2005 American Control Conference
June 8-10, 2005. Portland, OR, USA

Multiple UAV Search Using Agent Based Negotiation Scheme

P.B. Sujit and D. Ghose

Abstract— **In this paper, we present an agent based negotiation scheme for multiple UAVs performing search operation on an unknown region. The UAVs are subjected to limited sensor range and can communicate with their neighbouring UAVs only. The UAVs use negotiation as the decision making mechanism for obtaining search paths. The scheme is scalable to large number of UAVs without much computational burden. We study the performance of uncertainty reduction strategies using negotiation schemes with different levels of information structures. The simulation results show that the search based on negotiation scheme with various kinds of information structures outperforms random and greedy strategies with identical information structures.**

I. INTRODUCTION

The use of unmanned aerial vehicles for search and surveillance operations in unknown and hostile regions is becoming a reality. Coordinating these aerial vehicles, which perform the operation autonomously, is a difficult task. Usually, these vehicles have limited communication and decision-making capability. With large number of agents the computational overhead on the decision-making and coordination mechanism becomes high. In this paper, we present an agent-based negotiation scheme that scales well with increase in number of agents/vehicles and demands modest computational time.

Cooperative search using multiple vehicles has wide variety of applications, such as search, surveillance, disaster management, task allocation, sensor networks, etc, and has attracted the attention of several researchers in recent times. In [1], [2], the authors address the problem of searching an unknown region with multiple vehicles cooperatively. A recursive approach is used for cooperative search using a multi-objective cost function, and q-step path planning. In [3], a team of UAVs is given the task of searching a region with unknown opportunities and hazards with an objective that the team has to maximize the regions of opportunity visited, while minimizing visits to regions of hazard, subject to constraints that the UAVs must remain connected by a communication network at all times and avoid collisions among themselves. Coordinating large number of UAVs for wide area search munitions using an agent architecture is discussed in [4], where the authors use the concept of teamwork in multi-agent systems for team plan sharing and instantiation, role allocation for the UAVs, and real-time onboard information updates. In [5], a knowledge-based framework is proposed for multiple agents performing search and attack task with multiple UAVs. In [6], flying munitions carry out a parallel sweeping search using a decentralized behavior-based control philosophy with limited communication between UAVs.

The authors would like to acknowledge the financial support received from the IISc-DRDO Program on Advanced Research in Mathematical Engineering.

P.B. Sujit (Graduate student), D. Ghose (Professor) are with the Department of Aerospace Engineering, Indian Institute of Science, Bangalore 560 012, India, Email: sujit+dghose@aero.iisc.ernet.in

Negotiation schemes are used extensively in economics to obtain acceptable strategies for agents in conflict situations. Recently, many researchers have used the concept of negotiation to obtain a feasible solution for resource allocation, task allocation, etc. Argumentation-based negotiation is a powerful scheme used for conflict resolution, resource allocation, and to automate the negotiation process in multiagent systems. A comprehensive literature on argumentation-based negotiation schemes can be found in [7]-[9].

In this paper we use an argumentation-based negotiation scheme for route planning of UAVs engaged in search operations. In our earlier work [10], we addressed the problem of search using multiple UAVs with flight time restrictions, through a graph theoretical model of the uncertainty map and the k-shortest path algorithm. This approach had a strong assumption that the map is static during the sortie. To relax this assumption we introduced a game theoretical framework [11], [12], that generates a game optimal search strategies over a finite horizon. But this strategy has a large computational overhead when the number of agents is large.

II. PROBLEM FORMULATION

A. Scenario

Consider a region with some a *priori* information represented in the form of an uncertainty map. A search mission with multiple UAVs involves autonomous decision-making for on-line real-time route planning by each UAV in coordination with neighbouring agents via negotiation. A desirable search operation would be to search high uncertainty regions while spending less time in lower uncertainty regions. With such search paths the effectiveness of the search increases. However, obtaining such effective search paths depends on the computational, information sharing, and the communication capabilities of UAVs performing search operations. Here, we present a negotiation scheme with various levels of information sharing to study the effect of less communication and computational capabilities on the performance of uncertainty reduction strategies.

B. 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 [10].

C. 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. Once a cell C_i is visited by a UAV at time t, the uncertainty value of the cell 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 to a cell by an UAV.

