Karlsruhe Institute of Technology

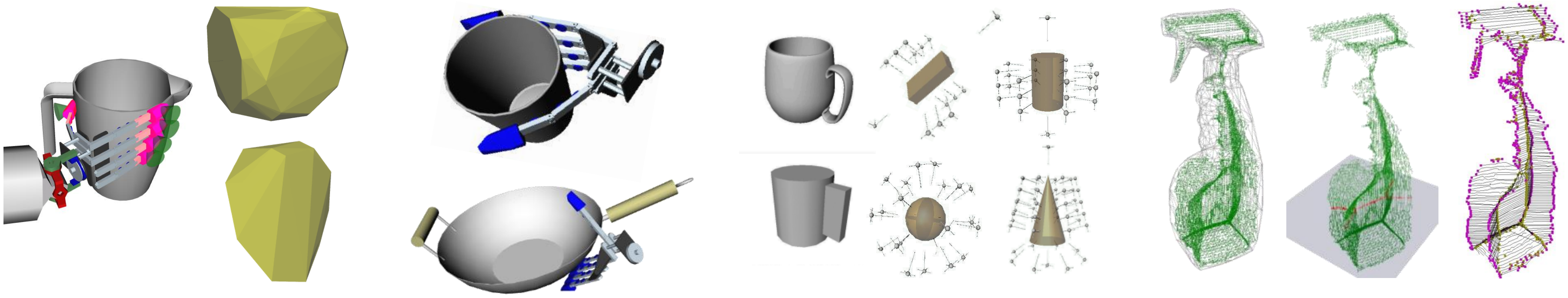
High Performance Humanoid Technologies

Robotics I: Introduction to Robotics

Chapter 8 – Fundamentals of Grasping

Tamim Asfour

https://www.humanoids.kit.edu



Content

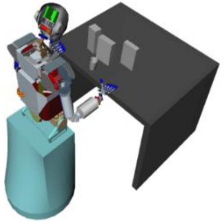
- Motivation
- Grasp Taxonomies
- Contact Models and Grasp Wrench Space
- Grasp Planning and Grasp Synthesis
- Examples: Grasping with ARMAR

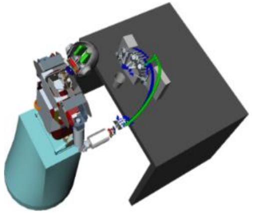
Content

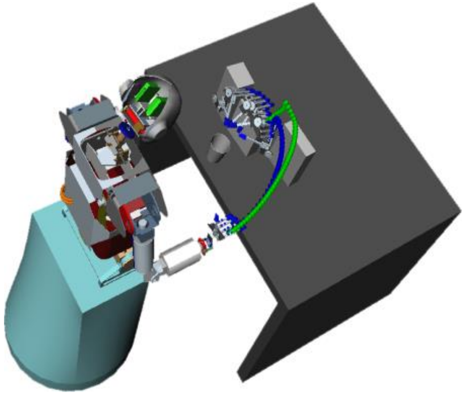
- Motivation
- Grasp Taxonomies
- Contact Models and Grasp Wrench Space
- Grasp Planning and Grasp Synthesis
- Examples: Grasping with ARMAR

Motivation

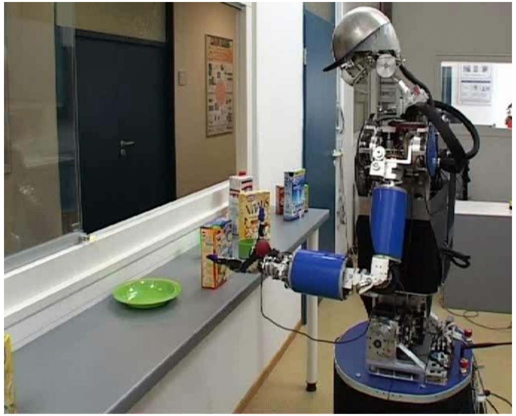
Grasping Task → Scene Representation → Motion Planning → Sensor-Based Execution



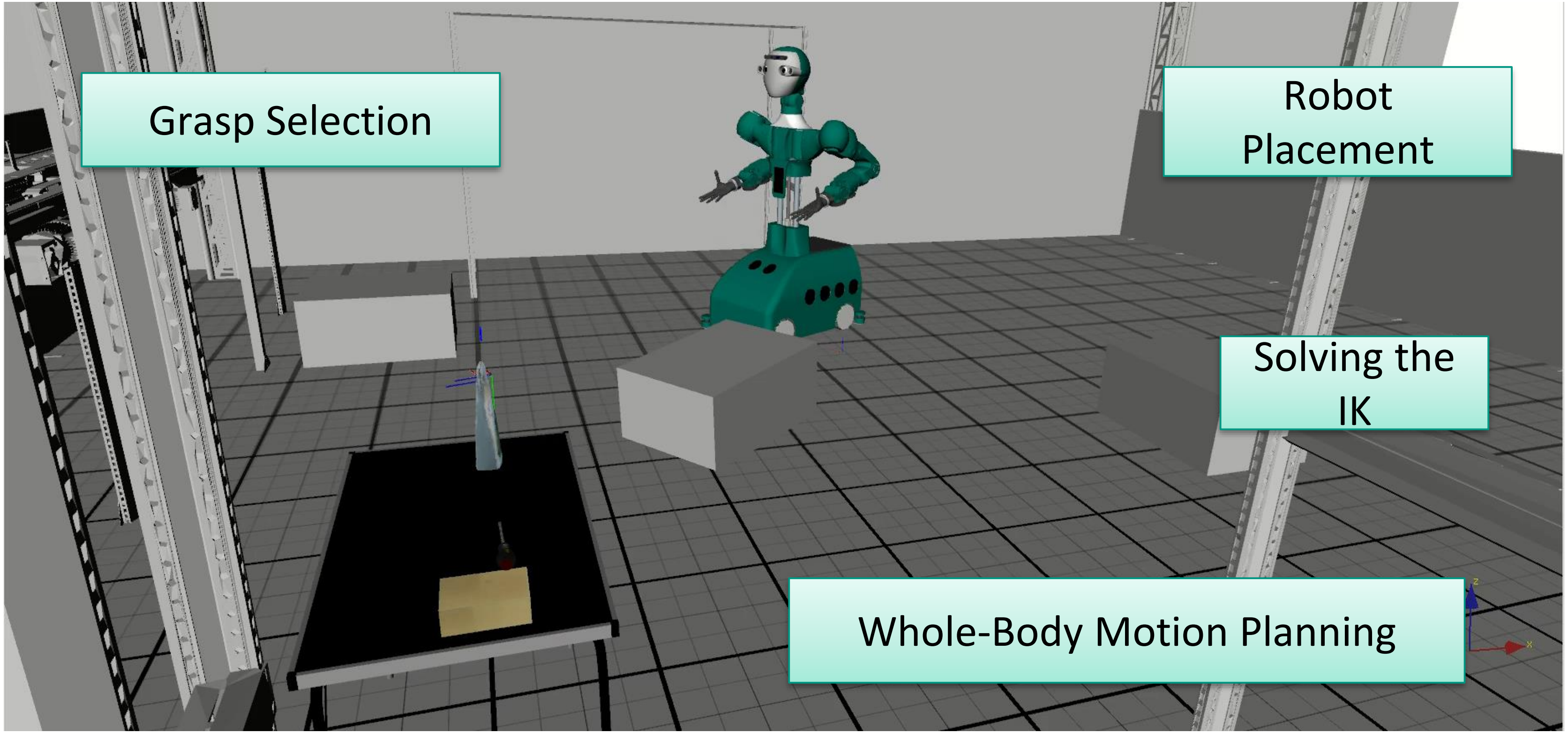




Motivation – Grasping

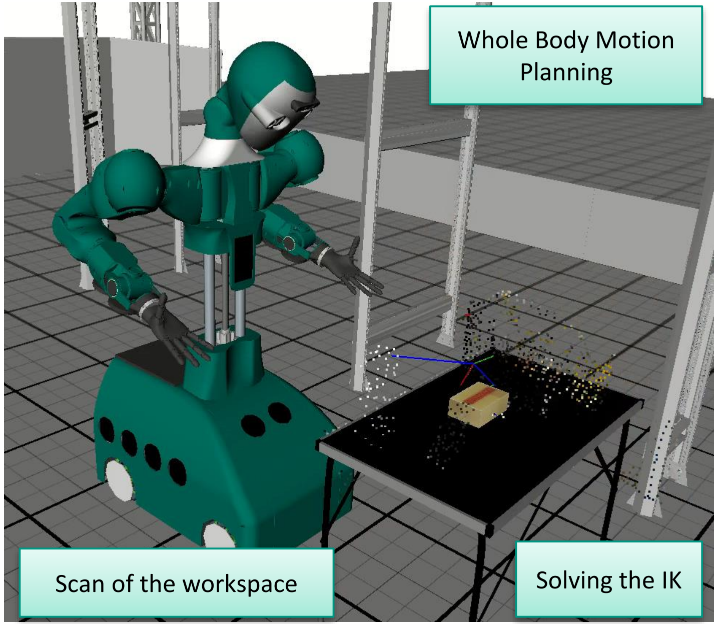


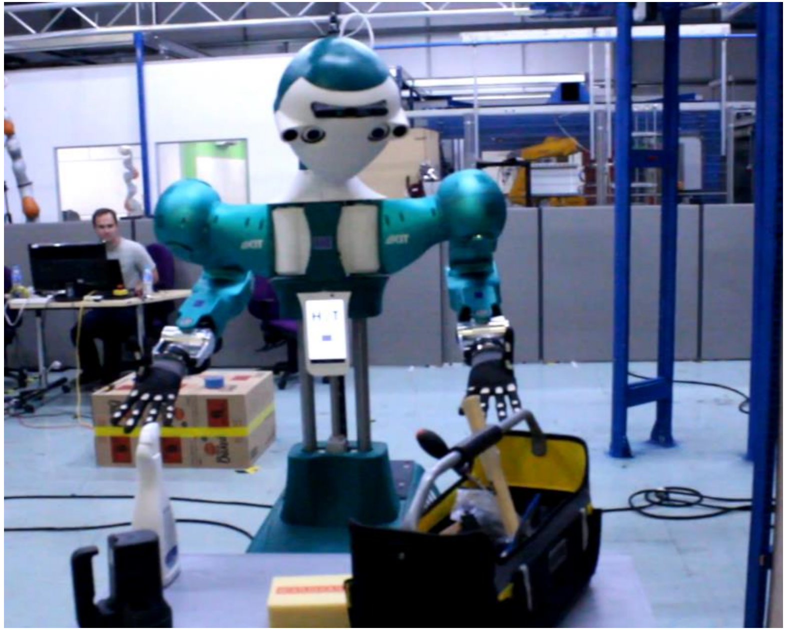
Motivation – Mobile Manipulation



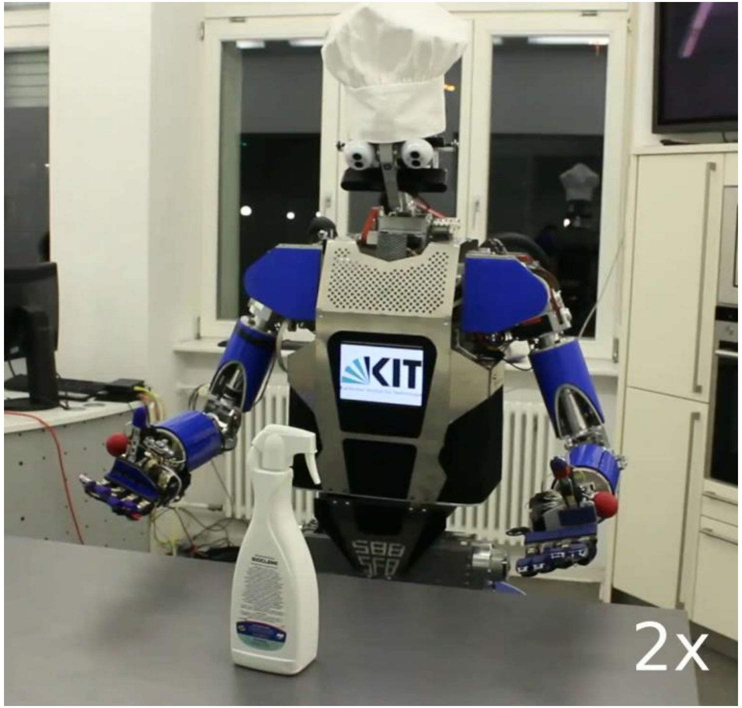
Motivation – Mobile Manipulation

Motivation – Grasping in Partially Unknown Environments

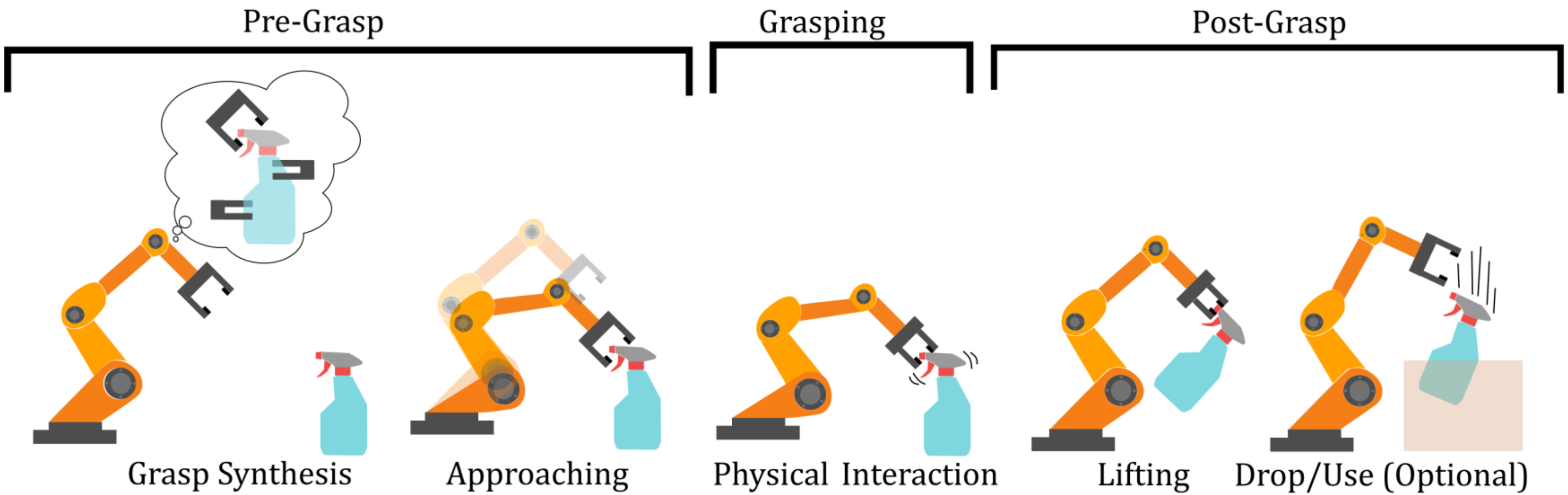




Motivation – Regrasp



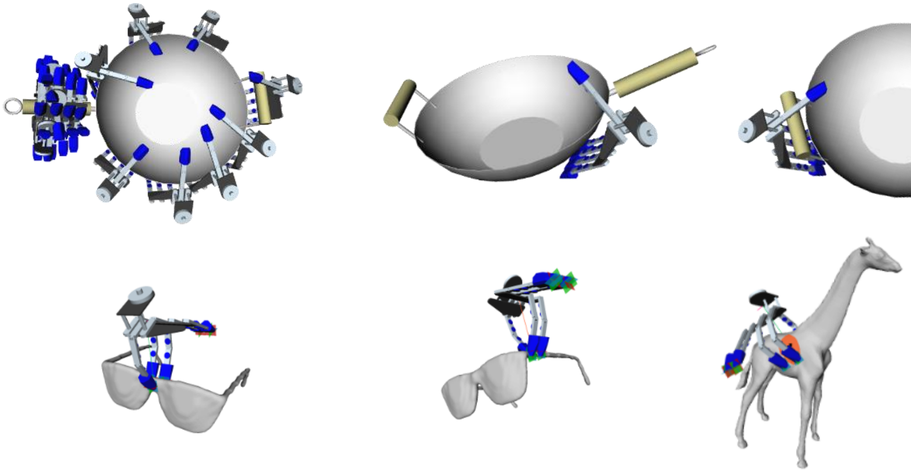
Phases of Grasping

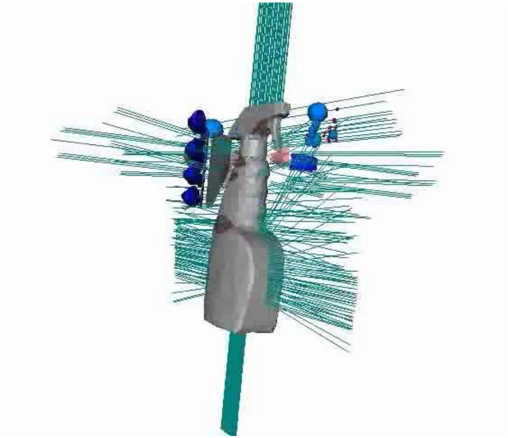


Newbury, R., Gu, M., Chumbley, L., Mousavian, A., Eppner, C., Leitner, J., Bohg, J., Morales, A., Asfour, T., Kragic, D., Fox, D. and Cosgun, A., *Deep Learning Approaches to Grasp Synthesis: A Review*, IEEE Transactions on Robotics, vol. 39, no. 5, pp. 3994-4015, 2023

Motivation – Determination and Selection of Grasps

- **Infinite** possibilities to grasp an object



- **How are grasps determined?**
- **How is a grasp selected?**

Motivation – Determination and Selection of Grasps

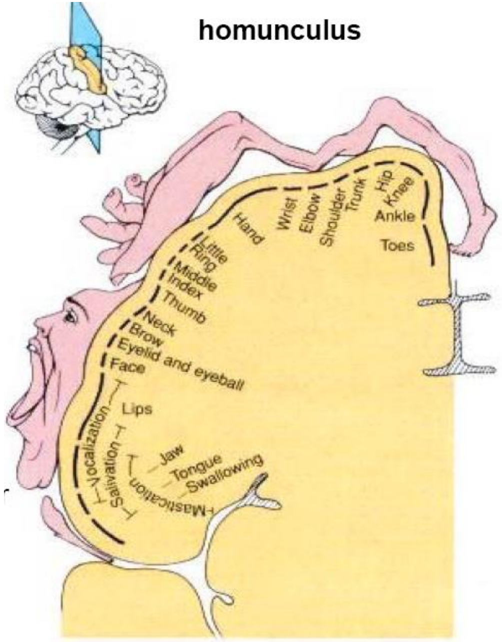
- Which information is crucial?
  - Geometry/shape of the objects
  - Interaction forces between finger and object
  - Kinematics of the hand

- **Grasping means contact between hand and object**

- **Lecture Robotics I**
  - Basics of grasping known objects
- **Lecture Robotics II**
  - Advanced methods of grasping (similar and unknown) objects

Motivation – Grasping in Humans

- Grasping as a (open- and closed-loop) control problem
  - Much is known
- Cognitive aspects of grasping
  - Little is known
  - Large part of the human cortex is dedicated to the hand
  - **Understanding hands = Understanding intelligence (?)**



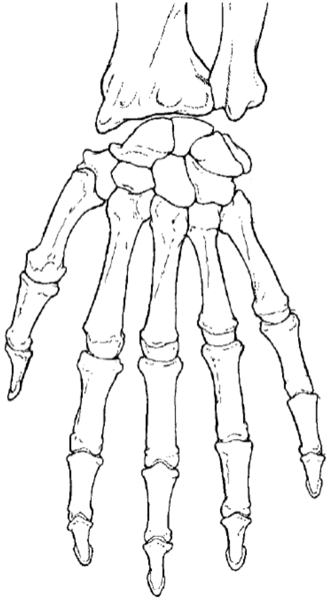
https://harmonicresolution.com/homunculus1.jpeg

The Human Hand

- 27 Bones
- 21 Degrees of Freedom (DoF)
  - 3 DoF Flexion/Extension per finger
  - 1 DoF Abduction/Adduction per Finger
  - 5 DoF Thumb
    - 3 DoF Flexion/Extension
    - 2 DoF Abduction/Adduction
  - 6 DoF for the wrist (palm)

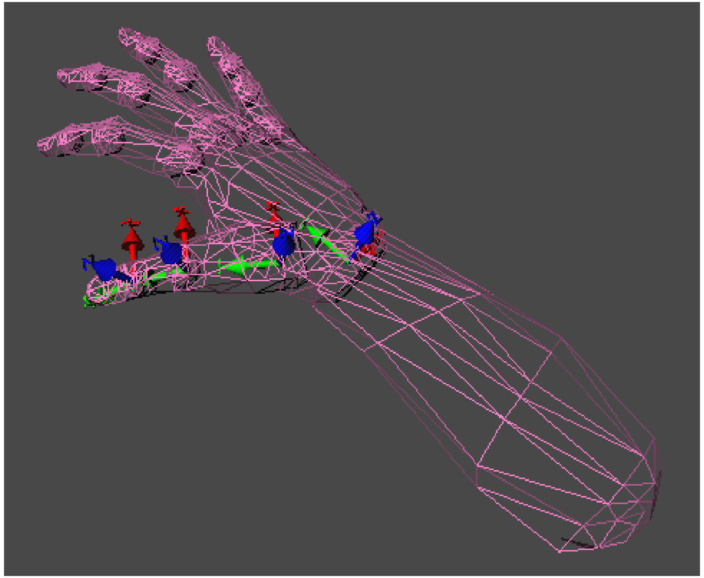
- **Total: 21 + 6 = 27 DoF ⇒ high-dimensional configuration space**

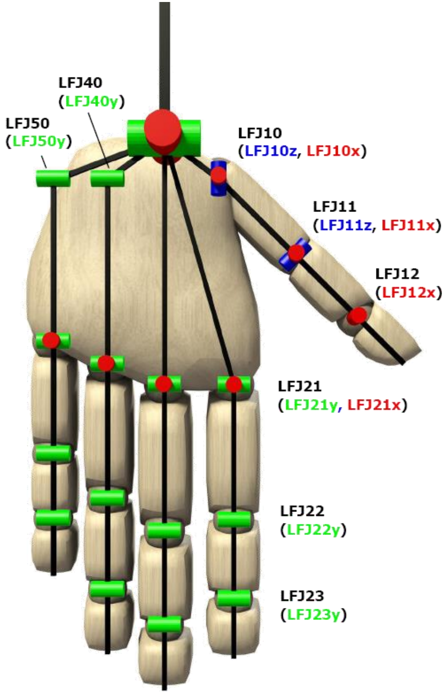
- Modeling:
  - Kinematic Model
  - Surface-based geometric model



U. Schmidt, Hans-Martin; Lanz. Chirurgische Anatomie der Hand. Stuttgart, New York, 2003. Georg Thieme Verlag

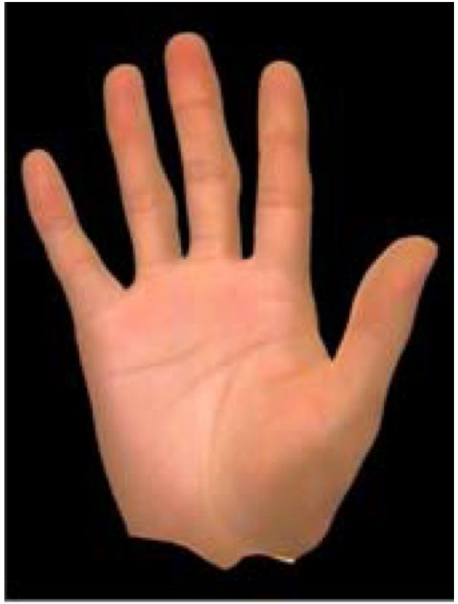
Kinematic Model

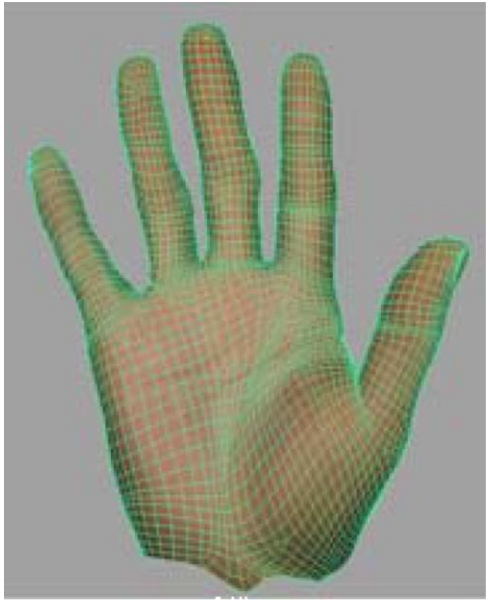




C. Mandery, Ö. Terlemez, M. Do, N. Vahrenkamp, T. Asfour, Unifying Representations and Large-Scale Whole-Body Motion Databases for Studying Human Motion, IEEE Transactions on Robotics 2016

Surface-based geometric model





T. Rhee, U. Neumann, J. P. Lewis. Human hand modeling from surface anatomy. Symposium on Interactive 3D Graphics and Games (I3D '06), 2006. DOI: https://doi.org/10.1145/1111411.1111417

MMM Model

The Human Hand

- High-dimensional configuration space

- How can the complexity of the problem be reduced?
  - Restriction/structuring of the configuration space by introducing grasp categories → **Grasp taxonomies**

Content

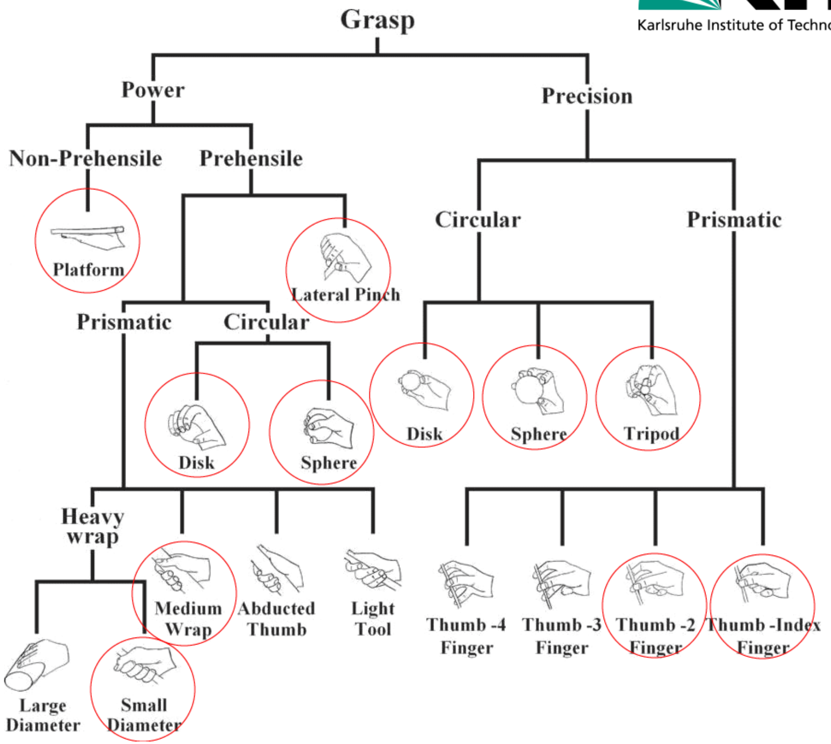
- Motivation
- **Grasp Taxonomies**
- Contact Models and Grasp Wrench Space
- Grasp Planning and Grasp Synthesis
- Examples: Grasping with ARMAR

Reducing the Problem‘s Complexity

- Introduce taxonomies
- **Benefits of a grasp taxonomy**
  - Simplification of grasp synthesis (determination of contact points on the object)
  - Fundamentals for the design of robot hands
  - Evaluation of robot hands
  - Support of autonomous grasp planning
  - Classification of (human) grasp types
- **Known grasp taxonomies**
  - In Robotics I: Cutkosky taxonomy
  - In Robotics II: Other taxonomies
    - Kamakura taxonomy, Feix taxonomy, Bullock and Dollar taxonomy, KIT Whole-Body Grasp taxonomy

Cutkosky Grasp Taxonomy

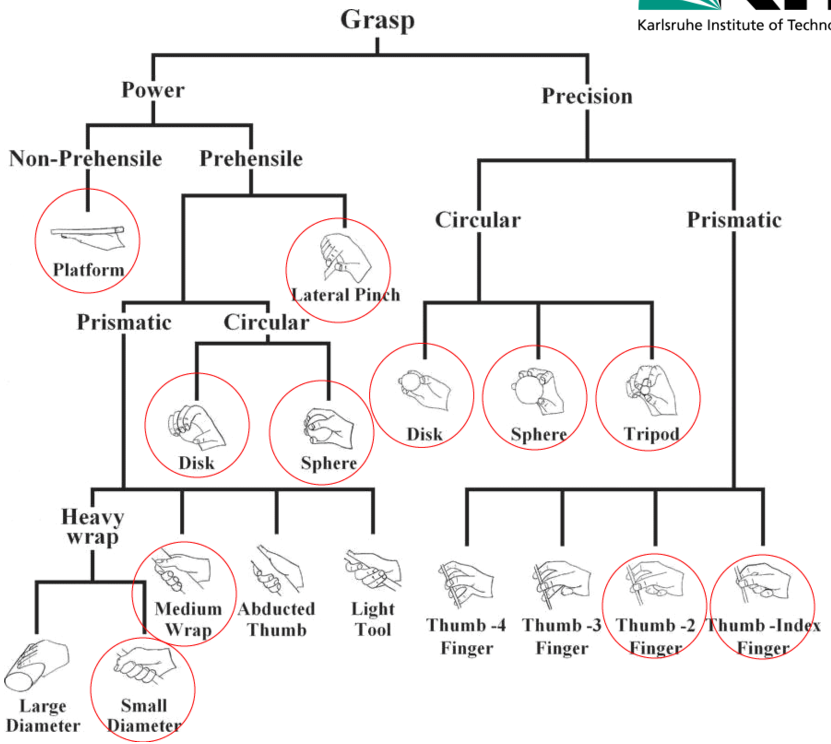
- **16 Types of Grasps**, which differ by the shape of the object and configuration of the hand (finger pose)
- **Hierarchy tree:** Grasp types are grouped together
- Top level: Differentiation between **Power and Precision grasps**



Mark Cutkosky, *On Grasp Choice, Grasp Models, and the Design of Hands for Manufacturing Tasks*. IEEE Transactions on Robotics and Automation, vol. 5, no. 3, pp. 269 – 279, 1989

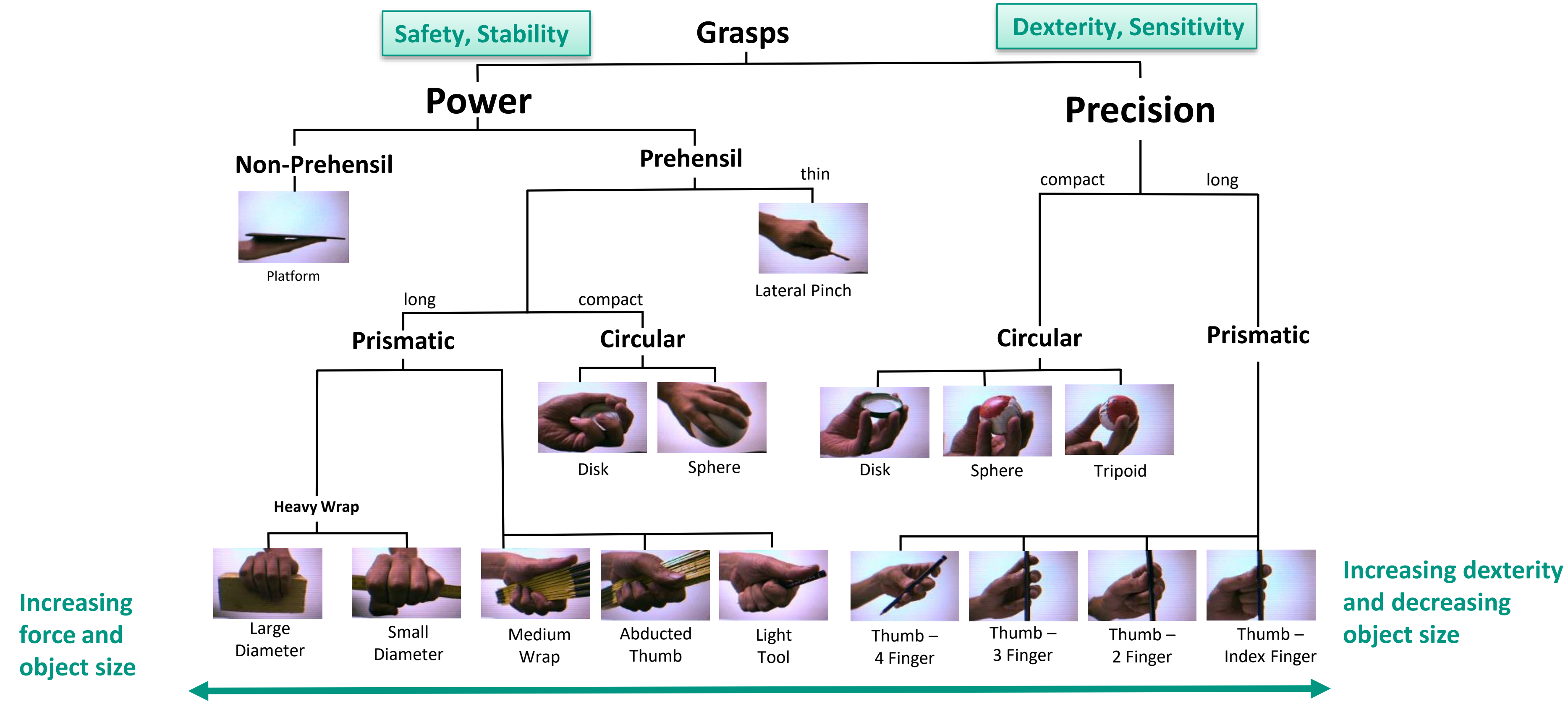
Cutkosky Grasp Taxonomy

- **Power** and **Precision Grasps**
- From observing humans perfoming manufacturing tasks
- Focus on the use of tools



Mark Cutkosky, *On Grasp Choice, Grasp Models, and the Design of Hands for Manufacturing Tasks*. IEEE Transactions on Robotics and Automation, vol. 5, no. 3, pp. 269 – 279, 1989

