Human-Robot Teaming for a Cooperative Game in a Shared Partially Observable Space

Brendan Mooers[a], Audrey L. Aldridge[b], Andrew Buck[a], Cindy L. Bethel[b], and Derek T. Anderson[a]

[a]Department of Electrical Engineering and Computer Science, University of Missouri, Columbia MO, USA
[b]Department of Computer Science and Engineering, Mississippi State University, Mississippi State MS, USA

ABSTRACT

An open research question is how to best pair a human and *agent* (e.g., AI, autonomous) relative to a complex, multi-objective task in a dynamic and unknown partially observable environment. At the heart of this challenge resides even deeper questions like what AI is needed and how can bi-directional and multi-directional human-robot trust be established. In this paper, the theoretical framework for a simple 2D grid world-based cooperative search and rescue game is explored. The resultant prototype interface enables the study of human-robot interaction for human-robot teaming. First, the design and implementation of a prototype interface is discussed. A 2D grid-world was selected to simplify the investigation and eliminate confounding factors that arise in more complicated simulated 3D and real world experiments. Next, different types of autonomous agents are introduced, as they impact our studies and ultimately are an integral element of the underlying research question. This is followed by three levels of increasing complexity open-ended games, easy, medium, and hard. The current paper does not contain human experimentation results. That is the next step in this research. Instead, this article introduces, explains, and defends a set of design choices and working examples are provided to facilitate open discussion.

Keywords: human-robot teaming, AI, agent, information sharing, trust, human-robot interaction

1. INTRODUCTION

As AI and robotic agents are used to complete more complex tasks in support of human collaborators, there is a need for the development of an effective, collaborative human-robot teaming paradigm. In order to create successful collaborations and cooperation between humans and robots is it necessary to understand how information sharing needs to occur. Additionally, there needs to be a calibration of trust for successful human-robot teaming. Information sharing and trust calibration, when neglected or improperly used, can cause collaborators to suffer from miscommunication, withholding of information, untrusting behavior, and overwhelmed workloads. When human-robot teams experience these types of issues, safety and mission success are jeopardized.

Researchers have explored metrics for measuring the trust a human has for a robot.[1,2] As autonomous agents become more capable of complex decision-making, there is an need for autonomous agents to learn to share information, rely on, and collaborate with their human teammates. In many human-robot teams, the robot is considered more of a tool, than a team member working with the human.[3] Autonomous agents may have better knowledge of the environment and situation than their human team members; however, there is not sufficient trust in their capabilities to allow them to make informed decisions or be included in decision-making. For optimal human-robot team performance, it is necessary to have appropriate information sharing that maintains a balanced cognitive load and accurate situation awareness. This establishes transparency and explainability between agents in the human-robot team. Transparency is important in maintaining trust, as it can provide insight into teammate behavior and intent. Establishing transparency through information sharing

Send correspondence to
Derek Anderson: E-mail: andersondt@missouri.edu
Cindy Bethel: E-mail: cbethel@cse.msstate.edu

can promote informed interdependent decision-making, influence human reliance on robots, and support system independence.

To address these teaming concerns, we present a grid world search game designed to test components of a novel human-robot teaming model.[4] To begin, we plan to test two components of the proposed model, information sharing and trust in human-robot teams. Because this is the first study, we aim to gather baseline data on information sharing and to learn how trust is influenced by the following factors: task performance, information sharing, situational awareness, cognitive load, and personality. By understanding and then optimizing these factors, a trust-based construct can be created among human-robot team members. The results of this upcoming study form the foundation for the performance of future, iterative evaluations of more complex aspects of the model.

This paper presents the theoretical framework used for designing and implementing a 2D grid world search game. First, existing simulations and games that were designed for similar testing or tasks are explored. Reasons for choosing the grid-world approach for initial testing of the model are discussed (Section 2). Next, specifics of the game creation, including environmental layout, interface functionality, goal of the game, and agent interactions are explained (Section 3). This leads to a discussion about the types of agents used for this study including a description of their capabilities (Section 4). Environment characteristics for the three levels of difficulty within the 2D grid world are explained along with how discrepancies in initial information from the robot can affect task completion (Section 5). Storyboard-type examples portraying possible situations encountered in each difficulty level and an example of a full environment, similar to what is used in the game, are presented. A brief description detailing environment traversal as part of game example is provided in Section 5.4. The next section (Section 6) of the paper provides a brief overview of the proposed model and why it is needed for future human-robot teaming. Task measures, survey metrics, and a partial study design are introduced in Section 7. The paper ends with a discussion of our research process (Section 8) and plans for the future.

2. RELATED WORKS

To evaluate the proposed human-robot teaming model, our research team created a novel 2D game where multiple agents cooperate in a shared, partially observable environment. Although many environments already exist for multi-agent cooperation[5–9] in a partially observable space,[10–12] testing the model requires a simulated environment designed for search and rescue tasks, similar to those outlined in Refs. 13,14. While these simulated environments were designed for perfecting algorithms[13] and evaluating multi-robot control paradigms,[14] they could be helpful in future testing of our model, but we are not currently prepared to test these low-level model details. In this beginning stage of testing, we focus on investigating and finalizing the high-level details of information sharing and trust calibration for the proposed model.

For this evaluation we do not need a life-like, real-world environment but rather a 2D grid-world representation of the search aspects of a search and rescue task, similar to Refs. 15–17 but more advanced than Refs. 18–22. We desired a 2D game-like simulation that could be edited/updated to test one or many of our model's components. The game needed to be able to: (1) explore a multi-objective military-type search and rescue task in a partially observable space, (2) include potential adversaries and dynamic field environments, and (3) scale with the increasing complexity to evaluate larger portions of our implemented model. Consequently, due to the size and intricacy of our model, our research team requires the creative freedom to develop a simulated environment that can be adapted to help test our model's components in a strategic step-wise manner. Constraining this game to a grid world allows us to simplify the representation of environment attributes and limits practical issues that come with representing a physical environments.[23]

3. GAME WORLD AND INTERFACE

To study information sharing and trust, we do not need an overly complex task. Therefore, we reduced search and rescue to a simple search task where human-drone teams must locate and reach a target. The rescue part of the task will be added in later studies once all high-level details are finalized. The goal of this multi-objective search game is to locate and reach the target in as little time and with as much energy points left as possible. We expect that using more cooperation (information sharing and trust) with a drone teammate will yield better task results than when using less cooperation with a drone teammate across all environments. For this study, we plan to utilize two types of drone agents: an agent with basic sensing functionality using way-point navigation "Agent 1" and an agent with path planning capabilities in addition to basic sensing and way-point navigation "Agent 2" (See Section 4).

3.1 Game Design

The 2D visualization of the world that the user is presented is programmed in Python using the Pygame library. The grid-world environment where the search task will take place consists of mixed terrain in a simulated outdoor field environment. We are using a 50 x 50 grid (1 square = $10m^2$) for easy viewing of a human player's behavior and interaction with an assigned drone teammate. Figure 1 shows an example of the game interface and a partial environment from the player's perspective. The brightened squares represent what the human has seen and are accurate to the ground truth of the map. The dimmed squares are places in the map that the player has not yet "seen" and might not be fully accurate when compared to the ground truth. The grey "stone" squares are impassable obstacles, the green "grass" squares are open areas that have a low traversal cost, and the brown "muddy" squares are open areas that have a higher traversal cost. As the player attempts to reach the target, they uncover the ground truth of their immediate surroundings, which may reveal more optimal paths to the goal.

