Continuous Collision Detection for Robotic Motion Planning

Semester’s Thesis/Master’s Thesis

Advisor: Jonas Wittmann (MW3133, jonas.wittmann@tum.de, +49 89 289 15208)

Introduction and Problem Description

Collision checking and distance computation are commonly integrated in existing path planning frameworks as third party libraries and are the most important submodules for achieving a collision free motion. Often the broad phase and the narrow phase are distinguished. The former aims to reduce the number of pairs for which collision calculations are performed so that the overall complexity of the algorithm is not \( O(n^2) \). Different methods are applied at this stage, including space partitioning, bounding volumes etc. The latter ones include among others axis-aligned bounding boxes (AABB) and oriented bounding boxes (OBB) [4].

"The motion planning framework MoveIt! currently only implements discrete collision checking for movements of the controlled robot. A major drawback of discrete collision checking methods is that they may miss collisions between the sampled time steps. While there exists techniques to alleviate this problem, resulting algorithms can be relatively slow. To provide stronger guarantees, continuous collision detection (CCD) techniques have been proposed by the research community. They compute the first time of contact between two moving objects along a path." [3]

A continuous collision detection approach recently was integrated in MoveIt! based on the Bullet open-source library [1]. It is an alternative for the default collision detection algorithm within MoveIt! that is based on the Flexible Collision Library [2].

This thesis investigates the current CCD implementation available in MoveIt! and compares the approaches with respect to their performance and reliability for task space motion planning (Point-to-Point motions). The robot used for evaluation and validation is a FRANKA EMiKA Panda robot.

Task Description

After a familiarization on the topic, the concepts within the Flexible Collision Library (FCL) [2] and within the Bullet library [1] have to be compared concerning the theoretical background, their drawbacks and advantages. The API that MoveIt! offers for both of the libraries has to be investigated for potential limitations and adapted if necessary. The corresponding literature survey has to provide and overview of the current state-of-the-art for real-time collision detection and distance calculations in robotic applications.

Both the concepts of discrete and continuous collision detection have to be implemented within the existing C++ motion planning framework of the FRANKA EMiKA Panda robot. Based on simple Point-To-Point motions in the robot's task space the two approaches have to be compared concerning their performance and reliability. Besides the comparison of discrete collision checking within FCL and continuous collision checking within Bullet, the current collision model improved. Instead of the mesh geometries, capsules (already available within FCL) or Swept Sphere Volumes should be used instead for more efficient distance calculations. The evaluations have to take place both in simulation and on the real system in the laboratory.

Requirements

- Good Programming Skills in C++ | Familiar with Linux
- Robotics
- Linear Algebra and Computational Geometry beneficial
- Ideally familiar with ROS and MoveIt!
- Reliable, independent and structured way of working

References