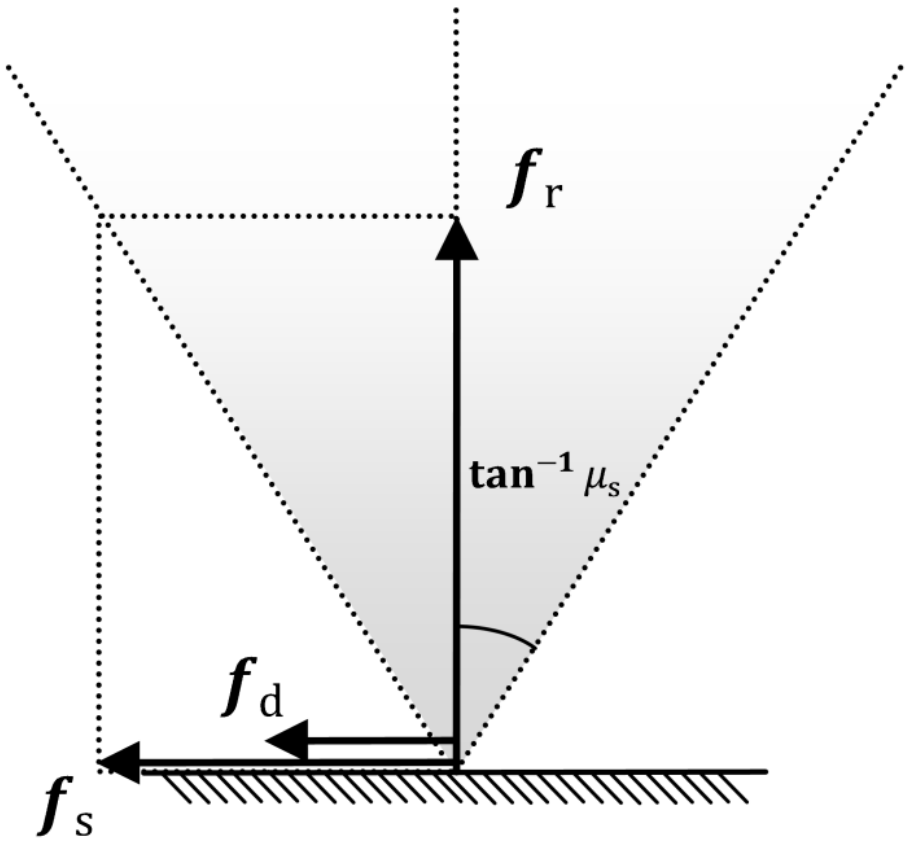
Practical 8: Rigid Bodies, part 3

Babis Koniaris



Introduction

The goal of this practical is to deliver a working rigid body simulation that includes the following features:

- Works for a solid cuboid
- Collision detection with a horizontal plane
- Impulse-based collision response
- Simulates friction with a horizontal plane

Tasks

Task 1: Application of an impulse

For this task and all subsequent tasks, create a default cubic rigid body scale it by a factor of **3 along the y axis**. You will end up with a cuboid that has a width of 2, a height of 6 and a depth of 2. Assign a **mass of 2** to it.

- Output the inverse inertia matrix of the rigid body on the console [1]
- Assign an initial velocity of (2,0,0) and initial angular velocity of (0,0,0) to the solid. Do not assign any force to it. After 2 seconds, apply a single impulse to the solid so that its center of mass comes to a stop and the solid starts spinning clockwise.

Task 2: Collision detection

Task 2: Detect which vertex or vertices of a cuboid collide with the ground plane

To demonstrate your solution, output the following information to the console when a collision is detected:

- Coordinates of all colliding vertices [2]

[1] Note that you can display any glm object as a string using glm::to_string, which is part of the glm/ext.hpp library

[2] 1 if the collision is with a vertex, 2 if it's with an edge and 4 if it's with a face

- Average of all colliding coordinates [3]

Task 3: Collision response

Task 3: Simulate the collision between a rigid body (cuboid) and the ground plane using impulse-based collision response

Here are the specific simulations you will demonstrate:

- Assign an angular velocity of (0,0,0.5) to the solid and an elasticity of 1 [4]
- Assign an angular velocity of (0.1,0.1,0.1) to the solid and an elasticity of 0.7

You are welcome to demonstrate other cases too.

Task 4: Friction

Task 4: Add a model of friction to your simulation and any other operations of your own design that will make the solid stop in a realistic fashion.

The focus of this task should be to achieve a realistic simulation for an elasticity of around 0.6. You should aim to make the solid stop translating and rotating realistically. Use the theory *liberally* and *creatively* to achieve your goal.

Deliverables

Marking scheme

Here's a summary of what you need to deliver and allocated marks:

- **Task 1: Impulse application: 5 marks**
- **Task 2: Collision detection: 6 marks**
- **Task 3: Collision response (no friction): 9 marks**

[3] Hopefully you will have figured out in last week's practical that this information is mandatory to implement the collision response!

[4] this refers to the elasticity value used in the collision response impulse

- **Task 4: Collision response (w/ friction): 5 marks**

Submission details

You must submit your work using the relevant Moodle assignment by the deadline specified on Moodle. Please submit your zipped code and executable.