Grasping

So far…

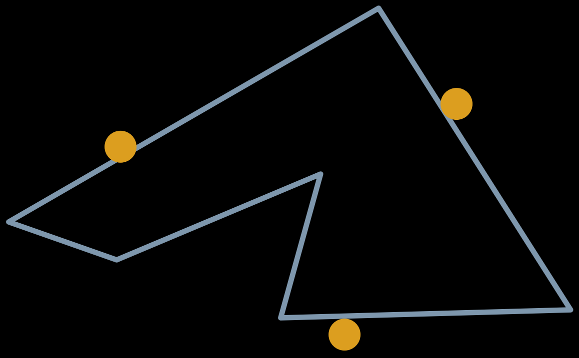
- We've talked about how to move robots so they don't collide
- But how do we get robots to move objects in the world?
  - *Grasping* studies how to stably make contact with objects and move them

- Now we want to make contact with objects
- But how do we know if a given grasp is *stable* or not?

Outline

- Definitions
- Form Closure
- Force Closure
- Searching for Grasps

Start Simple



Real World
- Complex mechanism
- Soft contacts
- Soft objects
- Bounded force
- Object is free-floating

Simplified Problem
- Ignore hand mechanism
- Assume *n* point contacts
- Assume rigid object
- Assume unlimited force
- Assume object is fixed

Definitions

- A point contact is sometimes called a *finger*
- A *wrench* is a combination of the force and torque applied to the object

$$\boldsymbol{g} = [\boldsymbol{f}^\top \boldsymbol{m}^\top]^\top$$

- *Wrench space* is space of wrenches applied to the object
  - 2D object: 3 dimensional wrench space (2 force, 1 torque)
  - 3D object: 6 dimensional wrench space (3 force, 3 torque)

- A grasp *immobilizes* an object if it can counter any wrench applied to the object
  - Immobilizing an object guarantees the stability of the grasp

Definitions

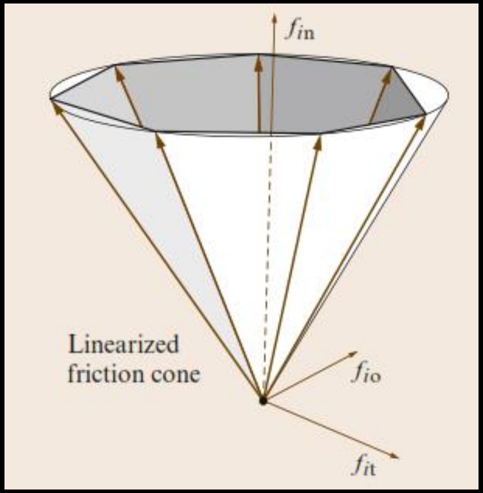
- A *friction cone* is the set of forces that can be applied at a contact point without sliding on the object
- Friction cone for the ith contact point is the set:

$$\mathcal{F}_i = \left\{ (f_{in}, f_{it}, f_{io}) \middle| \sqrt{f_{it}^2 + f_{io}^2} \le \mu_i f_{in} \right\}$$

f_{in} is the force applied normal to the surface

f_{io} and f_{it} are the forces applied along the surface

- Assume Coulomb friction
- Depends on coefficient of friction between hand and object (μ)
- Bigger μ implies wider friction cone

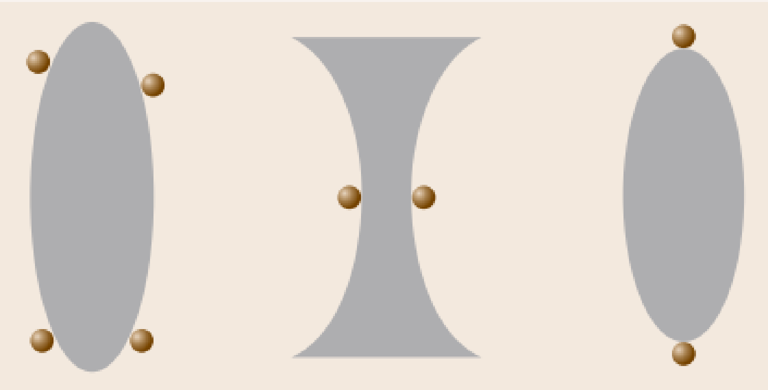


Computationally, it's easier to represent the friction cone with a discrete set of vectors on its boundary

Form Closure

- A *form closure* grasp is when the object cannot move **regardless of surface friction**

- Which of these is in form closure?