D. Searcher Objectives and Constraints

We represent the energy spent by the UAV in moving from one cell to another as equivalent to one unit step length. The objective of the UAVs is to select search routes, in coordination with neighbouring UAVs, that collectively maximize uncertainty reduction. In every time step, a UAV can either remain in the same cell (thus spending more search effort there), or move to a neighboring cell.

It is assumed that each UAV is equipped with a sensor using which it collects information about a cell. The amount of information collected at one search step by a UAV depends on the accuracy of the sensor, which is represented as uncertainty reduction factor (β). 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.

At the beginning of a time step every agent communicates and negotiates with its neighbouring agents and selects the next step in its search route. We propose an agent based negotiation scheme for the UAVs to achieve this objective.

III. Agent Based Negotiation Scheme

We consider every UAV as an agent. Every agent has limited communication capability and is able to communicate with its neighbours for information sharing and negotiation. Each agent carries an uncertainty map with it and updates the uncertainty map at every time step. The initial uncertainty map of all the agents is assumed to be the same – although this is not a requirement. The uncertainty map updating mechanism depends on various levels of information sharing between the agent and its neighbours. The information sharing methods are described in Section IV. The UAVs have limited memory and computational capability. The decision to move from one cell to the other is arrived at through a negotiation process between the neighbouring agents. Figure 1 describes the overall decision making scheme which is common to all the agents.

The negotiation scheme used is in the same class as the general argumentation-based negotiation models [7]. We consider the negotiation period to be a small part of the time taken for one search step. We assume that the negotiation process completes within the assigned negotiation period as shown in Figure 2(a). The negotiation period can consist of many negotiation cycles, where decisions are taken by some agents in each cycle. At the beginning of a search step, each negotiation cycle consists of sending proposals to neighbours, receiving proposal from neighbours and processing them, sending accept/reject decisions to neighbours, and decide on an action based on the accept/reject decisions received. The negotiation cycle is shown in Figure 2(b).

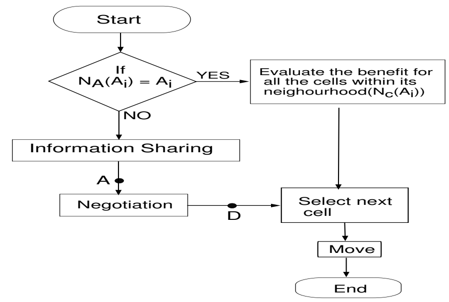

Fig. 1. Decision making schematic

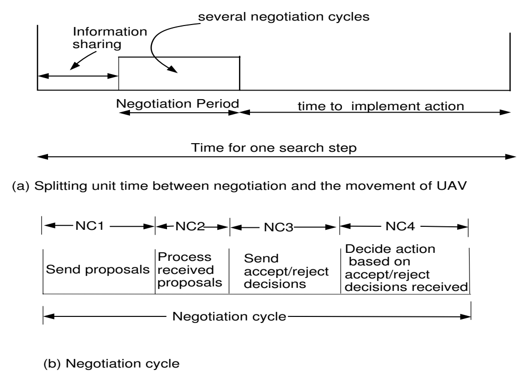

Fig. 2. Negotiation time

Figure 3 shows the flowchart of the negotiation scheme used by each agent. Consider N agents performing search in an unknown region. An agent $\mathrm{A}_i,\ i = 1, 2, \ldots, N$, has several possible options to select from in the search space. The maximum number of options an agent can have is 7, with six neighbouring cells and its own cell, denoted as $N_c(\mathrm{A_i})$. At every time step, each agent evaluates the benefit of all the options using its perception of the uncertainty map and inputs from neighbouring agents. The uncertainty value of each cell is denoted as $U_i(C_j)$, which is the uncertainty value of cell C_j as perceived by agent A_i. The expected benefit to an agent A_i, which has uncertainty reduction factor β_i, if it moves into cell C_j, is:

$$B_i(j) = \beta_i U_i(C_j), \quad \forall \;\; j \tag{1}$$

If the agent A_i has any neighbouring agents, denoted as $(N_A(\mathrm{A}_i))$, then it shares information with them and negotiates to decide its future action. Otherwise, agent A_i chooses S_i as its next move, where,

$$S_i \;=\; \arg\max_j \{B_i(j)\} \tag{2}$$

which represents an index to cell $C_{\mathrm{A_i}} \in N_c(\mathrm{A_i})$ with a

