

## Multiagent Adjustable Autonomy Framework (MAAF) for Multi-Robot, Multi-Human Teams

Amos Freedy, Onur Sert, Elan Freedy  
James McDonough, Gershon Weltman  
Perceptronics Solutions, Inc.  
[afreedy@perc solutions.com](mailto:afreedy@perc solutions.com)

Milind Tambe, Tapan Gupta  
USC Computer Science Department  
[tambe@usc.edu](mailto:tambe@usc.edu)  
William Grayson, Pedro Cabrera  
SAIC Orlando  
[William.S.Grayson@saic.com](mailto:William.S.Grayson@saic.com)

### ABSTRACT

*This paper describes the ongoing development of a Multiagent Adjustable Autonomy Framework (MAAF) for multi-robot, multi-human teams performing tactical maneuvers. The challenge being addressed in this SBIR Phase I R&D project is how to exploit fully the unique capabilities of heterogeneous teams composed of a mixture of Robots, Agents or Persons (RAPs): that is, how to improve the safety, efficiency, reliability and cost of achieving mission goals while maintaining dynamic adaptation to the unique limitations and contingencies of a real-world operating environment. Our response to this challenge is the creation of a new infrastructure that will facilitate cooperative and collaborative performance of human and robots as equal team partners through the application of advances in goal-oriented, multiagent planning and coordination technology. At the heart of our approach is the USC Teamcore Group's Machinetta, a state-of-the-art robot proxy framework with adjustable autonomy. Machinetta facilitates robot-human role allocation decisions and collaborative sharing of team tasks in the non-deterministic and unpredictable military environment through the use of a domain-independent teamwork model that supports flexible teamwork. This paper presents our innovative proxy architecture and its constituent algorithms, and also describes our initial demonstration of technical feasibility in a realistic simulation scenario.*

**KEYWORDS:** Human-robot teams, collaborative performance, Machinetta framework, multiagent planning and coordination, OneSAF Objective System

### 1. INTRODUCTION

Military forces of the future will use mixed manned and unmanned forces for a broad variety of functions: for reconnaissance and surveillance, for communications, in forward-deployed offensive operations, for logistics

and support, and as tactical decoys to conceal maneuvers by manned assets.

Robot-human teams introduce a new and unique aspect to the planning and coordination of team performance: the interaction of two cognitive systems – human and autonomous robot. In addition to the critical performance factors associated with human teams – which include information exchange, communication, supporting behavior and team leadership – the mixed robot-human team adds a number of challenging new dimensions.

Researchers have previously addressed the implications of the new human-robot team paradigm. One finding is that the goal of such robot-human teams is to extend manned and unmanned capabilities and act as “force multipliers”, as in the US Army Future Combat System [1][2].

In order for a robot-human team to work together effectively, it must collaborate and coordinate effectively; all members should be aware of the overall goals of the team, and coordinate their activities with their fellow team members in order to further the team's goals [6]. Team heterogeneity poses difficult challenges, since different entities may have differing “social” abilities and hence differing abilities to coordinate with their teammates.

### 2. R&D OBJECTIVE

The R&D problem addressed in this project is how to achieve fully integrated teams that will be able to take advantage of the unique capabilities and strengths of their heterogeneous team members while avoiding the limitations of individual team members or hampering the effectiveness of other team members or of the team as a whole. In other words, the challenge is to exploit fully the unique capabilities of heterogeneous teams composed of a mixture of Robots, Agents or Persons (RAPs).

The specific real-world objective is to improve the safety, efficiency, reliability and cost at which mission goals can be achieved while maintaining dynamic adaptation to the real-world operating environment with its unique limitations and contingencies.

### 3. TECHNICAL APPROACH

Our response to this challenge is to create a new infrastructure that will facilitate the cooperative and collaborative performance of human and robots as equal team partners through the application of advances in goal-oriented, multiagent planning and coordination technology. MAAF is an innovative robot-human team proxy architecture that will integrate *domain-independent* modules for coordination reasoning, maintaining local beliefs and adjustable autonomy, with *domain-specific* modules for mission completion.

