**Impedance Control and Compliance Control with UR5 Robot Study Report**

Jonathan Oh 941170383

Zitao Yu 941170755

1. **Abstract**

The purpose of this project is to implement an algorithm for manipulating flexible objects using impedance control techniques. (to be updated)

1. **Introduction**

To achieve the stated purpose, simulations are mainly done under MoJoCo environment. Therefore, the implementation of the project requires knowledge of impedance control and skills in python programming.

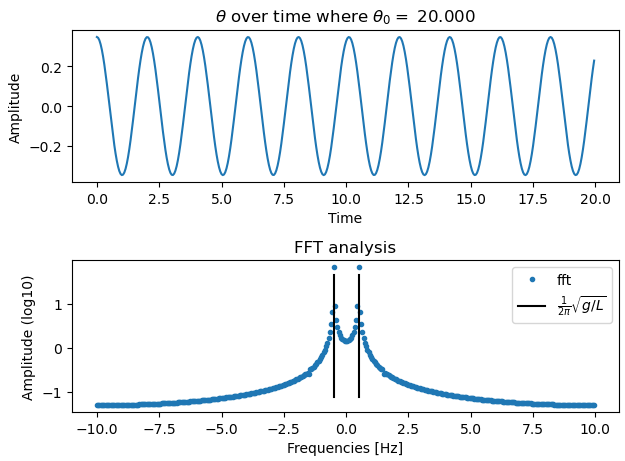
In this the first few weeks, we have learned the basic python programming and impedance control concepts by establishing some simple model related, as shown in the sub-section 1 of section III (Models).

In the following step, we (to be update)

1. **Models**
2. **Simple models executing python and impedance control** 
   1. **Pendulum and simple spring oscillator simulations in Python**

Based on the ideal double pendulum simulator, we added effect of damping on joint and elasticity of the rod. The main purpose of this practice is to be more familiar with Python. Related Python codes are on the related GitHub site.

Meanwhile, in order to verify if the simulation works the same as the physics, we analyzed the Oscillatory motion of one single pendulum with rod length equals to 1 meter and mass equals to 1 kilogram in gravity with G=9.8 . Horizontal displacement amplitude and Fourier transform of the oscillation are as presented in figure III.1.a)-1



**Figure III.1.a)-1: Oscillation and its Fourier transform of pendulum with ,**

* 1. **Impedance Control for a 2-Link Robot Arm**

The model is based on a project from the open resources in MathWorks. The original project is a user interactive 2-link robot arm that applied impedance control and returns demand joint torque values. On top of the original source code, we implemented few additional features. First, we separated the physics portion from the controller dynamics by assigning them different frequency as that is more similar to the real-world implementation. The robot failed to complete the desired missions when the controller frequency is too low, below 12% of the physical system’s calculation frequency. Also, we realized the dynamics-based and position-based control system basing on the existing model. Lastly, we observed the difference between the two control methods by comparing how the end-effector would follow the trajectory as well as the robustness and the accuracy of the systems.

For obtaining a better performance of the robotic arms, we examined the performance of the effects of controller parameters. Comparations between various values of K and B are shown in tables below. This comparation is done using DBIC in our codes with and as the controller group. Both trajectory results with and without wall are listed. The wall is located at x = 3[m], and is set to have a damping value of 50 and stiffness value of 100. (All these values have uncertain units and will be used only for comparations between variation of themselves.)

|  |  |  |  |
| --- | --- | --- | --- |
| K | 2 | 20 | 50 |
| Trajectory without wall | RMS = 0.3064 | RMS = 0.0716 | RMS = 0.0266 |
| Trajectory with wall |  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| B | 3 | 5 | 10 |
| Trajectory without wall | RMS = 0.1532 | RMS = 0.0716 | RMS = 0.0539 |
| Trajectory with wall | (Diverge after contact with wall) |  |  |

The results turned out that with larger stiffness value K, the faster tracking can be realized, however the motion will be stiffer with undesired oscillatory motions when large velocity change occurs. Meanwhile, it also turns out that increasing the value of B will significantly smoothen the trajectory and return a better tracking.

