`On a Tracking Scheme with Probabilistic Completeness for a Distributed Sensor Network

K Madhava Krishna Henry Hexmoor
Dept. of Computer Science and Computer Engineering
University of Arkansas
Fayetteville, AR 72701
mkrishna@uark.edu hexmoor@uark.edu

Abstract: *A framework and analysis for a distributed sensor network based surveillance system is presented here. In a previous effort [1] we have presented methodologies for coordination and decision-making amongst sensors for tracking targets while in [2] we presented the results of endowing the sensor network with autonomy. Sensors monitor targets that crisscross a rectangular surveillance zone. When a sensor pursuits a target it leaves areas unguarded through which other targets can get past undetected. In this paper we presents a methodology that computes the tracking time for a sensor that guarantee detection of a required fraction of the targets expected to crisscross its home area to an arbitrary probabilistic threshold. The home area of the sensor is the area guarded by it when it is stationed at its home position, its default position when it is not in pursuit of a target. Simulations are presented that corroborate with the probabilistic model developed and hence vindicate its correctness. A framework for extending the probabilistic model to a system where multiple sensors guard the same area is also presented.*

1. Introduction

This paper describes a methodology that guarantees probabilistic completeness for sensors that track targets in a multi-sensor setting. Each sensor guards in its default stationary state an area called the home area of the sensor. For a sensor s_j, its home area is denoted by H_j. The robots perform surveillance over a rectangular (square) surveillance zone. The surveillance zone is divided into number of square shaped cells as shown in figure 1 for the sake of modeling. The figure shows the sensors placed in their home positions. The radius of vision of the sensor equals the length of the diagonal of the cell. However the sensor only considers those targets that lie within its four neighboring cells as targets within its field of vision. This area representing its field of vision in its home position is also called as the home area of that sensor for the remaining of this paper. The home area of each sensor is depicted by thick squares. The simulation environment used for testing our strategies has been developed through *Borland's JBuilder IDE* on the Java platform.

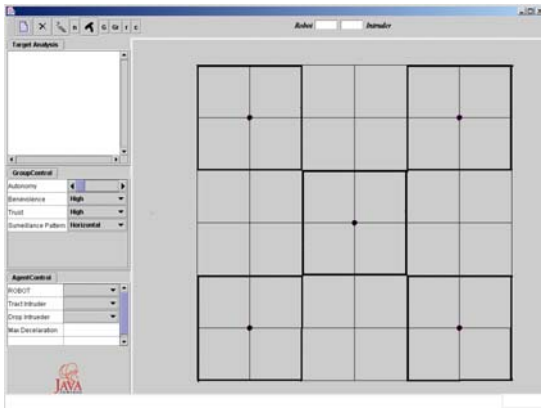

Figure 1: The rectangular surveillance zone with sensors depicted as circles ensconced in their home positions. The home area of each sensor is denoted by thick squares

A sensor allocates itself to one of the targets within its field of vision through a resource allocation process modeled through fuzzy rules [1]. The sensor further decides if it would monitor the target by remaining stationary or by pursuing (tracking) it. When a sensor tracks a target it leaves areas in its home position unguarded. The tracking time for a sensor s_j, denoted by t_j represents the time for which the target would be away from its home position H_j. The tracking time can be modulated based on the fraction of the number of targets that a sensor is expected to detect within a probabilistic threshold. If $\hat{N}_T$ denote the number of targets expected to crisscross H_j within a temporal window Γ and

f be the lower bound on the fraction of the number of targets, $\hat{N}_\Gamma$, to be detected and Ω represent the threshold the paper presents a framework for computing $t_j = g\left(\Gamma, f, \hat{N}_T, \Omega\right)$. Here g is a function that guarantees that at-least a fraction f of the targets, N_Γ, are detected to a probabilistic guarantee of Ω. In other words $g : P\left(n \geq f \cdot \hat{N}_\Gamma\right) \geq \Omega$, where P is the probability computation over the random variable n that denotes the number of targets detected.

The rest of the paper is organized as follows. Section 2 presents the current work in the context of similar works found in the literature. Section 3 presents the formulation of the methodology and section 4 depicts the efficacy of the methodology in simulations. Section 5 extends the formulation to an environment where multiple sensors are placed with the responsibility of guarding the same home area. Section 6 concludes and provides further scope of this work.

