Journal of Intelligent & Robotic Systems manuscript No.
(will be inserted by the editor)

Planning and Monitoring Multi-Job Type Swarm Search and Service Missions

Meghan Chandarana · Dana Hughes · Michael Lewis · Katia Sycara · Sebastian Scherer

Received: date / Accepted: date

Abstract To transition from control theory to real applications, it is important to study missions such as Swarm Search and Service (SSS) where vehicles are not only required to search an area, but also service all jobs that they find. In SSS missions, each type of job requires a group of vehicles to break off from the swarm for a given amount of time to service it. The required number of vehicles and the service rate are unique to each job type. Once a job has been completed, the vehicles are able to return to the swarm for use elsewhere. If not enough vehicles are present in the swarm at the time that the job is identified, that job is dropped without being serviced. In SSS missions that occur in open environments, the arrival rate of jobs varies dynamically as vehicles move in and out of the swarm to service jobs. Human operators are tasked with effectively planning and managing these complex missions. This paper presents a user study that seeks to test the efficacy and ease-of-use of a prediction model known as the Hybrid Model as an aid in planning and monitoring tasks. Results show that the novel computational model aid allows operators to more effectively choose the necessary swarm size to handle expected mission workload, as well as, maintain sufficient situation awareness to evaluate the performance of the swarm during missions.

Keywords swarm search and service · planning · monitoring · queuing theory · Markov chain

Sponsored by NASA LaRC through NIA Activity 201020.

Meghan Chandarana
Autonomous Integrated Systems Research Branch
NASA Langley Research Center
E-mail: meghan.chandarana@nasa.gov

Dana Hughes, Katia Sycara and Sebastian Scherer
Robotics Institute
Carnegie Mellon University
E-mail: danahugh@andrew.cmu.edu, katia@cs.cmu.edu, basti@andrew.cmu.edu

Michael Lewis
School of Computing and Information
University of Pittsburgh
E-mail: ml@sis.pitt.edu

1 Introduction

In many anticipated applications of swarms, vehicles will work together to simultaneously search an area while servicing tasks (or jobs) as they appear. We call these Swarm Search and Service (SSS) missions. In these missions, the primary role of the swarm is to fully search and map an environment. When possible, jobs discovered during searching must be immediately serviced to avoid potential negative consequences from occurring. For example, in the case of a swarm tasked with searching a section of a city after a natural disaster, the primary task of the swarm may be to search the assigned area and map any damage that has occurred; however, available agents in the swarm should repair gas leaks, if possible, to avoid explosions or additional infrastructure damage. Each type of job in the environment requires a different number of vehicles to service for a different amount of time.

As vehicles move in and out of the swarm, the amount of area seen by the swarm at any given time – coverage rate – changes dynamically. More specifically, as vehicles are added, the swarm expands and its coverage rate increases. The opposite is true when vehicles leave the swarm. The arrival rate also changes dynamically as a result. Predicting how and when the arrival rates will change is challenging due to the fact that job locations only appear when a job is within sensing range of the swarm vehicles. This makes it very difficult for operators to plan and manage SSS missions.

Swarm mission planning typically focuses on determining explicit paths [2] or general policies [6] for vehicles to service the expected jobs. Little to no work has focused on the resource allocation problem associated with determining the necessary number of vehicles needed to achieve mission success. In monitoring tasks, research has focused on how operators can manage multiple vehicles simultaneously to achieve the desired goal [15][20]. However, in SSS missions, it is important to understand the swarm's performance and any situational and environmental factors that can affect it over time. Therefore, current work must focus on developing tools and methods to assist operators in understanding the complex relationship between different objectives to inform better real-time decisions.

Previous work developed and validated a Hybrid Model for predicting the performance of swarms during SSS missions [8]. This paper presents a user study that explores the efficacy and ease-of-use of the Hybrid Model as an aid for operators tasked with planning and monitoring SSS missions where the arrival rate of jobs changes dynamically. Herein, we demonstrate that the Hybrid Model's ability to predict the expected relationship between various mission parameters enables operators to build a mental model of the trade-offs between different mission objectives. This enables operators to effectively assign vehicles to the swarm and maintain sufficient situation awareness to evaluate the performance of the swarm, as well as, identify anomalies that may arise.

2 Related Work

2.1 Planning Interfaces

To effectively plan large scale multi-agent missions, human operators must define mission objectives and allocate the necessary resources to the mission. In addition, they must determine how the vehicles will carry out the necessary tasks during execution. Regardless of what the mission is, the system's interface design determines the means by which the operator makes these decisions.

2.1.1 Interface Autonomy

Multi-agent mission planning interfaces range in autonomy level from manual to autonomous. Manual interfaces require that operators manually input the mission's area of interest, targets and step-by-step vehicle maneuvers. However, they do abstract away the controllers necessary to accomplish the desired maneuvers [16] [2]. Mission planning interfaces vary greatly in sophistication and scope, ranging from simple path plans relying on waypoints, to plan libraries that can generate complex multiphase plans, and cooperative planners that focus on satisfying an objective function under operator guidance. Autonomous interfaces use a variety of frameworks to autonomously plan the paths of all the vehicles, including Markov Decision Process frameworks [25], game theoretic [3], and integer programming [4]. In addition, grammar-based planning [5], task-based planning [19], behavior-based planning [37][33] and Petri Net-based planning [46] have been used in autonomous multi-agent planning. Multi-unmanned vehicle (multi-UV) mission planning interfaces can be divided into three general categories – path planners, task sequencers, and resource allocators – although a single interface may address multiple categories.

2.1.2 Path Planning

For missions with tasks requiring multiple UVs, optimal or near-optimal trajectories can be computed. These tasks range from more simplistic goals such as moving between locations [44] or searching an area [34] to more complex goals such as patrolling [40] or assigning robots to targets [30]. Area searching tasks can be further broken down into tasks where robots are deployed evenly [6] or they converge to a rendezvous location [35]. Due to the high computational complexity for solving these path planning problems for even a small number of vehicles, human operators are unable to perform these calculations. Therefore, automating path planning computations has been a major focus for mission planning researchers. Automated path planning is widely studied and several commercial tools are available [1]. Research shows that automated path planning is necessary for applications that require operators to supervise more than 4 UVs [15], and it results in better performance than aiding or automating monitoring for applications like foraging, which requires both monitoring for targets and directing UV movement [12] [23] [29] [26]. In systems such as RoboLeader where replanning is supervised by the operator, but initial path planning is automated, the automated path planning was also shown to improve performance without a loss of situational awareness [11] [13].

