Grasping

Jane Li

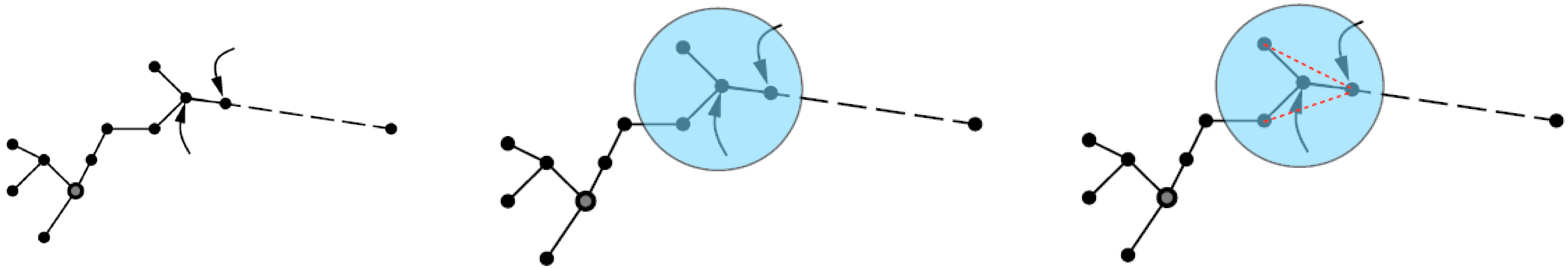
Assistant Professor
Mechanical Engineering Department, Robotic Engineering Program
Worcester Polytechnic Institute

Quiz (10 pts)

- (6 pts) What is the difference between RRG, RRT* and informed RRT
- (4 pts) How to use inverse optimal control to transfer human motion skills to humanoid robots? You may draw a block diagram to show the pipeline.

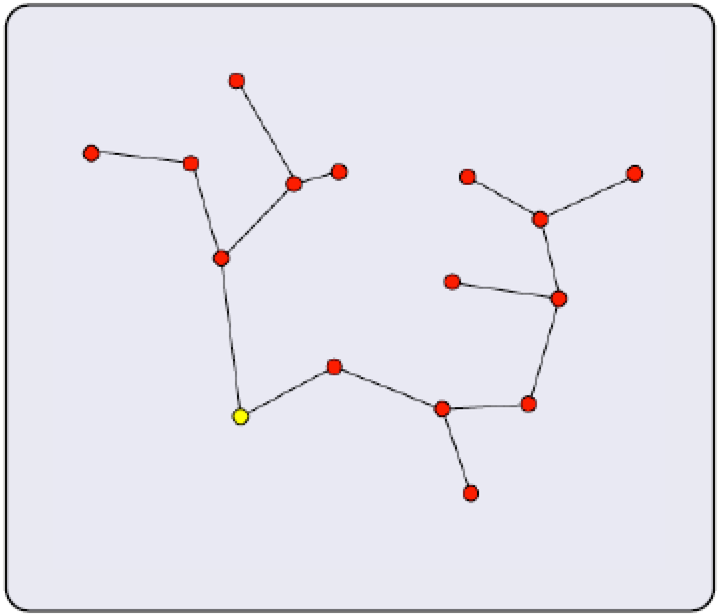
RRG

- RRT
  - Extends the nearest vertex towards the sample
- RRG
  - Extends all vertices returned by the Near procedure (if first was success).



RRT*

- Similar to RRG, except for "**rewiring**" the tree as better paths are discovered.
- After rewiring the **cost has to be propagated** along the leaves

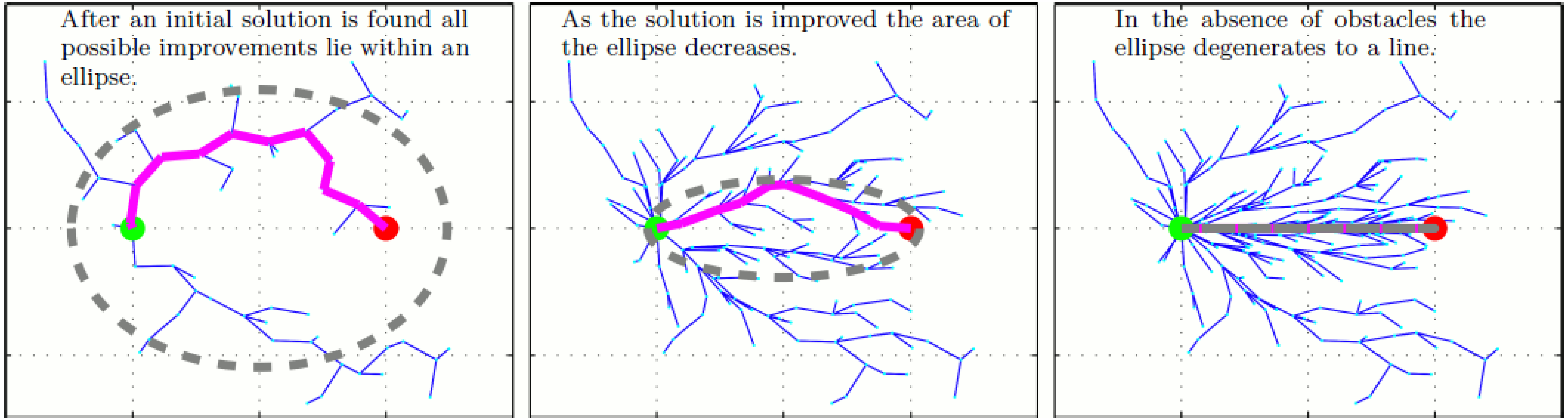


Limitation of RRT*

- RRT* is asymptotically optimal everywhere
- Not necessary for single-query planning
- Improvement
  - Limit the search to the **sub-problem** that would have a better solution
  - How to define the space of sub-problem?

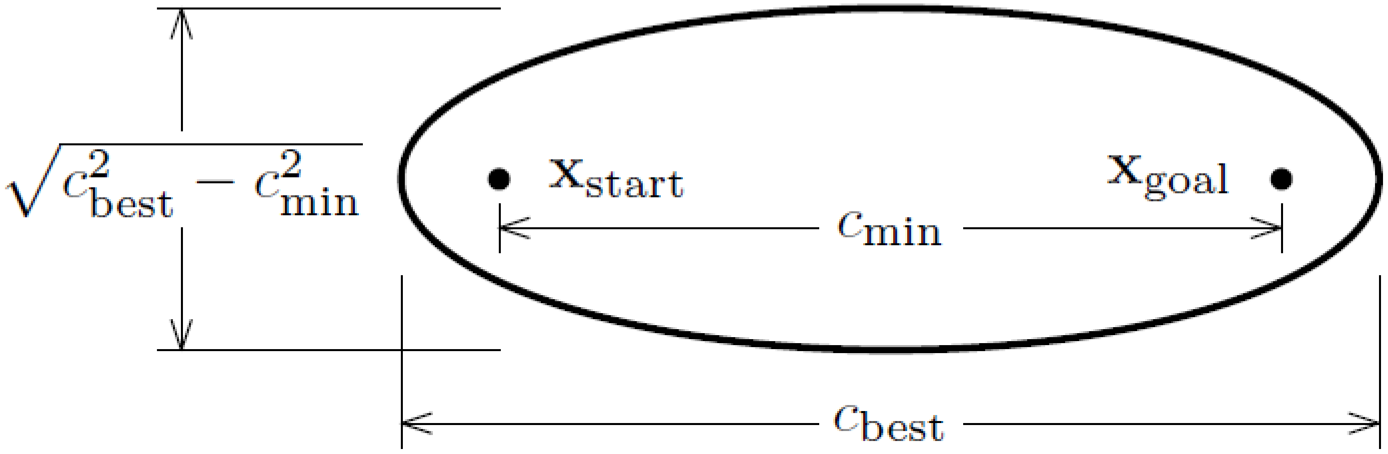
Informed RRT

- The sub-problem can be defined as “search in a n-dimensional ellipse”→ where to draw the new sample



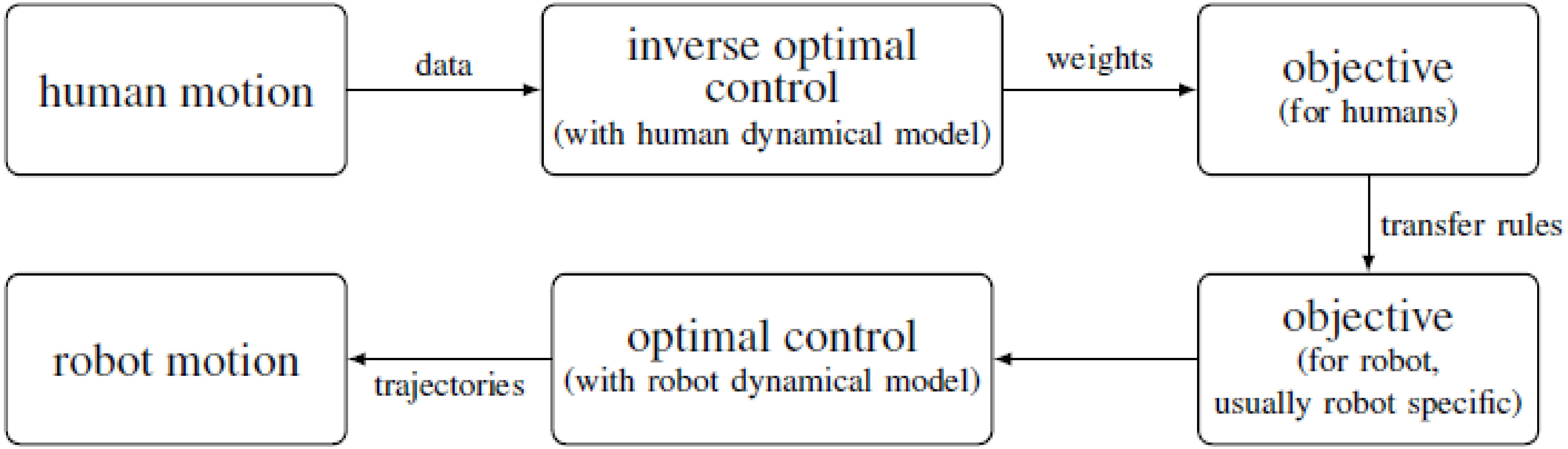
59 iterations, $c_{\text{best}} = 148.24$ | 175 iterations, $c_{\text{best}} = 107.12$ | 1142 iterations, $c_{\text{best}} = 100$

heuristic sampling domain



$$X_{\hat{f}} = \left\{ \mathbf{x} \in X \;\middle|\; ||\mathbf{x}_{start} - \mathbf{x}||_2 + ||\mathbf{x} - \mathbf{x}_{goal}||_2 \leq c_{best} \right\}$$

Inverse optimal control



Optimality criteria

- Actuation and energy consumption
  - Minimize actuation in the stance foot, swing foot, torque, hip torque of the swing foot, angular momentum in x and y direction, vertical center of mass oscillations, absolute swing foot velocity
- Motion fitting error
  - Minimize planar distance between foot position at touch down and capture point, periodicity gap in center of mass velocities
- Others
  - Minimize overall single support duration, absolute swing foot velocity at touch down

Logistics

- Upload student talk ppt
  - TA will create a submission post on Canvas
  - Count toward your extra credit, if you haven't used it
- Last lecture – May 1st
  - Course review
  - Course evaluation

Optional assignment

- Wednesday, April 4
  - Samruddhi Kadam spkadam@wpi.edu
- Friday, April 6
  - Nalin Raut nraut@wpi.edu
  - Abhilasha Rathod arathod@wpi.edu
  - Nathaniel Goldfarb
- Wednesday, April 11
  - Max Merlin – lecture with Gunnar on high-level motion planning
  - Guled Elmi ggelmi@wpi.edu
  - Gaurav Vikhe gsvikhe@wpi.edu

Literature review student talk

- 4/13/2018
  - Bimanual team, Mobile team, Swarm team

- 4/18/2018
  - High-level planning, pHRI, VR motion planning, grasping

Project presentation

- 4/25/2018
  - Mobile team, Bimannual team, High-level planning
  - Surgical robot (Sam)
- 4/27/2018
  - pHRI team, Swarm robot, grasping, VR motion planning

Grasping

Grasping

- ***Grasping*** studies how to <u>stably make contact</u> with objects and move them
- We've talked about how to move robots so they don't collide
- Now we want to collide! (i.e. make contact with objects)

