Compositional Synthesis of Multi-Robot Motion Plans via SMT Solving

Indranil Saha
(UC Berkeley & UPenn)

Background

Goal

To synthesize motion plans automatically for a group of robots with complex dynamics for complex specification

Existing Solutions

- Generate a finite abstraction for the robot dynamics
- Generate a finite model for the property
- Apply a game theoretic algorithm to generate a high level plan
- Generate low level control signals that satisfy the bisimulation property

Computationally expensive.. Not suitable for multi-robot systems

Approach

- We assume availability of a set of precomputed control laws for each robot
- We use an off-the-shelf SMT solver to generate motion plans composing these motion primitives

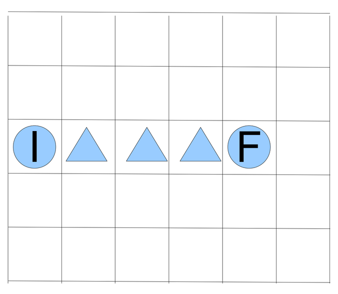
Motion Primitive

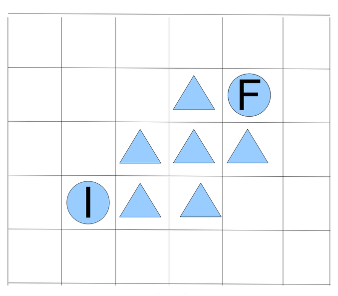
Captures closed-loop behavior of a robot under the action of a controller

A motion primitive is formally defined as a 7-tuple: $\langle u, \tau, q_i, q_f, X_{rf}, W, cost \rangle$

- u - a precomputed control input
- τ - the duration for which the control signal is applied
- q_i (q_f) - initial (final) velocity configuration
- X_{rf} - relative final position
- W - the set of relative blocks through which the robot passes
- $cost$ - an estimated energy consumption for executing the control law

$PRIM_i$ - the set of all primitives for robot i





Problem and Solution

Problem Instance

An input problem instance $\mathcal{P} = \langle N, I, F, PRIM, OBS, \xi \rangle$

- N - Number of robots
- I (F) - Initial (Final) state of the group of robots
- $PRIM = [PRIM_1, PRIM_2, \ldots, PRIM_N]$
- OBS - The set of obstacles
- ξ - $\Box\Psi$, conjunction of a set of invariant properties

Problem Definition

Motion Plan. A *motion plan* of a multi-robot system for an input problem instance $\mathcal{P} = \langle N, I, F, PRIM, OBSTACLES, \Box\Psi \rangle$ is defined as a sequence of states $\Phi = (\Phi(0), \Phi(1), \ldots, \Phi(L))$ such that

$$\Phi(0) \in I \qquad \Phi(L) \in F \qquad \Phi(0) \models \Psi$$

and the states are related by the transitions in the following way:

$$\Phi(0) \xrightarrow{Prim_1} \Phi(1) \xrightarrow{Prim_2} \Phi(2) \ldots \Phi(L-1) \xrightarrow{Prim_L} \Phi(L)$$

Motion Planning Problem. Given an input problem $\mathcal{P}$ and a positive integer L, synthesize a motion plan of length $L+1$

Transition Constraints

$\Phi_1 = [\phi_{11}, \ldots, \phi_{1N}]$, $\Phi_2 = [\phi_{21}, \ldots, \phi_{2N}]$

$Prim = [prim_1, \ldots, prim_N]$, where $prim_i \in PRIM_i$.

A transition

$$\Phi_1 \xrightarrow{Prim} \Phi_2$$

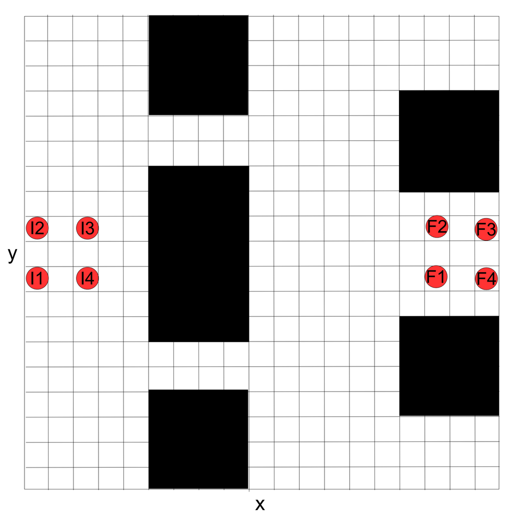
is associated with the following constraints:

- $\forall i \in \{1, .., N\}$:
  $\phi_{1i}.q = prim_i.q_i, \;\; \phi_{2i}.q = prim_i.q_f, \;\; \phi_{2i}.X = \phi_{1i}.X + prim_i.X_{rf}$
- *obstacle_avoidance*$(\Phi_1, \Phi_2, Prim, OBS)$
- *collision_avoidance*$(\Phi_1, \Phi_2, Prim)$
- $(\Phi_1 \models \Psi) \rightarrow (\Phi_2 \models \Psi)$