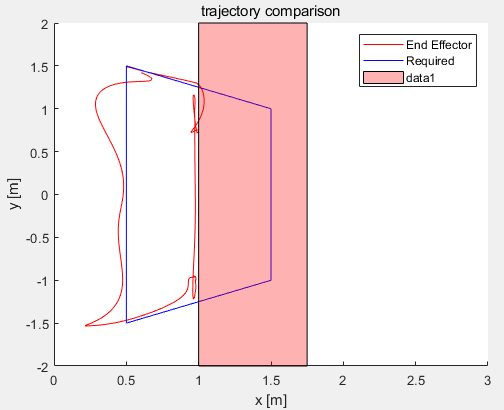
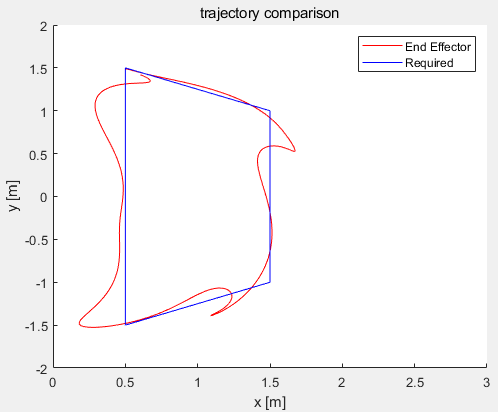
The improved DB-IC and PB-IC results obtaining by tuning with controller parameters are as shown below:

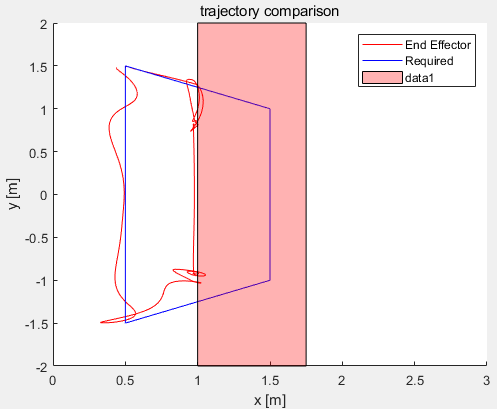
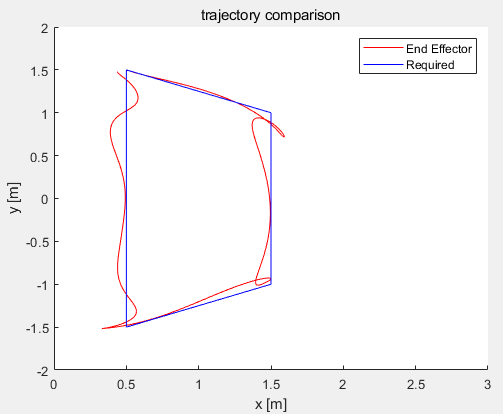
|  |  |  |
| --- | --- | --- |
|  | Without Wall | With Wall |
| DBIC (K=50, B=10, M=0.08) |  |  |
| PBIC (K=300, B=60) |  |  |

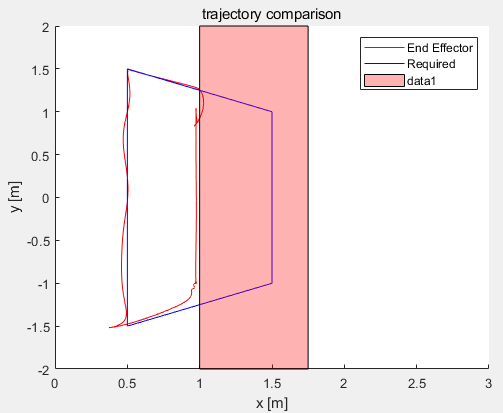
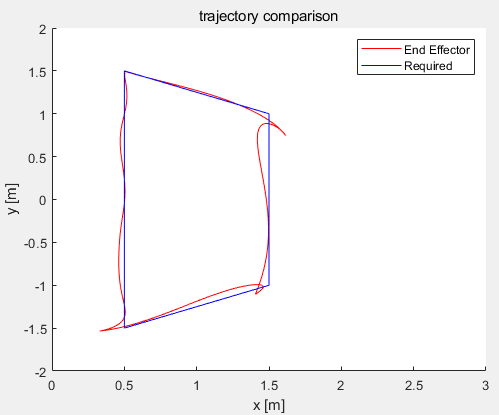
1. **Summary**
2. **Appendix**
3. **References**

