

Motion Planning with Differential Constraints as Guided Search over Continuous and Discrete Spaces

Erion Plaku

Department of Electrical Engineering and Computer Science
Catholic University of America, Washington DC, 22064

Abstract

To compute a motion trajectory that avoids collisions, reaches a goal region, and satisfies differential constraints imposed by robot dynamics, this paper proposes an approach that conducts a guided search over the continuous space of motions and over a discrete space obtained by a workspace decomposition. A tree of feasible motions and a frontier of workspace regions are expanded simultaneously by first determining the next region along which to expand the search and then using sampling-based motion planning to add trajectories to the tree to reach the selected region. When motion planning is not able to reach the selected region, its cost is increased so that the approach has the flexibility to expand the search along new regions. Comparisons to related work show significant computational speedups.

Introduction

In motion planning, the objective is to compute motions that enable a robot to move to a goal region while avoiding collisions. In order to follow the planned motions in the physical world, it is essential to take into account the underlying robot dynamics during planning. Robot dynamics express physical constraints on the feasible motions, such as ensuring a minimum turning radius or keeping the wheels from sliding sideways. Such constraints are often modeled as a set of differential equations of the form $\dot{s} = f(s, u)$, where u denotes the control inputs that are applied to the state s .

Even though motion planning with differential constraints poses significant computational challenges, considerable progress has been made, especially by sampling-based methods which approach it as a search problem over the continuous space of feasible motions (Hsu et al., 2002; LaValle and Kuffner, 2001; Plaku, Kavraki, and Vardi, 2010; Sucas and Kavraki, 2012). As in discrete search, sampling-based methods expand the search as a tree, starting from the initial state. While in discrete search each state has finitely many successors, due to the continuous nature of motion planning, there are uncountably many different ways of expanding the search from a state. To handle this complexity, sampling-based approaches rely on a function $\text{MOTION}(s, u, f, dt)$,

which provides the motion trajectory to a successor state obtained by applying the control u to the state s and integrating the differential equations of motion f for a short time dt . If not in collision, the motion trajectory and the successor state are added to the tree. The control is sampled according to a probability distribution, usually uniform, to expand the search along different directions. The search continues to expand until the tree reaches the goal region in which case a collision-free and dynamically-feasible solution trajectory is obtained by concatenating the trajectories associated with the tree edges connecting the initial state to the goal region.

Method

This paper builds upon the success of sampling-based motion planning. To further improve the computational efficiency when dealing with differential constraints, the proposed approach simultaneously conducts a guided sampling-based search over the continuous space of motions and a discrete search over a workspace decomposition. The proposed approach is motivated by (Plaku, Kavraki, and Vardi, 2010), which has been shown to improve over RRT (LaValle and Kuffner, 2001) and other approaches by using discrete search to guide sampling-based motion planning. The workspace on which the robot operates is triangulated into a number of regions. The physical adjacency of the workspace regions is captured by the edges of a graph whose vertices correspond to workspace regions. In distinction from (Plaku, Kavraki, and Vardi, 2010), which would compute sequences of regions along which to expand the sampling-based motion planning, the proposed approach maintains a frontier of unreached regions. The overall search starts by rooting the tree \mathcal{T} at the initial continuous state and inserting the neighbors of the initial region to the frontier. Proceeding in an A* fashion, the region r with the lowest overall cost $\text{cost}(r) = \text{gcost}(r) + \text{hcost}(r)$ is selected from the frontier at each iteration of the search, where $\text{gcost}(r)$ denotes the cost to reach r and $\text{hcost}(r)$ denotes a heuristic cost to reach the goal region r_{goal} from r computed as the length of the shortest path in the graph $G = (R, E)$ from r to r_{goal} .

Sampling-based motion planning is then invoked for a short time with to expand the tree from the parent region $\text{parent}(r)$ to r . This expansion adds new collision-free and dynamically-feasible trajectories starting from vertices associated with $\text{parent}(r)$. Such trajectories are obtained by sam-

pling control inputs and integrating the differential equations of motions for several steps, and stopping the integration earlier if a collision occurs. If the expansion is successful, r is removed from the frontier and is inserted into a closed list. The neighbors of r , which do not appear in the closed list, are then inserted into the frontier. If the expansion from $parent(r)$ to r is not successful, then r remains in the frontier but its cost to the goal is increased. This provides the flexibility to consider other regions toward which to expand \mathcal{T} , which is essential to ensure an effective exploration of the continuous state space, since, due to collision avoidance and differential constraints, it may be difficult or even impossible to reach a particular region r . The search continues to expand the frontier and the tree \mathcal{T} until it reaches r_{goal} . The proposed approach is shown to offer significant computational speedups over related work in solving challenging motion-planning problems with differential constraints.