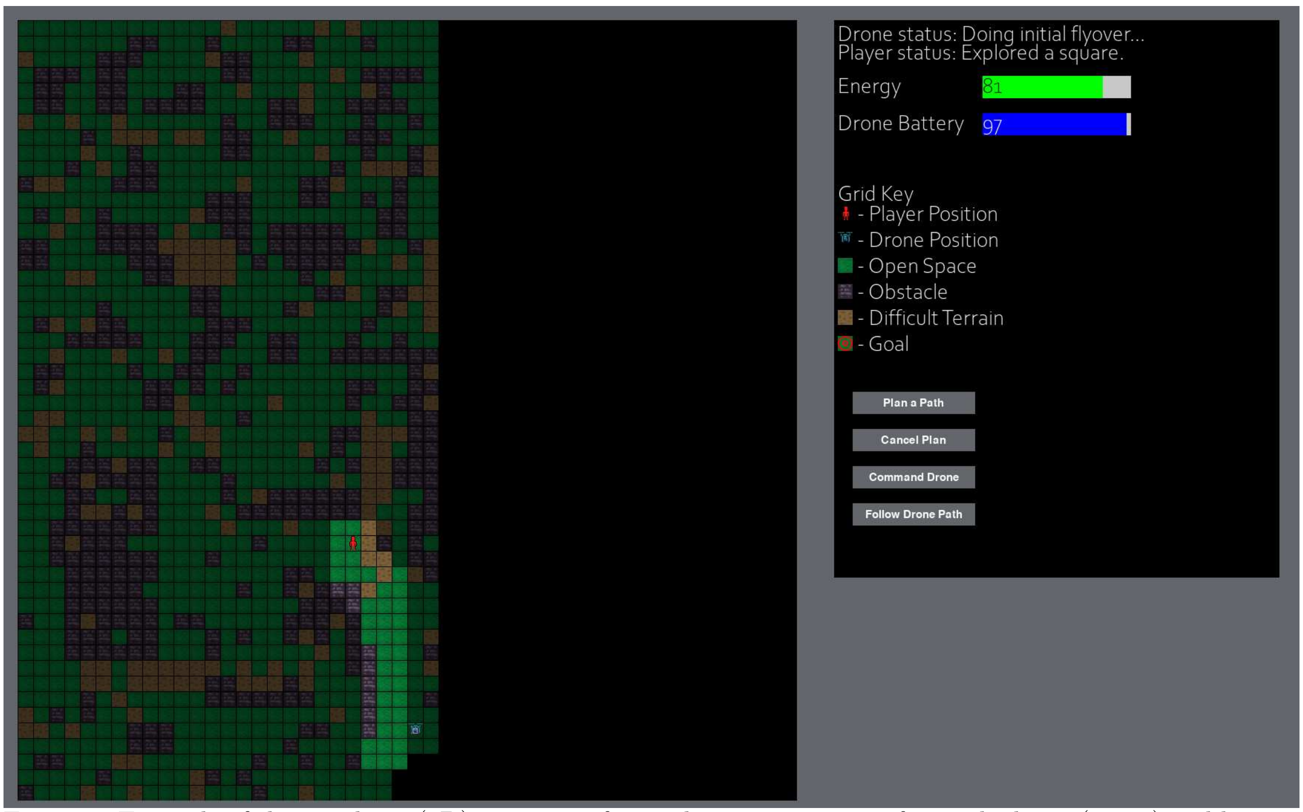

Figure 1: Example of the top down (2D) game interface. The game consists of a single drone (agent) and human (player) who can see their current location and where they have explored. The objective is for the user and agent to cooperate to solve a common task in a finite amount of time with limited resources.

The game world consists of three separate "views": the ground truth of the map, the human's view of the world, and the agent's view of the world. To simulate imperfect sensors and detection algorithms in the real world, the agents in this study do not provide a one-hundred percent accurate overview of the map. An agent provides a somewhat inaccurate but complete view of the world by performing an initial "raster scan" search pattern of the map. Because an agent can make mistakes, an agent's view may not match the ground truth exactly (Figure 2). These possible disagreements are what necessitates our three "views" of the world.

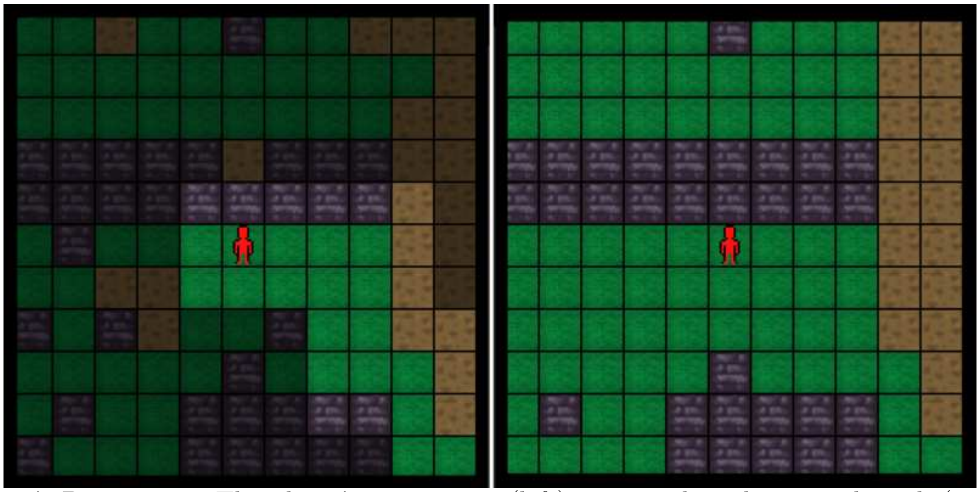

Figure 2: Player's Perspective: The player's perspective (left) compared to the ground truth (right). Dimmed squares are areas that the drone has mapped and the human hasn't visited yet and thus include some amount of error.

The human's view is initially unknown and limited to nearby and previously explored squares, but as the player explores the environment, the map becomes more accurate, matching the ground truth. The human player can only see information from the human's perspective and the agent's perspective. Squares incorrectly sensed and labeled by the agent during its initial flyover of the environment are outlined in magenta (Figure 3) and made visible to the player once the ground truth is revealed, either from traversing the environment or after task completion. By doing so, we give the player an idea of how accurate the drone has been throughout the mission, which allows the player to adjust their plan if necessary. Representing the world with these separate views allows us to simulate real-world situations where the agent's view of the world may not perfectly match the human's perspective or the ground truth.

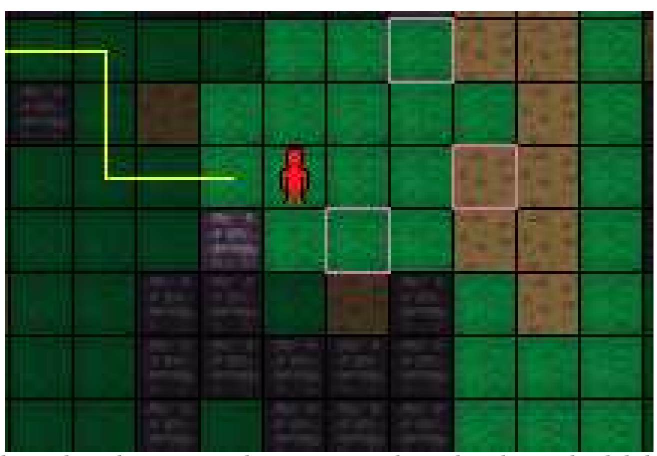

Figure 3: Agent Error: When the player reveals a square that the drone had labeled incorrectly, it is given a magenta border. This is to allow the player to make informed choices when deciding to follow the agent's suggested path or take an alternative route.

3.2 Task and Environment Traversal

In this study, a player assigned is tasked with reaching the goal via the most efficient path possible. There are multiple factors that a player must consider to complete the task. First, the human player has a limited amount of energy, and while traversing open squares costs 1 energy point, traversing difficult squares costs 4 energy points. If the human player completely runs out of energy, the game ends, and the mission is deemed a failure. Second, the drone agent is also limited in the amount of actions it can take, due to a constantly draining battery. After the drone's initial flyover of the map, a player can command it to re-investigate a 5 x 5 area in the grid-world. This costs a significant amount of the drone's battery level but rewards the player with the ground truth of a desired area. The player can make use of this feature to identify more efficient paths to the goal without expending too much of the human's energy investigating the area on foot. However, a player must be cognizant of the effect that overusing this feature will have on the lifespan of a drone's battery. If the drone's battery dies, the mission does not fail, but the player loses the ability to command the drone. Alternatively, neglecting to use this feature might cause a player to fail the mission, or it might be what stands between the player and a last ditch effort to find a path to the target.