Overview

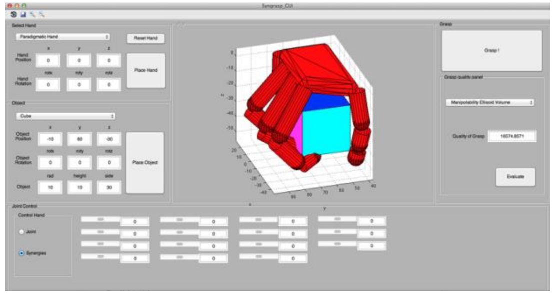
- Model & Definitions
- Form Closure
- Force Closure
- Grasp planning

Towards Dexterous Manipulation

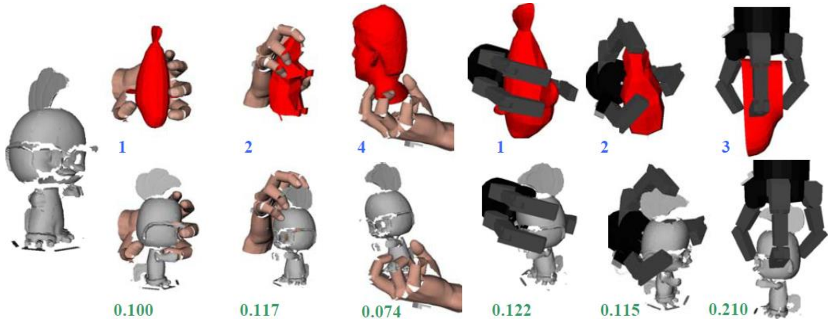
- First robotic hand for dexterous manipulation

- Software for grasp modeling & analysis

- Models for several robot hand
- Tools for grasp selection



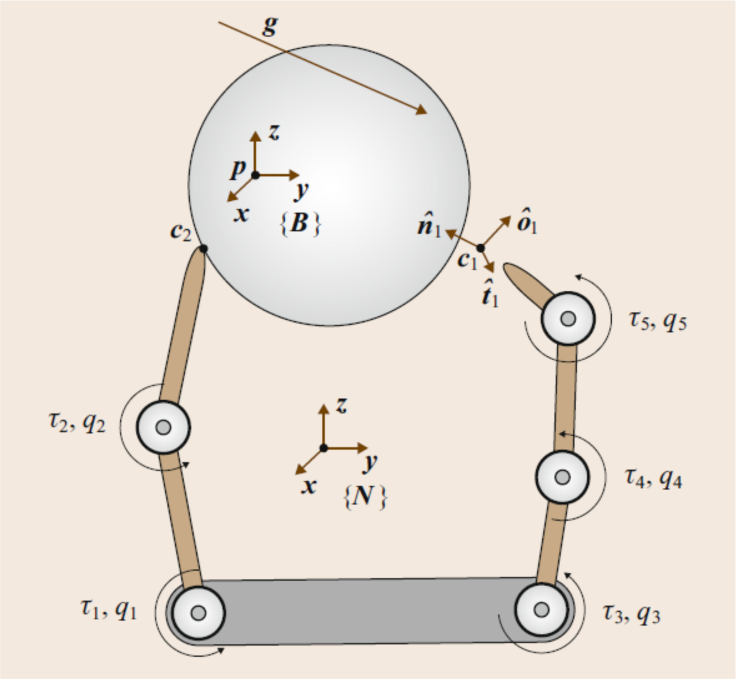
- Matlab toolbox
- Grasp analysis with fully/under-actuated hands



- Columbia Grasp Database
- Precomputed grasps on thousands of 3D models

Mathematical Model

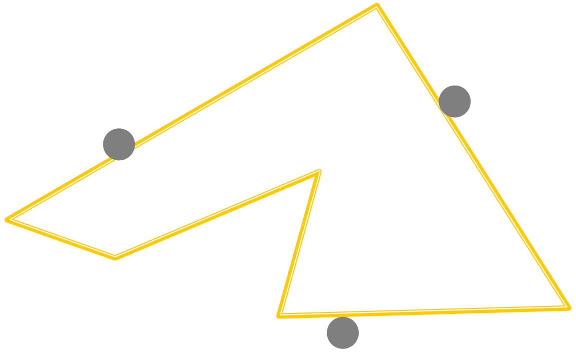
- Predict the behavior of the hand and object under <u>various loading conditions</u> that may arise during grasping
- Disturbance
  - Inertia force – e.g. fast motion
  - Applied force – e.g. Gravity
- Grasp maintenance
  - No contact separation
  - No unwanted contact sliding
- Closure grasp
  - The special class of grasps that can be maintained for every possible disturbing load



Model Simplification

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

Definition

- Finger – A point contact
- Twist
  - A combination of translational and rotational velocity of the object

$$\boldsymbol{\nu} = [\boldsymbol{v}^{\mathrm{T}}\ \boldsymbol{\omega}^{\mathrm{T}}]^{\mathrm{T}}$$

- Wrench
  - A combination of the force and torque applied to the object (at object origin)

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

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

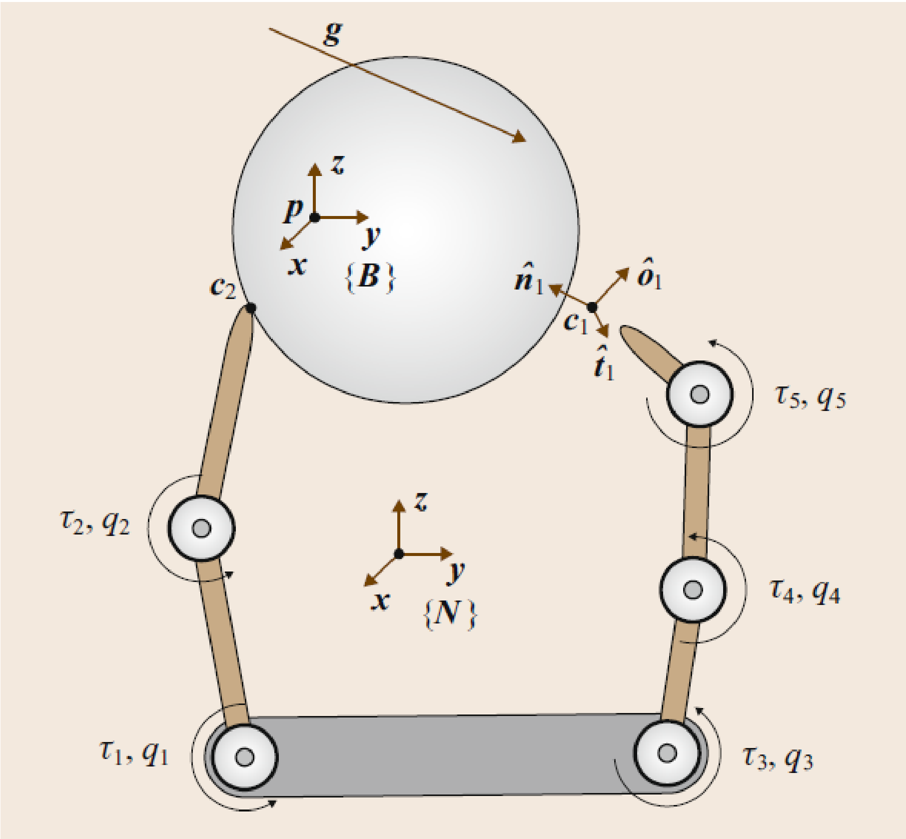
Grasp Kinematics

- Partial Grasp Matrix
  - Object twist in world frame {**N**} → Object twist in the contact frame {**C**}

$$\nu_{i,\text{obj}} = \tilde{\mathbf{G}}_i^{\mathrm{T}} \nu$$

  - where

$$\tilde{\mathbf{G}}_i^{\mathrm{T}} = \overline{\mathbf{R}}_i^{\mathrm{T}} \mathbf{P}_i^{\mathrm{T}}$$



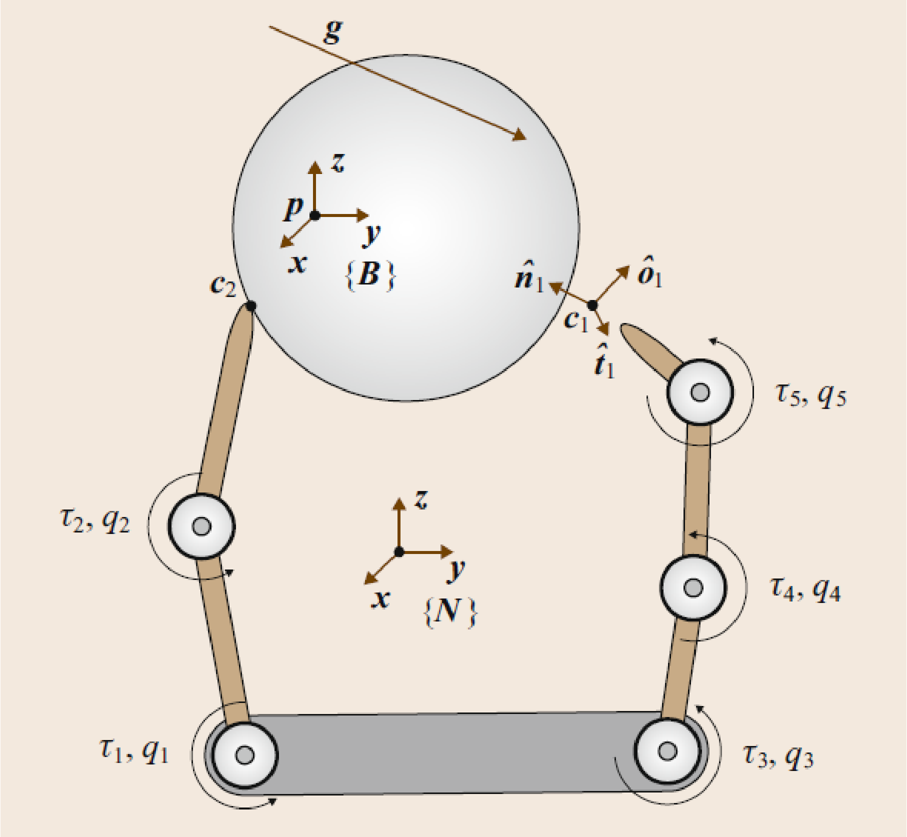
Grasp Kinematics

- Partial Hand Jacobian
  - Map joint velocities of hand → twist of the hand in {**N**} → twist of hand in {**C**}

$$\boldsymbol{\nu}_{i,\text{hnd}} = \tilde{\mathbf{J}}_i \dot{\boldsymbol{q}}$$

  - where

$$\tilde{\mathbf{J}}_i = \overline{\mathbf{R}}_i^{\mathrm{T}} \mathbf{Z}_i$$

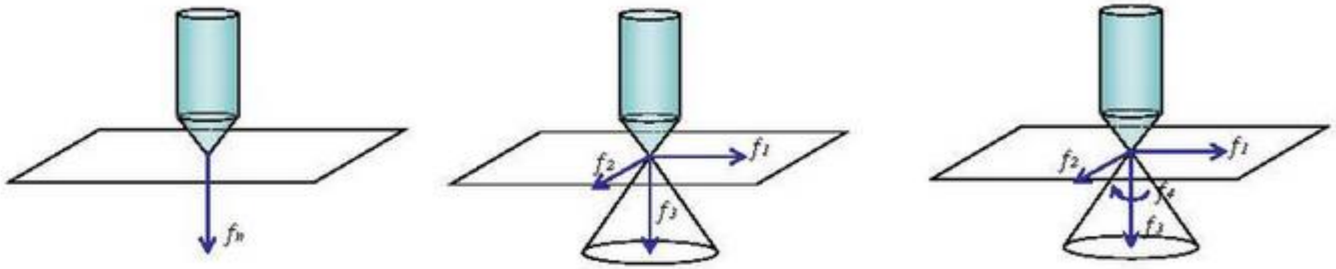


Definition

- Kinematics
  - $$v_{c,\mathrm{obj}} = \tilde{\mathbf{G}}^{\mathrm{T}} \boldsymbol{\nu} , \quad v_{c,\mathrm{hnd}} = \tilde{\mathbf{J}} \dot{q} ,$$ where $$\tilde{\mathbf{G}}^{\mathrm{T}} = \begin{pmatrix} \tilde{\mathbf{G}}_1^{\mathrm{T}} \\ \vdots \\ \tilde{\mathbf{G}}_{n_c}^{\mathrm{T}} \end{pmatrix}, \quad \tilde{\mathbf{J}} = \begin{pmatrix} \tilde{\mathbf{J}}_1 \\ \vdots \\ \tilde{\mathbf{J}}_{n_c} \end{pmatrix}$$
- Contact
  - Two coincident points – one on the hand, one on the object
- Immobilization
  - A grasp can counter any wrench applied to the object
  - Guarantees the stability of the grasp

