Developing for the OpenRTM Based Software Controller(Example: Arm Control)

*Yuichiro KAWASUMI(General Robotix, Inc.), Hajime SAITO (General Robotix, Inc.), Noriaki ANDO(AIST), Rosen Diankov(TokyoUniversity)

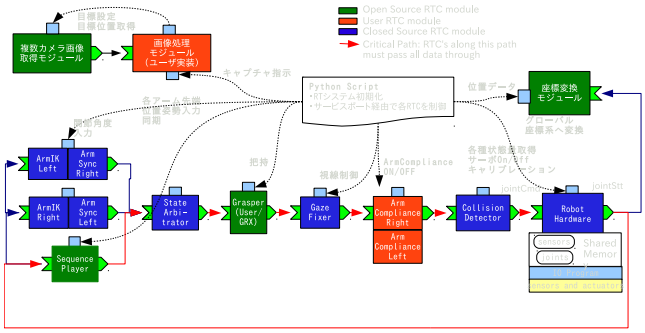

Fig.1 HIRO OpenRTM Controller(Planned Final Version)

1. Introduction

With the advent of open source in the robotics community, we are seeing more examples of systems being used together to get results. We have a sample RT component that uses OpenHRP[1][2][3], OpenRTM[4] and OpenRAVE[5], which shows what can be achieved by combining these open source systems. In this work we hope to show how to create basic controllers for the dual armed HIRO robot, and what we did to create a software controller that can move an arm using inverse kinematics(IK).

2. Control System

The control system of the HIRO dual armed robot is comprised of several components that communicate with each other over the OpenRTM middleware. The core controller components talk to each other in real-time. The way the components are linked is illustrated in Fig.1.

The Arm Control components are named ArmIK Right and ArmIK Left and are components that must respond in real-time. OpenHRP has a convergence based solver that will be used in cases where an analytic function does not exist. However, due to it's nature, convergence functions are not suited to real-time functions.

With the added support of Collada in OpenHRP, it is now possible to take full advantage of the ikfast[6] component available in OpenRAVE to generate an analytic IK function for most robots.

3. Method

The model required by OpenRAVE is Collada. However, the model used in OpenHRP has traditionally been VRML. With the added export Collada export capability of OpenHRP you can export a Collada file from the original VRML data. This Collada file is used by the ikfast function to generate a C++ source file of the analytic IK function.

Together with the basics required to create an RTC component for the HIRO robot, we will show how the generated source is made into a shared library, and loaded into the RT Component. We will also show what is required when connecting the OpenRAVE types with the OpenHRP types, typically used in our control software.

4. Summary

With the added capabilities to OpenHRP, it is now easy to create Arm control components for OpenRTM controllers. In the future we would like to further refine this component to support further different types of Arms.

References

[1] Hirohisa Hirukawa, Fumio Kanehiro and Shuuji Kajita: "OpenHRP: Open Architecture Humanoid Robotics Platform," Proc. Int. Symp. Robotics Research, 2001.

[2] Hirukawa,H., Kanehiro,F. and Yokoi,K.: "HRP develops OpenHRP," IEEE/RSJ IROS Workshop on Explorations towards Humanoid Robot Applications, 2001.

[3] Fumio Kanehiro and Hirohisa Hirukawa: "OpenHRP: A Software Platform for Humanoid Robotics," Humanoids 2006 Workshop on Humanoid Technologies, 2006.

[4] Tetsuo Kotoku, Noriaki Ando and Geoffery Biggs: "Tutorial Introduction to RT-middleware(OpenRTM-aist)," IROS'09 Proceedings of the 2009 IEEE/RSJ International Conference on Intellitgent Robots and Systems, 2009.

[5] Rosen Diankov: "OpenRAVE: Open Robotics Automation Virtual Environment," `http://openrave.org`

[6] "IKFast: The Robot Kinematics Compiler," `http://openrave.programmingvision.com/en/main/openravepy/ikfast.html?highlight=ikfast#openravepy.ikfast`