2.1.3 Task Sequencing

Path planning for multiple UVs to accomplish non-phased, geometric objectives, such as coverage and rendezvous, are fairly straight-forward to define mathematically. The objective function for phased tasks such as ingress, search, egress and role-based coordination, on the other hand, is much more challenging to define as these missions require sequential coordinated actions. If the appropriate coordination plan can be instantiated and executed, the mission has been accomplished. Interfaces for planning sequenced missions range from manual specification in variations of linear temporal logic (LTL) to plan libraries where stereotypical missions are retrieved and edited. Manual specification systems like Mission Lab [17] are programmed through assemblages that are temporally sequenced. An assemblage consists of groups of basic behaviors and coordination mechanisms. A finite state automata is used to represent the assemblages. Changes in perception trigger the transition between states. Therefore, expected perceptual changes act as a proxy for mission phases when operators are constructing mission plans.

On the other end of the spectrum are plan libraries that prespecify a sequence of coordinated actions. This allows for high plan complexity, but requires that situations encountered match the plans that are available. Plan libraries also allow operators to formulate plans from a high-level perspective where they consider role assignments, decision points and termination conditions as opposed to low-level path planning. These plans can be executed more quickly as the sequence of action is already defined and they can include cascading interdependencies, making them difficult to manage in real-time for operators. Hierarchical task network (HTN) planners break down larger tasks into more manageable recursive subtasks and represent constraints as a network. Completing the subtasks requires a sequence of actions to be carried out. HTNs such as the Playbook integrate an interface [38]. In general, a skeleton play provides a common task model that is necessary for interaction between the planner and operator. To interact with the system, these skeleton plays are then "called" by the operator. In addition, the operator provides needed parameters such as designating targets. Mission Lab provides a simplistic mechanism for modifying prior plans [43]. This mechanism is widely used in systems for planning sequenced missions.

2.1.4 Resource Allocation

While path planning and task sequencing are likely to occur in a known spatial-temporal context, planning for resource allocation is qualitatively different and typically involves determining policies to be applied in a dynamic environment rather than actual paths or behavior sequences. Where accurate models of the environmental dynamics are available, as in the case of wildfire propagation [24], distribution of resources can be optimized. Where jobs are not known in advance, however, auctions are the most commonly chosen mechanism for solving this problem [22][32][18] as they allow for distributed decision making by pitting agents' bids against one another to find the lowest cost allocation. While used most frequently to allocate tasks, resources such as spatial regions [45] or trade-offs among alternate fueling stations [31] can be allocated through auction as well. In addition to allocating the available resources effectively, current work in the multi-UV domain

also focuses on determining how many resources (i.e., vehicles) should be available for a given task [41].

2.1.5 Execution Monitoring

Mission planning interfaces often serve double duty in execution monitoring, most frequently through a map display [17][38] showing vehicle paths and progress. Additional information, such as current task in a sequence of tasks [17][38], is commonly displayed in the context of this map using devices such as Gantt charts [14]. By observing the mission unfold in the context within which it was planned, it is hoped the operator will be able to detect and correct departures in execution. This form of monitoring is particularly effective where mission progress can be represented through vehicle locations on a map (path planning and task sequencing), but may be less effective in portraying allocations of tasks or resources particularly when trade-offs are involved. For these more complex judgements, displays summarizing and characterizing trade-offs [14] may be needed to support effective control.

2.1.6 Planning for SSS Missions

When planning SSS missions, operators are often faced with balancing complex trade-offs to achieve a variety of conflicting mission objectives. For example, an operator may wish to increase swarm size to improve the swarm's ability to service jobs. However, doing so not only increases overall system cost, but failure to understand the resource requirements for the job types present may result in an inability to actually complete any additional jobs. This is due to the fact that each job requires a specified number of vehicles to service it, but if that threshold is not met, the vehicles will simply be added to the swarm without providing any additional use.

Therefore, the question arises: can model predictions be used as a planning tool to improve operators' ability to meet mission objectives? By incorporating a model into the mission planning tools and interfaces, human operators might be able to quickly and easily compare system performance across different mission configurations. In addition, by leveraging a predictive model, swarm mission success could become less dependent upon highly skilled operators, thereby making swarm systems more accessible to a broader user base.

Because the state of the art in route planning and task sequencing is well developed, our research addresses a resource allocation planning problem novel to SSS missions. Although our experimental task uses a lawnmower search pattern, our method is independent of the type of search, and depends only on the rate of coverage. Therefore, it can accommodate arbitrary patterns such as those resulting from expected information gain [39]. Job servicing in SSS missions represents a previously unstudied special case of task allocation in mission planning. Rather than choosing the way vehicles are assigned to jobs, SSS assigns any vehicles available from a common pool, which is depleted by vehicles leaving to service other jobs and enlarged as vehicles return. In the case of the hybrid model used in the experiment, this is further complicated by the dynamic arrival rate of jobs due to changes in the rate of coverage as vehicles leave and return to the searching swarm. Mission planning for the resource allocation portion of SSS requires estimating the

number of vehicles needed to balance this ebb and flow in such a way that the area is searched and jobs serviced efficiently.

This is a particularly difficult problem because its unfamiliar problem domain leads individuals to allocate a higher proportion of their mental capacity towards task representation rather than actual problem solving [21]. The large number of constraints on the problem (job numbers, sizes, service times, etc.) also increases the representation complexity [36]. This, along with the problem's non-monotonicity, makes finding a feasible solution difficult. The human strategy of constructing internal linear models to make predictions and human need to analyze complex trade-offs sequentially instead of simultaneously [28] also contribute to difficulty in making accurate decisions. To counter these difficulties, we have developed an experimental mission planning interface that makes a predictive model available for use in planning. By visualizing non-monotonic trade-offs between vehicles in the swarm and dropped jobs, the user can find a swarm size minimizing expected costs. In subsequent monitoring, the operator can have access to these familiar representations along with a map of the mission to follow the mission's progress and identify and diagnose anomalies.

Our previous work [10] assumed a constant coverage rate scenario such as in an urban environment where the swarm's sensing range is limited within narrow streets and thus experience a constant arrival rate despite changes in swarm size. Such would be the case if a swarm were tasked with searching for gas leaks in the downtown area of a city after a natural disaster. The SSS system was formulated as a variant of the Dynamic Vehicle Routing problem with time constraints [6], where the jobs are identified as they are sensed by a swarm member as opposed being sensed at the time they appear in the environment by an omniscient observer. The formulation utilized a patience time equal to the time a vehicle takes to travel to the edge of its sensing radius (i.e., jobs are immediately serviced). The probability of a job being serviced within an accepted patience time window was equated to the probability of a job being dropped.

In [10] SSS systems with constant coverage rate were modeled as an M/M/k/k queue – Queuing Model – with an infinite horizon where the number of servers was equal to the number of vehicles in the swarm. The system state was defined as $s_i = [n_1, ..., n_M]$, where n_m is the number of jobs of type m currently being serviced. The probability of being in a state for a given swarm size was given by:

$$P_\pi(s) = \prod_{m=1}^{k} \frac{\rho_m^{n_m}}{n_m!} \cdot C \tag{1}$$

$$C = \left(\sum_{s \in \mathcal{S}} \prod_{m=1}^{k} \frac{\rho_m^{n_m}}{n_n!} \right)^{-1} \tag{2}$$

$$\rho_m = \frac{\lambda_m}{\mu_m}, \tag{3}$$

where ρ_m is the utilization factor, λ_m is the arrival rate of a job of type m and μ_m is the service rate.

