

Manipulation Planning for the JSK Kitchen Assistant Robot Using OpenRAVE

Rosen Diankov (University of Tokyo)

Kenji Sato (University of Tokyo)

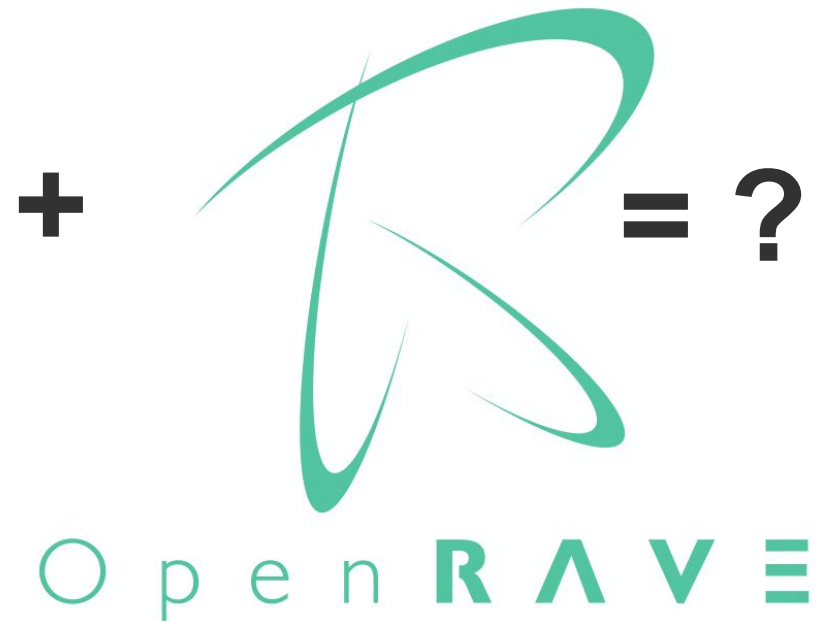
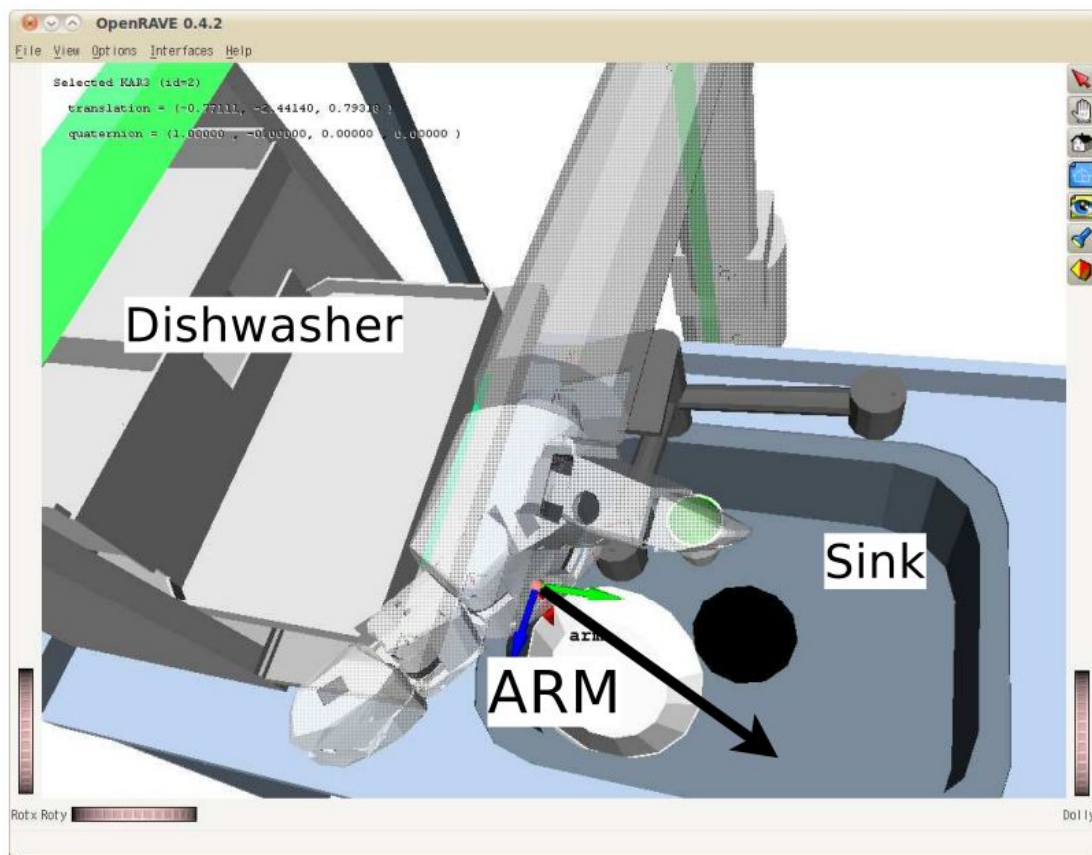
Hiroaki Yaguchi

