Guidelines for Using PPLANE

1. Download the contents of the folder "PhasePortraitCode" from coursework. It is located under Materials on the Coursework site. Be sure to keep these files together in one folder.
2. You will note that the folder contains several files. These simply contain the dynamics for a bicycle model with small angle assumptions and nonlinear tires (using the Fiala model from class) but with minor modifications to interface to PPLANE. You will be working directly with pplane8.m and GeneratePhasePortrait.m.
3. GeneratePhasePortrait.m is a function that passes computations of the state derivatives of the bike model to PPLANE with either linear or Fiala (nonlinear) tires. The syntax for this function is as follows:

GeneratePhasePortrait(uy, r, mode, desiredStateDerivative)

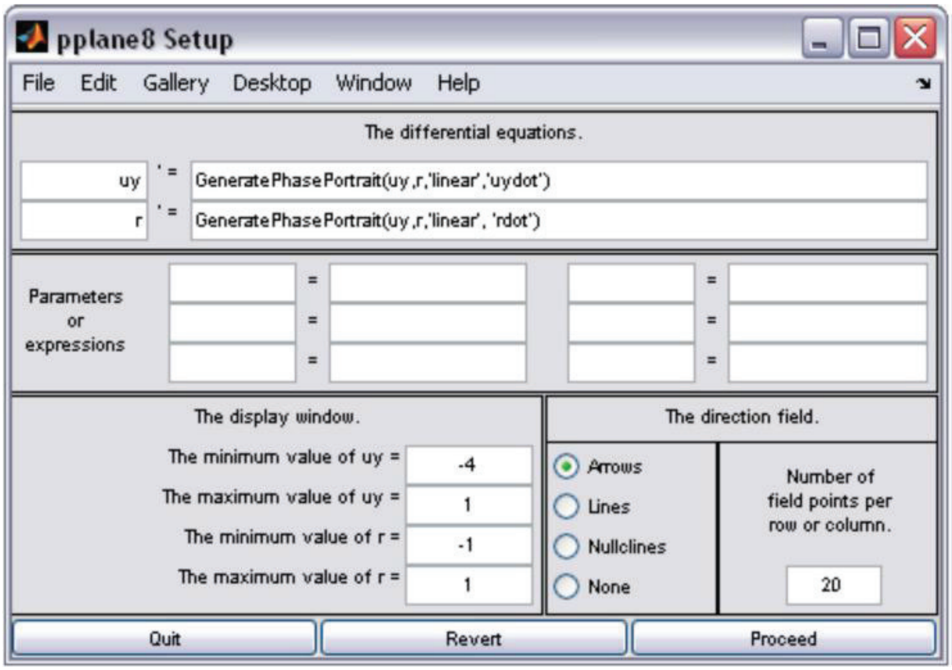
Of the inputs to the function, you will only be changing the last two:

mode – This input specifies whether a linear tire model or Fiala tire model will be used to compute lateral forces. Setting the mode to 'linear' will tell the function to use linear tires, while setting it to 'nonlinear' will tell it to use Fiala tires.

