Simulating Human-Robot Teamwork Dynamics for Evaluation of Work Strategies in Human-Robot Teams

Martijin Ijtsma

Sean Ye

Karen M. Feigh

Amy Pritchett

SIMULATING HUMAN-ROBOT TEAMWORK DYNAMICS FOR EVALUATION OF WORK STRATEGIES IN HUMAN-ROBOT TEAMS

Martijn IJtsma, Sean Ye, Karen M. Feigh
Georgia Institute of Technology
Atlanta, Georgia

Amy Pritchett
Pennsylvania State University
State College, Pennsylvania

To foster resilience in teams operating in complex work domains, design should allow for a range of work strategies as appropriate to context. This paper describes how computational simulation and network visualization of a team's work can identify feasible work strategies and assess their appropriateness for different contexts. Network visualizations can identify constraints and dependencies that drive the feasible set of work strategies. After preliminary network analysis, these dependencies and inter-dependencies can be simulated in detail to better understand their impact. To illustrate, we describe a case study that explores two different work strategies that can each address the dependencies in a human-robot (rover) team in a manned space exploration mission.

Studies of expert workers in complex work domains teach us that much of a system's resilience orginates from its workers' ability to adapt their work strategies, both to manage performance and workload levels (Woods and Hollnagel, 2006). To support such adaptation in design of teams, designers can build in flexibility to allow team members to "finish the design" (Vicente, 1999). Many current design methods for teams, however, inherently prescribe normative work strategies through implicit assumptions about how the work ought to be done. Additionally, existing attempts to create a more formative approach to the design of the team (e.g. Ashoori and Burns, 2013) have applied static work models that cannot account for the evolving dynamics of the work itself, and the coordination and synchronization it requires within a team. This paper introduces computational work analysis as a means to creating designs that can support multiple work strategies.

Background

A team is composed of agents, which can be human or technological agents (i.e. robots or other forms of automated systems). Conceptual design of teams specifies team composition, work allocation, and mechanisms for coordinating activities (IJtsma, Ma, Feigh, and Pritchett, 2019). Cognitive Work Analysis (CWA) (Vicente, 1999) is a design framework that formalized the idea of designing for expert workers to adapt through the support for different strategies. Contrary to normative design frameworks, which often prescribe an "optimal" work strategy that in practice limits adaptation, CWA lays out "formative" methods that help designers in supporting experts workers. Formative analysis of teams can provide insight in the constraints and dependencies that need to be managed by a team, regardless of the context.

Earlier work has applied CWA methods to study work in teams (Ashoori and Burns, 2013; Miller, McGuire, and Feigh, 2017). However, the formative methods proposed in CWA have been based on qualitative methods that, relative to the needs of designers not versed in CWA, can be rather vague and do not model the temporal dynamics of the team's joint work. The methods additionally are manual and therefore labour and time intensive (Bodin and Krupenia, 2016). While our earlier work demonstrated computational work models to evaluate patterns of work (IJtsma et al., 2019), a remaining challenge is identifying and supporting-in-design multiple different feasible work strategies to support team adaptation.

Identifying Work Strategies Through Network Visualization

This paper proposes an analysis of work through graph network visualization, aimed to inform designers by identifying constraints and dependencies that define a set of feasible work strategies. The analysis examines a model of the work to be conducted in the team to identify the constraints and dependencies in the work that drive which work strategies are feasible. Such identification of feasible work strategies can then be used as input to a computational model of work, to further examine the temporal components of these dependencies.

First, a team's work within their given work domain is described at various levels of abstraction. At the highest level, the team's work is described in terms of one or a number of goals, further elaborated as values and priorities one level down in the abstraction hierarchy. These values and priorities can be further abstracted into work functions, i.e. the actions that can be performed by the team members. The lowest level describes the tangible aspects of the work environment, i.e. physical and information resources.

Each level in such an abstraction hierarchy provides a complete description of the team's work domain; multiple heterarchic linkages can then be identified between the levels. To identify dependencies within the team's work arising from the work environment, three types of linkages need to be identified between actions and resources: (1) Actions can require as input specific information resources, formalized as *get* relationships; (2) actions can serve to change the environment, described as changing aspects of the environment state through *set* relationships where actions manipulate pieces of information as output; (3) actions can require physical resources (such as tools), described through *use* relationships.

These three types of linkages identified in the work model then constrain the paths that can be taken: *get* and *set* relationships imply that certain actions are sequential, in which one action's output is input to another; *use* relationships imply that two actions that are both linked to the same physical resource cannot be executed together. These dependencies can help with methods of identifying feasible strategies, similar to classic methods in which strategies can be shown on the abstraction hierarchy as feasible paths through the work domain (Vicente, 1999). With these specific formalizations, however, a more systematic approach can be undertaken using two types of network visualization: one that contains the information resources and one that contains the physical resources.

