Path Planning With Differential Constraints

COMP 765 Final Course Project

Monica Patel (260728093)

April 19, 2017

McGill University

Understanding the problem

Let X denote the state space that applies to Kinodynamic system and let C represent the general configuration space. We want to find time-varying control inputs u(t) in U for t in [0,T] that drives the system from initial state x-start to x-goal in finite time T. The resulting trajectory must be collision free and satisfy differential constraints according to the state transition function

$$\dot{x} = f(x, u) \tag{1}$$

Working of Vanilla - RRT

- Does not give Optimal path (Almost surely).
- Assumes fully observable state space while sampling.
- Does not take into account various differential Constraints

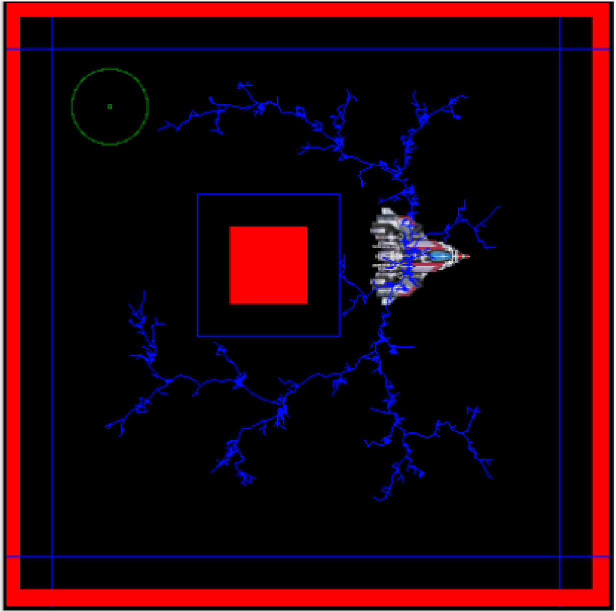
BUILD_RRT(q_{init})

$\mathcal{T}$.init(q_{init});
for $k = 1$ **to** K **do**
    $q_{rand} \leftarrow$ RANDOM_CONFIG();
    EXTEND($\mathcal{T}, q_{rand}$);
Return $\mathcal{T}$

EXTEND($\mathcal{T}, q$)

$q_{near} \leftarrow$ NEAREST_NEIGHBOR($q, \mathcal{T}$);
if NEW_CONFIG(q, q_{near}, q_{new}) **then**
    $\mathcal{T}$.add_vertex(q_{new});
    $\mathcal{T}$.add_edge(q_{near}, q_{new});
    if $q_{new} = q$ **then**
        Return *Reached*;
    else
        Return *Advanced*;
Return *Trapped*;

Implementation Results



Reachability-Guided Sampling for Planning

- RG-RRT takes the form of a modified RRT that explicitly accounts for the limitations of the system dynamics to shape the Voronoi bias so as to emphasize nodes within the tree that exhibit the greatest contribution towards exploring the state space.
- Main idea: Alleviate the sensitivity to distance metric.
- strictly ensure that any nodes added to the tree must make progress towards a given sample.
- For a given tree, (Subject to Dynamic Constraints) it is not able to expand into certain region of state space.
- Use of adaptive sampling strategy which constantly changes the allowed regions of samples for which expansion are attempted.

Reachability-Guided Sampling for Planning

Algorithm 1 $\mathcal{T} \leftarrow \textsc{BuildRRT}(x_{\text{init}})$

$\mathcal{T} \leftarrow \textsc{InitializeTree}()$;
$\mathcal{T} \leftarrow \textsc{InsertNode}(x_{\text{init}}, \mathcal{T})$;
for $k = 1$ to $k = K$ **do**
  $x_{\text{rand}} \leftarrow \textsc{RandomState}()$;
  $(x_{\text{near}}, x^{r}_{\text{near}}) \leftarrow \textsc{NearestState}(x_{\text{rand}}, \mathcal{T})$;
  while $x_{\text{near}} = \{\}$ **do**
    $x_{\text{rand}} \leftarrow \textsc{RandomState}()$;
    $(x_{\text{near}}, x^{r}_{\text{near}}) \leftarrow \textsc{NearestState}(x_{\text{rand}}, \mathcal{T})$;
  end while
  $u \leftarrow \textsc{SolveInput}(x_{\text{near}}, x^{r}_{\text{near}}, x_{\text{rand}}, \mathcal{T})$;
  $x_{\text{new}} \leftarrow \textsc{NewState}(x_{\text{near}}, u)$;
  $\mathcal{T} \leftarrow \textsc{InsertNode}(x_{\text{new}}, \mathcal{T})$;
end for
return $\mathcal{T}$

Reachability-Guided Sampling for Planning

- Sampling is less expensive than the process of finding the collision free trajectory from node while satisfying the differential constraints.
- RG-RRT actively constraints the set of nodes under consideration for nearest neighbor pairing with a sample to those that are actually able to expand towards the given sample.
- It does so by considering reachable region of state space associated with each node.
- For a state $x_0 \in \mathrm{X}$ and a finite local integration time step Δt, we define its reachable set, $R_{\Delta t}(x_0)$, to be the set of all points that can be achieved from x_0 in finite time, Δt and the set of available control inputs, U satisfying differential constraints.