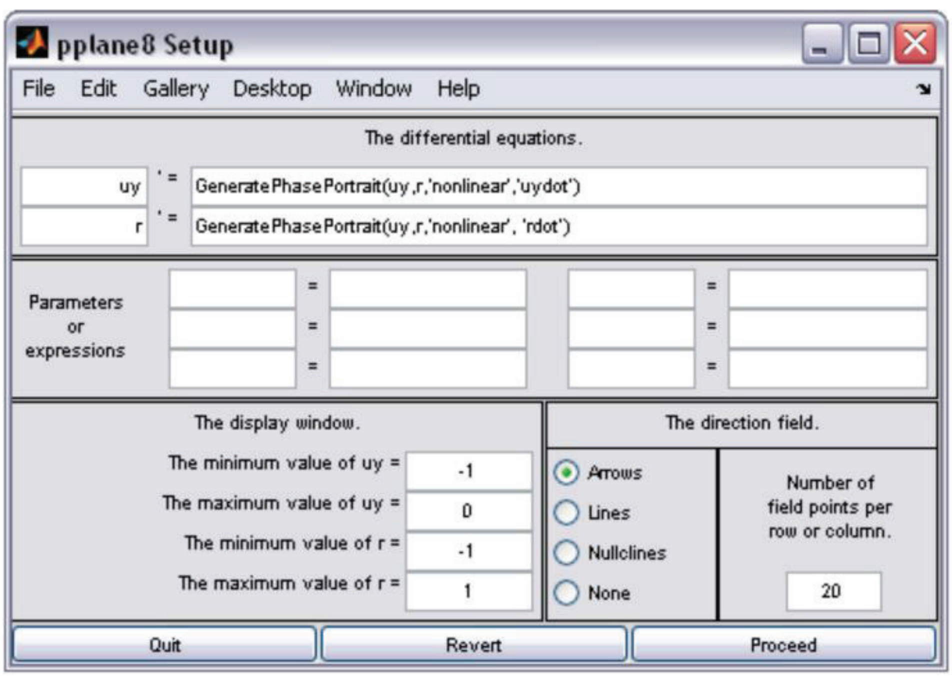
desiredStateDerivative – This input tells GeneratePhasePortrait which state derivative to pass to PPLANE. 'uydot' will pass the time derivative of the lateral velocity to PPLANE, while 'rdot' will pass the time derivative of yaw rate to PPLANE.

As provided, the function contains all required parameters for the P1 vehicle and is set up to compute phase portraits for a fixed steer angle δ = -10 deg and longitudinal velocity Ux =8 m/s. You are free to change these parameters to explore their effects on phase portraits, but you do not need to do so for the assignment.

4. In order to start PPLANE, simply enter "pplane8" into the MATLAB command window, and the "PPLANE setup" window will open.
5. In this window, we will be concerned with the fields labeled "The differential equations." and "The display window." The former will be used to enter the governing equations for our model of interest, while the latter sets the plotting limits for the phase portrait.
6. In the "The differential equations." field, we will use GeneratePhasePortrait with the appropriate combination of inputs to compute state derivatives for PPLANE. In order to compute state derivatives for a linear bike model and generate a phase portrait with Uy on the x-axis and r on the y-axis, we will set up the field as follows:



Similarly for a bike model with Fiala tires:



Note that the third input to GeneratePhasePortrait has been changed to ‘nonlinear’.

7. Change the plotting limits in “The display window.” field according to preference or instructions in the problem set.
8. Click “Proceed”. This will call up the “pplane 8 Display” window. Shortly thereafter, the flow field for the state derivatives will be plotted.
9. Click anywhere within the flow field to plot a state trajectory. PPLANE does this by simulating the bike model with the code in GeneratePhasePortrait.m forwards and backwards in time.
10. Once you have plotted enough state trajectories that you think you can see all equilibria in the phase portrait, you can use PPLANE to find the equilibria exactly and analyze their stability characteristics. In the “pplane8 Display” window, open the “Solutions” pull-down menu and select “Find an equilibrium point.” This will turn your mouse pointer into a set of crosshairs. If you click sufficiently close to the desired equilibrium, PPLANE will find the equilibrium, mark it with a red dot, and then open up a window entitled “pplane8 Equilbrium point data” containing the location of the equilibrium point, its type, and stability characteristics (eigenvalues and eigenvectors).
11. Once you are done with a particular phase portrait and would like to plot another, go to “File →Restart pplane 8” in the “pplane Display” window. This will close this window, at which point you can return to the “pplane8 Setup” and make changes as desired.

Have fun!

Compatibility notes:

pplane8.m and the code for this assignment have been tested successfully with MATLAB Releases 2010b and 2011a in OS X and Releases 2008b, 2009b and 2011a in Windows. If you are using an older version of MATLAB and encounter issues with pplane8.m, you are free to try the older versions available at http://math.rice.edu/~dfield/index.html

If you are having troubles making pplane work on your own computer, this assignment has been tested successfully on the computers at the Terman Engineering Library in the Huang Engineering Center.