Mixed-Initiative Adjustable Autonomy for Human/Unmanned System Teaming

Meghann Lomas de Brun, Vera Zaychik Moffitt, Jerry L. Franke, Dimitri Yiantsios, Trevor Housten, Adria Hughes, Shannon Fouse, and Drew Housten
Lockheed Martin Advanced Technology Laboratories
3 Executive Campus, Suite 600, Cherry Hill, NJ 08002
Phone: (856) 792-9681, Fax: (856) 792-9920, http://www.atl.lmco.com
mlomas@atl.lmco.com

Abstract

To increase the applicability and utility of unmanned systems, we propose a mixed-initiative adjustable autonomy system that will enable a more flexible interaction between humans and unmanned systems. Typically, the autonomy levels of unmanned systems are fixed; however, during operation there are a number of elements that are likely to change including the environment, the mission, workload, and trust. A fixed level of autonomy can result in an inability of the human/unmanned system team to respond as effectively as desired.

A mixed-initiative adjustable autonomy system enables either member of a human/unmanned system team to adjust autonomy during mission execution. In this paper we discuss the benefits and challenges of mixed-initiative adjustable autonomy, describe previous work on the subject, and present our own approach. We discuss our approach in the context of a driving mission for a human-robot team and present the human interfaces we have designed with this task in mind. We also describe ongoing work to validate the effectiveness of this approach.

I. INTRODUCTION

Unmanned systems are increasingly called upon to operate in real-world environments [6]. The complex, dynamic nature of these environments often decreases mission effectiveness because the human/unmanned system team is unable to respond gracefully to changes in the environment, the mission, and operator workload. To address these issues, the typical response is to design increased autonomous capabilities into the unmanned system to improve its performance under a wider variety of situations and subsequently move the human’s role further toward supervision [9]. However, this poses several problems. Increasing autonomous capabilities is a difficult problem that may result in decreased operator awareness, which is necessary for the operator to make mission-level decisions. It also does not always reduce operator workload because of the impact of trust on workload, an
important but often overlooked element in successful mission execution.

There are two primary reasons why increasing the autonomous capabilities of the unmanned system is not sufficient for reducing workload or improving trust. Both reasons stem from one unavoidable facet of the human/unmanned system relationship: the human is ultimately responsible for the unmanned system’s behavior. Typically, the operator and the programmer of the unmanned system are different people, which may result in a lack of understanding of how the system works on the part of the operator. As a result, the operator is unable to anticipate the unmanned system’s response to various situations, lowering the operator’s trust in the unmanned system’s autonomous functions. Experience shows that when the system is not trusted to act autonomously and the only other option is for the human to take complete manual control, many operators prefer using manual control. However, this dramatically increases the human’s workload, preventing multi-tasking, and resulting in an inefficient use of the autonomous capabilities of the unmanned system. Only when the operator trusts the autonomy can the operator take advantage of it.

Further complicating the matter is the dynamic nature of mission execution. Even if workload and trust are at satisfactory levels at the beginning of the mission, because the mission is dynamic, they will change over the course of the mission. At times, the human will be overloaded with tasks to perform, unable to monitor the unmanned system making it desirable for the unmanned system to operate autonomously. At other times during the mission, the human may be under-loaded and to maintain engagement, it is beneficial to be involved in the unmanned system’s operation. Likewise, while the unmanned system is operating in a simple, static, familiar environment, the human is likely to trust the unmanned system’s autonomous performance, whereas in complex, unexpected situations, the human is not likely to trust the unmanned system to operate without supervision. Mission execution is likely to be successful if workload and trust levels can be maintained at a constant level despite dynamic changes in the environment and mission.

In order for the human and unmanned system to operate effectively together, the interaction between them must be addressed [11]; and to promote and maintain trust in an unmanned system, the workings of that system must be apparent to their human teammate [7]. Currently, the control paradigm governing human/unmanned system interaction is rigid; typically there is an all-or-nothing approach with respect to autonomy. However, a more flexible control paradigm will improve mission execution performance in dynamic situations. Mixed-initiative adjustable autonomy has the potential to increase
this flexibility and promote a steady level of workload and trust throughout the mission. This approach can be immediately beneficial - without the need to develop more autonomous systems - and to a wide variety of platforms. In this paper, we present the subject of mixed-initiative adjustable autonomy, discuss related work, and describe our approach for applying mixed-initiative adjustable autonomy to unmanned systems. While our research can be applied to a variety of unmanned systems, for the purposes of this paper, we also use the term “robot” to represent the embodied unmanned systems on which we focus.

