Reactive Switching Protocols for Multi-Robot High-Level Tasks

Vasumathi Raman
California Institute of Technology (vasu@caltech.edu)

Motivation

Automatically generate provably correct control from high-level specifications for teams of interchangeable robots

Example:

Robot 1 starts in r1, robot 2 starts r6. If the door is closed, the robots cannot move through it. Regions r2 and r5 are "intrusion-sensitive" -- when an intruder is detected in them, one of the robots must go to that region to investigate.

Overview

- model robot team as a switched system
  \[ \dot{x}(t) = f_{\sigma(t)}(x(t)), \]
  \[ \text{mode} = \text{task assignment} \]
- construct motion controllers for each mode
- synthesize switching protocol to realize \( \varphi \)

Contributions

Novelty: concurrent task reassignment and planning via reactive synthesis

Computation: switched system representation yields exponential improvement during synthesis

Virtualization: explicit separation between motion controllers and robots allows solution of otherwise infeasible tasks
Reactive Switching Protocols for Multi-Robot High-Level Tasks
Vasumathi Raman

Automatically generating provably correct control from high-level specifications for teams of interchangeable robots

Reactive Synthesis

- Formulas in Linear Temporal Logic (LTL)*
  » Propositional logic +
  $$\mathcal{U}$$ (until)  $$\bigcirc$$ (next)  $$\square$$ (always)  $$\diamondsuit$$ (eventually)
- Propositions partitioned into controlled ($$\mathcal{Y}$$) and uncontrolled ($$\mathcal{X}$$) sets
- If the operating environment obeys $$\varphi_e$$, the system satisfies $$\varphi_s$$.
- Generalized Reactivity (1) (GR(1))
  $$\varphi_e^i \land \varphi_e^t \land \varphi_e^g \Rightarrow \varphi_s^i \land \varphi_s^t \land \varphi_s^g$$

Motion Control

- Need a controller for driving the team of robots from any current configuration to the goal configuration for each permutation of the goals
- Can use approach in [2] -- decompose the configuration space into obstacle-free polytopes, generate local smooth feedback laws that drive the team of robots from one cell to an adjoining one, sequence these local controllers using A* or incremental D* to reach the goal.


Related Work

Temporal Logic Synthesis for Multi-Robot Systems
Kloetzer and Belta, T-Ro 2007, 2010
Chen & Belta, T-Ro 2010
Loizou and Kyriakopoulos, CDC 2004
Raman and Kress-Gazit, ICRA 2014

Concurrent task assignment and planning
Turpin and Kumar, ICRA 2013
Ayanian, Rus and Kumar, NecSys 2013

Synthesis of switching protocols
Liu, Ozay, Topcu and Murray, TAC 2013
Example:

Robot 1 starts in r1, robot 2 starts r6.
If the door is closed, the robots cannot move through it.
Regions r2 and r5 are “intrusion-sensitive” -- when an intruder is detected in them, one of the robots must go to that region to investigate.

Discrete Abstraction

\[ G = \{r_2, r_5\} \quad \text{goals} \]
\[ P_N = \{12, 21\} \quad \text{goal permutations} \]
\[ X = \{\pi_{\text{intruder}_2}, \pi_{\text{intruder}_5}, \pi_{\text{door}}\} \]
\[ Y = \{\pi_p | p \in \{12, 21\}\} \]

completion of motion of robot i to room r

Synthesized Automaton (Excerpt)

The system switches modes from p12 to p21 when the door is closed and an intruder is sensed in r5.

Specification (Excerpt)

\[ (\varphi_{r_1} \land \varphi_{r_5}) \quad \text{#Initial (Environment)} \]
(Robot 1 starts in r1, Robot 2 in r6)
\[ (\neg \pi_{\text{intruder}_2} \land \neg \pi_{\text{intruder}_5} \land \neg \pi_{\text{door}}) \quad \text{#Initial (Environment)} \]
(Initially no intruders, open door)
\[ (\pi_{12}) \quad \text{#Initial (System)} \]
(Robot 1 is initially assigned goal r2,
Robot 2 is initially assigned goal r5)
\[ \land \Box (\varphi_{r_1} \land \Diamond \pi_{\text{door}} \Rightarrow \Diamond \neg \varphi_{r_2}) \quad \text{#Safety (Environment)} \]
(If door closed, Robot 1 can’t move from r1 to r2)
\[ \land \Box (\varphi_{r_2} \land \Diamond \pi_{\text{door}} \Rightarrow \Diamond \neg \varphi_{r_1}) \quad \text{#Safety (Environment)} \]
(If door closed, Robot 1 can’t move from r2 to r1)
\[ \ldots \]
\[ \land \Box (\varphi_{r_5} \land \Diamond \pi_{\text{door}} \Rightarrow \Diamond \neg \varphi_{r_6}) \quad \text{#Safety (Environment)} \]
(If door closed, Robot 2 can’t move from r5 to r6)
\[ \land \Box (\varphi_{r_6} \land \Diamond \pi_{\text{door}} \Rightarrow \Diamond \neg \varphi_{r_5}) \quad \text{#Safety (Environment)} \]
(If door closed, Robot 2 can’t move from r6 to r5)
\[ 1 \land \Box \Diamond (\pi_{\text{intruder}_2} \Rightarrow \Diamond (\varphi_{r_2} \lor \varphi_{r_5})) \quad \text{#Liveness (System)} \]
(If an intruder is detected in r2,
either Robot 1 or 2 should go to r2)
\[ 2 \land \Box \Diamond (\pi_{\text{intruder}_5} \Rightarrow \Diamond (\varphi_{r_5} \lor \varphi_{r_2})) \quad \text{#Liveness (System)} \]
(If an intruder is detected in r5,
either Robot 1 or 2 should go to r5)
Reactive Switching Protocols for Multi-Robot High-Level Tasks

Vasumathi Raman

 Automatically generating provably correct control from high-level specifications for teams of interchangeable robots