Latest Features in Robotics System Toolbox

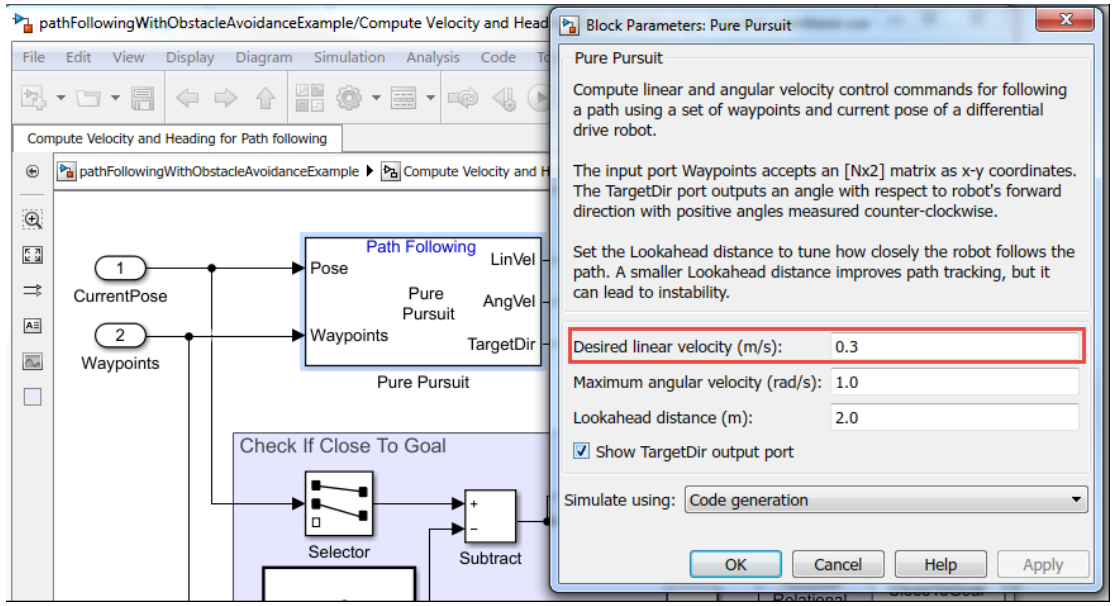
March 2017

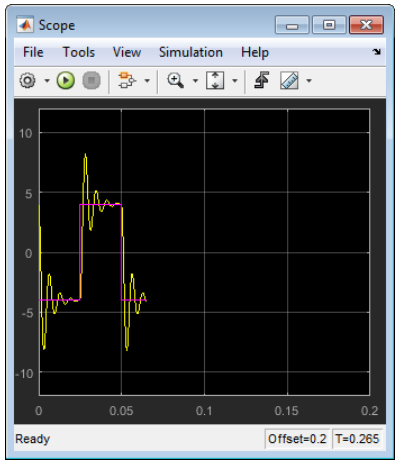
R2017a

Simulink External Mode for ROS

Tune parameters and view signal values of deployed ROS nodes over TCP/IP with Simulink Coder

- Connect to deployed ROS node through external mode
- Tune parameters in ROS node running on ROS device
- Get signal values and log data
- Set priority of external mode thread to adjust execution performance

