Automating Simulation-Based Air Traffic Control

Glenn Taylor
Soar Technology, Inc.
Ann Arbor, MI
glenn@soartech.com

MSG Jeff Miller (US Army, Ret.)
Navigator Development Group Inc.
Enterprise, AL
jeffrey.alan.miller@us.army.mil

Jeff Maddox
US Army AMRDEC
Huntsville, AL
jeffrey.a.maddox@us.army.mil

ABSTRACT

Air Traffic Control (ATC) receives little attention in simulation-based training and experimentation, in part because of the cost of including human operators to play ATC roles. Where ATC is used, it is typically very limited, reducing the realism of the experiment or training experience. This problem has become more apparent as UAVs and as joint battles are more often fought in simulation, requiring closer human management of the simulated airspace to coordinate air corridors, restricted airspaces, joint fire support, and the like. Furthermore, UAVs have become more prevalent in real battlefields, and the services are struggling with how to employ them safely and effectively within a broader air operations picture. Fighting ATC realistically in a simulated battlespace can help develop more realistic and appropriate employment tactics in the real battlespace. This paper describes the results of a Phase I SBIR investigating the feasibility of automating air traffic control (ATC) within simulation environments, for both experimentation and training. We leverage prior research analyzing human ATC tasks and situational awareness requirements in Tower, TRACON, and En Route operations, and describe how simulation environments can place different constraints and requirements on an ATC capability. We describe the use of human-driven ATC in recent joint experiments as a way to define some operational requirements of automated ATC. Key requirements include the ability to interact with both human pilots in virtual cockpits (using voice interaction), and with synthetic pilots and existing airspace management tools (using digital data links). We identify existing tools and technologies that can be used to fill these requirements, and where technology gaps still exist. Finally, we describe a cognitive systems approach to automating simulation-based ATC, and the development of a limited prototype that illustrates some of the key components of the architecture.

ABOUT THE AUTHORS

Glenn Taylor is a Research Scientist at Soar Technology, Inc. His research and development activities cover agent-based systems, human behavior modeling, intelligent user interfaces, and natural language systems. Recent efforts include: researching intelligent commander's assistants for C2; modeling an fixed-wing assets, rotary-wing assets, and artillery Fire Direction Center for a tactical air controller training system; building explanation systems for intelligent agents; developing speech interfaces to agents; and researching behavior design patterns for intelligent systems. He received a B.S. in Computer Science (1994), and an M.S. in Computer Science and Engineering (1996), both from the University of Michigan.

MSG Jeffrey A. Miller (US Army, Ret.) is a Senior Analyst at Navigator Development Group and has over twenty-one years of progressive operations and management experience in air traffic control, airspace management, air operations and resource management. His Army career culminated by serving as the Air Operations Training Committee (AOTC) Sergeant Major/Chief Instructor where he served as the lead SME for the procurement of the Enhanced Tower Operating System (ETOS) utilized to train students in the Army's ATC School. Mr. Miller is also a control tower operator certified by the Federal Aviation Administration (FAA) and holds a Master of Science in Aeronautical Science with a dual specialization in Aviation Management and Safety Systems from Embry Riddle Aeronautical University.

Jeffrey A. Maddox is a System Engineer for the US Army AMRDEC at Redstone Arsenal, Alabama, in the Advanced Prototyping, Engineering and eXperimentation (APEX) Lab. He earned a BS in Electrical Engineering from Auburn University and also holds a Math and Physics Education BS from Auburn University.

Report Documentation Page

Form Approved
OMB No. 0704-0188

1. REPORT DATE **2005**	2. REPORT TYPE	3. DATES COVERED **00-00-2005 to 00-00-2005**

4. TITLE AND SUBTITLE **Automating Simulation-Based Air Traffic Control**	5a. CONTRACT NUMBER
	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)	5d. PROJECT NUMBER
	5e. TASK NUMBER
	5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **Soar Technology, Inc,3600 Green Court, Suite 600,Ann Arbor,MI,48105**	8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR’S ACRONYM(S)
	11. SPONSOR/MONITOR’S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT

Approved for public release; distribution unlimited

13. SUPPLEMENTARY NOTES

Interservice/Industry Training, Simulation, and Education Conference (I/ITSEC), 2005. U.S. Government or Federal Rights License

14. ABSTRACT

Air Traffic Control (ATC) receives little attention in simulation-based training and experimentation, in part because of the cost of including human operators to play ATC roles. Where ATC is used, it is typically very limited, reducing the realism of the experiment or training experience. This problem has become more apparent as UAVs and as joint battles are more often fought in simulation, requiring closer human management of the simulated airspace to coordinate air corridors, restricted airspaces, joint fire support, and the like. Furthermore, UAVs have become more prevalent in real battlefields, and the services are struggling with how to employ them safely and effectively within a broader air operations picture. Fighting ATC realistically in a simulated battlespace can help develop more realistic and appropriate employment tactics in the real battlespace. This paper describes the results of a Phase I SBIR investigating the feasibility of automating air traffic control (ATC) within simulation environments, for both experimentation and training. We leverage prior research analyzing human ATC tasks and situational awareness requirements in Tower, TRACON, and En Route operations, and describe how simulation environments can place different constraints and requirements on an ATC capability. We describe the use of human-driven ATC in recent joint experiments as a way to define some operational requirements of automated ATC. Key requirements include the ability to interact with both human pilots in virtual cockpits (using voice interaction), and with synthetic pilots and existing airspace management tools (using digital data links). We identify existing tools and technologies that can be used to fill these requirements, and where technology gaps still exist. Finally, we describe a cognitive systems approach to automating simulation-based ATC and the development of a limited prototype that illustrates some of the key components of the architecture.

