EECS 206B Lab 2 Report: Grasp Planning with Baxter and Sawyer

Aakash Parikh, Federik Warburg, Jun Zeng

February 2019

1 Introduction

In this lab, we work on related topics about grasp. We plan grasp on different objects, execute each grasp on Baxter robot, and compare the success rate on the physical system with the success rate predicted by several grasp quality metrics.

A video of Baxter executing various grasps can be found on YouTube. We also include our GitHub repository here.

2 Approach and Methods

Grasp planning involves finding object, planning grasp, moving arm to gripper position, executing grasp, lifting and placing object. Furthermore, there is a feedback loop where automatic quality checking and setting are considered. In this section, we present the approaches and methods that have been used in this workflow.

2.1 Finding Object and Planning Grasp

Given an object, we can execute the mesh sampling to generate a set of contact points. After that, we can samples a bunch of candidate grasps whose pairs of vertices can be chosen randomly, while the ones which are too big or small for the gripper or too close too the table be thrown out.

In our implementation, the normal vector of each vertex is considered and the local frame of the vertex is constructed through an simple cross product multiplication. For each contact there are infinity choices of local frame construction. However, we can uniquely determine the orientation by constraining the y axis (plane of the grippers) to be parallel to the line connecting the two grasp points. We also decided that the approach direction will be determined by a ray from the midpoint to the the COM of the object. Based on it, we could calculate the transformation between from the local frame to the world frame. This transformation allows us to align the coordinates of the object, the body and the gripper.

2.2 Force Closure Metric

The definition of the force closure metric is that a grasp can resist any applied wrench. One equivalent definition of this property can be described if the line connecting the contact points lies inside both friction cones.

Our implementation of force closure metric exploits this fact, which makes the implementation quite simple. Based on the positions of two contact points A and B, we can compute the intersection angle α between the line from contact A to B. The normal vector $\vec{n}$ should be smaller than the angle of the cone α_0. Therefore for two contact points, we have the force closure metric as below,

$$\cos\alpha_1 = \frac{\vec{AB}\cdot\vec{n}_1}{2||\vec{AB}||||\vec{n}_1||} \geq \cos\alpha_0 \tag{1}$$

$$\cos\alpha_2 = \frac{\vec{BA}\cdot\vec{n}_2}{2||\vec{BA}||||\vec{n}_2||} \geq \cos\alpha_0 \tag{2}$$

The angle of cone α_0 is set as 0.2 during our implementation. We check this for both friction cones.

2.3 Gravity Resistance Metric

Given two contact points and the force magnitude vectors f_1 and f_2 are noted,

$$f_1 = [f_{11}, f_{12}, f_{13}, f_{14}]^T, \tag{3}$$

$$f_2 = [f_{21}, f_{22}, f_{23}, f_{24}]^T. \tag{4}$$

The grasp matrices for two contact point are written as G_1 and G_2 with dimension 6×4. Then the grasp metric can be regarded whether we have feasible values for f_1 and f_2 so that $G_1 f_1 = -w_1$ and $G_2 f_2 = -w_2$. This defines the gravity resistance as computing whether the grasp can resist gravity pulling down on the object.

To reformulate the feasibility of metric as an optimization matrix, we firstly construct the combined vector 16×1 matrix f^* from f_1 and f_2 defined in (3) and (4) as below,

$$f^* = [f_{11}, f_{12}, f_{13}, f_{14}, t_{11}, t_{12}, t_{14}, 0, f_{21}, f_{22}, f_{23}, f_{24}, t_{21}, t_{22}, t_{24}, 0]^T \tag{5}$$

where we have some constraints as

$$t_1 = |f_1|, \quad t_2 = |f_2|, \quad t_1 + t_2 - \mu f_3 \leq 0. \tag{6}$$

We also construct the combine matrix for the grasp matrix $G = [G_1, G_2]$. The related matrix P and vector q for the optimization are presented as below. Therefore, the feasibility of solution can be formulated as a QP optimization problem described as below,

$$\begin{aligned} &\min_{f^*} \frac{1}{2} f^{*T} P f^* + q^T f^* \\ &A f^* \leq u \end{aligned} \tag{7}$$

