Human Robot Team Development: An Operational and Technical Perspective

Jurriaan van Diggelen(✉), Rosemarijn Looije, Jasper van der Waa, and Mark Neerincx

TNO, Kampweg 5, 3769 DE Soesterberg, The Netherlands
{jurriaan.vandiggelen,rosemarijn.looije,jasper.vanderwaa,
mark.neerincx}@tno.nl

Abstract. Turning a robot into an effective team-player requires continuous adaptation during its lifecycle to human team-members, tasks, and the technological environment. This paper proposes a concept for human-robot team development over longer periods of time and discusses technological and operational implications. From an operational perspective, we discuss the types of adaptations to team behavior that are required in a military house search scenario. From a technological perspective, we explain how teamwork adaptations can be implemented using a teamwork module based on ontologies and policies. The approach is demonstrated in a virtual environment, in which humans and robots collaborate to find objects in a house search.

Keywords: Human robot teaming · Policies · Defense

1 Introduction

Turning a robot into an effective team player cannot be completely realized at design time. This is because many of its behavior requirements only become apparent after the system has been deployed. To illustrate this point, we consider a use case in which robots assist soldiers during a house search for explosive materials. To successfully participate in this mission, the robot must possess a diverse set of communication skills, e.g. for deciding to whom it should report its findings, or whether it can pick up some object without permission. It is highly unlikely that these behaviors have been perfectly pre-programmed by the robot development firm when the robot was delivered [1]. Therefore, they must be adaptable by the end user without the need of changing code.

As outlined in Fig. 1, we distinguish between three functional layers when designing human robot teams (HRT's). The lowest layer concerns robotic, human or joint activities of the team-members. The middle layer (*teamwork*) concerns all coordination activities that are required to enable these activities (e.g. allocating agents to tasks, or ensuring common ground [2]). These activities can be performed by one dedicated member or by all members of the HRT [3]. The upper layer concerns the functionality concerned with HRT development, i.e. monitoring and adapting the coordination activities to better match current circumstances.

J. Chen (ed.), *Advances in Human Factors in Robots and Unmanned Systems*,
Advances in Intelligent Systems and Computing 595, DOI 10.1007/978-3-319-60384-1_28

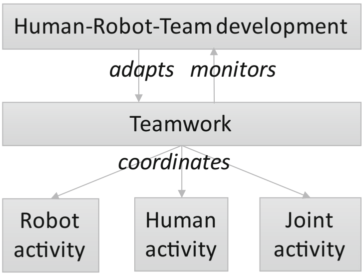

Fig. 1. Functional framework for human robot teamwork

In this paper, the focus is on HRT development which will be discussed from an operational and technological perspective. The operational perspective is illustrated in a military use case in which robots are allowed to participate in a debriefing of a mission. One of the goals of a debriefing is to allow the team to develop by sharing positive and negative experiences. In the case of the robotic participants, this means that explicit working agreements can be established between humans and robots which guides future behavior [4]. In this way, the functioning of the human-robot team is expected to improve as the team becomes more experienced.

The technological requirements are largely driven by this use case. To represent working agreements in a machine readable way, we have adopted a policy-based approach. Policies are a generic way to specify and govern an agent's behavior using rules for permissions and obligations. To make the policy engine applicable to our use case, we have built ontologies and a policy engine implemented in the Drools expert system [5]. The ontologies define the domain specific terms that are needed to specify relevant working agreements. For example, they specify what qualifies as a dangerous object. The policy engine defines rules of a specific format in a way that is computable and easily understandable to non-expert users. For example, using the ontologies and policy engine we can ensure that "the robot is not allowed to pick up dangerous objects" is understood by the robot.

The main contributions of this paper are an operationally relevant scenario containing debriefings for human-robot team (HRT) development, and a technological approach for HRT development based on ontologies and a rule-based policy engine.

