2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
October 25-29, 2020, Las Vegas, NV, USA (Virtual)

Generating Alerts to Assist With Task Assignments in Human-Supervised Multi-Robot Teams Operating in Challenging Environments

Sarah Al-Hussaini[1], Jason M. Gregory[2], Yuxiang Guan [1], Satyandra K. Gupta[1]

Abstract— **In a mission with considerable uncertainty due to intermittent communications, degraded information flow, and failures, humans need to assess both the current and expected future states, and update task assignments to robots as quickly as possible. We present a forward simulation-based alert system that proactively notifies the human supervisor of possible, negatively-impactful events, which provides an opportunity for the human to retask agents to avoid undesirable scenarios. We propose methods for speeding up mission simulations and extracting alerts from simulation data in order to enable real-time alert generation suitable for time-critical missions. We present the results from a user trial and verify our hypothesis that the decision making performance of human supervisors can be improved by introducing forward simulation-based alerts.**

I. Introduction

Multi-robot systems are becoming more adept to serve alongside humans in real-world missions as researchers make advancements in fundamental capabilities and resilience [1]. These types of systems will be especially useful for dangerous missions in complex environments, e.g., military operations and disaster response, because of their ability to increase standoff distances and reduce the risk to humans. In the context of multi-robot missions, we anticipate that human teammates will serve in a supervisory role to manage resources and make critical decisions because of the gravity of events, including life-or-death situations [2]. At a high-level, human supervisors will ensure that robots are continuously working on tasks that align with the mission objectives, and issue commands that account for the dynamically-changing context.

The events, circumstances, and outcomes of decisions in humanitarian-assistance and disaster-relief missions are severely consequential and complex in nature, which has a direct impact on a human's ability to supervise a multi-robot team. Large unstructured environments, restrictive communications, and the possibility of system- and task-level failures introduce uncertainty in the availability and efficiency of agents, and significant delay in the receipt of mission-critical information. In such missions, it is of the utmost importance that humans make the best use of the available information and constantly adapt their strategy and update task assignments to improve mission performance. However, perceiving the mission situation correctly from some incomplete or possibly-outdated information is a challenging task. Furthermore, these issues are exacerbated if humans are cognitively- and emotionally-fatigued, which could lead to the issuing of slow-paced or ill-conceived commands to the robotic teammates.

[1]Viterbi School of Engineering, University of Southern California, CA USA {salhussa, yuxiangg, guptask}@usc.edu
[2]CCDC Army Research Laboratory, Adelphi, MD USA jason.m.gregory1.civ@mail.mil

Alerts can serve as an effective means to prevent human-introduced inefficiency and mistakes, and can speed up the decision making process. Alerts are already being used in numerous technologies and application domains because of they offer measurable value. For example, lane departure warnings and blind spot detection systems alert drivers of potential collisions [3], which leads to prudent decision-making for safe driving. We feel that an intelligent alert system can provide tremendous benefit to human-supervised robot teams operating in time-sensitive, safety-critical scenarios.

Generating useful alerts for real-world, multi-agent systems requires making predictions on the future states of a mission, referred to as *forward simulation*. An obvious consideration is to use mission simulations, along with the latest updates from each robot, to make probabilistic estimations. In a time-critical mission, alerts will be beneficial only when they are generated in real-time. Therefore, we need to develop appropriate simulation models that reduce the computation time for alert generation.

In this work, we introduce an alert generation framework with a discrete-event, forward simulation model enhanced with smart features to significantly speed up computation. The traces from mission simulations are compared with user-customizable, mathematically-encoded alert conditions to automatically generate alerts in real-time. We conducted user tests and verified, with statistical significance, that alerts can make a significant improvement in the performance of human decision making.

II. Related Work

User-friendly, man-machine interfaces are important for communicating commands and information in critical missions, and ubiquitous computing in real-world applications is a research goal at present [4]. Researchers have focused on interface design to control the robots [5], the associated human factor concerns [6] [7], and system development [8] specifically for search-and-rescue missions.

There exist several different alert-generating architectures and interfaces for controlling robots, such as an augmented reality-based solution for collaborative assembly [9] and alert systems used by NASA in aviation [10]. There have also been several alert systems designed specifically for human-robot teams in the disaster response context [11] [12]. However, all these alerts are purely reactive, where a warning appears to the human once an undesirable event has occurred, e.g., navigation change, obstacle alert, etc. In [13], some predicted interface designs are depicted, where a trained neutral net predicts the operator's evaluation on risk and relevance of a robot performing some task. Since all these works are based on small-scale operation where humans are essentially controlling the robots directly, real-time alerts are generated solely using current information like sensor data. In large-scale missions, with intermittent, incomplete and delayed information flow, a fieldable system must make future mission predictions based on past or incomplete data.

