Robot programming by Teaching: Robot Franka Emika Panda

Group 2

Supervisor: Jacob Hernandez

Mariane Brodier
Mst. Robotic Minor. Biomedical Technologies
Sciper: 250488

Théophile Balestrini
Ms. Robotic Minor. Space Technologies
Sciper: 306674

Christopher J. Stocker
Mst. Robotic Minor. IoT
Sciper: 266575

I. Introduction

In this lab, we were introduced to the Franka Emika Panda robot and how to interact with it safely. In order to control it, we used an interface Desk. We have learned to program it by teaching it the movement to do a specific task.

Prior to the lab, we had to answer some theoretical questions. After that, we could familiarize with the Franka robot and its interface Desk. Then, we performed multiple tasks, such as pick and place, flip and place. Finally, we played with all the different functionalities in order to create our own custom task.

II. Theoretical questions

Describe different methods to program robots other than teaching.

Most popular method of robot programming is probably teach pendant. Then there is simulation which involves offline programming. This has the advantage to reduce downtime required for robot programming as these are developed offline, however, simulations will never represent the real world with 100% accuracy. Then other methods of learning besides teaching by demonstration includes using more modern tool such as machine learning, reinforcement learning, evolutive learning, shared learning.

What is the difference between the joint space and task space of a robot?

The joint space describes each joint position (Franka robot: angular position). The task space describes the end effector position in a defined coordinate system. The two are linked by the kinematic model.

How much redundancy does the Franka Panda have?

The Franka Panda have 7 joints for 6 DOF. This means it has one redundancy.

What are the uses of this redundancy?

Increased redundancy allows for greater robot flexibility, agility, and over all motion dexterity with the added degrees of freedom, in these cases the added degree of freedom give rise to infinite solution to the inverse kinematic problem.

What is forward/inverse kinematics?

Forward/inverse kinematics links the joint space to the task space position and vice-versa (coordinates). It links the generalized velocities (in joint space) to time-derivatives of the end-effector configuration representation.

What is inverse dynamics?

Inverse dynamics maps the speed/acceleration from the task space to the joint space.

What is a Jacobian matrix? A Jacobian matrix is the derivative of the direct kinematic model.

What is the difference between an analytical and geometrical Jacobian?

The difference remains in the number of joints and arms length (robot's geometry). The geometric Jacobian (basic jacobian) relates the generalized velocity ($\dot{q}$) to the velocity of the end-effector (linear v_e and angular ω_e).

What constrains the reachable space of an end-effector?

The reachable space of an end-effector is constrained by the physical mechanical constraints.

What is a singularity of a serial robot? In practice how do you deal with them?

A singularity is when the pose produce a situation where local movement of one or more joints does not affect end-effector's position/orientation.

What is manipulability, what is the formula?

Manipulability of a robot is the capacity to change the end effector's position as a function of the joint configuration. The manipulability measure reduced to zero when the robot is in a singular configuration and cannot generate velocities in at least one direction

Manipulability is calculated by first defining the robot's configuration in terms of joint angles (forward kinematics) and then calculating how the joint velocities generate velocity of the end effector (velocity kinematics). The manipulability measure is calculated as the product of the diagonal elements

σ_i in the Σ matrix that results from the singular value decomposition of the velocity Jacobian, $\mu = \prod_{i=1}^{m} \sigma_i$. It may also be defined relative to the determinant of the Jacobian as $\mu = \sqrt{\det{(JJ^T)}}$.

What could this robot be used for?

This robot could be used to pick and place, packing, grasping.

Franka Panda is a torque-controlled collaborative robot (COBOT). What advantages and disadvantages do torque-controlled robots have?

The advantages of torque-controlled robots are that they support their own weight in any position, allowing for the user to move the arm easily; it enhances precise control of forces in task space. The disadvantages are that they are less precise in positioning.

In what ways could the robot be potentially dangerous?

The robot could be dangerous because one could get pinched between the end effector and rigid objects; or because the robot could be manipulating sharp object; the robot could drop heavy objects in an unplanned manner; one could collide with the fast-moving arm.

What could be challenging for a robot of this type compared to parallel robots?

Parallel robots are intrinsically more accurate than serial robots because their errors are averaged instead of added cumulatively. So it might suffer from accuracy when fast movements are required.

For robot control, what are two relevant reference frames to consider in task-space control. Which are they?

The two relevant reference frames to consider in task-space control are the end-effector's frame and the grasped object's frame.

