Computer-Aided Design 70 (2016) 116–125

Contents lists available at ScienceDirect

Computer-Aided Design

journal homepage: www.elsevier.com/locate/cad

Efficient global penetration depth computation for articulated models[☆]

Hao Tian [a,b,c], Xinyu Zhang [a,b,c,*], Changbo Wang [a,c], Jia Pan [d], Dinesh Manocha [e]

[a] Shanghai Key Laboratory of Trustworthy Computing, Shanghai, China
[b] MoE Engineering Research Center for Software/Hardware Co-design Technology & Application, China
[c] East China Normal University, Shanghai, China
[d] The University of Hong Kong, China
[e] The University of North Carolina at Chapel Hill, USA

HIGHLIGHTS

- Novelty: The first global PD approach for high-DOF articulated models.
- Generality: Handling hybrid joints and links represented using polygonal models.
- Conservativeness: Guaranteeing that the configuration realizing PD is penetration free.
- Efficiency: Taking about 0.03–3 ms per runtime PD query in our experiments.

ARTICLE INFO

Keywords:
Configuration space
Articulated models
Penetration depth
Support vector machine

ABSTRACT

We present an algorithm for computing the global penetration depth between an articulated model and an obstacle or between the distinctive links of an articulated model. In so doing, we use a formulation of penetration depth derived in configuration space. We first compute an approximation of the boundary of the obstacle regions using a support vector machine in a learning stage. Then, we employ a nearest neighbor search to perform a runtime query for penetration depth. The computational complexity of the runtime query depends on the number of support vectors, and its computational time varies from 0.03 to 3 milliseconds in our benchmarks. We can guarantee that the configuration realizing the penetration depth is penetration free, and the algorithm can handle general articulated models. We tested our algorithm in robot motion planning and grasping simulations using many high degree of freedom (DOF) articulated models. Our algorithm is the first to efficiently compute global penetration depth for high-DOF articulated models.

© 2015 Elsevier Ltd. All rights reserved.

1. Introduction

Computing the magnitude of inter-penetration between two overlapping rigid/articulated objects is a fundamental problem in computational geometry. One metric that is widely used to measure the extent of inter-penetration is penetration depth (PD), which requires computing a minimum transformation (translation and rotation) to separate two overlapping objects. The resulting transformation motion can be used to compute the contact force for penalty-based methods, valid poses in grasping simulations, force/torque feedback in haptic rendering, sample generation in narrow passages for motion planning, etc.

The exact computation for PD, particularly the so-called generalized PD that involves both translation and rotation [1], can be reduced to arrangement computation in a high-dimensional configuration space that has high computational complexity. For instance, the combinatorial complexity of exact PD is as high as $\mathcal{O}(n^{12})$ [2] for two models with n triangles in three-dimensional space. As a result, all practical algorithms tend to compute an approximate solution. A wide variety of algorithms have been proposed in the literature for rigid models (e.g. [3,4,2,5–8]). For articulated models, the resulting configuration spaces are high-dimensional non-Euclidean spaces. For instance, the configuration

☆ This paper has been recommended for acceptance by Scott Schaefer and Charlie C.L. Wang.

* Correspondence to: East China Normal University, School of Software, 3663 North Zhongshan Road, Shanghai, 200062, China. Tel.: +86 21 62235706.
E-mail addresses: xyzhang@sei.ecnu.edu.cn, zhang.xinyu@live.com (X. Zhang).

http://dx.doi.org/10.1016/j.cad.2015.07.007
0010-4485/© 2015 Elsevier Ltd. All rights reserved.

space for a stationary obstacle and a six degrees of freedom (DOFs) robot arm fixed on the ground, is a six-dimensional non-Euclidean space. If we allow the arm base to move in space, its configuration space becomes nine-dimensional and non-Euclidean. As the number of joints increases, the complexity of a configuration space can become very high. In particular, if self-collisions between distinctive links must be considered, the complexity can increase rapidly. These self-collisions may correspond to many small and isolated components in the high-dimensional configuration space. Due to its high complexity, it is challenging to produce an exact representation of a space with such high dimensionality. To the best of our knowledge, only one recent study [9] has attempted to compute PD between articulated models, but its solution yielded only locally optimal PD.

Main result: We present an efficient algorithm to approximate the global PD in high-dimensional spaces for articulated models. Built on the early framework proposed in [8,10], we use a machine learning method to approximate the boundary of the obstacle regions in the configuration space for an articulated model and its surrounding obstacles. We generate a set of configuration samples to densely cover the boundary of obstacle regions. Given a query configuration for computing PD, the closest configuration can be found quickly using approximate k-nearest-neighbor search. The magnitude of PD can be computed using non-Euclidean distance metrics between the query configuration and the closest configuration. Compared with existing methods, our method has the following advantages:

Novelty: Our algorithm is the first global PD approach for high-DOF articulated models.

