RBDemo
graspingModule Manual

Task Description

Grasping Module:

Input: Bottle containing the enum of the part (hand) to be closed. It's the code of the touched hand of the robot. E.g, if it corresponds to the left (1) or right (4) hand, the module closes it.

The enumeration of the parts can be found in:
http://wiki.icub.org/iCub_documentation/common_8h_source.html

In particular:

```
// List of the parts composing the skin of iCub
enum SkinPart {
SKIN_PART_UNKNOWN=0,
SKIN_LEFT_HAND, SKIN_LEFT_FOREARM, SKIN_LEFT_UPPER_ARM,
SKIN_RIGHT_HAND, SKIN_RIGHT_FOREARM, SKIN_RIGHT_UPPER_ARM,
SKIN_FRONT_TORSO,
SKIN_PART_ALL, SKIN_PART_SIZE
};
```

Output: Implicit

Instructions to run the program on iCubSim

1- Launch the Cartesian Interface

In order to use the Cartesian Interface, make sure that the following steps are done. [Note that the term cluster refers to the set of computers directly connected to the robot network, whereas PC104 indicates the hub board mounted on the robot]

1.1 Update YARP and iCub repositories.

1.2 Compile YARP (always a good practice).

1.3 Install Ipopt on the cluster.

1.4 On the cluster: compile the repository with the switch ***ENABLE_icubmod_cartesiancontrollerclient enabled***. This will make the client part of the interface available.

1.5 On PC104: compile the repository with the switch ENABLE_icubmod_cartesiancontrollerserver enabled. This will make the server part of the interface available on the hub.

Henceforth rely on the installed copy of
$ICUB_ROOT/main/app/iCubStartup/scripts/iCubStartup.xml.template
application to launch both the cartesian solvers (and other useful tools as well).

Make sure that the machines where the solvers will be running are configured properly to work with the specific robot (e.g. the YARP_ROBOT_NAME correctly points to the name of the robot) in order to load at start-up the robot dependent kinematics.

NOTE: Once everything is launched from within the application, if the user accidentally stops one of these modules, it turns to be mandatory to restart all of them from the beginning; this comes from the requirement to keep the communication protocol as light as possible, avoiding requests for integrity check and reinitialization.

Then:

a. Launch the iCub Simulator:
iCub_SIM

b. Launch the simCartesianControl module.

Parameters:

--robot name // specifies the simulated robot name to connect to.
--no_legs // disable the control of the robot legs.

Other options are available but their default values should be fine for normal use. If you are really curious then get into the short code :)

e.g. in the most basic case on the simulator:
simCartesianControl --robot icubSim --no_legs

c. Launch the Cartesian Solvers for the left and right arms: have a look to the template located in the directory $ICUB_ROOT/main/app/simCartesianControl/scripts .

iKinCartesianSolver --context simCartesianControl --part left_arm
iKinCartesianSolver --context simCartesianControl --part right_arm

2- Launch "graspingModule.exe"
graspingModule.exe

2.1- Launch robotMotorGUI to manipulate joint positions (optional)
robotMotorGui --name icubSim

2.2- Send commands to the module by hand (optional)
yarp write /write /graspingModule/in