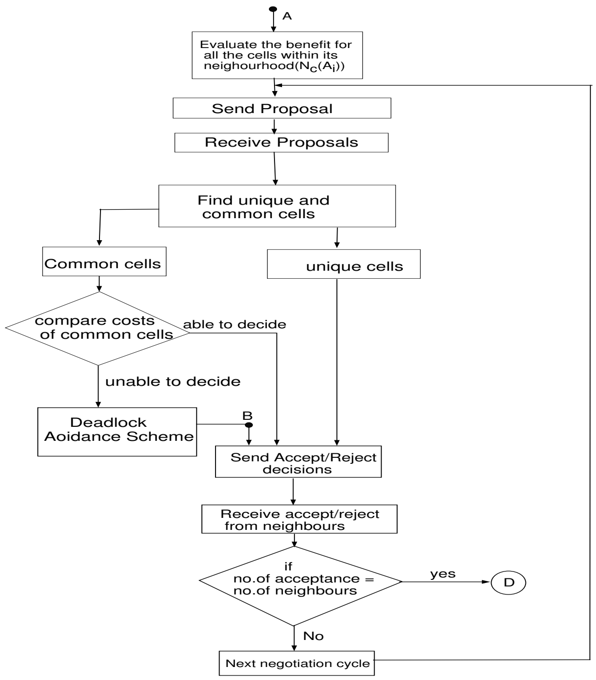

Fig. 3. Negotiation Model

benefit of $B_i(S_i)$.

Negotiation begins with the agents sending proposals to its neighbours and receiving proposals from them. The communication is synchronous in nature. We assume that there is no communication delay or loss of communication during the negotiation procedure. The decision making process in an agent is independent of other agents' decision making process and is processed parallelly by all agents.

An agent A_i sends proposal to its neighbours as $prop(\mathrm{A}_i)= \{C_{\mathrm{A}_i}, B_i(S_i)\}$. The agent receives $|N_A(\mathrm{A}_i)|$ proposals from its neighbouring agents. Let $\mathcal{C}_i$ represent a set of cells that A_i receives as proposals (including its own proposal). For each $C_{\mathrm{A}_j} \in \mathcal{C}_i$, we define a set $\mathrm{A}_i(C_{\mathrm{A}_j})$ consisting of agents that have proposed cell C_{A_j}. Let

$$C_u^i = \bigcup_{|\mathrm{A}_i(C_{\mathrm{A}_j})|=1} C_{\mathrm{A}_j},\ \ C_c^i = \bigcup_{|\mathrm{A}_i(C_{\mathrm{A}_j})|>1} C_{\mathrm{A}_j},\ \ \forall C_{\mathrm{A}_j} \in \mathcal{C}_i \tag{3}$$

where, C_u^i represents a set consisting of cells that have been proposed by only one agent, while C_c^i consists of cells that have been proposed by more than one agent. Agent A_i sends acceptance to all agents having $|\mathrm{A}_i(C_{\mathrm{A}_j})| = 1$. In other words, agent A_i sends acceptance to agents proposing for cell C_{A_j}, with $C_{\mathrm{A}_j} \in C_u^i$.

Rule 1: An agent A_i sends acceptance to agent A_j, $\mathrm{A}_j \in N_\mathrm{A}(\mathrm{A}_i)$ whose proposed cell C_{A_j}, $C_{A_j} \in \mathcal{C}_u^i$.

To send decisions to agents who have proposed cell $C_{A_j} \in \mathcal{C}_c^i$, the agent A_i compares the benefit $B_j(S_j)$ proposed by each agent A_j, $\mathrm{A}_j \in \mathrm{A}_i(C_{\mathrm{A}_j})$ and sends acceptance to an agent A_k, $\mathrm{A}_k \in \mathrm{A}_i(C_{\mathrm{A}_j})$ that has the maximum benefit and sends rejects to remaining agents $\mathrm{A}_m \in \mathrm{A}_i(C_{\mathrm{A}_j}), m \neq k$.

Rule 2: An agent A_i sends acceptance to that agent in $\mathrm{A}_i(C_{\mathrm{A}_j})$ whose proposal has the maximum benefit and reject to the other agents. When more than one agent's proposal has maximum benefit we need to invoke deadlock avoidance algorithm (discussed later).

In order to avoid more than one agent to move into the same cell, we introduce another rule which is necessary when multiple agents are proposing to a common cell.

Rule 3: An agent can send only one acceptance.

