Role of Autonomy in a Distributed Sensor Network for Surveillance

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: This paper studies the effect of endowing the social parameter, autonomy, on a distributed sensor network. The sensor network is used for surveillance of multiple targets that crisscross a rectangular surveillance zone. Crossing targets are ascribed priorities by a priority ascription scheme based on fuzzy inference methods. The scheme fuses local and global priorities of a target through autonomy modeled as a parameter that take values in $[0,1]$. Sensors coordinate to allocate themselves to a particular target. Autonomy affects the manner in which sensors allocate themselves to a target. Lower autonomy biases a sensor to allocate itself to a target that need not be tracked or tracked for shorter duration. Higher autonomy biases a sensor to allocate itself to a target that needs to be tracked for longer duration. Test cases are presented regarding how changes in autonomy affect the tracking performance of the system. Conclusions are derived that suggest when high or low values of autonomy are beneficial to the system.

1. Introduction

This paper is devoted to study and characterize the effects of autonomy on the performance of a multi-sensor based surveillance system. In an earlier effort of ours [1] we had presented the methodology for coordination between sensors and a set of rules that characterize the resource allocation scheme. Targets crossing the surveillance area are ascribed priorities, which is a fusion of global and local priorities. The global priority for a target is the priority from the point of view of the entire system of sensors. Each robot ascertains its own preference for a target, which is the local priority for that target from the point of view of that sensor. A weighted combination of global and local priorities is used to compute a balanced priority for a target from the point of every sensor. Thus, each sensor maintains a list of balanced priorities for every target. The weighted combination is achieved through the parameter, termed autonomy bias and denoted by w_{ab}. Higher values of w_{ab} indicate high autonomy for the sensor and hence the balanced priority for the target reflects the local priority of the sensor with respect to that target. Lower values of w_{ab} denote low autonomy for the sensor and the balanced priority for the target is reflective of the global preference of preference from the point of view of entire system. The performance of the system for the boundary values of the autonomy bias (zero and one values) is presented and conclusions derived.

The authors believe that the paper is one of the first of its kind that attempts to modify and control the behavior of the entire collective of agents (here sensors) through social notions such as autonomy. In the distributed sensor network approach of Victor Lesser's group [2, Lesser Autonomy] where the RSTA architecture and SPAM negotiation protocol were developed autonomy is characterized as the agent's freedom to decide what course of action to perform and to detect and resolve conflicts overriding the schedule formulated by the manager of that sector.

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.

Within this 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 method presented does not address collaborative or shared reasoning strategies for decision-making and action selection such as the decision for moving to a new area. The coordination between sensors is restricted to communicating their respective positions. Strategies presented by Lesser's group deal with sensor coordination from the point of view of tracking only one target.

3 Problem Description

Multi-sensor surveillance can be found in many military applications such as border patrol, beachfront surveillance and reconnaissance of secured rural areas and cities. Surveillance in military domain often involves rugged and uneven terrain over large areas with possible natural and manmade obstructions. In the formulation presented we make certain abstractions while transforming the real world situation to a simulated environment in that it does not have representations of the actual features of the landscape. The main thrust of this paper, deals with autonomy issues in a multi-sensor system is essentially independent of the approximations of the real world made in simulation models.

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 sensors indicated by circles are placed along the diagonals of the zone at the corners of the cell.

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 covering area of that sensor for the remaining of this paper. In other words a sensor positioned at 'a' in figure 1 considers only targets within the square region 'adce' as those within its field of vision. The four sides of the zone define its four boundaries. Boundaries serve as entry and exit points for the moving targets. Targets move across the area by entering from one boundary and leaving through any of the other three boundaries. Targets do not enter and exit via the same boundaries. Sensors model the path of the target as linear. Targets can change their directions but at any instant sensors model target trajectories as straight line paths along their current motion direction.