II. DESCRIPTION OF MIXED-INITIATIVE ADJUSTABLE AUTONOMY

Adjustable autonomy is a mechanism that enables a change in the autonomous level of an unmanned system during mission execution. This allows human/unmanned system teams to share responsibilities and to shift them back and forth as necessary. A human/unmanned system team capable of adjustable autonomy has the flexibility to change autonomy levels assignments during operation to maintain an appropriate workload level, trust level, and improve mission execution despite dynamic situations. Trust is promoted through the operator’s ability to revoke responsibility when not satisfied with the performance of the unmanned system. A mixed-initiative adjustable autonomy system further promotes this trust by enabling the unmanned system to make proactive recommendations about when the autonomy level should change. This further reduces the human’s workload by offloading the need to monitor the overall team performance. By giving the unmanned system the ability to provide recommendations for the autonomy level, a mixed-initiative adjustable autonomy system promotes the idea of the human and unmanned system operating as a team and better enables the use of information from the unmanned system.

In Section III we describe the existing approaches for adjusting autonomy and discuss their use in related research. We describe our approach for a mixed-initiative adjustable autonomy system in Section IV.

III. RELATED WORK

Adjustable autonomy is applicable to systems in which multiple agents (humans and/or unmanned systems) must cooperate. It has been used in a variety of scenarios such as human/unmanned system teaming for space applications [2]–[4], [16], for search and rescue [5], and for navigation of
There are three main approaches to adjusting autonomy: a linear approach, a hierarchical approach, and a policy-based approach. In a linear approach [1], [5], [10], linear autonomy “levels” ranging from manual operation by the human, to completely autonomous unmanned system operation, are predetermined by a designer. In a hierarchical approach [15], a hierarchy of behaviors is generated. The autonomy levels are set by which behaviors are active and autonomy is adjusted by adding, removing, or pausing behaviors in the hierarchy.

In policy-based approaches, the user defines a set of polices which are guidelines that establish permissions for the adjustment of autonomy. Policies are designed to ensure the system acts as desired by the user throughout the operation, even when the environment or mission changes. They are typically predefined, which unlike dial-based and hierarchical approaches, reduces the amount of input the user needs to have to the adjustable autonomy system during operation. As a result, the user is able to focus on accomplishing assigned tasks rather than monitoring the performance of the unmanned and adjustable autonomy systems. For this reason, policy-based approaches are particularly well suited to a mixed-initiative system. Policy-based approaches have been used to establish permissions on how unmanned systems should perform under various situations by adjusting the limitations on the set of permitted actions for the unmanned system. In [13], policies are used to ensure the unmanned system will consult with the human user under certain situations. Similarly, in [2]–[4], the policies establish the unmanned system’s permissions, obligations, and capabilities under various conditions. All three approaches for adjustable autonomy are described in further detail in [12].

We base our mixed-initiative adjustable autonomy approach on policy-based approaches because it is the most flexible of the three approaches. Real-world interactions between humans and unmanned systems are complex and the policy-based approach has the most potential to encompass a variety of interactions. It provides designer control, but by using guidelines rather than fixed levels, it provides the flexibility needed when there are many tasks and environmental possibilities (as is typically the case in real-world applications). As a bottom-up approach, it allows the designer to shape the interactions without imposing a rigid structure that could limit the versatility and adaptability of the human/unmanned system team. Furthermore, because the policies are written by the user, the performance of the unmanned system is better understood and therefore more trustworthy to the
There are a couple of challenges to using a policy-based approach. The interface must be designed so a human user both trusts and is capable of easily making adjustments to the adjustable autonomy system. The interface must clarify the workings of the mixed-initiative adjustable autonomy system and the unmanned system, not impose additional work on the human. Interfaces for linear and hierarchical approaches have been designed [5], [10], [14], [15], but the design of an interface for a policy-based approach is a challenging, open problem. We discuss this in Section V. The other challenge to policy-based approaches is writing the policies so they enable the system to act as desired. We discuss policies in more detail in Section IV-B.

