Using Agents to Enhance Performance Assessment of Team Communications

Susan Eitelman
Joan Ryder
Michael Szczepkowski
Thomas Santarelli
CHI Systems, Inc.
1035 Virginia Drive, Suite 300
Fort Washington, PA 19034
215-542-1400
seitelman, jryder, mszczepkowski, tsantarelli@chisystems.com

Keywords:
Performance measures, performance assessment, simulation-based training, communications training, instructional agents, cognitive agents

ABSTRACT: Cognitive Agent-based Real-time Measurement and Assessment (CARMA) is an automated performance assessment methodology for team training applications. This paper describes CARMA particularly focusing on its application to automated performance assessment of team communications. With CARMA, performance assessment is incorporated into the agent’s reasoning, allowing performance assessment to be intelligently adapted to the developing situational context. Examples of our approach are taken from the Virtual Interactive Pattern Environment and Radiocomms Simulator (VIPERS), an on-going Air Force effort developing a simulation-based training system for guided practice and feedback in radio communications and decision making in the Joint Primary Air Training System (JPATS) overhead pattern.

1. Introduction

Performance assessment in complex socio-technological domains, in which teams of operators must work together, is challenging. Significant progress has been made in the last twenty years in characterizing team decision making skills (e.g., Brannick, Salas & Prince, 1997; Cannon-Bowers & Salas, 1997; Swezey & Salas, 1992), defining the types of measures that are indicative of those skills (Fowles et al., 2005; Johnston, Smith-Jentsch & Cannon-Bowers, 1997), and developing methods for designing scenarios that exercise those skills and provide assessment opportunities (e.g., Fowlkes & Burke, 2005). Team training is on the rise. There are a variety of team training environments and technologies aimed to increase the utility and availability of team training such as through the use of scenario-based simulations (e.g., Means, Salas, Crandall & Jacobs, 1993), realistic cognitive agents as team members (e.g., Bell, Johnston, Freeman & Rody, 2004; Zachary, Weiland, Stokes, Scolaro, & Santarelli, 2004), and serious games with player characters for complex, multi-person training applications (e.g., Clark, 2003; Zyda, 2005). Assessing performance in these environments is critical, particularly in the team context in which communication is a key skill or important indicator of performance. Even with the automated data capture that some simulators and game environments provide, the need for role players, observers, and instructors, particularly to assess and debrief team-level skills, is still substantial.

We have developed a methodology for Cognitive Agent-based Real-time Measurement and Assessment (CARMA) to address the need for automated performance measurement and assessment in team training applications. Automated performance assessment in simulations and serious games can improve their instructional value and alleviate the burden on observers and instructors. This paper focuses on the application of CARMA to automated performance assessment of team communications. It incorporates speech recognition and generation, but the heart of the method is cognitive agent-based reasoning about the communications. The cognitive agent encodes instructional knowledge about communications, as well as domain expertise and situation understanding, to assess the communications. It is also able to use assessment results to provide real-time coaching or feedback delivered by a synthetic player within the simulation. Our approach does not involve a natural language tutorial dialog, a technology applicable to intelligent tutoring applications involving unstructured communications (e.g., Graesser, VanLehn, Rose, Jordan & Harter, 2001; Schultz et al., 2003). Instead, it is geared toward applications involving more structured communications, such as radio communications in military domains.