15. SUBJECT TERMS

<table>
<tr><td colspan="3">16. SECURITY CLASSIFICATION OF:</td><td rowspan="2">17. LIMITATION OF ABSTRACT
Same as Report (SAR)</td><td rowspan="2">18. NUMBER OF PAGES
11</td><td rowspan="2">19a. NAME OF RESPONSIBLE PERSON</td></tr>
<tr><td>a. REPORT
unclassified</td><td>b. ABSTRACT
unclassified</td><td>c. THIS PAGE
unclassified</td></tr>
</table>

Automating Simulation-Based Air Traffic Control

Glenn Taylor
Soar Technology, Inc.
Ann Arbor, MI
glenn@soartech.com

MSG Jeff Miller (US Army, Ret.)
Navigator Development Group Inc.
Enterprise, AL
jeffrey.alan.miller@us.army.mil

Jeff Maddox
US Army AMRDEC
Huntsville, AL
jeffrey.a.maddox@us.army.mil

INTRODUCTION

Air Traffic Control (ATC) is one area of military simulation that has not yet become fully automated, and is typically either left out of simulation entirely, reducing the simulation's realism, or is performed by a human operator, increasing its cost. Just as with the motto *Train as you fight*, experimentation must also place emphasis on creating realistic environments within which to experiment. Without such realism, the results may be incorrect or misleading, or may miss important issues that would otherwise be apparent in an operational environment. ATC is one of those aspects often left to the periphery. Automating ATC can address both the realism and cost aspects of playing ATC in simulation environments, thus improving overall effectiveness.

The need for a robust, automated ATC capability in simulation is manifold:

1) In experimentation, where either human role-players are used to play aspects of ATC without the benefit of automation, or ATC is left out entirely. Since aspects of ATC are played at most echelons, the need for automating ATC is broad across levels of experimentation, across the services, and especially in joint environments.
2) In simulation-based controller training or in sustainment training, where training is led by a human trainer, there are few automated decision support or intelligent tutoring tools to give the trainee opportunities to learn outside the classroom
3) In Army aviator training, where training aviators do not train for ATC in simulators (or in the classroom) until they find themselves in the terminal area, which can lead to negative skill transfer when in the air

This work focuses on simulation-based experimentation, but a robust solution would provide a good basis for the other two areas. This paper presents some background of air traffic control to motivate the requirements for what automating air traffic control in simulation would need to accomplish, discusses prior related work and analyses, then presents our approach to automation. Finally, we concluded with a discussion of a simple prototype to exercise some of our ideas.

BACKGROUND

In order to frame the problem and solution, we describe Air Traffic Control as happens in the Army, and automation that is currently used in ATC, for the Army and elsewhere.

Army Air Traffic Control

Army Air Traffic Control is placed within the more expansive Army Airspace Command and Control (A2C2), which provides a framework for managing Army air operations within broader Army operations (air and ground), other services, and other coalition members in a joint battlespace environment. ATC one component of A2C2. (Note that in the Army, ATC is known as Air Traffic Services (ATS). We will use ATC throughout this document.)

According to Army FM 3-52 Army Airspace Command and Control:

> Air traffic control is the use of active and passive measures to identify, locate, and regulate aircraft operating in the airspace control area. Regulating air traffic promotes air safety, facilitates identification of aerial platforms, and contributes to optimizing air defense assets. Air traffic control includes terminal procedures that focus on controlling aerial platforms at a specific landing or takeoff site, as well as en route procedures.

Army Air Traffic Control focuses on three types of operations: deep, close, and rear. The operations differ in the types of missions performed and the equipment available to the ATC to help in the task. For example, air traffic controllers assigned to airfields with immobile ATC towers and radar capabilities are likely to be employed only in areas far from the front lines; smaller mobile tactical teams, some with only radios and visual capabilities, are more likely to be employed in forward positions. These mobile ATC facilities are given missions in support of their assigned unit, and move as the battle moves.

There are two typical modes of control interaction between a controller and a pilot: en route control and terminal control. In en route operations, the controller typically receives a call from the aircraft, including information such as the aircraft identification, type of aircraft, location, and pilot intentions. Under normal conditions, the controller would simply allow the aircraft to pass through his or her Area of Operations (AO), and hand the aircraft off to the controller in the next ATC facility in the next AO. Where the aircraft needs adjustment, the controller would direct the aircraft to adjust its profile away from a potential conflict, an occurring conflict or an airspace violation. In terminal control, the controller is directing the aircraft to land at a heliport or airfield. In this mode, the task is one of scheduling the aircraft to land in some priority order, bringing the aircraft in from entry altitude down to the ground in a controlled fashion. Here, resources (airspace, runways, taxiways, ramps, zones) must be managed to keep aircraft smoothly moving in and out of the zone.

There are two primary means for controlling aircraft in a battlespace, positive control and procedural control. Positive control uses electronic means such as radar and other sensors to positively identify, track, and control aircraft within the airspace control area. Procedural control relies upon communicated orders and procedures to control how aircraft behave within the airspace control area. Tower environments and forward-placed controllers utilize primarily positive control to manage aircraft. Flight-following capabilities, and other en route operations, typically use only procedural control. The availability of airspace control facilities determines the method of control. Any tactical situation demands a combination of the two methods. In all cases, the controller must have a good understanding of the situation and doctrinal knowledge to guide the aircraft in any arena.

Along with aircraft in the environment, the controller may also communicate with other elements in the battlespace. Coordination with other facilities is necessary to request airspace clearances, to provide inbound and outbound information with other controllers, changes to routes and corridors, flight data, weather, etc.

Existing Automation Systems

Like many other areas in aviation, real air traffic control enjoys the benefits of some automation. Tools such as the Enhanced Traffic Management System (ETMS) and the Automated Radar Terminal System (ARTS) have been developed to automate many of the simpler tasks of ATC, allowing the human operator to focus on the more cognitively intensive tasks such as planning and interacting with pilots and other controllers.