The inequality constraint mentioned in (6) has been reformulated as $Af^* \leq u$ with A and u described as below,

$$A = \begin{bmatrix} A_0 & 0_{9\times 8} \\ 0_{9\times 8} & A_0 \end{bmatrix}, \tag{8}$$

$$u = 0_{16\times 1}, \tag{9}$$

$$A_0 = \begin{bmatrix} 0 & 0 & -\mu & 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & -1 & 0 & 0 \\ -1 & 0 & 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 & -1 & 0 \\ 0 & 0 & -1 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}. \tag{10}$$

Other parameters needed for the optimization in (7) can be easily formulated as below,

$$P = \begin{bmatrix} G_1^T G_1 & 0_{4\times 4} & G_1^T G_2 & 0_{4\times 4} \\ 0_{4\times 4} & 0_{4\times 4} & 0_{4\times 4} & 0_{4\times 4} \\ G_2^T G_1 & 0_4 & G_2^T G_2 & 0_{4\times 4} \\ 0_{4\times 4} & 0_{4\times 4} & 0_{4\times 4} & 0_{4\times 4} \end{bmatrix}, \tag{11}$$

$$q = [-G_1^T w, 0_{1\times 4}, -G_2^T w, 0_{1\times 4}]^T. \tag{12}$$

2.4 Custom Metric and Parameter Sweep

For our custom metric, we implement the robust force closure. This is similar to the force closure metric, but we compute if the grasp is still in force closure if we add some noise to the parameters of the object. We add a small Gaussian noise to the vertex positions and both the coefficients of friction. We do this many times to give a numeric score (number of grasps in force closure / number of grasps tests) to a grasp, while the force closure could only give a binary output. This in particular allows us to better rank grasp qualities and we can now pick the best one, where we previously had to randomly pick one of the best n grasps. Our parameter sweep was performed by varying the amount of noise provided to each parameter and determining which combination gave the best results.

2.5 Grasp Execution

During the grasp execution, many factors need to be taken into account. The most significant issue determinant the success of grasp is the grasp approach direction. Each grasp attempt is a multi stage process consisting of moving to an intermediate point, moving to the grasp position, and picking up the object and moving to a new position. The first step of this was determining the approach direction. We determined this direction by drawing a ray from the midpoint of the two grasp contacts and the COM of the object. This direction will be used to determine the intermediate position, which is a way point away from the the object in the inverse direction of the approach direction. We also choose match this orientation to the orientation of the grasp. Next we move to our grasp position. We constrain the orientation the grippers can move in to the final orientation and move forward in the direction of the approach direction to the final grasp position.

We can now close the grippers and move the object to an arbitrary location.

3 Discussion of Results

3.1 Effectiveness of approach and grasps

We found that our definition of the grasp, in terms of the orientation of the grasp and the approach direction was very effective. Our definition of the approach direction, was a fairly good heuristic in bringing our arm to a point where it could make the grasp.

3.2 Observations

We noticed that successful grasps generally tend to occur on large faces, where the tolerance in positioning is harder. Although certain extrusions on the objects may seem like good grasping positions, they are actually difficult as Baxter often misses the desired point. A problem we encountered was also in planning and executing a path to grasp positions. The most reliable approaches were from the front of the object, with the gripper oriented down. We found that grasps with these approaches were more likely to succeed.

3.3 Comparing Objects

Overall we found that the pawn was the easiest object to grasp and the gearbox was the most difficult object. This can be attributed to the gearbox object having very varying thickness, and thus if the gripper did not reach the exact desired contact points, the object was often too slim for the gripper to grasp it. Furthermore, this object easily toggled over, if the gripper touched it on its way to the final position.

