Agent Roles in Human Teams

Michael Lewis
School of Information Sciences
University of Pittsburgh
Pittsburgh, PA 15260, USA
001-412-624-9426
ml@sis.pitt.edu

Katia Sycara
School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213, USA
001-412-268-8825
katia@cs.cmu.edu

Terry Payne
Electronics and Computer Science
University of Southampton
Highfield, Southampton S017 1BJ, UK
440-238-059-8343
trp@ecs.soton.ac.uk

ABSTRACT
In this paper, we describe results of a series of experiments investigating the effects of agent aiding on human teams. The role an agent played, its task, and the ease with which it communicated with its human teammates all influenced team behavior. Team supporting tasks such as relaying and reminding seemed particularly effective.

Categories and Subject Descriptors
[**Agents**]:

General Terms
Performance, Design, Experimentation, Human Factors,

Keywords
Agents, Teams, Human-Agent Interaction

1. INTRODUCTION
The greatest impediment to assisting human users lies in communicating user intent to an agent and making the agent's results intelligible to the human. Today in almost all cases the limiting factor in human-agent interaction has become not computing cycles or connectivity (the machine side) but the user's ability and/or willingness to communicate his desires and sift, organize, and interpret the machine's response to satisfy them (the human side). The characteristics of increased flexibility and autonomy that make agents suitable to plan and execute tasks on behalf of human users also make monitoring and evaluating more difficult for the humans. For example, if you were to task an agent to book an inexpensive flight to Athens with a departure on Tuesday you should not be surprised to get back an itinerary with a 14 hour overnight layover in Memphis another in Warsaw and an arrival on Thursday. By the time you have enumerated your preferences in sufficient detail to have confidence in the agent's booking you might as well have gone online and booked it yourself. Except in cases where an agent's task performance is completely correct and deterministic, uncertainties as to agent progress in performing the task, alerting the user to potential failures, or protecting the user from unauthorized agent actions may need to be addressed for even the simplest interactions.

The degree of difficulty of these challenges varies with an agent's role. Some of these roles might be :

Supporting an individual team member by performing a subtask under the team member's management

Assuming the role of a (more or less) equal team member by :

performing the tasks appropriate to a human teammate

performing a human task in parallel to critique it

performing a human task subject to monitoring

Supporting the team as a whole.

While there are many other ways to categorize levels of automation (see for example Sheridan 1992 or Parasuraman et al. 2000) these seem representative of a range of roles proposed for agents.

The alternative to either the human or the agent performing an entire task is the subdivision of tasks between the human and the agent (or agents) to ensure high quality performance. In a scheduling task, for example, the agent might propose an initial schedule for the human to examine and suggest reordering based on knowledge about the team's goals not considered in the agent's computations. The agent could then prepare a new schedule incorporating the changes in priorities to be approved by the human and forwarded to the team. In this example the agent and human share task responsibility, with the agent providing algorithmic scheduling capabilities and the human supplying more detailed knowledge of team goals. Aiding individual tasks can improve team performance both by improving the individual's performance, in this case a better schedule, and by freeing the human's cognitive resources for teamwork.

To serve as a fullfledged teammate raises issues associated with communication and coordination among team members [2,3,5,9]. A software agent in this role must not only perform its assigned task but must use communications and modelling to share information, goals, and maintain its intelligibility to other team members. If the scheduling agent were promoted to team member status, for example, it would need a much more sophisticated model of the team's goals and interdependencies among its teammates' tasks in order to make the same adjustments to job priorities. Filling the role of a human teammate is extremely challenging for software agents because of the need to replicate the sorts of commonplace reasoning and ad hoc assistance we would expect of a human in the same role.

An intermediate step is to follow a management by consent/exception strategy by allowing the agent to perform the task but reserving approval/disapproval of the result. This approach frees the human from task performance but not monitoring. The complementary approach of intelligent critiquing presumes that expert task performance is very complicated and best performed by the human. However, because the human may overlook some aspect of the problem that would be accounted for in a much simpler model, the agent performs the task in parallel to point out discrepancies. This form of assistance has been applied most widely in the medical domain [4]. Both management by consent/exception and critiquing give some evidence of the agent's suitability to perform the task independently.