Contact Modeling

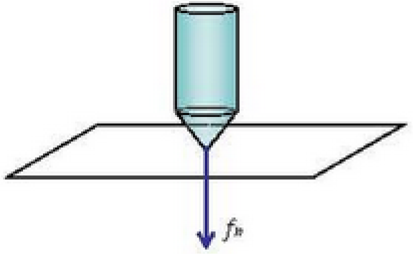
- Point contact without friction
- Hard-finger
- Soft-finger



Contact Modeling

- Point contact without friction (PwoF)
  - Contact properties
    - Contact patch is small
    - Contact surface is slippery → no surface friction

  - Transmitted to the object
    - Normal component of the translational velocity
    - Normal component of the contact force

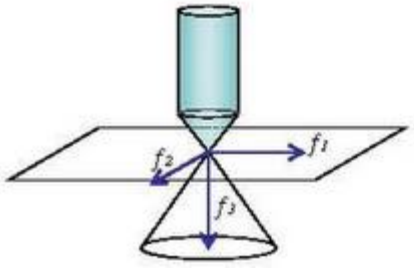


Contact Modeling

- Hard Finger (HF)
  - Contact properties
    - Small contact patch
    - Large enough surface friction

Friction force, but no appreciable friction moment

  - Transmitted to the object
    - All three components of the translational velocity
    - All three components of the contact force
    - No angular velocity or moment

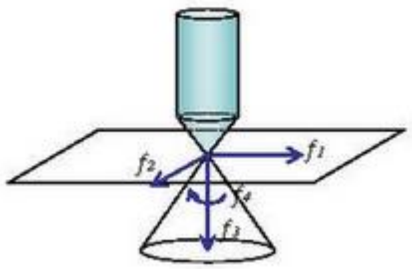


Contact Modeling

- Soft Finger (SF)
  - Contact properties
    - Large enough contact patch
    - Large enough surface friction

appreciable friction moment

  - Transmitted to the object
    - All three components of the translational velocity
    - All three components of the contact force
    - Normal component of contact moment



Contact Modeling

- Relative twist at each contact point

$$\begin{pmatrix}\tilde{\mathbf{J}}_i & -\tilde{\mathbf{G}}_i^{\mathrm{T}}\end{pmatrix}\begin{pmatrix}\dot{q} \\ \boldsymbol{\nu}\end{pmatrix} = \boldsymbol{\nu}_{i,\mathrm{hnd}} - \boldsymbol{\nu}_{i,\mathrm{obj}}$$

- When object is stably grasped

$$\mathbf{H}_i(\boldsymbol{\nu}_{i,\mathrm{hnd}} - \boldsymbol{\nu}_{i,\mathrm{obj}}) = \mathbf{0}.$$

- Kinematic contact constraint equation

$$\mathbf{H}(\boldsymbol{\nu}_{c,\mathrm{hnd}} - \boldsymbol{\nu}_{c,\mathrm{obj}}) = \mathbf{0}$$

which is

$$\begin{pmatrix}\mathbf{J} & -\mathbf{G}^{\mathrm{T}}\end{pmatrix}\begin{pmatrix}\dot{q} \\ \boldsymbol{\nu}\end{pmatrix} = \mathbf{0}$$

where

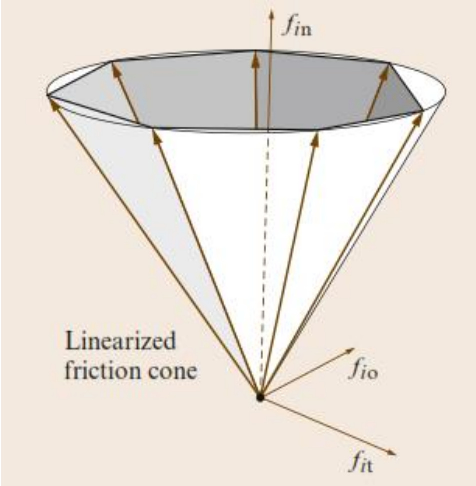
$$\mathbf{G}^{\mathrm{T}} = \mathbf{H}\tilde{\mathbf{G}}^{\mathrm{T}} \in \mathbb{R}^{n_\lambda \times 6}$$
$$\mathbf{J} = \mathbf{H}\tilde{\mathbf{J}} \in \mathbb{R}^{n_\lambda \times n_q}.$$

Contact Modeling

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

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

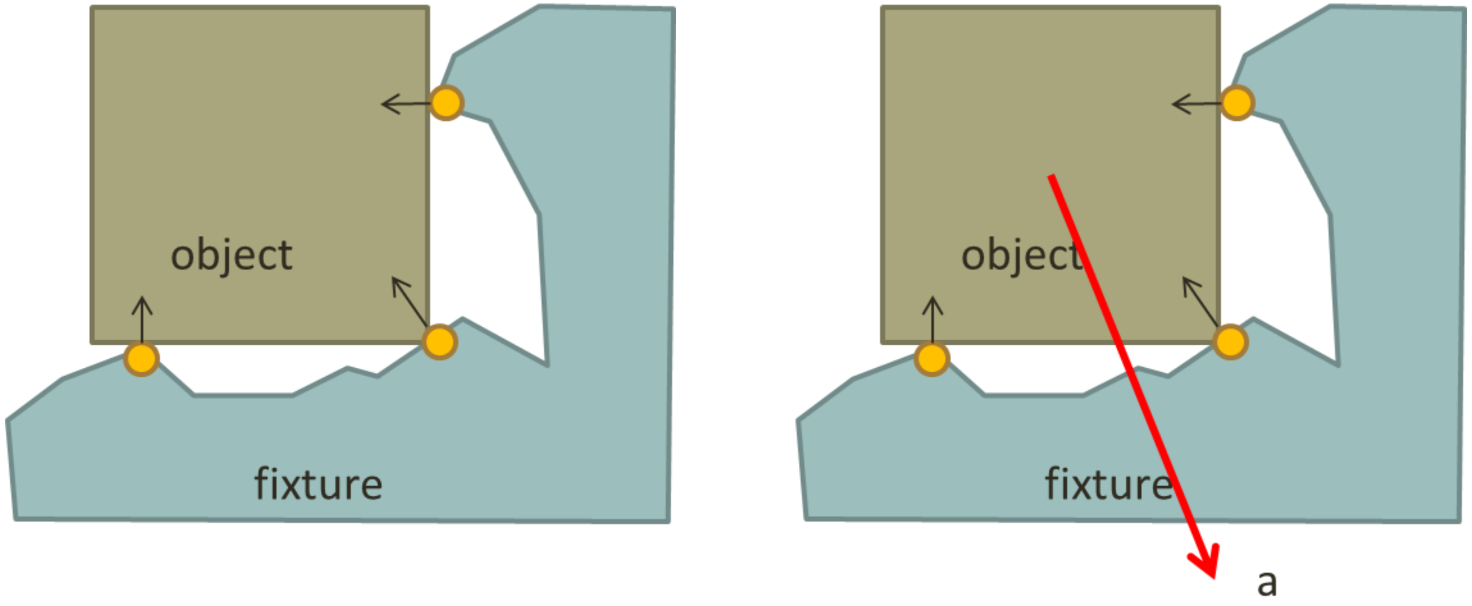
    - f_{in} is the force applied normal to the surface
    - f_{io} and f_{it} are the forces applied along the surface



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

Grasp Restraint

- Form closure
- Force closure



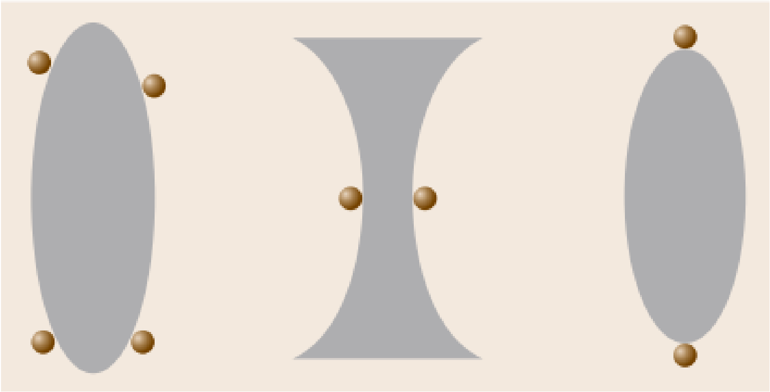
Form Closure

- Form closure grasp
  - The object cannot move **regardless of surface friction**

- What does this imply?
  - If the grasping hand has its joints locked, it is impossible to move the object, even infinitesimally

Form closure

- Which of these is in form closure?



- Example – power (enveloping) grasp
  - Palm and finger wrap around the object

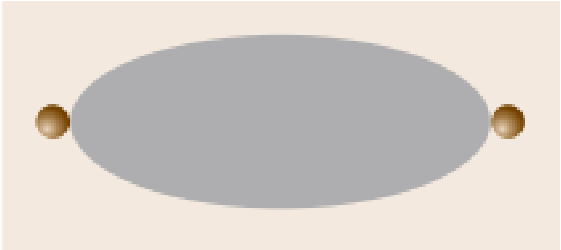
Form Closure

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

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

Force Closure

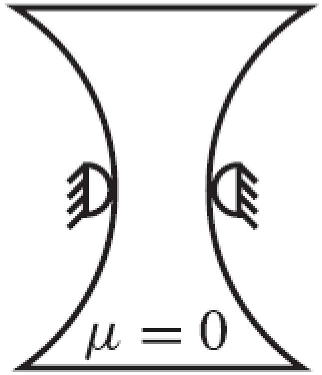
- Definition
  - **Frictional properties** of the object can be used to immobilize the object



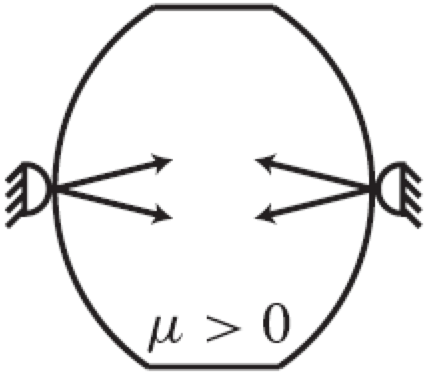
- What does it imply?
  - If the grasping hand has its joints locked, stability of this grasp depends on friction between contacts and object (μ)

Form closure VS Force closure

Form closure VS Force closure



Form closure $\not\longrightarrow$ force closure.



Force closure $\not\longrightarrow$ form closure.

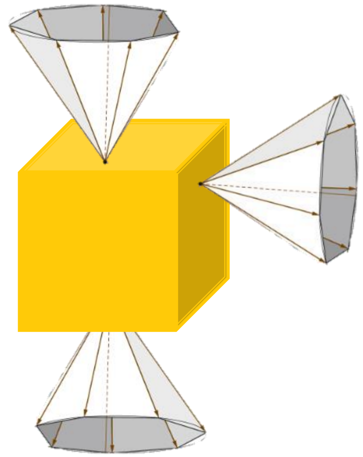
Testing of Force Closure

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

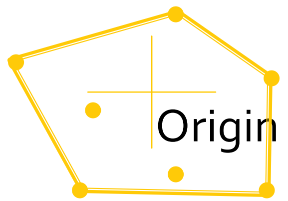
Input: Contact locations

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)

Why does this algorithm work?

Force Closure

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

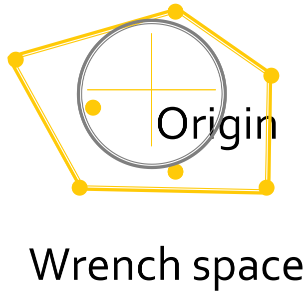
Wrench space **Wrench space**

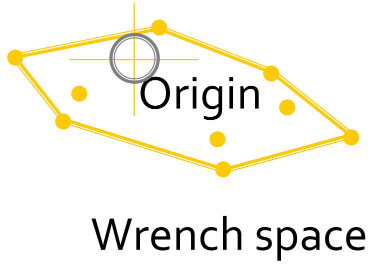
Note: wrench space is 6-dimensional, these are only cartoons

- Yes or no answer isn't enough to choose between grasps
  - How to measure grasp quality?

