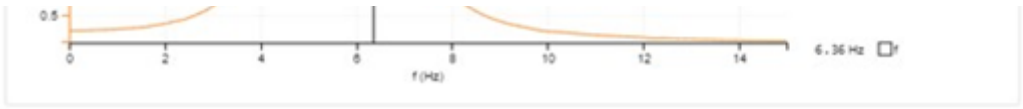

For all the channel names in the plot the format is outputChannel inputExcitation so gVert Heave is the frequency response of gVert to a heave excitation. These plots show typical features of F1 car ride behaviour: the heave mode at about 4Hz, the pitch mode at about 8Hz and some complicated modes associated with the hubs. Users wishing to generate similar plots can do so using the MATLAB function ‘tfestimate’.

For the *Standard Sweep* we specify:

1. vCar the speed of the car, which determines: (a) The speed of the airflow over the car. (b) How hard the bump and kerb are hit. (c) How long it takes to traverse the length of the bump and kerb.
2. zAmplitudeHeave the amplitude of the chirp signal used for the heave sweep.
3. zAmplitudePitch the amplitude of the chirp signal used for the pitch sweep.
4. zBump the height of the bump.
5. zKerb the height of the kerb. Note that if any of these amplitudes are set particularly high the simulation is likely to terminate early as the car is launched skywards.

3.1 Time History Correlation vs. Frequency Domain Correlation

It is rare verging on unheard of to achieve good time domain correlation between a simulation and the rig. This is because as a dynamic process the effect of any poor correlation will have a wide influence. I therefore recommend that the bulk of the correlation work is done in the frequency domain.

4 User Defined Sweeps

It is possible to define more detailed sweeps with the following information:

1. A lookup of wheel pan displacement versus time for each of the four corners.

And optionally:

2. Any number of external loads applied to
    - Any of the 4 hubs.
    - Chassis Front, Centre & Rear. In effect making it a Virtual N Post rig. For these external loads we input a lookup of force versus time. The connection position and load vector ([0 0 1] would be vertically down e.g. for aero downforce) need to be specified.

There are a number of self-explanatory options for example specifying the location of accelerometers.

5 The Future

As it stands Virtual 4 Post is a useful tool for exploring ride behaviour. Frequency sweeps such as these are an important part of searching for ride performance. However, there are some more advanced games we can play using track bump profiles. A future option to be included in V4P will work something like this:

1. Run a Dynamic Lap to generate a vCar profile and a set of tyre forces which ensure sensible car behaviour.
2. Push the car around the track using the precomputed tyre forces and along the precomputed vCar profile, but allowing the vertical dynamics to evolve freely over a user provided bump profile.
3. Use the lap time sensitivities computed in Dynamic Lap (SLS) to compute a ride score for that particular lap. This will then enable users to trade ride metrics against each other, for a specific setup and at a specific track. Where, you might ask, is the link to lap time? Well the short answer is that there isn’t one, and this is an area that will have be remain within the expert judgement of the race engineers. Over the years I’ve seen many simulations and methods which attempt to link ride performance directly to lap time, but have never seen a single one that has been validated with anything like the rigour required if we are to go from the broad range of good to optimal: a leap of a few hundredths of a second lap time. We therefore strongly favour giving race engineers superior tools to use their judgement.