Design Patterns and Ontologies for Situated Human-Agent Collaboration at Organization, Team and Interaction Level

Mark A. Neerincx[1,2] and Jurriaan van Diggelen[1], Martin L. Stolk[1]
[1]TNO Perceptual and Cognitive Systems, Kampweg 5, 3769 DE Soesterberg
[2]Delft University of Technology, Faculty of EEMCS, Van Mourik Broekmanweg 6, 2628 XE Delft
THE NETHERLANDS

<mark.neerincx, jurriaan.vandiggelen, martin.stolk>@tno.nl

ABSTRACT

For the integration of robots into the operations of rescue, reconnaissance, security and defence teams, there is a need for theoretical and empirical grounded human-agent collaboration models. This paper proposes the development of organization, team and interaction design patterns with corresponding ontologies, as re-usable building blocks of human-agent collaboration. A situated Cognitive Engineering (sCE) methodology supports the incremental development and implementation of these building blocks (i.e., ePartner abilities and behaviours) into a perceptual-cognitive agent framework. Core cognition concepts are formalized as ontology design patterns (e.g., "workload" and "situation awareness") and related to the concerning domain ontology (e.g., disaster response with concepts like "point of interest" and "urgency"). In parallel, relevant design patterns are identified or constructed with the corresponding ontologies, at the organization (e.g., norms), team (e.g., work agreements) and interaction level (e.g. explanation). The ontologies are implemented into the prototype and tested on their expected outcomes (i.e. the claims). The European TRADR project (www.tradr-project.eu) provides an example in which an agent-based system for robot-assisted disaster response is developed, where response team dynamically adapt the human-robot teamwork (i.e., task allocation and coordination) to the momentary context.

1.0 INTRODUCTION

Ongoing progress in artificial intelligence, robotics, sensors and network technology feeds the development of software agents that can act as teammates or so-called *ePartners.* These ePartners are applied in defense, safety and security missions [1],[2],[3] and operate robustly in hazardous and chaotic environments. They are supposed to show responsible, agile and flexible behaviors in response to foreseen *and* unforeseen events. Furthermore, their behaviors should be fluently integrated into the overall human-agent teamwork, and harmonized with the activities and expectations of the human teammates. To establish the envisioned human-ePartner collaboration, ePartners require an understanding of human's social, cognitive, affective and physical behaviors, and the ability to engage in partnership interactions (such as explanations of task performances, and the establishment of joint goals and work agreements).

To cope with the design complexity of these systems, theoretically and empirically grounded models are needed that capture these social, cognitive, affective and physical processes. This paper proposes organization, team and interaction design patterns with corresponding ontologies, that can be used as re-usable building blocks for human-agent collaboration in the design process. These patterns are abstractions and generalizations from successful individual design solutions. In our approach, humans and ePartners are part of a joint cognitive system that operates in a specific set of situations. In the system development phase, relevant *situation cognition* models have to be identified or constructed as core concepts of the human-ePartner collaboration. A *domain* model is constructed to explicate how cognition operates in the concerning situated collaborations. Subsequently, the *design patterns* are selected or constructed, which explicate how the actors of the joint cognitive system collaborate successfully in this domain.

As a running example, we will regard an agent-based system for robot-assisted disaster response, as used in the European TRADR project (www.tradr-project.eu). Within this application a response team needs to dynamically adapt the task allocation and coordination to the momentary context [4]. Figure 1 shows the Unmanned Ground Vehicles (UGV) and Unmanned Aerial Vehicles (UAV) that were deployed in the TRADR-project (the photos were taken during a realistic field evaluation of the TRADR-system).

Figure 1: Unmanned Ground Vehicle (UGV; left) and Unmanned Aerial Vehicle (UAV; right) during a field evaluation in the TRADR-project.

