

DOI:10.19651/j.cnki.emt.2314165

测试信号分析云计算技术研究*

黄江城 何岭松

(华中科技大学机械科学与工程学院 武汉 430074)

摘要: 随着通讯技术和云计算技术发展,前端获取数据后上传至云端分析,然后再将云端分析结果回传至前端进行显示的 APP 设计和影音模式逐渐在生活中普及。借鉴其思路,提出一种 JS+Python 的测试信号分析云平台设计技术;其中网页型的 JS 前端页面程序负责信号获取和分析结果显示,而运行在云服务器上的 Python 程序则负责上传数据的云计算和分析。运行测试实验平台,数据和脚本上传后,服务器端能成功接收并调用脚本中对应的测试技术和控制理论算法对数据进行处理,并返回浏览器实现结果可视化。这样可最大限度降低前端设备的软硬件要求,便于实现由前端测量装置+后端测试信号分析云平台构成的设备运行故障监测与诊断网络。

关键词: 云计算; Django; 测试技术; 故障诊断; Python

中图分类号: TP393 **文献标识码:** A **国家标准学科分类代码:** 520.407

Research on cloud computing technology for test signal analysis

Huang Jiangcheng He Lingsong

(School of Mechanical Science & Engineering, Huazhong University of Science and Technology, Wuhan 430074, China)

Abstract: With the development of communication technology and cloud computing technology, the front-end obtains data and uploads it to the cloud for analysis, and then transmits the cloud analysis results back to the front-end for display APP design and audio mode are gradually popularized in life. Drawing on its ideas, a JS+Python test signal analysis cloud platform design technology is proposed. Among them, the web-based JS front-end page program is responsible for signal acquisition and analysis result display, while the Python program running on the cloud server is responsible for cloud computing and analysis of uploading data. Run the test experiment platform. After the data and script are uploaded, the server can successfully receive and call the corresponding test technology and control theory algorithm in the script to process the data, and return it to the browser to visualize the results. This minimizes the hardware and software requirements of the front-end equipment, and facilitates the realization of the equipment operation fault monitoring and diagnosis network composed of front-end measurement device + back-end test signal analysis cloud platform.

Keywords: cloud computing; Django; testing techniques; fault diagnosis; Python

0 引言

随着 5G 通讯技术和云计算的普及推动了数字化时代的发展,国内外的云计算产业迅速崛起,并广泛应用于各行各业^[1]。5G 技术提供高效便捷的移动云计算,突破了移动终端硬件资源限制,提供强大的数据处理能力^[2]。在语音处理领域,科大讯飞提供了一套强大的语音处理接口,使用该接口,用户可以将语音音频文件上传至云服务器,并在云端进行各种音频数据处理,包括语音转文字等功能^[3]。在生活应用领域,手机端植物识别小程序已经成为日常使用

的工具之一,采用拍照上传未知植物的图片数据,服务器端将图片数据预处理后提取特征与数据库的数据进行比对查询即可返回识别结果及相关文献^[5-6]。生活中使用的智能手环产品实时采集一天中人体的健康数据如心率、步数、睡眠质量等,将数据上传至云端进行健康分析并生成个性化建议,用户可以通过手机前端界面获取数据曲线及报告和建议^[7-8]。在物流管理中,通过采集终端或扫描设备采集的物流信息,如包裹位置和运输状态等,通过上传数据到云计算平台对进行物流追踪和管理,用户也可以实时了解物流信息也是常见应用场景^[9-10]。

收稿日期:2023-07-21

* 基金项目:国家重点研发计划项目课题(2019YFB1310703)资助

目前国内外这类前端获取数据,云计算后台返回结果的模式在各领域应用都十分广泛,但在测试与控制领域应用较少。在测控领域常用的软件如美国 MathWorks 公司开发的 MATLAB 采用 MathWorks Cloud 实现,具有多种不同终端的单体软件和网页平台,但其操作繁复、安装复杂,学习成本较高;美国 NI 公司的 LabVIEW 采用图形化编程方便无需深入编程细节^[11-13];东方所的 DASP 数据采集分析软件具有相对完善的测试信号分析功能,但需要购买相关产品,且界面操作复杂^[14]。以上都需要安装软件,安装复杂且本体软件占用内存大,且对电脑性能要求高,部分在使用上具有版权问题。

