TRUST-BASED SYMBOLIC ROBOT MOTION PLANNING

Maziar Fooladi Mahani and Yue Wang

***Abstract*—In this study, we address symbolic robot motion planning with human-in-the-loop. Specifically, we explore (1) real-time trust-based switching between human and robot motion planning to address the tradeoffs between safety and performance, and (2) trust-based specification decomposition to address scalability. By bringing together approaches from trust model, symbolic motion planning, and runtime verification, 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. Specification decomposition based on assume-guarantee (A-G) contracts is developed to provide scalability and adaptability for high-level multi-robot tasking. We demonstrate the effectiveness of this framework as well as its feasibility through simulations using NuSMV and ROSRV.**

I. INTRODUCTION

Autonomy has made great strides over the history of robotics, dramatically decreasing physical and cognitive workload of operators and increasing task performance. Despite these advances, automation has yet to surpass the adaptability and high-level cognitive reasoning of a human operator. A human operator can adapt and devise plans that are computationally expensive for autonomy to develop alone. However, as the human operator becomes fatigued, he or she is prone to error. It is therefore desirable to devise novel methods of effective human-robot interaction (HRI) that take into account the strengths of both autonomy and human operation by detecting scenarios difficult for autonomy and weighting that difficulty against an operator's abilities. However, considering safety and efficiency of a system at the same time is somehow controversial. Runtime verification techniques by utilizing approaches that checks whether or not a system under scrutiny can satisfy a set of given specifications are undoubtedly necessary for systems with high level of complexities. Furthermore, the extension to multi-robot systems is challenging due to the "state-space explosion" problem since both the abstraction and the synthesis algorithms scale exponentially with the dimension of the configuration space. In the presented work, we develop a new trust-based framework to guarantee robot's safety while improving efficiency for a multi-robot motion planning task with human-in-the-loop.

II. METHODS

Our approach to the first problem is based on a trust-based runtime verification framework. This framework is logically divided into five subsystems: motion planner, controller, monitor, checker, and decision maker (Fig. 1). This system is designed in order to be able to control the robot in two modes: manual versus autonomous. The advanced sub-system 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 - NuSMV, a model checking tool. The baseline path planner generates a safe but lengthy path to goals while the human can signal the advanced path planner to use riskier but shorter paths (Fig. 2). 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. The event recognizer detects an event from the values received from the filter based on event definitions provided by a monitoring script. 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 a trust model, which is described below, to evaluate the trust level of the system. If the trust value falls below a predefined threshold, the decision module switches the system to the autonomous mode. The decision module will switch back to the manual mode only if the trust value is higher than that threshold again.

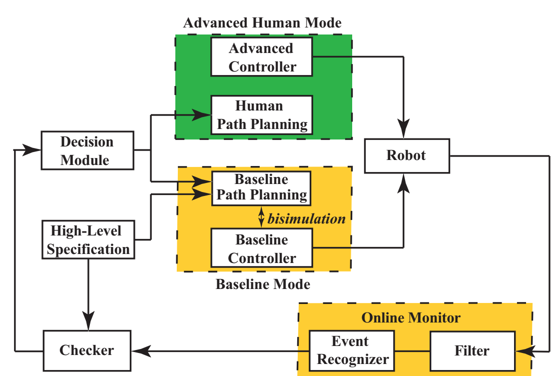

Fig. 1. The overall structure of the runtime verification framework

There are a variety of factors that may influence the trust level. Environmental characteristics, robot and human's performance are the major parameters in determining the trust

Department of Mechanical Engineering, Clemson University, Clemson, SC 29634, Email: {mfoolad,yue6}@g.clemson.edu

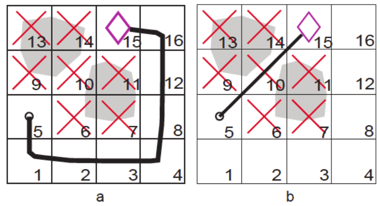

Fig. 2. (a) Safe robot motion planning in low trust scenario, and (b) advanced human motion planning in high trust scenario.

value [1]:

$$\begin{aligned} T(k) \quad &= \quad AT(k-1) + B_1 P_R(k) - B_2 P_R(k-1) \\ &\quad\quad + C_1 P_H(k) - C_2 P_H(k-1) \end{aligned} \tag{1}$$

where $T(k)$ represents human trust in the robot, P_R represents robot performance, and P_H represents human performance. The coefficients are subject to determination by human subjective tests in a given task.

