

FRAMEWORK FOR ANALYZING HUMAN COGNITION IN OPERATIONALLY-RELEVANT HUMAN SWARM INTERACTION

Joseph P. Distefano¹, Hanvit Cho¹, Prajit KrishshnaKumar¹, Souma Chowdhury¹, Ehsan Esfahani^{1,*}

¹Department of Mechanical and Aerospace Engineering
University at Buffalo
Buffalo, NY

ABSTRACT

For a wide variety of envisioned humanitarian and commercial applications that involve a human user commanding a swarm of robotic systems, developing human-swarm interaction (HSI) principles and interfaces calls for systematic virtual environments to study such HSI implementations. Specifically, such studies are fundamental to achieving HSI that is operationally efficient and can facilitate trust calibration through the collection–use–modeling of cognitive information. However, there is a lack of such virtual environments, especially in the context of studying HSI in different operationally relevant contexts. Building on our previous work in swarm simulation and computer game-based HSI, this paper develops a comprehensive virtual environment to study HSI under varying swarm size, swarm compliance, and swarm-to-human feedback. This paper demonstrates how this simulation environment informs the development of an indoor physical (experimentation) environment to evaluate the human cognitive model. New approaches are presented to simulate physical assets based on physical experiment-based calibration and the effects that this presents on the human users. Key features of the simulation environment include medium fidelity simulation of large teams of small aerial and ground vehicles (based on the Pybullet engine), a graphical user interface to receive human command and provide feedback (from swarm assets) to human in the case of non-compliance with commands, and a lab-streaming layer to synchronize physiological data collection (e.g., related to brain activity and eye gaze) with swarm state and human commands.

Keywords: Human Swarm Interaction, Brain Cognition

1. INTRODUCTION

Advances in Artificial Intelligence (AI) have significantly improved human-AI partnership in different capacities ranging

from augmenting human capabilities (e.g., human-swarm interaction[1]), to providing aids in decision making (e.g., medical decision making [2]), to fully automating tasks under human supervision (e.g. autonomous driving [3]). [4].

In goal oriented tasks, this partnership requires a bi-directional understanding between AI and human capabilities without which the team performance may not exceed the individual performance [5]. This is mostly referred to as shared mental model [6] in which AI and human create an understanding of their teammate ability to employ their obtained knowledge in perceiving and explaining an external stimulus and predicting the consequence of any response to it [7]. Explainable AI is extensively being explored to create a mental model of AI and enhance AI's transparency. Although it has been shown, AI accuracy does not necessarily translate to team performance [5, 8]. Most AI techniques are still developed in isolation from a human model or by considering a human as a source of uncertainty. In this regard, human performance has been long studied with regard to cognitive factors (including the workload, engagement, distraction, situation awareness, and trust) [9] in the ergonomic research that can enhance AI's understanding of the human counterpart. Understanding how these factors change with the AI assistance/decision making is imperative to create a synchronous Human-AI collaboration environment [10].

The main challenges in utilizing cognitive factors to describe the human mental model is the need to conduct human subject studies for specific tasks as most cognitive factors are task-dependent. Moreover, cognitive factors are hidden factors that are either utilized by self-reported surveys (e.g., NASA TLX) or by the need to be estimated from physiological information [11]. Finally, most human subject experiments are conducted in simulation environments where the effect of simplified simulations on the human mental model is unknown. To study these limitations, in this paper we use real-time brain activity monitor-

*Corresponding author: ehsanesf@buffalo.edu

ing techniques through an electroencephalogram (EEG) and eye movement to predict the human's cognitive factors and then study the effect of simulation simplifications on these cognitive factors by comparing the extracted mental model with real hardware experiments. To minimize the other effects in our study, we select the case of teleoperation where a human operator interacts with multiple agents to achieve a shared goal. By choosing this scenario, human will interact with the same interface regardless of how AI agents are represented (real robots or simulated robots).

2. BACKGROUND

Measuring teleoperators physiological data while performing human-swarm interaction tasks provides information about cognitive states of the operator and the opportunity to assess metrics such as cognitive load, situational awareness, and trust, to design a better human-swarm teaming [12] and predict human performance, task difficulty [13], and operator tactical decision (intention) [1] in real-time. In previous research, we have found that looking at the physiological data during HSI to complete time-intensive tasks provides information about the difficulty of the mission and the tactics the human is using [13], [1]. By utilizing the human physiological information and hardware calibration for simulation environments, we can explore how humans react to the difference between physical and simulated environments.

