AFRL-HE-AZ-TR-2003-0075

UNITED STATES AIR FORCE RESEARCH LABORATORY

AGENT-BASED PLANNING TEAM TRAINING PLATFORM

Daniel N. Mountjoy

Bala Ram

North Carolina A&T State University
Department of Industrial & Systems Engineering
Greensboro, NC 27411

JUNE 2003

AIR FORCE MATERIEL COMMAND
AIR FORCE RESEARCH LABORATORY
Human Effectiveness Directorate
Warfighter Training Research Division
6030 South Kent Street
Mesa AZ 85212-6061

NOTICES

SHARON K. GARCIA
Project Scientist

HERBERT H. BELL
Technical Advisor

CURTIS J. PAPKE, Colonel, USAF
Chief, Warfighter Training Research Division

Contract Number: F33615-01-1-6073
Contractor: Department of Industrial & Systems Engineering
North Carolina A&T State University

Direct requests for copies of this report to:

Defense Technical Information Center
8725 John J. Kingman Road, Suite 0944
Ft. Belvoir, VA 22060-6218
https://stinet.dtic.mil

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. REPORT DATE *(DD-MM-YYYY)*
June 2003

2. REPORT TYPE
Final

3. DATES COVERED *(From - To)*
Sept 02 to Mar 03

4. TITLE AND SUBTITLE
Agent-Based Planning Team Training Platform

5a. CONTRACT NUMBER
F33615-01-1-6073

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER
65502F

5d. PROJECT NUMBER
1123

5e. TASK NUMBER
B1

5f. WORK UNIT NUMBER
18

6. AUTHOR(S)
Daniel N. Mountjoy
Bala Ram

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Dept of Industrial & Systems Engineering
North Carolina A&T State University
Greensboro, NC 27411

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)
Air Force Research Laboratory
Human Effectiveness Directorate
Warfighter Training Research Division
6030 South Kent Street
Mesa AZ 85212-6061

10. SPONSOR/MONITOR'S ACRONYM(S)
AFRL; AFRL/HEAI

11. SPONSOR/MONITOR'S REPORT NUMBER(S)
AFRL-HE-AZ-TR-2003-0075

12. DISTRIBUTION / AVAILABILITY STATEMENT
Approved for public release; distribution is unlimited.

13. SUPPLEMENTARY NOTES
Air Force Research Laboratory Technical Monitor: Dr Sharon K. Garcia, (210) 536-3989; DSN 240-3989

14. ABSTRACT
With an increased emphasis on joint service operations, smaller force sizes and rapid deployment, the team paradigm within the U.S. military is moving more toward those with distributed members. Team members are often isolated from one another due to geographic or vehicular constraints, as in a "command on the move" situation. An undesirable outcome of this teaming arrangement is a possible reduction of team members' abilities to know and understand what others are doing around them, what decisions they are making, or what information they are using. Training under these circumstances also becomes complicated since members may be geographically distributed, and bringing everyone together under the same roof simultaneously is expensive in terms of time and travel, and is particularly difficult to schedule. An intelligent agent-based platform was developed as a proposed method to assist in the training of distributed teams. Agents within the platform were designed to represent specific functions within a supply chain management team. An experiment was performed to examine the training efficacy of the agent-based platform against a traditional instructional training technique. Results indicate that an agent-based platform can be an effective training tool, but larger studies should be performed before drawing firm conclusions.

15. SUBJECT TERMS
Agent-based training; Distributed training; E-learning, Electronic learning, HBCU; Historically Black Colleges and Universities; Team Training; Training tools; Web-based learning

16. SECURITY CLASSIFICATION OF: a. REPORT	b. ABSTRACT	c. THIS PAGE	17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
					Ms Liz Casey
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	UNLIMITED	21	**19b. TELEPHONE NUMBER** *(include area code)* 480.988.6561 x-188 DSN 474-6188

Contents

List of Figures

List of Tables

AGENT-BASED PLANNING TEAM TRAINING PLATFORM

Executive Summary

In general terms, a team can be considered to consist of several individuals (or other similar entities) acting together in some fashion to achieve a common goal. Team members may have the same or different skills and/or responsibilities. Specific behaviors of the individuals, when conducted in concert, contribute toward meeting the team's desired objective. While not necessarily the case, one typically considers a team to act in relative close spatial proximity. That is, team members are not generally very far from one another, and can "see" the effect of other members' actions on meeting the desired end-state. Based on the outcomes of such actions, other team members may alter their subsequent actions/behaviors to ensure continued progress toward the goal. This concept of team is somewhat analogous to a closed-loop control system where system output (feedback) augments the system input to keep the system in control (i.e., moving toward the desired end-state).

