

下载:

Simport_V201710171854



Simport_V2017101...4.zip
57.19KB

版本变更记录:

Simport_V201709291542	BUG修复: SimportFileDBC中, 根据dbc解析CAN数据时, 对单字节内
Simport_V201710021107	修复切换模型时未更新当前模型名BUG
Simport_V201710171854	修复GUI回调函数带有原GUI名称的BUG

1. 概述

Simport是一个用于在Simulink环境下进行快捷数据导入的工具。最早是在2010年抽空做了第一个版本, 当时只基于Mathworks File Exchange中Stuart Mc Garrity提供的mdfimport工具功能做了支持INCA数据的导入, 后来随着工作经历接触到其它很多数据格式, 包括CAN记录格式, 陆续写了一些解析脚本。近几个月抽空逐步做了一些整理集成工作, 现在释放出来希望能帮到更多同道的工程师。

由于是个人兴趣和需要结合业余搞出来的小玩具而非商业化工具, 所以在功能的全面性和鲁棒性上必然有欠缺, 发现BUG请不吝反馈, 会尽量抽时间解决。

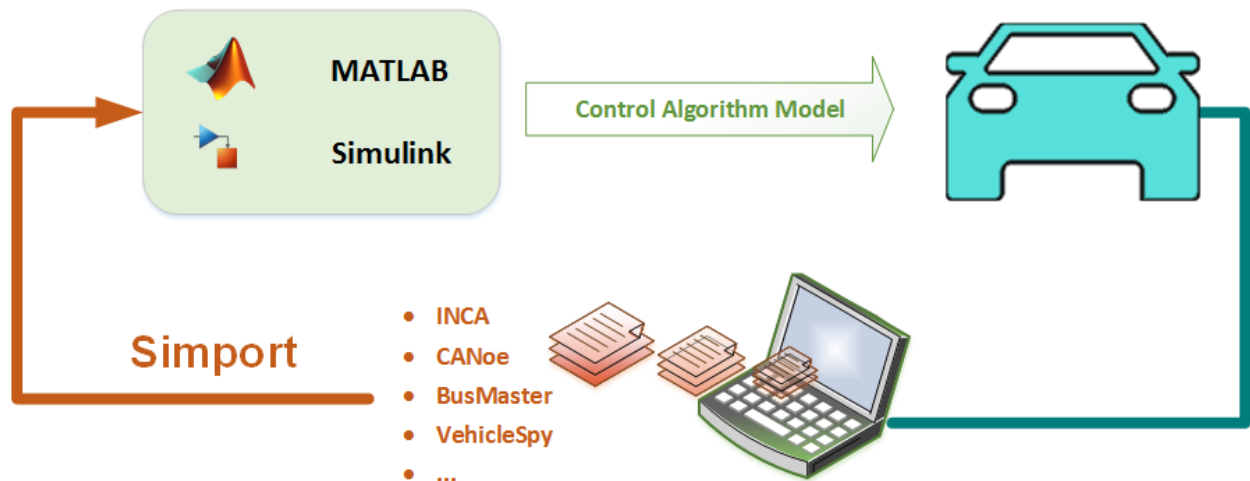
Email: jiangxinauto@163.com

时至今日在汽车电控行业内基于MATLAB/Simulink模型化开发(MBD) 大行其道, 整个算法建模、仿真、代码生成、软件集成、编译调试、标定开发工具链也非常成熟。Simulink本身就是个仿真工具, 因此天然就具有所建即所得的优点。在工程上这就带来一个可能性, 如果以后端实车采集数据反过来喂给开发端的仿真环境, 在开发和标定上形成闭环, 有很多好处, 比如,

- 在仿真环境下通过真实数据调试, 能在办公室相对高效地做一些原本需要在车上完成的调试;