At the heart of our approach is the USC Teamcore Group's Machinetta, a state-of-the-art robot proxy framework with adjustable autonomy [7]. Machinetta facilitates robot-human role allocation decisions and collaborative sharing of team tasks in the non-deterministic and unpredictable military environment through the use of a domain-independent teamwork model that supports flexible teamwork [5][9][10].

As a result, our Machinetta-based approach will ensure highly flexible role allocation and re-allocation, as well as enriched communication between RAPs and between a RAP and its proxy, to significantly improve teamwork flexibility.

### 4. MAAF SYSTEM CONCEPT

Figure 1 shows our concept for the Multiagent Adjustable Autonomy Framework (MAAF). The system itself consists of the following main components:

- **Battlefield.** Contains teams made up of air and ground robots and human team members.
- **Command Unit.** Establishes the top-level mission and or task objectives.
- **Human Interface.** Provides the mechanisms by which the human team members communicate with the robot-human team.
- **Teamwork Proxies.** Software agents that represent the various battlefield entities for the purpose of negotiating task assignments, monitoring status, and ensuring that the entities communicate necessary information among themselves.

The human interface builds on the US Army's OneSAF Objective System (OOS). OOS is a composable assembly

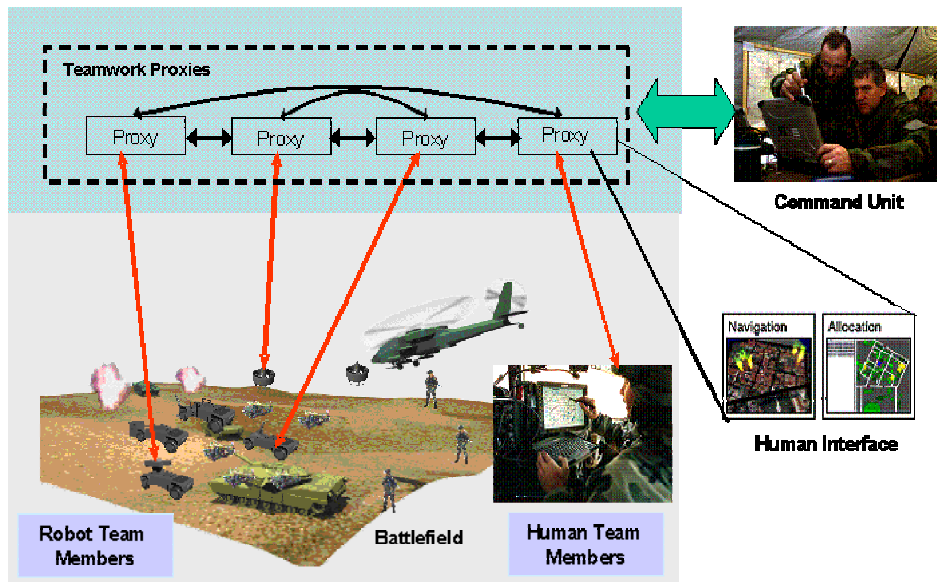


Figure 1. MAAF System

Our proxy architecture itself comprises a layered Mission Management and Control Hierarchy with the Machinetta Proxy Planning and Coordination Function on top, and the Robotic Vehicles Mission Execution Engine at the lower level. This provides for efficient communication between proxies as well as communication between a proxy and other team modules.

of computer generated forces (CGF) designed for brigade and below combat and non-combat simulations. OOS provides intelligent, doctrinally correct behaviors for both simulation and control of robots. OOS was built to simulate the modular future force, and to represent entities, units, and behaviors across the spectrum of military operations. It thus provides a leveraging infrastructure for testing human-robot

systems in simulation with robotic avatars as well as a means for supporting rich configurations of human interface options.