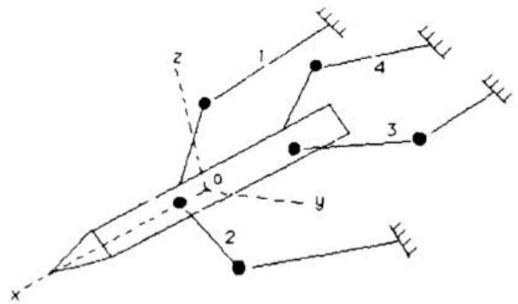
Force Closure Metrics

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



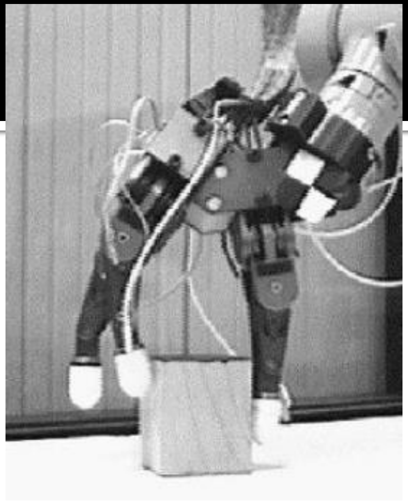


- Task-specific metric of Li and Sastry
  - Use **task-specific ellipsoid** instead of hyper-sphere



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

Searching for Force Closure Grasps

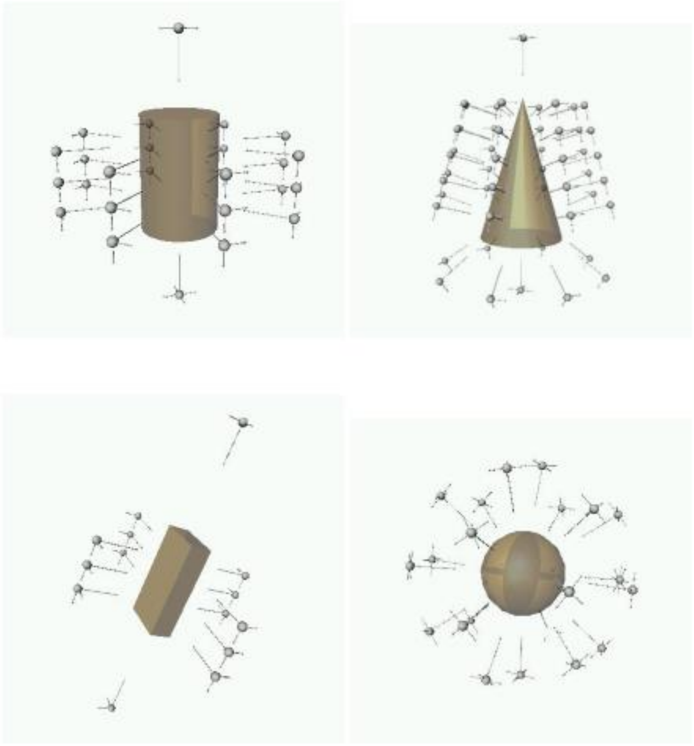
- In the 90s
  - Search for a **set of *n* point contacts** on **the surface of an object**, where *n* is the number of fingers of your hand

- 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.):
  - Sample hand pose relative to object with fingers in a pre-shape
  - Approach object until contact and close the fingers
  - Get contact points between fingers and object
  - Test these contact points for force closure

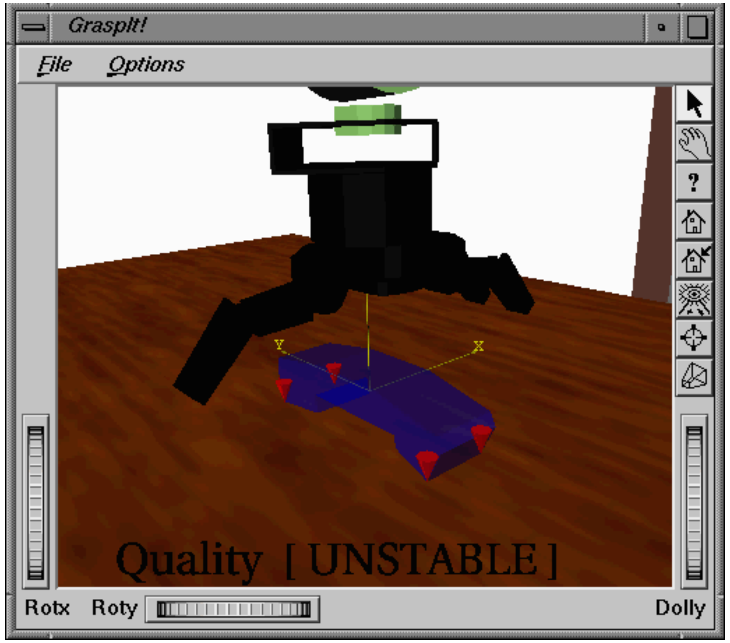


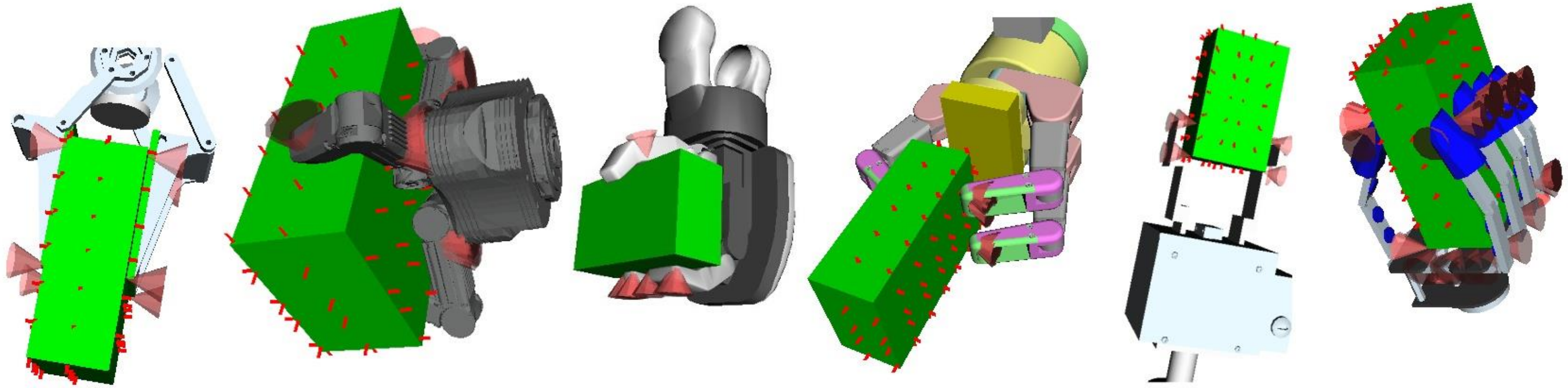
- Advantages?
  - **Search space** is only 6-dimensional (pose of hand) + set of pre-shapes
  - Search can be arranged so hand always approaches parallel to surface of object

Pre-computing Grasp Sets

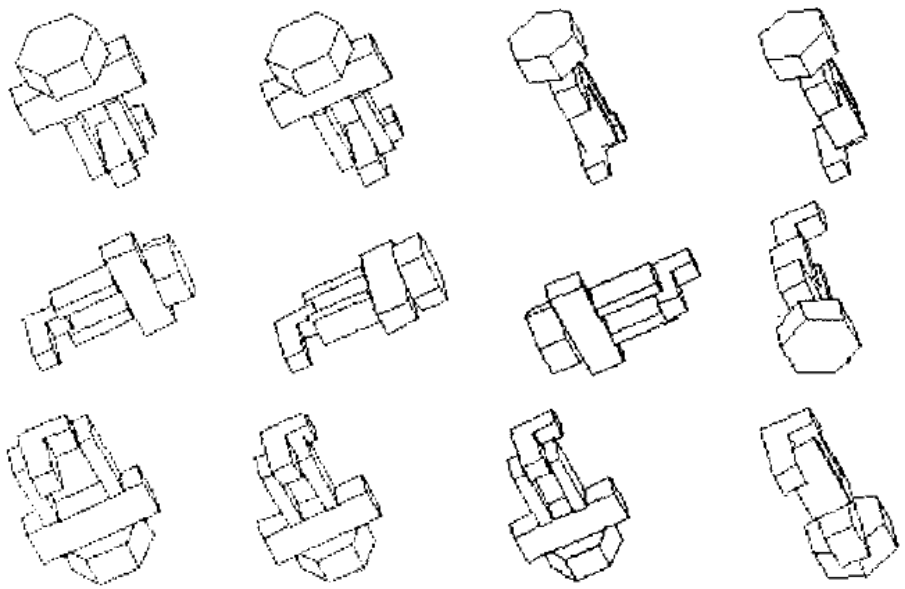
- Searching for grasps is slow!
  - Especially with dynamics (i.e. if you don't assume object is fixed)

pre-compute a set of stable grasps for a given object!





Pre-computing grasp sets is not new!



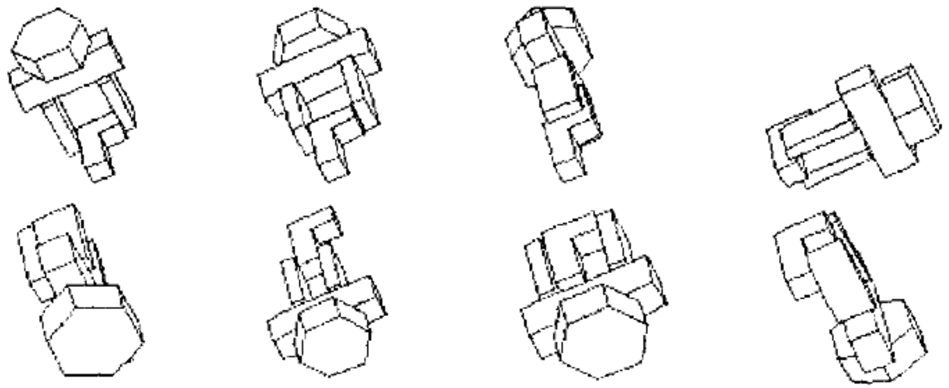


Figure 6. The different groups of approach directions and grasp classes for a particular orientation of an L-shaped object, heuristically ranked by desirability.

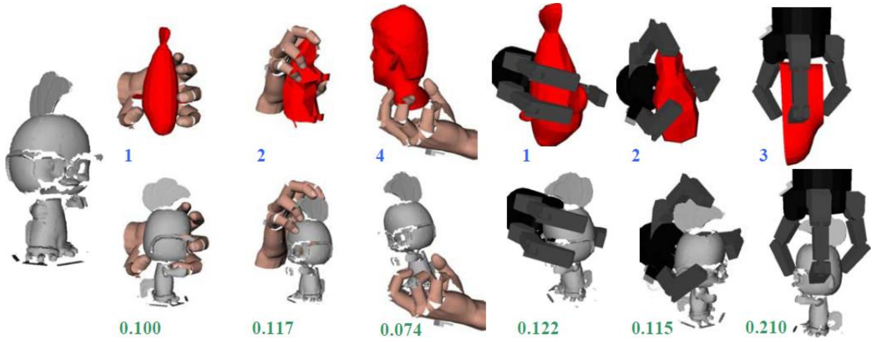
5.1 Choosing a grasp

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

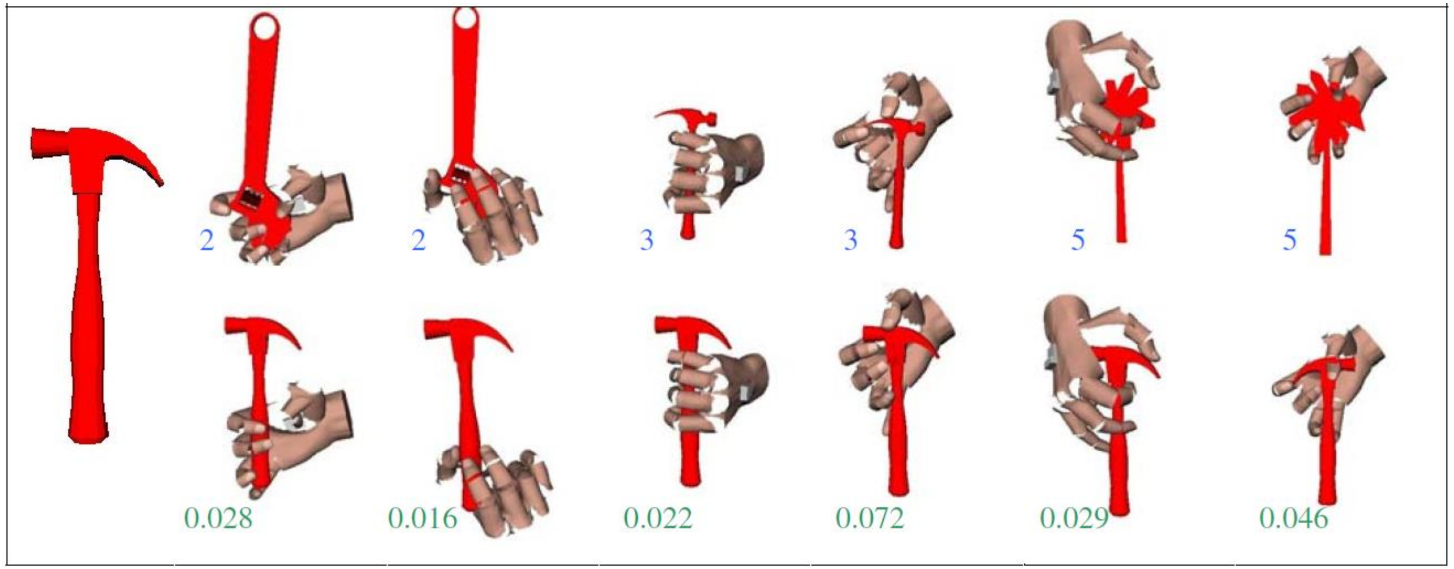
- Reuse the 3D models from the Princeton Shape Benchmark (PSB)
  - Well known academic dataset of 1,814 models
  - All models resized to “graspable” sizes
  - PSB models were not originally selected based on the need of robotic grasping
    - Some of the models are not obvious choices for grasping experiments.



http://grasping.cs.columbia.edu/

Columbia Grasp Database

- 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



Columbia Grasp Database

- How to compute a grasp given the database?

