High-Level Orders for Intelligent Agents to Rapidly Generate a Realistic Battlespace

Brian Mills
CAE USA
Arlington, TX
Brian.mills@caemilusa.com

Robert Ducharme
CAE USA
Arlington, TX
Robert.ducharme@caemilusa.com

ABSTRACT

Today the defense industry, the Warfighter, and government leaders are all looking to modeling and simulation to solve new technology and security challenges such as a) the creation of large-scale battlespaces to test new technologies and concepts, b) create the vast amounts of data required to train machine learning models, and c) provide new ways to inform decision makers with campaign-level simulation. Unfortunately, creating complex, multi-domain simulations is a time-consuming and expensive endeavor. This paper presents a unique hybrid Artificial Intelligence (AI) framework rooted in cognitive science and enabled by natural language understanding (NLU) for rapidly generating large and complex multi-domain battlespaces and communicating with intelligent agents within simulations. This hybrid AI framework is then demonstrated via several applied use cases introducing technologies illustrating how intuitive commands can be given to a simulation to generate large formations of entities as well as issue orders and commands to the formations. These use cases further illustrate how a system of intelligent synthetic agents interpret these commands as mission objectives, which are then further broken down into platform-specific tactics that can be applied to the specific scenario to get a desired outcome. Finally, specific research applications are presented to support a validation approach for achieving a true digital twin of the battlespace.

ABOUT THE AUTHORS

Brian Mills is a Modeling & Simulation Engineer at CAE USA Defense & Security in Arlington, TX and leads the engineering team in the CAE Cross-Domain Simulation, Training & Analysis Range (XDSTAR). Mr. Mills has previously worked at such companies as Toyota, FlightSafety International and Fives. Brian is a USAF veteran and received his B.S. in Electrical Engineering Technology from the University of Cincinnati.

Robert Ducharme is a Senior Scientist and Data Science Technical Authority at CAE USA Defense & Security in Arlington, TX. Dr. Ducharme has worked in the training and simulation industry for 23 years and currently specializes in applications of AI / ML in battlespace simulations. He gained his BSc and PhD degrees in theoretical physics from the University of Essex in England.

High-Level Orders for Intelligent Agents to Rapidly Generate a Realistic Battlespace

Brian Mills
CAE USA
Arlington, TX
Brian.mills@caemilusa.com

Robert Ducharme
CAE USA
Arlington, TX
Robert.ducharme@caemilusa.com

INTRODUCTION

Faced with new and significant security challenges posed by near-peer adversaries, we believe there will be an increased emphasis on the use of high-fidelity battlespace simulations to guide decision makers, test and validate new concepts and weapons, provide realistic Multi Domain Operations (MDO) training scenarios, and generate the vast amounts of high-quality data required to train artificial intelligence systems. To create truly realistic synthetic battlespace that can provide for realistic training and aid in decision making is a monumental undertaking because of the complexity of modern MDO scenarios. However, the potential payoff is huge in that the result will be a better prepared Warfighter, with better decision making to guide them and better platforms, weapons, sensors to enable them to achieve mission objectives. More succinctly, if Warfighters are to dominate the real battlespaces of the near future, they must first dominate the synthetic MDO battlefields that the M&S community creates in high-fidelity battlespace simulations (see Figure 1).

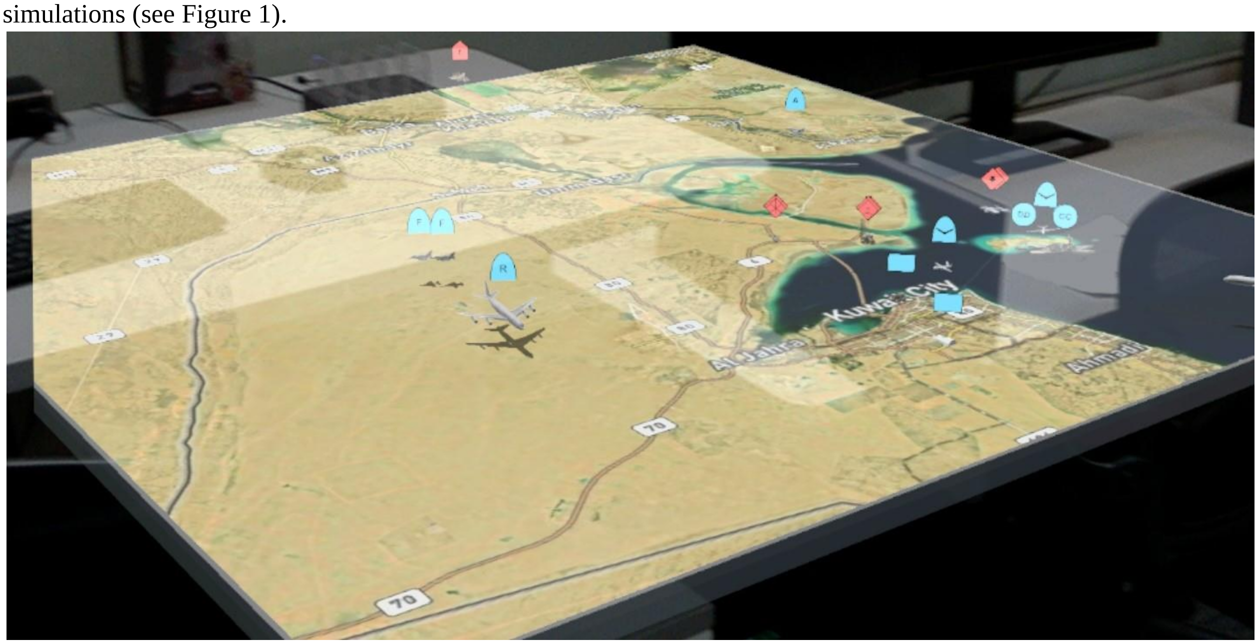

Figure 1. Augmented Reality view of a multi-domain operations scenario.

