Maintaining Team Training Efficacy with Autonomous Synthetic Teammates

Dr. Christopher W. Myers, Dr. Jerry Ball
Air Force Research Laboratory
Wright-Patterson AFB, OH
christopher.myers.29@us.af.mil,
jerryandauroraball@yahoo.com

Dr. Nancy Cooke, Mr. Mustafa Demir, Dr. Nathan McNeese
Arizona State University
Tempe, AZ
nancy.cooke@asu.edu,
mdemir@asu.edu,
nmcneese@asu.edu

Dr. Michelle Caisse
L3 Link Simulation
Mesa, AZ
michelle.caisse@l-3com.com

Mrs. Mary Freiman
Aptima
Williamsburg, VA
mary.freiman@lumirresearch.com

Dr. Tim Halverson
ORISE at AFRL
Portland, OR
thalverson@gmail.com

ABSTRACT

Team training can be an expensive, time-consuming endeavor—especially in complex domains with large teams. A regularly taken approach has been to develop intelligent tutoring systems (ITS) that provide simulated over-the-shoulder instruction (Koedinger & Anderson, 1997; Vanlehn et al., 2005). However, team training costs increase with team member availability, when members are unavailable, subject matter experts must be recruited and paid to participate as confederates. The ITS approach does not directly address the issue of teammate availability, whereas synthetic teammates provide a solution to this challenge. Synthetic teammates have been promised to replace human confederates in training scenarios while maintaining training efficacy (Zachary, et al., 2001). Delivering on this promise, we present an Autonomous Synthetic Teammate (AST) that is capable of skillfully operating in heterogeneous teams that complete multiple reconnaissance missions within a Remotely Piloted Aircraft (RPA) platform. The AST is a high-cognitive-fidelity simulation of a human piloting a simulated RPA that can forget, remember, acquire knowledge, and communicate in ways that closely approximate humans. We describe the AST and provide results from a first-of-its-kind empirical evaluation in which half of the tested teams included the AST and the remaining half were all-human teams. Results show that AST inclusion did not hinder mission-level team performance. More importantly for training, the AST did not hinder the performance of its human teammates. Interestingly, teams with the AST may have changed team processes to compensate for some AST weaknesses. Nonetheless, we demonstrated the ability to replace a human confederate with an AST while maintaining training efficacy. We conclude with issues faced in developing ASTs and desiderata for facilitating their development, use, and maintenance. Further, we detail new research that takes the best parts of the AST for developing an agent capable of performing procedural control in Air Support Operation Center training exercises.

ABOUT THE AUTHORS

Dr. Christopher Myers is a research cognitive scientist in the Cognitive Models & Agents branch of the 711th Human Performance Wing within the Air Force Research Laboratory. Previously, Dr. Myers was a postdoctoral scholar with Dr. Nancy Cooke at ASU. Dr. Myers' research interests range from the development of Autonomous synthetic teammates to computational cognitive process models of decision-making and perception-action systems.

Dr. Jerry Ball is a retired research cognitive scientist from the Cognitive Models & Agents branch of the 711th Human Performance Wing within the Air Force Research Laboratory. Dr. Ball is interested in models of natural language processing and understanding.

Dr. Nancy Cooke Nancy J. Cooke is a professor of Cognitive Science and Engineering in the Polytechnic School, one of the Ira A. Fulton Schools of Engineering at Arizona State University and is Science Director of the Cognitive Engineering Research Institute in Mesa, AZ. Nancy Cooke's research interests include the study of individual and team cognition and its application to the development of cognitive and knowledge engineering methodologies, healthcare, homeland security systems, remotely-piloted aircraft, and emergency response systems.

Maintaining Team Training Efficacy with Autonomous Synthetic Teammates

Dr. Christopher W. Myers, Dr. Jerry Ball
Air Force Research Laboratory
Wright-Patterson AFB, OH
christopher.myers.29@us.af.mil,
jerryandauroraball@gmail.com

Dr. Nancy Cooke, Mr. Mustafa Demir, Dr. Nathan McNeese
Arizona State University
Tempe, AZ
nancy.cooke@asu.edu,
mdemir@asu.edu,
nmcneese@asu.edu

Dr. Michelle Caisse
L3 Link Simulation
Mesa, AZ
michelle.caisse@l-3com.com

Mrs. Mary Freiman
Aptima
Williamsburg, VA
mary.freiman@lumirresearch.com

Dr. Tim Halverson
ORISE at AFRL
Portland, OR
thalverson@gmail.com

INTRODUCTION

Training is an integral part of the human ability to systematically progress socially and technologically. The cognitive sciences have sought ways to improve training efficiency and efficacy. A regularly taken approach is to develop intelligent tutoring systems (ITS) that provide simulated over-the-shoulder instruction to students (Koedinger & Anderson, 1997). Although much success has been achieved in ITS research on tutoring individuals in subjects like algebra (Heffernan, Koedinger, & Razzaq, 2008), Lisp programming (Anderson & Reiser, 1985), and physics (Vanlehn et al., 2005), few have attempted to train individuals operating specialized tasks within a team.

Training teams requires not only instructing individuals on their respective task(s), but also in how to interact with teammates. A further complication is that team training requires all team members to be available for training. In large teams, having all team members present for training may be intractable, at which point subject matter experts (SMEs) acting as confederates are brought in to "simulate" missing team members.

