**同济电车队GUI试车数据分析工具**

**内部开发文档**

Internal Development Documentation of DIAN Racing GUI Data Analysis Tool

开发人：**2018/9/5： Bruno Ma**

**2020/1/31:电控组\_叶哲惟**

# 概述

## 项目背景

## 用于以可视化方式分析试车数据，以便更快捷地分析数据。

## 项目开发日程

## 2018/9/5 工具制作by Bruno Ma

## 2020/1/31更新了小工具：by matlab 2018b

## 1，新增三个默认显示窗口，一打开文件直接显示diagno，actualV，actualT

## 2，检索diagno发生的时间，在所有图中标注出发生时图的值

## 3，plot重复画图时右上角有对应线颜色的标注（原来就有，de掉了不兼容新版本的bug）

## 4，starttime和endtime可以输入default返回初始值（0或者end）

## 5，显示数据的时间长度

## 6，检测到diagno有错误值时地下Error红色亮起

## 书写了开发文档算吗x

# 设计说明

## C:\Users\Flanker\AppData\Local\Temp\WeChat Files\15a4fa700214af5deb38ce1bb549052.png

7.

8.

5.

6. 

4.

1.

3.

2.

## 设计界面：

## 选择文件，将文件载入workspace，并将路径打印在2.白框中，同时查找6.中默认图像的数据源，对其进行分析，在8. 中显示数据持续时间，数据中是否有错误；

## 用于显示选择文件的路径

## 下拉列表，选择mat文件中记录的数据，选中后可以在4.中选择上图或下图的plot打印图像至7.中上图或下图

## 打印列表，用于记录在7.图中打印过图像的名称

## 起始时间和结束时间限定，在给出想要让6.、7.展示的起始时间和截止时间后，让所有图像展示这一时间段内的数据

## 默认图像，在选择玩文件后查找并展示文件中：实际转速、错误码、实际扭矩；并查找错误码出现/改变的转移时间点，并把时间点以及对应时间点数据的值标注在所有图像上；

## 显示3.、4.选择的图像，并且可以叠加。叠加图像由legend进行标注。

## 显示数据持续时间，帮助5.的选择；下框显示6.中是否有出现错误，若无错误则不会显示

## 接下来展示代码细节，并说明编写逻辑及原因

## 首先要知道guide的代码运作的机理：gui有两种存在方式：1是单纯靠.m文件，2也就是本工具使用的是.fig以及.m文件保存，如下图。

## 

## 运行时用.m文件运行。

## 当要进行编辑时，在command window输入guide，开始编辑；

## 通过弹出窗口选择要编辑的文件；Open，进入显示界面的编辑状态。在这个状态可以选择组件，对组件的属性进行编辑（如名称，大小，位置等）

## 

## 各种组件在此不进行介绍。

## 此处介绍两个概念：

## 句柄：句柄是一种结构，类似于c中的指针，可以存储各种形式的其他结构，且具有全局性。整个gui都由句柄“handles”进行调用，每个组件都对应一个句柄分支，如“handles.Plot1”指代plot1图像；句柄还可以作为组件函数间传值介质：新建一个handles分支（不需要声明），并对其进行赋值，保存操作，在其他组件函数下就可以对其进行调用。具体可参考<https://blog.csdn.net/xrdsjb001/article/details/76373984>

## Callback，Creat function的意义：Callback，回调函数，是使用最多的函数，在gui程序运行时，在对组件操作时执行。如，按钮按下执行；Creat function是在gui开始运行时执行的（不过本编辑者没有编写过）；

## 若要对组件进行功能编程，可以右键选择view Callbacks->Callback等function，跳转到m文件编辑器进行编辑

## 

## 接下来对各个组件的代码逻辑进行分析：

## FileSelection选择文件，是最长的一段代码：功能有将文件载入workspace，并将路径打印在2.白框中，同时查找6.中默认图像的数据源，对其进行分析并展示文件中：包括实际转速、错误码、实际扭矩；并查找错误码出现/改变的转移时间点，并把时间点以及对应时间点数据的值标注在所有图像上，在8. 中显示数据持续时间，数据中是否有错误；