IV. OUR APPROACH FOR MIXED-INITIATIVE ADJUSTABLE AUTONOMY

In our approach, agents (humans and unmanned systems) connect to the mixed-initiative adjustable autonomy (MIAA) system that assesses and supports the distribution of responsibilities. Our approach is applicable to any teaming scenarios where multiple agents share responsibilities. For initial experimentation, we consider a scenario that has two agents: a human sitting inside an autonomous vehicle. This human-robot team must traverse a varied, challenging driving course while answering questions based on information displayed in the environment. This experiment will be done in simulation. We have developed a virtual driving environment that uses a steering wheel and pedals as joystick inputs, allowing humans to realistically simulate driving the vehicle. In our scenario, we have driving tasks of steering, speed control (including braking), adjusting waypoints, deciding when to replan, and controlling the lights and turn signals.

Based on the tasks and the known strengths and weaknesses of the agents, the MIAA system developer establishes policies that can help govern task assignment under specific conditions. For our scenario, some policies are: “If the speed limit is over 15 miles per hour and the terrain is gravely, the car may not steer or control the speed,” and “If the car encounters a detour, the human must provide the car with waypoints.” Policies become active based on the current situational conditions which enter the MIAA system as data from the agents. This situation awareness information is scenario-dependent. For our example, situation awareness information includes: the GPS location and orientation of the vehicle, the type of terrain the robot is on (smooth, bumpy, slippery, or wet), the speed limit at the robot’s location, and the presence of any signs and information from the signs.

We chose the cooperative driving scenario for its versatility. The fundamental aspects of this can
be adapted to any type of unmanned vehicle including search and rescue robots or bomb diffusing robots and could apply to unmanned systems operating in any domain: air, land, or sea. Our approach supports any scenario in which a human and unmanned system work together and have a set of tasks that may be shared or exchanged but must be accomplished.

Several key issues drove our system design:

1) **Our objective is to develop a mixed-initiative adjustable autonomy system that applies to as many existing and future human/unmanned system teams as possible.** We designed a self-contained system with interfaces to connect to the agents (human and unmanned) in the team. Our system does not need to be reconfigured for each team; only the interfaces need to be integrated with the agents. To promote versatility, we chose a policy-based approach to adjustable autonomy because, as a bottom-up approach, it allows the most flexibility in autonomy adjustment and in the types of systems that can use it.

2) **Our goal is to enable an autonomous system to also be able to take the initiative to change the autonomy level.** This is the mixed-initiative aspect of our desired system and suggests a symmetry to the overall system diagram. In our approach, the robot shares and receives information the same way as the human. We also include a system component that evaluates the performance of the team members so the human team members are not the only agents capable of evaluation.

3) **To effectively use them, the human must trust both the unmanned system and the mixed-initiative adjustable autonomy system.** To promote trust, we enable sharing information and task assignments from the robot and our designed system to the human. We also use a policy-based approach, which promotes trust because it follows guidelines established (and therefore understood) by the human.

4) **The system must be able to use information from the environment, humans, and unmanned systems to assist with autonomy adjustment.** We use a situation awareness component that gathers information from the agents and shares relevant information with other components of the system.

From these needs we designed a system to address each issue. The system is described in Section IV-A. Policies and scoring, two major features of our approach, are described in Sections IV-B and IV-C, respectively.

**A. System Diagram and Components**

Our system diagram is shown in Figure 1. Each component is described in detail below.
1) **The Human’s Interface.** This interface connects the human to the mixed-initiative adjustable autonomy system (MIAA) by communicating information from both the system and the robot and responding to the human’s inputs. It allows humans to maintain awareness of their responsibilities at all times. The design of this interface establishes the human’s trust in the system. The interface itself is task-specific and is described in Section V. For teams with multiple humans, there would be an interface for each human.

2) **The Robot’s Interface.** This interface performs the same function as the human’s interface: it acts as the communication channel allowing the robot and the mixed-initiative adjustable autonomy system to share information and assignments. This interface is platform-specific and like the human interface, for multi-robot systems there would be an interface for each robot.