(University of Tokyo)

Kei Okada (University of Tokyo)

Masayuki Inaba (University of Tokyo)

Dish Loading Challenge



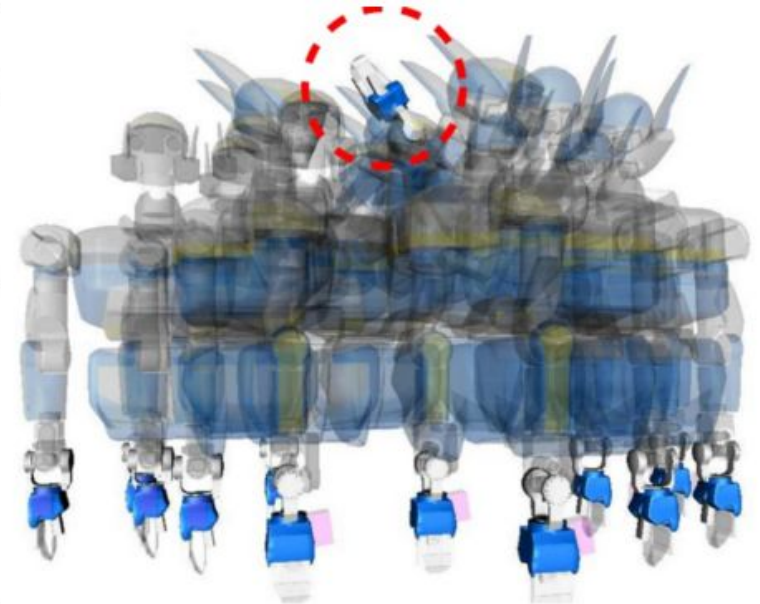
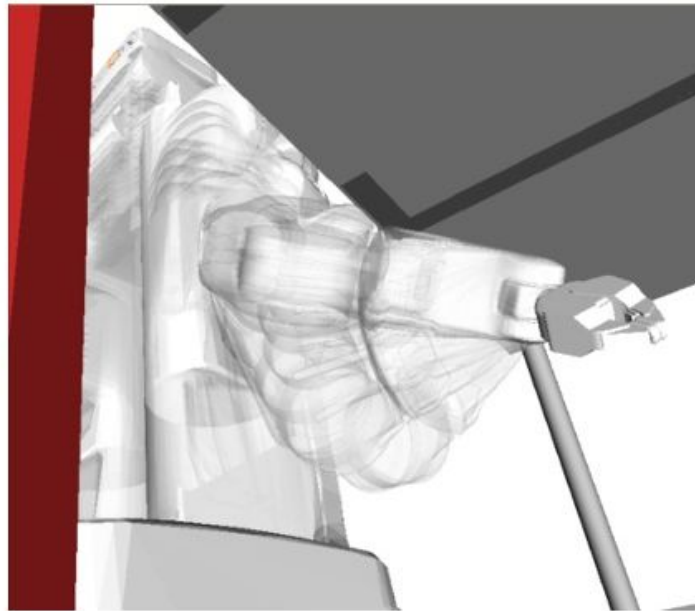
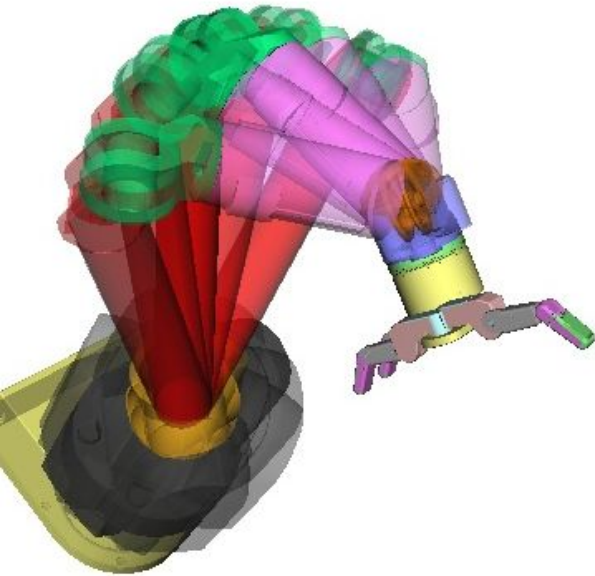
9-hours Programming Time