Form Closure

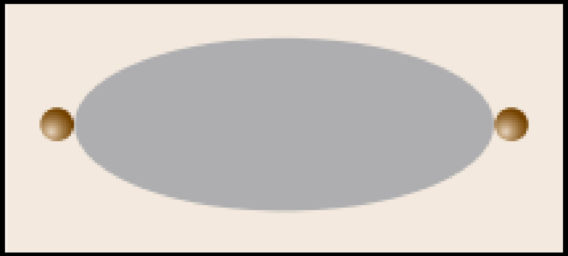
- You need *at least* N+1 contacts to achieve first-order form closure, where N is the number of DOF of the object

[K. Lakshminarayana: Mechanics of form closure, Amer. Soc. Mech. Eng. Tech. Rep. **78-DET-32** (1978)]

Dimension of Object	Minimum Number of Contacts for First-Order Form Closure
2D (3 DOF)	4
3D (6 DOF)	7

Force Closure

- Frictional properties of the object can be used to immobilize it
- Stability of a grasp depends on friction (μ) between contacts and object:



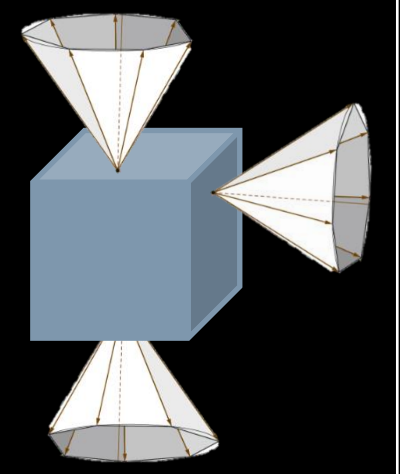
- If a grasp achieves *form* closure, does it also achieve *force* closure?

Testing for Force Closure

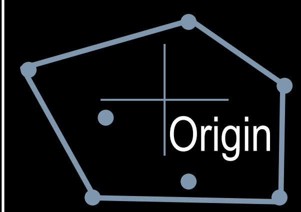
- Many algorithms exist to test for force closure, here is one:

Input: Contact locations
Output: Is the grasp in Force-Closure? (Yes or No)

1. Approximate the friction cone at each contact with a set of wrenches
2. Combine wrenches from all cones into a set of points *S* in wrench space
3. Compute the *convex hull* of *S*
4. If the origin is inside the convex hull, return YES. If not, return NO.



Friction cones at contacts



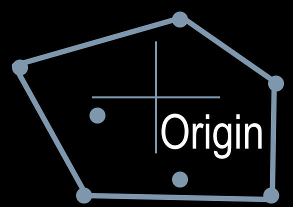
Wrench space (6 dimensional)

Testing for Force Closure

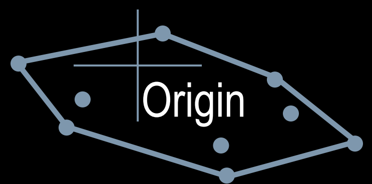
- Why does this algorithm work?
  - Hint: The convex hull represents the positive linear combination of all the wrenches you can apply at the given contact points

Force Closure

- Which grasp do you think is more sensitive to error in contact position?