In recent years, however, the previously described concept of "team" as having members in close proximity has become less common. Particularly within the U.S. military, with an increased emphasis on joint service operations, smaller force sizes and rapid deployment, the team paradigm is moving more and more toward distributed members. Team members are, in effect, often isolated from one another due to geographic or vehicular constraints, as in a "command on the move" situation. An undesirable outcome of this teaming arrangement is a possible reduction of team members' abilities to know and understand what others are doing around them; what decisions they are making, what information they are using, etc. In essence, to continue with the control system analogy, the feedback loop is compromised by noise – in this case generated from uncertainty due to missing information, incorrect or incomplete information. If the noise component is too great, the system will become unstable and may never converge toward the goal. Obviously, this is exactly opposite the intention of the military. Under both the team scenarios, training to function in the teams is very limited and is usually conducted with all team members in a single location. Training is complicated since members may be geographically distributed, and bringing everyone together under the same roof at the same time is expensive in terms of time and travel, and is particularly difficult to schedule. At issue, then, is how to facilitate team training even if many team members are not available. An intelligent agent-based platform for team training was developed to determine if such a system could be used as an effective team-training tool. The intelligent agents in this platform represented team members from different functions in a manufacturing planning task. Teams are commonly employed in planning tasks, and hence the choice. The system was designed in such a way as any combination of one-to-three team members could be replaced by a software agent, while the other team members participated along with the agents in the planning task. The platform was used as a training tool within the experimental design, and its relative performance was compared to that of a traditional, instruction-based approach. Experimental results indicate that the agent-based approach performed well, and most often paralleled the performance of the traditional training. While these results are somewhat preliminary in that a relatively small number of participants took part in the study, it is suggested that the agent-based approach to team training can be helpful, even if team members do not take part in the training at the same time.

AGENT-BASED PLANNING TEAM TRAINING PLATFORM

1.0 Introduction

The U.S. military is striving to be a smaller, faster, more lethal force. Beyond the obvious importance of technology and equipment development, one of the key elements to making this happen is proper training of its personnel. According to Joint Vision 2020, Chairman of the Joint Chiefs of Staff, General Henry H. Shelton (2000, p.3) writes, "…material superiority alone is not sufficient. Of greater importance is the development of doctrine, organizations, training and education…that effectively take advantage of the technology."

Winning the so-called "information war" is not just about having more information on which to base decisions (information superiority), rather it involves the ability to make superior decisions than the enemy (decision superiority). In this technology-rich information age, one cannot assume the United States forces are the only ones with information gathering capabilities; therefore, it is necessary to focus on the proper use of that information in order to make better, faster decisions than our adversaries. A primary difficulty in this area is how to sort through vast volumes of information, and, particularly in team arrangements, how to decide which information is important to share with other team members in order to form a shared mental model of the current environment.

A second problem presents itself in the relatively new concept of a "team." Where teams were once thought of as co-located individuals working together to accomplish a common goal, teams are more and more becoming distributed in nature; that is, team members may often be physically located in separate geographical areas both during training as well as during actual team operations. Operationally, it can be assumed that team members will be brought together if necessary to mission success, but this can be costly and difficult to schedule for training purposes. This poses obvious problems with team communication, and the ability to share information critical to success of the team in meeting its objectives.

In general terms, a team can be considered to consist of several individuals (or other similar entities) acting together in some fashion to achieve a common goal. Team members may have the same or different skills and/or responsibilities. Specific behaviors of the individuals, when conducted in concert, contribute toward meeting the team's desired objective. While not necessarily the case, one typically considers a team to act in relative close spatial proximity. That is, team members are not generally very far from one another, and can "see" the effect of other members' actions on meeting the desired end-state (see Figure 1). Based on the outcomes of such actions, other team members may alter their subsequent actions/behaviors to ensure continued progress toward the goal. This concept of team is somewhat analogous to a closed-loop control system where system output (feedback) augments the system input to keep the system in control (i.e., moving toward the desired end-state).

In recent years, however, the previously described concept of "team," has become more obscure. Particularly within the U.S. military, with an increased emphasis on joint service operations, smaller force sizes and rapid deployment, the team paradigm is moving more and more toward distributed members. Team members are, in effect, often isolated from one another due to geographic or vehicular constraints, as in a "command on the move" situation. An undesirable outcome of this teaming arrangement is a possible reduction of team members' abilities to know and understand what others are doing around them; what decisions they are making, what information they are using, etc. In essence, to continue with the control system analogy, the feedback loop is compromised by noise – in this case generated from uncertainty due to missing, incorrect, or incomplete information. If the noise component is too great, the system will become unstable and may never converge toward the goal. Obviously, this is exactly opposite the intention of the military.

Beyond team performance, issues involved with training itself are made more complicated since members are geographically distributed, and bringing everyone together under the same roof at the same time is expensive in terms of time and travel, and is particularly difficult to schedule. At issue, then, is how to reduce the noise in the system during team training if certain team members are either absent or are perhaps dealing with incomplete information. In a non-distributed team situation, even if individual team members have different information sources available to them, the relative close proximity allows other members the ability to observe their behavior and possibly even the information source itself. One of the major difficulties in distributed teams is the ability of members to form a common mental model of the environment when expertise differs from member to member, and each member and their respective information sources are physically isolated from one another. In this type of situation, the use of intelligent agents may help bridge the information gap.