3) **Situation Awareness Manager.** This component receives situation awareness information from both the human and the robot via their interfaces and discretizes the information so it is more manageable. To simplify the sharing of this information with the other components in the mixed-initiative adjustable autonomy system, we use a publish/subscribe approach.

4) **Task Assignment Coordinator.** This component monitors the current assignments and coordinates the Common Autonomy Model and the Autonomy Level Assessor during the assignment process. It informs the human(s) and the robot(s) of their assignments via their respective interfaces. This component also ensures that someone is in control of every task, even during reassignment, by
ensuring that no task is halted by an agent unless it has been successfully reassigned to another agent.

5) **Common Autonomy Model.** This component contains the policies. It determines which policies should govern the current task assignments by comparing the information it receives from the Situation Awareness Manager to the policies. For example, in our scenario, if the speed limit is above 15 mph and the terrain is gravely, the first example policy in Section IV will become active and, as dictated by the policy, the car cannot be assigned the tasks of steering or speed control. All policies are framed as if-then statements: *If* (condition), *then* (rule). The Common Autonomy Model uses the information it gets from the Situation Awareness Manager and matches it to the “conditions” in the policies. If a condition expression is matched, the policy is considered “active,” and the rule expressions of the active policies must be followed. These rules typically govern who may or may not perform a specific task under certain conditions. The policies are of primary importance to our approach and are described in Section IV-B.

One of the major challenges of a policy-based approach is defining the policies. Policies are task-specific and must be carefully designed by the programmer so they will cover as many situations as desired without conflicting with each other. As the task complexity increases, the number of policies typically increases, making policy conflicts more likely. To reduce the likelihood of conflicts, we use a reasoning engine in the Common Autonomy Model. While we specifically use CLIPS, a reasoning engine freely available and written in C, our system is designed so another reasoning engine could be substituted.

6) **Autonomy Level Assessor.** The Autonomy Level Assessor is an evaluation component that has two closely related responsibilities. Its first task is to monitor the current task assignments and make recommendations for change. This is done when the situation awareness information has not changed enough to trigger a change in the policy-based task assignment but has changed enough so the human-robot team is no longer operating as efficiently or effectively as desired. Its second responsibility is to assign tasks using a scoring method once the Common Autonomy Model has generated a set of possible assignments based on policies. The Autonomy Level Assessor’s scoring system is described in Section IV-C.

**B. Policies**

Policies facilitate the functionality of our system by incorporating information a human user may already have about the task(s) being performed. For instance, humans are aware of their limitations
and some of the robot’s limitations and can design policies to reduce the impact of these limitations on the system performance. By incorporating human experience into the system, the human-robot team is able to accommodate some of the challenges in the task operation and environment.

The policies govern assignments under specific conditions. We have identified three categories governing the strength of the assignment. Because we want the system to provide as much flexibility as possible, we try to avoid strict assignments and focus on which agents could, can, and should perform the task. The distinction between could, can, and should is subtle, but important. An agent could perform a task if it has the physical or computational capabilities to perform that task. The agent can perform that task if the situational conditions are such that the agent is still capable. If the agent is the best agent to perform that task under the given conditions, then that agent should perform the task. Policies can be either can or should policies. Whether an agent could perform a task is hard-coded into the system, as this will not change during operation (unless there is a physical or computational change in the agent). As expected, should policies are stricter than can policies because they provide fewer options than a can policy.

In our scenario there are two agents: a human and an autonomous vehicle. For the task of controlling the vehicle speed under slippery conditions, we could have the following sets: could: {human, vehicle}; can: {human, vehicle}; should: {human}. In this case, the human and autonomous vehicle are physically and computationally capable of performing the given task, even under slippery conditions. However, with a policy that says: “if the road is slippery, the human should drive,” the human is identified as the best agent to perform this task under these conditions and the task of steering control is assigned to the human.

For the same task but different conditions (a bumpy road), we could have the following sets: could: {human, vehicle}; can: {human, vehicle}; should: {}. Under these conditions, the human and vehicle are equally capable of performing the task while neither is preferred and so both are listed in the can set. Because it is a task only performed by one agent, the list of possible agents will be passed to the Autonomy Level Assessor which will use scoring to determine whether the human or vehicle will be assigned the task.