The examples of our approach to performance assessment are taken from an on-going Air Force effort developing a simulation-based training system for guided practice and
feedback in radio communications and decision making while flying the Joint Primary Air Training System (JPATS) overhead pattern. The system, called Virtual Interactive Pattern Environment and Radiocomms Simulator (VIPERS), is a simulation of the pattern environment with cognitive agents as synthetic role-players as well as performing instructional roles (Bell & Cain, 2005; Bell, Ryder & Cain, 2004). The synthetic players include the air traffic control function (called the Runway Supervisory Unit or RSU at JPATS fields), the other pilots/aircraft in the pattern, and the Instructor Pilot (IP), who flies with the student pilot. The IP agent also incorporates instructional functions including performance assessment, coaching, and real-time feedback. The trainee interacts with VIPERS by viewing a schematic representation of the pattern traffic, listening to communications from the RSU agent and simulated pilots, and making radio communications as appropriate and controlling his/her own plane using high-level aircraft control keystroke commands (e.g., “break”, “land”) in order to concentrate on radio procedures and decision making in the pattern. The focus of the VIPERS system is to enable entering pilots to begin to master understanding the JPATS flight pattern and the associated radio communications. Therefore the flying tasks have been simplified so that the trainee can focus on building situational awareness of the pattern and mastering the radio communications without being overburdened by flight tasks, which will be addressed in other aspects of the training curriculum. Also, the communications in the pattern are structured and follow a specific, constrained grammar.

2. CARMA Approach

Figure 1 illustrates the CARMA approach. First, the large, horizontal rectangular box depicts the sequence of steps initiated by a speech event, which are labeled numerically in the figure. For example, the trainee makes a communication (1), then the recognizer detects it (2) and provides data to the cognitive agents for further parsing (3). After the cognitive agent unwraps the speech in the context of the current situation and conducts performance measurement and assessment, then the resultant text and speech feedback are provided to the user (4). Finally, performance information generated by the agent’s assessment is provided to the After-Action Review module of the system for use after the scenario (5). Additionally, situation data about each aircraft are continually being communicated between the simulation and the agent as changes occur. The approach to handling the performance and situation data that is employed by the agent includes five primary steps, which will be covered in more detail in section four and are labeled alphabetically in the figure. The steps are: A) assess the situation in real-time, thereby providing context for use in the assessment, B) generate expected trainee behaviors, C) observe trainee behaviors, D) assess the behavior, by comparing the expected to the observed, and finally E) identify the appropriate feedback including instructional feedback, such as coaching or diagnostic information (for immediate use or in a debrief or after-action review) and also role-player responses. The steps are continually occurring as the situation unfolds and new expectations and observations are made. After the agent processes the behavior and identifies the appropriate feedback, then the results of the assessment are used to drive speech and/or text feedback expressed to the trainee via the speech generator.

The core of CARMA is the cognitive agent and the ability to assess performance in context. As shown in Figure 1, the cognitive agent reasons about the situation using encoded knowledge about the mission and the range of behavior (both communications and actions) that are possible. It also considers which behaviors are appropriate for varying mission phases or tasks. This knowledge, in combination with continually updated situation data, allows the cognitive agent to maintain a real-time assessment of the situation. This situation picture enables the cognitive agent to generate expected communications and actions against which to compare observed speech and action. The cognitive agent also contains performance measure knowledge that supports assessment of the comparison of observed and expected performance. It may involve combining multiple observations, complex assessment algorithms, or comparison to standards for levels of proficiency. The cognitive agent is able to use its knowledge of instructional strategies to determine what type of feedback to provide. Feedback can include immediate text feedback, coaching or evaluative feedback provided by a synthetic instructor or teammate. In addition, the cognitive agent can record performance assessments for later use in debrief or after-action review.

3. Speech Recognition

The first step in CARMA is recognizing the trainee’s speech inputs. Speech recognition enables the cognitive agent to access the content of a trainee’s verbal communications, thereby making it possible for the cognitive agent to reason about the information, assess performance, and respond intelligently. VIPERS utilizes the Nuance VoCon 3200 speech-recognition engine to process trainee-uttered speech communications. Each recognition attempt results in either a semantic frame (i.e., name-value pairs and semantic frame label) or a ‘failure’ message indicating that the utterance didn’t resolve to any known semantic frame.