Constraints are solved using an SMT Solver

Examples

Specification 1

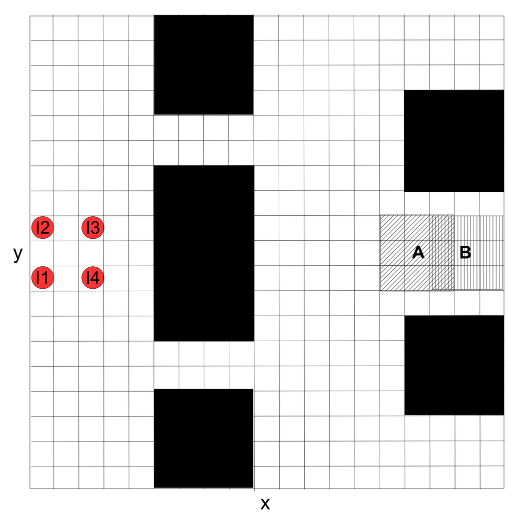


Goal: I1 → F1, I2 → F2, I3 → F3, I4 → F4

Invariants:

- Maintain a rectangular formation
- Maintain a precedence relationship
- Maintain a minimum distance

Specification 2

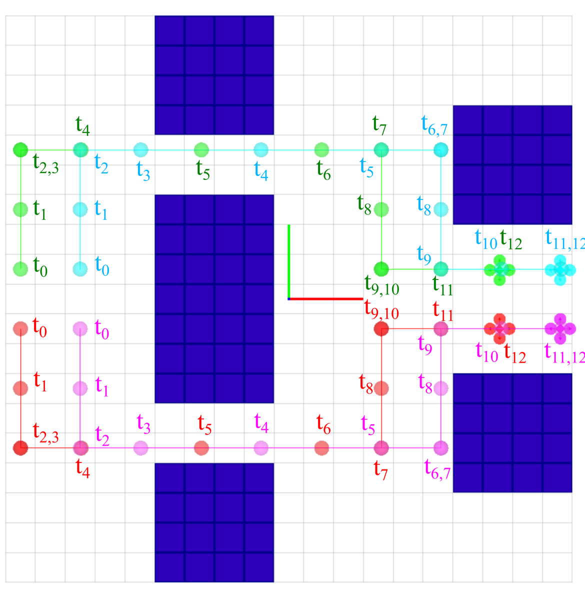


Goal: (I1 and I2) → B , (I3 and I4) → A

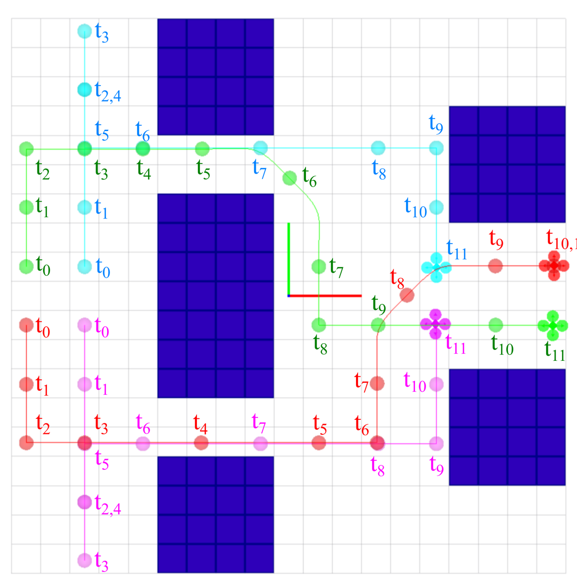
Invariants:

- Maintain a rectangular or linear formation
- Maintain a minimum distance

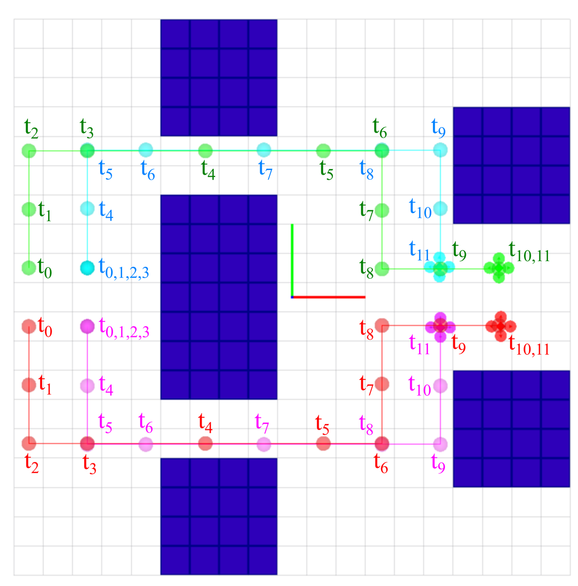
No motion plan that satisfies the formation constraint exists



Specification 1 (Optimal Trajectory)



Specification 2 (Sub-Optimal Trajectory)



Specification 2 (Optimal Trajectory)

Future Directions

- How to handle arbitrary LTL specification ?
- How to deal with change in environment?
- How to scale the synthesis to a large number of robots?
- How to deal with disturbance and uncertainty?

Potential Impact

- Automated and scalable mechanism to solve multi-robot planning problem for complex specification
- Many applications - monitoring, surveillance and disaster response, traffic control..

Joint work with Rattanachai Ramaithitima (UPenn), Vijay Kumar (UPenn), George Pappas (UPenn) and Sanjit Seshia (UC Berkeley)