Shape Matching → Find k models in the database most similar to a given model, under some shape similarity metric.

↓

Collocating

↓

Grasp computing

Shape metric?

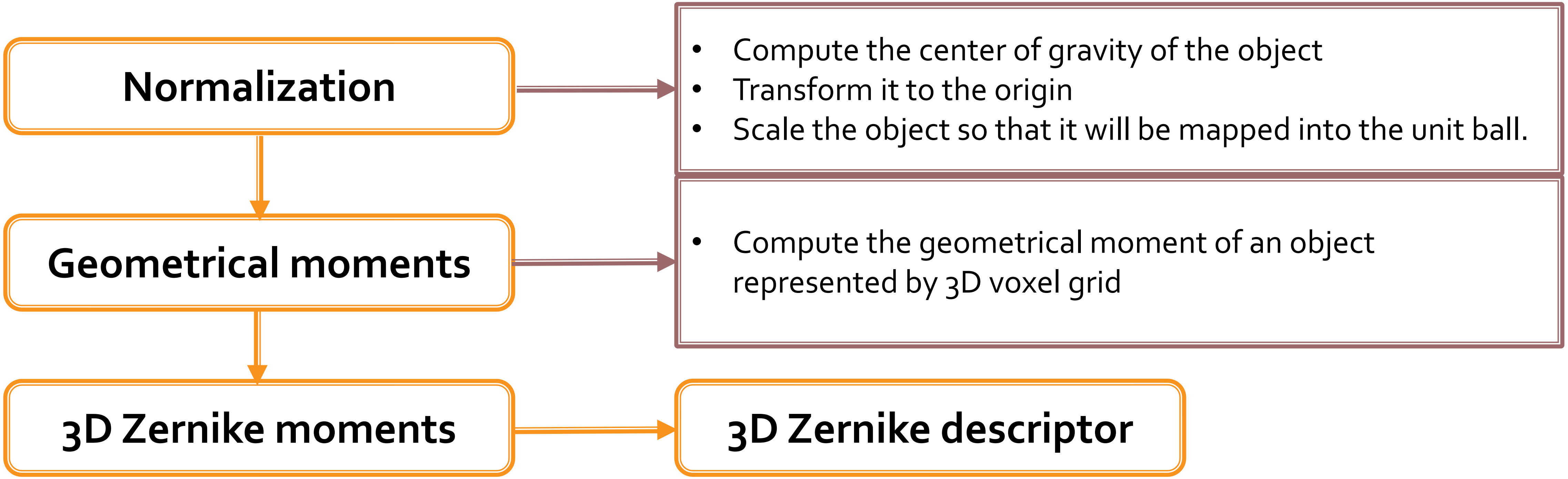
Shape metric – L^2 distance between **Zernike descriptor**

Metrics for similarity of 3D objects

- Content based 3D shape retrieval
  - Map a 3D objects into compact canonical representations referred to as **descriptors**, which serve as search keys during the retrieval process
- Ideal descriptor
  - Invariance under scaling, rotation and translation
  - Can measure the similarity of 3D objects

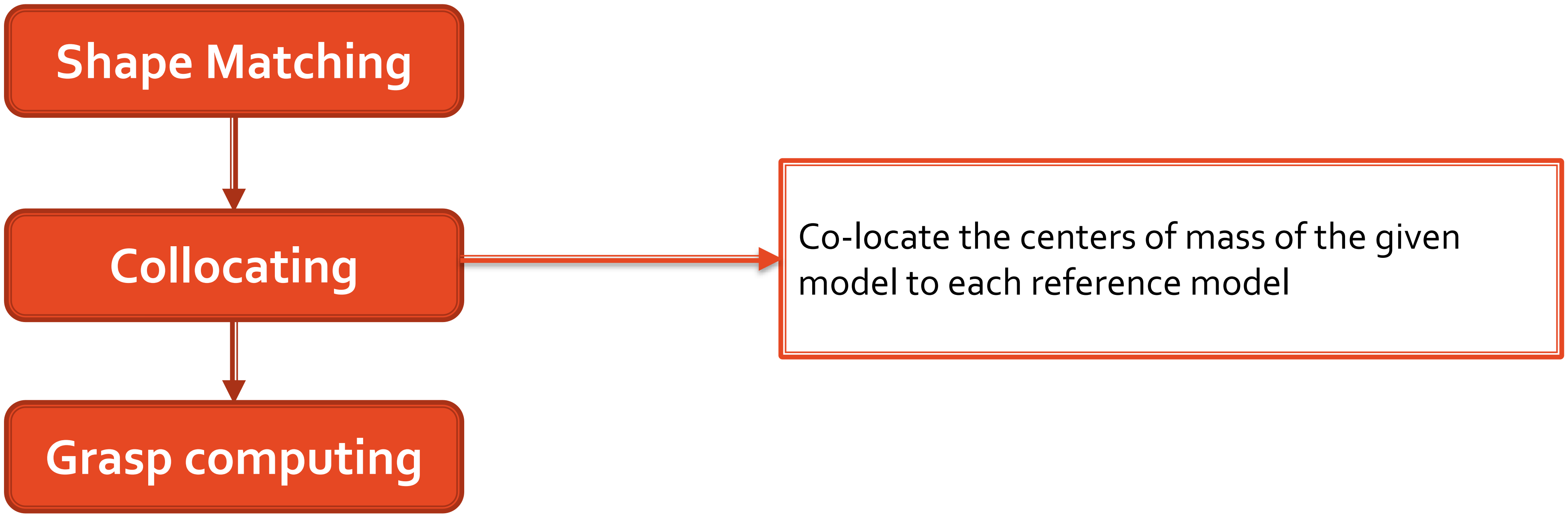
Zernike descriptor

- A scalar computed from polynomial bases that are invariant to transformations



Columbia Grasp Database

- How to compute a grasp given the database?



Columbia Grasp Database

- How to compute a grasp given the database?

Shape Matching → **Collocating** → **Grasp computing**

Performance:
20 seconds, from shape matching to final output

Choose the ones
- Without collision when open and close the hand
- Above the grasping quality threshold

Integrating Grasping and Manipulation Planning

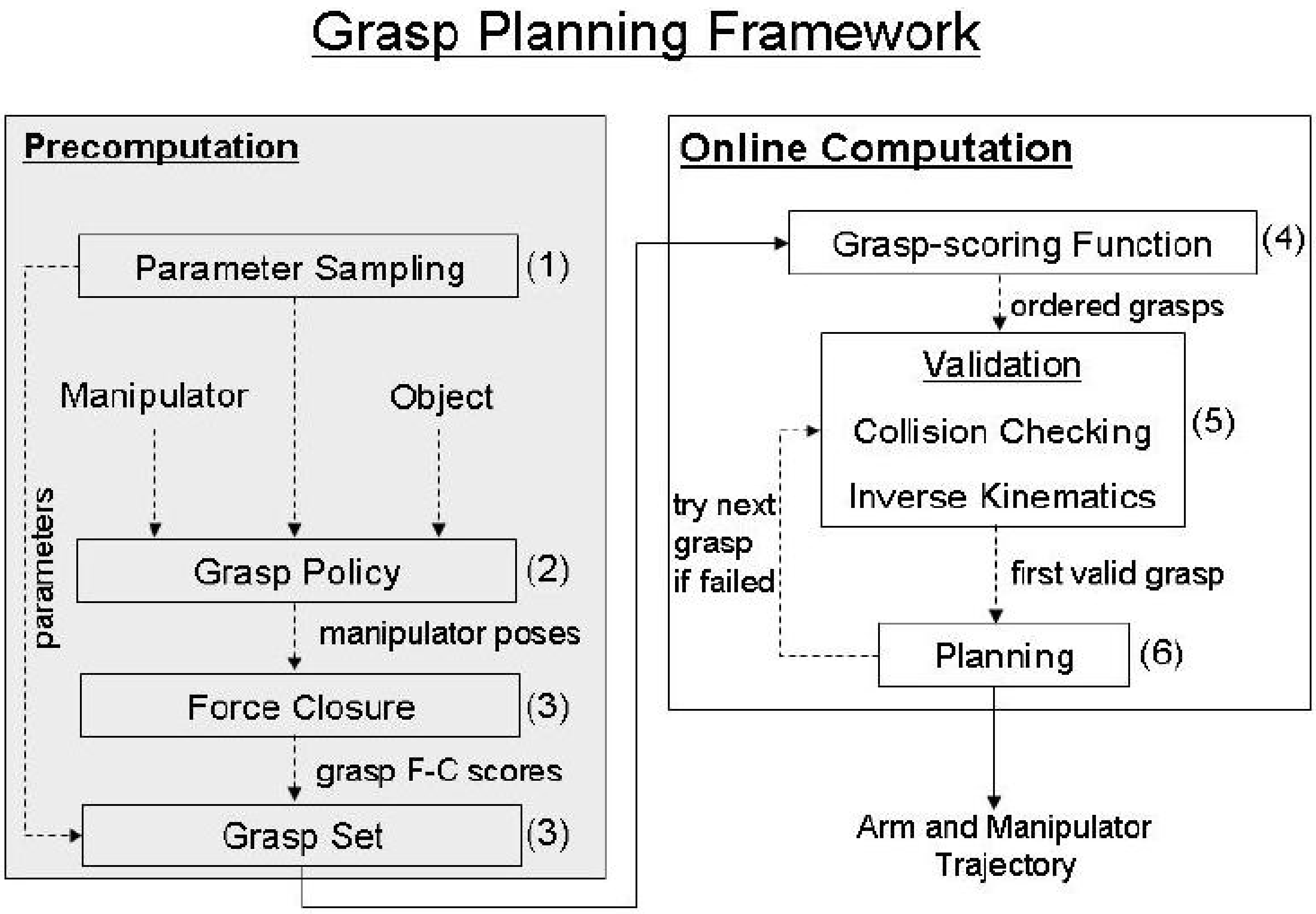
- So far, we only test for collision with obstacles online
  - Ignore them when computing grasp set)
- We wanted to integrate grasp planning motion planning
  - Consider obstacles and reachability

Integrating Grasping and Manipulation Planning

- Approach
  - Pre-compute grasp set offline, get force-closure score
  - Online, compute 2 scores for each grasp
    - Environment Clearance Score
    - Reachability Score

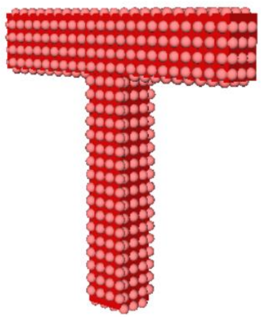
[Berenson, D., Diankov, R., Nishiwaki, K., Kagami, S., & Kuffner, J. (2007). Grasp Planning in Complex Scenes. *IEEE-RAS International Conference on Humanoid Robots (Humanoids07)*]

