PSO, which correspond to the maximum posterior conditional probability.

## V. Results

The mission manager, BN provides the performance requirements to the PSO, such as the probability of detection and update period for each aircraft.

Then, the PSO searches the fitness landscape to find an optimal schedule to the sensor scheduling problem. The plot of the solution's fitness is shown in Figure 5.

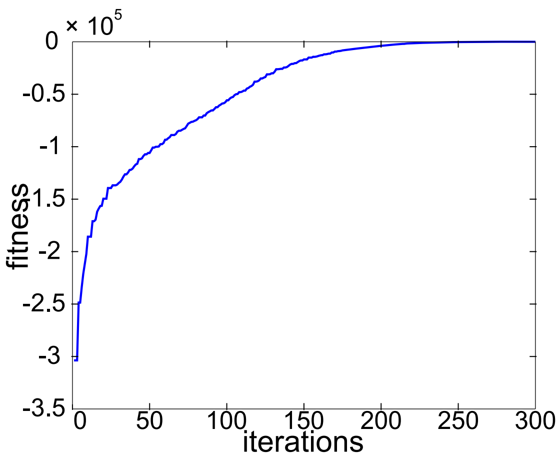


Fig. 5. Fitness Plot

The overall system is run over a 1000 second period with and without the BN, respectively. For each scheduling interval, the ratio between the number of aircraft that have not met the requirements to the total number of aircraft is computed. The probability of unsuccessful communication is then calculated by averaging these ratios over all the scheduling intervals.

Table 1 compares the percentages of unsuccessful communication averaged over all scheduling intervals in the two situations mentioned above. The BN gives better performance for smaller regions, such as region A. The percentage of unsuccessful communications in region A has improved 74.30%. The improvement is due to the fact that aircraft come for landing in region A, which in turn causes a higher aircraft density and maneuver concentration. The higher target density requires more interrogations causing the lower risk of no observation. Sensor coverage also impacts the probability of successful communication, as happened in region D, which has lower sensor coverage. BN has a large impact on the aircraft with simulated flight paths and little effect on simple, random motion.

TABLE I. Unsuccessful Communication Cumulative over All Scheduling Intervals

| | A | B | C | D | Sur. |
|---|---|---|---|---|---|
| Without BN | 0.0249 | 0.1077 | 0.0149 | 0.6892 | 0.1155 |
| With BN | 0.0064 | 0.0770 | 0.0109 | 0.7159 | 0.1459 |

## VI. Conclusion

In this paper, the BN and PSO algorithms have been applied to the aircraft traffic control problem. We show that the BN adapts to the ground situation at runtime and decides requirements for the PSO algorithm so that it can meet the overall mission objective of reducing the interrogation density. The PSO then comes up with a schedule that meets the given multi-objective requirements of performance to update intervals while at the same time increasing the probability of detection. The overall effect improves each aircraft detection results as well as increase the probability of successful communication.

### Acknowledgements

This research was supported by Sensis Corporation.

### References

[1] Denton, R.V., E.I. Alcaraz, J. Llinas, and K.J. Hintz, *Towards Modern Sensor Management Systems,* Science of Command and Control, Part III: Coping with Change, AFCEA International Press, Fairfax, Va., 1994, pp. 118-134.

[2] Gregory A. McIntyre, *A Comprehensive Approach to Sensor Management and Scheduling.* Doctoral Dissertation, George Mason University, Fairfax, VA, 1998.

[3] Stan Musick, Keith Kastella, *Comparison of Sensor Management Strategies for Detection and Classification,* 9th National Symposium on Sensor Fusion, Naval Postgraduate School, Monterey, CA, 1996.

[4] Lisa Osadciw, Kalyan Veeramachaneni, *A Controllable Sensor Management Algorithm Capable of Learning*, SPIE Homeland and Defense Symposium, April, 2005.

[5] L. Rothman and S. Bier, *Evaluation of Sensor Management Systems*, IEEE 1989 National Aerospace and Electronics Conference, 1747-1752 vol.4, NY, 1989.

[6] Eberhart, R. C. and Kennedy, J. *A new optimizer using particle swarm theory,* Proceedings of the Sixth International Symposium on Micromachine and Human Science, Nagoya, Japan. pp. 39-43, 1995.

[7] Kennedy, J. and Eberhart, R. C. *A discrete binary version of the particle swarm algorithm,* Proceedings of the World Multiconference on Systemics,Cybernetics and Informatics 1997, Piscataway, NJ. pp. 4104-4109, 1997.

[8] Kalyan Veeramachaneni, Lisa Osadciw, Ganapathi Kamath, *Probabilistically Driven Particle Swarms for Discrete Multi Valued Problems: Design and Analysis,* IEEE swarm intelligence symposium Hawaii, April, 2007.

[9] David Heckerman, *A Tutorial on Learning with Bayesian Networks,* Microsoft Research Technical Report, MSR-TR-95-06, March 1995.