There are currently many simulation environments that are used for human-swarm interaction. Most of these simulated environments employ physics engines to simulate their swarms and environments [14], [15] but do not compensate for other hardware difficulties such as compliance, noise, or problems sending signals.

There have been massive strides in simulation environments for various human swarm interaction applications. For instance, Kate et al. [16] has developed Simbee-otic which is a simulation framework for modeling and simulating micro aerial vehicles. Pinciroli et al. [17] has developed a similar robot simulator called ARGos to simulate large swarm robotics of many different types. Soria et al. [18] presents a novel Matlab simulator called SwarmLab that looks to standardize the processes and metrics of the robustness of swarm algorithms. Cheraghi et al. [19] have created an open-source simulator where they can evaluate large scale swarms in both 2D and 3D. All these simulators lack the ability to synchronize human swarm interaction while measuring the user's physiological information.

Previously we developed a new simulation framework and simulator for HSI named SHaSTA. This framework outperforms other simulation platforms by offering physiological data measuring, a comprehensive GUI for teleoperation, a light physics engine, and plugins for reinforcement learning. This simulator also provides a lab streaming layer that saves all game and human data for post-process analysis. This open-source simulator has been used in multiple human subject studies and swarm analysis including learning swarm tactics over complex adversarial environments [20].

Similar to the simulation environment, creating a physical environment is also important in human-swarm interaction experiments. Many indoor testbeds have been developed to test

human robot interaction [21]. While measuring physiological data from a subject, the resulting value depends on how the operation of the physical environment connects to the simulator. Our group uses e-puck2, a small ground robot to reflect the physical environment and conduct the physical testbed in a motion capture experiment using a VICON Tracker system for accurate movement of the swarm robots. By showing the positioning of each e-puck2 on the simulator screen, it is possible to check for malfunctions, delays or latency in the physical environment and provide feedback.

In this paper, we emphasize the importance of this end-to-end framework to properly analyze the human cognitive model during human swarm interaction teleoperation. We not only present a human subject study and analysis to illustrate three main goals, but we also discuss how this framework could be used by others with minimal effort to push the research community forward. Our three main goals in this paper are: (1) We enhance our current simulation platform SHaSTA by creating add-ons that allow for compliance models, feedback, hardware connection, and swarm calibration. (2) We present a new framework for conducting a human subject study where the subjects must interact with a GUI to control a simulation, real hardware, or simulated hardware to complete a disaster search mission. We also include all our design of experiment and calibration between the physical hardware, simulated hardware, and simulation. We also explicitly share how this end-to-end framework may be modified within a matter of hours to days for studying HSI in different tasks, settings, and environments (which would have previously required months or years of effort to set up). (3) Lastly, we analyze the data collected to show what effect the simulated hardware versus the real hardware versus the simulation have on the teleoperators cognitive model. We validate our framework by exploring the human cognitive model changes between simulation, simulated hardware, and real hardware.

3. METHODS AND MATERIALS

To study the teleoperator behavior and human physiological data during three different environments: swarm simulation, hardware simulation, and pure hardware, we created a framework to run a human subject study. Utilizing our previously developed SHaSTA environment we were able to create add-on packages for compliance, feedback, hardware, and simulated hardware. Using these add-ons, we conducted a human-subject study that required a teleoperator to perform a search mission in an urban environment with supervision and control of swarms both through simulation and a physical testbed. The details of the simulation environment, physical environment, calibration, physiological data recording, and analysis are reported below.

3.1 Simulation Framework

In any human-subject study, it is imperative to have an effective interaction platform that promotes engagement but does not overload the human workload. In recent years, game engines have progressed in photo-realistic rendering and physics which has attracted users from diverse fields, such as robotics, to develop various different application environments. We developed