In our design, OOS was adapted to allow the human team members to have an *omnipresent* interaction with remote robot team members. The interface thus combines two modes of operation and MAAF utilizes a proxy framework that can handle both the coordination and communication for the human-agent team.

A key assumption is that intelligent, distributed, unmanned robots will be a main element of future tactical teams. We believe that a critical role of these intelligent robots will be to manage coordination between all members of the response team, and that such coordination can take advantage of the MAAF emerging infrastructure. Specifically, we are using coordination algorithms inspired by theories of teamwork to manage the distributed robot-human team. The general coordination algorithms are encapsulated in *proxies*, with each RAP (Robot, Agent or Person) team member having its own proxy and representing it in the team.

## 5. ARCHITECTURE & ALGORITHMS

The current version of the proxies is called *Machinetta*. Machinetta is implemented in Java and is freely available on the web. It is important to note that the concept of a reusable proxy differs from many other “multiagent toolkits” in that it provides the coordination *algorithms* themselves, e.g., algorithms for allocating tasks, as opposed to the *infrastructure*, e.g., APIs for reliable communication.

- **Adjustable Autonomy**, reasoning about whether to act autonomously or pass control
- **RAP Interface**, communication among the team members

The Machinetta software consists of five main modules, three of which are domain independent and two of which are tailored for specific domains. The three domain-independent modules are for coordination reasoning, maintaining local beliefs (state), and adjustable autonomy. The domain specific modules are for communication between proxies and communication between a proxy and a team member. The modules interact with each other only via the local state with a blackboard design and are designed to be “plug and play.” Thus new adjustable autonomy algorithms can be used with existing coordination algorithms.

The coordination reasoning is responsible for reasoning about interactions with other proxies, thereby implementing the coordination algorithms. The adjustable autonomy algorithms reason about the interaction with the team member, providing the possibility for the team member to make any coordination decision instead of the proxy.

The key technique we have used in order to program robot-human teams is “team-oriented programming.” The basic idea here is to provide high-level team plans and their decomposition into sub-team plans and roles. The remarkable part of the program is what is missing from it. That is, there is no specification of which agent does what role or how agents should reallocate roles upon failures, nor is there any specification of which agent should communicate with whom and when.

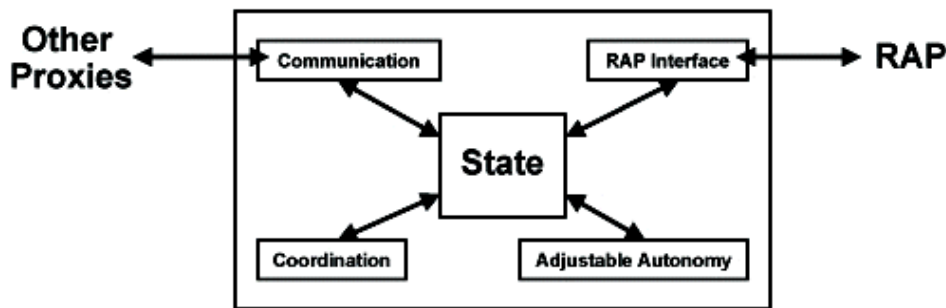


Figure 2. Teamwork Proxy Architecture

Figure 2 is an overall diagram for our teamwork proxy architecture, which includes the following:

- **Communication**, with other proxies
- **Coordination**, reasoning about team plans and communication
- **State**, the working memory of the proxy

The key idea is to relieve the programmer of the effort of programming the low-level coordination details of role allocation, communication, or adjustable autonomy. The main algorithms needed for team execution of the tactical maneuvers are described briefly below.