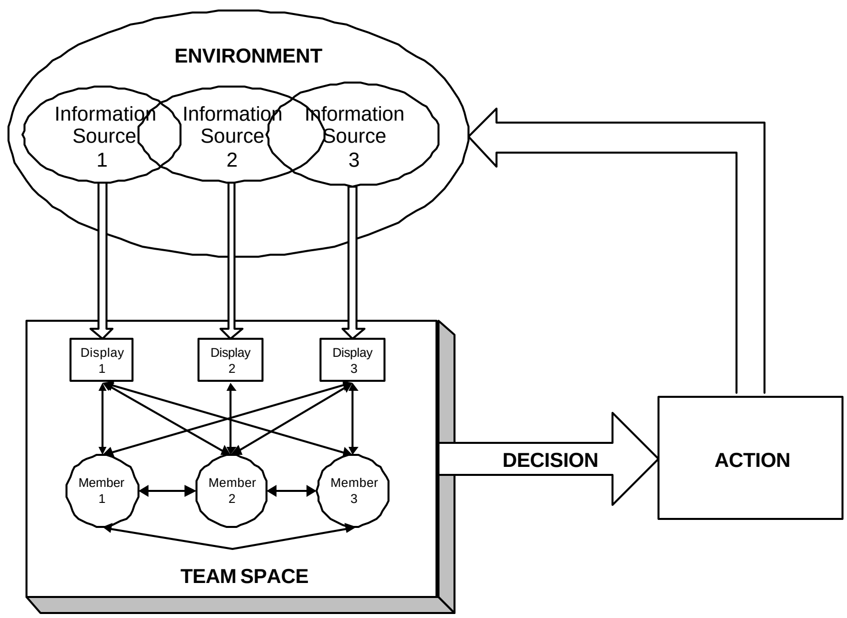

Figure 1. Theoretical model of a non-distributed team. Arrows within the team space represent lines of direct communication available to team members.

1.1 Purpose

The purpose of this research was two-fold: (1) to develop an agent-based team training platform, and (2) to evaluate its effectiveness in training team members to perform a team-oriented planning task.

1.2 Background

In the following sections, efforts of other researchers relevant to our proposal are reviewed. The first section addresses the use of agents in the design of systems to support supply chain management, while the second section addresses contemporary issues in team training.

1.2.1 Intelligent Agents

Multi-Agent Supply Chain cOordination Tool, MASCOT (Sahdeh, Hildum, Kjenstad & Tseng, 1999), is a reconfigurable, multilevel, agent-based architecture for coordinated supply chain planning and scheduling aimed at supporting some key functions required for agile environments. A key to agility in today's dynamic high-mix production environments is the ability to (1) effectively coordinate production across multiple facilities, whether internal or external to the company, and (2) quickly and accurately evaluate new product/subcomponent designs and strategic business decisions (e.g., make-or-buy or supplier selection decisions) with regard to capacity and material requirements across the supply chain. The architectural elements of MASCOT architecture have a special emphasis on the support of real-time mixed-initiative "what-if" functionalities, enabling end-users at different levels to rapidly evaluate alternative tradeoffs and their respective impact across the supply chain. MASCOT also incorporates new coordination protocols aimed at better exploiting the power of finite capacity scheduling functionalities across the supply chain. Empirical results are presented quantifying the benefits afforded by these new protocols under different loads and supply chain configurations.

Brinn and Carrico (1999) address the issue of inter-agent communication in the context of logistics. The application of distributed agent techniques to large-scale problems like global logistics planning raise interesting challenges regarding the communication and management of information between the agents. The distributed nature of the problem often makes for wide gaps of knowledge that need to be bridged through an expressive yet efficient language for describing requirements and service results. The Department of Defense's Advanced Logistics Program (ALP) has developed an infrastructure for representing extremely large problems in a distributed manner. The use of an expressive communication language along with techniques for state encapsulation and decision space tradeoff analysis have provided for efficient coordination among large numbers of medium complexity planning agents towards solving large, complex problems. This article describes in detail the language and structures developed under ALP to effect the inter-agent communications that enable successful distributed coordination.

Shen and Norrie (1999) provide a survey of efforts that have addressed the application of agent technology to manufacturing systems. The application areas include manufacturing enterprise integration, supply chain management, manufacturing planning, scheduling and control, and materials handling. In addition to the survey of efforts, the paper also discusses some key issues in developing agent-based manufacturing systems.

In their work with the Rock Island Arsenal, Parunak, Baker & Clark (1997) present a comprehensive example of the application of agents in a manufacturing environment. Their experience with agent application suggests that each functional agent needs detailed knowledge of many of the physical entities being managed. When the physical system changes, the functional agent needs to change as well. However, it is possible to define generic behaviors for physically defined agents from which the required functionality will emerge, for a widely varying overall population of agents. In the Rock Island Arsenal application separate agents are defined for material handling and processing resources. No agents are defined for a function like scheduling, as this function emerges from the interaction of the other agents.

Karsai, Bloor & Doyle (2000) describe an agent application for Autonomic Logistics which envisions a maintenance and support system that can autonomously respond to "events", e.g. problems detected on-board the aircraft. The response includes identifying the source, acquiring the correct parts and tools, and locating and scheduling the right maintenance personnel. Agents represent tasks that need to be performed in such a system, and they autonomously exit, navigate, and negotiate within the framework. Even in existing maintenance organizations human negotiation is an accepted practice, which, in many cases, "makes the system work", in spite of rigid, formal rules. The negotiation among the agents is modeled after human negotiation processes.

1.2.2 Issues in Team Training

When referring to team training, an abundance of literature revolves around Crew Resource Management (CRM) methodologies. CRM was developed to study and improve human performance in cockpit environments (Weiner, Kanki & Helmrich, 1993), with the intent that all information/resources available to cockpit crews was being properly utilized (Salas, Rhodenizer & Bowers, 2000). While developed for cockpit teaming arrangements, data gathered and lessons learned can be applied to teams operating in other operational environments.