In a perfect world, the speech recognition technology would result in 100% recognition rates. Despite recent advances in speech recognition technology there are still times when this technology can fail. Speech recognition
errors can occur for many reasons. The most common reasons are improper hardware setup, trainee accents, and the limitations of the technology itself (Stokes, 2001). The two specific errors that require mitigation are:

- Recognition failures on the part of the speech recognition system whereby a trainee speaks an utterance, but the system cannot resolve it to any known recognizable utterance/semantic frame based on the pre-specified grammar;
- Mis-recognitions whereby the trainee’s utterance is recognized as a domain-valid utterance, but not the semantically-intended utterance that was issued (i.e., a false-positive).

The basic approach that we utilize in order to improve recognition accuracy and recover from recognition errors is (1) to apply synthetic-teammate domain knowledge to frame expectations on what might be uttered and (2) to utilize cognitive agent-driven situational awareness that provides context to the recognition engine. Each of these approaches is explained below.

3.1 Agent-Based Speech Assist

It is imperative that trainees be able to interact with the synthetic teammates at all times so that they are able to prosecute their mission and complete their training. Thus, the potential for speech recognition failures necessitates the use of mitigation strategies that enable trainee-agent interaction when failures occur. In order to address this problem, we employ an expectation-based approach within the synthetic teammates to drive a ‘speech-assist’ capability within the trainee’s interface.

The speech-assist that we employ takes the form of a dialog box presented to the trainee when an unrecoverable speech recognition failure occurs. For our purposes, an unrecoverable failure is defined as a failure where the speech recognizer is not able to return enough information for the cognitive agent to infer the trainee’s intended meaning. In these cases, to keep the mission moving, the speech assist dialog box opens and presents the trainee with a list of potential communication choices. An example of this window from the VIPERS training system is shown in Figure 2.
Rather than simply presenting the trainee with a list of all the potential communication choices that one might say during an entire mission, we use an agent-driven approach to populate the list with only those communication choices that are appropriate for the current context. In VIPERS, the cognitive agent representing the RSU uses its underlying knowledge of the current mission and keeps track of the trainee’s actions to determine what the current context is and which communications the trainee should be able to make according to the training requirements. The RSU agent then populates the dialog box with this list.

The trainee is then able to review the choices and select the one that most closely matches what they were attempting to say. Alternatively, they may re-state their radio call instead of making a selection from the choices if they want to practice saying that particular communication. In either case, once a successful communication has been made, the dialog box disappears and the cognitive agent is able to accurately infer the trainee’s intended meaning.

This approach to mitigating recognition failures provides a number of benefits. First, it improves the recognition accuracy by providing communications that are valid within the scope of the particular mission context. Second, it acts as a coaching mechanism by providing domain context to the trainee that can help bolster their understanding of the range of appropriate communications in that particular situation.

### 3.2 Agent-Based Speech Adaptive Grammar

Similarly, to mitigate the problem of mis-recognitions, our approach utilizes the internal context and knowledge of the synthetic teammates to dynamically modulate the speech-recognition grammar in order to improve recognition rates overall by reducing the probability of misrecognitions.

Domain knowledge is used to frame expectations about what might be uttered. The knowledge is then captured by the development of a domain grammar that provides a priori indications of anticipated grammatical structures and the lexical tokens that will populate those structures. Specification of a grammar enables our approach to outperform traditional recognition systems by limiting the number of phrases that can be recognized at a given time to only those that are relevant to the domain or task at hand.

Our approach utilizes the situational awareness inherent to the synthetic teammates to provide context to the recognition engine to reduce misrecognitions. Misrecognition occurs when a spoken utterance is recognized as something allowable by the grammar but different from what was spoken. This approach reduces misrecognitions by using the cognitive agent’s situational awareness and reasoning ability to de-emphasize portions of the grammar that are less relevant to the current context, reducing the number of legitimate utterances at any given moment and thus improving accuracy. This dynamic activation and deactivation of sections of the grammar will increase recognition performance but require the sophisticated interpretations of the trainee’s current situation that only an advanced cognitive model-based agent can provide.