## 以下是通过调用ui选择文件的代码，并将文件的位置打印到2.FileDirectory中；

## 

## 在把mat文件中的数据导入工作区后，进行数据的打印，把所有数据类的名称查找出来，然后在下拉列表ParaList里呈现，方便查找

## 

## 这里使用了“eval（）”的语句，这是一种方便反复执行稍有变化的命令的一种简化命令，具体见：<https://blog.csdn.net/qq_38412868/article/details/79185883> 因为ParaName本身是以str形式存储，所以不加单引号

## 

## 有时命令中也有str，这时候需要用双重单引号进行表示，可以根据字的颜色判断是否书写有误，如果识别错误，可以用空格隔开以方便编辑器识别

## 接下来，在遍历的过程中找到6.对应默认图表想要显示的几个图象数据。这里并未按照上下顺序查找：因为所有的数据都要根据“Diagnostic”进行标注，所以先查找“Diag”以及“ErrorIn”（错误码的形式是“Diag-ErrIn”：如“3587-6”，所以一起两者在同一个循环里查找）。

## 

## 

## 

## 查找到Diag之后，为了能把图像打印在想要的位置，要根据句柄把打印位置确定

## 

## 此外，其中的“Plot4”是可以在fig中对组件属性Property Inspector中修改“Tag”改变的。

## 

## 定位到Diag之后，就开始在其数据中遍历数据变化的拐点：

## 

## 其中为了能清晰的显示错误的位置，本编辑者使用“stem”语句对出错数据点进行了标注，

## 又使用“text”打印数值信息。其中，使用[]框起不同元素，使之在显示中换行。关于“text”的更多使用方式，见<https://ww2.mathworks.cn/help/matlab/ref/text.html?s_tid=doc_ta>

## 

## 数据的拐点信息ErrorTime以句柄的形式保存（注意要使用guidata进行保存变更，否在在其他函数下无法查找到这个句柄），此外，数据的拐点是以这个数据在数据中的位置保存的：因为CAN负载率的问题，数采记录的各个数据更新频率有不同：譬如diag信号50ms发送一次，实际转速5ms发送一次。所以每个数据数组长度是不相等但长度呈倍数关系，所以根据Diag数组的位置，以及其他数据数组长度跟它的比值，就可以定位到其他数据对应物理时间的点，如下图：