因此提出了一种基于云计算的在线测试分析实验平台,将其功能集中在对数据的测试分析,能够实现对多种数据的在线测试分析,并具备一定的故障诊断功能。操作方式简单,提供十分友好的人机交互界面,封装脚本指令处理数据,可在多种不同类型终端直接联网使用,易于操作,方便快速学习,便于拓展。同时平台以开源的形式方便任何用户使用。

1 云计算在线测控实验平台整体设计

测试信号分析云平台的总体架构如图 1 所示。用户通过手机、pad 或 PC 等运行的浏览器程序访问测试信号分析云平台中的网址,返回具有前后端交互 JS 控制逻辑的 HTML 页面,在网页 IDE 界面用户只需调用已封装的 API 或者在文件处理界面选择相关功能控件并绑定参数和文件后上传,控件底层原理为已封装的脚本,脚本的底层原理采用 Python 实现,语法近似 MATLAB 编程;云服务器后台运行着 Python 环境,接收到前端 HTTP 请求,后台对请求中的数据和处理脚本进行解析,调用算法函数库对数据进行处理,处理完成后返回 HTTP 或者 JSON 响应,前端接收数据调用相关的可视化绘图显示工具完成数据的可视化。

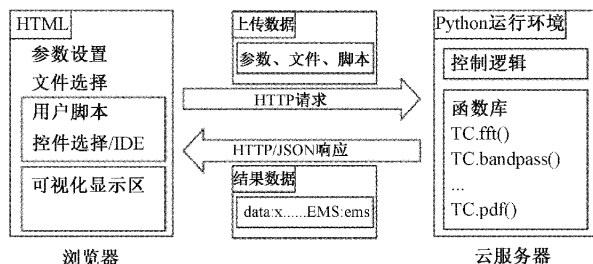


图 1 云计算在线测控实验平台整体架构

测试信号分析云平台的技术实现架构分为前端和云服务器后端两部分。云服务器端是系统的核心,承担着对接收到的请求进行分类相应的任务,后台集成了测试信号分析相关的实验算法和算法库,根据实验脚本需求处理和分析接收到的测试信号;同时云服务器的模板库还包含了用于前端界面渲染所需的 HTML 文件、JavaScript 文件、CSS

文件,构成了用户界面的控制逻辑和呈现形式;前端是由云服务器响应动态定义的程序,连接用户与测试信号分析云平台,提供友好的人机交互,实现用户与平台之间的数据读取、上传、解析与可视化等功能。此外,云计算后台的算法库具有很高的可扩展性,后续可以根据测试信号领域的发展和需求不断地丰富和完善。

2 测试信号分析云计算技术实现

2.1 云计算架构后端整体方案

1) 云计算架构后端整体方案

该云计算在线测试分析实验平台运行于阿里云服务器,采用基于 Python 的 Django 框架实现,并使用 Nginx 作为 HTTP 请求中的监听器^[15]。当 Nginx 接收到来自前端的请求时,将请求传递给 URL 调度器。在 URL 调度器中绑定了 URL 路径和对应视图函数之间的关系,根据 URL 将请求路由到相应的视图函数,后台架构整体如图 2 所示。

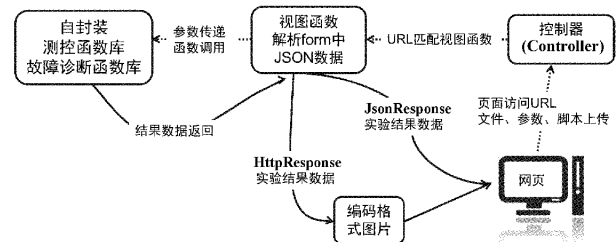


图 2 云计算架构后端整体方案

视图函数接受请求时会通过表单处理机制解析 POST 请求中表单信息以及存储在 FILES 中的文件信息并按照文件和参数分类后保存到后台缓冲区,服务器端的 Python 后台负责提取参数并调用解析脚本,执行测控算法函数处理文件。处理完成后,后台利用数据打包方法将结果重新封装为 JSON 格式,并以 JsonResponse 返回给前端网页。若上传为直接脚本指令,则直接对脚本指令进行解析,在 Python 运行环境下执行脚本并根据返回数据格式保存结果,将结果以 HttpResponse 的形式返回给前端网页。

2) 常用测控算法的实现原理

测试技术与控制理论相关实验涉及的算法实现依赖于 Python 丰富多样且功能强大的数据处理库^[16],主要包括科学计算库 Numpy、Scipy、控制系统仿真库 Python-control 和其它辅助计算库^[17]。这些库函数基本能实现测控技术设计的数学计算功能,但涉及的库比较杂乱。为此,参照 Matlab 中的函数名对它们进行了统一封装,以简化引用,如图 3 所示。

