Communication Styles in Human-AI Teams Tasked with Urban Search and Rescue Missions

Ashish Amresh, Nancy Cooke
Arizona State University
Tempe, AZ
amresh@asu.edu, nancy.cooke@asu.edu

Adam Fouse
Aptima Inc.
Arlington, VA
afouse@aptima.com

ABSTRACT

Urban search and rescue (US&R) refers to operations conducted in collapsed man-made structures. It has been recognized as a useful domain for studying human-AI interaction, due in part to the complications that arise out of introducing AI into an unpredictable environment such as a collapsed building. In this study, we investigate different AI communication styles in a team-based experimental search and rescue scenario. By inviting human participants via a simulated Minecraft-based reconstruction of urban search and rescue mission maps, we collect data gathered with both a "Wizard of Oz" design, with the researchers playing the role of an AI advisor and a rule-based AI advisor. In both cases, the advisor gave guidance to the team of participants during the experiment. The focus areas for the study are the adherence to guidance under different communication styles, usage of the styles, and participants' responses to these styles. Although the objective of the Minecraft-based experiment for the participants was to save as many victims trapped in collapsed buildings via fifteen-minute missions, this study evaluates team performance based on how teams adhere to the advisor's guidance, disregard the guidance, and ask for additional information. We present the results of our experiments via two specific guidance conditions EG- (Explicit Guidance or ISG- (Information Shaping Guidance) and modes of communication (push vs. pull). Our results indicate that there was a greater adherence to information-shaping guidance when compared with explicit guidance and participants seem to respond more to push over pull. The results and discussion presented in this study would help drive the design of human-AI teaming systems, especially when AI functions as an advisor to provide guidance to human teams.

ABOUT THE AUTHORS

Ashish Amresh is a Research Scientist at Arizona State University and an Associate Professor at Northern Arizona University primarily looking at how games can be used to evaluate human and AI teaming experiments.

Nancy Cooke is a Professor in Human Systems Engineering at Arizona State University and the Director of ASU's Center for Human, AI, and Robot Teaming under the Global Security Initiative and is an expert in team cognition.

Adam Fouse is a Senior Director in Aptima's Performance Augmentation Systems Division.

Communication Styles in Human-AI Teams Tasked with Urban Search and Rescue Missions

Ashish Amresh, Nancy Cooke
Arizona State University
Tempe, AZ
amresh@asu.edu, nancy.cooke@asu.edu

Adam Fouse
Aptima Inc.
Arlington, VA
afouse@aptima.com

INTRODUCTION

The fast-paced, information-rich environment of modern warfare, both in physical and cyber spaces, requires rapid strategic and tactical decision-making. As adversaries become more adept and capable, the future success of military teams operating in fast-changing battlespaces will depend on accelerating the observation-to-action loop. Artificial intelligence (AI) will be critical in assisting commanders with tactical and strategic decision-making. Urban search and rescue (US&R) refers to operations conducted in collapsed man-made structures. It has been recognized as a useful domain for studying human-AI interaction (Burke et al., 2004). Human-AI teaming in the domain of US&R is a widely researched area (Bartlett & Cooke, 2015), due in part to the complications that arise out of introducing AI into an unpredictable environment such as a collapsed building. A simulated task environment (STE) is one in which real-world tasks and cognitive decision-making can be performed and evaluated with a certain degree of precision and the results gathered have a strong correlation with operations in the real world (Cooke & Lawless, 2021). STEs are ideally suited for US&R tasks due to the difficulty and high cost of developing hands-on training exercises. There is potential to further improve the performance of the teams within the STEs by including AI agents that can assist the human performers within the task. In this study, we present a human-AI team design with an AI advisor that assists three-person human teams and helps improve their task performance during US&R missions. Specifically, we are investigating the use of different communication styles for the AI agent and its effect on human teams. The project was divided into two phases: a Wizard of Oz (WoZ) design phase followed by the incorporation of a rule-based AI agent. In the WoZ phase, researchers played the role of an AI advisor and gave guidance to a team of participants during the experiment. Data collected from this phase was used to develop the rule-based AI agent and further data was collected on how varying communication styles performed with human participants under two distinct conditions: 1) push (in which the AI agent provided distinct guidance to each team member) and 2) pull (in which the AI agent provided guidance when requested by the team member). We discuss the adherence to guidance under different communication styles, how those communication styles were decided upon and used, as well as the way participants responded to the WoZ and rule-based AIs over the course of the study. Figure 1 shows the major components of the STE, to the right we have the Minecraft game environment, to the top-left we have the top-down view of the map of the collapsed building, and to the bottom-left we have the communication interface for the AI. Although the objective of the experiment for the participants was to save as many trapped victims as possible in the 15-minute missions, this end goal, while important is not the focus of this study. Instead, we will evaluate teams by how often they adhered to the advisor's guidance, disregarded the guidance, and asked for further information from the advisor. In a recent study (Orth et al., 2021), US&R missions were shown to be a useful environment for looking at exploration (the guidance cannot be explicitly stated) vs. exploitation (where the guidance is explicit) behaviors in teams because though both are required to achieve success as a team, the correct balance can be difficult to achieve and is dependent on dynamic situational factors.

