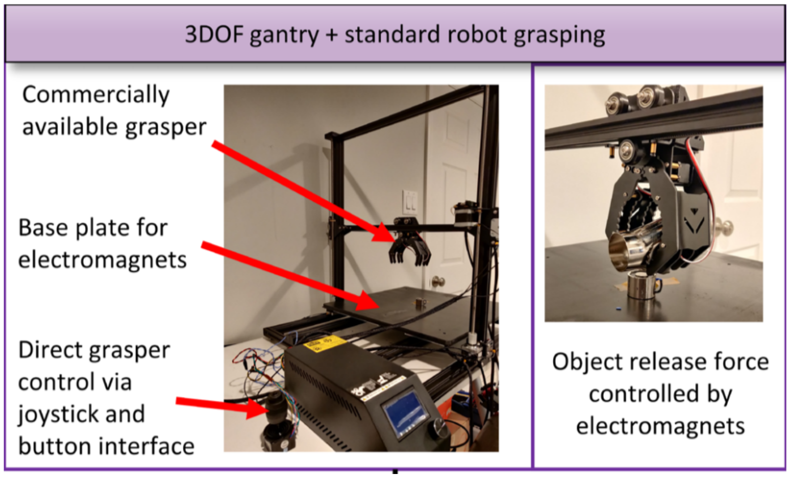
Gantry Simulation

1. Overview

This is a simulation environment (currently using PyBullet) to simulate the gantry interacting with objects for grasping applications.

The picture of the gantry from the FRR grant is shown to the right:



3DOF gantry + standard robot grasping

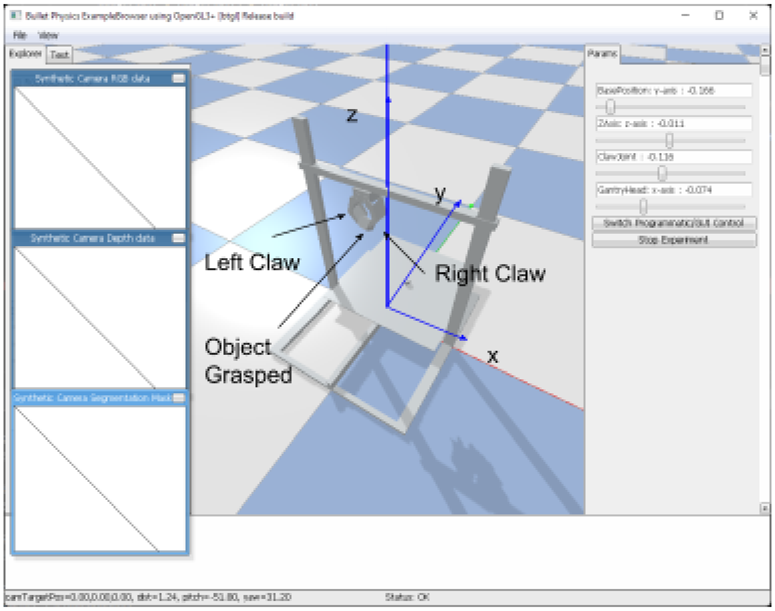
Commercially available grasper

Base plate for electromagnets

Direct grasper control via joystick and button interface

Object release force controlled by electromagnets

The PyBullet simulation is shown below.


Bullet Physics ExampleBrowser using OpenGL3+ [btgl] Release build
File
View
Explorer
Test
Synthetic Camera RGB data
Synthetic Camera Depth data
Synthetic Camera Segmentation Mask
Params
z
y
x
Left Claw
Right Claw
Object Grasped
Switch Programmatic/GUI control
Stop Experiment

BasePosition: y-axis : -0.168

ZAxis: z-axis : -0.011

ClawJoint : -0.136

GantryHead: x-axis : -0.074

camTargetPos=0.00,0.00,0.00, dist=1.24, pitch=-51.80, yaw=31.20 Status: OK

2. Packages Needed

I am currently running conda 4.12.0 and python 3.10.4

Please install the following packages:

- Pybullet
- Numpy
- Matplotlib
- math,pathlib,time ,warnings (should be installed by default)

The files should be under "Gantry_Simulation" branch in the FRR GitHub Repository

