The 2016 AAAI Fall Symposium Series:
Cross-Disciplinary Challenges for Autonomous Systems
Technical Report FS-16-03

# Trust-Based Symbolic Robot Motion Planning with Human-in-the-Loop

**Yue Wang**
Department of Mechanical Engineering
Clemson University
Clemson, SC 29634

### Abstract

Autonomous robots are becoming increasingly popular and such systems has led to complex design and analysis which brings the necessity of validation and verification. In particular, symbolic robot motion planning based on formal methods is verifiably correct. It is the process of specifying and planning robot tasks in a discrete space, then carrying them out in a continuous space in a manner that preserves the discrete-level task specifications. Despite progress in symbolic motion planning, many challenges remain, including addressing scalability for multi-robot systems and improving solutions by incorporating human intelligence in an adaptive fashion. On the other hand, extant works in human-robot interaction (HRI) often lack quantitative models and real-time analytical approaches. Here, we summarize our recent works on symbolic robot motion planning with human-in-the-loop as a step toward addressing these challenges. We specially focus on human trust in autonomous robots and embed trust analysis into the symbolic robot motion planning.

## Trust

Trust is a key parameter in determining human's acceptance and hence use of a robot (Hancock et al. 2011). Consideration of trust is especially important for supervisory control of multiple robots, since the tasks must be carefully allocated to ensure that time-critical issues are addressed while human workload is kept within acceptable bounds.

In both (Spencer, Wang, and Humphrey 2016) and (Mahani and Wang 2016), a quantitative and dynamic trust model based on robot performance/faults, human performance, and the environment is used to estimate human trust in each of the robots throughout the scenario. This type of time-series models characterize the dynamic relationship between human trust in the robots and the independent variables. The model is hence suitable for both real-time analysis and prediction of trust for control allocation.

In (Spencer, Wang, and Humphrey 2016), robot performance is modeled as a function of "rewards" the robot receives by detecting obstacles and reaching goals. This allows the robot to earn trust as it learns details of the environment. Human performance is calculated based on workload and the complexity of the environment surrounding the robot with which the human is currently collaborating. The concept of utilization ratio is used to measure workload. Complexity of the environment is based on the number of obstacles that lie within sensing range of the collaborating robot. The human's superior capability in creating more detailed paths will be enhanced in more complex environments, leading to increased performance in the presence of more obstacles. On the other hand, human performance decreases with respect to workload. Robot faults are modeled as the total number of obstacle regions the robot has entered before sensing the corresponding obstacle. In (Mahani and Wang 2016), the robot performance is a function of the distance of the robot to the closest obstacle and goal at each time step. In this model the robot performance decreases as the robot gets closer to an obstacle and further from a goal. The human performance model also takes into account human workload and environmental complexity.



## Multi-Robot Symbolic Motion Planning

In (Spencer, Wang, and Humphrey 2016), we utilize local communication, observation, control protocols, and compositional reasoning approaches to decompose the planning problem to address scalability. To address solution quality and adaptability, we use the dynamic and computational trust model to aid this decomposition and to implement real-time switching between automated and human motion planning.

We consider an intelligence, surveillance, and reconnaissance (ISR) scenario in which a team of robots, supervised by a human operator, must reach a set of goal destinations while avoiding collisions with stationary obstacles and with each other. The workspace is discretized into polytopic regions that are labeled with relevant properties, e.g., whether they contain an obstacle or goal. We assume a set of goal destinations is known from the start, and each goal must be reached by at least one robot while collisions with obstacles and between robots are avoided. This set of requirements forms a specification for the scenario. Our proposed planning scheme is implemented in a distributed manner, making use of compositional reasoning approaches to decompose the global specification. More specifically, the goal portion of the specification is decomposed such that each robot is assigned a subset of the goal destinations and locally synthesizes a plan to reach them. For obstacle avoidance, we assume obstacle locations are not known a priori, and so when a robot discovers an obstacle, it re-synthesizes a plan to reach its remaining goals after updating its representation of the workspace.