Figure 2 presents a part of the disaster response team and its structure (in this example with three human team-members, their ePartners and two robots). The team leader and her ePartner have the lead of two "units", each consisting of an operator, ePartner and robot. The domain model and cognition models of, for example workload, underpin the specification of design patterns at three levels. At the highest level, *organization* design patterns are specified about normative behaviors, such as the work and rest schedules of an organization to comply with a workload norm. At the intermediary level, successful collaborative behaviors of specific *teams* are specified, such as successful work agreements on the task (workload) allocations of a team leader, the two operators, the three ePartners and the two robots in a specific situation. At the lowest level, the *interaction* design patterns specify how the information is communicated between the actors, such as the user interfaces for sharing workload information.

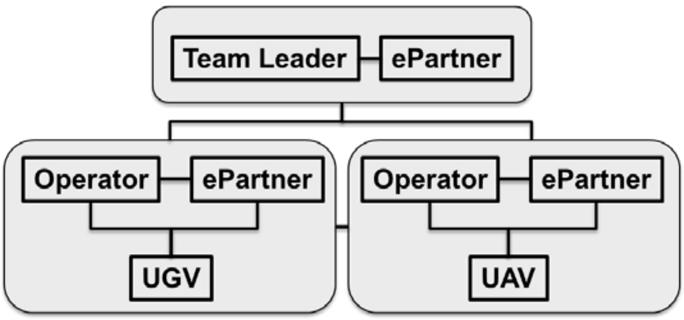

Figure 2: Part of the TRADR disaster response team with a UAV and a UGV.

Section 2 gives a brief overview of a situated Cognitive Engineering (sCE) method for the incremental development of the design patterns with the corresponding domain and cognition models (e.g., "workload" and "situation awareness"), which underpin system's functionality to collaborate. Section 3 discusses a perceptual-cognitive framework to implement this functionality. Subsequently, section 4 elaborates on the development and application of the design patterns, entailing an extendable, theory and empirical grounded, knowledge base of the ePartner functions and behaviors. Section 5 contains the conclusions of this paper.

2.0 SITUATED COGNITIVE ENGINEERING OF E-PARTNERS

Knowledge about the *operational demands*, *human factors* and *technology* (see Fig 3), underpin the requirements specification of the human-agent system. The sCE methodology supports the acquisition, construction and usage of knowledge from these three foundation components.

First, *ontologies* are used as formal, explicit specifications of the knowledge that the agents need to have about the domain ("operational demands") and the human factors (see Fig 3). An ontology is an explicit representation of declarative knowledge. It is structured around concepts, properties, and relations, which allows for automated reasoning (about this structure). For our purposes, the ontology encompasses consensual knowledge that is important for humans-agent collaboration. Also here, we aim at an extendable set of building blocks, i.e., Ontology design Patterns (OP; [5]). They distinguish knowledge at the upper ('classes'), and lower level ('individuals'). To systematically incorporate human factors knowledge, relevant concepts of situated cognition are formalized as an ontological model, such as situation awareness [6] and workload [7]. For example, Harbers and Neerincx [7] provide a workload model, which distinguishes cognitive load, affective load and mental effort, and can reason about task allocations. The upper level workload ontology (or parts of the ontology) has been studied in and applied to several domains: disaster response [8], space missions [9], ship control [10], and train control [7]. During operation, the domain ontology is added to instantiate the concerning concepts, e.g. for the computational reasoning (such as task allocation based on "urgency" and "workload"). Part of the domain ontology for disaster response is shown in figure 4 (for more information on the domain ontology, see [11]). Note that the ontology refers to human factors concepts (like workload) that are specified in the "Situated Cognition" ontologies.