At any time, a player has three actions they can take regardless of their assigned agent type: plan path, cancel plan, and command drone. A player's main mode of traversal is to plan a path consisting of one to five squares at a time and submitting it to their human representation in the game (Figure 4). The human representation in the game will move along the submitted path until it either reaches the end of the path, hits an obstacle, or the player inputs a "Cancel Plan" command. We chose this method of traversal to better capture the player's intent with their movements. As an alternative to plotting a path themselves, we've included a button that can be used with "Agent 2" that moves the human to the next square on the path suggested by that agent. Since the agent's view of the world is imperfect, simply clicking this button repeatedly won't provide the best route to the goal in most cases, but it allows the player to easily follow the agent's path when the player trusts that the agent is correct.

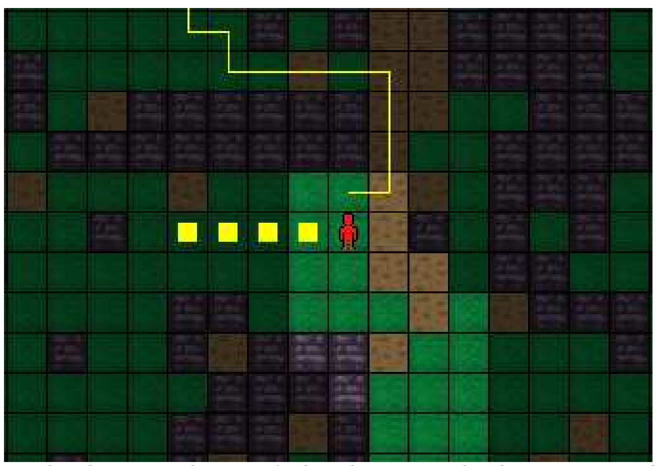

Figure 4: User Action: Example showing the user's local area and what actions they can take. At any given moment, the user can select their own path sequence (the yellow squares), or they can follow the agent's suggested path (the yellow line). In this case, the player may have determined that the agent's path through the difficult squares ahead could be less efficient than circumventing the obstacle by moving left.

Perhaps the most influential factor that a player must consider and one of the major aspects that affects a player's trust in an agent, is the imperfect nature of the information provided by the agent. In addition to trusting a drone's accuracy of its view of the world, a player might also have to consider whether to trust other information the drone might provide (e.g., path planning). In the real world, a drone may suggest that a path is blocked because a canopy of trees is obfuscating its view, while a human teammate may easily find a path through the trees on the ground level. A real, agent-controlled drone may also fail to identify a possible hazard, such as a fallen tree or a treacherous river, that can cause the human to have to find an alternate path. These types of example situations for imperfect information sharing from the agent are representative of the imperfect world-view a drone agent has from its bird's-eye-view and are vital for assessing how a human teammate's trust in an agent is affected and how that teammate will rely on an agent in future interactions. Due to their importance, similar imperfect information sharing situations were implemented in the game environments. For example, there might be cases where the drone recommends circumventing an impassable object that is not actually there, or it suggests that a human teammate walk through an impassable obstacle. Ideally, because we are using a grid world for the game, we should be able to tell if a player is blindly following a recommended path (blind trust) or completely ignoring path suggestions (distrust or untrust).

4. AGENTS

An overarching question of this work is what is an ideal agent that can be paired with a human to efficiently and effectively complete a cooperative multi-objective task. To explore this question, two agents were created. The first, “Agent 1” is considered basic technology (e.g., a drone) that has basic sensing but no onboard intelligence. The sensing capabilities are error prone and may not provide accurate environment information. The second agent, “Agent 2” has basic sensing similar to Agent 1 and is capable of proposing planned paths through the environment for optimal task completion. The remainder of this section describes these levels in further detail and how they ultimately impact the research outlined in this article.

“Agent 1” Eye in the Sky: “Agent 1” represents a simple drone technology, with low-level autonomy for way-point navigation that mirrors drones currently being used in the field today. The agent performs an initial flyover of the map performing a raster scan to get an overview of the world and broadcasts that perspective to a human team member. It also locates the target and loiters nearby while awaiting instruction to perform searches of identified areas. It essentially acts as an “eye in the sky” for a human team member. An example is a drone with a camera operated using a flight controller with way-point navigation capabilities. Intelligent features of the agent may include low-level signal processing (GPS, IMU, etc.), navigation, and obstacle detection/avoidance. We decided to start with a simple agent to determine the level of assistance it could provide as a team member, relative to a more complex and capable autonomous agent.

“Agent 2” Path Planner: The second agent has the same capabilities as Agent 1 with the addition of optimal path recommendations. Similar to Agent 1’s capabilities, Agent 2 completes an initial flyover of the map and relays that view to a human team member; it locates the target and loiters near the target while awaiting instructions to perform searches of identified areas. The key distinction is that in addition to Agent 1 behaviors, it calculates the player’s optimal path to the target from the player’s current location from the agent’s perspective of the map. It then displays that path to the player (the yellow line in Figure 4). This is meant to emulate a more advanced real-world agent with some path planning capability, allowing the agent to propose the safest, most efficient path from the human teammate’s location to the goal. An advantage of using a 2D grid world representation is that we can simulate advanced behaviors using an optimal path-planning algorithm. The “best” path for the human would be one that expends the least amount of energy and time needed to reach the goal. This is accomplished by using path planning with weighted terrain types in the 2D grid world environment. Due to the relatively small size of the graph that makes up the game, we used Dijkstra’s algorithm because the computational time reduction obtained by further optimizing this search using for example an A* search was negligible. The path planning capability of Agent 2 allowed us to focus on information sharing and trust in human-robot teaming for this study. The agent’s recommended path was optimal, but since the agent’s view of the world was prone to some level of error, the presented path may not actually be the best path for the human to take to reach the goal.

5. ENVIRONMENT DESIGN

We designed the maps for our study with three levels of difficulty in mind: easy, medium, and hard. The goal of having multiple environments of varying complexity is to study how a player’s trust changes across various decision points encountered while traversing the environments. For instance, if the task is too simple and Agent 2 is too effective at providing a route, the player may be inclined to trust the agent blindly. Inversely, a near-impossible task with a less helpful agent would likely cause a player’s trust to plummet to the point of complete disregard for that agent. Not only do we want to learn how a player’s trust is influenced by certain events, but we also want to study how a player’s trust is impacted across multiple environments of varying difficulty, when a player stops trusting the agent, if and when a player’s trust is recovered, and if there is a breaking point at which a player’s trust cannot recover. We expect to see results to these inquiries through randomizing and counterbalancing the order the environments are presented to the players. To this end, each participant is assigned an easy, medium, and hard environment map in a random order. The factors that determine the difficulty of any given map are the complexity of obstacle/terrain placement, the severity of the inaccuracies of the drone's view of the map, and the margin for error allowed before the user fails the mission. In this section, we outline simplified views of maps that we consider to be easy, medium, and hard. These are not the true representations of maps used in our study but are instead meant to illustrate our design philosophy when creating the final maps.

The remainder of this section explains how a player might traverse through an easy, medium, and hard environment with Agent 2 as a teammate. Examples are not given for teams with Agent 1 because environment traversal in those instances are self-guided. Regardless, players assigned to either agent will be presented with the same environments and will encounter many of the same decision points. In all of the example images below, the blue circle represents the player's location, the yellow lines represent possible paths suggested by Agent 2, the green 'X' represents the goal location, and the solid black lines represent impassable obstacles. Dotted black lines are impassable obstacles that the agent falsely detected, and red lines are impassable obstacles that the agent did not detect. Finally, orange areas (medium and hard examples) are places of difficult terrain, which a player can move through at a greater cost to their energy level.

5.1 Easy Map

A map is deemed "easy" based on the distribution of obstacles and difficult terrain being very clumped, scattered, and sparse. Obstacles and difficult terrain are clumped in what look like giant land masses. The small portion of "land masses" present in an easy map tend to be spread out with a lot of open space around them for easy environment traversal. The final component of environment difficulty is the severity of errors made by the agents. These kinds of errors include minor mislabeling of terrain near or in the path of a player without affecting the path a player is taking. Another example would be mislabeling a patch of terrain as an obstacle instead of a gap in an obstacle in an area away from a player's original optimal path but in a spot where an alternate path could have existed. The easy environment was designed to take no more than five minutes to traverse, and players assigned to either agent should have no difficulty reaching the target without instructing the drone to perform extra searches. Figure 5 shows an example of an easy environment with some of the decision points a player might encounter.