3. Python Implementation

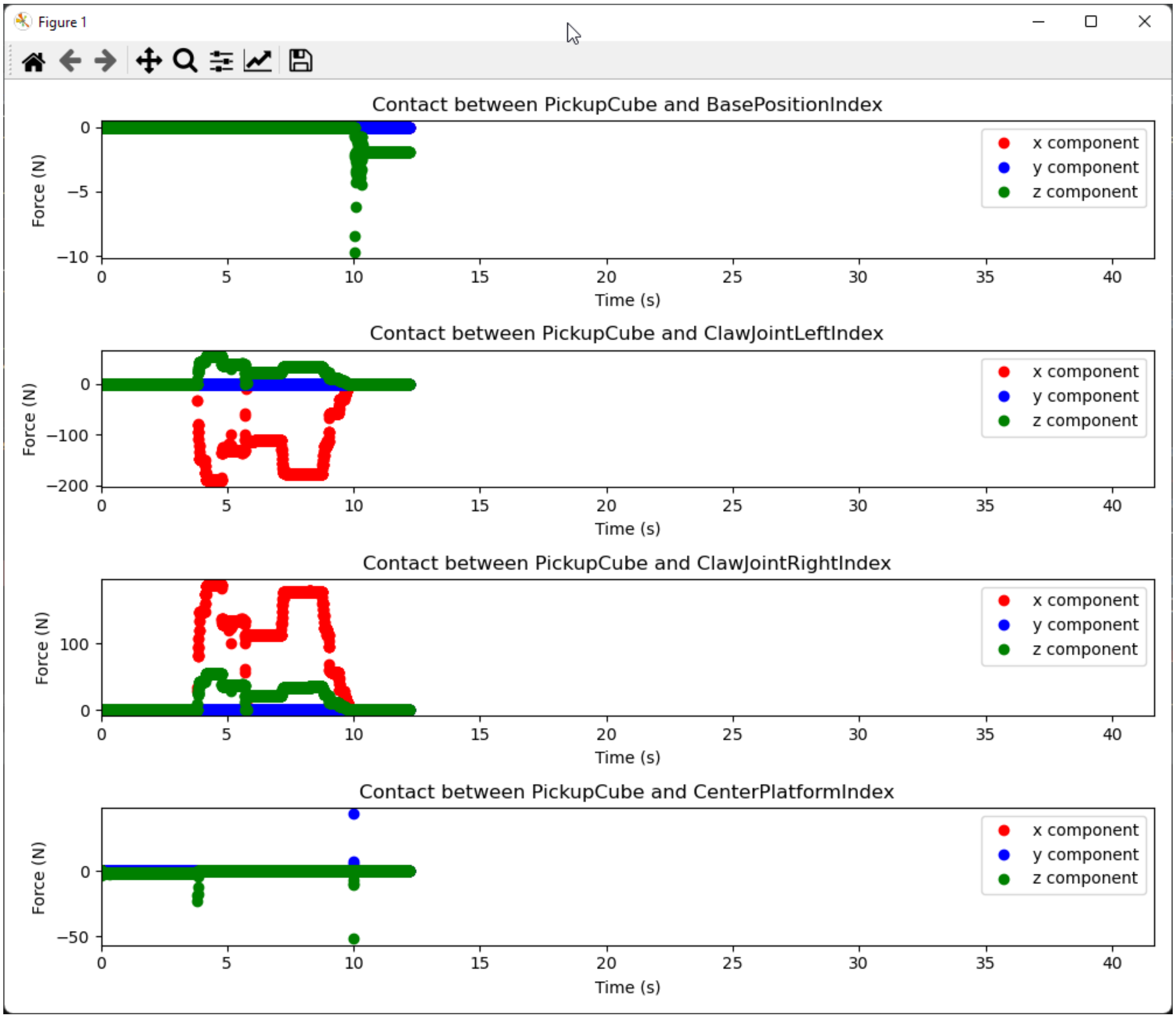
The class implementation is implemented in "**GantrySimulation.py**". The class has methods for instantiating the PyBullet simulation, loading the Gantry and objects, stepping the simulation (default 1/240 seconds per step), and getting and plotting contact forces.

To see how to instantiate the GantrySimulation class and use it for simulation, see the file "**SinusoidGUI_ProgrammaticCheck.py**". Note In this simulation, you can switch between GUI control and programmatic control. In GUI control, you can use the **sliders** on the right side to move the different axes of the gantry, as well as open and close the grasper.