Cutkosky Grasp Taxonomy



Further Grasp Taxonomies

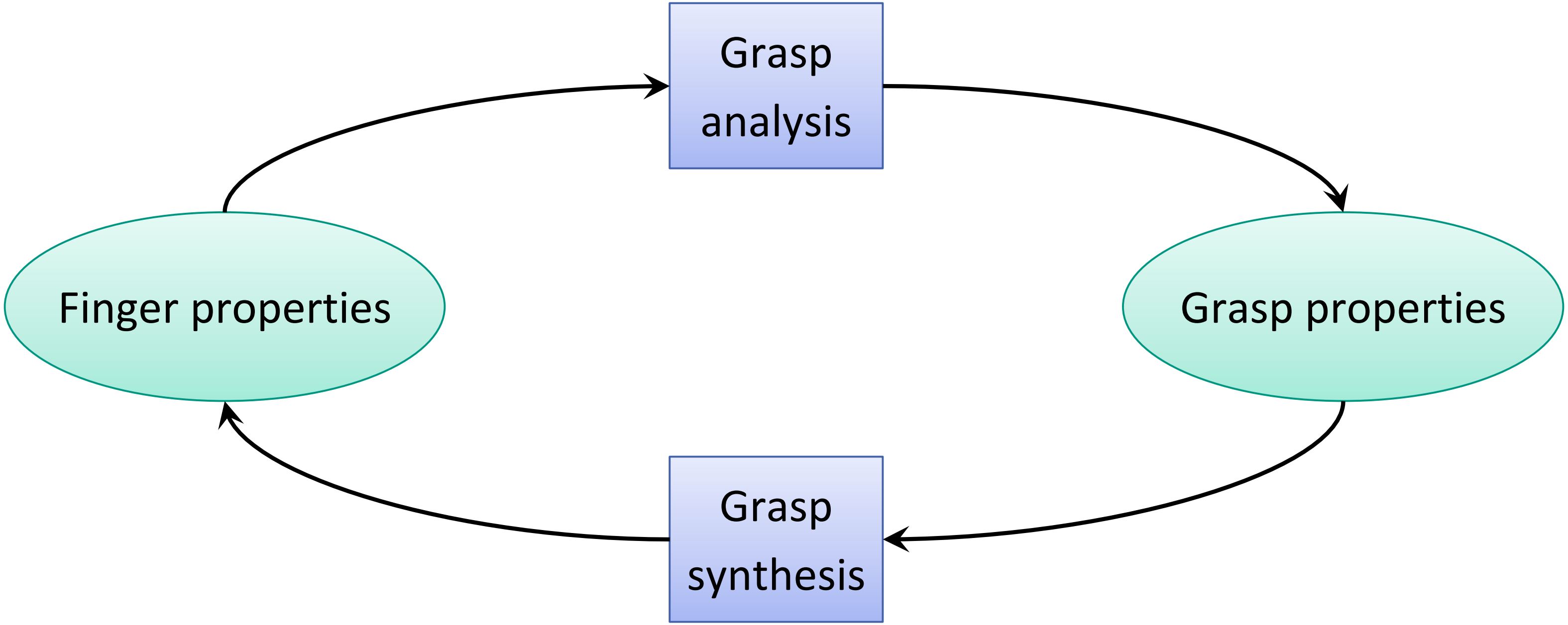
- Kamakura Taxonomy
- Feix Taxonomy
- Bullock and Dollar Taxonomy
- KIT Whole-Body Grasp Taxonomy
- More details in Robotics-II in the Summer Term

Important Factors for Grasp Generation

- Kinematics of the hand
- Grasp Representation
- Prior object knowledge
- Grasp synthesis: analytical, data-driven
- Object features: 2D, 2.5D 3D, visual, haptic, ...
- The specific task

Jeannette Bohg, Antonio Morales, Tamim Asfour, Danica Kragic, *Data-Driven Grasp Synthesis - A Survey.* IEEE Tran. on Robotics, pp. 289-309, vol. 30, no. 2, 2014

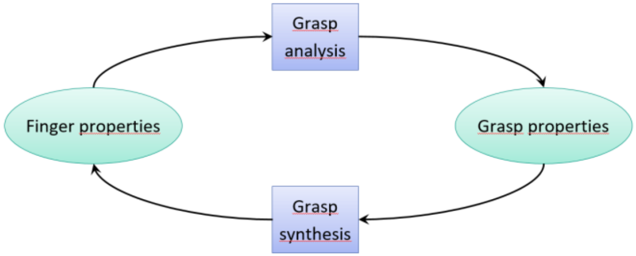
Grasp Analysis and Grasp Synthesis



Shimoga, K.B., "Robot Grasp Synthesis Algorithms: A Survey."
The International Journal of Robotics Research (1996): 15 230-266

Grasp Analysis and Grasp Synthesis

- **Grasp:**
  - A set of **contact points** on the surface of an object that restricts or compensates for potential movements of the object under the influence of external forces
- **Grasp analysis**
  - **Given**: Object and a grasp as a set of contact points
  - **Required**: Statements about the quality/stability of the grasp
- **Grasp synthesis**
  - **Given**: Object and a grasp (with certain properties)
  - **Required**: A set of contact points



Content

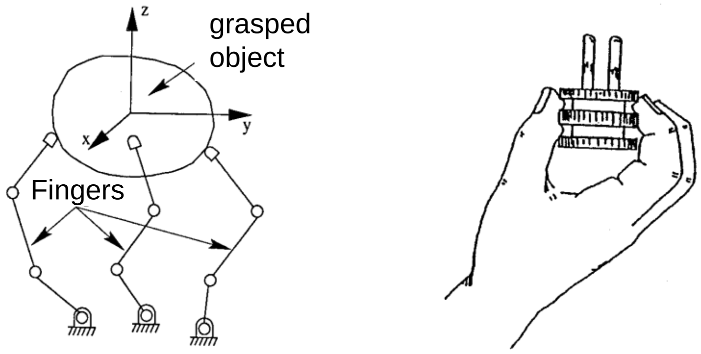
- Motivation
- Grasp Taxonomies
- **Contact Models and Grasp Wrench Space**
- Grasp Planning and Grasp Synthesis
- Examples: Grasping with ARMAR

Finger Tip Grasp Model

Model: **Grasp = Set of contact points** on the object‘s surface

⇒ Only the arrangement of the contact points on the surface needs to be determined.

- **Advantage:** Simplifies algorithms for synthesizing possible grasps of an object.
- **Disadvantage**: Ignores fundamental constraints of the grasping process.
  - Collision-free (finger, hand, arm, ...)
  - Accessibility of a grasp (approach/retraction movement)



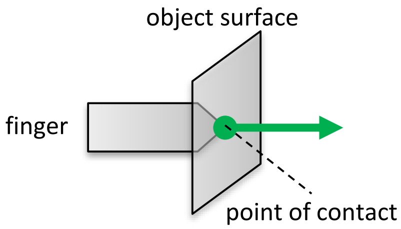
(a) (b)

Contact Models

There are different models of fingertip contact with the object surface:

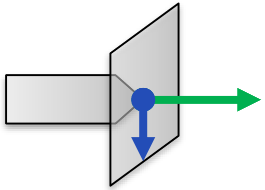
Point contact without friction

- Force only acts **normal** to the object surface.



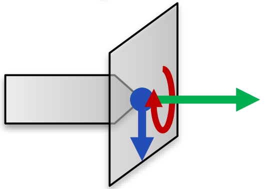
Rigid point contact with (adhesive) friction

- Force acts both normal and **tangential** to the object surface.
- Both forces are linked via Coulomb's law of friction.



Non-rigid point contact with friction (soft contact)

- Force acts both **normal** and **tangential** to the object surface.
- In addition, **axial moments** around the normal also act at the contact point.
- Coulomb's law of friction also applies.

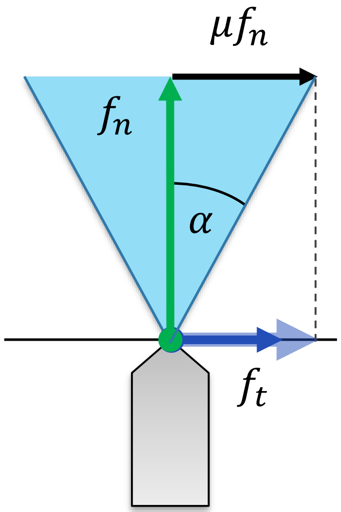


Contact Models: Coulomb's Friction Law

- Empirical Law
- Describes the relation of tangential force f_t to normal force f_n:

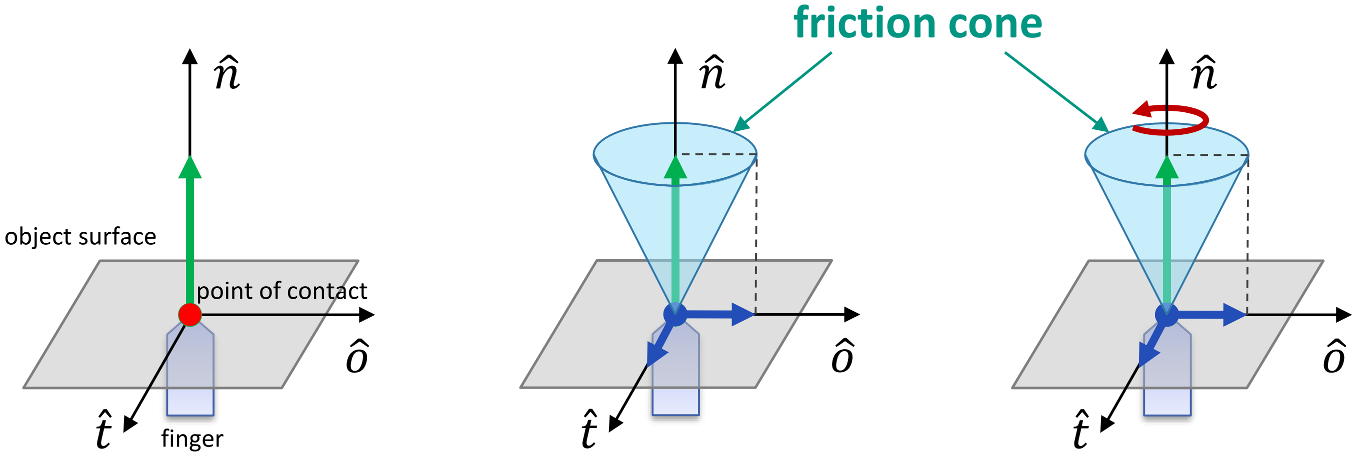
$$f_t \leq \mu \cdot f_n$$

- **Friction coefficient $\mu > 0$** (material dependent)
- For static contact:
  - $f_t = f_e < \mu \cdot f_n$
  - Tangential force acts against exercised force f_e.
- For dynamic contact (sliding):
  - $f_t = \mu \cdot f_n$
  - Tangential force acts against direction of motion.



$\alpha = \tan^{-1} \mu$

Contact Models: Geometric Interpretation



contact without friction

rigid contact with friction

soft-contact (Non-rigid contact with friction)

- The forces that can be generated by friction lie within a cone (friction cone), i.e. they satisfy (with normals in $+z$-direction):

$$\sqrt{f_x^2 + f_y^2} \leq \mu f_z \quad (f_z \geq 0)$$

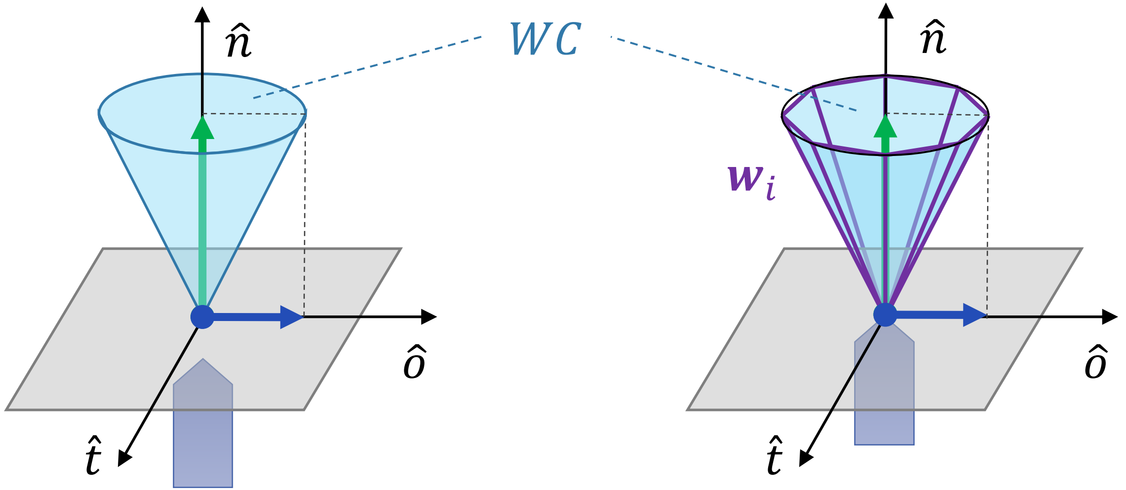
Wrench

The **forces** $\boldsymbol{f} = (f_x \quad f_y \quad f_z)$ and **rotational moments** $\boldsymbol{\tau} = (t_x \quad t_y \quad t_z)$ acting at a contact point p can be combined to form a **wrench** w (generalized force).

- Planar wrench (2D): $\boldsymbol{w} = (f_x \quad f_y \quad \tau_z)^T \in R^3$
- Spatial wrench (3D): $\boldsymbol{w} = \left(f_x \quad f_y \quad f_z \quad \tau_x \quad \tau_y \quad \tau_z\right)^T \in R^6$

engl. *wrench*

Contact Models: Approximation of the Friction Cone



Continous representation

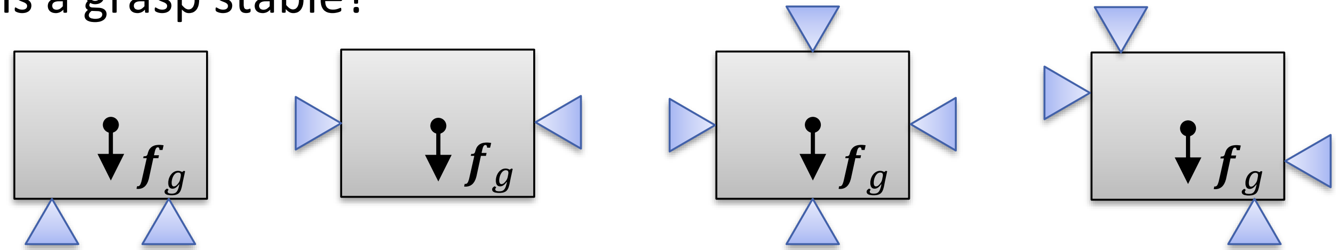
Approximated representation by an inscribed polyeder

The friction cone **(Wrench Cone)** $WC \subseteq \mathbb{R}^6$ is the positive linear hull (pos) of $\boldsymbol{w}_i$:

$$WC = \mathrm{pos}(\{\boldsymbol{w}_1, \dots, \boldsymbol{w}_n\}) = \{\sum_{i=1}^{n} k_i \boldsymbol{w}_i \mid k_i \geq 0\}$$

Stability of a Grasp

- We want to grasp objects in a stable manner, i.e. so that they do not move relative to the hand.
- When is a grasp stable?



⇒ Different concepts of stability:

- Equilibrium grasp
- Force-closure grasp
- Form-closure grasp
- Quality measures

Equilibrium Grasp

A grasp is called a **Equilibrium grasp** if the sum of all forces and moments acting on the grasped object is zero.

$\Leftrightarrow$ The object is at rest.

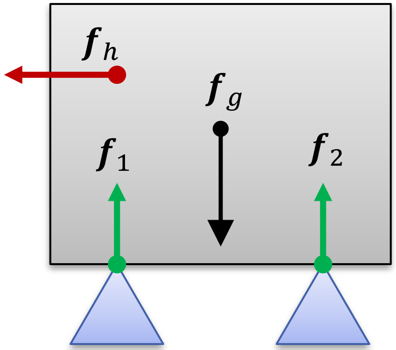
$\Leftrightarrow$ The wrenches $\boldsymbol{w}_i$ exerted by contacts exactly resist the external wrenches $\boldsymbol{w}_{\text{ext}}$ acting on the object.

$\Leftrightarrow \qquad \boldsymbol{w}_{\text{ext}} + \sum_i \boldsymbol{w}_i = 0$

Equilibrium Grasp: Example

Equilibrium Grasp: The wrenches exerted by contacts exactly resist the external wrenches acting on the object.

$$\Leftrightarrow \boldsymbol{w}_{\text{ext}} + \sum_i \boldsymbol{w}_i = 0 \qquad \boldsymbol{w}_{\text{ext}} = \begin{pmatrix} \boldsymbol{f}_{\text{ext}} \\ \boldsymbol{\tau}_{\text{ext}} \end{pmatrix}$$



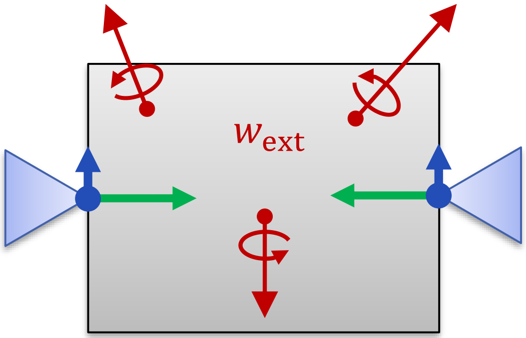
Points of contact without friction

$$\left.\begin{array}{l} \blacksquare\ \boldsymbol{f}_{\text{ext}} = \boldsymbol{f}_g \\ \blacksquare\ \boldsymbol{f}_1 + \boldsymbol{f}_2 = -\boldsymbol{f}_g \\ \blacksquare\ \tau_{\text{ext}} = 0, \qquad \tau_1 = -\tau_2 \end{array}\right\} \Rightarrow \boldsymbol{w}_{\text{ext}} + \sum_{i=1}^{2} \boldsymbol{w}_i = 0$$

- What if $\boldsymbol{f}_{\text{ext}} = \boldsymbol{f}_g + \boldsymbol{f}_h$?
- $\Rightarrow$ Grasp no longer in equilibrium.

Force-Closure Grasp (I)

- **Question:** Can the grasp counteract **any external wrenches**?
  - Object must remain in force equilibrium (relative to the hand).
  - Required to be able to manipulate the object effectively.



- **Assumption:** Contacts can exert arbitrarily large wrenches.
- Resulting question: Can the grasp generate **arbitrary wrenches**?
- In the following, consider rigid point contacts with friction.

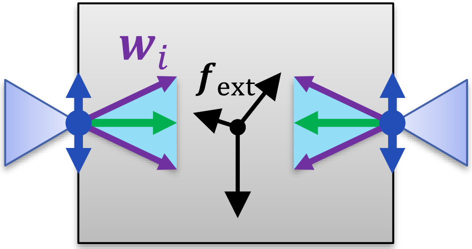
Force-closure Grasp (II)

- **Assumption:** Contacts can exert arbitrary wrenches.
- Question: Can the grasp generate arbitrary wrenches?

Let $\boldsymbol{w}_{i1}, \dots, \boldsymbol{w}_{im} \in \mathbb{R}^6$ be the vectors which approximate the wrench cone of contact i.

Wrench cone $WC_i \subseteq \mathbb{R}^6$ is the positive linear hull of $\boldsymbol{w}_{ij}$:

$$WC_i = \mathrm{pos}(\{\boldsymbol{w}_{i1}, \dots, \boldsymbol{w}_{im}\}) = \{\textstyle\sum_{j=1}^{m} k_j \boldsymbol{w}_{ij} \mid k_j \geq 0\}$$



Rigid point contacts with friction

WC_i is the set of wrenches that can be generated by contact i.

The **total set of wrenches** WC that can be generated by all contacts is the positive linear hull of each WC_i:

$$WC = \mathrm{pos}(\{WC_i\}) = \{\textstyle\sum_i \sum_{j=1}^{m} k_{ij} \boldsymbol{w}_{ij} \mid k_{ij} \geq 0\}$$

Force-Closure Grasp (III)

The total set of wrenches that can be generated by all contacts is the positive linear hull of each WC_i:

$$WC = \text{pos}(\{WC_i\}) = \left\{\sum_i \sum_{j=1}^{m} k_{ij}\boldsymbol{w}_{ij} \,\middle|\, k_{ij} \geq 0\right\}$$

The grasp is in **force-closure** if WC contains a sphere with radius $\varepsilon > 0$ around the origin.
($WC = \mathbb{R}^d$, as WC is a linear hull.)

⇒ **Then wrenches can be created in all directions (dimensions of the wrench space).**

Assumption: Contacts can exert forces of any magnitude.

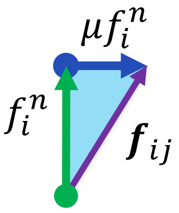
⇒ Wrenches of any magnitude can be generated.

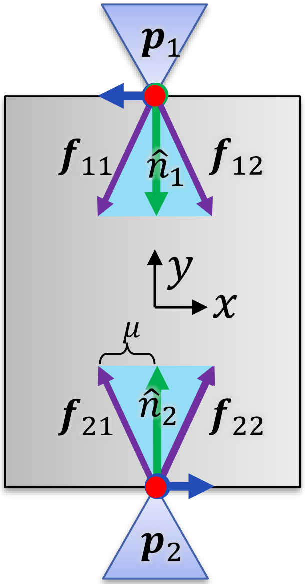
⇒ Any external wrenches can be resisted ⇒ force-closure grasp

Very large normal forces can be required to generate the necessary tangential forces.

Force-Closure Grasps: Example in 2D (I)

- Two contact points with positions $\boldsymbol{p}_1, \boldsymbol{p}_2$ and normals $\hat{n}_1, \hat{n}_2$
  - $\boldsymbol{p}_1 = (0 \quad 3)^T \qquad \hat{n}_1 = (0 \quad -1)^T$
  - $\boldsymbol{p}_2 = (0 \quad -3)^T \qquad \hat{n}_2 = (0 \quad 1)^T$
- Both contacts have friction coefficients $\mu > 0$.
  - Unity Force: $f_i^n := 1 \Rightarrow f_i^t = \mu f_i^n = \mu$
  - $\boldsymbol{f}_{11} = (-\mu \quad -1)^T, \ \boldsymbol{f}_{12} = (\mu \quad -1)^T$
  - $\boldsymbol{f}_{21} = (-\mu \quad 1)^T, \quad \boldsymbol{f}_{22} = (\mu \quad 1)^T$
- Related torques: $\tau_{ij} = \boldsymbol{d}_i \times \boldsymbol{f}_{ij}$
  - Here: $\boldsymbol{d}_i = \boldsymbol{p}_i$, since the origin is in the object's center of gravity.
  - $\tau_{11} = 3\mu, \qquad \tau_{12} = -3\mu$
  - $\tau_{21} = -3\mu, \qquad \tau_{22} = 3\mu$



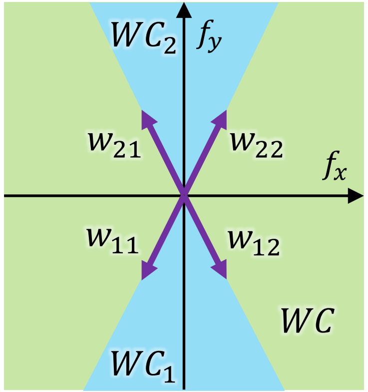


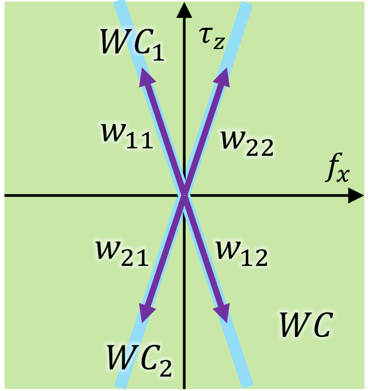
2D cross product:

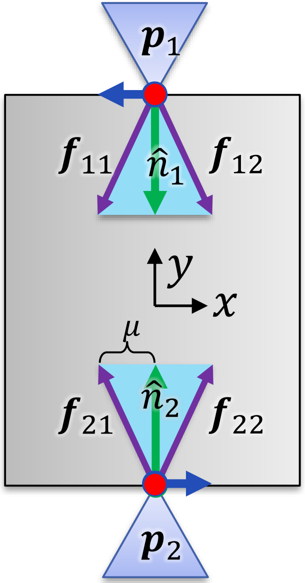
$$\begin{pmatrix} a_1 \\ a_2 \end{pmatrix} \times \begin{pmatrix} b_1 \\ b_2 \end{pmatrix} = a_1 b_2 - a_2 b_1$$

Force-Closure Grasps: Example in 2D (II)

- Wrenches (edges of the friction cones):
  - $\boldsymbol{w}_{11} = (-\mu \quad -1 \quad 3\mu)^T, \qquad \boldsymbol{w}_{12} = (\mu \quad -1 \quad -3\mu)^T$
  - $\boldsymbol{w}_{21} = (-\mu \quad 1 \quad -3\mu)^T, \qquad \boldsymbol{w}_{22} = (\mu \quad 1 \quad 3\mu)^T$
- Projections of the 3D Wrench space onto subspaces:

f_x, f_y-Space: 

f_x, τ_z-Space: 



- Friction cones span $\mathbb{R}^3$.
- $\Rightarrow$ **Force-closure Grasp.**

Grasp Matrix (I)

Let $\boldsymbol{w}_1, \dots, \boldsymbol{w}_m \in \mathbb{R}^6$ be the wrenches which span the wrench cones of all contacts.

The **grasp matrix** G contains the wrenches of all wrench cones as columns.

$$G = \begin{pmatrix} \boldsymbol{w}_1 & \cdots & \boldsymbol{w}_m \end{pmatrix} \in \mathbb{R}^{6\times m}$$

For $\boldsymbol{k} \in \mathbb{R}^m$, $\boldsymbol{w} = G \cdot \boldsymbol{k} \in \mathbb{R}^6$ is a linear combination of wrenches in G.

The grasp can withstand an external wrench $\boldsymbol{w}_{\text{ext}} \in \mathbb{R}^6$ if

- there is some $\boldsymbol{k} = (k_1 \quad \dots \quad k_m) \in \mathbb{R}^m$ with $k_j \geq 0$,
- such that $G \cdot \boldsymbol{k} + \boldsymbol{w}_{\text{ext}} = 0$

(Then $-w_{\text{ext}}$ is inside of the positive linear hull of $\boldsymbol{w}_j$.)

Grasp Matrix (II)

The **grasp matrix** G contains the wrenches of all wrench cones as columns.

$$G = \begin{pmatrix} \boldsymbol{w}_1 & \cdots & \boldsymbol{w}_m \end{pmatrix} \in \mathbb{R}^{6 \times m}$$

The grasp can withstand **arbitrary** external wrenches $\boldsymbol{w}_{\text{ext}} \in \mathbb{R}^6$ if

- G has full rank, i.e. $\operatorname{rang}(G) = 6$, and
- there is some $\boldsymbol{k} = (k_1 \quad \dots \quad k_m) \in \mathbb{R}^m$ with $k_j > 0$ such that $G \cdot \boldsymbol{k} = \boldsymbol{0}$.

The respective grasp is then in *force-closure*

Form-Closure Grasps

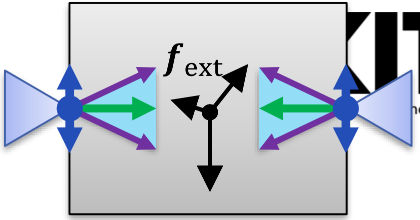
Question: What if no friction exists?

- → Point contact without friction
- Then: Contacts only excert forces along their normal

Can the grasp then continue to counteract any external wrenches?

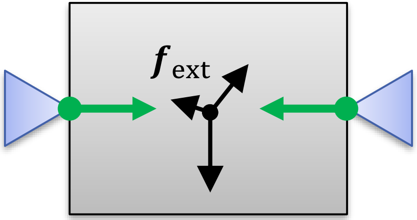
If so, the grasp is **form-closure**.

  - Similar to force-closure grasps, …
  - but **only forces along the normal** are taken into account.
  - Form-closure is a stricter condition than force-closure.



Rigid point contacts **with** friction

$\mu = 0$

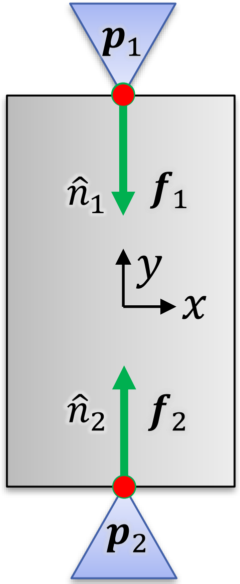


Rigid point contacts **without** friction

Form-closure = **Force-**closure **without** friction

Form-Closure Grasps: Example in 2D (I)

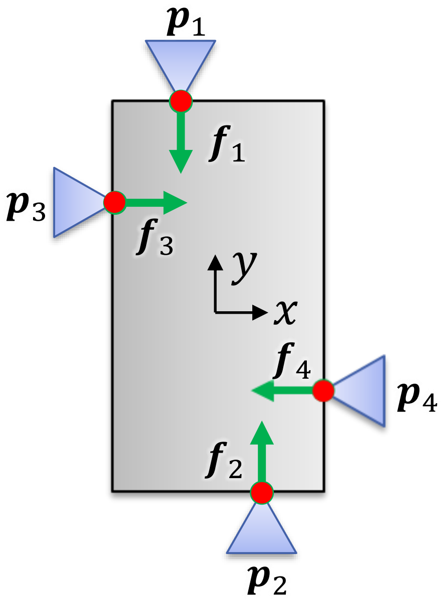
- Two contacts with positions $\boldsymbol{p}_1, \boldsymbol{p}_2$ and normals $\hat{n}_1, \hat{n}_2$
  - $\boldsymbol{p}_1 = (0 \quad 3)^T \qquad \hat{n}_1 = (0 \quad -1)^T$
  - $\boldsymbol{p}_2 = (0 \quad -3)^T \qquad \hat{n}_2 = (0 \quad 1)^T$
- No friction $\Rightarrow$ only two forces:
  - $\boldsymbol{f}_1 = (0 \quad -1)^T, \qquad \boldsymbol{f}_2 = (0 \quad 1)^T$
- Related torques : $\tau_i = \boldsymbol{d}_i \times \boldsymbol{f}_i$
  - Here: $\boldsymbol{d}_i = \boldsymbol{p}_i$, since the origin is in the object's center of gravity.
  - $\tau_1 = 0 \cdot (-1) - (-1) \cdot 0 = 0, \qquad \tau_2 = 0$ (analogously)
- Generated wrenches: ($\boldsymbol{w}_i = (f_x \quad f_y \quad \tau_z)^T$)
  - $\boldsymbol{w}_1 = (0 \quad -1 \quad 0)^T, \quad \boldsymbol{w}_2 = (0 \quad 1 \quad 0)^T$



- Contacts cannot exert forces along the x-axis nor torques around the z-axis.
- $\Rightarrow$ **Grasp is not form-closure**

Form-Closure Grasps: Example in 2D (II)

- Move contacts, add two more:
  - $\boldsymbol{p}_1 = (-1 \quad 3)^T$, $\quad \boldsymbol{f}_1 = (0 \quad -1)^T$
  - $\boldsymbol{p}_2 = (1 \quad -3)^T$, $\quad \boldsymbol{f}_2 = (0 \quad 1)^T$
  - $\boldsymbol{p}_3 = (-2 \quad 2)^T$, $\quad \boldsymbol{f}_3 = (1 \quad 0)^T$
  - $\boldsymbol{p}_4 = (2 \quad -2)^T$, $\quad \boldsymbol{f}_4 = (-1 \quad 0)^T$
- Generated torques and wrenches:
  - $\tau_1 = 1 \quad \Rightarrow \boldsymbol{w}_1 = (0 \quad -1 \quad 1)^T$
  - $\tau_2 = 1 \quad \Rightarrow \boldsymbol{w}_2 = (0 \quad 1 \quad 1)^T$
  - $\tau_3 = -2 \quad \Rightarrow \boldsymbol{w}_3 = (1 \quad 0 \quad -2)^T$
  - $\tau_4 = -2 \quad \Rightarrow \boldsymbol{w}_4 = (-1 \quad 0 \quad -2)^T$



- $\Rightarrow WC = \text{pos}(\{\boldsymbol{w}_i\}) = \mathbb{R}^3$, i.e. the $\boldsymbol{w}_i$ completely cover the wrench space.
- $\Rightarrow$ **Form-closure Grasp.**

Dimension of the Grasp Matrix

The **grasp matrix** G contains the wrenches of all friction cones as columns:

$$G = \begin{pmatrix} \boldsymbol{w}_1 & \cdots & \boldsymbol{w}_m \end{pmatrix} \in \mathbb{R}^{6\times m}$$

- m depends on the number of contacts n and the contact model.

Rigid point contact without friction:

- **2D & 3D:** Each contact generates exactly one wrench. $\Rightarrow m = n$

Rigid point contact with friction :

- **2D:** Friction triangle spanned by exactly two wrenches. $\Rightarrow m = 2n$
- **3D:** Friction cone approximated by k wrenches (e.g. k=8). $\Rightarrow m = kn$

Number of Contacts Required

Karlsruhe Institute of Technology

In general: **To span $\mathbb{R}^d$, $d+1$ vectors are required.**

	Planar (Wrench-Space: $\mathbb{R}^3$)	**Spatial** (Wrench-Space: $\mathbb{R}^6$)

Form-closure (without friction)	▪ At least 4 contacts
	▪ Maximum 6 contacts
		▪ At least 7 contacts
		▪ Maximum number:
		  ▪ Arbitrary objects: 12
		  ▪ Polyeder: 7

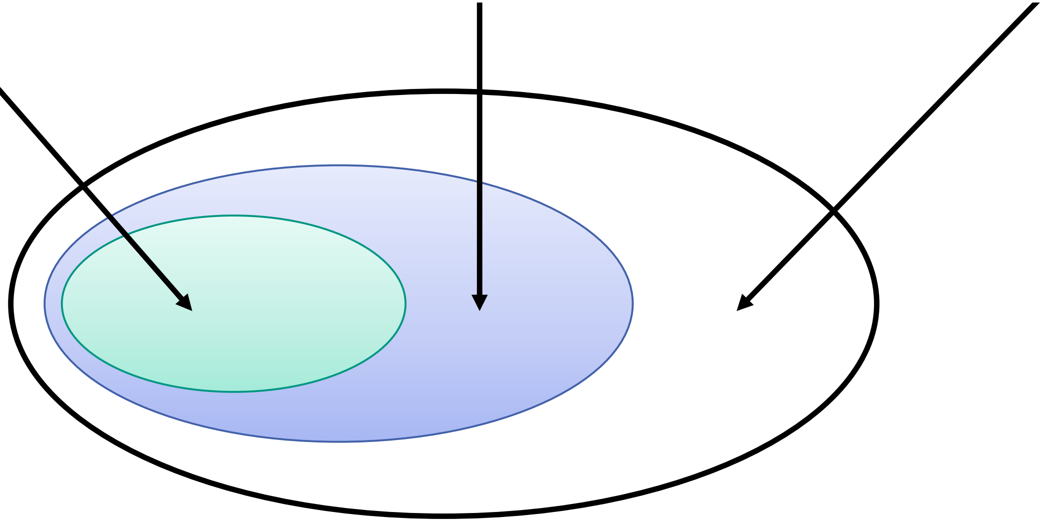
Force-closure (with friction)	▪ At least 2 contacts ▪ Maximum 3 contacts	▪ At least 4 contacts

There are "*exceptional objects*", e.g. the circle in 2D or the sphere in 3D, for which no form-closure grasp (without friction) exist.

