

Controller - rojas70@gmail.com - Gmail

Hi Juan,

I finished implementing your code in Baxter.

In the attach compressed file, there are all the files needed. It is a ros package that should be placed on your baxter working space directory (something like /home/ros_ws) and then run the command: `$ catkin_make`

Run the controller of the left arm with the command: `$ roslaunch force_controller left_controller.launch`

and for the right arm: `$ roslaunch force_controller right_controller.launch`

The controller is using baxter's URDF to construct the kinematic chain by KDL that will compute the Jacobian.

Based on the data that I had I tried to characterize the "noise" or "offset" in the torque readings of each joint depending on the joints angles. For some joints it is very noisy so I think it will be difficult to get good precision (the error could be around 2 or 3 Newtons).

I think it may be better to add a force sensor to Baxter's wrists.

The controller reads the torques from the joints and the gravitational torques, then compute the offset torque based on the joint angles and subtracts it from the actual torque (the torque measured minus the gravitational torques), then this torque is converted to a force/torque acting at the endpoint through the Jacobian, computes the error with the desired force or torque given through a service call, the response from the service call is the desired update joint angles.

The ros service call is of the form: `$rosservice call /left/force_controller "{num_ctrls: 1, type: [force], desired: [{x: 0.0, y: 0.0, z: 1.0}], gains: []}"`

`num_ctrls` -> the number of controllers (could be one or two, should be an integer)

`type` -> force or moment (should be a vector of string)

`desired` -> the desired force/moment (should be a vector of `geometry_msgs::Vector3`)

`gains` -> this is optional in case you want to change the default values (the same as in your code), if not use just leave it like in the example, if use it should be a vector of `geometry_msgs::Vector3`, the size of the vector should be the same size of the number of controllers.

As in your code I could not find a loop for the position control and the force error I did not make any loop, but if you want the loop for controlling the joint's angles while checking for the force/moment error, let me know. I have in a different package a simple position controller that could be easily added to the package that I am sending now.

I hope this can help you.

I am not in AIST anymore, so I can not run experiments, but if you need any help with respect to Baxter software/hardware, let me know I will be happy to help you.

Best Regards.

Ixchel

Ixchel G. RAMIREZ ALPIZAR

Postdoctoral Researcher

Manipulation Research Group,

Intelligent Systems Research Institute,

National Institute of Advanced Industrial Science and Technology (AIST)

E-mail: ixchel.ramiresuarupi-saru@aist.go.jp^[1]

Links

1. <mailto:ixchel.ramiresuarupi-saru@aist.go.jp>