Wrench space



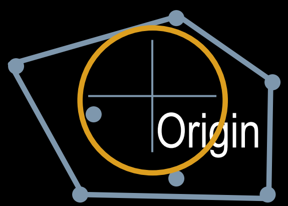
Wrench space

Note: wrench space is 6-dimensional, these are only cartoons

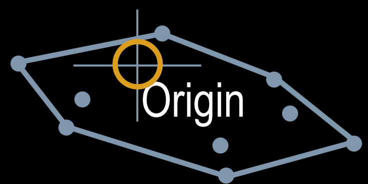
- Yes-or-no answer isn't enough to choose between grasps

Force Closure Metrics

- A popular metric: Radius of largest hyper-sphere you can fit in convex hull (centered at origin)

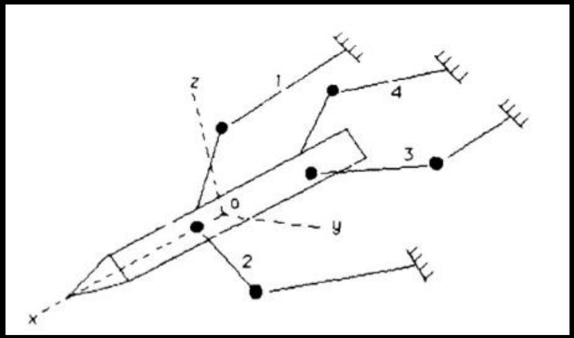


Wrench space



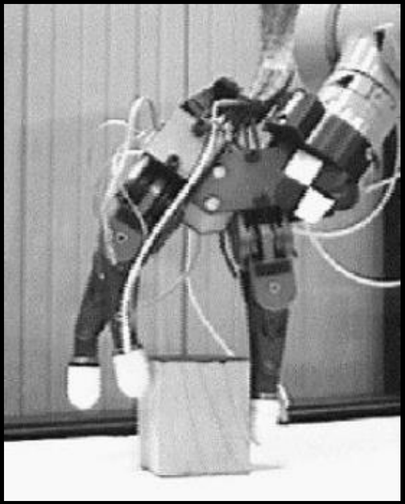
Wrench space

- Task-specific metric of Li and Sastry
  - Use task-specific ellipsoid instead of hyper-sphere
  - Does not make the infinite-force assumption



Force Closure

- For a 3D object, minimum number of contacts to achieve force closure is 3 (compare to 7 for form closure)
- Not surprisingly, 3-finger grippers are very popular