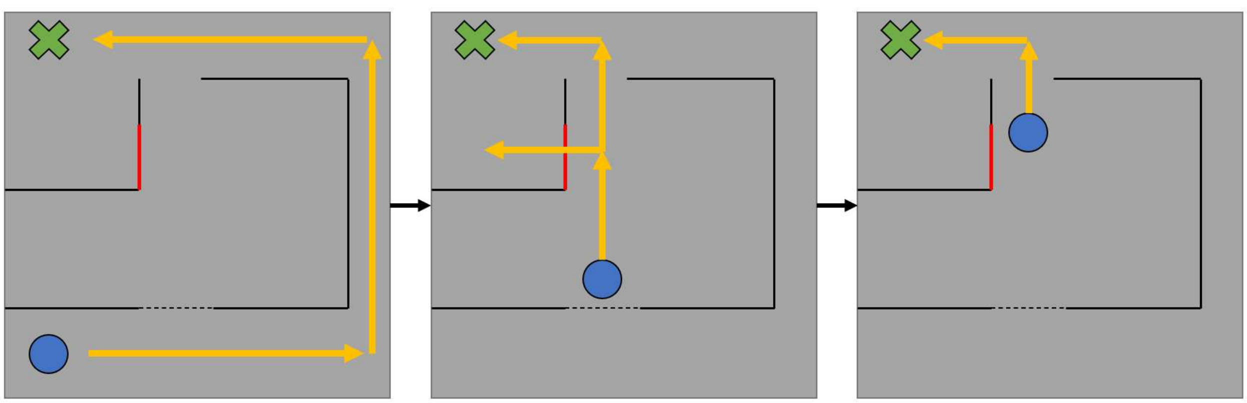

Figure 5: Easy Map: The start state of the game (left), the game state if the player moves through the falsely labeled obstacle (middle), and the state of the game as the player moves past the previously unseen obstacle (right).

In this example, the agent offers a path that is optimal based on mostly-accurate information gathered during an initial flyover of the environment. In truth the path is not optimal, due to minute inaccuracies from the initial scan of the environment, but it will still allow the player to reach the goal. However, the player can get a slightly better score (time- and energy-wise) by paying attention to their surroundings as they follow the agent's suggested path. The gap in the obstacle is easily visible to the player as they move to the right, and they have the choice to move through that space instead of circumventing the entire obstacle as the agent suggests. As this player moves through the previously undetected gap in the obstacle, the agent provides a new optimal path based on the player's current position (the middle image in Figure 5). To the agent, the two new paths it could provide are equivalent, but as the player continues toward the goal, they will see that there is an obstacle blocking their path to the left. With no other option, the player continues along the straight path to the top and reaches the goal with little difficulty (the right image in Figure 5). Since the player is able to reach the goal successfully without much deliberation and is provided a fast, straightforward route by the drone, we consider this scenario to be easy.

5.2 Medium Map

A medium map has larger masses of obstacles and difficult terrain. There is less open space to easily traverse the environment, and the path to the goal might not appear as straightforward as in the easy environment. The initial scan of the environment includes more mislabeled areas than in the easy map. Such mislabelings could include at least one semi-serious overlooked gap in an obstacle adjacent to the initial, optimal path provided by the agent. The mislabeled gap could open to an alternate, more direct path. Inversely, an obstacle may be mislabeled as a gap that the player wants to pass through to reach the goal. Other types of inaccuracies include mislabeling difficult terrain for open terrain and vice versa. Some of these patches of terrain could cause other paths to be the better choice. This environment was designed to take no longer than seven minutes to traverse, and to reach the target, players assigned to Agent 1 will most likely need to instruct the drone to perform more extra searches than the players assigned to Agent 2. Figure 6 displays an example traversal for a medium difficulty environment.

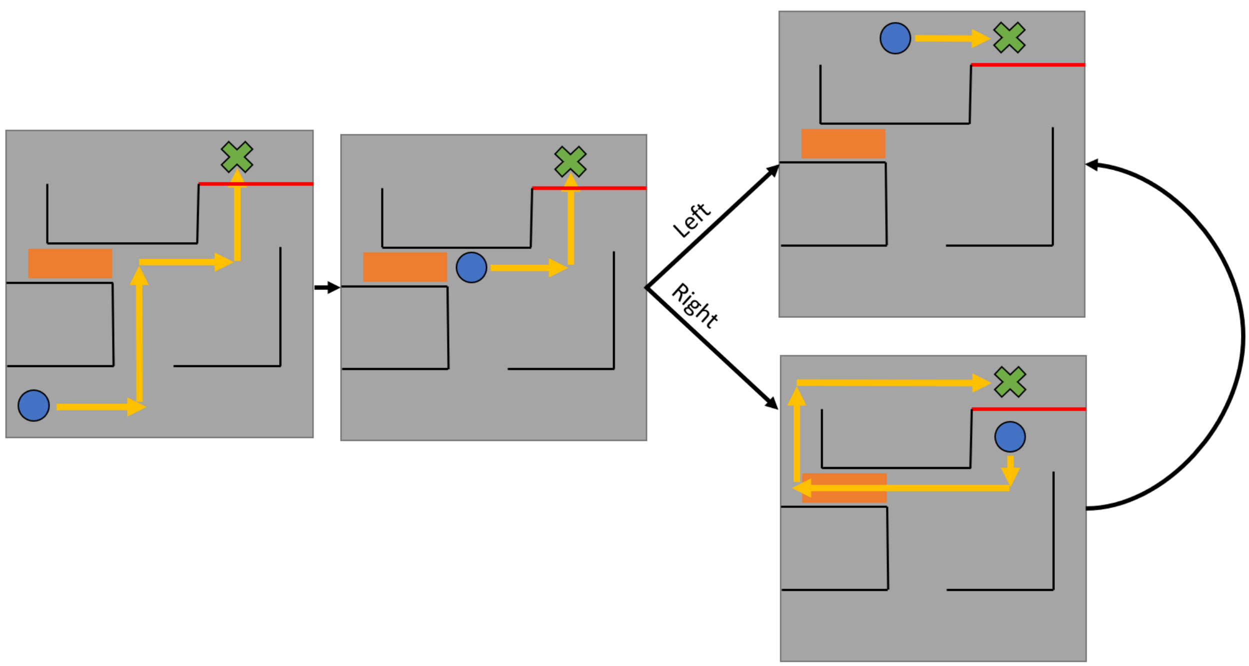

Figure 6: Medium Map: The start state of the game (left), the game state as the player reaches a major decision (middle), and the two outcomes if the player moves to the right or the left (right).

As the difficulty increases, the player is required to make more impactful decisions. Here, the player follows the agent's path with little issue until reaching a crossroads (middle image in Figure 6). The agent suggests they move to the right, and the player may be inclined to agree as they are able to see the difficult terrain to their left. However, following the agent's suggestion leads to a mislabeled dead-end and would cause the player to have to backtrack and move through the difficult terrain anyway. A cautious player may send the drone to re-investigate the area around the goal before making a decision, and thus avoid wasting energy on following the proposed path. A less cautious player, or one who may have too much trust in the drone's accuracy, may continue to follow the drone's path to the detriment of their performance. With a map designed for medium difficulty, the player should be able to make a few of these mistakes without completely failing the mission, but failure is still possible if they repeatedly require backtracking and reduce the human character's energy to zero.