Virtual battlespace environments have evolved over time to do a reasonably good job of simulating the physics of the various platforms, weapons, and sensors. From our perspective, these applications typically fall into two categories: computer-generated forces (CGF) and modeling and simulation (M&S) applications. CGF applications are geared more towards training, while M&S applications are focused on analysis and engineering. Since they both model the battlespace using physics-based computations and often can be used interchangeably, for this paper we group them together as "battlespace simulations". Examples include the Advanced Framework for Simulation, Integration and Modeling (AFSIM)) (e.g., Zeh, Birkmire, Clive, Johnson, Krisby, Marjamaa, Miklos, Moss, and Yallaly, 2014), Modern Air Combat Environment (MACE) (Battlespace Simulations, Inc, 2022) and Next

Generation Threat System (NGTS) (Naval Air Warfare Center Aircraft Division (NAWCAD), 2018). AFSIM and NGTS are examples of Government Off-The-Shelf (GOTS) solutions, while MACE is a Commercial Off-The-Shelf (COTS) solution. All use physics modeling to represent vehicle dynamics, sensors, and weapons to a high degree of fidelity across multiple domains.

What we see missing from current battlespace simulation is the modeling of the Warfighters inside the platforms. We believe this is important and the lack of human modeling limits the realism (thus the utility) of modern battlespace simulations. Warfighters, for example, communicate with each other in natural language, can learn through experience to perform tasks more quickly with less conscious effort and aren't limited to decisions based on scripted behaviors. It has been convention to exclude these kinds of details in the synthetic battlespace as complicated physics modeling was previously not available before the advent of adequate computational power. However, now that Artificial Intelligence (AI) is improving, so are the prospects for the rise of more realistic intelligent agents that leverage AI technology. With the recent explosion in capability of modern AI, we believe we can create battlespace simulations that feature intelligent synthetic agents having Warfighter characteristics (anthropomorphic) that can take the place of real Warfighters to facilitate training and analysis.

A second area where we believe AI can improve high-fidelity battlespace simulation is to aid in the creation of large, complex scenarios that will be required for decision support and large-scale training events. Battlespace simulations with large numbers of air, surface, ground, sub-surface, space, and cyber assets will be necessary to support training and decision-making. Since most scenario "laydowns", which are files that contain the locations, type and even behaviors of platforms, are typically done manually using scripts or by placing entities on a computer-generated map in a scenario laydown editor, as scenario scope and complexity increases, so will the cost and time to create them. As an example, the 100,000 Entity Scenario in Figure 2 took over 6 months and a full team to enable proper low-, mid-, and high-level behaviors.

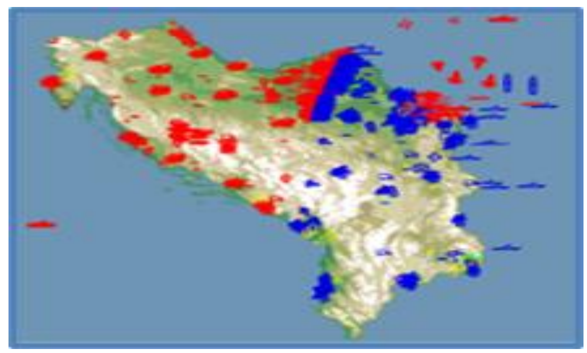

Figure 2 - 100,000 Entity Scenario for Training

Paper Overview

This paper aims to demonstrate that by adding intelligent synthetic agents enabled by AI and Natural Language Understanding (NLU) to existing battlespace simulations, the following can be achieved:

1. Enhance the realism of existing battlespace simulations by modeling the Warfighters who control the platforms, weapons, and sensors.
2. Create command, control, and communication structures in the battlespace.
3. Greatly simplify the creation of large, complex, MDO battlespace scenarios.
4. Reduce the reliance on Warfighter participation in large scale training events such as exercises and mission rehearsals.

Our aim is to share with the M&S community a description of the technical approach utilized to build a set of capabilities enabling the creation of effective intelligent synthetic agents to act as Warfighters that can communicate via a natural language framework. Additionally, we also show the importance of making NLU a first class foundational AI capability, and how it can be leveraged to support the generation of complex scenarios.

This paper first introduces a technical overview of traditional approaches and corresponding limitations. This leads to the presentation of our hybrid AI framework, inspired by the cognitive sciences, to promote communication with intelligent agents via a symbolic AI inferencing engine. After presenting our hybrid AI framework, we share some actual applied use-cases, identifying the various challenges they posed and some key findings. The paper concludes with a discussion of future research plans as well as possible use cases where we believe our approach can provide tangible benefit.

TECHNICAL OVERVIEW

This section provides an overview of current approaches to model entity behaviors, the need for anthropomorphizing communications with intelligent synthetic Warfighters, and then a few of the main issues associated with the exclusive use of ML methods.

Current Approaches to Modeling Entity Behaviors

Before diving deeper into our approach of modeling the Warfighters, we present a brief survey of some the current methods for modeling entity behaviors in M&S and CGF applications like MACE, AFSIM and NGTS. These range from scripts and state machines to behavior graphs (see Table 1) and typically model platform and Warfighter (abstractly) together as a single entity.

