

REPORT DOCUMENTATION PAGE				Form Approved OMB No. 0704-0188	
<small>Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden to Washington Headquarters Service, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188) Washington, DC 20503.</small>					
PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ADDRESS.					
1. REPORT DATE (DD-MM-YYYY) 23 Feb 2024		2. REPORT TYPE Final Report		3. DATES COVERED (From - To) 15 Aug 2023 through 15 March 2024	
4. TITLE AND SUBTITLE Final Technical Report - AI-Enabled Rapid Operational Decision Making (AEROD)				5a. CONTRACT NUMBER H9 240523 P00 11	
				5b. GRANT NUMBER	
				5c. PROGRAM ELEMENT NUMBER	
6. AUTHOR(S) Dr. Ross Hoehn (X R 2 Lead)				5d. PROJECT NUMBER	
				5e. TASK NUMBER	
				5f. WORK UNIT NUMBER	
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) X R 2 Lead LLC 15795 LAZYDAY L N DUMFRIES VA 22025-1823				8. PERFORMING ORGANIZATION REPORT NUMBER Final Technical Report - Deliverable 9	
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) Civilian Institutions Office (Code 522) Naval Postgraduate School 1 University Circle, Herrmann Hall Rm HE046 Monterey, CA 93943-5033				10. SPONSOR/MONITOR'S ACRONYM(S) NPS CIVINS	
				11. SPONSORING/MONITORING AGENCY REPORT NUMBER CDRL A003	
12. DISTRIBUTION AVAILABILITY STATEMENT Approved for public release; distribution is unlimited					
13. SUPPLEMENTARY NOTES					
14. ABSTRACT The XR 2 LEAD Team – including the Georgia Institute of Technology (GTRI) – prototyped the AEROD solution, an AI-enabled agent human-machine team with the aviation crews providing alerts and facilitating decisions in response to key mission environment changes. Specifically, an AI-enabled agent that can accept, structure, fuse, and analyze multiple incoming data streams to act as the decision-point for the selection of an alternate aircraft route and altitude in response to changes, allowing the building of trust using AI in a mission environment, while improving the timeliness and quality of decision making and reducing crew workload . This agent will digest incoming data streams and couple this with path optimization algorithms and measures of human trust.					
15. SUBJECT TERMS Artificial Intelligence, Machine Learning, Human machine interface, Decision Making, Planning, rotary wing, pilot assistance					
16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT UU	18. NUMBER OF PAGES 28	19a. NAME OF RESPONSIBLE PERSON Lev Mazeres
a. REPORT U	b. ABSTRACT U	c. THIS PAGE U			19b. TELEPHONE NUMBER (Include area code) 757-878-0119

AI-Enabled Rapid Operational Decision Making (AEROD)

Technical Report of Initial Study

1 Background

Modern aviators are inundated with an overwhelming deluge of unstructured data from disparate sources, often imposing substantial cognitive burdens throughout different phases of flight operations. As mission conditions frequently deviate from designated plans, the exigency and quality of decisions can significantly influence both flight safety and mission success. Over decades, the convergence of national and mission-driven imperatives has led to the development of a wide range of capabilities; each of these capabilities is tailored to meet specific sensing requirements across different segments of the electromagnetic spectrum and other situational awareness domains. The independent development of these technologies by multiple programs and various original equipment manufacturers (OEMs) has resulted in a fragmented information landscape. This fragmentation has led to crew procedures and processes requiring the individualized monitoring of a growing number of real-time data sources to inform a crew's operational decision-making. This leads to an increased cognitive burden for crews – in terms of both volume of information and fusion of this incoming information – and results in less effective decision-making, especially in high-tempo/high-stress mission contexts.

Although many of the technologies and techniques in use by modern Special Operating Forces (SOF) crews were initially developed through the independent progression of multiple original equipment manufacturers (OEMs), this Phase 1 Feasibility Study sought to explore the practical application of recent advances in Artificial Intelligence and Machine Learning to increase SOF operational capabilities while reducing platform crew cognitive burden through information fusion using Machine Learning and reducing cognitive load through Artificial Intelligence *that collaborates with human crew members*. The premise is that using AI and ML in a human collaborative decision-making system will 1) reduce cognitive burden, 2) enhance human performance, and 3) enable emergent capabilities/solutions not possible with either human or AI alone. This report documents the *concepts explored*, the *methods of exploration*, and the *discoveries* made in this Phase I pursuit. The findings of this study indicate a strong potential for seamless integration of disparate information sources, thereby mitigating the previously observed fragmentation and cognitive burden. As a result, modern aviators stand to benefit from a more streamlined flow of structured data and recommendations, which could significantly alleviate cognitive burdens throughout different phases of flight operations. This study also underscored the critical importance of timely and high-quality decision support in response to dynamic mission environments and suggests a realizable capability enabling high-confidence operational decision-making that exceeds the capabilities of either human or AI alone.

1.1 Approach

Within expansive or multifaceted data environments, several strategies exist to alleviate human cognitive strain and information overload. Techniques such as *data fusion* serve to condense incoming data streams, while *machine learning* algorithms excel at discerning patterns and classifications. Tailored *dashboards* further refine analysis outputs, directing user attention to critical insights, and Natural Language Processing (NLP) provides an intuitive and adaptive means to achieve human-machine collaboration. The integration of such tools stands to revolutionize user

interaction with data, culminating in a unified source of human-machine interaction (HMI) capable of *navigating diverse data streams and facilitating informed decision-making*. XR2Lead, in collaboration with the Georgia Institute of Technology (GTRI), have developed the AEROD solution: an AI-enabled information processing framework that collaborates with aviation crews to find correlations across an array of information sets, deliver alerts of anomalies which may only be discernable across multiple disparate data sources, and inform timely decisions in response to dynamic mission environments. In this Phase 1 study, the designed AEROD framework demonstrates the potential to ingest, structure, fuse, and analyze multiple data streams to provide route guidance. It strongly suggested the ability to provide real-time guidance for selecting alternate aircraft routes and altitudes, while fostering trust in AI through continuous availability and interaction with the crew. This interaction enhances the timeliness and quality of decision-making, ultimately reducing crew workload.

2 Discussion of Study Methodology.

The AEROD framework resulted from investigations and experiments across five key objective areas:

- 1) Requirements Setting and Defining Adequate Surrogate Data Identification,
- 2) Modeling Sensor/Aircraft Data AI,
- 3) Assessing Human Systems Integration/Interaction Principles Required,
- 4) Modeling AI Architecture/Compute Environment,
- 5) Designing with Performance and Implementation Requirements.

The sections below describe and discuss this framework.

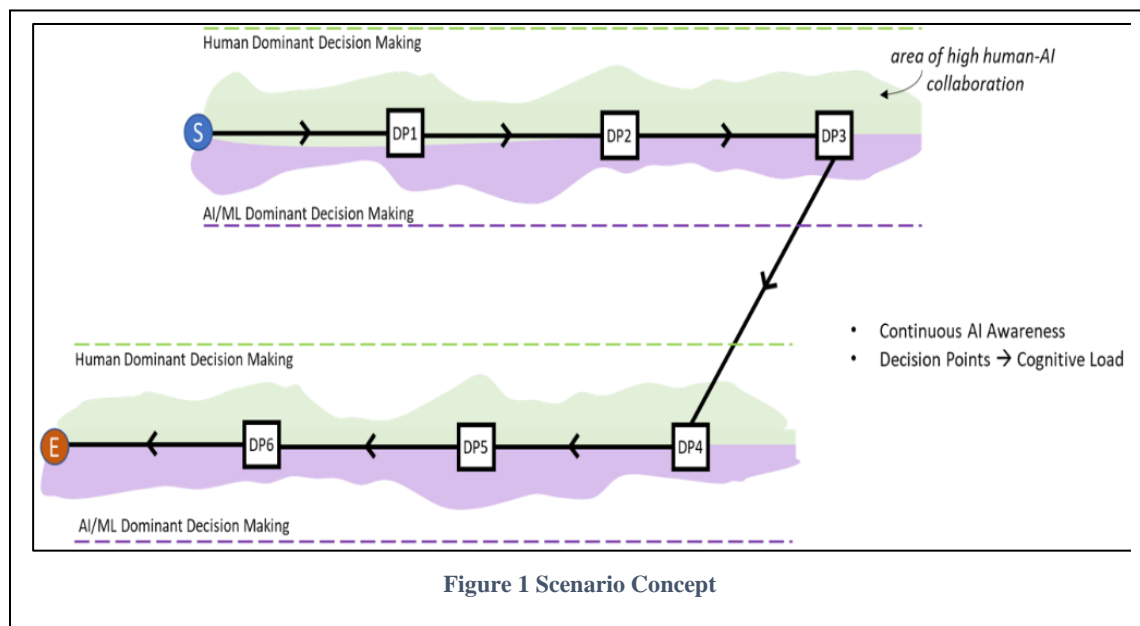
2.1 Requirements Setting and Defining Adequate Surrogate Data Identification

Requirement Setting: Our initial objective was to develop a comprehensive understanding of the project's technical requirements by examining Use Cases and AI Requirements via an abstract yet representative operational scenario. We began by refining our understanding of the underlying need or problem to be addressed, accomplished by the formulation of a use case and its communication via an operational scenario, or vignette. We subsequently presented this vignette, along with our hypothesis (narrative) and assumptions regarding the capabilities AI would provide, to representatives of stakeholders. The purpose was to refine and validate our proposed use case and scenario through discussion and feedback.

The primary use case involves the an AI capability capable of continuously offering recommendations for flight paths based on mission objectives and environmental risks (e.g., threats, weather conditions, and terrain) that are revealed during the mission via data collected from various onboard sensors. This particular use case serves as the focal point, or "problem," through which we analyzed and understood the overarching AI requirements, encompassing machine learning methodologies, computational infrastructure, and crew interaction dynamics.

Furthermore, this scenario served as the basis for modeling the type of data and architecture necessary for the implementation of a collaborative AI capability.

The hypothesized SOF flight scenario revolves around a single MH-60 – or another similar rotary wing platform – tasked with conducting an exfiltration mission. This mission entails launching from a ship offshore, with the objective situated on a contested island. In this context, the scenario assumes that AI support for the crew is provided as a continuous availability and accessibility to establish and maintain crew trust in the AI. Trust can be built during low cognitive load situations or mission periods. In situations of high cognitive loads – those demanding real-time decision-making – the demonstrated trust instills confidence in the crew, and thus reducing the cognitive dissonance, cognitive overload, and other stressors associated with high-stress decision-making situations. **Figure 1** depicts this scenario use case for AI at a high level of abstraction and shows where decision making is human dominant, AI dominant, or human-AI collaborative.



Within Figure 1, the profile assumes mission start, **S** through Ingress and Egress to mission end, **E**. The Decision Point (DP) boxes represent periods of dynamic inputs – threats, weather, mission change – that increase cognitive workloads due to need for decision making. Decisions – likely occurring at discrete points during a mission – require the crew to perform rapid information fusion from available data. For example, at DP 1, a change to mission timing may occur, or threat area update is received. The shaded areas of green and purple represent the level of collaboration with the AI, and “autonomy” afforded to the AI, throughout the hypothetical mission. During high decision demand segments, crew trust in the AI may allow it to become central to decision making – allowing the crew to trust recommendations while at moderate to low workload segments, the crew retains more decision making, using AI as an input (throttling the degree of reliance on the AI). The idea is: the *AI agent crew-member* is trusted to varying degrees, as a decision aid. This Human-AI teaming – in an operational flight environment – is a unique challenge.

