Computational Geometry 48 (2015) 589–605

Contents lists available at ScienceDirect

Computational Geometry: Theory and Applications

www.elsevier.com/locate/comgeo

On soft predicates in subdivision motion planning ☆

Cong Wang [a], Yi-Jen Chiang [a,*], Chee Yap [b]

[a] Department of Computer Science and Engineering, Polytechnic School of Engineering, New York University, Brooklyn, NY, USA
[b] Department of Computer Science, New York University, New York, NY, USA

ARTICLE INFO

Article history:
Available online 14 April 2015

Keywords:
Exact algorithms
Subdivision algorithms
Motion planning
Soft predicates
Resolution-exact algorithms

ABSTRACT

We propose to design new algorithms for motion planning problems using the well-known Domain Subdivision paradigm, coupled with "soft" predicates. Unlike the traditional exact predicates in computational geometry, our primitives are only exact in the limit. We introduce the notion of **resolution-exact algorithms** in motion planning: such an algorithm has an "accuracy" constant $K > 1$, and takes an arbitrary input "resolution" parameter $\varepsilon > 0$ such that: if there is a path with clearance $K\varepsilon$, it will output a path with clearance ε/K; if there are no paths with clearance ε/K, it reports "NO PATH". Besides the focus on soft predicates, our framework also admits a variety of global search strategies including forms of the A* search and probabilistic search.

Our algorithms are theoretically sound, practical, easy to implement, without implementation gaps, and have adaptive complexity. Our deterministic and probabilistic strategies avoid the Halting Problem of current probabilistically complete algorithms. We develop the first provably resolution-exact algorithms for motion-planning problems in $\mathrm{SE}(2) = \mathbb{R}^2 \times S^1$. To validate this approach, we implement our algorithms and the experiments demonstrate the efficiency of our approach, even compared to probabilistic algorithms.

1. Introduction

A central problem of robotics is motion planning [4,20,21,10]. In the early 80's there was strong interest in this problem among computational geometers [15,32]. This period saw the introduction of strong algorithmic techniques with complexity analysis, and the careful investigation of the algebraic configuration space (C-space). In particular, Schwartz and Sharir [31] showed that the method of algebraic cell decomposition is a universal solution for motion planning. We introduced the retraction method in [24,33,34]. In the first survey of algorithmic motion planning [40], we also showed the universality of the retraction method. This method is now commonly known as the road map approach, popularized by Canny [8] who showed that its algebraic complexity is in single exponential time. Typical of algorithms in Computational Geometry, these exact motion planning algorithms assume a computational model in which exact primitives are available in constant time. Implementing these primitives exactly is non-trivial (certainly not constant time), involving computation with algebraic numbers.

In the 1990s, interest shifted back to more practical techniques. Today, the dominant approach is based on sampling, usually combined with randomization. The most well-known representative of the sampling approach is the **probabilistic**

☆ An extended abstract of this paper appeared in *Proc. ACM Symposium on Computational Geometry (SoCG'13), pp. 349–358, 2013.* This work is supported by NSF Grant CCF-0917093 and DOE Grant DE-SC0004874.

* Corresponding author.
E-mail addresses: cwang05@students.poly.edu (C. Wang), chiang@nyu.edu (Y.-J. Chiang), yap@cs.nyu.edu (C. Yap).

http://dx.doi.org/10.1016/j.comgeo.2015.04.002
0925-7721/

roadmap method (PRM) [19]. The idea is to compute a partial road map by random sampling of the C-space. PRM offers a computational framework for a large class of algorithms. Moreover, many variants[1] of the basic framework have been developed (see [21,10]). Most sampling methods take sample points in configuration space, but the recent paper from Halperin's group [29] takes sample (parametrized) subsets of configuration space. In an invited talk at the IROS 2011 Workshop on Progress and Open Problems in Motion Planning.[2] J.C. Latombe stated that the major open problem of such **Sampling Methods** is that they do not know how to terminate when there is no free path. In practice, one would simply time-out the algorithm, but this leads to issues such as "Climber's Dilemma" [16, p. 4] that arose in the work of Bretl (2005). We call this the **halting problem** of PRM, viewed as the ultimate form of what is popularly known as the "Narrow Passage Problem" [10, p. 216]. Latombe's talk suggested promising approaches such as Lazy PRM [3]. The theoretical foundation of PRM is based on two principles: probabilistic completeness, and fast convergence under certain "expansiveness" assumptions [18] about the environment. It is unclear how to check these assumptions on specific environments. For a comprehensive overview of motion planning, see Lavalle [21] and Choset et al. [10].

