Abstract

Several randomized path planners have been proposed during the last few years. Their attractiveness stems from their applicability to virtually any type of robots, and their empirically observed success. In this paper we attempt to present a unifying view of these planners and to theoretically explain their success. First, we introduce a general planning scheme that consists of randomly sampling the robot's configuration space. We then describe two previously developed planners as instances of planners based on this scheme, but applying very different sampling strategies. These planners are probabilistically complete: if a path exists, they will find one with high probability, if we let them run long enough. Next, for one of the planners, we analyze the relation between the probability of failure and the running time. Under assumptions characterizing the “goodness” of the robot’s free space, we show that the running time only grows as the absolute value of the logarithm of the probability of failure that we are willing to tolerate. We also show that it increases at a reasonable rate as the space goodness degrades. In the last section we suggest directions for future research.

1 Introduction

Robot path planning has been proven a hard problem [40]. There is strong evidence that its solution requires exponential time in the number of dimensions of the configuration space, i.e., the number of degrees of freedom (dofs) of the robot. This result is remarkably stable: it still holds for rather specific robots, e.g., planar linkages consisting of links serially connected by revolute joints [17] and sets of rectangles executing axis-parallel translations in a rectangular workspace [13, 14]. Though general and complete algorithms have been proposed [6, 42], their high complexity precludes any useful application. This negative result has led some researchers to seek heuristic algorithms. While several such planners solve difficult problems, they also often fail or take prohibitive time on seemingly simpler ones. The fact that their behavior is not well characterized is a major drawback: they cannot be used as blackboxes in larger robot control systems.

The number of dofs beyond which existing complete algorithms become practically useless is low, somewhere between 3 and 5. This means that they cannot be used to compute paths for rigid objects translating and rotating in three dimensions, nor for six-dof manipulator arms, two important cases in practice. On the other hand, robot applications tend to involve more degrees of freedom than ever before. For example, an increasing number of manufacturing workcells use several cooperating robots to augment throughput and flexibility. Cells with more than twenty dofs are no longer exceptions. As costs and time for designing and deploying them become more critical, path planners integrated with CAD systems will be in higher demand to facilitate robot programming. Eventually, planners will run online to allow for non-deterministic sequences of goals and events [31]. Robots in domains other than manufacturing (e.g., medical surgery, space exploration) will also require efficient and reliable path planners. Some non-robotics domains raise a similar need as well. In computer graphics, animation of synthetic actors to produce digital movies or video games requires dealing with several dozen dofs. Here, motion planning may drastically reduce the work of human animators.
who currently input large numbers of key frames. In molecular biology, motion planning can help compute plausible docking motions of molecules modeled as spatial linkages with many dofs.

Collision-free path planning, which assumes perfect knowledge of the world and stationary obstacles, is only the most basic motion planning problem in robotics. Clearly, we would ultimately like robot planners to also deal with issues such as uncertainties, moving obstacles, movable objects, and dynamic constraints [29, 30]. But every extension of the basic problem adds in computational complexity. For instance, allowing moving obstacles makes the problem exponential in the number of moving obstacles [6, 41]; uncertainties in control and sensing make the problem exponential in the complexity of the robot environment [6]. Before we can effectively investigate such extensions in large configuration spaces, it seems that we must better understand how to practically solve basic path planning.

Path-planning applications are so diverse that it is infeasible to design a tailor-made algorithm for every possible robot. Instead, we need general path planning algorithms not bound to the specifics of any particular robot. We believe that between the two extreme types of planners suggested above — complete and heuristic — there is place for practically efficient general planners achieving a weaker form of completeness. In other words, we may perhaps trade a limited amount of completeness against a major gain in computing efficiency. Full completeness requires the planner to always answer a path-planning query correctly, in asymptotically bounded time. A weaker, but still interesting form of completeness is the following: if a solution path exists, the planner will find one in bounded time, with high probability. We call it probabilistic completeness. This weaker completeness becomes particularly interesting if we can show that the planner’s running time grows slowly with the inverse of the failure probability that we are willing to tolerate.

With this philosophy in mind, we have designed new path planners and experimented with them in large configuration spaces. One of them, described in [3, 4, 29], is a potential-field-based planner that escapes local minima by performing random walks; in the following, we will refer to it as the potential-field planner. Another planner, presented in [18, 20, 21, 24], precomputes a “roadmap” (network) of simple paths connecting randomly selected configurations and tries to construct a path between any two input configurations by connecting them to this roadmap; we will refer to this planner as the roadmap planner. Both these planners have been successfully applied to complex problems. For example, in [26], the potential-field planner was used to automatically synthesize a video clip with graphically simulated human and robot characters entailing a 78-dimension configuration space. Both the potential-field and the roadmap planners have been used to check that parts can be removed from an aircraft engine for inspection and maintenance [8]; here, paths are generated in configuration spaces having only six dimensions, but the parts have particularly complex geometry.

These two planners achieve probabilistic completeness. For the potential-field planner, this property remains qualitative: if there exists a path, the probability that the planner finds one tends toward one as the running time increases; but the convergence speed is unknown. But, for the roadmap planner, we have proven stronger results that relate the probability that it finds a path, when one exists, to its running time [18, 19, 22, 23]. In turn, these theoretical results suggest improvements of the planner.

Other work investigating similar or related randomized planning approaches include [1, 5, 15, 16, 35, 36]. Formal attempts to predict the behavior of specific random planners are proposed in very few papers [9, 28].

This paper proposes a consistent framework to describe and study the randomized planners cited above, with the goal to eventually build more powerful planners. In the planners cited above, the robot’s free space is not explicitly represented, but randomly sampled. We believe that this is the central concept underlying their success. In Section 2 we capture this concept into a general computational scheme for path planning in large configuration spaces. In Section 3 we make our discussion more precise by presenting our potential-field and roadmap planners as two instances of planners using this scheme, but applying two different sampling strategies. In Section 4 we give two formal analyses of the probabilistic completeness of variants of the roadmap planner. These analyses provide a theoretical explanation for the empirically observed success of the roadmap planner.

\footnote{In any case, very few tailor-made planners have been successfully designed for specific robots with more than four dofs.}

\footnote{This scheme can also be applied to configuration spaces having few dimensions; but it is less interesting in that case, since complete algorithms are then available.}