**Team Plan Execution and Role Allocation.** The robot-human team will need to be able to execute joint sequences of actions while taking into account the current state of the world. With the help of the proxies, the robot-human team will implement *team oriented plans* (TOPs) which describe joint activities to be performed in terms of the individual *roles* to be performed and any constraints between those roles. Generally, TOPs are instantiated dynamically from TOP templates at runtime when pre-conditions associated with the templates are filled. Typically, a large team will be simultaneously executing many TOPs. In a planned route reconnaissance scenario, for example, a team would execute a TOP similar to the following when the suspect vehicle is reported:

The Team Oriented Program (TOP) was designed to be general and has nothing agent-centric built in. The TOP has three main sections:

- A collection of team plans that can be applied to the current domain. This is also where team level goals are explicitly defined.
- A list of the team members and their characteristics: RAP type (robot, agent or person), capabilities, how to communicate to them, etc.
- A team hierarchy or structure. This section would define team and sub-team size and authority. This section is not emphasized in current versions of Machinetta; however, it might become more important in this project, where we don't have just a flat hierarchy.

Current versions of Machinetta cast the role allocation problem as a Distributed Constraints Optimization Problem (DCOP) and solve it via the LA-DCOP role allocation algorithm [6].

**Adjustable Autonomy.** Adjustable Autonomy uses transfer-of-control strategies to allow the best team-mate (most available and capable) to have autonomy over a decision at a given time. More clearly, a transfer-of-control strategy is a preplanned sequence of actions to transfer control over a decision among multiple entities. For example, an AGH strategy implies that a UAV (A) attempts a decision and if the UAV cannot make the decision then the control over the decision is passed to a UGV (G), and then if G cannot reach a decision, then the control is passed to the human (H).

**Mapping and Path Planning.** Path planning is a widely studied problem in robotics, and for this function we are examining the many existing algorithms, such as the A\* and D\* algorithms, that can give optimal plans for navigation provided the map is certain. Although often teams act in terrains that are completely known, other standard search methods also exist, such as probabilistic road maps (PRMS), rapidly region growing trees, gradient descent methods or graph based methods such as Voronoi graphs. Issues similar to the ones that arise in our project have been studied in UAV landing in uncertain terrain [8] and autonomous mapping of unknown terrain [3].