3.4 Comparing Grasp metrics

As expected, we found that the robust force closure metric gave the best grasp. This metric is able to rank the grasp whereas the to other metrics only returns binary values. For the robust force closure, we simulated 10 force closure grasp for each potential grasp. We believe that the performance of the method would increase, if we had simulated more force closure grasps. However, this metric was already quite computational expensive, and decided that 10 simulations was sufficient to differentiate the quality of the grasps.

We found that the gravity resistance metric was slightly better than the force closure metric, however, also somewhat slower to calculate. It would be interesting to combine the these two metrics in a robust force closure and robust gravity resistance metric in order to get even more grasps.

3.5 Parameter Sweep Results

We implemented our parameter sweep with Gaussian noise on both the friction coefficients and the vertices. In order to keep within a realistic amount of parameter combinations, we decided to use a global noise scale constant s that multiplied the γ, μ, and the position of the vertices p with respectively $\gamma + N(0, \gamma \cdot s)$, $\mu + N(0, \mu \cdot s)$, $p + N(0, 0.1 \cdot s)$, where $s \in \{0.5, 1, 1.5\}$. Note that the default values $\gamma = 0.5$ and $\mu = 0.1$ was kept. We chose to vary the position of the position with 10 cm as smaller values caused undefined behavior by the built-in function `intersects_location`. For future work, it would be interesting to make a grid search where the parameters are varied independently of the others. However, for the scope of this project, this was not feasible as the number of combinations has cubic growth.

3.6 Results

We scored our grasps on a qualitative scale ranging from 1 - 4, where a 1 indicated a completely failed grasp, and 4 was a success. A score of a 2 was given to an grasp if the manipulator closed around the target points, and a 3 was a partial lift. This can be differentiated from a success by slipping or falling on route to a new position. Results can be found in Figure 1 and 2.

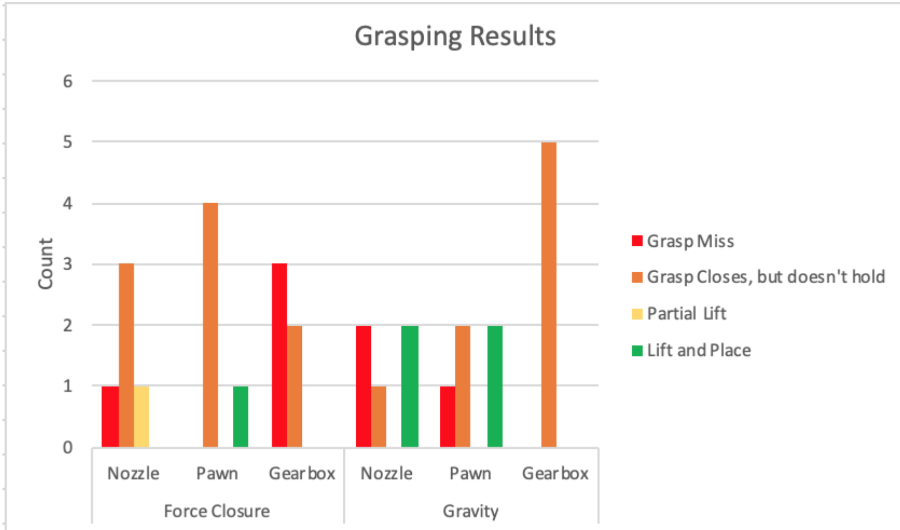

Figure 1: Grasp Metrics Results. The results about force closure and gravity resistance metric are compared. The grasp manipulation are evaluated with four different level: grasp miss, grasp closes but doesn't hold, partial lift and successful displacement, with four representation colors.