According to Salas, Rhodenizer & Bowers (2000), CRM training is aimed at improving both inter- and intra-crew performance in order that the team reaches its desired results. The goals as they apply CRM in the flight domain are to reduce accidents and errors, increase flight efficiency, and optimize utilization of resources. While they mention that there is no general agreement on an absolute definition of CRM training, the goals are much the same as in any evaluation of teamwork.

Cannon-Bowers, Salas, and Converse (1993) contend that the development of shared mental models is an important aspect of team training. They recommend cross training as a way for team members to become more aware of teammates tasks and to predict their information needs. This view is supported by McIntyre and Salas (1995) who maintain that some of the important components of teamwork include monitoring of other team members' performance, providing feedback to each other, and practicing closed-loop communication.

While much has been learned through CRM research over the last several years, Salas et al. (2000) point out that not enough is known about the benefits of advanced technology such as various display types and computer simulations for team training. Furthermore, CRM focuses on teams where members are co-located. The evolving situation of interest is one where team members are either absent altogether, or are geographically separated from each other.

1.3 Approach

There is a need to train individuals to perform in team-oriented deliberative tasks. We addressed this issue by developing an approach to provide such training for individual team members, without assuming the availability of the other team members.

The following specific tasks were conducted during the performance period of this grant:
1. Development of an intelligent agent-based platform for training in a team-oriented planning task.

2. Investigation of the efficacy of training on such a platform in terms of performance in the team-oriented planning task.

The training scenario developed reflected a short-term production-planning task for a PC manufacturing operation. While this scenario does not reflect a military domain directly, it is believed that the conclusions drawn regarding the training effectiveness can be generalized to other task domains. A description of the agent-architecture development and training effectiveness evaluation efforts are provided in the following sections.

2.0 Training Material Development

2.1 Multi-Agent System Implementation

The multi-agent training system consists of five agents assigned to specific tasks. Four of them are corresponding to the elements shown in Figure 3. The five agents are the Manager Agent, Design Agent, Manufacturing Agent, Purchasing Agent, and a Facilitator Agent. The agents' functions are described in Table 1.

Table 1. Description of Agent Functions

Agent Name	Agent Function
Manager Agent	Creates a pc specification for the new product request
Design Agent	Plans a list of pc components based on the new product request
Manufacturing Agent	-Evaluates the part list from Design; -Evaluates the vendor list from Purchasing
Purchasing Agent	Plans a list of vendors for the new product components
Facilitator Agent	Registers trainees into the system

We have used a high-level agent building software called AgentBuilder (Reticular Systems, 2000) in building the prototype system. AgentBuilder is an integrated toolkit for constructing intelligent software agents [9]. It provides graphical tools for supporting all phases of the agent construction process. Programming software agents is accomplished by specifying intuitive concepts such as the beliefs, commitments, and actions of the agent.

2.1.1 Agents Architecture

The system architecture can be divided into three layers. The rules, beliefs, commitments, and actions associated with each agent are in the first layer. They are all defined by using the agent constructing functions in AgentBuilder. The ontology and classes it uses are in the second layer. The ontology used by agents is first defined in the Object Modeler in AgentBuilder. The class entities are realized by using Java JDK 1.4. The third layer is the database. All the source data, messages, and final results are stored there. It is realized using Microsoft Access 2000. The connection between the database and ontology classes is handled by JDBC-ODBC.

2.1.2 Agents Deployment

There are two modes each for Design Agent, Manufacturing Agent, and Purchasing Agent: one with the agent replaced by a trainee, or one with the agent by itself. More than one agent can be replaced simultaneously. When an agent is not replaced, it will allow the real person to function and make decisions. When an agent is replaced, the trainee will be given a user interface window to participate in the training environment. The Facilitator Agent controls the two types of ontologies for the two modes during the initial agent registration process.

The Facilitator Agent is created when the training platform is started. It keeps a list of registered agents. Agents are not able to communicate directly with each other. Agents first send messages to the Facilitator Agent. . The Facilitator Agent then sends messages to appropriate agents. The use of the Facilitator Agent eliminates the chance that agents will not receive their messages and provides a structure for monitoring the messages passed between agents. All messages are passed in a modified KQML format.

For evaluation of the prototype, a geographically distributed team training environment can be simulated by deploying the five agents into five different computers that are located in five different rooms in one or buildings. All the agents communicate through a local area network.

2.1.3 Run-time Screenshots

The agent registration process can be seen from the AgentBuilder-AgencyViewer window that is shown in Figure 2. Figure 3 shows the user interface for the design engineer. The trainee can see the product specification from the table in the window and makes choices. Each agent has a message window as shown in Figure 4. Agent status and message exchanging processes can be seen from this window.