针对测控试验需求,Python 库函数只能实现部分关键数据的计算功能,函数集成度不够高,如直接在此基础上使用的话,还需要花费大量的时间进行训练,不利于使用。为此在同一封装函数名的基础上,基于测控试验的需求,新增了幅值、相位谱、功率谱等 134 个更便于应用的函数,根据

```
import numpy as np
from scipy.fftpack import fft,ifft
from scipy.io import wavfile
import math
from scipy import signal
from scipy.stats import kurtosis
```

import drvi

图3 测控算法封装原理

现有的实验、教学和使用要求对每一个封装的函数进行理论算法与实验结果验证。以单边频谱的数据处理为例,在解析脚本的之前将文件解析作为数据处理的参数 y 传递,基于 Python 的数据处理库函数实现自身的测控实验算法伪代码如图 4 所示。使用了 numpy 库进行数组操作和数学运算,使用 scipy 库中的 fft 函数进行傅里叶变换。函数 get_single_fy 用于归一化并取单边频谱数据,函数 get_x 用于生成频率轴数据,最后返回所需数据。至此完成对自定义函数的封装。

```
def SingleFFT(y, Fs):
    # 傅里叶变换
    fy = fft(y)
    # 归一化并取单边频谱数据
    single_fy = get_single_fy(fy)
    # 获取频率轴数据
    x = get_x(y, Fs)
    return x, fy
```

图4 测控算法伪代码示例

2.2 前端平台人机交互实现

1)前端人机交互整体方案

实验平台的前端人机交互网页 HTML 的关键功能是基于 Form 表单实现,并辅以其它控件搭建整体的前端网页,如图 5 所示。网页界面提供数据文件选择功能、Python 脚本编辑窗、ECharts 曲线显示控件,以及文本框、滚动条等参数输入控件。为便于使用,将常用的测控函数、前端绘图函数放在网页 HTML 菜单控件并采用 JS 实现后台逻辑,选择处理方式的分类列表,点击对应的处理方法函数即可在 Python 脚本编辑窗插入对应的处理函数,用户可以选择在现有函数基础上进行编辑或者直接使用现有处理方法的函数与文件一起以 FormData 的形式直接发送给云服务器后台进行数据处理。

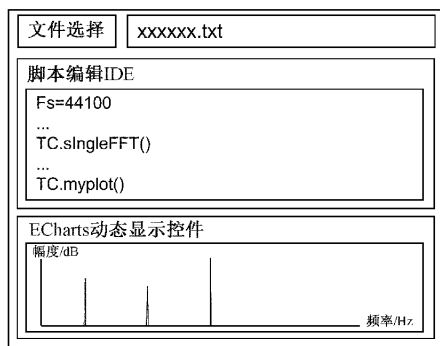


图5 前端人机交互界面设计

文件选择的主要功能是选择待分析的数据文件,然后上传到服务器的暂存文件区,服务器后台会根据上传文件名进行专门的标识,文件选择可以选择本地文件或者服务器上现有文件。Python 脚本编辑窗的作用分为两部分:(1)显示脚本控件中的文件数据处理算法,(2)在已有算法脚本基础上或者直接用扩展了前台绘图函数的 Python 书写测控分析程序,并通过表单传递给服务器后台进行分析和返回执行结果。基于 ECharts 封装了可以动态添加或删除的显示控件,实现对后台返回数据的可视化显示,同时提供放大、缩小、数据值显示等人机交互功能。

2)前后台的数据交换

前端界面初始化完成后,用户选择对应的实验模块,在文件选择栏选择需要分析的文件,部分处理函数需要添加新的参数,用户可以在参数输入栏输入参数,参数输入控件本身是 Form 表单的子控件,可以直接以键值对的形式跟随 FormData 上传,参数填写完毕后,在分析函数列表中选择需要的函数,完成对 Python 脚本编辑脚本代码的填充显示,用户根据自己需要选择直接上传处理或者修改函数后上传处理。在 FormData 的数据格式中,文件以 file 对象独立上传,函数脚本和参数以 QueryDict 字典的形式上传,函数脚本由 code 标识,参数则以参数名标识。基于此,函数脚本中就可以使用统一的参数名 file 标识文件,脚本和参数直接以各自标识获取,各参数名与文件的绑定工作由上传云服务器后台的处理逻辑完成。

