Swarm Size Planning Tool for Multi-Job Type Missions

Meghan Chandarana *
Carnegie Mellon University, Pittsburgh, PA 15213, USA

Michael Lewis†
University of Pittsburgh, Pittsburgh, PA 15260, USA

B. Danette Allen ‡
NASA Langley Research Center, Hampton, VA 23681, USA

Katia Sycara § and Sebastian Scherer ¶
Carnegie Mellon University, Pittsburgh, PA 15213, USA

As part of swarm search and service (SSS) missions, swarms are tasked with searching an area while simultaneously servicing jobs as they are encountered. Jobs must be immediately serviced and can be one of multiple types. Each type requires that vehicle(s) break off from the swarm and travel to the job site for a specified amount of time. The number of vehicles needed and the service time for each job type are known. Once a job has been successfully serviced, vehicles return to the swarm and are available for reallocation. When planning SSS missions, human operators are tasked with determining the required number of vehicles needed to handle the expected job demand. The complex relationship between job type parameters makes this choice challenging. This work presents a prediction model used to estimate the swarm size necessary to achieve a given performance. User studies were conducted to determine the usefulness and ease of use of such a prediction model as an aid during mission planning. Results show that using the planning tool leads to 7x less missed area and a 50% cost reduction.

I. Nomenclature

π	=	size of swarm
i	=	job type requiring i vehicles
R_π	=	probability of dropped a job given swarm size π
δ	=	accepted number of dropped jobs
s	=	system state
S	=	set of all system states s
$P_\pi(s)$	=	probability of being in a given state s given swarm size π
k	=	number of servers in system
ρ_i	=	utilization factor of job type i
λ_i	=	arrival rate of job type i
μ_i	=	service rate of job type i
n_i	=	number of jobs of type i
C	=	normalizing constant
$r_\pi(s)$	=	probability of dropping a job given state s
$\mathcal{D}$	=	set of all states that would cause the system to drop at job given its current state s
λ_d	=	arrival rate of a job that would cause the system to drop a job
R_π	=	total drop job rate

*PhD Candidate, Dept. of Mechanical Engineering, 5000 Forbes Ave., Pittsburgh, PA AIAA Student Member
†Professor of Informatics and Distributed Systems and Intelligent Systems, 135 North Bellefield Ave., Pittsburgh, PA 15260
‡Senior Technologist for Intelligent Flight Systems, Crew Systems & Aviation Operations Branch, MS 492, AIAA Senior Member
§Research Professor, Robotics Institute, 5000 Forbes Ave., Pittsburgh, PA 15213
¶Systems Scientist, Robotics Institute, 5000 Forbes Ave., Pittsburgh, PA 15213

II. Introduction

In swarm search and service (SSS) missions robot swarms are tasked with searching a specified area while simultaneously servicing jobs as they are sensed. For many applications jobs must be immediately serviced. Failure to do so may result in catastrophic consequences. In cases where the swarm is tasked with monitoring wildfire containment, for example, failing to send vehicles to douse a newly ignited brush fire could result in the loss of wildfire containment. As a direct consequence, additional area may be burned, negatively affecting local populations and wildlife.

Each newly sensed job requires the swarm to break off and send a vehicle or group of vehicles to the job site for a specific amount of time. The number of vehicles and the service time are determined by the type of job that is sensed. Once the job has been successfully serviced, the vehicles return to the swarm and are available for reallocation to future job sites. If not enough vehicles are present in the swarm to service a job that has been sensed, the job is dropped and will not be serviced. In many applications, multiple job types may be present within the environment. Although the expected number of each type may be known ahead of time, each job's exact location is only known when the swarm comes within sensing range.

Figure 1 shows an example SSS mission where the swarm is searching an area effected by a recent natural disaster. Within the environment in this search and rescue context, the swarm sees 3 different job types: fires, people trapped in crumbling buildings and injured people. Each job type requires the indicated number of vehicles and service times to complete them. When many jobs appear at once (as shown in the middle of the swarm's search path) jobs are dropped because not enough vehicles are present in the swarm.

The work presented in this paper aims to improve overall SSS mission performance. We specifically explore how the developed swarm size prediction model can be utilized by human operators to more effectively chose swarm sizes for SSS missions. The evaluation of a mission's efficacy provides metrics for trust and trustworthiness in multi-agent team interactions, as well as, a basis for the certification of autonomous systems. Building this basis is the focus of Autonomy Teaming and TRAjectories for Complex Trusted Operational Reliability (ATTRACTOR), a new NASA Convergent Aeronautics Solution (CAS) project. The remainder of the paper describes and analyzes a user study designed to examine the effect of the human operator's utilization of the swarm size prediction model on mission performance. A discussion of the results is presented, as well as, their implications for future swarm mission planning.

