DATAPLOTTER USAGE

This file is used for plotting any 2D data for a given plot-able data.

It subscribes to any data that has to be plotted and then either writes it to a text or CSV file, or plots the data on a graph, or both depending on the configuration.

Steps to operate it:

1) Below "//Create subscribers", Add the subscriber to the topic that you want to plot.

1.1) Subscriber is written like this:

ros::Subscriber "OBJECT\_NAME" = n.subscribe("TOPIC\_NAME", QUEUE\_SIZE, POINTER\_TO\_CALLBACK\_FUNCTION);

-> all the capitalized things in above command has to be replaced by what they signify.

TOPIC\_NAME: Topic that has to be subscribed.

QUEUE\_SIZE: (an integer value), in case we are not able to process messages fast enough, this number of messages will be stored.

POINTER\_TO\_CALLBACK\_FUNCTION: callback function that will get called when a new message has arrived on the subscribed topic.

1.2) Whenever a new message arrives, the CALLBACK\_FUNCTION gets called, inside that function take the values/data (that we are interested in) into some global variable which can be easily written to a file or plotted on a graph.

2) Below "//Optionally add named labels", Add the name of data or label that you want to be displayed over the graph.

e.g.,

labels.push\_back(std::string("LABEL"));

->change the capital "LABEL" by appropriate label name.

3) Below "//Open/Create the file to be written to", there is a command:

myfile.open("FILENAME.txt", std::ios::out);

This is used for opening/creating(in case file doesn't exist) the file in which data has to be written.

-> change the capital "FILENAME" by appropriate name for the file in which you want to write the data.

If multiple topics has to be written, then depending on whether we want to write it to same file or different files, it can be modified.

4) Inside

"while( !pangolin::ShouldQuit() && ros::ok() )"

4.1) "myfile << l\_tick\_count << "," << VARIABLE\_NAME << std::endl;"

This is used for writing to file, change the VARIABLE\_NAME by appropriate variable that you want to write to the file.

4.2) "log.Log(VARIABLE\_NAME1, VARIABLE\_NAME2);"

This is used for plotting, change the VARIABLE\_NAME by appropriate variable that you want to plot.

Running it:

1. go inside your catkin workspace:

$ cd ~/catkin\_ws

1. build the files:

$ catkin\_make

3) run it using:

$ rosrun /PACKAGE\_NAME/dataplotter