2. Background Review

The problem of multi sensor surveillance involves detection of multiple intrusions and/or tracking through coordination between the sensors. Detection and target tracking has been researched from multiple viewpoints. Some efforts have focused on the problem of identifying targets from a given set of data through particle filters [3, and probabilistic methods [4]. The problem of data association or assigning sensor measurements to the corresponding targets were tackled by Joint Probabilistic Data Association Filters by the same researchers such as in [4]. Kluge and others [5] use dynamic timestamps for tracking multiple targets. Krishna and Kalra [6] presented clustering based approaches for target detection and further extended it to tracking and avoidance in [7]. The focus of these approaches has been on building reliable estimators and trackers. They do not use distributed sensors and are not directly useful for the problem of large area surveillance.

In the context of distributed task allocation and sensor coordination Parker [8] proposed a scheme for delegating and withdrawing robots to and from targets through the ALLIANCE architecture. The protocol for allocation was one based on "impatience" of the robot towards a target while the withdrawal was based on "acquiescence". Jung and Sukhatme [9] present a strategy for tracking multiple intruders through a distributed mobile sensor network. Lesser's group have made significant advances to the area of distributed sensor networks [10] and sensor management [11]. In [9] robots are distributed across a region using density estimates in a manner that facilitates maximal tracking of targets in that region. The decision for a robot to move to another region or to stay in its current region is based on certain heuristics. The authors of this paper present their scheme for resource allocation and coordination in a distributed sensor system through a set of fuzzy rules in [1] and further analyze the behavior of system by varying the autonomy of the sensors in [2].

In none of the above efforts is a strategy for guaranteeing some form of completeness is presented. This paper is unique from other efforts in multi-sensor systems in that it offers a tracking strategy for sensors that modulates their tracking time such that a required number of targets are detected within probabilistic guarantees. The authors in [12] present a framework that provides for meeting a targeted search time within probabilistic guarantees for a cooperated UAV search. However the computations and the basis for their framework is disparate from this effort and is presented for a different application and motivation.

3. The Methodology

Targets are modeled percolating in a Poisson fashion at the rate λ through each cell, which has one of its edges on the boundary or the perimeter of the surveillance area. For a surveillance zone such as in figure 1 consisting of six cells along each side of the perimeter, the rate of entry is 6λ. λ is fixed at 0.1 for all the examples discussed in this paper. Then the apparent rate at which each sensor would see a target, λ_{SJ} provided it is stationary is given by the following approximation:

$\lambda_{SJ} = \lambda \sum_{k=1}^{P} \frac{\theta_k}{\pi}$, where, θ_k is the angle subtended from the point where the target enters the boundary at the home position of sensor sj. Since the entry points of the arriving targets are not known a-priori, θ_k is computed assuming

that the target infiltrates at the midpoint of the cell edge that coincides with the perimeter of the zone. In the figure below (figure 2) the targets are assumed to enter at points p1, p2, p3, … along the perimeter of the surveillance zone. For the sensor centered at 'b', the angle subtended by the target entering at p4 is shown marked θ. This angle covers the span of all the targets that will cross the region of surveillance of the sensor at 'b' by targets entering at p4. The total span of the angle for a target entering at all those points is π radians or in other words all targets that enter the surveillance zone have to necessarily be within a span of π radians from the point of entry for them to be within the surveillance zone.

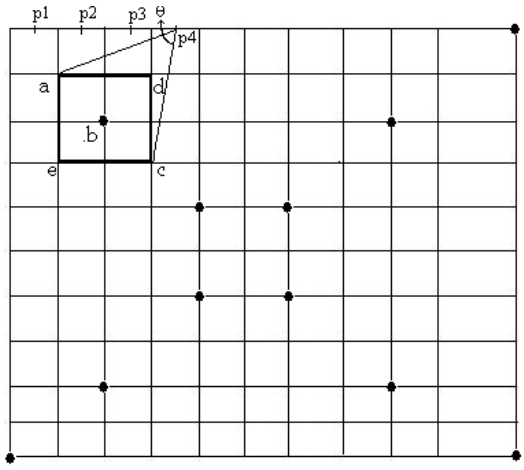

Figure 2: Targets are modeled as entering at locations p1, p2, …, p4 at the midpoints of the cell edges that coincides with the perimeter of the zone.

