

PROBOT Anno机械臂ikfast运动学求解器配置

笔记本： 六轴步进机械臂

创建时间： 2018/11/7 16:31

更新时间： 2019/4/2 15:21

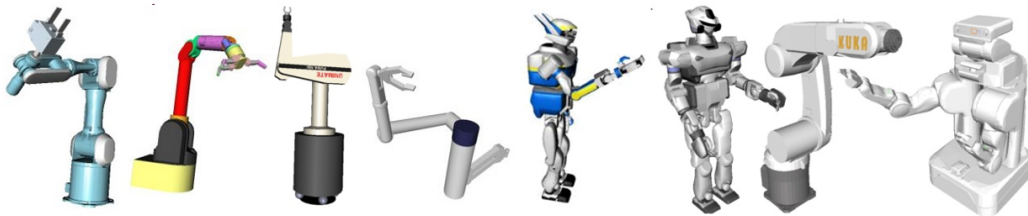
作者： hcx

Ubuntu 16.04 + ROS Kinetic

IKFAST是一种基于解析算法的运动学插件，可以保证每次求解的一致性。



- IKFast, 由Rosen Diankov编写的OpenRAVE运动规划软件提供;
- 可以求解任意复杂运动链的运动学方程（解析解），并产生特定语言的文件（如C++）后供使用;
- 比较稳定、速度快，在最新的处理器上能以5微秒的速度完成运算。



* 参考链接: <http://openrave.org/docs/0.8.2/openravepy/ikfast/>

相比KDL和TRAC-IK, IKFAST的安装过程就比较复杂了, 不过就笔者的使用经验来讲, IKFAST的效果还是很推荐的, 所以不妨一试, 以下就是IKFAST的安装配置过程。

安装程序:

```
sudo apt-get install cmake g++ git ipython minizip python-dev python-h5py python-numpy python-scipy qt4-dev-tools
```

安装依赖库:

```
sudo apt-get install libassimp-dev libavcodec-dev libavformat-dev libavformat-dev libboost-all-dev libboost-date-time-dev libbullet-dev libfaac-dev libglew-dev libgsm1-dev liblapack-dev liblog4cxx-dev libmpfr-dev libode-dev libogg-dev libpcrecpp0v5 libpcre3-dev libqhull-dev libqt4-dev libsoqt-dev-common libsoqt4-dev libswscale-dev libswscale-dev libvorbis-dev libx264-dev libxml2-dev libxvidcore-dev
```

一、源码安装OpenRave

```
git clone --branch latest_stable https://github.com/rdiankov/openrave.git
cd openrave && mkdir build && cd build
cmake -DODE_USE_MULTITHREAD=ON -DOSG_DIR=/usr/local/lib64/ ..
make -j$(nproc)
sudo make install
```

安装OpenSceneGraph-3.4:

```
sudo apt-get install libcairo2-dev libjasper-dev libpoppler-glib-dev libsdl2-dev libtiff5-dev libxrandr-dev
```

```
git clone https://github.com/openscenegraph/OpenSceneGraph.git --branch
OpenSceneGraph-3.4
cd OpenSceneGraph
mkdir build
cd build
cmake .. -DDESIRED_QT_VERSION=4
make -j$(nproc)
sudo make install
```

