

Multiple Agent Search of an Unknown Environment Using Game Theoretical Models

P.B. Sujit

Department of Aerospace Engineering
Indian Institute of Science
Bangalore 560 012, India
sujit@aero.iisc.ernet.in

D. Ghose

Department of Aerospace Engineering
Indian Institute of Science
Bangalore 560 012, India
dghose@aero.iisc.ernet.in

Abstract—This paper addresses the problem of obtaining optimal strategies for searching an unknown environment given in the form of an uncertainty map. Several strategies, in the form of variable length look-ahead policies, that depend on the level of communication between searchers, are proposed based on Nash equilibrium, security, and cooperative notions in game theory. Simulations are carried out for two searchers on a planar uncertainty map and the performance results are compared with respect to the type of strategies and the length of the look-ahead policies. These simulations show that longer look-ahead policies do not yield better performance than shorter ones, but need high computational effort. The results also show that although communication plays a major role, the performance of Nash and security strategies, that do not depend on communication between searchers, are comparable with the full-information centralized cooperative case.

I. INTRODUCTION

Searching for targets in an unknown environment has been a topic of intense research activity in the search theory literature for several decades [1]. Recent efforts to improve the efficacy of search and surveillance operations using Unmanned Aerial Vehicles (UAV's) has brought this area of work to the forefront. The model that is normally used for these applications is based on discretization of the search space and time, where the search region is represented as a collection of cells, and the dynamics of the aerial vehicles are suppressed [2]. The *a priori* knowledge of the location of the target is represented as an uncertainty map which could be interpreted as a function of a probability distribution on the cells constituting the search space. The uncertainty values associated with each cell reduces as these cells are visited by a UAV. Hence, the objective of UAVs would be to decide on a route so as to maximize the reduction in uncertainty as it searches in the unknown region. In this paper, we propose three search strategies based on notions in game theory in order to reduce uncertainty in a search space. We consider two UAVs in the search space and model the strategies used by the UAV's to decide their future actions. We model the problem as a non-zero sum game and explore noncooperative Nash strategies, security strategies, and cooperative strategies. Simulation results are presented to demonstrate the performance of these strategies and also the effect of increasing the length of the finite horizon look-ahead policy.

Some relevant papers that address problems in the same

domain are [2]-[13]. In our previous work [14], we developed an algorithm for uncertainty reduction in an unknown region using the *k*-shortest path algorithm. The objective of the UAVs was to maximize the uncertainty reduction per sortie, under the practical constraint that the UAVs have limited fuel and hence can travel limited depth of the search area, as they have to return back to base for refuelling.

Here, we formulate the problem of uncertainty reduction by using multiple searchers and obtain strategies for search route determination using a game theoretical framework. Depending on the level of information and communication between searchers, we propose three strategies and implement them in the form of a finite steps look ahead policies.

II. PROBLEM FORMULATION

A. Discretization of the Search Space

The search space is partitioned into a collection of identical regular hexagonal cells. We use hexagons as the basic unit since it offers the flexibility to the searcher to move in six uniformly distributed directions at each time step and reach a neighboring cell while expending the same amount of energy (see Figure 1).

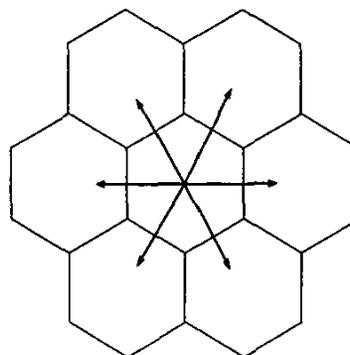


Fig. 1. Partitioning the search space into hexagonal cells

B. Uncertainty Map

The uncertainty map constitutes of real numbers between 0 and 1 associated with each cell in the search space. These numbers represent the undetected mass in that cell, or it represents the uncertainty with which the location of the

target is known in that cell. A probabilistic interpretation of the uncertainty map would be as follows: An uncertainty value of 0.6 would imply that any statement about the target's location in cell C_i (say) would be true only with probability 0.4. An uncertainty value of 0 would imply that everything is known about the cell (that is, one can say with certainty whether a target is located in that cell or not). On the other hand, an uncertainty value of 1 would imply that nothing can be said about the location of a target in that cell. Hence, a successful search operation is one that visits those cells that have large uncertainty values. Once a cell C_i is visited by a UAV at time t , its uncertainty value U_i reduces to $U_i(t+1) = U_i(t)(1 - \beta)$, where $\beta \in [0, 1)$ is the uncertainty reduction factor associated with each visit.