We report the benefits and detriments of including an autonomous synthetic teammate (AST) within team training scenarios as an artificially intelligent confederate. The empirical evaluation of an AST with naïve participants has never been reported, if ever conducted. Typically, they are developed and anecdotally evaluated by a subject matter expert, its developers, or its users. We go beyond this typical approach and situate the AST with naive human participants and compare team and individual performance to all-human teams. Results from the empirical evaluation demonstrate that ASTs can be used to replace team members and maintain team and individual training effectiveness. In the following sections we first describe the simulated task environment of interest—a remotely piloted aerial system (RPAS) comprised of three individuals operating as a team to complete multiple reconnaissance tasks. Next, we introduce the AST, providing details on its processes and mechanisms. Finally, we present the evaluative study's design, results, and provide conclusions on successes, failures, and future directions ASTs.

REMOTELY PILOTED AERIAL SYSTEM—SYNTHETIC TASK ENVIRONMENT

The RPAS-STE (Remotely Piloted Aerial System--Synthetic Task Environment) is a testbed for studying team cognition within a three-person heterogeneous team (Cooke & Shope, 2004). In this STE, three participants coordinate to "fly" a simulated RPAS to take reconnaissance photographs. The RPAS-STE task was modeled after team task components from the United States Air Force Predator ground control station (Cooke & Shope, 2004). In the RPAS-STE three individuals are assigned to the role of pilot, photographer, or navigator (see Figure 1). Individuals are first trained on the tasks specific to their roles and, after attaining proficiency in their individual roles, come together to work as a team to complete multiple 40-minute missions to photograph ground targets.

Each team member is seated in front of two monitors that display unique role information, as well as common vehicle information (heading, speed, altitude). Team member interaction occurs through text-based communications similar to instant messaging and email. A number of team and individual measures have been designed, validated, and embedded in the task software and collected apart from the task (Cooke & Gorman, 2009). To determine team

performance, a composite outcome score is computed for teams at the end of each 40-minute mission based on the number of successful photographs, route violations, and film used. Scores are also calculated for each team member. Additionally, individual- and team-level knowledge, team process, team situation awareness, and team coordination metrics are measured.

THE AUTONOMOUS SYNTHETIC TEAMMATE

We selected the RPAS pilot as the position for AST development. The development of our AST has been previously documented in the journal *Computational & Mathematical Organization Theory* (for significantly more detail and information on the development of the AST, please see Ball et al., 2010, and Rodgers, Myers, Ball, and Freiman, 2013). The AST was implemented within the ACT-R cognitive architecture (Anderson, 2007)—a framework instantiated in computer software for developing models of human cognitive processes and actions that closely approximate human behavior, down below the half-second timescale (i.e., *high-cognitive-fidelity*).

There were two important reasons for a focus on high-cognitive-fidelity. First, for inherently human behaviors (e.g., language comprehension and generation) the use of a high-cognitive-fidelity cognitive architecture to guide and constrain the implementation of a system may actually facilitate development rather than hinder it (Ball, 2006). The constraints imposed by the cognitive architecture push system development in cognitively plausible directions which are assumed more likely to lead to human-like behavior than purely algorithmic solutions that ignore such constraints.

Second, an AST should be of high-cognitive-fidelity for robust team training. If one trains with an AST that performs perfectly with no delay, then the trainee will learn to expect perfection when operating in the field. Alternatively, if one trains with an AST that fails and succeeds in ways and durations similar to humans, then the trainee will be prepared when faced with human-related failures in the operational environment. Adhering to high-cognitive-fidelity led to the AST becoming one of, if not the, largest cognitive models ever built. Further, it operates over a timescale atypical for computational cognitive process models, performing multiple 40-minute missions.

The major components of the AST include the *natural language analysis* component, the *situation representation* component, the *agent-environment interaction* component, and the *dialog management* and *language generation* components (see Figure 2). The language analysis component interacts with a situation representation component that contains spatial-imaginal/propositional representations of the current state of affairs encoded from reading text chat messages and interacting with the task environment. The situation representation component is intended to be a computational implementation of theoretical perspectives on situation models (Zwaan & Radvansky, 1998) and situation awareness (Endsley, 1995). The dialog management and language generations components interact with the situation representation to determine when to say what, and to whom. The agent-environment interaction component implements the 'observable behaviors' of the system, controlling shifts of visual attention and motor actions needed to communicate with teammates and to pilot the RPAS. Input to the AST is mediated by ACT-Rs visual perception module and motor actions are mediated by ACT-R's motor module.

Each component of the AST makes use of the same declarative memory (DM) and production system provided by ACT-R. Consequently, there is a central processing bottleneck, the production system, as each component needs it to process information. This is managed with a simple obtain-relinquish control mechanism. For example, if the language analysis component is reading and processing text, then the agent-environment interaction component cannot process any information, and vice-versa. When a component completes processing, then it relinquishes control of the system by returning it to an 'empty' goal state. At this point, the other components that can execute given the current situation are free to obtain control and carry on processing.