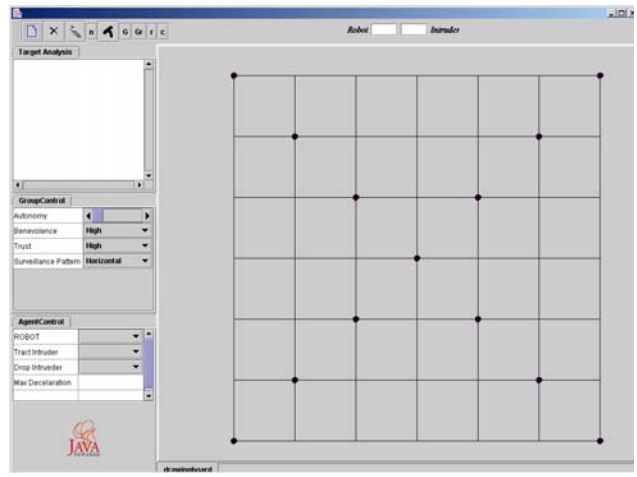

Figure 1. The rectangular surveillance zone with the sensors shown as circles placed along the two zone diagonals at the corners of the cells. The sensors have a radius of vision equal to the diagonal of the cell

4 The Methodology

The following notations may be useful before embarking on the discussion for ascribing priorities to targets. Let n_s represent the total number of sensors and n_t the total number of targets in the system. Let N_S be the set of all sensors in the system, i.e., $N_S = \{s0, s1, ..., sn_s\}$, where si denotes the sensor with label i. Hence n_s is the cardinality of N_S or $n_s = |N_S|$. In the same vein N_T is the set of all targets, $N_T = \{t0, t1, ..., tn_t\}$ and n_t is the cardinality of N_T. We define S_{ti} as the set of all sensors currently monitoring target ti and T_{si} as the set of all targets being monitored by sensor si. Then $\overline{S}_{ti}$ is the set of all sensors currently not detecting ti. We denote the number of samples a target ti is likely to be detected or observed by sensor sj by o_{tisj}, where $sj \in S_{ti}$. If the time for which ti is likely to be within the field of vision of sj is t_{tisj} and the sampling interval is dt then $o_{tisj} = t_{tisj} / dt$. The sampling time indicates the time interval between two successive scans of the environment by the sensors. SF_{ti} is the set of all sensors that are *likely*

to detect ti in the future though they are not observing it currently. Hence $SF_{ti} \subset \overline{S}_{ti}$.

4.1 Priority ascription to targets

Sensors reason about targets by ascribing priorities to them. Priorities serve as handles that aid in decision making. Whenever a sensor detects a target it updates information about the target regarding its current position, velocity and motion direction to a whiteboard. The whiteboard is a common data store of resources that includes public methods and variables, which can be modified and accessed by other programs of the project. Other sensors can come to know about this target by accessing these variable or invoking methods from this common pool.

Every target is given a global priority that portrays the priority to that target from the point of view of the entire sensing apparatus.

Ascribing global priority:
The global priority for a target ti is determined by three parameters namely:

- $p1_{ti}$: The maximum number of times target ti is likely to be further observed by one of the sensors currently monitoring it. In other words $p1_{ti} = \max_{sj}(o_{tisj}) \forall sj \in S_{ti}$
- $p2_{ti}$: The number of sensors currently not detecting ti but are expected to detect it in future. $p2_{ti} = |SF_{ti}|$
- $p3_{ti}$: The measure of possibility that sensors in SF_{ti} would be in a position to monitor ti in the future. This is elaborated in the subsequent section on sensor coordination. $p3_{ti} = \sum_{sj} m_{tisj} \forall s_j \in SF_{ti}$, where m_{tisj} is the measure of the possibility of sj to take care of ti

The fuzzy rulebase that infers the global priority gp_{ti} for any target in the system is tabulated in table 1. For notational convenience we remove the suffix ti associated with the parameters henceforth.

The membership functions associated with the antecedent and consequent variables are not shown here due to brevity of space and since they are not what that constitute the main thrust of this effort.

p1	p2	p3	gp
H	L	X	M
H	M	X	LM
H	H	X	L
L	L	X	H
L	M	X	HM
L	H	X	M
H	X	H	L
H	X	L	LM
L	X	H	M
L	X	L	H

Table 1. Fuzzy rulebase for global priority inference. The linguistic labels for the fuzzy sets are: H=high, L=low, M=medium. LM=low medium and HM=high medium

Ascribing local priority:
Every target ti is associated a local priority from the point of view of every sensor si in N_S. The parameteric basis for local priority computations vary marginally with regard to whether si belongs to S_{ti} or not. If $si \in S_{ti}$ the local priority is based on the time for which si would have to track S_{ti} before another sensor engages ti. If $si \notin S_{ti}$ the computation is based on the time for which si would have to wait for ti before si can engage ti. The local priority for a target ti from the point of view of a sensor sj is denoted as lp_{sj}. Denoting either of the times as t_{wait} the rulebase for computing local priority is tabulated in table 2 where the symbols carry the same linguistic labels as in table 1.