The Army shares many of the technologies in ATC automation with other services and the commercial world. Radar and terminal display technology has improved Army ATC operations in stationary environments, such as at fixed bases at Division level and above. The Army has also funded its own technology efforts that have or are expected to improve its ATC capabilities, such as with the Tactical Airspace Integration System (TAIS) and Blue Force Tracker (BFT), which are meant to fit into the network operating systems the entire military infrastructure is moving toward, such as the Global Information Grid (GIG). The Army still faces a host of problems, many of which are unique to the Army. One issue is that, because most of Army Aviation operations occur below 3000 feet, there is ever increasing amounts of traffic with the increased prevalence of UAVs. A Platoon leader can insert a UAV at any time, and often does so without checking in with a controller or with aircraft in the area. Recent episodes in Iraq have pointed to the dangers of this, where in at least one case a small UAV collided with a helicopter because of lack of coordination, causing extensive damage to the helicopter, and in three other cases hazard reports were filed for near misses. As UAVs mature and begin to use the same refueling points as human-piloted aircraft, there will be an ever increasing burden on the human controllers in simulation, and in the field.

Fully automating the ATC capability requires the development of a system that can include these higher cognitive abilities as well. Research in cognitive systems over the past decade has produced tools that can be used to address these higher-cognitive capabilities. In the development and deployment of such systems, it is critical to capture and produce the behavior of ATCs in ways that human controllers and other participants in the environment find believable. Advancements in planning, plan recognition, learning, and natural interaction with humans place cognitive systems at the forefront of tools for modeling human capabilities. Furthermore, because an automated ATC capability must fit within a larger system that includes human pilots, trainees and operators, the ATC system's behavior must be understandable by the participants in the simulation. An automated ATC capability needs not only the ability to perform the ATC tasks in a doctrinally correct manner, but also the ability to explain its decision-making processes and results. This includes inspectable and traceable decision-making, and interactive debrief capabilities. Without the ability to explain its own behavior, human participants are less likely to trust the system, and so are less likely to use it.

ANALYSIS OF AIR TRAFFIC CONTROL

In our research, we extended prior analyses of ATC from a task analysis perspective, and conducted language and conversation analysis on pilot-controller interactions. This section details these findings.

Task Analysis

There has been a great deal of attention paid to Air Traffic Control, especially with respect to the effects of automation on human controllers. (Wickens, Mavor, Parasuraman, & McGee, 1998) describe the human factors issues associated with introducing automation within ATC operations. At the highest level, all controllers share three goals described by the motto safe, expeditious, orderly flow of traffic. Safety is the most important goal, where expeditious and orderly may vary based on the situation and the experience of the controller. Anecdotal evidence from expert controllers suggests that the importance of the expeditious flow goal and order flow goal will vary based on experience, with experts preferring expeditious over orderly, and novices preferring the opposite.

There have also been a number of prior studies with the purpose of analyzing the human tasks associated with ATC, in its various roles. Here, we review the work of several researchers who have contributed the most toward analyzing the ATC task. It should be noted that there is a great deal of overlap in the content of these evaluations. However, (Endsley & Rogers, 1994, amongh others) identifies not only the tasks, but also the kinds of information required to perform the task, which is critical in being able to build an automated model of task performance.

(Wickens et al., 1998) identify six major tasks performed by a controller, on a spectrum with smaller numbers indicating low-cognitive tasks, and larger numbers indicating high-cognitive tasks. Typically, it has been the low-cognitive tasks that have been automated in ATC environments.

1) Identifying relevant items of information – identify aircraft air speed and ground speed; identify aircraft type/designation; identify aircraft position (altitude, plan position); identify navigation fixes; identify weather features; etc.
2) Remembering – remember history of aircraft position; remember flight plans and updates; record conflict situations; remember non-controlled objects; remember clearances; etc.
3) Transmitting information; receive clearance requests and generate clearances; receive/send traffic management restrictions; receive flight plan information; input/send flight plan information; instruct pilots (heading, speed, altitude); instruct pilots (flight paths); receive/send conflict information; inform pilots of unsafe flying conditions; update flight plan information; receive/send handoff; etc.
4) Comparing criteria and predicting short-term events – determine violation of separation standards; determine violation of conformance criteria; determine deviation; determine equipment and system problems; compare reported versus actual position of aircraft; etc.
5) Predicting long-term events – predict violation of separation standards; predict aircraft trajectory; predict aircraft heading and speed; predict aircraft position; predict traffic sequences for arrival/departure flows; predict clearance slots; etc.
6) Planning strategies and resolving conflicts – plan/resolve traffic management constraints; plan clearances; resolve tactical conflicts; resolve strategic conflicts; resolve consequences of deviation; plan departure and arrival flows; plan emergency response; etc.

Others, including (Klein, 2001) and (Kallus, Barbarina, & Van Damme, 1997), offer similar high-level models of the ATC task which, at a high level of detail, is no different than most human problem solving processes. There is an expectation component to the problem solving, in which 1) prior models or expectations are brought to bear to identify and understand the situation, 2) an assessment is made, 3) actions may be taken, and 4) the effects of those actions are monitored and assessed, all in a large (sometimes parallel) performance loop.

Endsley in several studies (Endsley & Rodgers, 1994; Endsley & Jones, 1995; Endsley & Jones, 1996; Endsley & Smolensky, 1998) has been key in identifying the situational awareness requirements of ATC, especially of terminal radar approach control (TRACON) and en route control. These studies include a goal-directed task analysis (GDTA) (Endsley, 1993), which focuses on both the goals and the knowledge requirements for the identified tasks. Situational awareness according to Endsley (Endsley & Smolensky, 1998), refers to three levels: level 1 has to do with simple perception of the surrounding environment; level 2 covers the relating of the information in the environment to the actor's goals; level 3 covers the projection of activities into the future. We borrow from this work to define the requirements of automating air traffic control procedures.

