Course: Advanced Robotics

A report
of

PW4: Force Processing with C3 Robot (EPSON): Study and Implementation

Supervisor: Abdenbi Mohand Ousaid

Done by:

UMUHOZA Jean d'Amour

UGWU Hillary Emeka

Département Automatique et Robotique

List of figures

Contents

1.Introduction

Currently the world is developing more and more, and this development come with the increase of automated robotics technology which is mostly contributing in the industrialization [1]. Robotics is mostly applied in industries because, robot can do the jobs that are dangerous for humans, repetitive jobs that are boring, stressful, or labor-intensive for humans, menial tasks that human don’t want to do. However, Implementation of automated robotic solutions for complex tasks currently faces a few major hurdles [2]. For instance, lack of effective sensing and task variability – especially in high-mix, low-volume processes – creates too much uncertainty to reliably hard-code a robotic work cell. Currently, collaborative frameworks generally focus on integrating the sensing required for a physically collaborative implementation.

In this practical work, we have implemented some program using the effort tensor, and as it was stated, the information was provided by a 6 Degree of Freedom force (DoF) sensor embedded on the robot wrist.

Six degrees of freedom (6-DoF) refers to the freedom of movement of a rigid body in three-dimensional space. Specifically, the body is free to change position as forward/backward (surge), up/down (heave), left/right (sway) translation in three perpendicular axes, combined with changes in orientation through rotation about three perpendicular axes, often termed yaw (normal axis), pitch (transverse axis), and roll (longitudinal axis).

2.The lab stet-up

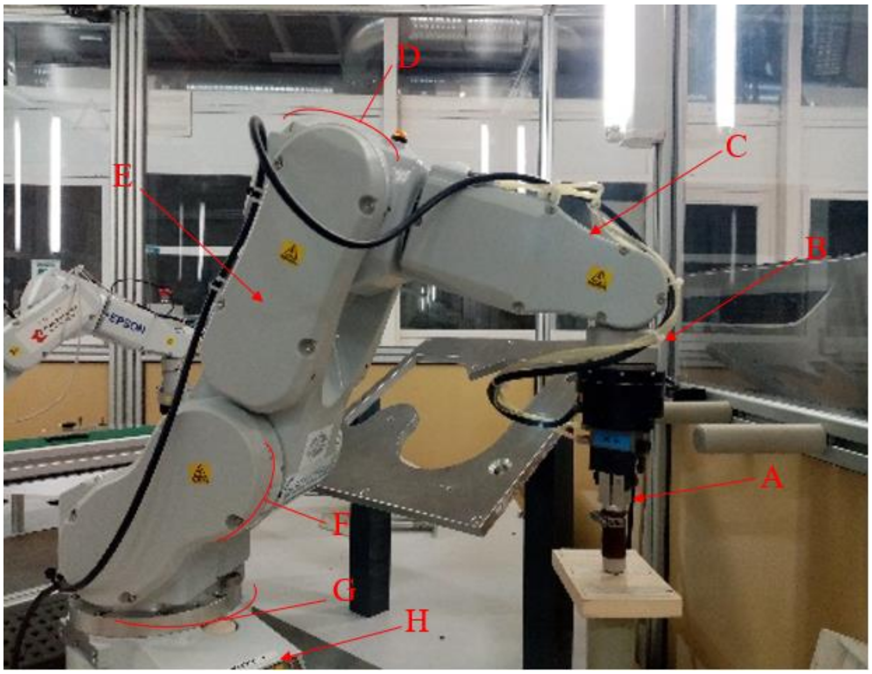

Figure 1: PW4 Set-up

A = Gripper

B = Wrist

C = Forearm

D = Elbow

E = Upper Arm

F= Shoulder joint

G = Waist

H = Base

3. The work done

3.1 Introduction to the work done

During this lab, the first thing was to be familiar with the C3 robot (EPSON) and its environment. We have integrated with some manual control of the this C3 robot (EPSON) and the interface as show in the figure 2 and figure 3 below.