云服务器后台完成对 FormData 的解析,以及参数名的绑定后,调用脚本解析函数对传入的脚本以及参数进行处理,实现对文件数据的处理或者直接脚本的执行。Python 提供常用的数据可视化软件包 Matplotlib,可以用 plot 等函数在 python 环境下十分方便的实现函数绘图功能。但是对本文的前后端应用的交互模式并不能使用此种方式实现。为此该平台对后台返回的数据格式采用了两种解决方式。

第 1 种方式是基于 Matplotlib 的函数进行封装,封装后函数名由 plot 转换为 PNG_plot,与原始函数进行区分的同时在函数中封装了图片数据转换的功能。封装后的函数的使用方法与原始的 Matplotlib 函数相同,使用相同的参数、调用方式和绘图逻辑来生成图像。使用封装后的函数将不再直接在 Python 后台上绘制图像,而是将所绘制的图像数据使用 Base64 编码,将后台图片数据通过 HTTP 响应返回给前端,为此引入用法一致的 PNG_plot 函数,如图 6 所示。网页中回调函数接收到数据后,判断数据格式后将数据解码调用封装的动态图片插入函数完成局部动态显示图像。

第 2 种是采用通过前端网页的 JS 绘图函数,基于现有的绘图函数,引入了 JSON_plot 函数。JSON_plot 函数将计算后的数组数据序列化,并利用 dumps 函数将其封装成 JSON 格式。这样,我们能够以与常规 plot 函数相同的方式使用 JSON_plot 函数。云平台将数据以 JSON 响应的形式返回给前端,触发回调函数将数据解析,并通过处理函数



图 6 动态图片显示原理

JSON 数据对应的提取后,调用基于 ECharts 的 JS 动态绘图函数完成局部绘图显示控件的添加,原理如图 7 所示。

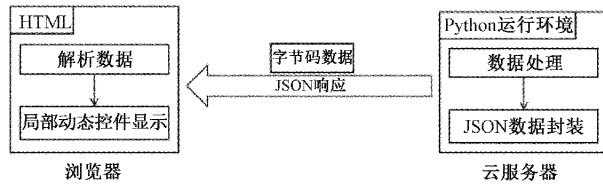


图 7 动态 ECharts 绘图原理

3 应用实例与验证

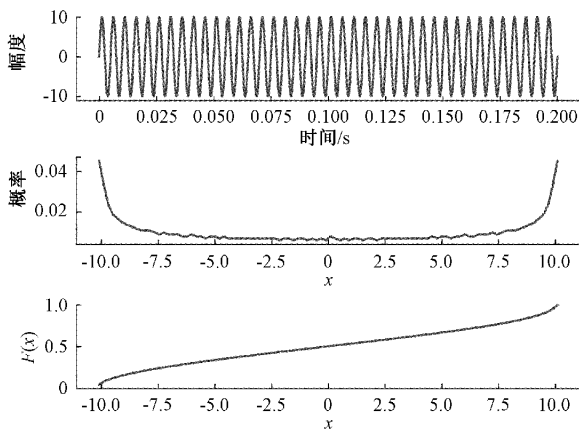
3.1 在线测控 IDE 应用实例

为了验证该实验平台的可行性、对数据分析的准确性、易操作性与测试样例的丰富性,在实验列表根据实验类型选择测试和控制理论实验案例,在 IDE 编辑器中完成案例的脚本输入,云服务器后台将处理结果返回,实例如图 8 所示。

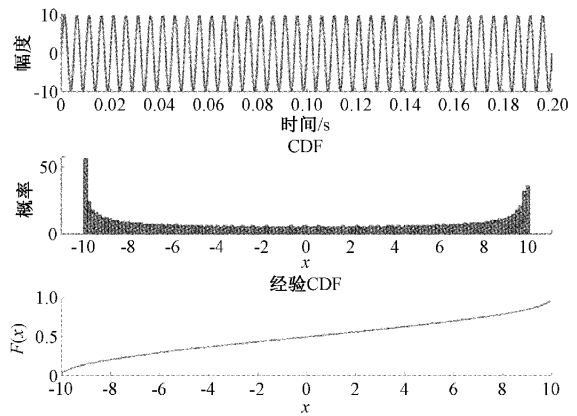
```

测试技术线上实验平台 DRVI IDE
运行 返回
1 Fs = 5120
2 N = 1024
3 dt = 1.0 / Fs
4 T = dt * N
5 t = linspace(0, T, N)
6 y = 10 * sin(2 * pi * 200 * t)
7 subplot(311)
8 plot(t, y)
9 subplot(312)
10 pdfplot(y)
11 subplot(313)
12 cdfplot(y)
lin: 12, Col: 13
    
```