The remainder of the paper is outlined as follows. Section III reviews current related work. Section IV introduces a dynamic vehicle routing framework for SSS missions and presents the dynamic vehicle routing problem with time constraints applied to SSS missions. Section V presents the swarm size prediction model that solves the dynamic vehicle routing problem with time constraints. Section VI describes the interfaces developed to allow human operators to utilize the predictive model for planning SSS missions. Section VII outlines the user study conducted. The results

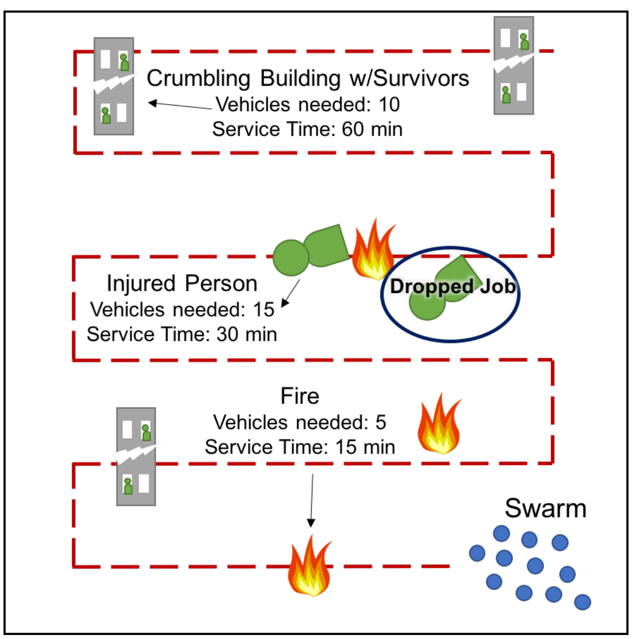


Fig. 1 Representative SSS mission which includes 3 different job types.

and discussion are given in Sections VIII and IX respectively. Finally, Section X provides concluding remarks.

III. Related Work

Current multi-agent mission planning interfaces focus on prescribing explicit vehicle paths for each vehicle throughout the duration of the mission. They assume known job site parameters and locations. In addition, traditional mission planning interfaces assume some predefined swarm size. Typically, this size is the maximum number of vehicles available to operators [1].

One of the most common types of mission planning interfaces used for multi-agent systems is ecological interfaces. They are derived from work in ecological visual perception and allow operators to define control actions based on graphical representations of abstract system data rather than the explicit data [2], thereby reducing operator workload [3]. Ecological interfaces were originally developed for use in nuclear power generation [4] and petro-chemical refinement [5], but were later extended for use in multi-robot control interfaces [6, 7].

Multi-agent mission planning interfaces range in autonomy level from fully manual to fully autonomous. Fully manual interfaces require that operators manually input mission area of interest, targets and step-by-step vehicle maneuvers. However, they do abstract away the controllers necessary to accomplish the desired maneuvers [8, 9]. Fully autonomous interfaces use a variety of frameworks to autonomously plan the paths of all the vehicles including Markov Decision Process frameworks [10], game theoretic [11], and integer programming [12]. In addition, grammar based planning [13], task based planning [14], behavior based planning [15, 16] and Petri Net based planning [17] have been used in autonomous multi-agent mission planning.

IV. Dynamic Vehicle Routing for Swarms

In dynamic vehicle routing (DVR) problems, vehicles are assigned to jobs as they arrive to the system during execution. Rather than finding specific routes for each vehicle, solutions are found by defining vehicle routing policies which can be used to assign a vehicle or group of vehicles to service a job as it arrives to the system. Within the context of SSS missions, the arrival rate of jobs is defined as the time required for a swarm vehicle to travel within sensing range of the job. Individual arrival rates for each job type can be determined. We assume that jobs are uniformly and randomly distributed within the environment. Jobs arriving to the system can be thought of as customers arriving to a queue. Since jobs must be immediately serviced or they are dropped, the queue is always empty. Algorithmic queuing theory is used to analyze the steady state performance of these SSS mission systems.

For the remainder of the paper we consider the DVR problem with time constraints. Bullo et al. define this problem as finding the minimum number of vehicles needed to successfully service the expected incoming job demands with a given steady state probability [18]. In SSS missions where we require jobs to be serviced immediately, the DVR problem with time constraints is seen as finding the minimum number of vehicles to allocate to the swarm so that fewer than an accepted number of dropped jobs is seen:

$$\min_{\pi} |\pi|, \text{subject to} \lim_{i \to \infty} R_{\pi} \leq \delta \tag{1}$$

where π is the number of vehicles allocated to the swarm, R_π is the probability of the system dropping a job upon its arrival due to lack of available vehicles, i is the job type and δ is the accepted number of dropped jobs.

V. Swarm Size Prediction Model

The steady state performance of an SSS mission system can be modeled as an infinite horizon M/M/k/k queuing system where jobs enter the system as they are sensed by swarm vehicles. For uniformly and randomly distributed jobs, constant velocity swarm, and systematic, non-redundant search, this leads to jobs arriving according to a Poisson distribution. In addition, service times are considered exponentially distributed. The system contains k servers. These servers are synonymous with the vehicles allocated to the swarm at the start of the mission. As compared to more commonly seen M/M/k systems where there is no limit on the size of the queue, the size of the queues in an M/M/k/k system is at most k in length. For SSS systems, jobs can require the use of multiple servers simultaneously and are of varying types. Each job type arrives at a different rate λ_i, where i is the job type and requires a different number of servers. Type-i jobs require i vehicles to service it. Each job has an exponential service time, $Exp(\mu_i)$. Once the job has been serviced, all vehicles are free to be used in the system for future jobs.