Stanford/JPL Hand

Barrett Hand

Robotiq Hand

Schunk SDH Hand

Break

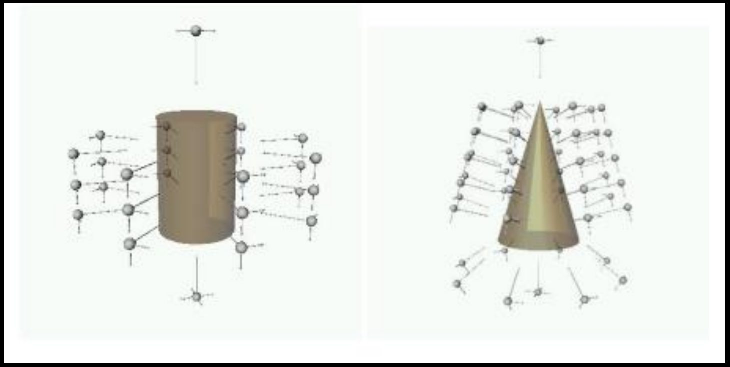
Searching for Force Closure Grasps

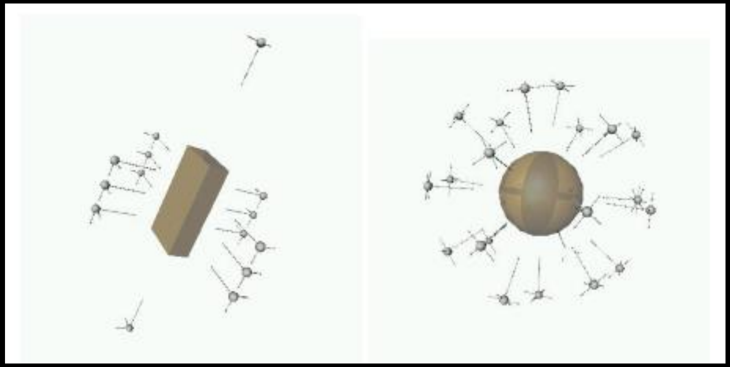
- In the 90s: Search for a set of *n* point contacts on the surface of an object, where *n* is the number of fingers of your hand
- Search is in 2*n* dimensional space (since surface of object is 2-dimensional)
- Disadvantage
  - Ignores hand kinematics: probability that these contacts are reachable while obeying hand kinematics is low
  - Search space scales poorly with number of fingers

Searching for Force Closure Grasps

- In the 2000s (Peter Allen et al.):
    1. Sample pose of hand relative to object with fingers in a *pre-shape*
    2. Approach object until contact and close the fingers
    3. Get contact points between fingers and object
    4. Test these contact points for force closure

- Search space is only 6-dimensional (pose of hand) + set of pre-shapes
- Search can be arranged so hand always approaches parallel to surface of object