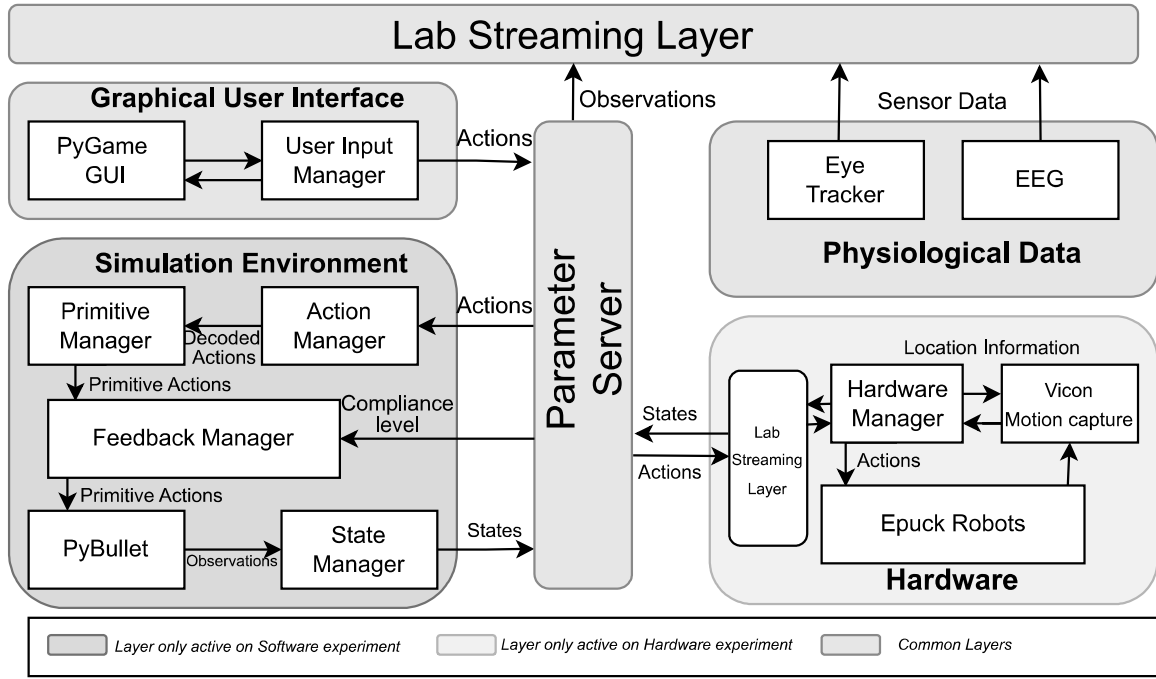


FIGURE 1: THE FRAMEWORK CONSISTS OF A SIMULATION ENVIRONMENT, A GRAPHICAL USER INTERFACE, A HARDWARE PLUGIN MODULE, AND PHYSIOLOGICAL DATA MEASURING. ALL DATA IS STORED IN THE LAB STREAMING LAYER AND PROVIDED A TIME STAMP FOR POST-PROCESSING

a game-based GUI while incorporating physics using an open-source library PyBullet. To allow for communication between the hardware and the simulation, a parameter server is utilized for all data exchange. All modules are run in parallel using an open-source library called Ray. Further details are explained below:

3.1.1 Parameter server. The parameter server is developed using an open-source package for distributed computing named Ray. The parameter server initializes and manages the information exchange from PyGame (GUI) and PyBullet (Simulation). This module tracks all information such as the position of robots, user input, and game status. It is also responsible for transmitting this information to the lab streaming layer for data synchronization and time stamps.

3.1.2 Graphical User Interface. The main GUI displayed to the teleoperator is designed using Pygame as shown in Figure 2. The GUI contains a 2D representation of the environment, feedback, and the mission status grid. The GUI takes the swarm location from the parameter server and displays the location on the 2D map to the user. The user must assign target locations to the platoons by pausing the game, drawing a rectangle around the wanted platoon, and selecting a target location. The target position along with the allocated platoon information will be sent to the parameter server, which will be used by the bullet engine to execute the tasks.

The feedback grid on the right side of the screen provides the user with notifications whenever the platoon fails to follow the user's instructions. More details on compliance and feedback are explained in section 3.5. As the name suggests, the score grid and time grid display the score and the mission time.

3.1.3 Simulation Environment. The simulation environment controls task allocation, path planning, and managing compliance. The action manager controls task information for each swarm that is received from the parameter server. It then assigns the swarms different primitives using a primitive manager. The feedback manager evaluates the platoon's compliance and feedback level to determine if given action should be performed. The Pybullet engine performs the user's actions and sends the platoon's data to the parameter server through the state manager.