t_{wait}	lp_{sj}
L	H
M	M
H	L

Ascribing balanced priority:

The balanced priority for ti from the point of view of sj is obtained by fusing local and global priorities as follows:
$bp_{sj} = w_{g'}(gp) + (1 - w_{g'})lp_{sj}$, where $w_{g'} = w_g(1 - w_{ab})$. Here w_g takes unitary value if $sj \in S_{ti}$, else its value decreases linearly with the time

taken by ti to enter the field of vision of sj. w_{ab} represents the autonomy bias of a sensor towards its own preference for the target [local priority] rather than the preference as ascribed to the target by the entire system [global priority]. Under high values of w_{ab} the balanced priority would reflect the individual sensor's preference for the target more than the global preference.

5 The Role of Autonomy

The effect of autonomy on the sensor system is best portrayed through the cause effect diagram of figure 2.

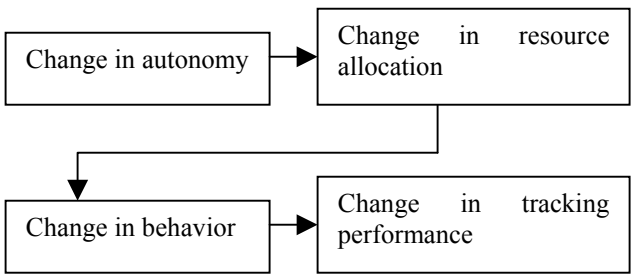

Figure 2: The cause effect chain diagram depicts how the effects of autonomy propagate down the chain

The most immediate effect autonomy changes have is in affecting the resource allocation process, the manner in which sensors allocate themselves to targets. Since resource allocation also depends on several other factors characterized through a set of rules in [1], similar changes in autonomy need not necessarily produce similar changes in resource allocation for different scenarios. Nonetheless the nearest change due to autonomy is in the manner of resource allocation. Similarly a change in sensor behavior is contingent on other parameters apart from changes in resource allocation. Hence the effect of autonomy on tracking performance is a trickle down effect, nonetheless as elaborated in next section is a useful social parameter to control the behavior of the group (and hence its performance) in a predictable and repeatable manner for a class of scenarios.

The effect of autonomy in changing resource allocation is illustrated through figures 3a and 3b. In figure 3a sensor $s2$ allocates itself to target $t1$, with autonomy bias, $w_{ab} = 0$. The balanced priority for $t1$ in this case is 0.76, which essentially is also its global priority. The role of $s2$'s own preference has little say here since $w_{ab} = 0$. The balanced priority for $t0$ in such a case is 0.4 from the point of view of $s2$ and hence $s2$ allocates itself to the target with the higher priority. However in figure 3b when sensors have high autonomy, $w_{ab} = 1$ the balanced priority of $t0$ rises to 0.64 while that of $t1$ slips to 0.46 with respect to $s2$ and hence $s2$ allocates itself to $t0$ instead of $t1$.

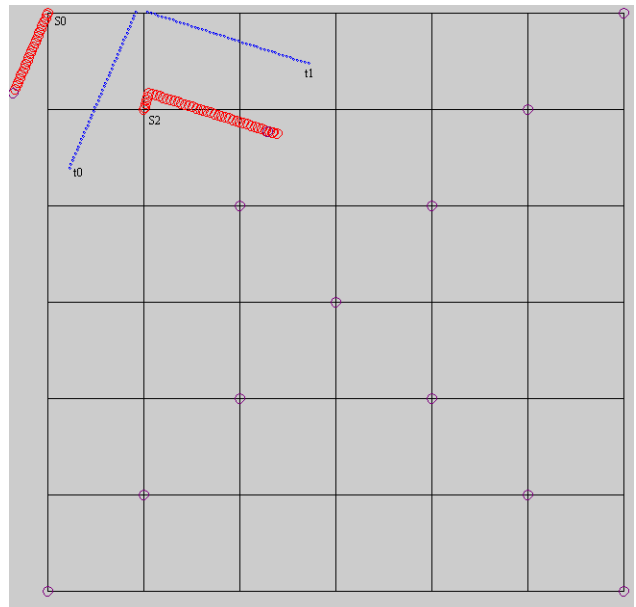

Figure 3a: With zero autonomy sensor S2 alloctaes itself to target t1in mobile tracking mode