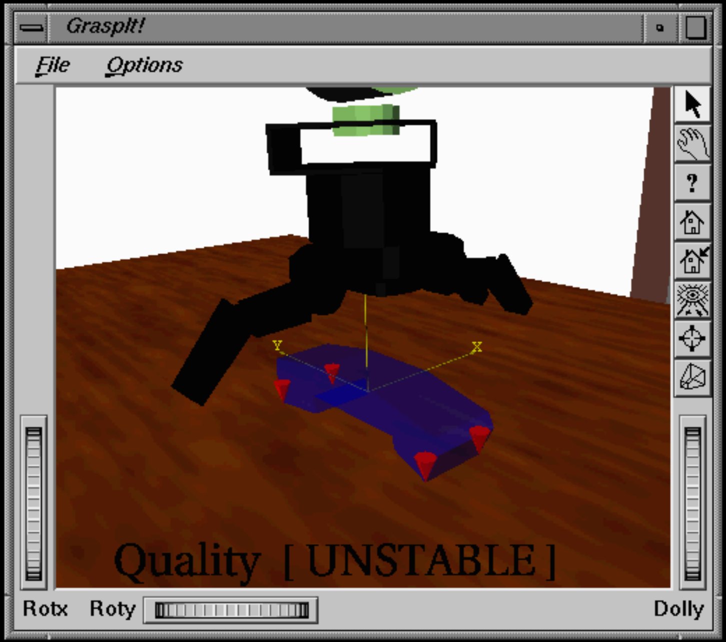




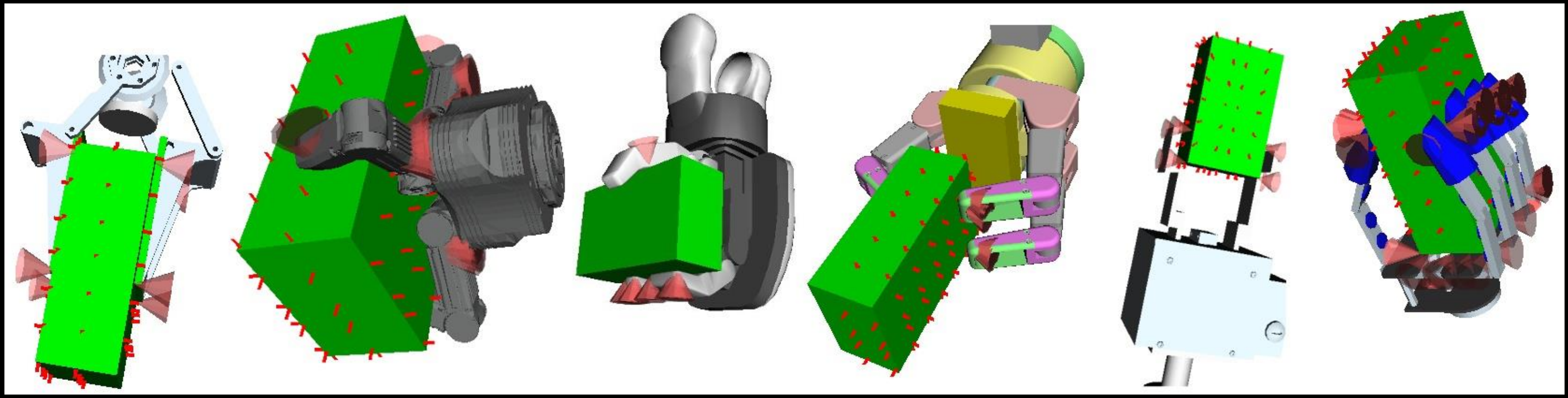
Pre-computing grasp sets

- Searching for grasps is slow!
  - Especially with dynamics

    (i.e. if you don't assume object is fixed)



- But, we can **pre-compute** a set of stable grasps for a given object



Pre-computing grasp sets is not new!

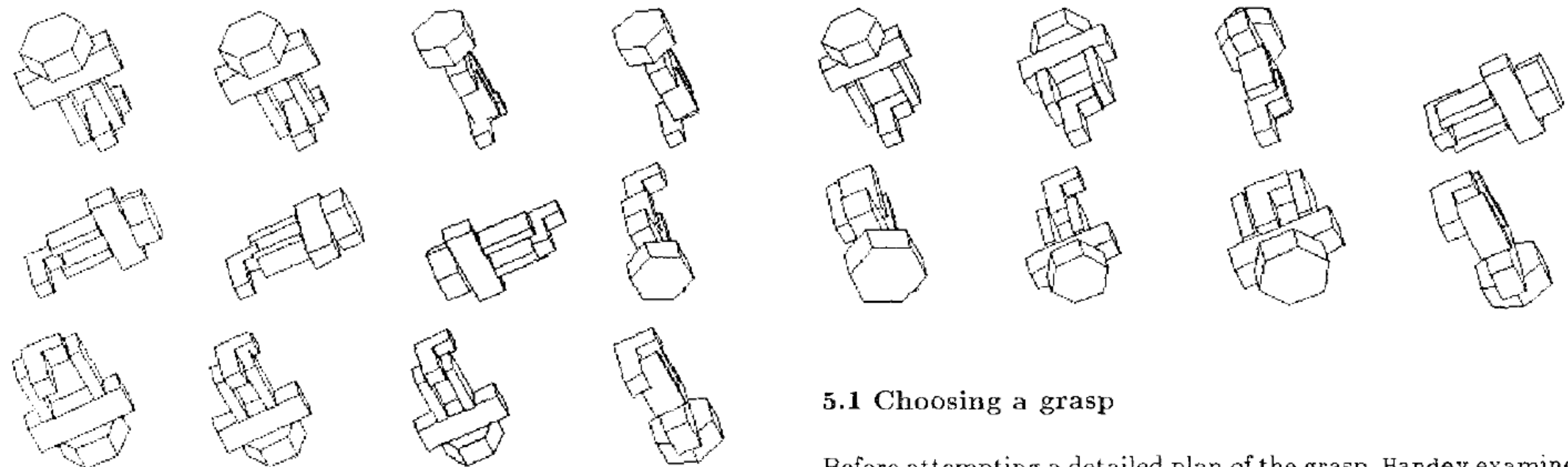


Figure 6. The different groups of approach directions and grasp classes for a particular orientation of an L-shaped object, heuristically ranked by desirability.