C. Searcher Objectives and Constraints

The search space is composed of identical sized cells. The energy spent by the UAV in moving from one cell to another is equivalent to one unit step length. The objective of the UAVs is to select search routes that maximizes uncertainty reduction. In every time step, the UAV can move from one cell to the neighboring cell. We also consider the possibility that the searcher may devote more search effort in a single cell, in terms of multiple visits, because of higher uncertainty value there. Each UAV is equipped with sensors that it uses to collect data or information about a cell. So, a UAV that spends a certain number of steps (one or more) in searching any given cell is essentially spending this time in collecting data about the cell and thus the uncertainty in that cell reduces as a function of the time spent there.

We assume that at any given time t the UAV's know the position and route (upto time (t)) of the other UAV's and also the updated uncertainty map. So, at time t , each searcher's objective is to determine its future route. The problem can be looked upon as a centralized one if we assume the searchers to communicate with each other and decide upon a globally beneficial decision. This would be the cooperative solution. In the absence of any such communications each searcher has to decide its next search route in different ways. We will explore decision strategies using a game theoretical model.

D. A Game Theoretical Model

We use q -step look ahead planning [2], where q determines the depth of the exploratory search environment to obtain optimal strategies. In this paper we develop the model for two agents. We assume that agent A_1 knows the current location of agent A_2 , and vice versa. Each UAV starts with an initial uncertainty map and updates it after every time step using information about the route taken by the other agents till that time. So, at any given time the agents have the same uncertainty map and also know the past route and present location of the other agents. In this sense the model is that of a perfect information one. The objective of the agents is to select their next action or path at time t in order to maximize their benefits (maximize

uncertainty reduction). This problem can be modelled as a two person non-zero sum game with each agent as a player and the set of paths available to it as the set of strategies. The payoff to an agent can be expressed as a search effectiveness matrix.

Computing the Search Effectiveness Matrix: Let a cell's uncertainty value be U_i . Let $\mathcal{P}_i^q(C_{s_i})$, $i \in \{1, 2\}$ be the set of all possible paths of length q for agent A_i , emanating from cell C_{s_i} . A path $P_i^j(C_{s_i}) \in \mathcal{P}_i^q(C_{s_i})$, $j = 1, 2, \dots, |\mathcal{P}_i^q(C_{s_i})|$ is a sequence of cells

$$P_i^j(C_{s_i}) = [C^1, C^2, C^3, \dots, C^q]$$

where $C^k \in \mathcal{C}$ (\mathcal{C} is the collection of all cells), $C^1 = C_{s_i}$ which is the current position of A_i and $C^{k+1} \in \mathcal{N}(C^k)$, where $\mathcal{N}(C^k)$ is the set of all neighboring cells of C^k .

Let the uncertainty value of cell C^k at time t be $U(C^k, t)$. Given a path $P_i^j(C_{s_i})$ of agent A_i , suppose A_i is at cell C^l at time t then the reduction of uncertainty associated with C^l , and the subsequent updated value of uncertainty, for both the agents is evaluated as:

Case 1: Only A_i is at cell C^l at time t , then

$$v_i(t) = U(C^l, t)\beta_i \\ U(C^l, t+1) = U(C^l, t) - v_i(t)$$

Case 2: A_1 and A_2 are both at cell C^l , then

$$v_i(t) = \frac{\beta_i}{\beta_1 + \beta_2} U(C^l, t)[1 - (1 - \beta_1)(1 - \beta_2)] \\ U(C^l, t+1) = U(C^l, t) - (v_1(t) + v_2(t))$$

