

# Hydra: A Framework and Algorithms for Mixed-Initiative UAV-Assisted Search and Rescue

Ephrat Bitton  
Department of IEOR  
University of California, Berkeley  
ebitton@berkeley.edu

Ken Goldberg  
Departments of IEOR and EECS  
University of California, Berkeley  
goldberg@berkeley.edu

**Abstract**—We demonstrate a testbed and algorithms for collaborative human and automated (or *mixed-initiative*) decision making within the context of outdoor search and rescue. Hydra is a networked simulation tool that allows  $n$  human and  $k$  automated agents operating under different assumptions to share control over  $m$  unmanned aerial vehicles (UAVs) with cameras, with the goal of locating a hidden subject  $\theta$  as quickly as possible. The agents are modeled on a pre-defined hierarchy of authority, and the search space is characterized by varying degrees of obstructions.

Search is based on iterating the following cycle of four steps: 1) all agents generate image requests based on their individual probability density functions (pdfs), 2) Hydra collects requests and computes an optimal assignment of images to the UAVs, 3) Hydra processes the resulting image data and specifies whether or not the subject was detected, and 4) all agents update their pdfs. We propose initial models and algorithms under this framework, and we show via simulations of a scenario with three agents and one UAV that our method performs 57.7 percent better than a theoretical upper bound for a single agent and UAV.

## I. INTRODUCTION

Recent technological advances in unmanned flight have provided equipment useful in designing automated search and rescue systems that allow searchers to cover ground more quickly, without putting human operators at risk. Because such missions are too complex to be either fully automated or entirely manual, we seek to understand how humans and automation should share authority over complex command and control systems in order to maximize a measure of information about the system as quickly as possible. By integrating human intuition and reasoning capabilities (which can be difficult to quantify) with computational resources, we provide a robust framework for collaborative, mixed-initiative decision making that can accommodate different agent authority structures.

## II. RELATED WORK

Early work in collaborative control for multiple human operator, single robot systems includes a project by Cannon that enabled remote waste cleanup by having users specify locations in a shared image for a robot to excavate. [1] In this scenario, the author demonstrated an improvement in cleanup

This work is sponsored in part by the National Science Foundation Graduate Research Fellowship Program, and by the AFOSR Human Centric Design Environments for Command and Control Systems: The C2 Wind Tunnel, under the Partnership for Research Excellence and Transitions (PRET) in Human Systems Interaction.

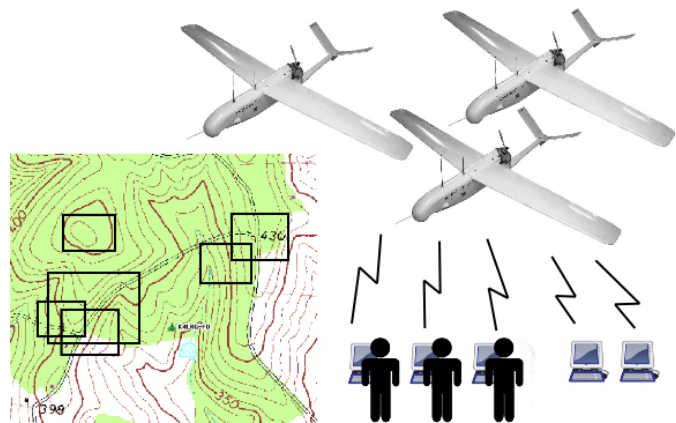


Fig. 1: Three UAVs with mounted cameras are controlled by a sequence of frame requests from distributed human and automated agents.

time, although he does not address conflict resolution between users.

Other precedents for online collaborative robot systems are described in [2]. In [3], Goldberg, Chen, et al present a theoretical framework for collaborative control, which they demonstrate with a system that averages multiple vector inputs to control the position of a moving point resource.