Figure 1. Major components of a human AI interaction system built in Minecraft.

Background

Research on human-AI collaboration in defense contexts has grown dramatically in recent years. Although technological advances have shown improvement in AI systems, researchers have noted the importance of understanding the psychology of humans teaming with those AI systems (Seeber et al., 2020). Even with the best AI systems available, any human team member introduces dynamics that must be understood to ensure successful collaboration. In operational contexts in which AI systems are meant to augment Warfighter awareness and decision-making, both overcompliance and undercompliance could prove deadly. While we work on creating AI that produces helpful results, we also must understand how we can best integrate an AI agent into a pre-existing or newly formed human team.

Minecraft offers a natural STE to develop and conduct such human AI teaming experiments. Minecraft allows players to build and explore virtual worlds. Recently, it has been used as a research software testbed for studying human-AI teaming, which is the collaboration between humans and artificial intelligence (AI) agents (Hofmann, 2019). In Minecraft, researchers can create environments that simulate real-world scenarios, such as disaster response or exploration missions, and study how human players work with AI agents to achieve goals in these scenarios. This approach allows researchers to study human-AI teaming in a controlled, yet realistic, environment. Minecraft also offers several advantages over other research testbeds. First, it has a large and diverse player base, which allows researchers to recruit participants easily. Second, it is highly customizable, allowing researchers to create specific scenarios and tasks that fit their research questions. Finally, it is a low-risk environment, as failure or mistakes do not have real-world consequences, allowing researchers to study teaming in a safe and controlled environment.

Overall, Minecraft has emerged as a promising research testbed for studying human-AI teaming and has the potential to provide valuable insights into how humans and AI can work together effectively in various domains. Teamwork processes are developed over time (Salas et al., 2008), and the time it takes a team to become fully effective depends on the task and the team. A laboratory-based simulation of a remotely piloted aerial vehicle ground control station required four 40-minute missions for teams to reach asymptotic levels of development (Cooke et al., 2007). Other real-world tasks such as space exploration or ones involving medical teams could require even more development time with a team. As a result of team progress, measures of team state—such as workload, trust, and situation awareness—can also be expected to change over time (Huang et al., 2021). Unfortunately, the team literature is based on laboratory studies in which teams are together for hours, not days or months. It is of theoretical and applied interest to understand how teams evolve over time in terms of team processes and states, as well as how the effectiveness of interventions to improve team performance fares over time. Longitudinal studies as teams evolve can also open the door for interventions that adapt to team development. Minecraft-based testbeds are uniquely suited to address these factors due to their ubiquitous availability, low resource requirements, and readily available participant base.

STUDY DESIGN

Figure 2. The three roles (H – Engineer; M -Medic; S-Search).