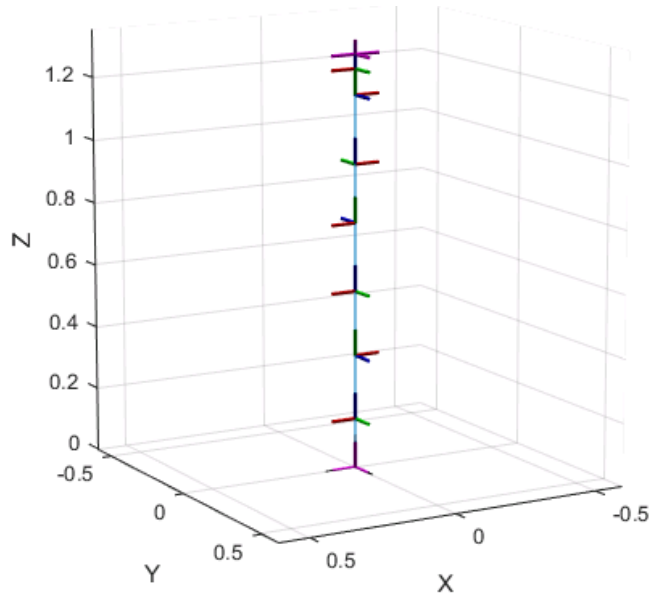


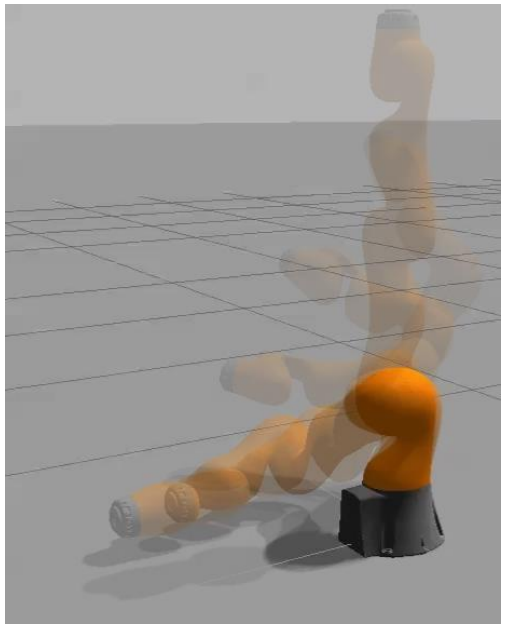


Dynamics for Robot Manipulators

Solve inverse and forward dynamics for `RigidBodyTree` objects

- Specify rigid body inertial properties
- Compute for the rigid body tree
  - Forward dynamics
  - Inverse dynamics
  - Mass matrix
  - Velocity product
  - Gravity torque
  - Center of mass position and Jacobian



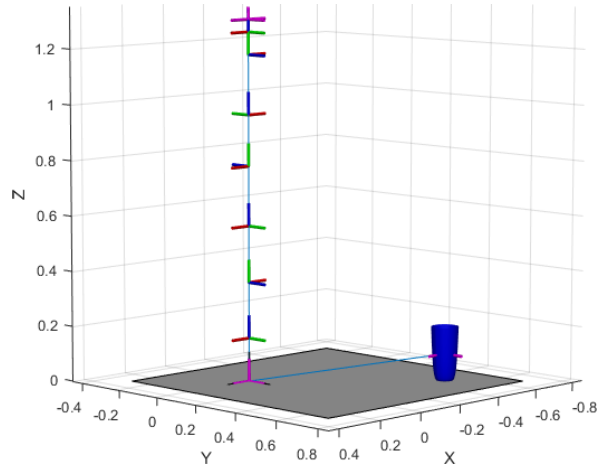


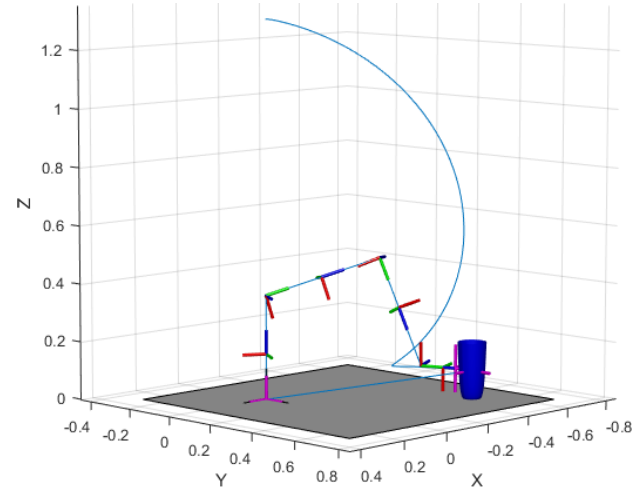
```
» load exampleRobots.mat
» lbr.DataFormat = 'column';
» q = randomConfiguration(lbr);
» tau = inverseDynamics(lbr, q);
```

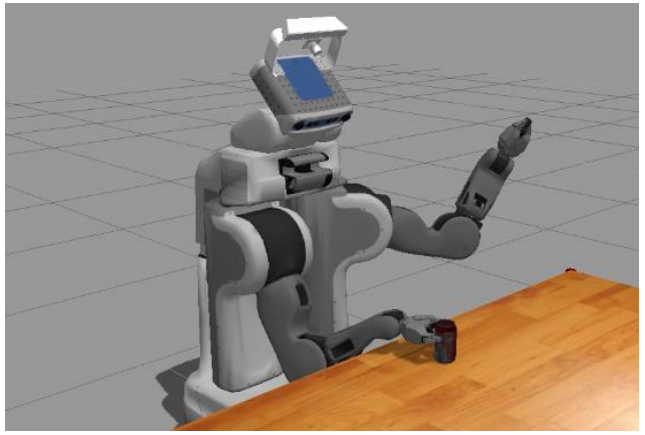
Generalized Inverse Kinematics

Solve multiconstrained inverse kinematics for robot manipulators

- Designed for generic tree or chain-structured manipulators
- Supports multiple constraint types
  - Body orientation, position, pose
  - Aiming (“aim the camera towards that point”)
  - Cartesian bounds (“keep the hand inside this box”)
  - Joint positions
- Combine constraints to produce more complex queries







URDF File Importer

Import Universal Robotic Description Format (URDF) Files

- Import arbitrary robot from URDF (as URDF file or URDF string) as `RigidBodyTree` object
- Extract kinematic and dynamic information, but not for visual and collision elements
- URDF parsing engine shared with **Simscape Multibody** for consistent user experience