How do you compute the transformation between these two coordinate systems?

In order to compute the transformation between these two coordinate systems, one should multiply the coordinates in the first reference frame by the rotation and translation matrices that link it to the second reference frame.

III. Familiarize with Franka Emika Panda

How is the robot limited?

The robot is limited when it is in a singular position, e.g. arm fully extended, segments aligned. Then of course it is limited by the motors in speed and acceleration, as well in accuracy.

Explain briefly what each guiding mode does.

Guiding modes facilitate guiding by locking or unlocking different directions or rotations in space. For example, the arm can be moved in three directions in space. You can switch between guiding modes either using the guiding mode button on the Grip or directly from Desk.

Global versus local coordinate system. What do you notice? Global: +10z $\rightarrow$ moves up, +10x $\rightarrow$ moves front. Local: +10z $\rightarrow$ moves down, +10x $\rightarrow$ moves front. Coordinate systems are rotated 2pi around X.

7.4 How is the global coordinate system oriented?

Fig. 1: Schematic of global coordinate system.

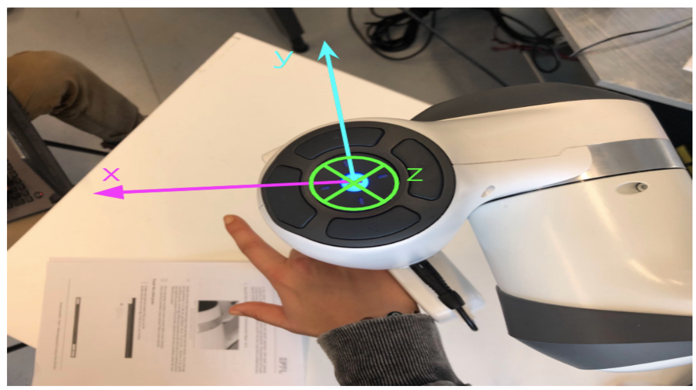

Fig. 2: Schematic of global coordinate system from above.

7.5 How is the local coordinate system oriented? Make a clear schematic.

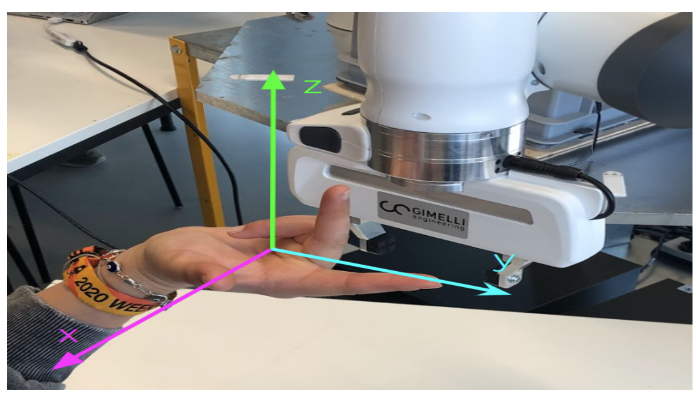

Fig. 3: Schematic of local coordinate system.

IV. Exercise 2: Pick and place task

8.1 Using the unclasp move and snatch move apps only, make a task that picks up a pen (10 grams) and releases it to another location. Add a screenshot of your task in the report.

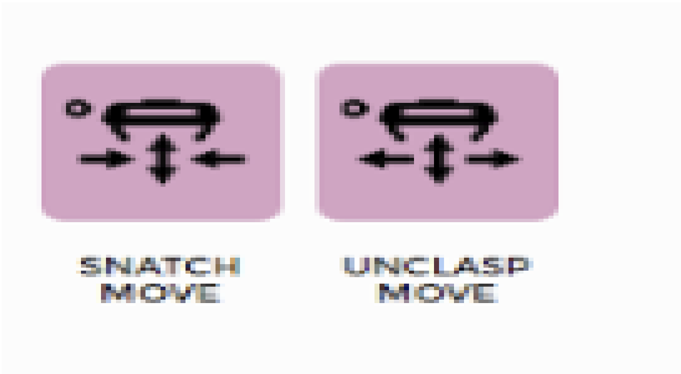

Fig. 4: DESK view of task: snatch and unclasp move.

8.2 Between the pick-up location and the release location, add another object (i.e., your backpack for example). The robot now has to avoid colliding with the object. Add to your task the cart motion app to program the task and generate a trajectory that follows approximately the one in red in the figure below. Add the screenshot of your task in the report.