Table 1. **A comparison of three different methods for creating entity behaviors with benefits and drawbacks based on our experience.**

Method	CGF/M&S Application	Benefit	Drawback
Scripts	MACE, AFSIM	Good for simple scenarios	Limited dynamics behavior. Often must be modified when scenario is changed
State Machine	AFSIM	Can result in relatively complex behaviors.	AFSIM state machines can be difficult to construct. Have logic gates that must be evaluated.
Behavior Graph	NGTS	Can create moderately complex behaviors through use of a graphical editor.	Graphs can grow to be quite complex. Can have many logic gates that require evaluation.

The above table shows a comparison between different methods of creating entity behaviors in a few of the common battlespace simulations. Scripts can be relatively quick and useful for very simple scenarios. State machines and behavior graphs require more time to build but can result in more dynamic behaviors than what a script can typically provide. Behavior graphs like those used in NGTS are a feature in gaming engines such as Epic's Unreal engine and can work quite well for the creation of convincing behaviors. However, for complex behaviors the graphs can become quite time consuming to create by hand and may contain a very large number of logic gates that need to be evaluated, potentially slowing execution of the simulation.

Using AI to Model the Warfighter as a Separate Entity

Something common to the approach to creating entity behaviors in battlespace simulations regardless of the implementation method is that none model the Warfighter as a separate and distinct entity from the platform they control. Also, they completely lack any natural language communication capability which makes intuitive interaction with them difficult. Thus, our research has led to the idea of modeling the Warfighter by using intelligent synthetic agents that are separate from the platforms in the simulation, and that can be directed by issuing them high-level orders with natural language commands as well as military-specific brevity language commands. This research has resulted in the creation of a standalone AI software application that communicates with battlespace simulations such as MACE. This AI software is responsible for modeling the Warfighter decision making and communication networks that are critical to achieving realism and leaves the battlespace simulations to handle the physics of the platforms and environment.

Thus, our goal is the creation of intelligent synthetic agents using AI to model the Warfighter in the platform. We then define AI as the study and design of intelligent agents that can perceive their environment and take autonomous actions. However, to properly model Warfighters, other disciplines of science should be considered to better understand how they think and make decisions. It is also necessary to give them the ability to communicate with understandable language, including brevity codes or other shorthand. In our view, the modeling of Warfighters is the intersection of neuroscience, cognitive science, and AI. This view has defined our approach to the modeling of Warfighters and has led to an interesting conclusion: ML alone may not to be the best way to model Warfighter behavior in a battlespace simulation because of some practical limitations and that ML does not really reflect how

Warfighters think and make decisions. Before we dive into this last point, we shall first discuss the current state of ML in battlespace simulation and our perspective on where it can fit in.

Why Not Machine Learning?

With its inherent strengths in NLU, image recognition and making predictions, ML would seem to be the "go-to" solution for building intelligent synthetic agents that behave like real Warfighters. However, ML is still not found in today's battlespace simulations (e.g., Oijen and Toubman, 2021) despite what would appear to be inherent benefits to its use. Presented at the Interservice/Industry Training, Simulation, and Education Conference (IITSEC), Oijen and Toubman (2021) revealed that when they surveyed 9 different battlespace simulation applications, there was not one mention of ML. There are practical reasons for this apparent lack of ML in modern battlespace applications:

1. The sheer breadth of platforms that must be modeled would require numerous trained ML models. Training even a single ML model to get reliable results is a non-trivial undertaking.
2. The lack of quality data available and/or difficulty in generating quality data to train ML models to perform as Warfighters in a modern battlespace.
3. The recent focus on assurance and trust in ML algorithms. If we cannot trust that our ML models will behave according to the laws of physics and other real constraints, then how can we trust the outcome of the simulation as a whole?

A HYBRID AI FRAMEWORK INSPIRED BY COGNITIVE SCIENCE

A few practical reasons have been established for not relying solely on ML. However, there is a more fundamental concern with ML in that it does not necessarily represent how Warfighters think and make decisions. What resonates and has shaped our approach to AI is the Kahneman Decision Making Model (e.g., Kahneman, 2011). In this model, decision making is broken down into so called System-1 and System-2 thinking. System-1 thinking represents the fast brain which uses associative thinking based on experience and intuition. Driving a car, catching a ball, and making a gut decision all use System-1 thinking. Decisions made using System-1 thinking are based on repeated training, association, or heuristics. We can think of System-1 as being roughly analogous to how ML models operate and arrive at decisions. System-2 thinking represents the slow brain which relies upon on rules and logic to arrive at decisions. System-2 thinking is comparatively slow and deliberate. Symbolic logic as found in traditional AI is representative of this type of reasoning. By leveraging both systems together, Warfighters can learn new tasks more quickly and offload these learned tasks from the slow brain to the fast brain.

To do a better job of modeling Warfighters with AI we strive to leverage both traditional symbolic code that mimics System-2 and ML that mimics System-1 in a hybrid AI. Thus our AI solution to model Warfighter decision making in battlespace simulations is to take a hybrid approach. Oijen and Toubman (2021) also mention the benefit of a hybrid AI approach in the realm of battlespace simulations in that Warfighters may have constraints in how they can arrive at decisions (rules) as well as a need to react to unfamiliar situations. A distinction can be made with our hybrid AI approach in that we are relying primarily on symbolic AI for Warfighter decision making, and ML techniques for communication and NLU. Specifically, our hybrid AI relies on a symbolic core based on a specialized form of directed graph and ML for its natural language understanding capability (see Figure 3).