Hypothesis an AI Architecture and Compute Environment: Since the use of actual platform and sensor

data was beyond the scope of the Phase 1 feasibility study, the hypothetical scenario was used as a basis to develop a model of what might be required of an AI system. This included such as information and data observations, AI functional components, algorithms, and crew interactions which may be required. For the *Route Recommendation* problem, the major informational elements include the kinds of external environment and aircraft platform state observations to select, a Reinforcement Learning Neural Network (RL-NN) machine learning algorithm, and a display of the AI output as might be appropriate to an overlay on existing Multi-Function Displays (MFD). For example, this could include a running or *moving map* as a means of continuously interacting with the crew. This architecture model was used in subsequent steps of our study to develop a *surrogate simulation model of the approach*, discussed in c.

It is important to understand the distinction between AI-based information fusion and classical sensor data fusion approaches. Classical data fusion approaches are typically based on Machine Learning (ML) methods. They are designed to improve the performance or accuracy of information provided by any one sensor and/or to provide a “fused” situational awareness “picture” by using data from multiple sensors. It has a long pedigree and history of success [REF]. Modern AI methods builds upon the classical data fusing approaches by combining data, ML methods, and AI techniques such as deep neural networks (DNN) to infer, i.e., *discover* and present *useful information* – or Knowledge – that is usually done exclusively by the human crew. The AEROD framework developed in Phase 1 integrates ML methods and AI methods while maintaining a Human-Over-The-Loop strategy. It includes the concept of an AI Agent that uses Reinforcement Learning to maintain decision memory (which may be forgotten by an AI alone), and validate inferences produced by the AI methods against physical rules, tactics, techniques, and procedures (TTPs). In this way, human *knowledge* is incorporated into the framework so that it can continuously refine the AI recommendations during a mission.

2.2 Modeling Sensor/Aircraft Data AI

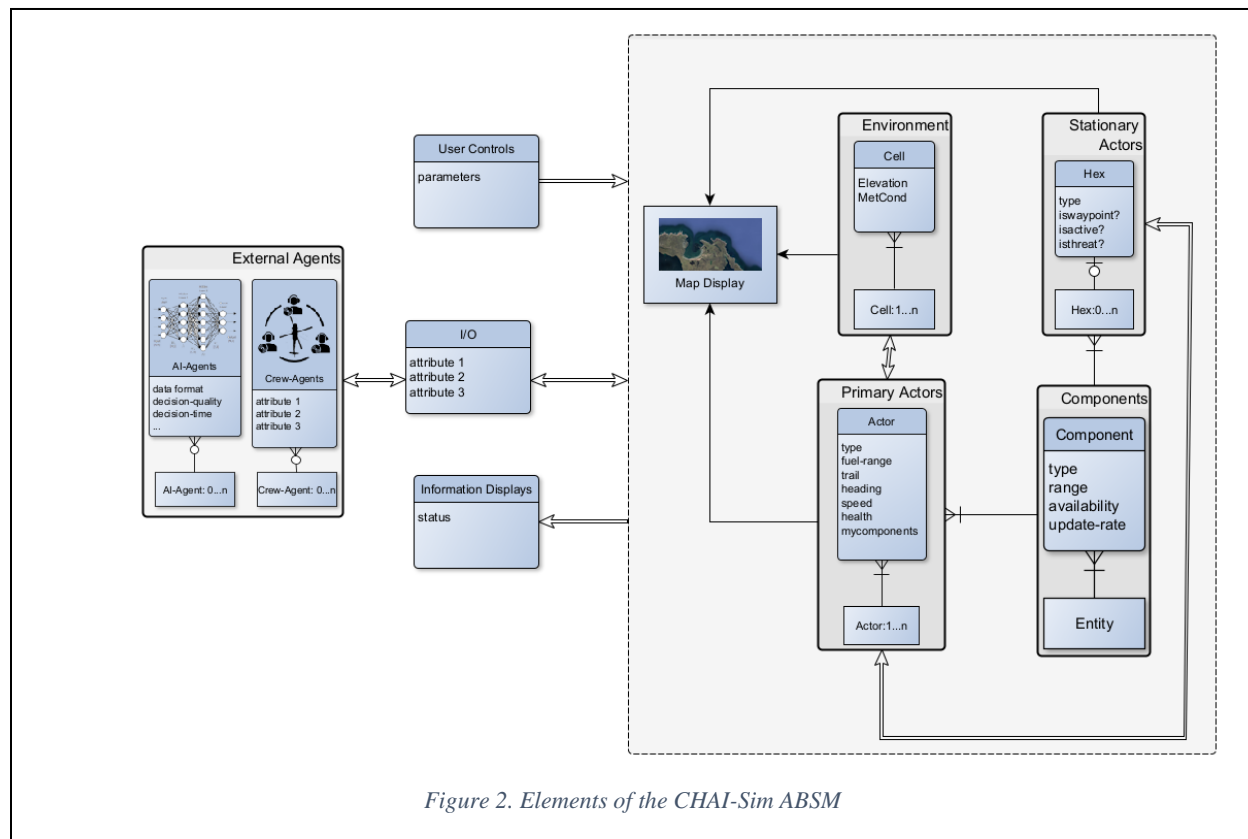
NetLogo is an opensource agent-based modeling platform that allows users to simulate complex systems by defining individual agents with behaviors and interactions, enabling the study of emergent phenomena and the exploration of various scenarios (Northwestern University, 2024). Though often used in social, biological, and environmental domains, we here opt to use Agent Based Simulation Model (ABSM) tools to develop surrogate models of scenarios and system elements prior to real-world system fabrication, integration, and testing. In the AEROD study, we use the ABSM approach to provide a means to analyze how required functions, such as Human-AI interaction (“pilot-over-the-loop”), sensor data/information use, and presentation of route recommendation predictions should be designed to provide a continuous, mission aware situational awareness capability to the pilot/crew. This benefits our overall understanding of the problem domain, including operational constraints, decomposing system elements, and exploration of interactions between multiple synchronous and asynchronous mission elements.

To depict a SOCOM MH-60 mission scenario using the described AI architecture (see Section 2.4), we used the NetLogo framework to build a simulation we called the Collaborative Human-AI Interaction in Military Operations Exploratory (CHAI-Sim) to provide surrogate data representative of the environment, mission objectives, and aircraft state.

CHAI-Sim ABSM for System Decomposition and Requirements: As mentioned previously,

CHAI-Sim is an Agent-Based Simulation Model. In ABSM, the terms "agent" and "actor" are often used interchangeably, but they can have nuanced differences based on the specific context or the preferences of the modeler or the field of study. Generally, both "agent" and "actor" refer to the entities within the simulation that exhibit independent and self-directed behavior while interacting with each other within their environment. They are typically designed with a sensorium, i.e., the ability to perceive their surroundings, they make decisions (even very basic stimulus and response), and often influence the environment based on their actions. These terms are often used in different disciplines or research areas where simulations are employed to study complex systems.

In some contexts, the term "agent" might be more commonly associated with computer science, artificial intelligence, and complex systems modeling. On the other hand, "actor" might be prevalent in fields such as social sciences, economics, or certain branches of simulation where the emphasis is on human or social behavior. CHAI-Sim is intended to provide a digital "sandbox" to explore interactions between entities in a SOF scenario. As such, the CHAI-Sim Actors are basic elements with autonomy within the simulation. We refer to External Agents as those entities outside the simulation that interact with the simulation. In the approach used for CHAI-Sim, the significant agents are the AI-Route-Recommender (AI-RR) and the Crew Model (if so determined). These are at this stage surrogate models, i.e., simplified representations that we



expect to further develop in follow-on work and use real entities when possible. For example, Crew-Agents can be simple decision-making rules in the surrogate. This allows information channels, decision timing, etc., to be explored quickly and cheaply. In subsequent phases of development, these surrogates, owing to the modular nature of the CHAI-Sim architecture, can be

readily exchanged with high-resolution models, recorded data, or even real-life crew in a training environment.

While the core concept of these terms, Actor and Agent, remains quite similar, the subtle differences exist in the way they are defined or used within CHAI-Sim. For this effort, where we are exploring what is feasible and have the need for well-understood and controlled data that can serve as a surrogate for actual data, it is preferable to refer to the more common entities in the digital sandbox as Actors, while those actors such as the AI-RR, being more sophisticated actors, we will refer to as agents. The insights to be gained from this Phase 1 exploration in CHAI-Sim are significant, providing a foundational understanding to inform subsequent phases of practical system prototyping and demonstration in Phase 2. CHAI-Sim, even its nascent state, has shown to be a valuable tool for facilitating a strategic progression from theoretical exploration to practical implementation in the developmental trajectory of AI integration into military helicopter operations. As a cheap and rapid surrogate for real-world system development, The CHAI-Sim ABSM, as a virtual sandbox, encapsulates the intricate interplay among diverse entities within the operational milieu. Figure 3 depicts the CHAI-Sim ABSM architecture in an entity-relationship format. This depicts the functional architecture of CHAI-Sim, reflected in the coding of the simulation. Here we can see the entities and their relationships, decomposed to reflect our

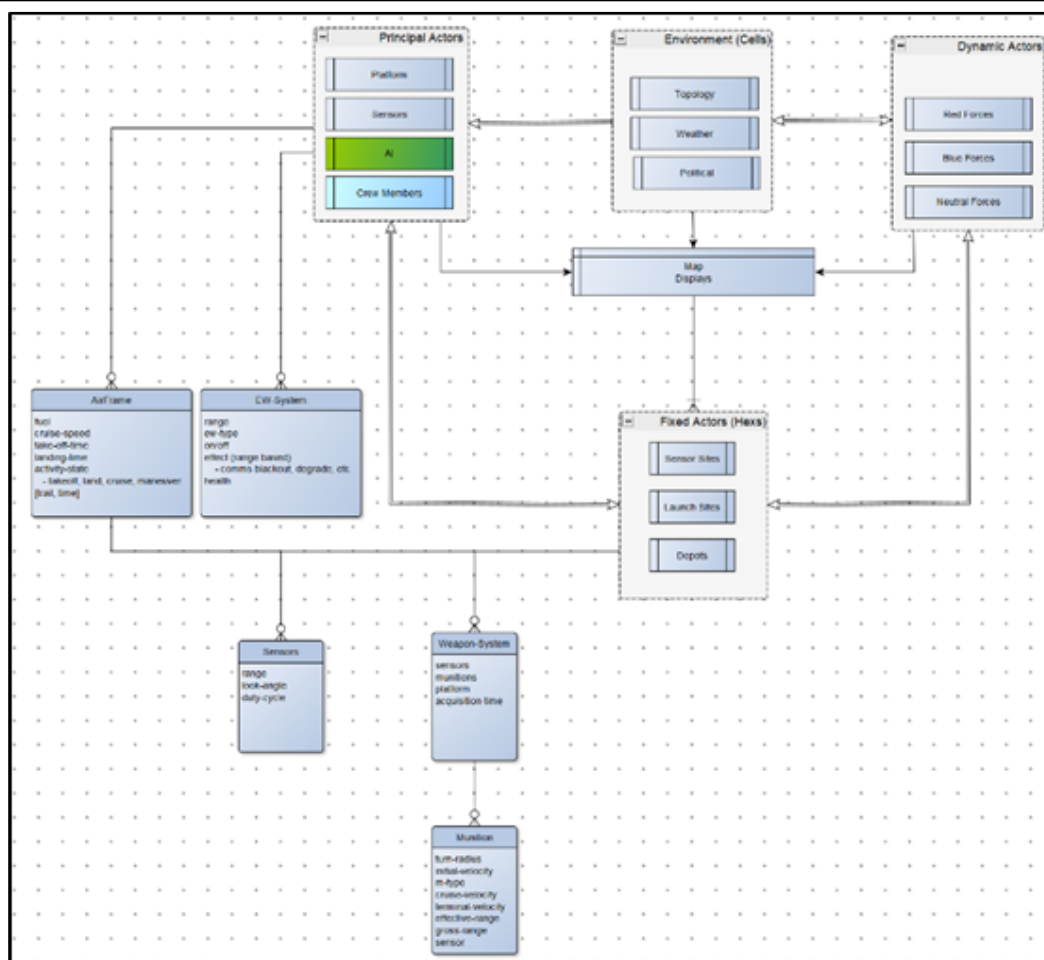


Figure 3. CHAI-Sim Entity-Relationships

understanding of the SOF mission space, including platforms, sensors, threats, and environment. As a Phase 1 study tool, we do not claim that CHAI-Sim is a complete digital twin of the SOF environment and use case, but it does allow us to explore concepts, data flows, and interactions between many of the significant elements of interest to developing a human-AI decision framework for increasing SOF capabilities while reducing cognitive load.