We have tested our approach for HRT development with domain experts in the field using an implemented demonstrator. We have implemented the working agreements in a policy engine, and local agent behavior using behavior tree in a virtual environment. We implemented a test environment in which a soldier and robot jointly perform a house search and engage in a debriefing afterwards for HRT development.

The paper is organized as follows. Section 2 discusses related work. Section 3 discusses the operational aspects of HRT development using scenarios and possible working agreements. Section 4 describes our policy language and ontologies that we use to formalize these working agreements. Section 5 describes a demonstrator, followed by a conclusion in Sect. 6.

2 Related Work

The operational benefits of integrating robots in tactical teams (such as Special Weapons and Tactics (SWAT) [6], or Search and Rescue teams [7]) have been widely recognized. The most prominent reasons are enhanced capabilities offered by robots (e.g. using sensors such as radar or laser, or using strong robotic arms), and enhanced human safety by using robots as forward observers and allocating dangerous tasks to robots.

Most current applications of robots in the defense domain still require a high degree of operator involvement (e.g. tele-operated drones), and usually regard robots as a tool used by humans [8]. On the other hand observations and interviews with Explosive Ordinance Disposal (EOD) personnel [9] and industrial workers [10] who work in close cooperation with a robot show that people are inherently social creatures and attribute social features to a robot even when it is not autonomous. The expectation of researchers is that increased autonomous capabilities will increase the attribution of social features to robots. They expect that this will happen to such an extent that robots will be regarded as a teammate of humans, freeing up valuable human resources for other tasks. A practical military application of this idea is known as *manned-unmanned teaming (MUM-T)* where helicopter- and fighter jet pilots collaborate with UAV's during their operations [11].

From the academic community, the idea of human-agent teaming is quite old.

The first papers appeared in the early nineties and focused on capturing team properties in logical formalisms [12]. Based on these formalizations, executional concepts were soon proposed. Examples are agent communication languages such as KQML [13], which allow software agents to communicate on a higher level of abstraction. Another example is SharedPlans [14], which allows software agents to establish a common team plan. Another area of research on teamwork technology focusses on policies or norms as a way to govern and constrain the behavior of autonomous systems such that uncoordinated activities are ruled out [15]. This research area is still very active; a number of current challenges are reported in [16].

This paper builds upon insights from these technical and operational/human factors communities.

3 Operational Perspective

This section explains the steps we have taken to obtain the operational perspective of HRT. Because teamwork may occur in many forms, we have focused on one particular form of working together and described it in a scenario.

Relating to Fig. 1, the scenario should describe:

- The *coordination* that is required from the *teamwork* module. This includes all coordination activities, such as ordering, notifying, prohibiting, from robot to operator and from operator to robot.
- The *adaptation* and *monitoring* that is required from the *HRT development* module. This includes deciding when and how often the teamwork adaptation should take place, and which types of adaptations can be expected.

3.1 Scenario

In interaction with domain experts, we have developed the following scenario.

A house search is ordered for a house where there is a supposition of explosive materials. A tactical site exploitation team is deployed, the team consists of a team leader (TL), assistant team leader, and three search teams each consisting of two actors. The focus in the rest of the scenario will be on one of the search teams which consists of a human soldier (S1) and a robot (R1); search team 1 (ST1).

All search teams receive information from the TL about the situation (no people or boobytraps expected, two story house with stairs after entrance and several rooms on the ground floor, consolidation point for carefully retrieved evidence is outside the house).

Then ST1 receives instructions from the TL. ST1 is the first team to enter the house, it should check the hallway and route to stairs, check if provided layout of ground floor is correct and search the entrance. Inform me (the team leader) when route to stairs is clear.