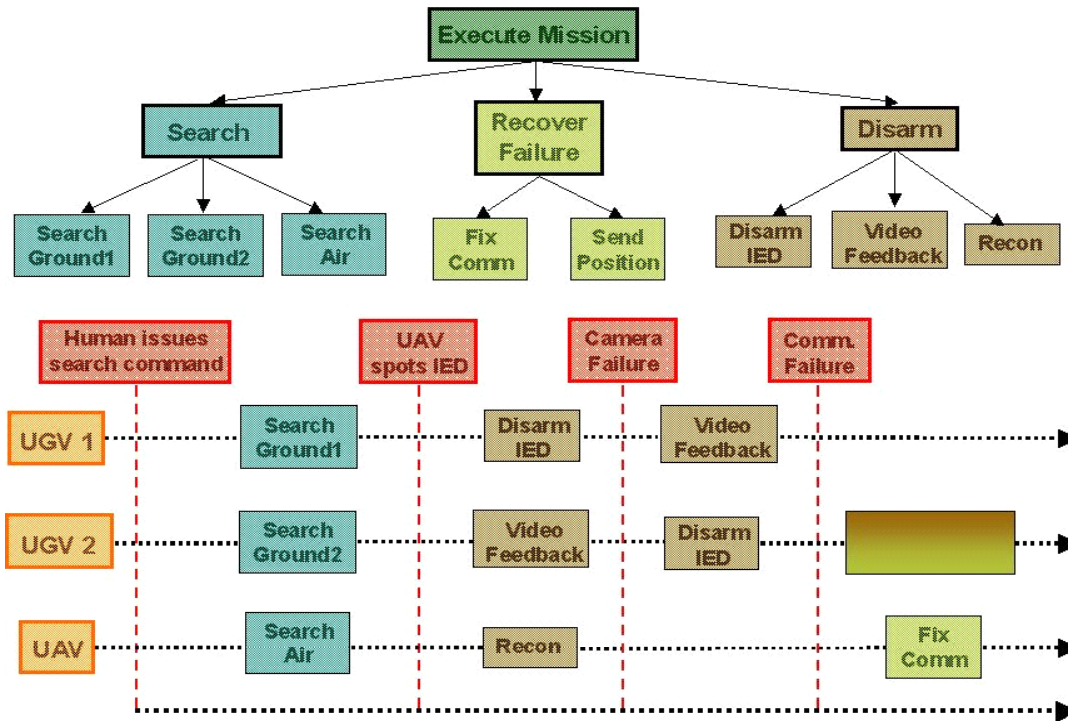


Figure 3. Team-Oriented Program

**Incorporation of Uncertainty.** One of the challenging issues in building real robots is that sensors can be faulty – and hence their observations are imperfect or the robot cannot actually meet the rigid assumptions made in theoretical models. To address this we plan to use the standard uncertainty models existing in artificial intelligence literature for transition and observational uncertainties. The most popular models are the Markov decision processes (MDPs) and the partially observable Markov decision processes (POMDPs). MDPs handle the transition uncertainty while the POMDPs can handle the transition as well as the observation uncertainty. Although these models can handle a fairly large class of uncertainties existing in real world, the main issue with them is the complexity of planning. Additional research is now focused on finding approximate solutions for these models while gaining on complexity issue. One promising direction is the building of BDI-MDP/POMDP hybrids [4].

As we plan to use a proxy based architecture which is primarily a BDI based system, without ignoring the fact that uncertainty is an important issue we can handle it using the BDI-MDP hybrids.

## 6. TEAM-ORIENTED PROGRAM

Figure 3 illustrates a portion of our team-oriented program, which is shown via a plan-hierarchy. The boxes with thick outlines show team-plans in the team-oriented program, while the remaining boxes show plans that are to be done by individuals. Thus, ‘execute-mission’ is a team-plan that is decomposed into three separate team

plans, i.e. ‘search,’ ‘disarm’ and ‘recover-failure.’ The search team-plan is further decomposed into plans for individual agents, including ‘search-ground1,’ ‘search-ground2’ and ‘search-air.’ Other team-plans similarly decompose into activities for individuals. At execution time, the individual plans are allocated to individual agents automatically by Machinetta based on their capabilities and their availability and constraints among agents’ roles.

In particular, the bottom half of Figure 3 shows the flow of execution of the team plans. When there is a communication failure, the recover-failure team-plan is activated. We see that because multiple team-plans are active, the active plans are shaded in two colors.

## 7. DEMONSTRATION

Our Phase I Proof-of-Concept demonstration focused on a typical route reconnaissance task of detecting and disarming IEDs: That is, a team of UAVs, UGVs and humans coordinate in a dynamic environment in order to accomplish specific tasks. The UAV and UGVs all have the capability of autonomous behavior under certain conditions. The humans will be engaged at moments where human skills are more desirable. When these moments pass, the humans shift back to more global monitoring.

The team needs to be able to adjust for unknown contingencies and dynamics in the environment. For example, the UGVs are equipped for obstacle avoidance during waypoint navigation. Information about obstacles is