So, given two routes $P_1^j(C_{s_1})$ and $P_2^j(C_{s_2})$ of the two agents, the reduction in uncertainty achieved by A_i at each step t ($t = 1, 2, \dots, q$) is given by $v_i(t)$ and is computed using Case 1 or Case 2 as the case may be. Note that this computation has to be done simultaneously for both agents. The total benefit to A_i due to path $P_i^j(C_{s_i})$ is

$$V(P_i^j(C_{s_i})) = \sum_{t=1}^q v_i(t)$$

The search effectiveness matrix M^i has dimension $|\mathcal{P}_1^q(C_{s_1})| \times |\mathcal{P}_2^q(C_{s_2})|$ and every element $m_{(k,t)}^i$ of the matrix represents the payoff ($V(P_i^j(C_{s_i}))$) obtained by the agents, when A_1 chooses the path $P_1^k(C_{s_1}) \in \mathcal{P}_1^q(C_{s_1})$ and A_2 selects $P_2^l(C_{s_2}) \in \mathcal{P}_2^q(C_{s_2})$.

III. SOLUTION METHODS

Every agent computes the search effectiveness matrix. The decision to choose a particular path are based on three common strategies used in game theory.

Noncooperative Nash Equilibrium: This is relevant when agents do not communicate with each other to decide on their action at time t . So each agent plays a Nash game assuming the other agent to be an adversary.

Security Strategy: Agents do not communicate with each other to decide on their action at time t , but they adopt actions to get the best guaranteed individual payoff.

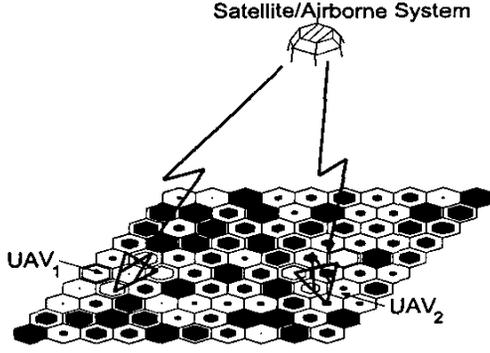


Fig. 2. Scenario for the application of noncooperative Nash or security strategies

Cooperative Strategy: In this strategy, the agents communicate with each other and decide collectively (jointly) to take the best possible action. This is also the centralized case.

In the above strategies (especially in the noncooperative Nash and security solutions) we assumed that each UAV knows the route of the other UAV till the instant of time when the next action is to be taken, but do not communicate with each other to decide on the next possible action jointly (as in the cooperative case). This scenario is a practical one in the following context: Assume that two UAV's are searching an unknown region and are tracked by a satellite or some airborne system which sends this information to the UAV's so that they can update their uncertainty maps. There is no direct communication between the UAV's and hence although each UAV is aware of the route of the other UAV, they do not know each other's contemplated actions. In such a situation the noncooperative Nash Equilibrium and security strategies are viable choice for a solution. This scenario is shown in Figure 2.

A. Non-cooperative Strategy using Nash Equilibrium

We define a non-cooperative bimatrix games for two agents [15]. A bimatrix game consists of two search effectiveness matrices, $M^1 = \{m_{ki}^1\}$ and $M^2 = \{m_{kl}^2\}$, with each pair of entries (m_{ki}^1, m_{kl}^2) denoting the payoff to each agent respectively, corresponding to a pair of decisions made by the players. The players do not cooperate with each other and arrive at their decisions independently. In such a situation the equilibrium solution can be stated as:

A pair of strategies {row k^* , column l^* } is said to constitute a noncooperative (Nash) equilibrium solution to the bimatrix game, if the following pair of inequalities are satisfied, $\forall k = 1, 2, \dots, |\mathcal{P}_1^q(C_{s_1})|$ and $\forall l = 1, 2, \dots, |\mathcal{P}_2^q(C_{s_2})|$

$$m_{k^*l^*}^1 \geq m_{kl^*}^1, \quad m_{k^*l^*}^2 \geq m_{k^*l}^2 \quad (1)$$

The pure strategy Nash equilibrium may not exist always, in which case we need to compute mixed strategies that guarantee a solution to the noncooperative game.

Mixed Strategies: A mixed strategy for a player is a probability distribution on the space of its pure strategies.

