

1-4-2019

Design of Fault Tolerant Control Visual Simulation System For Flying Wing Aircrafts

Shaojie Zhang

College of Automation Engineering, Nanjing University of Aeronautics and Astronautics, Nanjing 211106, China;

Qingkai Meng

College of Automation Engineering, Nanjing University of Aeronautics and Astronautics, Nanjing 211106, China;

Follow this and additional works at: <https://dc-china-simulation.researchcommons.org/journal>



Part of the [Artificial Intelligence and Robotics Commons](#), [Computer Engineering Commons](#), [Numerical Analysis and Scientific Computing Commons](#), [Operations Research](#), [Systems Engineering and Industrial Engineering Commons](#), and the [Systems Science Commons](#)

This Paper is brought to you for free and open access by Journal of System Simulation. It has been accepted for inclusion in Journal of System Simulation by an authorized editor of Journal of System Simulation.

Design of Fault Tolerant Control Visual Simulation System For Flying Wing Aircrafts

Abstract

Abstract: A visual simulation platform based on Qt Quick was designed with respect to the simulation about fault tolerant control for flying wing aircrafts. Aircraft mathematical model and control law were developed using Simulink. Man-machine interface was developed based on Qt Quick to inject fault to Simulink model, load control law, setting up the simulation parameters and process, the interface can monitor and display graphically the key parameters to test and verify fault tolerant control laws. FlightGear visual engine was driven by MATLAB to achieve real-time visual flight simulation and the aircraft were showed in 3-D model. Baidu Map's API function was adopted by QtWebView to achieve the navigation map of the aircraft, which designed by JavaScript. The simulation results show that this simulation platform is convenient, full-featured, and scalable.

Keywords

Qt Quick, fault tolerant, flying wing aircrafts, visual simulation, FlightGear, Baidu Map API

Recommended Citation

Zhang Shaojie, Meng Qingkai. Design of Fault Tolerant Control Visual Simulation System For Flying Wing Aircrafts[J]. Journal of System Simulation, 2018, 30(10): 3732-3738.

飞翼布局飞行器容错控制可视化仿真平台开发

张绍杰, 孟庆开

(南京航空航天大学自动化学院, 江苏 南京 211106)

摘要: 针对某型飞翼布局飞行器的容错飞行控制可视化仿真验证问题, 开发了一个基于 QtQuick 的仿真平台。使用 Simulink 搭建飞行器数学模型及容错飞行控制律, 作为可视化平台模型驱动端。以 Qt Quick 开发人机交互界面进行模型故障注入、控制律加载、仿真参数设置和仿真过程控制, 并对关键飞行参数进行实时监控和曲线显示, 验证容错飞行控制律的效果。基于 MATLAB/FlightGear 实现飞行三维仿真, 对飞行器三维飞行状态实时显示。使用 QtWebEngineView 加载地图, 通过 JavaScript 脚本语言设计实现飞行器导航地图。仿真案例表明设计的仿真平台操作方便, 功能完善, 可扩展性强。

关键词: QtQuick; 容错控制; 飞翼飞行器; 可视化仿真; FlightGear; 百度地图 API

中图分类号: TP391.9 文献标识码: A 文章编号: 1004-731X (2018) 10-3732-07

DOI: 10.16182/j.issn1004731x.joss.201810017

Design of Fault Tolerant Control Visual Simulation System For Flying Wing Aircrafts

Zhang Shaojie, Meng Qingkai

(College of Automation Engineering, Nanjing University of Aeronautics and Astronautics, Nanjing 211106, China)

Abstract: A visual simulation platform based on Qt Quick was designed with respect to the simulation about fault tolerant control for flying wing aircrafts. Aircraft mathematical model and control law were developed using Simulink. Man-machine interface was developed based on Qt Quick to inject fault to Simulink model, load control law, setting up the simulation parameters and process, the interface can monitor and display graphically the key parameters to test and verify fault tolerant control laws. FlightGear visual engine was driven by MATLAB to achieve real-time visual flight simulation and the aircraft were showed in 3-D model. Baidu Map's API function was adopted by QtWebEngineView to achieve the navigation map of the aircraft, which designed by JavaScript. The simulation results show that this simulation platform is convenient, full-featured, and scalable.