A third possibility is to support the team as a whole by facilitating communication, allocation of tasks, coordination among the human agents, and focus of attention. Issues here deal with how to support interactions among team members using agents [5], what kind of software agent architecture and processing allows agents to monitor team activity, access and distribute information and results of their reasoning to human team members that need them. Specifically, the focus is on how software agents could be used to support and promote teamwork along the dimensions identified by Cannon-Bowers and Salas [2]. Surprisingly, the task of supporting teamwork explicitly appears more amenable to agent assistance than that of incorporating teamwork into the performance of individual tasks. As part of the communications infrastructure, a software agent can initiate searches for supporting and related information or facilitate passing information to appropriate teammates without the sophistication of modelling needed to fill a human role.

Our research has investigated each of these roles for agents using a time critical synthetic radar task and a deliberative path planning task . Both tasks were designed to require team work and cooperation to be completed successfully. In this paper we will introduce both tasks then examine our results as they relate to agent roles.

2. TANDEM SYNTHETIC RADAR TASK

Two experiments used a moderate fidelity simulation (TANDEM) of a target identification task, jointly developed at the Naval Air Warfare Center-Training Systems Division and the University of Central Florida and modified for these experiments. The TANDEM simulation was developed under the TADMUS (tactical decision making under stress) program of the US Office of Naval Research and simulates cognitive characteristics of tasks performed in the command information center (CIC) of an Aegis missile cruiser. Figure 1 shows a typical TANDEM display. Information about the hooked target (highlighted asterisk) is obtained from the pull-down menus 'A','B', and 'C'. The cognitive aspects of the Aegis command and control tasks which are captured include time stress, memory loading, data

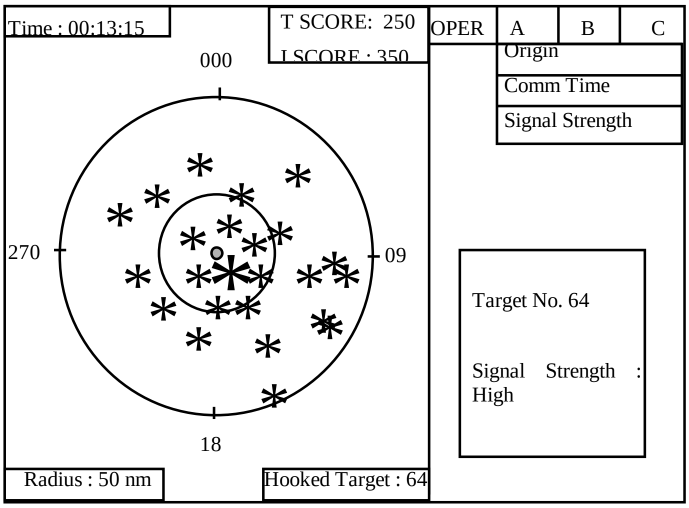

Figure 1. Tandem Display

aggregation for decision making and the need to rely on and cooperate with other team members (team mode) to successfully perform the task. The more highly skilled tasks of the individual team members that involved extracting and interpreting information from radar, sonar, and intelligence displays is not modeled in the simulation. Instead of interpreting displayed signals to acquire diagnostic information about targets, TANDEM participants access this information manually from menus. In accessing new information, old information is cleared from the display creating the memory load of simultaneously maintaining up to 5 parameter values and their interpretation.