The probability of being in each state can then be used to calculate the total expected number of jobs that will be dropped for a given swarm size using the following equations:

$$R_\pi = \sum_{s \in \mathcal{S}} r_\pi(s), \tag{4}$$

$$r_\pi(s) = P_\pi(s) \cdot \sum_{d \in \mathcal{D}} \lambda_d(s). \tag{5}$$

To examine the effect of the human operator's utilization of the swarm size prediction model (Queuing Model) on mission performance, a user study was designed [9]. Twenty subjects took part in 10 trials each and were evenly split between a control and an experimental group. The experimental group had access to the mission parameters and predicted performance of the swarm given by the Queuing Model, while the control group only knew the mission parameters. Each trial presented participants with an SSS mission where jobs of 3 types were expected in the environment. For each trial, participants were asked to use the information provided to determine what they thought was the swarm size required to handle the expected job workload.

Results from the study showed that participants in the experimental group were able to plan missions that resulted in a lower cost and had a smaller swarm size. The result were statistically significant. Large effect sizes showed that the differences in the data were in fact a result of the differences between the groups.

3 Hybrid Model for Performance Prediction

To model the performance of an SSS mission in an open environment with dynamically changing coverage rates, a Hybrid Model was developed [8]. The Hybrid Model utilizes a Markov chain state space representation to capture the dynamically changing swarm size. Queuing theory is then applied to determine the transition dynamics between states. In doing so, the steady state performance of the system can then be analyzed to model the expected overall performance of the swarm.

The dynamically changing swarm size between states is captured by representing the current size of the swarm in a given state as $s_i = [n_1, ..., n_M]$, where n_m is the number of jobs of type m that are currently being serviced. The complete state space is given by $S = \{s_1, ..., s_K | N - \overline{n}_{service} \cdot s_i \geq 0\}$, where N is the size of the swarm and $\overline{n}_{service}$ is the vector of the number of vehicles needed to service each of the job types. For each state, the swarm is comprised of vehicles that are either out servicing jobs or remain in the swarm and can be used to search: $N = N_{busy} + N_{search}$. The number of jobs in the system for a given state and the required number of vehicles for each job type allows both N_{busy} and N_{search} to be calculated.

Assuming that the jobs are randomly and uniformly distributed, their arrivals follow a Poisson distribution. Therefore, from queuing theory, the probability of k jobs of type m arriving when the current swarm size is n is given by

$$p_a^m(\lambda_m(n), k) = e^{-\lambda_m(n)} \frac{\lambda_m(n)^k}{k!}, \tag{6}$$

where $\lambda_m(n) = \phi_m * \mathcal{F}(n)$ is the arrival rate of a job of type m. $\lambda_m(n)$ is dependent upon the density of jobs of type m in the environment and $\mathcal{F}(n)$ is the coverage rate of the swarm. The coverage rate function determined by the formation of the vehicles, their sensing radius, vehicle spacing and the expected velocity of the vehicles. As a result, each mission's parameters provide a functional representation of the relationship between coverage rate and the number of vehicles currently in the swarm. This ensures that the model accounts for the actual dynamics associated with the mission. By incorporating the coverage rate as a function dependent on the scenario specific parameters, the arrival probability calculation is generalizable to any application and its associated job parameters.

For the purposes of this study, the swarm is assumed to travel in a line formation where the vehicles are separated by their sensing radius of 1 meter ($\mathcal{F}(n) = n$). Other formations, such as a wedge, are equally viable and could provide advantages of redundant coverage to compensate for fallible sensing. The line formation was chosen for illustration because it provides maximal coverage and the simplest relation between swarm size and coverage rate. This scenario mimics that of one in which the robots are tasked with finding a target as quickly as possible, such as in the case of a swarm tasked with finding a hidden munitions depot in a given area of interest. In such a scenario, spreading out as much as possible with a small (or no) overlap is preferred due to the priority of minimizing the search time.

The probability of χ jobs of type m being completed is given by

$$p_c^m(\chi) = \binom{n_m(s_i)}{\chi} (p_c^m)^{\chi} (1 - p_c^m)^{(n_m(s_i) - \chi)}, \tag{7}$$

where $n_m(s_i)$ is the number of jobs currently being serviced of type m in state s_i and p_c^m for exponentially distributed service times with mean μ_m is $p_c^m = 1 - e^{-\mu_m}$. The transition probabilities are then calculated by considering all the possible combinations of jobs arriving and being completed for each state. The fully populated transition matrix, T, is then used to determine the steady state distribution, π, across the states as follows

$$\pi = \pi \cdot T \text{ and } \sum_i \pi_i = 1. \tag{8}$$

The steady state distribution is then used to determine the expected number of dropped jobs, the mission time, average utilization, average coverage rate, and average power consumption for a given starting swarm size.

4 Method

4.1 Experimental Design

A user study was conducted to evaluate the efficacy of the Hybrid Model as a planning and monitoring aid for general SSS missions. Twenty-four participants were included in the study (10 female, 14 male) and were split evenly across two groups: experimental and control. Most of the participants were students at a university. All participants completed 5 trials. No training trials were provided.

After all the trials, participants were asked to fill out a NASA TLX workload measure survey [27].

Table 1: Job Parameters for Each Trial

	Job Type 1			**Job Type 2**		
Trial	**n**	V_n	μ	**n**	V_n	μ
1	10	10	20	10	15	10
2	15	10	10	10	15	20
3	15	5	40	10	8	20
4	5	5	20	15	8	30
5	15	5	15	15	8	25

Each trial in the study consisted of 2 parts: (1) a planning task and (2) a monitoring task. All trials presented missions comprised of 2 job types. The job parameters (expected number of jobs, required number of vehicles and service time) varied across trials. Table 1 shows the job parameters for each of the trials. The expected number of jobs (n), required number of service vehicles (V_n) and service time (μ) are shown for both job types. In the planning portion, participants were asked to determine the swarm size required to handle the expected workload of jobs present in the environment. To make this decision, 3 cost values were provided for consideration: a vehicle cost, a dropped job cost and a missed area cost. These cost values were meant to simulate the relative importance between the commonly seen high level mission goals of minimizing swarm size, servicing all jobs and searching all of a given area. Although explicit numerical values for each cost may not be available in real SSS missions, they were provided so that all participants maintained the same notion of relative importance between the performance metrics. The same cost values were used across trials and were 10, 30 and 5 for each vehicle, each dropped job and each cell of missed area, respectively.