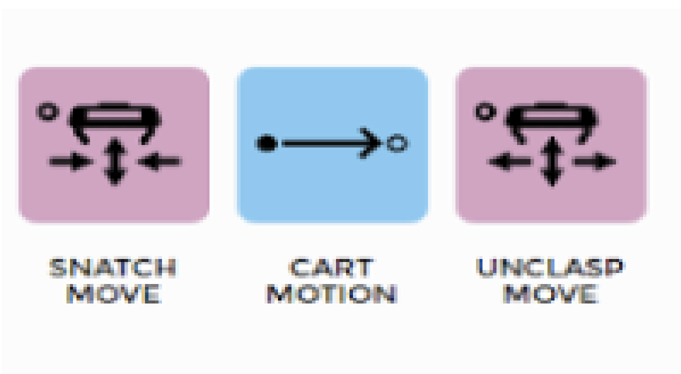

Fig. 5: DESK view of task: snatch, move, and unclasp move.

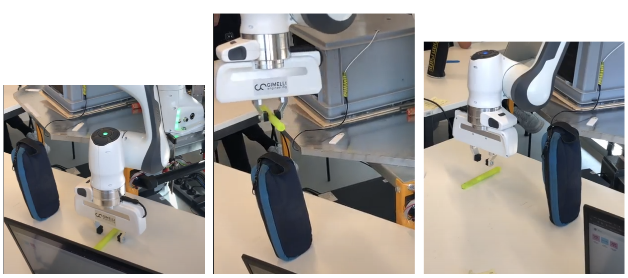

Fig. 6: Programming of motion above pencil case.

8.3 When you want to generate a trajectory that doesn't solicitate too brutally the actuators what do you want to avoid in the acceleration profiles?

You want to avoid large spikes in the acceleration profiles, meaning slower speeding and braking between target point.

8.4 The figure below shows the velocity profile of panda joint n°7 when performing a simple motion. Is the corresponding acceleration profile continuous? or discontinuous? What can you conclude?

The acceleration profile is continuous. We can calculate the acceleration as the derivative of velocity. Between $t = 1.75s$ to $t = 2.4s$ we see that the acceleration is equal to $46.1\frac{um}{s^2}$ and between $t = 2.4s$ to $t = 3s$ the acceleration is equal to $-46.1\frac{um}{s^2}$ and zero everywhere else including at $t = 2.4s$.

8.5 Using the available apps in Desk, find one method to make the trajectory even smoother. Explain your approach and add a screenshot of your task to the report.

We added the smooth motion to our trajectory.

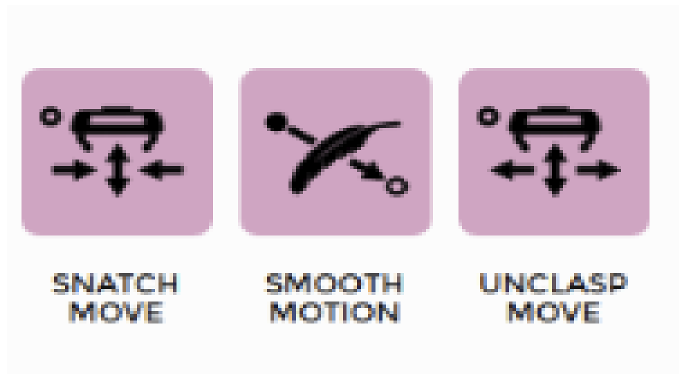

Fig. 7: DESK view of task: to smooth the trajectory.

V. Exercise 3: Flip and place task

This is the DESK view for programming the pen manufacturing plant.

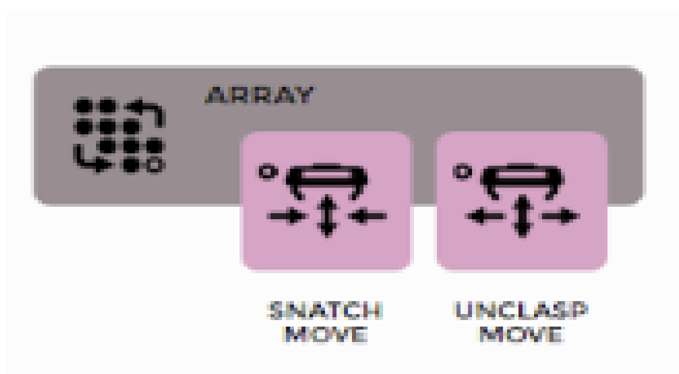

Fig. 8: DESK view for pen manufacturing plant.