The study required three-person teams to enter a Minecraft simulation of a US&R scenario with the goal of saving as many victims as possible in 15 minutes. Teams played two missions on the same map, designed to simulate a collapsed office building. Experienced video game players were chosen as participants to ensure they would be able to understand and perform the task. Participants were able to select between three roles, each having one unique tool. The three roles were 1.) medical specialist with the ability to save victims, 2.) hammer specialist, a slow player with the ability to break through rubble-blocking areas of the map, and 3.) search specialist, a speedy player with the ability to pick up and place victims as shown in Figure 2. Participants could select any combination of these roles, and during the mission, could also switch roles by returning to a starting area on the map. The unique abilities possessed by the roles allowed for teams to respond to the changing environment of the map by changing the composition of roles on their team. Prior to each mission, participants engaged in a 3-minute preplanning session in which they would be shown a top-down view of the map in Zoom and would be given the opportunity to discuss their strategy for the upcoming mission (Figure 3). Participants were prompted to discuss what roles they would begin with, where they would go on the map, and how they would work together as a team. In each mission, one perturbation, a sudden change intended to alter a team's behavior, would take place. In the first mission, a rubble collapse occurs seven minutes after the mission starts, players would be notified that the roof had collapsed, and extra rubble has been added to the map. In the second mission, a gas leak occurs five minutes after the start. Players would be notified that three gas leaks had appeared on the map, but they were given six possible locations for the gas leaks with three minutes to "fix" them to receive extra points. The project included two phases: I) A WoZ guidance using a human advisor that looked at the effect of EG and ISG conditions and II) An AI-based guidance that incorporated the findings of Phase I and looked at the effect of push and pull conditions.

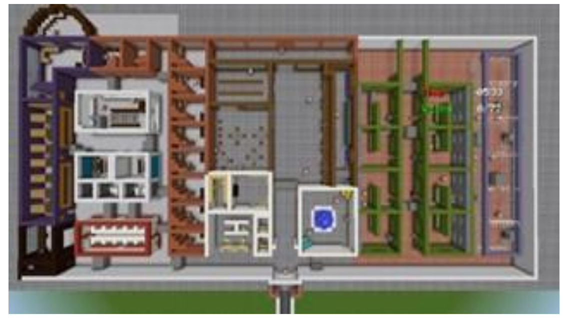

Figure 3. Top-down view of the collapsed building.

Phase I Wizard of OZ Guidance

The first phase used a human advisor pretending to be AI and providing guidance. The use of WoZ allowed us to focus on the effect of different types of guidance on the team rather than how that guidance was generated. Players were provided with a map they could access through their internet browser that would show their individual locations on the map. This map was also used to give the text advice under the explicit (EG) and information shaping (ISG) guidance conditions. Participants were able to communicate with each other via Zoom and were assigned callsigns to protect their identities. The advisor did not communicate by voice on Zoom but rather created guidance that was shown on the screen along with the map to simulate how guidance would appear from an AI agent. The advisor, a confederate, gave advice to the participants regarding the composition of roles on their team, areas they should move to on the map, and goals they should be attending to. The two major styles of communication used over the course of the study depended on the type of information given, EG vs. ISG. EG involves the use of direct commands (e.g., switch to the medical specialist; go to the library) while ISG involves providing relevant information (e.g., there is rubble on the right side that hasn't been cleared; more of the map needs to be explored) about the state of the map or mission without explicit instruction.

To come up with scripted advice for the two conditions, we developed optimal and sub-optimal strategies for completing the mission and decided to use advice that would move teams towards optimal strategies and away from suboptimal strategies. Some example optimal strategies included: having a distribution of roles on the team, using the correct role for the correct task, emphasizing exploration in the early stages of the mission, staying in close enough proximity to help teammates, using a search specialist for clearing rooms/exploring, and using the search specialist in the early stages of the missions.

Data was gathered from 54 participants (48 Men, 8 Women) separated into 18 teams. 85% of the participants were regular video game players, and 71% of participants were Minecraft players specifically. Prior to participating in the experiment, all participants were briefed on the objectives and rules of the experiment with a video presentation. Following this, all participants engaged in an in-game tutorial allowing them to become familiar with the tools used in the experiment. Zoom was used to conduct the experiment remotely and to allow the participants to interact with each other. Later, the Zoom recording of the Minecraft environment and audio transcripts were used to determine adherence to the advisor's guidance.

Data on participant adherence to advice was gathered through Zoom transcripts and video recordings of the Minecraft experiment. Two human coders independently verified the data and any time a participant tried to follow the advisor's guidance; it was counted as adherence. Due to the changing requirements of this mission, there were several instances in which a participant intended to follow the advisor's guidance initially but had to change their plans en route. Instances in which participants disregarded or failed to adhere to the guidance were noted as "failed to adhere", and instances in which the guidance led to a discussion amongst two or more teammates were noted as "prompted discussion." The intention of the Phase I study was to prepare for the implementation of the rule-based AI during Phase II.