H2T

Grasp Type Hierarchy

Form-closure grasp	$\subseteq$	Force-cloure grasp	$\subseteq$	Equilibrium grasps



Force- and Form-Closure Grasps

Karlsruhe Institute of Technology

Force-Closure:

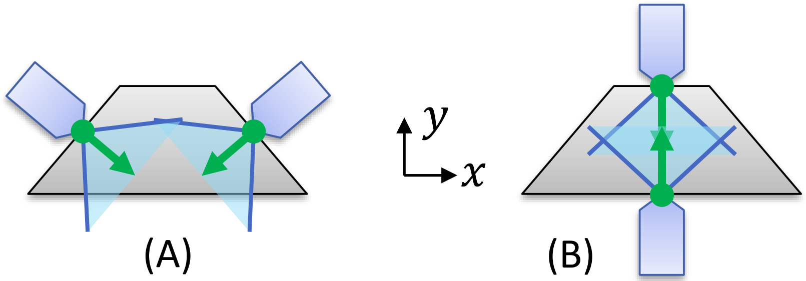
- The kinematics of the hand can actively generate forces to resist an external disturbance.
- Friction forces also play a role.

Form-Closure:

- The contacts prevent the object from moving.
- No friction forces are required/taken into account.

- Force closure is possible with fewer contact points and is therefore used in **precision grasps**. However, it requires control of the forces occurring internally in a grasp.
- Form closure is a stronger condition than force closure and is often used when performing **power grasps**.

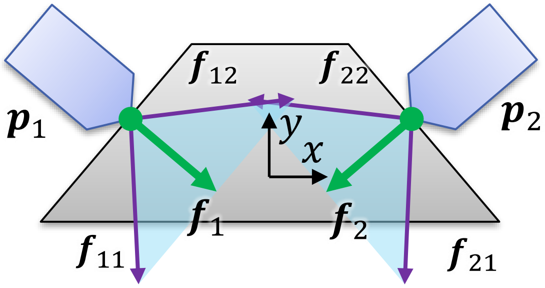
Grasp Quality



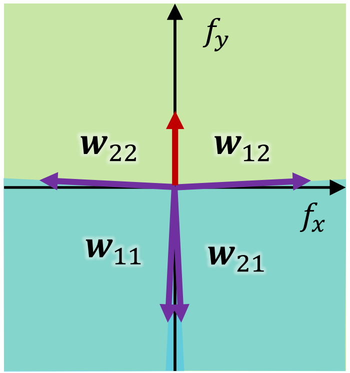
- Observe Grasps (A) and (B):
- Which grasp is **force-cluser grasp**?
  - (A) and (B)
- Are both grasps **equally good**?
  - With (A), high normal forces have to be excerted to generate friction forces in y-direction.
  - With (B), it is easier to excercise forces in all directions.
- How can this be **quantified**?

Grasp Quality: Grasp (A)

- Positions and normal forces
  - $\boldsymbol{p}_1 = (-4 \quad 1)^T$, $\qquad \boldsymbol{f}_1^n = \frac{1}{\sqrt{2}}(1 \quad -1)^T$
  - $\boldsymbol{p}_2 = (4 \quad 1)^T$, $\qquad \boldsymbol{f}_2^n = \frac{1}{\sqrt{2}}(-1 \quad -1)^T$
- $\mu = 1.1$, friction forces $f_{ij} = f_i^n + \mu \cdot f_i^{t_j}$
  - $f_{11} = \frac{1}{\sqrt{2}}(-0.1 \quad -2.1)^T$, $f_{12} = \frac{1}{\sqrt{2}}(2.1 \quad 0.1)^T$
  - $f_{21} = \frac{1}{\sqrt{2}}(0.1 \quad -2.1)^T$, $\quad f_{22} = \frac{1}{\sqrt{2}}(-2.1 \quad 0.1)^T$
- Generated torques and wrenches:
  - $\tau_{11} = \frac{8.5}{\sqrt{2}} \qquad \Rightarrow \boldsymbol{w}_{11} = \frac{1}{\sqrt{2}}(-0.1 \quad -2.1 \quad 8.5)^T = (-0.0707 \quad -1.48 \quad 6.01)^T$
  - $\tau_{12} = \frac{-2.5}{\sqrt{2}} \qquad \Rightarrow \boldsymbol{w}_{12} = \frac{1}{\sqrt{2}}(2.1 \quad 0.1 \quad -2.5)^T = (1.48 \quad \mathbf{0.0707} \quad -1.76)^T$
  - $\tau_{21} = \frac{-8.5}{\sqrt{2}} \qquad \Rightarrow \boldsymbol{w}_{21} = \frac{1}{\sqrt{2}}(0.1 \quad -2.1 \quad -8.5)^T = (0.0707 \quad -1.48 \quad -6.01)^T$
  - $\tau_{22} = \frac{2.5}{\sqrt{2}} \qquad \Rightarrow \boldsymbol{w}_{22} = \frac{1}{\sqrt{2}}(-2.1 \quad 0.1 \quad 2.5)^T = (-1.48 \quad \mathbf{0.0707} \quad 1.76)^T$

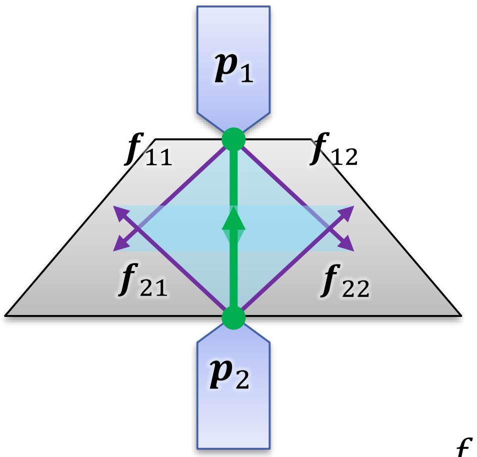


f_x, f_y-space

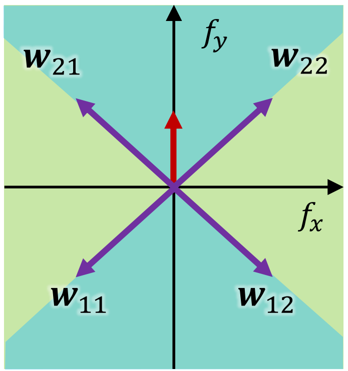


Grasp Quality: Grasp (B)

- Positions and normal forces
  - $\boldsymbol{p}_1 = (0 \quad 2)^T$, $\boldsymbol{f}_1^n = (0 \quad -1)^T$
  - $\boldsymbol{p}_2 = (0 \quad -1)^T$, $\boldsymbol{f}_2^n = (0 \quad 1)^T$
- $\mu = 1.1$, friction forces $f_{ij} = f_i^n + \mu \cdot f_i^{t_j}$
  - $f_{11} = (-1.1 \quad -1)^T$, $f_{12} = (1.1 \quad -1)^T$
  - $f_{21} = (-1.1 \quad 1)^T$, $f_{22} = (1.1 \quad 1)^T$
- Generated torques and wrenches:
  - $\tau_{11} = 2.2 \quad \Rightarrow \boldsymbol{w}_{11} = (-1.1 \quad -1 \quad 2.2)^T$
  - $\tau_{12} = -2.2 \quad \Rightarrow \boldsymbol{w}_{12} = (1.1 \quad -1 \quad -2.2)^T$
  - $\tau_{21} = -1.1 \quad \Rightarrow \boldsymbol{w}_{21} = (-1.1 \quad 1 \quad -1.1)^T$
  - $\tau_{22} = 1.1 \quad \Rightarrow \boldsymbol{w}_{22} = (1.1 \quad 1 \quad 1.1)^T$

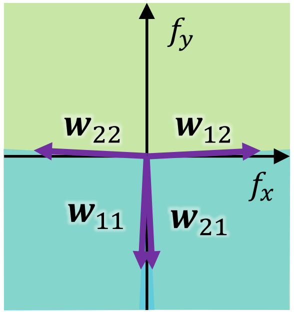


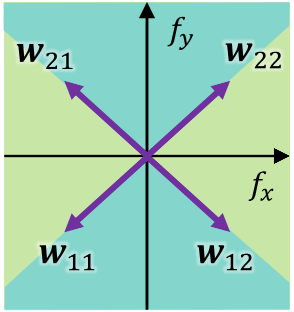
f_x, f_y-space

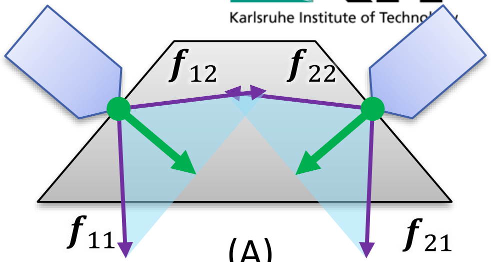


Grasp Quality: Grasp-Wrench-Space

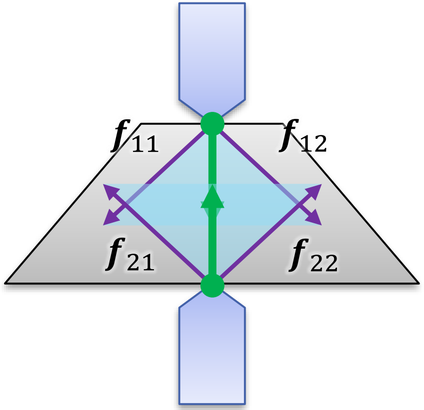
Compare f_x, f_y-layer of the wrench spaces:







(A)



(B)

How can wrench $\boldsymbol{w} = (0 \quad 1 \quad 0)^T$ be generated?

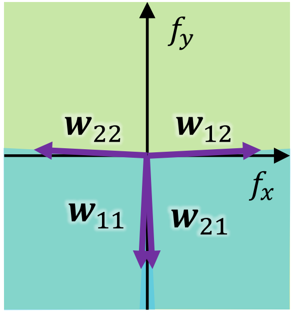
- (A) e.g: $\boldsymbol{w} = 7.1\ \boldsymbol{w}_{12} + 7.1\ \boldsymbol{w}_{22}$
- (B) e.g: $\boldsymbol{w} = 0.5\ \boldsymbol{w}_{21} + 0.5\ \boldsymbol{w}_{22}$

$\Rightarrow$ (B) requires less force to generate the same wrench

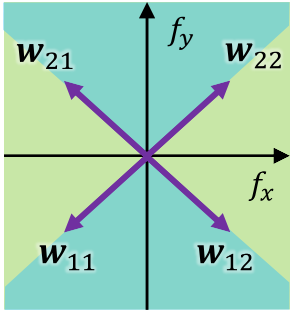
Grasp Quality: Grasp-Wrench-Space-Metric

- Force closure implies that any wrench can be generated when forces of any magnitude are applied.
- What if …
  - some wrenches are “difficult” to produce?
  - not arbitrarily large forces can be applied?
- Previous condition for coefficients of forces: $k_i \geq 0$
- Now: Imaginary upper bound: $k_i \in [0, f_{\max}]$

- ⇒ Friction cones are no longer infinite
- ⇒ Linear hull becomes **convex hull**



(A)



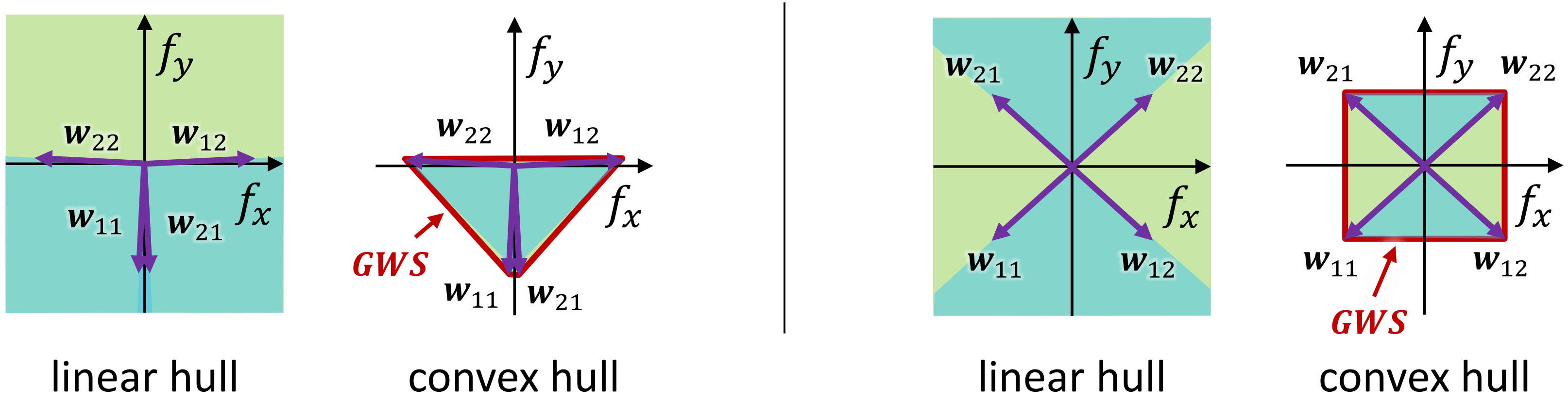
(B)

Grasp Quality: Grasp-Wrench-Space

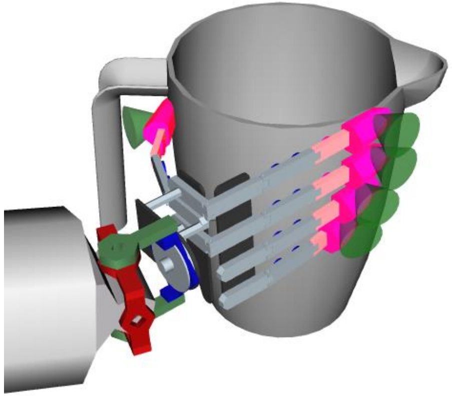
Let $\boldsymbol{w}_1, \dots, \boldsymbol{w}_m \in \mathbb{R}^6$ be the vectors that approximate the friction cones of all contacts.

The **Grasp-Wrench-Space** GWS is the convex hull of the $\boldsymbol{w}_i$

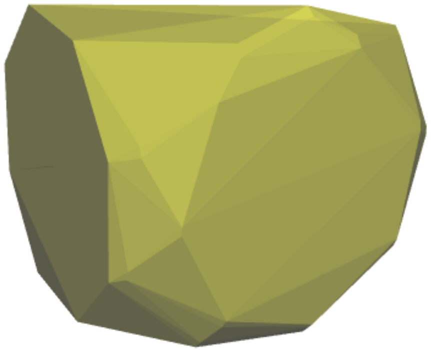
$$GWS = \operatorname{conv}(\{\boldsymbol{w}_i\}) = \left\{ \sum_{i=1}^{m} k_i \boldsymbol{w}_i \;\middle|\; k_i \geq 0 \text{ and } \sum_{i=1}^{m} k_i = 1 \right\}$$



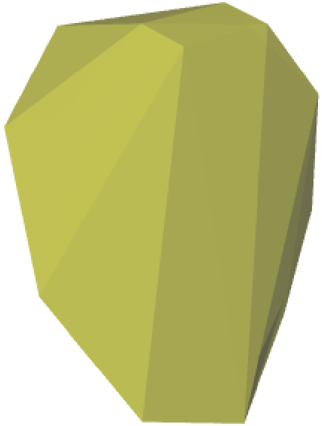
Visualization of the Grasp-Wrench-Space in 3D



Grasp in
Cartesian space



Force-Sub-Space of
the GWS



Torque-Sub-Space
of the GWS

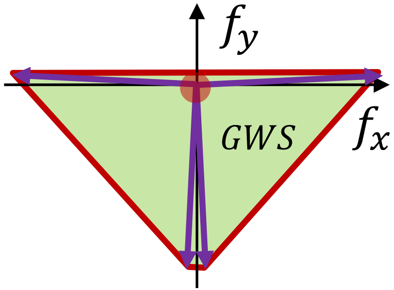
N. Vahrenkamp, T. Asfour and R. Dillmann, *Simultaneous Grasp and Motion Planning*, IEEE Robotics and Automation Magazine, Vol. 19, No. 2, pp. 43 - 57, June, 2012

Grasp Quality: Grasp-Wrench-Space

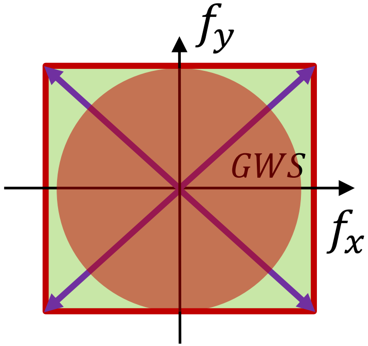
The **Grasp-Wrench-Space** GWS is the convex hull of the $\boldsymbol{w}_i$

$$GWS = \text{conv}(\{\boldsymbol{w}_i\}) = \{\sum_{i=1}^{m} k_i \boldsymbol{w}_i \mid k_i \geq 0 \text{ and } \sum_{i=1}^{m} k_i = 1\}$$

How could one define a **measure for the quality** of a grasp using GWS?



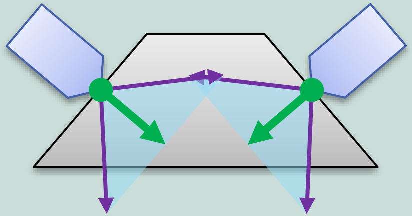
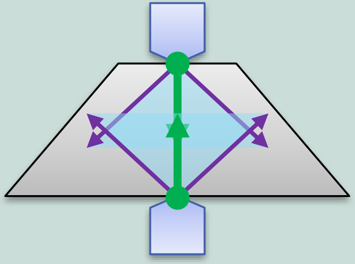
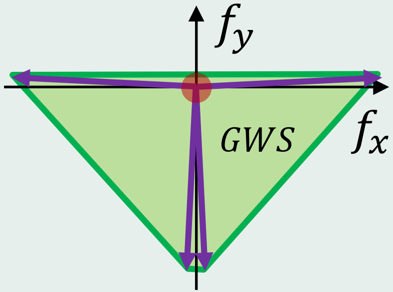
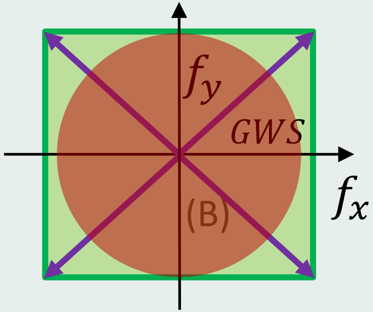
> The **ε-metric** is the **radius of the largest sphere** around the origin of the GWS that is still completely contained in GWS.
>
> It is sometimes also called the Grasp-Wrench-Space metric.



Intuition:

- ε is the magnitude of the smallest wrench which breakes the grasp.
- The grasp withstands all wrenches with a magnitude less than ε.
- If $\varepsilon > 0 \rightarrow$ force-clusure grasp
- The larger ε, the more „stable“ the grasp.

Grasp-Wrench-Space-Metric: Examples

Grasp	(A)	(B)
2D workspace		
f_x, f_y-Plane of the Grasp-Wrench-Space with ε-sphere around the origin		
ε_{xy}-metric in f_x, f_y-space	$\varepsilon_{xy} = 0.0707$	$\varepsilon_{xy} = 1$

Grasp Wrench Space Metric: Remarks

Problem 1: Forces and torques have different units.

- ε-metric is defined over entire GWS, i.e. forces and torques.
- How do you scale forces to torques?

Possible solutions:

- Choose the characteristic length r of the object (e.g. largest dimension).
- Convert torques τ to forces τ/r.

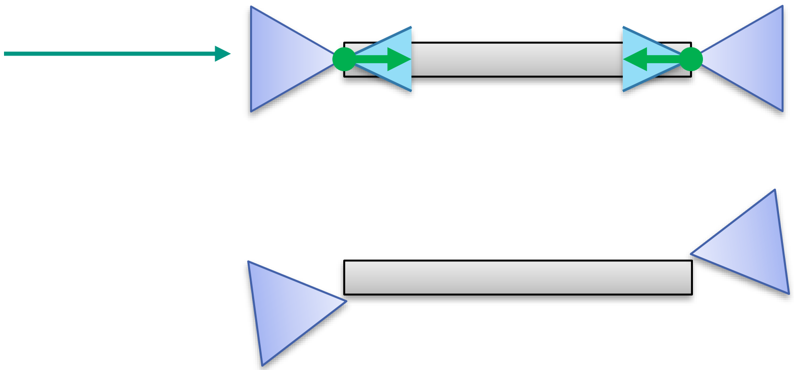
Problem 2: Torques depend on the choice of reference point (OCS).

- Thus, the ε-metric also depends on the choice of the reference point.

Possible solution:

- Choose the origin of the OCS near the geometric center or center of mass

Grasp-Wrench-Space-Metric: Annotations



Problem 3: Consider this grasp

- ε-metric provides a very good value.
- However, inaccuracies in execution can lead to a very poor grasp.

Possible solution:

- Calculate the ε-metric multiple times for randomly or systematically perturbed object poses.

Option 1:

- Select the minimum, 10^{th} percentile, median, …, of the ε-metric from the random poses.

Option 2:

- Use the proportion of grasps with $\varepsilon > 0$ (force-closure) → "Grasp success rate"

Grasp Wrench Space Metric: Summary

The ε-metric …

- is a general and widely used **quality metric for fingertip grasps**
- models how **resilient** a grasp is to external wrenches
- can be used to assess grasps for **force closure**
- requires **model knowledge**
  - geometry of the fingers and object, friction coefficients, contact positions and contact normals

The optimal grasp also depends on the **application**!

- Certain forces occur predominantly or exclusively
- The object must be manipulated in a specific way
- The object must be handed over or used for a task (e.g., hammering)
- $\rightarrow$ This is not considered in the basic ε-metric

Metrics for grasp quality are still a **research topic**!

Content

- Motivation
- Grasp Taxonomies
- Contact Models and Grasp Wrench Space
- **Grasp Planning and Grasp Synthesis**
- Examples: Grasping with ARMAR

Classification of grasp planning systems

Grasp planning systems can be classified based on the following criteria:

- Type of **gripper**: two-finger, three-finger, multi-finger grippers, ...
- Type of underlying **grasp planning algorithms**
  - Geometry-based: use of CAD data
  - Physics-based: consider forces and torques
- Type of geometry of the **object** to be grasped: polygons, polyhedra, arbitrary geometry...
- Type of **scenes** to be manipulated
  - Known: The position and orientation of all objects in the scene are known
  - (Partially) unknown: Unknown objects appear in unknown poses
- Type of **sensing**: No sensing, tactile sensing, visual sensing, ...

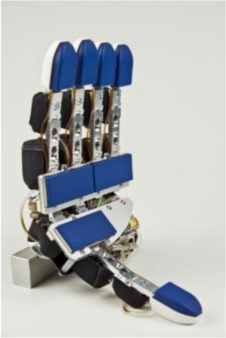
Search space of grasping

- If a valid grasp is to be planned, **the dimension of the search space** (for both the physical and the geometric approach) is

$$\boldsymbol{6 + n}$$

  - 6 degrees of freedom for the position and orientation of the hand in space
  - n Number of configuration parameters of the fingers

- Parallel jaw gripper:
  Search space of the dimension 7 (1 for the gripper + 6)
- ARMAR-III hand:
  Search space of the dimension 14 (8 for the gripper + 6)
- Human hand:
  Search space of the dimension 27 (21 for the gripper + 6)



Object categories for grasping

Known objects

- Known object geometry (we have a complete object model)
- **Approach:** Use a grasp planner that works with known object geometries
- Domain for classical grasp planning
- **Difficult**

Familiar objects

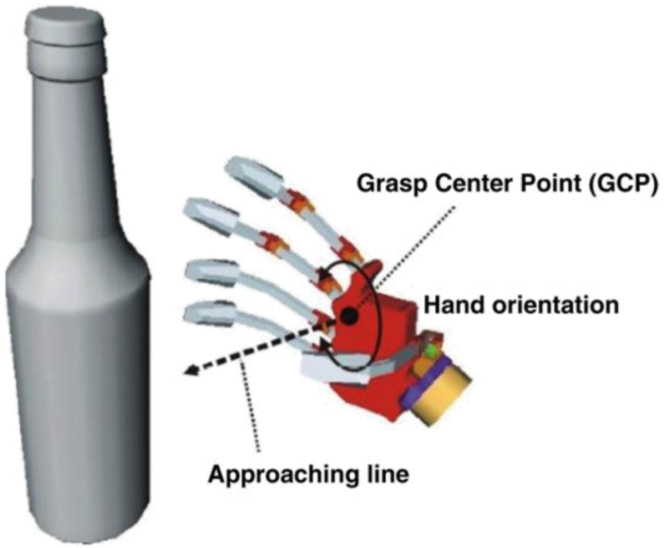
- The concrete object geometry is not known (e.g., only "object is of type bottle")
- **Approach:** Transfer knowledge from known class elements to the object to be grasped
- **More difficult**

Unknown objects

- Neither object geometry nor object class is known
- **Problems** (among others)
  - Dealing with incomplete sensor data (Stereo Vision, RGB-D, Laser scans, haptic data)
  - Segmentation of the object from the background
  - Creating a (partial) object model
- **Approaches** (among others)
  - Sensor fusion
  - Interactive perception (moving the object)
- **Most difficult!**

Definition of a grasp

- A grasp can be defined by
  - The **Grasp Center Point (GCP)** on the object where the Tool Center Point (TCP) should be aligned (approach point)
  - The **approach vector** describes the angle at which the hand approaches the grasp center point (approach direction)
  - **Orientation of the wrist** of the robot hand
  - **Initial finger configuration (pre-shape)**



Algorithms for Grasp Synthesis

Requirements

- Definition of hand kinematics
- Contact model (usually: point contact with friction)
- Object model (usually: fully known)

Algorithms for grasp synthesis with known objects

- ***Survey***: J. Bohg, A. Morales, T. Asfour and D. Kragic, *Data-Driven Grasp Synthesis – A Survey*, IEEE Transactions on Robotics, Vol. 30, No. 2, pp. 289 - 309, 2014
- **Randomized methods using forward planning** [Diankov 2010, Berenson et al. 2007, Vahrenkamp et al. 2011]
- ***Grasp synthesis on object parts*** [Goldfelder et al. 2007, Huebner et. al 2008, …], Representation of objects by (approximated) object parts, grasp synthesis on the object parts.
- Grasp planning using the **medial axis or skeleton of the objects** [Przybylski et al. 2010, Vahrenkamp et al. 2018]
- Grasp synthesis based on **eigengrasps** (principal grasps)[Ciocarlie et al. 2007, Bicchi et al. 2011, …]
- Grasp planning based on ***Independent Contact Regions*** [Roa et al. 2009]
- ….

Grasp Synthesis using Forward Planning(I)

Approach

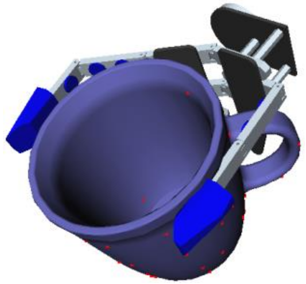
- Planning in simulation
- Determination of the approach point and approach direction
- The hand approaches the object until contact is detected
- The fingers close around the object until contact is made
- Evaluation of contacts between the hand and the object
  Interesting: Is the grasp in **force-closure**?

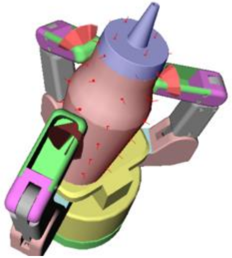
Advantages

- Forward planning is similar to executing a grasp on a real robot.
- Grasps that have been successfully evaluated in simulation can likely be performed with a real robot; however we have to deal with the sim2real gap problem

Grasp synthesis using forward planning– Algorithm

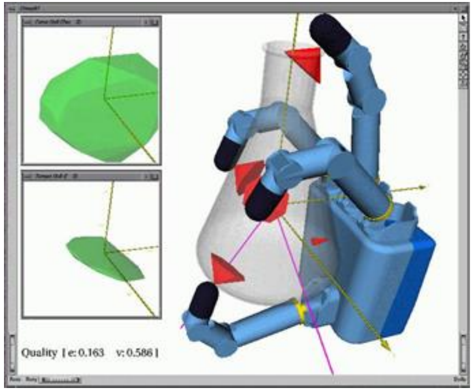
1. Load the hand and object model into a simulation environment.
2. Generate grasp candidates
   - Determine the approach direction of the hand towards the object
   - Determine the orientation of the hand
   - Determine the hand configuration (begin with an open hand)
   - Use heuristics for generating grasp candidates (reduce the search space)
3. Evaluation of grasp candidates
   - Move the hand along the approach direction until contact with the object is made.
   - Close the hand until contact with the object is made
   - Determine the contact points
   - Determine the grasp quality (e.g. **ε-/GWS-metric**)





Grasp Quality (Force-Closure Metric)

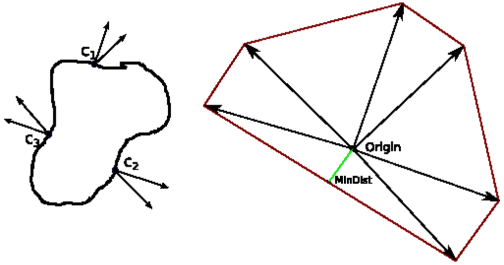
- Principle: How well can a grasp resist external forces?
- Approach
  - Determine contact points and contact normals between the hand and the object.
  - Determine the friction cone at each contact point (the cone’s opening angle depends on the friction coefficients).
  - Calculate the **Grasp Wrench Space** (GWS, 6D) as the convex hull over all friction cones.
  - **ε-metric:** The minimum distance from the center to the edge of the GWS (radius of the sphere) is a measure of the grasp's stability.



- Red: Friction cone
- Green: Projection of the GWS

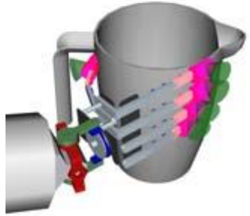
Grasp Wrench Space

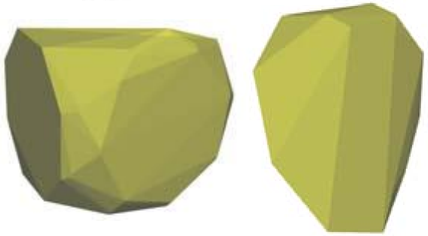
- Grasp Wrench Space (GWS):
  Convex hull over all
  contact wrenches



2D example with 3 contacts (Force)

- Quality measure
  - Force Closure:
    GWS contains origin
  - Volume (V): Volume of the GWS
  - Epsilon (ε): largest enclosing sphere,
    or the smallest distance ϵ from the origin
    to the edge of the GWS.



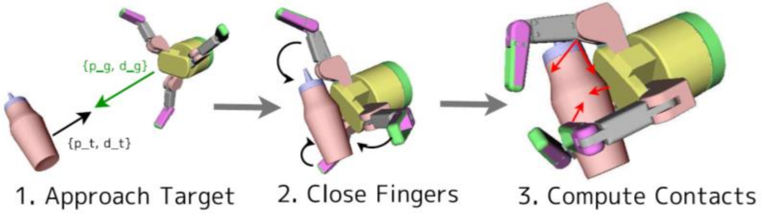


N. Vahrenkamp, T. Asfour and R. Dillmann, *Simultaneous Grasp and Motion Planning*, IEEE Robotics and Automation Magazine, Vol. 19, No. 2, pp. 43 - 57, June, 2012

Randomized Forward Grasp Planning (I)

Procedure:

1. Randomized generation of grasp hypotheses
   - Position and orientation of the hand (relation to the object)
   - Configuration of the fingers
2. Determination of contact
3. Evaluation of hypotheses
   - Force closure
   - Collision
   - Robustness

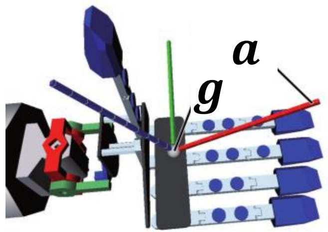


1. R. Diankov, *Automated construction of robotic manipulation programs*, Ph.D. dissertation, Carnegie Mellon University, Robotics Institute, Aug 2010.
2. Dmitry Berenson, Rosen Diankov, Koichi Nishiwaki, Satoshi Kagami, and James Kuffner, *Grasp Planning in Complex Scenes,* IEEE-RAS International Conference on Humanoid Robots (Humanoids07), December, 2007.
3. N. Vahrenkamp, M. Kröhnert, S. Ulbrich, T. Asfour, G. Metta, R. Dillmann and G. Sandini, *Simox: A Robotics Toolbox for Simulation, Motion and Grasp Planning*, International Conference on Intelligent Autonomous Systems (IAS), pp. 585 - 594, 2012

Randomized Forward Grasp Planning (II)

Hand model

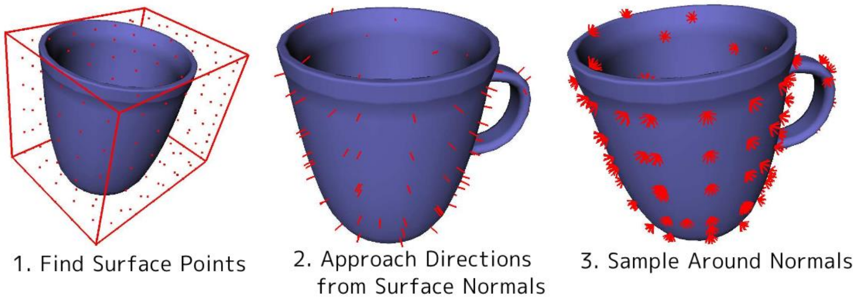
- **Grasp Center Point (GCP)**:
  Defines the center $\boldsymbol{g}$ as well as
  the approach direction $\boldsymbol{a}$ for a grasp type



Definition of the GCP of the ARMAR-III hand for a power grasp

Generation of grasp hypotheses

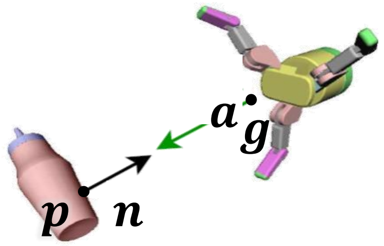
- Determination of the **approach direction**:
  - Random selection of a surface point $\boldsymbol{p}$
  - Determination of the surface normal $\boldsymbol{n}$



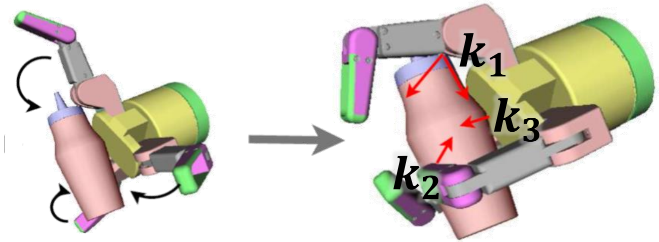
Randomized Forward Grasp Planning (III)

Determination of the grasp hypothesis

- Positioning of the hand
  - Position of the hand: $\boldsymbol{g}$ lies on the line defined by $\boldsymbol{p}$ and $\boldsymbol{n}$
  - Align the hand orientation so that $\boldsymbol{a}$ and $\boldsymbol{n}$ are collinear. The free parameter (hand orientation around $\boldsymbol{a}$) is chosen randomly.



