

Manipulator Modeling & Control

1 Content of this lab

The goal of this lab is to program in C++ the three fundamentals models for robot manipulators:

- Direct Geometric Model: what is the pose of the end-effector for a given joint configuration?
- Inverse Geometric Model: what joint values correspond to a given end-effector pose?
- Kinematic Model: what is the velocity screw of the end-effector when the joints move?

These models will be applied on three different robots, and used to perform basic point-to-point control.

As it is not a lab on C++, most of the code is already written and you only have to fill the required functions:

- `Robot::fMw(q)` for the direct geometric model wrist-to-fixed frame
- `Robot::inverseGeometry(M)` for...well, the inverse geometry
- `Robot::fJw` for the kinematic model wrist-to-fixed frame

This project uses the ROS¹ framework which imposes some particular steps to configure the environment. They are detailed in Appendix A.

The **only** files to be modified are:

- `control.cpp`: main file where the control is done depending on the current robot mode
- `robot_turret.cpp`: model of the RRP turret robot
- `robot_kr16.cpp`: model of the industrial Kuka KR16 robot
- `robot_ur10.cpp`: model of the Universal Robot UR-10

We will use the ViSP² library to manipulate mathematical vectors and matrices, including frame transforms, rotations, etc. The main classes are detailed in Appendix B.

¹Robot Operating System, <http://www.ros.org>

²Visual Servoing Platform, <http://visp.inria.fr>

A How to load the C++ project inside Qt Creator

An actual ROS course will be held in the second semester, for now just configure as follows. You should have a folder named `ros` in your home directory. Open a terminal inside and follow these steps:

1. Clone this project inside `ros/src`

```
mkdir src
cd src
git clone https://github.com/oKermorgant/ecn_manip.git
```

2. Compile using catkin (you'll discover soon enough what it is) : `catkin build`
3. Run Qt Creator from the top icon
4. Load the `ros/src/ecn_manip/CMakeLists.txt` file through `File...open project`
5. QtCreator asks for a compilation folder: give `ros/build/ecn_manip`
6. The files should be displayed and ready to compile and run
7. Compilation is done by clicking the bottom-left hammer
8. Run your program with the green triangle. It can be stopped by clicking on the red square

B Using ViSP