The probability that an M/M/k/k system is in a given state $s = (n_1, ..., n_k)$, where n_i is the number of jobs of Type-i is defined as [19, 20]:

$$P_\pi(s) = \prod_{i=1}^{k} \frac{\rho_i^{n_i}}{n_i!} \cdot C \tag{2}$$

$$C = \left(\sum_{s \in \mathcal{S}} \prod_{i=1}^{k} \frac{\rho_i^{n_i}}{n_i!} \right)^{-1} \tag{3}$$

$$\rho_i = \frac{\lambda_i}{\mu_i} \tag{4}$$

where $P_\pi(s)$ is the probability of the system being in a state s given an initial swarm size π. C is a normalizing factor. ρ_i is known as the utilization factor. We define $\mathcal{S}$ as the set of all possible states that the system can be in. In SSS missions, $\mathcal{S}$ is composed of states where the combination of jobs results in the system being as utilized as possible. In such systems, if no jobs of Type-i exist, all states with $n_i > 0$ are removed from $\mathcal{S}$.

The probability of the system not having enough vehicles to service an incoming job (i.e., the probability that a job will be dropped) given a prescribed swarm size π can then be defined as:

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

The set of all jobs which would make the system fail (i.e., drop a job) for a system state s is defined as $\mathcal{D}$. λ_d is the arrival rate of a job type that would cause the system to drop the job. The total drop job rate, R_π, is then defined as:

$$R_\pi = \sum_{s \in \mathcal{S}} r_\pi(s). \tag{6}$$

Figure 2 shows an example of the relationship derived between the swarm size and the number of dropped jobs using the described M/M/k/k model. In this example, 3 job types were expected. 15 jobs of Type 1 were expected, 5 of Type 2 and 15 of Type 3. Type 1 required 5 vehicles to service it for 100 seconds, while Type 2 and Type 3 needed 10 vehicles for 100 seconds and 15 vehicles for 100 seconds respectively. The blue points represent the values predicted from the model. An exponential curve (red) is fit to the data ($R^2 = 0.9674$).