- Stable Grasp Generation
- Manipulation Planning
- Collision Avoidance
- Analytical IK
- Physics

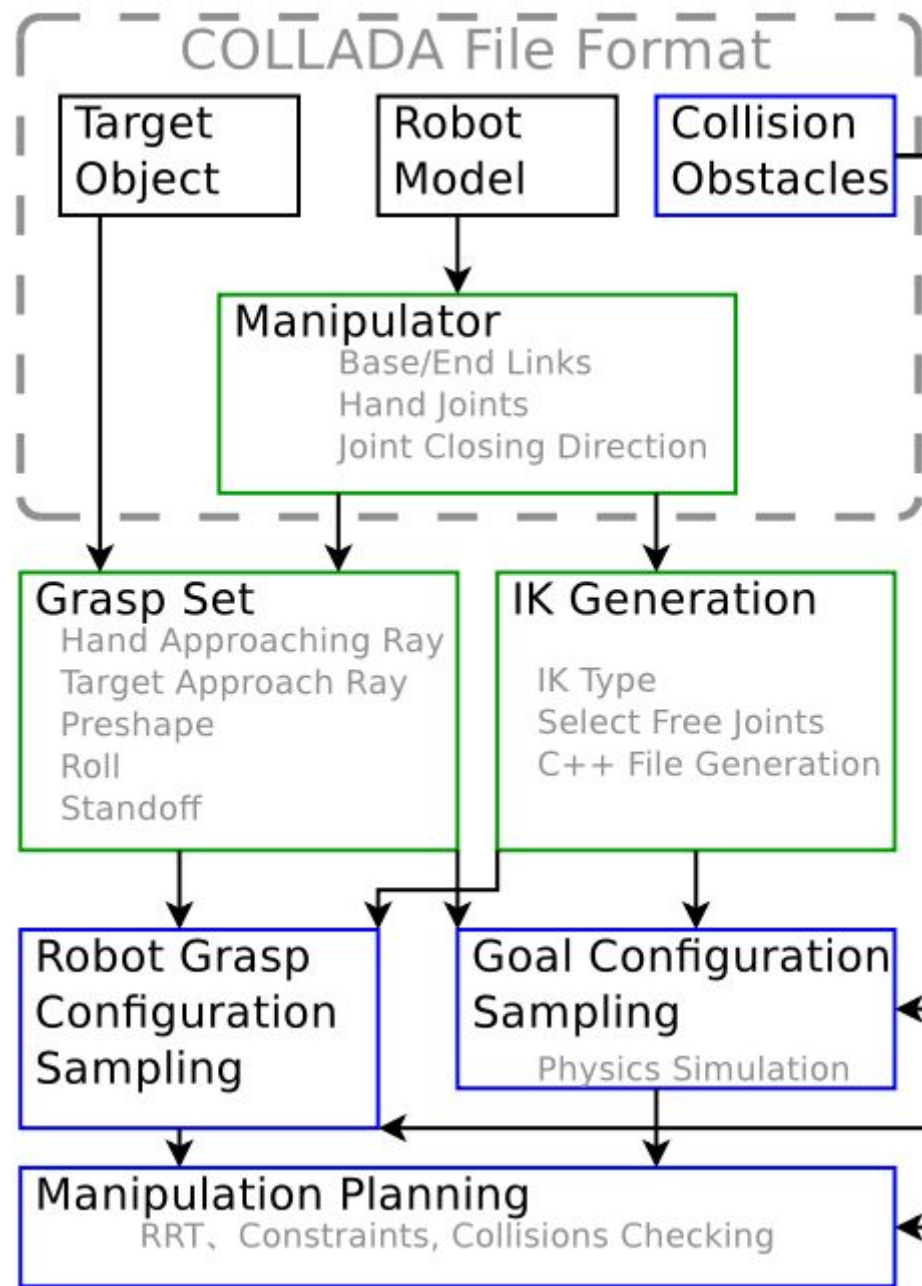


What is OpenRAVE?

- Open Robotics Automation Virtual Environment
 - Manipulation Planning
 - Sensor Visibility Planning
 - Inverse Kinematics: 10 types
 - Workspace Analysis



Pick-and-Place Architecture



IK Generation

```
openrave.py --database inversekinematics --robot=myrobot.dae  
--manipname=arm --iktests=100 --iktype=transform6d  
--freejoint=base_joint
```



Grasp Set Generation

```
openrave.py --database grasping --robot=myrobot.dae  
--manipname=arm --target=myobject.dae --friction=0.1 --preshape="-  
0.685 -1.48 0 0 0" --manipulatordirection="0.09 0.8 0.58"  
--graspingnoise=0.01
```



Grasp Configuration Sampler

python

```
openravepy.databases.grasping.GraspingModel.computeValidGrasps
```



Sampling Goal Configurations



Grasp Planning (6D IK)

- Sample grasps both valid at initial and goal states.