Keywords: Qt Quick; fault tolerant; flying wing aircrafts; visual simulation; FlightGear; Baidu Map API

引言

飞翼飞行器具有良好的气动性和隐身性, 是

飞行器发展的重要方向之一, 但其对飞行控制系统的可靠性也提出了更高的要求。操纵面作为飞行器的执行机构, 长期频繁地执行任务, 是系统中最易发生故障的部件之一, 解决飞翼飞行器的操纵面故障控制问题, 是飞行控制律设计研究人员面临的难题^[1]。在控制律设计仿真过程中, 为了问题描述的直观性, 以及辅助验证飞行器容错控制律设计



收稿日期: 2017-05-22 修回日期: 2017-08-06;
基金项目: 国家自然科学基金(61473147);
作者简介: 张绍杰(1978-), 男, 河南平顶山, 博士, 副教授, 研究方向为非线性系统控制和容错控制; 孟庆开(1993-), 男, 满族, 辽宁抚顺, 博士生, 研究方向为非线性系统控制和容错控制。

<http://www.china-simulation.com>

• 3732 •

的效果, 开发可视化仿真平台具有重要的意义。

飞行控制系统可视化仿真平台主体可分为视景系统和模型驱动端 2 个部分。早期的飞行控制系统可视化仿真的视景系统采用底层的三维图形建模应用接口技术, 这种方法虽然能实现 3D 多媒体飞行仿真, 但是开发过程繁琐, 开发周期较长, 3D 效果欠佳。文献[2]采用 DirectX 技术针对无尾式无人机开发了实时飞行仿真系统, 文献[3]在 Vega Prime 中引入 OpenGL 设计了基于 C++ 编程的可视化仿真平台。随着专业模拟飞行软件的发展, 尤其是以 X-plane、FlightGear 等较为出色的模拟飞行软件的成功应用, 视景系统转向对模拟飞行软件的二次开发, 这种方法开发周期短, 可视化仿真效果逼真。文献[4]使用 X-plane 作为视景系统开发了一种自动着陆仿真平台, 取得了较好的仿真效果, 但由于 X-plane 是一款闭源商业软件, 对其二次开发存在诸多限制。FlightGear 作为一款开源的模拟飞行软件, 其扩展性强, 移植性好, 是进行科研和学术研究的较好选择^[5-6]。

模型驱动端可以是半实物模型^[4,7], 也可以由软件编程实现, 但这种方法较为繁琐和不实用, 调试时间长, 修改起来较为麻烦。目前较为受欢迎的方式是使用仿真软件搭建数学模型, 开发实时仿真系统^[8-9], 尤其是使用 MATLAB 仿真软件, 这种方式开发的模型端扩展性强, 实时性好, 而且相关建模技术成熟, 适用范围广。

为了实现实时仿真, 提高仿真系统的可操作性和功能可扩展性, 合理科学地验证控制律, 本文采用 Simulink 搭建模型端, 以 FlightGear 作为视景仿真软件, 开发了一套基于 Qt Quick 的飞翼飞行器容错控制可视化仿真平台。

1 开发方案

1.1 飞翼飞行器操纵面故障分析

本文研究的飞行器的操纵面配置如图 1 所示。所有的操纵面置于齿状的“双 W”机翼后缘, 包括三

对升降副翼、一对阻力方向舵和一块“海狸尾”舵面。

升降副翼是飞翼飞行器引进的新的控制舵面, 兼具升降舵和副翼的功能。该飞翼飞行器设置了三对升降副翼, 其中左右对称安装在最内侧的一对升降副翼作为升降舵使用, 控制俯仰运动; 靠近升降舵外侧的一对升降副翼作为余度舵面, 根据操纵需求而改变控制作用; 靠近余度舵面外侧的一对升降副翼作为副翼使用, 控制滚转运动。飞行器的航向运动由最外侧的阻力方向舵的开合实现。