### 4. Agent-based Performance Assessment

With CARMA, performance assessment is incorporated into the cognitive agent’s reasoning. In VIPERS, both the IP and RSU agents reason about the pattern situation and, as such, are able to communicate appropriately with the student pilot; however, the primary instructional functions, including performance assessment, are performed by the IP agent. The RSU and the IP agents are implemented with the iGEN® cognitive modeling tool. Declarative knowledge is represented on a blackboard, and procedural knowledge is represented using a task/goal hierarchy (Zachary, Ryder, Stokes, Glenn, Le Mentec, & Santarelli, 2005). As state conditions are met by the declarative knowledge on the blackboard, various cognitive tasks are triggered, resulting in exhibited behaviors, which in turn change the state of the knowledge represented on the blackboard. Thus, the blackboard represents the dynamically evolving state of the environment, enabling procedural tasks to be executed at appropriate times based on the context of the evolving situation. The iGEN® engine also includes scheduling management and mechanisms that facilitate managing context-dependent behaviors that may become out-of-date before execution is complete. For example, if a student is seemingly late in performing a particular radio communication, the IP transforms an expectation to hear the communication into a coaching behavior to remind the student to perform the communication. However, if the situation is such that the student might be waiting to perform the communication because of a condition in the pattern, such as an emergency, in which other aircraft may take priority for using the radio, then the instructor pilot task could be ‘turned off’ before being executed in light of the situation.

Furthermore, the blackboard architecture sets the
foundation for performance assessment by providing a repository for recording behaviors and the related performance measures that for assessing the appropriate feedback and data to drive a debrief and after-action review.

4.1 Real-time Situation Assessment

The initial step in both RSU and IP reasoning is dynamic situation assessment. This is based on knowledge encoded in the cognitive agent’s memory (as represented on the blackboard) combined with real-time situation data from the simulation environment. In particular, basic information about the pattern and the rules associated with flying and making the appropriate radio calls while in the pattern are encoded as part of the cognitive agent’s knowledge. For example, the symbolic representations of different areas in the pattern (e.g., the closed downwind leg) are stored on the blackboard (see Figure 3 for the leg segments associated with the overhead portion of the JPATS pattern.). “Mission Understanding” in Figure 1 depicts this knowledge. Other examples of “Mission Understanding” include knowledge indicating the sequencing of pattern legs and the priority of flying one part of the pattern versus another.

**Figure 3. JPATS Overhead Pattern Segments**

The instructors in the live environment generally describe the required student pilot behaviors in relation to the area of the pattern in which the student is flying. For example, a student may request permission to “pull closed” into the inner portion of the pattern by making the call, “Texan one five, request closed” where “Texan one five” is the student’s call sign. If there is another aircraft performing a straight-in approach (coming from outside the overhead pattern) and is located between five and two miles (from the approach end of the runway), as indicated by that aircraft’s radio calls, then the student should know that, by pulling closed, a positional conflict between the two aircraft will be created, and thus their request to pull closed will be denied. More advanced students will recognize that it is not appropriate to make the request in that context. Both the RSU and IP agents will have this knowledge and as they receive data on aircraft positions in the simulation, will be able to reason about what communications to expect from the student pilot based on their pattern situation assessment.

Further, information about the radio call formats and the appropriate times for the trainee to make the calls are examples of static information stored on the blackboard, and are depicted in Figure 1 by “Domain Behavior Space.” Such memory is represented by describing relationships between the radio calls and the areas of the pattern where the calls should occur. Other behaviors may also be associated with specific pattern segments such as that the landing gear should be raised after takeoff, on the departure leg. The domain behavior space includes information to generate expectations for those trainee behaviors as well as for communication timing and phraseology.

The dynamically updated situation data enables the real-time assessment of the pattern situation by the RSU and IP agents. So for example, as the situation is dynamically updated and maintained on the blackboard, the RSU agent maintains an updated understanding of the pattern and is able to adaptively respond. When a conflict situation is about to manifest between two aircraft, the RSU will make a radio call that directs one aircraft to break out of the pattern, resolving the conflict. Once the directed aircraft has broken out of the pattern, the updated situation data feeds the new RSU situation assessment the understanding that the conflict has been resolved. This adaptive behavior of the synthetic players in the situation provides natural feedback to the student.