As a result of the dynamically changing coverage rate of the swarm, non-monotonic relationships between mission parameters arise. The causation between parameters is not bidirectional. These complex, and often times conflicting, relationships make planning and monitoring SSS missions extremely challenging for operators. Figure 1 shows a sample relationship between 2 mission parameters: swarm size and dropped jobs. As shown, the number of dropped jobs increases as swarm size increases for various ranges of swarm size. This counterintuitive relationship can be attributed to the fact that jobs require a set number of vehicles to service them. Although the number of vehicles in the swarm is in fact increasing, until a certain threshold is met, there still are not enough additional vehicles to service additional jobs. Additionally, as the number of vehicles increases, the coverage rate of the swarm increases, resulting in a faster arrival rate of jobs.

For the planning tasks, participants in the experimental group were also given access to the predicted relationships between various mission parameters determined by the Hybrid Model (Section 3). They were able to explore the relationship between swarm size, expected number of dropped jobs, power consumption, coverage rate, swarm utilization and mission time in 2D and 3D interactive plots (Figure 2). In addition, they were given the job parameters (expected number of jobs, required number of vehicles and service time) for each job type and the cost values. Participants in the control group only had the job parameters and

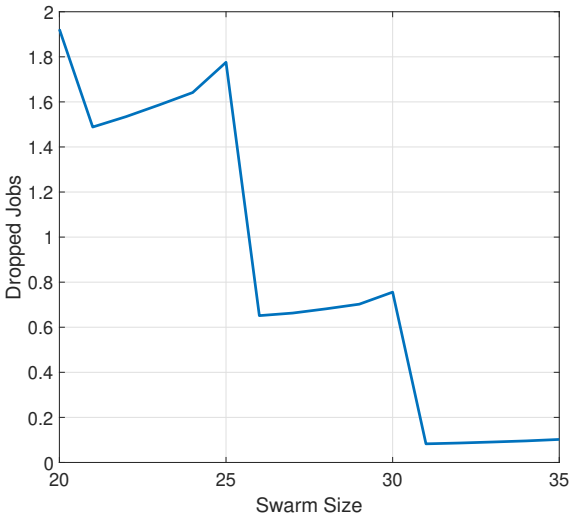


Fig. 1: Example relationship between dropped jobs and swarm size.

cost values. All participants were also asked to ensure that vehicles used less than an allowed 40Wh of battery power and were given the numerical values for the amount of power consumed in each time step per robot for each job type and for searching. Once participants had chosen a swarm size, their cost (red text) was shown to them at the bottom of the interface in comparison to the optimal cost (green text). For each swarm size, the average cost over 100 missions comprised of the job types given, was used to calculate the optimal cost. In each mission, jobs were randomly distributed in the environment. Similarly, 100 missions with the participants' chosen swarm size were run to determine their average cost value.

After completing the planning task, participants moved on to the next portion of the trial – the monitoring task. In the monitoring task, participants were asked to watch a simulated mission using the monitoring interface (Figure 3). The mission was comprised of the job parameters that they planned for in the first part. Their task was to determine if they thought the mission was running normally or if there was a performance issue. One of three issues was possible: there were more jobs in the environment than expected when the planning was done, there were too few vehicles in the swarm to handle the workload of the jobs needing service, or there were too many vehicles in the swarm (i.e., they were unnecessary to carry out the mission effectively). In missions where extra jobs are present the jobs arrive more quickly than anticipated and the swarm drops jobs while maintaining a fairly high utilization. This is shown in Figure 4a where the Type 1 jobs by themselves appear to be arriving fairly quickly. When too few vehicles are allocated and the expected number of jobs is in fact in the environment, all of the jobs together appear to arrive at a rate that matches the length of the mission, but jobs are still dropped and the swarm utilization remains high (Figure 4b). For cases where too many vehicles are allocated, the swarm see jobs at the expected rate, but the swarm appears to be underutilized throughout the mission (Figure 4c).

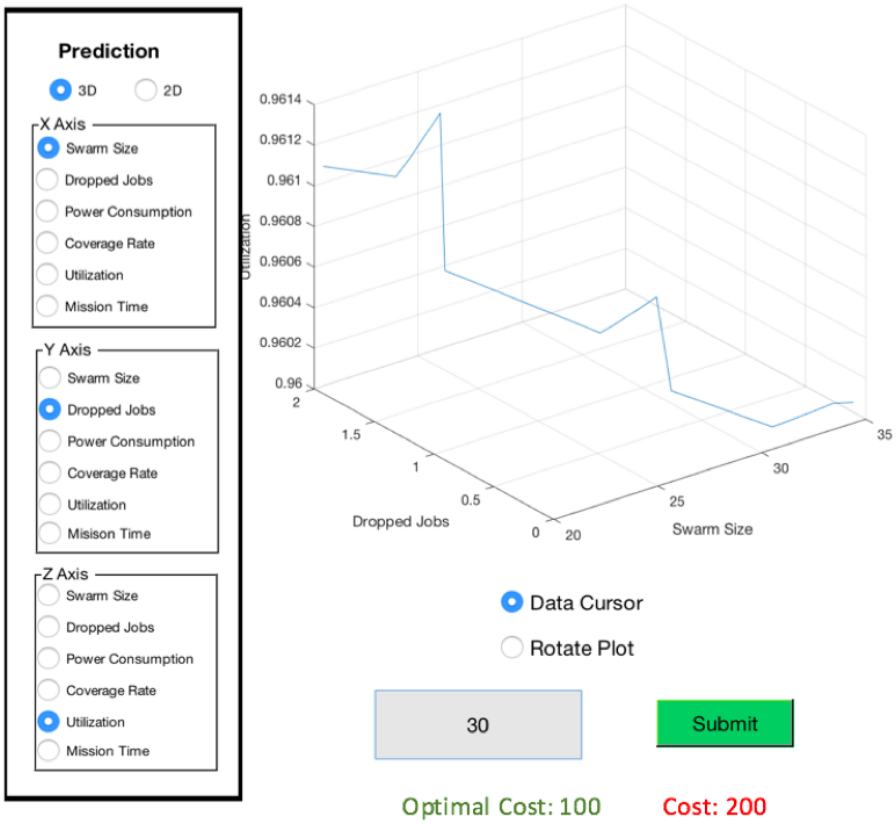


Fig. 2: Planning panel of experimental interface showing non-monotonic relation between parameters.

If the participant thought one of these issues was occurring at any time throughout the mission, they were asked to click one of the corresponding orange buttons at the bottom of the interface (Figure 3). This would indicate that they believed the mission needed to be replanned (i.e., adjusted online) and thus the mission would be paused and the trial would then be over. If they thought the mission was normal (i.e., performing such that the optimal cost would result from the mission) they simply let the mission progress until the end. This portion of the study was conducted using fixed swarm sizes to ensure that all participants saw the same missions regardless of what swarm size they chose in planning. Jobs were randomly distributed in the environment for each of the 5 trials, but all participants encountered the same job distributions.