In the TANDEM task subjects must identify and take action on a large number of targets (high workload) and are awarded points for correctly identifying the targets (type, intent, and threat). and taking the correct action (clear or shoot). A maximum of 100 points is awarded per target for correct identification and correct action. Users "hook" a target on their screen by left-clicking on the target or selecting "hook" from a menu and specifying a target's unique contact number. Only after a target is hooked can they access information relative to that target. In team configuration TANDEM consists of three networked pc's each providing access through menus to five parameters relative to a "hooked" target. Their tasks involve identifying the type of contact (submarine, surface, or aircraft), its classification (military or civilian), and its intent (peaceful or hostile). Each of these decisions is made at a different control station and depends on five distinct parameter values, only three of which are available at that station. Subjects therefore must communicate among themselves to exchange parameter values to classify the target. If the team finds a target to be hostile it is shot, otherwise it is cleared and the team moves on to another target.

In standalone mode all of the information is made available on a single pc with the station specific parameters accessed using three distinct menus in unaided conditions. Menus in standalone mode present 5 parameters each. In team mode the three menus present 3 (overlapping among team members) parameters per menu. Just as TANDEM simulates cognitive aspects of the Aegis missile command and control task in team mode, it provides a context to simulate the gathering, aggregation, and presentation of communications, command, control and intelligence information by intelligent agents in standalone mode. Our experiments explored both human-agent dyads in which the agent provided assistance for one of the classifications and three person teams for which agents provided individual or team directed assistance.

3. MOKSAF PATH PLANNING TASK

Typically, human decision-makers, particularly military commanders, face time pressures and an environment where changes may occur in the task, division of labor, and allocation of resources. Information such as terrain characteristics, location and capabilities of enemy forces, direct objectives and doctrinal constraints must all play a part in the commander's decisions. Software agents have privileged access to the masses of information in the digital infosphere and can plan, criticize, and predict consequences from this sea of information with greater accuracy and finer granularity than a human commander could. Information within this infosphere can be used for data fusion, "what-if" simulations, or visualized to provide situation awareness. There is also, however, information that may not be explicitly represented electronically and is therefore inaccessible to software agents. Such information includes intangible or multiple objectives involving morale, the political impact of actions (or inaction), intangible constraints, and the symbolic importance of different actions or objectives. Before agents can consider information that is outside their infosphere, this information must be re-expressed in agent-accessible terms.. Military commanders, like other professional decision-makers, have vast experiential information that is not easily quantifiable. Commanders must deal with idiosyncratic and situation-specific factors such as non-quantified information, complex or vaguely specified mission objectives and dynamically changing situations (e.g., incomplete/changing/ new information, obstacles, and enemy actions). In order to cooperate with software agents in planning tasks commanders must find ways to translate these intangible constraints into tangible ones their agents can understand. The issue therefore becomes how should software agents interact with human teams to assist with problems which may be vague, ill-specified, with multi-attribute goals.

We have developed a computer-based simulation called *MokSAF* to evaluate how humans can interact and obtain assistance from agents within a team environment. *MokSAF* is a simplified version of a virtual battlefield simulation called ModSAF (modular semi-automated forces). *MokSAF* allows two or more