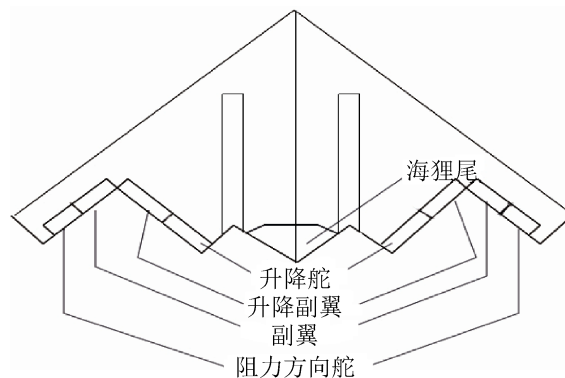


图 1 飞翼飞行器操纵面配置示意图

Fig. 1 Assignment of flying wing aircrafts' control surfaces

本文考虑的执行器故障类型包括卡死故障、部分/完全失效故障以及它们的组合故障^[10]。执行器故障形式表示为

$$\begin{aligned} u_i &= \lambda_i u_{ci}(t) + \bar{u}_i \\ \text{Rank}(\text{diag}\{\lambda_i, \bar{u}_i\}) &\leq 1 \\ t &\geq t_i, 0 \leq \lambda_i \leq 1, i=1, \dots, m \end{aligned} \quad (1)$$

式中 u_i 是第 i 个执行器的实际输出; u_{ci} 是控制器对第 i 个执行器的控制输入; $\lambda_i \in R$ 表示系统第 i 个执行器发生部分失效故障时有效部分的比例; $\bar{u}_i \in R$ 表示第 i 个执行器的卡死故障的卡死位置。式(1)可以表示所有执行器卡死、部分失效的组合故障情况:

当 $\bar{u}_i = 0$, $\lambda_i = 0$, 第 i 个执行器完全失效;

当 $\bar{u}_i = 0$, $0 < \lambda_i < 1$, 第 i 个执行器部分失效, 如 $\lambda_i = 0.6$ 时, 执行器的失效率为 40%;

当 $\bar{u}_i = 0$, $\lambda_i = 1$, 第 i 个执行器正常;

当 $\bar{u}_i \neq 0$, 第 i 个执行器卡死在 \bar{u}_i 度。

1.2 平台架构

仿真平台的开发架构如图2所示。Simulink模型端驱动容错飞行控制可视化仿真平台；通过Qt Quick设计模型控制界面实现故障注入、控制律选择和模型仿真运行控制；开发航空仪表、航空驾驶设备、故障显示窗口以及曲线打印界面直观的显示相关参数。以FlightGear作为飞行三维模拟器展示飞机实时飞行姿态，调用百度地图API对飞行器的位置信息进行实时显示。根据上述平台架构开发的仿真平台实物图如图3所示，左侧为百度地图，中间为人机交互界面，右侧为三维视景仿真。

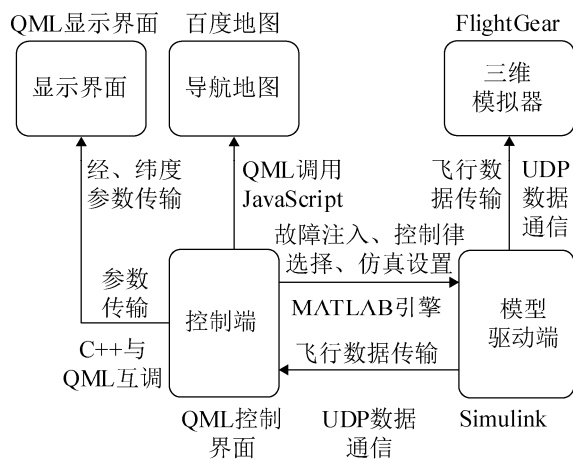


图2 平台架构图

Fig. 2 Framework of the platform



图3 仿真平台实物图

Fig. 3 Simulation platform

2 开发过程

2.1 Simulink 模型端开发

在飞行控制系统仿真领域，MATLAB 尤其是

其提供的 Simulink 工具利用图形窗口技术为建立复杂的仿真系统提供了便利，相关研究较为成熟^[11-13]。考虑到仿真的科学性和数学模型搭建的快速性，本文开发的平台采用 Simulink 搭建数学模型作为飞行器模型端，驱动整个平台的运行。本文以某型飞翼布局飞行器的执行器故障容错飞行控制作为仿真对象，建立 Simulink 模型，Simulink 模型图如图4所示。