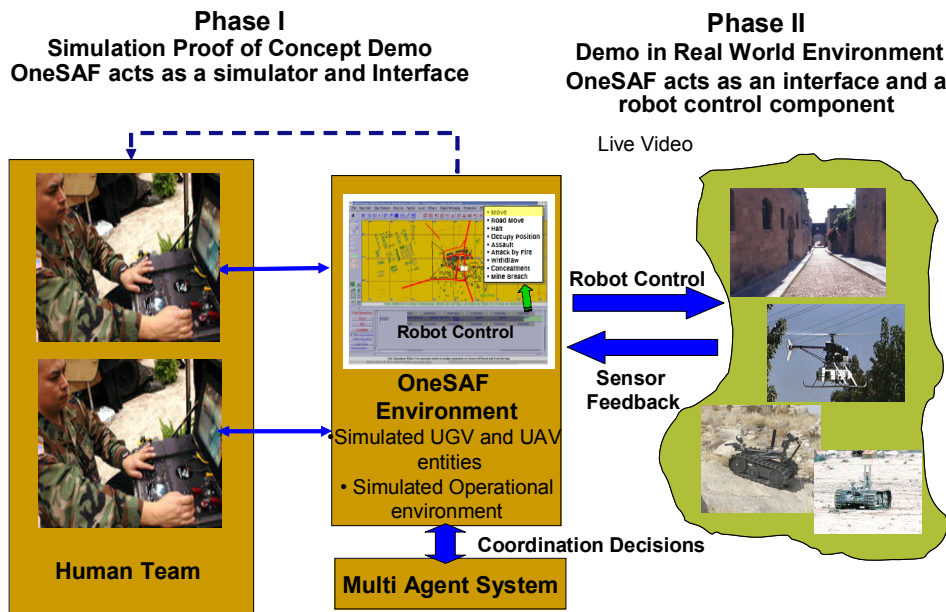


Figure 4. MAAF System Showing OneSAF Role

passed on by the UAV which has an aerial perspective of the routes being followed by the UGVs. Depending on the status of the world and the status of the team, our approach will choose the best strategy for allocating roles and adjusting the autonomy of unmanned vehicles. The goal of choosing the best strategy is extremely challenging given the distributed knowledge and execution of the team plans. In order to allow the team to coordinate better, we develop a team structure in which each team member has a proxy for handling communication and coordination with other team members, as well as decision making.

In the actual proof of concept demonstration we used the OneSAF Objective System (OOS) as both a simulation and user interface and used an existing scenario for Route Reconnaissance and Obstacle Clearance (RROC). OOS extends directly to real world robot control, so that our use of OOS will support seamless transition of MAAF to other DoD programs where OOS will be used for doctrine development and training.

Figure 4 shows the top-level demonstration system architecture. For the proof of concept demonstration the Machinetta multiagent system (MAS) was integrated with the OneSAF Objective System (OOS) to allow MAS proxies to obtain sensor and status data from the simulated the unmanned systems and assign tasks to the unmanned systems.

The integration approach and method is analogous to how MAAF would be integrated with actual real world

unmanned systems. That is, in our approach OneSAF will be used as intermediary control and sensory feedback interface for the integration with actual unmanned systems. By integrating OneSAF and Machinetta we have constructed a test-bed for evaluating higher levels of autonomy and coordination of unmanned systems in tactical scenarios. Integration provides both the functionality to perform basic human-robot missions and a foundation for future work.

## 8. RESULTS

Figure 5 illustrates the main human-robot, mixed-initiative events handled by MAAF in the Route Reconnaissance and Obstacle Clearance scenario adapted for our Phase I proof-of-concept demonstration. The numbers next to the events depict the sequence of execution.

- The scenario begins with Event 1, when the human issues a search order. The human's proxy communicates this order to other team members via their proxies. This triggers a plan to search the given area for IEDs. The two UGVs then proceed to navigate towards the designated area.
- Event 2 shows autonomous behavior of a UGV, by its ability to successfully maneuver around an obstacle. Once the UGV senses an obstacle, its proxies compute a path around it and reroute the UGV in order to avoid the obstacle. Meanwhile, the UAV detects a potential IED. This information is conveyed to all team members, and it triggers a team plan to disarm the IED.



Figure 5. MAAF Events in the RROC Scenario

Once the UGVs are aware of the new team plan, they start moving towards the potential IED.