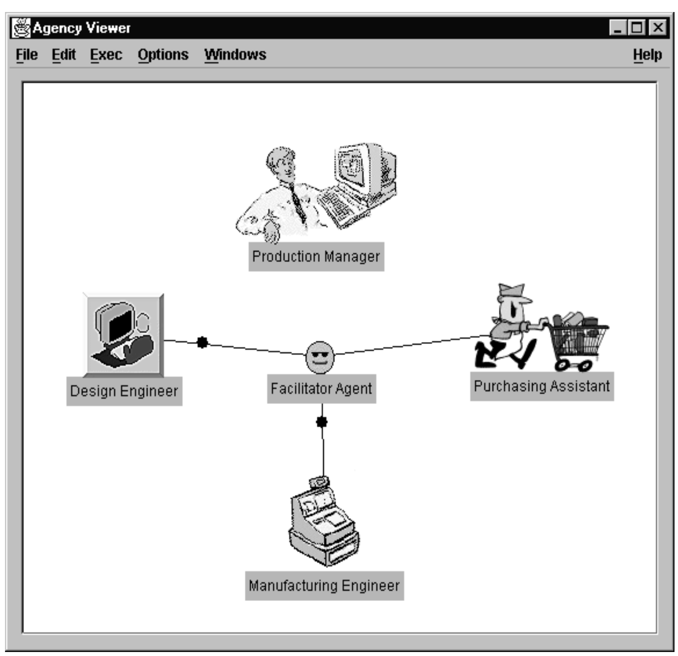

Figure 2. Agents registration with Facilitator.

Figure 3. User interface for Design Engineer.

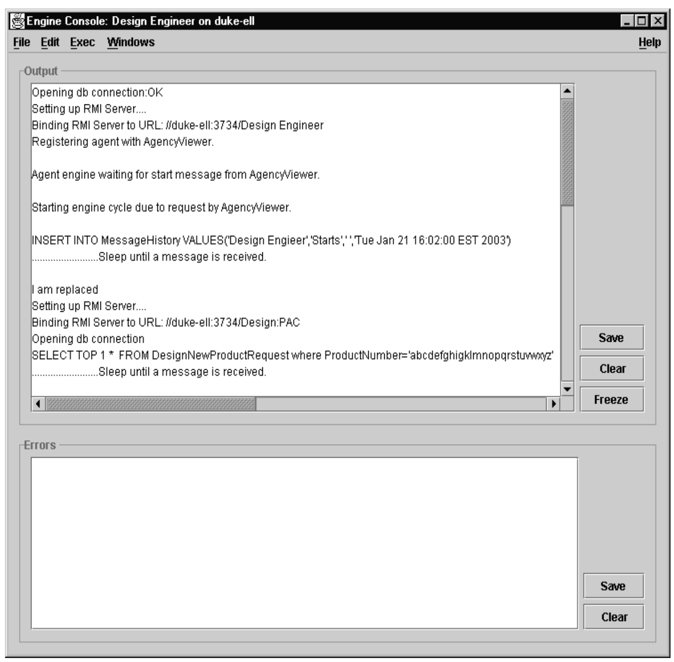

Figure 4. Message window of the Design Agent.

2.2 Instruction Manuals

The instructional manual included an overview to provide an introduction and background of the decision-making process. A job description was included to give each participant a description of what his/her responsibilities are. The manual also consisted of a section for the terminology and definitions that the participant needed to be aware of. There was also a copy of the criteria and constraints that assisted in the decision-making process. A conscious effort was made to provide the same type of information that can be received through the agent-based system to reduce the possibilities that any training effects would be due to the training technique itself, not the quality of information presented in either method. The instruction manuals are provided in Appendix A.

3.0 Experimental Method

The goal of this study was to assess the effectiveness of agent-based team training versus a more traditional training approach. To address this issue, a simplistic manufacturing supply chain scenario was formulated that required teams to "produce" a personal computer through the correct identification of components to meet a production manager's performance and delivery date specifications.

3.1 Participants

Thirty individuals volunteered to participate in this study. Participants were a mixture of undergraduate and graduate industrial engineering students at North Carolina A&T State University, ranging from 20 to 41 years of age (average age = 24 years).

3.2 Experimental Design

The experiment was conducted as a between-groups, pre-training-post-training design. Each participant was randomly assigned to one three-member team, and each team received only one type of training. The independent variable in the study was Training Type, which had five levels: Control (C), Traditional (T), Traditional with Cross-Training (TX), Agent-Based (A), and Agent-Based with Cross Training (AX). Dependent variables were task completion time, number of correct selections made during the task, number of interactions amongst team members, proportion of correct interaction amongst team members, and the average score on the knowledge assessment prior-to and after training.

3.3 Procedure

The experiment was conducted in three different phases: pre-training, training and post-training. These phases took place over the course of three days for all participants.

During the pre-training phase participants signed a consent form, then completed demographics and pre-training knowledge assessment questionnaires. The pre-training knowledge assessment was designed to determine the subject's knowledge of each department's roles represented on the team prior to training. The team completed a task that required them to make decisions based on given criteria and constraints initially provided by the production manager (the experimenter). Each personal computer consisted of four major components: a motherboard, RAM (Random Access memory), hard drive, and CPU (central processing unit). The team's goal was to produce a PC with for the lowest price that would meet the production manager's specified performance requirements and delivery date. An overview of the required information flow and decisions made by each team member is portrayed in Figure 5. The task was videotaped to assist in data collection. The post-training phase was identical to the pre-training phase, with the exception of the consent form and demographics questionnaire.