Experiments and Results

Experimental validation is provided by comparing the proposed approach to related work. The experiments use a second-order dynamics model of a tractor-trailer robot in which several links are attached to each other, as detailed in (Laumond, 1993; Plaku, Kavraki, and Vardi, 2010). By increasing the number of trailer links, the robot provides challenging test cases for high-dimensional motion-planning problems with differential constraints.

The workspace benchmarks provide challenging environments where the robot has to wiggle its way through numerous obstacles and narrow passages to reach the goal. The “random obstacles” benchmark is parametrized by the percentage p of the workspace area covered by randomly-added obstacles. The “random mazes” benchmark is parametrized by the number of dimensions p . A $p \times p$ maze is generated using a randomized version of the Kruskal’s algorithm. The computational efficiency of a method for a fixed benchmark type, parameter value, and number DOFs is measured as the average time to solve 30 random instances.

Fig. 1 provides a summary of the results when varying the number of DOFs. As more and more links are added to the robot, it becomes increasingly difficult to navigate through narrow passages and reach the goal. As shown in Fig. 1, the running time of RRT increases rapidly with the number of DOFs and times out on the high-dimensional problems. Syclop is considerably faster than RRT, but the running time still starts to slow down as more and more DOFs are added to the robot. The proposed approach yields significant speedups over RRT and Syclop, and efficiently solves even the high-dimensional problems.

Discussion

This paper focused on motion planning with differential constraints, where the objective is to compute a trajectory that reaches a goal region while not only avoiding collisions, but also satisfying differential constraints imposed by the robot dynamics. The motivation comes from navigation, exploration, search-and-rescue, and other robotics applications

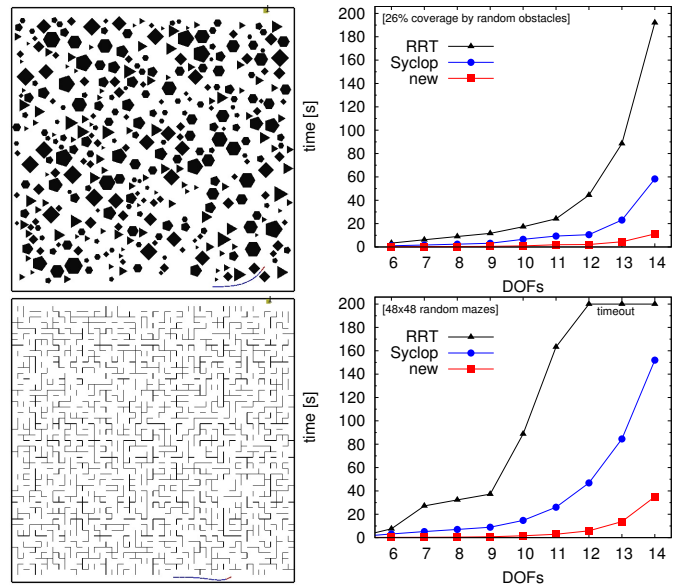


Figure 1: Results of the experiments for the “random obstacles” and “random mazes” benchmarks when varying the DOFs of the snake-like robot. The label new indicates the results of the proposed approach.

where it is essential to plan motions that the robot can follow in the physical world.

The proposed approach couples discrete search and sampling-based motion-planning in continuous state spaces, and by doing so opens up several venues for further research. In particular, the approach could benefit from more advanced discrete search techniques and improved heuristics to more effectively couple discrete search and sampling-based motion planning. Pruning or branch-and-bound techniques could be used in connection with sampling-based motion planning to determine tree vertices and regions from which to expand the search.

References

- Hsu, D.; Kindel, R.; Latombe, J. C.; and Rock, S. 2002. Randomized kinodynamic motion planning with moving obstacles. *International Journal of Robotics Research* 21(3):233–255.
- Laumond, J. 1993. Controllability of a multibody mobile robot. *IEEE Transactions on Robotics and Automation* 9(6):755–763.
- LaValle, S. M., and Kuffner, J. J. 2001. Randomized kinodynamic planning. *International Journal of Robotics Research* 20(5):378–400.
- Plaku, E.; Kavraki, L. E.; and Vardi, M. Y. 2010. Motion planning with dynamics by a synergistic combination of layers of planning. *IEEE Transactions on Robotics* 26(3):469–482.
- Sucan, I., and Kavraki, L. 2012. A sampling-based tree planner for systems with complex dynamics. *IEEE Transactions on Robotics* (99):1–16.