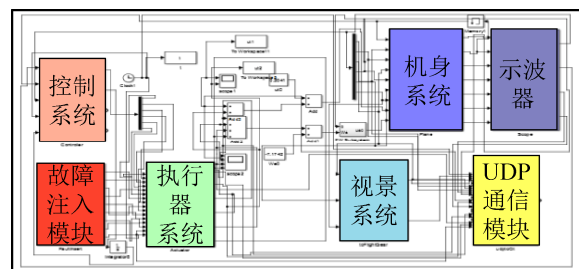


图4 模型端 Simulink 图

Fig. 4 Simulink model of Platform Driver

建立飞翼飞行器的六自由度数学模型作为控制对象；根据式(1)搭建故障注入模块对执行器故障进行设置，本平台中，故障注入模块直接与执行器系统模块相连，为执行器提供式(1)中的系数 λ_i 和 \bar{u}_i ，并在设定的时间完成故障注入，实现对操纵面卡死和部分失效的模拟。控制系统模块与机身系统和执行器模块相连，在固定接口的基础上，采用文[1,14-16]所述控制律，转换控制系统模块的 MATLAB function 代码，以实现平台对不同控制律的可行性验证。设计基于 s-function 的 UDP 通信模块，作为连接控制端的输出接口。设计视景系统实现飞行器模型的三维可视化飞行。

2.2 基于 Qt Quick 的控制端开发

Qt Quick 是 Qt 4.7 版本之后推出的一种高级用户界面新技术，其由两部分组成^[17]，首先是 QML 语言，用于快速构建用户界面；其次是 QtDeclarative 模块用于执行 QML 组件以及将它们与 C++代码整合。虽然 Qt Quick 技术的引入，使得我们能够快速构建艺术级的图形用户界面，但它存在很多局限性^[18]，传统 Qt 的一些技术，在 QML 中要么不可

用, 要么使用不便, 在调用其他软件及通信时, 存在诸多不便。故本文在平台开发时, 各取所长, 采用 QML 和 C++ 混合编程: 使用 QML 构建界面, 使用 C++ 实现非界面的业务逻辑和复杂运算。

将 C++ 类注册为 QML 环境类型的语句如下:
qmlRegisterType<Matlabctr>("an.qt.Matlabctr", 1, 0, "MatlabCtr");

这个语句将自定义的控制 MATLAB 引擎的 C++ 类 Matlabctr 注册为 QML 类 MatlabCtr, 封装的包名为 an.qt.Matlabctr, 主版本号为 1, 次版本号为 0。在 QML 中使用注册的 C++ 类需要在文件头部引用注册时规定的包, 语句如下:

```
import an.qt.Matlabctr 1.0
```

在 C++ 中通过元对象系统调用 QML 对象的语句如下:

```
QObject *AviationInstrument=root->findChild<QObject*>("AviationInstrument");
```

```
AviationInstrument->setProperty("PitchAngle", angle);
```

```
QMetaObject::invokeMethod(AviationInstrument, "reload");
```

第一句用于查找根目录的子对象 AviationInstrument, 这个子对象在 QML 中通过 ObjectName 属性定义。二、三句分别设置该对象的属性 PitchAngle, 以及调用该对象内定义的 reload 函数。

按照上述 C++ 与 QML 混合编程的机制, 完成模型控制端开发。菜单栏和控制面板用于完成模型相关操作, 如故障注入、仿真时间设置、控制律选择和加载等。操纵面故障指示窗口对飞行器的操纵面故障类型及各舵面的偏转角度进行显示, 无故障情况下舵面颜色为绿色; 当舵面发生卡死故障时, 故障舵面颜色为红色; 失效舵面故障为黄色; 舵面上实时显示当前舵面偏角度数。虚拟仪表指示飞机的姿态角以及空速、高度。控制输入设备窗口, 包括驾驶盘、脚踏和油门杆, 用于指示控制输入量。其右侧的飞行参数列表实时显示当前的各种飞行

参数。曲线打印窗口此时打印出空速、俯仰角、高度随时间变化的曲线。

2.3 MATLAB 与 Qt 通信的实现

MATLAB 提供了灵活而又强大的接口技术^[19], 这也为其与 Qt 相互调用实现通信提供了便利。本文通过 C++ 与 MATLAB 混合编程技术, 以调用 MATLAB 引擎的方法实现对 Simulink 模型参数的设置和对仿真过程的控制。用 C++ 编写 s-function 模块开发通信服务端用于发送数据, 在 Qt 中编写通信客户端用于接收数据, 囿于飞行数据的多样性和大量性, 采用简单高效的 UDP 协议进行通信。