The high-level situational awareness requirements across the different ATC roles are largely the same.

Differences become apparent in the level and type of control the controller has over the aircraft, and in knowledge about the specific locale in which the aircraft and controller are operating. Endsley's work identified a single overriding top-level goal, *assure flight safety*, and its immediate subgoals, *avoid collisions, provide flight services,* and *handle perturbations*. She continues to break these subgoals down to the questions that must be answered to meet the goal, and the high- and low-level knowledge required to answer the questions. Given the detail provided by Endsley, her work has been critical in designing an automated ATC system that is human-like and transparent in its behavior.

Language and Conversation Analysis

In our own work, we have collected pilot-controller conversations and have performed some analyses on those data, at the level of single utterances, and at the level of dialog exchanges. There are expected notions of formality in the exchanges between participants, including the doctrinal nature of the conversation and the well-established turn-taking that occurs in a dialog. ATC conversation is also marked by use of external references to presumably shared common knowledge, whether its background knowledge about artifacts (particular aircraft or mission types), mission information (route or point names), or situational/environmental aspects (names of reference points). In typical situations, very few grounding acts appear to take place, because of the assumption that everyone has this information. Pilots are expected to know the area they fly into, so that controller commands are unambiguous. In rare cases, however, the controller and the pilot may be expected to establish grounding, such as in bad weather conditions where some references are not visible, or where a pilot is disoriented or otherwise unfamiliar with the area.

One aspect of analyzing spoken language, and dialog in particular, is to examine what is being done with each utterance. In linguistic terms, the actions of an utterance are *dialog acts* that serve (at least) two purposes: performance of a task (*task management acts*) and management of the dialog (*dialog management acts*) (Harris, 2005). These aspects of an utterance are important from a hearer's perspective to help maintain the thread of conversation, to recognize intent of the speaker, and recognize when assumptions in the dialog no longer hold and need to be repaired. From a speaker's perspective, the utterance serves to move a task forward, but also to manage rules of the dialog (such as turn-taking) and establish grounding when needed. Table 1 illustrates some of our analysis of ATC dialog along these dimensions. As can be seen in this small snippet, a large component of the language content is situational awareness, and request/permission exchanges, with many references to physical or environmental elements – reference points, landing zones, etc.

Table 1: Conversation Analysis of a sample pilot-controller dialog

Turn	Speaker	Utterance	Dialog Mgmt Act	Task Mgmt Act
T11	Eagle6 (pilot)	Eagle Tower, Eagle 6, holding short for departure	Flow-regulating: initiate-dialog, take-turn, release turn	Assertive: introduce Directive: request
T12	Eagle Tower (controller)	Eagle 6, Eagle Tower, No delay on the runway, traffic, CH-47 on final approach inbound for landing, wind calm, cleared for takeoff, report frequency change	Flow-regulating: take-turn, assign-turn	Assertive: describe Directive: request Declarative: authorize
T13	Eagle6	Eagle Tower, Eagle 6, Roger on the go	Flow-regulating: take-turn, terminate-exchange	Commissive: accept-to (takeoff, report)
T14	Eagle6	Eagle Tower, Eagle 6, frequency change ACP 1	Flow-regulating: initiate-dialog, take-turn, assign-turn	Declarative: announce Directive: request
T15	Eagle Tower	Eagle 6, Eagle Tower, Frequency change approved	Flow-regulating: take-turn, terminate-dialog	Directives: approve

SIMULATION TECHNOLOGY SURVEY

This section gives a summary of military simulations, interoperability standards, and prior work in computational models that are relevant to this work.

ATC in Military Simulation

There are specialized simulations used for controller training, for example in TowerSim/ETOS, by Adacel, for training tower and ground controllers, and A2 Coach by UFA, for training radar controllers. There has been very little work including elements of ATC in simulation environments such as ModSAF or JSAF. Work funded by the Navy in the Battle Force Tactical Trainer, based in JSAF, was used in a way similar to TowerSim, where a human controller was directing computer-driven aircraft. In another case, also Navy work in JSAF, was a simple implementation of a controller built under the Navy's WARCON program, in which the computer-generated-forces (CGF)-based controller would attempt to recognize and resolve simple conflicts, and which would track scheduled departures and arrivals. By and large, where ATC has been played in simulation, humans have played those roles. Furthermore, there is little in the way of artifacts or objects in these simulations that pertain to ATC, such as runways and heliports, or signals such as inverted-Y's or lighted-T's. Obviously, these can be added either to standards such as HLA/DIS, or as part of the terrain databases. In such cases, at least some of the simulation systems would have to know how to interpret these artifacts, and they would need to be made available to the simulation participants, such as in rendering the artifacts to cockpit displays for human pilots or allowing CGFs to sense the presence of lights.

Communication and Interoperability

There are numerous relevant digital military protocols for this program. First, at the simulation level, network standards such as HLA or DIS, allow for federating together networks of simulations. There are also digital messaging formats used by the Army Battlefield Command System, such as US Message Text Format ting (USMTF) and Variable Message Formatting (VMF), that are designed to transmit messages between humans who are using these systems, and can include free text, voice recordings, etc. A now-defunct messaging format called the Command and Control Simulation Interface Language (CCSIL) was designed to be used as a strict way to communicate between computer systems, but was deemed too limiting, and has since been abandoned. The most relevant recent effort at creating a standard for mixed human/computer communications is the Command and Control Information Exchange Data Model (C2IEDM) effort sponsored by the Multilateral Interoperability Programme (NATO, 2005) and, separately, the Extensible Battle Management Language (XBML) (Turnitsa et al, 2004) effort sponsored by DMSO, both of which attempt to take doctrinal concepts from such sources as Army FM 101-5-1 Operational Terms and Graphics, and encode them into digital message concepts and formats to allow interoperability between human and computer systems. In simulation environments that attempt to mimic the operational Army battle command systems, Army Information Systems (INFOSYS) formats such as the TADILs become relevant as extant communication standards.