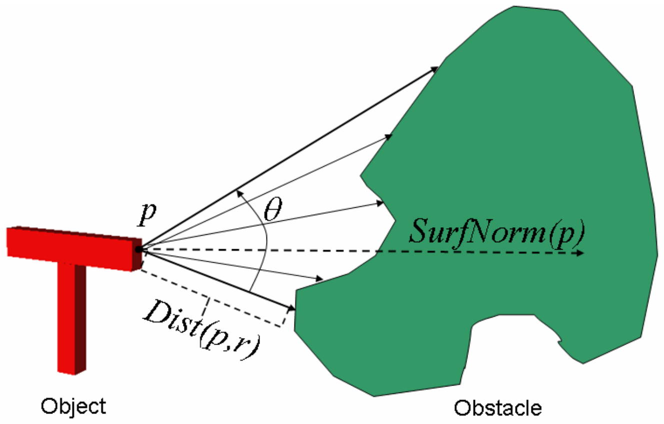
Integrating Grasping and Manipulation Planning

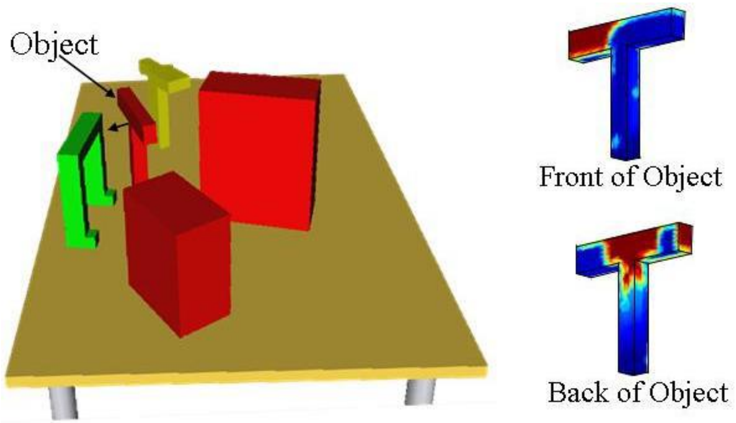


Computing Environment Clearance Score

- Step 1 – sampling
  - Sample on object surface
  - For every **target point Pt**, define the **approach directions Pd** as the negative of Object's surface normal at Pt.

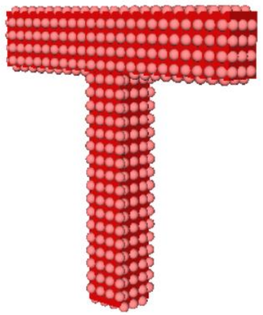


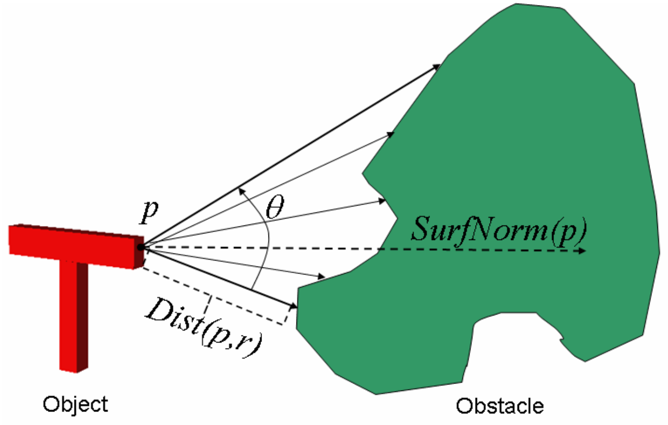


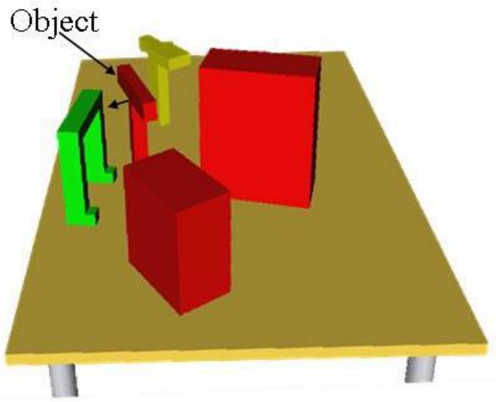


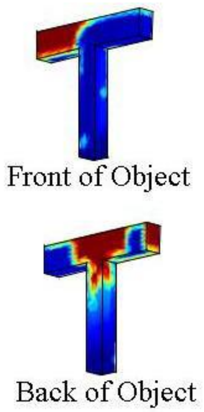
Computing Environment Clearance Score

- Step 2– compute distance map
  - Theta is the width of the cone whose tip is at P and whose alignment is the surface normal at P.
  - Dist(p, r) is the minimum distance as evaluate over all rays, r, in the cone.



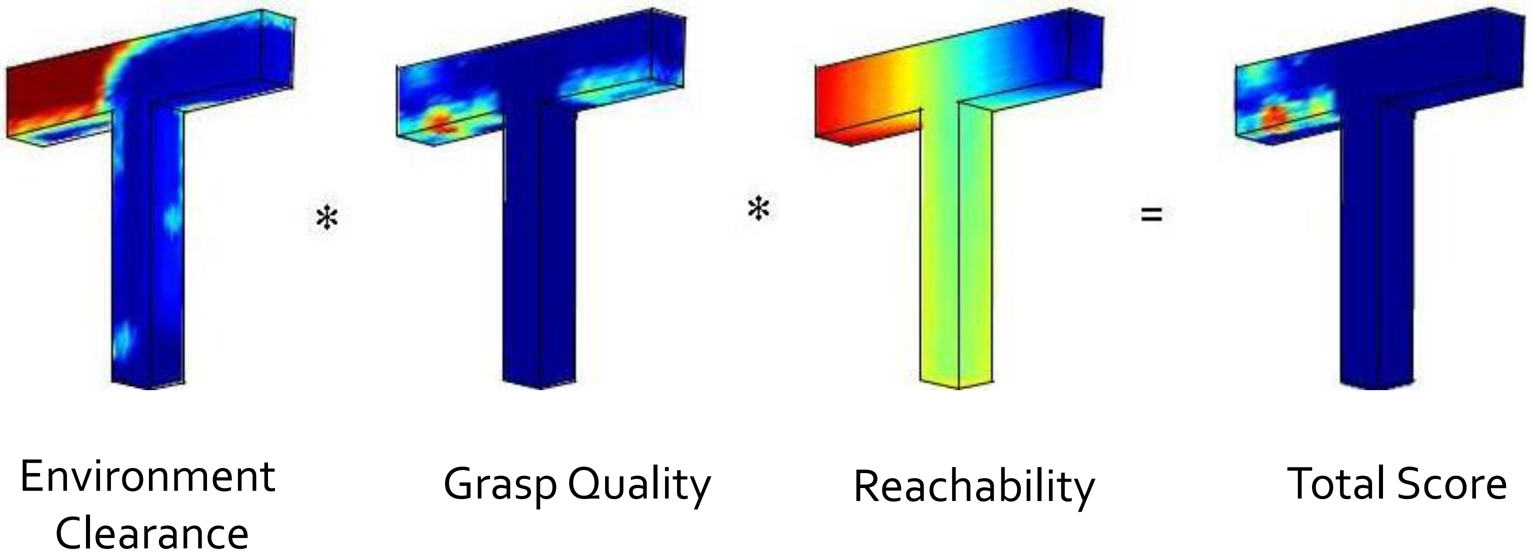






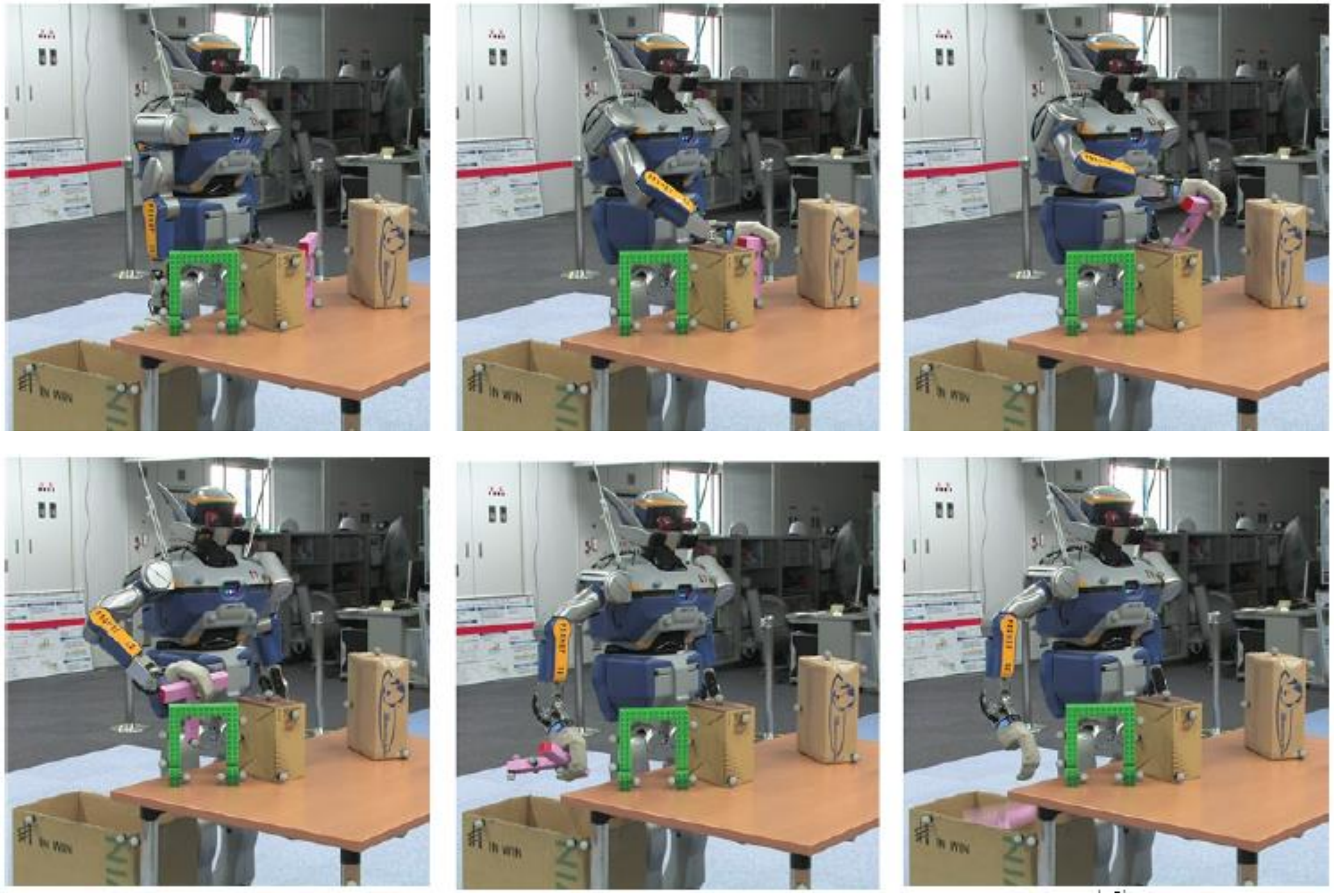
Integrating Grasping and Manipulation Planning

- Combine scores to create grasp ranking



- Test grasps in order of ranking
  - Much faster than testing in random order

Integrating Grasping and Manipulation Planning



Grasp Planning in Complex Scenes

- Motivation
  - Integration of grasp and manipulation planning is still limited to a fixed set of grasps
  - Next, we tried searching for grasps **online** using similar scoring

Online grasping planning in cluttered environment

- Challenges
  - High dimensionality of the hand configuration space
  - High cost of validating a candidate grasp
    - collision-checking, testing for force-closure.