Here, we seek a proactive approach for providing alerts based on probabilistic outcomes which can not be known for certain yet, and might not be immediate from the known states. Forward simulations can make predictions on possible future findings, and improve decision-making, which has been utilized in other applications [14]. In the context of military and disaster response, researchers are actively pursuing simulators for environments and applications, and simulators are getting progressively more accurate in these harsh environments compared to real-world results [15] [16]. This motivates us to develop forward simulation-based alert generation for our application, and explore the computational costs and benefits of the generated alerts.

III. System overview

We have developed an alert generation framework for human-supervised, multi-robot teaming applications in large unstructured environments, e.g., disaster response missions and military operations. In these scenarios, humans dispatch a team of robots into the operational environment to explore the affected regions efficiently, collect mission-critical information, and perform certain tasks. The humans provide the robots complete task plans which may include a nominal task sequence, along with some interrupt tasks, and contingency task plans. Thus the high-level mission strategy is crafted by the human supervisors to meet some mission objectives. Meanwhile, the robots are designed to be capable of making the low-level decisions for performing the tasks given by the humans, such as navigation and exploration, identification and manipulation of objects of interest, and execution of other mission-specific tasks.

The entire system architecture is depicted in Figure 1, which includes the alert generation framework from our earlier work [17]. Humans receive and view mission updates from the ongoing mission, via a user interface. This mission update, along with current task plan of robots in the ongoing mission, and human-specified unwanted situations to trigger alerts, are then fed into the alert generation framework. We have outlined in [17] the mathematical framework and representation of different alert triggering conditions as Metric Temporal Logic (MTL) formulae, the complex task description structure of robots, and the state machine representation of robot behavior to be used in mission simulation. In the current paper, we focus on mission- and task-models for forward simulation, computational speed up techniques, alert extractions, and finally, the usefulness of humans being notified of the alerts via the interface while assigning tasks to the robots.

In a large-scale environment, there are usually some regions of higher importance which we call *areas-of-interest* (AoIs). We assume that the human supervisors use their expertise and protocols to identify these regions and mission goals are set with priorities given to searching or performing tasks in the AoIs. We also assume limited communication in this mission, which causes intermittent or degraded data flow to humans. The humans can only receive mission updates in some time intervals depending upon the communication constraints, instructions given to the robots, and how the mission progresses. As the robots operate in the complex mission space, there is a non-zero probability of failure, which can be due to environmental factors like complex terrain, or stochastic events like hardware or software failures. In some cases, if a robot is disabled or immobilized, we assume that another robot might be able to revive the disabled robot by providing assistance. We call this task *robot rescue* [18] and the difficulty and risk of these rescue operations depends on the specific situation.

The human supervisors issue task allocations to the robots from a command center, which is not necessarily co-located with the robots. We assume that the humans have some method by which they can effectively communicate with the robots and monitor mission progress using some display. Display modalities could include computer monitors, tablets, augmented or virtual reality displays, or any other interface by which humans receive mission-relevant updates from the robots. In order to be most effective, the humans need to adapt their strategy as they receive information from the robots. Whenever a robot becomes available to be tasked, the humans must make informed decisions about how to best use the asset to accomplish the mission.

For every real-world mission, there is likely a set of critical situations that a human supervisor is concerned about. Some of these events may be unwanted situations that are detrimental to the team or mission performance and, in the best case, are prevented or mitigated through improved decision making. However, intermittent data flow makes this process very challenging because humans will receive information at a delayed time. Whenever there is new mission information available it is crucial that humans assess the mission situation and formulate a mental model for how the mission has been progressing since the past updates and how it will progress in near future. This way humans will be able to allocate available resources appropriately.

We propose a framework to assist the human supervisor with informed decision making in an effort to overcome the burden of mission modeling and mitigate the negative impact of delayed information flow due to intermittent communication. We introduce forward simulation-based alert generation, which we believe will be very useful for the human commanders of multi-robot teams because it will reduce the cognitive load required to manage teams. The forward simulation uses updates from the past to predict what is currently happening outside of communications range, or

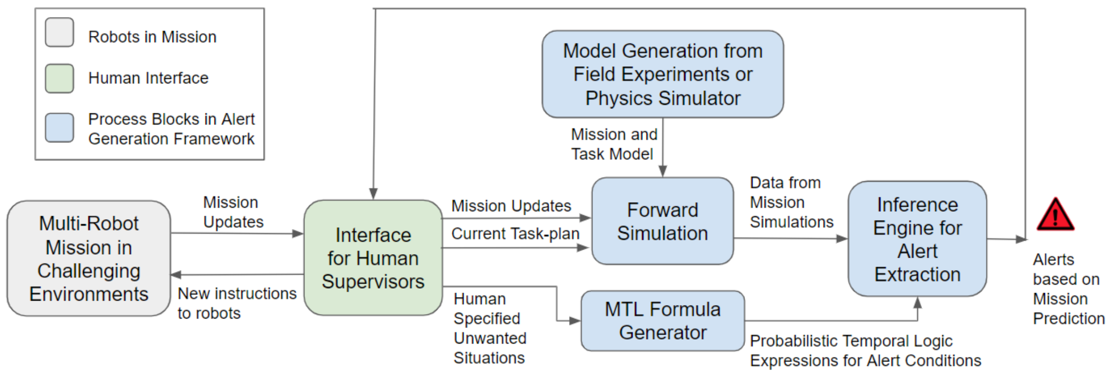


Fig. 1: System architecture proposed in this work that builds on our previous work [17].