If agent A_i finds that more than one agent has the same benefit, then it uses some deadlock avoidance scheme to decide to whom it has to send accept or reject. Finally, we have a decision rule.

Rule 4: For A_i to move to the next position it has to get acceptance to its proposal from all its neighbouring agents.

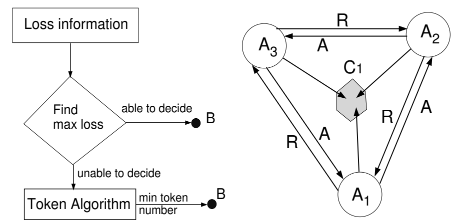

Fig. 4. Deadlock avoidance scheme

In case of same benefit expected by more than one agent from the same proposal (this may happen when the agents are homogeneous, that is, have the same value of β), the deadloack avoidance scheme as shown in Figure 4(a), is used. Then the deciding agent (A_i) requests for more information. For instance, agent A_i requests the value of possible loss for each proposing agent if they do not choose their proposed action but choose an alternative action which is the best among their remaining options. Let $\hat{B}_j$ be the set of benefits obtained by agent A_j without considering the proposed strategy S_j. The loss for each agent $\mathrm{A}_j \in N_A(\mathrm{A}_i)$ is calculated as

$$L_j = \max(B_j) - \max(\hat{B}_j), \tag{4}$$

This loss information is sent to the agent A_i for further decision making purpose. The loss L_j is compared by agent A_i for all the agents. An acceptance is sent to an agent A_j who has the highest loss and the other agents' proposals are rejected.

Another possibility is to generate a random number and base the decision on it, instead of requesting for loss information. However, the random number generation method can generate a sequence that would cause a deadlock as shown in Figure 4(b). Here, every agent gets an acceptance and a reject and a deadlock situation may occur.

Using the loss information also does not guarantee that a decision will be made. This situation can arise when agents have the same choices (for example, when multiple agents are present in the same cell). Hence, the loss by these agents will also be the same. In this case, we use a token algorithm as given below.

Token Algorithm: Every agent A_i carries a unique token number T_i. Whenever the above situation (the loss being equal) occurs, wherein the agent is unable to decide to whom it has to send acceptance, the agent compares these token numbers and chooses an agent whose token number is the lowest. Once the agent has been identified (say agent A_j), its token number is increased by a number N, where N is the number of agents. The new token number of agent A_j would be $T_j + N$. This scheme ensures that an agent that has been selected earlier in this situation, will not be selected again in a similar situation if there is at least one other agent which has not been selected before.

Whenever a situation occurs, where many agents propose for the same cell, using the rules specified and the deadlock avoidance mechanisms, it is guaranteed that there would be at least one agent that gets acceptance from all its neighbouring agents and the situation as shown in Figure 4(b) never arises.

Theorem 1: If more than one agent is proposing a cell C_j which has not been assigned to any agent in earlier negotiation cycles, then at least one of these agents will receive all acceptances from its neighbours.

Proof: We will show that, the agent in this set, that has the maximum benefit, will always receive all acceptances from its neighbours. Let this agent be A_i. We can partition the neighbour set $N_A(A_i)$ into two sets, one containing those agents that are proposing to the same cell C_j and the other containing those agents that are proposing to cells other than C_j. All the agents in the first set will send acceptance to A_i's proposal (according to Rule 2). All the agents in the second set will send acceptance to A_i's proposal (according to Rule 1 and/or Rule 2). Note that when there are two or more agents having the same maximum benefit, the deadlock avoidance algorithm is invoked. □

The *Rules(1-4)* and *Theorem 1* ensure that only one agent will be assigned to one cell among the neighbours. However, there can be multiple assignment to the same cell. This can happen for example, when the agents are not neighbours but have a cell that is common to both with high uncertainty compared to other cells in their field of view. So, there is a possibility that both agents will choose this cell as their next step. To avoid this situation, we have to increase the look-ahead step length.

After processing the received proposals each agent sends accept or reject decisions to their neighbours. Similarly they themselves receive accept or reject decisions for their proposals from their neighbours. If an agent receives acceptance from all its neighbours, then it adopts the proposal as its next action. Such agents participate in the next stage of negotiation by receiving and processing other agents' proposals only, but they do not send any proposals. Hence, they will not have NC1 and NC2 in Figure 2(b). Those agents who receive at least one reject continue with the next stage in the negotiation by sending, receiving, and processing new proposals and will go through the complete negotiation cycle as given in Figure 2(b). The participation of the agent even after getting all accept decisions ensures coordination among the agents in deciding their future decision. When an agent receives at least one reject, then the agent sends the next best proposal to its neighbours. This process continues till all the agents get all accept decisions from all their neighbours as seen in Figure 3.