Computational Models of Controller Behavior

There have been a number of computational approaches to developing controller behavior. Note that there are other purely mathematical or planning models that deal solely with the planning and scheduling aspects of ATC; the agent models we examined represent a much fuller slice of the controller task (see, for example, (Harper, et al, 2002)), including interacting with other elements in the environment, so illustrate more of the issues associated with creating an automated ATC system. There are a few cases of building high-fidelity cognitive models that might be suitable for human factors analysis – see, for example, the Agent-based Modeling and Behavior Representation (AMBR) program including (Lebiere, Anderson, and Bothell, 2001; Chong & Wray, 2002). However, it should be noted that these models typically address very narrow aspects of the ATC task – such as learning simple responses to inputs, or simply computing new routes to avoid conflicts. One reason for this is the high cost associated with collecting data from human subjects, building and tuning the model, and testing against the data. To date, no single architecture has demonstrated the full range of capabilities of a human performing a highly complex task such as air traffic control with the fidelity of a human in terms of performance measures such as time on task, attention, mental workload, language understanding and generation, error rates, etc.

OBJECTIVE PARADIGM: KNOWLEDGE-RICH COGNITIVE SYSTEMS FOR AUTOMATED AIR TRAFFIC CONTROL

Given the high cognitive requirements of a human ATC, as detailed earlier (Wickens et al., 1998), it is only natural to look to cognitive systems as the foundation for an automated air traffic control system. A basic definition of cognitive systems is a class of systems that exhibit intelligent behavior across a wide range of problems and domains. Such behavior may include the ability to solve problems in different ways, learn from experience, and interact with other entities, including humans. The cognitive systems field is a varied one, with different researchers taking very different stances on the formal definition of what comprises a cognitive system. For instance, intelligent agents are often categorized with cognitive systems. (Wooldridge, 2000) presumes intelligent agents are autonomous, reactive, proactive, and social. The Beliefs-Desires-Intents paradigm (Bratman, 1987) extends this to a stronger view of agency, with a few more characteristics: knowledge and beliefs, desires and goals, intentions, obligations, and rationality.

Cognitive systems are often situated in complex environments and encounter many different problem-solving opportunities. In order to operate in these complex environments, the systems must use many different types of knowledge and reasoning, including: knowledge about goals and problem solving; about how to interact with the environment, other agents, and the user; knowledge about how to manage its own knowledge; how to recover from failure, etc. That is, we can describe cognitive systems as being knowledge rich. Furthermore, cognitive systems are often meant to interact intelligently with humans, which places different constraints on them than if they had to interact only with other computer systems. For this reason, capabilities such as self-explanation and communication are important in cognitive systems, which affords the system a level of transparency that encourages acceptance among human users. In these ways, cognitive systems can be distinguished from other approaches on the intelligent agent spectrum. Given the requirements in the proposed AutoATC, it is clear that a knowledge-rich, cognitive systems solution is required.

A strong view of cognitive systems is that they solve problems in human-like ways. One approach to developing these human-like cognitive systems is to build them using a cognitive architecture, which embodies a theory of human problem solving. As such, cognitive architectures provide an integrated approach to problem solving and reasoning in complex tasks. These systems also tend to provide a parsimonious framework for problem solving, such as processes for goal-directed behavior, planning, and belief maintenance. We believe it is necessary to build an automated ATC system on a unified framework that provides these capabilities, regardless of whether the final system is expected to perform as a high-fidelity human performance model of ATC.

APPROACH

Our approach to automating ATC within simulation is to develop a network Air Traffic Control appliance that can sit on the network, receive information over standard simulation network protocols (DIS, HLA) and using voice and data inputs from pilots, then reason about the situation and respond in a doctrinally correct manner (see Figure 1). With a network-appliance approach, the system should be able to be placed on any standard military simulation network, be told what

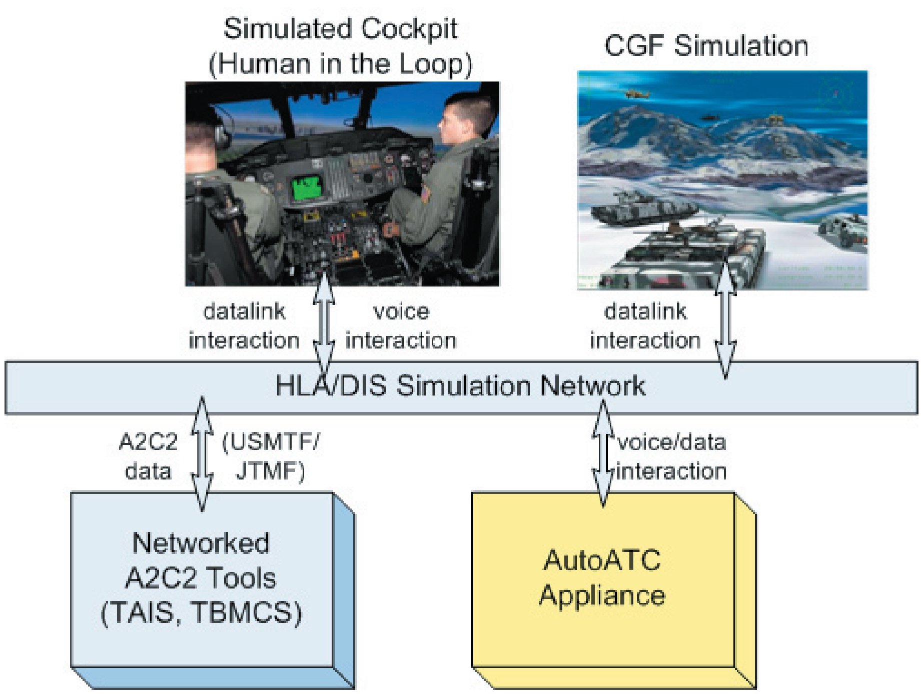

Figure 1: AutoATC Network Appliance Conceptual Diagram

its area of operations is, information about its roles and responsibilities and its current mission, and be unleashed to control aircraft in that airspace.