The situation representation component did not strictly use the obtain-relinquish process control mechanism. The situation representation component was developed to maintain information derived from processing within each of the components. In an effort to have a more thorough, complete situation representation, rules from the situation component could execute at any time, independent of which component was in control of information processing. To summarize, the AST is composed of five major components, all developed in the ACT-R cognitive architecture. They share DM and production system resources, and manage the production system using an obtain-relinquish control mechanism. In the following section, we provide more details for each of the five components.

The Language Analysis Component
The language analysis component is intended to be a domain general system capable of handling a broad range of English constructions. It is a construction-driven processing system (Ball, 2007b) based on a linguistic theory of the grammatical encoding of referential and relational meaning (Ball, 2007a) which is aligned with basic principles of Cognitive and Construction Grammar (c.f., Langacker, 1987, 1991). Lexical items in the linguistic input activate constructions that drive processing.

The component adheres to two well-established cognitive constraints on language processing–incremental and interactive processing (c.f., Gibson & Pearlmutter, 1998). The component processes the input incrementally (one word at time), constructing a linguistic representation of the input based on the current word, constructions activated by the word, and the prior context. If necessary, the current input is accommodated by adjusting the current representation or coercing the current input into that representation without backtracking or look-ahead. The mechanism of context accommodation is part and parcel of the basic left-to-right, incremental processing mechanism.

The Situation Representation Component
The situation representation is the component which functions as the primary interface between the other AST components, and provides the primary meaning representation and inference capability of the overall system. Currently, the AST's situation component is limited to linguistic representations and some task situations. The concept of a situation model originates in the psycholinguistic research of van Dijk and Kintsch (1983), who describe a situation model as a mental representation of the propositional content of a text—including the addition of propositions corresponding to inferences that are derived from the text.

The situation representation is operationally defined as a set of objects, actions, events, and relationships associated with a task that is sufficient for reasoning about the agent's potential set of actions. The situation representation is separate from the AST's world knowledge but is related to, and affected by, world knowledge. Hence, the situation representation can be thought of as the AST's mental model of the RPAS situation, including the RPAS, its flight parameters, its location relative to waypoints, the waypoints, etc. The AST's situation representation reflects the dynamically integrated input from not only linguistic and discourse sources, but also task processes and the model's knowledge. This illustrates the integration of information from a variety of sources consistent with Baddeley's (2003) episodic memory proposal, and provided an integration of linguistic situation models with the psychological construct of situation awareness (Endsley, 1995; 2015). For a detailed discussion on the design goals of the situation model in the context of AST research and its implementation in the ACT-R cognitive architecture, please see Rodgers et al. (2013).

The Agent-Environment Interaction Component
The agent-environment interaction component was developed to fly the RPAS from waypoint to waypoint in a manner similar to a human pilot. Flying to waypoints involves interacting with the RPAS-STE to queue the correct waypoint and enter the correct course (see Figure 3). The pilot must also set the airspeed and altitude within restrictions provided by the photographer and navigator. The agent-environment interaction component interacts with the RPAS-STE using the same device as humans–it uses the mouse pointer to interact with the RPAS flight controls in a point-and-click fashion, and uses the keyboard to send and receive messages to and from its teammates.

There is enormous complexity in developing computational process models that interact with graphical user interfaces (GUIs). This complexity results from the need to model mouse pointer movement, attention selection and eye movement times, etc. The use of the ACT-R architecture made it tractable to develop a model capable of interacting with the RPAS-STE GUI, as it provided built-in functionality for modeling shifts of visual attention, mouse pointer movements, button clicks, etc. The architecture uses Fitts' Law for determining mouse pointer and attention shift movement times. In the following sections an overview of the piloting task is provided, and is followed by brief descriptions of modeled strategies used by the AST.

To fly the RPAS, the pilot must complete six goals: 1) set the airspeed, 2) altitude, 3) course, 4) waypoint, and 5) send and 6) receive text messages. Of these six, only details setting the waypoint is covered for the purpose of brevity. To maneuver the RPAS from one waypoint to another, the pilot uses a point-and-click interface to enter settings (see Figure 3). To set the waypoint, the pilot toggles through a list of 109 alphabetically organized waypoints by pressing the setting adjustment buttons (see Figure 3). Each time an adjustment button is pressed, a

new waypoint value is queued (e.g., BEP in Figure 3). The waypoint list operates as a continuous loop so that the letter A comes after the letter Z.

When setting a waypoint, the AVO can either advance (+) or retreat (–) through the list of waypoints, one at a time. It was assumed that participants come to the task with extensive knowledge and experience in the English alphabet, and that the choice to advance or retreat through waypoints results from bringing the alphabet knowledge to bear on the waypoint setting goal. Thus, the Klahr et al. (1983) model was used as a representation of the English alphabet for comparing letters.

In the Klahr et al. (1983) model of letter retrieval and comparison, letters were stored as hierarchical subgroups in a link-node structure (e.g., α to τ in Figure 4). Letters within a node (e.g., D in node α) can only be reached through node entry points. Entry points for each node are the first letter of the node (e.g., A for α, see Figure 4). Node contents are based on empirical evidence of entry point consistency with phrasing in "Twinkle, Twinkle Little Star," used to teach the alphabet.