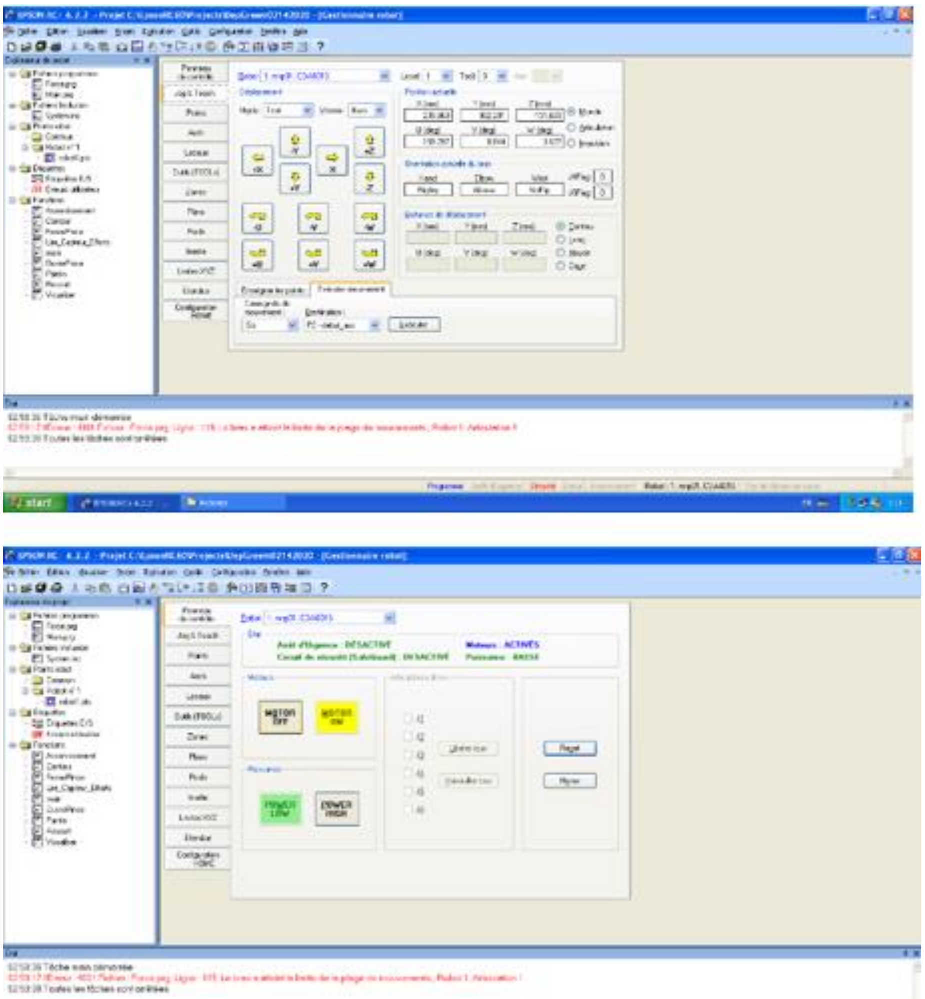

Figure 2:interface for manual control

The second step was to execute the provided pre-canned program. Initially, the program allows only visualizing the effort tensors applied on the robot wrist. With these pre-scanned programs, we have found the forces and torque of the robot which have also used in developing the program that makes the robot responding as a spring about z axis.