安装正确版本的sympy
pip install --upgrade --user sympy==0.7.1

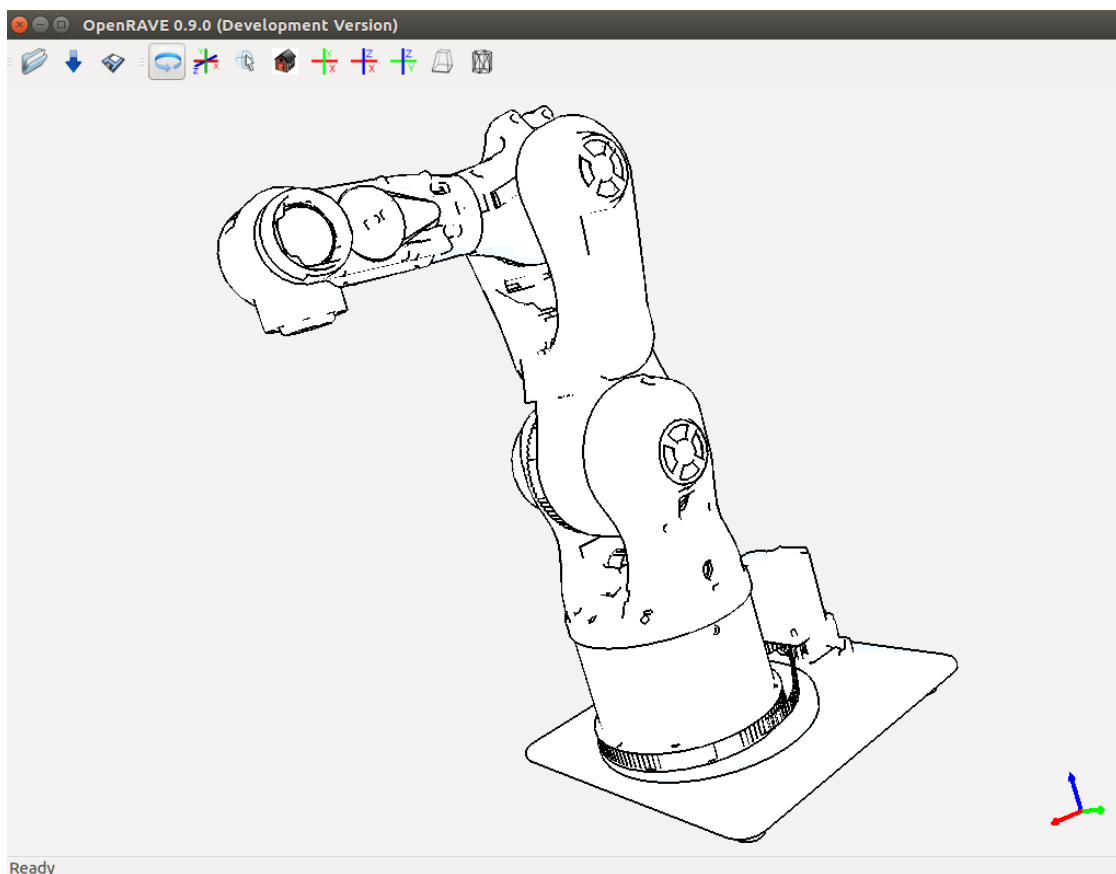
确定没有安装mpmath
sudo apt remove python-mpmath

创建collada文件
export MYROBOT_NAME="probot_anno"
roslaunch xacro xacro --inorder -o "\$MYROBOT_NAME".urdf "\$MYROBOT_NAME".xacro
roslaunch collada_urdf urdf_to_collada "\$MYROBOT_NAME".urdf "\$MYROBOT_NAME".dae

export IKFAST_PRECISION="5"
roslaunch moveit_kinematics round_collada_numbers.py "\$MYROBOT_NAME".dae
"\$MYROBOT_NAME".dae "\$IKFAST_PRECISION"

openrave-robot.py "\$MYROBOT_NAME".dae --info links

```
→ urdf openrave-robot.py "$MYROBOT_NAME".dae --info links
name          index parents
-----
base_footprint 0
base_link      1      base_footprint
link_1         2      base_link
link_2         3      link_1
link_3         4      link_2
link_4         5      link_3
link_5         6      link_4
link_6         7      link_5
tool0          8      link_6
-----
name          index parents
openrave "$MYROBOT_NAME".dae
```



设置运动规划组

```
export PLANNING_GROUP="manipulator"
```

设置运动规划的关节组，以上面的模型关节数据为基础设置

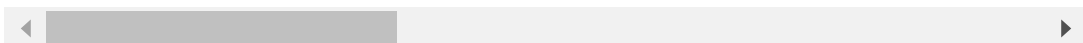
```
export BASE_LINK="1"  
export EEK_LINK="8"
```

设置IKFAST输出路径

```
export IKFAST_OUTPUT_PATH=`pwd`/ikfast61_"$PLANNING_GROUP".cpp
```

生成IKFAST文件（6轴）

```
python `openrave-config --python-dir`/openravepy/_openravepy_/ikfast.py
```



需要注意的是：机械臂的初始位置不能是奇异姿态，否则会报错

创建插件

export

MOVEIT_IK_PLUGIN_PKG="\$MYROBOT_NAME"_ikfast_"\$PLANNING_GROUP"_plugin

cd ~/catkin_ws/src

catkin_create_pkg "\$MOVEIT_IK_PLUGIN_PKG"

roslaunch moveit_kinematics create_ikfast_moveit_plugin.py "\$MYROBOT_NAME"

"\$PLANNING_GROUP" "\$MOVEIT_IK_PLUGIN_PKG" "\$IKFAST_OUTPUT_PATH"

重新编译工作空间

catkin_make

修改使用的插件

roscd "\$MYROBOT_NAME"_moveit_config/config/kinematics.yaml

<planning_group>:

kinematics_solver:

<myrobot_name>_<planning_group>_kinematics/IKFastKinematicsPlugin

-INSTEAD OF-

kinematics_solver: kdl_kinematics_plugin/KDLKinematicsPlugin