what will happen in near future. Human supervisors provide the mission conditions that they value and want notifications for as to ensure that the alerts provided to the human are relevant and not distracting. These conditions are expressed mathematically in a probabilistic temporal logic framework [19]. Forward simulations of the mission generate traces of information, which are compared to the alert conditions set by the humans. Whenever a condition is evaluated to be true, an alert message with additional details is provided to the human commanders. This, in turn, improves the humans' understanding of mission progression and helps them to prioritize important issues and resources.

IV. Speeding up forward simulations

Forward simulations are needed to estimate how the mission will progress, and to generate alerts regarding whether any unwanted situations are likely to occur. Each robot initiates a simulation using the most up-to-date information they have received and computes for some simulated time into the future. Some robots whose latest updates are further into the past, need to be simulated for a longer period of time which may be on the order of hours. These simulations will produce data on the distribution of robot locations over time and other mission parameters.

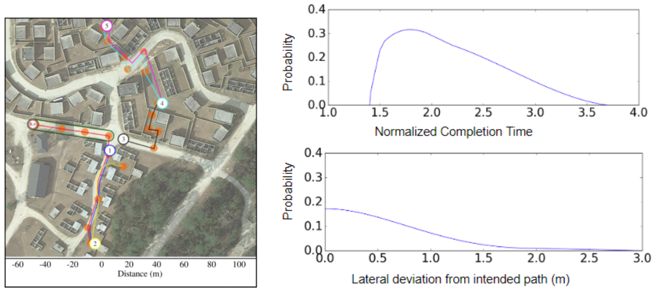
A physics-based simulator could be used to perform Monte-Carlo runs for the mission; however, simulating a couple of hours of tasks for a team of robots can be very time-consuming. Available physics engines typically operate with constant, very small (millisecond) time steps. For example, 30 simulated minutes of navigation for a single Husky[1] driving in a straight line requires 8.12 seconds using the PyBullet[2] physics-based simulator on an Intel Xeon CPU E3-1245 v5, 3.50GHz processor. This example assumes the simplest scenario of one robot on a flat surface using a default friction model and the largest suggested time step (10 ms) indicated in the software documentation. Extrapolating this example to a team of 10 robots in a long-duration mission, a single run for a navigation task will require more than 4 minutes. A realistic mission will require several kinds of complex tasks, update many variables, and use hundreds, possibly thousands, of simulations for a single instance of probabilistic state estimation and mission outcome prediction. Therefore, conventional simulation techniques will be prohibitively slow. In the following subsections, we propose ways to expedite this forward simulation process to generate alerts within a reasonable amount of time for time-critical missions.

A. Discrete event based simulation model

We use a discrete event-based simulation paradigm with fixed-increment time progression for forward simulating the missions. So, the operation of the system is modelled as a discrete sequence of events in time. The states of the robots are updated at every time step according to the instructions given to the robots and mission updates. To help mitigate computational cost, we supplement the discrete event simulation with representative task performance models, which are generated offline before the actual mission begins and then used in forward simulations during mission execution.

The enabling technology in our discrete event simulation is the task performance models. Because this is a pre-processing step, these can be constructed from physics-based simulations or real-world experimentation with robots performing relevant tasks in similar environments. To demonstrate the feasibility of this approach, we built distributions for the normalized completion time and position-based error of autonomous navigation using data from a real-world field experiment, described in our earlier work [20]. Each mission in the experiment consists of a ground robot autonomously navigating from one precise location to another, given no a-priori information, through a variety of terrains in a complex, urban setting (Figure 2 (a)). This navigation task resembles the conditions of what a robot could encounter in military or disaster relief operations. For each successful navigation, we computed the amount of additional time required to complete the navigation, relative to the Euclidean path, which encapsulates the uncertainty in the observed speed of the robot in an unknown environment. We also constructed a distribution of the position-based error by computing the distance between the robot's observed position and the corresponding positions

[1]https://clearpathrobotics.com/husky-unmanned-ground-vehicle-robot/
[2]https://pybullet.org/wordpress/



(a) Navigation Missions

(b) Empirical distributions for normalized completion time and position uncertainty

Fig. 2: Generating distributions of navigation parameters for discrete event simulation using data from field experiments.

in the commanded global plan. The empirical distributions for normalized completion time and position uncertainty for autonomous navigation are shown in Figure 2 (b). One could then use these distributions by sampling each distribution at each time step of the discrete event simulation to determine a possible speed and error that produce the next location of the robot performing the navigation task.