In [4] and [5], Song et al develop a model for shared camera control where human users request sensor data corresponding to a specific rectangular region of a shared image. By treating these image requests as spatial votes, the authors leverage the geometric properties of these votes to formulate the model as an optimization problem, which they term *spatial dynamic voting*. Given a set of image requests, they provide both exact and approximate algorithms to determine the single rectangular region that maximizes user satisfaction. The authors make use of these algorithms in [6], where they describe a system that allows multiple users to simultaneously share control over a single robotic camera. They consider the problem of sharing control over more than one camera in [7], and we provide an alternate formulation and algorithm in section III-B2.

At the Center for Robot Assisted Search and Rescue, Murphy led several studies in human-robot interaction [8], [9], where the authors examined the use of robots in urban

search and rescue settings to understand the workflow of such operations and the types of errors encountered. Murphy has also led work in cooperative control of mobile robots based on modeling and simulating societal behavior [10].

In designing a UAV control framework for search and rescue applications, two important considerations are collision avoidance (safety) and ground coverage. In [11], Ryan and Hedrick give a control algorithm for a set of UAVs flying in formation to sweep the search space using four basic path rules. Hu et al study the optimal formation constrained multi-agent coordination problem in [12] and identify geometric properties of its solutions. Ryan, Nguyen, and Hedrick consider a Coast Guard search and rescue scenario where two UAVs assist a manned helicopter by expanding the range of visual data available to the pilot. [13] The authors present a decentralized controller for maneuvering the UAVs safely.

Baum and Passino present a search-theoretic approach for fully automated cooperative control of UAVs tasked with locating stationary targets. [14] They extend classic search theory techniques to incorporate trajectory generation and to allow for multiple information seekers. Although we do not explicitly consider the trajectory generation problem in this paper, we provide a framework for coordinated control of multiple UAVs that can incorporate such models.

Chaimowicz and Kumar study the problem of using a set of UAVs to coordinate and control a swarm of ground vehicles in urban environments. [15] They develop probabilistic and behavioral models for *shepherding* based on an hierarchical framework. In [16] Caffarelli et al present algorithms for directing UAVs to monitor known stationary targets by paying them a minimum number of visits per unit of time while in consideration of the energy consumption of the UAVs and the uncertainty of the trajectories.

The following two groups consider the complete automation of a set of UAVs tasked with locating a target. We expand on their work by proposing image frames as a unified vocabulary by which both humans *and* automation can easily request sensor information from the system, in accordance with a user-defined hierarchy of agent authority. We also characterize the underlying search space with varying degrees of obstructions, which in turn affects the quality of information collected. We then present a model for extracting data from the sensors that provides a tradeoff between the size of the sampled image and the quality (reliability) of the information.

In [17], Bourgault et al describe a decentralized Bayesian approach for locating a single target by coordinating multiple autonomous agents. In their framework, automated search agents make individual decisions based only on their knowledge (prior probability distribution), and the information gathered by the different sensing platforms. Information is combined using a fully decentralized Bayesian data fusion technique, and controls are given using a decentralized coordinated control scheme. Furukawa expands on this work in [18] with the development of a coordinated control method for autonomously searching for and tracking multiple targets using multiple vehicles. Hoffmann et al [19] also consider the

#### PROBLEM FORMULATION

**Input:** Search domain  $\Theta$ ; number of UAVs  $m$ ; density of obstructions  $c(x, y)$ ; termination threshold  $a$

**Goal:** Minimize  $t$ , the number of iterations to termination

**Steps in Each Iteration:**

- 1) Agent frame request
- 2) UAV frame allocation
- 3) Sensor data extraction
- 4) Prior distribution update

**Termination Condition:** A sampled frame  $f$  such that  $B_t(f) = 1$  and  $\Pr(\theta \in f | B_t(f) = 1) \geq 1 - a$

TABLE I: Summary of the system inputs, goal, steps, and termination condition.

automation of a set of networked UAVs and develop a non-parametric technique based on particle filtering to determine in real-time the optimal control sensor locations to minimize the number of future observations required to determine the state of a target. In their setup, each UAV maintains its own estimate of the target’s current state and uses an onboard particle filter to approximate the posterior distribution once sensor data has been gathered. The authors present two polynomial-time approximation algorithms, making the network scalable while maintaining a high degree of descriptiveness.

