

# 基于Python语言的轮胎模型测试数据格式系统设计

刘凤阳, 李 赛, 于 萍

中汽研汽车检验中心(天津)有限公司, 天津

收稿日期: 2023年4月12日; 录用日期: 2023年6月1日; 发布日期: 2023年6月8日

---

## 摘 要

轮胎动力学特性对车辆行驶稳定性、平顺性以及安全性等均有重大影响。随着仿真技术的发展, 轮胎测试数据的准确性对轮胎模型的建立及车辆仿真起着至关重要的作用。由于建立轮胎模型所需的数据量较大, 并且数据是由不同的设备测试得到的, 而软件对测试数据的格式和内容又有一系列的要求, 逐一调整内容格式会耗费较多的时间和精力。为了解决这个问题, 我们基于Python语言开发了数据处理系统, 既可以快速准确地处理数据, 又可以实现绘制图像检查数据的功能。

## 关键词

轮胎, 数据格式, Python

---

# Design of Tire Model Test Data Format System Based on Python

Fengyang Liu, Sa Li, Ping Yu

CATARC Automotive Test Center (Tianjin) Co., Ltd., Tianjin

Received: Apr. 12<sup>th</sup>, 2023; accepted: Jun. 1<sup>st</sup>, 2023; published: Jun. 8<sup>th</sup>, 2023

---

## Abstract

Tire dynamic characteristics have great influence on vehicle driving stability, ride comfort and safety. With the development of simulation technology, the accuracy of tire test data plays a vital role in the establishment of tire model and vehicle simulation. Because of the large amount of data needed to establish tire model, and the data are obtained by different equipment tests, and the software has a series of requirements for the format and content of test data, it will take more time and energy to adjust the content format one by one. In order to solve this problem, we developed a da-

ta processing system based on Python, which can not only process data quickly and accurately, but also realize the function of drawing images and checking data.

## Keywords

Tire, Data Format, Python

Copyright © 2023 by author(s) and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

## 1. 概述

轮胎动力学是车辆动力学研究和车辆控制算法开发的基石，轮胎的力学性能对于车辆的操纵稳定性和行驶安全性能的影响非常大。随着社会经济和科技的快速发展，国内各大整车开发企业均已具备自主汽车正向开发实力，但在轮胎力学特性上的研究与应用上还有明显不足。轮胎模型对轮胎力学特性的准确表达和应用是整车动力学仿真与底盘开发效率提升的关键点之一。

目前被广泛使用的轮胎动力学模型有纯经验的 PAC2002 模型、MF 模型、半经验的 swift 模型，以及物理 FTire 模型、CDTire 模型。PAC2002 模型和 MF 模型是基于魔术公式构建的，使用在操纵稳定性方向的。FTire 模型更多地使用在平顺性及耐久性上。CDTire 模型更多地使用在路噪仿真方向上。

目前构建模型的商用软件仅提供数据模板，并不提供数据格式的调整功能。而相关软件又要求数据格式有相关标识能够自动化识别种类，时间序列严格单调增加，指定通道有详细数据不为空列等要求。目前以米其林为代表的技术前列公司均自行编写相关程序进行格式化处理。大多数进行技术储备的公司仍旧以手动处理的方式进行数据格式转换。

以 FTire 为例，我们为了完成一款 FTire 模型，我们需要使用的设备有轮胎静态刚度设备，轮胎高速均匀性测试设备，轮胎六分力测试设备，具体见表 1。我们需要将数据处理成标准格式，能够让软件正常读取，进行自动工况分类，并进行初步分析。在处理数据的同时，我们要能够看到当前工况的数据情况，即绘制相应的曲线，人工进行数据筛选，以便提升后续工作效率。

基于以上需求和日常使用的办公电脑性能，我们选择了 PyCharm 开发工具，基于 Python 语言开发一套数据格式系统，旨在能够将 excel 数据，txt 数据，tdx 数据转化为软件可以识别的 tydex 标准格式数据 [1]，并能够绘图检查数据是否存在问题。

**Table 1.** Classification of test items

**表 1.** 试验项目分类

试验项目	使用设备	试验项目	使用设备
径向刚度		侧偏刚度	
纵向刚度		纵滑特性	轮胎六分力测试设备
侧向刚度	轮胎静态刚度设备	动刚度	
扭转刚度		Cleat 试验	轮胎高速均匀性测试设备
包覆刚度		/	/