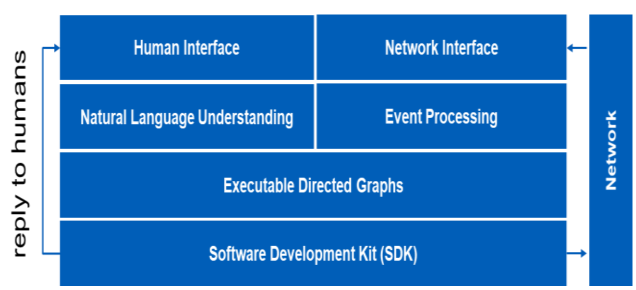

Figure 3. Simplified Diagram of our Technology Stack

In taking this hybrid approach some of the practical difficulties of relying on ML for decision making are avoided, though we have a desire to eventually use ML for some decision making where it makes sense. The resulting intelligent agents think and make decisions symbolically by using a system of semantic graphs that are autogenerated from facts. The key difference between this form of graph and an Artificial Neural Network (ANN) found in ML is that our graphs are trained by facts, and not by probabilistic learning. The resulting intelligent agents can be commanded using high-level natural-language orders via a chat interface or orders file read in at runtime, and control platforms modeled in battlespace simulation applications. This design can potentially also address two of the drawbacks for finite state machines and behavior graphs in that there are no logic gates that require evaluation and with the addition of an ingestion engine (using ML) the creation of large and complex graphs can be (at least partially) automated.

Communicating with Intelligent Agents

Intelligent agents require a means of communication that is understandable by the Warfighter to maximize the agent's flexibility and utility. Thus, we have chosen to focus much of our AI efforts on the use of NLU in our intelligent agents to give them a means of communicating. By taking this approach, we believe the following can be achieved:

1. Facilitate natural communication between intelligent synthetic agents and Warfighters .
2. Build Command, Control, and Communication (C3) networks within the simulation.
3. Rapidly generate complex, multi-domain scenarios.

Using the hybrid AI, an analyst or engineer can send orders to an intelligent synthetic Warfighter using layperson language, or a Warfighter can communicate using the North Atlantic Treaty Organization (NATO) brevity language. These orders are processed in an NLU engine where they are broken down into structured language. This structured language flows down a command chain populated by other intelligent agents, who communicate to each other with the same structured language. A simple example is ordering an attack on a portion of an enemy Integrated Air Defense System (IADS). This order is given in layperson language and is directed to the theater commander "BlueBoss" (see Figure 4). The system's NLU engine processes the command and generates the correct structured language which is

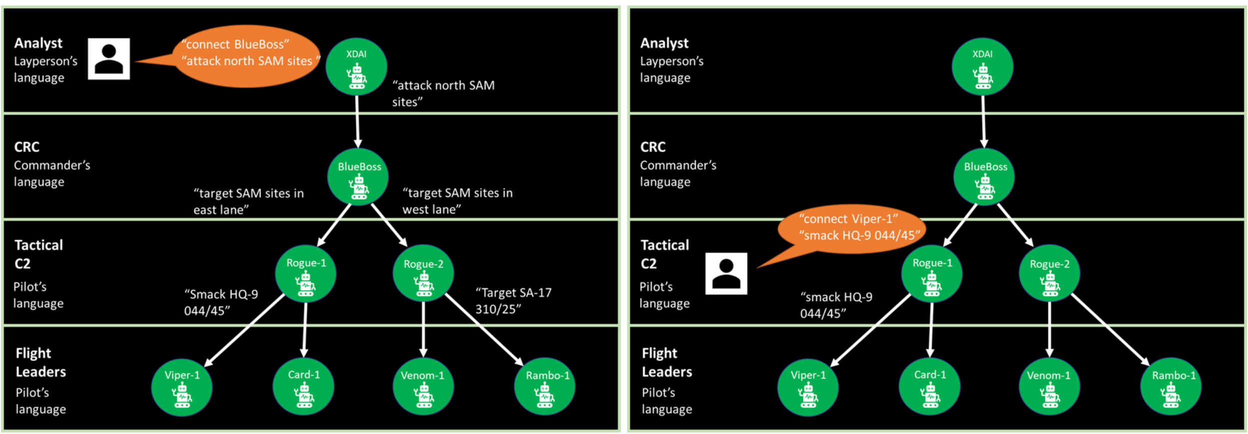

Figure 4. Orders given for a user logged into the "Analyst" view as compared to a user logged in as "Rogue-1".

passed on to the next agent in the command chain. As orders flow down the command chain, each intelligent agent issues orders to a subordinate via structured language. This structured language can be translated back to the correct natural language or brevity language based on the identity of the sender and receiver as well as their location in the command hierarchy.

To facilitate different types of users, in this hierarchical language framework we have the notion of having views into the NLU. A view can be thought of as a type of user in the system. Examples of views are "analyst", "Viper-1", and "IADS commander." If the user is logged into the "analyst" view, they are at the top of the command chain and can command any entity using layperson language to allow them to fully control a simulation without having to learn military jargon or brevity language. The view that a user is logged into determines who they can command and what language they use. If logged in as an "analyst", then they communicate in layperson language, or if the user is logged into the system as "Viper-1", then they communicate in brevity language. With multiple views and the associated language, the hope is to allow for maximum flexibility of use cases.

The high-level orders given to the intelligent synthetic agents result in specific tactics being carried out by platforms in the scenario, as well as the generation of scenarios containing formations of synthetic warfighters (and their platforms) along with their mission objectives. While the initial focus has primarily been in the air-domain, eventually we aim to map the same high-level orders command set across multiple domains and to expand the scenario generation capability, so that an analyst can rapidly generate a realistic scenario in multiple domains and issue the same commands (for example “destroy”, “defend”) to synthetic pilots as they can to synthetic ground vehicle operators and have the resulting behavior adhere to doctrine while contributing to a desired mission outcome.