`openrave.py --example graspplanning`

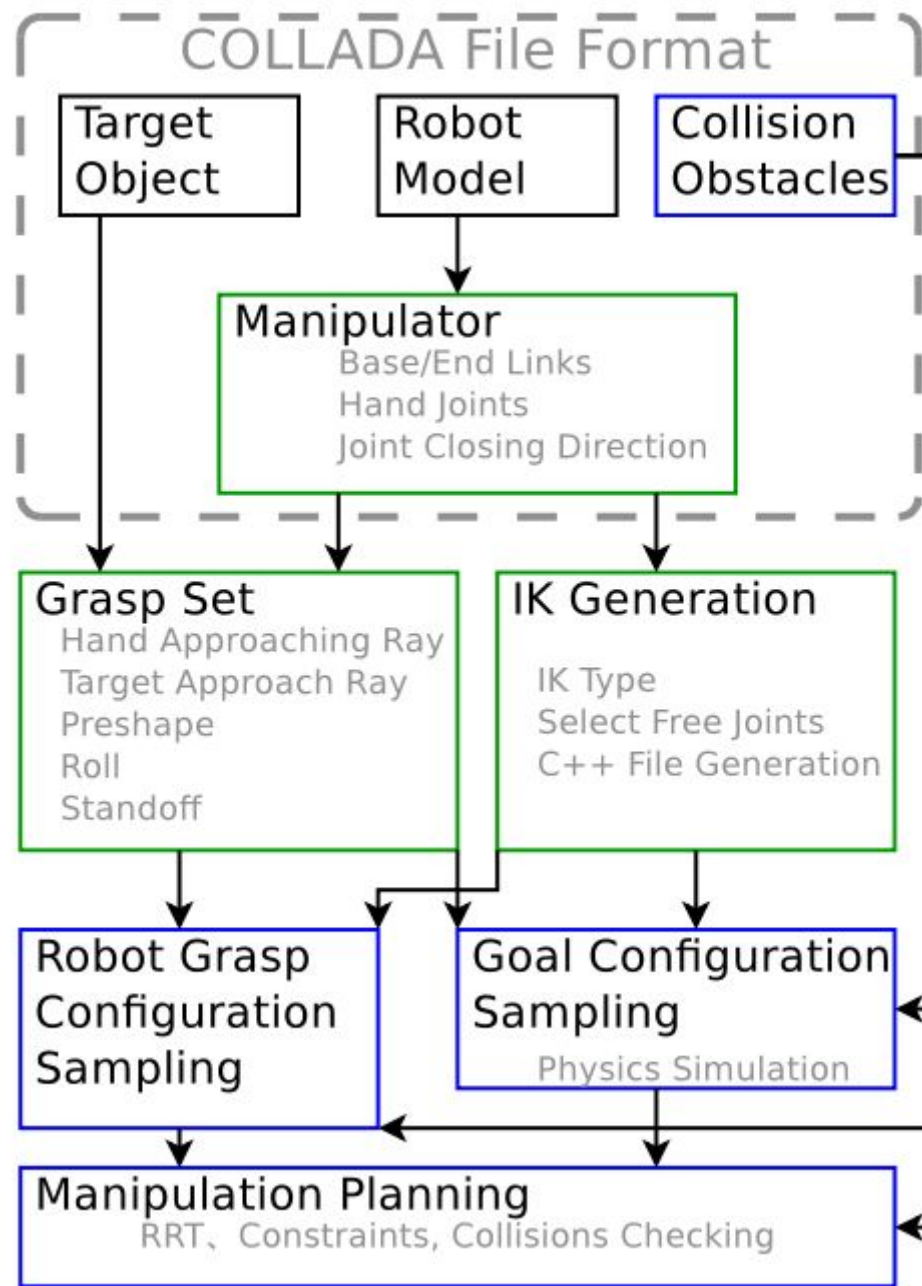


Grasp Planning (5D IK)

`openrave.py --example graspplanning --scene=data/katanatable.env.xml`



Pick-and-Place Architecture

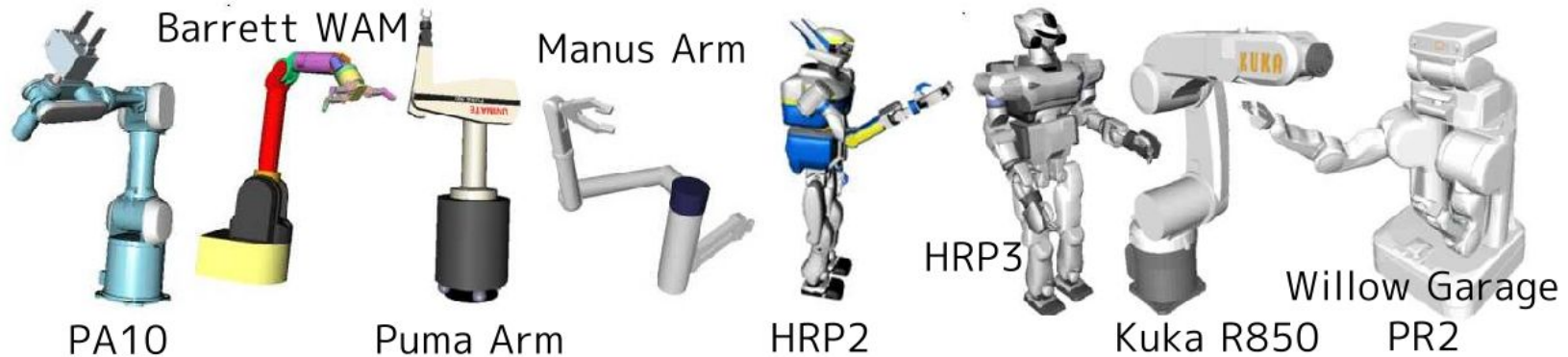


Planning with Multiple Robots



Inverse Kinematics

- **ikfast** - Generate Optimized/Stable C++ Code
- **Any robot**
- Handle **all** divide by zero conditions
- Handle degenerate cases
 - Axes align/DOFs are lost
- Return all solutions in **~6 microseconds**
- **10 types:** 6D, 3D translation, 3D rotation, 4D Ray, 2D direction, 5D, ...



Please use
OpenRAVE ;0)