在 Qt 中使用 MATLAB 引擎需要在 Qt 工程中添加 MATLAB 引擎的头文件和库函数路径, Qt Creator 调用外部库设置是在工程中的 pro 文件添加以下语句, \$dir\$ 为 MATLAB 的安装路径。

```
INCLUDEPATH += $dir$\extern\include
```

```
LIBS += $dir$\extern\lib\win64\microsoft\libmex.lib
```

```
LIBS += $dir$\extern\lib\win64\microsoft\libmx.lib
```

```
LIBS += $dir$\extern\lib\win64\microsoft\libeng.lib
```

2.4 基于 FlightGear 和 Simulink 的可视化仿真

FlightGear 作为一款优秀的开源飞行模拟器, 其预留了多种输入输出接口, 方便用户对其功能进行扩展。在 Simulink 的 AeroSpace 工具箱中提供了一个基于 FlightGear 官方发布的传输协议开发的 UDP 飞行仿真数据通信接口^[20], 可用于实现三维画面显示。本文中, 使用该接口封装成视景系统实现某飞翼布局飞行器的飞行姿态显示。

为了使该坐标在 FlightGear 中适用, 必须将由欧式几何得到的坐标系转换为大地坐标系, 通用的坐标系为 World Geodetic System(WGS), 本文使用 WGS 84 版本的大地坐标系, 这种坐标系主要的椭球参数^[9]如下, 长半轴为 6, 378, 137.0 m, 短半轴为 6,356,752.314245 m, 扁率为 298.257 223 563。

飞行器的初始位置选择也是很重要的。本文中，选择旧金山机场(KSFO)作为初始位置，机场的经纬度分别为 122.3870 W, 37.6162 N。

2.5 Qt 调用百度地图 API

百度地图 API 是由百度公司开发的基于百度地图服务的应用程序接口，其全部由 JavaScript 语言编写而成。目前常见的百度地图嵌入 Qt 程序多是使用 C++ 语言开发^[21-22]，在移动端百度地图的载体可以选择使用浏览器或者客户端，但在 PC 端百度地图只能由浏览器加载。本文平台开发中使用的百度地图使用 QML 网页引擎加载 HTML 文件，并由 QML 直接调用 JavaScript 完成二者通信。因为同为描述型语言，QML 除了提供网页调用引擎外，其语法结构还部分继承了 JavaScript 语法规则，避免了使用 C++ 与 JavaScript 复杂繁琐的混合编程。

使用百度地图的服务，需使用百度坐标系 (BD09 坐标)。若使用非 BD09 坐标、未经过坐标转换直接叠加在地图上，地图展示位置会偏移，因此通过其他坐标调用服务时，需先将其他坐标转换为 BD09。网页版百度地图 API 提供调用服务器端和在 JS 的前端网页中转换坐标的功能，本文采用在前端网页转换坐标的方式。百度地图提供如下方法完成其他坐标向百度坐标的转换，其中 pointArr 保存原始坐标点；from 位为 1，表示源坐标类型为 WGS84 坐标；to 位为 5，表示目的坐标类型为百度经纬坐标；在 translateCallback 中定义坐标转换完成后的回调函数，本文中回调函数将代表飞行器位置信息的飞机图标定位在百度地图上。

convertor.translate(pointArr, 1, 5, translateCallback); WebEngineView 是 QML 提供的一个组件，用于查看和编辑网络文档。将保存有百度地图相关配置的网页文件 map.html 使用 WebEngineView 加载，并使用 runJavaScript() 方法调用 html 中由 JavaScript 所定义的函数，实现地图的导航功能。

3 仿真案例

利用本仿真平台对本实验室提出的相关容错控制方法研究成果^[1, 14-16]进行验证，同时说明本仿真平台的有效性和可行性。以文献[1]所述方法为例，具体说明仿真过程。