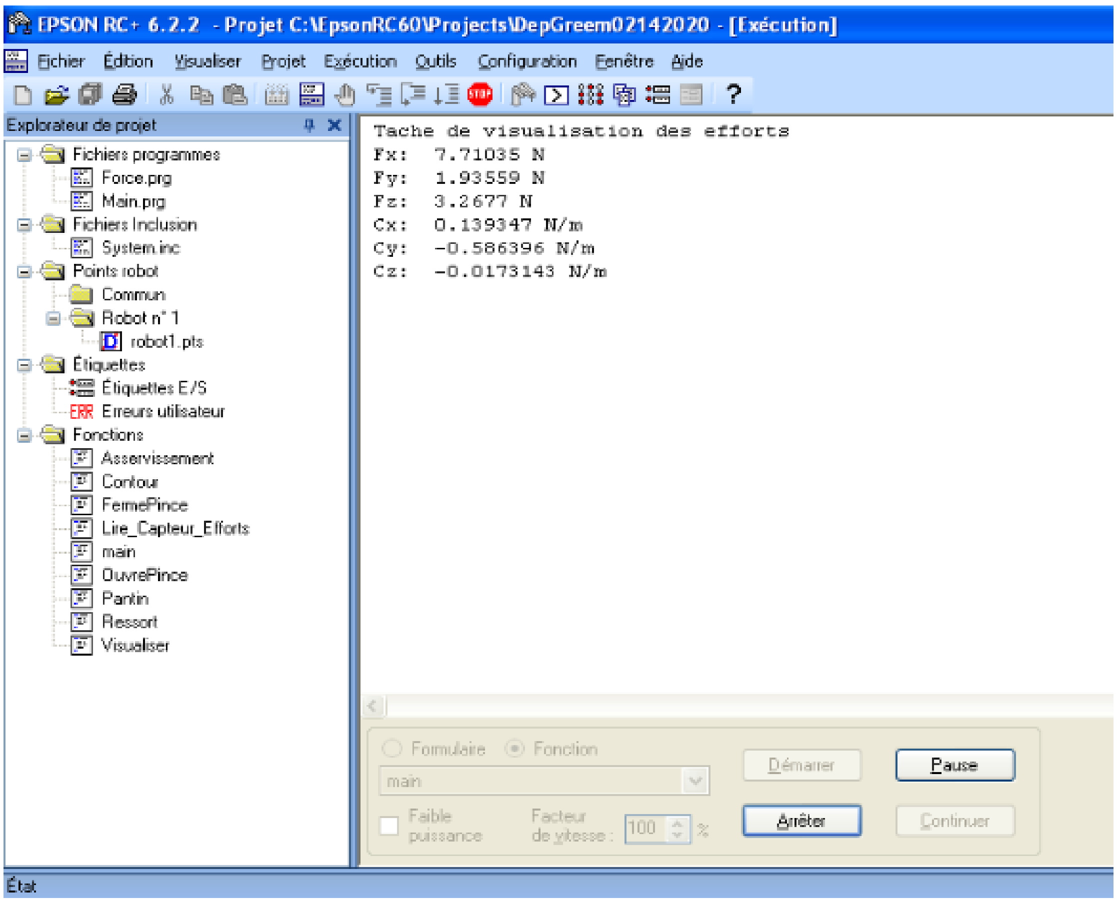

Figure 3: Forces and Torques found with pre-scanned program

3.2 Program that makes the robot responding as a spring about z-axis (Active compliance)

A robot arm can exhibit several different behaviors depending on the task and its environment. It can act as a source of programmed motions for tasks such as moving an object from one place to another or tracing a trajectory for manufacturing applications. It can act as a source of forces, for example when grinding or polishing a work piece.

So, during this lab with the use of the formula of the force in the equation (1), we were able to develop the following code which was useful for the purposed described below.

$$F = KX \qquad (1)$$

with K to be the stiffness

and

$$X = SF \qquad (2)$$

With S to be the compliance, inverse of the stiffness ($S = \frac{1}{C}$)

The figure below does not show all the codes used. However, it shows the important codes developed that makes the robot responding as a spring about z-axis. So, the codes are as follow:

```
Do While MemSw(Utilise_Efforts) = On
    'recuperer la valeur de l'effort en X,Y,Z (EffortX, EffortY et EffortZ)

    EffortX = Force_GetForce(FORCE_XFORCE)
    EffortY = Force_GetForce(FORCE_YFORCE)
    EffortZ = Force_GetForce(FORCE_ZFORCE)

    '.........
    'Calculer la souplesse

    Souplesse = 1 / k

    '.........
    'Calculer la valeur du deplacement (l1,l2 et l3) selon X, Y et z respectivement
    l1 = Souplesse * (EffortX - 7.8)
    l2 = Souplesse * (EffortY - 2.2)
    l3 = Souplesse * (EffortZ - 3.9)
    '.........

     'grace a la valeur du deplacement obtenue, faites evoluer le robot de la position initial

     Move Here +TLX(l1) +TLY(l2) +TLZ(l3) CP

    '............
    Wait 0.500
```

Figure 4:Program for the robot to respond to the applied force

With the help of these code in the figure above, we were able to give our robot the ability to responds to the force applied on it. Suppose when you push or pull this robot because of the ability given by the above code it can move in the direction of the force applied on it.

So, in practice the advantage of such program is that robots are more than capable of performing a wide range of machining tasks desired by the operator since it can move in the direction for the desired wishes of the operator.

3.3 Robot Force control

Today's industrial robots are almost always programmed using a position control scheme. Typically, the robot tool follows a prescribed trajectory in space which has been pre-programmed or taught by operator before run-time as shown in the above section 3.2. sometimes, the robot gets its reference position from a vision system, which enables it to adapt its motion to its changing environment. However, for some applications, it is more important to precisely control the force applied by the end-effector rather than controlling the robots positioning. Again, industrial manipulators are under various limitations against high quality motion control, such as both frictional and dynamic disturbances should be dealt with a simple PID structure [3].

In that case, our aim in this section was to implement the PID controller in order to master the force applied by the robot and the force control programming about Z axis of the tool was developed.

