R5-COP

Reconfigurable ROS-based Resilient Reasoning Robotic Cooperating Systems

Kinematic chains

Jan Fraś (PIAP)

Project	R5-COP	**Grant agreement no.**	621447
Deliverable	D23.30	**Date**	25.01.2016
Contact Person	Adam Dąbrowski	**Organisation**	PIAP
E-Mail	adabrowski@piap.pl	**Diss. Level**	PU

Document History			
Ver.	**Date**	**Changes**	**Author**
0.1	25.01.2016	Initial version	Jan Fraś, PIAP
0.2	26.01.2016	For internal review	Jan Fraś, PIAP
0.3	29.01.2016	Final version	Jan Fraś, PIAP

Content

1 Introduction

1.1. Summary (abstract)

Change of robot hardware results in change of its physical properties. The physical structure of the robot is crucial from software point of view. In particular the inverse kinematics algorithms have to be aware of a kinematics of the robot in order to determinate correct control. In order to remain the robot able to control its movement, such a change has to be handled in software. In this task a proper software has been developed. The software solves an inverse kinematics of any kinematic chain described in URDF format, and enables easy reconfiguration of the robot structure.

1.2. Purpose of document

This Document describes a software that has been developed in PIAP in task 23.3. It discuss the main features and interfaces of the software. It is meant to be an overview description of the existing prototype, a source of interface descriptions, and as a presentation of concept and it's realization. The simple use case is proposed and described.

1.3. Partners involved

Partners and Contribution	
Short Name	**Contribution**
PIAP	Author

2 Issue

The problem related with robot actuator reconfiguration is that any change of the hardware results in a change in robots kinematics. The geometry not consistent with the setup that software is aware of, may result in unexpected behavior or unpredictable movements. It is likely that after any reconfiguration, type of joints and their localization in the actual manipulator may change. Two example kinematic chains are presented in Figure 1. They are composed of the same number of joints and links. Joint coordinates are set to the same values for both chains, but third joint is of different type in both cases. The final tip pose is fairly different for both chains.

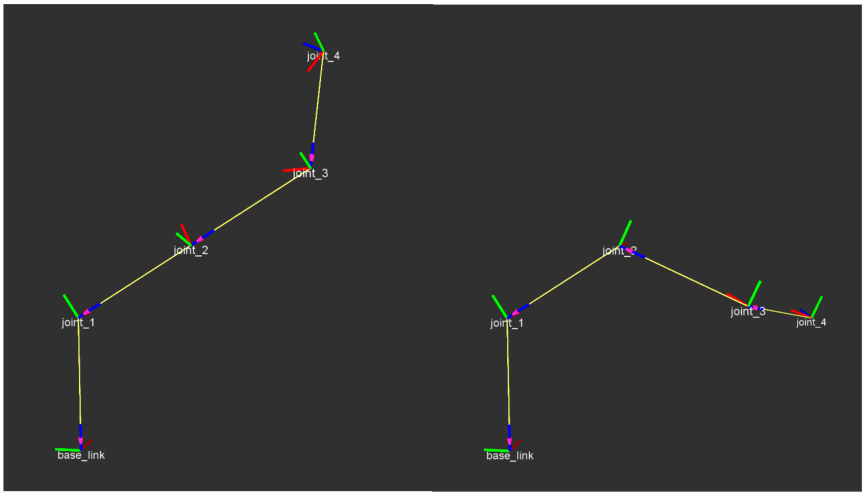


Figure 1: Example of two different kinematic chains. Joint coordinates set to the same values in both cases.

Example use case of runtime manipulator reconfiguration is a change of an end effector. In such case, a geometry of the whole system changes, the tool-central-point moves to another location, and both the joints and the links are likely to change its types and locations.

3 Solution

A ROS node for inverse kinematics solving that is able to handle runtime manipulator reconfiguration has been developed. The package provides the inverse kinematics service that returns JointState message as a response for the requested effector goal position expressed in Cartesian coordinate system. The node creates a virtual model of the manipulator resulting from the URDF description. The node expects to be provided with two separated URDF descriptions one for the arm and another for the effector. Change of the effector description is possible at any time. Such functionality is realized by another service provided by the node. In order to initialize the software, URDF descriptions of the arm and the effector has to be loaded into the ROS parameter served.

Each description change is followed by re-initialization of all the IK solving components. Re-initialization is done having regard to the current manipulator configuration in order to minimize impact on the system work. The re-initialization takes relatively short time and is done in real-time.

3.1 Interfaces

The node provides three services with the following definitions:

- ***CalculateIK.srv*** for inverse kinematics request***:***

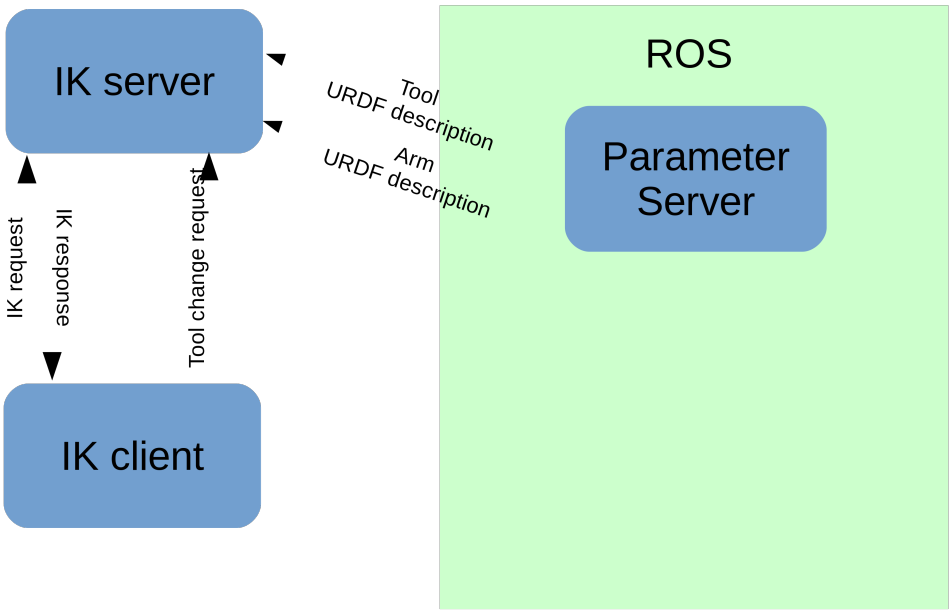
```
geometry_msgs/Pose goal
---
sensor_msgs/JointState jntCoordinates
```

- ***InitJoints.srv*** for joint coordinates initialization:

```
string manipulator_part #arm | effector | both
float64[] jointCoordinates
---
bool return_value
```

- ***ChangeEffectorDescription.srv*** for changing the effector of the manipulator:

```
string parameter_name
string base_joint
string tip_joint
---
bool return_value
```



4 Summary

Change of robot hardware results in change of its physical properties. In order to handle such a situation from inverse kinematics task point of view, a ROS package has been implemented. The package provides mechanisms that enables an exchange of a robot effector while system is running. It implements a services for inverse kinematics solving, effector change and joints coordinates initialization. The software solves an inverse kinematics of any kinematic chain described in URDF format. The package has been tested in simulation for the proposed tool-exchange use case. The behavior of the system is stable and change of the effector does not cause any unexpected movements.

The code of the package can be found at: https://github.com/piappl/kinematic_chains