Preliminary AutoATC System Design

Based on our research and analysis of the tasks associated with ATC, and the requirements derived from simulation-based experimentation, we have developed an initial framework to account for the different tasks of a human ATC expected in the operating environment (see Figure 2). This section describes an architecture for an automated ATC appliance, which has led to a simple prototype to illustrate the concepts.

System Modules

There are a few primary modules within the system, covering behavioral, interoperability, knowledge management, and interaction aspects. Figure 2 illustrates a schematic of the preliminary architecture for AutoATC, which include the following modules:

- Behavior Engine – provides a parsimonious architecture for behavior representation and execution; based on earlier discussions, this would be have to be a knowledge-rich cognitive systems architecture capable of real-time behavior execution. The architecture here provides the primitives for behavior execution, and constraints on how they interact.
- Goal-Directed Problem-Solving Behavior Module – represents the strong view of agency to include knowledge and beliefs, desires and goals, intentions, obligations, and rationality to generate goal-directed behavior
- Situational Awareness Reasoning Module – explicitly represents and manage the system's awareness of environment in current and projected states, including models of other participants
- Task Knowledge Module – specific knowledge about how to perform ATC tasks across the different controller roles (tower, en route, approach/departure control) – these may be turned on or off, depending on the role(s) of the particular AutoATC agent within the simulation environment
- Communication Knowledge Module – manages high-level language understanding and generation, and dialog management
- Communication/Simulation Interface – manage low-level message, speech, and transport aspects of communication, as well as network simulation interactions

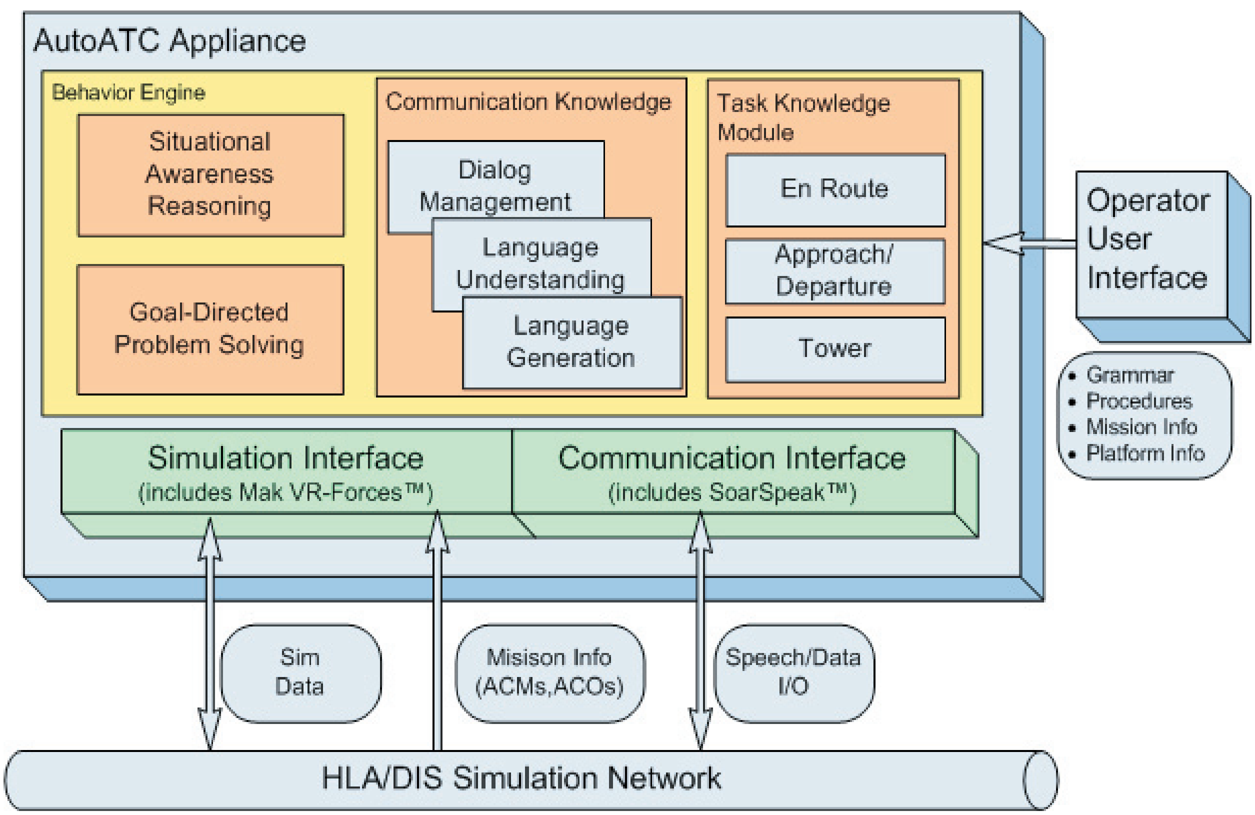

Figure 2: An initial architectural design for AutoATC

- System User Interface – used for tailoring system behavior to specific exercises or network environments and for visualizing the system behavior/performance

We feel these are necessary components for a network appliance approach to automating air traffic control in a simulation environment. Obviously, there are several technologies that might fit in some of the component modules, and this program has identified some existing technologies (such as given in earlier sections) that can play these roles. Our approach in these early phases of this research was to select well-fitting components and look at the integration of them into a prototype.

Prototype System and Results

To assess the feasibility of this approach, we have developed a limited prototype to explore some of the research and integration issues inherent in building such a system. Listed here are some extant tools we have pulled together and used as a baseline for new development required in building the ATC capabilities.