### III. MIXED-INITIATIVE SEARCH AND RESCUE

Every search is initialized with the following inputs. We limit the search space to a bounded area of the plane  $\Theta$  that contains a hidden, stationary subject with location  $\theta \in \Theta$ . We assume that the search space is characterized by a pre-specified parameter  $c(x, y)$ , which corresponds to the density of obstructions in  $\Theta$  at point  $(x, y)$ . (These could be buildings, fog, vegetation, etc.) A distributed group of  $n$  human and  $k$  automated agents collaboratively control a set of  $m$  UAVs with mounted cameras for data collection. Each agent has an associated authority level  $\alpha_i \in [0, 1]$  and maintains a pdf  $P_{i,t}$  over  $\Theta$  of the subject’s location.

In each iteration of a session, every agent specifies a rectangular frame to investigate further.

**Definition** A frame  $f(x, y, z, t)$  corresponds to a rectangular subregion of the search space, centered at point  $(x, y)$ , with zoom level  $z$ , and indexed by time  $t$ .

Both human and automated agents submit requests in this unified format. We assume that the combined number of agents is larger than the number of UAVs available. In each iteration, the system must compute a set of  $m$  frames that maximize total “satisfaction” among the agents.

Once data for a frame  $f$  is collected, the information is processed and a binary value  $B(f)$  is returned indicating whether or not the subject is detected within  $f$ .  $B(f)$  is a Bernoulli random variable that is more likely to return a correct answer when a frame is of high resolution (i.e. covers a small area) and the density of obstructions is low.

We assume that as the area spanned by an image and/or the density of obstructions increases, the quality of information decreases. Let  $a$  be a pre-specified termination threshold corresponding to the maximum acceptable probability of a false positive in a candidate frame. Then, a search session terminates when the sensor detects the subject in a frame with small enough area so that we can ascertain with probability  $1 - a$  that the sensor information is accurate. We determine the maximum area of a terminating frame solely as a function of  $a$  and the average density of obstructions in the frame,

$$c_f = \frac{1}{\text{Area}(f)} \int_f c(x, y) dy dx. \quad (1)$$

Since time is a major factor determining the success of a search and rescue operation and hence the system, *the goal of each agent is to minimize  $t$ , the number of iterations required to locate the subject.* We develop models and algorithms for the following four steps of the search process:

- 1) **Agent Frame Request:** All agents generate frame requests based on their individual pdfs of the subject's location.
- 2) **UAV Frame Allocation:** Hydra collects requests and computes an optimal frame assignment to the UAVs.
- 3) **Sensor Data Extraction:** Hydra processes the resulting image data and specifies whether or not the subject was detected.
- 4) **Prior Distribution Update:** All agents update their pdfs to incorporate the new data.

The system is illustrated in Figure 1 and outlined in Table I. In the following subsections we present our approach to each of the above tasks in technical detail.

#### A. Step 1: Agent Frame Request

At the beginning of each iteration, participating agents submit frame requests corresponding to the rectangular subregions of the search space they wish to investigate further. We consider different strategies for each class of agent.

1) *Human Agents:* To facilitate rapid decision-making, the Hydra interface maintains for each agent a visual representation of his or her probability distribution  $P_{i,t}$  of the subject's location. Each cell of the search space is filled with a shade of blue, where a darker shade corresponds to a higher likelihood that the subject is located within that cell.

With this visual representation system, human agents can quickly get a feel for what regions of the search space have higher probabilities of finding the subject. The agents can then decide which frame to request based on intuition.

2) *Automated Agents:* We consider a search strategy for a single automated agent using results from information theory. The information entropy of a probability distribution is a measure of uncertainty, where higher entropy corresponds to greater uncertainty regarding the outcome of a random variable. The entropy of agent  $i$ 's distribution is given by

$$H(\theta) = - \int_x \int_y P_{i,t}(x, y) \log_2 P_{i,t}(x, y) dy dx \quad (2)$$

As shown by Shannon in [20], the uncertainty in the agent's distribution is minimized by sampling the frame that minimizes the expected information entropy of the posterior distribution. This is equivalent to maximizing the expected log-likelihood of the posterior, known as *information gain*.

