# **Weekly Report 14 – Fall 2022**

## **Problem & Task**

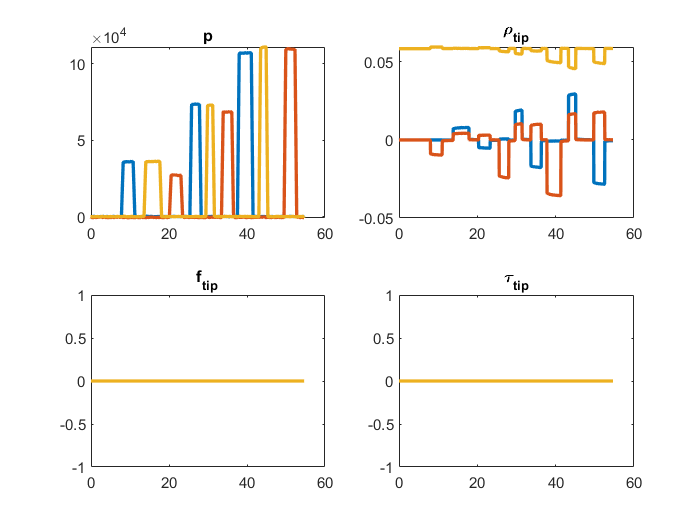
1. Mimic the simulations in paper [1] [2] [3] and check if using the same package “TMTDyn” will lead to similar results;
2. Adjust some basic parameters to see if the package would work;
3. Find the external force/load input file for the package, which could be used for our unique falling SRA simulation.

## **Solution**

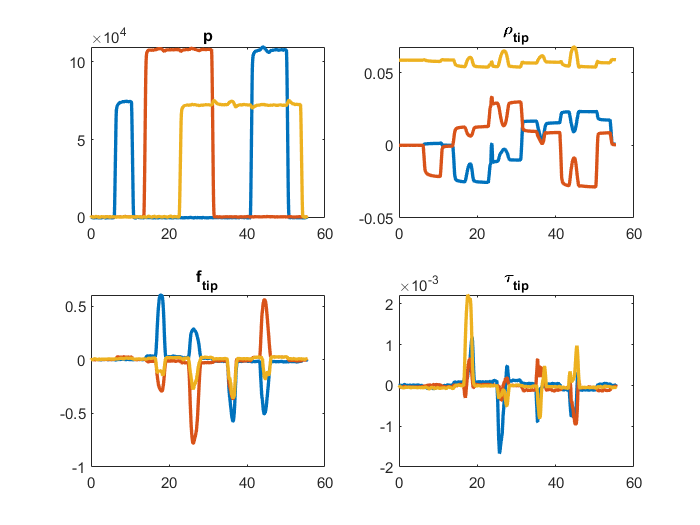
1. *Simulation Part*

There are basically 2 sets of tests performed based on the “TMTDyn” package in all the three papers mentioned previously, no load and with load for four different types of modelling methods, namely, SRL (series rigid link), EBR (Euler-Bernoulli Beam Theory with relative states), EBA (Euler-Bernoulli Beam Theory with absolute states) and ROM (reduced order method). We can practice using the package via doing exactly the same simulation in the papers, if the simulation results comply with or are similar to what were showcased in the paper, it can be proved that we are using it in the correct way.

The simulation results did not different so much in terms of different modelling methods, I will just post some key ones here. Firstly, no matter for which method, the inputs should be the same for no load and with load on the soft robot tip conditions respectively, which can be seen from below.



**Fig. W14-1** The input pressure without tip load



**Fig. W14-2** The input pressure with tip load

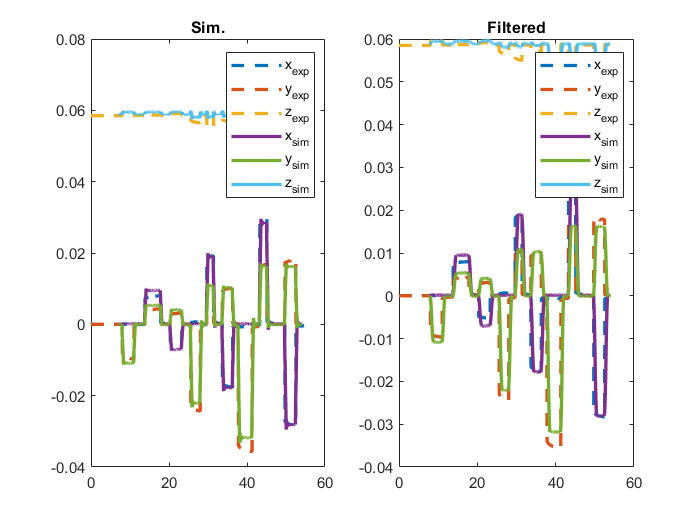
According to the guidance in the paper, to guarantee the accuracy of the simulation results, the bending damping coefficient and twist damping coefficient of the soft robot were set to be different values as shown in the table below.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Modelling Method** | **SRL** | **ROM** | **EBA** | **EBR** |
|  |  | |  |  |
|  |  | |  |  |

