Assignment 3 - System Identification

Dr. Joseph Conroy (conroyj@umd.edu)
Dr. Krishna Kidambi (kidambi@umd.edu)

5 points
Due: 23 November 2021

1 Overview

This assignment is designed to introduce you to basic single-input-single-output (SISO) system identification and tools available for implementation thereof. You will estimate transfer functions from input to output independently for roll input to roll angle, ϕ, and lateral velocity v, pitch input to pitch angle, θ, and longitudinal velocity, u, yaw input to yaw rate, ψ, and heave input to heave velocity, w. For this assignment, each TEAM may turn in one assignment with everyone's name on it. You should utilize the MATLAB 'systemIdentification' command/toolbox to aid in the evaluation.

1.1 Download Data Bags

Bags are a few GB in size each and can be downloaded from here:

https://drive.google.com/drive/folders/1dqb_XB3qfs0xn6TimuqMyCvFqnDQOyHW?usp=sharing

20211012_orientationctrl_test.bag is the bag you can use for lateral system identification
20211028_sysid_longitudinal.bag is the bag for longitudinal system identification
20211028_sysid_yaw_heave.bag has yaw motion in the first half and heave excitation in the second half

1.2 Coordinate Systems

Note that the onboard vehicle topics utilize the standard ROS coordinate system of FLU (forward, left, up) for the body-fixed axes and ENU (east, north, up) for the local inertial frame. The vicon coordinate systems were set such that the coordinate system in the body frame is (X=right, Y=forward, Z=up) and the inertial frame was (X=towards the wall mounted TV, Y towards the window outside in the Brin lab, and Z up). The vicon body coordinate system was co-aligned with the vicon local inertial system at the start of each data bag collection. You will find that the MAVROS on-board data and vicon data are 90 degrees off from each other in yaw.

1.3 Relevant Topics

In this case, the onboard generated data is sufficient to use for system identification, however you are welcome to leverage the Vicon-data as well. Relevant onboard data for actual vehicle orientation is '/mavros/imu/data'. Onboard estimated velocity is '/mavros/local_position/velocity_body'. Inputs will be '/mavros/rc/in'. Vicon data, as usual, is '/vicon/m500_joec/m500_joec'. For this assignment, due to how the 'systemIdentification' toolbox works, you'll need to utilize the 'interp1' function in MATLAB to resample one of the signals. As mentioned before, you should generally resample the lower/slower sampled data into the higher sampled data as this does not result in loss of signal fidelity. In the "inputs" topic, there are 4 controls you need to be aware of: roll, pitch, throttle, and yaw. The inputs are in this order: 1.) throttle 2.) roll 3.) pitch 4.) yaw.

2 Perform Identification

In the system identification toolbox GUI, begin by "importing" the time series data from the workspace. This is easy if you resampled your data already. When you import, Input and Output are vectors (or matrices in more advanced cases). Start Time can be left as 1 or changed to 0, it doesn't matter much. Sample time needs to be the '*dt*' of your signals and this depends on what you resampled to.

The “Working Data” and “Validation Data” boxes are self explanatory and you can click and drag data into these boxes. To start, you should clip your data if needed; this will be highly advised for the heave/yaw dataset to separate them out, but is optional for the lateral and longitudinal sets. Then it is highly recommended that you remove the means of the signals under the “pre-processing” operations drop-down. If you resampled ahead of time, you won’t need to do anything with resampling, merging, and transforming. Then simply go to “estimate”. It’s advised that you take time to examine each option, but the three you will use are: “Transfer Function Model”, “Process Model”, and ”Spectral Model”. For the first two options, begin with an educated guess regarding the number of poles and zeros. Try to find the best fit you can. For the “Spectral Model”, use the “spafdr” option, ’linear’ spacing, ’100 -1000’ frequencies, and “Default resolution”. The “Transfer function” and “Process Models” closely align to the prediction error minimization strategy discussed in class and will result in ‘s’ domain transfer functions. The “Spectral Model” is a method of numerically obtaining the frequency response of the transfer function estimate, related loosely to the idea of estimating $Y(\omega)/U(\omega)$.

MATLAB has a few nice tutorial videos here as well: https://www.mathworks.com/products/sysid.html

3 Plot Results

Professionally document and provide plots showing the inputs, outputs, and model estimated outputs for the “Process Model” and “Transfer Function” models. For these, you’ll note that clicking the “Model Output” checkbox in the system identification GUI provides a comparison plot with a metric of how well the model fits the data (higher is better). For the “Spectral Model”, simply plot the resulting $\hat{G}(\omega)$ estimate; this is seen when checking the “Frequency Response” checkbox.