Klahr et al. validated their model with response time data collected from human participants that were shown letters of the alphabet and asked to respond with the name of the letter that either occurs before or after the probe letter. However, determining whether to advance or retreat through the waypoint list in the UAV-STE is quite different. Rather than responding with an adjacent letter, the model must determine whether the desired waypoint (e.g., BEO) occurs before or after the queued waypoint (BEP in Figure 3). This requires determining if a letter occurs before or after another letter in the alphabet, and these comparisons can occur between and within letter nodes.

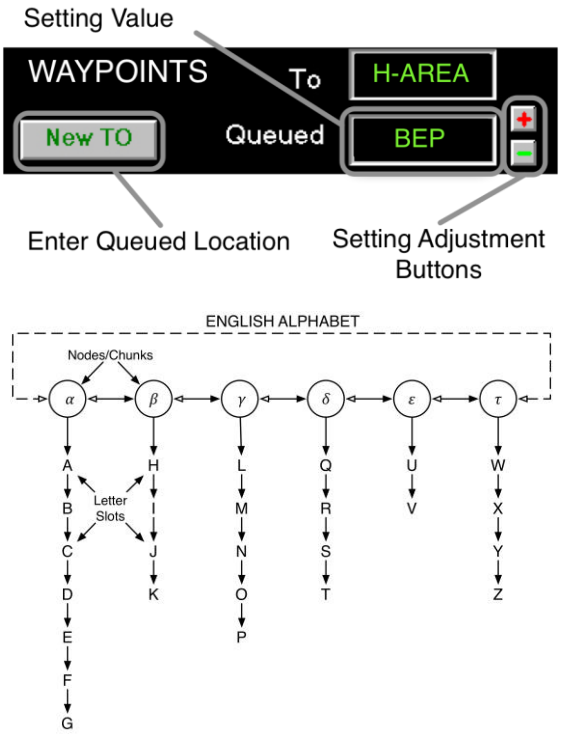


Figure 1. Alphabet representation adapted from Klahr, Chase, and Lovelace (1983). Dashed lines and open arrows represent capabilities added to their model.

The English alphabet was divided into six letter nodes called *alpha-chunks* (see Figure 4) that contained letters and were stored as items in the AST's declarative memory. A two-step process was developed to select the appropriate setting adjustment button. The process began by comparing the first letter of each waypoint name. If they were equal, subsequent letters were compared until two were different (e.g., O and P from the desired waypoint 'BEO' and the queued waypoint 'BEP'). At this point the second step began. The second step involved retrieving alpha-chunks from declarative memory for each of the different letters for comparison (in our example letters O and P) and making comparisons across the chunks. Thus, alpha-chunks were retrieved independently, without the need to serially traverse the alpha-chunks/nodes until the desired alphabet-chunk was reached. This non-serial retrieval of alphabet-chunks differs from the Klahr et al. (1983) model, and allows traversing the alphabet nodes in either direction. The Klahr et al. (1983) model provided an excellent approach to representing the English alphabet in the agent-environment interaction component (Myers, 2009). Once the pilot had queued the next waypoint to visit, s/he presses the 'New TO' button, changing 'H-AREA' to 'BEP' in Figure 3.

This walkthrough of how the agent-environment interaction component queues new waypoints demonstrates the level of detail most of the tasks were modeled. This level of detail provides a high-cognitive-fidelity model of completing RPAS tasks (Myers, 2009).

The Language Generation & Dialog Management Components

The language generation and dialog management components were developed to capture the dynamic nature of human language production, following earlier approaches involving dynamic dialog constraints (Ericsson, 2004), accommodation (Matessa, 2000), and adaptive content selection (Walker et al., 2004). The focus of the language

generation component was on selecting from a set of possible utterances, akin to overgeneration-and-ranking approaches (Varges, 2006). The focus of the dialog management component was the management of communication obligations (Traum & Allen, 1994), with messages abstracted as dialog acts (Core & Allen, 1997).

The language generation component used Optimality Theory (Prince & Smolensky, 1993) to select an optimal utterance, given a set of utterances and their constraints. Constraints are simple, violable, conflicting, and motivated by cross-linguistic evidence. Constraints are arranged in a strict dominance hierarchy; the optimal utterance is the one that least violates the hierarchy. Constraint ranking is expressed through ACT-R declarative memory activation: the most important constraint is most highly activated. Activation spreads from constraints to utterances to determine the utterance retrieved from memory; the most important constraint has the greatest effect on the retrieval. Factors from the situation model component dynamically affect the constraint ranking, providing a principled variation in utterances over time. Language generation is based on retrieval of complete messages with one or two variabilized slots. These message templates are akin to constructions.

The dialog management component managed the push and pull of information to and from the AST. Messages were abstracted as speech acts based on the DAMSL annotation scheme developed by Core and Allen (1997). For example, the message 'Can I proceed to the next waypoint?' is classified as a forward-looking check question. In normal conversation, there is a sense of obligation to follow rules of communication. For example, if a question is asked then that question should be answered, or at least addressed somehow. Some of these obligation rules were made explicit by Traum and Allen (1994) and were found to be useful in other dialog management systems (e.g., Matessa, 2000). Obligation rules facilitate effective discourse by setting up expectations for future messages. Messages with little local context information can then be understood because of the context from previous expectations. In the dialog management component, obligations were stored as declarative memory chunks in a specialized module with separate buffers for self and others. Dialog management productions create, release, and use obligations to fill in context. In addition to the obligation module, the dialog management component also used a temporal module extension to ACT-R to avoid repeatedly requesting or providing the same information.