A mixed strategy for A_1 is to choose 'row 1' with probability (w.p.) y_1 , 'row 2', w.p. $y_2 \dots$, 'row $|\mathcal{P}_1^q(C_{s_1})|$ ' w.p. $y_{|\mathcal{P}_1^q(C_{s_1})|}$, so that,

$$\sum_{k=1}^{|\mathcal{P}_1^q(C_{s_1})|} y_k = 1 \quad (2)$$

The mixed strategy space for A_1 is denoted as 'Y', while for A_2 it is denoted as 'Z'. A pair $\{y^* \in Y, z^* \in Z\}$ is said to constitute a *noncooperative (Nash) Equilibrium solution* to the bimatrix game (M^1, M^2) in *mixed strategies*, if the following inequalities are satisfied $\forall y \in Y, \forall z \in Z$:

$$y^* M^1 z^* \geq y' M^1 z^*, \quad y \in Y$$

$$y^* M^2 z^* \geq y^* M^2 z, \quad z \in Z$$

Computation of mixed strategy equilibrium solution can be posed as a nonlinear programming problem [15] as follows:

A pair $\{y^*, z^*\}$ constitutes a mixed-strategy Nash Equilibrium solution to a bimatrix game (M^1, M^2) if, and only if, there exists a pair (f^*, g^*) such that $\{y^*, z^*, f^*, g^*\}$ is a solution of the following bilinear programming problem:

$$\min_{y, z, f, g} [y' M^1 z + y' M^2 z + f + g]$$

subject to

$$M^1 z \geq -f \cdot \mathbf{1}_{|\mathcal{P}_1^q(C_{s_1})|}, \quad M^2 z \geq -g \cdot \mathbf{1}_{|\mathcal{P}_2^q(C_{s_2})|}$$

$$y \geq 0, \quad z \geq 0, \quad y' \cdot \mathbf{1}_{|\mathcal{P}_1^q(C_{s_1})|} = 1, \quad z' \cdot \mathbf{1}_{|\mathcal{P}_2^q(C_{s_2})|} = 1$$

where $\mathbf{1}_{|\mathcal{P}_1^q(C_{s_1})|}$ and $\mathbf{1}_{|\mathcal{P}_2^q(C_{s_2})|}$ are column vectors of dimensions $|\mathcal{P}_1^q(C_{s_1})|$ and $|\mathcal{P}_2^q(C_{s_2})|$, with all elements 1.

The dimension of the search effectiveness matrix increases with q . Hence, computing the mixed strategy equilibrium using the bilinear programming formulation may become computationally time consuming.

Dominating Strategies [15]: We may eliminate some rows and columns that have no influence on the equilibrium solution. We say that in matrix M_1 , 'row i ' dominates row k if $m_{ij}^1 \geq m_{kj}^1, \forall j = 1, 2, \dots, |\mathcal{P}_2^q(C_{s_2})|$ and if, for at least one j , the strict inequality holds. Similarly, for A_2 , 'column j ' of M^2 is said to dominate 'column l ' if $m_{ij}^2 \geq m_{il}^2, \forall i = 1, 2, \dots, |\mathcal{P}_1^q(C_{s_1})|$, and if, for at least one i , the strict inequality holds. The dominated strategies (row k and column l , in the above example) can be eliminated without affecting the equilibrium solution. The resultant matrix dimensions will be smaller than the original $|\mathcal{P}_1^q(C_{s_1})| \times |\mathcal{P}_2^q(C_{s_2})|$ and will need less computational time to compute the mixed equilibrium strategy.

B. Security Strategy

Using security strategies agents try to secure their profits assuming unpredictable behavior of other agents. For a bimatrix game, agent A_1 chooses 'row k^* ' whose smallest entry is no smaller than the smallest entry of any other row.

$$\forall(M^1) = \max_k \min_l m_{kl}^1, \quad k^* = \arg \max_k \{\min_l m_{kl}^1\}$$

where $k = 1, 2, \dots, |\mathcal{P}_1^q(C_{s_1})|$ and $l = 1, 2, \dots, |\mathcal{P}_2^q(C_{s_2})|$, k^* is the security strategy for A_1 , and $\mathbb{V}(M^1)$ is the guaranteed payoff to A_1 . Similarly, for agent A_2 , the guaranteed payoff and security strategy are,

$$\mathbb{V}(M^2) = \max_l \min_k m_{kl}^2, \quad l^* = \arg \max_l \{ \min_k m_{kl}^2 \}$$