In the following section we present some screenshots of the CHAI-Sim representing a SOF exfiltration scenario.

Surrogate Simulation: As previously discussed, CHAI-Sim serves as a versatile platform for exploring the complexities of human-AI teaming within operational contexts. Through CHAI-Sim, this study examined notional scenarios to simulate interactions between human operators and artificial intelligence systems across various mission phases, such as operational planning, ingress and egress, threat detection, and risk assessment. Figure 4 shows CHAI-Sim initialized with a scenario which is a more practical interpretation of the abstract scenario discussed in Section **Error! Reference source not found.** (see Figure 1 Scenario Concept). We will describe the operation and results of the CHAI-Sim in greater detail in the subsequent paragraphs.

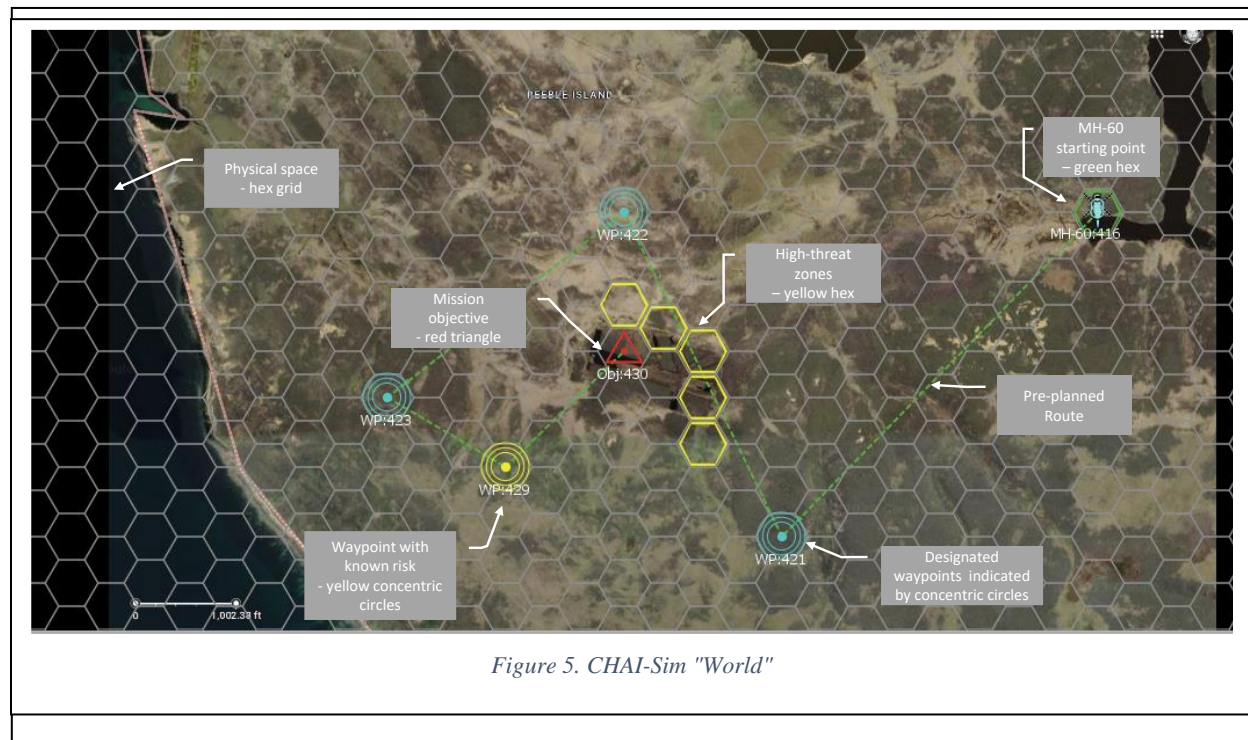
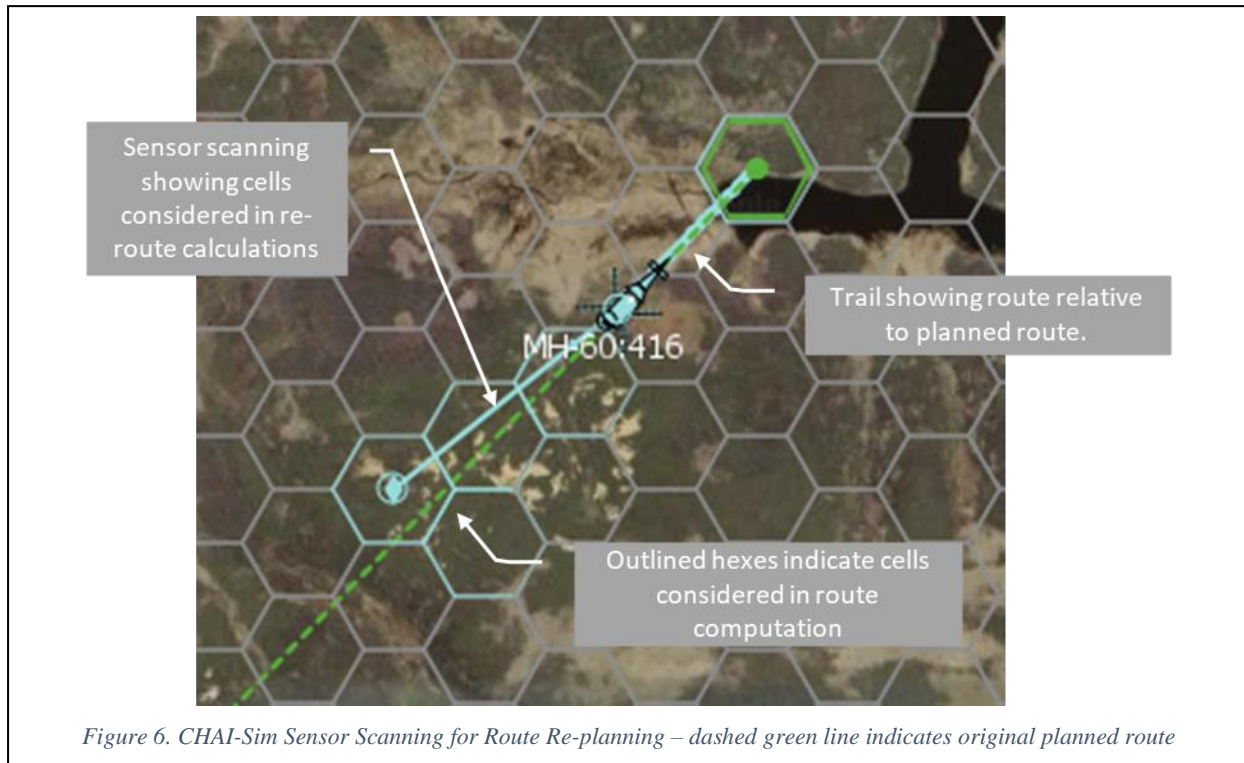


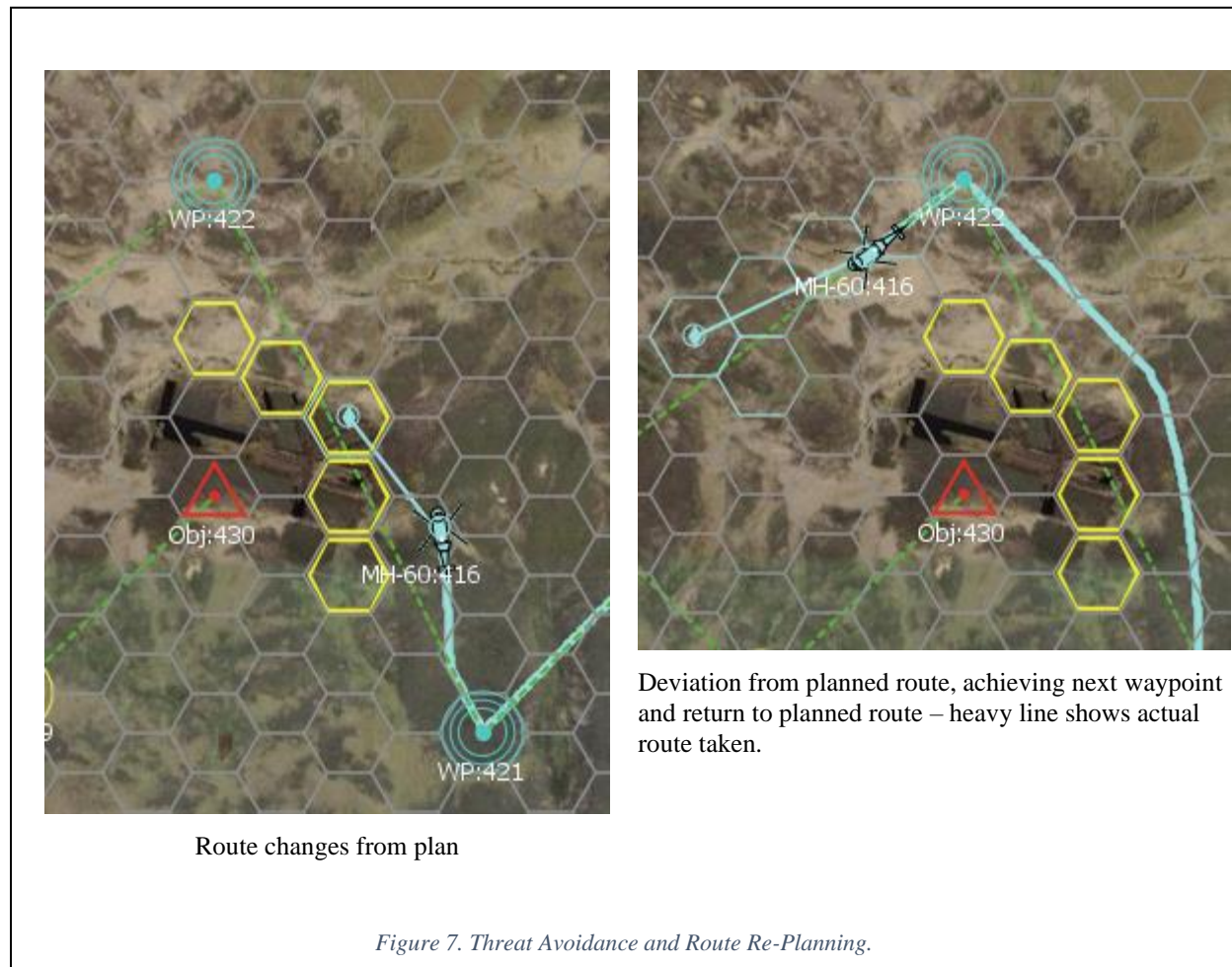
Figure 5 shows the “world view” of the CHAI-Sim. Remember, that every element of interest in the simulation is represented by a software agent. For example, each hexagonal grid represents approximately 234 sq ft and contains elevation and/or other information. This is where ABMS methods excel. In this case, the entire geographical space is represented by a set of agents (hex grid) that can take on values and even contain sense and respond behaviors if desired. This provides a great deal of latitude and scalability in modeling environmental features such as elevation, wind, visibility, etc. In this way, the environment is modeled by a set of hexes which we can see on the world. The principal “actor” in this scenario is a single MH-60 and its crew. It is shown to the upper right of Figure 5 and can be seen on a green hex that indicates the MH-60’s starting position.

