Modeling Humans for Coordinating Distributed Human-Agent Teams in Space Operations

Debra Schreckenghost, Cheryl Martin, Tod Milam, and R. Peter Bonasso
NASA Johnson Space Center, TRACLabs
MS: ER2, 2101 NASA Road Ones, Houston, TX 77058
ghost@ieee.org, cmartin@traclabs.com, tmilam@traclabs.com, r.p.bonasso@jsc.nasa.gov

1. Introduction

Current manned space operations consist of crew members performing predefined procedures assisted by flight controllers located in Mission Control at Johnson Space Center (JSC). For future operations such as space exploration, where missions are extended and communication delays may be a factor, we are investigating new concepts of operations. Specifically we are developing software agents to assist both crew and flight controllers in their duties. We have developed control agents for crew and vehicles system that reduce human workload by automating routine operations such as reconfiguration and fault management. We also have developed human support agents that aid in distributing operations. These agents relieve humans of vigilant data monitoring by automatically drawing human attention to important events when they occur. They allow flight controllers to perform some operations from their offices. And they increase crew autonomy by providing services such as event notification, procedure tracking, and schedule management that reduce the need for close tending by flight controllers. Taken together these humans and agents form an operational team.

To coordinate effectively, team members need to be aware of each other. This includes knowing where team members are located, what activities they are performing, what roles they fulfill, and if they are healthy. This information is used both as a context for communication as well as a means of coordinating activity. Tracking human state can be difficult, however, because you often are reasoning with incomplete or incorrect knowledge. Humans can be difficult to sense because you must rely on sensing the effects of their actions and the environment they occupy and manipulate often is poorly instrumented. Humans also are difficult to model because they tend to do things in different ways over time and they can commit a variety of errors.

We have developed the Distributed Collaboration and Interaction (DCI) System to assist humans and automated control agents working together in support of distributed space operations [1]. The DCI system includes Ariel agents that observe their human users' location and activities for the purposes of team coordination. In this paper we describe these software agents. We discuss our approach for modeling the human in these software agents and how we use this model to support coordination among the human-agent team. We then describe our experience in using these agents with a life support application at JSC.

2. Agents for Coordinating Teams

The DCI System provides an environment for collaboration and interaction among humans and automated control agents. Within this environment, each human in the group has an Ariel agent to assist him or her in working with other members of the team. The Ariel agent provides services that are integral to the agent. Other capabilities are provided in the environment that are external to the Ariel agents but are integrated with them and exchange information with them.

The following services are provided by the Ariel agent

- *Notification Service*: filters & annotates incoming notices based on the user's assigned roles.
- *Location Service*: receives user logins/logouts and wireless RF tracking signal strengths and translates them user location changes.
- *Task Status Service*: tracks completion status of the user's activities and transmits changes to the Conversion Assistant for Planning.
- *Command and Authorization Service*: works with Command and Authorization Manager to prevent concurrent activities from interfering with each other by authorizing human activities only if the requested activity does not conflict with ongoing activities. In conjunction with the Assistant for Commanding, it also reconfigures the automated control agents to ensure compliance with the authorization granted.
- *User Interface Service*: manages the presentation of all information to the user, including displays, paging, and sending of emails.
- *State Management Service/State Data Service*: manages the user's state model, makes that state available to other services, and ensures that updates to that state are handled consistently.

The following set of capabilities external to the Ariel agent are available in the DCI environment:

- *Activity Planner (AP):* builds and updates the daily duty schedules for all humans in team
- *Conversion Assistant for Planning (CAP)*: manages plan monitoring and execution for the Activity Planner, including both environmental monitoring and information exchange between the Ariel agents and the Activity Planner.
- *Event Detection Service (EDA):* computes Caution & Warning events and failure impacts using simple pattern-matching software.
- *Complex Event Recognition Service (CERA)*: summarizes operational situations that consist of sets of events that are hierarchical and temporal in nature. CERA was developed by I/Net.

The DCI system has been built using Java 1.4.2, Allegro Lisp 6.1, and CORBA ORB implementations JacORB 1.4.1 and ILU 2.0b1.