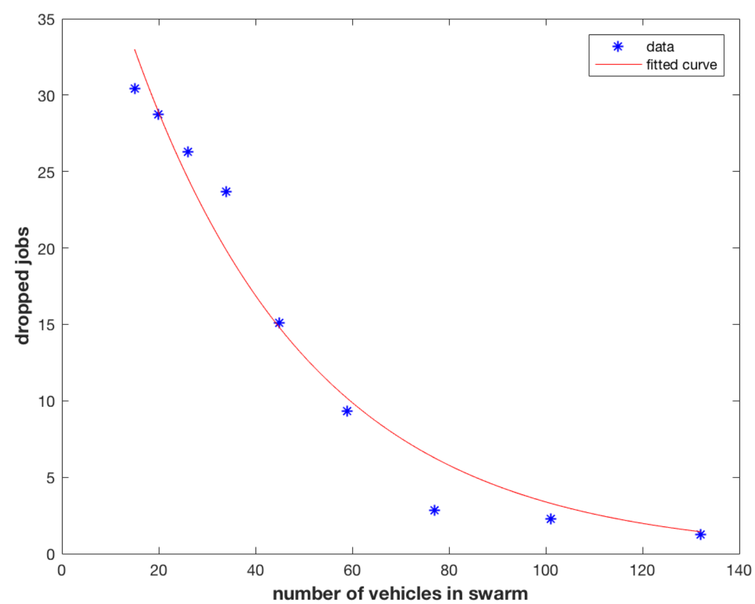
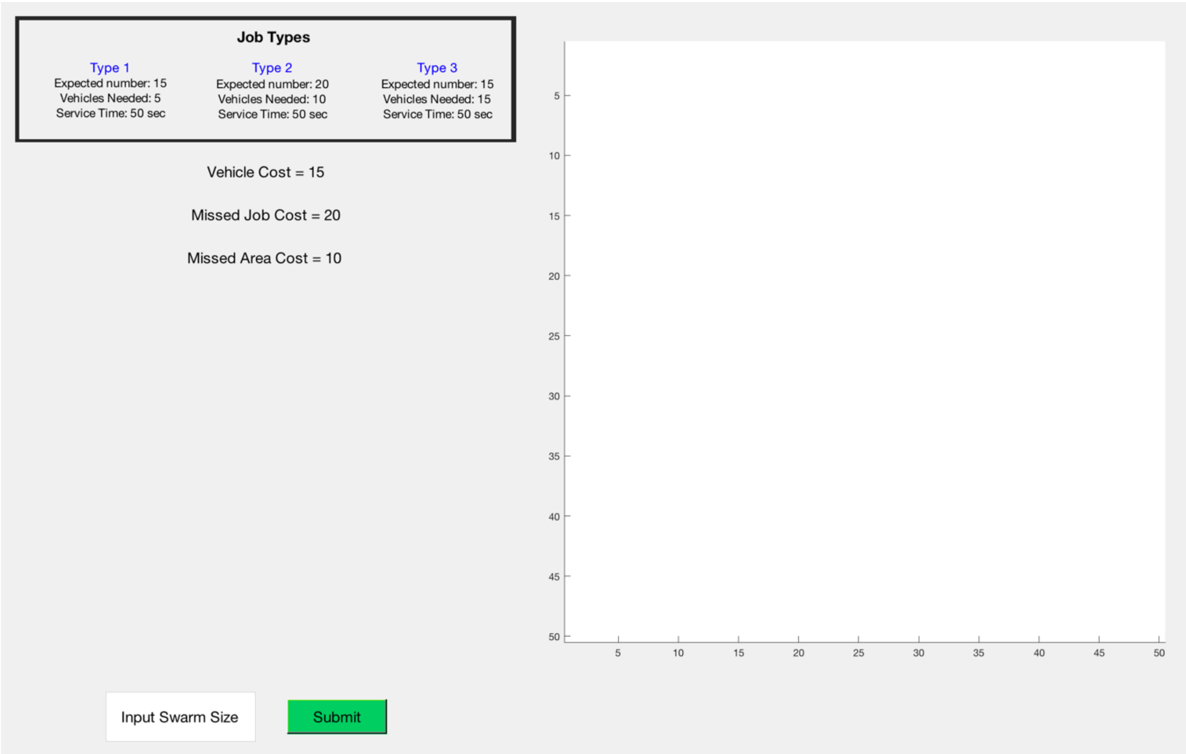
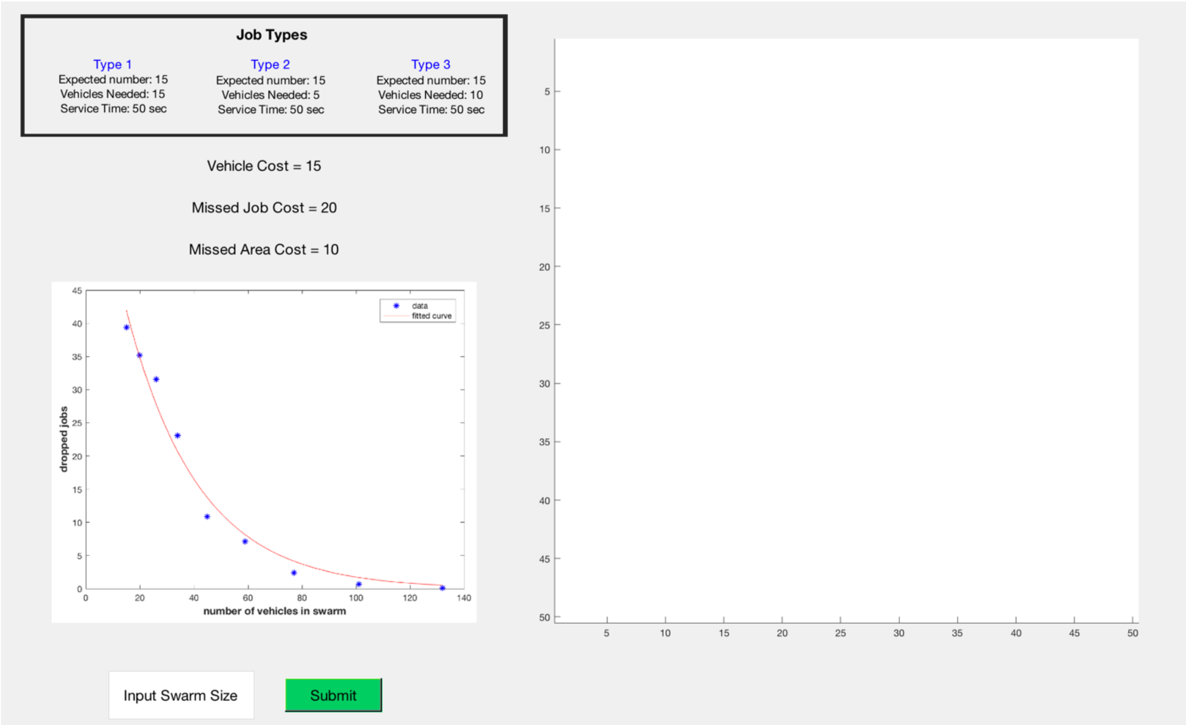


Fig. 2 Example relationship between swarm size and number of dropped jobs using the M/M/k/k model.



(a) Control Interface



(b) Experimental Interface

Fig. 3 SSS mission planning interfaces utilized in user study.