Natural Language Understanding and Semantic Networks

NLU takes spoken utterances in the form of text and transforms them into structured data. A user, for example, may say “climb to 10000ft” and the NLU will output {“intent”: “set data”, “entity name”: “altitude”, “entity value”: “10000ft”}. Of course, for this to be useful the intelligent agent must be able to derive meaning from the structured data so it can act on it. The branch of linguistics concerned with meaning is called semantics and semantic graphs are concerned with concepts and relationships between them. These graphs are not sufficient for our purposes since the graph needs to be executable in the sense of driving Application Programming Interface (API) calls to our Software Development Kit (SDK). Regardless, in seeking to construct a form of executable directed graph that goes beyond behavior graphs to incorporate semantics and other traits such the ability to compose complex data like natural language text, we have been inspired by a highly specialized form of semantic graph called an executable semantic graph (Sowa, J. 1991). Executable semantic graphs (see Figure 5) differ from semantic graphs in that they allow for specialized features such messaging passing, attached procedures and learning.

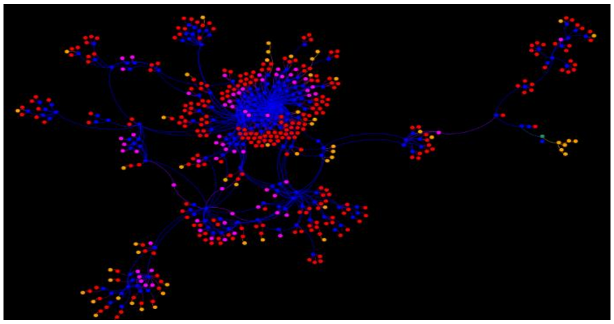

Figure 5. A graph generated with Gephi showing our chat application. The different colors reflect the different types of nodes. See body of this section for further explanation.

Directed graphs of the kind that we need are too big and complex for manual generation. This has led us to seeking automation to create them from requirements. One key aspect of this process is visualization of the graph using an open-source graphing tool called Gephi. These visualizations distinguish between 5 different categories of node:

1. Action nodes (red) make one or more API calls to our SDK.
2. Branching nodes (purple) enable joins between fact patterns in decision making.
3. Compose nodes (yellow) represent nested data. Edges between nodes allow for higher degrees of nesting.
4. Decision nodes (blue) represent fact patterns. The edges between these nodes add additional information to the fact patterns (logical and) and the splits on nodes (logical or) allows for alternate paths through the graph.
5. Event nodes (green) constitute the input layer to the directed graph.

It is our practice to package all the event data entering the intelligent synthetic agents into datagrams giving the appearance of a single input. This single input could represent many things such as a Distributed Interactive Simulation (DIS) packet, an event from a Graphical User Interface (GUI), or structured data from our NLU algorithm.

Mapping Inputs to Actions

Natural-language orders are eventually mapped to actions, which can be anything from calling a network API to setting event triggers for reactive behaviors. However, orders must first be converted into structured language that can be interpreted by the nodes in the executable semantic network. It is also here that context (such as intent, nationality, platform type, and domain as well as other factors) is evaluated so that the correct tactics and doctrines can be followed. This is possible because the AI knows a wealth of information about the synthetic Warfighters as well as the platforms and systems. The information is captured in data files that define a “knowledge database” for the scenario that the AI can access to influence which actions to take for a given Warfighter in a mission. The knowledge database is still early in development and currently focuses on the air domain. However, we look to eventually map high-level orders given in layperson language to tactics and behaviors across all domains. For example, a simple “defend” command could map into Combat Air Patrol (CAP) or Defensive Counter-Air (DCA) missions for the air domain, or convoy escort missions in both land and sea domains.

Achieving the goal of issuing orders in layperson language to create realistic and doctrinally correct behaviors will require the entry of a large amount of data into the hybrid AI to create a suitable knowledge database. The hope is that eventually the construction of the database can be automated using an ingestion engine and ML. This approach of using ML to populate a knowledge database which is then accessed by a symbolic AI inferencing engine is gaining traction and was notably demonstrated by a research project that studied the use of neuro-symbolic AI with reasoning abilities similar to those of humans (Ananthaswamy, A. 2020).

USE CASE 1: VIRTUAL ISR TRAINER

Our first use of intelligent agents with natural language capabilities came with the development of the Virtual Intelligence Surveillance Reconnaissance Synthetic Training Application (VISTA) prototype in late 2018. This software allows for the training of intelligence officers without the use of expensive remotely piloted drones and their associated personnel and equipment infrastructure. It does this by placing the student in a realistic scenario where they can task drone operators and receive synthetic data from virtual sensor feeds. This normally requires an operator to pilot each drone while another operator controls the drone's sensors. In a typical scenario, four people would be required to facilitate a realistic training session (and two real drones). To reduce the reliance on humans and equipment, constructive drones piloted by intelligent synthetic agents were created, and a chat interface was developed to communicate with the synthetic agents.

Tactical Chat

Chat interfaces based on Internet Relay Chat (IRC) are commonly used by intelligence officers to communicate with drone operators, so they are a natural fit to serve as the user interface to communicate to the intelligent synthetic agents that replace the drone operators. Chat interfaces (see figure 6) have the additional advantages of being highly effective for distributed, low bandwidth environments and they provide a persistent record of the communication. This enables the user to participate in multiple chat rooms simultaneously for updating a common operational picture.