## 2. 系统设计

### 2.1. 操作界面设计

如图 1 所示, 为本系统操作主界面。主界面内由几部分组成, 分别为轮胎信息、轮胎试验工况信息和选择试验工况、选择数据源文件以及 tdx 数据转换几个部分。

Figure 1. Main interface of system operation

图 1. 系统操作主界面

轮胎信息模块主要包括轮胎编号、生产厂商、花纹型号, 轮胎规格(包括速度符号、速度等级)、使用轮辋宽度、参考胎压、轮胎质量和标称质量输入框, 轮胎编号前缀根据外部试验和内部试验分为 CATARC 和 Tire App 两种, 可通过图 2 所示的轮胎编号后的下箭头进行选择, 选择 CATARC 或者 Tire App 后可在选择框后输入轮胎测试编号, 生产厂商输入框用于输入轮胎生产厂家, 花纹型号输入框用于输入测试轮胎的花纹信息, 轮胎规格输入框用于输入标准格式的轮胎规格信息, 轮胎质量输入框用于输入测试轮胎的质量信息, 标称载荷输入框用于输入轮胎的载荷指数信息。



Figure 2. Selection of tire number

图 2. 轮胎编号选择

轮胎试验工况信息和选择试验工况模块主要包括测试工况类型、试验载荷、试验速度、试验胎压、侧倾角、cleat 凸块信息。对于测试工况类型，我们直接将其罗列在界面上便于手动选择。对于试验载荷、试验速度、试验胎压以及侧倾角信息的输入，我们将常用信息汇总至左侧，便于手动选择。当然，也存在试验情况较为特别的存在，这时我们可以在点选后面的空白框，在框内输入我们的实际情况。同时，对于凸块的放置，考虑到其放置形式的多样性，我们对这些部分的输入作了细化输入功能设计。

选择数据源文件以及 tdx 数据转换模块则是功能实现模块。我们选择好源文件后，使用“汇总工况信息”按钮将所选工况信息进行汇总，统一进行格式化名称编写，选择数据保存文件夹。在转化保存数据前，使用绘图功能将当前工况重要性能曲线绘制，检查曲线图的标题与所选工况信息输入是否一致，并进一步检查曲线走向是否符合逻辑，最后进行点击“tdx 数据转换”按钮，将处理好格式的数据保存，这样我们的工作就可以完成了。

## 2.2. 自动分类功能实现

由于各项试验名称不同，同样的试验类型还会有不同载荷，不同速度的不同工况，我们需要对数据名称进行标识化处理。用  $v$  来代表速度，用  $p$  来代表气压，用  $fz$  代表垂向载荷，用  $cam$  代表外倾角度。我们还需要给不同的试验名称加以命名区分，如表 2 所示。

例如，我们在这个阶段用 `cleat_2p7_03fz500_cam0_30v0` 来表示垂向载荷 3500 N，气压 270 kPa， $0^\circ$  外倾角，30 km/h 工况下的 cleat 试验数据。通过对试验数据名称赋予特殊字段，可以实现软件读取数据后的自动分类功能。

Table 2. Marks of different test items

表 2. 不同试验项目标识

试验类别	标识	试验类别	标识
cleat 试验	cleat	侧偏刚度	side
径向刚度	cvert	纵滑特性	muslip
纵向刚度	clong	动刚度	dyn
侧向刚度	clat	横置凸块包容特性	trcl
扭转刚度	ctors	斜置凸块包容特性	obcl
轮辋触地	bottoming	纵置凸块包容特性	locl

为了尽量便捷同时尽可能详细的描述每条试验数据的试验种类及具体的试验工况，我们在轮胎试验工况信息和选择试验工况模块做了进一步细化输入功能设计。我们可以在前面直接点击选择不同载荷、不同气压、不同速度等。也可以选择后面的方框内手动输入的实际内容作为信息输入。

另一方面，我们考虑到包覆刚度和 cleat 试验都会使用到不同尺寸，不同放置方式的凸块，为了避免混淆，我们将不同设备上，不同放置形式，不同尺寸的凸块进行了编号，并通过几级下拉菜单进行选择，避免混淆。关键代码如下：

```
if TestInfo [5] == "横置":
    cl_angle = '_dP00'
if TestInfo [7] == '矩形':
if TestInfo [6] == '10x20':
    cl_symbol = 'M'
elif TestInfo [6] == '10x25':
```

```

cl_symbol = 'O'
elif TestInfo [6] == '15x15':
cl_symbol = 'P'
elif TestInfo [6] == '20x20':
cl_symbol = 'R'
elif TestInfo [7] == '半圆形':
cl_symbol = 'T'
elif TestInfo [5] == "斜置":
cl_angle = '_dP45'
if TestInfo [7] == '矩形':
if TestInfo [6] == '10x20':
cl_symbol = 'N'
elif TestInfo [6] == '15x15':
cl_symbol = 'Q'
elif TestInfo [6] == '20x20':
cl_symbol = 'S'
self.cl_Symbol.setText (cl_symbol)
TestInfoStr = "测试工况: Cleat 试验载荷:" + str (TestFz) +"N 试验速度: " + str (TestVx) +"km/h 胎压:" + str
(TestIP) +" kPa 侧倾角:" + str (TestIA) +"deg "\
"凸块:" + TestInfo [5] + "" + TestInfo[6] + "" + TestInfo [7] + "" + cl_symbol
FileSaveName = "cleat_" + str(p1) + "p" + str (p2) + "_" + str (fz3) + "fz" + str(fz4) + "_" + \
'P00cam0_' + str(TestVx) + "v0_" + cl_symbol + cl_angle + ".tdx"
self.testCondition.setText (TestInfoStr)
self.dataSaveName.setText (FileSaveName)