3. Modeling the Human Agent

We model the human agent and update this model agent team. We use these models to (1) notify team members based on their roles, location, and individual preferences, (2) track human activities as part of coordinating the group using a centralized plan, and (3) maintain group awareness among a distributed team. In this section we describe our model of the human and how we use it to coordinate the human-agent team.

3.1. The Model Description

Our model for the human has these characteristics:

- *Roles*: the operational responsibilities the human fulfills within the team. Our model for roles includes a descriptor (Prime, Backup, Coordinator) and a domain (e.g., life support). The Prime is responsible to handle all problems in the life support system that the automated control agent cannot handle. The Backup is called in when the Prime is unavailable. Both these roles rotate weekly. The Coordinator is knowledgeable about the life support domain. This role does not rotate.
- *Health*: the person's physical state. Our model for health identifies whether the person is sick or well.
- *Location*: the locale of the human. We define this as (1) *user presence* defining whether the user has logged into his or her agent (i.e., online or offline), (2) a mapping onto a physical ontology, and (3) the list of network addresses into which the user has been logged (called *cyber-location*).
- *Activities*: the duties a person performs for his/her assigned roles. We model activities in the agent for two purposes: to track the completion status as a part of plan execution monitoring and to display these activities to the user in a daily schedule. Our model for plan execution monitoring for each task that is ready to execute is an activity name, a unique identifier, activity start and end time, assigned agent, and completion status value. We also provide a compilation of activity changes to the agent for building the daily schedule.

The roles and health are not sensed. These states cannot be changed by the user, but are managed by the organization. The health and the roles can be changed by lead personnel within each group, such at the Coordinator in our role ontology. These organizational states are used for schedule management and group awareness.

The Location Service of the Ariel agent records network address, host name, and platform type from logins into the DCI system. It maps network address to a physical location. Since the user can log into the agent from multiple locations, the most recent login is considered to be the current location. When the user logs out, we decay our confidence in the user's location based on the time since the logout. As time passes, we move up the ontology, becoming more general in our location mapping. For example, we move from the WRF to Building 7 that contains the WRF to NASA where Building 7 is located. We also map machine logins to user presence. This ontology lets us distinguish when the user is both online or offline and nearby or remote. We use this ontology to determine how to notify users (e.g., whether to page a user or notify using the agent GUI).

The Task Status Service (TSS) of the Ariel agent tracks the completion status of user activities. It determines when tasks are initiated, completed, or when they are overcome by events such that they cannot be completed. When a status changes, it passes this to the CAP to support plan execution monitoring. The TSS can have a unique strategy for tracking the progress of each activity. When human activities are computer-mediated, they are observable to the Ariel agent. In some cases, however, human activities are conducted by direct manipulation of physical devices. Such *manual activities* can be difficult for agents to observe. Strategies we have used to track manual activities include monitoring for direct evidence of manipulation in sensed data (e.g., valve closed), indirect evidence such as observing the human changing to the location where the activity will be performed, or asking the human to acknowledge the activity. We also have combined strategies, such as observing the human moving to the expected location and observing her using the software tool needed to perform the task. When tasks are critical, we often require user acknowledgement when the task is assigned. If the user fails to acknowledge within a reasonable amount of time, we send increasingly more salient requests. When tasks are less critical, we often assume they are done when the time for completion passes and let the user inform the agent otherwise at a later time (using a strategy called *plan reconciliation*).

Plan reconciliation permits the user to update task completion assessments made by the Ariel agent. The user can change the task status set by the TSS at any time during the day. Options are simple: (1) *complete*: goal achieved and the task is not considered for replanning, and (2) *not-complete*: goal not achieved and thus is considered for replanning. The task may be replanned immediately if the TSS considers it to be in progress when the status is changed by the user. Otherwise, the task will be reconsidered at the next plan update.

3.2. Use of the Model

The Ariel agent uses the model of the human to aid him in performing his duties. It notifies its user when events occur that are relevant to his roles and determines how to present those notices depending upon his location. It tracks his activities and provides that information to an automated planner that manages his daily schedule. And it publishes his progress on activities and his location to other team members to support team awareness. We describe our approach for each of these below.