Let t_j be the time for which a sensor sj is away from its home position in pursuit of a target as mentioned in section 1. We compute the apparent time T_a (the time for which a target that would have been in the field of vision of sj had sj been stationary at its home position perceives sj to be away) as:

$$T_a = \begin{cases} 2\int_0^{t_j/2} \frac{t}{T_\kappa} dt \quad ; t_j \le 2T_\kappa \\ 2\int_0^{T_\kappa} \frac{t}{T_\kappa} dt + 2\left(t - T_\kappa\right)_{T_\kappa}^{t_j} ; t_j > 2T_\kappa \end{cases} \quad (1)$$

Here, κ is the fraction of the home area left unguarded by sj as it moves away from its home. The upper limit of the integral T_κ denotes the time at which the sensor leaves its original area completely unguarded. If T_{esc} represents the average time for which a target stays in the home area of a sensor, the probability that a target is detected by a sensor is given by:

$$p = \frac{T_{esc}}{T_a}. \quad (2)$$

Let n be the random variable denoting the number of detections made over a temporal window Γ as before. The probability of detecting exactly k of the N_Γ targets expected to arrive in Γ is given by the familiar binomial distribution:

$$P\left(n = k / X = \hat{N}_\Gamma\right) = \binom{\hat{N}_\Gamma}{k} p^k \left(1-p\right)^{\hat{N}_\Gamma - k}$$

Here X is a Poisson random variable that measures the number of targets arriving. The resultant probability of detecting k such targets then becomes

$$P\left(n = k\right) = P\left(n = k / X = \hat{N}_\Gamma\right) \cdot P\left(X = \hat{N}_\Gamma\right)$$

It can be shown that the above resultant probability once again has a poisson distribution with parameter $\lambda_{SJ} p$. Hence the probability of detecting k targets has the representation

$$P\left(n = k\right) = e^{-\lambda_{SJ} p} \frac{\left(\lambda_{SJ} p\right)^k}{k!} \quad (3).$$

The tracking time t_j is eventually computed by making use of equations (1), (2) and (3) and that which would satisfy the following guarantee condition:

$$P\left(n = k \ge f \cdot \hat{N}_t\right) \ge \Omega \quad (4)$$

The formulation is described graphically through figures 3a and 3b. Consider the sensor stationed at home position 'b' as in figure 2. Targets are expected to enter its home area at a rate $\lambda_{SJ} = 0.5$. Then over a time interval of 20 units the expected number of arrivals is either 9 or 10. This is shown in figure 3a with the mean expected arrival showing a peak in the probability distribution function at 9 and 10. If a fraction of at-least $f = 0.7$ of these targets (7 of them) are to be detected with a minimum threshold of $\Omega = 0.55$ the rate at which the sensor detects, $\lambda_{SJD} = \lambda_{SJ} p$ is computed to be 0.4 and the

binomial detection probability of a target turns out as $p = 0.8$. Having computed p the wandering time of the sensor t_j is derived from equations (2) and (1). The dotted vertical line in figure 3a indicates the lower bound on the fraction of the number of arriving targets that needs to be detected using the above threshold. The only constraint on the user in fixing the threshold is that $\lambda_{SJD} \leq \lambda_{SJ}$ needs to be obeyed. In other words the rate of detection cannot be greater than the rate of arrival.

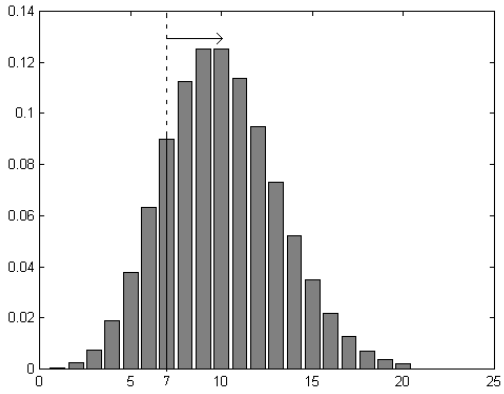

Figure 3a: The probability distribution function for target arrivals. The mean expected number of target arrivals over an interval of 20 units is 10. The dotted line and the arrow to the right indicates a minimum of 7 targets need to be detected by the sensor.