Since discrete event simulation is being done at a high level, we can use larger time steps without loss of performance; we typically would not intervals smaller than 30 seconds. We tested alert generation for four representative scenarios (more details in Sections V and VI). We found that alert generation using 100 simulations took between $3.95-4.76$ seconds with a time step of 30 seconds, which is at least tens of thousands times faster than running physics-based simulation. Even though this computation time may seem small enough for usage, a mission model with higher complexity might have significantly more computational cost. Therefore, further improvement in computational time would be beneficial.

B. Variable time-step

The goal of our discrete event simulation is to explore the possible ways a mission could progress and identify when certain situations occur, if at all. Therefore, we need sufficiently small time steps in order to capture the salient phenomena. The traditional way of doing discrete event simulation is to run the simulation using a constant time step, and identifying an appropriate time step is crucial. Smaller time steps ensure higher fidelity at the expense of higher computational cost, while the low-fidelity with larger time step may cause some events of interest to be overlooked.

We propose a variable time step to reduce the number of steps in the discrete event simulation, while generating sufficient, representative mission data. In applicable missions, there are usually some time intervals for each robot when it operates in accordance with its nominal plan with no external events or interactions, and the *interesting phenomena* occur during other time periods. To identify this for each individual robot, we perform 10 initial simulated runs with a small, constant time step to identify the less-consequential time periods where we can use a larger time step for updating its status, and other times where we require smaller time steps. It is important to identify what the *interesting* or *consequential* items are to be searched in those initial simulations. In some of our representative mission scenarios, interesting events included robot-to-robot interactions, robotic failures and rescues, detection of objects of interest, and any external event. We produce some preliminary results using a two-tier structure of time steps, dt (30 seconds) and $10dt$ (5 minutes), in our adaptive time simulation. For the four scenarios, the two-tier adaptive method achieved between $(3.15-5.70)\times$ speed up in computation process as compared to using constant dt time step. In the future, we plan to perform hierarchical time steps, so that we can support multiple different step sizes instead of only two.

V. Extracting alerts from simulation data

The data generated by forward simulations need to be compared with the alert triggering conditions to issue alerts. This section denotes a few potential unwanted situations and relevant alert conditions, mathematically encoded in a probabilistic logic framework. One representative mission scenario is used as an example, and we demonstrate how simulation data are processed in our scheme. Time-dependent probability traces for certain mission variables are produced from a series of forward simulations. Then alerts are issued by checking whether the associated temporal logic formulas are found to be $true$ from the traces.

Based on each mission objectives, human commanders may care about different situations. Some alerts can be based on whether there are or will be sufficient number of robots in a region of need. For example, new findings can make some regions of higher priority than others, which was not initially known, and still not known to many robots in the field, so humans may want to track whether the high priority regions have a sufficient number of robots operating there. If there are survivors involved, humans may want to check whether survivors are getting sufficient assistance. If alerts are given, humans can prioritize those regions early by allocating new resources. Alerts can be also useful when humans do not want robots in certain regions any more. If humans receive information on some risky regions, they might prefer other robots not to navigate around that region to avoid potential failures. Another example might be, if one robot unknowingly plans to explore a region that another robot has completely explored already, time and energy will be wasted. These future unwanted situations can be avoided if humans send an available robot to those robots of concern, and update their instructions. In some cases, it might be worthwhile to track different mission parameters, e.g., some robot's status related information like failure, or rescue operations. The specific situations that humans prioritize will be dependent on the mission details, objectives, and updates.

One of the example mission scenarios we constructed, which can be representative of an actual mission, is shown in Figure 3. There is a few kilometer square sub-urban

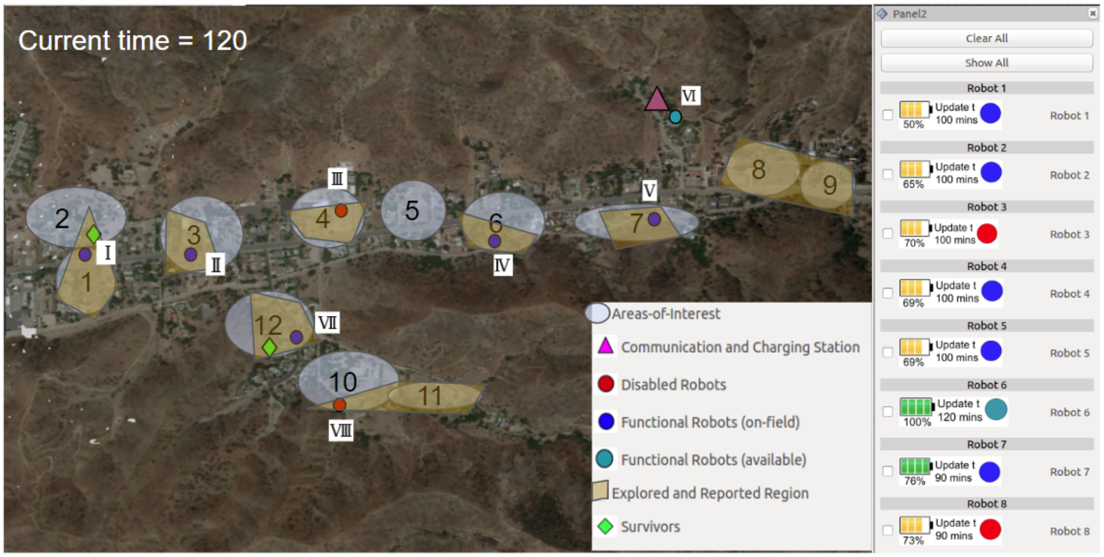


Fig. 3: An example representative mission scenario with some reported information