VI. Mission Planning Interface

To test the efficacy and usefulness of the developed swarm size prediction model, two user interfaces were developed (Figure 3). Both interfaces provide users with SSS job type mission parameters for 3 different job types. These parameters include the expected number of jobs of each type that will be present in the environment, as well as, the required service times and number of vehicles needed to successfully service jobs of each type. One of the interfaces provides an additional graph showing the calculated relationship between the number of dropped jobs versus the starting size of the swarm given by the predictive model (Figure 3b). The relationship varies with the job type parameters provided. The second interface provides only the job type parameters. In both interfaces users are able to specify the size of a swarm for the given SSS mission parameters. In addition to the job type parameters provided, each interface also provides users with the cost values associated with each vehicle assigned to the swarm, as well as, the number of dropped jobs and missed area seen during the mission.

Upon hitting submit, a grid world simulation of a mission with the given job type parameters and the input swarm size is shown (Figure 4). Within the 50x50 world, the swarm (pink) traverses 1x1 grid cells in a lawn mower pattern. The positions of the job sites are uniformly and randomly distributed each time. Each job site is color coded to indicate which of the 3 types it is. Job type 1 is green, type 2 is yellow and type 3 is blue. Uncovered area is shown in white, while area that the swarm has covered is shown in black. When the swarm reaches a job site, if enough vehicles are available, they are allocated. The site remains active and colored until it is finished being serviced. After this point the job disappears and the vehicles are added back to the swarm. If not enough vehicles are available, the job site turns red, indicating that it is a dropped job site. At any point if all of the vehicles have been allocated and the swarm is no longer able to search, gray grids are shown for locations that have been missed and would have been searched by the swarm if vehicles were still present. The simulation terminates when all the area has been covered. After the simulation is complete the cost for the mission is displayed.

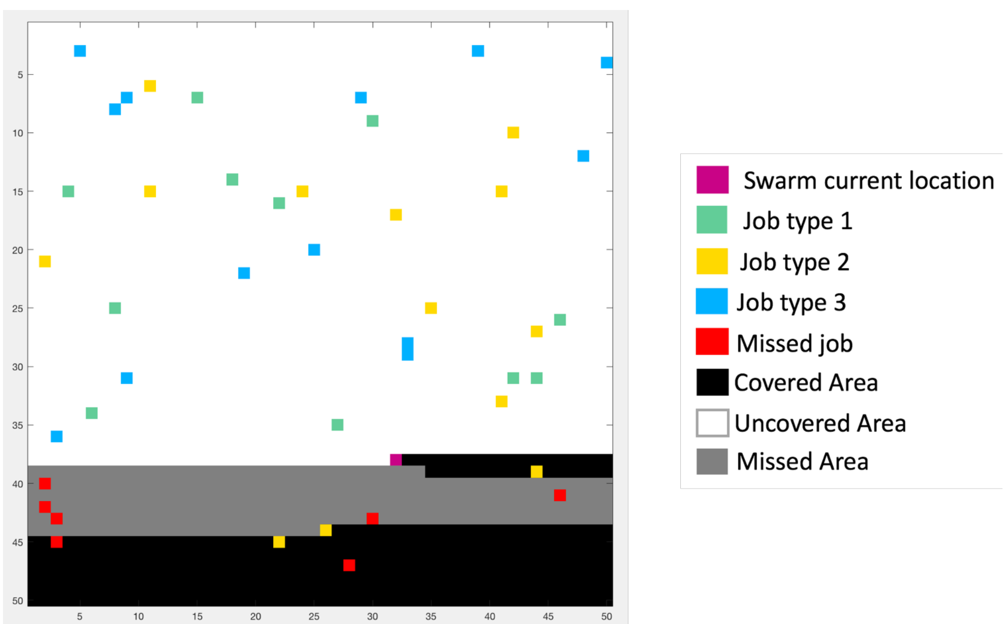


Fig. 4 Screen shot of the grid world simulation shown to a user as feedback after choosing a swarm size.

VII. Experimental Design

A total of 20 subjects took part in the user study. The subjects were evenly split between the control group and the experimental group. All 20 subjects participated in a total of 10 trials. Subjects placed in the experimental group utilized the interface with the additional predictive model data (Figure 3b), while those in the control group used the interface without the model (Figure 3a).

Across the 10 trials the job parameters provided to subjects for each of the 3 types varied. Table 1 shows the parameters for each trial. n is the expected number of jobs for that type that will be present in the environment. V_n is the

Table 1 Job Type Parameters for Each Trial

	Type 1			Type 2			Type 3		
Trial	**n**	V_n	μ_i **(sec)**	**n**	V_n	μ_i**(sec)**	**n**	V_n	μ_i**(sec)**
1	5	15	25	10	5	50	15	10	30
2	5	15	25	5	5	50	5	10	30
3	15	15	50	15	5	50	15	10	50
4	5	15	100	10	5	50	15	10	30
5	5	5	100	10	10	50	15	15	30
6	5	5	50	10	10	50	15	15	50
7	15	5	50	20	10	50	15	15	50
8	15	5	100	10	10	100	5	15	100
9	15	5	100	5	10	100	15	15	100
10	15	10	30	5	15	100	15	15	50

required number of vehicles that will be needed to service the job for μ_i seconds. All subjects were given the parameter sets in the same order. For subjects in the experimental group, the graph displaying the relationship between the starting swarm size and the expected number of dropped jobs varied with the job parameters. Using the information provided, subjects were asked to specify a swarm size that they believed would result in 5 dropped jobs or fewer such that the total cost for the mission was minimized. Cost values of 15, 20 and 10 were chosen for each vehicle's cost, each drop job's cost and each grid cell missed respectively.