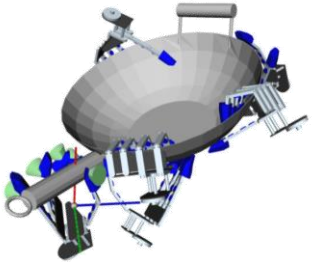
- Contact determination
  - Moving the hand to the object along $\boldsymbol{a}$
  - Closing the fingers
  - Determination of n contact points $\boldsymbol{k_1}, \dots, \boldsymbol{k_n}$

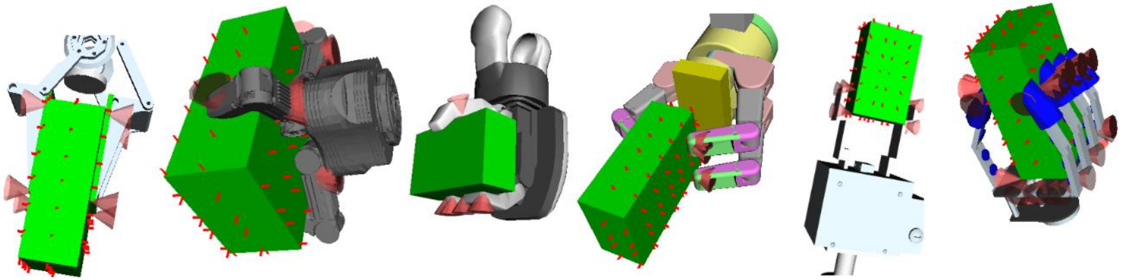


Randomized Forward Grasp Planning (IV)

Analysis of the grasp hypothesis

- Analyse the contact points $\boldsymbol{k_1}, \dots, \boldsymbol{k_n}$
  - Force closure
  - Grasp qualitiy $(V, \varepsilon, \dots)$
- Valid grasps are saved
- Grasp hypotheses with insufficient properties are discarded

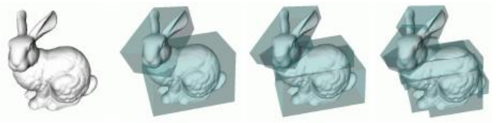
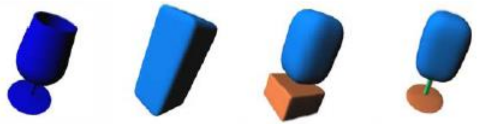




Grasp Syntheses on Object Parts

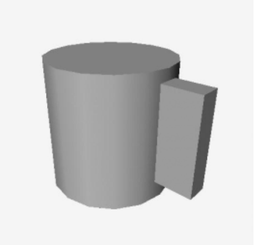
- Complex object geometry: Difficult grasp calculation
- **Solution:** Determine **good grasp candidates** on parts of an object
- Approaches for different object approximations


Shape primitives	Manual subdivision into primitives (boxes, cylinders, spheres, cones, etc.)
Box decomposition	Automatic subdivision (boxes only)
Superquadrics	Automatic subdivision
Medial Axis Transformation	Spheres only
Surface normals	

Grasp Planning with Shape Primitives (I)

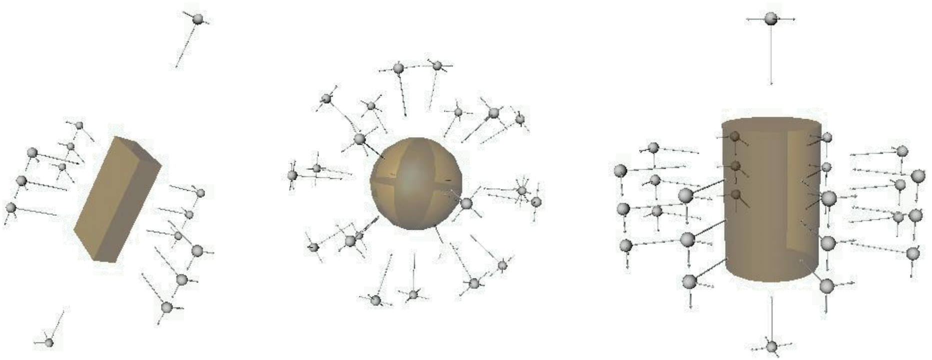
- Objects are represented by simple **shape primitives**
- For each shape primitive,
  different **grasp strategies** are predefined
  (including starting point and approach direction).
- **Forward simulation** (as before):
  - The starting point defines the initial hand position
  - Based on the approach direction, the hand is moved towards the object until contact is detected
  - Contact points are determined by closing the hand
  - Evaluation of the contacts is done using the GWS approach



A. T. Miller, S. Knoop, H. I. Christensen, and P. K. Allen, *Automatic Grasp Planning Using Shape Primitives*, in IEEE Int. Conf. On Robotics and Automation (ICRA), 2003, pp. 1824–1829.

Grasp Planning with Shape Primitives (II)

- **Grasp strategies for**
  - Box
  - Sphere
  - Cylinder
  - Cone

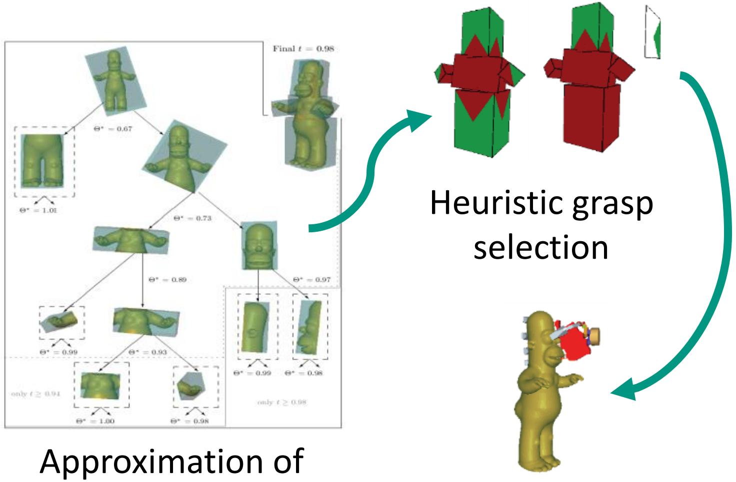


- **Parameter**
  - Number of parallel subdivisions (boxes, sides of cylinder and cone)
  - Number of circular subdivisions (sphere, sides of cylinder and cone)
  - Number of hand orientations (around the approach direction)
  - Mirroring of grasps (for symmetric primitives)
- The parameters are automatically determined based on the object information

Grasp planning with shape primitives – Results

Grasping Known Objects: Box-Based Approach (I)

- **Approximation** of object geometry by boxes (*box decomposition*)
- **Grasp hypotheses** generation for boxes
- **Evaluation** of grasp hypotheses
  in a grasp planner (here GraspIt!)



Approximation of the object geometry

Heuristic grasp selection

Final grasp

Grasping Known Objects: Box-Based Approach (II)

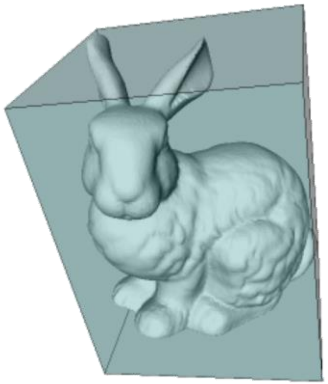
- Household objects with known geometry
  - From the **KIT ObjectModels Web Database**
    http://h2t-projects.webarchiv.kit.edu/Projects/ObjectModelsWebUI/
  - Object representation
    - 3D point cloud (stereo image, multiple object views)
    - (Textured) triangle (meshes)

- Approximation of object geometry through box decomposition
  - Efficient **Minimum Volume Bounding Box** (MVBB) algorithm
    [Barequet and Har-Peled, 2001]
  - Based on point clouds

1. K. Huebner, S. Ruthotto and D. Kragic, *Minimum Volume Bounding Box Decomposition for Shape Approximation in Robot Grasping.* In 2008 IEEE International Conference on Robotics and Automation (ICRA 2008)
2. Huebner, K., Welke, K., Przybylski, M., Vahrenkamp, N., Asfour, T., Kragic, D., and Dillmann, R. *Grasping Known Objects with Humanoid Robots: A Box-Based Approach.* In 14th International Conference on Advanced Robotics, 2009

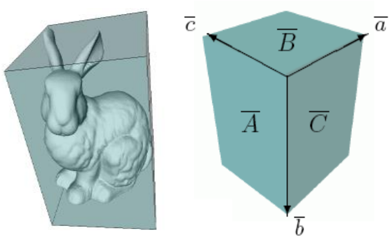
Decomposition Algorithm: From Points to Boxes (I)

- Efficient algorithm for calculating the *Minimum Volume Bounding Box (MVBB)* by Barequet and Har-Peled (2001)
- Takes a 3D point cloud with n points and calculates the MVBB with a complexity of $O(n \log n + n/\varepsilon^3)$ (ε: GWS metric for force-closure grasps)
- **Requires**: A splitting criterion to divide the root of the MVBB into a set of smaller boxes that more **accurately** approximate the shape.
- **Approach**:
  - Check the 6 axis-aligned planes of the MVBB for minimizing the enclosed volume.
  - Iteratively split the point cloud and fit two new child boxes.

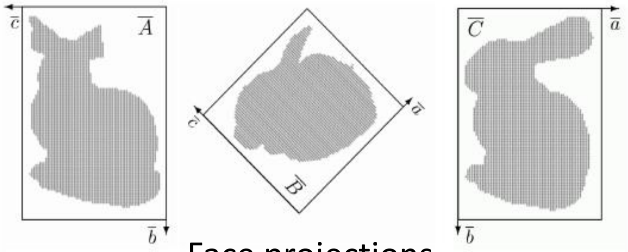


Decomposition algorithm: From points to boxes (II)

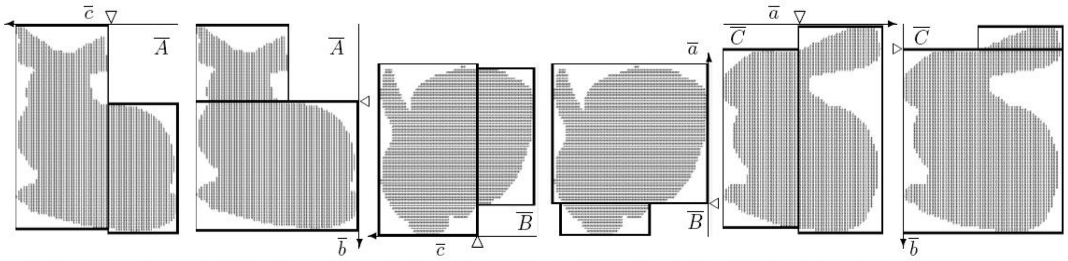
- Reduces the problem
  of splitting a 3D cuboid by a surface-aligned plane
  to splitting a 2D rectangle by an edge-aligned line.



Source and MVBB



Face projections

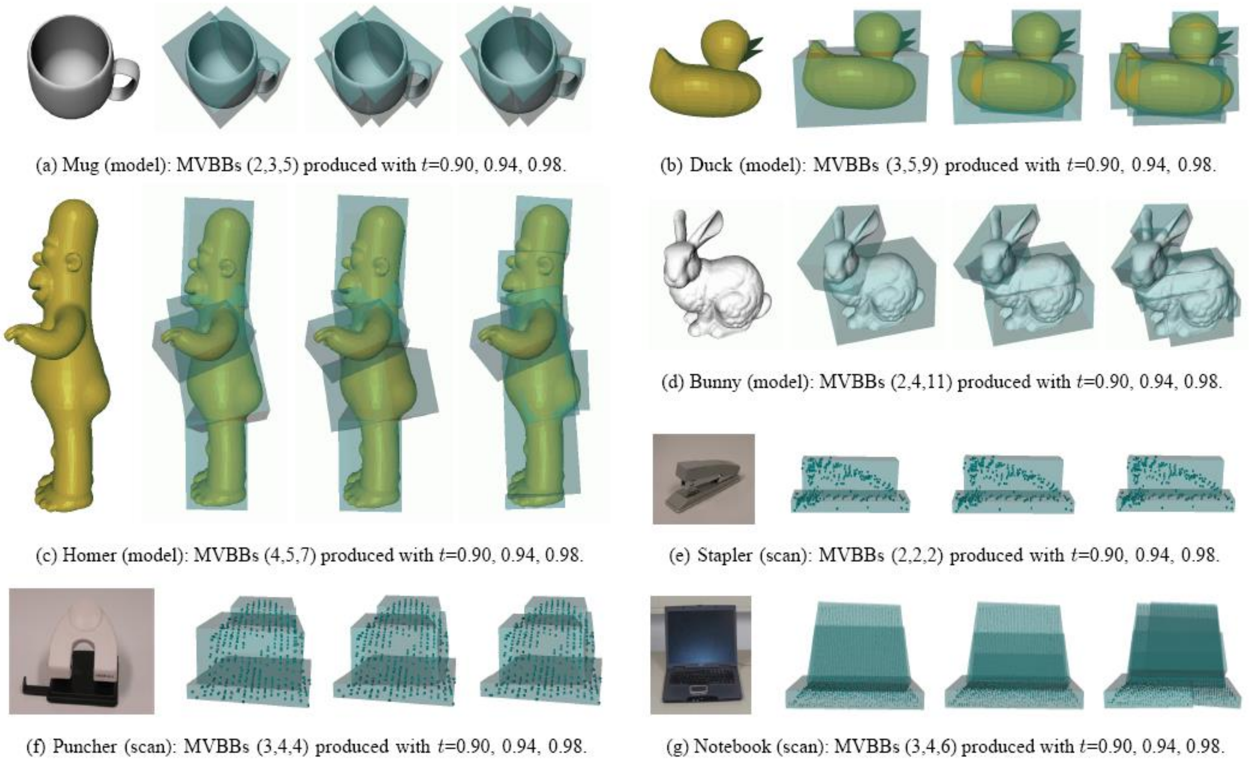


$\theta_1(\overline{A},\overline{c},i_1) = 0.84$ $\theta_2(\overline{A},\overline{b},i_2) = 0.84$ $\theta_3(\overline{B},\overline{c},i_3) = 0.92$ $\theta_4(\overline{B},\overline{a},i_4) = 0.92$ $\theta_5(\overline{C},\overline{a},i_5) = 0.93$ $\theta_6(\overline{C},\overline{b},i_6) = 0.94$

Fig. 4. Best cuts along the six box directions and cut positions i marked by triangles. The corresponding volume values $\theta(\overline{\mathcal{F}},\overline{f},i)$ are presented below.

K. Huebner and D. Kragic, "Selection of robot pre-grasps using box-based shape approximation," 2008 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2008, pp. 1765-1770.

Decomposition algorithm: From points to boxes (III)



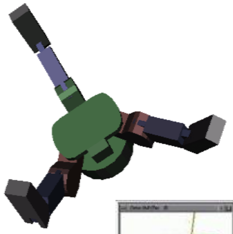
Details in [Huebner et al., 2008]

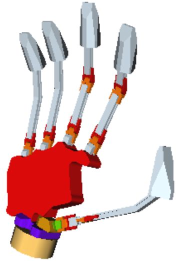
Generating grasp hypotheses: From boxes to grasps

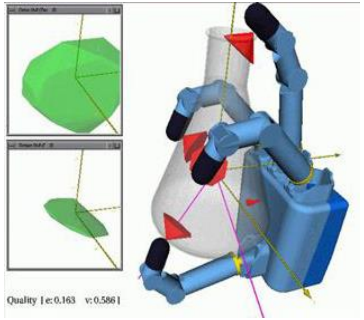
- Each box has six faces, each of which implies four grasp hypotheses.
  - **Grasp point**: Center of the face
  - **Grasp direction**: Along the normal of the face
  - **Hand orientation**: Four possibilities, oriented along the edges of the face
- *Impossible* grasps can be directly excluded based on the sizes of the faces.
  - E.g. if the face is larger than the hand opening
- Grasp hypotheses for **blocked** or **hidden** faces can be discarded through simple geometric calculations.

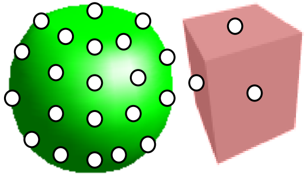
Evaluation (I)

- Experimental evaluation through the simulation environment **GraspIt!** (Miller et al., 2004)
- Two hand models used
  - Barrett Technologies (3 Finger, 4 DOF)
  - Karlsruhe Humanoid Hand (5 Finger, 8 DOF)
- Two quality measures
  - V: Volume of the GWS
  - ε: Biggest sphere of the GWS
- Two methods were compared
  - Spherical: 22,104 grasp hypotheses (points on the spherical surface)
  - Box: Grasp hypotheses from box subdivision

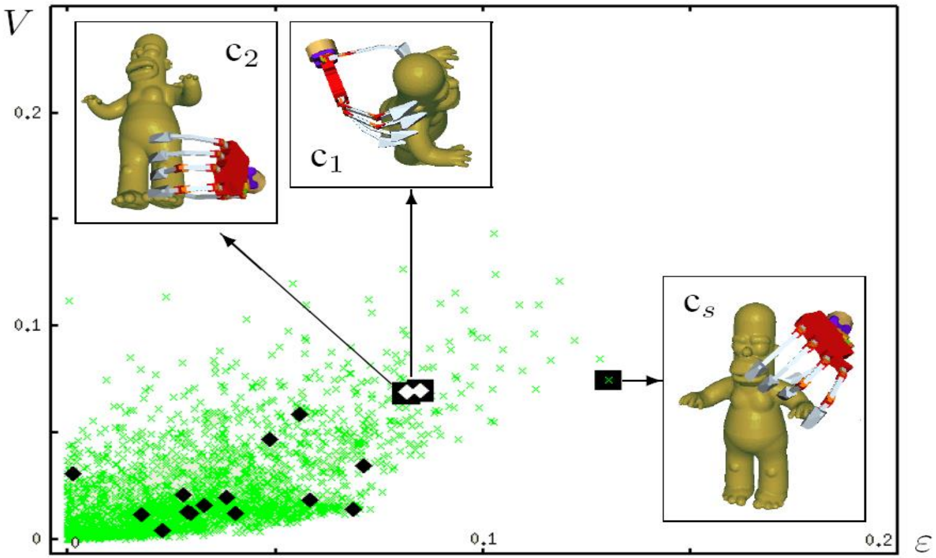








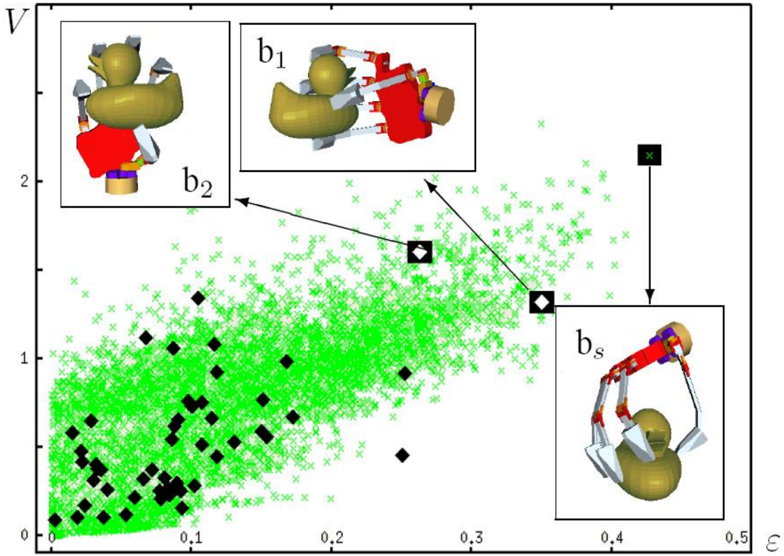
Evaluation (II)



Sample of spherical grasp
Sample of box grasp
“Best” spherical grasp
“Best” box grasp(s)

Box decomposition generates **fewer but higher-quality** grasp hypotheses

Evaluation (III)



Sample of spherical grasp

Sample of box grasp

“Best” spherical grasp

“Best” box grasp(s)

Box decomposition generates **fewer but higher-quality** grasp hypotheses

Evaluation of ARMAR-III

Karlsruhe Institute of Technology

paco plus

perception, action and cognition
through learning of object-action complexes

CogSys Cognitive Systems

Information Society Technologies

Grasping Objects using Box Decomposition on the Humanoid Robot ARMAR-III

University of Karlsruhe
M. Przybylski, K. Welke, T. Asfour,
R. Dillmann

Kungliga Tekniska Högskolan
K. Hübner, D. Kragic

March 2009

Decomposition algorithm: Results

- **Exact form** is not necessary to generate grasps
- Objects can be processed better through **simplified shapes** (boxes). **Point clouds** of the surfaces can be used for this.
- Easy **parameterization** of the algorithms
- Evaluation: **Box decomposition** approach produces grasps that can be successfully executed by a humanoid robot.

Grasp planning with superquadrics (I)

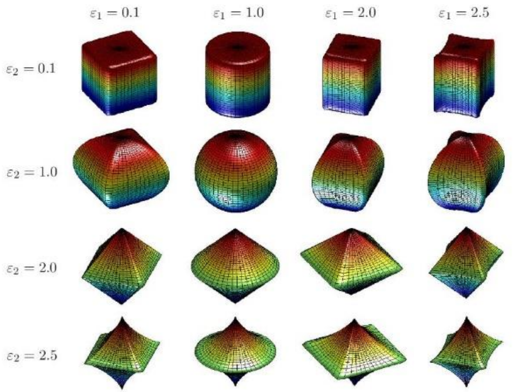
Superquadric:

- Parameterizable functions define the shape of the geometric object
- A superquadric aligned to the coordinate axes at the origin of the coordinate system is described by five parameters with the following equation:

$$X(\eta, \omega) = \begin{pmatrix} a_1 \cos^{\varepsilon_1}(\eta) \cos^{\varepsilon_2}(\omega) \\ a_2 \cos^{\varepsilon_1}(\eta) \sin^{\varepsilon_2}(\omega) \\ a_3 \sin^{\varepsilon_1}(\eta) \end{pmatrix}$$

for $-\frac{\pi}{2} \leq \eta < \frac{\pi}{2}$ and $-\pi \leq \omega < \pi$.

The parameters a_1, a_2, a_3 describe the size of the superquadric in the direction of the three spatial axes. The exponents $\varepsilon_1, \varepsilon_2 \in [0,1]$ determine the sharpness of the edges.



C. Goldfeder, P. K. Allen, C. Lackner, and R. Pelossof, *Grasp Planning Via Decomposition Trees*, in IEEE Int. Conf. on Robotics and Automation (ICRA), 2007, pp. 4679–4684.

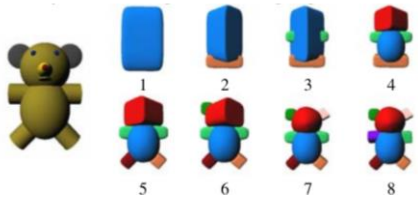
Grasp planning with superquadrics (II)

Representation with Superquadrics

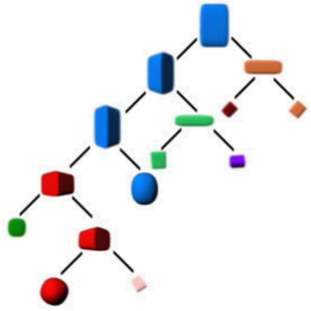
- Object surface is displayed as a point cloud
- Determine the superquadric whose surface best approximates the point cloud (nonlinear optimization algorithm, e.g. Levenberg-Marquardt)

Decomposition Tree

- Refine the approximation step by step
  - Find the superquadric with the largest error for the point cloud
  - Divide the corresponding point cloud
  - Create two new superquadrics
- Store results for each approximation step in the **decomposition tree**



superquadrics of 8 decomposition steps

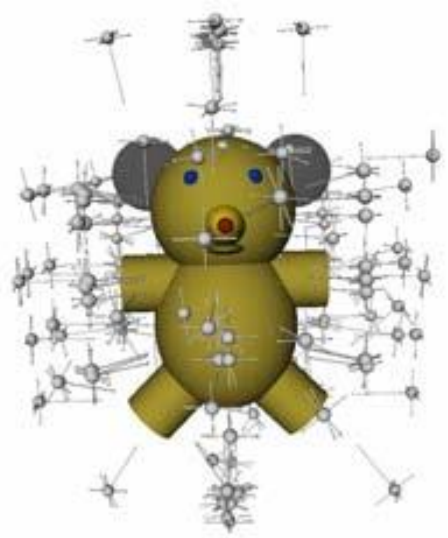


Decomposition Tree

Grasp planning with superquadrics (III)

Grasp planning

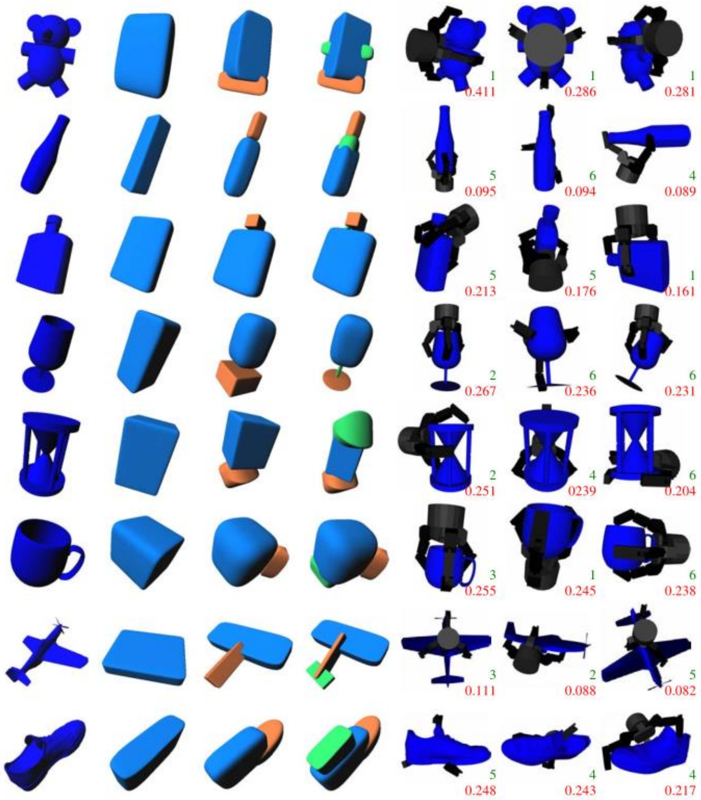
- For each superquadric, **grasp hypotheses** are generated. Equally distributed approach points and approach directions relative to the surface of the corresponding superquadric.



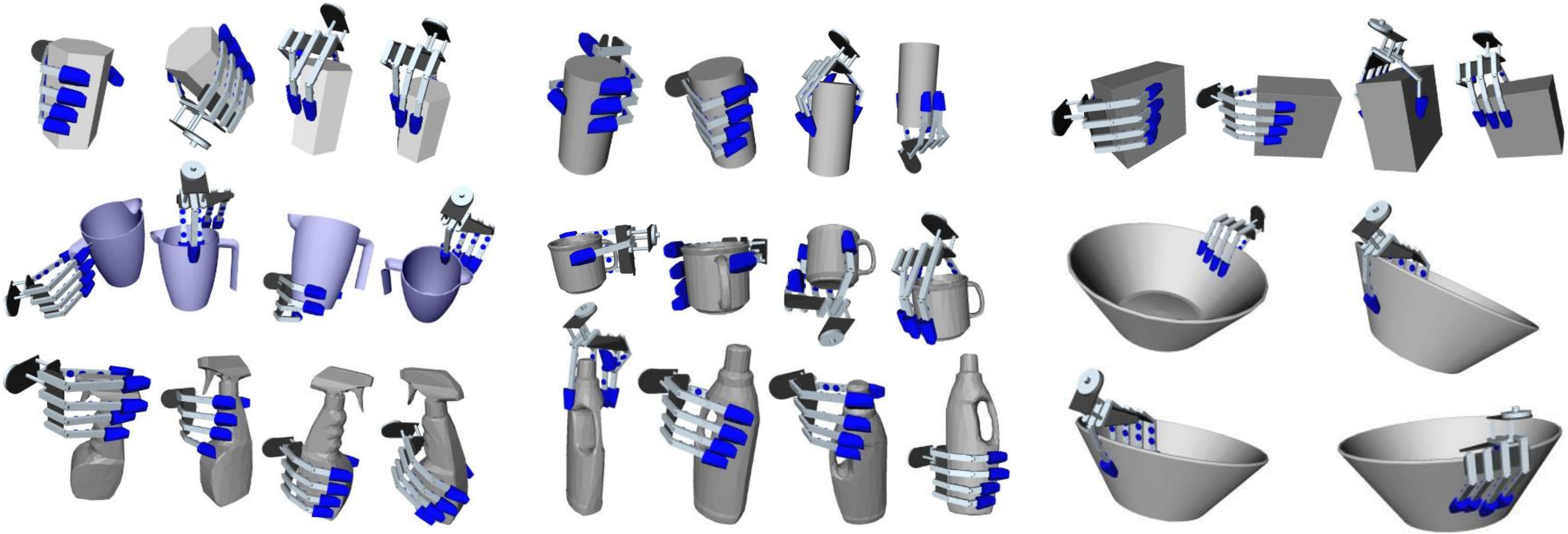
- **Evaluation** of the grip hypotheses using GWS approaches. **Detailed object model** (3D mesh)

- For **all levels** of the decomposition tree: grasp planning on coarse and fine structures of the object

Grasp planning with superquadrics – Results



Grasp Planning with Medial Axes



Motivation

- **State of the art:** Algorithms for grasp planning in simulation environments
  - Grasp hypotheses are examined for stability
  - Efficiency depends on the heuristics for generating grip hypotheses
- **Goal:** Improve the efficiency of grasp planning algorithms by only examining *"geometrically meaningful"* grasps

- **Approach:** Use **local symmetries** of the object geometry.
  The representation of the object is important
  - Triangular mesh: Too low level of abstraction
  - **Medial axis as object representation**

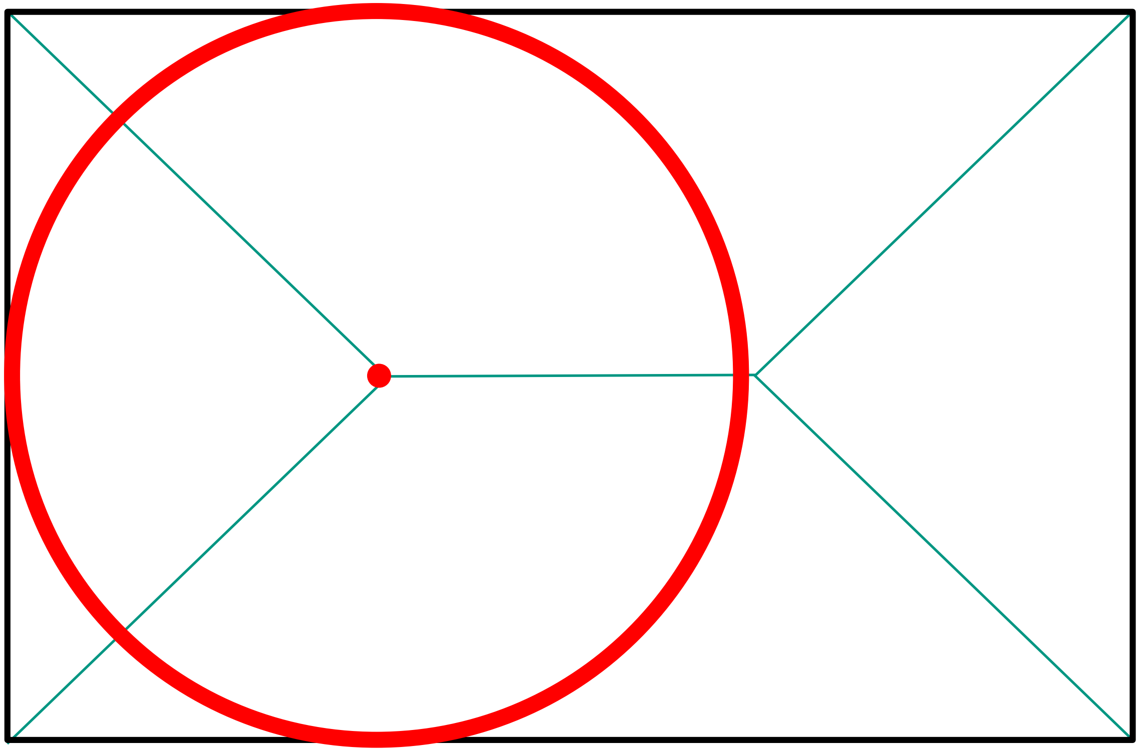
1. Przybylski, M., Asfour, T. & Dillmann, R. *Unions of Balls for Shape Approximation in Robot Grasping*. 2010 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2010
2. Przybylski, M., Wächter, M., Asfour, T. & Dillmann*, A Skeleton-based Approach to Grasp Known Objects with a Humanoid Robot*, IEEE/RAS International Conference on Humanoid Robots (Humanoids), pp. 376 - 383, 2012

Grasp Planning with Medial Axes

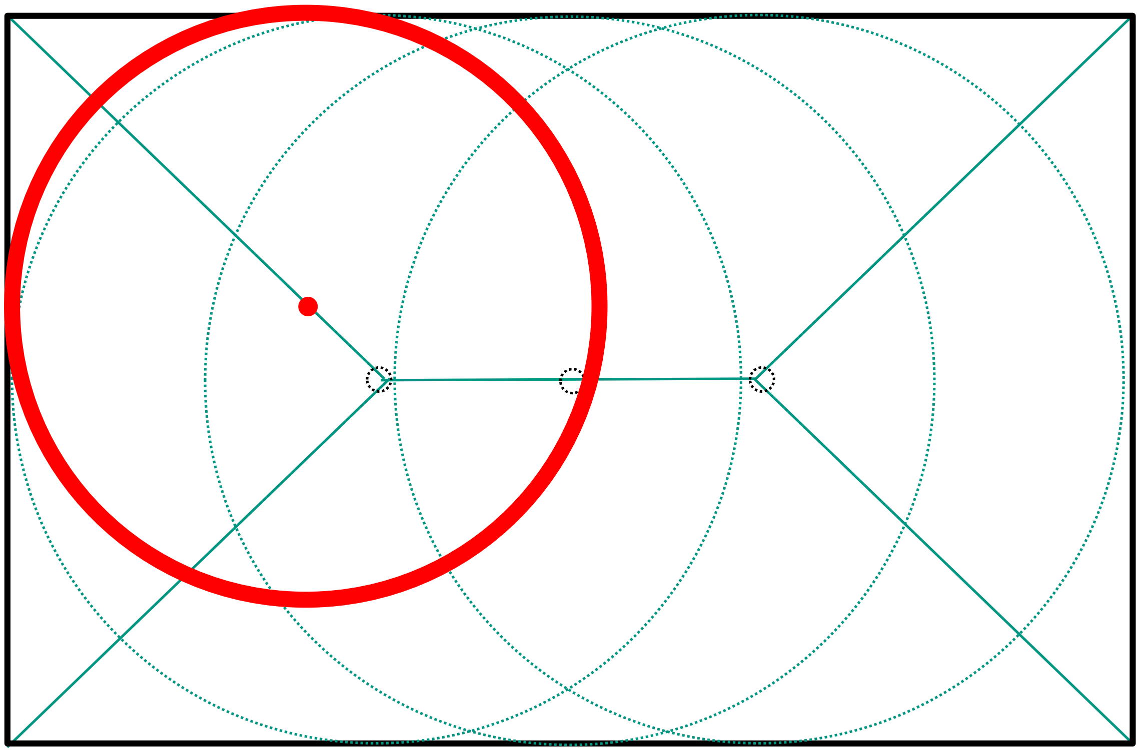
- Medial axis (Blum 1967)
  - Object shape is approximated using **contained spheres** with maximum diameter
  - Contained spheres must touch the object shell at **at least two** points
- **The medial axis is defined as the union of the centers of all contained spheres**
- The medial axis describes the **topological skeleton** of the object
- Advantages:
  - Good approximation of the object geometry
  - Details are retained
  - Good description of the symmetries

H. Blum, Models for the Perception of Speech and Visual Form. A transformation for extracting new descriptors of shape, Cambridge, Massachusetts: MIT Press, 1967, pp. 362–380.