3.2 Physical Environment

E-puck2, a wheeled mobile robot from GCTronic, was employed to play the physical role of swarm robots in the physical testbed. E-puck2s have a built-in Bluetooth module allowing communication with multiple E-pucks from a single PC. In this experiment, we utilized three E-puck2 robots as different swarms to enhance the communication speed between the robots and controller in the physical testbed.

In order to conduct the physical experiment with E-puck2s, the position data sent to the simulation must be precise. To achieve this, we needed to create a mapping from the physical testbed to the simulation. We used Vicon Tracker, a software used for motion capture labs, to allow for accurate tracking of the swarm robots. By utilizing Vicon Tracker data-streaming, we were able to receive information on motion-captured objects at the maximum speed of 300 Hz, which significantly reduced the delay and layover between the robots and teleoperator.

Lastly, as shown in Figure 1, in order to harmonize the environment of the simulator and the physical experiment, we set up a projector vertically on the floor to project the map which

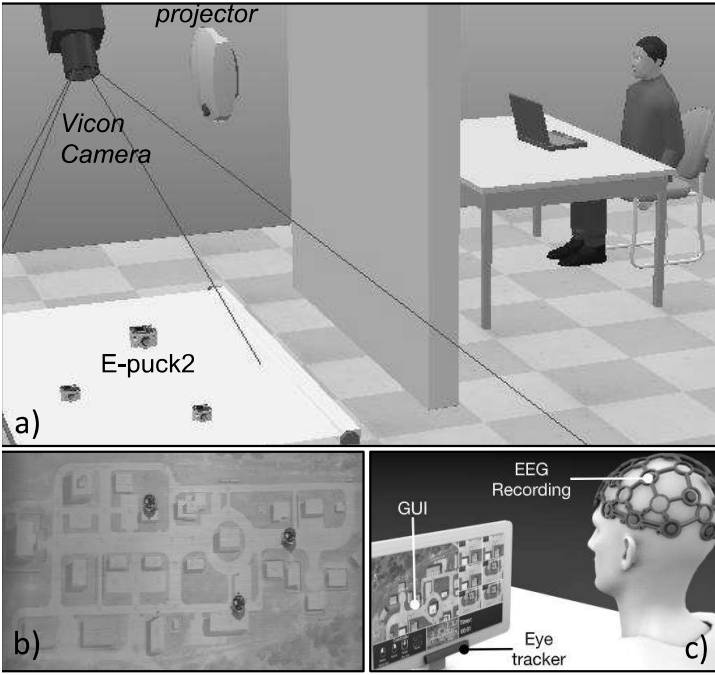


FIGURE 2: A) THE EXPERIMENTAL SET UP USED FOR RECORDING DATA. THE HARDWARE IS BLOCKED FROM THE HUMAN SIGHT BUT TRACKED TROUGH A VICON SYSTEM AND DISPLAYED THROUGH THE GUI. THE HUMAN CONTROLS THE SWARMS THROUGH THE GUI. B) THE PHYSICAL HARDWARE BEING CONTROLLED BY THE HUMAN ON THE PROJECTED 2D MAP. C) THE PHYSIOLOGICAL DATA RECORDING SETUP.

was displayed in the simulator. Calibration between the map projected on the floor and the map displayed on the simulator screen required adjustments in the different coordinate systems. The way to calibrate the experiment is explained in the next section.

3.3 Experiment calibration

The physical experiments returned the exact position of each swarm robot to the simulator in real-time. When a subject assigned a platoon's target position through the simulation, the location was sent to the hardware. The hardware moved to its target location and the swarm's movement was displayed to the user in real-time. In order to effectively control and display the physical experiments through the simulator, it was necessary to calibrate the map projected on the floor with the map in the simulator. Mapping calibration was executed by extracting the four corner points of both the hardware and simulation environments and performing affine transformation with translation and scaling. This transformation is shown below:

$$\begin{aligned} x_{phy} &= (y_{sim} + 80) \times w_{phy} / w_{sim} \\ y_{phy} &= (-x_{sim} + 80) \times h_{phy} / h_{sim} \end{aligned} \quad (1)$$

where x_{phy} and y_{phy} stand for the coordinates in physical environment and x_{sim} and y_{sim} stand for the coordinate in simulation environment. w_{phy} and h_{phy} are the width and height of the physically projected map, and w_{sim} and h_{sim} are the width and height of the map displayed on the simulator.