C. Cooperative Strategy

In this strategy the agents communicate during the decision process and choose a strategy that maximizes the joint payoff. For two agents, let the search effectiveness matrix be $M = M^1 + M^2 = \{m_{kl}\}$, where $k = 1, \dots, |\mathcal{P}_1^q(C_{s_1})|$ and $l = 1, \dots, |\mathcal{P}_2^q(C_{s_2})|$ and every element of the matrix represents the joint payoff to both the agents. A pair of strategies $\{\text{row } k^*, \text{column } l^*\}$ is said to be a cooperative strategy, if the following condition is satisfied.

$$m_{k^*l^*} \geq m_{kl}$$

$\forall k = 1, 2, \dots, |\mathcal{P}_1^q(C_{s_1})|$ and $\forall l = 1, 2, \dots, |\mathcal{P}_2^q(C_{s_2})|$. The information communicated among the agents should be kept to the minimum to reduce the computational and communication complexity. Hence, the agents communicate only their current position and future actions up to q steps. Each agent computes the search effectiveness matrix individually rather than communicating these matrices which increases the communication complexity.

D. Selection of Strategies

When there are multiple solutions, the selection of strategies by players becomes a crucial issue. The security strategies and cooperative strategies are straightforward to implement. If there exists multiple security strategies, any one of them will guarantee the same payoff level. In fact, the actual payoff is bound to be higher for both players so long as they stick to their security strategies. In the case of multiple cooperative strategies, since players communicate with each other during the decision process, they can decide on a strategy which is beneficial to the overall goal. But when multiple solutions occur for pure or mixed strategy Nash equilibrium, we need to select one of them. Since every agent has the search effectiveness matrix of all the agents, we have to select a solution whose joint payoff due to A_1 and A_2 is maximum. The selection of solution does not involve any communication with the other agent, but uses the available data in the search effectiveness matrix.

When mixed strategy equilibrium exists, a random number may be generated according to the probability distribution of the optimal mixed strategy to select the appropriate strategy. There can be other kind of selections such as choosing the strategy that has the highest probability (maximum likelihood). In our simulation, we use the random number generation method.

IV. SIMULATION RESULTS

For the purpose of simulation we consider a region composed of a 10×10 grid of hexagonal cells with two

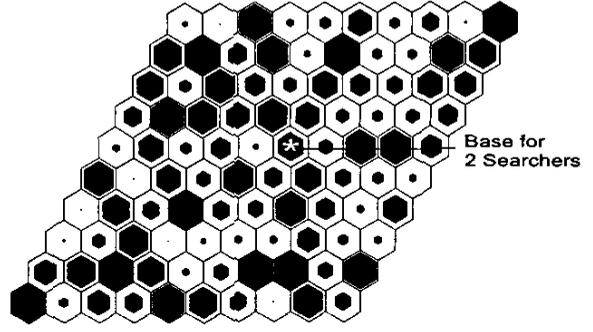


Fig. 3. Initial Uncertainty Map

searchers. The initial uncertainty map is created by generating random numbers between 0 and 100 (thus representing uncertainty as a percentage). This is shown in Figure 3. The uncertainty in a cell is shown as the grey area within a cell. The percentage of uncertainty in a cell is proportional to the size of the grey area in the cell. Initially, we assume both the searchers to be situated in the same cell. The base station location is marked with a '**' in the figure. The search is limited to 10 steps. The simulation study is carried out for $q = 1, 2,$ and 3 steps and the uncertainty reduction factors assumed are $\beta_1 = 0.5$ and $\beta_2 = 0.25$. The aim behind such a simulation is to study the performance of the proposed strategies and the effect of q on these strategies. In all the figures, the route of Searcher 1 is shown by a thick white line and that of Searcher 2 is shown by a thin white line.