Agent Environment: Soar

For the prototype system, we are using the Soar agent architecture for its track record in CGF-like systems, and also because it was already integrated into the simulation engine used in the prototype. Soar, as a general cognitive architecture, lets us explore a wide range of approaches to problem solving, perception, situational awareness, and other tasks associated with air traffic control. The flexibility afforded by Soar was ideal in the early phases of research, when we could explore a few different solutions to a problem.

Simulation Environment: Mak VRForces

Mak's VR-Forces simulation environment is widely used throughout the industry, and provides a well-engineered basis for integrating CGFs. VR-Forces has built-in support for DIS- or HLA-compliant execution, so is well-suited as the basis for a network appliance approach to AutoATC.

Voice Interface: SoarSpeak

SoarSpeak is a generalized speech interface module that encapsulates multiple speech-to-text and text-to-speech engines transparently, allowing a developer to integrate speech into a CGF solution. Despite its name, SoarSpeak is not specific to Soar, and, furthermore, can be run as an HLA federate or DIS application, thus allowing it to be used in any standard military simulation environment for humans to communicate with CGFs. In the configuration used in our prototype here, we used Nuance for speech-to-text and AT&T Natural Voices for text-to-speech.

Agent Display: VISTA Situational Awareness Panel

In order to illustrate the agent's behavior, we used a tool called the Situational Awareness Panel (SAP) developed using the Visualization Toolkit for Agents (VISTA), a Java-based tool builder for visualizing agent behavior and awareness (Taylor, Jones, Goldstein, Frederiksen & Wray, 2002). Using a tool like this allows a user to get insight into the behavior of the agent, besides just the outward behavior of speech interaction. The SAP indicates such things as the agent's awareness of other entities, airspace control measures, current incoming and outgoing requests, the interaction history, etc.

New development to fill in the pieces included behaviors to cover the ATC tasks including communication and dialog management, grammars for the speech interface, and some domain-specific display elements for the agent display. The AutoATC agent has goals to maintain situational awareness and avoid conflicts. Maintaining situational awareness is performed passively, by receiving updates from the aircraft, and actively, by requesting information the agent does not have, such as requesting that the pilot call out when he reaches the next waypoint in his route. When potential conflicts arise, such as with known restricted operating zones or other aircraft operating in the vicinity, the controller issues advisories regarding those potential conflicts, and relies upon the pilot to maintain appropriate distance after the advisory is given. In this simple prototype, we did not give the system elaborate planning or scheduling capabilities to resolve complicated conflicts, and used a single active flight for controlling. Despite several simplicities, the system behavior was not scripted – all behavior was derived from a combination of the agent's goals and the current situation as the agent perceives it, and the dynamic interactions with the aircraft.

We developed an example scenario that spans terminal and en route control in rear and forward operations (see Figure 3). In the example, a human pilot's task (in a simulation environment) is to deliver a sling load and passengers from TAA Eagle to LZ Judy, ingressing on route Blue, and egressing on route Red. At TAA Eagle is a tower controller; along the routes is a en route controller; and at LZ Judy is another terminal controller. The pilot must interact with each during the mission, and switch between controlling agencies at required points during flight. In this prototype, the same agent plays the role of each of these controllers and simply simulates the handoffs. The human pilot begins interaction with the controller at TAA Eagle, as in the example given in Table 1. The pilot then changes frequency to interact with the agent playing the en route controller, then on to the Terminal Air Control Team (TACT) at LZ Judy, back