For a given mission the total cost consisted of an individual cost for each vehicle allocated to the swarm, as well as the cost associated with dropping a job. In addition, if all the vehicles are ever allocated at once and the swarm is empty and unable to continue searching, a cost for each grid cell that would have been searched if vehicles were still present in the swarm is added to the total cost. This is meant to represent missions where search time is limited and the swarm is unable to return to the unsearched area at a later point.

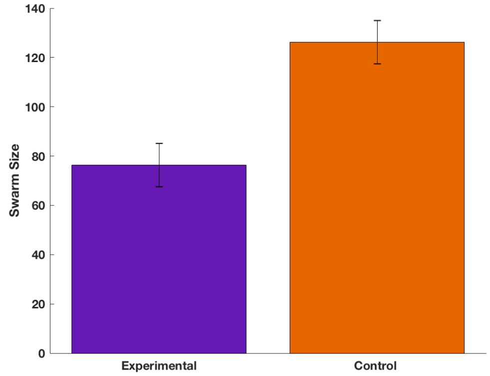
For each subject the following protocol is used. Before beginning the trials, subjects are asked to sign a consent form and fill out a background questionnaire. Then, they were shown a video which provided an overview of the SSS missions, as well as, an explanation of the information they would be provided with and their task. Subjects then completed the 10 trials. During each trial the following data was collected: (1) time taken to input swarm size, (2) chosen swarm size, (3) number of dropped jobs, (4) number of missed grid cells, and (5) the total mission cost. All data except for the input time was generated from running the simulation with the chosen swarm size and the given job type parameters.

VIII. Results

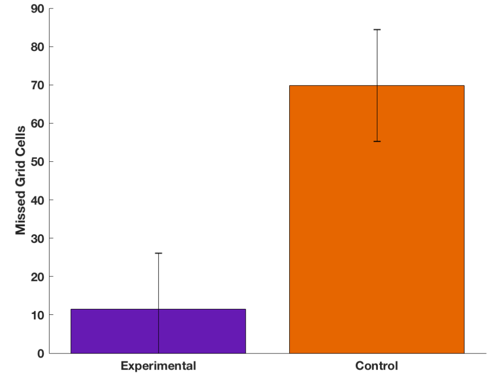
The results shown here are taken from the 10 trials conducted for each subject. The swarm size values reported are the values input by the subject in each of their trials. All remaining data is taken as a result of the simulation run with the given job type parameters, the subject's choice of swarm size and uniformly and randomly generated job site locations. A one way ANOVA with repeated measures was conducted on the data using IBM SPSS version 24. Subject's assigned group (control or experimental) were the independent variables. Input time, swarm size, number of missed grid cells, number of dropped jobs and cost were used as dependent variables. Results are reported using a significance level of $p < 0.05$. Error bars for the standard error of the means are shown in all plots.

Figure 5 shows the comparison between the experimental group (those with use of the predictive model graphs) and the control group. The results for subject chosen swarm size, as well as, the simulation's resulting number of missed grid cells, dropped jobs and total cost are shown. Overall subjects who had use of the predictive model chose a lower swarm size (Figure 5a) and maintained a lower total cost (Figure 5d). Subjects in the control group maintained a lower number of dropped jobs (Figure 5c), but had a higher missed area (Figure 5b). All results are statistically significant (Table 2). Differences in input time were not significant.

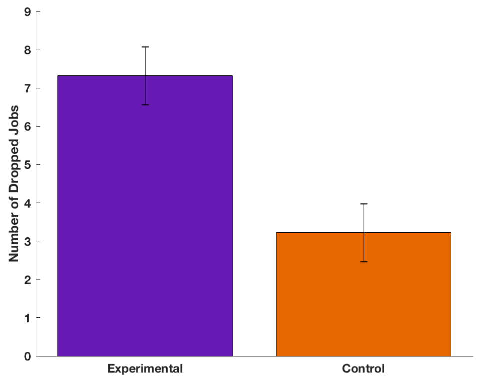
Table 3 shows an effect was seen in the interaction between trials and group on subjects' chosen swarm size, as well as, the trials' resulting number of dropped jobs and cost. No effect was seen on the amount of missed area. Figure 6 (left) shows that this effect is a result of the experimental group learning to almost eliminate any missed area, while



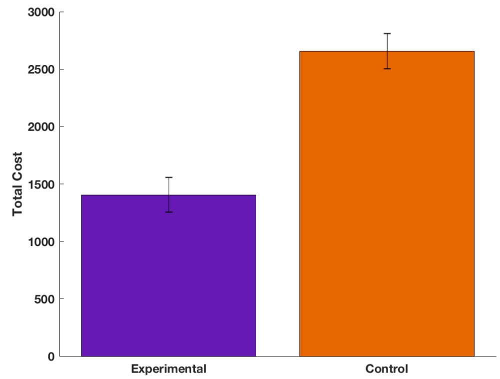
(a) Swarm Size



(b) Missed Grid Cells



(c) Dropped Jobs



(d) Cost

Fig. 5 Comparison of results given the subject's group (experimental vs. control).

