**Weekly Report – W6 Spring 2023**

Task & Problem

1. Reidentify some of the key figures in the experiment plan, e.g. the time for repeated experiments and how much extra weight to mount on the mannequin;
2. Coding work regarding SRA dynamics in MATLAB, communication between MATLAB and Arduino, realizing control in Arduino;
3. Some other issues need to report.

Solution

1. Reidentify some key figures for the experiment plan

To make our experimental process and results more convincing, we need to make sure all the settings in our experiment are well referenced, for example, for how many times we should do repeated experiments, too small or large will not be a good idea, one or two times experiments would be affected by occurrence and too many times will cost a lot of time, which is not worth to do that. So maybe we can look for some related papers to find answers.

(1). The proper repeated experiment time

According to past research papers about metabolic energy cost, human walking scenarios with extra weight, the general experimental trial is 8 [1].

(2). The percentage extra weight added

About 25~50% additional weight of the human will be added to the subject in the experiments, so assume the total mass of the mannequin and backpack is 20 kg, the range of extra weight will be from 5 kg to 10 kg, since we have already bought 4 sand bags with 10 lbs max for each one, that’s enough for our experiments.

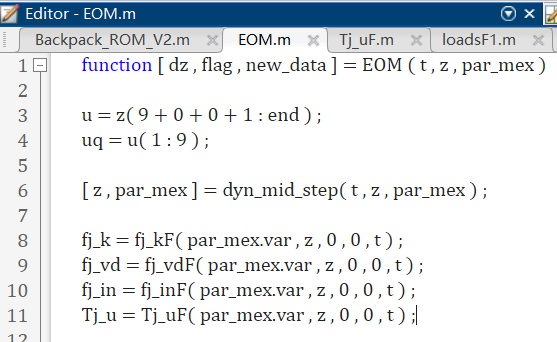
1. Coding work

(1). SRA dynamics

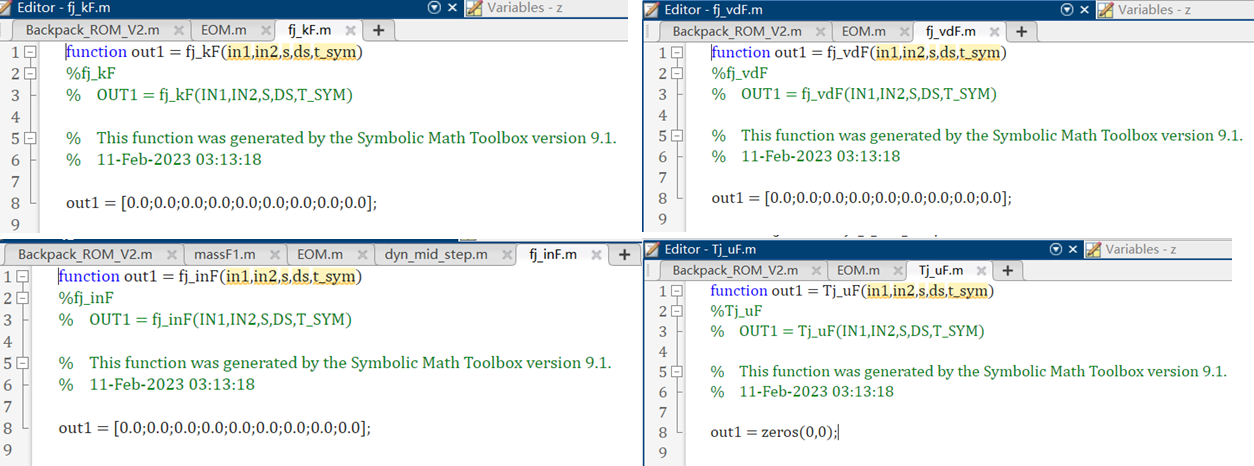
1). Simplification process

When modifying the MATLAB simulation code into online control code, one of the key issues is that we have to avoid too many times function called, to increase the communication efficiency between MATLAB and Arduino, we have to simplify the functions as much as possible.

Because I have assigned different jobs for myself and Charanjit, I will be responsible for the dynamics part, the EOMs for ROM method contain the following matrices, the inertia matrix (*massF1.m* and *massF2.m*), spring & damper matrix (*sprdmpF10.m*), the external load matrix (*loadsF1.m*), eventually we will obtain the numeric matrix for second order term (acceleration, ), then it will be brought into ode solvers to compute the states and using numerical integration. The second order terms are all derived in *EOM.m*, in which some of the matrices seem to be initialization process which are shown as follows,



**Fig. W6-1** The “initialization process” related functions



**Fig. W6-2** The practical outputs of the “initialization” functions

We can see from Fig. W6-2 that all the outputs are zero vectors or matrices and all of them do not change during the simulation process after they have been derived, so we can redefine them in EOM.m with specific values to decrease the utilization rate of the functions.

2). Coupling process

The coupling process for the dynamics of arms can be complex due to the function nests, besides the basic EOMs, we have to pick out all the related the functions, the main challenges are shown as follows:

* The values stored in the struct cannot be passed to the EOMs, the temporary solution is to set up a global variable, but this feature will not exist in the coming versions of MATLAB, we still have to face this issue in the future;
* The function nest makes it hard to convert the code into C/C++ in the future, before I figured out the global variable solution, I was trying to put all the functions into Simulink block which was enlightened by Chase’s model, but it failed because some of functions were used to avoid singularity issues that have been used for twice not in a for loop, which made it hard to realize in Simulink;
* The resultant states of the arm were different from the original simulation results with exactly the same inputs, I still have to check the reason behind.

1. Some other issues

(1). After pressurizing only one channel of the left side of the backpack, there was one still deforming as an “S” shape, I was thinking about to add some control strategies to avoid this issue to let it be a beautiful curve in the swing stage by pressurizing the same amount of input in the channel beside it to cancel out the “error”.

(2). The swing lateral distance and height were not so promising after test, given 48 psi, the lateral distance was about 25 cm, and the height was around 15 cm compared with the equilibrium state, the thing is the threshold pressure input is 65 psi, obviously the pressure input for “pull back” stage will be much larger, which made me doubt about the ability for the arm to help mannequin regain balance with extra weight, by increasing pressure input above 65 psi can be a risk.

Reference

[1] Grabowski, A., Farley, C.T. and Kram, R., 2005. Independent metabolic costs of supporting body weight and accelerating body mass during walking. *Journal of applied physiology*, *98*(2), pp.579-583.