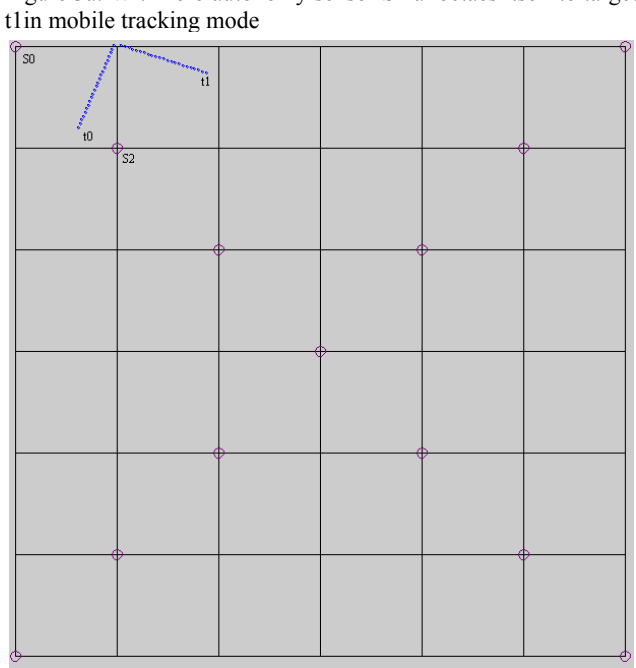

Figure 3b: With autonomy 1, sensor S2 allocates itself to t0 under stationary tracking mode.

The changes in resource allocation in figures 3a and 3b also cause change in behavior of the sensor $s2$. Under low autonomy when $s2$ allocates itself to $t1$ it does so by following $t1$ since $t1$ would not have been tracked sufficient number of times had $s2$ remained stationary. In the latter case of high autonomy $s2$ allocates itself to $t0$ but does not follow it. This was expected since a sensor's local preference is guided by the amount of time it would be away from its home position. Longer a target requires a sensor to be away from the home position, lesser is the sensor's own preference for that target. And since $t1$ entails $s2$ to be away longer under high autonomy resource allocation tends to reflect the local priority with $s2$ allocating itself to $t0$ that does not entail $s2$ tracking $t0$ for a long time. In this particular case the tracking quality for the figure 3a is better since both targets are tracked for a longer duration, whereas in figure 3b since $s2$ is stationary $t1$ is not tracked for the same time length as in figure 3a. However it can be seen intuitively that since $s2$ is stationary its better situated to detect targets that enter its local area of surveillance in the imminent future. The subsequent section discusses this tradeoff in tracking performance vis-à-vis low and high values of autonomy.

6 Evaluating Tradeoff due to Autonomy

The following parameters are used to measure the performance of the system.

- Mean Tracking Quality $MTQ = \sum_i \frac{d(ti)}{N}$, where, $d(ti)$ is the number of detections of the target ti and N is a normalization constant. N is the minimum number of times a target needs to be detected for a sufficient characterization of it. N is fixed as 50 in all the examples discussed in this paper. MTQ can have a value greater than unity and higher the tracking quality the performance is considered better
- Median Tracking Quality, $MdTQ$, the median value of the tracking quality is obtained by sorting $\frac{d(ti)}{N}$ and picking or averaging the middle value(s).
- Total expected number of misses, $TEM = \sum EM(sj)$. Here $EM(sj)$ represents the expected number of targets that will go undetected by sj during the time sj is in pursuit of some other target. Lower the value of TEM, better is the tracking performance.

It is evident that MTQ and TEM would conflict with each other since higher the value of MTQ higher would also be the value of TEM and hence indicate a poorer tracking performance though from the point of view of MTQ the same performance would be considered better. The computation of MTQ is not straightforward. Targets are assumed to come at rate λ in the space occupied between any two of the vertices formed by the meeting point of the rows and columns on the boundary. For a surveillance zone such as in figure 3b consisting of seven rows and columns targets enter at rate λ at six of those spaces formed between the points of intersection of the rows and columns on the boundary. Hence from each boundary of the rectangle 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 arrives at the midpoint of the region between the intersection of rows and columns along the perimeter of the surveillance zone. In the figure below (figure 4) 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 agents 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.

Let T be the time for which a sensor sj is away from its home position in pursuit of a target. 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:

$T_a = 2\int_0^{T_f} f.dt + \left(T - T_f\right)$, where, f is the fraction of the original area left unguarded by sj as it moves away from its home. The upper limit of the integral T_f denotes the time at which the sensor leaves its original area completely unguarded. If T_{esc} represents the average time for which a sensor should be away such that a target can get past it original covering area completely undetected, the expected number of targets that would be missed by the sensor is given by the approximation $\lambda_{SJ}\dfrac{T_a}{T_{esc}}$. In other words the final expression states if λ_{SJ} is the average rate of entry within the original covering area of sj and if sj is away from the point of view of the entering target for a time T_{esc} (i.e., $T_a = T_{esc}$) then λ_{SJ} number of targets can be expected to go undetected on an average. T_{esc} is fixed as the time taken by a target to move from one vertex to its opposite vertex along the main diagonal of the covering area.