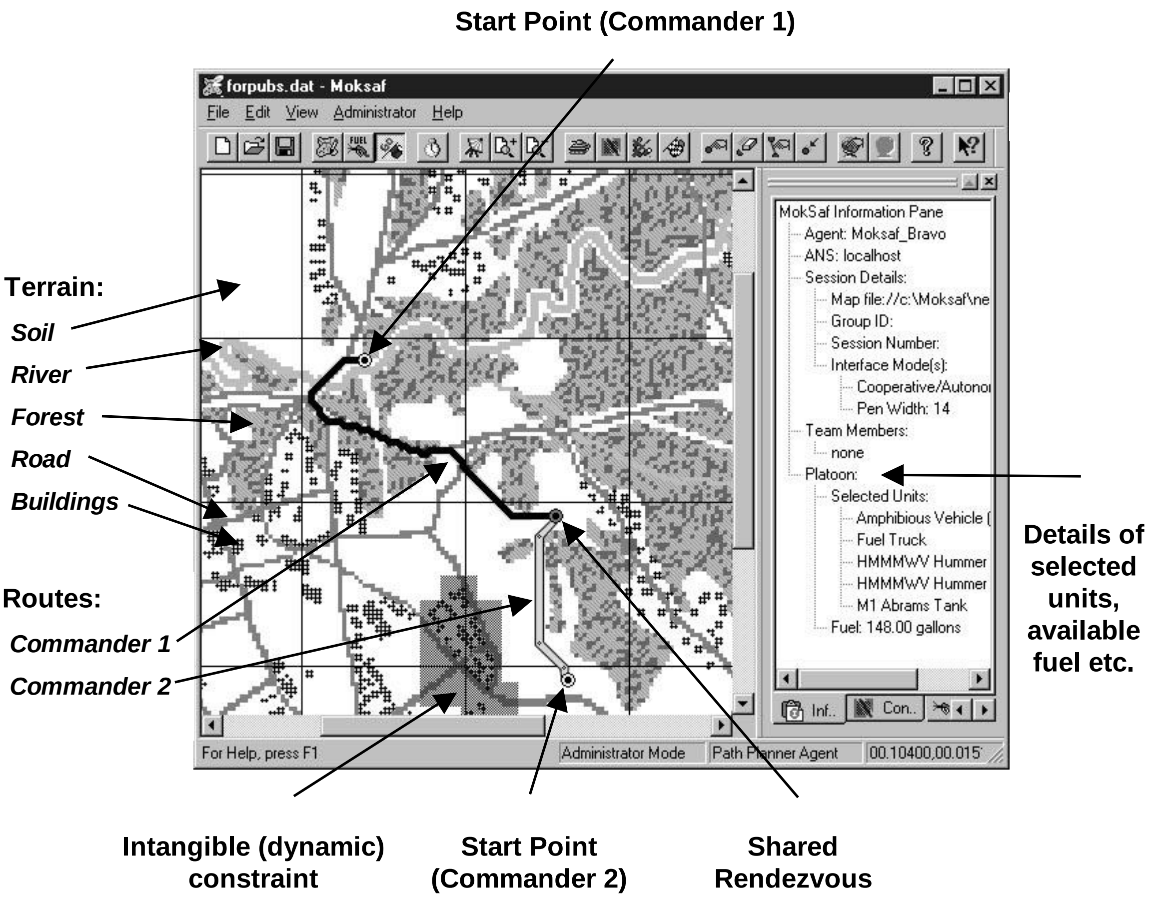

Figure 2: MokSAF display

commanders to interact with one another to plan routes over a particular terrain. Each commander is tasked with planning a route from a starting point to a rendezvous point by a certain time. The individual commanders must then evaluate their plans from a team perspective and iteratively modify these plans until an acceptable team solution is developed. One of the interface agents used within the *MokSAF* Environment is illustrated in Figure 9. This agent presents a terrain map, a toolbar, and details of the team plan. The terrains displayed on the map include soil (plain areas), roads (solid lines), freeways (thicker lines), buildings (black dots), rivers and forests. The rendezvous point is represented as a red circle and the start point as a yellow circle on the terrain map. As participants create routes with the help of a *route-planning agent* (see below), the routes are shown in bright green. The second route shown is from another *MokSAF* commander who has agreed to share his planned route. The partially transparent rectangles represent intangible constraints that the user has drawn on the terrain map. These indicate which areas should be avoided when determining a route.

4. AGENTS AS TEAMMATES

We have conditions in both the TANDEM and MokSAF studies in which agents took one of the modified (managed or critiquing) teammate roles The first TANDEM study investigated different forms of information aggregation and integration using a single workstation for which the agent assisted in one of the three decision tasks.

4.1 Teammates in TANDEM

Agents presented either :

1) aggregated information (*list*) -- a list of parameters and values

2) integrated information (*table*) -- a table showing categorized values

3) synthesized information (*oracle*) -- target type assignment with certainty factor.

Agent presented information was subject to errors of several forms. Although more precise monitoring of performance was possible for the list and table conditions, radar operators showed their best performance when working with the oracle which functioned as a teammate rather than a subtask assistant by performing the full decision task. This effect was found for both the aided classification task (air/surface/sub) and the unaided (civilian/military) decision.