Initial policies are essentially educated guesses by the human programmer about what guidelines should be used to ensure a reasonable performance of the system, so we allow the policies to be modified and new policies to be added during system operation. This is done through our human user
interface which is described in Section V.

C. Scoring Tasks

Sometimes the policies do not dictate an absolute assignment; as seen in the second example above, a policy may only reduce the set of possible agents. But if more than one agent is possible, which agent should perform the task? For this, we use the Autonomy Level Assessor.

The Autonomy Level Assessor maintains a list of the current assignments (from the Task Assignment Coordinator) and a list of all possible tasks. It has a set of values for each agent for all possible tasks that it uses to evaluate and assign. For our case, with one human and one robot, there are two sets: one for the human and one for the robot. The Autonomy Level Assessor uses a function to determine the score for an agent and task by inputing the values for that agent and task into a function.

There are several elements that are important to the system for scoring. Their values are task dependent, but the elements apply to many scenarios. They are:

- **Workload.** The workload the task generates for the agent.
- **Skill.** How skilled the agent is at performing the task.
- **Preference.** How much the agent prefers to perform the task. For example, humans enjoy some tasks more than others. For tasks they enjoy, their preference to perform the task is higher.
- **Priority.** How important the task is to mission completion.
- **Resistance.** A piecewise function where resistance = 1 if the agent is performing the task, and 0 if the agent is not performing the task. This term helps prevent assignment chattering.

Currently, these values are hard-coded into our system. Improving the computation of these values is discussed in Section VI.

Our scoring function is a function of these variables; we currently use a weighted sum:

\[
\text{score} = \alpha \frac{1}{\text{workload}} + \beta \text{skill} + \gamma \text{preference} + \epsilon \text{priority} + \lambda \text{resistance}
\]

(1)

where \(\alpha, \beta, \gamma, \epsilon,\) and \(\lambda\) are constants. The agent with the highest score is assigned the task.

D. System Operation

During operation, the system constantly monitors situation information and agent performance and maintains a list of all tasks and current task assignments. Each task reassignment has two triggers:
the situation has changed so the current task assignments violate the policies, or the Autonomy Level Assessor determines the system is not performing as well as desired (e.g., if a human agent’s workload is too high). In both cases, steps for assignment are the same (Figure 2).

The first step is to check the policies for any required assignments. If there are any, these are stored by the Task Assignment Coordinator. The Autonomy Level Assessor scores the remaining tasks to determine which agent should be assigned each task. As tasks are assigned through scoring, they are checked against the policies to ensure that no conflicts arise from these assignments. Scoring and policy checking continue until all the tasks have been assigned. Because tasks have corresponding priorities, the agents will perform the tasks with the highest priority first.

Fig. 2. Flowchart for task assignment. The situation information affects which policies are activated. Any tasks that are explicitly assigned by the policies are recorded. Any tasks that have not been assigned are scored and assigned through scoring. These assignments are checked against the policies to verify that the assignments do not violate the policies.

Our mixed-initiative adjustable autonomy approach allows us to monitor the progress and performance of both the human and robotic agents and determine when and how the autonomy levels must change. It is not mandatory that the change in autonomy be initiated by a human; the system, using the information available to it, can also adjust autonomy levels. We have also structured our system so the human is able to provide guidance in the form of both policies and Autonomy Level Assessor values, increasing the human’s trust in the system. As a result, the human/unmanned system team can act more efficiently and effectively rather than simply improving the performance of one of the agents at a cost to the team as a whole.

V. INTERFACE DESIGN

As with tasks and policies, interfaces are scenario specific; however, several key features of our human interface design translate to other scenarios and systems that may use our mixed-initiative adjustable autonomy approach. These features are based on what we identified as necessary information
for informed, confident operation. They are:

- **A list of the tasks the human should perform.** To avoid assignment misunderstandings, the human must be aware of what tasks the MIAA system believes the human should perform. This is particularly useful for scenarios with many smaller tasks to be performed rather than scenarios with one or two larger tasks that completely occupy the agents.

- **Situational awareness information.** To perform the listed tasks, humans, robots, and the mixed-initiative adjustable autonomy system use information about the scenario and situation to make informed decisions.