When a computational form of these networks is available, the set of feasible action sequences can be formally identified. For example, one can identify all actions that need to be executed to reach a final state, or alternatively, from an initial state, all possible next actions can be identified by forward propagating through the network. Moreover, network theory has several constructs that can be used to analyze and characterize the work model. For instance, the connectivity of the networks provides insight into how constrained the work is.

The way these constraints are coordinated results in a pattern of actions that we refer to as the *work dynamics* (IJtsma et al., 2019). Thus, once the network has been analyzed and feasible action sequences have been identified, the work model can be extended into a computational form that can be used to simulate the work dynamics. Simulation can provide more detailed insight into the temporal aspects of each work strategy, where the interplay of the various constraints and dependencies might result in emergent behavior in the team.

Case Study: Manned Space Exploration with a Rover

Our earlier research has focused on simulation of work dynamics in human-robot teams for space operations (IJtsma et al., 2019). The case study presented here will further demonstrate how network visualizations can provide formative insight to designers, through analyzing constraints in the work and identifying feasible work strategies. Here we model a simple team consisting of a rover and two astronaut drivers on the lunar surface. Following a brief literature survey on lunar EVA and Mars Rover operations (Hooey, Toy, Carvalho, Fong, and Gore, 2017; Miller et al., 2017), as well as several informal discussions with space robotics and operations researchers, a work model was created in the form of an abstraction hierarchy, shown in Table 1.

Table 1.
Abstraction hierarchy of the work in a rover/astronaut team.

Functional purpose	EVA objectives			
Abstract functions	Resource consumption	EVA priorities	Rover safety	
Generalized functions	Life support system monitoring	Local navigation	Translation, control, and orientation	Imagery
Physical functions	1. Check battery levels	4. Plan path for rover	7. Select next waypoint	9. Change camera angle
	2. Check temperature	5. Estimate size of object	8. Move rover	10. Capture imagery
	3. Assess location and attitude	6. Localize obstacles		
Physical resources	a. Batteries	b. Rover vehicle	c. Rocks/obstacles	d. Camera
Information resources	A-B. (Observed) battery levels	Q. Goal location	T. Terrain map	W. Camera angles
	C-D. (Observed) subsystem temp	R. Planned path	U. Rock locations	X. Imagery
	E-P. (Observed) rover states	S. Next waypoint	V. Rock size	Y. Goal reached

Linkages between elements in this model identify constraints on the work. Figure 1 shows these linkages in a graph network representation. The nodes are actions (squares) and information resources (circles). The edges represent *set* and *get* linkages. The directional graph clearly shows the centrality of the rover dynamics (`Move Rover`), as well as the path planning

(`Plan Rover Path`). Several feedback loops are apparent in this representation, including the feedback loop for selecting the next waypoint and moving the rover iteratively, as well as a longer-timescale loop for replanning the path on a larger scale. A designer can further refine this model of the work iteratively by identifying new information requirements for actions and thereby creating new edges.

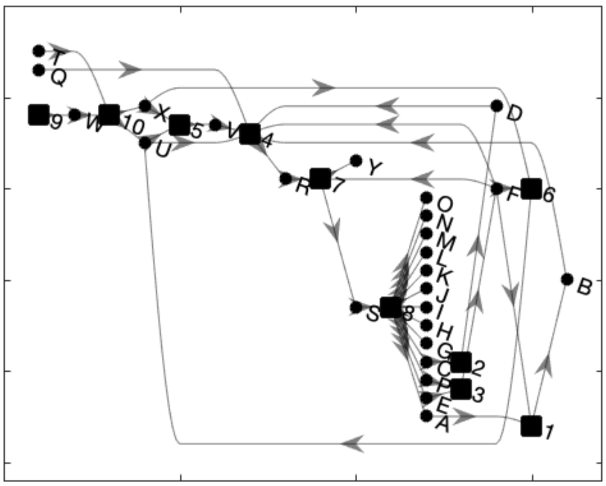

Figure 1. Network visualization of precedence relationships.

Once the network is fully flashed out, it can be used to identify feasible work strategies between two or more points. As an example, Figure 2 shows two feasible work strategies between setting the camera angle (`Change Camera Angle`) and moving the rover (`Move Rover`). In Strategy 1 the rover imagery is used straight away without formal analysis of obstacles, and is appropriate when high quality information on the "rock locations" is already available and does not require constant sampling. Strategy 2 steps through localization of obstacles and estimation of their size before doing path planning. This strategy would be appropriate when the quality of information for "rock locations" is low, and therefore the imagery should be frequently sampled and used to inform path planning. In many potential missions, the information quality can vary with location and terrain, and thus the team may need to adapt between these strategies.