5.3 Hard Map

A hard environment is one that is nearly impossible to see the optimal path to the goal, however players will most likely be able to see at least one path to the goal. In a hard environment, obstacles and difficult terrain are more heavily scattered than clumped together. There is little open space for environment traversal, and all potential paths will lead through a lot of difficult terrain. The agent's initial view of the map is more error-prone than the previous environments and players are allotted just enough energy capacity to only afford a few mistakes. The environment is designed to take no more than ten minutes to complete, although failure is more likely in this environment than in the others. The types of inaccuracies made by the agents include ones that cause a player to do a lot of backtracking on a path. A lot of obstacles are mislabeled in very critical points on potential paths. Players of both agent assignments will need to instruct the drone to perform extra area searches in order to reach the target. Because of this, the drone is expected to run out of battery before the human reaches the target. Essentially, this is the environment designed to test for a breaking point in players' trust. Figure 7 displays the kinds of choices a player might encounter in a hard environment.

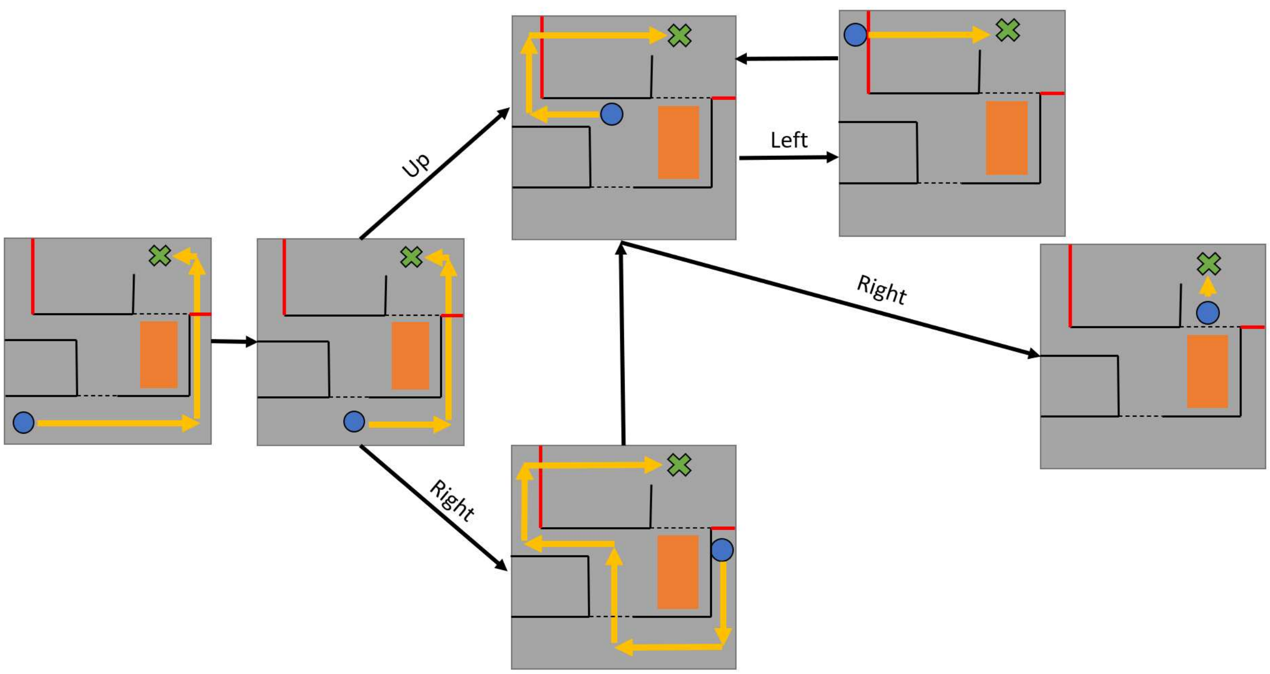

Figure 7: Hard Map: The game state flow as a player explores a hard environment.

In our hard environments, there may be several dead-ends caused by imperfect agent information that can lead to significant energy expenditure and would likely result in a failed mission. Following the flow chart in Figure 7, the agent's initial path requires the player to move through an undetected obstacle. Following the initial path, they might notice that there is a gap in the obstacle above them that the agent did not notice. If they continue along the agent's path, they'll end up having to backtrack to this spot. Moving upward offers an alternative route for the player to traverse. Once through the gap, they can see difficult terrain to the right, and based on the drone's view of the world they may also think the path is blocked in that direction. An attentive player may use this time to send the drone to re-investigate the left or the right path before making a decision, but another player may continue following the drone's suggested path only to meet another dead-end. Notably, if the player reaches this dead-end, the goal is impossible to reach from the agent's perspective since it believes the false obstacle below the goal exists. The only remaining path is to navigate through the difficult terrain, revealing that the false obstacle is traversable, and the goal can be reached. Maps designed for this difficulty level require the player to be attentive, to utilize their agent teammate, and to question their trust in their agent teammate more frequently.

5.4 Example Environment

Figure 8 shows the player's view of the map on the left and the ground truth map on the right with additional annotations around major errors in the drone's perspective. We will describe how both a "Naive player" and a "Cautious player" may experience this particular environment in order to illustrate how a single map can provide us different insights depending on how the player interacts with the environment.

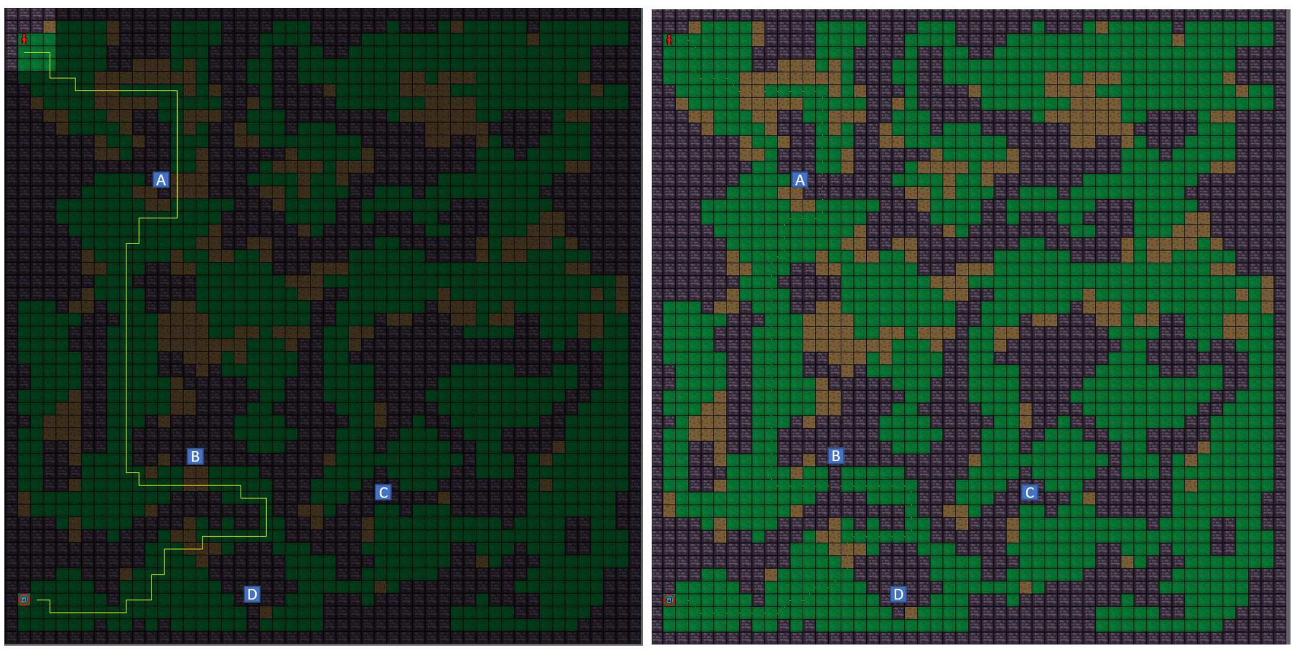

Figure 8: The start state of an example map including the initial recommended path (left) and the "ground truth" view of the map that is unseen by the player (right).