Phase II Rule-based AI Guidance

The first phase used a human advisor pretending to be AI and providing guidance. In the second phase, we designed an experiment to identify the effects of how guidance from an AI agent is communicated; and impacts compliance and team performance with within-subjects (repeated-measures) design; with communication style (push or pull) as the independent variable and compliance; performance (number of victims saved and time); map exploration as a dependent variable. We hypothesized that there would be a difference in the compliance of the teams between AI communication methods. The communication methods were defined as "push" or "pull". The push condition had the agent send out guidance to a team member when that team member and the environment satisfied certain conditions. The pull condition had the agent only send guidance to a team member when guidance was asked for. There were no modifiers for asked-for guidance in that the team member would simply indicate that they would like any guidance available from the agent. Figure 4 illustrates the web-based interface that displays visual markers placed by teammates on the map as well as guidance from the agent.

Figure 4. Markers placed on the map and agent guidance interface.

Four teams of three players participated in two sessions per week of three games over 4 weeks for 24 game plays. The order of manipulation of the role of the AI agent was counterbalanced within and between groups. The manipulation mode switched halfway through the experiment at the end of week 2. Initially, participants completed a survey for demographic information, plus additional questions about how often and what kinds of video games they played as well as their general feeling towards AI agents. After each session, participants completed an additional survey to gauge their feelings about their team's performance, their individual performance, and how the AI agent impacted their team.

Based on what we observed in Phase I, we identified the optimal strategy for playing the Minecraft scenario and created guidance accordingly. Table 1 captures a subset of the conditions that we used to develop the rule-based AI. For the scenario, it is best to have two engineers and a search specialist, in the beginning, to prioritize exploring new areas to discover where victims are located. In addition, because perturbations will drop rubble onto the map at various intervals, there should always be at least one engineer on the team so that players do not get stuck in the building. The search specialist can move victims quickly to a central location, while engineers clear pathways. Markers should be placed as new rooms are explored. By mid-game, the teams should have switched one of their engineers to a medic and started saving victims, clearing out markers once victims are safe (Figure 5). By the end of the game, a good team composition is either an engineer, medic, and search specialist or two medics with one engineer. By the last few minutes of the game, medics should be focused on saving victims before time runs out (Figure 6).

Figure 6. The medical specialist saving a victim.

Table 1. Example Conditions That Trigger Agent Guidance

Guidance	Rule	Triggered by
Your team currently does not have an Engineering Specialist. It is usually a good idea to have at least one Engineering Specialist on your team.	Have at least 1 engineer on the mission start	No engineers at the start of the mission
The building is unstable, it is recommended to always have an engineer in case someone gets trapped.	The number of engineers should always be > 1	Selecting a role and # engineers <1
You have not been placing enough markers. Please place a marker each time you scout a room to let your teammates know what you found.	Remember to place markers to communicate with your team.	Entering / Exiting a room and the number of markers placed when those events occur.
You broke through some obstructive rubble. Good job getting back on track!	Reinforcement for behaving well.	If an engineer has not broken rubble in 30 seconds – has received guidance about that, and then breaks rubble.
You have not saved a victim in over 30 seconds. Please save the victim in {ROOMNAME}.	Medic needs to be efficient in their role.	Medic has not saved a victim in over 30 seconds AND we have passed the first minute of the mission.
There has been a gas leak. Your team should be more spread out over the map to quickly address it. OR There has been a gas leak. Your team should be more spread out over the map to quickly address it.	Coordinate on Gas Leak perturbations	If # explored regions < 2
It's always good to have an engineer or two handy in these situations. Coordinate with your team to spread out and address it.	Coordinate on rubble perturbations	If # explored regions < 2
The mission is in its last stage. Focus on finding and saving victims as quickly as possible.	Focus on treating as many victims as possible.	@ 10 mins in: If greater than 30% of total victims remain.
Your team is halfway through the mission. Make sure you have at least one Medic going forward. Time is your enemy. OR We are now halfway through the mission. It looks like you have at least one Medic - good job! Focus on those victims!	Prioritize victim saving towards the latter half of the mission.	@halfway mark #medics < 1