When an agent A_i has neighbours who are in greater number than the number of options, then there is every chance for the agent A_i to get all its proposals rejected. In that case we allow the agent to stay in the same cell for the next time step.

IV. Information Sharing

Negotiation takes place at every search step, after the agents have updated their information, which may refer to their past actions and can be complete or partial. This updation can be through sharing of information among neighbouring agents or obtained from external sources.

Complete Information: In this case, we assume that there is a satellite or some airborne system that tracks these agents and communicates the current position of all the agents to every agent, using which each agent can recreate the past route of all the agents. We also assume that each agent knows the other agents' uncertainty reduction factors. Thus, each agent can update its uncertainty map synchronously with other agents and so, at any point in time t, the uncertainty map of each agent is the same. This is an ideal situation so far as the information updating is concerned. However, the decision making mechanism to decide on the next step in the search route is through the proposed negotiation scheme. But, in reality, such a satellite/airborne system for monitoring purpose may not be available. Hence it is necessary to study the performance of the search strategies with partial information.

Partial Information: In this case, each agents' uncertainty map is different from the others since the route which each agent has taken is different and it does not have information about other agents. The only kind of communication can be local communication with its neighbours. The information that an agent can share with its neighbours can have different levels of sophistication.

Case 1 (No information exchange): No route information is exchanged between neighbouring agents. Here, the agents update their uncertainty map based only on their own routes. This scheme models low cost UAVs with less communication cost and low memory requirements.

Case 2 (Current location information exchange): In this case, an agent A_i updates its uncertainty map with his neighbouring agents' current cell position. So, each agent knows where its neighbouring agents are and requests for β_i of its neighbouring agents (or its perception of uncertainty reduction in the visited cells), based on which it updates its uncertainty map. Agents are homogeneous and do not have any kind of agent identification mechanism.

Case 3 (Past route information exchange): In this case, each agent A_i has an unique identification number (could be the token number itself) and also a record which contains the time step at which agent A_i has met agent A_j. When an agent A_i has $|N_A(A_i)|$ number of neighbours at time t, it updates its uncertainty map using the information of past routes travelled by its neighbours. Each agent keeps a record of when it has met its neighbour previously. So, when the agent A_i meets an agent A_j at time $t+n$, A_i updates its uncertainty map based on the route travelled by agent A_j from time $(t+1)$ to $(t+n)$. Each agent has an additional memory requirement to store the agent identification and the meeting time t. Even with this sophistication, this updating scheme does not give the same result as the complete information case and agents may have different uncertainty maps at intermediate search steps.

An additional advantage of the information sharing scheme between agents is that an agent can also update the uncertainty of a cell using its current search effectiveness, which may not be reflected through an a priori fixed β. This variation in β value does not affect the negotiation scheme.

V. Simulation Results

For the purpose of simulation we consider a region composed of a 10×10 grid of hexagonal cells with ten searchers. The initial uncertainty map is created by generating random numbers between 0 and 100 (thus representing uncertainty as a percentage) and is common to all the agents. This is shown in Figure 5. The percentage of uncertainty in a cell is proportional to the size of the grey area in the cell. The position of the base station is marked with a '*' and the agents are currently located at that position as shown in the figure. The search is limited to 30 steps. The uncertainty reduction factors are assumed to be $\beta_1 = 0.5$, $\beta_2 = 0.4$, $\beta_3 = 0.6$, $\beta_4 = 0.8$, $\beta_5 = 0.9$, $\beta_6 = 0.55$, $\beta_7 = 0.45$, $\beta_8 = 0.68$, $\beta_9 = 0.7$ and $\beta_{10} = 0.75$. The objective of such a simulation is to study the performance of the proposed negotiation scheme. The simulation also gives us insight into the effect different levels of sophistication has on the information sharing. We carried out simulations with partial information and various updating rules for the agents as described in Section IV. The negotiation scheme is shown to scale well with increase in number of agents.