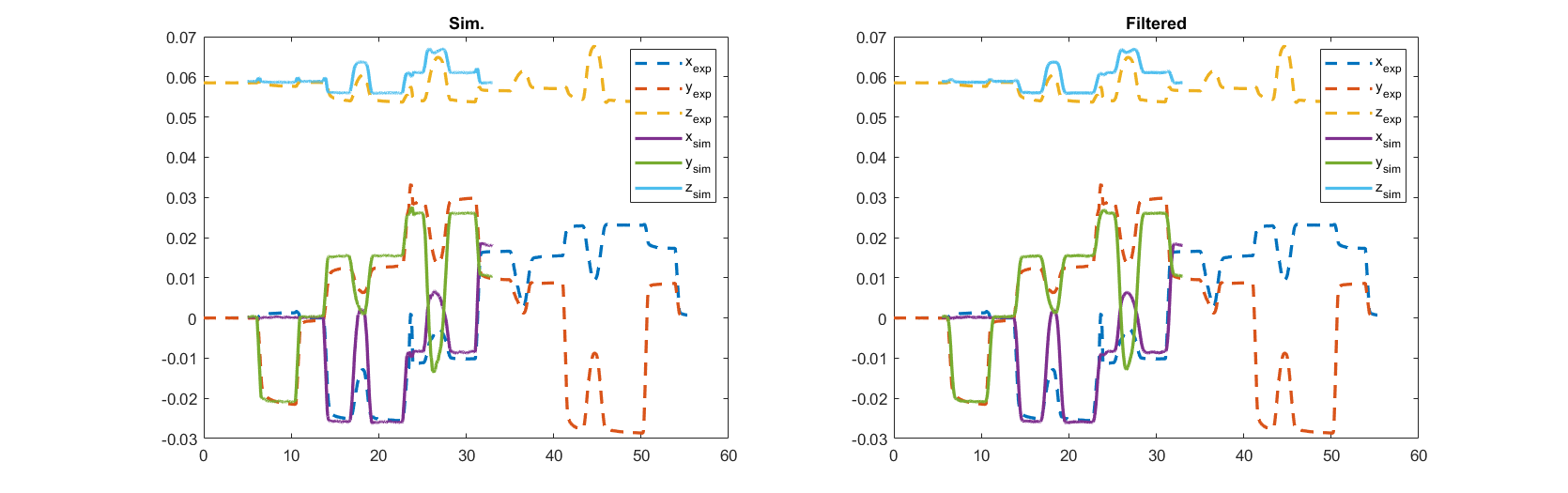
**Table W14-1.** The bending and twist damping coefficient setting for SRA

We can clearly find that the damping coefficients for EBA and EBR are much larger than the rest two modelling methods, that is because these two methods are pretty sensitive to the numerical instability, to compensate the system sensitivity to rapid changes in the states in dynamic simulations, higher damping coefficients are needed (in my understanding, the system is probably very stiff according to the investigation several weeks ago, so there will be some numerical integration problems). And another thing I could not understand is that why the damping coefficient would fluctuate within such a wide range for a same sort of material (as stated in the paper, it was measured experimentally).