In this paper, we turn to a third popular approach [46] for motion planning, which we call **Subdivision Methods**. The general idea is to subdivide some bounded domain B_0, typically a subset of $\mathbb{R}^d$. In motion planning, the domain is a subset of configuration space. In its simplest form, the subdivision of B_0 can be represented as a **subdivision tree**, which is a generalization of binary trees ($d = 1$) or quad-trees ($d = 2$). An early reference for this approach is Brooks and Lozano-Perez [5]. Recent subdivision references include [46,2,45,12,26]. Manocha's group has been active and highly successful in producing practical subdivision algorithms for a variety of tasks, not just in motion planning (e.g., [38,36]). Domain subdivisions are sometimes known as "cell decomposition" (e.g., [46]), but we reserve "cell decomposition" for the approaches based on partitioning the configuration space into algebraic "cells" with bounded combinatorial complexity that are directly correlated with the combinatorial features on the obstacles (e.g., [30,40]). In contrast to such cells, the boxes in subdivision approaches are more related to "resolution". Nevertheless, subdivision that takes into account combinatorial complexity may be seen in [45,46]. Such kinds of subdivision algorithms offer tantalizing opportunities for new kinds of complexity analysis. Examples of such analysis may be seen in [28,35,7].

¶1. Contributions of this paper Although subdivision algorithms have been widely used by practitioners, their theoretical foundations have so far been lacking. This paper begins this task.

The notion of "resolution completeness" is widely used in the motion planning literature [10] but rarely analyzed (Section 5 discusses some issues). Our first contribution is to introduce the concept of **resolution-exact** (or ε**-exact**) **planners**. Such planners accept an input **resolution parameter** $\varepsilon > 0$. The planner has an **accuracy constant** $K > 1$, independent of the input, such that if there is a path of clearance $K\varepsilon$, it will output a path with clearance ε/K; if there is no path of clearance ε/K, it will output "NO PATH". As this paper shows, our definition allows us to devise planners that avoid the halting problem of PRM. Moreover, Section 5 notes that the usual concept of "resolution completeness" does not automatically solve the halting problem. But in what sense have we "solved" the halting problem? To be sure, we are *not* solving the halting problem for exact motion planning—this would require exact computation, something we wish to avoid in robotics. Instead, ε-exactness weakens the requirements for the "NO PATH" output. But is this just a trick to solve the halting problem by fiat? No, we argue that our weakening is not only justifiable, but desirable: good engineers know the limits of accuracy in their sensors, actuators, robot dimensions, etc. Path planning that depends on accuracy beyond these limits is not realistic, even dangerous. Note that when we output "NO PATH", we guarantee that there exists no path with clearance $K\varepsilon$ (this is the contrapositive of the statement just mentioned above: "if there is a path of clearance $K\varepsilon$, it will output a path with clearance ε/K"); no similar guarantees can come from PRM. With this information, users can choose ε based on engineering limits so that when we declare "NO PATH", no further search is warranted.

Our second contribution is the introduction of **soft primitives** for designing resolution-exact planners. Briefly, soft primitives are suitable numerical approximations of exact (hard) primitives. Such primitives are perhaps nascent in previous literature. But by making this idea explicit, we open up many new possibilities, as well as lay the groundwork for a systematic investigation of such algorithms. Such primitives are relatively easy to implement correctly (i.e., there are no "implementation gaps" in such algorithms).

Third, we design new planners based on soft predicates. These algorithms are the first explicit examples of resolution-exact planners. Our algorithms can use various search strategies, including probabilistic ones. Halting is guaranteed even in our probabilistic planners.

Our final contribution is the development and implementation of the first resolution-exact algorithms for rigid robots with configuration space $\mathrm{SE}(2) = \mathbb{R}^2 \times S^1$. Our experiments demonstrate their effectiveness.

2. On numerical computational geometry

Computational Geometry (CG) has traditionally concentrated on **Exact Methods**. The attractive features of exact algorithms are well-known. The drawback of such methods is exposed when we start to implement the algorithms. The inability

[1] A partial list includes Expansive-Spaces Tree planner (EST), Rapidly-exploring Random Tree planner (RRT), Sampling-Based Roadmap of Trees planner (SRT).

[2] http://www.cse.unr.edu/robotics/tc-apc/ws-iros2011. Sept. 30, 2011, San Francisco.