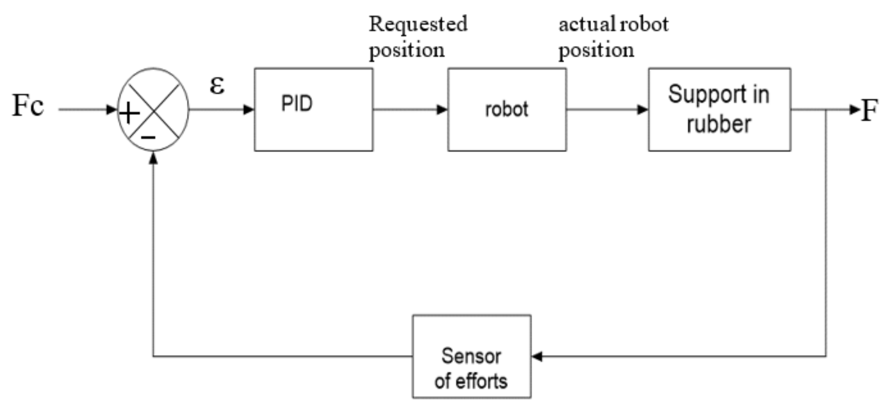

Figure 5: PID control feedback block diagram

The codes developed for the force control was the following as shown in the figure below.

```
     'definir les valeur de la consigne, action integrale et
proportionnelle


      'Fc = 'force de consigne
      Fc = -2
      ' Kp = 'action proportionnel
      Kp = 0.01
      ' Ki =  'action intgrale
      Ki = 0.001
     Ai = 0

     Do While MemSw(Utilise_Efforts) = On
         'recuperer la valeur de l'effort en Z
         EffortZ = Force_GetForce(FORCE_ZFORCE)
         EffortZ = EffortZ - 3.9

         'calculer l'erreur d asservissement en effort
(Erreur)
       Erreur = Fc - EffortZ

         'calculer l action integrale (Ai)
       Ai = Ki * Erreur

         'calculer l action proportionnelle (Ap)

         Ap = Kp * Erreur


         'appliquer les calcul de votre shéma fonctionnel
(PosDec)

       PosDec = Ap + Ai
         'grace a la valeur de la force obtenue faite evoluer
le robot de la position initial
```

Figure 6: codes for force control about Z-axis

During this lab, the controller action implemented is shown in the above code, where we have considered only the variations of proportional and integral gains. This PI controller was used because it is needed for non-integrating processes, meaning any process that eventually returns to the same output given the same set of inputs and disturbances and is the same case as our robot. A P-only controller is best suited to integrating processes. Integral action is used to remove offset and can be thought of as an adjustable.

The example of where we can use this type of PID is a control is in a pick-and-place robot. Our controller will reside in the electrical drive and the control section of a pick-and-place robot. For example, there are typically three control loops: torque, velocity, and position. Typically, the A drive will control the torque loop, with the "controller" handling the speed and position control.

3.5 Reflection about force control

As said in the section 3.4 it is sometimes occurring that the robot can get the reference position with the help of vision system such as 6 DoF force sensor embedded on the robot wrist in our case, and with this it is able to adapt its motion to its changing environment. However, for some applications, it is more important to precisely control the force applied by the end-effector rather than controlling the robots positioning. Some application of this is the finishing of a machined part for example in robot polishing [3].

3.6 conclusion

During this practical work, as it aim was to deal with the complex robotic task and be familiar with the force processing of the C3 Robot (EPSON), we have investigated and be able to explain the subroutine program “pantin”, we were able to develop a short program with the help of given force formula for giving the ability to the robot to respond for the applied forces and finally, by the use PID control we have also developed and investigated the force control of the C3 Robot about Z axis.

Therefore, with this Practical work we have gained the basics programming of the industrial robot and some robot point teaching.

3.7 References

1. European Commission. 2015. Analysis of the impact of robotic systems on employment in the European Union. Luxembourg: Publications Office of the European Union.
2. Robotic Industries Association. (2008-2020). Dangerous Robot Jobs. Retrieved from https://www.robotics.org/blog-article.cfm/Dangerous-Robot- Jobs/15
3. [Hannaford 99] Rosen J., Hannaford B. et al., « Force Controlled and Teleoperated Endoscopic Grasper for Minimally Invasive Surgery – Experimental Performance Evaluation », IEEE Trans. on Biomedical Engineering, vol. 46(10), 1999, pp. 1212-1221