This is the video showing its results.

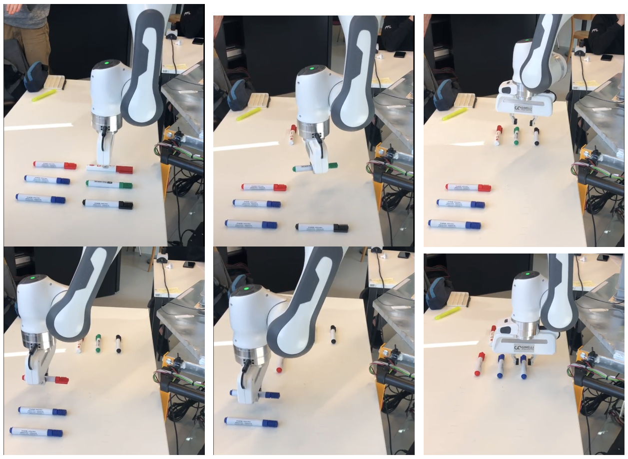

Fig. 9: Programming pen manufacturing plant on two conveyor belt.

VI. Exercise 4: Create your own custom task

10.1 What is your task doing? We tasked our robot to use a marker and play the brain game of drawing a house without lifting the marker.

10.2 Explain how your task works, explain the choices you've made. We first had to register the robot to hold a marker. Then to find and touch the table to mark the position of the surface of the table. Then to apply pressure up to a certain threshold (0.3 N). Lastly to apply a sequence of planar displacements to create the drawing. At every two stroke interval we decided to add pressure between the marker and the surface to ensure there wasn't any loss of contact.

10.3 For which applications could your task be used for? In more general terms this type of procedure and application could be used to replicate position tracing. We cold replicate images and the marker could be swapped out with a different tool. For example take the case of a patient needing precise knife incision to open a tissue. Ours of course is much more basic level but the general concept is there.

10.4 Add a screenshot of your task to the report Unfortunately for this part we forgot to take a screenshot as Professor Bouri asked us to his his project on rehabilitation robots.

The figure below shows the results of our tasks at different time intervals.

VII. Final questions

What limitations did you find in programming by teaching? Programming by teaching many practical benefits, however, one of the limitations we found was the fact that the learning was teacher pendant, meaning it needed the physical robot for programming. Also precise coordinates were difficult to input as we were limited to our hand-eye precision.

Which type of behaviors cannot be easily programmed? Behaviors which include very precise control of the systems trajectory where difficult to program. This type of teaching by demonstration involves moving the robot around to sometimes target points. The program fills in the empty points of the trajectory sometimes resulting in some jerky movements of the robotic arm.

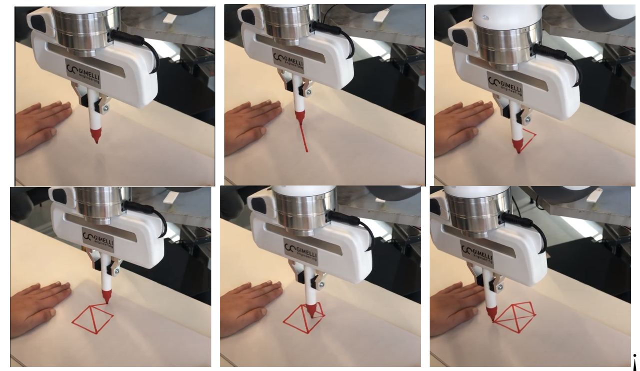

Fig. 10: Programming pen manufacturing plant on two conveyor belt.

Which alternatives do you have?
As alternative we had the possibility to

- Add the number of target points between the trajectory
- Use the *SMOOTH* feature in the **DESK** programming.
- Change the start and end positions so the trajectory is as "intuitive" as possible for the robot.
- Use some other pre-programmed features of the **DESK** interface, such as *ARRAY*

VIII. Conclusion

We learned the advantages and disadvantages of robot programming by teaching. We saw that the practicality of the software eliminated the need of specialized technician and was able to accomplish some mildly complex tax (pen manufacturing conveyor belt) with relative ease. However programming by teaching give certain key instructions to the robot and leaves the rest open for it to extrapolate. This leaves room for certain unwanted behaviors such as jerky motions, and movement's that are either too slow , or too quick. This method of teaching is optimized for fast programming but not for efficient motion.