Dubin’s Car and Reachability Set

Dubins´ car

$$\dot{x} = \sin\theta$$

$$\dot{y} = \cos\theta$$

$$\dot{\theta} = u, \qquad |u| \leqslant 1$$

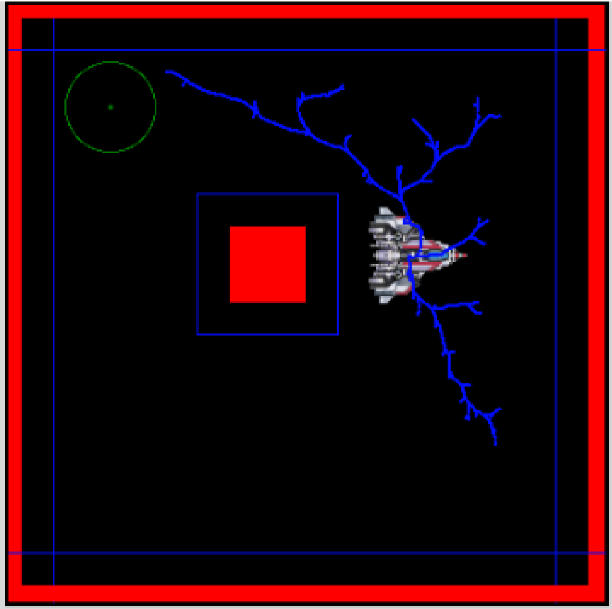
L. E. Dubins (1957). On curves of minimal length with a constraint on average curvature and with prescribed initial and terminal positions and tangents. Amer. J. Math., Vol. 79, 497516.

Reachability set

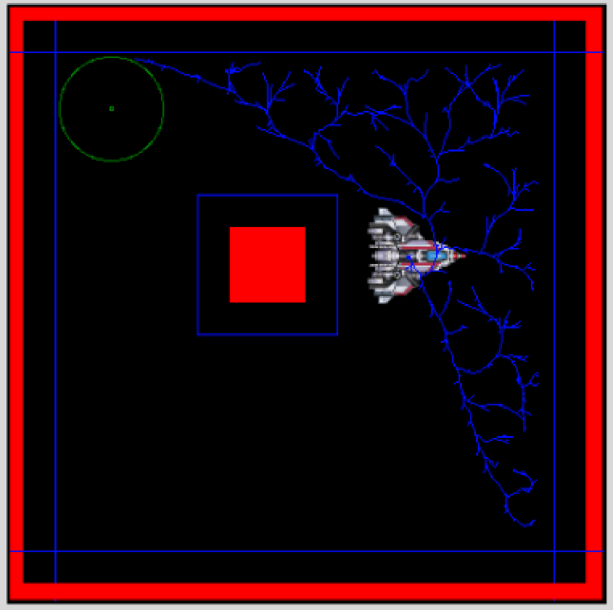
The time-limited reachable set for the Dubins car facing to the right (Planning Algorithms: Steven M. LaValle)

- The Dubins car can only drive forward. From an arbitrary configuration, the time-limited reachable set appears as shown in Figure. The time limit t is small enough so that the car cannot rotate by more than $\pi/2$

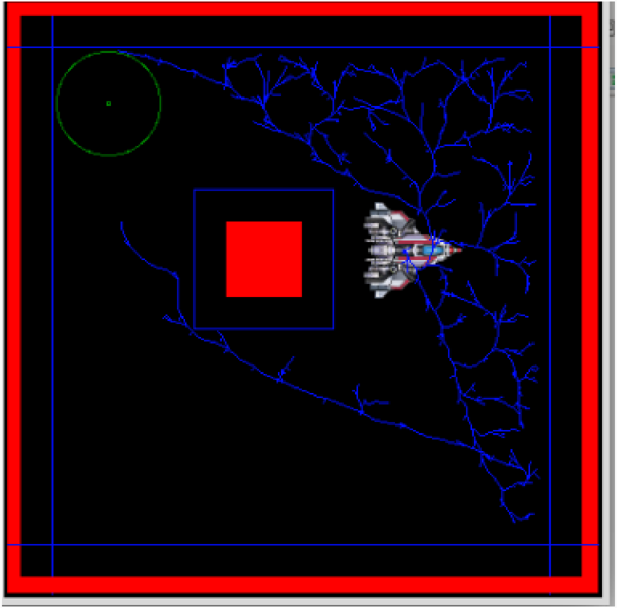
Implementation Results



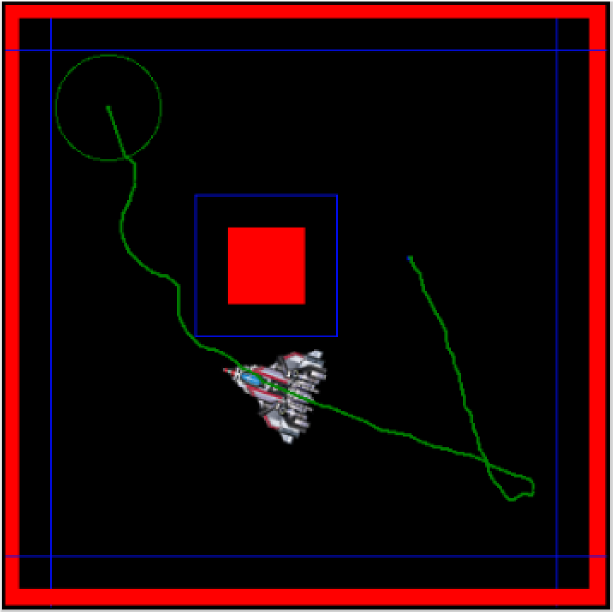
Implementation Results



Implementation Results



Implementation Results



Discussion

- Not the shortest path. No RG-RRG or RG-RRT*
- Since the sampling still done over the complete space, RG-RRT finds the path if it exists.
- Problem with sampling based planner, they do not stop if path doesn't exist, therefore best to restrict number of nodes. But How many?