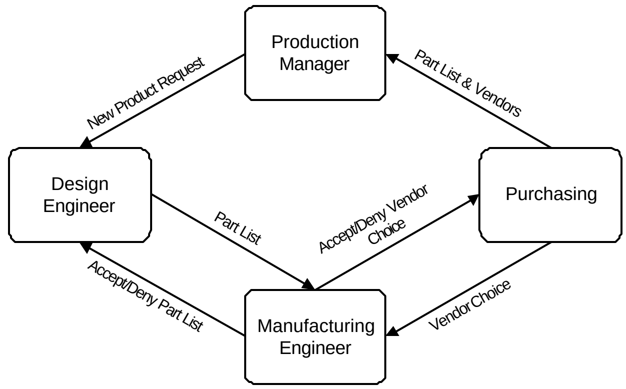

Figure 5. Flow of information required for the PC production task.

The training phase consisted of the team participating in the training method that corresponded to their assigned training type. The training simulated the process of the task that was completed in the pre-training phase. The process in each training type was similar in sequence, the only difference being the training approach. The control groups were the only teams that did not receive any training. The traditional trained groups and the traditional cross-trained groups received the traditional instruction-based training. These teams were provided training manuals and used role-playing as a simulation method. The agent and agent cross-trained groups received the intelligent agent-based training, in which they used the intelligent agent software as a basis of their training. In both training techniques, team members participated as their assigned role – the difference being, the traditional trainees participated with other people, the agent-based trainees participated along with intelligent agent counterparts.

4.0 Results

Due to a small sample size and the likelihood of dealing with non-normally distributed data, all data was analyzed via a non-parametric statistical technique, namely the Wilcoxon Signed Rank test. The analysis was run to determine if there was a statistically significant effect of training on subject matter knowledge (Figure 6), task completion time (Figure 7), and the number of correct interactions between team members (Figure 8). Statistically significant results were obtained for subject matter knowledge, $T(10) = 0, p < .05$, and task completion time, $T(10) = 8, p < .05$. Implications of these results are discussed in the next section.

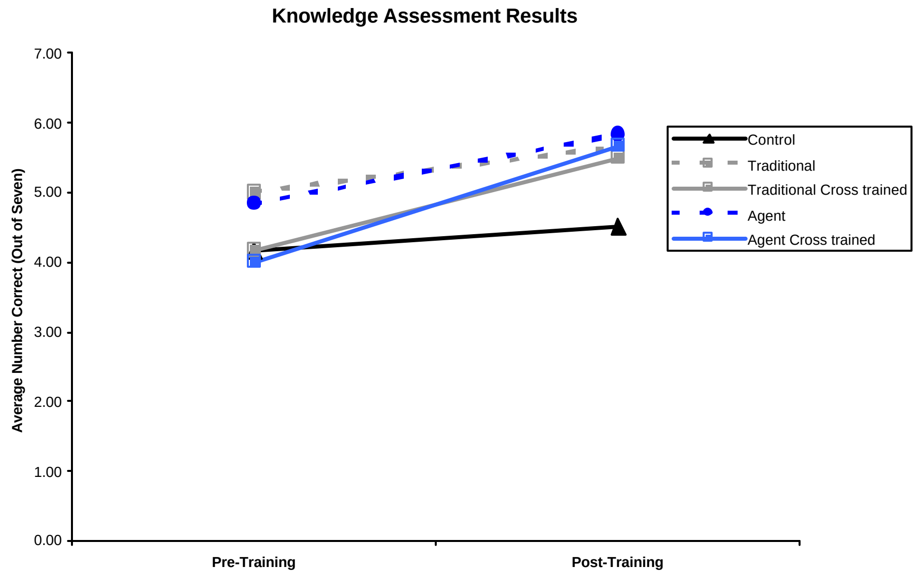

Figure 6. Results of the knowledge assessment questionnaire.

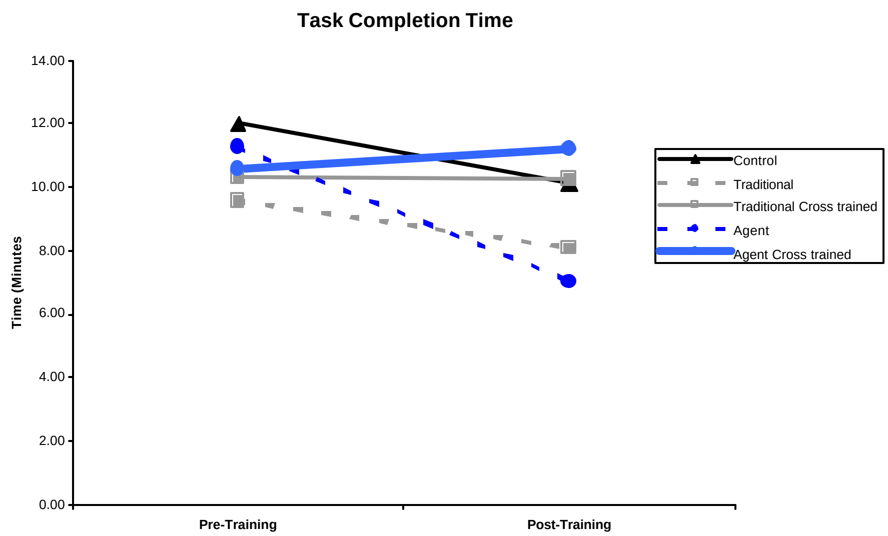

Figure 7. Task completion times recorded prior to- and after training.

