

Plotting Data using Octave

Auto-generated Octave script

The UDK script will generate an Octave script and dump it in the UDKGame/Logs folder. To do this you must customise the following functions in the UDK script where you decide which variables you want to export to Octave.

Let's take the example of MAS14P_SkyMaster.

(1) Identify the variables you wish to log in the **function setLogFileColumnLabels()** Here's my choices, I want to be able to plot theta, omega and drive as a function of time.

```
function setLogFileColumnLabels() {  
  
    columnLabels.length = 0; //empty array.  
    columnLabels[0] = "time";  
    columnLabels[1] = "theta";  
    columnLabels[2] = "omega";  
    columnLabels[3] = "drive";  
  
    writeMatlabFileHeaderNew(columnLabels);  
  
}
```

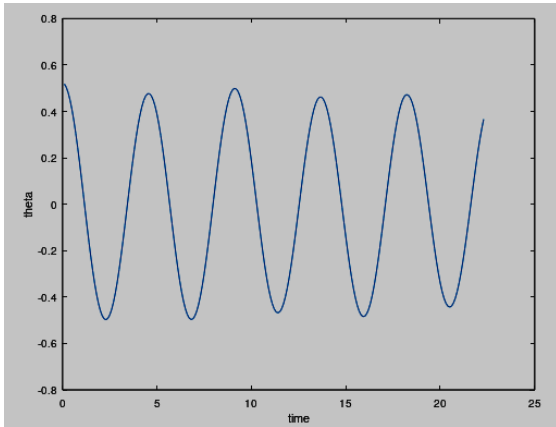
(2) Then you should write these variables into the dataArray in the function **logDataRecord()** like this

```
function logDataRecord() {  
  
    local array<float> dataArray;  
  
    dataArray.length = 0; //empty array.  
    dataArray[0] = time;  
    dataArray[1] = theta;  
    dataArray[2] = omega;  
    dataArray[3] = drive;  
  
    writeMatlabFileRecordNew(dataArray);  
  
}
```

All of these variables will be exported to the Octave script, but only two will be plotted automatically. The independent variable (horizontal axis) will be dataArray[0] and the dependent variable (vertical axis) will be dataArray[1]. How to plot other dependent variables will be explained below.

(3) Data is collected as the simulation runs, to dump the data to file **you must press F12**. The name of the script is what you specify in the actor's editable variables, I chose SM for the Skymaster. When you run SM_3 (third simulation) you will get the following output (below). First you get the plot of variables explained, also on the console you get the names of variables you can plot

(4) So if you want to see how omega varies with time then you type **plot(time,omega);** at the Octave command prompt.



```
>> SM_1
>>>> File auto-generated by UDK magic
>>>> Variables available for plot:
      time
      theta
      omega
      drive
>> |
```

More Complex Plots

You can open and modify the auto-generated .m file to get some more advanced plots.

(1) To get several plots on the same axes you would type

```
plot(time, variable1, time, variable2);
```

and so on.

(2) If you want several plots on the same figure then you would do this

```
subplot(3,1,1);           (nrRows,nrCols,plotNr);
plot(time,vari1);
subplot(3,1,2);
plot(time,vari2);
subplot(3,1,3);
```

(3) If you want several figures then you would do this

```
figure(1);
plot(time,vari1);
figure(2);
plot(time,vari2);
```

(4) You might want to rescale the axes of a plot. You can do this interactively on the command line. Let's say you have a plot which goes from 0 to 100 seconds, and the data is between -1 and +1. To see the last 10 seconds of the data you would type

```
axes([90,100,-1.2,1.2]);
```

(5) You might want to overlay two plots on the same axes, e.g., from two simulation runs. You can 'hold' a plot like this

```
plot(time,vari1);
hold on;
plot(time,vari2);
hold off;
```