(a) 测控实验 IDE 编辑器



(b) 后台结果返回



(c) MATLAB 分析结果

图 8 在线测控 IDE 应用实例

实验结果如图 8 所示,实现了正弦函数及其概率分布和累积分布图表绘制,与 MATLAB 进行对比分析其结果正确无误,对比 MATLAB 的直方图绘制,本平台采用线性绘制更精细的展示概率分布变化情况,表明该测试实验平台能够正确的应用于测控实验,完成对前端传来的测控实验脚本正确解析与返回,具有很好的灵活性。

3.2 数据分析与故障诊断拓展应用实例

在测控分析实验中,往往要使用传感器采集机器运行过程中的振动、声音等模拟信号,并将其转换成数字信号。因此设计了如图 9 所示的在线数字信号分析实验模块。选择上传本地文件或者直接调用云服务器端的数据,通过实验类型控件选择实验类型列表下的实验脚本,完成数据和处理脚本的上传,服务器调用相关算法和计算资源完成对数据的处理,前端接收结果数据完成数据的可视化。

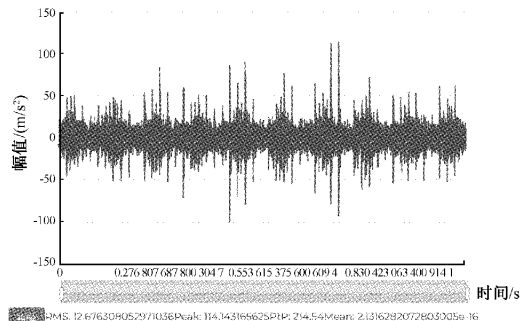
通过添加大量实验样本文件,可以在此基础上将功能拓展至故障诊断。将大量实验样本打包上传至云服务器识别后进行批处理,本文采用辛辛那提大学的轴承实验数据集 2 中的轴承 1 的数据为例,如图 10 所示,当轴承持续运转较长时间后出现故障,有效值明显波动。

数据上传与参数设置

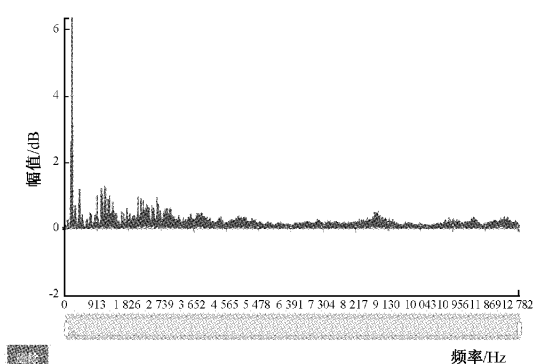
选择本地文件 选择服务器文件

选择文件 加速度 m 采样率 25 600 HZ.txt 采样频率Fs 25 600

(a) 文件选取参数填写



(b) 波形显示



(c) 单边频谱
图9 数据分析应用实例

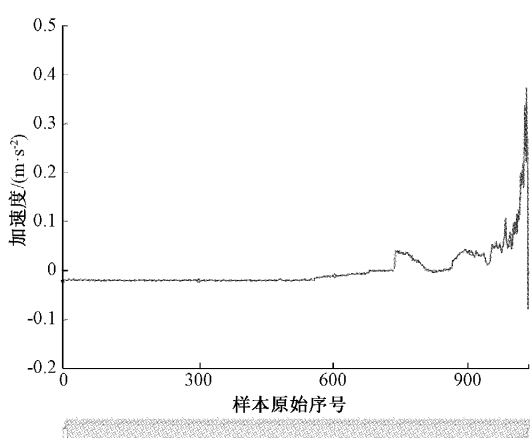


图10 故障诊断拓展应用实例

从实验可以看出,用户可以选择多种数据分析方法对数据进行分析并且可拓展至故障诊断领域,同时使用人机交互友好的控件对结果数据进行操作。控件化封装脚本指令,使得实验平台具有较高的可拓展性以及易操作性。