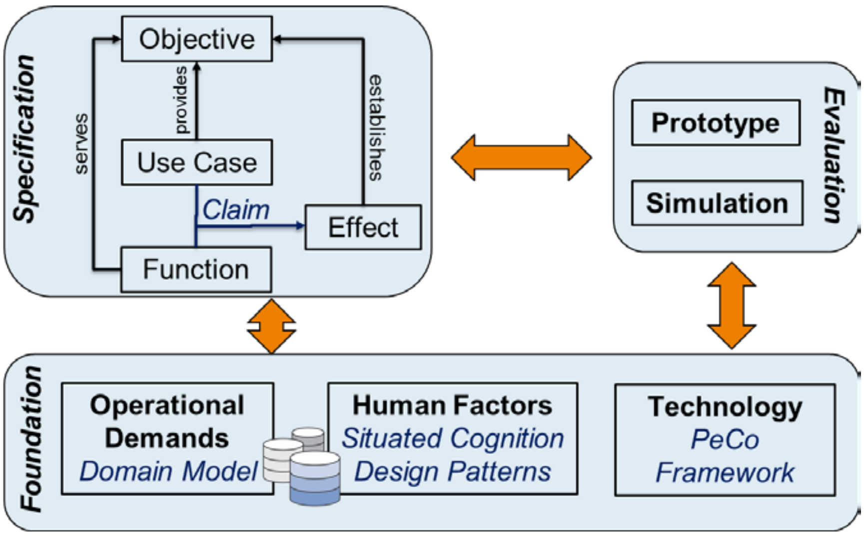

Figure 3: Situated Cognitive Engineering entails an iterative grounding ("foundation"), requirements derivation with its rationale ("specification"), and prototype and simulation testing ("evaluation").

Second, to build on past and current agent systems, and to relate the concerning agent behaviors to each other, it is important to understand and share agents' *design rationales*. In the specification or the functional requirements (see Fig 3), *claims* explicate the design rationale of a specific function: The expected effect of this functionality in a (set of) use case(s). In other words, claims describe the effects of agents behavior on the condition and performance of (the ensemble of) team actors; the effects can entail an expected positive trade-off of the advantages (upsides) over the disadvantages (downsides). Explicating possible downsides is important, because they have to be checked in the evaluation. For example, a contrastive explanation function for a task reallocation advice might be expected to enhance the trust of a team-leader in the advising agent, with only a minor increase of workload. Claims have to be testable and, if appropriate, refer to stakeholders' values.

Third, to generalize over functions and claims, so-called "*design patterns*" are used to capture the "generic" design rationale by explicating common solutions for classes of problems (e.g., as formulated in a use case [12]). As far as possible, proven models and solutions have to be looked for and maintained as an accessible library in the foundation component. Section 4 will elaborate on the identification and construction of design patterns.

Fourth, the design rationales with their claims are evaluated via simulation and prototyping, resulting in approval, rejection or refinement of the concerning models. In TRADR, the ontology, team and interaction design patterns have been developed and tested in several cycles, applying simulations and realistic field tests with a complete team, including unmanned ground vehicles and unmanned aerial vehicles.