Let  $p_1 = P(B(f) = 1)$  be the probability that the sensor data for frame  $f$  indicates that  $\theta \in f$ , and let  $p_0 = P(B(f) = 0)$  be the probability that the subject was not detected in  $f$ . We assume a general probability model for now and give an explicit one below. The information entropy conditioned on the sensor data is defined as

$$H(\theta|B(f)) = - (p_1 \log_2 p_1) H(\theta|B(f) = 1) - (p_0 \log_2 p_0) H(\theta|B(f) = 0) \quad (3)$$

Thus, the frame  $f_i^*$  that maximizes the information gained for agent  $i$  is

$$f_i^* = \arg \max_f H(\theta) - H(\theta|B(f)) \quad (4)$$

as given by [21].

While choosing the frame that maximizes information gain helps concentrate the agent's pdf, it is not designed to zero in on the areas with highest probability and hence the subject's most likely location. We thus propose a two-state search process for automated agents. In the first stage, the agent's strategy is to request the frame that minimizes the expected entropy of his or her posterior distribution. In the second stage, the agent employs a greedy strategy by requesting the frame of maximum acceptable size for termination (given  $a$  and  $c$ ) that holds the greatest expected probability.

#### B. Step 2: UAV Frame Allocation

In each cycle, we have a queue of frame requests initiated by distributed human and automated agents. Due to limited resources, not all of these requests can be met within a reasonable amount of time, since the UAVs can take several seconds to physically adjust their positions, focus, and record data. Consequently, we require a method that considers certain user attributes to determine which frames to record and in what order; we call this the *UAV frame allocation problem*.

We present a geometric approach that uses agent authority coupled with cumulative dissatisfaction to prioritize frame requests. Since both human and automated agents submit requests in the format of a frame, we do not distinguish between the two. We first consider the case where the agents share control over a single UAV (i.e.  $m = 1$ ) by formulating a spatial dynamic voting optimization problem. We then provide a heuristic that extends this solution to consider multiple available resources.

1) *Single Frame Allocation:* We adopt the model given in [4], [5] to mathematically define a user's satisfaction with a proposed frame. Our objective then becomes to maximize the priority-weighted sum of the users' individual satisfaction measures, which we denote as the *global satisfaction function*.

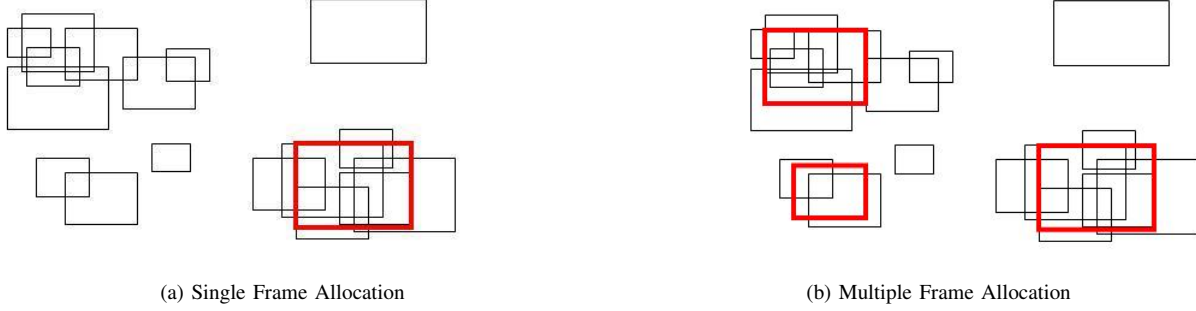


Fig. 2: A sample snapshot of the frame requests in the queue at a given moment in time. The solution to the single frame allocation problem is given by the thick rectangle in (a), and the corresponding solution to the multi-frame allocation problem is shown in (b).