At the start of the game, the player may have no reason to fully trust, distrust, or untrust the agent's recommended path. However, the Cautious player may decide to send the agent to re-investigate the area around site A before committing to the left or the right path and discovers that only the left path is passable. The Naive player immediately follows the agent's recommendation, hits a dead-end, and is forced to backtrack to find that the left path is in fact passable. This first obstacle tells the player that the agent's perspective isn't always perfect and they may need to utilize all of the tools at their disposal to reach the goal. This is especially apparent as both players approach site B. The Cautious player may be suspicious of the difficult terrain that the agent believes is below site B, especially if in their investigation they discovered that the path to the right of site A was an impassable cluster of obstacles that the agent mislabeled as difficult terrain. Again, the Cautious player decides to send the drone to re-investigate the area around site B and chooses to head to the right rather than following the path to a dead end. The Naive player decides that the agent's error at site A was a fluke and elects to follow the recommended path, only to hit a wall at site B and be forced to backtrack toward site A before moving to the right of the map with a large hit to their energy.

At this point, the Cautious player has made liberal use of the drone, and the drone's battery is too low for them to check if the fork in the road that leads to site C is clear or not. They decide to circumvent site C by continuing along to the right edge of the screen instead of expending energy to walk down to site C and check if it is clear with their own eyes. The Naive player's drone has battery life to spare as they didn't make use of the drone at all to get past sites A and B, so they decide to send the drone down the path toward site C and happen to reveal the shortcut, managing to conserve a considerable amount of energy. As the players approach site D, The Cautious player may decide to check the lower path because the agent has been incorrect previously, and the risk of losing too much energy to backtracking appears to be low. The Cautious player makes their way to the goal from here without incident after discovering that the obstacle in the agent's map does not exist. The Naive player decides to trust the agent's path again, and loses a bit more energy going through the difficult terrain on the upper path but otherwise reaches the target.

6. HUMAN-ROBOT INTERACTION

The successful completion of complex, multi-objective tasks in dynamic or unknown environments requires extensive planning and adaptive teamwork. For a team involving humans and autonomous systems to perform a task safely and efficiently, trust needs to be established and transparency needs to be maintained. When performing a task, a human and autonomous machine work toward the same overall goal while remaining transparent in their communication and operation. The human and machine might work asynchronously or have different sub-goals, but an appropriate level of trust and transparency should be established for both parties from the beginning. If either teammate experiences degraded transparency or information sharing from another teammate, trust will likely start to break down.

When working with autonomous systems, humans must be able to calibrate their trust based on the reliability of the system.[24] In an environment filled with unforeseeable events, a system's reliability changes to adapt to uncertainty, and the human's trust in the system is tested. Ideally in this situation, a human will re-calibrate their trust to match the system's new (current) reliability. This iterative process of trust calibration is necessary for successful cooperation in human-machine teams. Also, it is often the point where humans fail. Their inability to update their trust level in time with a machine's actions (or inactions) is the event that leads humans to over- or under-trust an autonomous system. Excessively trusting a system could result in an inefficient performance where a human might have been able to out-perform the system. Alternatively, blind or excessive trust in a system could result in an intelligent machine exploiting a human's trust to achieve an unknown objective.[25] Insufficiently trusting a system, however, could mean ignoring the system completely when it might know pertinent information. The same concepts apply to an autonomous system over- and under-trusting its human team members. Neither of these levels of trust are ideal for humans and autonomous systems working together. With this in mind, we set out to create a human-robot teaming model that encourages the incorporation of trusted autonomous systems into human-robot teams. Our innovative approach to formulating such a model can be seen in Figure 9. Because we plan to utilize autonomous systems, trust and authority were taken into consideration. We want the agents of human-robot teams to work together without a strict hierarchy of authority. To achieve this, maintaining trust and transparency between the robot and the human is vital.

The proposed human-Robot Teaming Model is a shared meta-model (Figure 9) and is designed to handle teams of varying agent configurations: one-to-one, one-to-many, and many-many.[4] Due to its scalability, the framework is applicable for teams utilizing mixed types of unmanned systems (UxSs) (e.g., ground, aerial, surface, underwater) and systems of varying levels of autonomy, including cognitively-compatible intelligent systems. With intelligence comes independent thinking and interdependent decision-making. To keep agents informed and to increase transparency within a team, shared mental models and trust evaluations will help agents predict how their teammates will act and how they, themselves, should act as well as. Shared mental models also offer an "explainable" channel of communication where reasoning and intent can be provided for unexpected or sudden actions. While mental models are helpful, especially for situations where there is a loss of communications, the majority of functionality within out human-robot teaming model relies on trust. Because trust plays a vital role in successful, cooperative teaming,[26,27] we have integrated a novel multi-directional trust-based calibration to allow for human-human, human-system, system-system, and system-human trust evaluations. As previously stated, with intelligence comes independent thinking and interdependent decision-making. If a team plans to incorporate cognitively-compatible, intelligent systems, and trust influences human reliance on autonomous systems,[27] then perhaps teaching intelligent systems to trust will provide a means of influencing intelligent systems to rely on their human teammates and others.

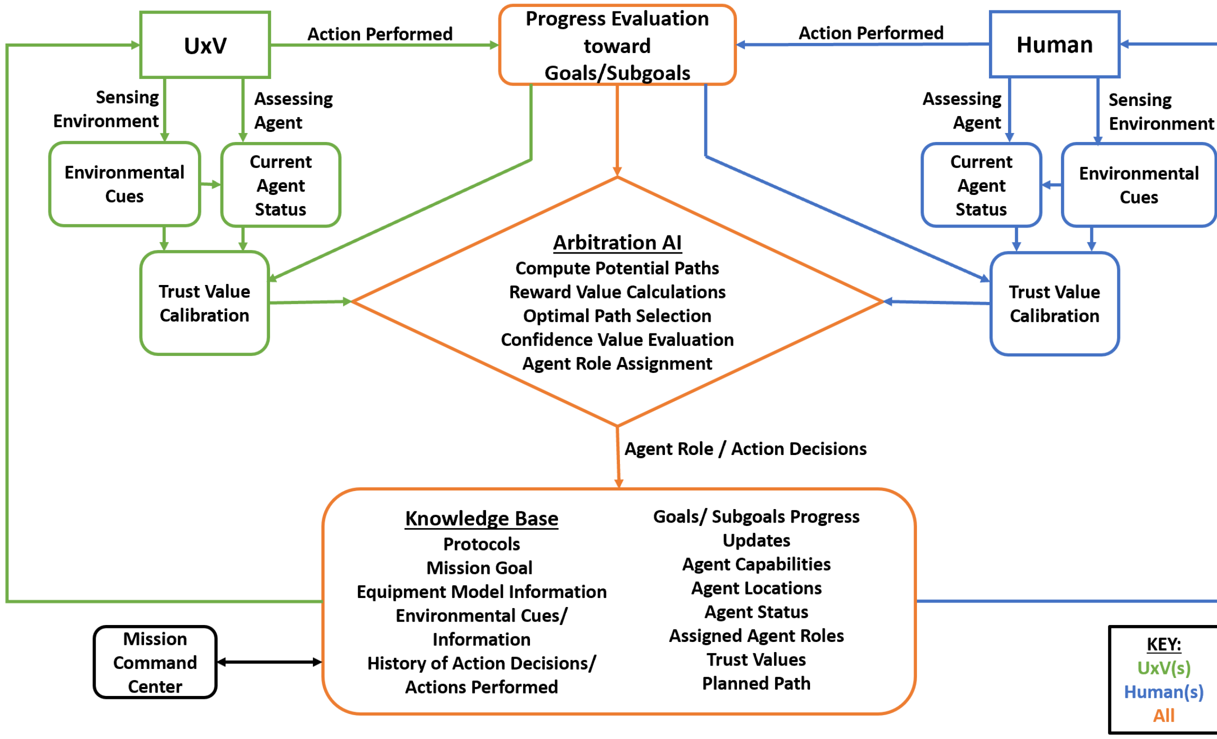

Figure 9: Proposed human-Robot Teaming Model

7. STUDY MEASURES AND METRICS

The goal of this study is to determine how performance, efficiency, situational awareness, and cognitive load are affected by personality, information sharing, and trust in human-machine teams completing multi-objective tasks. With this information, optimized teaming factors, such as information sharing and trust calibration, can be determined and incorporated into our human-robot teaming model designed for enhancing collaboration, cooperation, performance, efficiency, and safety in human-machine teams. To begin formalizing the framework, we must investigate how performance and efficiency are impacted by information sharing, trust, cognitive load, and situational awareness.