Table 2 Significance Values

Measure	df	F	Significance	Partial η^2
Swarm Size	1	16.253	0.001	0.474
Number of Missed Grid Cells	1	7.962	0.011	0.307
Number of Dropped Jobs	1	14.624	0.001	0.488
Cost	1	34.062	0.000	0.654

the control group never learns a strategy for reducing their amount of missed area. In addition, the average number of dropped jobs across trials increases for the experimental group, whereas the control group's number of dropped jobs fluctuates around a range (Figure 6 right).

IX. Discussion

Overall, the results show that when using the additional prediction model tool, subjects were able to plan missions with better performance than their counterparts who did so without the tool. Their consistently higher performance, even without a training session, indicates that the tool was fairly intuitive and easy to use. Subjects were able to effectively use the tool as a baseline for making their choice of swarm size given the job type parameters and mission cost values. The cost values, job type parameters and desired drop rate provided a metric for determining where along the curve

Table 3 Effects Seen in Interaction Between Trial and Group

Measure	Partial η^2
Swarm Size	0.144
Number of Missed Grid Cells	0.047
Number of Dropped Jobs	0.234
Cost	0.145

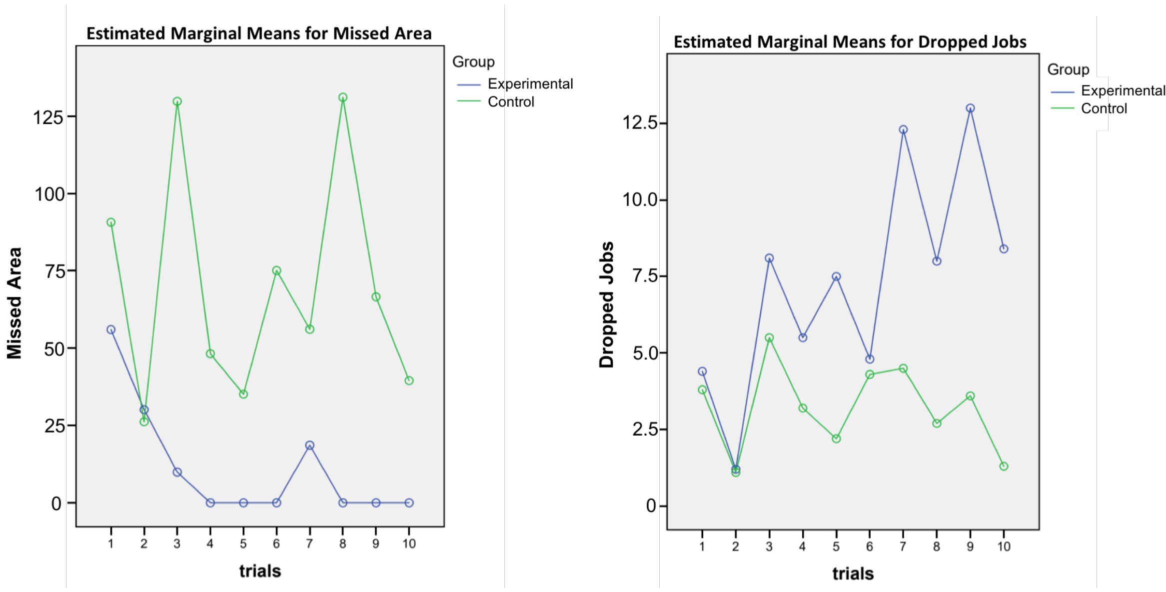


Fig. 6 Average number of missed grid cells and dropped jobs across trials for both groups.

subjects should be when making their swarm size choice. The results were statistically significant with partial η^2 values indicating that the difference was in fact robust.

Analysis shows that subjects in the control group generally prioritized dropped jobs over missed area leading to fewer dropped jobs than those in the experimental group. However, on average, control participants still had a larger extent of missed area and overall cost. In part this may be due to their choice of larger swarm sizes. This is also an indication that subjects within the control group did not look at the parameters of the job types closely enough to realize that by giving certain swarm sizes, they were still likely to have a combination of jobs that would lead to all the robots being occupied.

In contrast, subjects in the experimental group realized that by adding an extra 1 or 2 vehicles (e.g., 57 instead of 55) they were able to ensure that no combination of jobs would ever lead to all the robots being occupied (Figure 6 left), leading to 7x less missed area on average. Subjects in this group also prioritized total cost over the number of dropped jobs. Given the cost of vehicles was about the same as a dropped job, they chose to allocate fewer vehicles to the swarm even though this meant that more jobs would be dropped as a direct result (Figure 6 right). However, in doing so, they were able to reduce their overall cost by 50% as compared to the subjects in the control group.