Participants in the control group were given a tally of the number of jobs completed and missed, as well as, the percentage of area covered (Figure 3). Participants in the experimental group had access to the Hybrid Model predictions from the planning portion, allowing them to cross-reference the current mission with the expected values. The number of finished jobs, missed jobs, amount of power consumed per vehicle, and swarm utilization were tracked graphically for

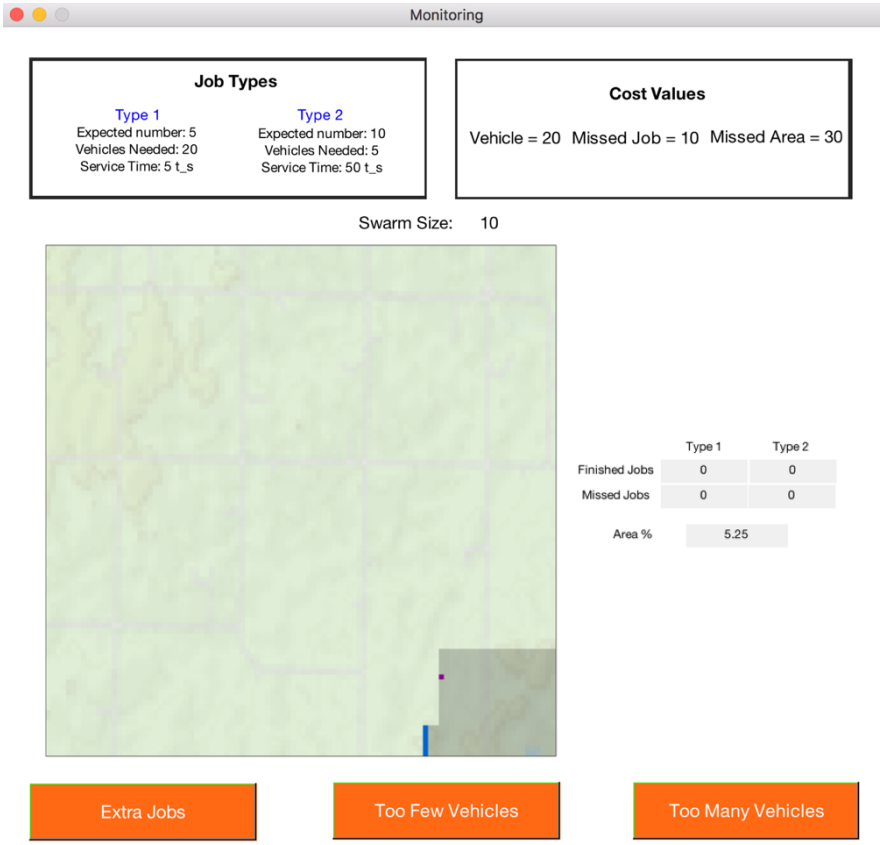
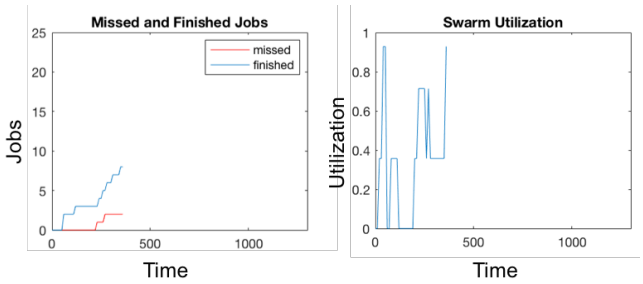


Fig. 3: Monitoring interface for control group.

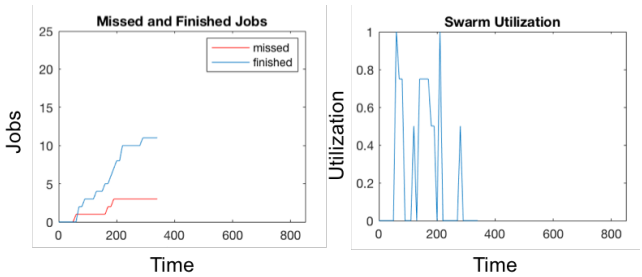
participants in the experimental group during the mission (Figure 5). Area covered is also given.

4.2 Data Collected

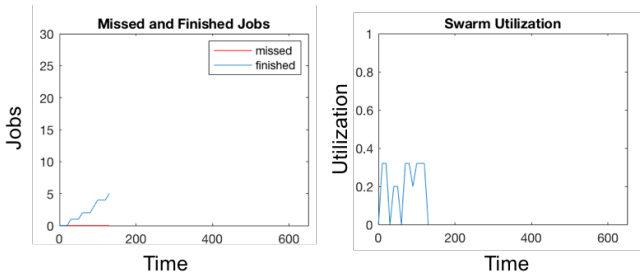
In both the planning and monitoring portions of the trials, the time taken for participants to make their decisions was recorded. For the planning task, the swarm size chosen by the participant, as well as, the associated cost was collected. During monitoring, the participants decision about the performance of the swarm was recorded. The workload felt by users across 6 categories – mental, physical, temporal, performance, effort and frustration – were collected using the standard NASA TLX survey [27].



(a) Extra Jobs



(b) Too Few Vehicles



(c) Too Many Vehicles

Fig. 4: Resulting performance indicators from suboptimal missions.

5 Results

The data for the planning and monitoring portion of the trials will be shown separately below. IBM SPSS version 25 was used to perform a one-way ANOVA with repeated measures. Three dependent variables were used for analyzing the planning results: input time, swarm size and cost. Two dependent variables were used for the monitoring analysis: time to make a decision and the number of correct decisions. For both the planning and monitoring portions, participants' group (experimental/control) was used as an independent variable. All results are reported with a significance level of $p < 0.05$. The effects between participants' assigned group and the trials will be shown. In addition, error bars will be shown on plots when appropriate.

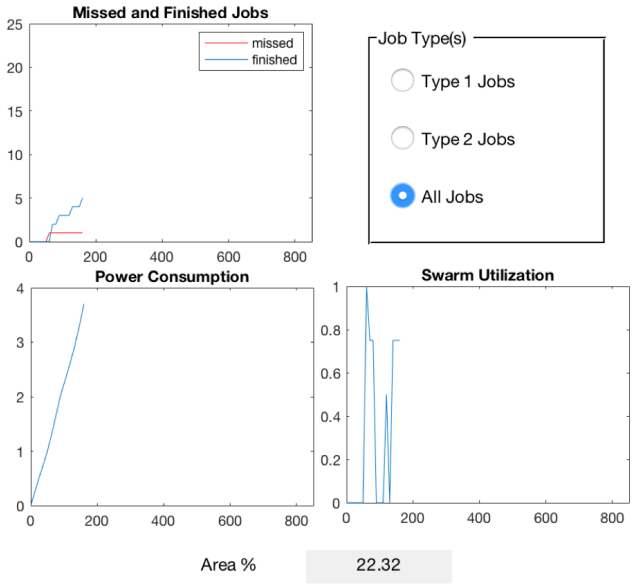
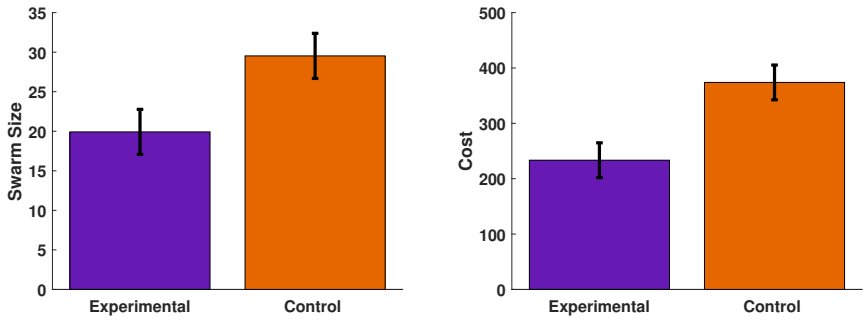


Fig. 5: Graphical parameter tracking portion of monitoring interface for experimental group.