Let  $F = \{f_1, \dots, f_n\}$  be a set of axis-parallel rectangles with fixed aspect ratio that represents the frame requests currently on the queue. We define agent  $i$ 's *individual satisfaction*  $s(f, f_i)$  as a measure of the similarity or overlap between candidate frame  $f$  and frame request  $f_i \in F$ . We use the *intersection over maximum* to measure the similarity between two rectangles, given by:

$$s(f, f_i) = \frac{\text{Area}(f \cap f_i)}{\max(\text{Area}(f), \text{Area}(f_i))} \quad (5)$$

This function exhibits the following property:  $0 \leq s(f, f_i) \leq 1$ . The agent's satisfaction is therefore 0 when the intersection of  $f_i$  with  $f$  is the empty set (i.e. they are disjoint), and 1 when  $f_i = f$ . A sample solution to the problem is illustrated in Figure 2 (a). Furthermore, the function is piecewise linear, allowing the use of computationally efficient optimization algorithms.

Let  $s_{i,t}$  be the  $i^{\text{th}}$  agent's satisfaction with the frame allocated at time step  $t$ , and let the agent's dissatisfaction with the frame be  $\bar{s}_{i,t} = 1 - s_{i,t}$ . We model the priority  $\rho_i$  of agent  $i$ 's frame request by taking the product of the agent's authority level and a normalized exponentially decaying function of the agent's dissatisfaction with the three previously allocated frames:

$$\rho_i = \frac{\alpha_i}{0.875} \left( \frac{1}{2} \bar{s}_{i,t-1} + \frac{1}{4} \bar{s}_{i,t-2} + \frac{1}{8} \bar{s}_{i,t-3} \right) \quad (6)$$

Hence, the more dissatisfied an agent is with the three previously allocated frames, the higher his priority will be. Furthermore, the model is constructed so that an agent's priority can never exceed his authority, and greater weight is given to dissatisfaction from more recently allocated frames.

We desire an axis-parallel rectangle  $f^*$  that maximizes total satisfaction for all agents, weighted by priority:

$$f^* = \arg \max_f \sum_{f_i \in F} \rho_i s(f, f_i), \quad (7)$$

subject to  $f \leq \max \{ \text{Area}(f_i) : f_i \in F \}$

We constrain the maximum area of an allocated frame to be less than or equal to the area of the largest frame request in the queue; this prevents the algorithm from selecting larger frames in an attempt to satisfy more agents, which would in turn significantly delay the time until the search can terminate successfully.

Polynomial-time exact and approximation algorithms for identifying a single optimal frame  $f^*$  in this context are given in [4], [5]. In the following section we extend this to find the  $m$  best frames.

2) *Multiple Frame Allocation*: Given that there are  $m$  UAVs available for use, we construct an optimization problem that seeks to determine a sequence of  $m$  frames that maximize the sum of all users' individual satisfaction. A graphical example of this problem is given in Figure 2 (b). It can be likened to the p-center or facility location problem in Operations Research, in which a set of facilities must be chosen and located to minimize the distance between customers and their nearest facility. [5]

To extend the single frame allocation algorithm to find the  $m$  best frames, we prioritize the frame requests using an  $m^{\text{th}}$  order exponentially decaying function of the agent's dissatisfaction as follows:

$$\rho_i = \frac{\alpha_i}{1 - 2^{-m}} \sum_{j=1}^m \left( \frac{1}{2} \right)^j \bar{s}_{i,t-j} \quad (8)$$

We then run the single frame allocation algorithm  $m$  sequential times, updating the priority of each frame request appropriately with each newly allocated frame.

### C. Step 3: Sensor Data Extraction

The sensor is a camera and image processing system. Given a frame specification, the UAV flies to the appropriate height and location and takes a photo with the camera. The photo is analyzed and a binary value  $\{0,1\}$  is returned, indicating 1 if the subject is detected in the frame and 0 otherwise. Since the size of the frame is related to the level of detail/resolution available to the image processing system, the sensor output value is based on two factors: 1) whether or not the subject

$a \setminus c$	0	0.02	0.04	0.06	0.08	0.10	0.12	0.14	0.16
0.02	6.47								
0.06	4.67	4.20	7.20						
0.10	4.83	5.17	5.00	6.17	11.03				
0.14	4.17	4.43	5.10	5.40	5.97	6.47	8.30		
0.18	4.30	3.57	5.40	4.87	7.13	5.40	5.10	9.27	9.07

TABLE II: The average number of steps required for five agents to locate the subject using three UAVs and with termination threshold  $a$  and obstruction density  $c$ .

is in fact located inside the frame, and 2) the accuracy of the sensor, which corresponds to the size of the frame and the density of obstructions in the underlying scene.