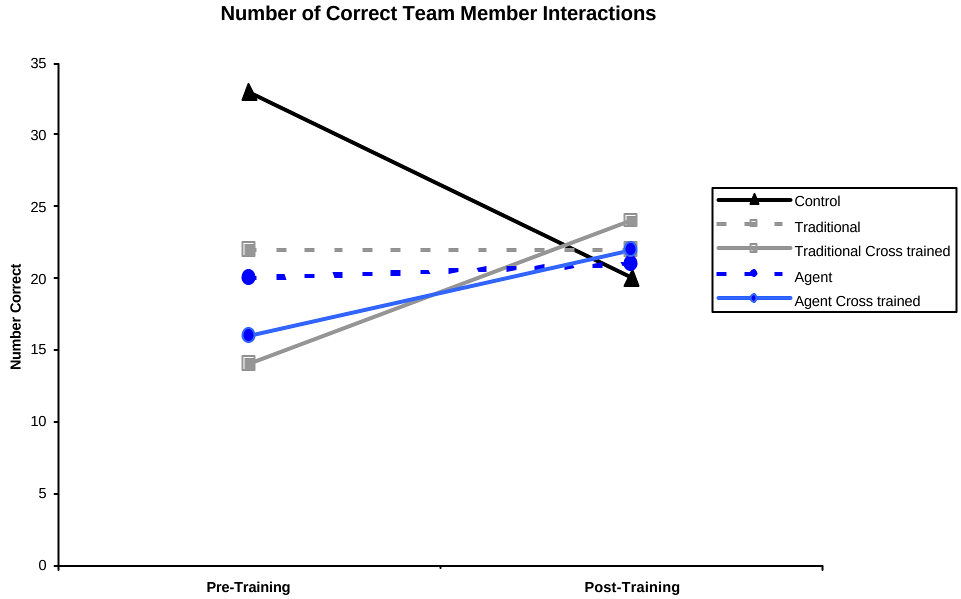

Figure 8. Number of correct interactions between team members during the PC production task recorded prior to- and after training.

5. Discussion

It should be noted that the main point of interest in all results is the difference between the pre-training and post-training data. Participants were randomly assigned to training groups in an attempt to even out the knowledge and experience level between groups, but this does not absolutely guarantee that all groups are equal, if for no other reason than chance.

5.1 Subject Matter Knowledge Assessment

It is readily apparent from Figure 2 that all training types resulted in a positive effect on the participants' knowledge of supply chain management subject matter. The control group also shows a slight increase in knowledge level, although to a lesser degree than the experimental groups as would be expected. It is interesting to note that the results of the cross-trained teams (*AX* and *TX*) paralleled each other, as did those of the non-cross-trained teams (*A* and *T*), regardless if they participated in the traditional or agent-based training. Overall, the cross-trained teams showed a greater increase in their knowledge level than their non-cross-trained counterparts.

5.2 Task Completion Time

While it was expected that task completion time would decrease following training, Figure 3 reveals mixed results, with training shown to benefit only some of the groups. The *A* and *T* training groups and the control group were able to decrease their respective task performance times, the *TX* group was virtually unchanged, while the *AX* group saw a slight increase in completion time. The *AX* group data was a bit skewed due to poor performance by one of the two *AX* groups in the post-training task. It is suggested that the cross-trained groups would have also seen a decrease in times (on average) had there been a larger subject pool.

5.3 Number of Correct Team Member Interactions

Since in the pre-training phase, team members were not provided any sort of guidance on which other team members they should communicate with in order to complete the task, it was expected that the number of correct interactions would increase following the training phase. Indeed, as shown in Figure 4, the *A*, *AX*, and *TX* groups each increased the number of meaningful interactions following training, while the *T* group held steady. Curiously, the number of correct interactions in the control group *decreased* markedly during their second task performance. It is noteworthy that the *AX* and *TX* groups exhibit the largest increase in correct interactions since cross training is often viewed as being particularly beneficial to team performance.

6.0 Conclusions

Since teams are increasingly more likely to include members from various geographic regions, it is wise to investigate team training methods that can help reduce travel costs and associated logistics problems. The research summarized here helps to support the idea that intelligent agents can be a beneficial piece of that solution. While no evidence was found to show that agent-based training is *better* than more traditional approaches such as instruction and role-playing, this was not entirely the goal of this work. The intent was to determine if intelligent agent based software can be used to effectively train teams. In that sense, the overall results seem to indicate that agent-based training does result in a positive training effect (often paralleling results from the traditional methods), and indeed can be used effectively as a tool for team training. It is recommended that similar studies with larger sample sizes be conducted in order to draw more rigid conclusions than those suggested here. Two articles submitted as conference proceedings that summarize the training platform and the evaluation are provided in Appendices B and C.

7.0 Recommendations

Based on the work performed as part of this research effort, the following recommendations are made for future work:

- Perform a larger scale experiment in order to draw more robust conclusions
- Determine effectiveness of agent-based training in a different domain (i.e., military planning)
- Determine the extent to which team member training absenteeism affects team performance (i.e., one person/two agents, two people/one agent, three people/no agents)

References

Bigus, J. & Bigus, J. (1997). *Constructing intelligent agents with Java*. New York: John Wiley & Sons.

Brinn, M. and T. Carrico (1999). The Language of Dynamic Planning: Inter-Agent Communication in ALP, http://www.arpa.mil/iso2/alp/.