- 拿实车数据做后处理，进行特定工况数据的统计分析；
- 开发一些新算法时，如果用之前的真实数据做为输入，比人造一些规整的信号变化更具鲁棒性。

Simport 目的是基于上述设想，在常见的数据记录格式文件与Simulink之间搭个桥，能够比较方便的把它们导入Simulink环境，如下图所示。



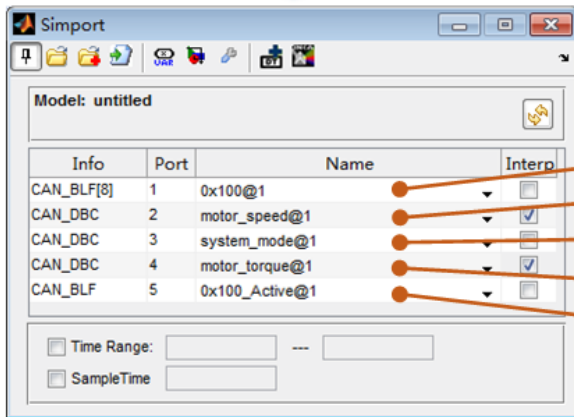
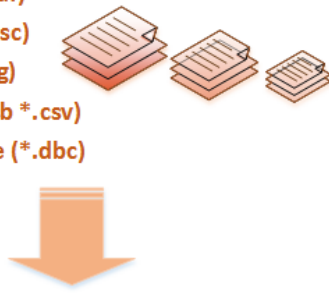
2. 原理说明

2.1. 基本流程

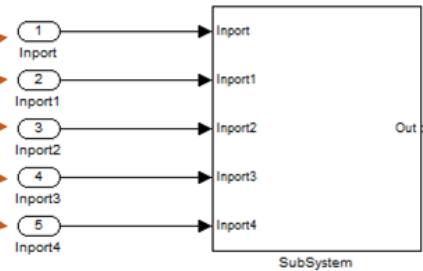
Simport利用了Simulink的Root System下可以接受外部数据输入的特性，对数据文件进行解析后，进行数据统一化处理（时间窗，重采样等），然后导入数据并配置Simulink环境，如下图，

Simport启动时，会根据当前模型Root层的Inport个数，生成对应的信号行，每个信号行对应一个输入Inport。

- INCA (*.dat *.mdf)
- CANoe (*.blf *.asc)
- BusMaster (*.log)
- VehicleSpy (*.vsb *.csv)
- CANdb database (*.dbc)
- ...



Simulink Root System



2.2. 文件解析

目前支持导入的软件数据格式包括：

- INCA (*.dat *.mdf)
- CANoe (*.blf *.asc)
- BusMaster (*.log)
- VehicleSpy (*.vsb *.csv)
- CANdb database (*.dbc)

为节省资源开销，解析文件时尽可能避免一次性读入所有数据。对于各种CAN数据文件，无法避免所有报文扫描，因此相对读入效率会慢一点。

此外，对于dbc文件，其本质只是一种数据字典，并不包含真实数据，因此导入dbc文件时必须同时指定一个对应的CAN数据文件。

2.3. 信号列表

每个文件导入后，会将其中包含的信号列表添加到下拉菜单列表中，用户可以选择：

- 对于MDF格式，其中所有信号添加到下拉菜单列表；
- 对于各类CAN记录格式，对于每一个CAN报文ID生成两个信号，分别代表其8*uint8的数据值以及发送时间(Active)，例如对于16进制"100"帧，下拉菜单中会生成"0x100"和"0x100_Active"供选择；

- 对于dbc格式，将其中所有信号放在下拉菜单中供选择，在导入时，根据其对应的数据文件相应解析成物理值；
- 当多个文件中信号出现重名时，在其后会后缀@N表示信号属于第N个文件。文件序号可以通过配置按钮查看；