- Event 3 is a camera failure on one of the UGVs, which disrupts the live video feed on the human's monitor. As soon as this failure occurs, the proxies for the two UGVs switch the role for providing video feed to the other UGV. The camera view is transferred to the other UGV, and subsequently the video feed is restored on the human's monitor.
- Event 4 shows the ability to recover from a communication failure. One of the UGVs loses communication with the other team members. When this is detected, a team plan is triggered to restore communication. The proxy for the other UGV sends in position coordinates of the two UGVs to the UAV's proxy, which computes a point the UAV would need to reach in order for the communication to be restored. The UAV then flies in to this point, acting as a relay station between the two UGVs, and communication is restored.
- Finally, Event 5 shows adjustable autonomy in action. The UGVs are approaching the IED, and UGV2 has the role of disarming the IED. As the UGVs close in on the IED, the confidence level of UGV2 that the object is indeed an IED increases. When this confidence level crosses a certain threshold, a message is sent to the human asking to confirm the presence of an IED, using the video feed from the other UGV. The human responds by confirming that the object is an IED, and UGV2 switches from autonomous mode to manual control. The human remotely disarms the IED, and once the disarming is complete, an instruction is sent to the UGVs to fall back, which completes the scenario.

The demonstration results validated the full objective range of MAAF capabilities using the architecture and algorithms developed in the Phase I effort, and showed that the technology is ready for real world application in the planned Phase II continuation.

## 9. MACHINETTA KEY STRENGTHS

The application of Machinetta to the robot-human domain revealed major strengths as well as important areas of future research for the novel teamwork architecture. One of the exciting aspects of this project is that this is among the first application of theory-driven, reusable teamwork architectures in robot-human teams. Key strengths of Machinetta that were emphasized and validated in this application include:

- *Addressing unanticipated teamwork failures:* Our Machinetta architecture was able to handle many such failures without requiring new capabilities to be programmed in. For example, a role failure is

automatically handled by reallocation of roles among agents by algorithms that are already in-built in Machinetta; we did not have to program in new algorithms.

- *Team-oriented programming, avoiding many domain-specific low-level coordination plans:* A key advantage of Machinetta teamwork architecture is that we can program it using high-level team-oriented programs. For example, the team-oriented program does not state what happens if a role fails; instead, the architecture handles such role failures via its in-built algorithms. This reusable set of algorithms within Machinetta reduces the programming effort by avoiding the requirement to write low level coordination plans.
- *Advances over prior generation of teamwork architecture (use of hybrids):* While Machinetta builds on over a decade of research in teamwork architectures --- there have been significant improvements in Machinetta over those architectures. For example, one major improvement is that it uses hybrid techniques to encapsulate a larger group of capabilities. In particular, while communication is based on logical theories of teamwork, the role allocation and reallocation is based on DCOPs, and adjustable autonomy within the architecture is based on MDPs (Markov Decision Problems).
- *No major barriers in extension to robot human teams and handling contingencies:* We found no major conceptual barriers in applying our teamwork framework to robot-human teams. In particular, the distributed nature of the proxy-based architecture allows it to be scalable. Furthermore, its reusability allows it to migrate from domain to domain with relative ease.

We have also identified a number of very interesting open research issues in this project, which we plan to address further in subsequent R&D efforts. Some of these issues include:

- *Adjustable autonomy given delayed or lacking human intervention:* We have yet to fully exploit adjustable autonomy algorithms. Bringing complex adjustable autonomy algorithms from the agent literature into robot-human teams will be an exciting and significant development.
- *Communication failures and other potential failures in using real-world robots:* While our architecture provided flexible responses in many failures, failures such as communication failures and others require new failure recovery algorithms.
- *Exploration of hybrid architectures:* From a research perspective, we must explore hybrid architectures that combine logic-based symbolic approaches for teamwork with decision-theoretic approaches such as

MDPs (Markov decision problems) to address the uncertainty that we must face in robot-human teams.

## 10. CONCLUSIONS

Our project of building robot-human teams for a real-world military application involves using the latest technologies from Artificial Intelligence and Robotics. Such applications are still in their incubation stages. So our project is a major step toward achieving practical usage. Its two most innovative features that will promote additional research and development are in the areas of robust human-robot teams and adjustable autonomy for the real world.