By clicking the "**Switch Programmatic/GUI Control**" button, you can switch back and forth between GUI control (i.e. using the sliders) and by sending commands from "**SinusoidGUI_ProgrammaticCheck.p**y" to the **GantrySimulation** class using the "**stepSim**" method (i.e. programmatic control). In the programmatic control, all joints will sinusoidally move between their joint limits with a nominal period of 4 seconds (this may be slightly off).

The button "**Stop Experiment**" will stop the simulation and disconnect from the PyBullet server. It will then plot contact forces between the objects on the gantry and the claws and baseplate of the gantry. The forces are aligned with the global x,y,z directions. An example of the plot is shown below (see class method plotContactForces() for more details). You can access the link and object positions and orientation in the global world frame using the getPositionsAndOrientations method. See the code for more details.

Note: while you have the ability to scale the simulation in both length (lengthScale) and mass (massScale) by passing those arguments to the GantrySimulation initialization.



You can use the “**SinusoidGUI_ProgrammaticCheck.py**” as a reference to implement your own programmatic control via neural networks etc.

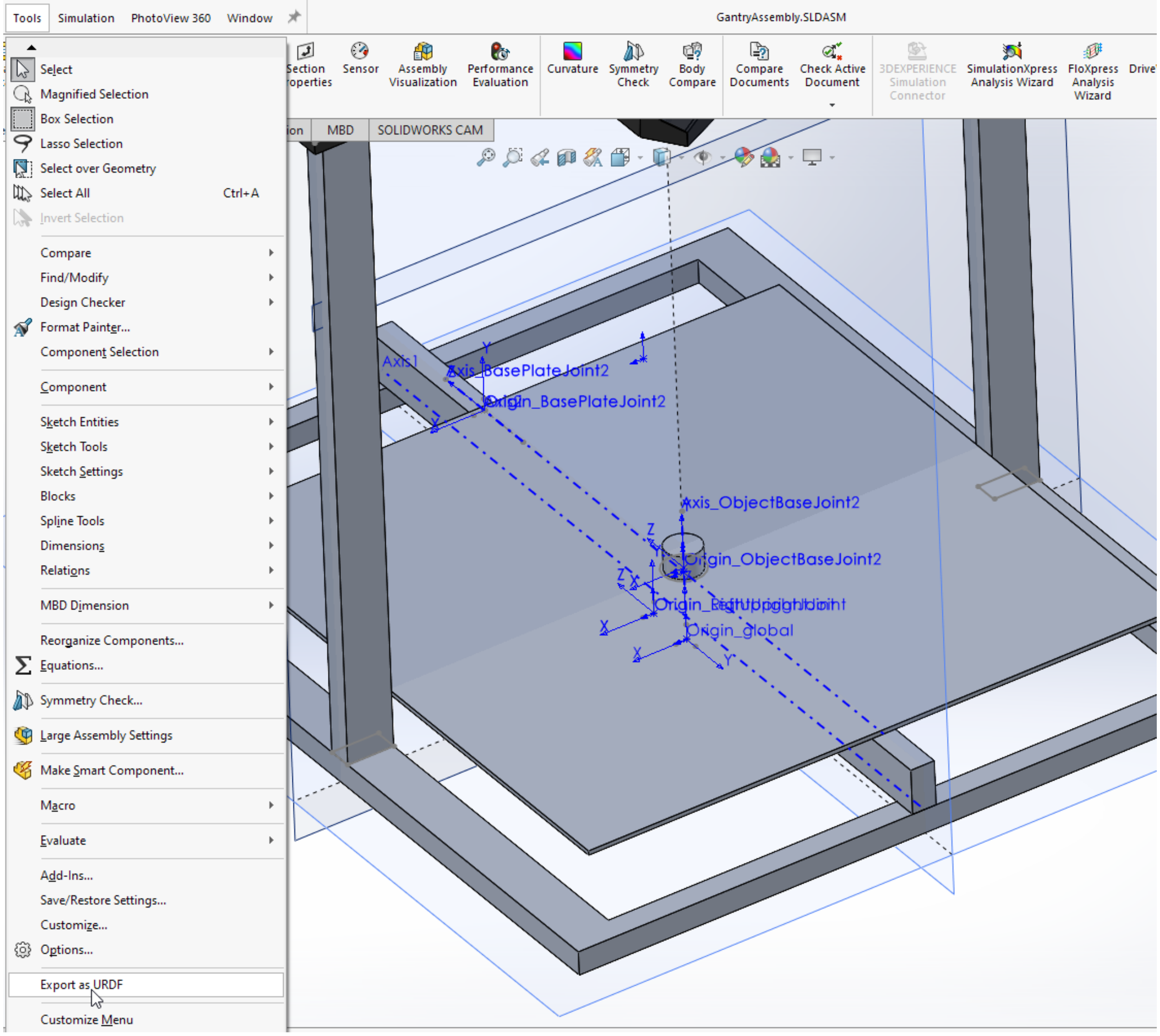
4. Loading Objects

Currently, loading objects is only supported via URDF files. The gantry is automatically loaded when initializing the GantrySimulation class. To add other objects, see the “addObjectsToSim” method call in the “SinusoidGUI_ProgrammaticCheck.py”.

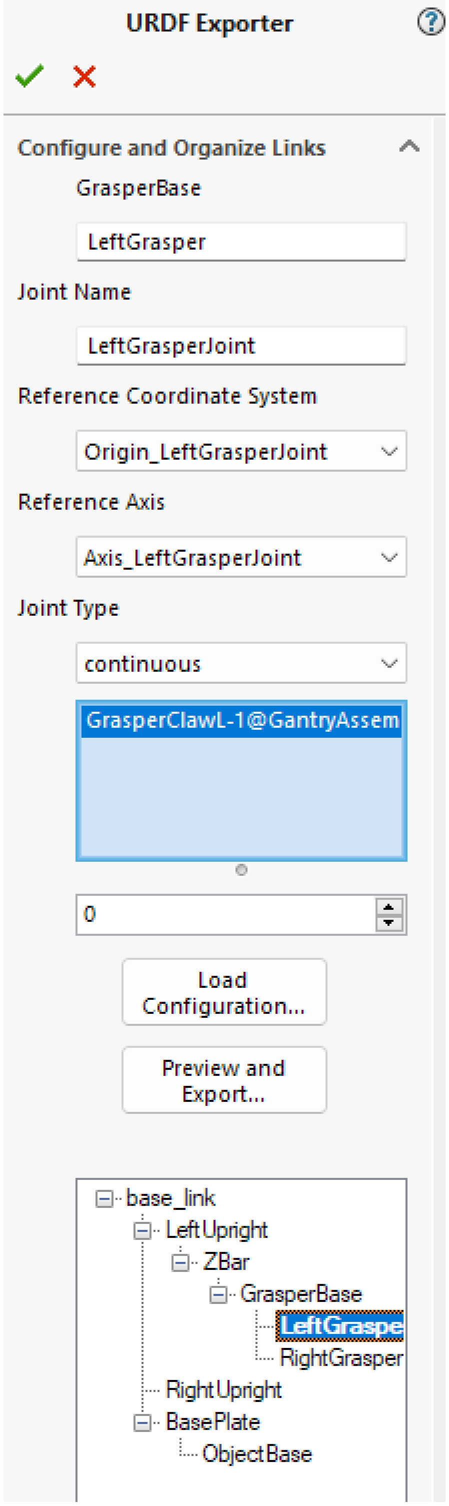
You can easily create URDF files from a Solidworks assembly.