2.4. 信号属性

导入的信号其数据属性为，

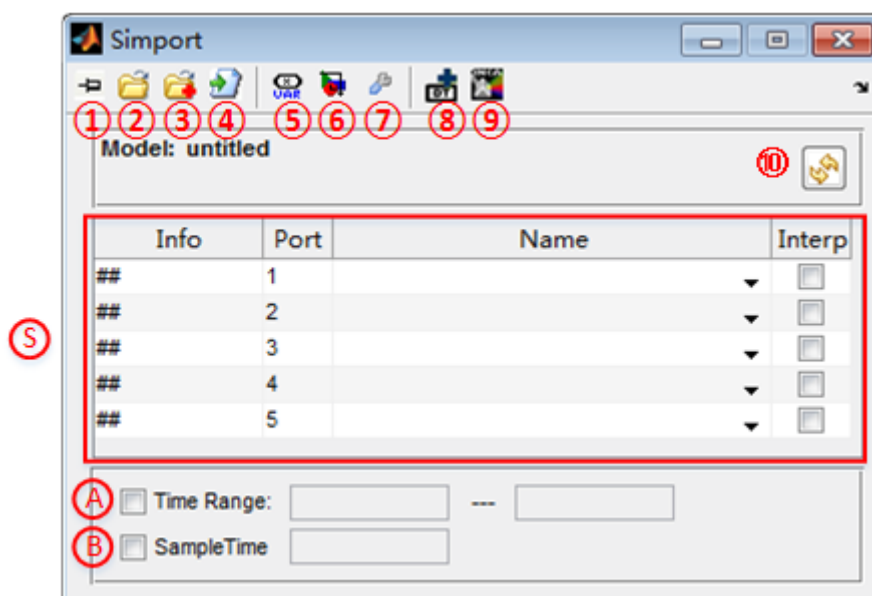
- CAN信号: DataType=uint8, Dimension=8
- CAN Message Active: DataType=boolean, Dimension=1
- 其它信号: DataType=double, Dimension=1

Simport信号列表的第一列显示了信号的来源和维度信息。

此外，最后一列"Interp"表示采样插值方法，如果打勾则中间采样点用线性插值。对于速度、扭矩、电压、电流等模拟量，应该选用插值（打勾），对于状态枚举量、布尔量、CAN数据等，应不插值（不打勾）。对此Simport首先会尝试自动判断，但使用者需要注意此处，必要时人工干预。

3. 使用方法

3.1. 工具栏



- 1) 窗口总在最上
- 2) 全新导入数据文件，覆盖已有导入

3) 添加另一/多个数据文件，从而实现多数据源联合仿真，或者同一工况多次试验数据的对比

4) 导入信号数据，配置Simulink仿真环境。导入配置成功后即可在Simulink下仿真

5) 自动匹配信号：如果Inport模块的名称与数据文件中的信号名一致，则自动选择匹配

6) 切换目标模型：如果同时打开多个模型，可用该按钮切换目标仿真模型

7) 显示已导入文件列表，以及更改文件时间轴（起始点、偏移量、系数）

8) 辅助功能，在当前Root System下的每个Inport后添加DataTypeConverter模块

9) 辅助功能，将MATLAB Workspace中的标定量数值更新为INCA中的值（INCA中对应Experiment需要打开，但标定量不需要拖出）

10) 刷新模型状态，用于增删模型Inport后，刷新下方信号列表行数

3.2. 信号列表 (S)

信号列表分四列，

- Info列：信号的来源、维度等信息
- Port列：端口号，与模型对应
- Name列：信号名称，由导入文件所包含信号合并形成的下拉式选择
- Interp列：指定对应信号是否进行线性插值

3.3. 时间配置

- A - Time Range：手动指定仿真时间窗口。在配置信号时，默认设置为信号起始时间。
- B - Sample Time：手动指定仿真步长。在配置信号时，会默认设置为其中最短的信号采样周期。

Tips:

要使用CAN数据进行按通信协议解析信号仿真，可以直接使用本工具中的dbc格式进行导入，也可以用本工具导入CAN原始数据，然后利用Vehicle Network Toolbox自带的CAN Pack/Unpack模块进行信号解析。