ST1 walks to door, R1 waits on direct order from S1 as it is prohibited to enter without approval. S1 provides order to enter and check route to stairs. R1 enters hallway and checks route to stairs, informs S1 that situation is clear. S1 also enters and informs TL that route is clear. S1 checks ground floor plan and informs TL that it is according to intelligence. S1 then provides R1 with new shared task; in the middle of the room there is a consolidation point for evidence, you look clockwise, I'll look counterclockwise. R1 provides updates when it sees something (e.g. table, no danger). When room is clear according to R1 it reports this to S1, when S1 is also finished it reports the conclusion to TL. They then proceed to the next room, as ordered by TL, S1 enters R1 follows. S1 again introduces the consolidation point for this room and divides the task in clockwise (R1) and counterclockwise (S1). R1 sees something that might be an explosive, as it does not have the capabilities to investigate this it informs S1. S1 is busy with another object and does not react on R1. R1 then informs TL who sends S2 to the room to support.

After the house search is finished the complete team sits together for a debrief.

In this scenario R1 is prohibited to enter a room and obliged to inform S1 of everything. It is prohibited to inform TL directly unless S1 does not react. Adaptation of these policies can occur during the scenario, e.g. when S1 gets irritated about the updates S1 can say S1 prefers only updates on dangerous objects and not of everything. Another adaptation opportunity is during the debrief, where the decision can be weight between the different team members.

3.2 Working Agreements

After development of the scenario it was formalized in smaller steps (e.g. S1 provides new assignment to R1). These steps were then analyzed further, describing the specific working agreements they might require.

A situation as "before door" requires the following working agreement: If I (the robot) see a closed door or entrance to another room on the route, I will wait for instructions unless my human team member walks on or I already received an assignment for this entrance.

"Can't do it" requires the working agreement: If I can't perform my assignment I report to my human team member. Which in its turn requires the working agreement on reporting to S1. A working agreement that is contained in "report to human team member" is a further refinement that says: If my team member does not react I'll rapport to the team leader.

A prerequisite for all these working agreements is that there is also an ontology present that for example defines *team member*. Furthermore, all working agreements must be described in such a way that they can be adapted, changed or refined. The first working agreement can for instance be changed so that the robot always walks in a room before the human team member. When the robot encounters a situation where it is not able to perform its assignment it could have the refinement of its working agreement that it informs S1 unless S1 is busy and it will then inform TL directly.

4 Technological Perspective

4.1 Hybrid Agent Architecture Model

The Hybrid Agent Architecture Model (HAAM) is shown in Fig. 2. Contrary to the *functional model* shown in Fig. 1, HAAM is a *system model* serving to identify the main components which are relevant when building Human Agent Teamwork.

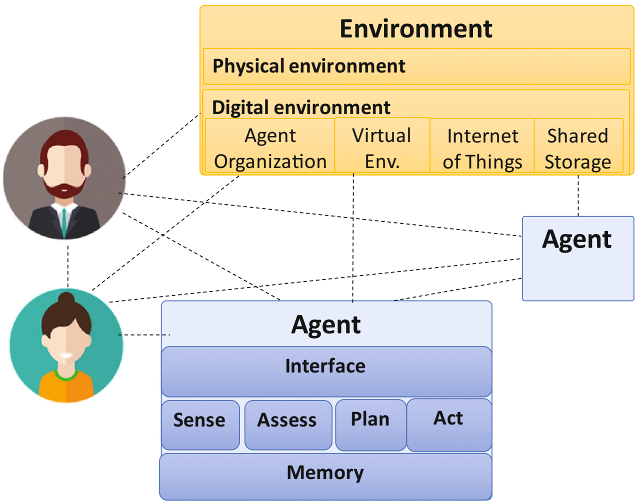

Fig. 2. Hybrid Agent Architecture Model (HAAM)

The main components of HAAM are humans, agents and a digital environment. Interaction may occur between all components (as indicated by the dashed lines). In our case, the humans are the members of the search teams, that collaborate with robots. The agents are embodied as robots (UGV's), and interact with each other and with humans. Agents and humans also interact with their environment, which consists of a physical parts (which we don't have to design, and hence is not specified further in this model), and a digital part (which we can design). Depending on the application, the digital environment specifies various shared components establishing common ground between the agents.