Let  $r(f) = \frac{\text{Area}(f)}{\text{Area}(\Theta)}$  be the ratio of the area of a frame  $f$  to the size of the search space, so that  $r(f) = 1$  if the frame is maximally large, and  $0 < r(f) < 1$  for smaller frame requests.  $f$  either contains the subject or it does not. Let  $c_f$  be a value between 0 and 1 that corresponds to the average density of obstructions in frame  $f$ , and let  $B(f)$  be the binary sensor output. Conditioned on the frame containing the subject, we model  $B(f)$  as a Bernoulli random variable, where the evidence is more likely to be accurate when the frame is small (the image is of high resolution) and the density of obstructions is low. According to agent  $i$ 's distribution, the probability that the subject will be detected in frame  $f$  at time  $t$  is given by:

$$\begin{aligned} \mathbb{P}[B(f) = 1] &= (1 - c_f)(1 - r(f))\mathbb{P}_{i,t}(f) \\ &+ [1 - (1 - c_f)(1 - r(f))](1 - \mathbb{P}_{i,t}(f)) \end{aligned} \quad (9)$$

That is, if  $f$  contains the subject, the sensor returns 1 with probability  $(1 - c_f)(1 - r(f))$ , and if  $f$  does not contain the subject, the sensor returns 1 with probability  $1 - (1 - c_f)(1 - r(f))$ . With this sensor model, we determine an upper bound on the acceptable frame size for termination  $\bar{r}(f)$  by solving the following:

$$\begin{aligned} 1 - a &\leq (1 - c_f)(1 - \bar{r}(f)) \\ \Rightarrow \bar{r}(f) &\leq 1 - \frac{1 - a}{1 - c_f} \end{aligned} \quad (10)$$

Observe that frame  $f$  meets the size requirement for termination only when  $a > c_f$ .

#### D. Step 4: Updating Priors

As evidence is collected during each cycle, every searcher's individual, spatial probability distribution for the subject's location must be updated to account for new information.

Let  $f_t^*$  be the frame sampled at time  $t$ , and let  $B_t$  be the corresponding evidence collected. For agent  $k$ , we compute the probability that  $\theta \in f_t^*$  by integrating over the marginals:

$$\mathbb{P}_{k,t}(f_t^*) := \mathbb{P}_{k,t}(\theta \in f_t^*) = \int_{f_t^*} \mathbb{P}_{k,t} \quad (11)$$

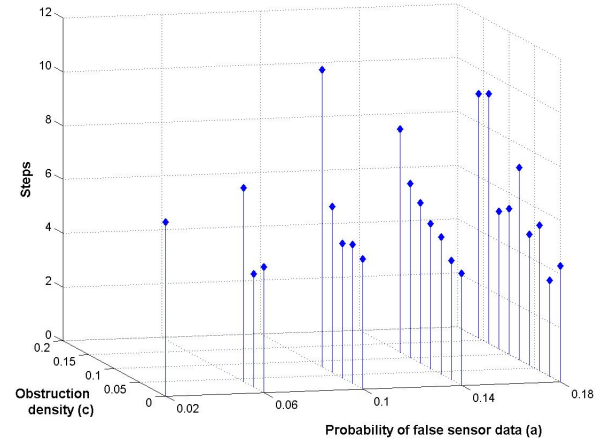


Fig. 3: Average number of steps to termination as given in Table II. The number of steps increases when we require a smaller frame to terminate.