DATA ENGINEERING

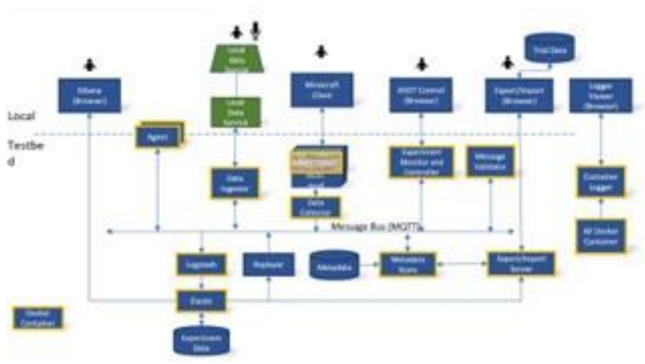

Figure 7. The data engineering architecture used by the testbed.

The Minecraft testbed was engineered as a modular set of discrete containers utilizing Docker containerization technology. Each container communicates within the system over a common message bus built on the MQTT messaging protocol. To encourage readable, performant messages, the JSON message format was selected. This message format is ideal for data analysis and storage via Elasticsearch—a popular, JSON-based database.

When running the experiment, a variety of messages from different containers must be collected and stored to accurately log all the events in a session. This includes control messages which activate and deactivate individual trials; scenario messages which communicate the actual state of the session and participants during the experiment; and agent messages which are the guidance messages sent to participants by the agent container.

The Minecraft “modding” environment offers several “hooks” (software events) that can be monitored, logged, and converted to JSON messages, then subsequently published on the message bus. In the testbed, the Minecraft mod captures all events dealing with player movement (walking, swinging a tool, pitch, yaw, acceleration), custom scenario control (mission control, perturbation control), and custom object interaction (use of custom tools, custom behaviors of objects, point scoring).

All events for a particular team are associated with an ‘Experiment ID’ and a ‘Trial ID’, along with other relevant metadata capturing the session conditions. When a session is complete, a purpose-built graphical user interface (GUI) is accessed via the browser and a complete set of messages associated with a trial are exported. These messages are ordered according to the time they arrived at the database but have multiple timestamps that allow different ordering when necessary. A standard session numbers in the tens of thousands of messages depending on the duration of the trial. The complete data engineering architecture is shown in Figure 7.

RESULTS

For Phase I, a chi-square test of independence was performed to examine whether there was any difference in adherence to the advisor's guidance depending on the style of communication used. The chi-square test showed that communication style had a significant effect on adherence to guidance $X2$ (4, $N = 353$) = 92.3, $p < .001$. Further chi-square tests were performed to understand which communication styles prompted the greatest adherence to the guidance given. In terms of guidance, ISG prompted greater adherence than EG $X2$ (2, $N = 138$) = 17.2, $p < .01$.
Information-shaping guidance appears to be associated with the highest rates of compliance in participants. This relationship calls for further research, considering there are a few reasons that participants could have preferred ISG over EG. It is possible that participants didn’t have enough trust in the advisor to follow commands from them, but they may have been more likely to accept that the advisor had information about the state of the map. There is also a difference between being told what to do and being told why you might want to do something. It is worth mentioning that many of the participants in this usability study were experienced Minecraft players. While they weren’t familiar with the specific experiment they were participating in, it is possible that they preferred to rely on their own expertise as opposed to following explicit instructions.

For Phase II, we examined the differences between the push and pull conditions. Outcomes examined across these differences included the number of victims saved in each trial, the number of guidance messages given by the agent, and the proportion of guidance followed. Table 2 displays the results of three t-tests, along with their effect sizes. Few differences were found between the push and pull conditions across the entire study. As expected, we found significantly more guidance messages given in the push condition per trial (M = 26.43, SD = 5.00) than in the pull condition (M = 6.84, SD = 4.03). However, there was no difference in the proportion of guidance followed, even though as expected more guidance was followed when more guidance was given.