```

在这样的设计方案下,可以实现将试验类型及凸块方向、尺寸、形状全部选中后,点击汇总信息,自动选择凸块编号,在数据中对凸块进行放置方式以及尺寸的描述。同时,只需要制作一份凸块编号对应的凸块信息,即可让软件自动识别试验时所用凸块的详细情况。

### 2.3. 绘制曲线功能设计

本功能为系统内关键功能之一。根据试验工况选择以及输入信息,对所选数据的重点工况进行选择,同时绘制出本工况最重要的性能曲线[2]。如图3以cleat试验为例,将轮胎信息、凸块信息及工况信息在相应位置输入后,先后点击“汇总试验信息”按钮和“查看试验曲线按钮后”,名字及相应曲线便显示出来。此时数据名称显示“cleat\_2p5\_07fz800\_P00cam0\_90v0\_M\_dp00”。即为cleat试验,试验气压是250 kPa,垂向载荷为7800 N,外倾角为0°,速度为90 km/h,使用凸块编号为M(安装于转鼓上,横置放置,20 mm宽,10 mm高),角度0°放置。详细曲线如图4所示,关键的载荷及速度信息也在图例中显示。

本工况关键代码如下:

```

elif TireTestInfo[0] == 'cleat':
FX0 = float (data [0] [5])
FY0 = float (data [0] [6])

```

```
lastTime = float (data [0] [0])
Time = []
Fx = []
Fy = []
Fz = []
Vx = []
for i in range(len (data)):
    writedata = data [i]
    RUNTIME = float(writedata [0])
    Time.append(RUNTIME)
    FXW = float (writedata [5]) - FX0
    Fx.append (FXW)
    FYW = float (writedata [6]) - FY0
    Fy.append(FYW)
    FZW = abs (float (writedata [1]))
    Fz.append (FZW)
    Speed = float(writedata [3]) * 2 * 3.14 * 1/60 * 3.6
    Vx.append(Speed)
    FZI = int(Fz [0])
    VXMEAN = int(np.mean (Vx))
    plt.rcParams['font.sans-serif'] = ['SimHei']
    plt.rcParams['axes.unicode_minus'] = False
    plt.figure(figsize=(10, 6))
    plt.subplot(3, 1, 1)
    plt.plot(Time, Fz, 'r-')
    titleStr = '动态冲击 径向载荷: '+str(FZI)+' N+' 速度: '+str(VXMEAN)+' km/h'
    plt.title(titleStr)
    plt.ylabel('径向力 Fz [N]')
    plt.grid()
    plt.subplot(3, 1, 2)
    plt.plot(Time, Fx, 'b-')
    plt.ylabel('纵向力 Fx [N]')
    plt.grid()
    plt.subplot(3, 1, 3)
    plt.plot(Time, Fy, 'k-')
    plt.xlabel('时间 [t]')
    plt.ylabel('侧向力 Fy [N]')
    plt.grid()
    plt.show()
```

CATARC tdxMaker

轮胎信息

轮胎编号: CATARC 001

生产厂商: A 花纹型号: A

轮胎规格: 245 / 40 R 18 100 Y 8 J

参考胎压: 250 kPa 轮胎质量: 14 kg 标称载荷: 7845 N

轮胎试验工况信息

纵向刚度  侧向刚度  径向刚度  扭转刚度  包裹刚度  轮胎触地  Cleat  侧偏刚度  纵弯特性

试验载荷:  40  50  80  100  120  150 %  other LI  % or  Fz: 7800 N

试验速度:  0  2  20  30  40  60  90 km/h or  Vx:  km/h

请选择试验工况: 试验胎压:  80  100  120 kPa or 0.5bar or other IP Perc.  % or IP  kPa

侧倾角:  -6  -4  -2  0  2  4  6 deg or other  deg

凸块方向:  凸块尺寸:  凸块形状:  凸块编号:

试验工况信息汇总: 测试工况: Cleat 试验载荷: 7800N 试验速度: 90km/h 胎压: 250kPa 侧倾角: 0deg 凸块: 10x20 矩形 M

试验数据:

tdx文件转换

文件保存目录:

文件保存名: cleat\_2p5\_07fz800\_F00cam0\_90v0\_M\_4P00.tdx

Figure 3. Cleat example condition

图 3. Cleat 实例工况

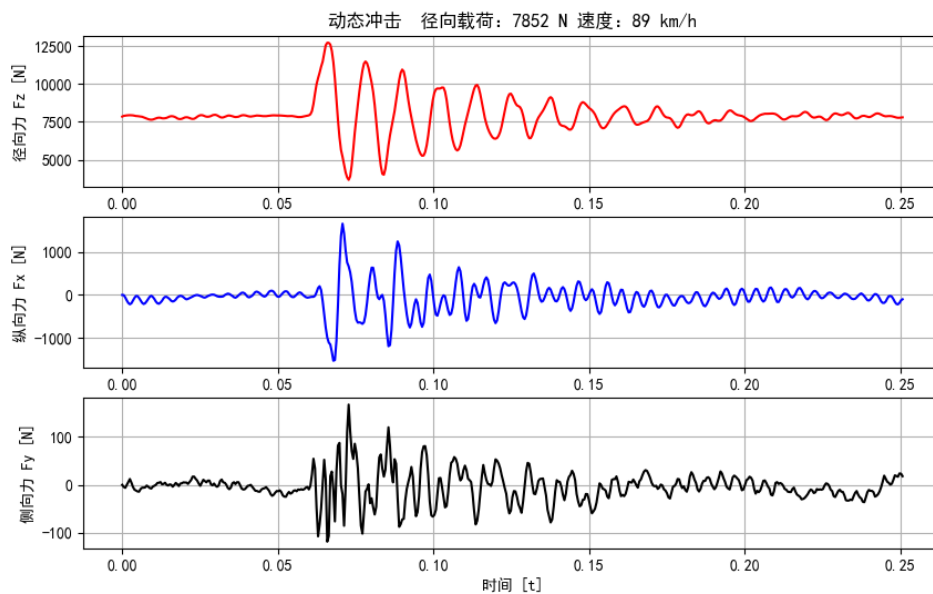


Figure 4. Cleat condition curve drawing

图 4. Cleat 工况绘图

## 2.4. 数据补充功能实现

对于径向刚度工况,模型在辨识时不仅需要形变量的数据,同样还需要轮心距地面的距离数据。而设备并不具备输出此部分数据的功能。为此,我们结合轮胎基本信息,通过计算的方式补全本通道。我们可以通过轮胎的规格大致估算出轮胎直径,如下式:

$$D = \frac{2 \times A \times B}{100} + 25.4 \times C$$

式中,  $A$  为轮胎断面宽度(mm),  $B$  为轮胎高宽比(%),  $C$  为轮辋名义直径(in),  $D$  为轮胎直径(mm)。我们默认轮胎在进行刚度测试时, 刚接触的瞬间轮胎不发生形变, 则初始轮心距地面距离为轮胎半径。因此, 测试过程中每个时间点的轮心距地面距离可以通过下式计算:

$$D_g = \frac{D}{2} - D_t$$

式中,  $D_g$  为轮心距地面距离(mm),  $D_t$  为轮胎形变量(mm)。根据以上计算方式, 可以实现轮心距地面距离通道数据的补充计算[3]。关键代码如下:

```
tireDiameter=(2 * int(TireInfo [4]) * int(TireInfo [5])/100+int(TireInfo [6]) * 25.4)/1000
    rimDiameter = int(TireInfo [6]) * 25.4/1000
    rimWidth = float(TireInfo [9]) * 25.4/1000
INCLANGL = float(writedata [2])
TYREDEFW = float(writedata [4])-TYREDEFW0
DSTGRWHC = tireDiameter * 500-float(TYREDEFW)
```

### 3. 结论

本文基于 Python 语言, 设计了面向轮胎动力学模型测试数据的格式处理系统, 该系统能够实现通过给予特定字符方式实现分类功能, 通过绘图功能针对特定通道数据进行图像趋势检查, 通过调用已用通道数据结合轮胎基本信息进行计算补充其他需求通道的功能。通过本系统, 可以在不同设备上对不同测试项目输出的不同格式的数据转化成标准格式的 Tydex 格式文件, 避免了人工调整数据格式时潜在的问题, 避免后续工作软件反复误读造成时间浪费。使用本系统处理数据可以将数据处理工作变繁为简, 大大提升工作效率与准确性。

### 参考文献

- [1] Mark Lutz. PYTHON 学习手册[M]. 第3版. 北京: 机械工业出版社, 2009.
- [2] Wes Mckinney. 利用 Python 进行数据分析[M]. 北京: 机械工业出版社, 2014.
- [3] William F. Punch, Richard Enbody, 伊鲍德, 等. Python 入门经典: 以解决计算问题为导向的 Python 编程实践[M]. 北京: 机械工业出版社, 2012.