Figure 5 shows the route followed by each agent after the first step. The routes demonstrate the effect of negotiation. Consider the position of agents 4 and 5, since these two agents are neighbours to each other they negotiate to find a feasible strategy. The proposal for agent A_4 is different from agent A_5, which can be easily seen and hence in the first proposal itself *Rule 1* is applied and both the agents get acceptance. The decisions for these two agents are optimal as they select paths that give them maximum benefit.

Figure 6 shows the performance of uncertainty reduction for 30 steps, with various information sharing schemes. The complete information case, where every agent gets information about other agents and updates its uncertainty map, has

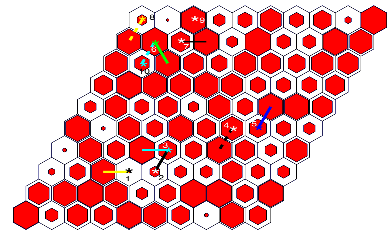

Fig. 5. Initial uncertainty map and Routes taken by agents after one time step

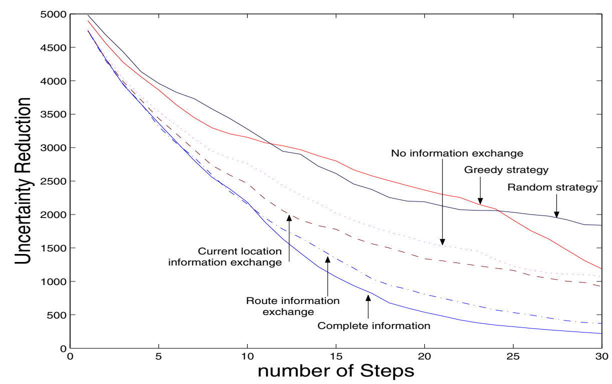

Fig. 6. Uncertainty reduction for various information levels

the best performance. But, in reality, this is not a feasible strategy as we require some mechanism to track all the agent movements. In the case of partial information where there is no mechanism to track agents and information sharing through local communication between neighbouring agents is adopted, the uncertainty reduction is less compared to the complete information case but is still better than the greedy (which has complete information) or random strategies. In the case of partial information the route updating mechanism is the best as the information sharing in this mechanism is more compared to the other two mechanisms, hence negotiation based updating mechanism is superior to random and greedy strategies.

Computational Time: Table I shows the time required for each agent per step with different information structures on 10×10 and 50×50 search space. For 10×10 search space, the time taken by the random strategy is the least as it does

Type of information	10×10 search with 10 UAVs for 30 steps	50×50 search with 30 UAVs for 500 steps
Complete information	4.43 ms	2.29 ms
Greedy strategy	2.13 ms	0.92 ms
Random strategy	2.1 ms	0.7 ms
Partial Information		
Route update	4.56 ms	3.13 ms
Current location update	4.06 ms	2.14 ms
No update	4.97 ms	2.25 ms

TABLE I

Time consumption by each strategy for per step per agent not have to negotiate nor evaluate its options. The time consumption of greedy strategy is higher as it evaluates all its options compared to the random strategy. In the complete information case, at every time step, each agent updates its uncertainty map based on the routes of all the agents and also have to negotiate if there are any neighbours, hence the time consumption is higher than greedy or random strategy. The current location update scheme has lower computational time than complete information or route updating schemes. It can be seen that the time taken for an agent using the negotiation schemes for decision making and moving to the next step takes less than 5ms. These times are for simulations carried out on a 2.4 GHz P4 machine.

To demonstrate that the negotiation scheme scales up well for a large number of agents, we consider an uncertainty map consisting of 50×50 cells and 30 agents, for a duration of 500 steps. The time, as shown in Table I, for complete information case, is higher than the greedy or random strategies and this phenomenon was also seen in the earlier simulation results with a smaller uncertainty map. In case of partial information case, the route update takes more time than the current position update or no update schemes. An important point is that for 500 steps with 30 agents, the time consumed by an agent is less compared to the smaller search space case. This is because, in 10×10 search space, the search area is comparatively small and hence more number of negotiations takes place increasing the computation, while in larger search space (50×50), the number of interactions between the agents is less and hence lower number of negotiations occur leading to lesser computational time. The simulation results show that the negotiation scheme scales well with increase in number of agents and also performs well. Figure 7 shows the performance of various strategies for $50{\times}50$ search space. In this figure we also include the performance of greedy strategy with different information sharing schemes. From the figure we can see that the complete information case performs the best and the route information exchange performs next best. When no information is exchanged, negotiation scheme performs better than random strategy. The figure also shows that for various levels of sophistication in the information structure the uncertainty reduction with negotiation scheme is far better than the corresponding greedy strategy. Since the performance of random strategies with complete information, as shown in the figure, is the worst, we did not simulate the performance of the random strategies for the other information structures.