3.4 Physiological Data Collection

In this study, we measure the participant's brain activity. For the brain activity, we use the non-invasive B-Alert X24 electroencephalogram headset from Advanced Brain Monitoring[®] to record brain activity of the subject at 20 locations on the scalp with a sampling rate of 256.0 Hz.

We utilize the EEG to extract mental workload, situational awareness, engagement, and distraction. A quadratic discriminant function analysis used by the B-Alert software is used to classify the engagement level. The coefficients of this model are individualized for each participant according to the baseline tasks [22]. The EEG signals must be cleaned to remove any electrical noise, slow drift, or muscle movements from the electrodes. To do this, we begin by filtering the EEG with a bandpass filter (1-60 Hz). We then run an Independent Component Analysis to remove all muscle movement and eye blink artifacts. Autorejection [23] was used to remove any epochs from the data that contained very large artifacts that skewed the data. After each experiment the participant completed a NASA-TLX survey rating their perceived mental, physical, and temporal demands as well as their effort, performance, and frustration during the experiment.

3.5 Feedback and Compliance

In this particular human subject study, we focused on two main applications utilizing this framework. The first study was looking at different levels of feedback and compliance, and the second study was creating a simulated hardware that replicated the real hardware testbed. To study feedback and compliance, we ran nine different iterations of a target search mission in an urban environment. The nine different iterations consisted of a combination of three levels of feedback and three levels of compliance.

The three levels of feedback (low, medium, high) all progressively provided more information to the user about the platoons. The compliance model in the simulation was done by randomly denying the users action 50, 70, and 90 percent of the time.

The simulated hardware was based on the real hardware testbed experiments. The drone speed was reduced to one third of the simulation speed and 30 pixels of noise was added to target position of each drone. The drone also had a 10 percent chance of either not complying or going to the wrong location.

3.6 Experiment and Framework

The framework for this study is illustrated in detail in Figure 1. The end-to-end framework includes a simulation environment, a hardware plugin, a simulated hardware plugin, an interactive GUI, physiological data measuring, and a real-time physics engine. The framework in this paper was utilized by performing a search and extract mission while providing the different levels of feedback and compliance. It was then utilized to do the same mission in the physical testbed and then in the simulated hardware testbed. The goal of the user was to search an urban environment to find as many targets as possible in one minute.

To test the simulated versus real hardware, all participants were informed they were controlling the real hardware. Half the participants were controlling the E-puck2s and the other half were working with the simulated hardware. The experimental setup

denoted in Figure 2 consists of a human controlling the swarms through a computer setup while getting their physiological data measured. The human cannot see the hardware testbed that is being measured through the Vicon Tracker system.

4. RESULTS AND DISCUSSION

4.1 Simulation, Hardware, Simulated Hardware

In Figure 3 and Table 1 we show the efficiency of our end-to-end framework and demonstrate some of the capabilities in this study. In Figure 3 we plot the x and y positions of one UAV for the simulation, hardware, and simulated hardware. The user assigns target positions through the GUI, and the simulator and hardware must both try to reach that position. In the top plot, we see the simulated hardware verse the simulator, and in the bottom we see the physical hardware vs the simulator.

In supplement to Figure 3, in Table 1 we report the time the user inputted a target location, the simulated UAV arrival time, and the physical/simulation hardware arrival time. We also show the average travel time of the simulation and simulated/physical hardware. It can be seen that the simulated hardware is a just representation of the physical hardware delays and travel speed. In the physical testbed, the hardware takes an average of 4.9 seconds longer and the simulation hardware takes an average of 3.67 seconds longer than the simulation to reach the target position. It should be noted that the physical hardware takes about 5 seconds to orient itself at the beginning of the experiment. In the simulated hardware we show a case where the noise of position affects the hardware.