- Solution
  - For a given pre-shape, focus grasp search to the **good region** of hand pose space
  - Cost function?
    - Object geometry, environment density, force-closure

Grasp Planning in Complex Scenes

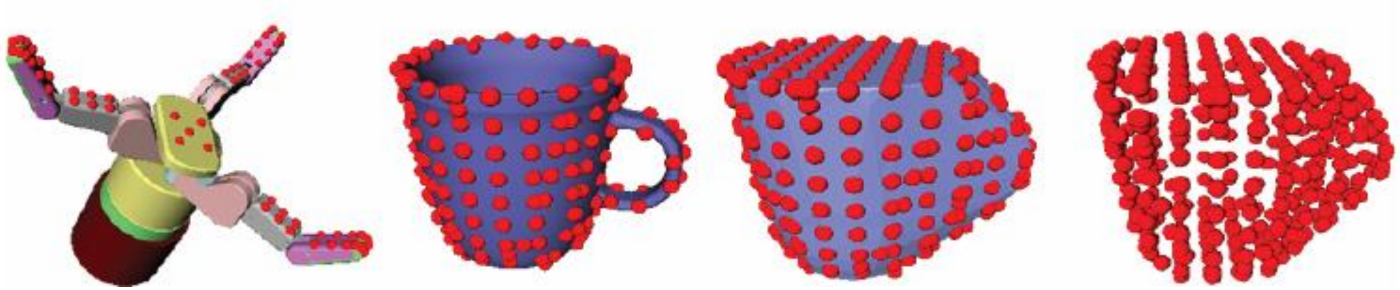
- Cost function for optimization

$$C(\mathrm{HPO}, O, E) = \frac{F(\mathrm{HPO}, O) + \zeta S(\mathrm{HPO}, O)}{X(\mathrm{HPO}, E)}$$

  - Approximate Collision – F(HPO,O)
    - Whether the fixed part of the hand will be in collision
  - Fit Cost – S(HPO,O)
    - The error of the fit between the pre-shape and the object at this hand pose
  - Contact Safety Cost – X(HPO, E)
    - The likelihood of the fingers being able to reach the desired contact points without collision – **how**?

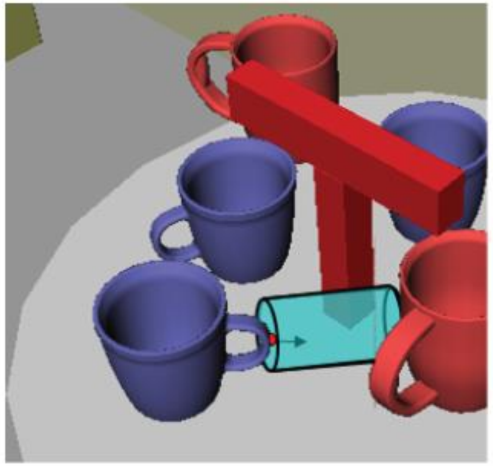
Grasp Planning in Complex Scenes

- To compute the clearance map
  - Sample on the surface of the object to be grasp offline
  - For each sample point, compute the cylinder/conical clearance map

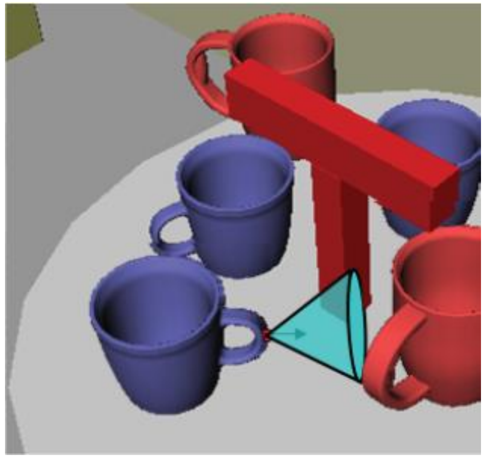


Clearance Map (CylCM)

- Cylinder Clearance Map (CylCM)
  - To evaluate the fixed part of the hand will be in collision

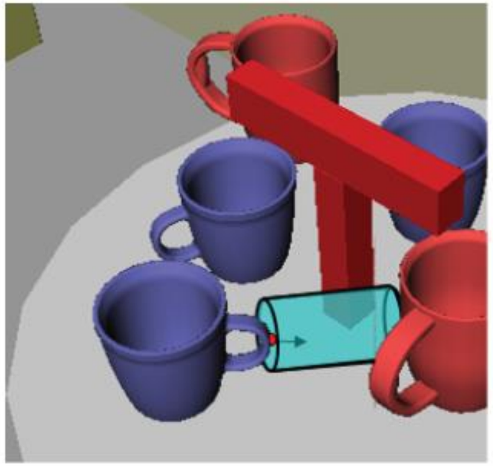


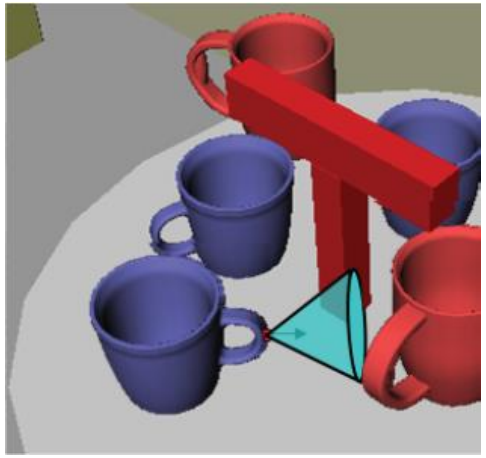
- Conical Clearance Map (ConCM)
  - To evaluate the cost of contacting the object when the fingers close the grasp



Computing the Clearance Maps

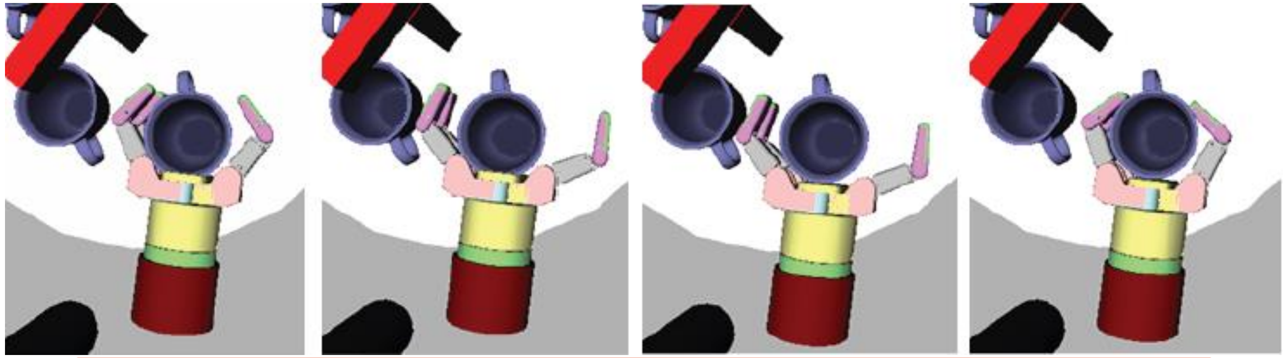
- Steps
  - At each of the sample points of the object, we compute the **Conical/Cylinder Clearance Map**
  - The height of the **longest collision free cone/cylinder** directed along the outward-facing surface normal at a point on the surface of the object becomes that point's score





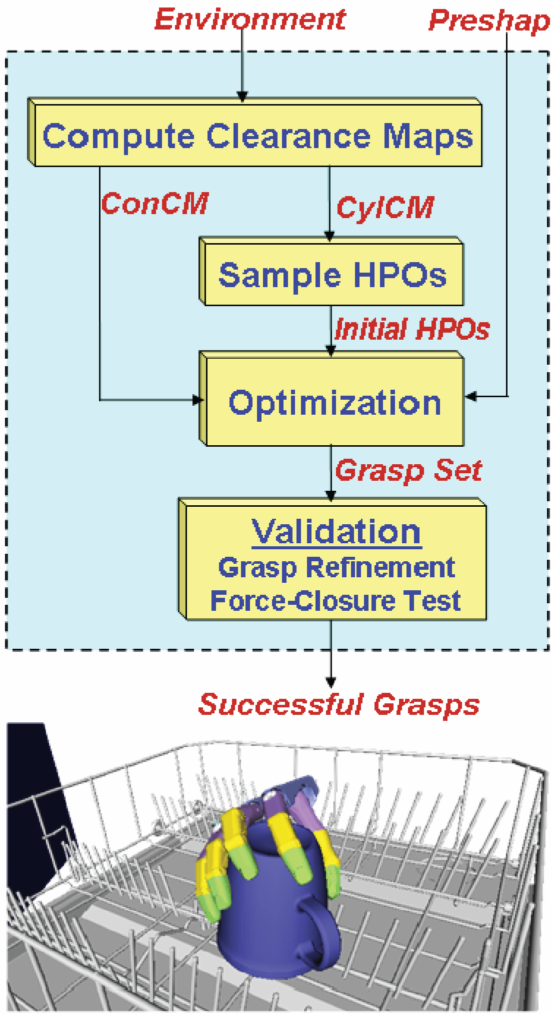
Online grasping selection and refinement

- The fingers are uncurled until they reach collision or a joint limit
- Curl the fingers until they are halfway between their starting position and the obstacle in collision
- If in collision, move the hand backward along the line define hand cylinder, and then moved forward slightly so that the hand is barely colliding with the object