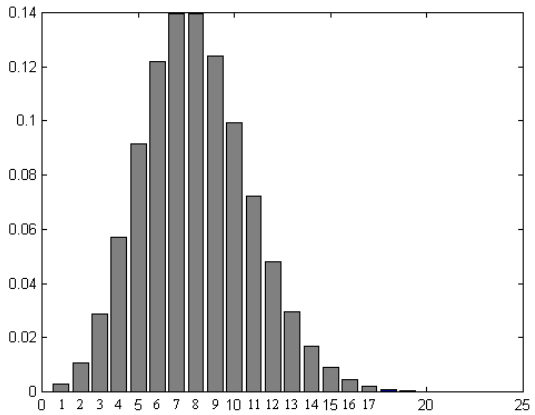

Figure 3b: The probability distribution function for target detections for a minimum threshold of 0.55 for the arrival distribution shown in figure 3a. The mean expected number of detections is 7.

Since double precision arithmetic does not allow computation of factorials beyond 20 the normal approximation to poisson distribution is used in our computations.

4. Simulation Results

The first objective is to evaluate empirically the validity of equation (2) that ascertains the probability of detecting a target by a sensor while it is in motion. For this purpose a single cell environment with one sensor such as in figure 4 is considered. Targets are introduced in poisson fashion at the midpoints of the four boundaries of the cell. The sensor's home position is at the center of the cell. The wandering time t_j is calculated for a given value of Ω and f. The sensor is away from its home position for t_j units. The sensor does not track a particular target. It is merely away from its home position. The number of targets that crisscrossed the cell during this time interval and the number of those detected were recorded.

The results are tabulated in table 1. The first column represents the desired fraction of the targets that need to be detected and the second the minimum probabilistic threshold of detecting the fraction. The third column is the actual fraction of the targets detected averaged over twenty runs. The fourth column signifies the relative frequency of times a fraction greater than or equal to the desired fraction was detected. The fourth column is then a means of evaluating whether the desired probabilistic threshold was obtained. If the desired fraction to be obtained is 0.6 and thirteen times out of twenty a fraction more than 0.6 was detected, the entry in the last column of the table is 0.65 that signifies the required performance was met. It is seen that the average obtained fraction is above the desired fraction whenever the minimum probabilistic threshold is high indicating that the desired fraction was detected in most of the runs as entailed by the threshold. The average fraction obtained is lesser than the desired fraction when the desired probabilistic threshold is low. This is indeed expected as a low desired threshold indicates that the sensor is entailed to detect the desired fraction of targets only in a few of those twenty runs. The relative frequencies in the fourth column also do not fall significantly below the desired minimum probabilistic threshold in any of the runs. That the relative frequency is within $5-10\%$ of the desired threshold in all the runs validates the

probability definition of equation (2) and the computation of apparent time in equation (1).

In simulations T_{esc} is computed as the average of the minimum and maximum a target is within the home area of a sensor. The minimum time is the time taken by the target to traverse half the edge of a cell. The maximum time is the time taken to traverse from midpoint of a side to the farthest opposite vertex of the cell. Similarly T_{κ} is the average of the minimum and maximum time taken by the sensor to result in its home area becoming completely unguarded. The details of computing T_{κ} are omitted here for brevity.

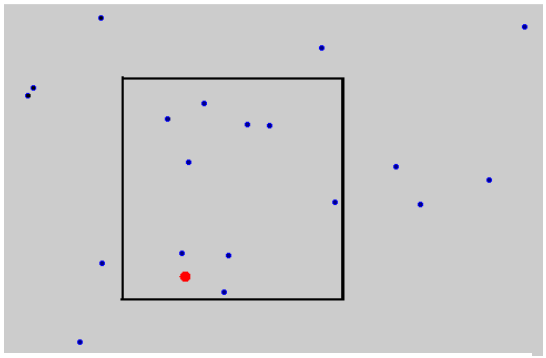

Figure 4: A single cell environment used for validating the definitions of equations (2) and (1). The bigger circle denotes the sensor and the smaller circles the crisscrossing targets introduced in poisson fashion.

