Zachary Weiss

ME570 Final Project Abstract

Prof. Tron

6 November 2020

Rapidly exploring random trees are a commonly employed method of efficiently searching nonconvex and/or high-dimensional spaces. Within the context of robot motion planning, RRTs can be used to handle problems with additional constraints, in the form of degrees of freedom, kinematic bounds, and obstacles. In all algorithms intended for real-time use in autonomous robots, computational time is an important consideration. As such, this project sets out to consider the implementation of a pre-emptive RRT algorithm.

The jumping-off point will be a similar pre-emption scheme as proposed in the context of RANSAC (Nistér 2005), in which hypotheses are iteratively scored against observations, quitting with the best remaining hypothesis as the preferred one when the time/iteration budget is reached. Time permitting, and for easier benchmarking against current state-of-the-art, this scheme would also be integrated with other improvements / variants upon RRTs, that additionally decrease the convergence rate, or improve the solution optimality, such as A\*-RRT\*, informed RRT\*, and RT-RRT\*.

The solution will likely be best evaluated by a receiver-operating-characteristics-style charting of optimality (as gauged in relation to the non-preemptive version of the RRT variant the preemption is implemented on, as well as potentially to a ground truth optimality) to the time (or iteration) budget allotted. To do so, a large set of environments and start / goal positions will need to be generated and tested against, for both the unmodified and modified RRT variant(s). Generating such a comparison metric will enable easy evaluation of the algorithm to others, that can be plotted as a point within the curve’s space (time to convergence / budgeted time vs. path optimality), as well as an easy way to evaluate the practicality of the algorithm’s implementation in hardware that would allow for greater/lesser time budget allotments: i.e. what is the minimum hardware to achieve some optimality threshold.