4. 应用场景

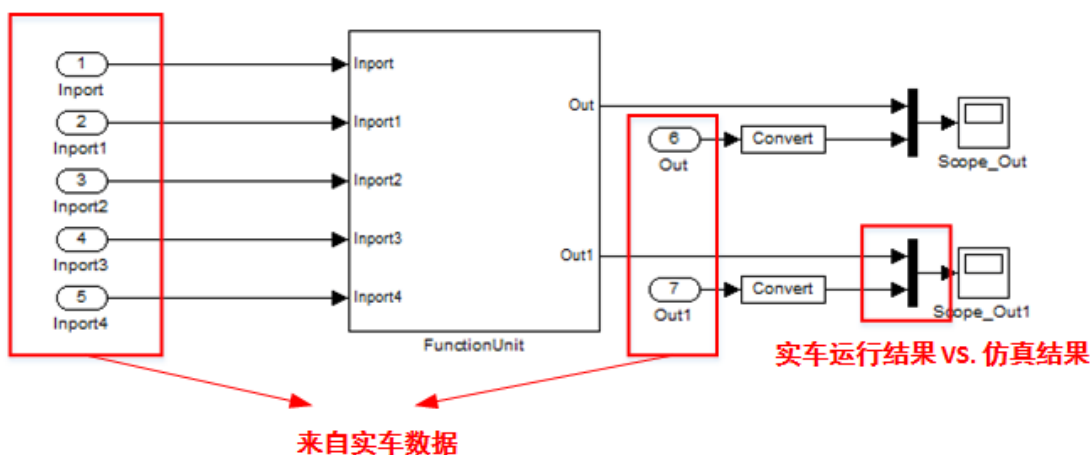
分享几种平时工作中几种常用的场景，

4.1. 策略运行Replay/问题分析

对于一些控制功能模块，利用实车数据可以在Simulink环境下几乎完全复现实车控制器内的运行状态。有时遇到一些杂症，中间变量录不全，在脑子里假装跑模型分析又太累，或许可借由此途径，导入数据后仿真Replay，可以进行一些细节分析进而排察问题点。

具体步骤，如果一个功能单元SubSystem没有内部状态或者内部状态初始化比较简单，可以

- 用标定工具录下它的输入（或者部分输入，例如有些不变的boolean输入可以给常数）
- 用提供的INCA更新按钮（见前述）更新实车状态标定数据
- 根据需要调整初始状态，和外部调度关系
- 仿真Replay



4.2. 模型学习

对于新手来说，拿一个实车数据在Simulink环境原样复现后，可以从容学习其中数据流、逻辑，甚至于改改删删或者试着重新搭一遍。这是种有效的学习途径。

4.3. 工况数据分析统计

实车数据不光可以喂给原来的策略模型，也可以喂给新搭的模型，比如用来做工况数据统计。例如，需要不同车速区间下能量回收的能量比，或者需要统计不同混动模式的换挡次数等算法里没有的量，用简单的计算模型加上实车数据就能后期统计出来。对于做一些优化的工作，这个办法经常有很好的效果。

4.4. 工况比较

可以同时导入同一工况的两个数据，叠在一起进行比较分析，便于找出差异。举个例子例如做混动NEDC能耗优化，不同次试验数据在同一个模型里计算比较，能较为容易地发现异同。

4.5. 新算法开发

用过往的数据做一些新算法的开发，尤其是一些工况识别类的标志位时，非常有助于提高效率。例如VCU中判断某种加速踏板模式，BMS中判断特定的电流形式等情况，用实车数据作为输入比人造信号效果好的多。

4.6. 模型测试/回测

实车数据可以部分作为单元测试输入，同样有其真实性优点。

另外，例如对已有功能块模型做整理，除了正常的测试流程外，也可以用过往的各种工况数据回测，比较结果与整理前是否一致就大致有谱了。

5. 命令行和函数

对于MATLAB脚本熟悉的朋友，用命令行更容易完成一些批处理的脚本，比如批量导入处理和分析一批数据。

这部分容后再详写，暂且举个栗子，一般的格式，

```
blf = SimportFileBLF(blffile); % Initialize and load the blf file
blf.LoadData({'var1', 'var2', ...}); % Load data of the specified signal (cell or string)
var1obj = blf.GetVar('var1'); % Get variable object, within which the data has been
prepared with Time and Data
```

其它文件格式类似。

6. 潜在问题和待完善

目前考虑到的一些潜在问题：

- 已支持的文件格式解析未必完整，比如blf文件、vsb文件的结构定义本身就有版本更新；
- 目前不支持MDF4.0的数据格式(*.mf4)

- 部分文本文件格式如asc, log 等没找到明确的specification定义，对着文本格式写得解析代码，可能有未遇到的特殊情况会出问题；
- 格式解析都是用m文件写的，尤其是CAN报文格式普遍解析效率不够，文件很大的话读入过程太慢了；
- 过往的开发环境基本没有超过2012a版本，高版本MATLAB下可能会有兼容性问题。

后续主要的方向是完善现有文件格式解析，并且逐步增加对更多常用数据记录格式的支持。

有什么意见建议，或者发现bug，请反馈到 jiangxinauto@163.com，谢谢！