Bayes' rule can then be used to obtain the posterior probability that the subject is located within  $f_t^*$ , conditioned on the new evidence as follows:

$$\mathbb{P}_{k,t}(f_t^* | B_t) = \frac{\mathbb{P}_{k,t}(B_t | \theta \in f_t^*) \mathbb{P}_{k,t}(f_t^*)}{\mathbb{P}_{k,t}(B_t)} \quad (12)$$

Once each searcher's posterior distribution has been updated, we can incorporate the evidence from the next frame ( $f_{t+1}^*$ ) in a similar fashion. Our prior distribution is now given by the posterior from the first step, and we use Bayes' rule and the information quality model to find the new posterior distribution.

## IV. EXPERIMENTAL RESULTS

We are interested in determining how quickly the search strategy described for automated agents can locate the hidden subject. We created a scenario with a subject hidden uniformly at random in a  $25 \times 25$  grid with five automated agents of equal authority and three UAVs; additionally, we randomly generated a different prior distribution for each agent, which we used to seed each run of the simulation. We tested the agents' combined performance 30 times each for a range of termination thresholds ( $a$ ) and constant obstruction densities ( $c$ ). We limited the first (information-seeking) phase of the automated agent frame request algorithm to three steps and set an upper bound of 200 iterations. Table II and Figure 3 reflect the average number of iterations required for the runs that successfully terminated within 200 iterations. Approximately 6.93 percent of the simulation runs diverged and were unable to locate the subject; this behavior was particularly pronounced when the difference between  $a$  and  $c$  was small (i.e. when we require a smaller frame to terminate).

We derive an upper bound on the expected number of steps for a single automated agent to detect the subject by following a naïve frame request algorithm. Let  $a$  be the pre-specified termination threshold, and let  $c$  be the constant density of obstructions across  $\Theta$ . Then the largest acceptable frame for

termination has area  $\bar{r} = 1 - \frac{1-a}{1-c}$ . If we only consider frames with area equal to  $\bar{r}$ , then we can sweep the entire search space with  $1/\bar{r}$  frames, and we terminate with the first frame in which the subject is detected. To find an upper bound on the number of frames we must take before the subject is detected, assume that we continue search until the termination conditions are met *and* the subject is truly located. Let  $f^*$  be the region of the search space in the partition of  $1/\bar{r}$  frames that contains the subject. Sampling from the frame is equivalent to sampling from a geometric distribution with a probability of success equal to  $(1-c)(1-\bar{r}) = 1-a$ . Hence, the expected number of samples of just frame  $f^*$  required before the subject is detected is  $\frac{1}{1-a}$ , and

$$E[\text{time to truly locate subject}] = \frac{1}{\bar{r}} \left( \frac{1}{1-a} \right) \quad (13)$$

To compare the maximum information gain algorithm against the naïve search strategy, we ran each simulation until an appropriately sized frame was found that truly contained the subject. In our experiments we observed that on average the number of steps taken by the maximum information gain frame request algorithm is 65% fewer than the expected number of steps required by the sweep strategy, with a standard deviation of 15.8%. In experiments with three automated agents and a single UAV, we observed a 57.7% improvement over the naïve search strategy.

## V. CONCLUSION AND FUTURE WORK

In this paper we describe a framework for collaborative control that is designed to accommodate different models for sensor data extraction, agent authority hierarchies, prior distributions, and termination conditions. Future work will include extensive experiments with automated and human agents. We will also seek to extend the frame allocation algorithm to account for the current positions of the UAVs and the cost of travel, incorporating path planning and scheduling algorithms.

## ACKNOWLEDGMENTS

The authors thank Claire Tomlin, Pravin Varaiya, Janos Sztipanovits, Anand Kulkarni, Jijie Xu, and Menasheh Fogel for helpful comments and review.