Target	Input Time	Arrival Time (Simulation)	Arrival Time (Alternative)	Alternative
1	11.3	14.6 (3.3 sec)	26.8 (15.5 sec)	Physical Hardware
2	24.9	30.2 (5.3 sec)	33.0 (8.1 sec)	
3	52.4	57.2 (4.8 sec)	58.8 (6.4 sec)	
4	60.5	65.0 (4.5 sec)	68.0 (7.5 sec)	
Average Travel Time		4.475 sec	9.375 sec	
1	5.9	14.7 (8.8 sec)	15.4 (9.5 sec)	Simulated Hardware
2	19.4	34.1 (14.7 sec)	36.9 (17.5 sec)	
3	40.4	43.7 (3.3 sec)	49.9 (9.5 sec)	
4	53.7	DNF	DNF	
Average Travel Time		8.93 sec	12.16 sec	

TABLE 1: COMPARISON OF THE SIMULATION UAV WITH THE HARDWARE AND SIMULATION HARDWARE. THIS TABLE SUPPLEMENTS FIGURE 3 AND SHOWS THE TIME THE USER INPUTTED A TARGET LOCATION, THE TIME THE SIMULATION UAV REACHES A TARGET, AND THE TIME THE SIMULATED/PHYSICAL HARDWARE REACHES A TARGET. WE ALSO SHOW THE AVERAGE TRAVEL TIMES TO COMPARE THE SIMULATED HARDWARE AND THE PHYSICAL HARDWARE DELAYS.

4.2 Teleoperator Cognitive Analysis

Focusing on analyzing the human decisions and performance, we look at Figure 4. This plot shows the variance of the Z-score values of the NASA-TLX survey. These results give insight on the framework, design of experiments, and the teleoperators cognitive load. The NASA-TLX survey has five questions

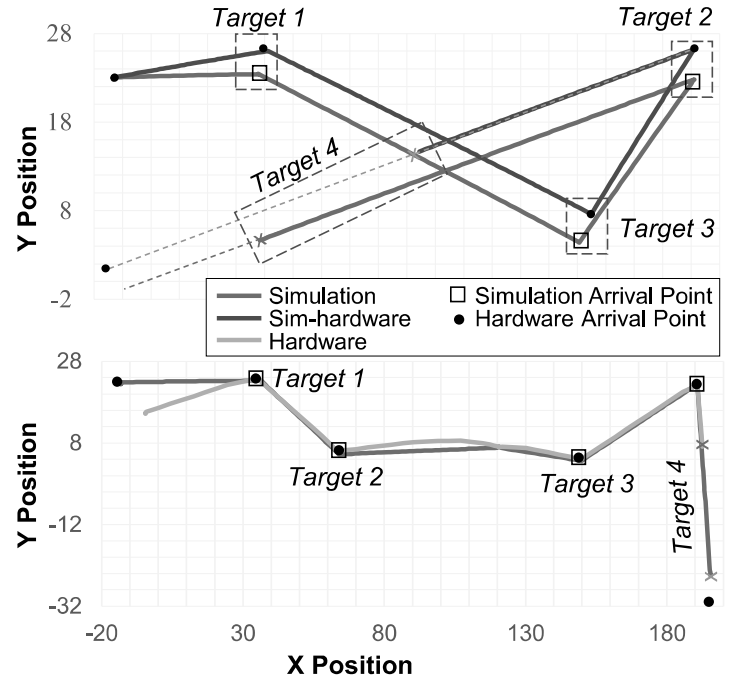


FIGURE 3: COMPARISON OF THE SIMULATION WITH HARDWARE AND SIMULATION HARDWARE. THE TOP PLOT SHOWS A SINGLE UAV POSITION FOR BOTH SIMULATION AND SIMULATED HARDWARE. THE BOTTOM PLOT SHOWS A SINGLE UAV POSITION FOR BOTH SIMULATION AND PHYSICAL HARDWARE. THE TARGETS ARE ASSIGNED BY THE USER AND THE ARRIVAL OF THE UAVS IS DENOTED BY A CIRCLE OR SQUARE.

to measure the human cognitive state during the experiment. Due to individual differences in subjects, it is not sufficient to look at the averages, so the Z-score is calculated to get rid of subject bias.

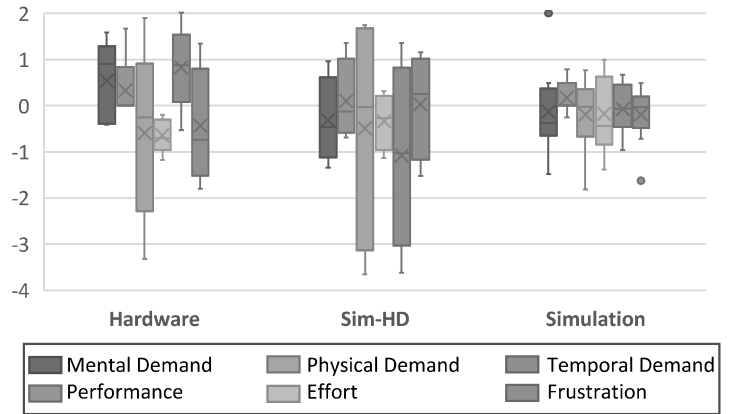


FIGURE 4: THIS PLOT SHOWS THE Z-SCORE VARIANCE OF THE NASA-TLX SURVEY RESULTS FOR THE PHYSICAL HARDWARE, SIMULATED HARDWARE, AND SIMULATION.