Similarly, colored hexes represent entities of interest such as waypoints, objectives, threats, etc. As we have implemented CHAI-Sim for this study, yellow hexes represent areas of high threat.



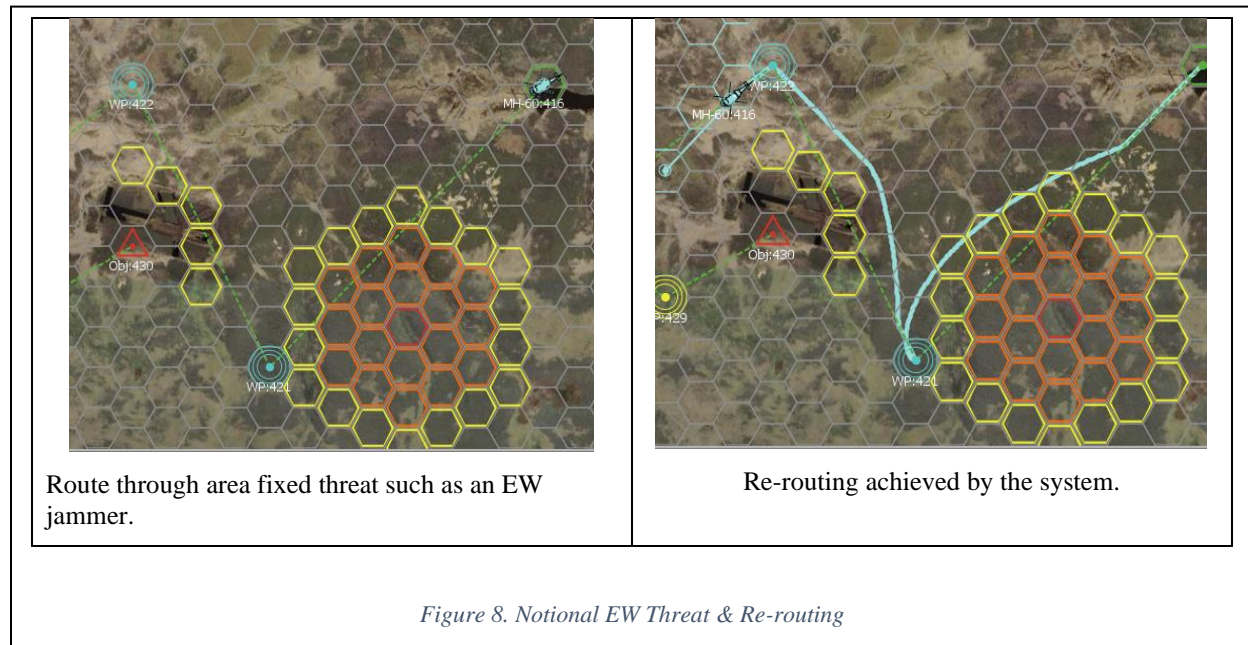
We consider threat generically in this feasibility study and associated it with a high “cost” factor for our primary actor, in this case the MH-60 and its crew. Waypoints, i.e., points through which the MH-60 must pass on its route to its objective, are depicted as hexes with concentric circles. Blue waypoints are known, or expected, to be low threat, whilst yellow waypoints are high threat. The yellow waypoints can produce significant cost in terms of platform and/or crew depending on the scenario, threat, and other conditions in the scenario. Figure 6 is a close-up view showing the CHAI-Sim as it executes a scenario. Sensors are modeled as individual agents in CHAI-Sim. This allows the greatest flexibility to define new sensors and sensor capabilities. It allows simple sensors behaviors to be used initially, and to be later refined and improved as sensor specifications are determined or actual sensor data is obtained. In this study, terrain sensors, radar, electronic warfare, and LIDAR, are considered. As shown in Figure 6, MH-60 sensors are used to provide input to decision-making surrogate model, used in lieu of actual AI route re-planner, to suggest alternative routes. The line extending from the MH-60 is the scan at an instant (during the simulation run, this is constantly changing according to sensor scan rates). Outlined hexes indicate the various hexes from which information was obtained to inform the route recommendation.

Figure 7 shows a point in the simulation where threat information is significant to the extent that the crew accepts a re-routing from the system.



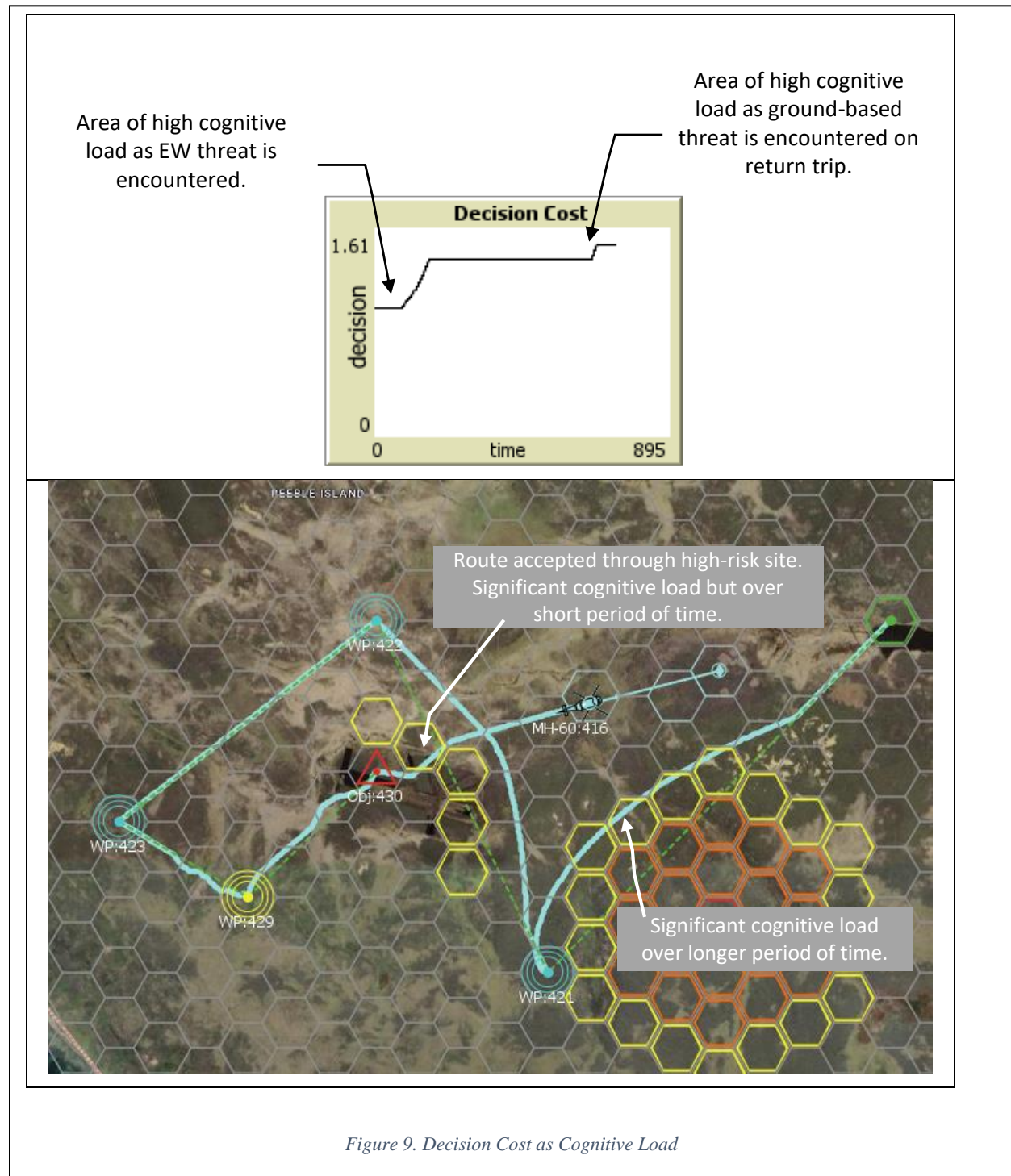
Here we can see how the system uses sensor information about threats and the environment to suggest an alternative route. Although greatly simplified for the purposes and scope of this Phase 1 feasibility study, a good deal of insight is garnered into how sensor data can be fused into information that can be easily understood by SOF crew members.

Whereas the yellow hexagons in the scenario shown here depict high-cost local threats, the large patchwork of hexagons in Figure 8 represents a fixed threat site centered around the red outlined hex in the center. The cost of the threat decreases radially from the center, in this case representative of an electronic warfare jammer. As can be seen, the planned route travels through areas where jamming will be encountered. In the right panel of Figure 8, the system has recommended a route that minimizes the effects of the jammer while still achieving the required



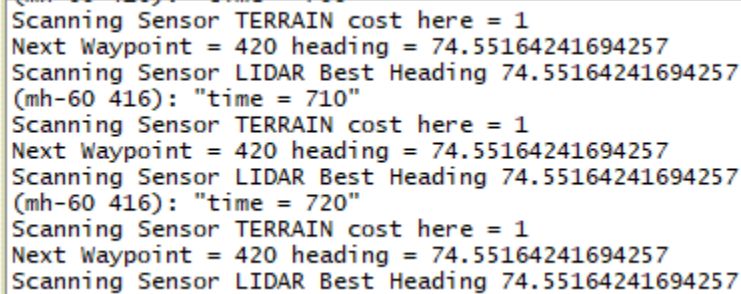
waypoints.

In addition to route planning, we track the Decision Cost, i.e., the relative cognitive load associated with the activities of the human crewed system. In this study, this is notional and related to the number and intensity of the choices or other factors associated with decision-making. We show the Decision Cost over the duration of the mission, representing a total value of cognitive load. The top panel of Figure 9 shows how the decision cost increases during the course of the mission. The first increase we observe is related to the cognitive load associated with traversing the EW threat zone. This can be seen in the bottom panel where the MH-60 traverses the lower threat areas (yellow) but avoids the higher threat areas (orange). The next significant decision cost we observe in the top panel occurs after completion of the objective as the MH-60 chooses to traverse the fixed threat zone (yellow) for a very short period in its return to its starting point.