5.1 Choosing a grasp

Before attempting a detailed plan of the grasp, Handey examines different classes of candidate grasps and evaluates their feasibility both at the pickup point and the putdown point. A grasp class is characterized by a choice of object surfaces. Within a

[Handey: A robot system that recognizes, plans, and manipulates, Lozano-Perez, T., Jones, J., Mazer, E.. O'Donnell, P., Grimson, W., Tournassoud, P., Lanusse, A., ICRA 1987]

Columbia Grasp Database
http://grasping.cs.columbia.edu/

- We reused the 3D models from the Princeton Shape Benchmark*
  - Well known academic dataset of 1,814 models
  - All models resized to "graspable" sizes
- We provide grasps at 4 scales
  - …because grasping is scale dependent
  - .75, 1.0, 1.25 and 1.5 times the size of each model
  - 7,256 3D models in all

*Shilane *et al.*, SMI 2004

The Columbia Grasp Database. Goldfeder, Ciocarlie, Dang, and Allen, ICRA 2009

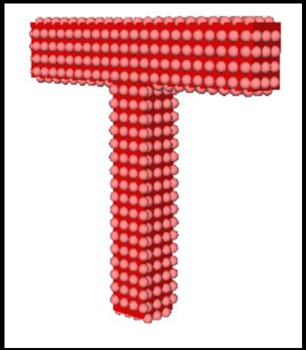
Integrating Grasping and Motion Planning

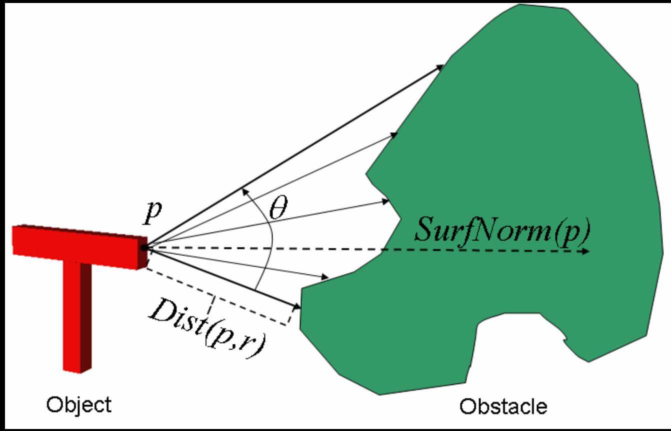
- So far, we only test for collision with obstacles online (ignore them when computing grasp set)
- We wanted to integrate grasp planning and motion planning (consider obstacles and reachability, too)
- Approach:
  1. Pre-compute grasp set offline, get force-closure score
  2. Online, compute 2 scores for each grasp
     - Environment Clearance Score
     - Reachability Score

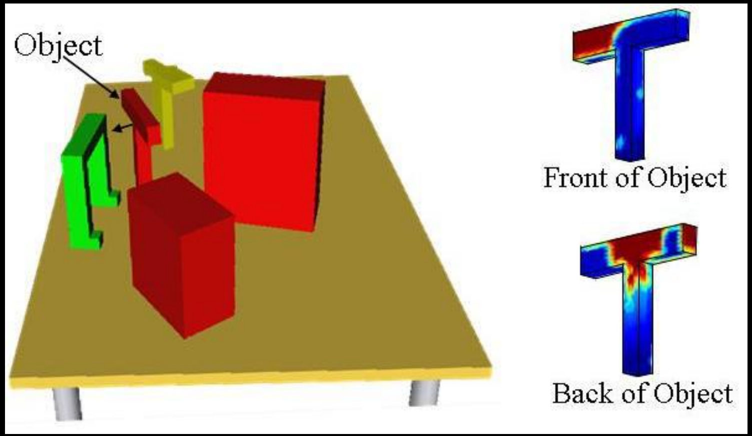
[Berenson, D., Diankov, R., Nishiwaki, K., Kagami, S., & Kuffner, J. (2007). Grasp Planning in Complex Scenes. *IEEE-RAS International Conference on Humanoid Robots (Humanoids07)*]