Desired minimum fraction (f)	Desired minimum probabilistic threshold (Ω)	Obtained average fraction (20 runs)	Obtained relative frequency (20 runs)
0.9	0.9	0.91	0.95
0.9	0.1	0.23	0.05
0.6	0.9	0.73	0.85
0.6	0.1	0.38	0.2
0.4	0.6	0.42	0.65
0.4	0.4	0.32	0.3
0.1	0.9	0.28	1
0.1	0.1	0.08	0.2

Table 1: Tabulation of the results obtained for the environment of figure 4 for different desired fraction and threshold values.

The framework developed in section 2 is now tested for an environment shown in figure 1 with multiple sensors. Each sensor tracks targets such that the probabilistic guarantee is maintained with respect to its home area. The overall quality of track (*QoT)* at the end of a simulation interval Γ is defined by:

$$QoT = \frac{1}{N_\Gamma}\sum_{i=1}^{N_\Gamma}\frac{d_i(\Gamma)}{c_i(\Gamma)} \qquad (5)$$

Here N_Γ is the actual number of targets that crossed the surveillance zone in Γ. The numerator d_i indicates the number of sensors that detected the target i in Γ while the denominator, c_i denotes the number of sensor home areas that the target i had got past in Γ. Essentially QoT is an average measure on whether a target that crossed the home area of a sensor was detected by that sensor with the difference that the QoT would also reflect cases of targets that are detected by sensors whose home area it did not cross elsewhere in the surveillance zone. In the summation of (5) if the fraction $\frac{d_i}{c_i}$ exceeds unity it is clipped to unity. This is done since a fraction greater than unity tends to offset for fractions less than unity and does not reflect those cases. A stricter definition of QoT that specifically captures the number of targets missed by a sensor that crossed its home area is given by: $QoT = \frac{1}{N_S}\sum_{j=1}^{N_S}\frac{n_{dj}}{n_{cj}}$ (6).

Here N_S denotes the total number of sensors in the environment, n_{dj} denotes the number of targets detected by the sensor among those that had visited its home area H_j, while n_{cj} is the number of targets that had been through H_j. Thus $\frac{n_d}{n_c}$ represents the fraction of the targets that entered a sensor's H_j and were detected by it and can never exceed unity. However we found that at the end of the simulation interval the values as computed by (5) and (6) vary only marginally. Hence in this paper the results of QoT are those computed as in (5).

In the simulations that follow a sensor leaves its environment in pursuit of a target. Sensors can reallocate themselves to other targets during the course of a track as dictated by the resource

allocation strategy. The time t_j for a sensor is updated after every fixed number of samples based on the fraction of the targets that were detected thus far and the fraction that need to be detected in the remaining time window to meet the objective of (4). The algorithm operates through the steps given below.

Algorithm for Target Tracking with Probabilistic Guarantees:

1. Allocate sensor to a target based on resource allocation strategy discussed in [1].
2. After allocation decide if the target would be tracked or monitored with the sensor in stationary position based on the autonomy of the sensor [2] and decision making strategy presented in [1].
3. If step 2 results in a tracking decision compute t_j for the sensor based on equations (1) – (4) in section 2.
4. If the current cycle is the update cycle for a sensor that is in tracking mode perform steps 4a to 4c.
   - 4a. Compute the fraction of the targets detected so far, f_c.
   - 4b. Compute the fraction, f_r, of the targets that need to be detected in the remaining time $\Gamma - t_c$ in order to detect the desired fraction of targets f, where t_c is the time elapsed since the start of simulation.
   - 4c. Update time t_j for which the target can be away from H_j based on f_r.
5. Repeat steps 1 to 4 till the end of simulation time Γ is reached.

Figure 5 shows a snapshot of a simulation run. The bigger circles represent the sensors and the smaller circles the targets. The dashed rectangles enclosing the sensor and target identify the sensor-target pair (the target to which the sensor has allocated itself to). It is to be noted while a sensor tracks a target it also detects all other targets within its field of vision. Currently the problems of data association and target occlusion are not considered.