Additionally, CHAI-Sim presents a running report of system activities and sensor information. Figure 10 is a sample system report. During this feasibility study, we have used the system report

to inform a generative pre-trained large language model (LLM). This was done to explore the use of such LLMs better achieve human-AI understanding and natural information interface. Although more experimentation needs to be done, initial results are promising.



```
Scanning Sensor TERRAIN cost here = 1
Next Waypoint = 420 heading = 74.55164241694257
Scanning Sensor LIDAR Best Heading 74.55164241694257
(mh-60 416): "time = 710"
Scanning Sensor TERRAIN cost here = 1
Next Waypoint = 420 heading = 74.55164241694257
Scanning Sensor LIDAR Best Heading 74.55164241694257
(mh-60 416): "time = 720"
Scanning Sensor TERRAIN cost here = 1
Next Waypoint = 420 heading = 74.55164241694257
Scanning Sensor LIDAR Best Heading 74.55164241694257
```

Figure 10. System Report

The stream of mission information like that shown in Figure 10 was presented to a LLM and various queries were presented. In one case, the LLM was asked to summarize the mission from the data stream and provided rather impressive results. Figure 11 shows the response from the LLM. It's important to note that this was produced nearly instantaneous from the data provided. We anticipate further research and development of using LLMs to infer knowledge from diverse information streams, both on-platform and off-platform, and to use LLMs as another facet of AI-crew communication.

It appears that the data provided is a log of the MH-60's movement and sensor scanning during a mission. Here are some observations:

MH-60 Configuration: The MH-60 is equipped with [RADAR TERRAIN] sensors.

Waypoints and Movement: The MH-60 is navigating through waypoints, starting at waypoint 420 and moving toward the next waypoints. The heading (direction) of the MH-60 is consistently around 224.17 degrees during the initial phase.

Sensor Scanning: The scanning sensor (sensor 417) is providing information on the best heading for the MH-60 based on the terrain. The course correction appears to be consistent with the initial heading.

Cost Evaluation: There is a "cost here" value associated with each time interval. This might represent some cost function related to the terrain or obstacles in the path. The cost remains 0 initially and then increases to 1 after some time.

Waypoint Changes: After reaching waypoint 421, there is a change in the MH-60's heading to 334.03 degrees. Subsequently, there are further changes in headings as the MH-60 progresses through waypoints.

End of Mission: The data seems to end with the MH-60 approaching waypoint 423, and then there is a transition to waypoint 429 and subsequent waypoints. The mission might be ongoing.

Unclear Data: There are some instances where the "cost here" suddenly becomes 1000 or 3, which might indicate an issue or a specific condition in the mission.

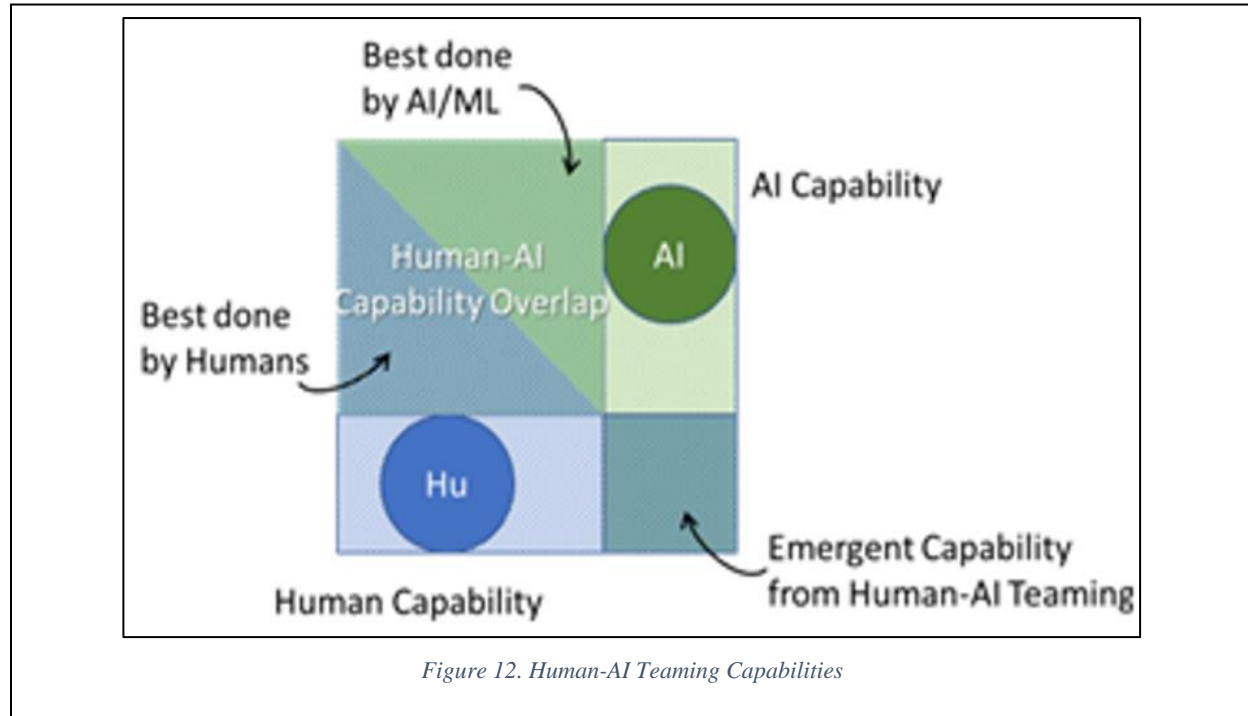
Without additional context or specific questions, it's challenging to provide more detailed insights.

Figure 11. Using an LLM to Infer Information for System Data

2.3 Assessing Human Systems Integration/Interaction Principles Required

Integrating Human and AI Decision-making: The integration of artificial intelligence into SOF helicopter operations presents a profound paradigm shift in flight crew operations and capabilities, offering not only the potential to alleviate the cognitive load and mitigate stress-related uncertainties inherent in human decision-making through sophisticated automation, but to enhance overall crew capabilities through the collaborative dynamic of Human-AI teaming. Such teaming stands to optimize mission outcomes in ways unattainable by either AI or humans in isolation, and even can produce emergent capabilities, i.e., new crew capabilities previously unanticipated (Holland, Foundations for the Modeling and Simulation of Emergent Behavior Systems, 2018). These Emergent Capabilities result from the dynamic interaction between the human crew and their AI crewmember. Part of this Phase 1 feasibility study was to consider that there are activities that can potentially be accomplished by either AI or humans, but to recognize that there are activities “best done by AI”, e.g., route optimization, activities “best done by humans”, e.g., mission prioritization. During the study it became evident that perhaps the greatest increase in capability and subsequent reduction in human cognitive load, will likely come about through

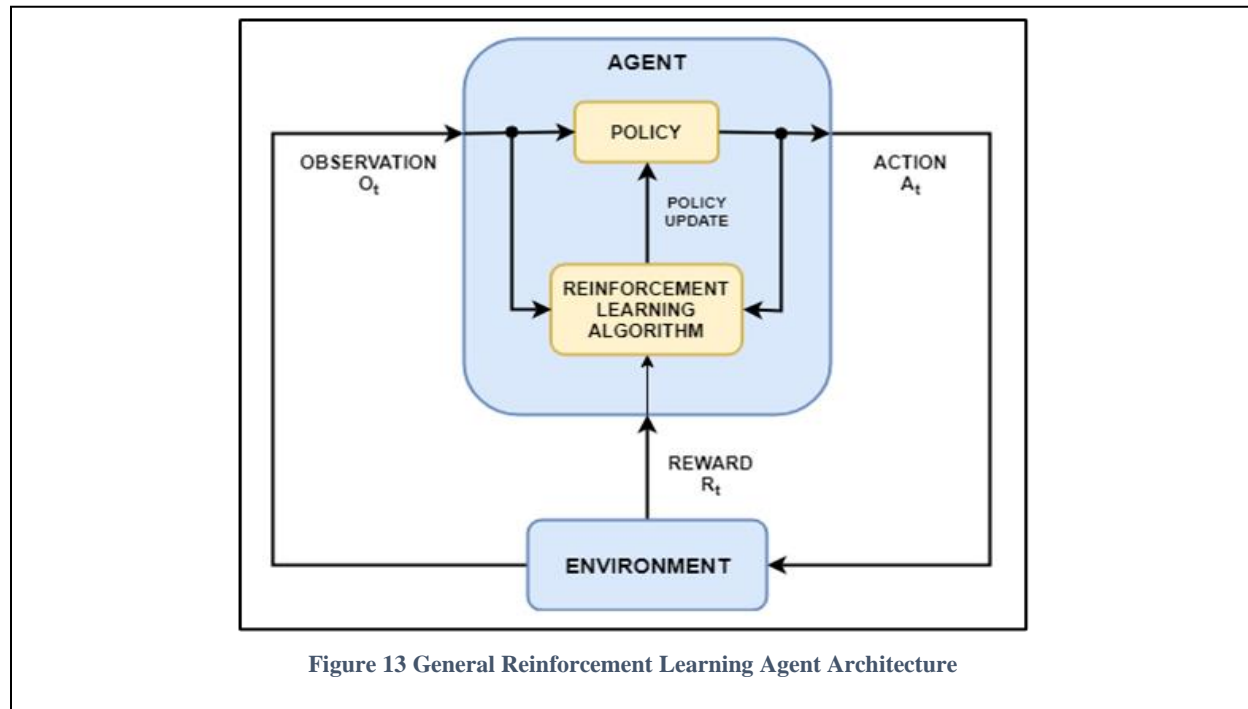
activities that may be more optimal, or only even possible, through the emergent capabilities of the Human-AI team (Human Machine Teaming, 2023) (Rickli, 2023) (Johnson & Vera, 2019). Figure 12 depicts these capability spaces of human, AI, and the emergent capability of Human-AI Teaming.



Exploring Human-AI Teaming in the CHAI-Sim Model: As alluded to previously, ABSMs are adept at analyzing intricate systems with interacting entities, making them invaluable for examining the challenges and system considerations associated with complex systems problems such as integrating AI into military helicopter operations. In our endeavor to systematically explore the complexities of integrating human and AI capabilities for SOF helicopter operations, we utilized our CHAI-Sim to simulate experimental data instead of relying on actual sensors and platforms which was beyond the scope of this Phase 1 study. By employing the CHAI-Sim ABMS, we were able to simulate and convey our comprehension of SOF missions effectively, enabling a thorough investigation into the integration of human and AI capabilities within this domain. Utilizing CHAI-Sim for feasibility exploration served as a virtual experimental platform—a nuanced sandbox, so to speak. This virtual environment facilitated rapid and controlled exploration of theoretical approaches and frameworks. It allowed for the identification of challenges that might arise in a real system and provided representative notional data to support detailed analysis.

2.4 Modeling AI Architecture/Compute Environment

Reinforcement Learning Agent Architecture: Briefly stated, an AI Agent using Reinforcement Learning such as we have described can be considered a kind of “goal directed computational approach where a computer learns to perform a task by interacting with an unknown dynamic environment”. 13 is a high-level functional depiction of this concept. (MATLAB)



In our problem, the **Action** (an output) represents a Route (Flight Path) recommendation. The **Observations** (inputs) are the included sensor and aircraft state and mission data captured by aircraft on-board sensors and contained within the respective receivers and processors.