4.2 Teammates in MokSAF

Three different *route-planning agents* (*RPA*) were developed to interact with the human team members in the planning task. The first agent, the *Autonomous RPA*, performs the routing task itself. This agent acts like a "black box." The agent creates the route using its knowledge of the physical terrain and an artificial intelligence planning algorithm that seeks to find the shortest path. The agent is only aware of physical constraints, which are defined by the terrain map and the platoon composition, and intangible constraints, which are graphically specified by the commanders.

The second agent, the *Cooperative RPA*, analyzes routes through a corridor drawn by the human team members, selects the optimal route and helps them to refine their plans. In this mode,

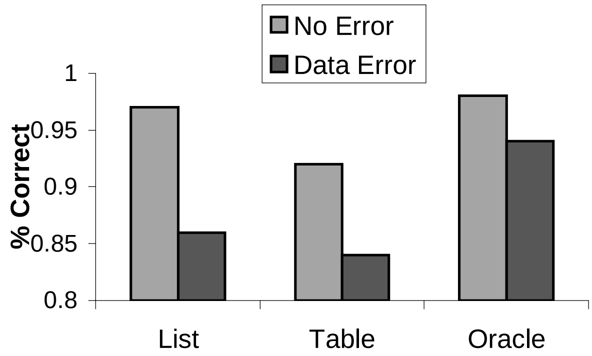

Figure 3. Accuracy for the aided (air/surface/sub) decision

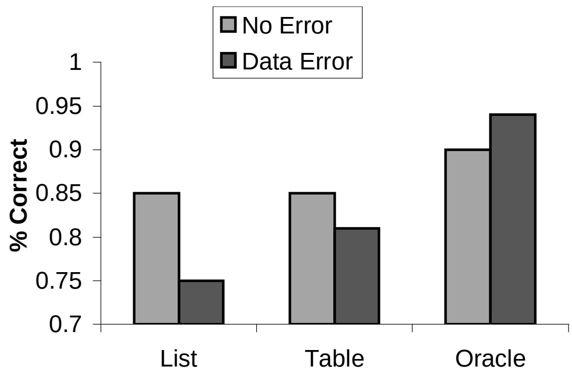

Figure 4. Accuracy for the unaided (civilian/military) decision

the human and agent work jointly to solve the problem (e.g. plan a route to a rendezvous point). The workload should be distributed such that each component matched to its strengths. Thus, the commander, who has a privileged understanding of the intangible constraints and utilities associated with the mission, can direct the route around these constraints as desired. However, the commander may not have detailed knowledge about the terrain, and so the agent can indicate where the path is sub-optimal due to violations of local physical constraints such as traversing swamp or wooded areas.

The third condition, the *Naïve RPA* (or control), provides minimal assistance to the human commanders in their task of drawing and refining routes. Using this RPA, the commander draws a route that the agent then critiques for constraint violations such as impassible terrain or insufficient fuel. The commander is allowed to iteratively alter his failed route until a plan is found which passes muster. All three RPAs are intended to be used for iterative cooperative refinement of routes and the task of coordinating with other commanders requires continuous replanning as the team searches for its own best solution..

4.3 Experimental Methodology

The *MokSAF* experiments examine a deliberative, iterative and flexible planning task. There are three commanders (Alpha, Bravo and Charlie), each with a different starting point but the same rendezvous point. Each commander selects units for his/her platoon from a list of available units. This list currently contains M60A3 tanks, M109A2 artillery units, M1 Abrams tanks, AAV-7 amphibious assault vehicles, HMMWVs (i.e., hummers), ambulances, combat engineer units, fuel trucks and dismounted infantry. This list can be easily modified to add or delete unit types. With the help of an RPA, each commander plans a route from his starting point to the rendezvous point for the specified forces.