VI. Conclusions

In this paper, we demonstrated the efficacy of negotiation mechanism for multi-UAV search in an unknown region. The negotiation scheme outperforms random and greedy strategy for identical information sharing paradigms. The computational time required by the negotiation scheme, even for large number of agents, is fairly low. The simulation results illustrate the effect of information sharing and

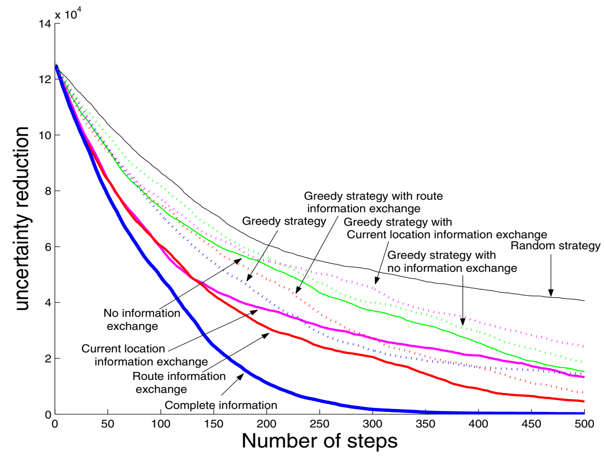

Fig. 7. Uncertainty reduction for large search space

show that with more information we get better uncertainty reduction. The effect of communication delays and loss of communication which also play important roles in determining search effectiveness are ignored and these aspects will be addressed in future research efforts.

References

[1] K. Passino, et al.: Cooperative control for autonomous air vehicles, In *Proceedings of the Cooperative Control and Optimization Workshop*, Florida, December 2000.

[2] 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.

[3] R.W. Beard, T.W. McLain: Multiple UAV cooperative search under collision aviodance and limited range communication constraints, *IEEE Conference on Decision and Control*, Maui, Hawaii, December 2003, pp. 25-30.

[4] P. Scerri, E. Liao, J. Lai, K. Sycara: Coordinating very large groups of wide area search munitions, *Theory and Algorithms for Cooperative Systems*, (Ed.) D. Grundel, R. Murphey, and P. Pardalos, Kluwer Academic Publishers, 2004 (to appear).

[5] A. Dargar, A. Kamel, G. Christensen and K.E. Nygard: An agent based framework for UAV collaboration, *Proc. ISCA International Conference on Intelligent Systems*, pp 54-59, Boston, MA, 2002.

[6] J. Schlecht, K. Altenburg, B.M. Ahmed, and K.E. Nygard: Decentralized search by unmanned air vehicles using local communication, *Proc. of International Conference on Artificial Intelligence,* Las Vegas, Nevada, 2003, Vol. II, pp. 757-762.

[7] N.R. Jennings and S. Parsons and P. Noriega and C.G. Sierra: On argumentation-based negotiation, *Proceedings of the International Workshop on Multi-Agent Systems*, Boston, USA, 1998.

[8] I. Rahwan, S.D. Ramchurn, N.R. Jennings, P. McBurney, S. Parsons, and L. Sonenberg: Argumentation-based negotiation, *The Knowledge Engineering Review*, 2004. (to appear)

[9] P. Faratin and C. Sierra and N.R. Jennings : Negotiation decision functions for autonomous agents, *International Journal of Robotics and Autonomous Systems*, 24, 3-4, pp. 159-182, 1998.

[10] P.B. Sujit and D. Ghose: Search using multiple UAVs with flight time constraints, *IEEE Transactions on Aerospace and Electronic Systems*, Vol. 40, No. 2, pp. 491-509, April 2004.

[11] P.B. Sujit and D. Ghose: Multiple agent search of an unknown environment using game theoretical models, *Proc. of the American Control Conference*, Boston, pp. 5564-5569, 2004.

[12] P.B. Sujit and D. Ghose: Decentralized game theoretical search strategies for autonomous UAVs, Submitted to *IEEE Transactions on Aerospace and Electronic Systems*.