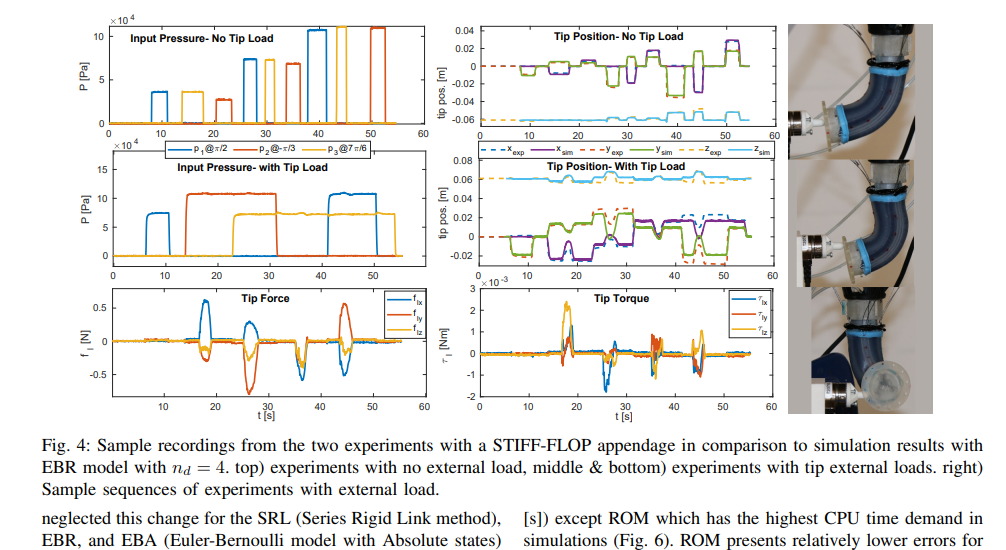
From the simulation results shown as follows, we can conclude that though there seems to be a larger gap between the positions of experimental measurement and simulation results for scenario with tip load (EBR method), the maximum error is less than 0.01 m, which is quite acceptable.



**Fig. W14-3** The simulation and experimental tip positions without tip load (EBR)



**Fig. W14-4** The simulation and experimental tip positions with tip load (EBR)



**Fig. W14-5** The screen shot of the simulation results from the paper

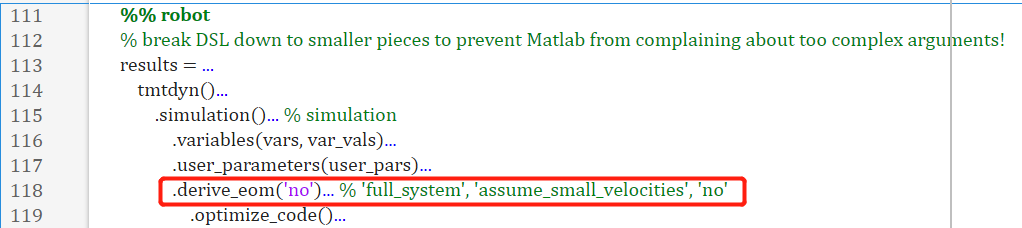
Compared with the simulation results in the paper, mine is basically the same, which proved that I was using the package correctly and I think I can move forward to think more about how to set up the simulation environment for our project, a free falling SRA with the base fixed on the wall.

1. *Self-Exploration Section*

Since the lecture video post on Canvas about this package on Youtube is not sufficient for us to make a thorough knowledge about all the functions inside, the best way to learn about it is to play with it first by adjusting some parameters. Initially I started from the example simulation provided by the author of the package, which is called “exp2\_EBR.m”, the geometry factors of the SRA were also mentioned in the three papers above, to test the applicability of the package, my simple test plan is shown as follows.

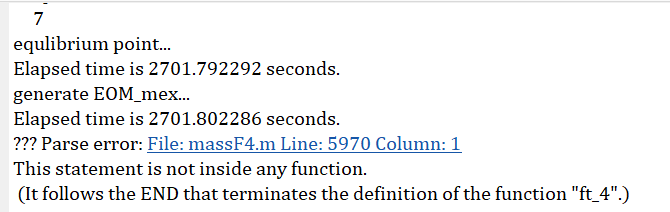
(1). Change the number of links

The number of SRA links was initially set to be 3, to reduplicate the simulation in the paper, I changed it to 4. The equation(s) of motion (EOM) will be derived and over-written in the file “eom” for each time we run the simulation if we choose to derive the “full system” EOM in the “robot” section of exp2\_EBR.m (around line 118) which is shown as follows. For newly system with 4 links,



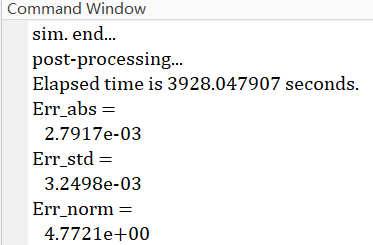
**Fig. W14-6** The switch mode for deriving the EOM of the system

we have to switch it to “full system” first and then I found that such error in the figure below emerged for 12 times for the whole process, the problem is that the EOM was derived in the massFn.m (where n is an integer starting from 1, the maximum of n depends on the number of links and complexity of the system) file using Symbolic Math Toolbox, which is exactly the same method as Chase, however, with the increasing number of links (or complexity of the system), the redundant “end” in the function of massFn.m file will be increasing simultaneously, for number of links less than 3, there will not be any issues. I haven’t found the reason behind or any indications from the author so far. The only thing I could do is to delete the additional “end” when the simulation was stuck by such error. I think I need to contact the author for more detailed information.