## 

## 在确定有错误之后，则将8.处“Error”改变为红色，

## 

## 同时把数据时长打印出来，以方便5.TimeRange选择数据起始时间

## 

## 要注意，因为可能会多次不同文件，所以不重复打印的图象都需要加上清理图像函数

## 

## FileDirectory：用于显示选择文件的路径，在1.中进行过讲解

## ParaList：下拉列表，选择mat文件中记录的数据，选中后可以在4.中选择上图或下图的plot打印图像至7.中上图或下图。

## 由于主要编辑在1.FileSelection中，ParaList中没有对应的函数。只有在打开gui的时候有进行初始化，但这些也可以在Property Inspector中进行修改

## 

## SelectedParaList：打印列表，用于记录在7.图中打印过图像的名称，PlotButton，用于PlotParaList选中的图像，ClearPlot，用于清空对应plot中所有的图像

## 在按下button之后，把ParaList选择的数据导入，把数据名称记录进SelectedParaList并显示出来，如下图

## 

## 然后，通过“\_”定位关键字符，把关键字符提取出来存入“Name”中，如下图

## 

## 这样就得到了代表每一条直线的字符串数组。接下来对选中的这个图像进行打印。但是这仍然有一个问题：对每个数据图像都会画出两个图像：plot（）的连续曲线，以及显示错误所在位置的stem（）函数。而负责标注的legend函数默认会把Plot中所有的线全部进行标注（即标注plot和stem）。为了只标注plot（）画的图像对象，本编辑者设置了一个临时变量p用于传递plot的图像对象，并使用cell结构p1进行装载（数组[]如果不用图像对象进行初始化不能成功装载，只会装载一个num），在legend第一个变量输入用数组[]把p1的所有内容进行重新装载，以满足legend的语法，这样可以对指定对象进行标注。具体见<https://ww2.mathworks.cn/help/matlab/ref/legend.html?searchHighlight=legend&s_tid=doc_srchtitle>中“在图例中包含部分图形对象”

## 

## 

## ClearPlot就是对Plot进行清理

## 

## 起始时间和结束时间限定，在给出想要让6.、7.展示的起始时间和截止时间后，让所有图像展示这一时间段内的数据

## 在按下“Plot in Time Range”之后，会读取“Start\_Time”和“End\_Time”两个框内的内容。此处本编辑者设置了一个新功能，即当输入为“default”字样时返回初始值（start为0，end为x（end））

## 

## 若不是default字符，则将输入转换为double，将所有Plot中的图像限定在其范围内

## 

## 默认图像，在1.中已经详细说明；

## 显示3.、4.选择的图像，并且可以叠加。叠加图像由legend进行标注，已经在2.进行说明

## 显示数据持续时间，显示数据中是否有错误，已在1.中进行说明

## Gui生成exe：这个玩玩就好了很鸡肋，需要电脑装了matlab运行库runtime才行，要1.6个g

## 有两种方法：

## 1，使用自带deploytool，见<https://www.cnblogs.com/mat-wu/p/5732008.html>，可以勾选included in pakage打包，不然没有装matlab的使用者网上下很慢；电脑如果没有runtime可能无法打包，runtime文件MCRinstaller见： https://ww2.mathworks.cn/products/compiler/matlab-runtime.html注意要安装对应版本的，且放到对应文件夹内

## 2，使用c编译器emm开启可能会快一丢丢吧但是还是跑不掉matlab.runtime运行库<https://blog.csdn.net/sinat_35026778/article/details/89889434>

# 开发回顾

## 3.1 走过的弯路

## 2018/9/5: 前辈没留下啥，只说有问题可以联系他（

## 2020/1/31: 弯路大概就是，句柄、回调函数的定义花了好久才搞明白，还有其实也不是我的弯路，就是前辈没有留下啥备注，得一条一条命令对着查。多写备注多写备注多写备注！

## 前辈有的语法后来不支持了，有的功能需要debug；

## 以及开发的时候建议新增功能可以先通过command window进行论证验实后，再加入进m文件。对于m文件的运行情况，可以在代码中加入“keyboard”，程序运行到这里时可以在这里中断，允许使用command window进行操作。这个时候你同时进入了m文件的工作空间，可以查看此时变量的情况，可以更好地了解这个时候各个变量的情况；

## 

## 有个地方没有搞得很清楚，就是handles的新分支的声明以及更改，在1.FileSelection的详细讲解中有提到过，若新建了句柄分支，注意要使用guidata（hObject,handles）进行保存变更，否在在其他函数下无法查找到这个句柄，但我不明确使用set（）的方式修改句柄，其声明应该是怎样的。

## 3.2 未来的展望

## 有个问题，没有对其他的mat文件做适配，具体适配在1. FileSelection的strcmp中改掉就行了

## 新想法：1，1.FileSelection里循环太多了，其实可以改掉；

## 可以增加一个Slider，适配时间范围选择，缩小图像以后可以用slider拖动让所有图像前后移动，看数据更方便

## Matlab的gui生成exe十分鸡肋，需要电脑装了matlab运行库runtime才行，要1.6个g，以后有代码狂魔可以用C的mfc或者py啥的重写一下[滑稽]