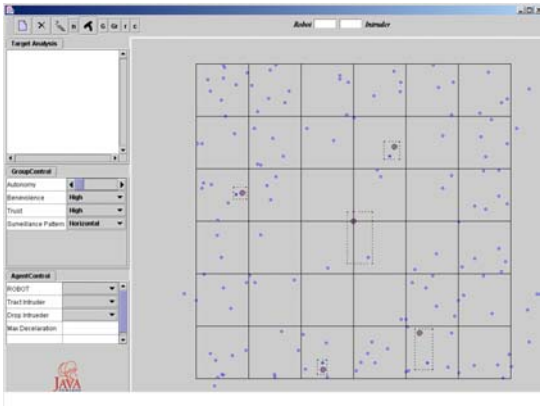

Figure 5: A snapshot of a simulation run. The dotted rectangles enclosing a sensor-target pair indicate the target to which that sensor is currently allocated to. Sensors are shown by larger circles while targets are depicted smaller

4.1 Analysis:

Figure 5a shows two graphs that plot f_r, the fraction remaining and wander time, t_j, along the y-axis. In both the graphs the abscissa denotes the time in samples. Sampling measurements on f_r, t_j are done once in every ten cycles of a simulation run. The total number of simulation cycles is 150 or in other words $\Gamma = 150$ in these simulations. Each cycle is repeated every 500ms. The graphs cover the entire simulation run of 150 cycles or 15 samples of measurements. The plot of figure 5b depicts QoT on y-axis and sample time on x. Both graphs 5a and 5b are for a simulation run with parameter $f = 0.8, \Omega = 0.75$. The graphs of figure 5a are for one of the sensors of figure 5 only the graph of 5b depicts QoT of the entire system. Graphs in figure 5c and 5d have the same connotations as in figures 5a and 5b except that they are for parameters $f = 0.8, \Omega = 0.3$. The horizontal dashed line in the graph of 5b and 5d indicate the desired fraction of target fraction of targets, f, expected to be detected at the end of the simulation. Since QoT as defined in (6) computes the fraction of targets detected averaged across all the sensors in the system, the horizontal line serves as an indicator if the QoT was achieved or not.

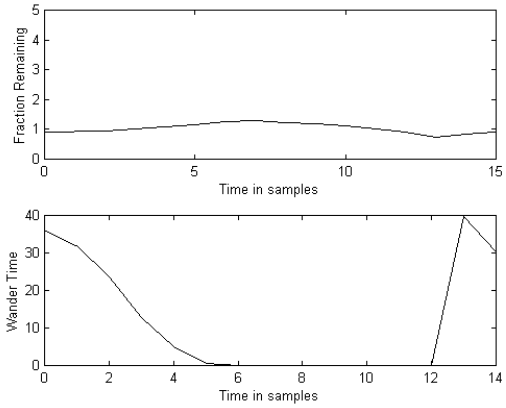

Figure 5a: The top graph shows f_r plotted against sample time while the bottom graph is a plot of t_j against samples. The plots are for a simulation such as in figure 5 run with $f = 0.8, \Omega = 0.75$

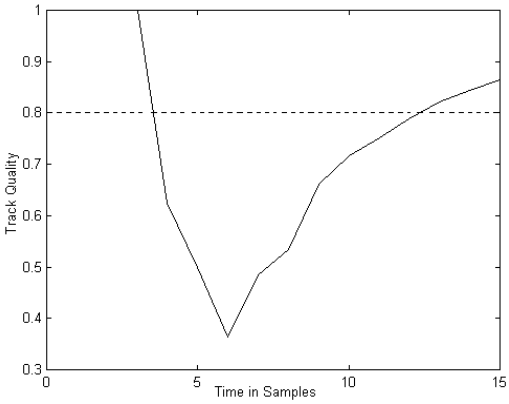

Figure 5b: A plot of track quality QoT . The dashed horizontal line denotes the desired fraction of the total targets that need to be detected.