Once a commander is satisfied with the individual plan, she can share it with the other commanders and resolve any conflicts. Conflicts could arise due to several issues including shared routes and/or resources or the inability of a commander to reach the rendezvous point at the specified time. The commanders also must coordinate regarding the number and types of vehicles they can take to the rendezvous because their mission specifies the number and composition of forces needed at the rendezvous point. Commanders were additionally instructed not to plan routes that took them on the same paths as any other commander which required them to coordinate routes to avoid shared paths.

Data was examined from two critical points in the session – the time that individuals first shared their individual routes (first share) and at the end of the 15 minute session (final). Overall, we found that the *Autonomous RPA* and *Cooperative RPA* achieved

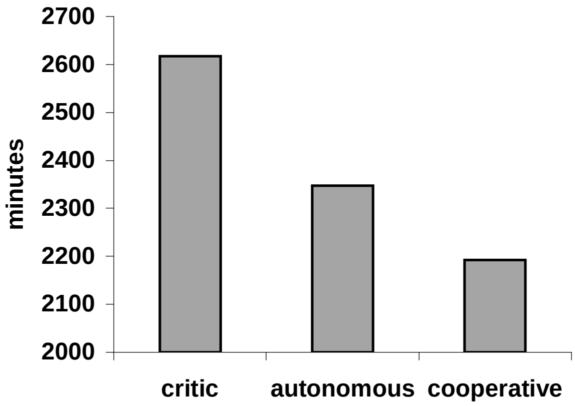

Figure 5. Path lengths were shorter for active agents

lower cost paths, earlier rendezvous, and lower fuel usage than the RPA critic. These results held true both for the team as a whole and for individual participants. It was expected that path lengths between the first time a route was shared and at the end of a trial would vary due to issues related to conflict resolution among the teammates. Participants in the active conditions managed to maintain the quality of their plans despite the modifications and replanning needed to coordinate with other team members. Although the RPA agents did not support teamwork directly, their assistance for the individual planning task allowed the commanders to find new routes as short as the ones they abandoned. Unaided commanders by contrast were forced to resort to longer paths in order to accommodate the requirements of coordinating with their team.

In its current form, the active conditions, *Autonomous RPA* and *Cooperative RPA* have been shown to provide a better interface for both individual route planning and team-based re-planning. Despite this clear superiority over the passive condition (*Critic RPA*), participants in the *Autonomous RPA* group frequently expressed frustration with the indirection required to arrange constraints in the ways needed to steer the agent's behavior and often remarked that they wished they could "just draw the route by hand".

Comments on the *Critic RPA* focused more closely on the minutiae of interaction. In its current form, the user "draws" a route on the *interface agent* by specifying a sequence of points at the resolution of the terrain database A route is built up incrementally by piecing together a long sequence of

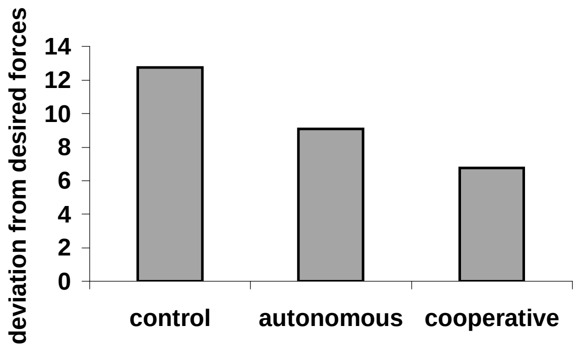

Figure 6. Deviation in force composition was less for active agents

such segments. Although tools are provided for deleting unwanted points and moving control points, the process of manually constructing a long route is both tedious and error prone. While the Critic RPA gratuitously forced the human to deal with physical constraints to which it already had direct access, the Autonomous RPA also diverted the user from the conceptual task of choosing and coordinating routes to that of representing intangible constraints. We believe this conceptual incongruity between the route planning and representation tasks left Autonomous RPA subjects unprimed for route planning related communications and coordination. Although the quality (path length/fuel usage) of routes between these two groups was very similar Cooperative RPA teams were better able to coordinate the composition of their forces and deal with deconflicting routes. These results emphasize the importance of designing human-agent interactions that promote direct interaction with the problem domain rather than focusing on information needs of the automation.