启动容错飞行控制可视化仿真平台，完成 MATLAB、FlightGear 和百度地图初始化，在控制面板故障注入界面设置左升降舵卡死在 -6° ，左升降副翼失效系数为 0.6，故障注入时刻为 10 s，仿真时间为 100 s；分别加载不具有容错控制功能的传统控制律和文献[1]所述的反步自适应控制律，运行模型，设定参考跟踪信号，验证控制效果。

该飞翼飞行器的巡航高度为 11 000 m，巡航速度为 0.6 马赫，配平的迎角为 4° ，俯仰角为 4° ，油门开度之和为 0.3041，升降舵偏角之和为 -7.1742° 。参考速度信号为 $V_r = 130 + 20\cos(0.1t)$ ；参考俯仰角信号为 $\theta_r = 18 + 10\sin(0.1t)$ 。

如图 5 所示，分别为相同视角下两种控制律在故障发生时的视景效果图。受左侧升降副翼卡死故障影响，左图所示传统控制律作用下的飞行器发生了滚转，无法正常跟踪给定信号。而文献[1]所述容错控制律在舵面故障下仍能较好跟踪给定参考信号，实现操纵面故障下的容错控制飞行。图 6 为文献[1]所述的容错控制律仿真结束后的人机交互界面。

飞行器舵面状态指示窗口标识出左升降舵卡死在 -6° ，左侧升降舵发生失效故障。从右侧速度响应曲线和俯仰角响应曲线可以看出，在该容错控制律控制下，发生故障的飞翼飞行器仍能跟踪给定参考信号，验证了控制律的有效性。