First we consider robot collision avoidance tasks that require collaborations between neighboring robots. This requires defining the atomic propositions for each robot, which correspond to communication, observation, and control. The communication proposition is true if another robot is within the communication range of the robot under consideration and false otherwise. When the communication atomic proposition is true, two robots can communicate with each other to exchange sensing and path information. This information can also be used to detect possible collisions between the two robots, expressed in the observation proposition. If it is observed that the current robot's motion plan will cause an imminent collision with the second robot, then the observation proposition is true; otherwise it is false. For the control proposition, when it is true, the robot is executing a nominal linear quadratic regulator (LQR) control law; when false, the robot pauses, replans its path automatically or requests for human intervention. When both the communication and the observation propositions are true, the robot has detected a potential collision and communicates its path with the involved robot. At this moment, the control proposition is set to false, and the robot pauses, replans, or asks for human intervention dependent upon the collision type. At each time step, the propositions are checked, and local specifications are dynamically updated. Through these propositions, we are able to decompose the robot collision avoidance task and guarantee there is no collision between the robots. Next, we use a general compositional reasoning approach to show that the robots are able to collectively fulfill the reachability and obstacle avoidance portions of the global specification using a distributed planning approach. Compositional reasoning in this context relies on concepts of interleaving of transition systems and unconditional fairness. Last but not least, the estimate of trust affects the specification decomposition, with more trusted robots assigned more destinations. A real-time trust-based switching between human and automated motion planning is also proposed and designed and implemented in (Mahani and Wang 2016) as explained in the next section. We show that our planning approach is guaranteed to meet the task specifications under some mild assumptions.

## Runtime Verification Framework for Switches between Human and Robot Motion Planners

In (Mahani and Wang 2016), we investigate runtime verification approaches for robot motion planning with human-in-the-loop. By bringing together approaches from runtime verification, trust model, and symbolic motion planning, we develop a framework which guarantees that a robot is able to safely satisfy task specifications while improving task efficiency by switches between human supervision and autonomous motion planning. A simple robot model in a domain path planning scenario is considered and the robot is assumed to have perfect localization capabilities. The task domain is partitioned into a finite number of identical cells. A computational trust model based on the robot and human performance is used to provide a switching logic between different modes.

The switching framework is consisted of two controllers: autonomous baseline controller for safety and manual advanced controller for efficiency. This framework is logically divided into five subsystems: Motion Planner, Controller, Monitor, Checker, and Decision Maker. The advanced subsystem is less safe but contains human-in-the-loop which refers to the manual mode. The baseline subsystem refers to the autonomous mode and uses a symbolic motion planner, which is guaranteed to be correct and hence safe. In the autonomous mode, the system generates plans using NuSMV. In the Monitor subsystem, we have two modules: Filter and Event Recognizer. The filter is designed to extract the information and send them to the event recognizer. Although the filter can be merged with the event recognizer, but having them separated prevents the system from the overhead of abstracting out events from the extracted information and consequently we can minimize intervention with the monitored system run. The next module in the monitoring subsystem is the event recognizer which detects an event from the values received from the filter based on event definitions provided by a monitoring script. The monitoring script renders the systems states to the events at the requirement level in order to be analyzed by the system checker. Once the event recognizer detects an event, it will send that information to the checker module. The runtime checker uses the specifications provided by the user and checks whether or not the current execution of the system meets the requirements. Based on the information received from the runtime checker, the decision module determines under which mode the system should run for motion planning and it uses the trust model to evaluate the trust level of the system. The simulation is conducted in ROS and model checking is computed using NuSMV. Runtime verification that mainly contains the monitor, checker, and decision module is implemented using ROSRV.

## References

Hancock, P. A.; Billings, D. R.; Schaefer, K. E.; Chen, J. Y.; De Visser, E. J.; and Parasuraman, R. 2011. A meta-analysis of factors affecting trust in human-robot interaction. *Human Factors: The Journal of the Human Factors and Ergonomics Society* 53(5).

Mahani, M., and Wang, Y. 2016. Runtime verification of trust-based symbolic robot motion planning with human-in-the-loop. *Proceedings of the ASME 2016 Dynamic Systems and Control Conference*.

Spencer, D.; Wang, Y.; and Humphrey, L. 2016. Trust-based human-robot interaction for safe and scalable multi-robot symbolic motion planning. *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*.