## REFERENCES

- [1] D.J. Cannon. *Point-and-direct telerobotics: object level strategic supervisory control in unstructured interactive human-machine system environments*. PhD thesis, Stanford University Department of Mechanical Engineering, 1992.
- [2] M. McDonald, D. Small, C. Graves, and D. Cannon. Virtual collaborative control to improve intelligent robotic system efficiency and quality. *IEEE International Conference on Robotics and Automation*, 1:418–424, 1997.
- [3] K. Goldberg and B. Chen. Collaborative control of robot motion: Robustness to error. *International Conference on Intelligent Robots and Systems (IROS)*, 2:655–660, 2001.
- [4] D. Song, A.F. van der Stappen, and K. Goldberg. Exact Algorithms for Automated Satellite Frame Selection. *IEEE Transactions on Automation Science and Engineering*, 3(1):16–28, 2006.

- [5] D. Song and K. Goldberg. Approximate Algorithms for a Collaboratively Controlled Robotic Camera. *IEEE Transactions on Robotics*, 23(5):1016–1070, 2007.
- [6] D. Song and K. Goldberg. Sharecam part i: Interface, system architecture, and implementation of a collaboratively controlled robotic webcam. *IEEE/RSJ International Conference on Intelligent Robots (IROS)*, 2003.
- [7] Y. Xu, D. Song, J. Yi, and A.F. van der Stappen. An Approximation Algorithm for the Least Overlapping p-Frame Problem with Non-Partial Coverage for Networked Robotic Cameras. *IEEE International Conference on Robotics and Automation (ICRA)*, 2008.
- [8] RR Murphy. Human-robot interaction in rescue robotics. *Systems, Man and Cybernetics, Part C, IEEE Transactions on*, 34(2):138–153, 2004.
- [9] JL Casper and RR Murphy. Workflow study on human-robot interaction in USAR. *Robotics and Automation, 2002. Proceedings. ICRA '02. IEEE International Conference on*, 2, 2002.
- [10] RR Murphy, CL Lisetti, R. Tardif, L. Irish, and A. Gage. Emotion-based control of cooperating heterogeneous mobile robots. *Robotics and Automation, IEEE Transactions on*, 18(5):744–757, 2002.
- [11] A. Ryan and JK Hedrick. A mode-switching path planner for UAV-assisted search and rescue. *IEEE Conference on Decision and Control and the European Control Conference*, pages 1471–1476, 2005.
- [12] J. Hu, M. Prandini, and C. Tomlin. Conjugate points in formation constrained optimal multi-agent coordination: A case study. *SIAM J. Control and Optimization*, 2006.
- [13] A.D. Ryan, D.L. Nguyen, and J.K. Hedrick. Hybrid control for uav-assisted search and rescue. *the 2005 International Mechanical Engineering Congress and Exposition, Orlando, FL*, 2005.
- [14] M. Baum and K. Passino. A search-theoretic approach to cooperative control for uninhabited air vehicles. *Proc. of the 2002 AIAA Guidance, Navigation, and Control Conference and Exhibit*, 2002.
- [15] L. Chaimowicz and V. Kumar. Aerial shepherds: Coordination among uavs and swarms of robots. *Proceedings of the 7th International Symposium on Distributed Autonomous Robotic Systems (DARS 2004)*, pages 231–240, 2004.
- [16] L. Caffarelli, V. Crespi, G. Cybenko, I. Gamba, and D. Rus. Stochastic Distributed Algorithms for Target Surveillance. *Intelligent Systems Design and Applications*, 2003.
- [17] F. Bourgault, T. Furukawa, and HF Durrant-Whyte. Coordinated decentralized search for a lost target in a bayesian world. *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, 1, 2003.
- [18] T. Furukawa, F. Bourgault, B. Lavis, and H.F. Durrant-Whyte. Recursive Bayesian Search-and-Tracking Using Coordinated UAVs for Lost Targets. *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 2521–2526, 2006.
- [19] G.M. Hoffmann, S.L. Waslander, and C.J. Tomlin. Mutual Information Methods with Particle Filters for Mobile Sensor Network Control. *45th IEEE Conference on Decision and Control*, pages 1019–1024, 2006.
- [20] CE Shannon. A mathematical theory of communication. *Bell Systems Technical Journal*, 27:379–423, 1948.
- [21] S. Kullback and RA Leibler. On Information and Sufficiency. *The Annals of Mathematical Statistics*, 22(1):79–86, 1951.