Table 1: Gaming features measured during study.

Data Logged Per Study	
Success or Failure	Indicates whether the player was successful or not in completing the task.
Total Moves	Total number of moves the player made before the end of the current task.
Path Convergence	Degree of overlap between the player's chosen path and the agent's suggested path.
Path Taken	Exact path that the player took through the environment.
Squares Revisited	List of grid squares that the player "backtracked" to.
Final Energy	The human character's remaining energy at task completion.
Final Charge	The drone's final battery level at task completion.
Game Duration	Total time elapsed for this task.

For this study, we capture various metrics, seen in Table 1, to evaluate how a player's intent, trust and performance are all connected. We begin by investigating how trust, performance, and information sharing affect each other in three environments of varying levels of difficulty and with two types of agents (see section 4). Performance and efficiency between players and their assigned agent type will be studied through mission status (success or failure), total moves, final human energy, final drone charge, and game duration. By using two agent types, "Agent 1" and "Agent 2", we hope to show how working with an agent of higher intelligence can benefit a human-robot team's performance, efficiency, and safety when completing a task. How a human user trusts an agent (and the information it shared) and how that trust changes after a successful or failed mission will also be studied. We can determine if a player was blindly trusting the agent or following their own intended path by their path convergence, path taken, and squares revisited. Along with these metrics captured directly from the game, we plan to measure and assess personality type, trust, workload, cognitive load, and situational awareness through the dual-task paradigm and pre- and post-game surveys.

This research is a mixed-model study design. For the between-subjects factor, the participants will be randomly assigned to one of two drone conditions (Agent 1 or Agent 2) that were previously discussed. Participants assigned to the Agent 1 teammate will receive no path planning help when traversing an environment, while participants assigned with the Agent 2 teammate will receive path planning help. In both conditions, the drone will do an initial flyover of the environment to provide a map of the terrain, obstacles, and target location. Players from both conditions can call upon their drone teammate to search a particular area that the player would like investigated. When the drones are called upon to search an area, the drone provides an accurate update of the area, regardless of the environment difficulty and the drones' corresponding accuracies for that environment.

Table 2: Study metrics collected pre- and post-task.

Metric	Interest
Big Five Personality Test	five domains that comprise most personality traits
Trust/Untrust Scale	measures pre-existing bias toward autonomous machines
HRI Trust Scale	measures trust in a specific robot
SWAT Scale	subjectively measures workload
SOS Scale	subjectively measures experienced cognitive load
Dual-Task Paradigm	objectively measures cognitive load
SAGAT Queries	objective measure of situational awareness
SART Assessment	subjective measure of situational awareness

Measures of evaluation (seen in Table 2) that will be recorded during this study include team/task performance, personality, trust, workload, cognitive load, and situation awareness. Analyzing team performance will reveal how our shared meta-model framework affects performance, especially as more components of the framework are added to later studies. Study metrics used to assess situation awareness and cognitive load will be modified to acquire additional information regarding task allocation and information sharing/understanding. By learning how and what type of information impacts trust and performance, we can optimize information sharing and trust calibration in our framework.

Real-time SAGAT (Situation Awareness Global Assessment Technique) queries will be used to assess situation awareness and the understanding and sharing of information.[28–32] This hybrid technique of asking SAGAT-type queries via real-time probes allows for situation awareness to be objectively measured concurrent to ongoing operations during a mission. Cognitive load assessment will use an adapted SOS scale, like used in Ref. 33, paired with the dual-task paradigm. The SOS scale provides a subjective measure of experienced cognitive load. It is designed to gather insight into the understandability of subject matter, system operation ease of use, and whether support tools aid subject matter comprehension.[34] In conjunction with the SOS scale's measure of cognitive load, the dual-task paradigm, which involves performing two tasks simultaneously, will objectively measure cognitive load by recording response time and response accuracy.[35] Studying these cognitive load measures will reveal the optimality of task allocations.

A participant's trust in general autonomous systems will be measured with the Trust/Untrust scale developed by Ref. 36, while trust in an assigned drone teammate will be measured with the HRI Trust scale developed by Ref. 37. Blind trust will be measured by comparing the number of times a human user moves in a different direction than the drone's suggested 'optimal' path. In this study, we want to see how preconceived notions of autonomous systems in general and preconceived notions learned from exposure to previous environments affect trust across all environments and at critical points of decision-making during the search task. We expect to see a correlation in personality, cooperative behavior during the task, and trust in an assigned drone agent.

8. CONCLUSIONS AND FUTURE WORK

In a multi-objective, multi-agent, human-robot planning problem, understanding and explaining how teammate actions impact goal attainment and how actions are interdependent among teammates will assist with effectively planning, executing, and evaluating missions in complex environments and with supporting the uptake of trusted autonomous systems in defense teams. In this paper, we laid the foundational framework for a 2D game designed for studying HRI in teams following our human-robot teaming model. We present a game interface that allows us to measure trust-related features as human participants collaborate with increasingly advanced agents by abstracting the agent's contribution to a mission in a simplified grid-based world. This study is the first to test and evaluate our model by fine-tuning the cooperative, multi-task game (presented in this paper) and specific teaming factors, such as information sharing and trust, that are vital to successful, collaborative human-robot teaming. The proposed game sets the stage for future research into our human-robot teaming model's functionality and allows us to explore human-robot interactions that become more and more relevant as autonomous agents increase in cognitive intelligence over time.

Moving forward, we intend to clearly define the intelligence and capabilities specific to each agent type used as teammates in the human-robot game. As more AI and higher levels of autonomy are integrated into the drone, the intelligence across agent types should increase. These defined agent traits will likely determine in which order our human-robot teaming model's components are formalized and tested. We also plan to outline the characteristics that distinguish easy, medium, and hard environments. Creating these more stringent definitions will allow for scalable gaming environments that can adapt to future agent types and study protocols. They will also provide a guide for consistent designs across studies.

After an initial pilot study is run, further refinement to the game may be required if unknown weaknesses within our gaming framework are revealed. Upon Institutional Review Board approval, we will conduct a full-scale study using the 2D grid-world game outlined in this paper. The data collected from this study will provide a baseline for future testing of model components, such as shared mental models and multi-directional trust-based evaluation. After some or all major components of our model are formally developed, we will extend our game's framework into a 3D virtual simulation to create a more direct parallel to real-world applications. Because our model is designed to support multi-agent teaming, we will perform testing in the 3D virtual environment on heterogeneous, multi-agent human-robot teams. An interface for communication between the human and robot(s) will need to be implemented and tested. Once comfortable with our model's formalization, we plan to integrate AI and higher levels of autonomy onto an Unmanned Aerial Vehicle and validate our model in the field with a human-robot team performing a multi-objective search and rescue task.

From recent exploration in human-robot teaming trust literature, it is clear that trust, being an abstract concept, is always defined based on its application, making it too subjective for universally reliable metrics. Henceforth, our research team has moved on from measuring "trust" and will be approaching the multi-directional trust-based calibration component of our model from a different angle for all future work. We plan to design a novel evaluation that is trust-adjacent and adaptable for most human-robot teams.

REFERENCES

[1] Huang, L., Cooke, N. J., Gutzwiller, R. S., Berman, S., Chiou, E. K., Demir, M., and Zhang, W., "Chapter 13 - distributed dynamic team trust in human, artificial intelligence, and robot teaming," in [*Trust in Human-Robot Interaction*], Nam, C. S. and Lyons, J. B., eds., 301–319, Academic Press (2021).

[2] Zhang, G., Chong, L., Kotovsky, K., and Cagan, J., "Trust in an ai versus a human teammate: The effects of teammate identity and performance on human-ai cooperation," *Computers in Human Behavior* **139**(107536) (2023).

[3] Seeber, I., Bittner, E., Briggs, R. O., de Vreede, T., de Vreede, G.-J., Elkins, A., Maier, R., Merz, A. B., Oeste-Reiß, S., Randrup, N., Schwabe, G., and Söllner, M., "Machines as teammates: A research agenda on ai in team collaboration," *Information Management* **57**(2), 103174 (2020).