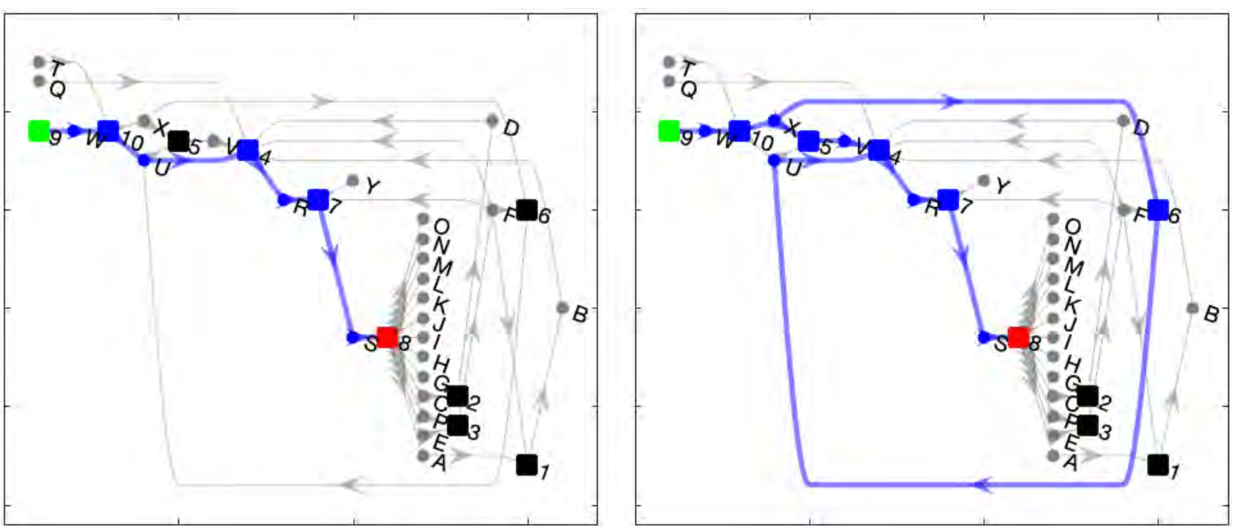

Figure 2. Two work strategies for coordinating "Change Camera Angle" and "Move Rover".

Once feasible action sequences are identified, more detailed evaluation can be performed through simulation of the work. A computational version of the work model was created to evaluate the timing of actions within each work strategy. From the graph networks it is clear that the rover dynamics play a significant role in the work dynamics. Thus, the computational model of the action `Move Rover` contains a dynamic model of the rover (updated at 2Hz), and models of how a human or automatic system would control speed and heading to track to a given waypoint. A lunar terrain model identified whether any location was open or has obstacles that the rover needed to maneuver around.. The `Localize Obstacles` action checks the surrounding area and updates the known obstacles in the environment. The action `Plan Rover Path` produces a path from the rover's current location to desired location. The `Select Next Waypoint` action provides the rover with the next desired waypoint to traverse to.

The two feasible work strategies in Figure 2 were simulated using this computational work model. In this case, all actions were performed by a single agent to illustrate the dynamics inherent to the work even before accounting for inter-agent coordination; the simulation can also examine a range of potential work allocations. For example, an astronaut might plan the paths based on an assumed database of terrain information and the rover then move through them autonomously (correspondingly closely with Strategy 1), or the human may actively steer the camera to confirm the feasibility of the planned path and interact more closely with the rover in path following (as may be necessary with Strategy 2).

Figure 3 shows the path that the rover traversed in each strategy. Clearly, having prior knowledge of the obstacles resulted in a more optimized path. With lower information quality in the right figure, the rover first attempts to drive straight to the goal position, only later finding out (through `Localize Obstacles` and `Estimate Size Object`) that this route is not a viable option, and then needing several iterations to find a path around obstacles.