1. First install the Solidworks URDF exporter.

2. Then under Tools-> Export as URDF



3. Configure the different joints if needed. Then select “Preview and Export”



Configure and Organize Links

GrasperBase

LeftGrasper

Joint Name

LeftGrasperJoint

Reference Coordinate System

Origin_LeftGrasperJoint

Reference Axis

Axis_LeftGrasperJoint

Joint Type

continuous

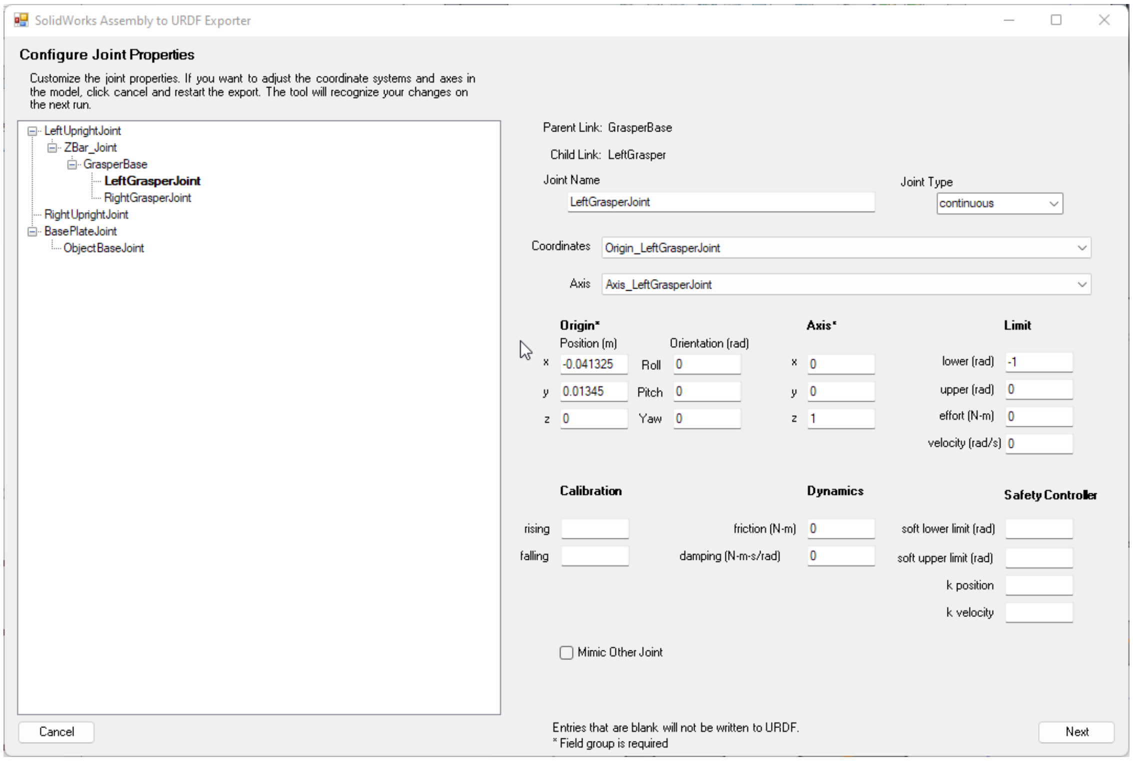
GrasperClawL-1@GantryAssem

0

```
base_link
  LeftUpright
    ZBar
      GrasperBase
        LeftGraspe
        RightGrasper
  RightUpright
  BasePlate
    ObjectBase
```

If you are exporting an object to be grasped, then it most likely doesn’t have any joints (constraints) and you can directly select “Preview and Export”

4. You can then modify options for the joint properties. Click Next when done.



SolidWorks Assembly to URDF Exporter

Configure Joint Properties

Customize the joint properties. If you want to adjust the coordinate systems and axes in the model, click cancel and restart the export. The tool will recognize your changes on the next run.

- LeftUprightJoint
  - ZBar_Joint
    - GrasperBase
      - **LeftGrasperJoint**
      - RightGrasperJoint
- RightUprightJoint
- BasePlateJoint
  - ObjectBaseJoint

Parent Link: GrasperBase

Child Link: LeftGrasper

Joint Name: LeftGrasperJoint

Joint Type: continuous

Coordinates: Origin_LeftGrasperJoint

Axis: Axis_LeftGrasperJoint

Origin*

Position (m)		Orientation (rad)	
x	-0.041325	Roll	0
y	0.01345	Pitch	0
z	0	Yaw	0

Axis*

x	0
y	0
z	1

Limit

lower (rad)	-1
upper (rad)	0
effort (N-m)	0
velocity (rad/s)	0

Calibration

rising

falling

Dynamics

friction (N-m)	0
damping (N-m-s/rad)	0

Safety Controller

soft lower limit (rad)

soft upper limit (rad)