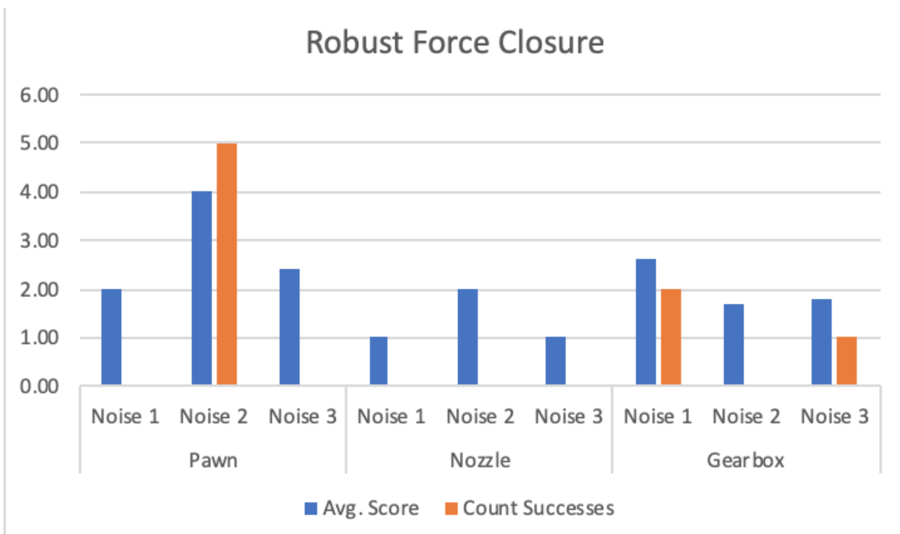

Figure 2: Robust Grasp metric Results. Experiments have been launched with pawn, nozzle and gear box respectively. Average score and number of successes are indicated for each case if available.

4 Conclusion

4.1 Proposed Extension

To improve the grasp performance effectively, we should fully use the probabilistic theory or learning theory. This can either be accomplished through a network training with prior data, or a direct Bayesian framework.

One possible method to improve the performance of grasp is to use a neural network called Dexterity Network [2], where a new dataset and associated algorithm to study the scaling effects of Big Data and Cloud Computation on robust grasp planning are exploited. This method was proved to have the computational efficiency (less numbers of iterations) and better normalized quality. Dex-Net 1.0 uses Multi-View Convolutional Neural Networks (MV-CNNs), a new deep learning method for 3D object classification and this method has been proved to benefit the quality and complexity of robust grasp planning, as we have used the prior data.

Without network, we can also apply some improved traditional method based on probabilistic theory. For example, a Bayesian framework for grasp planning is introduced in [1]. In this paper, the uncertainty in object shape or pose, as well as robot motion error have been taken into account. Baxter has quite an error about motion itself and object recognition from AR markers. However, the probabilistic framework takes into account all of these factors while trying to estimate the overall probability of success of each grasp, allowing us to select grasps that are robust to incorrect object recognition as well as motion error due to imperfect robot calibration. This method still need data-driven planning, but it will take less time than the Dex-Net.

4.2 Lab Documentation (Bonus)

4.2.1 Improved visualization tool

We found that by plotting few more things in the visualization tool we could get a better intuition of how the grasp would work. In particular, we added plotting of the base coordinate frame, the approach direction, and the desired orientation of the grasp. Please look at vis() in policies for this update.

4.2.2 Update on look at general

The original look at function tries to make the x axis the cross of z and up, however, we should actually make this the cross of the desired y axis and z. This was a simple modification of the look at general method, simplifying allowing for the specification of the up vector.

References

[1] K. Hsiao, M. Ciocarlie, P. Brook, and W. Garage. Bayesian grasp planning. In *ICRA 2011 Workshop on Mobile Manipulation: Integrating Perception and Manipulation*, 2011.

[2] J. Mahler, F. T. Pokorny, B. Hou, M. Roderick, M. Laskey, M. Aubry, K. Kohlhoff, T. Kröger, J. Kuffner, and K. Goldberg. Dex-net 1.0: A cloud-based network of 3d objects for robust grasp planning using a multi-armed bandit model with correlated rewards. In *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1957–1964. IEEE, 2016.