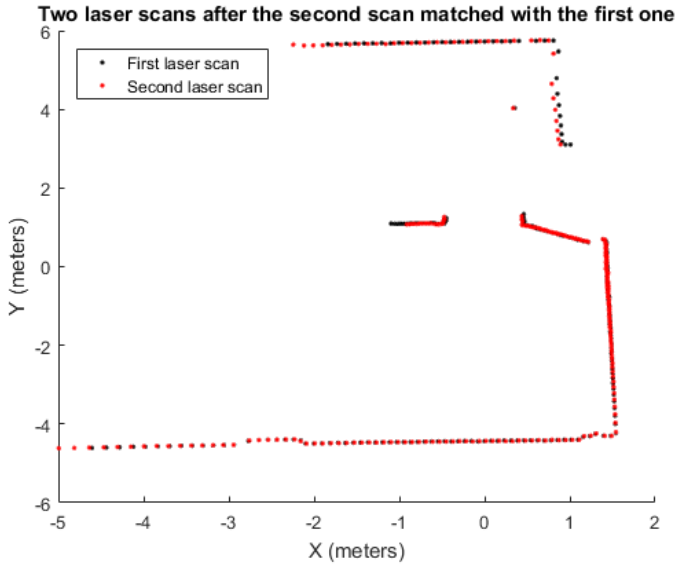
```
</link>
<joint name="iiwa_joint_1" type="revolute">
  <parent link="iiwa_link_0"/>
  <child link="iiwa_link_1"/>
  <origin rpy="0 0 0" xyz="0 0 0.15"/>
  <axis xyz="0 0 1"/>
  <limit effort="300" lower="-2.96705972839" upper="2.96705972839" velocity="10"/>
  <safety_controller k_position="100" k_velocity="2" soft_lower_limit="-2.93215314335"
  <dynamics damping="0.5"/>
</joint>
<link name="iiwa_link_1">
  <inertial>
    <origin rpy="0 0 0" xyz="0 -0.03 0.12"/>
    <mass value="4"/>
    <inertia ixx="0.1" ixy="0" ixz="0" iyy="0.09" iyz="0" izz="0.02"/>
  </inertial>
  <visual>
    <origin rpy="0 0 0" xyz="0 0 0.0075"/>
    <geometry>
      <mesh filename="package://iiwa_description/meshes/iiwa7/visual/link_1.stl"/>
    </geometry>
    <material name="Orange"/>
  </visual>
  <collision>
    <origin rpy="0 0 0" xyz="0 0 0.0075"/>
    <geometry>
      <mesh filename="package://iiwa_description/meshes/iiwa7/collision/link_1.stl"/>
    </geometry>
    <material name="Orange"/>
  </collision>
</link>
<joint name="iiwa_joint_2" type="revolute">
  <parent link="iiwa_link_1"/>
  <child link="iiwa_link_2"/>
  <origin rpy="1.57079632679   0 3.14159265359" xyz="0 0 0.19"/>
```

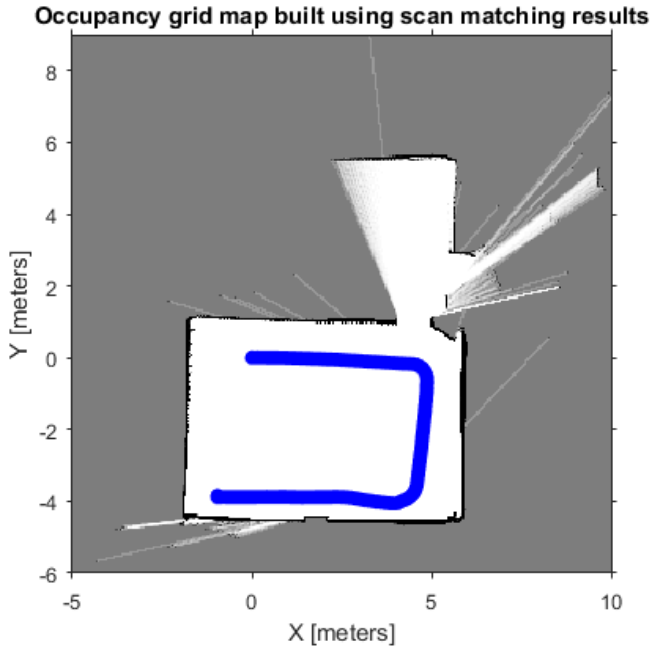
```
» lbr = importrobot('iiwa7.urdf')
```

Scan Matching

Calculate pose difference between laser scans

- Estimate relative transformation (robot pose) between two LIDAR scans
- Recover robot motion if no odometry or other position information is available
- Use as algorithmic building block in mapping applications





`rosparam` Simplified Commands

Modify ROS parameters using a simplified interface

- List, get, set, and delete ROS parameters in a more convenient way with single-line MATLAB commands.
- Work with ROS namespaces as dictionaries (map to MATLAB structs).

```
% Get the list of parameters using command line
rosparam list

% Get the list of parameters under a certain namespace
rosparam list /gazebo

% Set a string parameter
rosparam set /string_param param_value

% Get the value of a string parameter
rosparam get /string_param

% Delete a parameter
rosparam delete /string_param

% Set a double parameter
rosparam set /double_param 1.2

% Create a parameter tree object for more advanced operatio
ptree = rosparam;

% Set an integer parameter
```