EMPIRICALLY EVALUATING THE AUTONOMOUS SYNTHETIC TEAMMATE

The following analyses are from an AST evaluation study in the RPAS-STE. To determine if the AST could provide piloting skill and communication capabilities to facilitate performance at the team and individual levels of analysis, we manipulated team condition, in which there were three between-subjects conditions: Synthetic, Experimenter, & Control. The *Control* condition was an all-human team. The *Synthetic* condition had the AST performing as the RPAS-STE pilot. The *Experimenter* condition had an expert human serve as the RPAS-STE pilot; however, for the purposes of this paper, we leave the Experimenter condition in all analyses, but focus on any differences and similarities between the Control and Synthetic conditions.

Individuals were randomly assigned to form teams of three and then randomly assigned to each condition. Each team completed five unique missions, with the last mission being one of high workload (many more reconnaissance targets than missions 1-4). There were 10 teams per team condition. With respect to the AST, we were interested in three questions: (1) whether teams in the Synthetic condition demonstrated a performance increase across missions, (2) whether teams in the Synthetic condition reached a level of performance similar to the Control condition, and (3) if the performance of human teammates working with an AST differed from humans performing the same tasks in the Control condition.

There are a set of well-validated performance metrics for team performance and pilot, navigator, and photographer performance. Each performance score is the sum of a set of penalties subtracted from 1000. Consequently, we can evaluate performance differences between teams and individual positions at the aggregate and penalty level of analyses in order to have a thorough understanding of where participants are performing well and poorly and the effects the AST has on its teammates' performance.

Team Performance

The team performance score is composed of four sub-scores, each derived from penalties resulting from important team tasks defined by the domain. The sub-scores are Warning Duration Penalty, Alarm Duration Penalty, Critical Waypoints Visited Penalty, and Good Unique Photos Penalty. Each of these sub-scores are weighted to correspond with task importance. The overall team score is the sum of the four penalties subtracted from 1000.

To determine how teams differed in their performance, we performed a 3 (team condition) × 5 (missions) repeated measures Multivariate ANOVA on four team sub-scores (alarm, warning, the route sequence, and good unique photos penalties). Mauchly's test indicated that the assumption of sphericity was violated for mission, $\chi^2(9) = 22.110$, $p < .05$, $\varepsilon = .762$, team subscores, $\chi^2(5) = 51.696$, $p < .001$, $\varepsilon = .486$, and the mission–subscore interaction, χ^2 (77) = 376.01, $p < .001$, $\varepsilon = .286$. We used the Greenhouse-Geisser correction for degrees of freedom for each of the three within subject effects.

The mission-by-condition interaction was not significant $F(6.092, 82.246) = 1.884$, $p = .092$, demonstrating team conditions did not significantly differ with increasing task experience. The team condition main effect was statistically significant, $F(2,27) = 11.496$, $p < .001$, $\eta^2 = .460$, demonstrating a performance difference across the different team conditions (i.e., Synthetic, Experimenter, and Control).

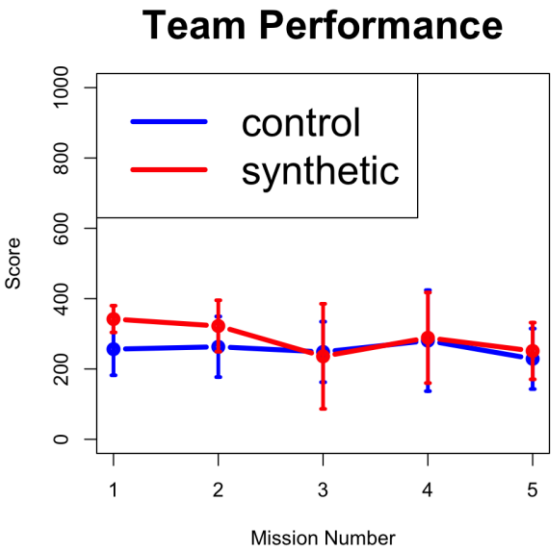


Figure 2. Team performance for Control and Synthetic conditions across missions.

To determine which team conditions differed to drive the main effect, we planned comparisons between each of the team conditions. The results of the pairwise dependent *t*–tests indicate that the Synthetic and Control teams were not significantly different ($M_{Synthetic} = 178.996$, $M_{Control} = 187.833$, $p = .286$; see Figure 4). There was also a significant main effect of mission, $F(3.046, 82.246) = 2.991$, $p < .05$, $\eta^2 = .100$, demonstrating overall team sub-scores increased from low-load (i.e., Mission 4) to high-load missions (i.e., Mission 5).

To summarize, Synthetic and Control conditions performed the same across missions. These results establish that teams performing with an AST perform at the same level as all-human teams, and demonstrate that supplanting SMEs with ASTs could maintain team performance during training scenarios. While this is a promising result, it does not demonstrate that the trainees operating with the AST (i.e. the navigator and photographer) maintain a level of proficiency compared with operating on a team with all human teammates. In the following sections we evaluate how well the navigator and photographer perform when working with an AST compared to working with a human pilot.