Figure 4 shows the search route adopted by two searchers using noncooperative Nash equilibrium strategy. In this simulation, there was at least one pure strategy Nash equilibrium for $q = 1$, hence there was no need to use mixed strategies. For $q = 2$ at the starting position there was no pure strategy Nash equilibrium and hence there was a necessity to use mixed strategy to get a solution. For the second step there were multiple pure strategy Nash equilibriums and a particular solution was chosen based on random number generation as described in Section 3.4. From the third step onwards a unique pure strategy Nash equilibrium was available. For $q = 3$, for the first two steps there were no pure strategy Nash equilibrium and mixed strategies had to be used. In fact, there were multiple mixed strategy Nash solutions and a solution was chosen based on random number generation. Figure 5 shows the routes of security strategy for $q = 1, 2, 3$, respectively, while figure 6 shows the routes for the cooperative strategy. Note that in Figure 5(a) and (b) both searchers follow the same route.

Figure 7 shows the uncertainty reduction graph of various strategies for 10 search steps. Figure 7(a) is for $q = 1$, in this strategy the graphs of uncertainty reduction for Nash and cooperative strategies are the same, as the search routes for both the strategies are also same (see Figure 4(a) and 6(a)). From Figures 7 (b) and (c), the security strategy is the least beneficial strategy while cooperative strategy is the most beneficial and noncooperative strategy is in

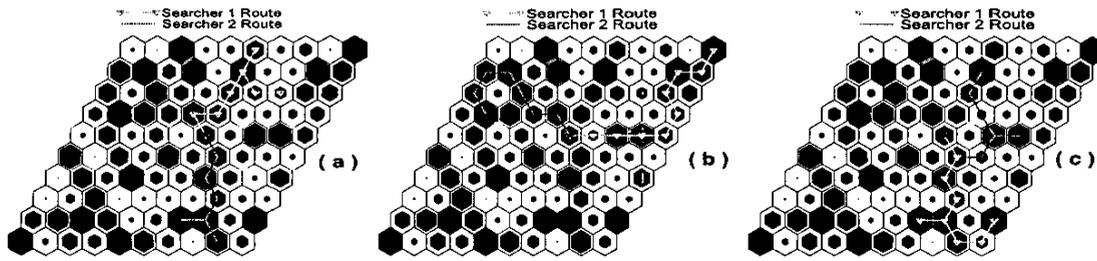


Fig. 4. Search route for two searchers using the noncooperative Nash equilibrium strategy, (a) $q = 1$ (b) $q = 2$ (c) $q = 3$

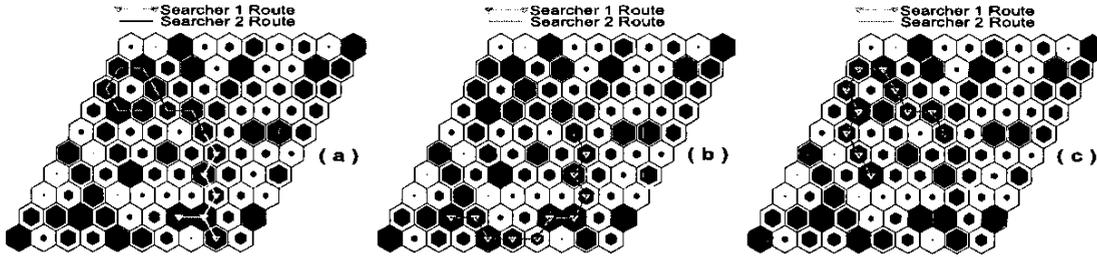


Fig. 5. Search route for two searchers using security strategy, (a) $q = 1$ (b) $q = 2$ (c) $q = 3$



Fig. 6. Search route for two searchers using cooperative strategy, (a) $q = 1$ (b) $q = 2$ (c) $q = 3$

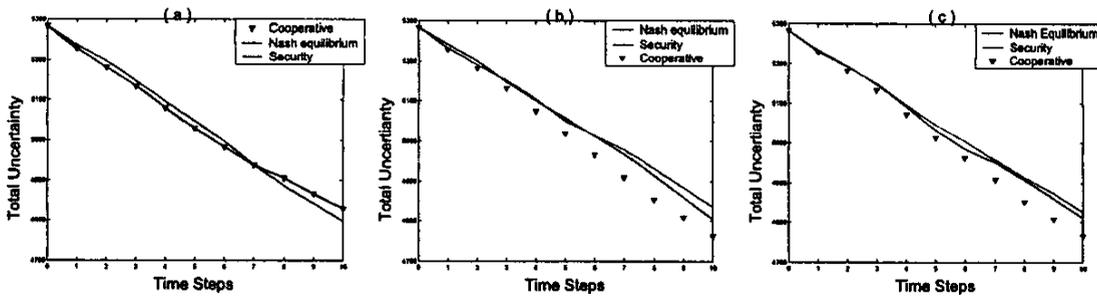


Fig. 7. Uncertainty Reduction for noncooperative Nash equilibrium, security and cooperative strategies, (a) $q = 1$ (b) $q = 2$ (c) $q = 3$