For our application, a shared agent organizational model is implemented as part of the digital environment, to specify the roles, permissions and obligations that are required for agent coordination. These policies can intervene directly in an agent's action selection cycle (sense; assess; plan; act), for example by blocking or enforcing agent's actions. The agent organization model and the agent's internal action selection cycle are discussed in Sects. 4.2 and 4.3 respectively. To test different concepts early in the development cycle, we have adopted a virtual environment (as a component in the digital environment, which will be discussed in Sect. 4.4.

4.2 Agent Organization

The agent organization model coordinates communication and activities. We follow a policy-based approach (similar to [17]). The policy engine uses contextual information and a set of working agreements to intervene in an agent's action selection cycle.

Working agreements (or policies) are formatted as rules in the form "*if* <*conditions are met*> *then* <*create policy decision*>". The conditions of a policy may contains contextual information which is provided by agents or the environment. When the conditions of a policy are met, a policy decision is created that states whether an action is obligated or prohibited. An example of a policy is:

If *a robot finds a dangerous object and can secure the object* ***then*** *the robot must ask its team member if it is allowed to secure that object.*

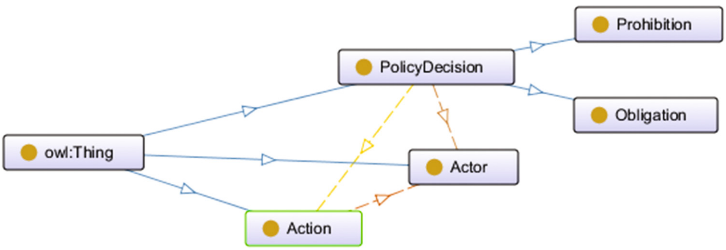

Fig. 3. The core ontology of the agent organization model

The policies are specified in the Drools expert system language. The information itself is structured according to a set of ontologies [18]. These define the common concepts between agents. The policy-engine has a single core ontology that defines the basic concepts of an action, actor and policy decision which can be obligations or prohibitions (see Fig. 3).

Besides the core ontology, which contains the most abstract and generic concepts required for HRT development, several domain specific ontologies specify the concepts that are needed to define the specific types of working agreements discussed in the previous Section.

4.3 Agent Action Selection

In our demonstration we used simulated robots in a virtual environment whose action selection loops are implemented with behavior trees [19]. A behavior tree is a tree-like graph where each node resembles an action, a condition check or a method how to propagate through the tree below it (e.g. breadth-first or depth-first). The agent's behavior is defined by propagating through the tree and perform each node's instruction. The resulting action chain can depend on environmental conditions, on previous actions or entirely external influences.

The agent organization model (i.e. the policy engine described in Sect. 4.2) intervenes in the action nodes; when an action node becomes active a query is send to the model and the policy decision is returned. In the case of a prohibited action, a different node becomes active either due to regular propagation through the tree or when the policy decision is accompanied by an obligated action. For example if a policy prohibits the action of securing a dangerous object, an additional policy may become active that obligates the agent in keeping its distance and send this action to the agent. Figure 4 shows a portion of the implemented behavior tree. The tree visualizes the real-time state of the agent's behavior as the game is played. The highlighted nodes are currently active.