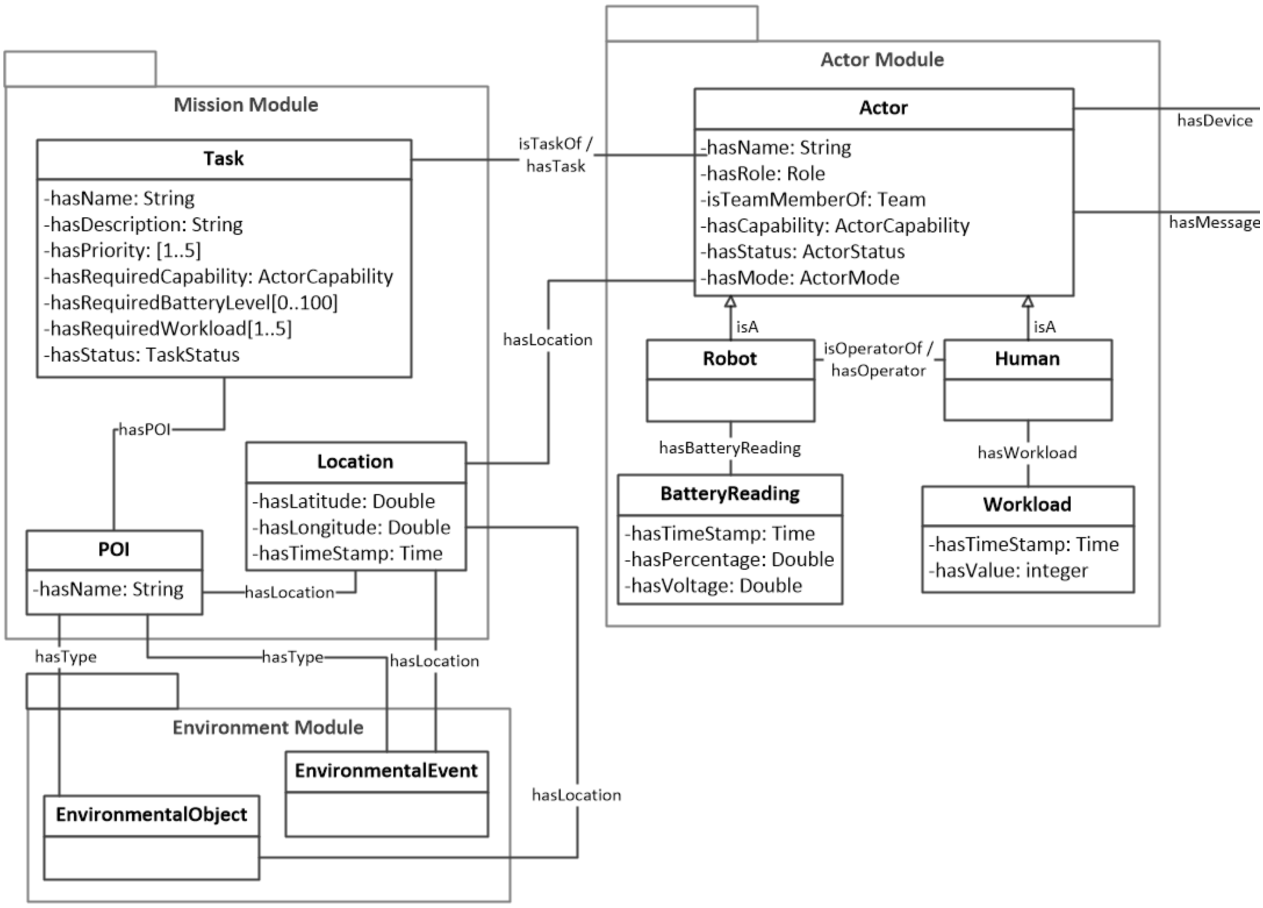

Figure 4: Part of the TRADR ontology (domain model of sCE in Fig 3; more information on the TRADR ontology, see [11]).

3.0 PERCEPTUAL AND COGNITIVE E-PARTNER FRAMEWORK

As a team-member, ePartners need to acquire and show understanding of the meaning, weighting and implementation method of the joint goals they are committed to. This type of understanding and commitment is being built and communicated via three partnership interactions in Human-Agent Teams. Currently, a generic perceptual-cognitive (PeCo) framework-suite is being developed for this purpose ([15], [16], see Fig. 5). This PeCo-framework distinguishes three components: An Objective, Work Agreement and Explanation component.

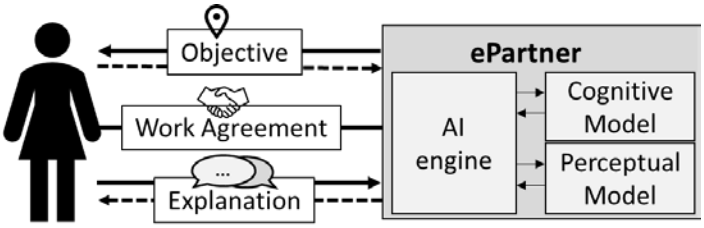

Figure 5: Three human-agent partnership interactions: objective sharing, harmonizing activities via work agreements and explaining task outcomes.