**Fig. W14-7** The frequent error emerged in the “mass” file for deriving the EOM of the system

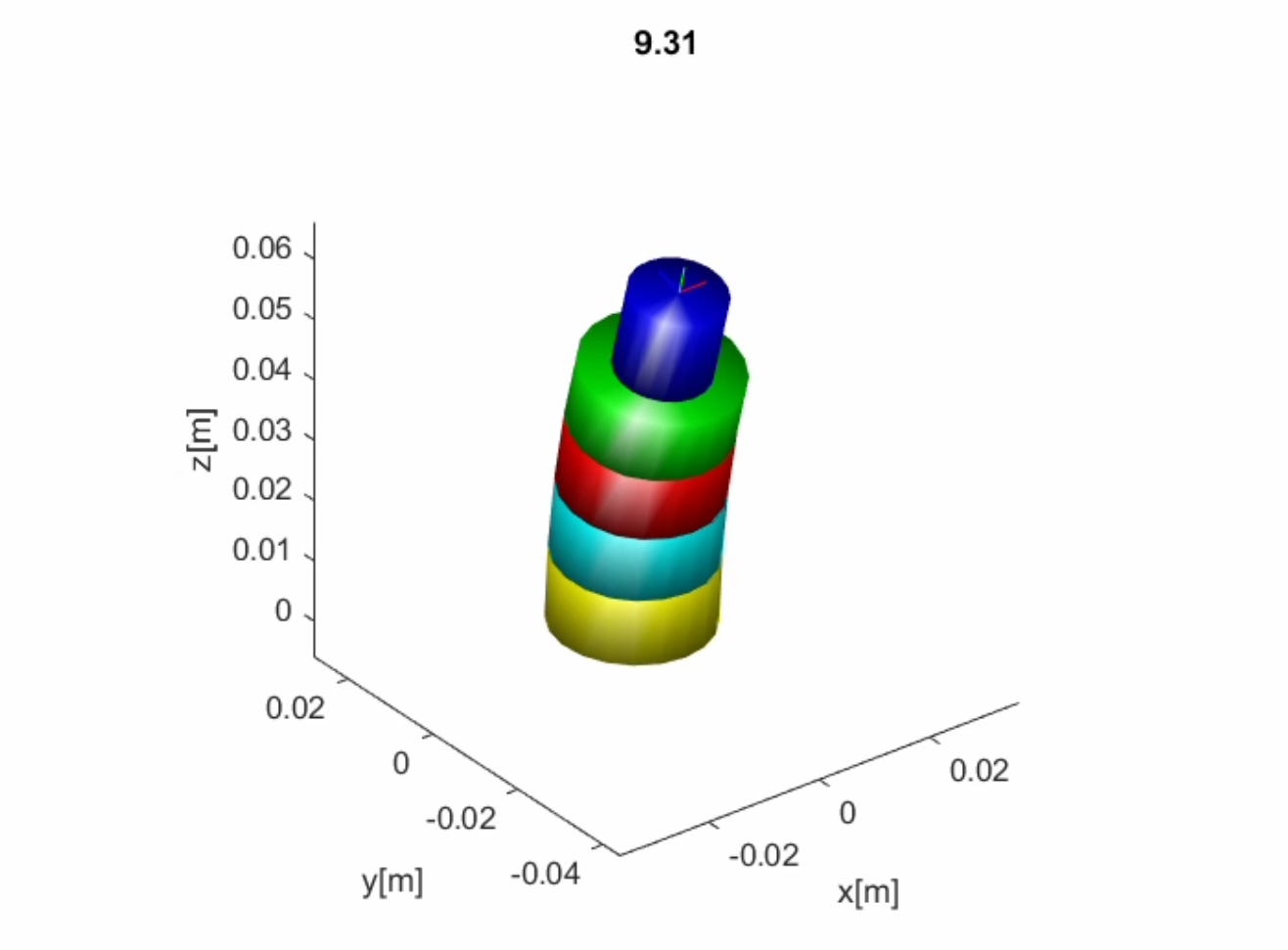
After deleting all the redundant “end”s, the simulation can continue smoothly and the animation looked pretty cool. Also I compared my results with those in the paper, they are basically the same (more explicit contents have already been covered in the previous section, Simulation Part). The overall simulation time is about 1 hr 5 mins, which is shown below.