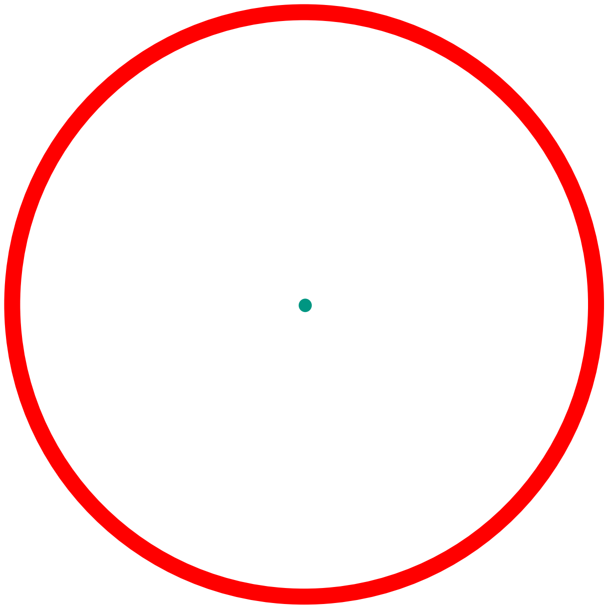
Example: Medial Axes G_1



Example: Medial Axes G_1

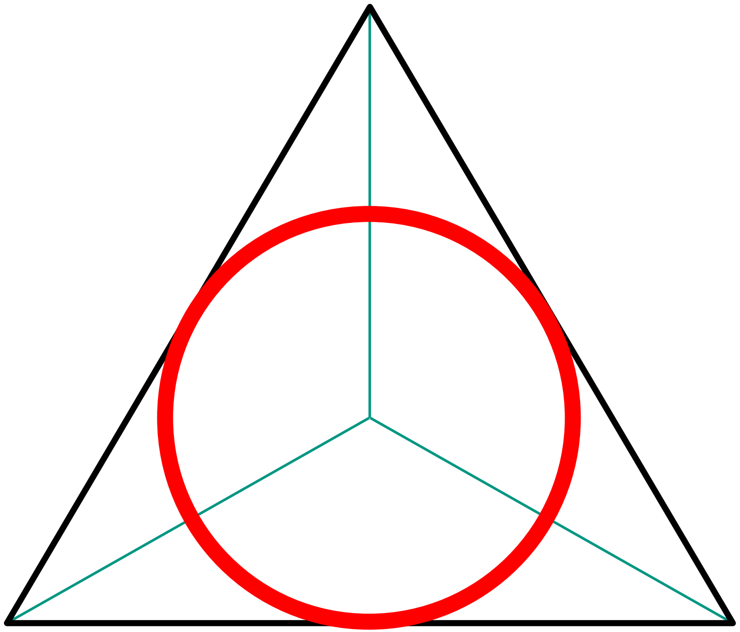


Example: Medial Axes G_2



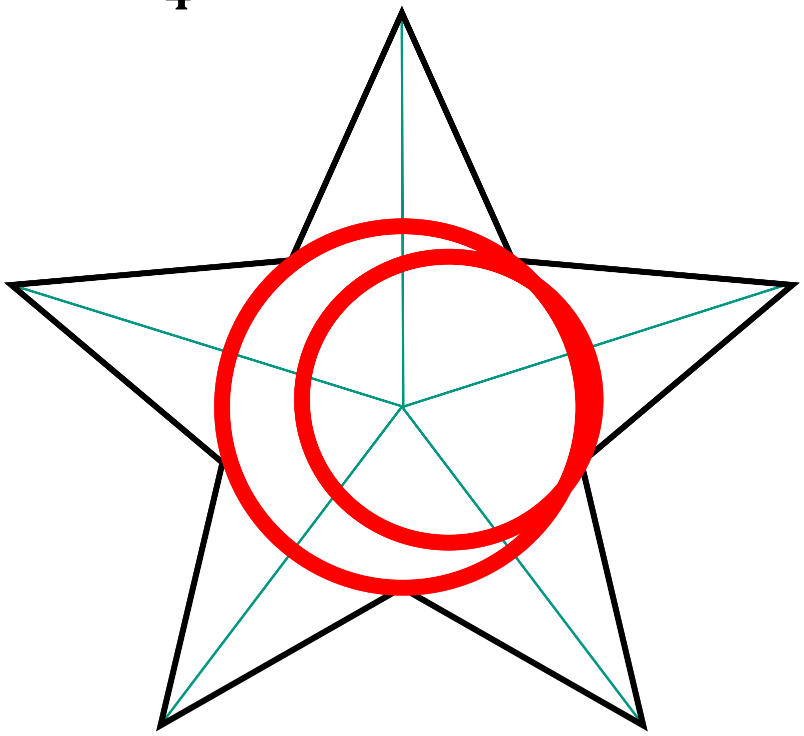
G_2

Example: Medial Axes G_3



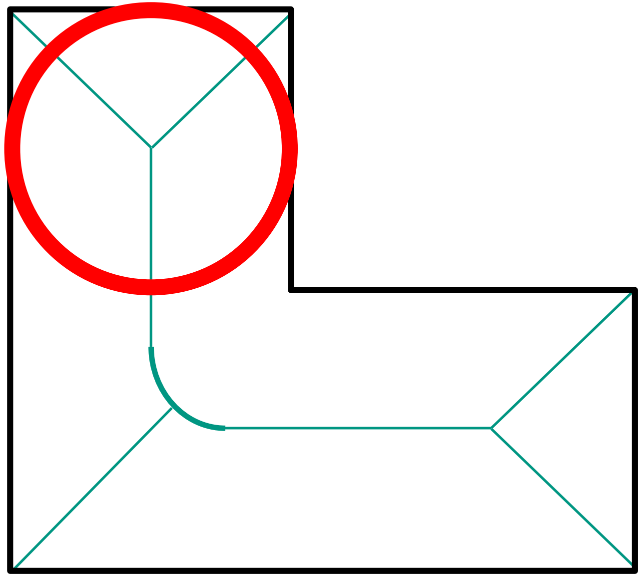
G_3

Example: Medial Axes G_4



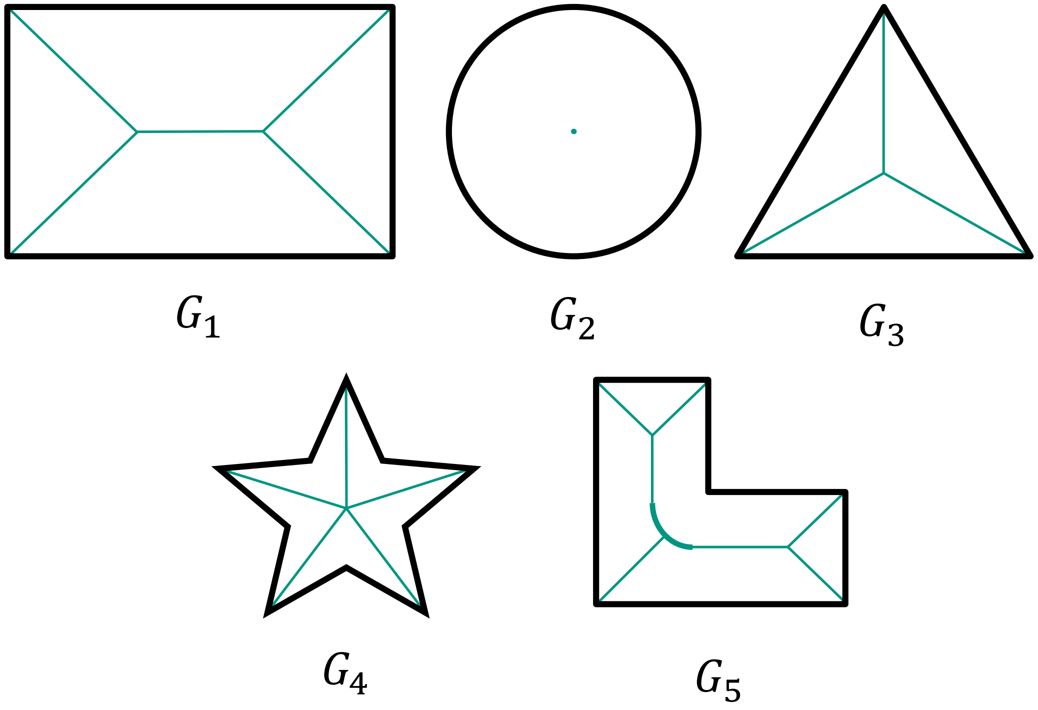
G_4

Beispiel: Mediale Achsen G_5



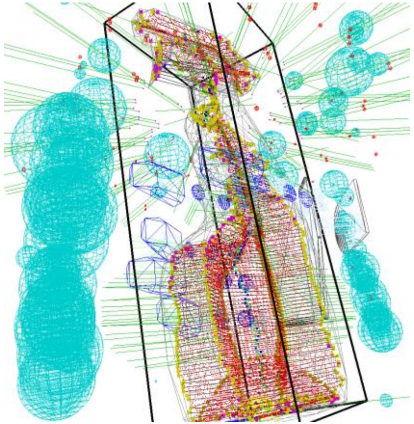
G_5

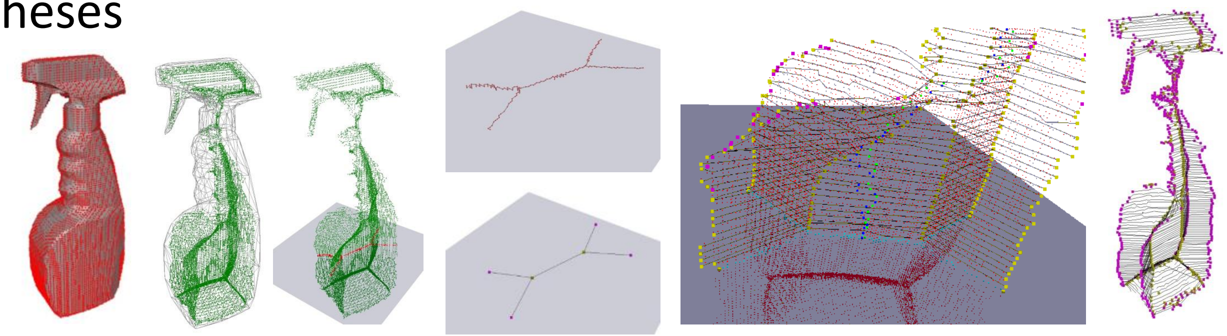
Example: Medial Axes $G_1 \ldots G_5$



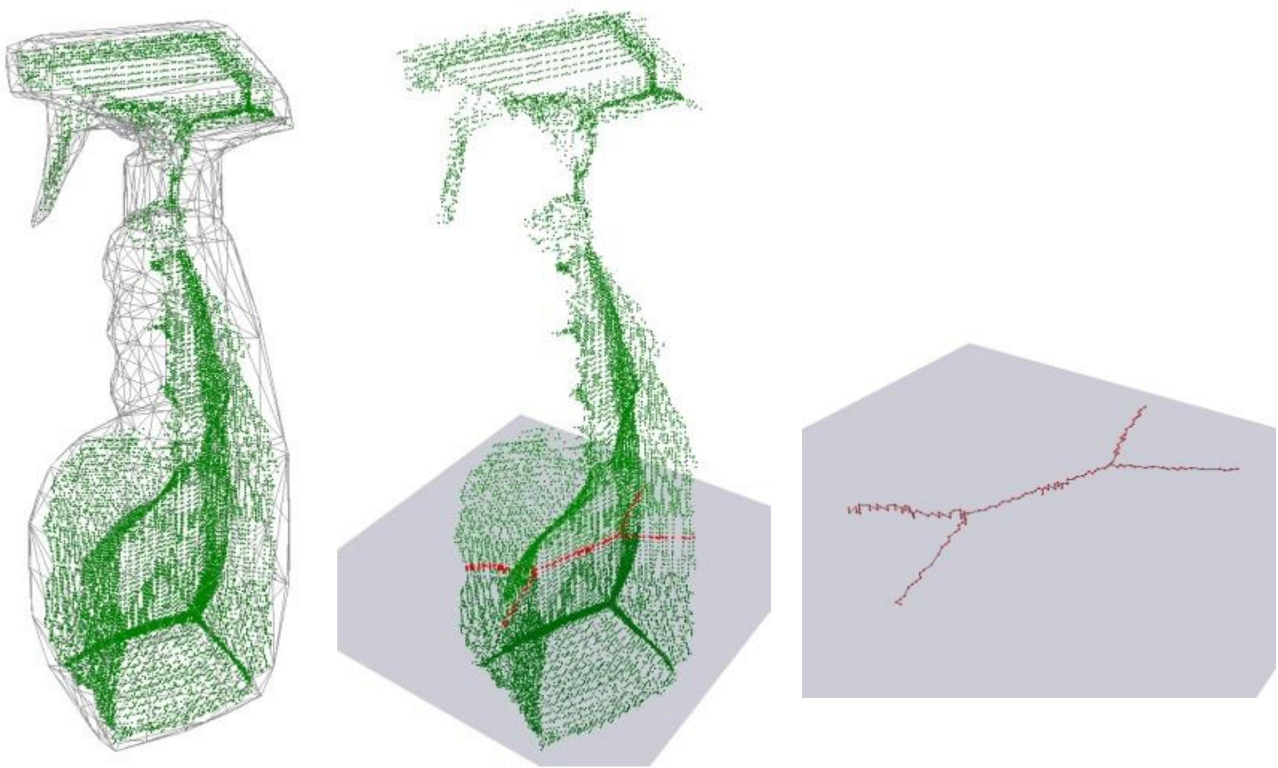
Grasp Planning with Medial Axes – Algorithm

1. Sample the object surface
2. Calculate the medial axis
3. Analyze the cross sections of the medial axis
   - Minimum Spanning Tree (MST)
   - Cluster
   - Convex hull
4. Generate grasp hypotheses
5. Evaluate
   grasp stability

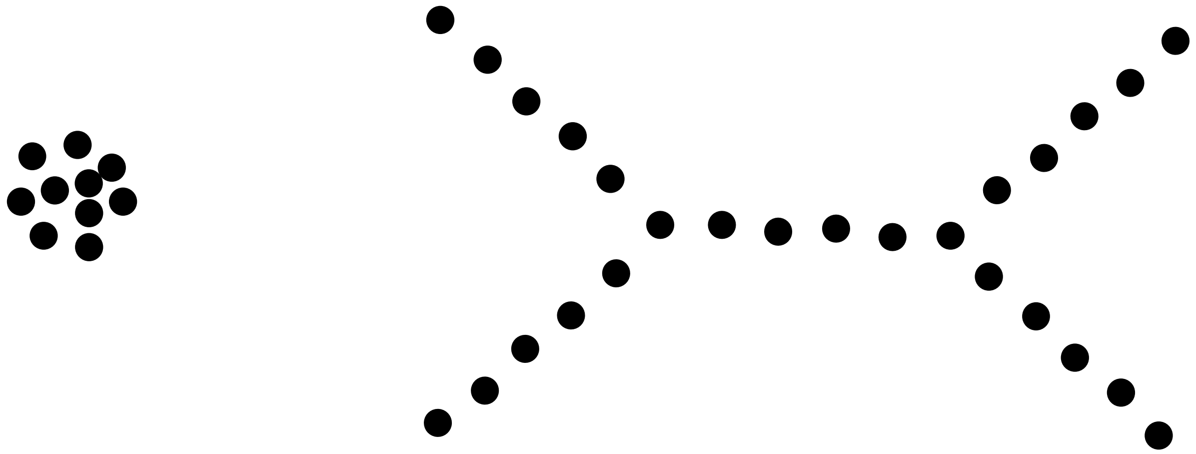




Analysis of Cross Sections of the Medial Axis

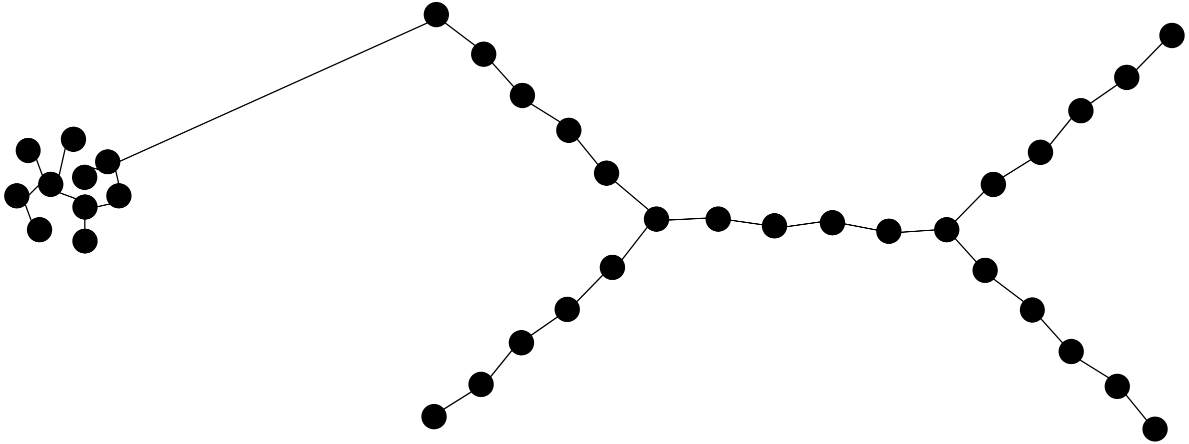


Analysis of Cross Sections of the Medial Axis I



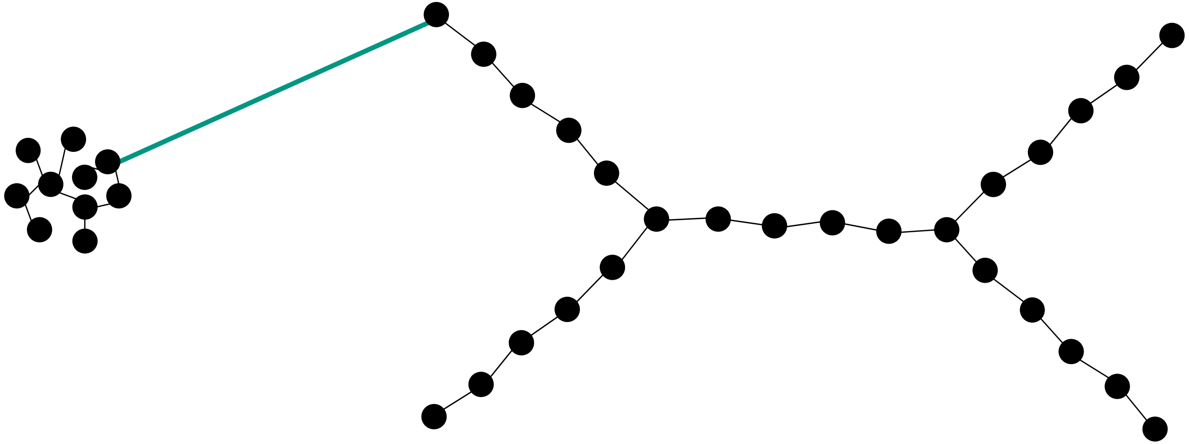
- Medial axis represented as a point cloud
- We are looking for the nearest neighbor relationships, branching points
- Approach: **Minimum Spanning Tree (MST)**

Analysis of Cross Sections of the Medial Axis II



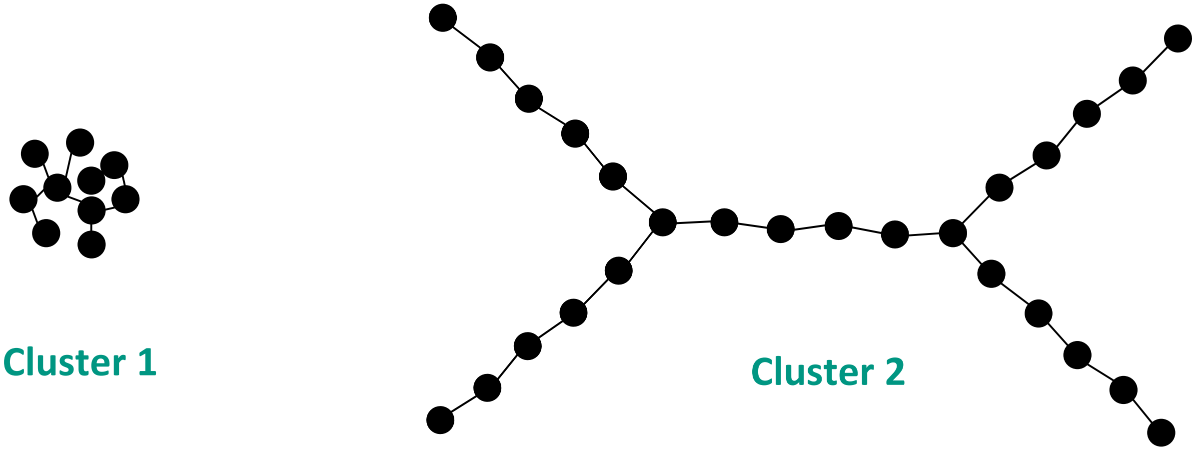
- **Step 1**: Determine MST
- Result: All points are connected to their nearest neighbors

Analysis of Cross Sections of the Medial Axis III



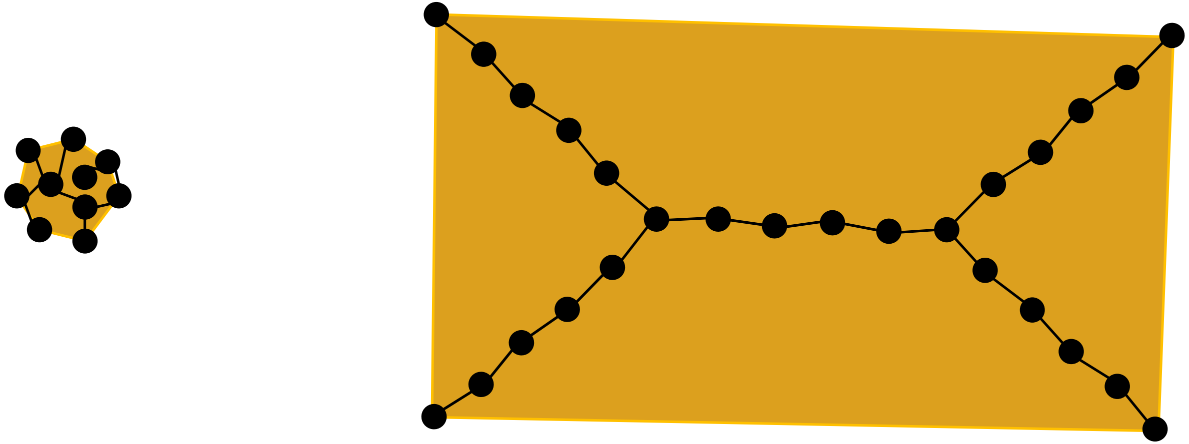
- **Step 2:** Remove edges that are longer than a threshold d_{cut}
- Result: Segmentation of the medial axis into clusters

Analysis of Cross Sections of the Medial Axis III



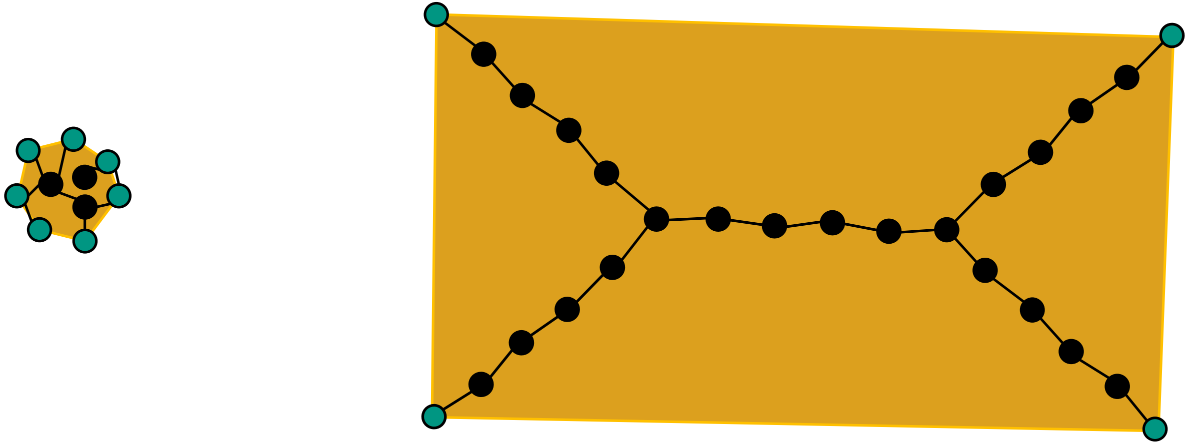
- **Step 2:** Remove edges that are longer than a threshold d_{cut}
- Result: Segmentation of the medial axis into clusters

Analysis of Cross Sections of the Medial Axis IV



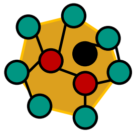
- **Step 3:** Calculate for each cluster
  - **Volume of the convex hull**
  - Boundary points of the convex hull
  - Branching points of the MST
  - Center of mass

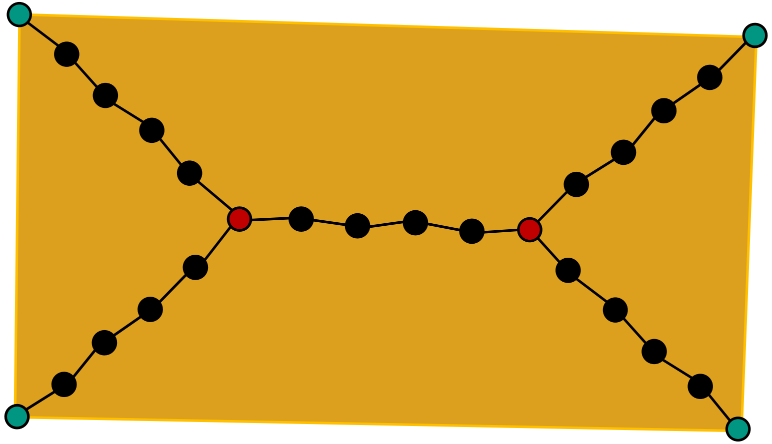
Analysis of Cross Sections of the Medial Axis V



- **Step 3:** Calculate for each cluster
  - Volume of the convex hull
  - **Boundary points of the convex hull**
  - Branching points of the MST
  - Center of mass

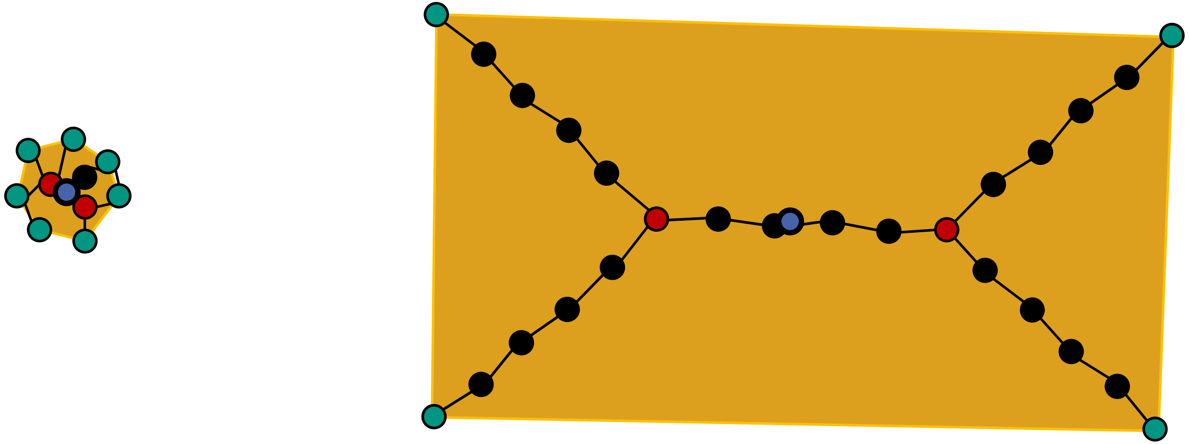
Analysis of Cross Sections of the Medial Axis VI





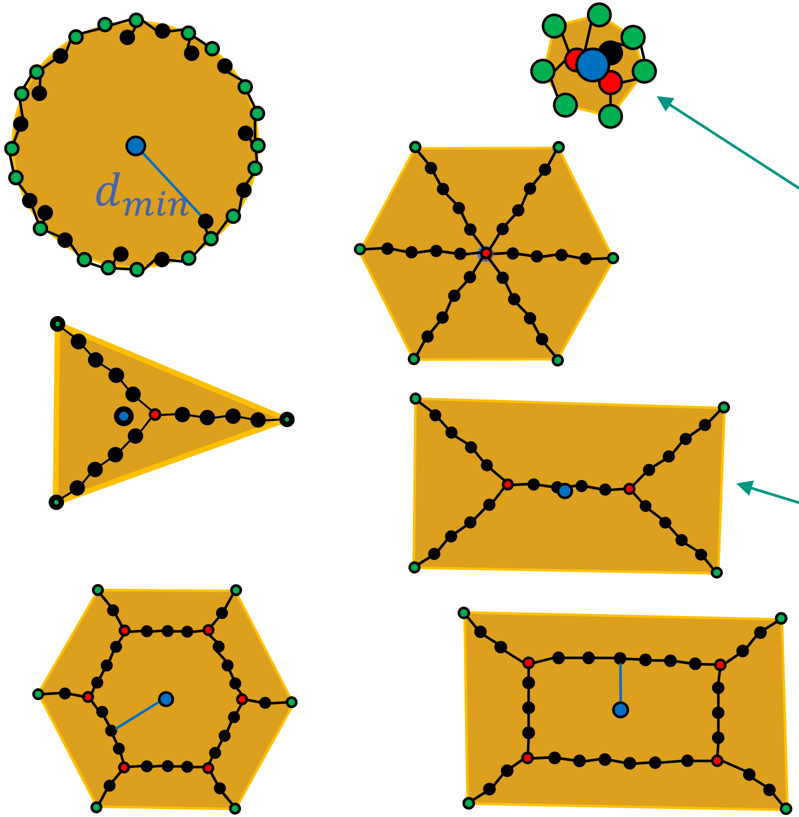
- **Step 3:** Calculate for each cluster
  - Volume of the convex hull
  - Boundary points of the convex hull
  - **Branching points of the MST**
  - Center of mass

Analysis of Cross Sections of the Medial Axis VII



- **Step 3:** Calculate for each cluster
  - Volume of the convex hull
  - Boundary points of the convex hull
  - Branching points of the MST
  - **Center of mass**

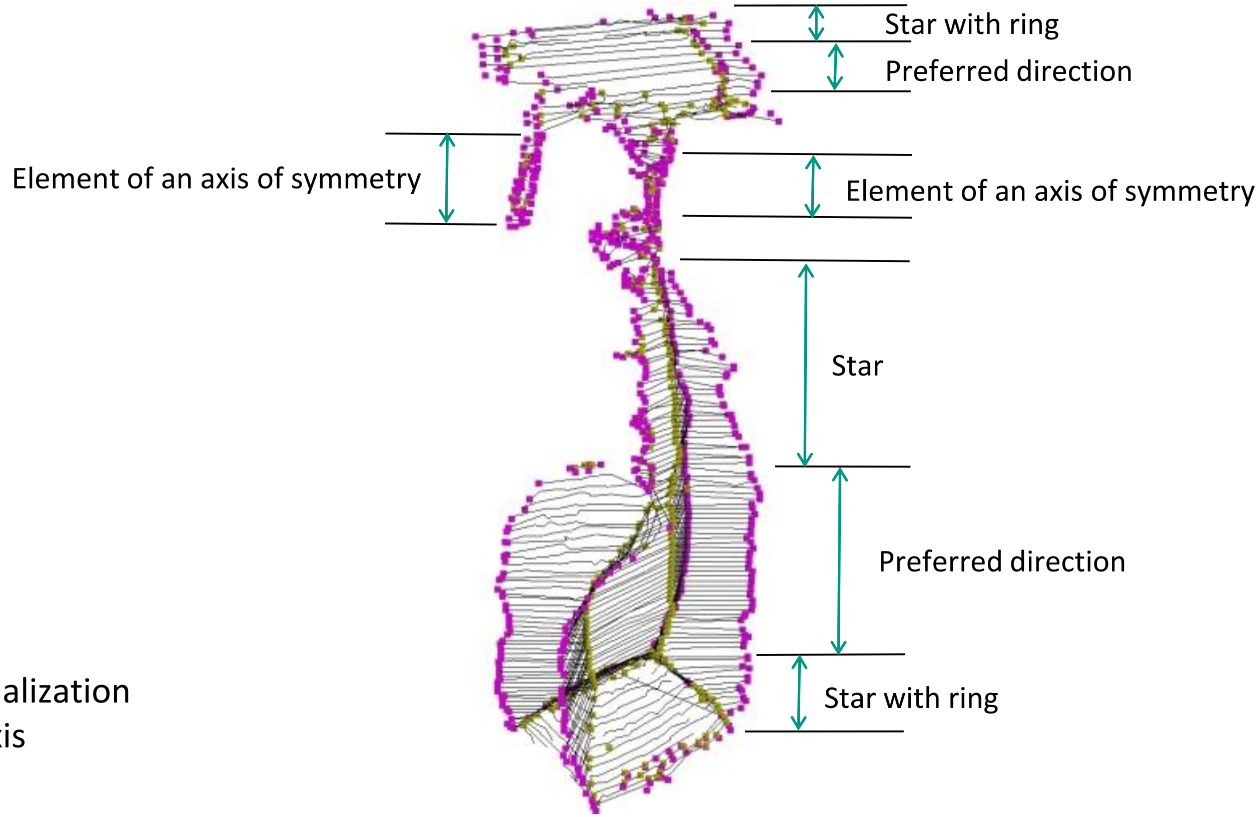
Anaysis of Clusters Found



- **Circle/Ring**
  - Large distance between the center of mass and the nearest point on the medial axis
  - Many points on the edge of the convex hull
- **Element of a symmetry axis**
  - Lower volume of the convex hull
  - Fewer points on the edge of the convex hull
- **Star**
  - Exactly one branching point
- **Preferred direction**
  - Exactly two branching points
- **Star with ring**
  - Large distance between the center of mass and the nearest point on the medial axis