[4] Aldridge, A. L. and Bethel, C. L., "M-oat shared meta-model framework for effective collaborative human-autonomy teaming," in [*Companion of the 2023 ACM/IEEE International Conference on Human-Robot Interaction*], *HRI '23*, 663–666, Association for Computing Machinery, New York, NY, USA (2023).

[5] Resnick, C., Eldridge, W., Ha, D., Britz, D., Foerster, J., Togelius, J., Cho, K., and Bruna, J., "Pommerman: A multi-agent playground," *CEUR Workshop Proceedings* **2282** (2018).

[6] Lowe, R., Wu, Y., Tamar, A., Harb, J., Abbeel, P., and Mordatch, I., "Multi-agent actor-critic for mixed cooperative-competitive environments," in [*Proceedings of the 31st International Conference on Neural Information Processing Systems*], *NIPS'17*, 6382–6393, Curran Associates Inc., Red Hook, NY, USA (2017).

[7] Leibo, J. Z., Zambaldi, V., Lanctot, M., Marecki, J., and Graepel, T., "Multi-agent reinforcement learning in sequential social dilemmas," in [*Proceedings of the 16th Conference on Autonomous Agents and MultiAgent Systems*], *AAMAS '17*, 464–473, International Foundation for Autonomous Agents and Multiagent Systems, Richland, SC (2017).

[8] Jia, H., Hu, Y., Chen, Y., Ren, C., Lv, T., Fan, C., and Zhang, C., "Fever basketball: A complex, flexible, and asynchronized sports game environment for multi-agent reinforcement learning," *ArXiv* **abs/2012.03204** (2020).

[9] Yang, Y., Luo, R., Li, M., Zhou, M., Zhang, W., and Wang, J., "Mean field multi-agent reinforcement learning," in [*Proceedings of the 35th International Conference on Machine Learning*], Dy, J. and Krause, A., eds., *Proceedings of Machine Learning Research* **80**, 5571–5580, PMLR (10–15 Jul 2018).

[10] Samvelyan, M., Rashid, T., de Witt, C. S., Farquhar, G., Nardelli, N., Rudner, T. G. J., Hung, C.-M., Torr, P. H. S., Foerster, J., and Whiteson, S., "The starcraft multi-agent challenge," in [*Proceedings of the 18th International Conference on Autonomous Agents and MultiAgent Systems*], *AAMAS '19*, 2186–2188, International Foundation for Autonomous Agents and Multiagent Systems, Richland, SC (2019).

[11] Kurach, K., Raichuk, A., Stańczyk, P., Zajac, M., Bachem, O., Espeholt, L., Riquelme, C., Vincent, D., Michalski, M., Bousquet, O., and Gelly, S., "Google research football: A novel reinforcement learning environment," *Proceedings of the AAAI Conference on Artificial Intelligence* **34**, 4501–4510 (Apr. 2020).

[12] Yao, M., Yin, Q., Yang, J., Yu, T., Shen, S., Zhang, J., Liang, B., and Huang, K., "The partially observable asynchronous multi-agent cooperation challenge," (2021).

[13] Balakirsky, S., Scrapper, C., Carpin, S., and Lewis, M., "Usarsim: Providing a framework for multi-robot performance evaluation," Performance Metrics for Intelligent Systems Workshop— 2006 — PerMIS, Gaithersburg, MD (January 2006).

[14] Lewis, B. and Sukthankar, G., "Rsarsim: A toolkit for evaluating hri in robotic search and rescue tasks," *Florida Conference on Recent Advances in Robotics, FCRAR 2009* (May 2009).

[15] Eaton, E., "Gridworld search and rescue: A project framework for a course in artificial intelligence," in [*Proceedings of the AAAI-08 AI Education Colloquium*], (13 Jul 2008).

[16] Peschl, M., "Training for implicit norms in deep reinforcement learning agents through adversarial multi-objective reward optimization," in [*Proceedings of the 2021 AAAI/ACM Conference on AI, Ethics, and Society*], *AIES '21*, 275–276, Association for Computing Machinery, New York, NY, USA (2021).

[17] Abdelfattah, S., Kasmarik, K., and Hu, J., "Evolving robust policy coverage sets in multi-objective markov decision processes through intrinsically motivated self-play," *Front. Neurorobot.* **12** (Oct 2018).

[18] Hao, B., Zhao, J., Du, H., Wang, Q., Yuan, Q., and Zhao, S., "A search and rescue search method based on flower pollination algorithm and q-learning fusion algorithm," *PLoS ONE* **18**(3) (2023).

[19] Rahman, A., Bhattacharya, A., Ramachandran, T., Mukherjee, S., Sharma, H., Fujimoto, T., and Chatterjee, S., "Adversar: Adversarial search and rescue via multi-agent reinforcement learning," (2022).

[20] Bihl, T. J., Jones, A., Farr, P., Straub, K., Bontempo, B., and Jones, F., "Assessing multi-agent reinforcement learning algorithms for autonomous sensor resourse management," in [*Proceedings of the 55th Hawaii International Conference on System Sciences*], 10 (January 2022).

[21] Liu, M., Sivakumar, K., Omidshafiei, S., Amato, C., and How, J., "Learning for multi-robot cooperation in partially observable stochastic environments with macro-actions," 1853–1860 (09 2017).

[22] Boggess, K., Kraus, S., and Feng, L., "Toward policy explanations for multi-agent reinforcement learning," in [*Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence (IJCAI-22)*], 109–115 (2022).

[23] Buck, A., "Multicriteria pathfinding in uncertain simulated environments, ph.d. dissertation," (2018).

[24] Okamura, K. and Yamada, S., "Calibrating trust in human-drone cooperative navigation," in [*2020 29th IEEE International Conference on Robot and Human Interactive Communication (RO-MAN)*], 1274–1279 (2020).

[25] Losey, D. P. and Sadigh, D., "Robots that take advantage of human trust," in [*2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*], 7001–7008 (2019).

[26] Parasuraman, R. and Manzey, D. H., "Complacency and bias in human use of automation: an attentional integration," *Hum. Factors* **52**, 381–410 (Oct. 2010).

[27] Lee, J. D. and See, K. A., "Trust in automation: Designing for appropriate reliance," *Hum. Factors* **46**(1), 50–80 (2004).

[28] Endsley, M. R., "Measurement of situation awareness in dynamic systems," *Hum. Factors* **37**(1), 65–84 (1995).

[29] Endsley, M. R., "Situation awareness global assessment technique (sagat)," in [*Proceedings of the IEEE 1988 National Aerospace and Electronics Conference*], **3**, 789–795 (1988).

[30] Endsley, M. R., "Theoretical underpinnings of situation awareness: A critical review," *Situation awareness analysis and measurement* , 3–32 (2000).

[31] Jones, D. G. and Endsley, M. R., "Can real-time probes provide a valid measure of situation awareness?," in [*In Proceedings of the Human Performance, Situation Awareness and Automation: User Centered Design for the New Millennium Conference*], (2000).

[32] Jones, D. G. and Endsley, M. R., "Use of real-time probes for measuring situation awareness," *The International Journal of Aviation Psychology* **14**(4), 343–367 (2004).

[33] Eysink, T. H. S., de Jong, T., Berthold, K., Kollofel, B., Opfermann, M., and Wouters, P., "Learner performance in multimedia learning arrangements: An analysis across instructional approaches," *American Educational Research Journal* **46**(4), 1107–1149 (2009).

[34] Swaak, J. and de Jong, T., "Learner vs. system control in using online support for simulation-based discovery learning," *Learning Environments Research* (4), 217–241 (2001).

[35] Klepsch, M., Florian, S., and Tina, S., "Development and validation of two instruments measuring intrinsic, extraneous, and germane cognitive load," *Front. in Psych.* (8) (2017).

[36] Swaak, J. and de Jong, T., "Foundations for an empirically determined scale of trust in automated systems," *International journal of cognitive ergonomics* **4**(1), 53–71 (2000).

[37] Gulati, S., Sousa, S., and Lamas, D., "Design, development and evaluation of a human-computer trust scale," *Behaviour and Information Technology* **38**(10), 1004–1015 (2019).