ROS URDF

Nicolas Limpert

MASKOR

August 10, 2017

Outline

- Motivation
- Setting up a 3DOF robot
- XACRO

Motivation

Task

Provide a human-readable platform independent style to define robots, sensors and / or scenes including:

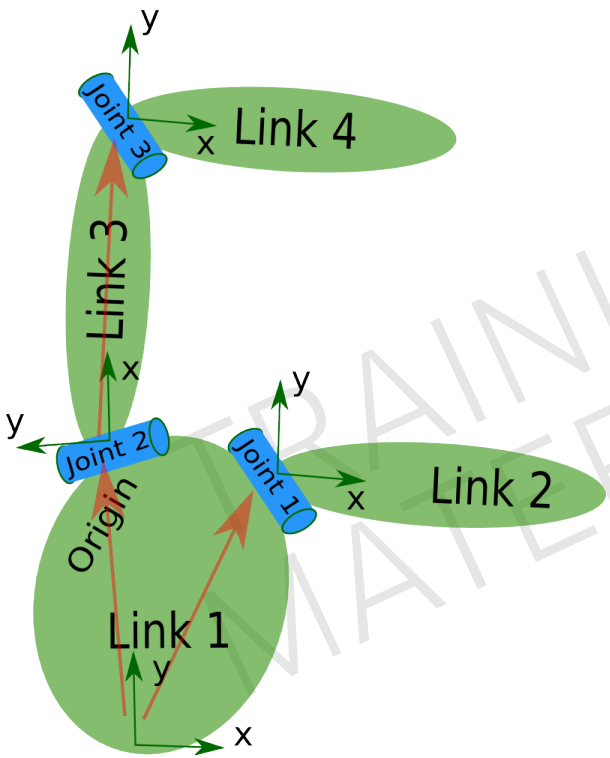
- collision properties
- visible properties

} either .stl or geometric primitives (box, cylinder, etc.)

- joints / links
- physical properties
  - inertia
- joint dynamics
  - friction
  - damping

Setting up a 3DOF robot

http://wiki.ros.org/urdf/Tutorials/Create your own urdf file




```
<robot name="test_robot">
  <link name="link1" />
  <link name="link2" />
  <link name="link3" />
  <link name="link4" />

  <joint name="joint1" type="continuous">
    <parent link="link1"/>
    <child link="link2"/>
  </joint>

  <joint name="joint2" type="continuous">
    <parent link="link1"/>
    <child link="link3"/>
  </joint>

  <joint name="joint3" type="continuous">
    <parent link="link3"/>
    <child link="link4"/>
  </joint>
</robot>
```

Adding dimensions

http://wiki.ros.org/urdf/Tutorials/Create your own urdf file

```
<robot name="test_robot">
  <link name="link1" />
  <link name="link2" />
  <link name="link3" />
  <link name="link4" />


  <joint name="joint1" type="continuous">
    <parent link="link1"/>
    <child link="link2"/>
    <origin xyz="5 3 0" rpy="0 0 0" />
  </joint>

  <joint name="joint2" type="continuous">
    <parent link="link1"/>
    <child link="link3"/>
    <origin xyz="-2 5 0" rpy="0 0 1.57" />
  </joint>

  <joint name="joint3" type="continuous">
    <parent link="link3"/>
    <child link="link4"/>
    <origin xyz="5 0 0" rpy="0 0 -1.57" />
  </joint>
</robot>
```

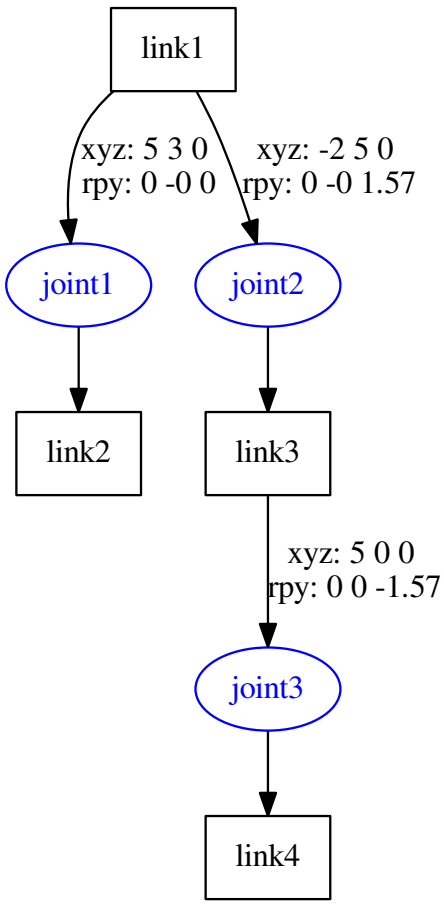
Completing kinematics

http://wiki.ros.org/urdf/Tutorials/Create your own urdf file

```
<robot name="test_robot">
  <link name="link1" />
  <link name="link2" />
  <link name="link3" />
  <link name="link4" />
  <joint name="joint1" type="continuous">
    <parent link="link1"/>
    <child link="link2"/>
    <origin xyz="5 3 0" rpy="0 0 0" />
    <axis xyz="-0.9 0.15 0" />
  </joint>
  <joint name="joint2" type="continuous">
    <parent link="link1"/>
    <child link="link3"/>
    <origin xyz="-2 5 0" rpy="0 0 1.57" />
    <axis xyz="-0.707 0.707 0" />
  </joint>
  <joint name="joint3" type="continuous">
    <parent link="link3"/>
    <child link="link4"/>
    <origin xyz="5 0 0" rpy="0 0 -1.57" />
    <axis xyz="0.707 -0.707 0" />
  </joint>
</robot>
```