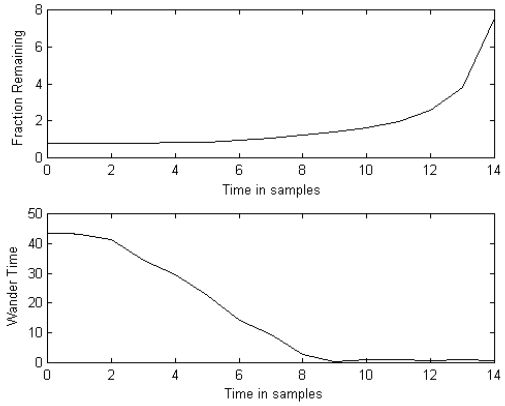

Figure 5c: Graph same as figure 5a for parameters $f = 0.8, \Omega = 0.3$

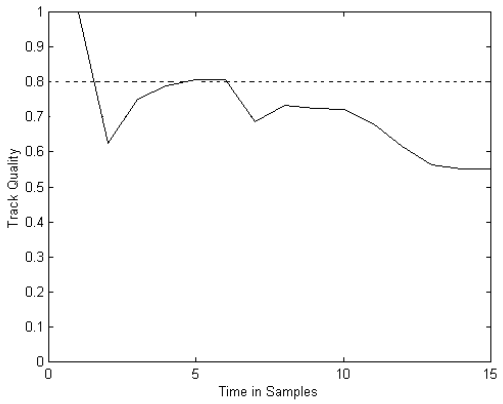

Figure 5d: Graph with same connotations as figure 5b for parameters $f = 0.8, \Omega = 0.3$. At the end of simulation time the track quality is below the horizontal line indicating that performance criterion was not met

For $\Omega = 0.75$ the track time is modulated such that the desired fraction f averaged over all sensors is detected at the end of a simulation run for majority of such runs. Figure 5b corresponds to one such run where the track quality at the end of the simulation is 0.85 and is above the expected criterion of 0.8 and lies above the horizontal. Figure 5d corresponds where the QoT at the end of the simulation does not achieve the desired fraction. This is expected for a run with $\Omega = 0.3$ where most of the runs are not required to detect a fraction greater than 0.8.

For a marginal increase in f_r in the top plot of figure 5a the corresponding decrease in t_j is steeper in figure 5a when compared with figure 5c. The decrease in wander time t_j as f_r is less steep in 5c than in figure 5a. For a given f the variations in t_j are due to the variations in Ω . A higher Ω entails that the sensor cannot move too far away from its home due to lower values of t_j . As the sensor moves away from its home and misses targets the required remaining fraction to be detected f_r may tend to increase. In such a case the decrease in wander time also tends to be steeper for a similar increase in f_r for a higher Ω . A steeper increase translates as quicker returns to home by the sensor to detect more targets.

6 Extension to Multi-sensor Surveillance

The benefits of having more than one sensor guard the same home area is now considered. For example let each of the home area in figure 1 (shown by thick squares) be guarded by 3 sensors. One such area is shown in figure 6.

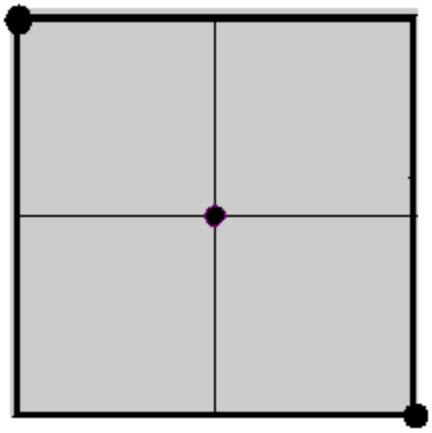

Figure: An area guarded by three sensors, one at the center and two at the corners

It can be shown that T_κ, the time taken by the sensor to leave the area completely unguarded is the same on an average for all the three sensors. Let the sensors be labeled as A, B and C. The probability that sensor A detects a target is denoted as p_A and has the same form as (2). Similarly the individual probabilities of detection for sensors B and C are denoted as p_B and p_C. Since T_κ and hence T_{app} are the same for the three sensors $p_A = p_B = p_C$. The probability that at-least one of the sensors detect a target is given by $p = p_A \cup p_B \cup p_C$. Determining p from equations (3) and (4) and along with the condition $p_A = p_B = p_C$ leads to the following cubic in the individual probability of a sensor:

$$p_A^3 - 3p_A^2 + 3p_A - p = 0 \qquad (7).$$

The solution to the cubic solves the individual probability of a sensor detecting a target.