4.2 Generate Expectations

The situation assessment facilitates the cognitive agents’ ability to develop expectations of student behavior. Once formed, the expectations will be compared to observed behavior. Various legs of the JPATS pattern require the student to exhibit leg-specific behaviors (communications or aircraft manipulations). For example, once a student pilot has been granted permission to pull closed, the student then enters the closed downwind leg of the pattern. The closed downwind leg is associated with the following series of required behaviors: a radio call to the RSU to report location, an inter-cockpit request to the instructor indicating readiness to lower the gear; a change in the gear setting upon approval from the IP; and finally another radio call to the RSU to report location when turning from closed downwind into the final turn, which leads to landing or a touch and go maneuver. So given the a priori pattern behavior knowledge encoded in the IP agent, together with the agent’s real-time situation assessment, the IP agent generates expectations for each of these behaviors. Since there are interdependencies among the expected behaviors, the IP agent only generates expected behaviors once the preconditions for the next behavior have been met.
4.3 Compare with Observed

Agent generated expectations enable a comparison against observed behavior. As the situation data is updated, the IP compares what the student does (e.g., makes the call correctly, makes the call incorrectly, or doesn’t make the call) to the expected performance (a correct call made at the appropriate time). For example, the IP first expects to hear the student communicate a “closed downwind” location report. If the student makes the call correctly and within the appropriate time interval, the observation matches the agent’s expectation. Otherwise, a mismatch is noted.

4.4 Assess Performance

The IP agent assesses performance against predefined performance measures that are encoded in its memory. Specific expected behaviors are associated with each performance measure. However, a particular behavior (either a communication or an airplane control action) may contribute to multiple performance measures. For example, a ‘communication timeliness’ performance measure is assessed with every instance of an expected communication. A performance measure of ‘avoiding conflicts’ is assessed at each decision point at which one choice (often indicated by making a clearance request) would lead to a conflict. In such cases, the RSU agent would deny the clearance, and at the same time, the IP agent would record an instance of poor decision making regarding avoiding conflicts.

Using the comparison of expected and observed behaviors, the IP agent is able to assess each performance measurement opportunity. Should observation match expectation, assessment is straight forward. However, if there is a difference between the expected and observed behavior, the assessment may involve further analysis. For example, there are cases in which the position of the student’s plane is not sufficient for an assessment. In those cases, it is necessary to consider the path taken to get to that point in addition to the observed behavior. A plane entering the overhead pattern from a takeoff makes a different radio call on the downwind leg than one entering from the outside. The cognitive agents maintain pattern history as well as current airplane positions as part of their situation understanding, and performance is assessed accordingly.

In another example, if the student is supposed to make a full stop landing after completing the overhead pattern, the IP expects the student to make the closed downwind call with a fuel report appended to the regular call. This expected communication contributes to three performance measures: proper phraseology, communication timeliness, and situation understanding (knowing when to include optional information). In addition, some communications require adjustment if they cannot be made at the correct time due to waiting for another pilot to finish speaking on the radio. For example, a position report is supposed to be made five miles from the runway. However, if it is not possible to report until four miles away, the call should report four miles instead of five.

Each measurement opportunity for each performance measure is recorded to the blackboard as correct or incorrect for use in providing feedback or contributing to an after-action review. If the behavior is incorrect, brief descriptive text is recorded that indicates the diagnosis of the error, which is used later to enable the trainee to identify which part of the behavior needs improvement.

4.5 Determine Feedback

The final cognitive agent reasoning step involves determining the appropriate feedback to provide the student based on the behavior assessment. Feedback includes the cognitive agent responses that the student should expect when flying the live pattern, coaching by the IP, text feedback provided in a feedback window, as well as a post-mission debrief, or after-action review.