The intelligent synthetic agent used here is capable of responding to 20 instructions giving the chat user fine control over the size, shape, and location of the orbit as well as more basic flight control information such as altitude and speed. Also, the sensor operator agent will need instruction on where to point the sensor ball, what zoom level to apply and what scan pattern to use. The user can also command additional options for video mode, switching between manual and automated tracking, buddy lasing and weapons deployment.

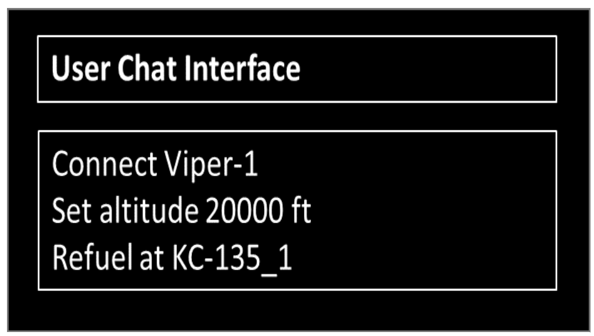

Figure 6. Example of chat commands to the intelligent synthetic agents.

The Intelligence Surveillance Reconnaissance (ISR) use case is similar to pilot training using synthetic Air Traffic Controller (ATC) agents (e.g., Harrison, Hobbs, Howes, and Cope, 1986; Lin, 2021) that listen to pilots through speech recognition and send instructions back using speech synthesis. The synthetic ATC agents must be smart in the sense of inferencing situational awareness as the basis of formulating their instructions to the pilots but clearly differ from our synthetic agents both in the use of speech versus IRC chat as well as in the detail that our intelligent synthetic agents are the actual equipment operators.

Key Findings

A few key findings from this ISR research and development effort:

1. There is a loss of fidelity in speech to text translation, since no speech recognition system is perfect, that is avoided if the text is typed rather than spoken.
2. Our first approach to interpreting tactical chat in 2018 used Rasa NLU (Bocklisch, Faulkner, Pawlowski, and Nichol, 2017) for intent classification. This worked quite well but did not give a significant benefit over simply picking out key phrases and abbreviations using a regular expression approach owing to the small size and highly specialized nature of the language.

3. Our intelligent agents have been successful in reducing the number of drone and sensor operators by a factor of four but have not yet met our goal of a completely dependable autonomous system.
4. The challenge of interpreting tactical chat is that it is not a perfectly defined standard. There are rules and standard terminology and abbreviations, but it needs be flexible enough to support multi-service tactics and behaviors.

USE CASE 2: AVIONICS RESOURCE MANAGER

This study required a scenario with dozens of constructive entities to serve as a threat environment for a virtual mission trainer (with human pilot) outfitted with a ML model for a new avionics system. The goal was to generate a large amount of electromagnetic data to train the ML model in an avionics resource manager. The pilot in the virtual aircraft configured the avionics resource manager as the mission unfolded to set the correct mode for the ML model training. The scenario laydown consisted of enemy fighters, enemy IADS, and a high-value target. Friendly assets included virtual and constructive fighters. The scenario required that the human pilot in the virtual mission trainer task constructive wingmen in real-time as the mission unfolded to help generate the necessary stimulating data for the ML model.

The challenges in this study were:

- Ingestion of an existing scenario laydown into the simulation environment.
- The composition of simulated avionics data from low-level data in network packets.

Both challenges were solved by leveraging the NLU capability that is foundational to our AI and that underpins the chat interface. An ingestion engine (leveraging existing NLU capability) was developed to read in natural language orders files at runtime. An data composition agent was then designed using the same executable semantic network architecture that serves as the enabling technology for the NLU capability to successfully compose complex avionics bus data from low level data observed by the aircraft's sensors.

Laydown Via Natural Language Orders

For the scenario laydown, we realized that reformatting an AFSIM laydown script into a natural language text file and using our new natural language orders file ingestion engine to read the file into our AI at runtime (example in Figure 7) was the quickest way to build the scenario. Once the AI ingests the scenario laydown file, it

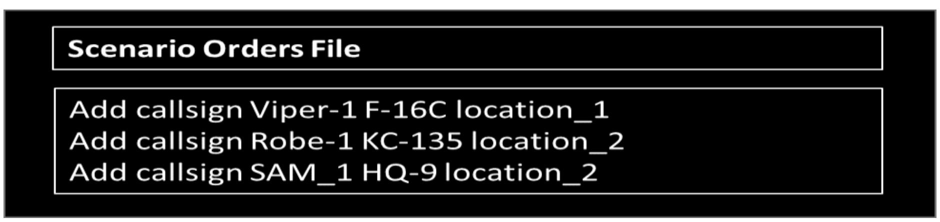

Figure 7. Example Scenario Via Natural Language Orders.

communicates to MACE via a network API to generate the scenario laydown. This is significant in that our AI can ingest scenario laydowns from one simulation application and then control the execution of the scenario in a different battlespace simulation application if desired. MACE has been the preferred battlespace simulation because it has a well-developed network API that allows for an external application to have full control of a mission. Network interfaces to other battlespace simulations that enable a similar level of control (as we have with MACE) are planned to be developed to give expand the interoperability of the hybrid AI.

Leveraging the NLU Architecture to Solve the Data Composition Problem

Since the goal for the study was to properly train the ML model for the avionics resource manager, a large amount of training data in the form of Open Mission Systems (OMS) Universal Command and Control Interface (UCI) data packets needed to be generated. This presented an interesting problem in that well-composed avionic bus data packets would be synthesized from raw electromagnetic data generated by the simulation environment. A solution was straightforward since the executable semantic networks that our AI is built upon can handle more than mapping the structured data from natural language understanding into actions. In particular, the agents can implement behaviors and compose information from data in memory. The resulting graphs use compose nodes to access deeply nested data from simulated electromagnetic emissions observed by aircraft sensors. This raw data is then used to construct more complex data blocks via a process of backwards chaining. The resulting fully formed UCI-OMS compliant data packets were then used to train the avionics resource ML model.