Cannon-Bowers, Salas, and Converse (1993). Shared mental models in expert team decision making. In N. J. Castellan (Ed.), *Current issues in individual and group decision making* (pp. 221-246). Mahwah, NJ: Erlbaum

Karsai, G., G. Bloor, and J. Doyle (2000). Automating human based negotiation processes for autonomic logistics, Proceedings of the 2000 IEEE Aerospace Conference, Nashville: Tennessee, Vol. 6, pp. 147-153.

McIntyre, R. M., and Salas, E. (1995). Measuring and managing for team performance: emerging principles from complex environments. In R. A. Guzzo and E. Salas (Eds.), *Team effectiveness and decision making in organizations* (pp. 9-45). San Francisco: Jossey-Bass.

Mountjoy (2001). Perception-based development and performance testing of a non-linear map display. Unpublished doctoral dissertation. http://www.lib.ncsu.edu/etd/public/etd-8421519310141081/etd.pdf. North Carolina State University, Raleigh, North Carolina.

Mountjoy, D. N., and Marshak, W. P. (1999). Impact of non-linear mapping on mileage estimation. In *Proceedings of the Army Research Laboratory Advanced Displays and Interactive Displays Federated Laboratory Third Annual Symposium* (pp. 97-101). Adelphi, MD: US Army Research Laboratory.

Mountjoy, D. N., Ntuen, C. A., Converse, S. A., and Marshak, W. P. (2000). Basing non-linear displays on vector map formats. *Journal of Navigation, 53,* 68-78.

Mountjoy, D. N., Ntuen, C. A., and Yarbrough, P. L. (1998). Comparison of three display methodologies for battlefield resource management. In *Proceedings of the Human Factors and Ergonomics Society 42nd Annual Meeting* (pp. 476-480). Santa Monica, CA: Human Factors and Ergonomics Society.

Naghshineh-pour, R., Williams, N. & Ram, B. (1999). Logistics issues in autonomous food production systems for extended duration space exploration. In *Proceedings of Winter Simulation Conference*, San Diego, CA.

Ntuen, C. A., Park, E. H., Eastman, S., Mountjoy, D. N. and Yarbrough, P. L. (2000). ACAD: A decision support display for commander's visualization of alternative courses of action during battle planning. In *Proceedings of the Army Research Laboratory Advanced Displays and Interactive Displays Federated Laboratory Fourth Annual Symposium*. Adelphi, MD: US Army Research Laboratory.

Parunak, H.V.D., A.D. Baker, and S.J. Clark (1997). The AARIA agent architecture: an example of requirements-driven agent-based system design. In *Proceedings of the First International Conference on Autonomous Agents (ICAA-97).*

Ram, B., M.R. Naghshineh-pour and N. Williams (1999). Logistics Issues In Autonomous Food Production Systems For Extended Duration Space Exploration, In *Proceedings of the 1999 Winter Simulation Conference*, San Diego, California.

Reticular Systems (1999). Agent-Builder: An Integrated Toolkit for Constructing Intelligent Software Agents, San Diego, CA.

Sadeh, N., D. Hildum, D. Kjenstad, and A. Tseng (1999). MASCOT: An agent-based architectural for coordinated mixed-initiative supply chain planning and scheduling. In *Proceedings of the Third International Conference on Autonomous Agents (Agents'99).* Seattle, WA.

Salas, E., Rhodenizer, L., & Bowers, C. A. (2000). The design of crew resource management training: exploiting available resources. *Human Factors, 42*, 490-511.

Shen, W. and D. H. Norrie (1999). Agent-based systems for intelligent manufacturing: a state-of-the-art survey," *Knowledge and Information Systems*, *1*(2), 129-156.

Smith, N. and Ram, B. (1999). An application of the agent paradigm in autonomous production systems. In *Proceedings of the Industrial Engineering and Management Systems (IEMS) Conference*.

Weiner, E. L., Kanki, B. G., & Helmreich, R. L. (Eds.). (1993). *Cockpit resource management.* San Diego: Academic Press.

GLOSSARY OF TERMS

Abbreviations and Definitions

CPU - Central Processing Unit; the processor that controls the operation of a microcomputer.
CPUspeed – speed of the CPU that data is retrieved and processed
CPUArchitecture –
CPUMicron –
CPUMfr – Manufacturer of the CPU

HD - Hard Drive; a device that stores programs and data that consists of a drive and one or more hard disks.
HDCapacity – the capacity or amount of data that can be stored on the hard drive
HDRPM – the speed at which the data is stored or retrieved on the hard drive
HDPin –
HDMfr – Manufacturer of the hard drive

MBRAMtype – is the type of RAM that the motherboard is compatible with
MBChipset – is a semiconductor that enables the processor and the memory with capabilities of multitasking and multimedia applications
MBCPUspeed – is the speed of the MB that has to be compatible with the speed of the CPU
MBMfr – Manufacturer of the motherboard

RAM - Random Access Memory; is volatile memory that stores programs and data that are being accessed by the user. This memory is erased when the power goes off.
RAMCapacity – is the capacity or the amount of storage space for this memory
RAMSpeed – is the speed at which the programs or data is stored or retrieved in the RAM
RAMType – the type of RAM; it is either SDRAM or DDR
RAMMfr – Manufacturer of the RAM

LatestDate is the latest date that the item will arrive at the company to be used in manufacturing

MB – Motherboard; is the primary circuit board in a computer that controls all of the elements that make up a computer system.