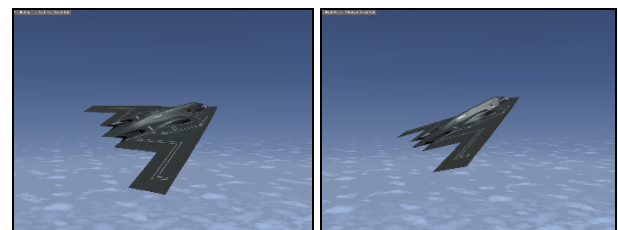


图 5 故障下两种控制律的视景效果图

Fig. 5 Visual renderings of two control laws under faults

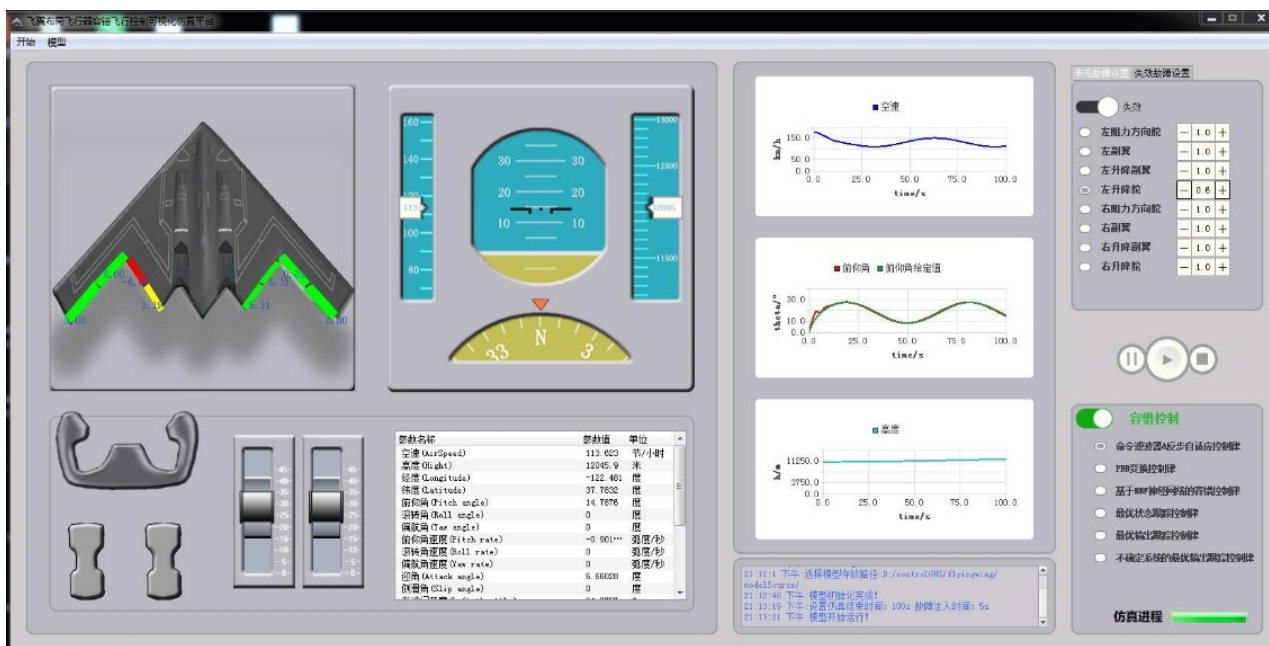


图 6 仿真结束后的界面
Fig. 6 The interface of simulation

4 结论

本文研究了基于 Qt Quick 的容错飞行控制可视化仿真平台及其开发方法, 为多软件联合仿真提供了一种开发思路。本平台多软件协同通信组成了功能完善的飞行控制系统可视化仿真平台。本平台软件界面美观、人机交互操作便捷, 软件间通信和系统控制满足实时性要求。通过某型飞翼布局飞行器的操纵面故障容错飞行控制仿真实例表明, 本平台能够很好地展现飞行仿真中的飞行器控制设备输入、故障状况、飞行航迹、姿态等相关信息, 且使用方便、易于维护。本平台的扩展性强, 经修改可用于车辆、船舶、导弹等可视化仿真, 有着广阔的应用前景。

参考文献:

[1] Shuang W, Zhang S, Wu X, et al. An Anti-windup Fault Tolerant Control Scheme with Guaranteed Transient Performance for Tailless Flying Wing Aircrafts [C]// AIAA Guidance, Navigation, and Control Conference. Gaylord Texan USA: AIAA, 2017-1253.
[2] 刘泽坤, 昂海松, 罗东明. 基于 DirectX 的无人机实时飞行仿真系统开发 [J]. 系统仿真学报, 2006, 18(4):

918-920.

Liu Zekun, Ang Haisong, Luo Dong-ming. Development of Real-time Flight Simulation System Based on DirectX for Unmanned Aerial Vehicle [J]. Journal of System Simulation, 2006, 18(4): 918-920.

[3] 陈力威, 朱凡, 刘希, 等. Vega Prime 与 OpenGL 飞行控制系统可视化仿真平台设计 [J]. 火力与指挥控制, 2012, 37(8): 191-194.

Chen Liwei, Zhu Fan, Liu Xi, et al. Visual Simulation Platform Design of Flight Control System Based on VegaPrime and OpenGL [J]. Fire Control & Command Control, 2012, 37(8): 191-194.

[4] Seo Y, Min C, Lee D. Development and Flight Test of Auto-landing System Using X-Plane Simulator [C]// Proceedings of KSAS Fall Meeting. Korea: KSAS, 2011: 435-440.

[5] 应进, 潘浩曼, 代冀阳, 等. Matlab/FlightGear 直升机视景仿真研究 [J]. 实验技术与管理, 2014, 31(8): 106-109.

Ying Jin, Pan Haoman, Dai Jinyang, et al. Research on visual flight simulation based on Matlab/FlightGear [J]. Experimental Technology and Management, 2014, 31(8): 106-109.

[6] 王仓库. 基于 FlightGear 的无人机编队空战可视化系统研发 [D]. 沈阳: 沈阳航空航天大学, 2016.

Wang Cangku. Research and development of FlightGear-Based UAVS formation combat visualization