Computing Environment Clearance Score

2 (more detail). Compute clearance from points on object to nearest obstacle

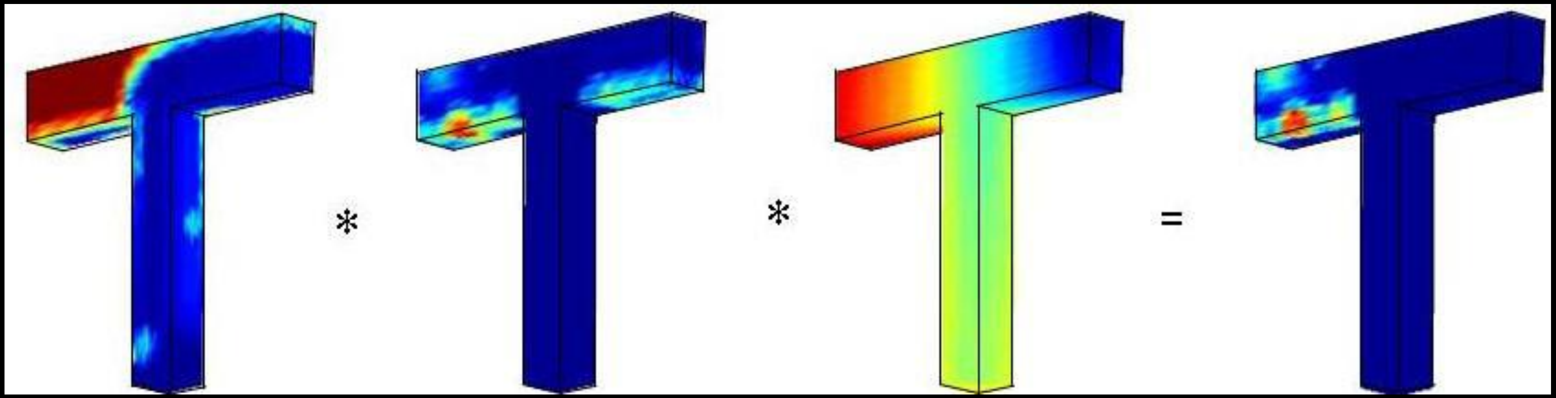






Integrating Grasping and Motion Planning

3. Combine scores to create grasp ranking



Environment Clearance | Grasp Quality | Reachability | Total Score

4. Test grasps in order of ranking
    - We showed this is much faster than testing in random order

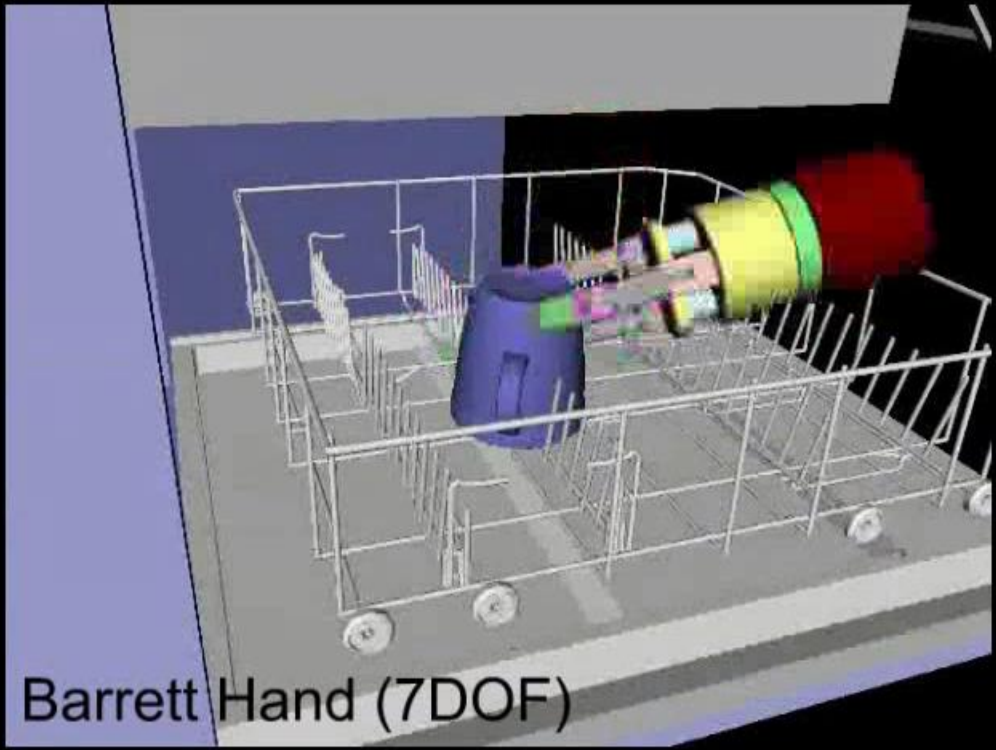
Grasp Planning in Complex Scenes

Dmitry Berenson
Rosen Diankov
Koichi Nishiwaki
Satoshi Kagami
James Kuffner

Carnegie Mellon Robotics Institute — Pittsburgh, PA, USA
Digital Human Research Center (AIST) — Tokyo, Japan

Integrating Grasping and Motion Planning

- But this method is still limited to a fixed set of grasps
- Next, we tried searching for grasps online using similar scoring
  - Search was based on a genetic algorithm

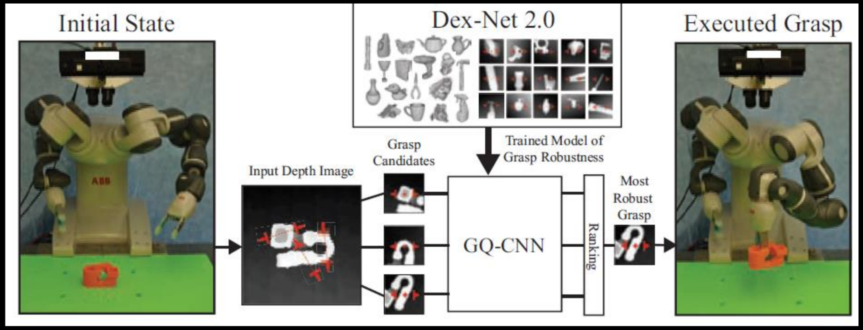


[Grasp Synthesis in Cluttered Environments for Dexterous Hands. Berenson and Srinivasa, Humanoids 2008]

Recent work in grasping

- Deep Learning methods have taken over the grasping field
- General Idea:
  1. Generate many grasp candidates
  2. Learn a quality metric that uses the point cloud data directly
  3. Output highest quality grasp

Grasp Pose Detection