The **Policy** function selects or maps the Actions to be taken, based on the **Observations** of the **Environment**, and is composed of multiple layers of simple mathematical functions in a neural network. These functions have “tunable” scalar *policy parameters*, i.e., weights, that are updated by the **Reinforcement Learning Algorithm**. Observations are specific features of interest and are the inputs to a neural network that is initialized with various weights (often random). This forms a Reinforcement Learning Neural Network (RLNN) that continuously updates the policy parameters based on the Environment observations, Actions and **Reward**. The reward is generated by an RL algorithm based on the current environment state relative to a desired “end state”. In our case, the “end state” consists of both a *global objective* and a *local objective*. The global objective would consist of the aircraft arriving at an intended target location (exfiltration site), approaching at a desired course and having a pre-determined minimum fuel remaining (flight time). The environment includes some select aircraft state and external environment observations. The RL NN encodes both the Policy (action selection) and the RL algorithm (e.g., Actor-Critic). Significantly in our AEROD framework, the Human pilot (or Crew) must be able to aid

Compute environment. The software, hardware and AI deployment “pipeline” required to implement the modeled approach is informed by research into *embedded AI applications*. Embedded, On-platform, “Edge”, or on-device ML-AI applications are used when constraints on bandwidth (i.e., cloud-based AI-ML); size, weight, power (SWAP) exists, or requirements for privacy/security or latency exist. The compute environment – processors, clock-speed, cooling, and data storage to name a few – are tailored to the specialized application. **Figure** (Lin & Fance, 2023) is an abstract workflow concept of AI-ML deployed to the “edge”. This represents two major components: 1) software tool chains for Training and Compiling tasks and 2) hardware accelerators for running the ML at the edge (to reduce inference latency, efficiency and training time).

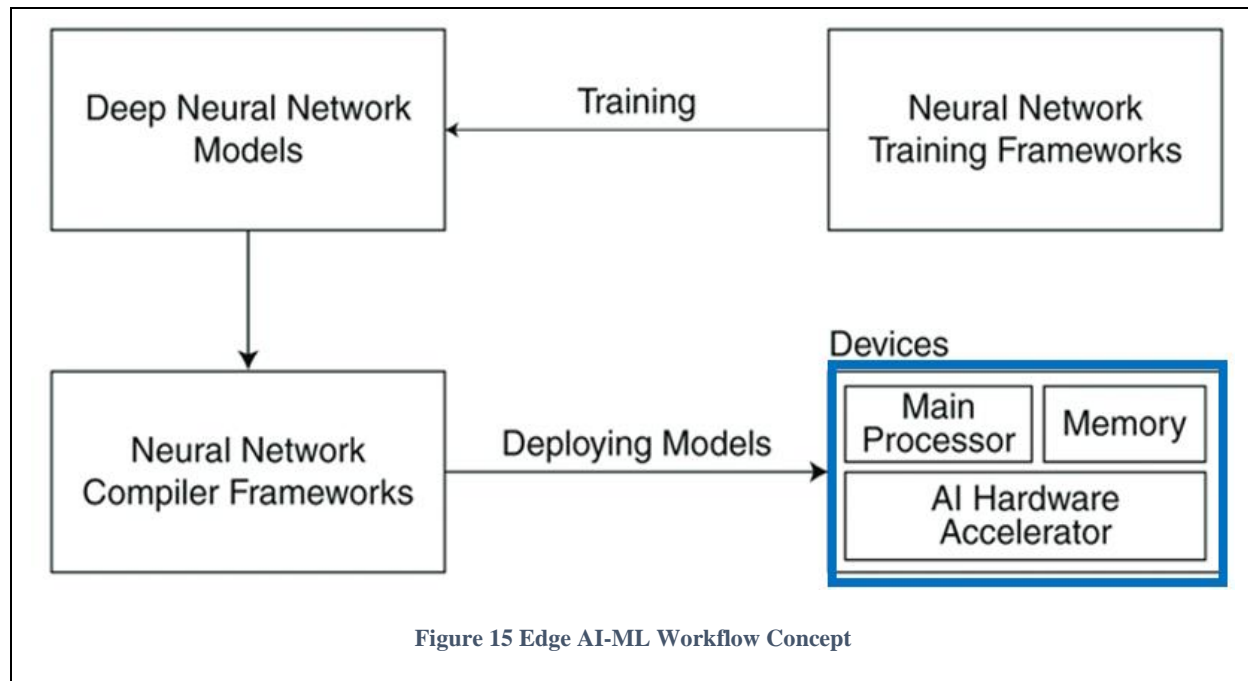


Figure 15 Edge AI-ML Workflow Concept

The NN Training, Models, and Compiler functions represent *off-board* compute environment functions. The *Devices* block denotes the *platform AI* compute environment where the trained AI model is compiled, compressed, and deployed. The AI Hardware Accelerator are the processor chip(s) and design architecture selected that perform the processing. While GPUs are among the most commonly utilized accelerators for Deep Neural Network (DNN) training, their application for us *on* the platform is appropriate. In addition, there are AI System on a Chip (SOC) solutions rapidly entering the market, such as NVIDIA Jetson™. In the design use case of AEROD, embedded constraints concerning Size, Weight, and Power (SWAP) are unlikely to be a factor. Moreover, tailored GPU designs offer the capability to scale AI applications for more data-intensive tasks, such as computer vision. COTs AI system cards, like Jetson™, offer the ability to prototype at very low costs.

Other acceleration techniques are also available. DNN optimization techniques are in active research – and include: *pruning “nodes”* in the DNN, training models in different environment, and/or using a teacher-student model approach to transfer AI training (knowledge) from a larger complex model to a smaller “student” model before compilation. (Lin & Fance, 2023).

Thus, the complete compute environment is composed of two elements: the on-board environment and the training and deployment environment. The on-board compute environment integrates with the aircraft, which will have constraints and resources in terms of power, data-throughput, and processing limits of any aircraft systems that require integration.

The AI on platform requirements such as processor, memory, I/O, and any associated aircraft OS integration for I/O with aircraft data-buses and systems (CAAS, et. al), is dependent on both the aircraft resources *in addition to* the AI *technical* performance needs – that is how many operations per second is required to perform route recommendation inferences.

This technical performance – continuous output of route recommendation - is driven by at least two key measures: (Want, Wang, & al, 2020).

- *AI inference latency* – The time for the AI to process an input “cycle” and produce an output
- *Inferences per second* – The number of times output can be provided/second. This is determined by pilot need - useability and trust.

2.5 Designing with Performance and Implementation Requirements

Design Methodology. Since the performance requirements are derived from the use case – continuous route recommendation (the ML “problem”) – the selection of platform AI design require a flow as reflected in Figure 16. From this figure, we see that the AI application – inferring an optimum flight path will drive the required AI accelerator needs – latency and inferences per second required on-board (Compute Environment).

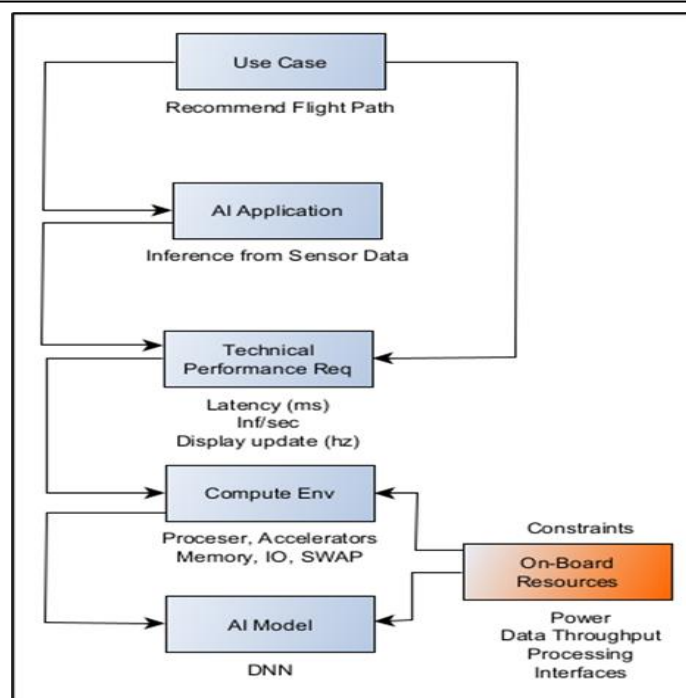


Figure 16. Performance and Compute Environment Design Flow

The *Training and Compiler* (off-board compute environment) can be employed to “seek” the appropriate AI model that can operate within the embedded resource constraints. This model is then deployed to the on-platform AI environment by a specialized compiler that also provides any customized instructions or any memory constraints.

Specific AI applications perform best with different deep network models. For example, Image classification and object detection use Convolutional Neural Networks (CNN) models for automatic pixel level image feature extraction, while Natural Language Processing (NLP) applications use Recurrent Neural Network (RNN) models. Attention-based models – Transformers – are used by Generative AI or “machine translation” tasks. (Want, Wang, & al, 2020).

Deep Reinforcement Learning (DRL) is an approach that enables input of large amounts of unstructured data to “learn” about the environment, which closely matches the AEROD use case requirement.

3 Significant Findings

3.1 Surrogate Simulation Model Use

Within the context of an Armed Personnel Exfiltration mission, CHAI-Sim facilitated the examination of how agent models of AI and human performance can serve as surrogate models for more detailed simulations or actual operational data we would anticipate in a follow-on effort. Operational planning, for example, emerges as a collaborative endeavor where human operators and AI agents work in tandem. AI agents offer data-driven insights and identify patterns, while human operators contribute strategic understanding, contextual knowledge, and ethical considerations.

During ingress and egress phases, CHAI-Sim demonstrates how AI agents can continuously monitor and optimize routes and fuel usage based on real-time data, alleviating human operators from the burdens of distraction and cognitive overload. However, human decision-making remains crucial for adapting to unforeseen obstacles and assessing team members' emotional states, underscoring the complementary roles of humans and AI agents.

Moreover, CHAI-Sim showcases how AI agents enhance threat detection capabilities by actively scanning disparate data sources and synthesizing information for human operators. Human-AI collaboration empowers operators to assess threat significance, apply situational awareness, and make informed decisions regarding threat responses. AI agents, in turn, learn from human feedback to refine their algorithms and improve detection accuracy over time.

Beyond threat detection, CHAI-Sim illustrates how human-AI collaboration extends to data-driven risk assessments. Human operators integrate AI-generated risk assessments with qualitative factors such as cultural nuances and geopolitical context, fostering a symbiotic relationship where AI agents continuously learn and adapt their models based on human input. In essence, CHAI-Sim provides a comprehensive framework for investigating the dynamics, challenges, and opportunities inherent in human-AI teaming across diverse operational contexts. The agent models of AI and human performance within CHAI-Sim serve as surrogates for more detailed simulations, paving the way for future research and development in the field of human-AI interaction and operational decision-making.

3.2 AI System Architecture Evolution

In evaluating the architecture through surrogate simulation modeling, we discovered that a separate agent was highly desired to provide constraints to the ML output, reflecting limits based on physics and Tactics, Techniques and Procedures (TTPs) or other elements that could change mission to mission. Without the need to retrain the ML model, we employ two additional components to the AI system. GTRI research into Large-Language Model (LLM) Routing led us to a more complete AI Agent description, without specifying the ML design, shown in Figure 2.