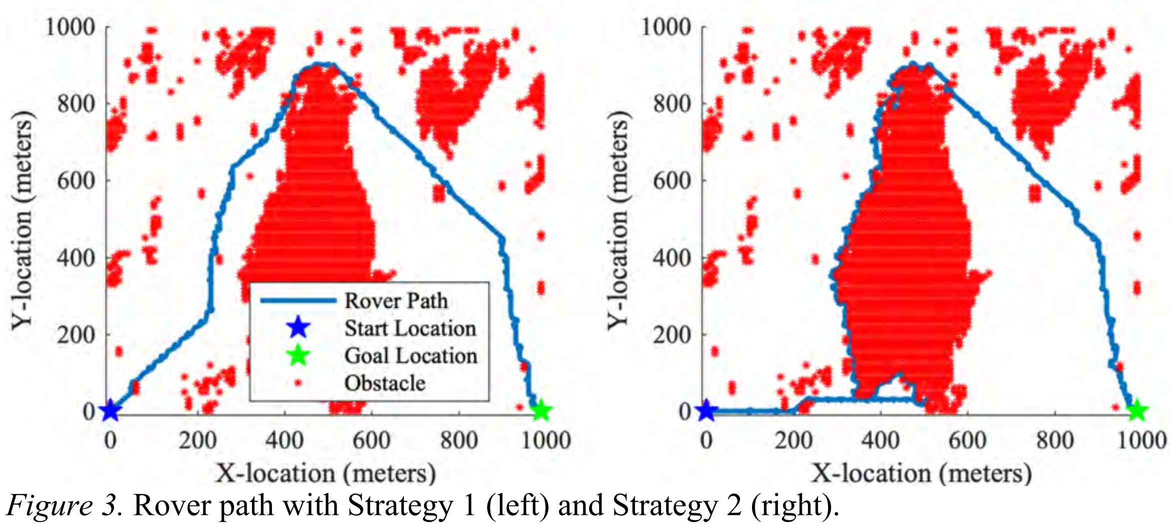

Figure 3. Rover path with Strategy 1 (left) and Strategy 2 (right).

Conclusions

This paper introduced a formative and computational work modeling approach that can support designers of teams for complex work domains by identifying and designing for multiple work strategies. Many further iterations are possible with this method. The network analysis and simulation described in this paper comprised a first step in examining primarily the task work that the team must collectively perform. Once the specific strategies (or attributes of a wide range of feasible strategies) are thus identified within the taskwork, subsequent design decisions can then examine the team composition, and the allocation of work within the team, by which this taskwork will be conducted.

Further, the method of coordination will impact the feasibility of different work strategies and, if they remain feasible, their timing and performance. In human-human coordination within teams, the humans are typically reasonably flexible and adaptive in their modes of interaction. On the other hand, in the case of human-robot or human-automation interaction, the machine is comparatively inflexible in the modes of interaction, requiring predictable patterns in the human's activity for commanding, controlling, monitoring and/or confirming the machine. These dynamics can be included in the network analysis as additional teamwork actions that require particular sequences of actions (e.g. the robot cannot perform an action until its human supervisor is free to command and monitor it – and the human may not be able to perform her/his own physical tasks in parallel). Further, these teamwork actions add a further temporal component to the collective dynamics of the team's work, which computational simulation can further predict.

Acknowledgements

This work is sponsored by the NASA Human Research Program with Jessica Marquez serving as Technical Monitor under grant number NNJ15ZSA001N. The authors would also like to thank the other WMC developers.

References

Ashoori, M., & Burns, C. (2013). Team cognitive work analysis: Structure and control tasks. *Journal of Cognitive Engineering and Decision Making*, *7*, 123–140.

Bodin, I., & Krupenia, S. S. (2016). Activity Prioritization to Focus the Control Task Analysis. *Journal of Cognitive Engineering and Decision Making*, *10*, 91–104.

Hooey, B. L., Toy, J. J. N. T., Carvalho, R. E., Fong, T., & Gore, B. F. (2017). Modeling Operator Workload for the Resource Prospector Lunar Rover Mission. In D. Cassenti (Ed.), *Advances in Human Factors in Simulation and Modeling. AHFE 2017. Advances in Intelligent Systems and Computing* (Advances i, Vol. 591, pp. 183–194). Springer, Cham.

IJtsma, M., Ma, L. M., Feigh, K. M., & Pritchett, A. R. (2019). Analysis of Work Dynamics for Objective Function Allocation in Manned Spaceflight Operations. In M. A. Vidulich & P. S. Tsang (Eds.), *Advances in Aviation Psychology* (Vol. 3). Ashgate Publishing Limited.

Miller, M. J., McGuire, K. M., & Feigh, K. M. (2017). Decision Support System Requirements Definition for Human Extravehicular Activity Based on Cognitive Work Analysis. *Journal of Cognitive Engineering and Decision Making*, *11*, 136–165.

Vicente, K. (1999). Cognitive Work Analysis: Modeling Behavior-Shaping Constraints. In *Cognitive Work Analysis* (pp. 113–136).

Woods, D. D., & Hollnagel, E. (2006). *Joint Cognitive Systems: Patterns in Cognitive Systems Engineering* (1st ed.). Boca Raton, FL: CRC Press.