(a) Swarm Size (b) Cost

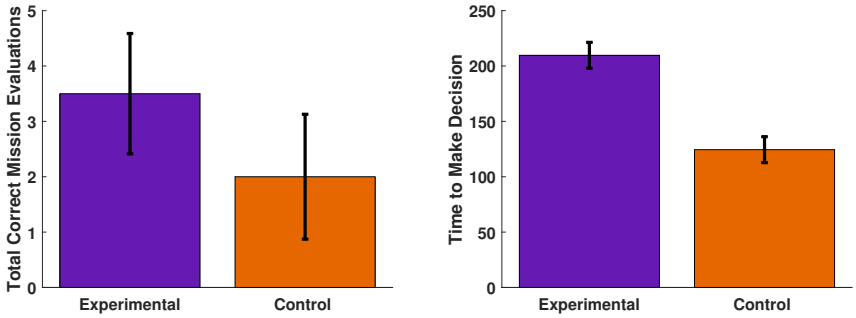
Fig. 6: Comparison of planning results by group.

Table 2: Significance Values for Planning Measures

Measure	df	F	Significance	Partial η^2
Swarm Size	1	5.70	0.026	0.206
Cost	1	10.03	0.004	0.313

5.1 Planning Task

Figure 6 shows that the participants in the experimental group who had the Hybrid Model aid were able to plan missions with lower cost and smaller swarm size. The



(a) Num. of Correct (b) Decision Time

Fig. 7: Comparison of monitoring results by group.

Table 3: Significance Values for Monitoring Measures

Measure	**df**	**F**	**Significance**	**Partial η^2**
Decision Time	1	26.46	0.000	0.546
Performance Decision	1	1.993	0.172	0.083

Table 4: Effects in Interaction Between Trial and Group

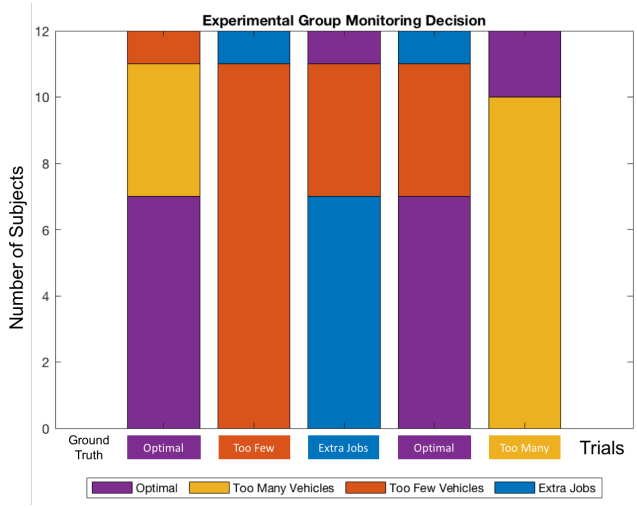
Measure	**Partial η^2**
Decision Time	0.143
Decision	0.113

results were statistically significant (Table 2). In addition, they were consistent across job parameter changes. However, participants in the control group took a shorter amount of time to choose a swarm size on average compared to the experimental group (111.82 sec versus 154.04 sec).

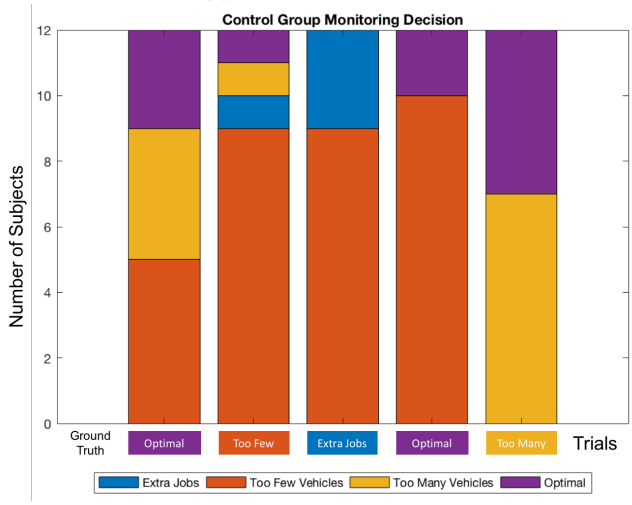
5.2 Monitoring Task

In the monitoring portion of the trials, the Hybrid Model aid enabled participants in the experimental group to determine the performance of the swarm and identify anomalies more accurately than their counterparts in the control group (Figure 7a). However, once again, they took longer to make their decision (Figure 7b). The results were statistically significant (Table 3). As seen in Table 4, participants' group had an effect on participants' ability to make monitoring decisions and was robust across trials. There was a medium-level interaction seen between trial and group for the performance decision participants.

Figure 8 shows a tally of performance decisions across trials for both groups. The ground truth decisions for the 5 monitoring trials (in order) were: (1) optimal, (2) too few vehicles, (3) too many jobs, (4) optimal, and (5) too many vehicles. They are indicated in Figure 8 with the associated ground truth color next to the trial number. As seen in Figure 8a, participants in the experimental group were able to not only able to determine if an anomaly occurred during the mission, but also distinguish between the different types of anomalies. In contrast, participants



(a) Experimental Group



(b) Control Group

Fig. 8: Accuracy in anomaly detection.

in the control group (Figure 8b) thought that unless no jobs were missed (Trial 5), the swarm had too few vehicles.

5.3 Workload Measures

Table 5 shows the average workload rating given by participants in both groups. Ratings of mental demand, physical demand, temporal demand, performance, effort and frustration are shown. Results show that participants in the experimental

group felt a lower average workload in all measures except for physical demand. The performance values were statistically significant ($F = 7.65$ and $p = 0.011$).