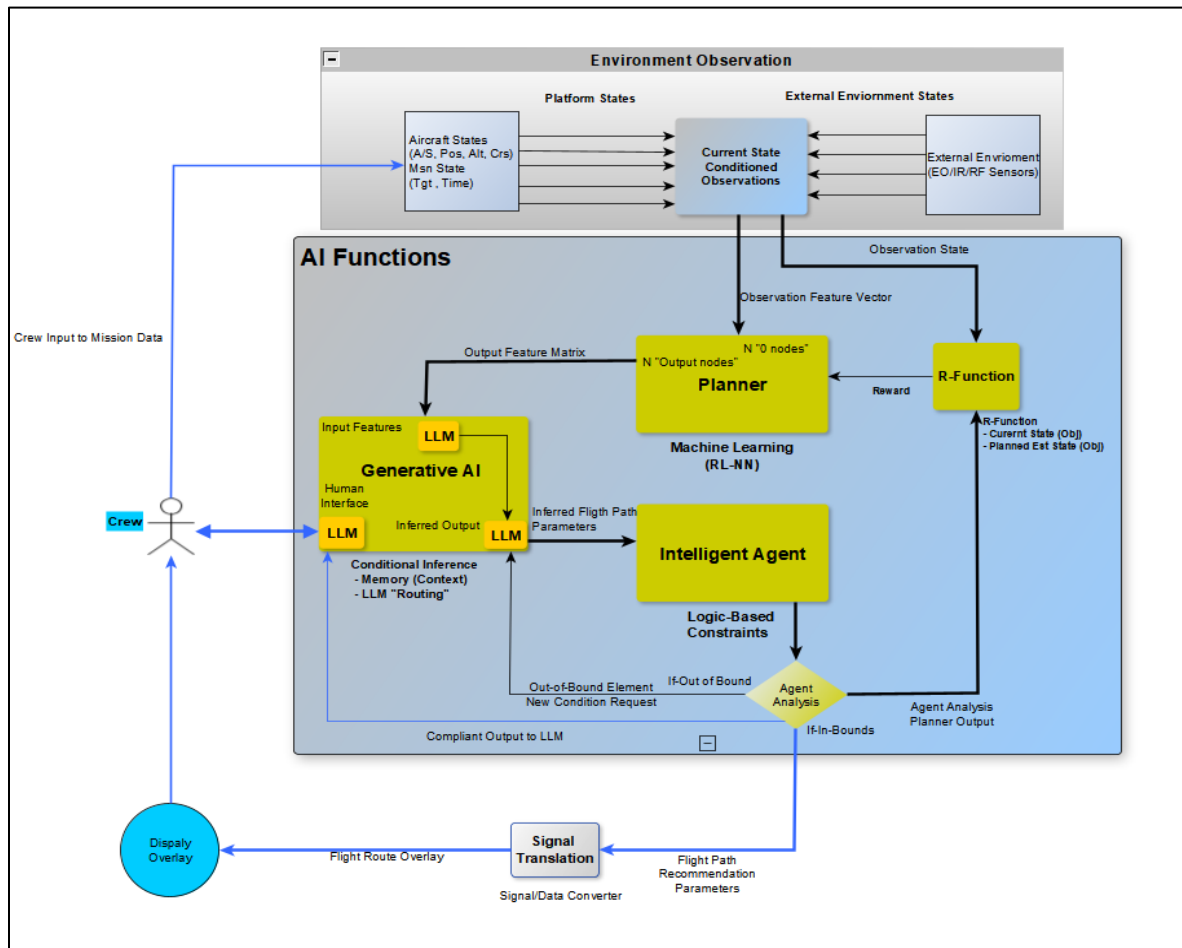


Figure 2 AI System Model Using LLMs

Here we have decomposed the AI System into three components: The *Planner*, *Intelligent Agent* and *Generative AI*.

The *Planner* is the ML, using a DNN architecture, that estimates and plans the optimum flight path to present. The *Intelligent Agent*, is the logic entity module that contains the *constraints* that are applied to the *Planner* output *before* it is presented to the crew. It is the path through which the RL NN is rewarded.

The *Generative AI (GAI)* contains several, discrete LLMs – trained and optimized for the specific type of inference (domain) required - and provides the interaction or “translation:

between the crew and the Agent. The Agent communicates with the Planner through an LLM and is the source of the *Reinforcement Learning* to the Planner.

The Agent analysis returns any parameters out of bounds (constraints) back to the GAI LLM – as a refinement “prompt”. *The Agent also passes all Planner output – as it is communicated via the GAI – to the R-Function to assess how well recommendations meet state objectives (Global and Local).*

The crew also has direct access to mission objectives (Global) through their normal interface with the aircraft mission data, as shown in the line from the crew to Platform states.

With additional crew interaction via the GAI, queries about current state, threats, route recommendations – as demonstrated through the surrogate simulation model – are also possible, ideally by audio.

A hypothesized benefit of modularizing the AI system in this way – is providing an additional mode for *transparency* in the AI – by using a programmable and transparent Agent for critical concerns (i.e., mission safety, aircraft limits, etc.) that are not necessarily observed by the AI data. It provides a check on the *opaque* black box function of the ML Planner to provide *trust and further reduces potential cognitive workload* by removing, or reducing, the need for the crew to evaluate recommendations that are not useful – and offloading that task to a logical Agent.

3.3 Advanced Cognitive Agents

The need for a reinforcement learning (RL) solution – see Section 2.4 – predicates the need to encapsulate this RL solution within an agent-based system. Such a system leverages the knowledge-based RL decision making with a series of observational ‘organs’ (e.g., sensors provided to the agent by which they retrieve information about the world) and affordances to interact with the environment in kind.

The agent design opted into for the Phase II of AEROD is the Soar Cognitive Architecture. Soar is part of a class of AI architectures – which includes ACT-R – that attempts to formally exemplify a theory of cognition (either related to or based on human cognitive processes). Bringing up ACT-R begs the question, “why Soar”? ACT-R is primarily used to model human cognition; this design principle is exemplified in some of the limitations placed on ACT-R, including a limit on the amount of working memory that can be used to human-like levels. Soar was designed to mimic the methods of human cognition to achieve general artificial intelligence.

In brief the processing cycle of Soar begins with the acceptance and Perception of exterior information (like sensors) that work as the foundation for building a model of the world through Scene Graphs relating the observations to relationships known within the semantic memory. This representation of the world is stored in Symbolic Working Memory and compared to already understood Semantic memory graphs. Now comes in Soar’s Procedural Memory, these are actions organized as condition-action pairs with a weighted one-to-many relationships. When a condition within the Procedural Memory is matched with the Working Memory contents, it is selected; but ALL procedures are selected if they match Working Memory conditions. The outcomes are approximated within Working Memory and evaluated by the Decision Procedures recursively until one can be selected and enacted on the environment through the Controllers.

The real iterative procedure is slightly more complex, involving Episodic Learning where the agent memorizes the situation-action-outcome related to a representation in Working Memory for rapid recognition and to support future applications of the Decision Procedure. Furthermore, the weighted one-to-many procedural relationships can be updated iteratively over time once actions are tested over conditions. This can become extended to generate sub-instances of a procedure that are many-to-many rules, where several conditions must be met to marry the rule to the Working Memory.

The affordances this provides within the context of SOCOM's ask:

- Tractable decision making – understanding WHY the system behaved as it did.
 - This will empower future users and engineers to understand why decisions were made and improve the system.
- Simple rule authoring with broad utility allows for the introduction of singular and seemingly trivial relationships that can be trained for broad implications.
- Facilitates RL, meaning that continual improved performance across scenarios can be engineered through additional training and on-performance/mission learning.
- Can enable selective updating based on RL, preserving certain condition-action relations (dictated military policy that cannot be altered)
- Does not accept user feedback on any/all (up to SOCOM) relationships, thereby preventing users from manipulating fixed policies.

Within the Phase II period of performance, development of a baseline scenario very early on will be critical. This baseline scenario will – after some interviews with SMEs and policy documents – enable the generation of the first set of condition-action rules and the authoring of the initial semantic knowledge. This will be tied both to the scenario simulation environment NETLogo and the C (or JAVA) environment-based Soar agent. Variations on the base scenario will be authored automatically for training of the model with the NetLOGO simulation environment.

This will require us to appropriately select the flavor of Soar to be used. The ideal variant is openSoar, this distro is already available in a Docker image, making it rapidly deployable in multiple environments while also running stand-alone from the NETLogo simulation. The binding between the simulation and agent would be best through application programming interface messaging through a helper application called Postman. Postman will accept broadcasted API call from NetLOGO, which accepts and transmits RESTFUL API; Postman shall direct these messages to the Docker's API ports from provision to the openSoar model within. This schema will enable the necessary back-and-forth communication between the AEROD RL agent and the NetLOGO simulation environment.

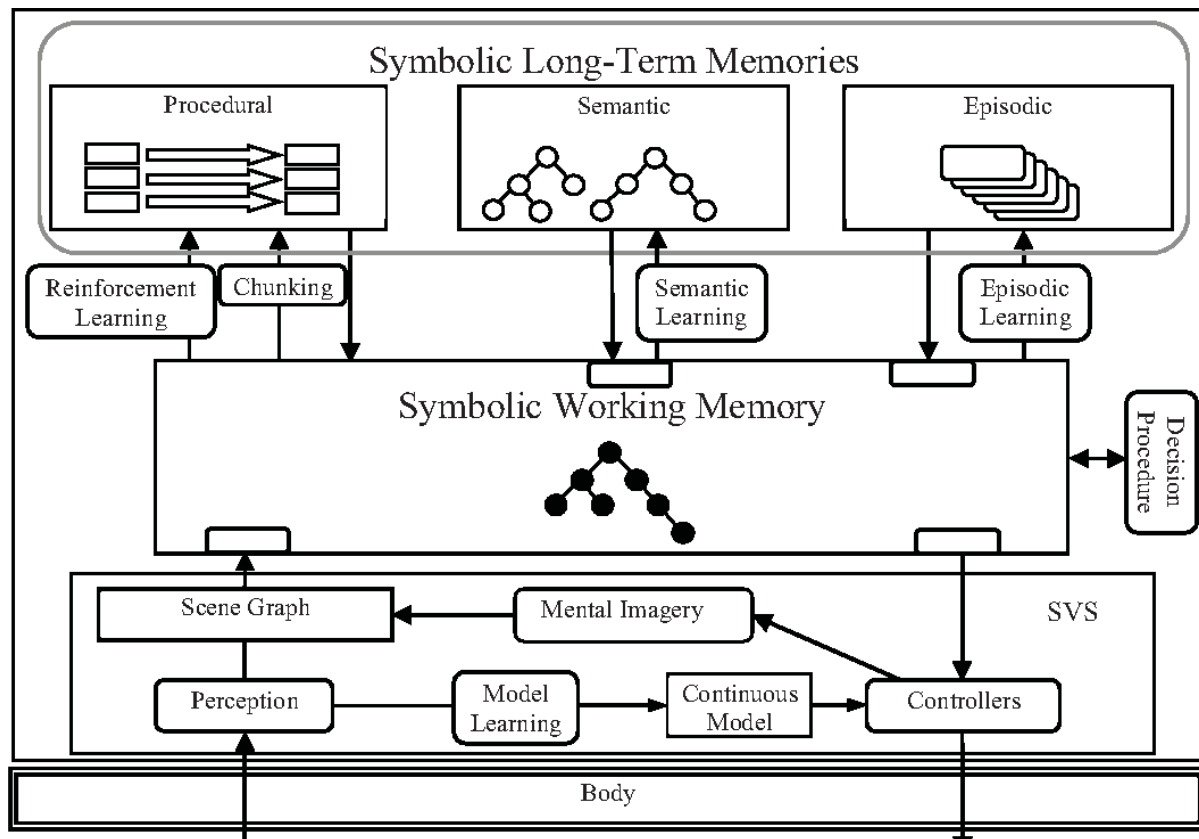


Figure 17 Soar Cognitive Architecture

3.4 System Deployment Environment

Within the designated deployment environment for the overall AEROD software package (simulation, visualization, and ML) each of the major components shall be developed and deployed within a Docker container. This deployment strategy enables us to generate each element for a single operating condition and repeatedly deploy within new environment. This proves most of our effort against any future requirements governing environment OS; a great many government systems operate within different Linux flavors, and Docker is a great way of generating agnostic tools and dropping them in various environments. Our Docker containers will be originally deployed within an OpenSuse Linux, both free and very similar to more enterprise distributions of Linux such as RedHat.