Notification**:** The Notification Service within the Ariel agent filters and annotates incoming notices based on the roles assigned to its user. The annotation defines how the notice will be presented to the user (pager, email, display) and how much urgency and emphasis will be associated with the presentation of the message (modeled as latency and focus of attention). Determining how the notice is presented takes into account the user presence at the time the notice is delivered. If the user is offline, we use presentation media such as pagers or email; if the user is online, we use media such as the agent display. We define notice specifications for the roles within the team as the conditions for filtering and the directives that implement the annotations. When a user's roles change, the notice specifications loaded for that user automatically update as well. Most notice specifications are managed by the organization, and cannot be changed by the individual. We can encode personal preferences that do not override the organizational requirements.

Team Coordination: One way the activities of the humans among the team are coordinated is by means of a centralized plan. In the future we plan to extend this work to model the activities of both humans and automated control agents in one plan. This plan is generated using an automated planner. The CAP performs many of the plan execution and monitoring functions, translating from the different users' perspectives as represented by their Ariel agents to the perspective of the automated planner. It transmits new activities to each Ariel agent when a new plan is built. All activity changes are sent to the User Interface Service that builds the display of the user's schedule. If activities are ready to execute, they are sent to the TSS in the Ariel agent. The TSS assesses the activity's completion status, stores it in the human model, and transmits changes to the CAP. The CAP translates this into terms the automated planner can use (e.g., "overcome by events" maps to "fail"). The CAP monitors for changes in the operational situation and user state indicating that the assumptions or constraints under which the current plan was created have changed, requiring that a new plan be created. Examples of such changes include a change in user health, a failure of equipment requiring a repair activity, or a waiver of a flight rule requiring daily exercise. The CAP also monitors the availability of the human agent for activities, and when this availability changes, initiates replanning. A person is set temporarily unavailable when he or she fails to respond to a critical activity. Finally, the CAP is responsible to maintain the continuity of activities across planning. Thus, if a replan results in a new identifier for a former activity with the same goal, the CAP will unify the two activity data structures. This avoids confusing the user with an apparent "new" activity for what is essentially the same task.

Team Awareness*:* The physical distribution of the operational team poses challenges for maintaining awareness of other team members. Even in a relatively small facility like the International Space Station, astronauts find it difficult to know what other team members are doing. To address this difficulty, the Ariel agents collect information about other users' state changes to permit generating a team awareness display. This display shows the current role assignments, physical location, user presence, and schedule for all members of the group. It also provides a history of changes for each of these user states for all agents in the team.

4. Life Support Application

We are evaluating the use of the DCI system in the Water Research Facility (WRF) at JSC. We are using it to assist control engineers in performing duties associated with crew water recovery systems. Specifically we are aiding interaction with the Post Processing System (PPS) that is being evaluated in ground testing. The PPS is a water "polisher" that removes the trace inorganic wastes and ammonium in recycled water using a series of ion exchange beds and removes the trace organic carbons using a series of ultraviolet lamps. This is a good analog to crew activities with life support systems in space because, like the crew, control engineers do other duties most of the time, but are responsible to handle periodic PPS maintenance and any PPS anomalies that arise.

The PPS is controlled by the 3T automated control software [2]. The 3T architecture consists of three tiers of parallel control processing:

- *Deliberative Planner*. hierarchical task net planner to manage activities with resources or temporal constraints, or requiring multi-agent coordination,
- *Reactive Sequencer*. reactive planner to encode operational procedures that can be dynamically constructed based on situational context, and
- *Skill Manager*. layer for closed loop controllers.

This approach is designed to handle the uncertainty inherent in complex domains. Control commands flow down the hierarchy and feedback flows back up the hierarchy to close the control loop at all tiers. If a command fails at any level, it can initiate a repair action (e.g., replanning at the deliberative level, selection of an alternative sequence at the reactive level). Each layer operates at a different time constant, allowing high speed controllers at the lower level to operate in parallel with the slower, deliberative algorithms at the high level. The 3T architecture has been used extensively during ground tests of crew life support systems at JSC.