First, active partnerships require the setting, progress monitoring, refinement and adjustment of joint *objectives*, based on a shared understanding, awareness and communication of the objectives and their situated urgencies. Stakeholders' values, strategic intentions and momentary context conditions determine the urgency and feasibility of the objectives to pursue. The choices of the objectives can require re-assessment and adjustment to deal with changing circumstances, possibly just in advance or even during the work processes. Objective ontology design patterns [5] [17] [18] and utility elicitation methods [19] are being developed to support the setting of joint objectives. Adaptivity is an important capability of human-machine teams, which may involve creativity to find a new work process (possibly with an adjusted objective) when the current process is expected to fail. Humans perceive, associate and reason differently than machines; consequently, the perspective-taking and solution space for new (possibly wicked) problems is larger when machines and humans share their perceptions, associations and reasoning.

Second, the proposed human-agent partnerships involve explicit, univocal and possibly adjustable, Work Agreements (WA) on how the tasks should be allocated and performed [16]. Similar to the modeling of the objectives, an ontology design pattern has been developed for work agreements, defining the knowledge required for a team member to set, reason about and adjust work agreements. The WA-ontology defines core concepts with their relations encompassing the knowledge to specify, activate, monitor, and reason about work agreements: ***<creditor, debtor, antecedent, consequent, lifespan, acceptance>***. This ontology was implemented and tested for human-robot partnerships in disaster response teams (i.e., in the TRADR-project). Such a team needs to dynamically adapt the task allocation and coordination to the momentary context. A disaster response field test showed that the fire-brigade officers could specify work agreements as required, which, subsequently, brought forward the desired adaptive team behavior of the concerning robot.

Third, we developed explanation models for ePartners that should invoke adequate trust calibration (i.e., to mitigate over- or under-reliance) and improve the learning of the team-members. Currently, a generic perceptual-cognitive explanation (PeCo) framework is being developed for this purpose [15]. The perceptual level provides an Intuitive Confidence Measure [18] and identifies the "foil" that can be used in a contrastive explanation [19]. Such an explanation fits with human explanation dialogues, centering on the question "Why this output (the fact) instead of that output (the foil)?" It reduces the factors in the explanation to the ones that are of interest to the human, so that it can be better interpreted by the human. The cognitive level provides the beliefs, goals and emotions for explanations [15]. Initial evaluations showed that humans can understand the explanations that PeCo will provide, but further personalization and contextualization is needed.

4.0 DESIGN PATTERNS FOR HUMAN-E-PARTNER COLLABORATION

As mentioned in section 2, design patterns explicate the rationale of the chosen design solution as a common solution for a class of design problems. The following procedure aims at a common pattern library for human-agent collaboration:

1. *Identify key design problems for specific functionality*
2. *Search for available design patterns*
3. *If no pattern can be found,*
   and if it is a general, recurrent design problem:
   - *Start with a Proto Pattern, a pattern "in construction", i.e., a design problem and solution documented in a pattern form (yet lacking empirical grounding)*
4. *Provide different instantiations (examples)*
5. *Test, refine and validate these examples*
6. *If successful:*
   - *Make the Design Pattern accessible in library (of best practices)*

In our approach, the pattern language comprises a structure, template and ontology. Patterns can be constructed at different abstraction levels with different types of relationships, forming the building blocks of the research and development of team agents. We distinguish three pattern levels: Organization (e.g., work and rest schedules), team (e.g., constructive vs. destructive) and interaction (e.g., direct feedback).

Organizations often maintain norms for the joint activities of the workers. These norms can be explicated and formalized as policies: Explicit enforceable constraints on human and agents performance in a given situation, as authorization ("permissions") and obligation [23]. This way, policies govern the behavior of specific agent roles within the organization to enforce normative behaviors (i.e., they apply to a set of actors). The normative model allows to reason about (and to verify) obligations, permissions, and prohibitions, based on deontic logic [25]. Example policies for human-robot collaborations come from the space [23] and automotive domain [24]. In the last domain, the policies specified when an autonomous car should provide the driver with personalized supportive information, and when to hand over control to the driver. Both policies were linked to the driver's workload ([24], cf. the workload ontology of section 2). In these examples and in the TRADR project, organization design patterns have not yet been formulated. However, we have identified some "Proto Patterns" (see above) that need to refined and validated, such as the rest and work schedule pattern mentioned in the introduction of this paper, and design patterns for dealing with hazardous substances in populated areas. Following such patterns, the fire fighters and robots would operate within the safety and health norms of the organization. Van Diggelen et al. [13] provide a design pattern language for teamwork, which seems to be a suitable starting point to describe the normative joint activities at the level of the organization.

