Policy-Based Design of Human-Machine Collaboration in Manned Space Missions

Jurriaan van Diggelen
Utrecht University, ICS
The Netherlands
jurriaan@cs.uu.nl

Tim Grant
Science & Technology b.v.
The Netherlands
tgrant@cs.up.ac.za

Mark Neerincx
TNO / Delft University of Technology
The Netherlands
mark.neerincx@tno.nl

***Abstract*— This paper presents a policy-based approach to designing teams and collaboration support in manned planetary exploration missions.**

I. Introduction

Manned space missions invariably assume a hierarchical organization, giving control to mission control on Earth, while leaving little autonomy to the astronauts. This type of organization has its roots in the military and offers clear advantages in terms of attributing responsibility and creating a sense of control over an uncertain environment. However, communication in space missions can be severely complicated due to latency, delays and interruptions. For example, 20 to 50 minutes radio traffic delay between Earth and Mars makes it impossible for astronauts on Mars to discuss time-critical problems with Mission Control on Earth. This raises the need for new organizational forms and support tools that allow astronauts to act more autonomously [1].

This paper presents the development of an implemented prototype of a mixed human and machine team. This effort forms part of the iterative development process of a Mission Execution Crew Assistant (MECA) [2]. First, we use the prototype as a way to extract requirements for the design of resilient intra- and inter-team collaboration in MECA. Then, we use these requirements to develop an improved teamwork prototype assessing its resilience (e.g., to delays and interruptions) and investigating novel techniques to support the collaboration during off-nominal operations.

II. Background

We develop the teamwork models by writing KAoS policies, i.e. behavioral constraints that are imposed and enforced prescriptively on agents [3]. In this way, we force agents to give up some degree of individual autonomy in favour of collective team objectives. For example, a policy might state that a team member is obliged to send a notification to the requester after it has finished performing a requested action. Policies are written in OWL (Web Ontology Language) and are automatically distributed and enforced by KAoS.

III. Designing the MECA team

We design the MECA team by focusing on a use case of a hypothermic astronaut who needs to be brought to the habitat for recovery. The joint team objective, i.e. ensuring safety, is modeled as a collective obligation (CO). The policies serve to translate the CO (what the team as a whole is obliged to do) to individual obligations (what an individual is obliged to do). In total, we have written fifteen policies. Some of these can be enabled or disabled to regulate the degree of team member autonomy in three dimensions: leadership assumption, plan coordination, and task allocation. This allows us to create eight different teams, which all successfully fulfill the team objective of rescuing the astronaut. However, as the different event traces show, all teams function differently. Some teams heavily rely on a central authority, whereas others leave more room for member autonomy.

IV. Discussion

A. Extracting Requirements

Our implementation of different teamwork models allows us to analyze team performance by running simulations. We can apply different validation criteria, such as robustness against communication delays, against inaccessible agents, and against network interruptions. From the simulations, we are deriving requirements for an improved teamwork model for MECA.

B. System Development

Our approach to teamwork also offers many opportunities for developing collaboration support. Firstly, as many team policies describe which information team members must share with each other, a computer can now take over this task from humans. Secondly, as the team policies describe important aspects of teamwork, they can make computers more predictable and understandable to humans. Thirdly, we can use the model to enable computers to recognize when policies are not met and to start recovery or contingency planning.

V. Conclusion

Team design is an important issue in manned planetary exploration. We have demonstrated the feasibility of studying team design by using a computational policy approach for requirements engineering and system implementation.

We plan to develop more complex teamwork models, to further investigate ways to validate and compare different teamwork models, and to find ways to recognize policy breaches and to enforce policies, e.g. using doctrine.

VI. References

[1] Grant, T.J., et al. (2006) Space Autonomy as Migration of Functionality: The Mars case. Proceedings of SMC-IT 2006.
[2] Neerincx, M.A. et al. (2008). The Mission Execution Crew Assistant: Improving Human-Machine Team Resilience for Long Duration Missions. Proceedings of IAC2008
[3] Bradshaw, J. M., et al. (2004). Making agents acceptable to people. In Intelligent Technologies for Information Analysis. (pp. 355-400). Springer Verlag