HyRRT/HySST: Sampling-based Motion Planning Algorithms for Hybrid Dynamical Systems
Nan Wang and Ricardo G. Sanfelice

Overview

Summary

We propose two RRT-type algorithms to address motion planning problems for hybrid dynamical systems, which are characterized by their rapid search capabilities and are accompanied by the above theoretical guarantees:

1. Our RRT[1]-type algorithm to solve feasible motion planning problems for hybrid systems, which we refer to as HyRRT, is guaranteed to be probabilistically complete.
2. Our SST[2]-type algorithm to solve optimal motion planning problems for hybrid systems, which we refer to as HySST, is guaranteed to be asymptotically near-optimal.
3. Both algorithms possess the ability to rapidly search through high-dimensional problems.

Hybrid Systems

The Modeling Framework [3]

A hybrid system \( \mathcal{H} \) with state \( x \in \mathbb{R}^n \) and input \( u \in \mathbb{R}^m \):

\[
\mathcal{H} = \left\{ (x, u) \to \begin{cases} \dot{x}(t) = f(x, u), & (x, u) \in C \setminus D \setminus \mathcal{A} \\ \begin{cases} x(t') = x(t) + v(t'), & C \setminus \mathcal{A} \end{cases} \end{cases} \right\}
\]

- \( f(x, u) \) is the flow map
- \( v(x, t) \) is the jump map
- \( \mathcal{A} \) is the set of actuated jumping times
- \( \mathcal{B} = \mathcal{A} \cup \mathcal{A}^+ \)

Probabilistically Complete HyRRT Algorithm [4]

The inputs to HyRRT are \( X_0, X_f, X_g, \mathcal{M} \), the input library \( U_U \), and \( U_D \), a parameter \( \rho \in (0, 1) \), which tunes the probability of proceeding with the flow regime or the jump regime, and an upper bound \( K \in \mathbb{N} \) for the number of iterations to execute.

Algorithm 1 (HyRRT Algorithm)

1. \( X = X_0 \), \( X_t = \emptyset \), \( C = \emptyset \), \( \mathcal{A} = \emptyset \), \( J = \emptyset \)
2. If \( \rho > 0.5 \)
   
   \( J = J \cup \{(x, u') \to \{x(t') = x(t) + u'(t'), \forall t' \in (t, t') \} \} \)

3. Else
   
   \( J = J \cup \{(x, u') \to \{x(t') = x(t) + u'(t'), \forall t' \in (t, t') \} \} \)

4. For each \( x \) in \( X \), do
   
   \( X_t = X_t \cup \{(x, t') \to \{x(t') = x(t) + u'(t'), \forall t' \in (t, t') \} \} \)

5. If \( X_t \neq \emptyset \)
   
   \( J = J \cup \{(x, u') \to \{x(t') = x(t) + u'(t'), \forall t' \in (t, t') \} \} \)

6. If \( X_t \neq \emptyset \)
   
   \( X = X \setminus \{(x, u') \to \{x(t') = x(t) + u'(t'), \forall t' \in (t, t') \} \} \)

7. Return \( J \)

Asymptotically Near-Optimal HySST Algorithm [5]

In addition to the inputs to HyRRT, HySST requires parameters \( \beta > 0 \) and \( \beta > 0 \) to tune the optimization of the cost and specification of the vertices.

Algorithm 2 (HySST Algorithm)

1. \( X = X_0 \)
2. If \( \rho > 0.5 \)
   
   \( J = J \cup \{(x, u') \to \{x(t') = x(t) + u'(t'), \forall t' \in (t, t') \} \} \)

3. Else
   
   \( J = J \cup \{(x, u') \to \{x(t') = x(t) + u'(t'), \forall t' \in (t, t') \} \} \)

4. For each \( x \) in \( X \), do
   
   \( X_t = X_t \cup \{(x, t') \to \{x(t') = x(t) + u'(t'), \forall t' \in (t, t') \} \} \)

5. If \( X_t \neq \emptyset \)
   
   \( J = J \cup \{(x, u') \to \{x(t') = x(t) + u'(t'), \forall t' \in (t, t') \} \} \)

6. If \( X_t \neq \emptyset \)
   
   \( X = X \setminus \{(x, u') \to \{x(t') = x(t) + u'(t'), \forall t' \in (t, t') \} \} \)

7. Return \( J \)

Sampling-based Motion Planning Algorithms for Hybrid Dynamical Systems

Probabilistically Complete HyRRT Algorithm [4]

The inputs to HyRRT are \( X_0, X_f, X_g, \mathcal{M} \), the input library \( U_U \), and \( U_D \), a parameter \( \rho \in (0, 1) \), which tunes the probability of proceeding with the flow regime or the jump regime, and an upper bound \( K \in \mathbb{N} \) for the number of iterations to execute.

Algorithm 1 (HyRRT Algorithm)

1. \( X = X_0 \), \( X_t = \emptyset \), \( C = \emptyset \), \( \mathcal{A} = \emptyset \), \( J = \emptyset \)
2. If \( \rho > 0.5 \)
   
   \( J = J \cup \{(x, u') \to \{x(t') = x(t) + u'(t'), \forall t' \in (t, t') \} \} \)

3. Else
   
   \( J = J \cup \{(x, u') \to \{x(t') = x(t) + u'(t'), \forall t' \in (t, t') \} \} \)

4. For each \( x \) in \( X \), do
   
   \( X_t = X_t \cup \{(x, t') \to \{x(t') = x(t) + u'(t'), \forall t' \in (t, t') \} \} \)

5. If \( X_t \neq \emptyset \)
   
   \( J = J \cup \{(x, u') \to \{x(t') = x(t) + u'(t'), \forall t' \in (t, t') \} \} \)

6. If \( X_t \neq \emptyset \)
   
   \( X = X \setminus \{(x, u') \to \{x(t') = x(t) + u'(t'), \forall t' \in (t, t') \} \} \)

7. Return \( J \)

Asymptotically Near-Optimal HySST Algorithm [5]

In addition to the inputs to HyRRT, HySST requires parameters \( \beta > 0 \) and \( \beta > 0 \) to tune the optimization of the cost and specification of the vertices.

Algorithm 2 (HySST Algorithm)

1. \( X = X_0 \)
2. If \( \rho > 0.5 \)
   
   \( J = J \cup \{(x, u') \to \{x(t') = x(t) + u'(t'), \forall t' \in (t, t') \} \} \)

3. Else
   
   \( J = J \cup \{(x, u') \to \{x(t') = x(t) + u'(t'), \forall t' \in (t, t') \} \)