X. Conclusion

This paper presented a prediction model for determining the swarm size in multi-job type swarm search and service missions. User study results show that the subjects who used the model were able to plan missions that resulted in better overall performance, fewer used vehicles, and a lower unsearched area than those who planned the mission without the tool. However, the choice of cost values resulted in a higher dropped jobs. The results were statistically significant and the effect sizes were fairly large, indicating that the differences in the data was in fact a result of the differences between the two groups. Future work will explore the effect of changing the ratio between cost values. In addition, mission planning for swarms with varying formations will be explored.

Acknowledgments

This work presented in this paper was sponsored by NASA LaRC (NIA Activity 201020).

References

[1] Nehme, C. E., Crandall, J. W., and Cummings, M., "An operator function taxonomy for unmanned aerial vehicle missions," *12th international command and control research and technology symposium*, 2007.

[2] Gibson, J. J., *The ecological approach to visual perception: classic edition*, Psychology Press, 2014.

[3] Kilgore, R., and Voshell, M., "Increasing the transparency of unmanned systems: Applications of ecological interface design," *International Conference on Virtual, Augmented and Mixed Reality*, Springer, 2014, pp. 378–389.

[4] Itoh, J., Sakuma, A., and Monta, K., "An ecological interface for supervisory control of BWR nuclear power plants," *Control Engineering Practice*, Vol. 3, No. 2, 1995, pp. 231–239.

[5] Jamieson, G. A., and Vicente, K. J., "Ecological interface design for petrochemical applications: supporting operator adaptation, continuous learning, and distributed, collaborative work," *Computers & Chemical Engineering*, Vol. 25, No. 7-8, 2001, pp. 1055–1074.

[6] Linegang, M. P., Stoner, H. A., Patterson, M. J., Seppelt, B. D., Hoffman, J. D., Crittendon, Z. B., and Lee, J. D., "Human-automation collaboration in dynamic mission planning: A challenge requiring an ecological approach," *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, Vol. 50, SAGE Publications Sage CA: Los Angeles, CA, 2006, pp. 2482–2486.

[7] Furukawa, H., "An ecological interface design approach to human supervision of a robot team," *Autonomous Robots and Agents*, Springer, 2007, pp. 163–170.

[8] Dias, P. S., Gomes, R. M., and Pinto, J., "Mission planning and specification in the Neptus framework," *Robotics and Automation, 2006. ICRA 2006. Proceedings 2006 IEEE International Conference on*, IEEE, 2006, pp. 3220–3225.

[9] Arkin, R. C., Collins, T. R., and Endo, Y., "Tactical mobile robot mission specification and execution," *Mobile Robots XIV*, Vol. 3838, International Society for Optics and Photonics, 1999, pp. 150–164.

[10] Guestrin, C., Koller, D., and Parr, R., "Multiagent planning with factored MDPs," *Advances in neural information processing systems*, 2002, pp. 1523–1530.

[11] Arslan, G., Marden, J. R., and Shamma, J. S., "Autonomous vehicle-target assignment: A game-theoretical formulation," *Journal of Dynamic Systems, Measurement, and Control*, Vol. 129, No. 5, 2007, pp. 584–596.

[12] Bertsimas, D., and Weismantel, R., *Optimization over integers*, Vol. 13, Dynamic Ideas Belmont, 2005.

[13] Brumitt, B. L., *A mission planning system for multiple mobile robots in unknown, unstructured, and changing environments*, Carnegie Mellon University, 1998.

[14] Fernández Perdomo, E., Cabrera Gámez, J., Domínguez Brito, A. C., and Hernández Sosa, D., "Mission specification in underwater robotics," 2010.

[15] Miller, C. A., and Parasuraman, R., "Beyond levels of automation: An architecture for more flexible human-automation collaboration," *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, Vol. 47, SAGE Publications Sage CA: Los Angeles, CA, 2003, pp. 182–186.

[16] MacKenzie, D. C., Arkin, R. C., and Cameron, J. M., "Multiagent mission specification and execution," *Robot colonies*, Springer, 1997, pp. 29–52.

[17] Ziparo, V. A., and Iocchi, L., "Petri net plans," *Proceedings of Fourth International Workshop on Modelling of Objects, Components, and Agents (MOCA)*, 2006, pp. 267–290.

[18] Bullo, F., Frazzoli, E., Pavone, M., Savla, K., and Smith, S. L., "Dynamic vehicle routing for robotic systems," *Proceedings of the IEEE*, Vol. 99, No. 9, 2011, pp. 1482–1504.

[19] Arthurs, E., and Kaufman, J., "Sizing a message store subject to blocking criteria," *Proceedings of the third international symposium on modelling and performance evaluation of computer systems: Performance of computer systems*, North-Holland Publishing Co., 1979, pp. 547–564.

[20] Harchol-Balter, M., *Performance modeling and design of computer systems: queueing theory in action*, Cambridge University Press, 2013.