Our approach is to build teams where humans and robots work as equal partners rather than a master slave relationship. The typical challenges faced in such projects are control issues. For example, humans are usually good at decision making without explicitly using the utility maximization theory or assigning utilities to each and every step in the project. On the other hand machines are very good at processing numbers and given certain logic on how to use these numbers can achieve the output of that logic using the input data quite fast.

Each member of the mixed initiative team needs to take decisions and perform actions. However, many situations can arise where a team member could get better inputs from other team members rather than just taking the decision by himself or itself. At the same time, there many issues that arise from being dependent on teammates.

Our project proposes an architecture that defines the control structure clearly so that potential conflicts can be resolved in a collaborative manner; we will continue to validate it through implementation, and we believe such implementation will motivate further R&D in this direction.

Each member of the mixed initiative team needs to take decisions and perform actions. However, many situations can arise where a team member could get better inputs from other team members rather than just taking the decision by himself or itself. At the same time, there many issues from being dependent on teammates. First, teammates can be quite busy and unable to respond. Second, communication can be limited or costly. Third, teammates might not be as capable as one thinks they are. In this project we introduce adjustable autonomy as a solution to such difficult and tricky problems, and again, our implementation of an adjustable autonomy framework will promote further R&D on similar issues.

In summary, in this project we met our SBIR Phase I objectives in full. We successfully designed and developed the basic MAAF proxy architecture and algorithms, demonstrated the feasibility of the concept in a realistic scenario. We are confident of being able to complete the full MAAF development, evaluation and transition project in the next phase.

## REFERENCES

- [1] Barnes, M., R. Parasuraman, and K. Cosenzo, "Adaptive Automation for Robotic Military Systems," Technical Report, NATO HFM, (In press)
- [2] Cosenzo, K., R. Parasuraman, A. Novak, and M. Barnes, "Adaptive Automation for Robotic Military Systems," ARL Technical Report, ARL-TR-3808, 2006
- [3] Howard, A., L.E. Parker, and G.S. Sukhatme, "Experiments with Large Heterogeneous Mobile Robot Team: Exploration, Mapping, Deployment and Detection," *International Journal of Robotics Research*, 2006
- [4] Nair, R., M. Tambe, and S. Marsella, "Role Allocation and Reallocation in Multiagent Teams: Towards a Practical Analysis," Proceedings of The Second International Joint Conference on Agents And Multiagent Systems (AAMAS), 2003
- [5] Pynadath, D. V., and M. Tambe, "Automated Teamwork Among Heterogeneous Software Agents and Humans," *Journal of Autonomous Agents and Multi-Agent Systems (JAAMAS)*, 2003, 7:71-100.
- [6] Saripalli, S., J.F. Montgomery, and G.S. Sukhatme, "Visually-Guided Landing of an Unmanned Aerial Vehicle," IEEE Transactions on Robotics and Automation, 2003, Vol. 19
- [7] Scerri, P.; D. Pynadath, and M. Tambe, "Towards Adjustable Autonomy for the Real World," *Journal of Artificial Intelligence Research*, 2002, 17:171-228.
- [8] Schurr, N., J. Marecki, P. Scerri, J.P. Lewis, and M. Tambe, "The DEFACTO System: Training tool for Incident Commanders," The Seventeenth Innovative Applications of Artificial Intelligence Conference, 2005
- [9] Tambe, M. "Towards Flexible Teamwork," *Journal of Artificial Intelligence Research*, 1997, 7:83-124
- [10] Tambe, M. and W. Zhang, "Towards Flexible Teamwork in Persistent Teams: Extended Report. *Journal of Autonomous Agents and Multi-agent Systems (Special Issue on "Best of ICMAS 98")*, 2000, Vol 3, 159-183

## ACKNOWLEDGEMENTS

The research and development described in this paper was performed as a DARPA SBIR Phase I project under Contract No. W31P4Q-07-C-0146. We thank DARPA-IPTO Program Manager Dr. Thomas Wagner for his valuable technical guidance, and Dr. Dan Corbett, DARPA-IPTO Science and Technical Support, for his contributions to our successful completion of this effort.