Navigator Performance

We begin with the overall navigator performance score to determine differences in performance across conditions and missions. We follow this analysis with a detailed analyses of each sub-score. The navigators' performance score is the summation of five sub-scores subtracted from 1000. The sub-scores are warning penalty, alarm penalty, the number critical waypoints visited, the number of planned waypoints, and route sequence violations. The following table demonstrate each of the navigator sub-score's calculations.

We conducted a 3 (team condition) × 5 (missions) repeated measures Multivariate ANOVA on the five navigator sub-scores. Mauchly's test of sphericity indicated that the assumption was not violated for mission, $\chi^2(9) = 15.52$, $p = .078$, $\varepsilon = .791$. However, it was violated for overall navigator sub-scores, $\chi^2(9) = 176.991$, $p < .001$, $\varepsilon = .417$, and interaction between mission and sub-scores, $\chi^2(135) = 770.535$, $p < .001$, $\varepsilon = .307$. Thus, the Greenhouse-Geisser correction for degrees of freedom was used. The three-way mission × sub-score × condition interaction was statistically significant $F(32, 432) = 2.465$, $p < .001$, the magnitude of which was consistent with a small effect size, $\eta^2 = .154$. The mission–team condition interaction was not significant, $F(8, 108) = 1.433$, $p = .191$. However, the sub-scores–condition interaction was statistically significant $F(8, 108) = 9.892$, $p < .001$, $\eta^2 = .423$, as was the sub-scores–mission interaction, $F(4.918, 132.796) = 13.147$, $p < .001$, $\eta^2 = .327$, driving the three-way interaction.

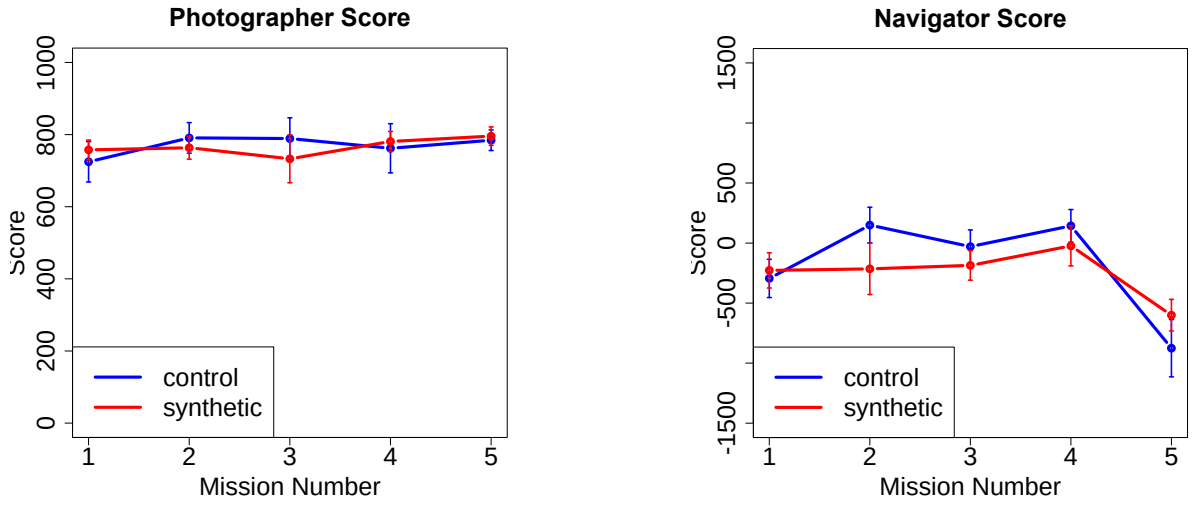
The team condition main effect was statistically significant, $F(2,27) = 7.441$, $p < .05$, ($\eta^2 = .355$), as was the mission main effect, $F(4, 108) = 25.614$, $p < .001$, $\eta^2 = .487$. The mission–condition interaction was not significant, $F(8, 108) = 1.433$, $p = .191$. Sub-scores interacted with team condition, $F(8, 108) = 9.892$, $p < .001$, $\eta^2 = .423$ and mission, $F(4.918, 132.79) = 13.147$, $p < .001$, $\eta^2 = .327$).

The navigation performance differed significantly across mission, $F(4, 108) = 25.614$, $p < .001$, $\eta^2 = .487$, and between team conditions, $F(2,27) = 7.441$, $p < .05$, $\eta^2 = .355$. Importantly, the results of the planned pairwise dependent t−tests between team conditions indicated that navigators in the Synthetic condition was not significantly different than navigators in the Control condition ($M_{Synthetic} = 250.124$, $M_{Control} = 236.258$, $p = .658$). However, the Synthetic condition did have a greater overall penalty relative to the Experimenter condition ($M_{Exp} = 140.514$, $p < .001$). Further, the navigators working with an expert pilot in the Experimenter condition had significantly less overall penalty than the navigators in the Control ($M_{Exp} = 140.514$, $M_{Control} = 236.258$, $p < .05$).