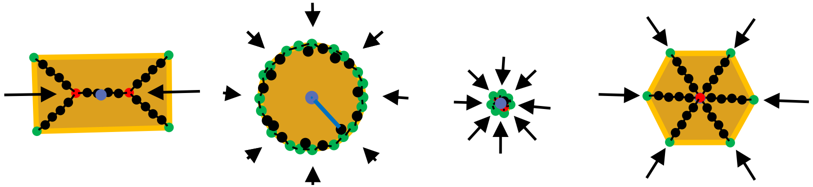
Example: Analysis of the Medial Axis

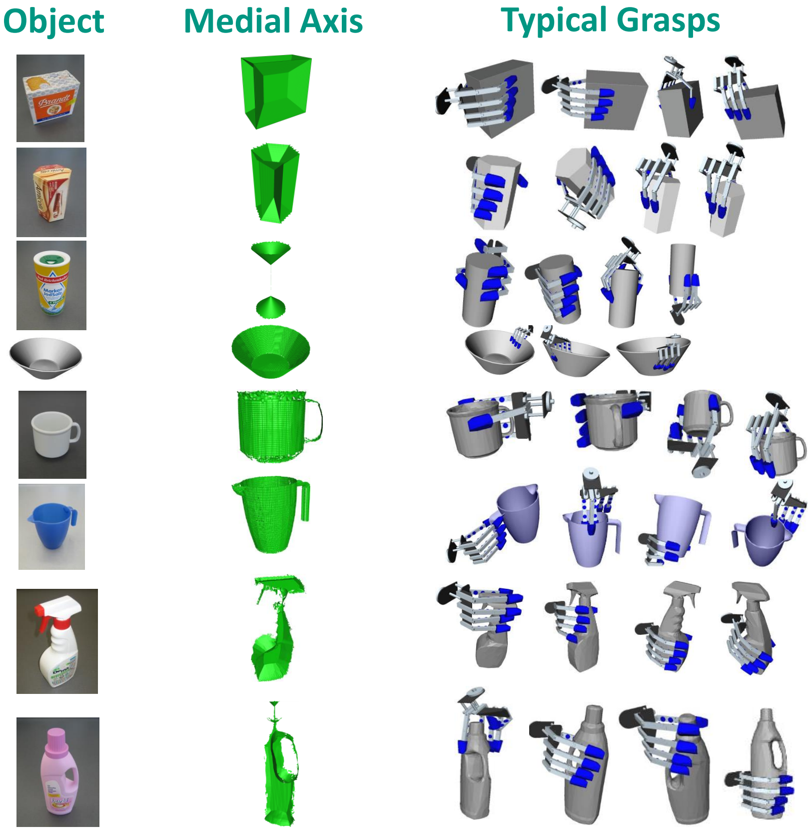
Object model with visualization of the medial axis



Heuristics for Generating Grasp Hypotheses (I)

- Approach direction of the hand
  - Branching points of the MST
  - Tips of a star
  - Symmetry axes
  - Edge of open objects

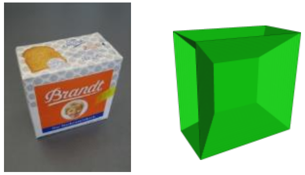


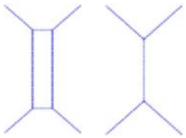


- Roll angle of the hand
  - Local symmetry axes
  - Local symmetry planes
- For more complex objects
  - Apply a combination of these rules

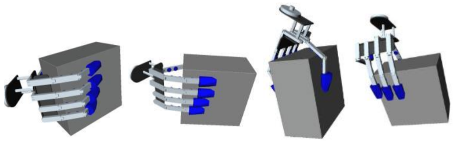
Heuristics for Generating Grasp Hypotheses (II)

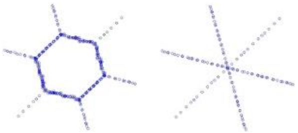
Medial Axis | **Cross section** | **Heuristics and Grasp Hypotheses**



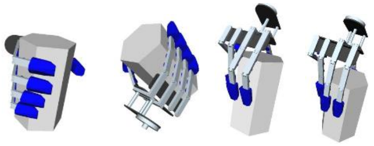


- Approaching branching points of the MST
- Align the roll angle of the hand according to the planes of symmetry



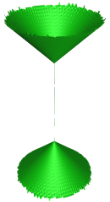


- Approaching star tips
- Align the roll angle of the hand according to the plane of symmetry

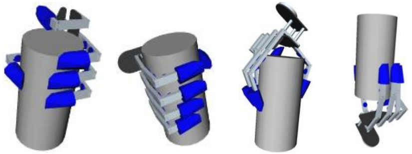


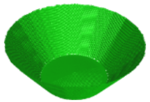
Heuristics for Generating Grasp Hypotheses (III)

Medial Axis	Cross section	Heuristics and Grasp Hypotheses



- Approaching the circle and the axis of symmetry from different directions
- Align the roll angle of the hand with the axis of symmetry



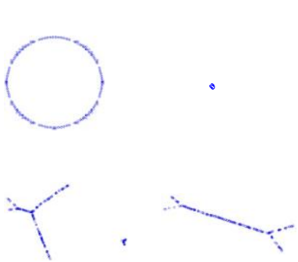


- Objects with openings: Approaching the edge of the opening

Heuristics for Generating Grasp Hypotheses (IV)

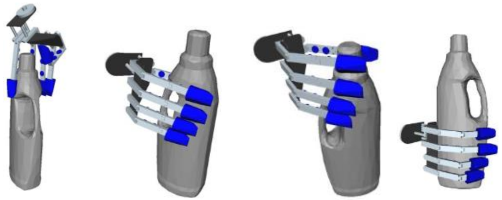
Medial Axis

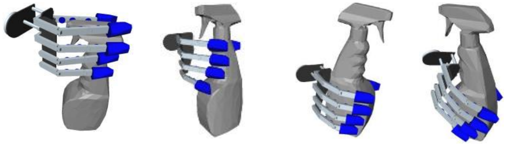
Cross section



Heuristics and Grasp Hypotheses

- Complex objects: combination of heuristics





Heuristics for Generating Grasp Hypotheses (V)

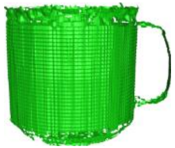
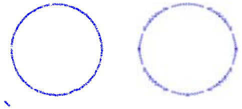
Karlsruhe Institute of Technology

Medial Axis

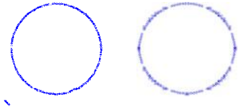
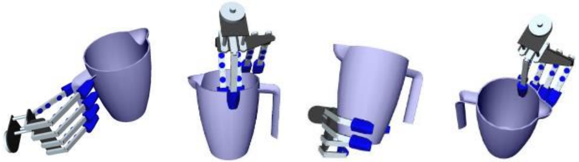
Cross section

Heuristics and Grasp Hypotheses

- Complex objects: combination of heuristics



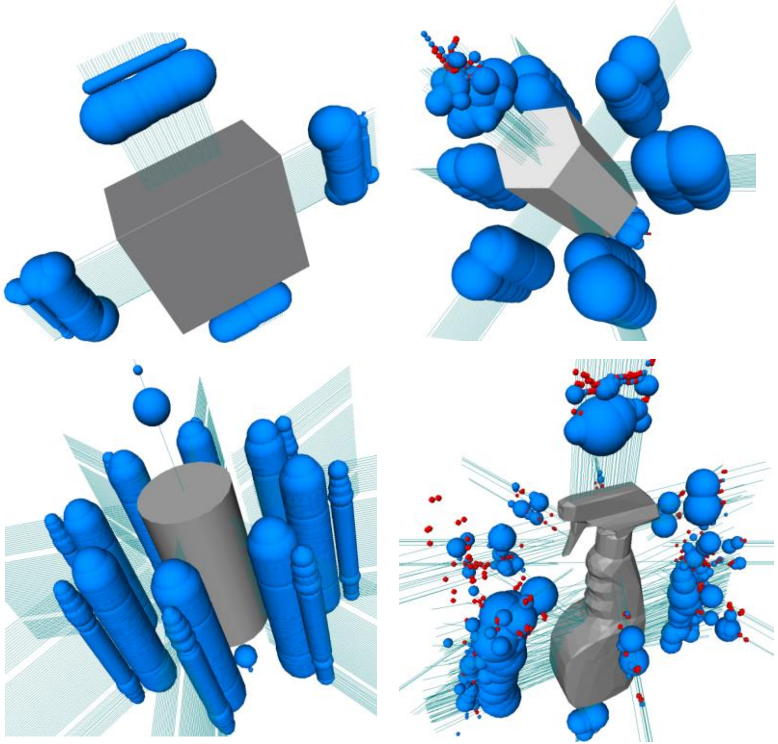




Results: Grasp Quality (Force-closure)

- Blue Spheres: stable Grasps
- Red Spheres: instable Grasps

- Position of the sphere: Position of the wrist during grip

- Diameter of the ball: measure of stability (largest ball = most stable grasp)

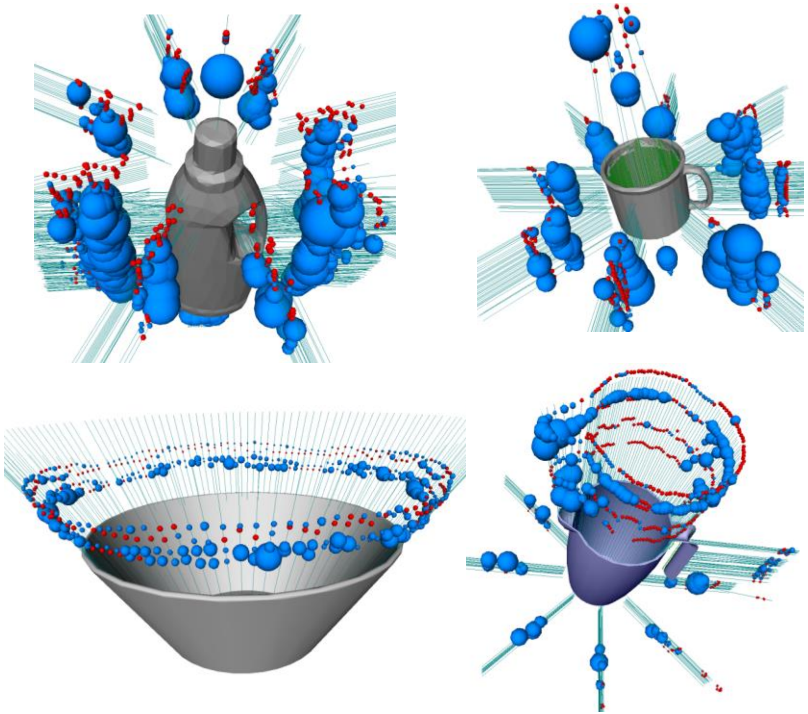


Results: Grasp Quality (Force Closure)

- Blue Spheres: stable Grasps
- Red Spheres: instable Grasps

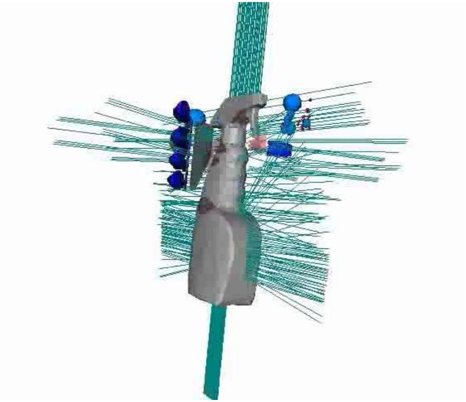
- Position of the sphere: Position of the wrist during grip

- Diameter of the ball: measure of stability (largest ball = most stable grasp)



Results: Efficiency

- Comparison with surface normal based grasp planner [Berenson07]
  - Number of possible grasps generated
  - Stable grasps

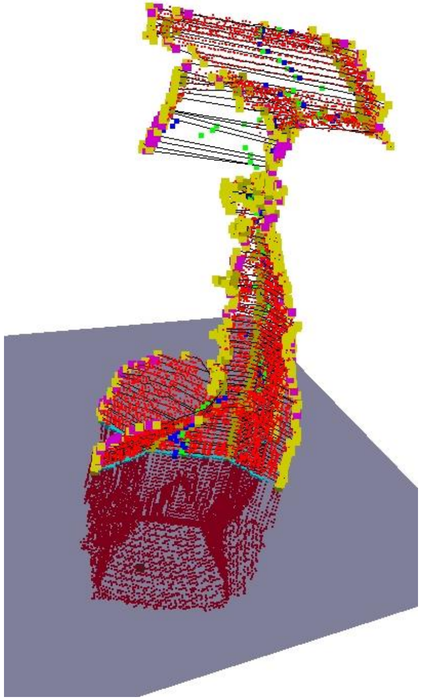


- Result:
  - Planning based on medial axis is more efficient

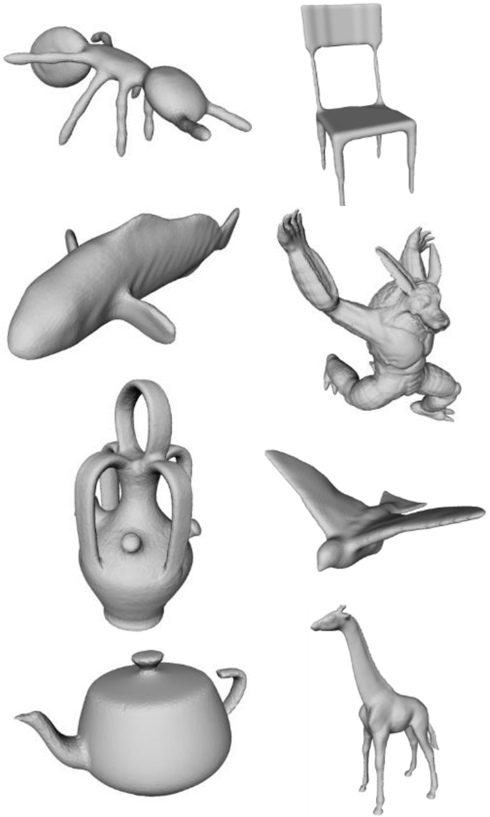
Objects	MA-based planner Candidates	MA-based planner Stable	Surface normals planner Candidates	Surface normals planner Stable
Bread box	632	86.2%	13440	15.5%
Prismatic box	1344	90.7%	8512	36.0%
Salt can	2144	96.9%	7904	45.7%
Detergent	1996	65.9%	12672	26.2%
Spray	1304	55.1%	11200	21.2%
Cup	1428	59.5%	6688	37.0%
Pitcher	1124	47.0%	15504	25.9%
Salad bowl	504	68.5%	13648	4.5%

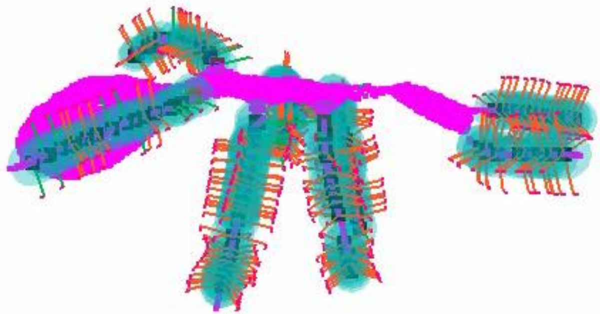
Summary

- The medial axis contains information about the structure and geometry of the object
- Generation of possible grasps with heuristics
  - Inclusion of object symmetry
  - Possible grasps are geometrically “useful“
    - High percentage of stable grasps
    - Resulting grasps appear "natural“
- Exact approximation of the object geometry
  - Possible grasps are not influenced by a poorly approximated object geometry
- Extendable concept
  - Extendable set of heuristics to generate possible grasps

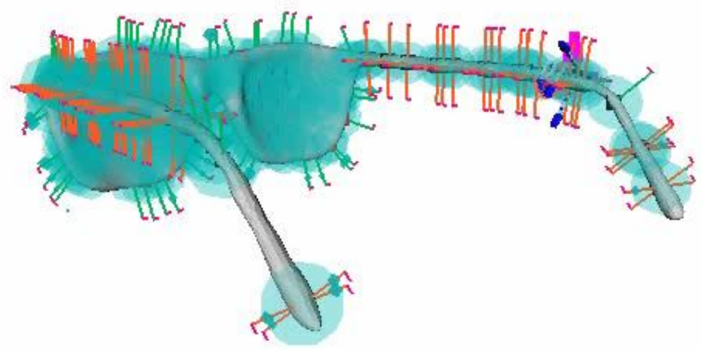


Extensions



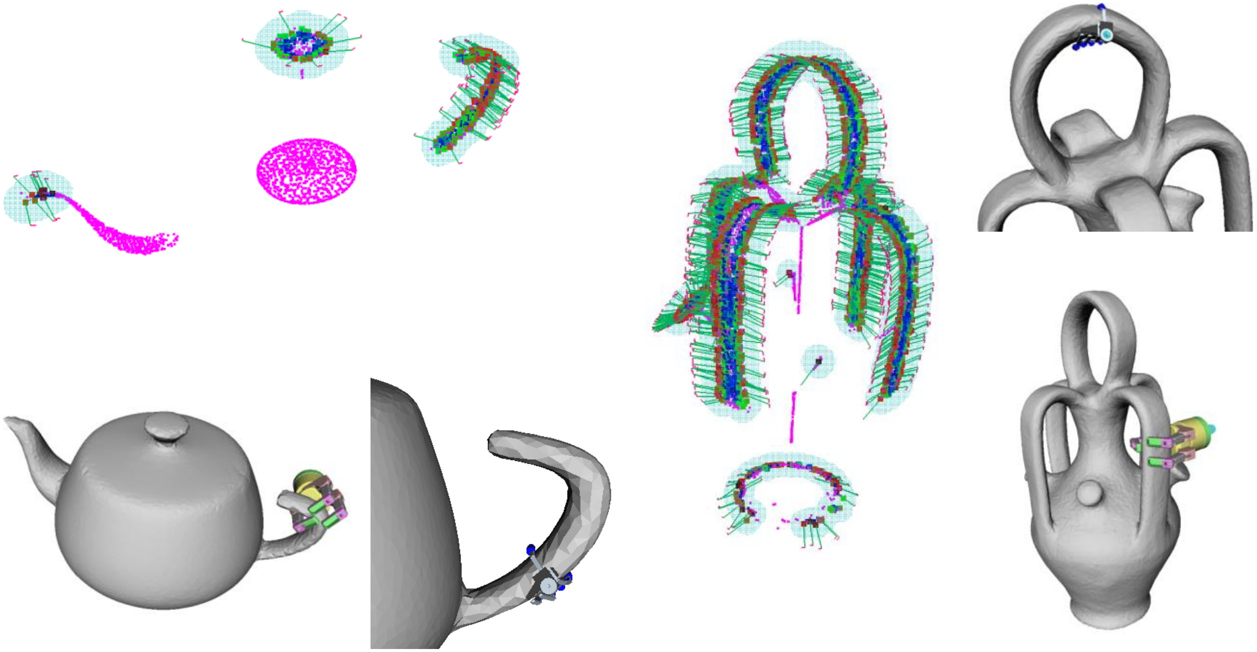


Extensions



More than 50% of grasps are force closed

Extensions : Grasping Handles

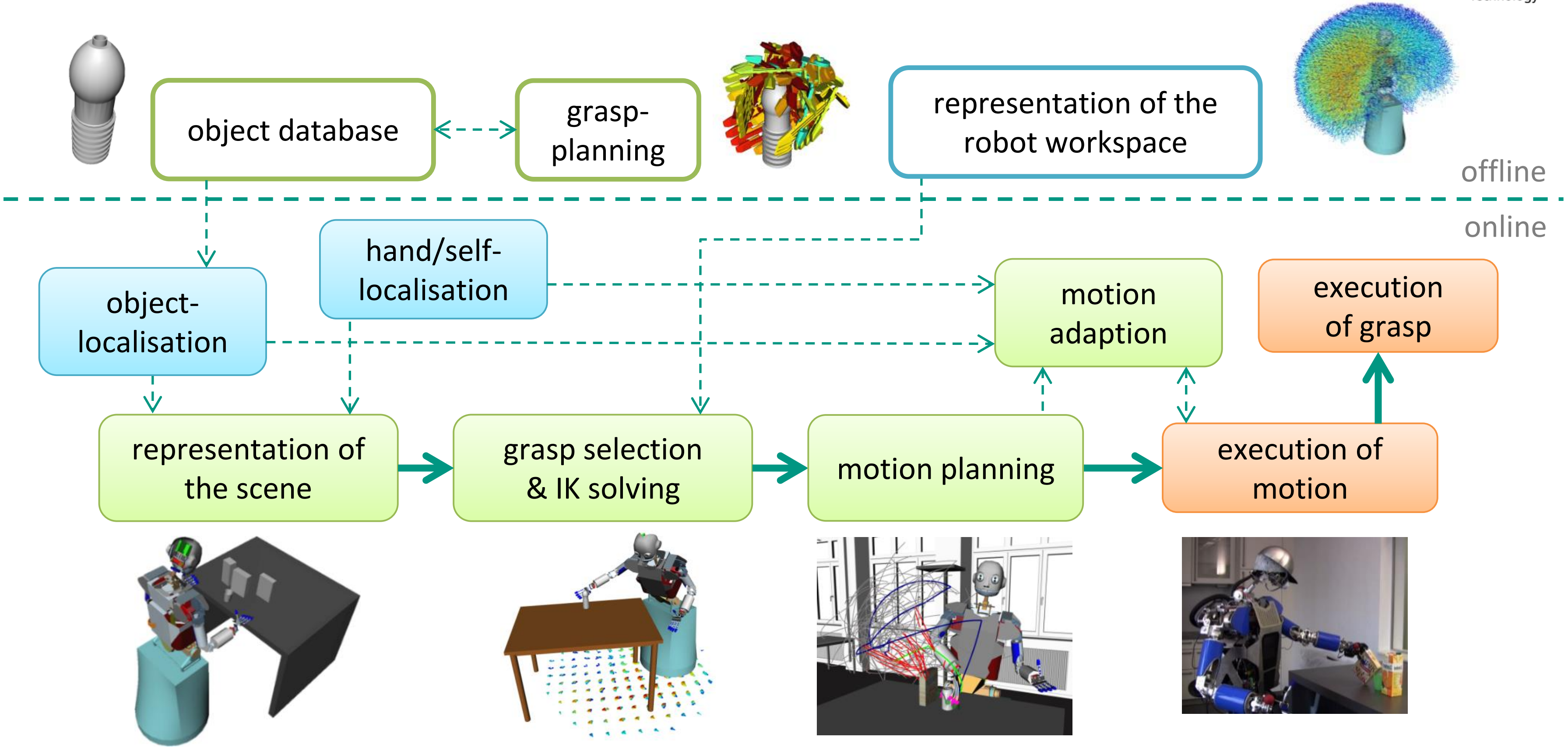


Content

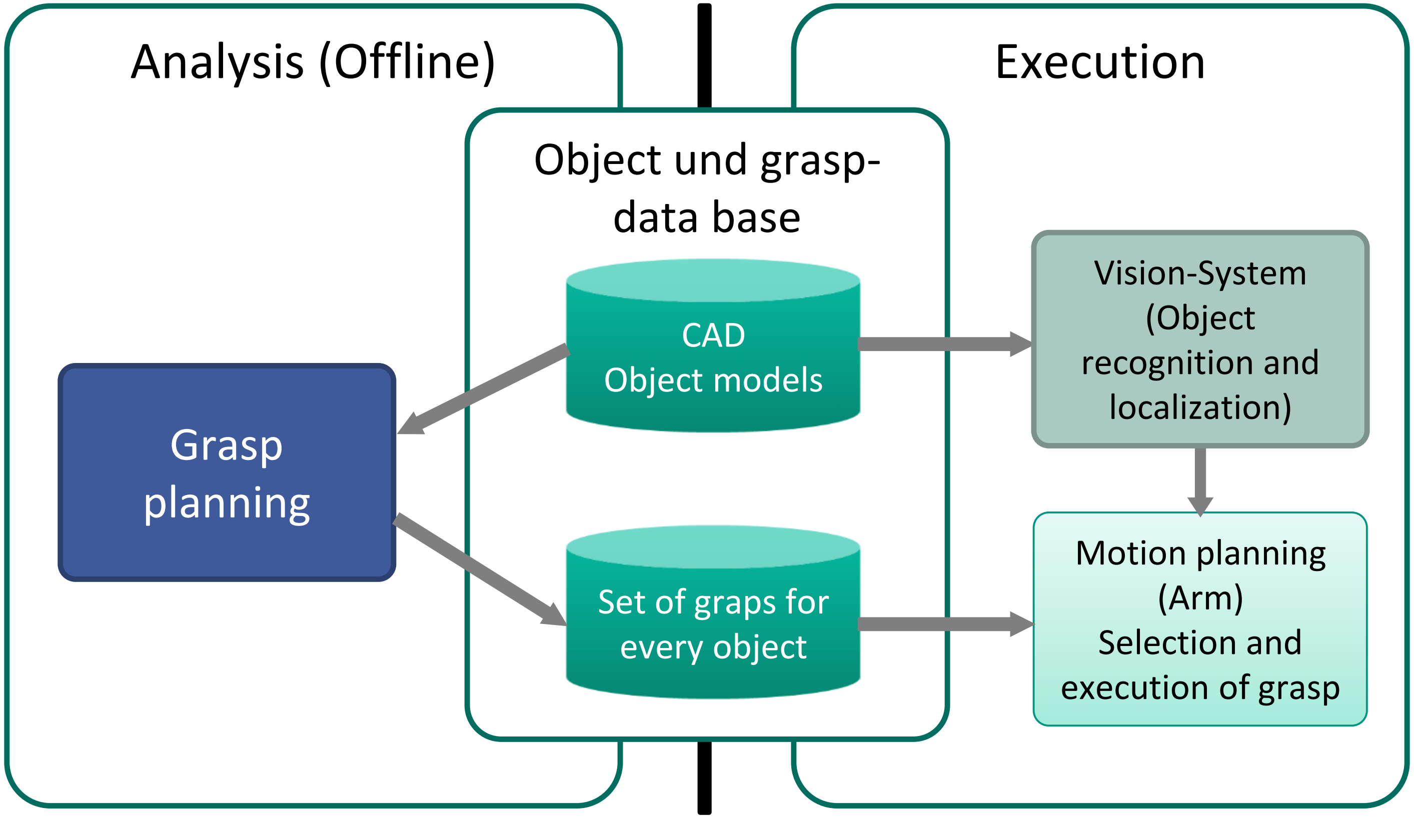
- Motivation
- Grasp Taxonomies
- Contact Models and Grasp Wrench Space
- Grasp Planning and Grasp Synthesis
- **Examples: Grasping with ARMAR**

Grasping with ARMAR

ARMAR-Grasping Pipeline (Known Objects)



ARMAR – Grasping Known Objects

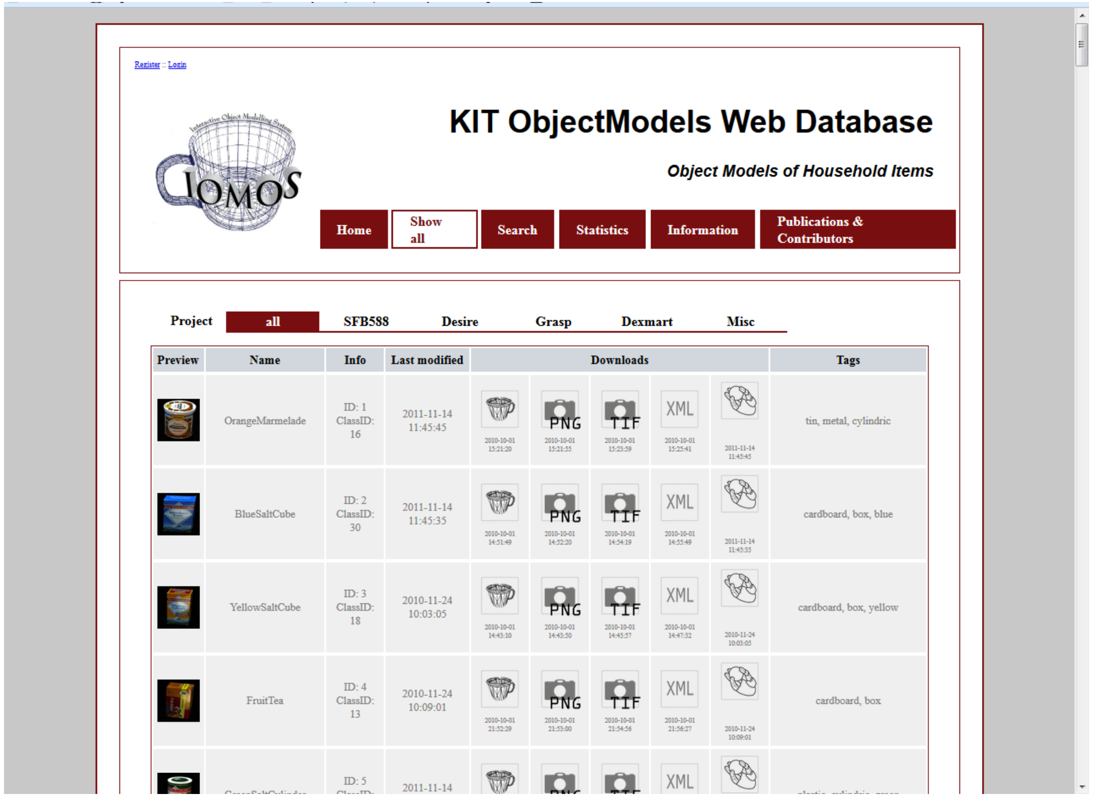


ARMAR – Object representation

- Household objects

Representation	Source	Usage
Point cloud	Object Modeling Center	Grasping based on box decomposition
Triangle mesh	Point cloud, simplified	Collision detection and visualization
Textured mesh	Additional textures	Visualization (vision simulation possible)
Vision data	Real images, synthetic views	Object detection and localization

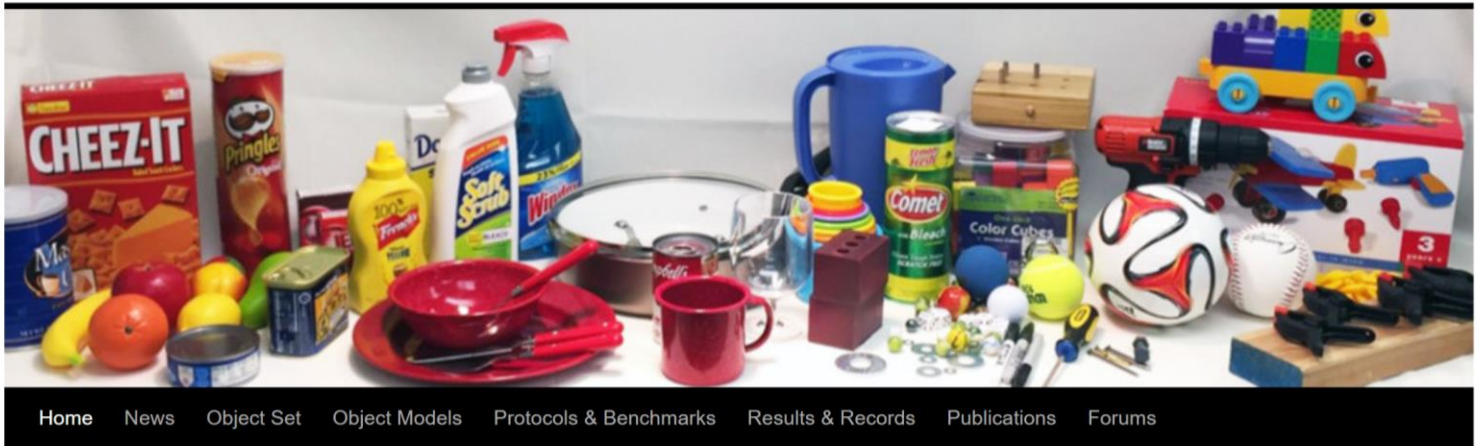
KIT Object Models Database



Preview	Name	Info	Last modified	Downloads					Tags
	OrangeMarmelade	ID: 1 ClassID: 16	2011-11-14 11:45:45	2010-10-01 15:21:20	PNG 2010-10-01 15:21:55	TIF 2010-10-01 15:23:59	XML 2010-10-01 15:25:41	2011-11-14 11:45:45	tin, metal, cylindric
	BlueSaltCube	ID: 2 ClassID: 30	2011-11-14 11:45:35	2010-10-01 14:51:49	PNG 2010-10-01 14:52:20	TIF 2010-10-01 14:54:19	XML 2010-10-01 14:55:49	2011-11-14 11:45:35	cardboard, box, blue
	YellowSaltCube	ID: 3 ClassID: 18	2010-11-24 10:03:05	2010-10-01 14:43:10	PNG 2010-10-01 14:43:50	TIF 2010-10-01 14:45:57	XML 2010-10-01 14:47:32	2010-11-24 10:03:05	cardboard, box, yellow
	FruitTea	ID: 4 ClassID: 13	2010-11-24 10:09:01	2010-10-01 21:52:29	PNG 2010-10-01 21:53:00	TIF 2010-10-01 21:54:56	XML 2010-10-01 21:56:27	2010-11-24 10:09:01	cardboard, box
		ID: 5	2011-11-14				XML		

YCB Benchmarks – Object and Model Set

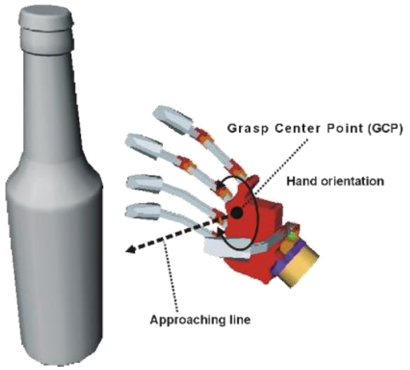
- Yale-CMU-Berkeley (YCB) Object and Model set
  https://www.ycbbenchmarks.com



Berk Calli, Aaron Walsman, Arjun Singh, Siddhartha Srinivasa, Pieter Abbeel, and Aaron M. Dollar, Benchmarking in Manipulation Research: The YCB Object and Model Set and Benchmarking Protocols, IEEE Robotics and Automation Magazine, pp. 36 – 52, Sept. 2015.