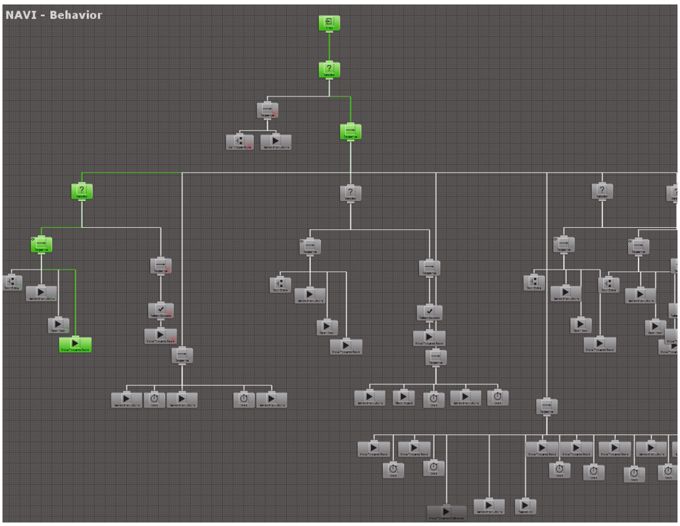

Fig. 4. A portion of the implemented behavior tree

The leaf node of the highlighted sub-tree is the node that is currently being executed. Nodes with a cross are rejected due to prohibiting policy decisions.

4.4 Virtual Environment

To test the human robot teamwork at an early stage of development, we implemented the house search scenario in a virtual environment (i.e. Unity [20]). A human participant can perform a house search form his or her own first person view. An autonomous robot helps the human in performing this task. Both robot and human can communicate with each other. The figure below shows the perspective of the robot. Note that this perspective is not visible by the human performing the house search, but can be generated for demonstration purposes (Fig. 5).

Fig. 5. Robot view of robot perceiving a black box

In the example above, the robot perceives a black box. Upon finding this box, ontology reasoning is used to classify the black box as a dangerous object, after which

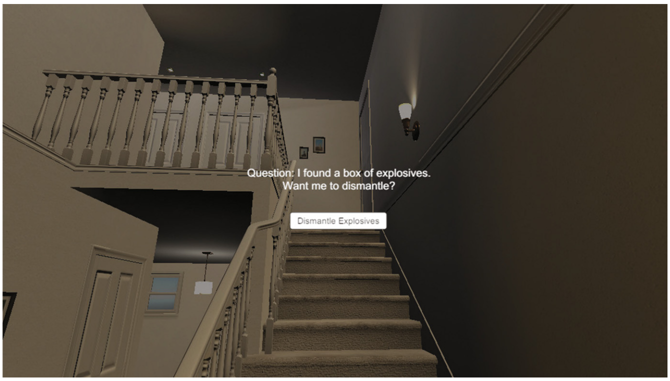

Fig. 6. Human view of human receiving a request from the robot

the policy engine fires a notification policy (discussed in Sect. 4.2) that this should be communicated to the operator. This is shown in the figure below (Fig. 6).

The human and robot can search the house in this way as a team, engaging in continuous communication. After their assignment is completed, the human and robot have the opportunity to reflect on their cooperation and may choose to adjust the policies that drive their communication, i.e. HRT development.

5 Conclusion

This paper presents an operational and technological concept for HRT development. The operational challenge is to enable operators to invest in improvements in team coordination over the long term, while maintaining their primary focus on the task at hand. We have developed a military house search scenario, and identified several types of human-robot team adaptations that are desirable in this domain. We propose that these adaptations should be pursued during the debriefing. From a technological perspective, we have proposed an ontological policy-based approach, where ontologies and a rule-based policy enforcement mechanisms serve as the primary means to represent and reason with these adaptations. The policies should be understandable by humans and machines. To enable human readability, we propose to use Domain Specific Languages. To enable machine-readability, we propose to use ontologies which are grounded in the underlying behavior mechanism (in our case behavior trees).

In the future, we intend to further investigate HRT development using this approach. We plan to develop the test environment further to also measure team performance. This enables measuring the effects of HRT development over longer periods of time.