5. AIDING TEAMWORK

The second team TANDEM study examined different ways of deploying machine agents to support multi-person teams: 1) supporting the individual (within a team context) by keeping track of the information he has collected and helping the individual with his task and with passing information to teammates (Individual Clipboard, Figure 5b); 2) supporting communication among team members by automatically passing information to the relevant person which should reduce communication errors and facilitate individual classification (Team Clipboard, Figure 5c); and 3) supporting task prioritization and coordination by providing a shared checklist of which team member had access to which data (Team Checklist, Figure 5d). We hypothesized that the Individual Agent should aid the individual task and aid communication among team members. This agent shows all data items available to an individual team member (in this case, ALPHA) and fills in the values for the data items as the subject selects them from them from the menu. The values under the TYPE heading assist the individual with his task while the other team members may need to request the remaining values. The Team Clipboard Agent should also aid the individual task and aid team communication to a greater degree than the Individual Agent. This agent aggregates values from all members of the team to help the individual with his/her task. It automatically passes values as they are selected from a menu to the appropriate team member. Thus, when altitude/depth is selected from some one else's menu, it is passed to an individual team member (ALPHA) who can use it to make the type identification. We hypothesized that this agent should reduce verbal communication among team members and reduce communication errors. The third agent, Team Checklist, should aid team coordination. This agent shows who has access to what data. For example, all three team members (ALPHA, BRAVO, CHARLIE) have access to speed, but only BRAVO has access to "Intelligence". The final condition is a control where we observed team performance without the aid of any machine agent. This is the standard TANDEM task described in Smith-Jentsch, et al. [8]. The goal of the study is to examine the impact of the aiding alternatives on: 1) communication patterns, 2) data gathering strategies, 3) reliance (i.e., use of) on the agents, and 4) performance.

Teams of three subjects were recruited for this study. Each team was assigned to one of four conditions: 1) control, 2) individual agent, 3) team clipboard agent, or 4) team checklist agent. TANDEM was used with three-person teams, each member with a different identification task to perform (air/surface/submarine, military/civilian, and peaceful/hostile). One person was assigned to ALPHA, one to BRAVO and one to CHARLIE. ALPHA, BRAVO and CHARLIE had different items on their menus and different tasks during the trials. ALPHA identified the type of target (air, surface or submarine); BRAVO determined whether the target was civilian or military; CHARLIE determined whether the target was peaceful or hostile. In addition, CHARLIE acted as the leader by indicating the type, classification and intent of each target to the system and taking the final action (to shoot or clear).

There were five pieces of information for each identification task, three of which must agree in order to make a positive identification. These pieces of information were distributed among the three team members. Each team member saw different data items on the menus and had three data items required for his/her identification task and several other items that the other team members might need to complete their tasks. Thus, the subjects needed to communicate with one another to perform their tasks for roughly two-thirds of the targets. All five pieces of information might agree for a particular target, however, in many cases, the ambiguity of the data was manipulated such that only three pieces agreed.

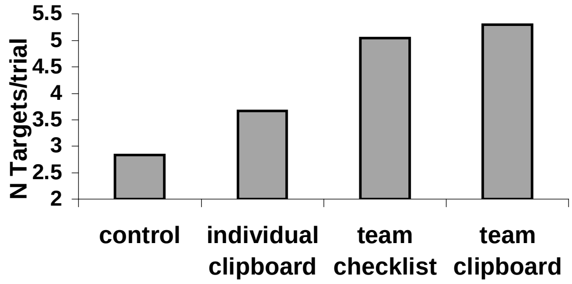

Figure 7. Aiding teamwork improved performance

In this study aiding teamwork directly (team clipboard/checklist) appeared more effective than supporting team members at their individual tasks despite the reductions in memory load and ready accessibility to parameters for sharing provided by the individual clipboard. The potential for coordinating human-human interactions through agent systems seems a particularly promising approach because of the high payoff and the reusable and largely domain independent character of the team supporting tasks.