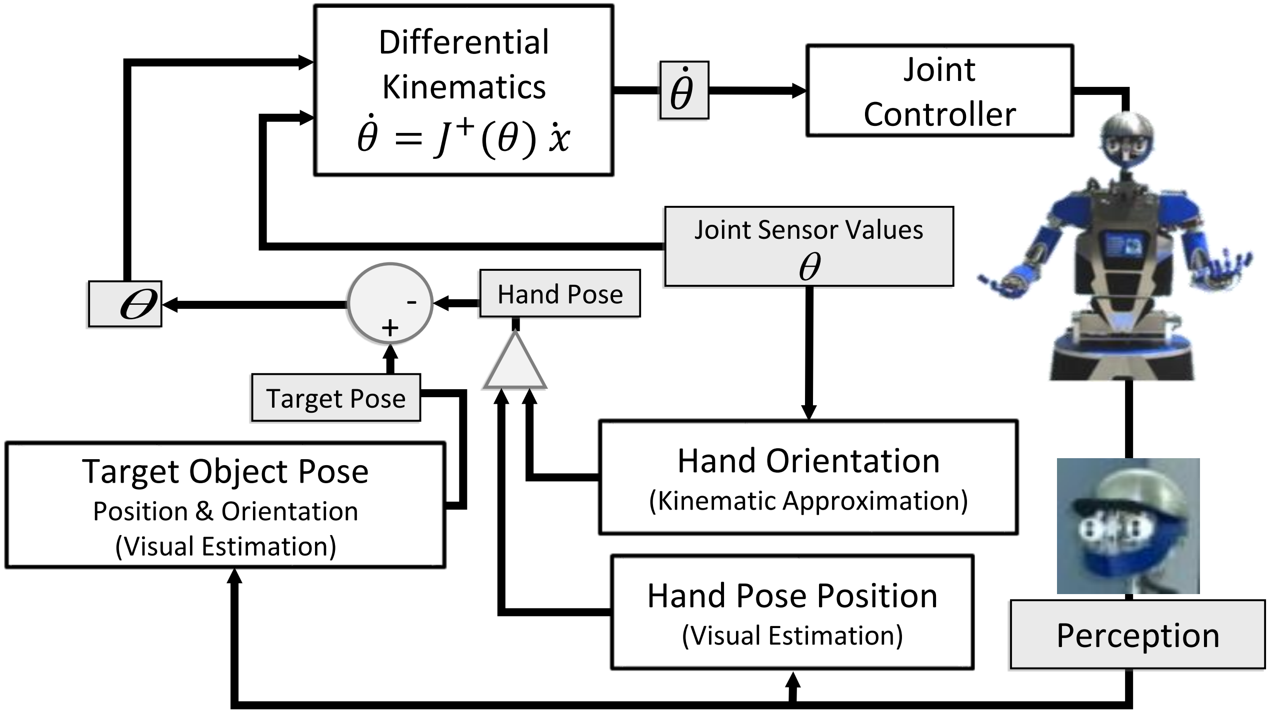
ARMAR – Offline Grasp Analysis

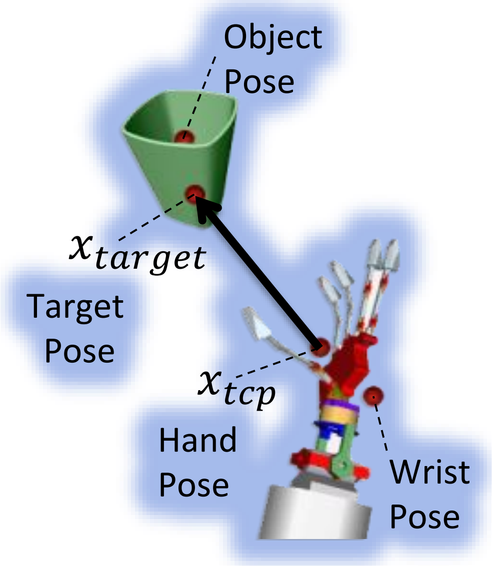
- Possible grasps are calculated offline for each object and saved together with the object
- Definition of a grasp:
  - **Grasp center point** on the object to which the Tool Center Point (TCP) is to be aligned
  - **Approach vector** describes the angle at which the hand approaches the grasp center point
  - **Orientation of the wrist** of the robot hand
  - **Initial finger configuration**
- Evaluation in simulation



Hook *Cylindrical* *Spherical* *Pinch grasp* *Three-finger grasp*

ARMAR – Execution: Position-based Visual Servoing





$$\dot{\theta} = J^{+}(\theta)\dot{x}$$

$$\delta^{t} = x^{t}_{vision} - x^{t}_{kinematic}$$

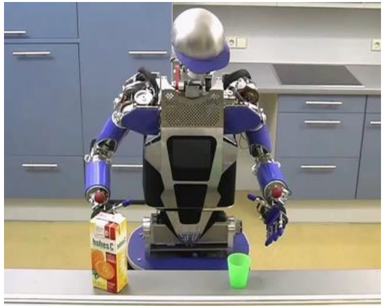
$$x^{t+1}_{tcp} = x^{t+1}_{kinematic} + \delta^{t}_{tcp}$$

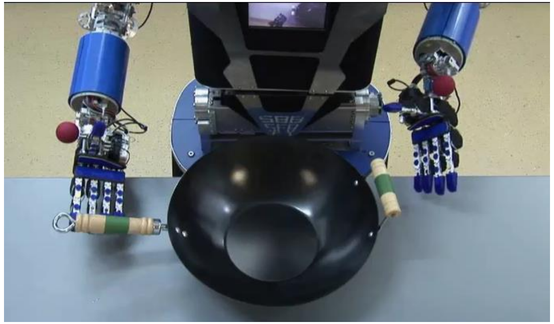
ARMAR-III in the RoboKITchen

45-minute task, more than 4500 times since 3rd February 2008

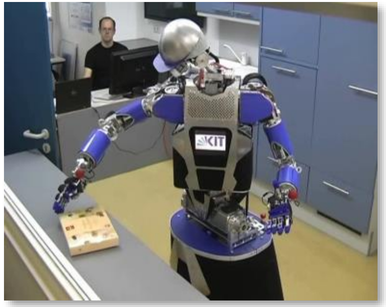
Advanced Grasping Capabilities

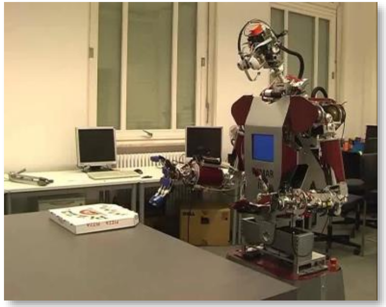
- *Bimanual* grasping and manipulation





- *Pre-grasp manipulation*





RAM 2012
IROS 2011
Humanoids 2010
Humanoids 2009
RAS 2008

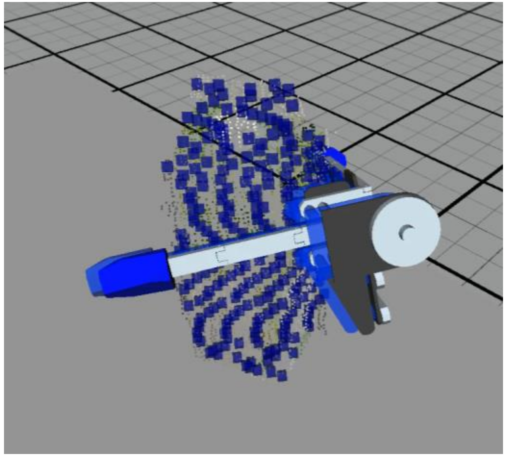
Task-Specific, Part-Based Grasping

- *Familiar objects*

Vahrenkamp, N., Westkamp, L., Yamanobe, N., Aksoy, E. E. and Asfour, T., Part-based Grasp Planning for Familiar Objects, Humanoids 2016

Grasping of Unknown Objects(I)

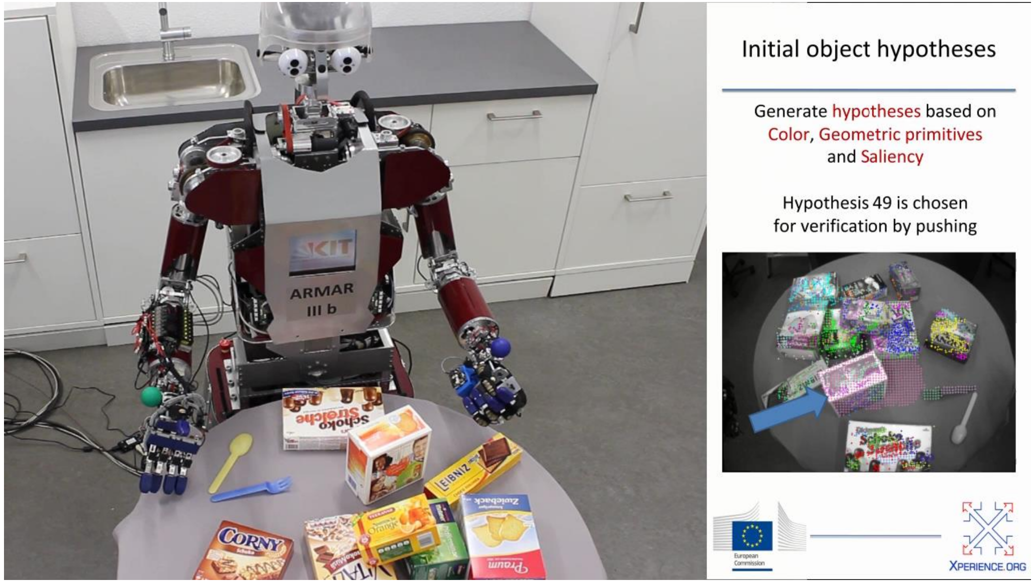
- CNN trained with depth images and 6D grasp information
- Only depth images during execution



Schmidt, P., Vahrenkamp, N., Wächter, M. and Asfour, T., *Grasping of Unknown Objects using Deep Convolutional Neural Networks based on Depth Images*, ICRA 2018

Grasping of Unknown Objects (II)

- *Active perception* for Discovery, Segmentation and Learning of objects



Schiebener, D., Ude, A. and Asfour, T., *Physical Interaction for Segmentation of Unknown Textured and Non-textured Rigid Objects*, ICRA 2014

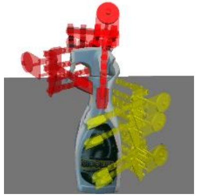
Understanding *Perception-Action Relations*

- **Affordace-based Manipulation:** Robot needs to understand (inter-)action possibilities in unkown environments.

Pohl, C., Hitzler, K., Grimm, R., Zea, A., Hanebeck, U. D. and Asfour, T., *Affordance-Based Grasping and Manipulation in Real World Applications*, IROS 2020

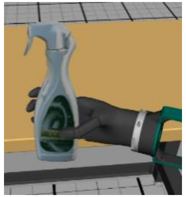
Re-grasping, Handover

Select initial grasp
- Inverse kinematics
- RRT-connect

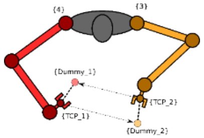


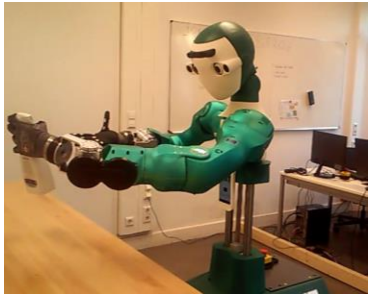
Grasp object

Detect object pose in hand
- Online in-hand object tracking (dbot)



Calculate re-grasp pose
- Inverse kinematics with bi-manual pose constraint

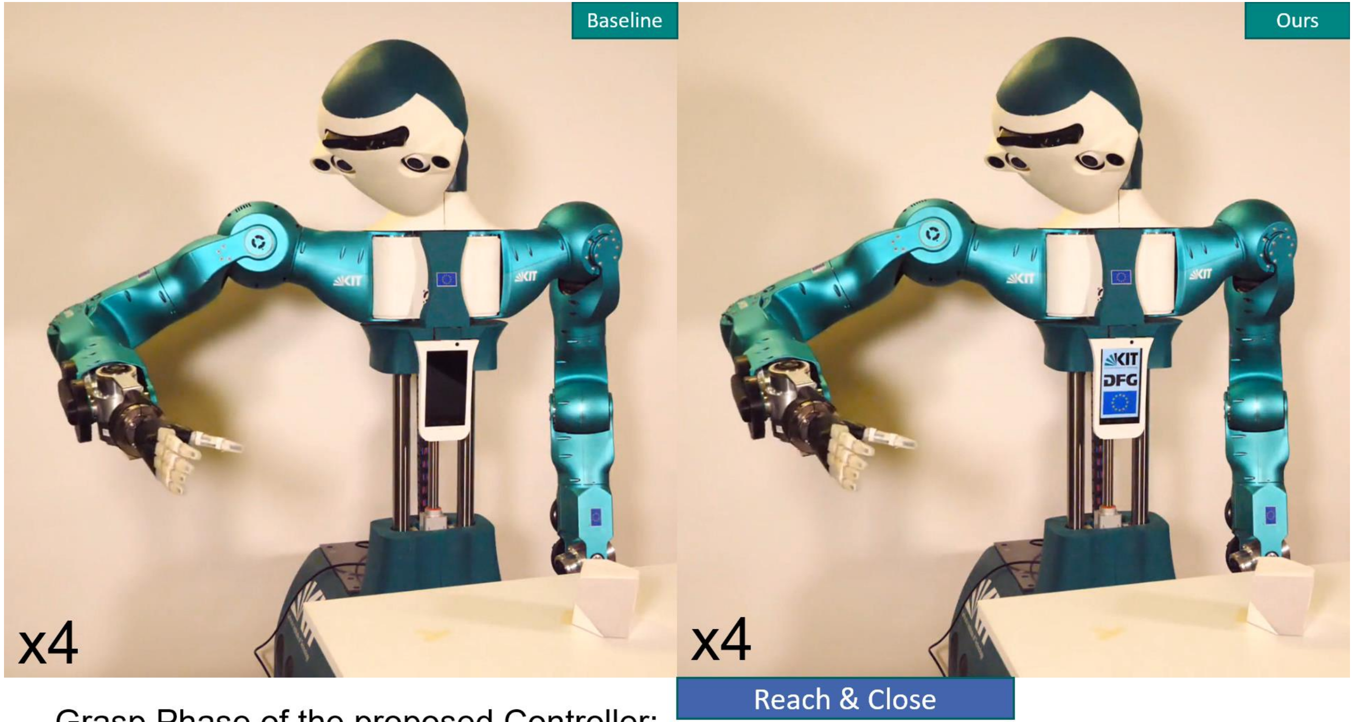




Re-grasp object

Depth Based Object Tracking Library (dbot)
https://github.com/bayesian-object-tracking/dbot

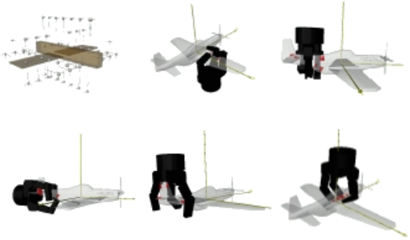
Grasping with Soft Hands and Tactile Feedback

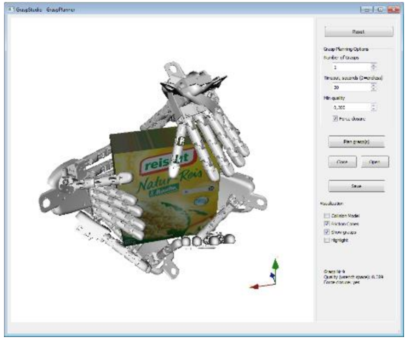


Grasp Phase of the proposed Controller:

Software

- **GraspIt!**
  - Developed by Columbia University Robotics Group
  - C++ Source Code: http://graspit-simulator.github.io/
- **Simox**
  - Developed by H2T, KIT
  - C++ Source Code: https://gitlab.com/Simox/simox
  - Documentation: https://gitlab.com/Simox/simox/wikis
  - VirtualRobot: Robot Simulation
  - Saba: Motion Planning Library
  - GraspStudio: Grasp Planning Library





Thank you very much!

- Grasping is still an unsolved problem in robotics
- More on the topic of grasping in the Robotics II lecture (summer term)

ADDITIONAL MATERIAL

KIT Object Database

https://h2t-projects.webarchiv.kit.edu/Projects/ObjectModelsWebUI

- 3D Data
  - Laserscanner Minolta „VI-900“
  - Active triangulation
  - Resolution: 640 x 480 Pixel
  - Precision: < 0.2 mm
  - Various formats / Resolutions
- Color images of a stereo camera
  - Allied Vision Technologies „Marlin 145C2“
  - Resolution: 1392 x 1038 Pixel
- XML Representation
  - Metadata
  - Parser

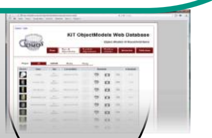
ARMAR – Object modeling

Object Modeling Center

- Generation of 3D point clouds with laser scanner
- Post-processing using triangulation in high-resolution triangular meshes
- Saving in various formats (Open Inventor, VRML, Wavefront)
- Generation of various object views using stereo cameras

Developed in „Sonderforschungsbereich 588 Humanoide Roboter“

Web data base



https://h2t-projects.webarchiv.kit.edu/Projects/ObjectModelsWebUI

ARMAR – Object detection and localization

Colored Objects **(IROS 2006, IROS 2009)**

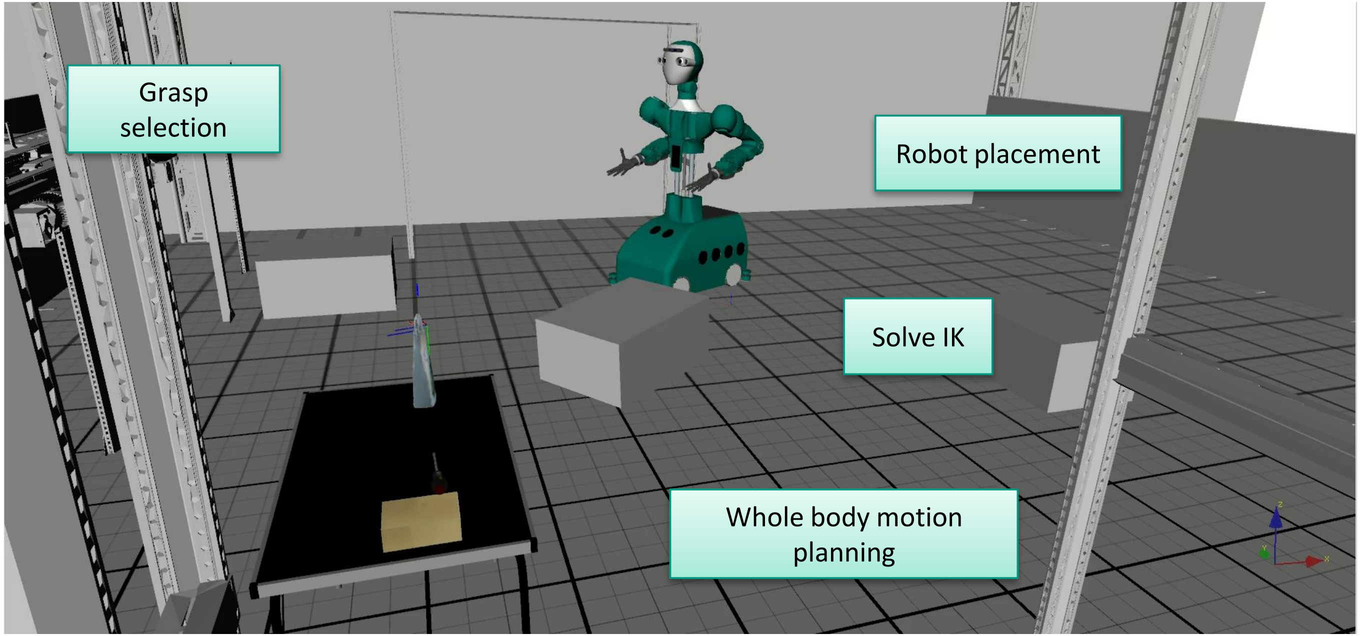
- Segmentation by color
- View-based recognition with a global approach
- Model-based generation of views
- Combination of stereo vision and stored orientation information for 6D pose estimation

Textured Objects **(Humanoids 2006, IROS 2009)**

- Recognition using local properties
- Calculation of consistent properties related to the object pose using Hough transform
- 2D localization using pixel correspondences
- 6D pose estimation using stereo vision

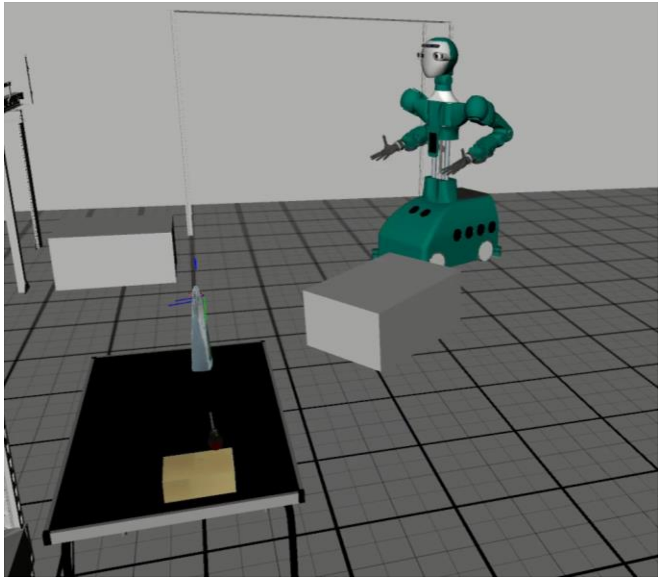
Correspondences between learned view and view in scene

Grasping and Manipulation Pipeline: Methods



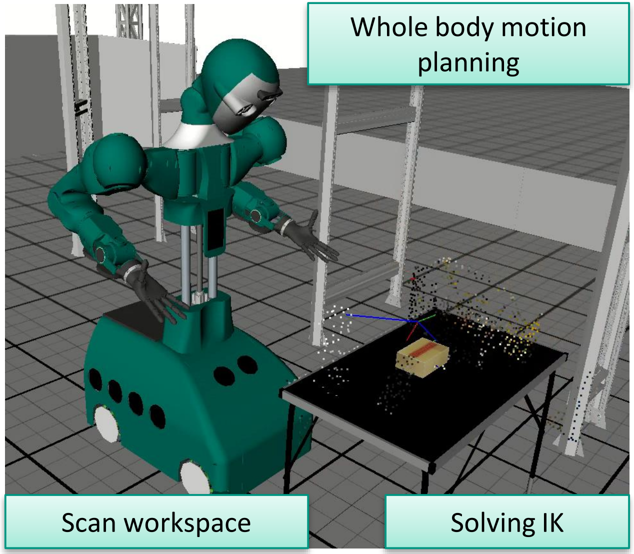
Video slowed down for visualization

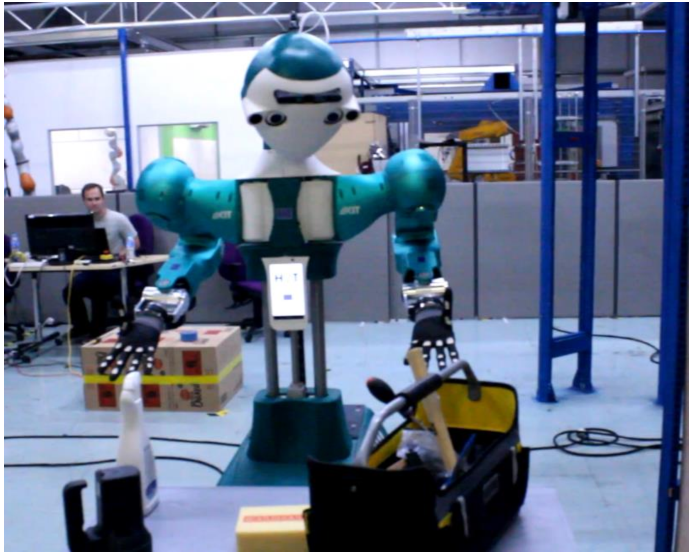
Execution in Known Environments



Execution in Partially Known Environments

- Based on environmental models and point clouds



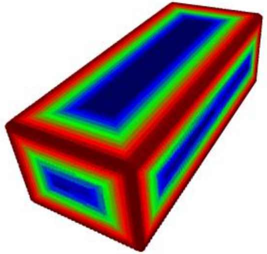


Video slowed down for visualization

Medial Axis Planner: Limitations

- Problems with slice structures
  - For complex objects, there are too many different kinds of slice structures, and each one has to be considered separately.
  - Too many parameters, heuristics, threshold constants…
- Better solution (Grid of medial spheres):
  - Consider complete medial axis transform (MAT = spheres including radii)
  - No slicing of the MA
    - Instead: Principal component analysis of sphere centers in a local environment around a query sphere
  - Only two cases for candidate grasp generation:
    - Spheres located on local symmetry axis → generate candidate grasps
    - Spheres located on the rim of local symmetry plane → generate candidate grasps
    - Spheres located inside a local symmetry plane (not interesting for grasping)

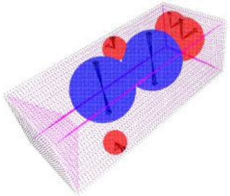
Grid of medial spheres grasp planner



- Based on the medial axis transform
  ⇒ Contains object‘s symmetry properties
- In addition: efficient access to spheres in local neighborhood (grid-based spatial indexing):

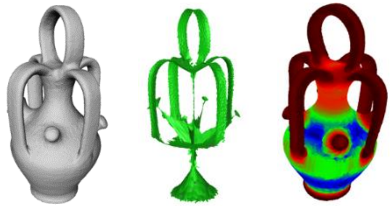
$$\begin{pmatrix} i_x \\ i_y \\ i_z \end{pmatrix} = \begin{pmatrix} \lfloor n_x(x - x_{min})/(x_{max} - x_{min}) \rfloor \\ \lfloor n_y(y - y_{min})/(y_{max} - y_{min}) \rfloor \\ \lfloor n_z(z - z_{min})/(z_{max} - z_{min}) \rfloor \end{pmatrix}$$

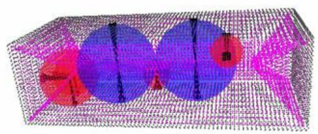
- Attributes of each sphere:
  - Center, radius, points where the sphere touches the object‘s surface
  - Object angle: maximum angle included by the sphere‘s center and two surface points touched by the sphere.
  - Example:
    - Blue spheres: object angle ~180°
    - Red spheres: object angle ~ 90°



Object representation for grasp planning

- *Grid of medial spheres*
  - Based on the Medial Axis Transform (MAT) [Blum67]
  - Contains object‘s symmetry properties
  - Additional: efficient access to spheres in local neighborhood
  - Attributes of each sphere:
    - Center
    - Radius
    - Points where the sphere touches the object‘s surface
    - Object angle: maximum angle defined by the sphere‘s center and two surface points touched by the sphere

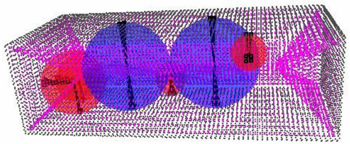




Blue spheres: object angle ~180°
Red spheres: object angle ~90°

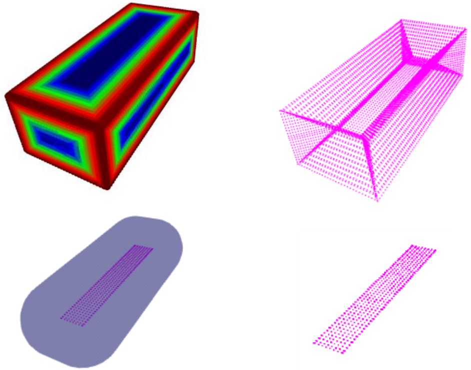
Selecting spheres for grasp planning

- **Which spheres are important for grasp planning?**
- Rough structure (occupied volume) vs. surface details of the object
- Goals:
  - Exploit local symmetry planes / axes for grasp planning
  - Generate grasps with two opposed virtual fingers
- Main parameters:
  - Object angle
  - Sphere radius
- Grasp planning:
  - Consider only spheres with object angle >= 120°
  - This removes edges and corners of the object
  - Symmetry planes and axes are preserved



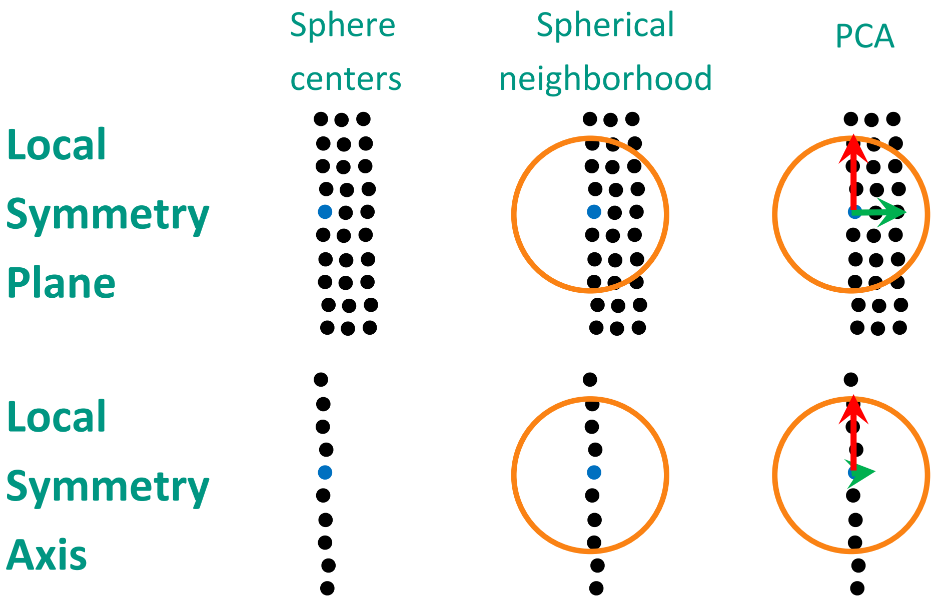
Grid of medial spheres

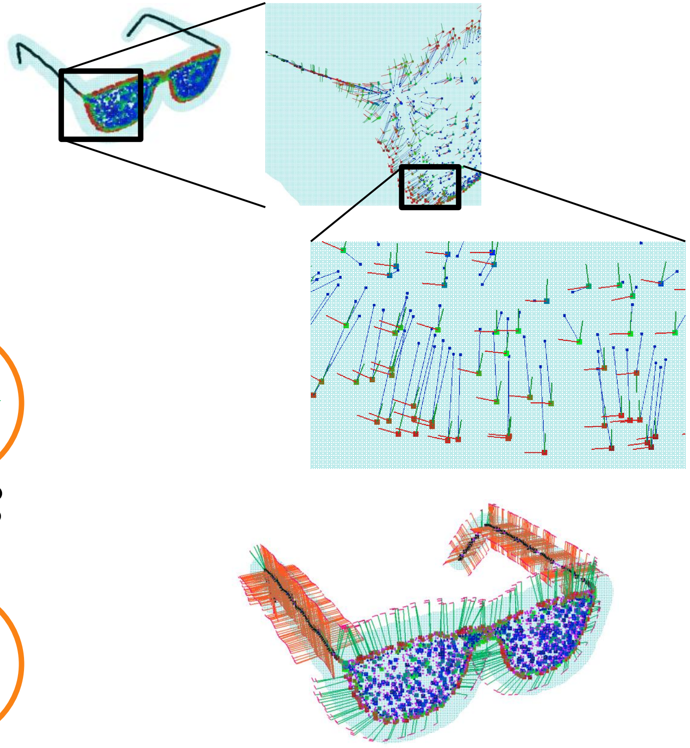
Medial Axis



Grasp Planning Algorithm Using MAT and PCA

Estimate symmetry properties of sphere centers in each sphere‘s local neighborhood

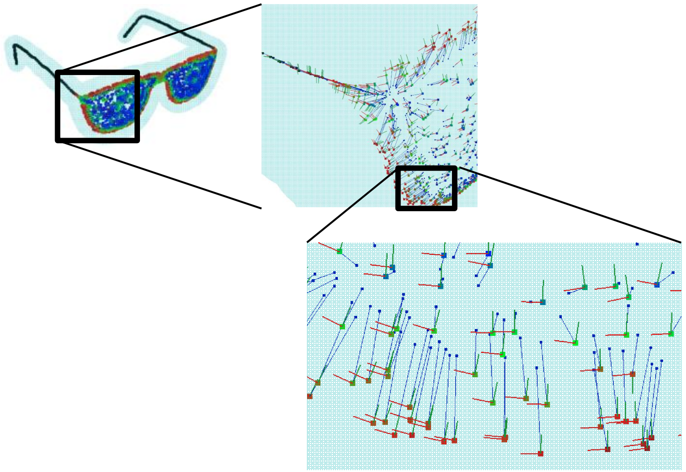


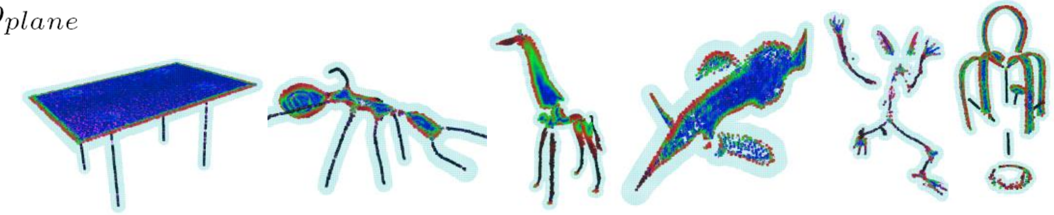


Hand approach directions

Analyzing an Object‘s Symmetry Properties

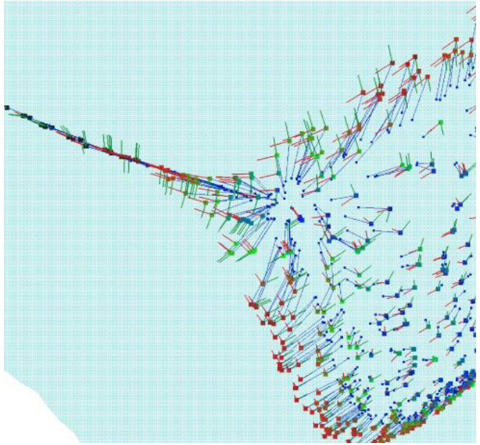
- Estimate symmetry properties of sphere centers in each sphere‘s local neighborhood
- Principal Component Analysis:
  - Directions of eigenvectors
  - Ratio of eigenvalues $\rho_{ev} = \frac{\lambda_2}{\lambda_1}$
- Classification of spheres:
  - On local symmetry axis $\rho_{ev} \leq \rho_{axis}$
  - On local symmetry plane
    - At the rim $\rho_{axis} \leq \rho_{ev} \leq \rho_{plane}$
    - Inside the plane $\rho_{ev} > \rho_{plane}$