References

1. Bradshaw, J.M., Hoffman, R.R., Woods, D.D., Johnson, M.: The Seven deadly myths of “autonomous systems”. IEEE Intell. Syst. **28**(3), 54–61 (2013)
2. Klein, G., Woods, D.D., Bradshaw, J.M., Hoffman, R.R., Feltovich, P.J.: Ten challenges for making automation a “team player” in joint human-agent activity. IEEE Intell. Syst. **19**(6), 91–95 (2004)
3. Van Diggelen, J., Bradshaw, J.M., Johnson, M., Uszok, A., Feltovich, P.J.: Implementing collective obligations in human-agent teams using KAoS policies. In: Padget, J., Artikis, A., Vasconcelos, W., Stathis, K., da Silva, V.T., Matson, E., Polleres, A. (eds.) Coordination, Organizations, Institutions and Norms in Agent Systems V, pp. 36–52. Springer, Berlin (2010)
4. Neerincx, M.A., van Diggelen, J., van Breda, L.: Interaction design patterns for adaptive human-agent-robot teamwork in high-risk domains. In: Harris, D. (ed.) International Conference on Engineering Psychology and Cognitive Ergonomics, pp. 211–220. Springer, Cham (2016)
5. Drools: https://www.drools.org/
6. Asif, K.I., Bethel, C.L., Carruth, D.W.: Iterative interface design for robot integration with tactical teams. In: Savage-Knepshield, P., Chen, J. (eds.) Advances in Human Factors in Robots and Unmanned Systems, pp. 3–16. Springer, Cham (2017)

7. Kruijff, G.J.M., Janíček, M., Keshavdas, S., Larochelle, B., Zender, H., Smets, N.J., Mioch, T., Neerincx, M.A., et al.: Experience in system design for human-robot teaming in urban search and rescue. In: Yoshida, K., Tadokoro, S. (eds.) Field and Service Robotics, pp. 111–125. Springer, Berlin (2014)
8. Murphy, R.: Introduction to AI Robotics. MIT Press, Cambridge (2000)
9. Carpenter, J.: Culture and Human-Robot Interaction in Militarized Spaces: A War Story. Routledge, Abingdon (2016)
10. Sauppé, A., Mutlu, B.: The social impact of a robot co-worker in industrial settings. In: Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems, pp. 3613–3622. ACM (2015)
11. Strenzke, R., Uhrmann, J., Benzler, A., Maiwald, F., Rauschert, A., Schulte, A.: Managing cockpit crew excess task load in military manned-unmanned teaming missions by dual-mode cognitive automation approaches. In: AIAA Guidance, Navigation, and Control Conference, p. 6237 (2011)
12. Cohen, P.R., Levesque, H.J.: Intention is choice with commitment. Artif. Intell. **42**(2–3), 213–261 (1990)
13. Finin, T., Fritzson, R., McKay, D., McEntire, R.: KQML as an agent communication language. In: Proceedings of the Third International Conference on Information and Knowledge Management, pp. 456–463. ACM (1994)
14. Tambe, M.: Towards flexible teamwork. J. Artif. Intell. Res. **7**, 83–124 (1997)
15. Bradshaw, J. M., Feltovich, P. J., Johnson, M. J., Bunch, L., Breedy, M. R., Eskridge, T., Jung, H., Lott, J., et al.: Coordination in human-agent-robot teamwork. In: International Symposium on Collaborative Technologies and Systems, 2008. CTS 2008, pp. 467–476. IEEE (2008)
16. Bradshaw, J.M., Montanari, R., Uszok, A.: Policy-based governance of complex distributed systems: what past trends can teach us about future requirements. In: Adaptive, Dynamic, and Resilient Systems, pp. 259–284. Auerbach Publications (2014)
17. Uszok, A., Bradshaw, J.M., Johnson, M., Jeffers, R., Tate, A., Dalton, J., Aitken, S.: KAoS policy management for semantic web services. IEEE Intell. Syst. **19**(4), 32–41 (2004)
18. Guarino, N.: Formal ontology, conceptual analysis and knowledge representation. Int. J. Hum. Comput. Stud. **43**(5–6), 625–640 (1995)
19. Marzinotto, A., Colledanchise, M., Smith, C., Ögren, P.: Towards a unified behavior trees framework for robot control. In: 2014 IEEE International Conference on Robotics and Automation (ICRA), pp. 5420–5427. IEEE (2014)
20. Unity: https://unity3d.com/