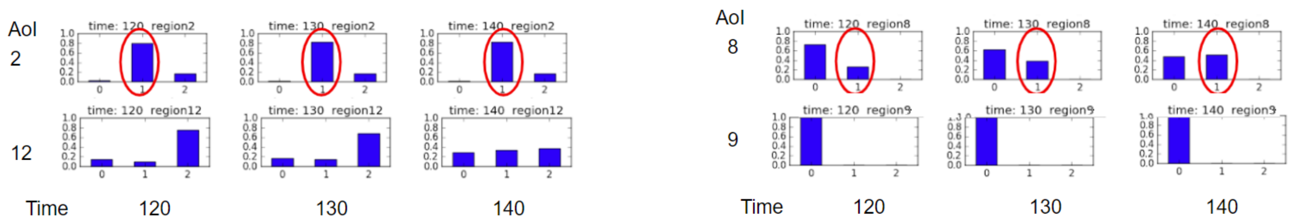
environment with 12 *Areas-of-Interest*, and a team of eight robots, I-VIII. The goal of the mission is to explore all these AoIs and find as many survivors as possible. The robots also need to assist each survivor based on the person's needs. There are some assumptions for this operation. Firstly, if there is one survivor found in a region, there is higher probability to find more survivors. Secondly, one robot may not be sufficient to provide the necessary aid to a survivor, and two robots will be needed in such cases. The instructions given to the robots in the field are in accordance with these assumptions and mission goals. A robot can get another robot from the same or nearby region to receive more assistance if needed. Also, robots need not explore a region if it is already explored and reported by another robot.

Under these assumptions, we can specify two alert triggering conditions which are relevant in this mission: (1) high probability for any survivor found not getting enough assistance, (2) redundant exploration by a robot. In this mission, robots are sent out with some initial task plans, and a couple of hours have passed since the start. In Figure 3 we can see that at current time, 120 minutes, robot VI has returned to the communication station after finishing exploration of AoIs 8 and 9, and is ready to receive new instructions. All other robots are still on the field performing their tasks based on their instructions and information states. Each robot has a last update time, and status from that time. We can also see that one survivor has been found in regions 2 and 12 separately. Since one survivor means more survivors will likely be found in a region, and most survivors need two robots, human commanders would want at least two robots in AoI 2 and 12. Also, humans would not want any robot in regions 8 and 9 since they are already explored. Thus, both the noted alert conditions can be tested from the number of operating robots in specific regions. The more precise alert conditions for this scenario can be as follows: (1) high probability of AoIs $2, 12$ (with identified survivors) having less than 2 operating robots, (2) possibility of any robot operating in AoIs $8, 9$; both conditions for present time or recent future (time $\in [120, 140]$ min). In our framework, these conditions are mathematically encoded using the framework provided in our previous work [17].

Our forward simulation uses all the reported information on initial states of the robots, and given instructions, and simulates the mission. We have more interest in knowing about the number of operating robots in regions $2, 8, 9, 12$ specifically, due to survivors needs or to prevent redundant exploration. If we perform 100 runs of forward simulation, we generate data for 100 possible mission outcomes throughout the time duration. We aggregate the results for the mission variables of our interest, in this case the number of functioning robots in each region. We create a probability distribution for the number of robots operating in a particular region at a specific time. The number of robots in a region can be between 0 to 2. Thus, we generate the plots given in Figure 4, and the unfavorable situations are marked in red. We can see that there is high (about 0.8) probability that region 2 does not have enough robots, which triggers the first alert. In order to handle this situation, an available robot can be tasked to that area and relay a retasking message. Secondly, there is a probability of $0.20 - 0.40$ of a robot operating in AoI 8, which triggers an alert for redundant exploration. To reflect to this second alert, humans need to check another mission variable, each robot's operating area ID. This is to see which robot goes to AoI 8, so that the human supervisor can update its instructions by sending the available robot to it.

VI. User Study

We conducted a preliminary human study to evaluate the merits of forward simulation-based alert generation. We identified mission scenarios that could be of interest in the context of human-supervised robot teams, and built a user



(a) Regions which should have least 2 operating robots in recent times, for the survivors

(b) Regions which should not have any robots to avoid redundant exploration

Fig. 4: Plots generated from forward simulation of mission scenario in Figure 3, showing probability distribution of having $0, 1, 2$ operating robot(s) in particular regions at specific times. The discrete times in the figures span from present time to $20 - 30$ minutes into the future, as required by the time window in alert conditions. The red markings denote concerning situations where the number of operating robots are not favorable.

interface for human commanders to view information for these missions. The participants in our user study were provided sufficient introduction and training, and then were asked to assume the role of commander, use the interface, assess the mission situation, and strategize on new task assignment for one or two available robots. Each participant served for four missions and in two of these we offered alert generation with forward simulation-based prediction information. The purpose of our human study is to quantify if, and how effectively, forward simulations might a) help the humans to understand the mission situations better and b) guide them to make more informed decisions to facilitate performance improvement.