The NASA-TLX survey is the human perception of their own mental workload. Our goal was to show that our framework is sufficient for physical hardware simulation, hardware simulation, and simulation. One important factor that we illustrate is that the simulated hardware is more like the physical hardware than the simulation. We can observe that in Figure 4 there is little

variance between any cognitive measures. We observe the human perception cognitive measures during simulated hardware are much closer to the physical hardware than the simulation. This proves that our framework is successful at simulating hardware to a degree that replicates physical hardware on the cognitive mind.

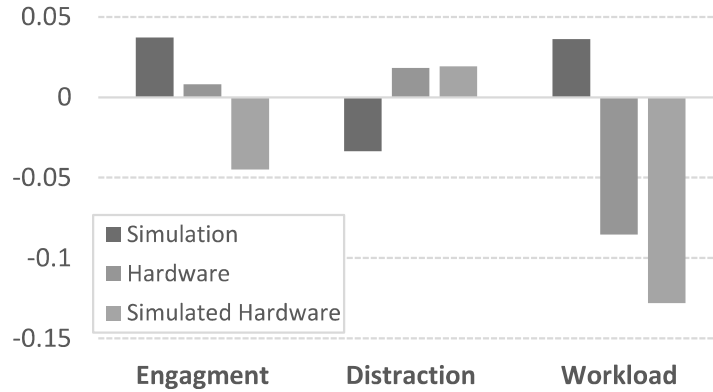


FIGURE 5: THIS PLOT SHOWS THE Z-SCORE AVERAGE COGNITIVE MEASURES FROM THE BRAIN ACTIVITY FOR THE SIMULATION, HARDWARE, AND SIMULATED HARDWARE.

Looking at Figure 5 we compare the teleoperators cognitive functions for the the simulation, the hardware, and the simulated hardware. To show that our end-to-end framework works for simulation, hardware, and physical hardware, we illustrate the teleoperator cognitive function of workload and distraction are very similar for hardware and simulated hardware, but not for simulation. The engagement and distraction difference between the three cases is very insignificant, but the workload shows substantial differences. This is expected due to the simulation having non-compliance and requiring the user to be more engaged and using more cognitive work with the feedback. For example, during the simulation, the swarms may be non-compliant, requiring the user to evaluate the feedback and re-evaluate to another platoon, but in the hardware studies, the swarms are always compliant. This will cause a higher workload and engagement in the simulation.

5. CONCLUSION

This end-to-end framework is incredibly important for the research community of human swarm interaction. We have gone through the difficulties of building this framework from the ground up so others in the community can study human swarm interaction with ease. This framework can trim months of future research opportunities in this community. The wide range of applications and ease to change the parameters in this HSI framework is a huge contribution.

In this paper specifically, we utilized this framework to explore how teleoperators react to different levels of feedback and compliance, hardware, and simulated hardware. Now that this framework is developed, the next step in this research is to do an in-depth analysis of the human physiological information to create a cognitive model that can be used for Human-Swarm Interaction and decision making.

ACKNOWLEDGMENT

This work was supported by the NSF awards CMMI-2048020 and IS-1927462. Any opinions, findings, conclusions, or recommendations expressed in this paper are those of the authors and do not necessarily reflect the views of the NSF.

