

# Homework 1

Robot Autonomy  
CMU 16-662, Spring 2016  
Due Date: 12 February, 5 pm

TAs: Laura Herlant, Shushman Choudhury\*

## 1 Introduction

In this homework, you'll be implementing some of the grasp metrics we discussed in class. Additionally, you'll consider how we can compensate for noise. You will be working in groups of three or four - check an enclosed PDF for your group assignments. Any one of you can submit the homework - please indicate all members of your group (name and andrew id) in the writeup.

Do note that nobody has a uniquely good solution for grasp metrics, so while we expect the results to be somewhat reasonable, there will be many things that look "funny". You're expected to point this out when you notice it.

## 2 Overall Goal

By the end of this, you should produce two sets of grasps, ordered by your evaluation function, and ordered by your evaluation function with noise taken into account. There are 3 target items you will try.

## 3 Grasp Metrics

Openrave allows you to generate grasps automatically, using `openravepy.databases.grasping`. The end result of that is an ordered set of grasps it believed were good using openrave's metric. First, check how their metric does. You can even look up the code to see the exact function, if you'd like (though you aren't required to).

Next, you'll have to devise your own grasp metric. We discussed a few ideas in class. In the code, we have already written up how to get the contacts from an openrave grasp, and extract the position and direction of each. You'll need to turn those into wrenches and evaluate with a function of your choosing. Fill in the `eval_grasp` function for this. Be sure to describe it, and include results in your writeup!

## 4 Adding Uncertainty

Often times, these grasps require your contacts be made at very specific locations to work well. However, it's very difficult to have the contacts made at those exact locations. Both the robot end effector and the object will likely not be at the location believed.

We'll try and account for this by incorporating this into our ranking. To do so, we will generate random grasps, and combine their scores.

First, fill in the function `sample_random_grasp` to take in a grasp and modify the position and orientation randomly. We have sampled using a Gaussian distribution with sigmas listed in the code (see

---

\*Adapted from assignment by Kyle Strabala

the function `numpy.random.normal`, but you are free to do something else, as long as it is reasonable. Do mention that in your report.

Next, you'll need to call your evaluation function on each of these, and combine the scores. Fill in `order_grasps_noisy.py` so that it outputs a new ranked list of grasps. How did you decide to combine them?

## 5 Deliverables Checklist

We've marked the sections that need to be filled in with a TODO. Please submit both the code and the writeup. You can code however you like so long as it's correct. However, writeup items cannot exceed 5 sentences - so be brief!

The following files are expected from you:

- A modified `hw1_grasp.py`
- A writeup in pdf form (name it `hw1-group<group-ID-from-teamlist>.pdf`)

Only one person per groups needs to submit the assignment, but the names and Andrew IDs of the partners should be in the report. Put everything in a zip file called `hw1-group<group-ID-from-teamlist>.zip`

## 6 Grading

This is how the assignment will be evaluated:

- Try out the openrave grasps on the champagne bottle - put in a picture of the top rated one. Do you think they will work in practice? (1 pt)
- Come up with an evaluation function for testing grasps. Describe it in your writeup. Put in images of your top 4 ranked grasps for each of the items in your writeup. (10 pts)
- Complete the code to sample a new position and orientation for a grasp (5 pts)
- Complete the code in `order_grasps_noisy.py` to evaluate the random samples and combine them. Put in images of the top 4 ranked grasps for each item in the writeup. Did adding noise help? How did you decide to combine the noisy scores? (3 pts)
- Tell us how long this homework took you. (1 pt)