- **Channels for communication between the human, the robot, and the mixed-initiative adjustable autonomy system.** The human must be able to provide input into the mixed-initiative adjustable autonomy system and share information (through the system) with the robot.

These key features drove our interface design. The resulting interface is shown in Figure 3. We designed a four-panel interface: the leftmost panel shows the list of tasks the human is performing, the middle two panels provide situation awareness information, and the rightmost panel provides the human with a place to communicate with the mixed-initiative adjustable autonomy system.

![Fig. 3. Our human interface. The interface shows the information necessary for the human to perform tasks and communicate with and understand both the MIAA system and the human’s teammates.](image)

For the leftmost panel (the list of tasks) we also provide the user with access to the list of the unmanned system’s assigned tasks. While the default display shows the human’s tasks, the robot’s tasks can be viewed by clicking a button (Figure 4). This feature promotes the human’s trust in the robot. By viewing the robot’s list, the human knows what type of behaviors to expect from the unmanned system.

The middle panels provide the human with the information necessary to perform the tasks. The second panel from the left reconfigures to show information relevant to the list of tasks in the leftmost
panel. In Figure 4, this panel shows the current speed next to the speed limit for comparison and also shows that the lights and turn signals are on. The third panel from the left shows the current, future, and past waypoints for the vehicle on a map of the environment. Although it updates as the vehicle moves, this panel is the most static of the four because it always shows position and waypoint information.

The right panel is the most flexible and varied. It is the communication portal between the human agent and the MIAA system. This is where the human enters information into the system, for example to add or modify a policy or to change the scoring values in the Autonomy Level Assessor. It is also designed to allow the human to garner as much information about the system as possible. This increases the human’s trust in the system and robot by presenting any information the human cannot get from the panels to the left, such as a list of scoring parameter values.

Fig. 4. Snapshot of our human interface showing how policies can be added or modified using the rightmost panel. The interface always shows the same four panels, but the content of the panels is scenario and task specific as seen when this figure is compared to Figure 3. In Figure 3, the rightmost panel shows the MIAA system asking the human a question; in this figure, the human is adding or adjusting policies.

In combination, the four interface panels provide the user with all the required information. While the human may not be able to constantly check all of the information, it is available if needed. It is left to the user to decide what information is shown; any windows selected will automatically remain open.

VI. FUTURE WORK AND CONCLUSIONS

We are in the process of preparing for experiments with humans using our simulated driving platform. These experiments are being designed to establish human’s perceptions of trust, workload, and ease of operation. From these experiments we will evaluate the MIAA system, specifically focusing on how successfully the policies and scoring system assign tasks in dynamic situations.
We plan to improve how we determine the values for the scoring function. Currently the values are estimated and fixed; we plan to extend our approach to enable calculation and modeling of these values. While we currently assume the workload for each task is independent of the conditions and other tasks being performed, a more sophisticated model would take the situation into account. For human workload levels, we plan to use physiological data to measure the human’s workload. An agent’s skill could be evaluated and monitored by the system. This should be periodically (if not continuously) evaluated and updated because an agent’s skill changes during task performance. Preference could also be modeled; this could be based on how often (and under what circumstances) an agent refuses to perform a task assigned to it. A sophisticated model of preference would be user-specific and would likely require training based on feedback from the user. Like workload, skill, and preference, priority could also be modeled based on the conditions and the other tasks being performed. Under different conditions and during the performance of different tasks, a task’s priority is likely to fluctuate. This is the most difficult of our values to model because it is a subjective parameter; different users prioritize tasks differently. An open question is how to accommodate personal preferences with the overall goals of the team.

We also plan to extend our approach to facilitate the coordination of many agents. We have focused on a single human-single unmanned system scenario but plan to extend our approach for use in multiple human-multiple unmanned systems scenarios. Multiple robot scenarios have generated considerable interest and the extension of our mixed-initiative adjustable autonomy approach could greatly increase the capabilities of this type of team.

Ultimately, mixed-initiative adjustable autonomy has the potential to facilitate the incorporation of many different unmanned systems into scenarios where current technology would limit them. We propose an approach and system that allows humans and unmanned systems to work as a team, leveraging the strengths of each team member and overcoming some of their weaknesses. In doing so, we enable the whole team to act more efficiently and effectively and therefore increase its applicability.

REFERENCES