The IP agent includes instructional strategies that specify what types of situations lead to each type of feedback, and tasks that provide the feedback. For example, several IP tasks relate to reminding the student to perform a call if the student does not make the call within the expected time period. These tasks reflect the IP’s expectations for a student to make a particular call, and when the expectation is not met within a certain amount of time, the IP assesses the communication as late and provides a reminder to the student to make the call. All measurements are recorded for the after-action review, with explanations provided for all incorrect behaviors.

More complex instructional strategies that depend upon the student’s expertise level may be added in the future. For example, an advanced student might not receive a reminder from the IP, but would simply hear the IP perform the call (the call can not be omitted for the safety of all aircraft in the pattern). Then, the system would record the call as an omission rather than a late call, thus adjusting both the assessment and the instructional strategy.

5. Use of Assessment for Feedback

There are two forms of instructional feedback that CARMA utilizes in order to convey the results of the assessed communications to the trainee:

- Real-time intelligent coaching and feedback – explanation and direction provided while the trainee is engaged in the simulated exercise.
- Intelligent After-Action Review (AAR) – assessment data is stored for more formal, guided presentation to the trainee after the exercise.

5.1 Real-time Intelligent Coaching and Feedback

Real-time coaching and feedback is provided by either synthetic teammates or coaches who interact with the trainee via speech communications. Synthetic teammates, or role-players, can ‘speak’ to a trainee, or other synthetic teammates, by sending communication actions to the speech output engine. In VIPERS, the IP and RSU agents and other simulated student pilots in the pattern are speech-enabled through using Festival, an open-source speech synthesis tool from the University of Edinburgh, which generates and plays wav-files reflective of the utterances issued by the simulated teammates.

The IP is the real-world “coach” for the student pilot. So, just as during a real mission, the synthetic IP provides synthesized feedback to the student pilot about the student’s use or misuse of radio procedures (e.g., “Make a ‘Gear Clear’ call.”, “You forgot to request closed.”). In addition, if the student communicates a request to the RSU that indicates that the student plans to make a dangerous maneuver, the IP intervenes with “Disregard” to the RSU and may even take control of the plane – “I have the plane”.

The RSU is responsible for controlling traffic in the pattern by verbally approving or disapproving pattern requests from the student pilots. Consequently, synthetic RSU responses to requests provide direct, real-time instructional feedback to the student. Examples include, “Negative request closed” and “Closed approved.”

5.2 After-action Review

Our approach provides the capability of automatically generating a post-mission review of communication performance, referred to as an AAR or debrief. This capability is enabled by the instructor agent (in this case the IP agent) and/or synthetic teammates, through the logging of performance data and assessment evaluations as well as the associated context at the time the data was collected and the assessment was made.

In VIPERS, an after-action debrief is provided to the trainee at the conclusion of the mission. It reviews the trainee’s performance on the aforementioned performance measures. Data is presented to the trainee on each measure indicating the total number of measurement opportunities and the number handled incorrectly, as well as an explanation of each error. For example, the review would include a summary of the number of late, omitted, or incorrect communications. An example of an error explanation would be “‘Closed downwind’ call omitted.” or “‘Gear down’ call late.” Additionally, context is provided to the user to facilitate remembering the specific situation in which a behavior occurred. For example, the “snapshot” of the moment a performance measure is recorded includes pertinent situation data, such as where another aircraft involved in a conflict is located, where the trainee’s aircraft is located, and what the last heard radio call was.