- system [D]. Shenyang, China: Shenyang Aerospace University, 2016.
- [7] 臧皖晋. 基于 FlightGear 的四旋翼无人机半实物仿真系统 [J]. 工业控制计算机, 2016, 29(8): 12-14.
Zang Huanjin. Hardware-in-the-loop Simulation System for Quadrotor UAV Based on FlightGear [J]. Industrial Control Computer, 2016, 29(8): 12-14.
- [8] 郭卫刚, 韩维, 王秀霞. 基于 Matlab/Flightgear 飞机飞行性能的可视化仿真系统 [J]. 实验技术与管理, 2010, 27(10): 110-112.
Guo Weigang, Han Wei, Wang Xiuxia. Visual simulation system of flight performance based on Matlab/FlightGear [J]. Experimental Technology and Management, 2010, 27(10): 110-112.
- [9] Nusyirwan I F. Engineering Flight Simulator Using MATLAB, Python and Flightgear [C]// Simtect 2011 Conference. Melbourne, Australia: SimTecT, 2011.
- [10] 邱相玮. 一类MIMO非线性系统的执行器故障自适应补偿控制 [D]. 南京: 南京航空航天大学, 2014.
Qiu Xiangwei. Actuator Fault Adaptive Compensation Control for a Class of MIMO Nonlinear System [D]. Nanjing, China: Nanjing University of Aeronautics and Astronautics, 2014.
- [11] 戴宁, 司云玲. 基于MATLAB的直升机飞行控制系统建模与仿真 [J]. 系统仿真学报, 2006, 18(1): 240-242.
Dai Ning, Si Yunling. Modeling and Simulation of helicopter flight control system based on MATLAB [J]. Journal of System Simulation, 2006, 18(1): 240-242.
- [12] 陈斌, 王力. 基于MATLAB的飞行控制实时仿真系统研究 [J]. 微计算机信息, 2007, 23(5S): 295-296.
Chen Bin, Wang Li. Study on the real-time simulation system of flight control based on MATLAB [J]. Microcomputer Information, 2007, 23(5S): 295-296.
- [13] 秦啸, 陈红林, 曲艺海. 基于 SIMULINK 的飞行控制系统设计与仿真 [J]. 系统仿真学报, 2007, 19(21): 5030-5033.
Qin Xiao, Chen Honglin, Qu Yihai. Design and Simulation of Fly Control System Based on SIMULINK Toolbox [J]. Journal of System Simulation, 2007, 19(21): 5030-5033.
- [14] Wu X, Zhang S, Shuang W, et al. Optimal adaptive compensation control for a class of MIMO nonlinear systems with actuator failures [C]// UKACC International Conference on Control. Belfast UK: IEEE, 2016: 1-6.
- [15] 张绍杰, 邱相玮, 刘春生, 等. 基于 MMST 分组的一类 MIMO 非线性系统执行器故障自适应补偿控制 [J]. 自动化学报, 2014, 40(11): 2445-2455.
Zhang Shaojie, Qiu Xiangwei, Liu Chunsheng, et al. Adaptive Compensation Control Based on MMST Grouping for a Class of MIMO Nonlinear Systems with Actuator Failures [J]. Acta Automatica Sinica (S0254-4156), 2014, 40(11): 2445-2455.
- [16] Zhang Shaojie, Qiu Xiangwei, Jiang Bin. Adaptive actuator failure compensation control based on MMST grouping for a class of MIMO nonlinear systems with guaranteed transient performance [J]. International Journal of Control (S0020-7179), 2015, 88(3): 593-601.
- [17] Thelin J. Quick user interfaces with Qt [J]. Linux Journal (S1075-3583), 2011(204): 71-74.
- [18] 安晓辉. Qt Quick 核心编程 [M]. 北京: 电子工业出版社, 2015: 228-240.
An Xiaohui. QtQuickCore Programming [M]. Beijing, China: Publishing House of Electronics Industry, 2015: 228-240.
- [19] 张亮, 王继阳. MATLAB 与 C/C++混合编程 [M]. 北京: 人民邮电出版社, 2008.
Zhang Liang, Wang Jiyang. Mixed programming of MATLAB and C/C++ [M]. Beijing, China: People's Posts and Telecommunications Press, 2008.
- [20] The Mathworks. Aerospace Blockset for Simulink, Matlab User's Guide [M]. USA: The MathWorks, Inc, 2012.
- [21] 席志红, 李俊文. 基于 BOA 服务器和百度 API 的北斗定位地图的设计 [J]. 应用科技, 2016, 43(6): 43-45.
Xi Zhihong, Li Junwen. Design of the Beidou positioning map based on BOA server and Baidu API [J]. Applied Science and Technology, 2016, 43(6): 43-45.
- [22] 赵意鹏, 赵河明. 基于 GPRS 和百度地图 API 的火灾定位系统 [J]. 自动化与仪表, 2016(1): 26-29,47.
Zhao Yipeng, Zhao Heming. Fire Positioning System Based on GPRS and Baidu Map API [J]. Automation & Instrumentation, 2016(1): 26-29,47.