A. Hypothesis on Performance Improvement with Alerts

We hypothesize that we can improve performance of human commanders by providing prediction-based alerts and information. We believe that there will always be a small percentage of people who are intellectually sharp enough to make reasonable decisions using only reported data, without alerts or forward simulation. But the remaining larger pool of people will likely be overwhelmed with the interrelated mission information and stress that they will fail to infer the data and make the best decisions. We anticipate that mission prediction information can significantly improve the performance for a large fraction of this population; while there will still be a small fraction of people who will be overloaded, emotionally and cognitively, and hence unable to make good decisions even with the additional help. Based on this assumption, we propose Hypothesis 1 with regards to our human study, and measure statistical significance from our data to be collected.

Hypothesis 1 ($\mathcal{H}1$): Of the population that are *unable* to make *any* good decision without alert messages, more than half of the participants will start making *all* correct decisions when provided assistance in the form of alerts and forward simulation-based mission predictions.

B. Preliminaries

We constructed four mission scenarios (an example in Figure 3), where each scenario starts at a particular time instance of a unique mission comprised of a team of eight to ten simulated robots. Each mission has progressed considerably, i.e., several hours have passed since the mission started with some nominal task plan being executed. One or more robots have returned with new mission information and are available for new task assignment. At this point, the participant is asked to assume the role of the commander. The participant can now access certain information using our interface, and decide on issuing commands to the available robots to facilitate mission progress. The information available to the human commander includes the instructions previously given to the currently out-of-communications-range robots in the field, the latest robot state updates, and other mission event updates with their corresponding timestamps. The usefulness of the updates varies because it may have been reported for an event that took place a few minutes ago to more than an hour in the past. Each of the scenarios are constructed such that it has one or two unwanted situations that may occur in the immediate future. These are not reported events because they have not been observed by the robots; rather, they are based on predictions from forward simulations. If a commander prioritizes certain tasks and assigns new instructions to the available robots, these undesirable situations can be prevented or alleviated. We refer to this new instruction set as the *ideal tasking* component, and this intelligent process as a *good decision*, for each scenario. This ideal tasking is ultimately compared with the tasking done by the participants in our human study.

We used two configurations of our interface in this study: (A) baseline in Round 1 and (B) with smart features in Round 2. Both versions provided the same reported information from past to present. However, version B offered one additional tab in the interface that showed alert messages generated by forward simulations based on a set of initial triggering conditions. With this version, the participants are also provided with all of the robots' probabilistically-computed locations and navigation information through out the forward simulation time period during an expected run where the alert triggering situation occurs.

C. Study Description

We conducted our study with engineering graduate students who are over 18 years old, and do not have color vision deficiencies. As each participant came in, and was given introduction to the mission and the system, we randomly assigned the person two scenarios with version A of the interface for Round 1. If the person makes any *good decision* in this round, they were considered reasonably capable and excused from the study. This was done in accordance with our hypothesis. Alternatively, the participants making *zero* good decisions moved to Round 2, where they used version B of the interface for two of the remaining scenarios. In this second round, we identified the participants who could make *all* good decisions, measured the performance improvement using forward simulation-based information, and verified our hypothesis with statistical significance.

Each experiment for a single participant in the study took up to 120 minutes. During the first $20 - 30$ minutes, the participants were given a presentation about the class of missions in this study, the mission-specific goals and assumptions, along with an introduction to the interface. They were also given generic guidelines on specific mission aspects to assess, and ways to handle relevant situations. The remaining $60 - 90$ minutes of the study consisted of the two rounds that controlled for the amount of available information by using a specific configuration of our custom interface shown on large computer monitors. At the beginning of each mission, each participant was briefed on the initial strategy and task plans that were initially executed, and then given ten minutes time to process the information using the interface, and decide on a strategy to task the available robots. After ten minutes, the participants explained their desired task allocation verbally, we recorded their responses and compared their instructions to the ideal tasking for that scenario. If their plan included the key components from the ideal tasking, it is labeled as *good decision*, otherwise it is labeled a failure to make the high quality decision.

D. Data Collection

In each round, there were a total of two or three good decisions to make, depending upon the two randomly-assigned scenarios that the participant was evaluated in. Only three out of the 13 participants made at least one good decision in the first round and were removed from the data set to be used in hypothesis testing. The re 10 participants, failed to make a single good decision in the first round with only the reported information and no alerts, so they continued on to the second round. We observed that 9 people, i.e. all except one, made all correct decisions in the second round. We terminated conducting our user study at this point since we had obtained enough data for claiming our hypothesis with significant statistical confidence.