Table 2. Mean Differences Across Conditions

	t	*df*	*p*	*d*
Number of victims saved	0.534	147	0.594	0.088
Guidance count	-25.996	147	< .001	-4.276
Guidance Proportion followed	-1.227	147	0.222	-0.202

DISCUSSION

By looking at the two phases, the quantitative findings from this study seemed to indicate that the experimental manipulation did not impact performance in terms of victims saved in the virtual search and rescue scenario. Although some differences in outcomes were observed between conditions, more research is needed to fully understand how communication style from an autonomous agent impacts performance and individual perceptions. Additionally, though we observed a pattern in terms of changes over the eight sessions of this experiment, more research is needed to determine the causes that led to these changes. For example, we could not determine empirically from our study whether boredom or the difficulty of the scenario played a role in how teams behaved and performed during the trials. A qualitative review of the video recordings and the open-ended survey item suggested that indeed, boredom and frustration were factors in participant and team behavior. For example, while teams changed their strategy early in the experiment, about halfway through the experiment, several teams refused to change their strategy if it was “good enough” and they felt that an imperfect score was due to their ability to execute the strategy, rather than the strategy itself. In addition, we noted teams seemed to coalesce and create a sense of comradery over the course of the experiment, despite any boredom or frustration. The study does provide insights into the use of STE’s like Minecraft for human-AI teaming experiments. The major takeaways from conducting the two phases are: 1) there is value in building unpredictable and uncertain environments to maximize the value of AI-based guidance in team-based tasks; 2) humans are more likely to adhere to information-shaping guidance by AI agents and 3) there is no significant difference in adherence to AI advice when AI provides guidance on its own compared to when it is asked to provide guidance.

ACKNOWLEDGEMENTS

This research was supported by the Defense Advanced Research Projects Agency (DARPA) through government contract W31P4Q-20-C-0031. The views, opinions, and/or findings expressed are those of the author(s) and should not be interpreted as representing the official views or policies of the Department of Defense or the U.S. Government.

REFERENCES

Bartlett, C. E., & Cooke, N. J. (2015, September). Human-robot teaming in urban search and rescue. *In Proceedings of the Human Factors and Ergonomics Society Annual Meeting* (Vol. 59, No. 1, pp. 250-254). Sage CA: Los Angeles, CA: SAGE Publications.

Burke, J. L., Murphy, R. R., Rogers, E., Lumelsky, V. J., & Scholtz, J. (2004). Final report for the DARPA/NSF interdisciplinary study on human-robot interaction. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 34(2), 103-112.

Cooke, N. J., Gorman, J. C., Duran, J. L., & Taylor, A. R. (2007). Team cognition in experienced command-and-control teams. *Journal of Experimental Psychology*: *Applied*, 13(3), 146.

Cooke, N. J., & Lawless, W. F. (2021). Effective human–artificial intelligence teaming. *Systems Engineering and Artificial Intelligence*, 61-75.

Hofmann, Katja. "Minecraft as ai playground and laboratory." *In Proceedings of the annual symposium on computer-human interaction in play*, pp. 1-1. 2019.

Huang, L., Cooke, N. J., Gutzwiller, R. S., Berman, S., Chiou, E. K., Demir, M., & Zhang, W. (2021). Distributed dynamic team trust in human, artificial intelligence, and robot teaming. *In Trust in human-robot interaction* (pp. 301-319). Academic Press.

Orth, D. A., Buchanan, M., Amresh, A., Smith, C., Lematta, G., Cooke, N., ... & Dubrow, S. (2021, September). Designing for exploration and exploitation in experimental search and rescue scenarios. *In Proceedings of the Human Factors and Ergonomics Society Annual Meeting* (Vol. 65, No. 1, pp. 720-725). Sage CA: Los Angeles, CA: SAGE Publications.

Salas, E., Cooke, N. J., & Rosen, M. A. (2008). On teams, teamwork, and team performance: Discoveries and developments. *Human factors*, 50(3), 540-547.

Seeber, I., Bittner, E., & Briggs, R. O. (2020). Triparna De Vreede, Gert-Jan De Vreede, Aaron Elkins, Ronald Maier, Alexander B Merz, Sarah Oeste-Reiß, Nils Randrup, et al. 2020. Machines as teammates: A research agenda on AI in team collaboration. *Information & management*, 57(2), 103174.