In addition to automatically generating the content for the AAR, we will add “guidance” that is automatically adapted to the specific training scenario and trainee performance in that scenario. The guidance may be as simple as ordering the presentation of feedback that considers the past criticality of errors and trainee past performance when determining presentation priority. Or, the guidance can take the form of structured and succinct narrative that communicates the type of information a human instructor would present. Thus, in the AAR for instance, an annotated birds-eye playback of the flight paths flown might be presented over a geographic map and combined with an automatically generated verbal communications log. These context aiding references can be referred to in the narrative presented to the trainee, while remaining available for further interactive exploration at the pace and level of detail chosen by the trainee. The AAR may be organized into meaningful episodes and categories—structured logically but not presented in one continuous flow. Instead, the trainee would be able to control the order and pace at which information is presented by selecting brief or AAR segments. While the instructional agent will provide a default order for the presentation of information, the trainee will be enabled to select a presentation order of their choice. In addition, the trainee will be able to freely navigate the complete information space both during (after the completion of guided chunks) and after the execution of the guided AAR. Within each segment, the instructional agent will focus the trainee attention on relevant information. Many of the intelligent features of the AAR were conceptualized concurrently with the development of the VIPERS system; however, they have not yet been implemented. They are all clearly achievable with the present technology, using the CARMA approach, and we plan to implement them in future iterations of development.

6. Conclusions

CARMA is an approach to automated performance assessment for team training situations involving structured verbal communications. A key component of the approach is the use of cognitive agents for performance assessment and instructional functions, allowing performance assessment to be intelligently adapted to the developing situational context.
The benefit of this approach is that it allows more complex, contextualized assessments. Performance measures may involve more complex analysis than a single expected-observed behavior pair. An example involves assessing counterattack skill in the air-to-air combat domain. There may be three or four options (behaviors), and the best strategy depends on the timing of the attack in the mission, what methods have already been used, altitude, and other factors. In addition, other team skills, such as coordination or situation understanding do not always have observable indicators. Often, by careful scenario construction, a specific verbal communication in a situation can be considered an indicator of coordination or information sharing skills that could not otherwise be assessed without a live instructor. A cognitive agent is able to maintain a performance history, keep track of mission phase, as well as maintain a real-time situation picture to use in assessing performance. Furthermore, a cognitive agent is able to reason about the probable cause of errors and use that to provide informative feedback, including automatically generated, guided AARs.

Overall, CARMA supports instructor-less remedial training opportunities and also supplements live instruction so that the instructor’s time may be used more effectively.

7. Acknowledgements

The VIPERS work was supported by the Air Force Research Laboratory under contract number F8650-04-C-6455. The authors also wish to thank their sponsors at AFRL and the Subject Matter Experts at Laughlin and Moody AFBs who provided invaluable knowledge, guidance, and ideas to the effort.

8. References


**Author Biographies**

**SUSAN EITELMAN** is currently a Cognitive Engineer for CHI Systems, Inc. in Orlando. She develops synthetic interactive agents for training systems, conducts cognitive task analyses, captures user needs and develops functional requirements for training and performance support tools. Her current interests are in human behavior modeling, human machine interaction, and training. Ms. Eitelman received her B.S. in Mechanical Engineering and Psychology from Carnegie Mellon University and she is presently working towards her M.S. in Industrial Engineering from the University of Central Florida.

**JOAN RYDER** is currently a Managing Cognitive Engineer at CHI Systems, Inc. She has over twenty-five years experience in training research and development, human factors engineering, and psychological research. Her interests are in the areas of cognitive modeling, cognitive task analysis, and advanced instructional systems. Dr. Ryder received a Ph.D. in psychology from Brandeis University.

**MICHAEL SZCZEPKOWSKI** is a Managing Cognitive Engineer at CHI Systems, Inc. He has over 16 years applied and research experience in industrial engineering, cognitive engineering, human engineering, and human-computer interaction. He has provided human factors design guidance and cognitive engineering analyses in the medical, military, telecommunication and manufacturing domains. He received his M.S. in Human Factors from the State University of New York at Buffalo.

**TOM SANTARELLI** is the Cognitive Agent Team manager in CHI Systems’ Philadelphia office. He has over fifteen years experience in research into cognitive modeling, the study of human performance in military weapon systems, HCI design and evaluation, and training system development. His research interests include modeling and simulation of human cognitive and organizational behavior and synthetic virtual actors applied to game-based training. He received his B.S. in Information Systems from Drexel University.