Figure 7a shows two graphs. The top graph depicts the individual probability of a sensor detecting a target for a fixed Ω (here 0.7) and varying f. The lower graph plots t_j versus f for a constant Ω (here 0.7). Each of these graphs shows two plots. The plot with a dashed line corresponds to the case where a single sensor guards an area. The plot with solid line corresponds to the case where multiple sensors guard an area. The graphs indicate that the individual probability of detection is consistently lower when multiple sensors guard an area than a single sensor guarding and the wander time is correspondingly high for a multi-sensor case. Hence a sensor can wander away from its home area for a longer duration when if there are more than one guarding its home area. In [2] we have shown longer tracking time enhances performance criteria based on median and mean number of detections of targets.

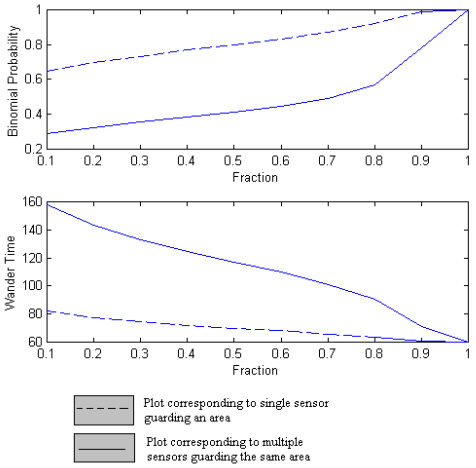

Figure 7a: Plots of individual probability and wander time against varying values of desired fraction f. Solid lines are plots corresponding to a single sensor while dashed lines correspond to multi sensor case

7 Conclusions:

A framework that provides for probabilistic guarantees for a multi-sensor based multi-target tracking system is provided here. Sensors modulate their tracking time based on the desired fraction of targets that need to be detected and the minimum probabilistic threshold of detection. Simulation results corroborate the efficacy of the formulation of the scheme for probabilistic guarantees. Extension of the scheme to multiple sensors guarding the same area enables longer tracking time for a sensor and hence better performance based on mean and median number of detections.

References:

[1] K Madhava Krishna, H Hexmoor and S Pasupuleti, "A Surveillance System Based on Multiple Mobile Sensors", *to appear in Proceedings of FLAIRS,* 2004, Special Track in Sensor Fusion.

[2] K Madhava Krishna, H Hexmoor and S Pasupuleti, "Role of Autonomy in a Distributed Sensor Network", *to appear in Proceedings of ICAI,* 2004.

[3] D Schulz; W Burgard, D Fox and A Cremers, "Tracking multiple moving targets with a mobile robot using particle filters and statistical data association", *IEEE International Conference on Robotics and Automation*, 1165-1170, 2001

[4] D Schulz and W Burgard, "Probabilistic state estimation of dynamic objects with a moving mobile robot", *Robotics and Autonomous Systems,* 2001.

[5] B. Kluge, C Kohler and E Prassler, "Fast and robust tracking of multiple objects through a laser range finder", *IEEE International Conference on Robotics and Automation*, 1165-1170, 2001

[6] K Madhava Krishna and P K Kalra, "When does the robot perceive a dynamic object", *Journal of Robotic Systems*, 19(2), 2002

[7] K Madhava Krishna and P K Kalra, "Detection tracking and avoidance of multiple dynamic objects", *Journal of Intelligent and Robotic Systems*, 33(4): 371-408, 2002

[8] L Parker, "Cooperative robotics for multi-target observation", *Intelligent Automation and Soft Computing,* 5[1]:5-19, 1999

[9] B Jung and G.S. Sukhatme, "Multi-Target Tracking using a Mobile Sensor Network", Proc., *IEEE Intl. Conf. On Robotics and Automation,* 2002

[10] B Horling, R Vincent, R Miller, J Shen, R Becker, K Rawlins, and V Lesser, "Distributed Sensor Network for Real Time Tracking", In *Proceedings of the 5th International Conference on Autonomous Agents*:. 417-424, 2001.

[11] B Horling, R Miller, M Sims, and V Lesser, "Using and Maintaining Organization in a Large-Scale Distributed Sensor Network", In *Proceedings of the Workshop on Autonomy, Delegation, and Control, (AAMAS 2003)*.

[12] P Vincent and I Rubin, "A Framework and Analysis for Cooperative Search Using UAV Swarms", *Proceedings of SAC'04*, March 14-17, Nicosia, Cyprus