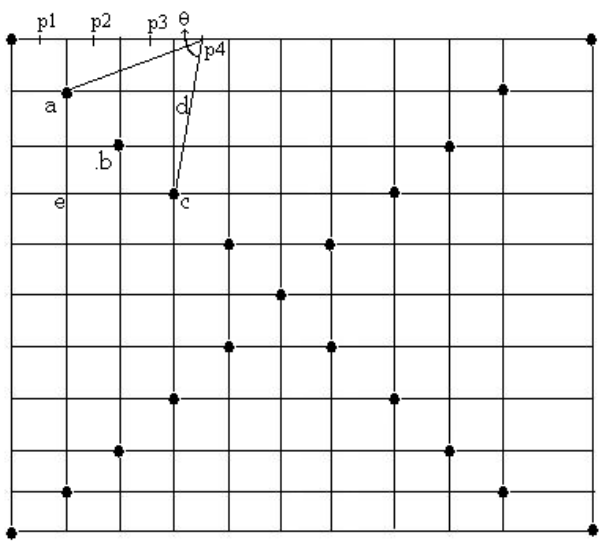

Figure 4: Targets are modeled as entering at locations p1, p2, …, as shown. These are the midpoints of the space between the meeting points of the columns and rows along the perimeter

Table 3 summarizes the results of a list of simulations. Each simulation was characteristic of a particular modality of target entry and was repeated for unity and zero values of autonomy. For example experiment 1a mentioned in table 3 involved targets entering in a manner shown in figure 3a with zero autonomy while experiment 1b involved the same simulation with unity autonomy as shown in figure 3b.

Experiment Number	w_{ab}	MTQ	$MdTQ$	TEM
1a	0	1.46	1.46	0.73
1b	1	1.32	1.32	0.0
2a	0	1.18	1.03	2.9
2b	1	0.85	0.82	0.0
3a	0	1.44	1.25	9
3b	1	1.13	1.08	0
4a	0	1.89	1.64	3..7
4b	1	1.65	1.63	1.4
5a	0	1.47	1.44	1.1
5b	1	1.38	1.39	0.9

Table 3: Summary of performance measures obtained for different simulation experiments

The table consistently reports higher mean tracking quality and higher expected number of misses when autonomy bias is zero and lower mean tracking quality and lower number of missed targets for all experiments but for the last one. The last experiment reports more or less similar values for both high and low values of w_{ab}. The reason for this difference is that while the first four experiments had targets introduced in a manner that facilitated change in resource allocation between high and low autonomy values the last experiment however could not facilitate such a change in autonomy. The last experiment involved introduction of ten targets from left boundary of the surveillance zone to the top as shown in figure 5. In this particular case a sensor did not find a particular target more to its preference under high autonomy than the others among the set of targets it monitored at any instant. The resource allocation proceeded on similar lines for both high and low autonomy values. Hence the tracking behavior and the performance metrics did not show appreciable changes. The table also portrays that the Median Tracking Quality, $MdTQ$, is lesser when compared with the mean, MTQ, in the first four experiments with zero w_{ab}. This indicates that while certain targets are tracked for longer number of samples others are not tracked for much lesser duration. This is expected since under low w_{ab} sensors tend to pursue certain targets for longer time than others. However for experiments with unity autonomy the median and mean tracking have negligible difference indicating a

more equally spread tracking of the targets. The median value for high autonomy experiments continues to be lower than that of the low autonomy as was the case with the mean values. Targets introduced with temporal delays with some of them along similar directions as their predecessors give the highest number of expected misses such as in experiment 3a. Targets entering along similar or parallel directions with time delays tend to keep sensors away from the home position for very long durations. Sensors that had pursued earlier targets detect the new targets at locations away from their home and start pursuing them thus keeping themselves away from the home for long durations. The simulation graph of experiment 3a depicted in figure 6 shows both sensors 0 and 2 moved far away from their home due to targets entering in similar directions at regular intervals. In such a scenario sensors 0 and 2 are prone to miss a lot of other targets that crisscross their original covering area. In figure 6 sensors 0 and 2 would both end up missing completely targets marked 7, 8 and 9.