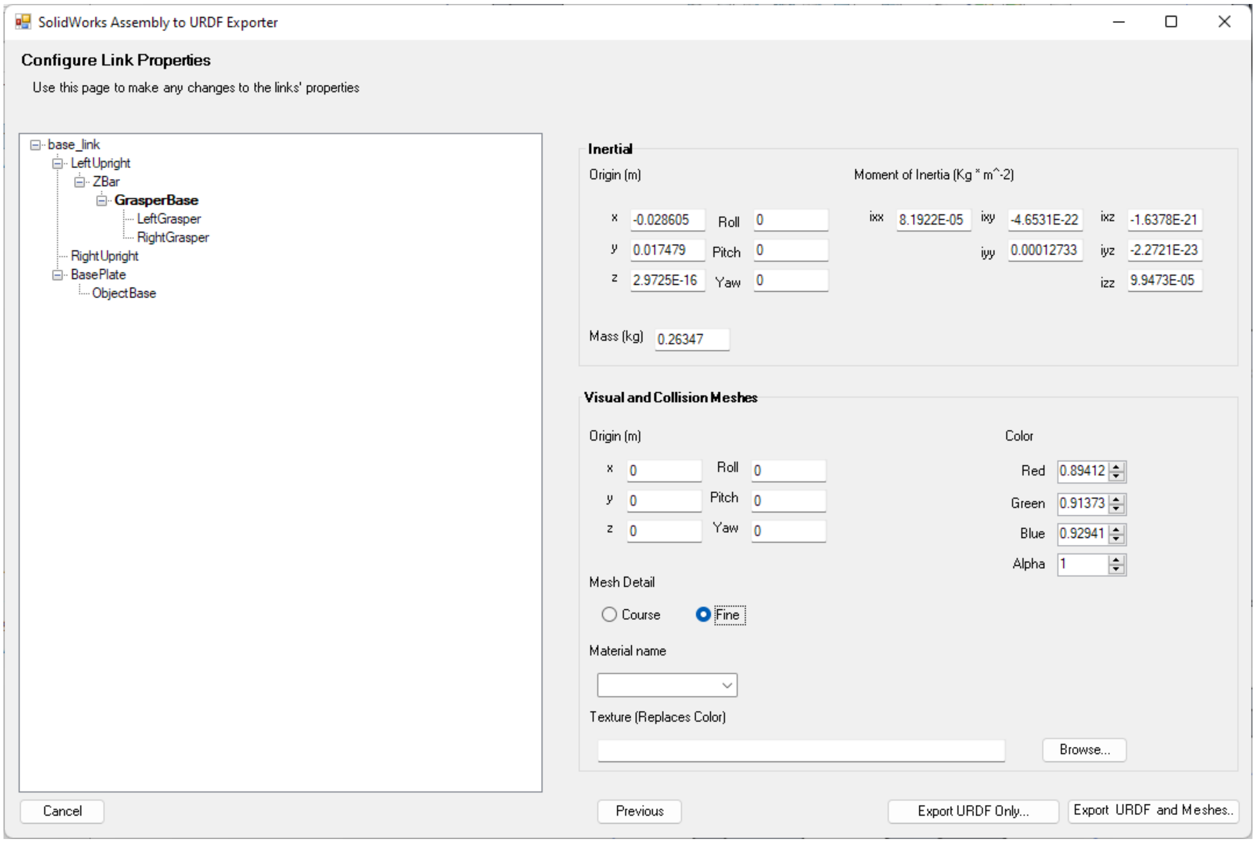
k position

k velocity

☐ Mimic Other Joint

Entries that are blank will not be written to URDF.
* Field group is required

5. You can then configure the link properties (check masses to make sure they are non zero). Note that it is recommended to select "Fine" for Mesh Detail.



SolidWorks Assembly to URDF Exporter

Configure Link Properties

Use this page to make any changes to the links' properties

- base_link
  - LeftUpright
    - ZBar
      - **GrasperBase**
        - LeftGrasper
        - RightGrasper
  - RightUpright
  - BasePlate
    - ObjectBase

Inertial

Origin (m)

x	-0.028605	Roll	0
y	0.017479	Pitch	0
z	2.9725E-16	Yaw	0

Moment of Inertia (Kg * m^-2)

ixx	8.1922E-05	ixy	-4.6531E-22	ixz	-1.6378E-21
		iyy	0.00012733	iyz	-2.2721E-23
				izz	9.9473E-05

Mass (kg) 0.26347

Visual and Collision Meshes

Origin (m)

x	0	Roll	0
y	0	Pitch	0
z	0	Yaw	0

Color

Red	0.89412
Green	0.91373
Blue	0.92941
Alpha	1

Mesh Detail

○ Course ● Fine

Material name

Texture (Replaces Color)

6. When finished select “Export URDF and Meshes”

Note that for the Gantry simulation, grasping is made more stable because of the addition of a “friction anchor”. Please see the .urdf file for more details.