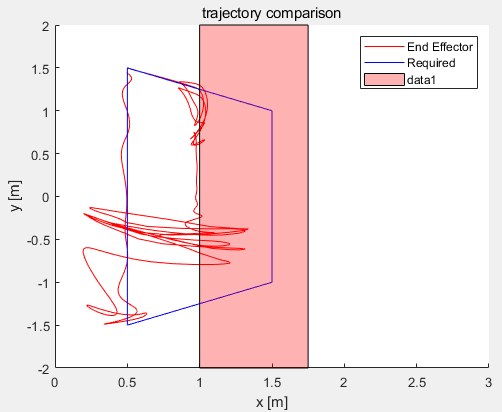
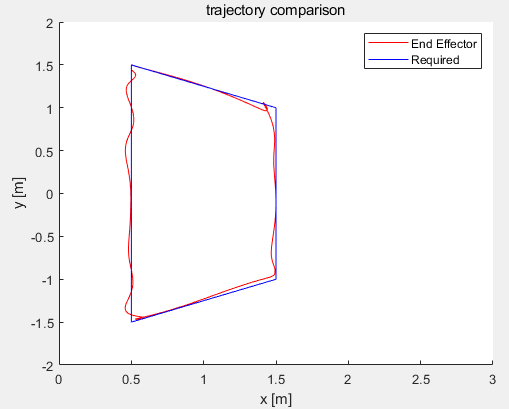
# [1] Matthew Sheen, MATLAB Impedance Control for a 2-Link Robot Arm - User-interactive <https://github.com/mws262/MATLABImpedanceControlExample>

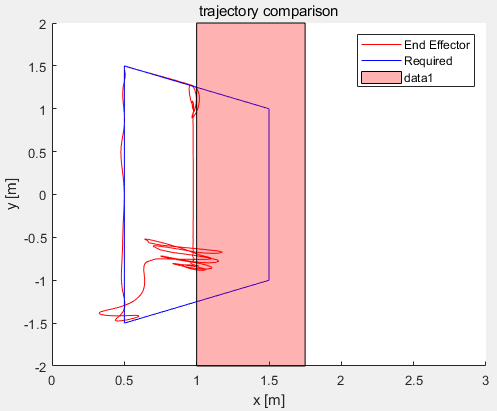
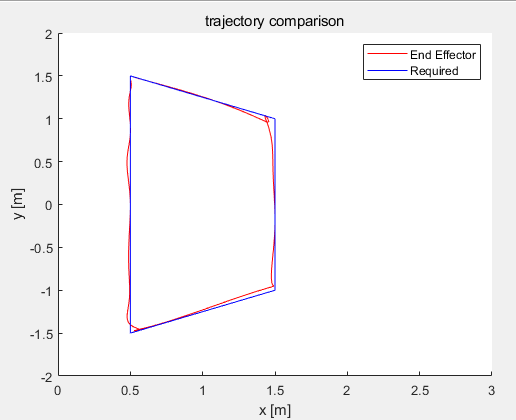
[2] Valency, T., & Zacksenhouse, M. (2003). Accuracy/robustness dilemma in impedance control. *J. Dyn. Sys., Meas., Control*, *125*(3), 310-319.

 K=150 B=30

 K=150\*2 B=30

 K=150\*2 B=60

 K=150\*5 B=30

 K=150\*5 B=60

|  |  |  |  |
| --- | --- | --- | --- |
| K | 2 | 20 | 50 |
| Trajectory without wall | RMS = 0.3064 | RMS = 0.0716 | RMS = 0.0266 |
| Trajectory with wall |  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| K | 2 | 20 | 50 |
| Trajectory without wall | RMS = 0.3064 | RMS = 0.0716 | RMS = 0.0266 |
| Trajectory with wall |  |  |  |