REFERENCES

- [1] Hemanth Manjunatha et al. "Using Physiological Measurements to Analyze the Tactical Decisions in Human Swarm Teams". In: *2020 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*. IEEE. 2020, pp. 256–261.
- [2] Kevin Dzobo et al. "Integrating artificial and human intelligence: a partnership for responsible innovation in biomedical engineering and medicine". In: *Omics: a journal of integrative biology* 24.5 (2020), pp. 247–263.
- [3] Min Chen et al. "Trust-aware decision making for human-robot collaboration: Model learning and planning". In: *ACM Transactions on Human-Robot Interaction (THRI)* 9.2 (2020), pp. 1–23.
- [4] Sarvapali D Ramchurn, Sebastian Stein, and Nicholas R Jennings. "Trustworthy human-AI partnerships". In: *Iscience* 24.8 (2021), p. 102891.
- [5] Ming Yin, Jennifer Wortman Vaughan, and Hanna Wallach. "Understanding the effect of accuracy on trust in machine learning models". In: *Proceedings of the 2019 chi conference on human factors in computing systems*. 2019, pp. 1–12.
- [6] Robert W Andrews et al. "The role of shared mental models in human-AI teams: a theoretical review". In: *Theoretical Issues in Ergonomics Science* 24.2 (2023), pp. 129–175.
- [7] Janis A Cannon-Bowers, Eduardo Salas, and Sharolyn Converse. "Shared mental models in expert team decision making." In: (1993).
- [8] Gagan Bansal et al. "Beyond accuracy: The role of mental models in human-AI team performance". In: *Proceedings of the AAAI conference on human computation and crowdsourcing*. Vol. 7. 2019, pp. 2–11.
- [9] Sarah Hopko, Jingkun Wang, and Ranjana Mehta. "Human factors considerations and metrics in shared space human-robot collaboration: a systematic review". In: *Frontiers in Robotics and AI* 9 (2022), p. 6.
- [10] Jinglu Jiang et al. "A situation awareness perspective on human-AI interaction: Tensions and opportunities". In: *International Journal of Human-Computer Interaction* (2022), pp. 1–18.
- [11] Amirhossein H Memar and Ehsan T Esfahani. "Physiological measures for human performance analysis in human-robot teamwork: Case of tele-exploration". In: *IEEE access* 6 (2018), pp. 3694–3705.
- [12] Aya Hussein et al. "Characterization of indicators for adaptive human-swarm teaming". In: *Frontiers in Robotics and AI* 9 (2022), p. 16.

- [13] Joseph P Distefano et al. "Using physiological information to classify task difficulty in human-swarm interaction". In: *2021 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*. IEEE. 2021, pp. 1198–1203.
- [14] Neel Dhanaraj et al. "Adaptable platform for interactive swarm robotics (apis): a human-swarm interaction research testbed". In: *2019 19th International Conference on Advanced Robotics (ICAR)*. IEEE. 2019, pp. 720–726.
- [15] Shital Shah et al. "Airsim: High-fidelity visual and physical simulation for autonomous vehicles". In: *Field and Service Robotics: Results of the 11th International Conference*. Springer. 2018, pp. 621–635.
- [16] Bryan Kate et al. "Simbeeotic: a simulator and testbed for micro-aerial vehicle swarm experiments". In: *2012 ACM/IEEE 11th International Conference on Information Processing in Sensor Networks (IPSN)*. IEEE. IEEE, 2012, pp. 49–60.
- [17] Carlo Pinciroli et al. "ARGoS: a modular, parallel, multi-engine simulator for multi-robot systems". In: *Swarm intelligence* 6.4 (2012), pp. 271–295.
- [18] Enrica Soria, Fabrizio Schiano, and Dario Floreano. "SwarmLab: A MATLAB drone swarm simulator". In: *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE. IEEE, 2020, pp. 8005–8011.
- [19] Ahmad Reza Cheraghi et al. "Swarm-sim: A 2d & 3d simulation core for swarm agents". In: *2020 3rd International Conference on Intelligent Robotic and Control Engineering (IRCE)*. IEEE. IEEE, 2020, pp. 1–10.
- [20] Amir Behjat et al. "Learning robot swarm tactics over complex adversarial environments". In: *2021 international symposium on multi-robot and multi-agent systems (MRS)*. IEEE. 2021, pp. 83–91.
- [21] Mario Valenti et al. "Indoor multi-vehicle flight testbed for fault detection, isolation, and recovery". In: *AIAA guidance, navigation, and control conference and exhibit*. 2006, p. 6200.
- [22] Chris Berka et al. "Real-time analysis of EEG indexes of alertness, cognition, and memory acquired with a wireless EEG headset". In: *International Journal of Human-Computer Interaction* 17.2 (2004), pp. 151–170.
- [23] Mainak Jas et al. "Autoreject: Automated artifact rejection for MEG and EEG data". In: *NeuroImage* 159 (2017), pp. 417–429.