We compared each condition across each navigator sub-score. There was statistically significant difference between Synthetic and Control conditions for the alarm penalty ($M_{Synthetic} = 1.421$, $M_{Control} = 78.058$, $p < .001$). The Synthetic and Control conditions exhibited no difference in warning penalty. The pairwise comparisons for the critical waypoints penalty indicated that there was a marginal difference between the Synthetic and Control conditions ($M_{Synthetic} = 904.112$, $M_{Control} = 681.961$, $p = .066$). There were no differences between the Synthetic and Control conditions for the planned waypoint penalty and route sequence violation penalty (see top of Figure 5).

Photographer Performance

Similar to the navigator performance, the photographer performance was compared between those who had the AST as a teammate and those who did not. The photographers' performance score is the summation of five sub-scores subtracted from 1000. The photographer sub-scores are warning penalty, alarm penalty, the number of good photos taken, the number of bad photos taken, and the number of good unique photos taken. Like the navigator analyses, we begin with the performance score to determine differences in overall photographer performance across conditions and missions. We follow this analysis with detailed analyses of each sub-score.



We conducted a 3 (conditions) by 5 (missions) repeated measures Multivariate ANOVA on five photographer sub-

Figure 3. Navigator & Photographer performance scores and penalties for the control and synthetic conditions.

scores. Mauchly's test of sphericity indicated that the assumption was not violated for mission, $\chi^2(9) = 14.819$, $p = .097$, $\varepsilon = .801$. However the sphericity assumption was violated for overall photographer sub-scores, $\chi^2(9) = 137.239$, $p < .001$, $\varepsilon = .325$, and the interaction between mission and overall photographer sub-scores, $\chi^2(135) = 639.395$, $p < .001$, $\varepsilon = .272$. Consequently, the Greenhouse-Geisser correction for degrees of freedom was used. Critically, the team condition main effect was not statistically significant, $F(2,27) = 1.892$, $p = .170$ ($M_{Synthetic} = 951.71$, $M_{Control} = 953.42$). The main effect of mission was not significant, $F(4, 108) = 1.591$, $p = .182$, nor was the mission–condition interaction, $F(8, 108) = .970$, $p = .463$.

CONCLUSIONS & DISCUSSION

Results demonstrate that team performance is maintained when an AST operates as a pilot within the RPAS-STE. Importantly, the AST's human teammates (i.e., navigator and photographer) maintain the same level of performance

as navigators and photographers who have a human pilot as a teammate. Consequently, we can conclude that training efficacy was maintained with the inclusion of an AST. This does not mean that the AST was without faults or that aspects of the research could have been done differently. In the rest of the section we discuss ways to improve on the current AST and future directions.

The AST was developed to operate as a pilot in the RPAS-STE, requiring a combination of domain-specific knowledge with a set of domain-general knowledge, domain-specific and domain-general components. The language-analysis component is domain-general, yet for the AST to comprehend communications from its teammates it must integrate those communications through a situation model component that is a combination of domain-general and domain specific knowledge. The dialog management and language generation components are largely domain-specific, and the agent-environment interaction component is completely domain-specific. As a consequence, we posit the language analysis component will be able to operate well within most domains; however for an agent to comprehend communications, changes to the situation model component will be required.

The AST evaluation study was a significant success. Much was learned on how to develop ASTs, their minimum requirements (e.g., obtain-relinquish control mechanism for component control, a lexicon of ~60,000 items, simple agents that act as the ASTs teammates for testing, *et cetera*), what not to do (e.g., delay connecting the AST to the STE, develop the AST components in complete isolation, wait on example communication corpora from STE, *et cetera*). Even with all of the knowledge gained, the AST development took over a decade to complete and cost millions of dollars to develop. If we are to be liberal in calculating the savings in training to be 33% in the RPAS-STE, because we replaced an individual in a team of three, it would likely take over a decade to recoup the R&D costs.

If we are to take the call to investigate approaches to more rapidly develop and test ASTs, then we must understand the appropriate level of cognitive fidelity required for the AST. With an increase in cognitive fidelity comes an increase in AST complexity; in turn comes an increase in development cost. It remains a question if ASTs require high-cognitive-fidelity for all applications. If so then we as a community must seek paths to rapidly develop ASTs through more general components, automated knowledge engineering, and rapid testing, evaluation, verification and validation methods. If high-fidelity is not required, then we must understand what components can be lo-fi and which must be hi-fi. With the success of the AST evaluation, the AFRL-ASU team is poised to lead the way in addressing these outstanding issues, and not only within the RPAS-STE described here, but also within an Air Support Operations Center where we are working on the development of an AST capable of performing procedural control.

ACKNOWLEDGEMENTS

We would like to thank the Office of Naval Research for their support in this research (N000140910201 to Christopher Myers and N000141110844 to Nancy Cooke), the Air Force Research Laboratory for their continuing support in Autonomous Synthetic Teammate research, and the many individuals who directly contributed to this project, specifically Dr. Jamie Gorman, Dr. Andrea Heiberg, Dr. Scott Douglas, and Dr. Michael Matessa.

REFERENCES

Anderson, J. R. (2007). How can the human mind exist in the physical universe? (F. E. Ritter, Ed.). Oxford University Press.

Anderson, J. R., & Reiser, B. J. (1985, apr). The Lisp tutor. BYTE, 159–175.