To address scalability, we employed the Assume-Guarantee (A-G) contracts to decompose task specifications for multi-robots [2]. Symbolically, the A-G reasoning states that

$$\frac{\begin{array}{c}\langle\varphi_2\rangle R_1 \langle\varphi_1\rangle \\ \langle\varphi_1\rangle R_2 \langle\varphi_2\rangle\end{array}}{\langle true\rangle R_1 \parallel R_2 \langle\varphi_1 \wedge \varphi_2\rangle} \tag{2}$$

where $R_1 \parallel R_2$ denotes the composition of Robot 1 and 2's transition systems, and the specification satisfies $\varphi_1 \wedge \varphi_2 \Rightarrow \varphi$. Formulas of the form $\langle\varphi_2\rangle R_1 \langle\varphi_1\rangle$ assert that R_1 guarantees φ_1 on the assumption that R_2 satisfies φ_2 and vice versa for R_2, so that $R_1 \parallel R_2$ guarantees φ unconditionally. This allows us to deduce properties about $R1 \parallel R2$ while reasoning about R_1 and R_2 separately, reducing the state space and making paths easier to compute and verify. The A-G reasoning approach is extended to the multi-robot motion planning.

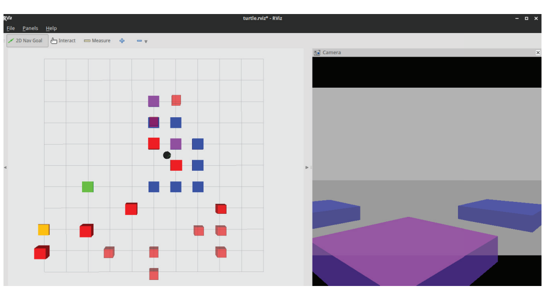

Fig. 3. Snapshot of simulations. Red blocks represent obstacles, blue blocks represent the path generated by model checker, yellow block is the start point, purple blocks represent unreached goals, green blocks represent reached goals and the black circle is the top view of the robot.

III. RESULTS

We first consider a mobile ground robot deployed in an unknown environment containing a set of obstacles to be avoided and a set of goals to be reached. The environment is partitioned with identical square cells. Fig. 3 shows a snapshot of the simulation using NuSMV and ROSRV and the right portion of this figure shows how the human operator sees the robot's environment from the robot onboard camera.

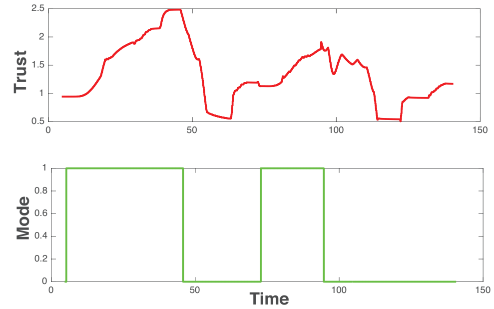

Fig. 4. Trust evolution and selected modes.

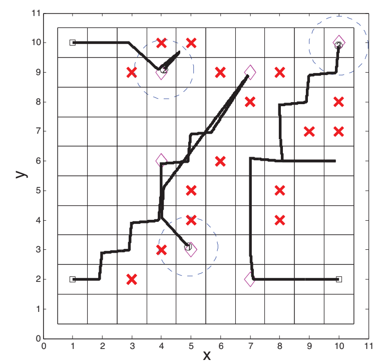

Fig. 5. Final paths of a 3-robot team operated by a human.

Fig. 4 presents the trust evolution and selected modes. Mode 1 refers to autonomous path planning and mode 2 refers to manual path planning. We then simulate the multi-robot case (see Fig. 5). The overall goal of the multi-robot system is to eventually reach each of the six goals. Scattered throughout the environment there are 16 obstacles, marked by x, which are initially hidden from the robots until they are sensed. The sensor range of a robot is marked by the dashed circle surrounding it. Once an obstacle is sensed, its position becomes known to that individual robot for the purpose of its path planning. Fig. 5 shows the final paths of the 3 robots planned under the A-G reasoning method.

REFERENCES

[1] Peter A Hancock, Deborah R Billings, Kristin E Schaefer, Jessie YC Chen, Ewart J De Visser, and Raja Parasuraman. A meta-analysis of factors affecting trust in human-robot interaction. *Human Factors: The Journal of the Human Factors and Ergonomics Society*, 53(5):517–527, 2011.

[2] Alireza Partovi and Hai Lin. Assume-guarantee cooperative satisfaction of multi-agent systems. In *American Control Conference (ACC), 2014*, pages 2053–2058. IEEE, 2014.