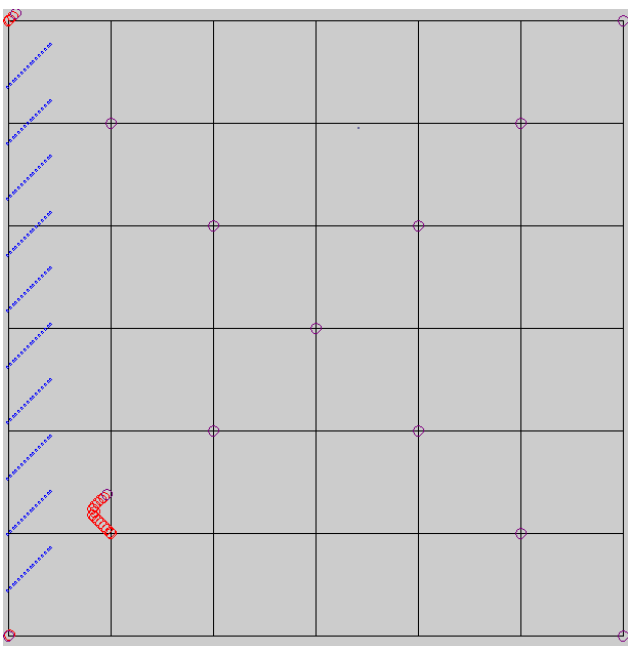

Figure 5: Targets introduced in manner such as above from left to top do not result in change in the manner of resource allocation between high and low values of autonomy. Hence there is not much change in the performance metrics between experiments 4a and 4b in table 3

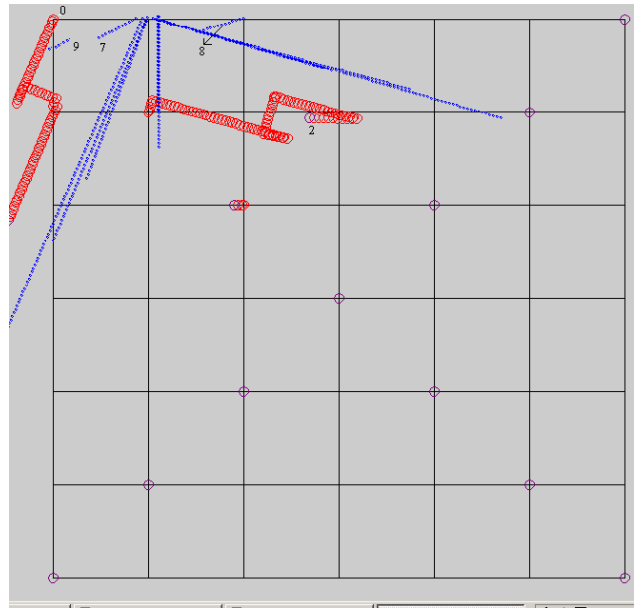

Figure 6: The simulation graph of experiment 3a shows targets pushed far enough such that targets 7, 8 and 9 can go completely undetected

It is to be noted that while for the purpose of computation the expected number of missed targets, targets are modeled as entering in poisson fashion, the simulation graphs shown do not depict the entry of targets in such a fashion. Targets are entered initially by the user through the GUI developed on Borland JBuilder's IDE for Java and further introduction of targets are automated through a random generation such as in figure 6.

7 Conclusions

The role of embedding sensors with autonomy in a multi-sensor based surveillance system has been analyzed in this paper. Autonomy is defined as the preference of the individual sensor in tracking a target vis-à-vis tracking the target by following the global priority order. For a class of environments where change in autonomy affects resource allocation it is observed zero values of autonomy allow for higher number of tracks of targets and hence a higher mean tracking quality when compared with unity autonomy values. However the expected number of targets to be missed is also high for zero autonomy values when compared with high autonomy values. The median tracking quality is noticeably lower than the mean tracking quality for systems with low autonomy indicating that certain targets are tracked for a longer time and others for much lesser time. Thus tuning autonomy values provides for a social order control for large group of

agents (sensors) in such a system as described in this paper.

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] B. Horling, R. Mailler, J. Shen, R. Vincent, and V. Lesser, "Using Autonomy, Organizational Design and Negotiation in a Distributed Sensor Network". In *Distributed Sensor Networks: A multiagent perspective*, pp. 139-183. 2003. Book chapter.

[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)*.