Table 5: NASA TLX Workload Measures

Group	Mental	Physical	Temporal	Performance	Effort	Frustration
Exp.	53.75	14.58	18.33	33.33	54.58	22.92
Ctrl.	66.67	12.08	30.42	52.50	55.42	27.92

6 Discussion

Human operators put in charge of SSS missions must successfully complete tasks that possess 3 out of the 4 sources of complexity [7]: multiple desired states, conflicting dependence among data, and uncertainty in the data. The multiple desired states are described by the various costs that the operators must minimize. The complex interdependence between mission parameters results in conflicting trade-offs that operators must balance. In addition, SSS missions have inherent uncertainties associated with unknown locations of the jobs.

In spite of the high task complexity and the lack of training trials, participants in the experimental group were able to plan missions more effectively than those in the control group. As a result, they were able to choose smaller swarm sizes and lower overall mission costs. In addition, they were able to overcome the difficulty associated with the negative correlations between attributes across the possible swarm sizes. This implies that the way that the data from the Hybrid Model predictions were displayed allowed the experimental group to not only understand the overall patterns in the data, but to also interpret the data in detail to pick a specific swarm size, unlike the visual aids given in the study conducted by Speier and Morris [42]. The control group weighed the trade-off between costs for less time than their counterparts. However, their higher overall cost and lower time to plan indicates that they developed a flawed mental model that may have been too simplistic to represent the actual interaction between mission parameters. The results were consistent across trials.

During monitoring tasks, participants in the experimental group were not only able to detect if missions were progressing optimally, but also distinguish between the various types of anomalies that arose. This indicates that participants in the experimental group were effective at cross-referencing the real-time mission parameters with the expected mission performance given by the Hybrid Model. In doing so, they were able to maintain a better situational awareness of the mission and understand more accurately what affect the mission parameter relationships had on the performance of the swarm. These results support previous work by Speier and Morris who showed that visual interfaces allowed operators to develop a deeper understanding of the data presented to them [42].

Participants in the control group tended to make quicker decisions than those in the experimental group during the monitoring tasks. As a result, they deemed missions with any number of missed jobs as one that had too few vehicles in the swarm. This once again indicates that the participants in the control group were

unable to develop an accurate mental model of the relationship between mission parameters, or understand their effect on the overall cost of the mission.

In both the planning and monitoring tasks, the study supervisor observed that participants in the experimental group rarely explored the prediction data given by the Hybrid Model in 3D. For the most part participants explored the trade-offs between parameters by comparing the performance of the swarm across multiple 2D relationships. As seen in the literature, this occurs when humans are considering complex trade-offs between possible choices. They are unable to consider all facets of the decision simultaneously and must instead do so sequentially [28]. In addition, during the monitoring task the investigator noticed that a fair number of participants (across both groups) did not notice before they made their decision that more jobs had been seen in the environment even though it was explicitly shown on the tally/graph they were given. This indicates that participants were not cross-referencing the expected parameters with the real-time feedback from the environment. Future monitoring interfaces may need to have additional alerts for deviations from expected mission parameters to help notify participants of such issues.

Although the Hybrid Model itself is built upon complex principles, the resulting low mental demand and effort for the experimental group indicate that the Hybrid Model was fairly easy to interpret. In addition, even though the prediction model aid required participants to examine and interpret a lot of additional data in comparison with the participants in the control group, their frustration level still remained lower. This once again reinforces the notion of ease-of-use for the Hybrid Model-based aid. Lastly, the ease-of-use of the aid allowed participants in the experimental group to accurately evaluate their performance as being better than those in the control group. This was indicated by the experimental group having a lower average performance workload measure than the control group.

7 Conclusion

This paper presented a user study that aimed to test the efficacy and ease-of-use of a prediction model as a planning and monitoring aid for SSS missions with dynamically changing arrival rates. Results show that participants who had use of the Hybrid Model predictions were able to plan missions that required fewer vehicles and resulted in lower overall cost. Participants without the aid developed a more simplistic mental model of the relationship between mission parameters enabling them to make quicker decisions, but whose inaccuracy lead to higher cost. During monitoring portion, the Hybrid Model allowed participants to maintain an accurate situational awareness of the real-time mission performance. This resulted in their ability to identify optimal missions, as well as, distinguish between mission issues that arose. Participants with the aid maintained a lower workload in all measures except physical demand, emphasizing the ease-of-use of the aid.

References

1. URL www.neyasystems.com/robotic-mission-planning-and-management/

2. Arkin, R.C., Collins, T.R., Endo, Y.: Tactical mobile robot mission specification and execution. In: Mobile Robots XIV, vol. 3838, pp. 150–164. International Society for Optics and Photonics (1999)
3. Arslan, G., Marden, J.R., Shamma, J.S.: Autonomous vehicle-target assignment: A game-theoretical formulation. Journal of Dynamic Systems, Measurement, and Control **129**(5), 584–596 (2007)
4. Bertsimas, D., Weismantel, R.: Optimization over integers, vol. 13. Dynamic Ideas Belmont (2005)
5. Brumitt, B.L.: A mission planning system for multiple mobile robots in unknown, unstructured, and changing environments. Carnegie Mellon University (1998)
6. Bullo, F., Frazzoli, E., Pavone, M., Savla, K., Smith, S.L.: Dynamic vehicle routing for robotic systems. Proceedings of the IEEE **99**(9), 1482–1504 (2011)
7. Campbell, D.J.: Task complexity: A review and analysis. Academy of management review **13**(1), 40–52 (1988)
8. Chandarana, M., Hughes, D., Lewis, M., Sycara, K., Scherer, S.: Hybrid model for a priori performance prediction of multi-job type swarm search and service missions. In: 2019 19th International Conference on Advanced Robotics (ICAR). IEEE (2019)
9. Chandarana, M., Lewis, M., Allen, B.D., Sycara, K., Scherer, S.: Swarm size planning tool for multi-job type missions. In: 2018 Aviation Technology, Integration, and Operations Conference, p. 3846 (2018)
10. Chandarana, M., Lewis, M., Sycara, K., Scherer, S.: Determining effective swarm sizes for multi-job type missions. In: 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 4848–4853. IEEE (2018)
11. Chen, J.Y., Barnes, M.J., Qu, Z.: Roboleader: An agent for supervisory control of multiple robots. In: Proceedings of the 5th ACM/IEEE international conference on Human-robot interaction, pp. 81–82. IEEE Press (2010)
12. Chen, J.Y., Barnes, M.J., Quinn, S.A., Plew, W.: Effectiveness of roboleader for dynamic re-tasking in an urban environment. Proceedings of the Human Factors and Ergonomics Society Annual Meeting **55**(1), 1501–1505 (2011). DOI 10.1177/1071181311551312. URL https://doi.org/10.1177/1071181311551312
13. Chen, J.Y., Barnes, M.J., Quinn, S.A., Plew, W.: Effectiveness of roboleader for dynamic re-tasking in an urban environment. In: Proceedings of the Human Factors and Ergonomics Society Annual Meeting, vol. 55, pp. 1501–1505. SAGE Publications Sage CA: Los Angeles, CA (2011)
14. Clare, A.S., Cummings, M.L., How, J.P., Whitten, A.K., Toupet, O.: Operator object function guidance for a real-time unmanned vehicle scheduling algorithm. Journal of Aerospace Computing, Information, and Communication **9**(4), 161–173 (2012). DOI 10.2514/1.I010019. URL https://doi.org/10.2514/1.I010019
15. Cummings, M.L., Mitchell, P.J.: Predicting controller capacity in supervisory control of multiple uavs. IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans **38**(2), 451–460 (2008)
16. Dias, P.S., Gomes, R.M., Pinto, J.: Mission planning and specification in the neptus framework. In: Robotics and Automation, 2006. ICRA 2006. Proceedings 2006 IEEE International Conference on, pp. 3220–3225. IEEE (2006)
17. Endo, Y., MacKenzie, D.C., Arkin, R.C.: Usability evaluation of high-level user assistance for robot mission specification. IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews) **34**(2), 168–180 (2004)
18. Farinelli, A., Iocchi, L., Nardi, D.: Distributed on-line dynamic task assignment for multi-robot patrolling. Autonomous Robots **41**(6), 1321–1345 (2017)
19. Fernández Perdomo, E., Cabrera Gámez, J., Domínguez Brito, A.C., Hernández Sosa, D.: Mission specification in underwater robotics (2010)
20. Fong, T., Thorpe, C.: Vehicle teleoperation interfaces. Autonomous robots **11**(1), 9–18 (2001)
21. Friedman, S.L., Scholnick, E.K.: The developmental psychology of planning: Why, how, and when do we plan? Psychology Press (2014)
22. Goldberg, D., Cicirello, V., Dias, M.B., Simmons, R., Smith, S., Stentz, A.: Task allocation using a distributed market-based planning mechanism (2003)
23. Goodrich, M.A., McLain, T.W., Anderson, J.D., Sun, J., Crandall, J.W.: Managing autonomy in robot teams: observations from four experiments. In: Proceedings of the ACM/IEEE international conference on Human-robot interaction, pp. 25–32. ACM (2007)