The Created Kinematic Chain

http://wiki.ros.org/urdf/Tutorials/Create your own urdf file



By making use of `graphviz`, one could display the created kinematic chain:

```
$>urdf_to_graphiz my_robot.urdf
$>evince test_robot.pdf
```

XACRO

http://wiki.ros.org/urdf/Tutorials/Using Xacro to Clean Up a URDF File

Until now:

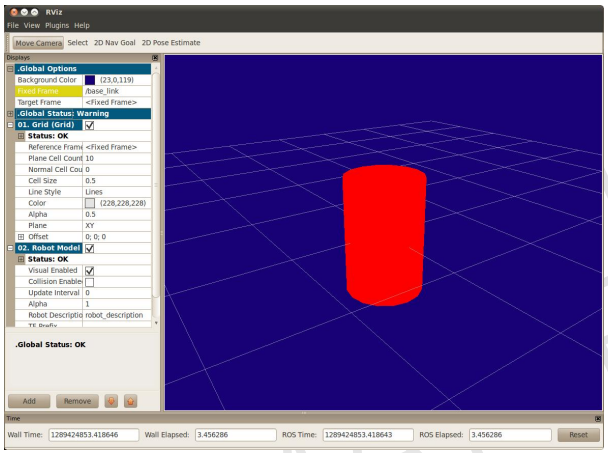
- Manual calculation of joints dimensions
- Manual copying of URDF if a second robot is to be used
- No constant definition available

XACRO

- Cleaning up URDF files
- Constants
- Simple math
- Macros (repeatibility)

XACRO example

http://wiki.ros.org/urdf/Tutorials/Using Xacro to Clean Up a URDF File




```
<link name="base_link">
  <visual>
    <geometry>
      <cylinder length="0.6" radius="0.2"/>
    </geometry>
    <material name="blue"/>
  </visual>
  <collision>
    <geometry>
      <cylinder length="0.6" radius="0.2"/>
    </geometry>
  </collision>
</link>
```

XACRO: adding constants

http://wiki.ros.org/urdf/Tutorials/Using Xacro to Clean Up a URDF File

```
<xacro:property name="width" value="0.2" />
<xacro:property name="bodylen" value="0.6" />
<link name="base_link">
    <visual>
        <geometry>
            <cylinder radius="${width}" length="${bodylen}"/>
        </geometry>
        <material name="blue"/>
    </visual>
    <collision>
        <geometry>
            <cylinder radius="${width}" length="${bodylen}"/>
        </geometry>
    </collision>
</link>
```

```
    <xacro:property name="robotname" value="marvin"/>
    <link name="${robotname}s_leg"/>
```

XACRO: Conditional blocks

http://wiki.ros.org/xacro

```
<xacro:if value="<expression>">
  <... some xml code here ...>
</xacro:if>
<xacro:unless value="<expression>">
  <... some xml code here ...>
</xacro:unless>
```

```
<xacro:property name="var" value="useit"/>
<xacro:if value="${var == 'useit'}"/>
<xacro:if value="${var.startswith('use') and var.endswith('it')}"/>

<xacro:property name="allowed" value="[1,2,3]"/>
<xacro:if value="${1 in allowed}"/>
```

XACRO: Math

http://wiki.ros.org/urdf/Tutorials/Using Xacro to Clean Up a URDF File

```
<xacro:property name="wheeldiam" value="0.5" />
<cylinder radius="${wheeldiam/2}" length="0.1"/>
<origin xyz="${reflect*(width+.02)} 0 0.25" />
```

```
<xacro:property name="pi" value="3.1415926535897931" />
<circle circumference="${2.5 * pi}" />
```

```
<xacro:property name="R" value="2" />
<xacro:property name="alpha" value="${30/180*pi}" />
<circle circumference="${2 * pi * R}" pos="${sin(alpha)} ${cos(alpha)}" />
```

XACRO: Macros

http://wiki.ros.org/urdf/Tutorials/Using Xacro to Clean Up a URDF File

Simple:

```
<xacro:macro name="default_origin">
    <origin xyz="0 0 0" rpy="0 0 0"/>
</xacro:macro>
<xacro:default_origin />
```

Generates:

```
<origin rpy="0 0 0" xyz="0 0 0"/>
```

Parameters:

```
<xacro:macro name="default_inertial" params="mass">
    <inertial>
            <mass value="${mass}" />
            <inertia ixx="1.0" ixy="0.0" ixz="0.0"
                iyy="1.0" iyz="0.0"
                izz="1.0" />
    </inertial>
</xacro:macro>

<xacro:default_inertial mass="10"/>
```

Executing XACRO

```
$> roscd youbot_description/urdf
$> rosrun xacro xacro youbot.urdf.xacro > youbot.urdf

$> roslaunch urdf_tutorial display.launch model:=youbot.urdf
```

Visualizing the generated model