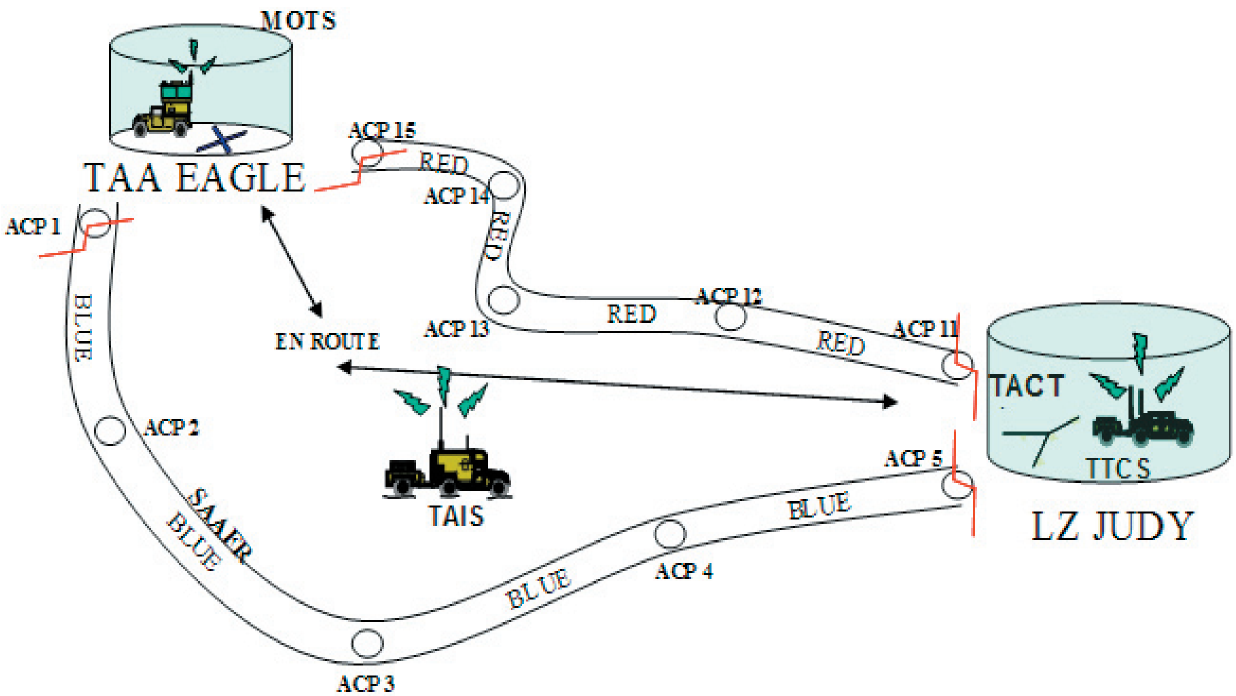

Figure 3: Example Scenario. The 17 Aviation Brigade is currently located in the vicinity of tactical assembly area (TAA) Eagle and is preparing to move to a forward operating base in the vicinity of LZ Judy. Bravo Company, 58th Air Traffic Control Company located at LZ Judy is establishing a Tactical Operations Center (TOC) and requires additional power and personnel to run the TOC. Echo Company, 1st Battalion, 17th Aviation Brigade will sling load one 30KW generator to LZ Judy and drop off support personnel to complete TOC operations and setup. Generator and support personnel to be on location LZ Judy no later than (NLT) 262200DEC07.

through the en route controller, then finally back to TAA Eagle and the Eagle Tower controller. All interactions are doctrinally correct within the narrow scope of the example. Figure 3 illustrates the basic scenario.

CONCLUSIONS AND FUTURE WORK

We have performed an initial assessment of the requirements for an automated ATC capability within simulation environments, and an assessment of the feasibility of developing such a system given existing technology. Where technology gaps exist, we have identified possible solutions for filling those gaps.

As part of assessing feasibility, even the simple prototype we developed indicates the scale of system integration required for implementing a network appliance approach to automating air traffic control in simulation environments. Though limited in several ways, the prototype demonstrates many of the key features required in a fully capable system – and proposes solutions to many of the issues identified in the initial stages of this work. Not surprisingly, one of the most limiting technologies is voice recognition and language understanding, which must improve greatly to facilitate more free-flowing interactions with humans. However, the generally constrained language and interaction protocols of ATC itself mitigates many of the issues found in other more fluid speech interaction domains.

Future steps in this endeavor include further analysis of the ATC task and interactions with other participants in the simulation, then developing a more complete system that can be validated and evaluated within a Joint simulation exercise. We will also explore the use of the system for training pilots and controllers.

ACKNOWLEDGEMENTS

This work was funded through a Department of the Army Small Business Innovation Research award, contract #W911W6-05-C-0035. We would like to thank the SMEs at Navigator Development Group for sharing their considerable expertise in Army and FAA air traffic control. We would also like to thank the research group at AMRDEC, especially Mr. Greg Tackett, and Dr. Nancy Bucher, for their encouragement and direction through this program.

REFERENCES

Bratman, M. (1987). *Intentions, Plans, and Practical Reason.* Cambridge, MA: Harvard University Press.

Chong, R. S., & Wray, R. E. (2002). *An EPIC-Soar model of concurrent performance on a category learning and a simplified air traffic control task.* Paper presented at the 20th Annual Conference of the Cognitive Science Society.

Endsley, M. R. (1993). *A survey of situation awareness requirements in air-to-air combat fighters.* International Journal of Aviation Psychology, 3(2), 157-168.

Endsley, M. R., & Jones, D. G. (1995). *Situation awareness requirements analysis for TRACON air traffic control (No. TTU-IE-95-01).* Lubbock, TX: Texas Tech University.

Endsley, M. R., & Jones, D. G. (1996). Situation awareness in air traffic control. Paper presented at the First International Conference on Applied Ergonomics, Instanbul, Turkey.

Endsley, M. R., & Rodgers, M. D. (1994). Situation awareness information requirements for en route air traffic control. Paper presented at the Human Factors and Ergonomics Society 38th Annual Meeting, Santa Monica, CA.

Endsley, M. R., & Smolensky, M. (1998). *Situation Awareness in Air Traffic Control: The Picture.* In S. E. (Ed.), Human Factors in Air Traffic Control. New York: Academic Press.

Harper, K., Guarino, S., White, A., Hanson, M. L., Bilimoria, K., & Mulfinger, D. (2002). *An Agent-Based Approach to Aircraft Conflict Resolution with Constraints.* Paper presented at the AIAA Guidance, Navigation, and Control, Monterey, CA.

Kallus, K. W., Barbarina, M., & Van Damme, D. (1997). *Model of the Cognitive Aspects of Air Traffic Control* (No. HUM.ET1.ST01.1000-Rep-02). Brüssel: EUROCONTROL.

Klein, G. (2001). *The Human Air Traffic Management Role in a Highly Automated Air Traffic System.* (Report) McLean, VA: MITRE.

Lebiere, C., Anderson, J. R., & Bothell, D. (2001). *Multi-tasking and cognitive workload in an ACT-R model of a simplified air traffic control task.* Paper presented at the Tenth Conference on Computere Generated Forces and Behavior Representation, Norfolk, VA.

NATO, M. I. P. (2005). *C2IEDM Specification,* from http://www.mip-site.org/

Taylor, G., Jones, R. M., Goldstein, M., Frederiksen, R., & Wray, R. E. (2002). *VISTA: A Generic Toolkit for Visualizing Agent Behavior.* Paper presented at the Eleventh Conference on Computer Generated Forces and Behavior Representation, Orlando, FL.

Turnitsa, C., Kovurri, S., Tolk, A., DeMasi, L., Dobbs, V., & Sudnikovich, W. (2004, September 2004). *Lessons Learned from C2IEDM Mappings within XBML.* Paper presented at the Fall Simulation Interoperability Workshop, Orlando, FL.

Wickens, C. D., Mavor, A. S., Parasuraman, R., & McGee, J. P. (Eds.). (1998). *The Future of Air Traffic Control: Human Operators and Automation.* Washington, DC: National Academies Press.

Wooldridge, M. (2000). Reasoning about Rational Agents. Boston, MA: MIT Press.