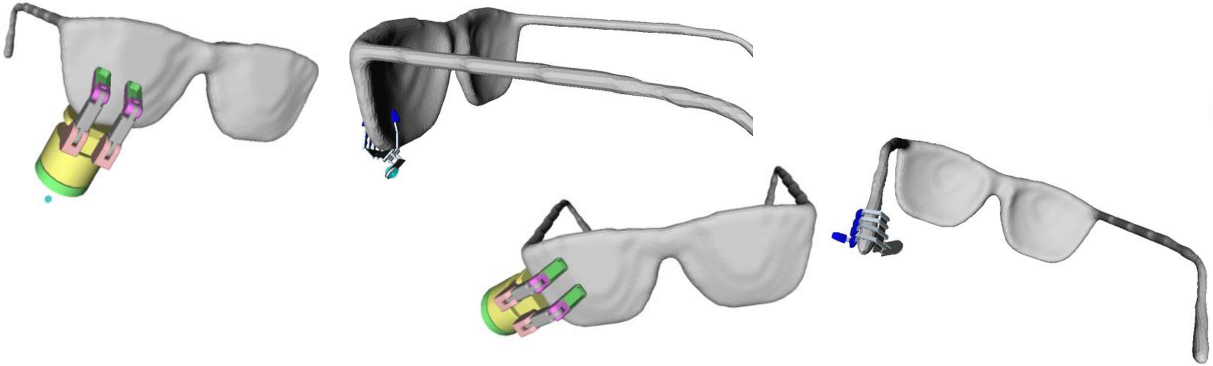


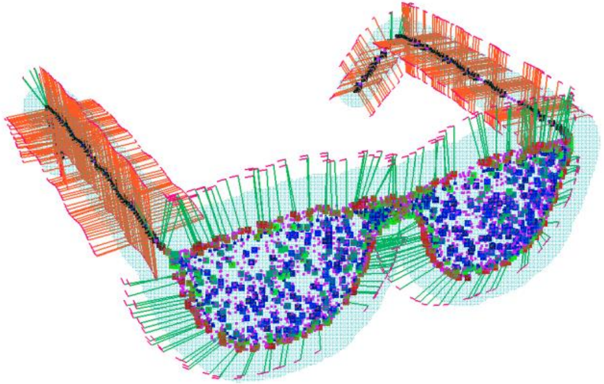


Generating Candidate Grasps

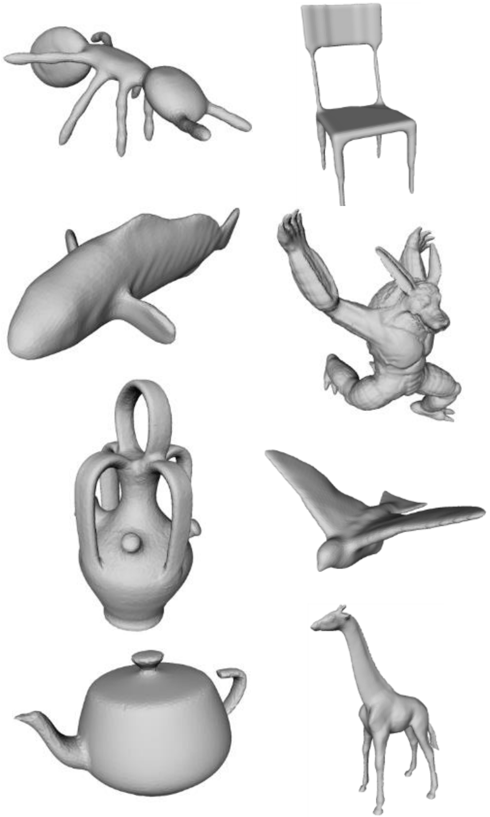
- Symmetry axis
  - Hand approach directions perpendicular to local symmetry axes
- Rim of symmetry plane
  - Hand approach directions perpendicular to local symmetry planes

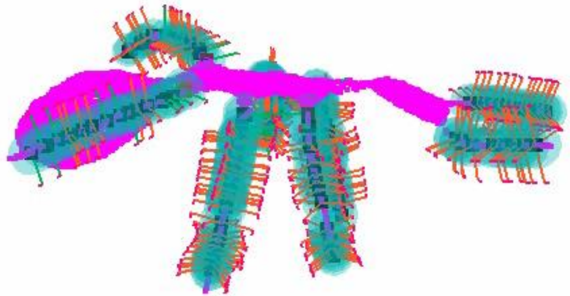






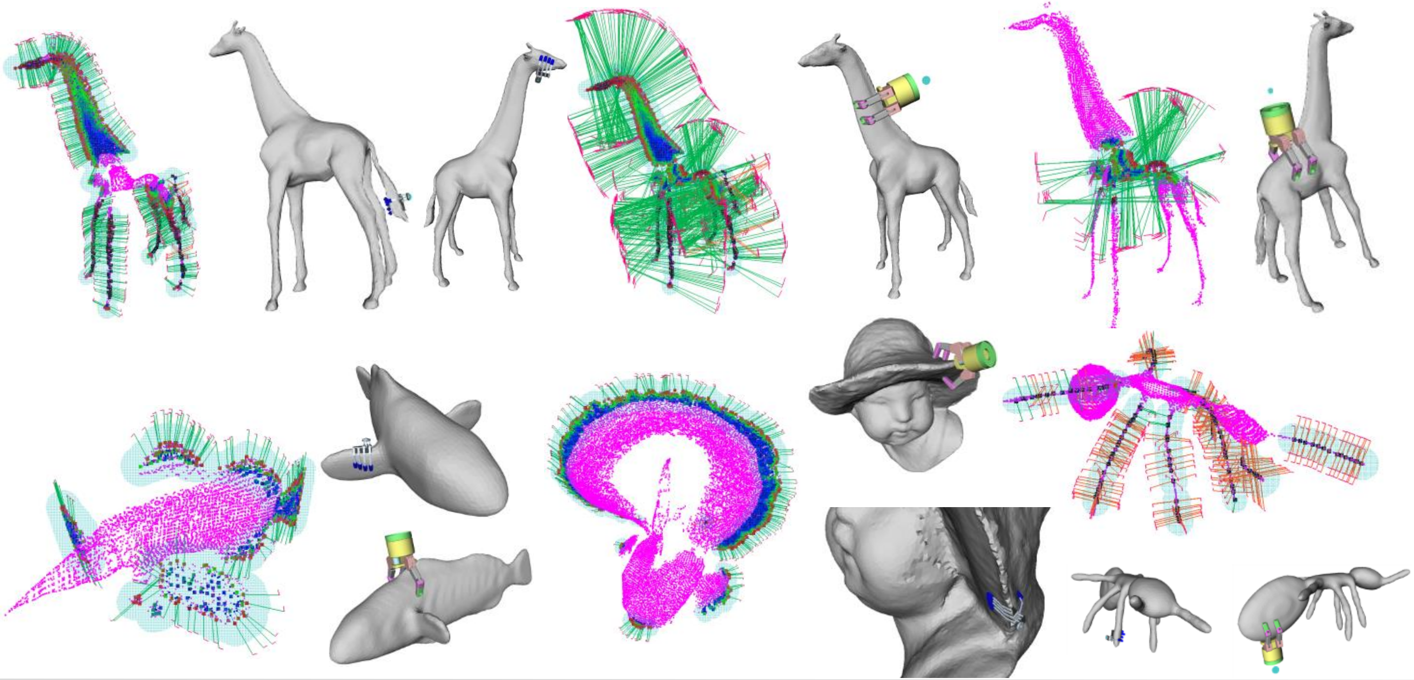
Candidate Grasps: Some Examples



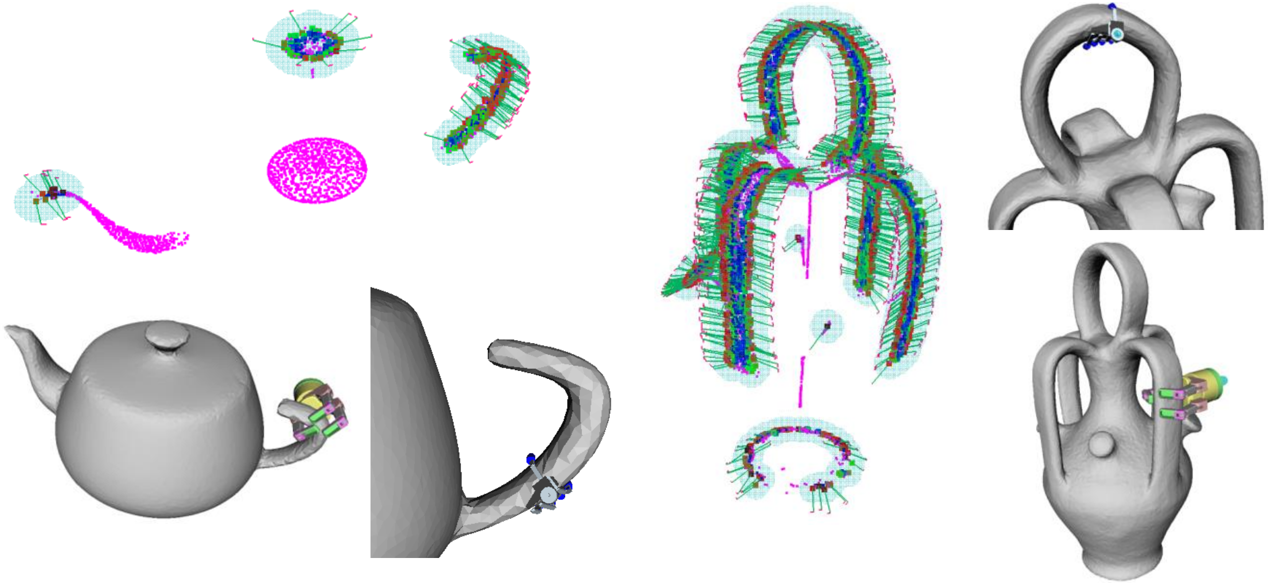


Advantages: Hand Size vs. Object Size

- Respect maximum sphere diameter graspable by the robot hand
- Optional: do not generate grasps for “small“ spheres



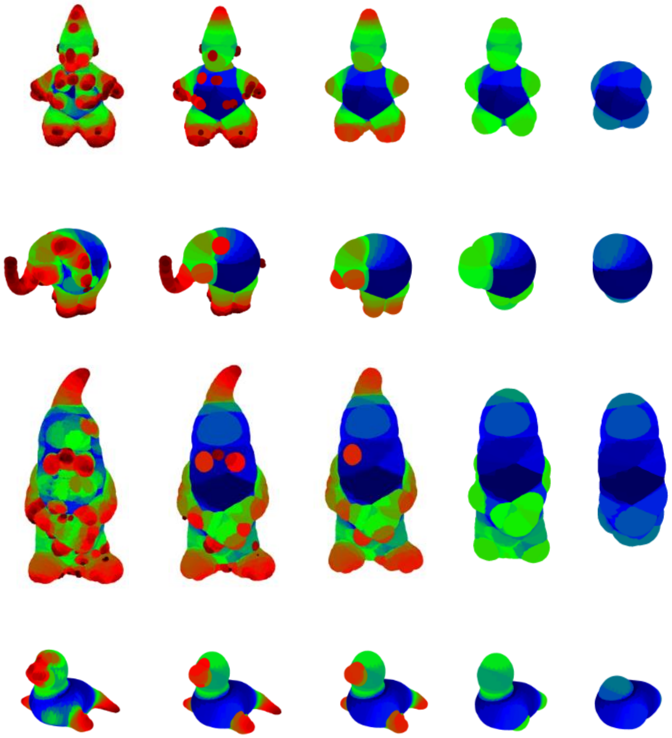
Advantages: Grasps on Handles



- For big objects, the algorithm finds many grasps at the handles
  - Simply due to geometric considerations, as the hollow bodies are too big to grasp
  - No semantic knowledge (task dependency) necessary

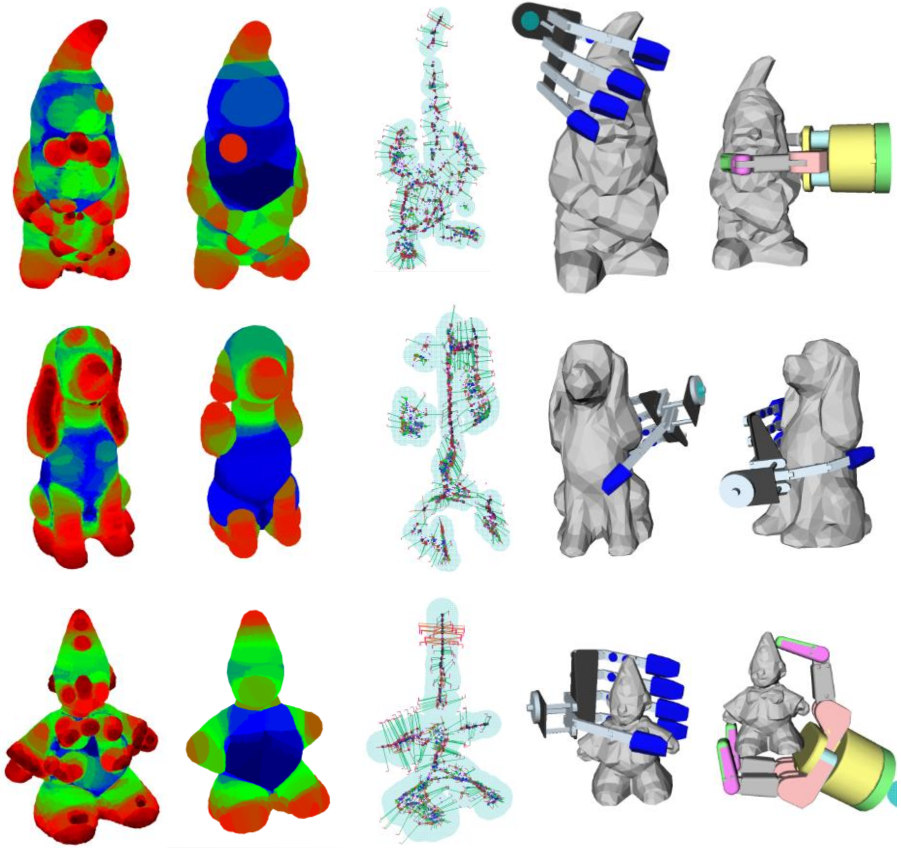
Dealing with Surface Details

- Surface details might disturb the planner.
  - Example: We don‘t want to plan grasps for the lawn gnome‘s nose.
- Solution: Discard small spheres and spheres with small object angles
  - Column 1: All spheres
  - Column 2: Only spheres with object angle $\geq 120°$
  - In addition: Discard spheres with small radii compared to biggest sphere in the object:
    - Column 3: r < 0.3 r_{max}
    - Column 4: r < 0.5 r_{max}
    - Column 5: r < 0.7 r_{max}
    - (r_{max} : Radius of the biggest sphere in the object)

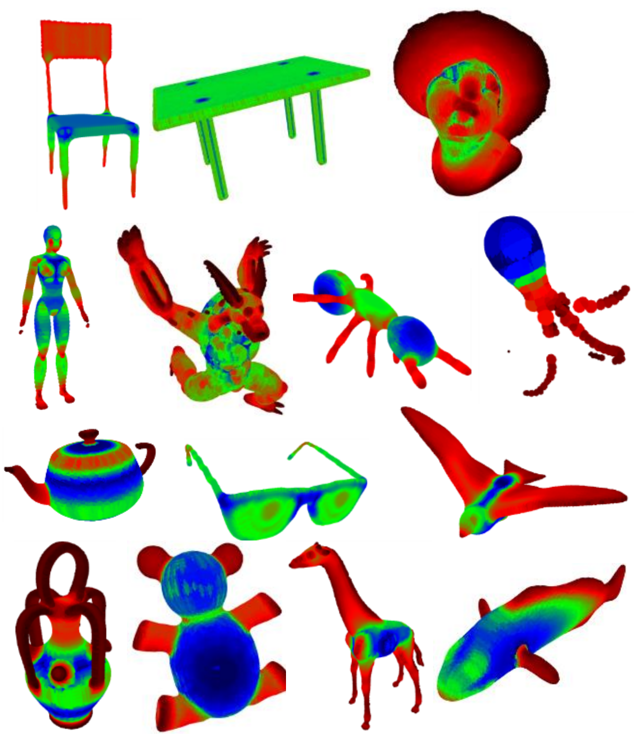


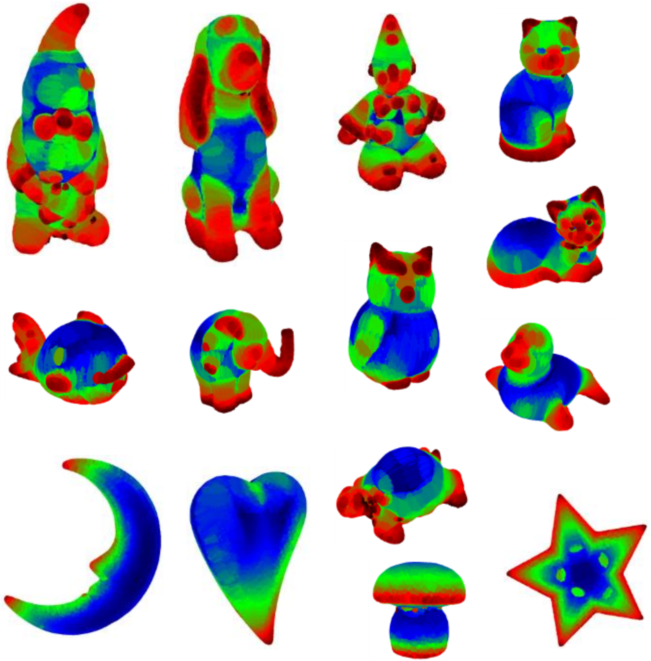
Advantages: Surface Details

- How to deal with surface details?
- Solution: discard “small“ spheres
- Planner considers only rough geometry of the object



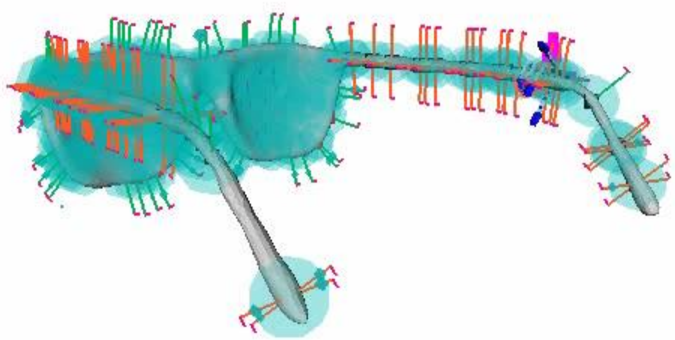
Object Sets for Testing





- Chen Mesh Segmentation Benchmark [Chen2009]
- Real objects

Force-closure Testing



Results

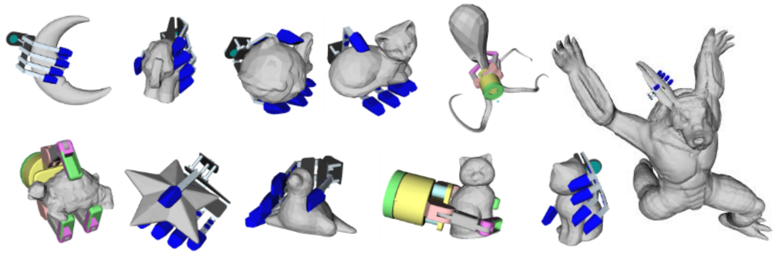
Chen benchmark

Objects	ARMAR-III hand		Barrett hand	
	scale 1.0	scale 0.5	scale 1.0	scale 0.5
1 Female doll	71.3%	54.6%	53.13%	37.9%
41 Glasses	93.9%	7.8%	73.7%	10.7%
81 Ant	94.4%	71.1%	61.3%	45.7%
101 Chair	89.6%	49.2%	73.9%	72.2%
125 Octopus	53.7%	55.2%	26.9%	44.7%
141 Table	91.9%	92.5%	94.6%	85.0%
161 Teddy	100.0%	83.3%	86.7%	51.2%
225 Fish	76.5%	83.3%	68.4%	81.1%
245 Bird	75.0%	68.3%	75.0%	65.6%
290 Monster	70.5%	64.7%	67.8%	38.2%
305 Bust	50.0%	70.0%	100.0%	92.9%
361 Vase	76.8%	65.3%	69.6%	55.1%
379 Tea kettle	78.9%	63.2%	75.7%	31.3%
390 Giraffe	85.5%	68.3%	71.4%	56.0%

Real objects

Objects	ARMAR-III hand	Barrett hand
1001 Clown	63.5%	61.2%
1002 Elefant	75.3%	76.0%
1003 Owl	78.0%	68.2%
1004 Spheric fish	59.0%	78.3%
1005 Lawn gnome	53.1%	57.7%
1006 Heart	89.0%	77.0%
1008 Dog	63.7%	69.2%
1009 Sitting cat	64.9%	59.5%
1010 Lying cat	80.7%	80.7%
1012 Moon	58.9%	64.4%
1013 Mushroom	80.0%	55.5%
1014 Turtle	57.1%	70.3%
1015 Seal (Seehund)	73.5%	59.2%
1016 Star	44.4%	66.7%

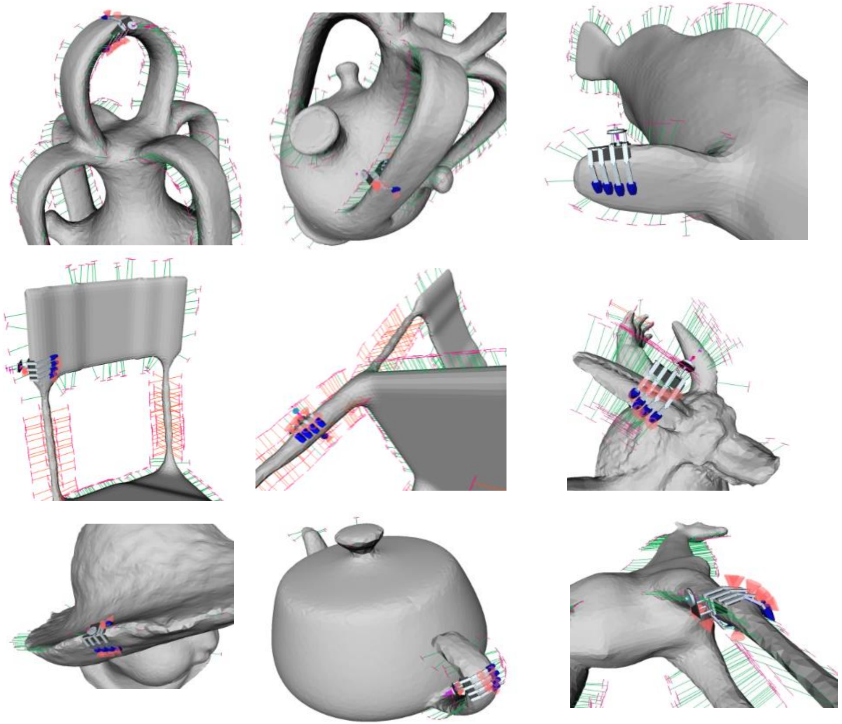
- Experiments:
  - Hand models: ARMAR-III, Barrett
  - Object models:
    - Chen benchmark
      - 100% scaled objects
      - 50% scaled objects
    - Real objects
- Results:
  - Most (> 50%) of the generated candidates are force-closure grasps



Results (Power Grasps for Big Objects)

Comparison to method based on surface normals [Berenson07]

Planner type	MAT	Berenson [34]
$r_{o,min}$	0.0	n.a.
r_N	0.06m	n.a.
Glasses	67.1%	14.1%
Ant	95.3%	15.3%
Chair	88.0%	14.4%
Fish	57.3%	0.8 %
Bird	66.8%	2.7 %
Monster	63.0%	4.7 %
Bust	41.6%	0.0 %
Vase	75.7%	3.6 %
Tea kettle	87.5%	1.9 %
Giraffe	89.5%	20.0%
Average	**73.2%**	**7.7 %**



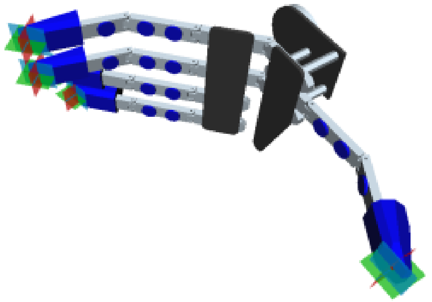
Summary

- *Grid of medial spheres* object representation:
  - Based on the medial axis transform
  - Volumetric approximation
  - Arbitrary level of detail
  - Symmetry properties as part of the object representation

- Grasp planning algorithm:
  - For arbitrarily shaped objects
  - Generates geometrically meaningful candidate grasps
  - Further advantages:
    - Hand size and object size considered
    - Grasps on handles simply due to geometric considerations
    - Surface details can be ignored, if necessary
  - High ratio of force-closure grasps

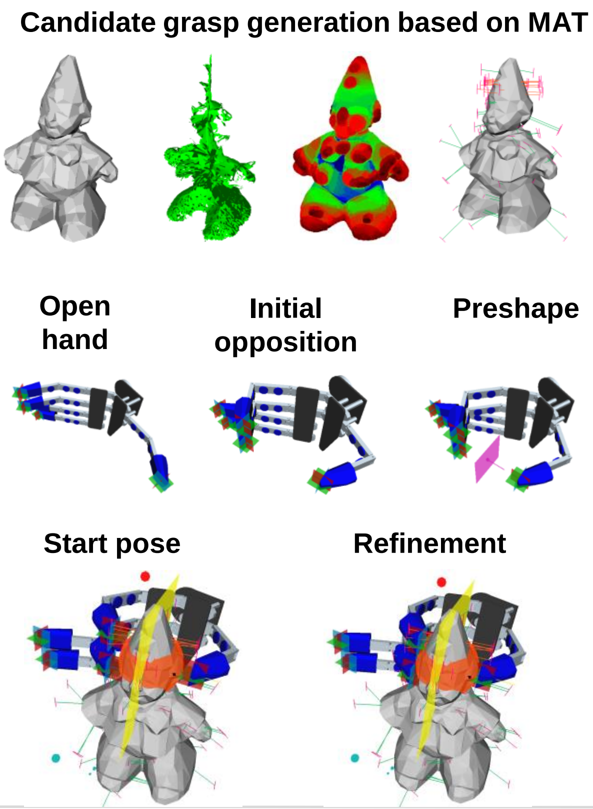
Planning precision grasps

- Problem:
  - Planning precision grasps requires simultaneous determination of hand pose and finger joint angle
- Idea:
  - Define desired contact areas at fingertips
  - Preshape and first guess for hand pose based on:
    - Object symmetry information
    - Contact areas
  - Refine hand pose and configuration while closing the fingers



Algorithm: Planning precision grasps

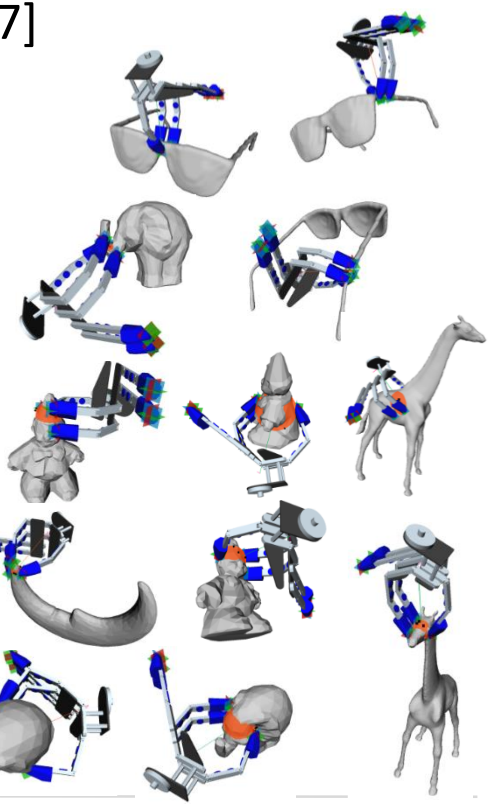
1. Generate candidate grasps using the Medial Axis Transform (MAT).
2. Compute preshape and first guess for hand pose:
   - Move hand to a configuration with initial opposition of predefined parallel contact areas.
   - Move the hand to the preshape configuration.
   - Extract grasp center transformation
   - Set hand to start pose
   - Discard candidate, if the hand is initially in collision with object.
3. Refinement of hand pose and configuration:
   - Close fingers, while maintaining parallel orientation of desired contact areas on the fingertips.
   - Compensate for translational fingertip movement by moving the TCP in the opposite direction.
4. Test for force-closure

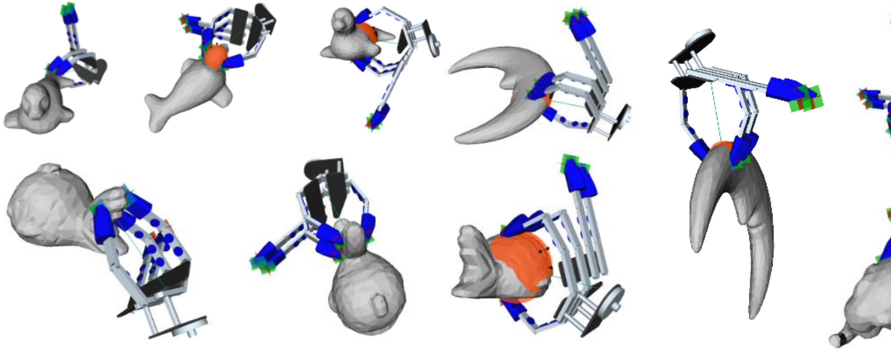


Results (Precision Grasps)

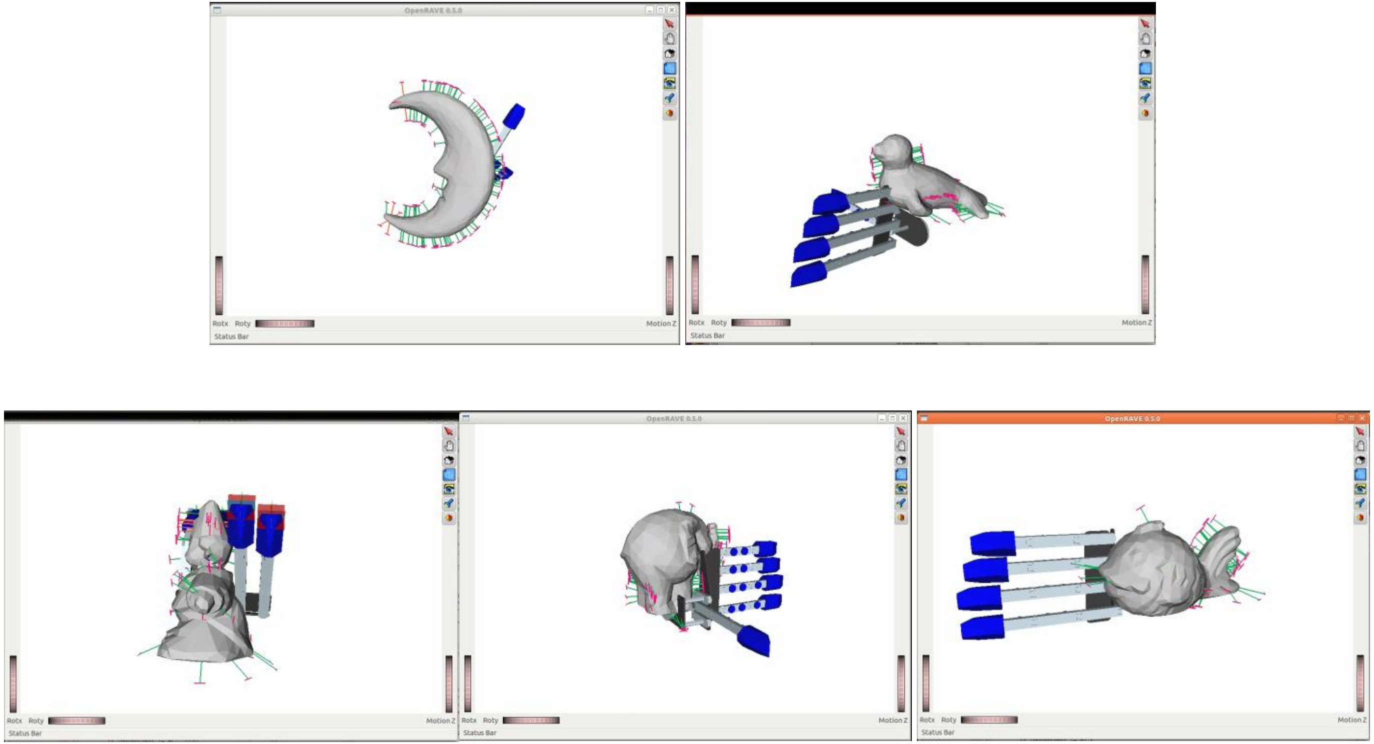
Comparison to method based on surface normals [Berenson07]

Planner type	MAT				Berenson [34]
$r_{o,min}$	0.0		$0.3r_{o,max}$		n.a.
r_N	0.01m	0.03m	0.01m	0.03m	n.a.
Clown	27.7%	15.8%	41.3%	29.9%	10.8%
Elefant	4.3 %	5.9 %	2.3 %	3.8 %	15.9%
Spheric fish	40.3%	24.7%	46.8%	29.6%	27.2%
Moon	83.0%	91.6%	84.2%	92.3%	5.3 %
Seal	59.8%	58.7%	67.1%	61.1%	10.8%
Glasses 25%	85.0%	88.7%	84.5%	88.7%	3.1 %
Giraffe 25%	36.1%	47.1%	19.0%	18.7%	9.4 %
Average	**48.0%**	**47.5%**	**49.3%**	**46.3%**	**11.8%**





Results (Precision Grasps) (II)



Execution on ARMAR-IIIb