1 Motivations

1. Given an object $\mathcal{O}$, find a grasp g^* that maximizes a binary success metric subjected to uncertainty in object, environment, robot.

2. Leverage a large dataset of obj-grasp-grasp quality to reduce the number of grasp evaluations required to find the optimal grasp.

2 Algorithmic Descriptions

1 **Input:** Object $\mathcal{O}$, Number of Candidate Grasps K, Number of Nearest Neighbors M, Dex-Net 1.0 Dataset $\mathcal{D}$, Feature maps φ, Maximum Iterations T, Prior beta shape α_0, β_0, Lower Bound Confidence q, Quality Metric S
Result: Estimate of the grasp with highest P_F, $\hat{\mathbf{g}}^*$
`// Generate candidate grasps and priors`
2 Γ = AntipodalGraspSample($\mathcal{O}, K$) ;
3 $\mathcal{A}_0 = \varnothing, \mathcal{B}_0 = \varnothing$;
4 **for** $\mathbf{g}_k \in \Gamma$ **do**
    `// Equations VI.2 and VI.3`
5     $\alpha_{k,0}, \beta_{k,0}$ = ComputePriors($\mathcal{O}, \mathbf{g}_k, \mathcal{D}, M, \varphi, \alpha_0, \beta_0$);
6     $\mathcal{A}_0 = \mathcal{A}_0 \cup \{\alpha_{k,0}\}, \mathcal{B}_0 = \mathcal{B}_0 \cup \{\beta_{k,0}\}$;
7 **end**
`// Run MAB to Evaluate Grasps`
8 **for** $t = 1, .., T$ **do**
9   j = ThompsonSample($\mathcal{A}_{t-1}, \mathcal{B}_{t-1}$);
10   S_j = SampleQuality($\mathbf{g}_j, \mathcal{O}, S$);
    `// Equations VI.4 and VI.5`
11     $\mathcal{A}_t, \mathcal{B}_t$ = UpdateBeta(j, S_j, Γ);
12     $\mathbf{g}_t^*$ =MaxLowerConfidence($q, \mathcal{A}_t, \mathcal{B}_t$);
13 **end**
14 return $\mathbf{g}_T^*$;
Dex-Net 1.0 Algorithm: Robust Grasp Planning Using Multi-Armed Bandits with Correlated Rewards

2.1 Line 2: AntipodalGraspSample

1. Def of antipodality from wiki: the antipodal point of a point on the surface of a sphere is the point which is diametrically opposite to it - so situated that a line drawn from one to the other passes through the center and forms a valid diameter.

2. A special algorithm is used to generate antipodal grasp.

2.2 Line 5: ComputePriors

1. After generating the candidate grasps, the goal is to find one grasp among the candidate grasp that maximizes the expected value of the success metric.

2. The expected value of the success metric can be written as the probability of success $P_S(\mathbf{g}) = \mathbb{E}[S(\mathbf{g})]$, which is modeled as a Bernoulli random variable.

3. They use a Continuous Correlated Beta Processes (CCBP) to model the joint distribution of the successes of all candidate grasps.

4. The CCBP allows for closed form update to obtain the posterior given data.

5. To compute the posterior update, they use a squared exponential as the similarity metric between object-grasp pair $\mathcal{P} = (\mathbf{g}, \mathcal{O})$

$$k\left(\mathcal{Y}_p, y_q\right) = \exp\left(-\frac{1}{2}\sum_{m=1}^{3}\left\|\varphi_m\left(\mathcal{P}_p\right) - \varphi_m\left(\mathcal{P}_q\right)\right\|_{C_m}^2\right)$$

6. The formulae to obtain the posterior is given in the paper

2.3 Line 9: Thompson Sampling

1. At each iteration, they need to pick a grasp to evaluate.
2. Let $\theta_j = P_S(\mathbf{g}_j)$, for each candiate grasp, they draw $\hat{\theta}_\ell \sim p(\theta_\ell | \alpha_{\ell,t}, \beta_{\ell,t})$ and pick the candidate grasp with the highest sampled value.

2.4 Line 10: SampleQuality

1. Given a success metric, an obj $\mathcal{O}$, grasp g, they need to evaluate the corresponding value of the success metric.
2. The success metric used here is force closure, which is the ability to resist force and torque in arbitrary directions.
3. The formula to compute the success metric is given in section 3, C and E.

2.5 Line 11; UpdateBeta

1. The posterior of the CCBP is updated following evaluation of the grasp.

3 Other interesting technical bits

3.1 Grasp parameterization

1. The exact obj shape is assumed to be given in the form of a signed distance function $f : \mathbb{R}^3 \to \mathbb{R}$, which is 0 on the obj surface, positive outside and negative inside.
2. A grasp is parameterized by $(\mathbf{x}, \mathbf{v})$ where $\mathbf{x}$ is the centroid of the jaw in 3D space $\mathbf{x} \in \mathbb{R}^3$ and $\mathbf{v}$ is an approach angle $\mathbf{v} \in \mathbb{S}^2$
3. They denote an obj by $\mathcal{O} = (f, \mathbf{z})$ with SDF f and center of mass $\mathbf{z}$.

3.2 Multi-view CNN for obj similarity comparison

1. They obtain different views of an obj from different viewing angle and distance.
2. Each view is passed through a CNN.
3. The output of the CNN for the different views are aggregated and max-pooled, then projected into a lower dimensional space.
4. The distance between objects in this lower dimensional space is used to the similarity metric to compare 2 objects.

4 Questions

1. They show how to evaluate force closure for one setting of environment and robot parameter in section 3, but how do they compute its expectation? With Monte Carlo integration?
2. Definition of interpretation of: soft finger contact model, contact wrenches, surface normal, friction cone,
3. What is the interpretation of the math in the UpdateBeta step?
4. How is max lower confidence computed?
5. What is the number of candidate grasp they use in the algorithm?