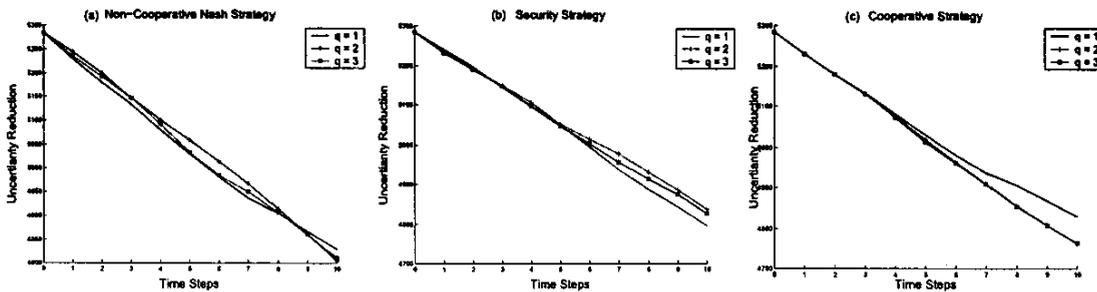


Fig. 8. Uncertainty Reduction for $q = 1, 2, 3$, (a) noncooperative Nash equilibrium strategy (b) security strategy (c) cooperative strategy

between. The performance of each strategy with respect to the value of q , the look ahead window, can be seen in Figure 8. Figure 8(a) shows that $q = 1$ performs better initially when Nash strategies are used, but higher values of q perform better when the search routes are longer. In case of security strategy, shown in Figure 8(b), initially all values of q perform equally well, but as the search routes become longer, $q = 1$ performs better. This is because, the search routes adopted by the two searchers are different and hence the net uncertainty reduction is higher than in the case of $q(2,3)$ where the searchers follow the same search route. In case of cooperative strategy, shown in Figure 8(c), higher values of q (2 and 3) give the same performance while $q = 1$ performs the worst. However, the effectiveness of the strategies lie in a narrow band for this example.

From these simulations it appears that increasing the size of q , with consequent increase in computational burden, does not yield any substantial benefits. However, with different placements and larger number of searchers the strategies may perform differently as the regions of overlaps will also increase. Also, the search space considered in these simulations is small and the simulation is carried out only for 10 steps which is also small and hence the decrease in uncertainty reduction, as shown in Figures 7 and 8, are not substantially different for different strategies and different look ahead steps. To arrive at a more concrete conclusion on the actual benefit of using game theoretical models requires the simulation to be carried out on a larger search space and for larger number of steps. These results will be reported in future articles.

Computational time: With increase in q the size of the search effectiveness matrix increases and hence the time to compute mixed Nash equilibrium solution also increases. For $q = 3$ the size of the search effectiveness matrix is 343×343 , the number of variables to be optimized is 688, and the number of constraints is 1374. So, to reduce the computational time we chose dominated strategies only and then the size of the matrix reduced to 6×13 , with number of variables to 21 and the number of constraints to 40, for the first step; and matrix size of 7×14 , with number of variables to 23 and number of constraints to 44, for the second step.

V. CONCLUSIONS

We proposed three strategies, cooperative, security and Nash Equilibrium to design strategies theory for uncertainty reduction in a search space. The strategies use the concept of q -step look ahead planning. From the results obtained it appears that the increase in q to large values does not increase the benefits substantially, but increases the computational time. Hence, $q = 2$ or $q = 3$ would give a compromise between the uncertainty reduction and computational time required. Here, it was assumed that the UAVs have complete information about other UAVs at every time step without loss of information. In reality there could be appreciable delays in updating

UAVs information [16]. Increasing the number of agents to N in the search space, increases the complexity of the strategies. The implementation of cooperative strategy is straightforward. For security strategies, we may model the problem as each agent playing against all the rest of the agents. The extension to Nash equilibrium can also be done similarly but requires careful and detailed formulation that are beyond the scope of this paper. These will be reported in a later paper.