Table I presents the performance of the 10 participants during Round 2, where they had access to forward simulation based alerts and information from predictions. For each user in this round, there were 2 or 3 decisions to be made, and we note the number of decisions the person made correctly. Thus, we calculate the percentage of good decisions each participant made which is also given in the table.

TABLE I: Performance of participants in Round 2, who were provided with forward simulation-based alerts and mission predictions.

User	# of Decisions to be made	# of Good Decisions	% Correct Decisions
1	3	3	100%
2	2	2	100%
3	2	2	100%
4	2	2	100%
5	2	1	50%
6	3	3	100%
7	3	3	100%
8	2	2	100%
9	3	3	100%
10	2	2	100%

E. Findings

We use one-sample hypothesis testing to test our Hypothesis 1 given in Section VI-A. If a participant can make *all* of the good decisions in the second round, it is considered a *success*, otherwise it is labeled a *failure*. We model the outcome for each participant in the second round as a Bernoulli random variable, X, that can take two values, 1 if there is success, and 0 otherwise, with probabilities P and $1-P$ respectively. We construct hypothesis tests for the Bernoulli parameter P where the null hypothesis H_0 is that P has some value P_0. The alternate hypothesis H_a is that the true value of P less or greater than P_0. We have a sample size of $N = 10$ corresponding to the number of participants in Round 2, who could not make a single good decision in Round 1. According to our hypothesis 1, we perform one-sided hypothesis testing for $P > P_0$, where $P_0 = 0.5$.

We compute the number of successes, $Y = \sum_{i=1}^{N} X_i$, where X_i denotes *success* or *failure* of the ith participant. By definition Y has a Binomial distribution with parameters N and P, defined as $P_Y(K) = \binom{N}{K} P^K (1-P)^{N-K}$. For significance level $\alpha \in (0,1)$, let $b_{n,p}(\alpha)$ denote the quantile of order α for Binomial distribution with parameters n, p. Since the Binomial distribution is discrete, only certain (exact) quantiles are possible. Our hypothesis testing is then, reject $H_0 : P \leq P_0$ versus $H_a : P > P_0$ iff $Y \geq b_{N,P_0}(1-\alpha)$.

We consider the significance level $\alpha = 0.05$ corresponding to 95% confidence. We calculate $b_{10,0.5}(0.95) = 9$. Table I shows that the total number of successes in our ten trials is 9. Therefore, we can reject the null hypothesis and claim our Hypothesis 1 to be true. In fact, we can say with 95% confidence that the percent of people making all good decisions with alerts, who could do zero without, is actually greater than 60%.

F. Discussion

The three participants who were dismissed from Round 1 could make exactly one correct decision, and thus scored between between $33.33 - 50.00\%$ correct decisions. The remaining 10 participants, comprising of 76.92% of the total population, failed to make even a single good decision without predictive assistance features. This poor performance in the first round indicates the inherent difficulty for humans in assessing a mission without any aid from forward simulation based predictions. We aggregated participants' answers in the questionnaires which asked about the challenges that they faced in each scenario. According to the participants, the main challenge was that they felt overloaded with information. Therefore, it was not possible for them to deduce how the mission will progress by assessing so many different variables. It is to be noted that the scenarios were designed in such a way that a reasonable human would be inclined to do some different tasking if they can not foresee the mission progression. This was done in an effort to minimize the possibility of participants arriving at the ideal tasking randomly.

In Round 1, after the participants made their decisions, we verbally provided the alert message and the robot navigation information, and asked for their revised answer. We found that 88.97% of the revised decisions given by the participants during this conversation included the ideal tasking. This high percentage can verify further that our chosen *good decisions* are indeed the superior choices, according to most people, when provided enough information.

In the second round, nine out of ten participants could make every good decision as they were given alert messages and predicted location data of the robots for the corresponding situation in version B of the interface. Collectively, the participants successfully made 95.83% of the good decisions, which is a substantial increase compared to Round 1 without alerts. This performance is also better than the revised decisions from the after-study conversation in Round 1. This indicates the usefulness of presenting forward simulation information up front in a structured way. Lastly, the participants were asked to provide free responses which revealed that the participants found the forward simulation-based prediction information very useful.

VII. Conclusions

We have shown that alerts can be extracted from forward simulations of the mission. We have demonstrated speeding up these simulations by more than three orders of magnitude, as compared to using physics-based simulations. We have verified that forward simulations based prediction information helps humans to make better decisions. In the future, we would like to explore smarter features for further speed up in forward simulations. We also want to assess the usefulness of different alert conditions from mission performance and how they are perceived by humans.

Acknowledgment

This work was supported by the CCDC Army Research Laboratory. Any opinions or conclusions expressed in this paper are those of the authors and do not necessarily reflect the views of the sponsor.

References

[1] B. Schlotfeldt, V. Tzoumas, D. Thakur, and G. J. Pappas, "Resilient active information gathering with mobile robots," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2018, pp. 4309–4316.