Each container will rely on those tools executing in the remaining containers. Communication between these containers shall be managed by Postman API, an Application Programming Interface management tool, that enabled addressed (directed to specific ports or applications) messages of a fixed and defined (well-structured) format. We shall endeavor to mimic DIS as much as possible when developing our API structures for the large degree of transferability it affords when plugging into preexisting simulators and system.

Within Figure XX, we can see that incoming sensor feeds (during an operational/simulator integrated use case) are provided to the NetLOGO container, where the updated hex map is then

provided to Tkinter and the updated world state is passed to a Soar Agent. This enables the Soar Agent to develop its policy-based actions to Tkinter and the onboard LLM. The LLM will generate textual information to the aviators and the routines within Tkinter will be responsible for visualizing any graphical overlays to the NetLOGO images and other HUD elements and providing the LLM text to the aviators. See Figure 18 for a graphical representation.

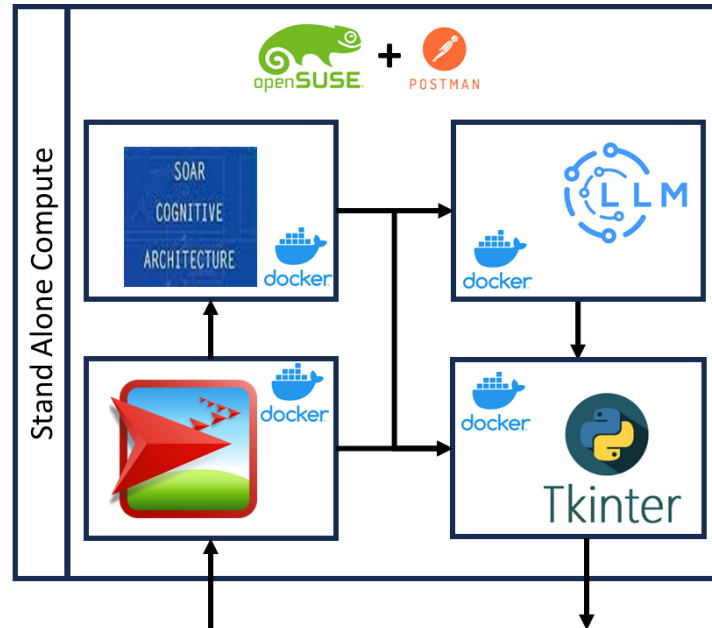


Figure 18 Overall software deployment scheme for Phase 2

4 Summary and Recommendations

Flight path optimization, or recommendation, is a dynamic field of research within the AI and aerospace communities. The concept involves utilizing data such as winds and convective weather to continuously adjust flight trajectories for commercial flight operations, aiming to enhance efficiency in terms of both time and fuel consumption. This approach has been put forward by the Georgia Institute of Technology in collaboration with the Joint Air Mobility Center (Georgia Institute of Technology, Justin, & Mavris, 2022). The research discussed in this report outlines a similar strategy, leveraging available tactical data to continually plan and visualize flight paths based on the associated risks and desired outcomes.

We posit that a software module stack comprising a machine learning algorithm and an agent for quality control, connected via LLM, offers a promising avenue for establishing trustworthy AI-human collaboration.

Specifically, we recommend:

1. Building a surrogate simulation that connects to the Reinforcement Learning (RL) module to generate pseudo flight data. This will facilitate the collection of detailed data on the agent's interactions and decision-making processes, which can be analyzed to gain insights into its behavior and performance. This data can inform the development of more effective RL algorithms and decision-making strategies.

2. Use of RL – an unsupervised ML method – that can more readily adapt to different mission contexts, without extensive, time-consuming, labeling of data required by supervised ML techniques.
3. Use a cognitive agent-model – i.e., SOAR Cognitive Architecture - within the Logical Agent. The SOAR is a “general cognitive architecture for developing systems that exhibit intelligent behavior” (SOAR - University of Michigan, n.d.). It employs a computationally inexpensive truth maintenance algorithm to update beliefs about the world and provides a means for the AI to better understand human decision-making contexts – such as TTPs, mission constraints and commanders’ intent.
4. Investigate the integration of LLM as the AI component that routes information within the AI system and provide a means of natural interaction with the human crew.
5. The computational needs to perform this architecture, will specify the hardware and computational acceleration techniques required to develop a prototype.

Research from the Georgia Institute of Technology and the Joint Air Mobility Center highlights the potential for this approach. This Phase I Study suggests that a software stack comprising a machine learning algorithm and a quality control agent, connected via LLMs, provides an innovative approach to reliable Human-AI teaming. Key recommendations include building a surrogate simulation for detailed data collection, utilizing reinforcement learning for adaptability, employing a cognitive agent-model like SOAR for intelligent behavior, integrating LLM for information routing and human interaction, and addressing computational requirements for prototype development.

5 Transition Strategy

“More is needed now, and at scale.” (*DOD Data, Analytics, and Artificial Intelligence Adoption Strategy, Jun 2023*)

Our strategy is to focus on the novel use of LLM techniques to enhance human adoption and interface with well-scoped machine learning problems to understand the AI ecosystem required to scale AI capabilities for future acquisitions.

AI technology “adoption” within the commercial (consumer and business) sector has created a demand signal to support rapid prototyping, AI-training and deployment pipelines. While hardware ML accelerators (e.g., NVIDIA Jetson™), cloud ML tools (e.g., MS Azure) and downloadable and tailorable LLMs (e.g., Llama), AI pipeline – model, data and testing/assurance, remain a challenge. Unlike mature software development – DevOps – with Continuous Integration/Continuous Delivery (CI/CD), to standardize and ease complex software deployment, application to AI is an active research area. CI/CD automation for AI deployment is particular difficult due to need for Data Handling and Model Learning tasks. (Steidl, Felderer, & Ramler, 2023)

DOD adoption of AI will require the employment of similar capabilities to test, deploy and sustain AI and related supporting technologies, with the specific needs to integrate with existing platforms. These novel capabilities must also respond to heterogeneous use case and user needs, and share a development and deployment ecosystems. From the DOD Data, Analytics, and Artificial Intelligence Adoption Strategy, five key outcomes are desired: (Department of Defense, 2023)

- Employing Agile development principles and approaches
- Building intuitive interfaces to accelerate human adoption
- Developing products with cross-functional teams focused on customer needs
- Offering product portfolios with shared digital foundations
- Experimenting with minimum viable products in operational environments

AI and Machine learning deployed on aircraft will also need to comply with Future Airborne Capability Environment (FACE) reference architecture and technical standards. Adopted capabilities must effectively integrate with aircraft Real-Time Operating System (RTOS), provide for specialized data management, implement specialized development and deployment pipelines and provide means to protect at least the Machine Learning from adversarial attack.

Beyond these common technology integration challenges, however, are some broad challenges facing DOD in realizing AI deployment – with trust. The recent National Academies of Sciences, Engineering, and Medicine report on “Test and Evaluation Challenges in Artificial Intelligence-Enabled Systems for the Department of the Air Force”, Sep 2023, highlighted some of these.

One is the need to adapt “traditional human-machine interaction approaches to tackle future AI-centric *human systems*.” That is, based on the kinds of AI anticipated, a “different approach to how humans learn to work with ‘smart’ machines” will be required (Heckman, 2023). Our second business strategy also focuses on the need to develop cyber security and AI vulnerability and risk assessment weighing. This “justified confidence” will require an “entirely new type of Test and Evaluation” (Heckman, 2023).

Because hardware and software technology elements will be widely available, but the types of AI-enabled capabilities in military platforms will be unique, the machine learning design, training, tuning and data management must be created for repeatable and adaptive AI capability implementation. Nothing will remain static.

The AEROD research effort will support two of the needed outcomes for adopting AI:

1. Support *acceleration of human* adoption by informing Human-AI teaming needs and requirements, and
2. Support *agile* development, through understanding the tailored AI-ML-Data ecosystem that will be required within DOD.

The use of rapid modeling, simulation and experimentation will be pursued to contribute to Service requirements for understanding Human-AI interaction, by defining measures, methods and evaluation criteria. Experimentation is critical for understanding confidence and trust levels of performance.

AEROD, in pursuing demonstration capabilities, will provide insight into embedded machine learning processes that scale will require. Any requirement for embedding AI on human-centric platforms will need to scale for fleet use. This cannot be done without understanding the needs of Machine Learning Operations (MLOps) for single use cases.

6 References

- Department of Defense. (2023). *Data, Analytics, and Artificial Intelligence Adoption Strategy*. Retrieved from https://media.defense.gov/2023/Nov/02/2003333300/-1/-1/1/DOD_DATA_ANALYTICS_AI_ADOPTION_STRATEGY.PDF
- Georgia Institute of Technology, K. J., Justin, C., & Mavris, D. (2022). Data-Driven Approach Using Machine Learning for Real-Time Flight-Path Optimization. *Journal of Aerospace Information Systems*.
- Heckman, L. (2023, Sep 7). Just In: National Academies Report Urges Air Force to Name AI Testing 'Champion'. *National Defense*. Retrieved from <https://www.nationaldefensemagazine.org/articles/2023/9/7/air-force-needs-ai-testing-champion-report-says>
- Holland, O. T. (2007). Taxonomy for the modeling and simulation of emergent behavior systems. *Proceedings of the 2007 spring simulation multiconference-Volume 2*.
- Holland, O. T. (2018). Foundations for the Modeling and Simulation of Emergent Behavior Systems. In *Engineering Emergence: A Modeling and Simulation Approach* (p. 219). Boca Raton, FL: CRC Press.
- Human Machine Teaming*. (2023). (DEFENSE INNOVATION MARKETPLACE) Retrieved from <https://defenseinnovationmarketplace.dtic.mil/technology-interchange-meetings/autonomy-tim/human-machine-teaming/>
- Johnson, M., & Vera, A. (2019, Mar 28). No AI is an island: the case for teaming intelligence. *40*(1), 16-28.
- Lin, H.-Y., & Fance, H. (2023, Sep). Embedded Artificial Intelligence: Intelligence on Devices. *Computer*, 56(2023.09). doi:<https://doi.ieeecomputersociety.org/10.1109/MC.2023.3280397>
- MATLAB. (n.d.). Deep Learning Fundamentals. *MATLAB Documentation*.
- Northwestern University. (2024). *NetLogo*. Retrieved from <https://ccl.northwestern.edu/netlogo/>
- Rickli, J.-M. (2023). Human-Machine Teaming in Artificial Intelligence-Driven Air Power: Future Challenges and Opportunities for the Air Force. *The Air Power Journal*. Retrieved from <https://theairpowerjournal.com/human-machine-teaming-in-artificial-intelligence-driven-air-power-future-challenges-and-opportunities-for-the-air-force/>
- SOAR - University of Michigan. (n.d.). Retrieved from SOAR EECS UMICH EDU: <https://soar.eecs.umich.edu/>
- Steidl, M., Felderer, M., & Ramler, R. (2023). The pipeline for the continuous development of artificial intelligence models - Current state of research and practice. *Journal of Systems and Software*, 199. doi:<https://doi.org/10.1016/j.jss.2023.111615>
- Want, Y., Wang, Q., & al, e. (2020). *Benchmarking the Performance and Energy Efficiency of AI Accelerators for AI Training*. Retrieved from IEEEEXPLORE: <https://ieeexplore.ieee.org/abstract/document/9139681>