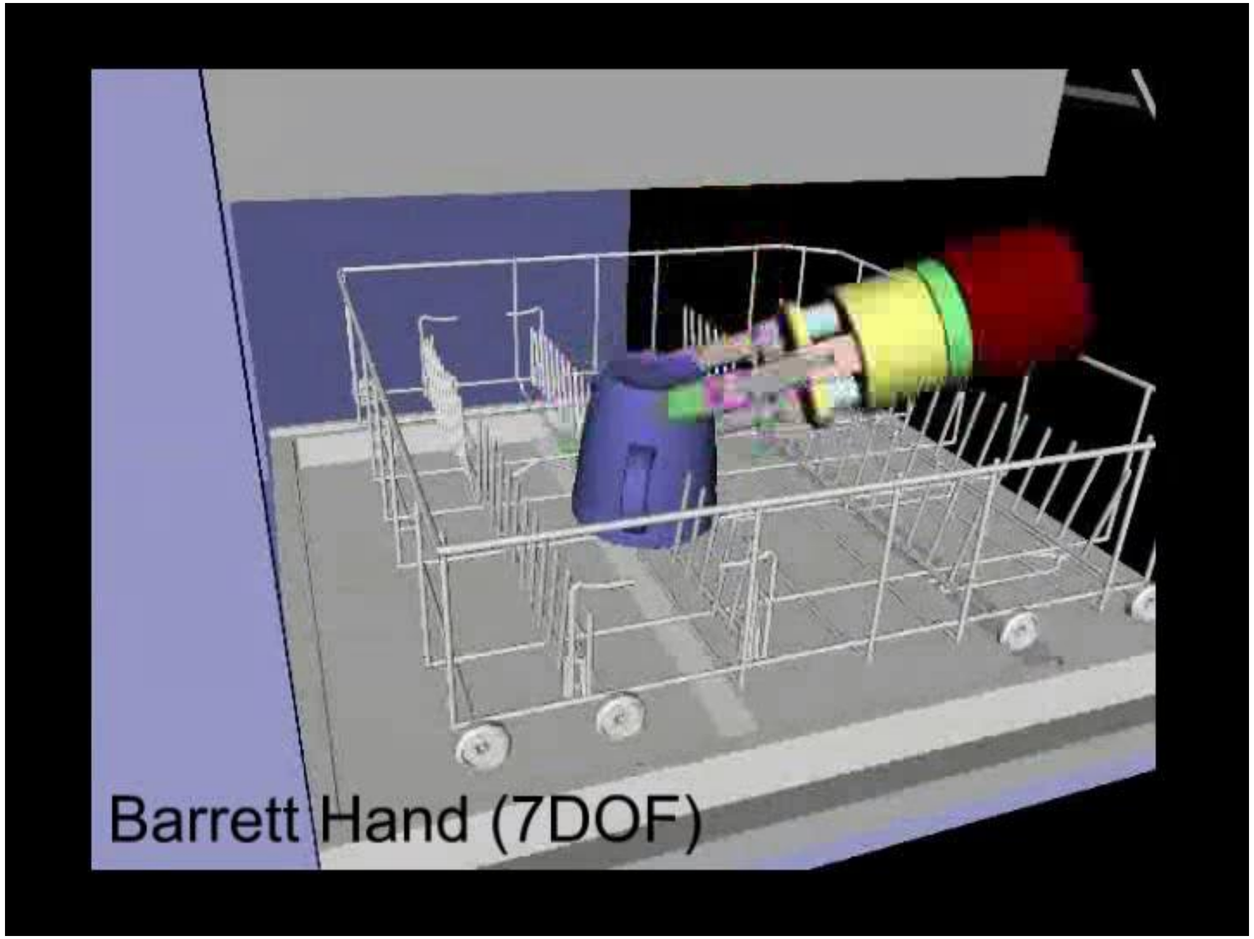


-

Grasp refinement to avoid **interpentration** of the palm



Grasping refinement

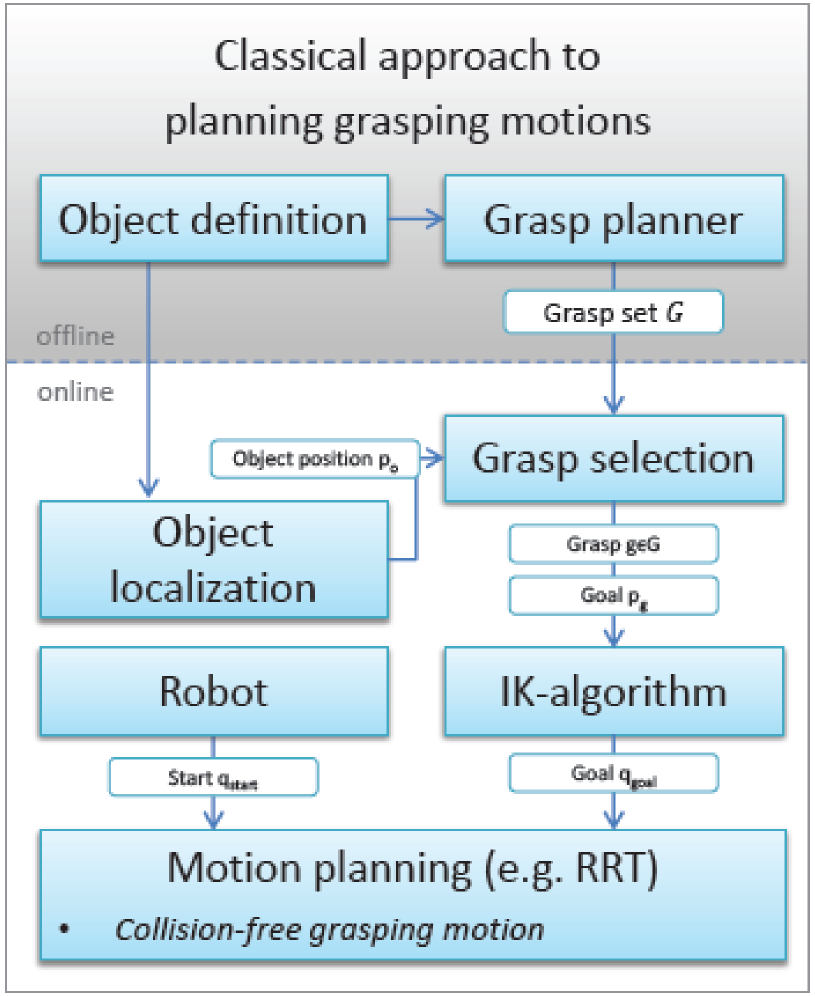


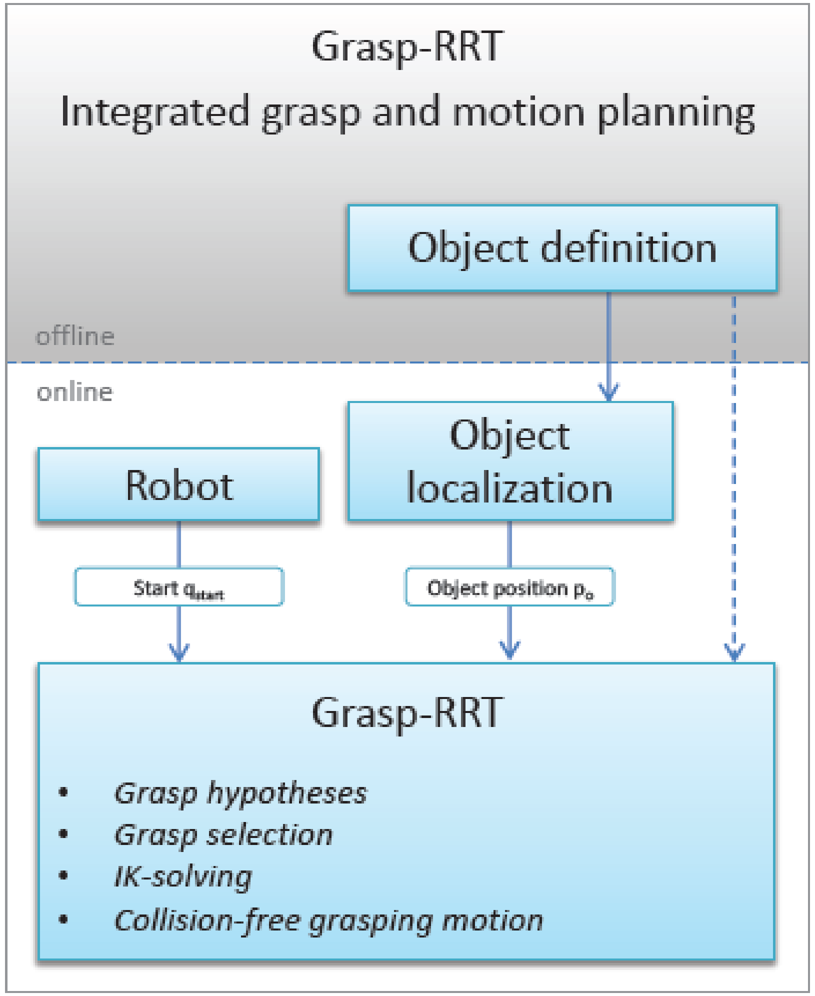
[Grasp Synthesis in Cluttered Environments for Dexterous Hands. Berenson and Srinivasa, Humanoids 2008]

Grasp–RRT planner

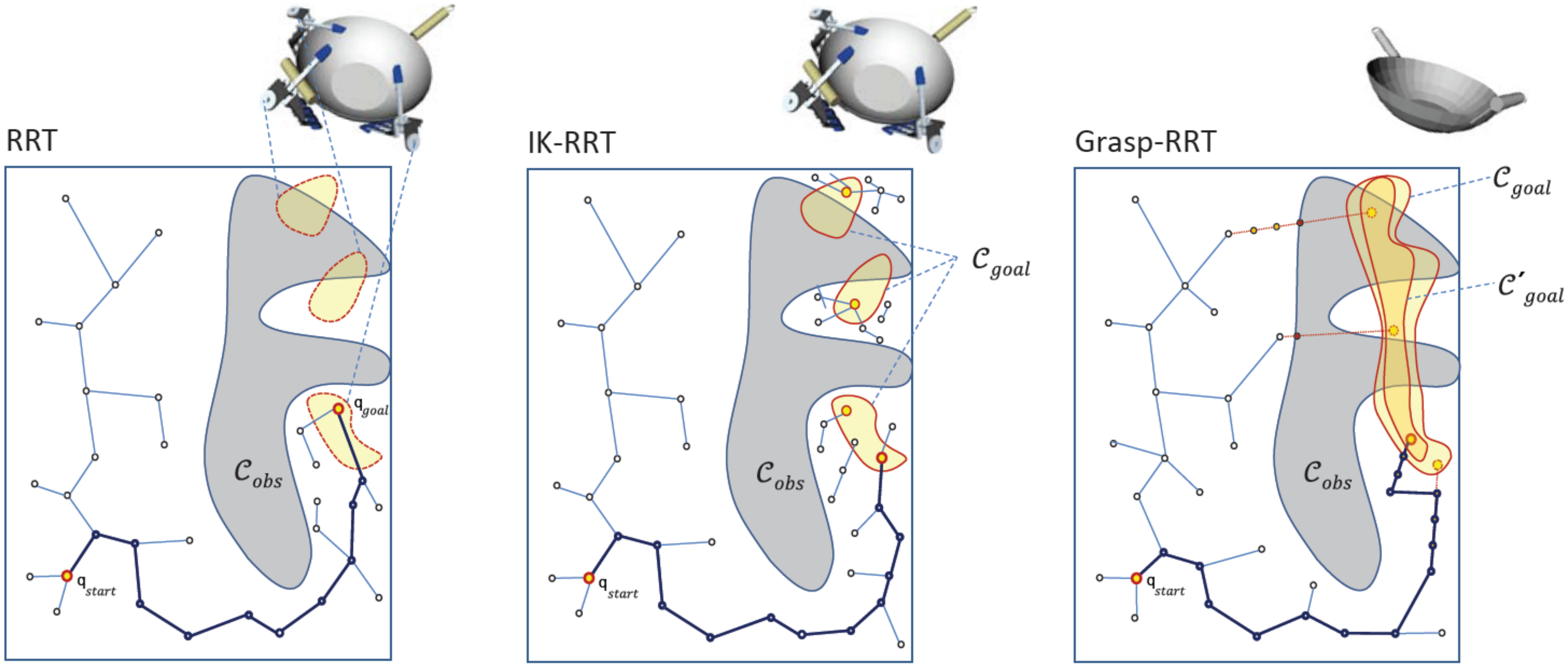
- Motivation
  - What if the object model is incomplete and/or inaccurate?
    - The pre-computed grasps may not fit well
  - No pre-calculated grasping data → pure online search
- Grasp–RRT planner
  - Build a feasible grasp +
  - Solving IK +
  - Search a collision-free trajectory to the grasping pose

Classical vs Grasp-RRT





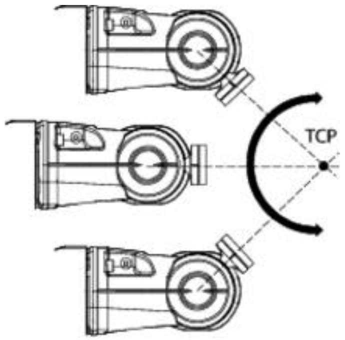
Comparison

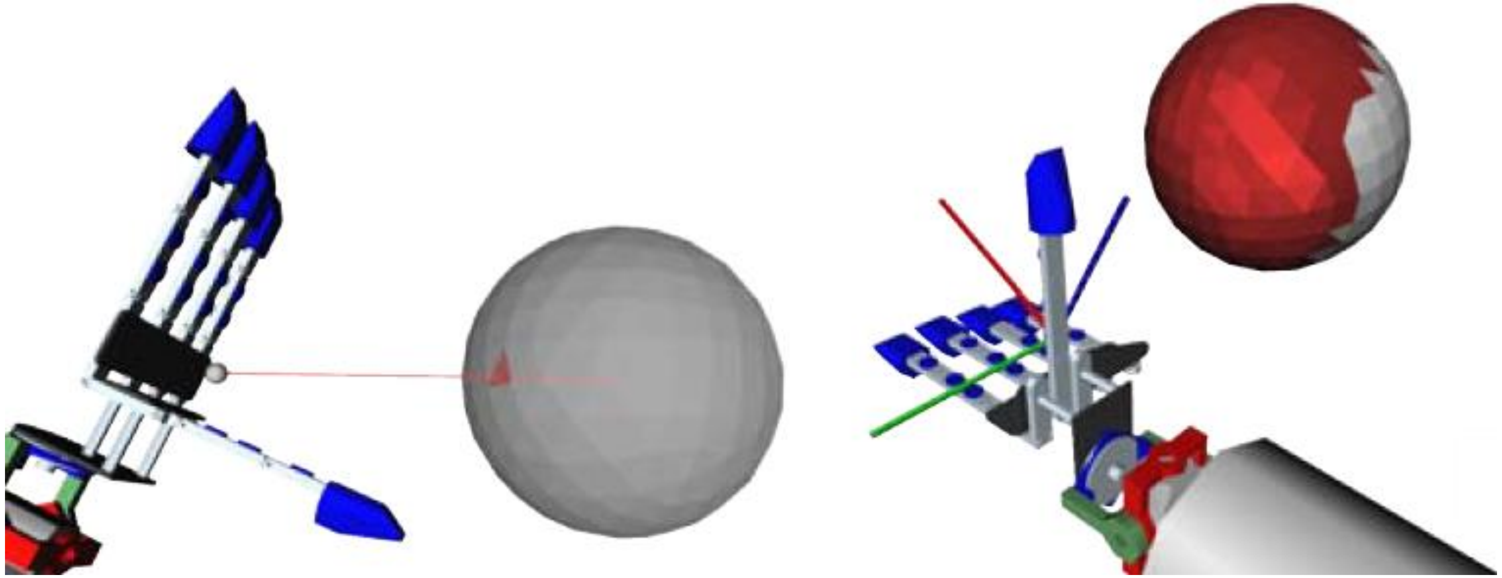


Vahrenkamp N, Asfour T, Dillmann R. Simultaneous grasp and motion planning: Humanoid robot ARMAR-III. IEEE Robotics & Automation Magazine. 2012 Jun;19(2):43-57.

Grasp–RRT planner

- Determine the approach direction
  - Approach sphere
  - Sampling distribution





Grasp–RRT planner

- Based on the approach direction, Compute a virtual target pose
- Resolve IK and move towards the target pose as far as possible
- Validate the grasping pose
  - Closing the fingers, determining the contacts and performing grasp wrench space analysis

Grasp–RRT planner

- Compute a target pose