Acknowledgements: The authors would like to acknowledge the financial support received from the IISc-DRDO program on advanced research in mathematical engineering.

REFERENCES

- [1] S.J. Benkoski, M.G. Monticino, and J.R. Weisinger: A Survey of the search theory literature, *Naval Research Logistics*, Vol. 38, No. 4, pp. 469-494, August 1991.
- [2] K. Passino, et al., Cooperative Control for autonomous air vehicles: In *Proceedings of the Cooperative Control and Optimization Workshop*, Florida, December 2000.
- [3] D. Enns, D. Bugajski, and S. Pratt: Guidance and control for cooperative search: In *Proc. of the American Control Conference*, Anchorage, Alaska, 2002.
- [4] C. Schumacher, P.R. Chandler and S.R. Rasmussen: Task allocation for wide area search munitions, *Proc. of the American Control Conference*, Chicago, Illinois, 2002, pp. 369-373.
- [5] W. Burgard, M. Moors, F. Schneider: Collaborative Exploration of Unknown Environment with teams of mobile robots, in *Advances in Plan-Based Control of Robotic Agents* (Eds. M. Beetz, J. Hestzberg, M. Ghallab, M.E. Pollack), Lecture Notes in Computer Science: 2466, Springer Verlag, 2002.
- [6] S.V. Spires and S.Y. Goldsmith: Exhaustive Geographic Search with Mobile robots along Space Filling Curves, *Collective Robotics* (Eds. A. Dragoul, M. Tambe, and T. Fukuda), Lecture Notes in Artificial Intelligence: 1456, Springer-Verlag, pp. 1-12, 1998.
- [7] S.Y. Goldsmith and R. Robinett: Collective search by mobile robots using alpha-beta coordination, in *Collective Robotics* (Eds. A. Dragoul, M. Tambe, and T. Fukuda), Lecture Notes in Artificial Intelligence: 1456, Springer-Verlag, pp. 135-146, 1998.
- [8] Y. Kitamura, Z.B. Chauang, S. Tatsumi, T. Okumoto, and S.M. Deen: A cooperative search scheme for dynamic problems, *Proceedings of the IEEE Conference on System, Man, and Cybernetics*, 1993.
- [9] R.F. Dell, J.N. Egel, G.H.A. Martins, and A.G. Santos: Using multiple searchers in constrained-path, moving-target search problems, *Naval Research Logistics*, Vol. 43, pp. 463-480, 1996.
- [10] Y. Yang, A.A. Minai and M.M. Polycarpou: Analysis of Opportunistic Method for Cooperative Search by Mobile Agents, *Proc. of the 41st IEEE Conference on Decision and Control*, pp. 576-577, December 2002.
- [11] R.W. Beard, T.W. McLain: Multiple UAV Cooperative Search under Collision Avoidance and Limited Range Communication Constraints, *IEEE Conference on Decision and Control*, 2003, (to appear)
- [12] M. Polycarpou, Y. Yang and K. Passino: A Cooperative Search Framework for Distributed Agents, In *Proc. of the 2001 IEEE International Symposium on Intelligent Control*, pp. 1-6, 2001.
- [13] D.R. Jacques: Search, classification and attack decisions for cooperative wide area search munitions: *Proceedings of the Cooperative Control Workshop*, FL, Dec. 2001
- [14] P.B. Sujit and D. Ghose, Optimal Uncertainty Reduction search using the k -Shortest Path Algorithm: In *Proc. of the American Control Conference*, pp. 3269-3274, June 2003.
- [15] T. Basar and G.J. Olsder, *Dynamic Noncooperative Game Theory*, Academic press, CA 1995.
- [16] D. Ghose, J.S. Shamma, and J.L. Speyer, Cooperative Search using teams of partially connected autonomous agents: Information latency and search strategies, Technical Report, Department of Mechanical and Aerospace Engineering, University of California at Los Angeles, July 2002.