Baddeley, A. (2003, oct). Working memory: looking back and looking forward. Nature reviews. Neuroscience, 4 (10), 829–39. doi: 10.1038/nrn1201

Ball, J. (2006). Can NLP Systems be a Cognitive Black Box? (Is Cognitive Science Relevant to AI Problems?). In Aaai spring symposium: Between a rock and a hard place, cognitive science principles meet ai hard problems. AAAI Press.

Ball, J. (2007a). A Bi-Polar Theory of Nominal and Clause Structure and Function. Annual Review of Cognitive Linguistics, 5 (1), 27–54.

Ball, J. (2007b). Construction-driven language processing. In S. Vosniadou, D. Kayser, &A. Protopapas (Eds.), 2nd European Cognitive Science Conference (pp. 722–727). New York: LEA.

Ball, J., Myers, C., Heiberg, A., Cooke, N. J., Matessa, M., Freiman, M., & Rodgers, S. (2010, aug). The synthetic teammate project. Computational and Mathematical Organization Theory, 16 (3), 271–299. doi: 10.1007/s10588-010-9065-3

Cooke, N. J., & Gorman, J. C. (2009). Interaction-Based Measures of Cognitive Systems. Journal of Cognitive Engineering and Decision Making, 3 (1), 27–46. doi: 10.1518/155534309X433302

Cooke, N. J., & Shope, S. M. (2004). Designing Synthetic Task Environments (Vol. 263).

Core, M. G., & Allen, J. F. (1997). Coding Dialogs with the DAMSL Annotation Scheme. In Proceedings of the 1997 AAAI fall symposium: Communicative action in humans and machines (pp. 28–35).

Endsley, M. (1995). Toward a theory of situation awarenss in dynamic systems. Human Factors, 37 , 32–64. doi: 10.1518/001872095779049543

Endsley, M. R. (2015, feb). Final Reflections: Situation Awareness Models and Measures. Journal of Cognitive Engineering and Decision Making, 9 (1), 101–111. doi: 10.1177/1555343415573911

Ericsson, S. (2004). Dynamic optimisation of information enrichment in dialogue. In the 8th international workshop on formal semantics and pragmatics of dialogue. Barcelona, Spain.

Gibson, E., & Pearlmutter, N. J. (1998). Constraints on sentence comprehension. Trends in Cognitive Sciences, 2 (7), 262–268. doi: 10.1016/S1364-6613(98)01187-5

Klahr, D., Chase, W. G., & Lovelace, E. A. (1983). Structure and Process in Alphabetic Retrieval. Journal of Experimental Psychology: Learning, Memory, & Cognition, 9 (3), 462–477.

Koedinger, K. R., & Anderson, J. R. (1997). Intelligent tutoring goes to school in the big city. International Journal of Artificial Intelligence in Education, 8 , 30–43.

Langacker, R. W. (1987). Foundations of Cognitive Grammar: Volume I: Theoretical Prerequisites. Stanford, CA: Stanford University Press.

Langacker, R. W. (1991). Foundations of cognitive grammar, vol. 2: Descriptive application (Vol. 2).

Matessa, M. (2000). Simulating Adaptive Communication (Unpublished doctoral dissertation). Carnegie Mellon University.

Myers, C. W. (2009). An Account of Model Inspiration, Integration, & Sub-task validation. In 9th international conference on cognitive modeling. Manchester, UK.

Prince, A., & Smolensky, P. (1993). Optimality Theory: Constraint Interaction in Generative Grammar. Studies in Second Language Acquisition, 28 (01), 1–262. doi: 10.1017/S0272263106220060

Rodgers, S., Myers, C., Ball, J., & Freiman, M. (2013). Toward a situation model in a cognitive architecture. Computational and Mathematical Organization Theory, 19, 313–345.

Traum, D. R., & Allen, J. F. (1994). Discourse obligations in dialogue processing. In Proceedings of the 32nd annual meeting on association for computational linguistics - (pp. 1–8). doi: 10.3115/981732.981733

van Dijk, T., & Kintsch, W. (1983). Strategies of discourse comprehension. New York: Academic Press, Inc.

VanLehn, K., Lynch, C., Schulze, K., Shapiro, J. A., Shelby, R., Taylor, L., … Wintersgill, M. (2005). The Andes Physics Tutoring System: Lessons learned. International Journal of Artificial Intelligence in Education, 15(3), 1–47. http://doi.org/citeulike-article-id:7925171

Varges, S. (2006). Overgeneration and ranking for spoken dialogue systems. In 4th international natural language generation conference (pp. 20–22). Sydney, Australia: Association for Computational Linguistics.

Walker, M. a., Whittaker, S. J., Stent, a., Maloor, P., Moore, J., Johnston, M., & Vasireddy, G. (2004). Generation and evaluation of user tailored responses in multimodal dialogue. Cognitive Science, 28 (5), 811–840. doi: 10.1016/j.cogsci.2004.06.002

Zachary, W., Santarelli, T., Lyons, D., Bergondy, M., & Johnston, J. (2001). Using a Community of Intelligent Synthetic Entities to Support Operational Team Training. In 10th Conference on Computer Generated Forces and Behavioral Representations (pp. 215–233).

Zwaan, R., & Radvansky, G. (1998). Situation models in language comprehension and memory. Psychological bulletin, 123 (2), 162–185.