Key Findings

There were two interesting findings from this research effort:

1. An estimated 20 hours in MACE laydown time was saved because we could ingest a modified version of an AFSIM laydown instead of duplicating the entire laydown by hand in MACE.
2. The NLU capability was something more fundamental and foundational than first realized. The same underlaying executable semantic network architecture used for communication can solve other computer science problems, such as the data composition problem.

USE CASE 3: OFFENSIVE COUNTER AIR STUDY

In the previous use cases, it was learned that we could easily generate a scenario and command individual constructive pilots via a chat interface. However, it became apparent that the relatively low-level orders used to command individual platforms would not be suitable for the command of the numerous entities in more complex scenarios. This led to the idea of commanding groups of entities with high-level orders to reduce the number of orders required to create a scenario. This updated approach was key to successful completion of a project requiring a sizable multi-domain scenario consisting of 160 air, maritime and ground platforms to simulate an anti-access, area-denial fight.

Enabling High-Level Orders to Synthetic Warfighters and Platforms

To create smarter constructive agents that can be issued high-level orders in the form of mission goals, a knowledge database was built within the AI that captures tactics and doctrines. This maps tactics to scenario specific input criteria, such as mission orders, nationality, weapons loadout, and target type among others (see Figure 8). With this approach, the number of orders to create a scenario is reduced and a library of synthetic Warfighters is created. In fact, several of the platforms used in this study were reused in follow-on efforts where they were given orders to attack targets in different situations and each used the correct tactics.

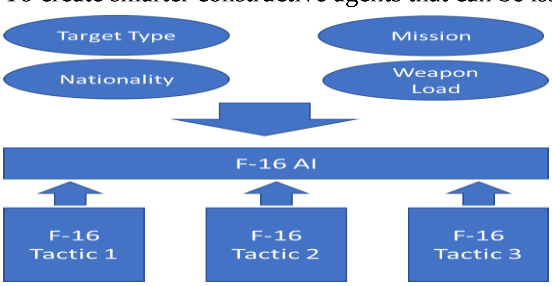

Figure 8. Simplified diagram showing how different input criteria influence which tactics the AI employs.

Commanding a Group of Synthetic Warfighters

By defining groups of Warfighters (and their platforms) and giving the group commander a callsign the number of orders required to obtain a desired mission outcome in a scenario was further reduced. Overall, by combining grouping with high-level orders, the number of orders needed to produce a given mission outcome was reduced by a factor of five. An orders file like the example shown in Figure 9 (the Gen 2 version) was created that puts orders on the group and not each individual entity.

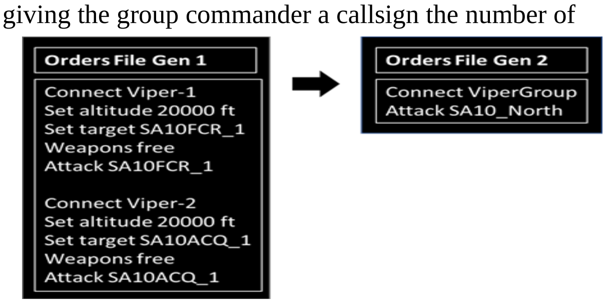

Figure 9. Comparison of low-level orders given to individual entities vs high-level orders to groups.

The creation of a database of predefined groups within the AI not only allows for smaller orders files, but for more efficient laydown of larger scenarios down the road. Scenario laydowns and their required entity behaviors can be constructed quickly via these prefabricated building blocks. For example, an entire fighter wing could be built up by simply defining new groups that are built from other smaller groups. It also serves as a convenient way to build command and control hierarchies within the AI that can be reused in future efforts.

Key Findings

There were three key findings in this study:

1. By issuing high-level orders to groups and group commanders, it was possible to reduce the number of orders by a factor of five. This approach allows for scenarios to be generated with less effort.
2. We found that the best way to command the intelligent synthetic agents is the same as how one would command real Warfighters – by assigning them high-level goals instead of low-level tasks. This brings us a step closer to the goal of creating anthropomorphic, intelligent synthetic agents that think and communicate like real Warfighters.
3. The grouping of Warfighters (and platforms) also enables the definition of command-and-control hierarchies within the groups.

FUTURE USE CASES AND POSSIBLE APPLICATIONS FOR INTELLIGENT AGENTS

We have shown three examples of how intelligent synthetic agents with NLU capability can overcome the challenges in creating realistic and complex battlespace simulations. For each project, key findings were provided as both lessons learned and to help shape follow-on research to build ever more capable intelligent synthetic agents that can communicate naturally. To further advance the state-of-the-art, research efforts should be pursued to develop the following capabilities:

1. Adding speech recognition capability to intelligent synthetic agents within battlespace simulation exercises.
2. Expanding high-level orders command set and additional "views." For example, we are developing a pilot "view" that allows them to communicate with constructive pilots using NATO brevity language.
3. Building command, control, and communications networks within the hybrid AI framework by using agent-to-agent communication that can be viewed as natural language. The aim is to improve complex battlespace simulations, such as VISTA or MACE, to allow for more realistic and cost-effective training.
4. Adding new interfaces to the hybrid AI to support additional battlespace simulations such as AFSIM and NGTS.
5. Further enhancing the hybrid AI framework through:
    a. Developing the capability to ingest facts to speed up the creation of large and complex executable semantic networks.
    b. Incorporating ML models in areas where the symbolic AI may not give the best solution. Ultimately, the goal is to create a neuro-symbolic AI that builds upon our current hybrid AI.
6. Automating laydowns through the creation of a synthetic lab assistant using the hybrid AI. This could drastically speed up scenario laydown in various ways such as creating an optimum IADS laydown based on the location of high value targets and automatic creation of realistic routes for strike missions.