24. Griffith, J.D., Kochenderfer, M.J., Moss, R.J., Mišić, V.V., Gupta, V., Bertsimas, D.: Automated dynamic resource allocation for wildfire suppression. Lincoln Laboratory Journal **22**(2) (2017)
25. Guestrin, C., Koller, D., Parr, R.: Multiagent planning with factored mdps. In: Advances in neural information processing systems, pp. 1523–1530 (2002)
26. H, W., Hsu, B., Velagapudi, P., Scerri, P., Sycara, K.: Choosing autonomy modes for multirobot search. Human Factors (special issue on Decision Making in Complex Environments), 52(2), 225-233 (2010)
27. Hart, S.G., Staveland, L.E.: Development of NASA-TLX (task load index): Results of empirical and theoretical research. Advances in psychology **52**, 139–183 (1988)
28. Hastie, R., Dawes, R.M.: Rational choice in an uncertain world: The psychology of judgment and decision making. Sage (2010)
29. Humphrey, C.M., Henk, C., Sewell, G., Williams, B.W., Adams, J.A.: Assessing the scalability of a multiple robot interface. In: Proceedings of the ACM/IEEE International Conference on Human-robot Interaction, HRI '07, pp. 239–246. ACM, New York, NY, USA (2007). DOI 10.1145/1228716.1228749. URL http://doi.acm.org/10.1145/1228716.1228749
30. Korsah, G.A., Stentz, A., Dias, M.B.: A comprehensive taxonomy for multi-robot task allocation. The International Journal of Robotics Research **32**(12), 1495–1512 (2013)
31. Lee, D.H.: Resource-based task allocation for multi-robot systems. Robotics and Autonomous Systems **103**, 151 – 161 (2018). DOI https://doi.org/10.1016/j.robot.2018.02.016
32. Luo, L., Chakraborty, N., Sycara, K.: Distributed algorithm design for multi-robot task assignment with deadlines for tasks. In: International Conference on Robotics and Automation (ICRA), Karlsruhe, Germany (2013)
33. MacKenzie, D.C., Arkin, R.C., Cameron, J.M.: Multiagent mission specification and execution. In: Robot colonies, pp. 29–52. Springer (1997)
34. Macwan, A., Vilela, J., Nejat, G., Benhabib, B.: A multirobot path-planning strategy for autonomous wilderness search and rescue. IEEE transactions on cybernetics **45**(9), 1784–1797 (2014)
35. Mathew, N., Smith, S.L., Waslander, S.L.: Multirobot rendezvous planning for recharging in persistent tasks. IEEE Transactions on Robotics **31**(1), 128–142 (2015)
36. Michalewicz, Z., Fogel, D.B.: Why are some problems difficult to solve? In: How to Solve It: Modern Heuristics, pp. 11–30. Springer (2000)
37. Miller, C.A., Parasuraman, R.: Beyond levels of automation: An architecture for more flexible human-automation collaboration. In: Proceedings of the Human Factors and Ergonomics Society Annual Meeting, vol. 47, pp. 182–186. SAGE Publications Sage CA: Los Angeles, CA (2003)
38. Miller, C.A., Parasuraman, R.: Designing for flexible interaction between humans and automation: Delegation interfaces for supervisory control. Human factors **49**(1), 57–75 (2007)
39. Nagavalli, S., Chakraborty, N., Sycara, K.: Automated sequencing of swarm behaviors for supervisory control of robotic swarms. In: 2017 IEEE International Conference on Robotics and Automation (ICRA) (2017)
40. Portugal, D., Rocha, R.P.: Cooperative multi-robot patrol with bayesian learning. Autonomous Robots **40**(5), 929–953 (2016)
41. Portugal, D., Rocha, R.P.: Performance estimation and dimensioning of team size for multirobot patrol. IEEE Intelligent Systems **32**(6), 30–38 (2017)
42. Speier, C., Morris, M.G.: The influence of query interface design on decision-making performance. MIS quarterly pp. 397–423 (2003)
43. Wagner, A.R., Endo, Y., Ulam, P., Arkin, R.C.: Multi-robot user interface modeling. In: Distributed Autonomous Robotic Systems 7, pp. 237–248. Springer (2006)
44. Wagner, G., Choset, H.: Subdimensional expansion for multirobot path planning. Artificial Intelligence **219**, 1–24 (2015)
45. Yu, L., Cai, Z., Gao, P., Liu, X.: A spatial orthogonal allocation and heterogeneous cultural hybrid algorithm for multirobot exploration mission planning. Journal of Control Theory and Applications **9**(2), 171–176 (2011). DOI 10.1007/s11768-011-9109-3. URL https://doi.org/10.1007/s11768-011-9109-3
46. Ziparo, V.A., Iocchi, L.: Petri net plans. In: Proceedings of Fourth International Workshop on Modelling of Objects, Components, and Agents (MOCA), pp. 267–290 (2006)