**Fig. W14-8** The simulation time for a 4-link SRA modelling with EBR method

(2). Change the geometry factor of SRA

To test more about the applicability of the package, I decided to adjust the length of the SRA from to (still with 4 links), the simulation still could proceed smoothly by eliminating the “end”s, but the animation seemed odd, which looks like antenna rotating around its axle rather than what it is supposed to be normally (example shown below), and the magnitude of the displacement (or deformation) along no matter what axle is in , which is apparently wrong. And in the lecture video, the author did another experiment by doubling the input force/torque, he obtained the similar animation, thereby the package has some limitations, we



**Fig. W14-9** Example animation of a normal 4-link SRA modelling with EBR

have to recognize them autonomously by setting the parameters within a rational range, or by increasing the number of links to acquire the accuracy of the simulation when necessary.

1. *Location of input*

To trigger and accomplish the simulation process, there must be some specific input and some raw data to make a comparison between the simulation and physical experimental results. And to confirm where we can add the input and the specific format of input are vital to us as well, since we have to perform our unique simulation with this package, which is a relatively mature one we can rely on.

Take the exp2\_EBR.m file as an example, the simulation will start through another external function which is called sample\_exp\_data in the exp.readings section of the main file (about line 70). There are three types of modes for the input, from case 1 to 3, which were defined by the formation of the actuator and external forces (e.g. direct force applied, pressure or torque). When selecting the mode, the data will be imported from mat file from exp folder, and the title of each column data for each mode is shown as follows,

|  |  |  |
| --- | --- | --- |
| **Mode 1** | **Mode 2** | **Mode 3** |
| Time |  | Time |
| Pressure 0 |  | Pressure 0 |
| Pressure 1 |  | Pressure 1 |
| Pressure 2 |  | Pressure 2 |
| X\_value |  | X\_value |
| Y\_value |  | Y\_value |
| Z\_value | Time | Z\_value |
| Q0\_value | Pressure 0 | Q0\_value |
| Qx\_value | Pressure 1 | Qx\_value |
| Qy\_value | Pressure 2 | Qy\_value |
| Qz\_value | X\_value | Qz\_value |
| X1\_value | Y\_value | X1\_value |
| Y1\_value | Z\_value | Y1\_value |
| Z1\_value | Q0\_value | Z1\_value |
| Q01\_value | Qx\_value | Q01\_value |
| Qx1\_value | Qy\_value | Qx1\_value |
| Qy1\_value | Qz\_value | Qy1\_value |
| Qz1\_value | X1\_value | Qz1\_value |
|  | Y1\_value |  |
|  | Z1\_value |  |
|  | Q01\_value |  |
|  | Qx1\_value |  |
|  | Qy1\_value |  |
|  | Qz1\_value |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

**Table W14-2.** Parameters defined in each mode of input

Contemporarily, the force and torque can be confirmed to be exerted on the tip (end-effector, which is blue in Fig. W14-4) of the SRA, Q and Q1 values need to be further confirmed.

## **Plan**

1. Since I have found where we can amend the input, next week I will set same unique inputs like sinusoid or step function to test the package; besides, in the data file, there must be some experimental measurements, we don’t need them in our simulation, I will try to eliminate them by controlled trial method;
2. As I measured quite a long time ago, the length of each segment of our SRA backpack is about 20 cm, in the Self Exploration Section, the package did not perform so well for very long soft robot, I decide to insert some real geometry features of our SRA to how many links we need to set for an relatively accurate simulation result, also make a balance between the computation cost and accuracy.

## **Reference**

[1] Sadati, S.M.H. *et al.* (2020) “*tmtdyn*: A Matlab package for modeling and control of Hybrid Rigid–continuum robots based on discretized lumped systems and reduced-order models,” *The International Journal of Robotics Research*, 40(1), pp. 296–347. Available at: <https://doi.org/10.1177/0278364919881685>.

[2] Sadati, S.M.H., Zschaler, S. and Bergeles, C. (2019) “A MATLAB-internal DSL for modelling hybrid rigid-continuum robots with tmtdyn,” *2019 ACM/IEEE 22nd International Conference on Model Driven Engineering Languages and Systems Companion (MODELS-C)* [Preprint]. Available at: <https://doi.org/10.1109/models-c.2019.00086>.

[3] Sadati, S. *et al.* (2019) “Reduced order vs. discretized lumped system models with absolute and relative states for continuum manipulators,” *Robotics: Science and Systems XV* [Preprint]. Available at: <https://doi.org/10.15607/rss.2019.xv.076>.