Team design patterns specify generic reusable behaviors of actors for supporting effective and resilient teamwork [13]. These patterns can be used to describe the dynamics of teamwork over longer periods of time. They can be constructive (i.e. leading to a more coherent team) or destructive (i.e. leading to a less coherent team), which can be observed in the behavior of an individual or group of actors. During a field test, for example, the TRADR team showed an individual, destructive micro-management pattern of the team-leader. By formalizing and implementing this pattern in the PeCO-framework, the agent can mitigate this behavior in future situations.

Interaction design patterns describe the shaping of the (multimodal) human-agent interactions, such as the dialogue acts for work agreements in TRADR [16]. Figure 6 presents part of the Interaction Design patterns for Human-Agent Collaboration (HAC) that have been specified so far. It distinguishes patterns for a Tailored Situation View (SV), Adjustable Work Agreements (WA) and Harmonized Interactive Notifications (IN) (see [13] for more details on these patterns).

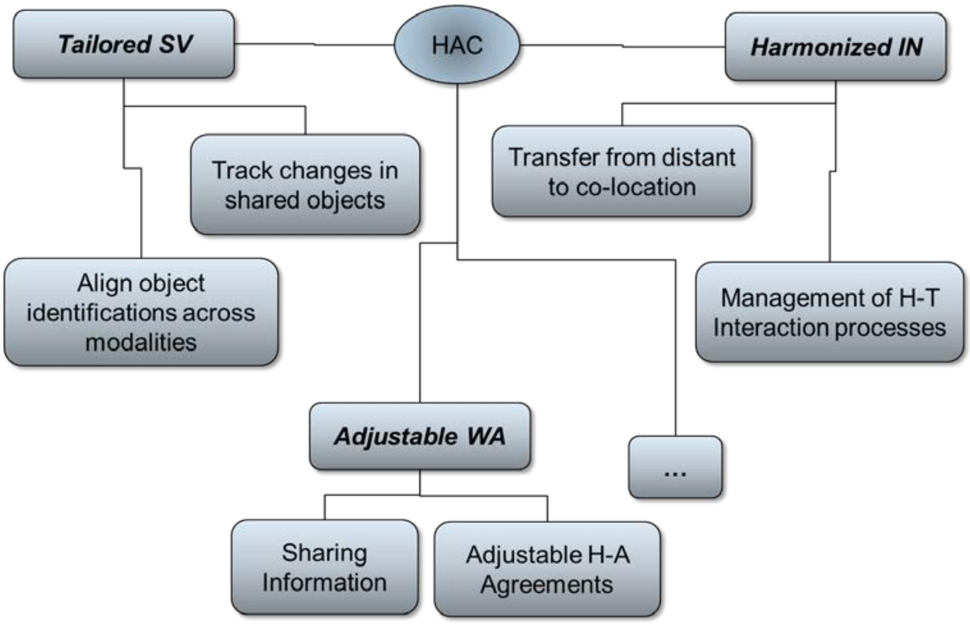

Figure 6: Part of the Interaction Design patterns for Human-Agent Collaboration (HAC), distinguishing patterns for a Tailored Situation View (SV), Adjustable Work Agreements (WA) and Harmonized Interactive Notifications (IN) (see [13] for more details on the patterns).