5. Conclusions

The Ariel agent in the DCI System builds a model of its user and uses this model to aid coordination with other members of the human agent team. The Location Service in the Ariel agent currently uses simple observations based on machine logins and logouts for tracking human location. Our approach to location tracking has been sufficient for tracking control engineers during business hours because it is based on logins to computers and engineers' duties require significant time spent using computers. Additionally, when not logged into the DCI system – when "offline" - we can still maintain contact by means of other media, such as pagers or email. Using these simple observations we have proven out our concept of personal agents to assist team coordination. We have demonstrated that modeling user presence (i.e., online/offline) and using this to determine how to present notices (e.g., whether to page or display) is effective.

These simple location observations are not adequate, however, for accurately tracking users in a deployed system. We plan to look at keyboard monitoring in the future as a means of ascertaining which computer the user is currently using and thus providing one means of detecting current location. We also plan to evaluate wireless radio frequency tracking for location tracking because it promises to provide a low-cost, ubiquitous source of location information. In this approach, a user's mobile computing platform measures signal strength from all access points bridging the wired and wireless networks visible to it, signal strength measurements taken by the handheld are then matched to a signal strength map, and the user's location is considered to be the strongest signal or closest access point. There are issues in using wireless radio frequent tracking, however. For example, mobile platforms are typically turned off when being transported and thus cannot communicate with the access points to gather necessary measurements. So measurements may be sparse from this source. Related projects such as the Electric Elves [3] have used the Global Positioning System (GPS) for location tracking. GPS is useful for tracking to a building, but not within a building at JSC.

We have anecdotal evidence that our approach of using an automated planner to assign life support anomaly handling responsibility based on agent tracking, including reassignment to backup personnel, can reduce time to recover from anomalies over the previous approach of coordination by phone. This centralized approach also is a good match for managing astronaut activities, which are planned in advance to ensure good use of crew time. Our use of a centralized automated planner to coordinate human activities differs from the use of distributed coordination approaches used among some multi-agent systems. For example, the Friday agent in the Electric Elves Project used a distributed auction approach to determine role assignments for its user [3].

The use of preplanned duty schedules in space operations means that the human's expected activities are known by the agent. This can simplify activity tracking to some extent by constraining it, although humans can depart from scheduled activities under both nominal and contingency situations. The Task Status Service in the Ariel agent makes use of a heterogeneous set of strategies for activity tracking. Where possible we monitor for direct evidence of state changes in sensed data resulting from human manipulation during the activity (e.g., handling a loss of communication anomaly). This approach is constrained by the fact that space environments are typically under-sensored due to cost and mass-to-space limitations. User acknowledgement is also effective when used sparingly in contingency situations. Based on our experience in the WRF, we are implementing the capability for engineers to accept or deny a task assignment via a pager for situations where they cannot get online quickly (e.g., a meeting, in a car). The use of indirect evidence, such as observing the human changing to the location where the activity will be performed, benefits from strategies that synthesize or fuse multiple, different observations to increase the confidence in the combined observations (e.g., move to WRF and launch the correct procedure). We have not yet evaluated the strategy of assuming an activity is successful when it's completion time passes and letting the human reconcile any differences later.

6. Acknowledgement

We acknowledge the support of Dr Michael Shafto, manager of Human-Centered Computing in NASA's Intelligent Systems Program, who sponsored this work.

7. References

[1] Schreckenghost, D., Martin, C., Bonasso, P., Kortenkamp, D., Milam, T., and Thronesbery, C. Supporting group interaction among humans and autonomous agents. *Connection Science*. 14(4) 2002. pp. 361-369.

[2] Bonasso, P., Firby, J., Gat, E., Kortenkamp, D., Miller, D, and Slack, M. 1997a. Experiences with an Architecture for Intelligent, Reactive Agents. *Journal of Experimental Theory of Artificial Intelligence*. 9: 237-256

[3] Scerri, P., David V. Pynadath, Milind Tambe: Why the elf acted autonomously: towards a theory of adjustable autonomy. AAMAS 2002: 857-864