[ten Pas and Platt, 2015]



DexNet 2.0
[Mahler et al. 2017]

Summary

- Grasping is the study of how to immobilize objects through contact
- We make simplifying assumptions so we can compute form/force closure for grasps
- Force closure can be evaluated by checking if the origin is within the convex hull of the contact wrenches
- Much recent work has focused on how to search for grasps
  - Search for points on object surface
  - Search in the space of hand pose and pre-shapes
- Our work sought to integrate motion planning and grasping more closely by considering collision and reachability in grasp planning
- Recent work uses deep learning to estimate grasp quality

Recent Work: Fast Grasp Detection

- "Volumetric Grasping Network: Real-time 6 DOF Grasp Detection in Clutter," Michel Breyer, Jen Jen Chung, Lionel Ott, Roland Siegwart, Juan Nieto, Conference on Robot Learning (CoRL), 2020

- https://www.youtube.com/watch?v=BFjsTX3vEH8

Recent Work: Task-Oriented Grasping

- "Same Object, Different Grasps: Data and Semantic Knowledge for Task-Oriented Grasping," Adithyavairavan Murali, Weiyu Liu, Kenneth Marino, Sonia Chernova, Abhinav Gupta, Conference on Robot Learning (CoRL), 2020
- https://www.youtube.com/watch?v=eV-KyT6OK14

Homework

- Read PCA tutorial
- HW3 due on Monday!