5.0 CONCLUSIONS

For the integration of robots into the operations of rescue, reconnaissance, security and defense teams, there is a need for theoretical and empirical grounded human-agent collaboration models. This paper proposes the development of organization, team and interaction design patterns with corresponding ontologies, as re-usable building blocks of human-agent collaboration. A situated Cognitive Engineering (sCE) methodology supports the incremental development and implementation of these building blocks (i.e., ePartner abilities and behaviors) into a perceptual-cognitive agent framework. Core cognition concepts are formalized as ontology design patterns (e.g., "workload" and "situation awareness") and related to the concerning domain ontology (e.g., disaster response with concepts like "point of interest" and "urgency"). In parallel, relevant design patterns are identified or constructed with the corresponding ontologies, at the organization (e.g., norms), team (e.g., work agreements, and interaction level (e.g. explanation). The ontologies are implemented into the prototype and tested on their expected outcomes (i.e. the claims).

A key proposal of this paper is to build on past and current agent systems via the specification and sharing of *design rationales*. This rationale includes the explication of testable claims ***and*** the derivation of design patterns. The design patterns are formulated at three abstraction levels as building blocks of the research and development of team agents or ePartners: Organization, team and interaction. A second key proposal is to use *ontologies* that formalize human factors concepts and relate them to the domain concepts. These ontologies underpin the knowledge base ("beliefs") and reasoning of the TRADR agents, for example for proposing, adjusting, accepting, fulfilling or rejecting a work agreement. By formalizing and implementing constructive and destructive team patterns in the knowledge base, the agent can detect and support constructive behaviors and mitigate destructive behaviors.

Following the proposed methodology, robots can evolve as partners in disaster response by instantiating design patterns for the (a) sharing objectives, beliefs and experiences, (b) committing to norms and work agreements, and (c) uptake and learning of explanations and feedback (Fig 5).

ACKNOWLEDGEMENTS

This work was partly funded by the EU FP7 TRADR project (grant no. 60963) and the Dutch Ministry of Defence’s research program (project V1719). We thank the fire fighters from the “Gezamenlijke Brandweer Rotterdam” in the Netherlands for participating in this study.

REFERENCES

[1] Neerincx, M. A. (2011). Situated cognitive engineering for crew support in space. *Personal and Ubiquitous Computing*, *15*(5), 445-456

[2] Bosse, T., Breebaart, L., Diggelen, J. V., Neerincx, M. A., Rosa, J., & Smets, N. J. (2017). Developing ePartners for human-robot teams in space based on ontologies and formal abstraction hierarchies. *International Journal of Agent-Oriented Software Engineering*, *5*(4), 366-398.

[3] Shively, R. J., Lachter, J., Brandt, S. L., Matessa, M., Battiste, V., & Johnson, W. W. (2017, July). Why Human-Autonomy Teaming?. In *International Conference on Applied Human Factors and Ergonomics* (pp. 3-11). Springer, Cham, Germany.

[4] Kruijff-Korbayová, I., Colas, F., Gianni, M., Pirri, F., de Greeff, J., Hindriks, K., Neerincx, M.A., Ögren, P., Svoboda, T. & Worst, R. (2015). TRADR project: Long-term human-robot teaming for robot assisted disaster response. *KI-Künstliche Intelligenz*, *29*(2), 193-201.

[5] Gangemi, A. and Presutti, V. (2009). Ontology design patterns. In *Handbook on ontologies*, 221-243. Springer, Berlin, Heidelberg, Germany.

[6] Smets, N. J., Neerincx, M. A., Jonker, C. M., & Båberg, F. (2017, March). Ontology-based situation awareness support for shared control. In *Proceedings of the Companion of the 2017 ACM/IEEE International Conference on Human-Robot Interaction* (pp. 289-290). ACM.

[7] Harbers, M., & Neerincx, M. A. (2017). Value sensitive design of a virtual assistant for workload harmonization in teams. *Cognition, Technology & Work*, *19*(2-3), 329-343.

[8] Colin, T. R., Mioch, T., Smets, N. J., & Neerincx, M. A. (2012). Estimating an operator's cognitive state in real time: A user modeling approach. In *RO-MAN* (pp. 627-633).

[9] Cohen, I., den Braber, N., Smets, N. J., van Diggelen, J., Brinkman, W. P., & Neerincx, M. A. (2016). Work content influences on cognitive task load, emotional state and performance during a simulated 520-days' Mars mission. *Computers in Human Behavior*, *55*, 642-652.