6. DISCUSSION

Researchers in the information agent area have conventionally focused on information access, while those in the personal associate tradition [1] have concentrated on monitoring (and intervention). Our research suggests that there may be fruitful applications of agent technology to communications and planning as well.

The Tandem experiments illustrate the extent to which assistance in information exchange can have a major impact particularly in real-time high workload tasks. As we come to rely more and more on ad hoc interlocking teams made possible by the increasing interconnection and digitalization of the battlefield there will be fewer opportunities for team training and co-adaptation shown to be essential to high performance teams. For newly formed teams, such as our Tandem subjects, assistance in selecting and directing communications provided major benefits. The agent's role of identifying salient changes in the tactical picture as humans uncovered them and relaying these updates to the appropriate teammates is a new role for software agents. Rather than acting as personal agents assisting the information recipients by seeking this information (pull) or a personal agent assisting the information provider in finding information, the agent supports distribution (push) of their products The closely related dimension of "communication" which characterizes the interpretability of communicated messages becomes a matter of good information presentation (agent-to-human) and design of effective interaction techniques (human-to-agent). Our findings highlight the importance of designing interactions that focus attention on the human's goals and problem domain rather than the agent's information needs. This need to tailor human-agent interactions to tasks and domain suggests that intelligent agents are likely to add to rather than detract from efforts needed in human-computer interface design. As software agents become more common in human teams we expect monitoring, correction, and intervention to become more acceptable but they are likely to be the last capabilities to be introduced into successful systems.. Our results suggest that software agents are well suited for this task. Because the domain independence of teamwork agents would allow them to be rapidly deployed across a broad range of tasks and settings teamwork appears to be a particularly high payoff area for further agent research.

ACKNOWLEDGMENTS

This research was supported by an Office of Naval Research grant N-00014-96-1-1222. Some of the studies reported were conducted as part of Terri Lenox's dissertation research.

7. REFERENCES

[1] Banks, S. & Lizza, C. (1991). Pilot's Associate: A cooperative, knowledge-based system application, IEEE Intelligent Systems and Their Applications, 6(3), pp. 18-29.

[2] Cannon-Bowers, J. A., & Salas, E. (Eds.). (1998). Making decisions under stress: Implications for individual and team training. Washington, DC: APA.

[3] Grosz, B. and Kraus, S. (1996). Collaborative Plans for Complex Group Action. Artificial Intelligence journal, 86(2), pp 269-357.

[4] Guerlain, S., Smith, P.J., Obradovich, J. Heintz, Rudmann, S., Strohm, P. Smith, J.W., Svirbely, J., and Sachs, L. (1999).Interactive Critiquing as a Form of Decision Support: An Empirical Evaluation. Human Factors 41, 72-89.

[5] Lenox, T., Lewis, M. Roth, E., Shern, R., Roberts, L., Rafalski, T., and Jacobson, J. (1998). Support of teamwork in human-agent teams. Proceedings of IEEE International Conference on Systems, Man, and Cybernetics, 1998, pp. 1341-1346.

[6] Parasuraman, R., Sheridan, T. B., & Wickens, C. D. (2000). A model for types and levels of human interaction with automation. IEEE Transactions on Systems, Man, and Cybernetics–Part A: Systems and Humans, 30(3), 286–297.

[7] Sheridan, T.B. (1992). *Telerobotics, Automation, and Human Supervisory Control*. Cambridge, MA: MIT Press.

[8] Smith-Jentsch, K. Johnston, J. H., and Payne, S. (1998). Measuring team-related expertise in complex environments. In J. A. Cannon-Bowers and E. Salas (eds.), Decision Making Under Stress: Implications for Individual and Team Training. Washington, DC: American Psychological Association.

[9] Tambe, M. and W. Zhang, W. (1998). Towards flexible teamwork in persistent teams, In: Proceedings of the International Conference on Multi-Agent Systems (ICMAS), 1998.