Generality: Our algorithm can handle hybrid joints and links represented using polygonal models.

Conservativeness: Our algorithm can guarantee that the configuration realizing PD is penetration free. This feature is particularly important for enforcing non-penetration constraints in certain applications, such as dynamic simulation, motion planning, and grasping simulation.

Efficiency: Our algorithm takes about 0.03–3 ms per runtime PD query on single core. The computational complexity of runtime query depends only on the number of support vectors used in learned obstacle regions.

The rest of the paper is organized as follows. In Section 2, we provide a review of the related work on PD computation. In Section 3, we introduce the notation that we use in the paper and present the algorithm for approximating obstacle regions for articulated models. In Section 4, we describe our approach to compute PD by using approximate obstacle regions and a solution for computing conservative PD. Section 5 describes the implementation details and some basic experimental results. Section 6 highlights the results on complex scenarios.

2. Related work

There are two types of PD: translational PD and generalized PD. Translational PD corresponds to a translational motion to separate two objects, whereas generalized PD corresponds to both translational and rotational motions. Various work on PD computation has been reported in computer graphics, geometric modeling, haptics, and robotics, and most of the associated algorithms address rigid models. In the following, we offer a brief overview of these algorithms.

Translational penetration depth: Exact translational PD can be formulated using the Minkowski sum; it is obtained by computing the closest distance from the origin to the boundary of the Minkowski sum [11,12]. The worst-case complexities for these approaches is $\mathcal{O}(n^2)$ for convex polytopes and $\mathcal{O}(n^6)$ for non-convex polytopes, where n is the number of features in the polytopes [11]. Because of the high computational cost involved in computing exact translational PD, most practical approaches compute an approximation instead. For convex polytopes, many methods simply compute an approximate Minkowski sum, which is then used as an approximate translational PD calculation [13,14]. Translational PD between non-convex objects is typically computed using convex decomposition, which is based on the fact that an exact Minkowski sum can be computed based on convex decomposition and the union of all the pairwise Minkowski sums [15]. As it is expensive and not particularly robust to compute the union explicitly, many approximate solutions have been proposed, including GPU-based approaches [2,16] and methods that are based on reduced convolution and filtering [17,18]. Other methods avoid the expensive Minkowski union entirely by computing only translational PD between each pair of convex components. These approaches are called local methods because the resulting PD only depends locally on the closest point on the penetrated surfaces and may not result in a globally consistent solution. Most local PD methods are based on local features [19,20,18,21,22], i.e., each convex piece generated by the convex decomposition is a mesh triangle. Some recent methods [7] uses iterative optimization to compute approximate translational PD.

Generalized penetration depth: Few algorithms can compute generalized PD, which considers both translation and rotation. [1] estimate the upper and lower bounds for generalized PD between two general polyhedral models by decomposing the models into convex components. Most practical algorithms for generalized PD computation follow the iterative, constrained optimization framework, which generates a series of configurations on the contact space with decreasing distances to the given in-collision query. [5] generates such a series of configurations by moving a small step size from a configuration along the gradient direction. [6] first compute an approximate local contact space near a configuration and then perform random sampling within the approximate contact space to find a suitable following configuration. [23] calculate a linearized contact space in the neighborhood of a configuration and then obtain an optimal following configuration by solving a linear complementary problem (LCP). Most of these approaches [1,6,5] are slow for interactive applications and do not have the necessary guarantees for a global solution.

Penetration depth for articulated models: Recently, [9] present an algorithm to approximate PD for articulated models. Their algorithm approximated a local configuration space using iterative and constrained optimization techniques and provided a locally optimal PD.

Other penetration depth metrics: In addition to the related work above, there are other definitions of PD. Intersection volume and its derivative can also be used for volume-based repulsion [24]. Distance fields are also used for local translational PD computation [25] and can be computed in realtime using GPUs. Point-based Minkowski sum approximation [17] can also be used to compute translational PD.

3. Configuration space approximation

3.1. Notation

We denote an articulated model as $\mathcal{A}$ and an obstacle as $\mathcal{O}$, as shown in Fig. 1. We assume that $\mathcal{A}$ has n links. The ith link of $\mathcal{A}$ is denoted as $\mathcal{L}_i$, and the link attached to $\mathcal{L}_i$ is denoted as $\mathcal{L}_{i+1}$. The joint connecting two links, $\mathcal{L}_i$ and $\mathcal{L}_{i+1}$, is denoted as $\mathcal{J}_i$. Joint $\mathcal{J}_i$'s parameter is denoted as θ_i. We denote the configuration space composed of $\mathcal{A}$ and $\mathcal{O}$ as $\mathcal{C}$ and each configuration $\mathbf{q}$ in $\mathcal{C}$ corresponds to the relative configuration (i.e., translation and rotation)

Download English Version:

https://daneshyari.com/en/article/440000

Download Persian Version:

https://daneshyari.com/article/440000

Daneshyari.com