[10] Neerincx, M. A., Kennedie, S., Grootjen, M., & Grootjen, F. (2009). Modeling the cognitive task load and performance of naval operators. In *International Conference on Foundations of Augmented Cognition* (pp. 260-269). Springer, Berlin, Heidelberg.

[11] Looije, R., Neerincx, M. A., & Hindriks, K. V. (2017). Specifying and testing the design rationale of social robots for behavior change in children. *Cognitive Systems Research*, *43*, 250-265.

[12] Schulte, A., Donath, D., & Lange, D. S. (2016). Design patterns for human-cognitive agent teaming. In *International Conference on Engineering Psychology and Cognitive Ergonomics* (pp. 231-243). Springer, Cham.

[13] Neerincx, M. A., van Diggelen, J., & van Breda, L. (2016). Interaction design patterns for adaptive human-agent-robot teamwork in high-risk domains. In *International Conference on Engineering Psychology and Cognitive Ergonomics* (pp. 211-220). Springer, Cham, Germany.

[14] Mioch, T., Peeters, M.M.M. and Neerincx, M.A. (2018). Improving Adaptive Human-Robot Cooperation through Work Agreements. *RO-MAN2018 27th IEEE International Symposium on Robot and Human Interactive Communication*.

[15] Saad, E., Hindriks, K. V., & Neerincx, M. A. Ontology Design for Task Allocation and Management in Urban Search and Rescue Missions. *ICAART 2018*.

[16] Van Diggelen, Neerincx, M.A., Peeters, M.M.M., and Schraagen, J.M.S. (in press). Developing effective and resilient human-agent teamwork using team design patterns. *IEEE Intelligent Systems*.

[17] Mioch, T., Ledegang, W., Paulissen, R., Neerincx, M. A., & van Diggelen, J. (2014). Interaction design patterns for coherent and re-usable shape specifications of human-robot collaboration. In *Proceedings of the 2014 ACM SIGCHI symposium on Engineering interactive computing systems* (pp. 75-83). Association for Computing machinery (ACM).

[18] Neerincx, M.A. , Van der Waa, J. , Kaptein, F. and Van Diggelen, J. (2018). Using perceptual and cognitive explanations for enhanced human-agent team performance. *EPCE 2018, LNAI 10906*, pp. 1–11, Springer.

[19] Peeters, M.M.M. and de Visser, E.J. (2018). “Longitudinal trust development in human-agent-robot teams,” in *Workshop on Longitudinal Human-Robot Teaming, held at the Human-Robot Interaction Conference*, February 2018, Chicago, IL, USA.

[20] González-Ortega, J., Radovic, V., and Insua, D.R. (2018). Utility elicitation. In *Elicitation*, pp. 241-264. Springer, Cham

[21] Van der Waa, J., Diggelen, J. and Neerincx, M.A. (2018). The design and validation of an intuitive certainty measure. *Proceedings of IUI 2018 Workshop on Explainable Smart Systems*.

[22] Van der Waa, J., Robeer, M..J., Van Diggelen, J., Brinkhuis, M.J.S. and Neerincx, M.A. (2018). Contrastive Explanations with Local Foil Trees. *IJCAI 2018 Workshop on Human Interpretability in Machine Learning (WHI)*.

[23] Van Diggelen, J. M. Bradshaw, J.M., Johnson, M. Uszok, A. and Feltovich, P.J. (2010). Implementing collective obligations in human-agent teams using KAOS policies. In: *Coordination, Organizations, Institutions and Norms in Agent Systems V*. Springer, pp. 36–52.

[24] Mioch, T., Kroon, L., & Neerincx, M. A. (2017). Driver readiness model for regulating the transfer from automation to human control. In *Proceedings of the 22nd International Conference on Intelligent User Interfaces* (pp. 205-213). ACM.

[25] Wieringa, R.J. and Meyer, J.-J. C. (1993). Applications of deontic logic in computer science: A concise overview. *Deontic logic in computer science,* pp. 17–40, 1993.