[2] A. H. DeCostanza, A. R. Marathe, A. Bohannon, A. W. Evans, E. T. Palazzolo, J. S. Metcalfe, and K. McDowell, "Enhancing human-agent teaming with individualized, adaptive technologies: A discussion of critical scientific questions," US Army Research Laboratory Aberdeen Proving Ground United States, Tech. Rep., 2018.

[3] H. Kim, J. L. Gabbard, A. M. Anon, and T. Misu, "Driver behavior and performance with augmented reality pedestrian collision warning: An outdoor user study," *IEEE transactions on visualization and computer graphics*, vol. 24, no. 4, pp. 1515–1524, 2018.

[4] W. He, Z. Li, and C. P. Chen, "A survey of human-centered intelligent robots: issues and challenges," *IEEE/CAA Journal of Automatica Sinica*, vol. 4, no. 4, pp. 602–609, 2017.

[5] F. Niroui, Y. Liu, R. Bichay, E. Barker, C. Elchami, O. Gillett, M. Ficocelli, and G. Nejat, "A graphical user interface for multi-robot control in urban search and rescue applications," in *2016 IEEE International Symposium on Robotics and Intelligent Sensors (IRIS)*. IEEE, 2016, pp. 217–222.

[6] C. Y. Wong and G. Seet, "Workload, awareness and automation in multiple-robot supervision," *International Journal of Advanced Robotic Systems*, vol. 14, no. 3, p. 1729881417710463, 2017.

[7] S.-Y. Chien, Y.-L. Lin, P.-J. Lee, S. Han, M. Lewis, and K. Sycara, "Attention allocation for human multi-robot control: Cognitive analysis based on behavior data and hidden states," *International Journal of Human-Computer Studies*, vol. 117, pp. 30–44, 2018.

[8] G.-J. M. Kruijff, M. Janíček, S. Keshavdas, B. Larochelle, H. Zender, N. J. Smets, T. Mioch, M. A. Neerincx, J. Diggelen, *et al.*, "Experience in system design for human-robot teaming in urban search and rescue," in *Field and Service Robotics*. Springer, 2014, pp. 111–125.

[9] S. Makris, P. Karagiannis, S. Koukas, and A.-S. Matthaiakis, "Augmented reality system for operator support in human–robot collaborative assembly," *CIRP Annals*, vol. 65, no. 1, pp. 61–64, 2016.

[10] K. L. Mosier, J. A. Kochan, B. K. Burian, and U. Fischer, "Autonomous, context-sensitive, task management systems and decision support tools i: Human-autonomy teaming fundamentals and state of the art [stub]," 2017.

[11] M. J. Barnes, J. Y. Chen, F. Jentsch, T. Oron-Gilad, E. Redden, L. Elliott, and A. W. Evans III, "Designing for humans in autonomous systems: Military applications," ARMY RESEARCH LAB ABERDEEN, Tech. Rep., 2014.

[12] F. Jentsch, *Human-robot interactions in future military operations*. CRC Press, 2016.

[13] J. J. Roldán, E. Peña-Tapia, A. Martín-Barrio, M. A. Olivares-Méndez, J. Del Cerro, and A. Barrientos, "Multi-robot interfaces and operator situational awareness: Study of the impact of immersion and prediction," *Sensors*, vol. 17, no. 8, p. 1720, 2017.

[14] M. Lauri and R. Ritala, "Planning for robotic exploration based on forward simulation," *Robotics and Autonomous Systems*, vol. 83, pp. 15–31, 2016.

[15] C. Goodin, J. T. Carrillo, D. P. McInnis, C. L. Cummins, P. J. Durst, B. Q. Gates, and B. S. Newell, "Unmanned ground vehicle simulation with the virtual autonomous navigation environment," in *2017 International Conference on Military Technologies (ICMT)*. IEEE, 2017, pp. 160–165.

[16] M. Cole, C. Lucas, K. B. Kulkarni, D. Carruth, C. Hudson, and P. Jayakumar, "Are m&s tools ready for assessing off-road mobility of autonomous vehicles?" in *Ground Vehicle Systems Engineering and Technology Symposium*, pp. 13–15.

[17] S. Al-Hussaini, J. M. Gregory, S. Shriyam, and S. K. Gupta, "An alert-generation framework for improving resiliency in human-supervised, multi-agent teams," in *Artificial Intelligence for Human-Robot Interaction Symposium, AAAI Fall Symposium Series*, November 2019.

[18] S. Al-Hussaini, J. M. Gregory, and S. K. Gupta, "Generation of context-dependent policies for robot rescue decision-making in multi-robot teams," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Madrid, Spain, October 2018.

[19] S. Konur, "A survey on temporal logics for specifying and verifying real-time systems," *Frontiers of Computer Science*, vol. 7, no. 3, pp. 370–403, 2013.

[20] J. M. Gregory, S. Al-Hussaini, and S. K. Gupta, "Heuristics-based multi-agent task allocation for resilient operations," in *2019 IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR)*. IEEE, 2019, pp. 1–8.