A few examples of research areas and use cases that we believe could benefit from the research presented in this paper are:

1. Development and testing of collaborative AI for Warfighter-Machine-Teaming applications. AI that works alongside Warfighters will leverage NLU to communicate effectively with their human teammates (and their synthetic teammates as well). Next generation air platforms and systems that are optionally crewed and that work alongside with unmanned systems are of particular interest to us.
2. Reducing the use of Warfighters in exercises and training activities as a force multiplier to generate more complex simulated multi-domain battlespaces. Specifically, using agents with NLU and speech recognition that can serve as replacements for Warfighters and other roles to facilitate lengthy large scale training activities.
3. The creation of large and complex scenarios to aid in decision-making. Here a hybrid AI could generate optimal force allocations that are based on rules and constraints, but with an adaptive learning capability as well. The NLU capability would serve as a convenient way to communicate constraints and desired outcomes for the scenario.

CONCLUSION

This paper introduces the novel concept of issuing high-level, natural language orders to intelligent agents in a hybrid AI framework along with three applied use-cases highlighting its viability and desirability for the generation of high-fidelity battlespace simulations. Through our research, applied use cases, and overall key findings we hope to help the M&S community build a true digital twin of the battlespace that is realistic and scalable, as well as to reduce the time and effort required to create complex simulations. We believe the key to the realization of this hyper-realistic, digital battlespace is by using intelligent synthetic agents that are anthropomorphic. These agents will both realistically model the Warfighter in the simulations and will assist in their creation and analysis. We believe this will be transformative by:

1. Enhancing Warfighter readiness by allowing for the frequent holding of large-scale, MDO training events where intelligent agents take the place of human players (effectively creating a "Red Flag event in a box").
2. Speeding up M&S efforts through the generation of complex and realistic scenarios as well as aiding in the presenting and inferencing of simulation data.
3. Enabling of commanders and decision makers to make the right decisions based on results of battlespace simulations featuring realistic Warfighter modeling.
4. Accelerating the development, testing, and validation of agents that can work alongside Warfighters in a collaborative way to achieve the goal of true Warfighter-Machine-Teaming.

Finally, a main driving point throughout this paper was that our hybrid AI conceptually aligns with cognitive science and how the brain processes information. In the research presented in this paper, we have adopted Kahneman's decision making model to produce an AI framework that behaves in a manner consistent with how the Warfighter brain operates. It is with this type of AI and natural language understanding that we will eventually realize the vision of anthropomorphic intelligent synthetic agents to support, extend, and optimize high-fidelity battlespace simulations.

REFERENCES

Ananthaswamy, Anil. (2020). AI's Next Big Leap. Retrieved June 17, 2022, from https://knowablemagazine.org/article/technology/2020/what-is-neurosymbolic-ai

Battlespace Simulations, Inc. (2022). Modern Air Combat Environment. Retrieved June 17, 2022, from http://www.simlabs.in/pdf/MACE_Brochure.pdf.

Bocklisch, T., Faulkner J., Pawlowski, N. & Nichol, A. (2017). Rasa: Open-Source Language Understanding and Dialogue Management. *arXiv preprint arXiv:1712.05181* .

Harrison, J. A., Hobbs, G. R., Howes, J. R., & Cope, N. (1986). The use of speech technology in air traffic control simulators. In *IN: International Conference on Simulators* (pp. 15-19).

Kahneman, Daniel. (2011). *Thinking, Fast and Slow*. New York: Farrar, Straus & Giroux.

Lake, B. M., Ullman, T. D., Tenenbaum, J. B., & Gershman, S. J. (2017). Building machines that learn and think like people. *Behavioral and Brain Sciences, 40*.

Lin, Y. (2021). Spoken Instruction Understanding in Air Traffic Control: Challenge, Technique, and Application. *Aerospace*, *8*(3), 65.

Martz, J., Carpenter, W.B., Miller Jr., W.L. & Miller, S. J. (2009). Tactical Chat: Multi-Service Tactics, Techniques, and Procedures for Internet Tactical Chat in Support of Operations. Retrieved June 17, 2022, from https://www.alsa.mil/mttps/tacchat/

Naval Air Warfare Center Aircraft Division (NAWCAD) (2018). Next Generation Threat System. Retrieved June 17, 2022, from https://www.navair.navy.mil/nawctsd/sites/g/files/jejdrs596/files/2018-11/2018-ngts.pdf

Oijen, J. van. & Toubman, A. (2021). Are We Machine Learning Yet? Computer Generated Forces with Learning Capabilities in Military Simulation. *The Interservice/Industry Training, Simulation, and Education Conference (I/ITSEC)*, Orlando, FL, 2021.

Sowa, J. F. (Ed.). (2014). *Principles of semantic networks: Explorations in the representation of knowledge*. Morgan Kaufmann.

Zeh, J., Birkmire, B., Clive, P. D., Johnson, J., Krisby, A. W., Marjamaa, J. E., Miklos, L. B., Moss, M. J., & Yallaly, S. P. (2014). Advanced framework for simulation, integration, and modeling (AFSIM) version 1.8 overview (Technical Report 88ABW-2014-5041). *Wright Patterson Air Force Base, OH: Air Force Research Laboratory, Aerospace Systems*.