4 结 论

该文利用前后台技术设计实现了一个 Python 云计算测试实验数据分析平台。用户通过手机、计算机等任何支持浏览器的设备,就可以在网页用 Python 语言编写测试实验数据分析脚本;然后调用布置在云端的 Python 引擎和函数库进行运算;最后,再将计算结果以美观的 ECharts 图表形式返回给前端界面。本文平台经过多次测试,同时在课堂交由学生在线进行学习运行,均能正常工作、运行稳定可靠。本问平台优点是学生、工程师等用户,无需安装任何专业软件,只需联网即可用类似 MATLAB 中 M 语言的 Python 语言快速进行自定义的实验数据分析应用。随着实验与教学需求的增长,以及测控技术的深入研究,不断扩充基于 Python 构建的测控算法库。能够封装更复杂的专业实验样例算法,以满足不断变化的测控实验需求。本平台的架构并不局限于测控技术领域的信号分析,在具备相关标准数据集的情况下,还可拓展至故障诊断领域。

参考文献

- [1] 栗菁. 5G 网络对云计算发展的影响探讨[J]. 数字通信世界, 2023, 218(2):173-175.
- [2] 王曦,周惠兴,王舜,等. 基于边缘计算的室内测量设备远程控制系统设计与实现[J]. 国外电子测量技术, 2022, 41(8):159-166.
- [3] 李霄垅,王梦婕. 基于语音识别 APP 的同声传译能力培养教学模式建构与研究——以科大讯飞语记 APP 为例[J]. 外语电化教学, 2018, 179(1): 12-18.
- [4] BANERJEE N, BORAH S, SETHI N. Intelligent stuttering speech recognition: A succinct review[J]. Multimedia Tools and Applications, 2022, 81(17): 24145-24166.
- [5] 许展慧,刘诗尧,赵莹,等. 国内 8 款常用植物识别软件的识别能力评价[J]. 生物多样性, 2020, 28(4): 524-533.
- [6] OTTER J, MAYER S, TOMASZEWSKI C A. Swipe right: A comparison of accuracy of plant identification apps for toxic plants [J]. Journal of Medical Toxicology: Official Journal of the American College of Medical Toxicology, 2020, (1):42-47.
- [7] 陈春谋. 基于 Android 系统的智能手环健康检测系统关键技术[J]. 微型电脑应用, 2019, 35(9):99-101.
- [8] ANIKWE C V, NWEKE H F, IKEGWU A C, et al. Mobile and wearable sensors for data-driven health monitoring system: State-of-the-art and future prospect [J]. Expert Systems With Applications, 2022, 202:117362. 1-117362. 32.
- [9] YU Q. Design of logistics tracking and monitoring system based on internet of things [J]. Journal of Residuals Science & Technology, 2016, 62(2):75-82.
- [10] 欧阳小迅. 智慧物流的发展特点及技术体系分析[J]. 物流工程与管理, 2019, 41(10):1-3, 22.
- [11] 高红英. 基于 MATLAB 的机械综合部件虚拟设计实现方法研究[J]. 国外电子测量技术, 2021, 40(3): 99-102.
- [12] 杨莉. 基于 LabVIEW 的虚拟仪器在中职电子技术中的应用[J]. 科学咨询(教育科研), 2020(12):118.
- [13] 黄彬,张春,王乐平. 基于 LabVIEW 的座椅耐久性可视化测试系统[J]. 电子测量技术, 2022, 45(10):33-38.
- [14] 康群英,刘劲涛,王琳. 电力机械关键零件加工状态检测与分析[J]. 沈阳工程学院学报(自然科学版), 2022, 18(3):77-85.
- [15] 邱红丽,张舒雅. 基于 Django 框架的 web 项目开发研究[J]. 科学技术创新, 2021(27):97-98.
- [16] WANG Y, HILL K J, FOLEY E C. Computer programming with Python for industrial and systems engineers: Perspectives from an instructor and students[J]. Computer Applications in Engineering Education, 2017, 25(5):800-811.
- [17] 赵海滨,颜世玉. 变形 Liu 混沌系统控制仿真实验设计[J]. 中国教育技术装备, 2020(16):39-41.

作者简介

黄江城,硕士研究生,主要研究方向为数字信号分析、云计算实验平台、测控技术。

E-mail: hustjiangcheng@qq.com

何岭松(通信作者),博士,教授,主要研究方向为工业测量与控制、计算机虚拟仪器技术、手机化移动测量仪器。

E-mail: helingsong@hust.edu.cn