

DOI: 10.19452/j.issn1007-5453.2017.04.057

直升机飞行载荷测量数据分析软件的控制开发

吴堂珍^{1,*}, 林斌², 曾玖海¹

1. 中国直升机设计研究所, 江西 景德镇 333001

2. 海军驻景德镇地区航空军事代表室, 江西 景德镇 333001

摘要: 实测载荷是直升机疲劳设计与评定的关键数据。由于直升机实测载荷具有通道多、频率高、数据量大等特点, 如何实现对实测数据高效快速处理显得十分关键。本文提出了一种用于直升机飞行载荷测量数据分析软件的控制程序开发技术, 包含二维曲线及三维瀑布图控件, 适用于大数据量的显示分析, 可实现多条二维曲线并列显示及数据点的游标读取, 不同视角的三维瀑布图旋转, 具有易用, 高效等特点。

关键词: 直升机; 飞行载荷; 控件开发; 测量数据

中图分类号: V217.32

文献标识码: A

文章编号: 1007-5453 (2017) 04-0057-06

直升机是一种带旋翼的航空飞行器。旋翼系统(旋翼和尾桨)在飞行中高速旋转,造成不对称流场作用产生的高频振动与耦合载荷^[1]。飞行载荷试验测试是一项大型的试验项目,在整个直升机研制中至关重要,直接影响动部件的寿命评定结果。

直升机飞行载荷测量采样频率较高、通道多、飞行时间长、数据量大,采用之前的一些自编软件,速度较慢,效率不高,难以满足快速数据分析、及时发现问题的需求,许多专业软件功能虽然强大,但是针对性不强,不能自如控制。在进行不同工况时间段确定时,需要在一个图上并列显示多条不同量纲的曲线,对曲线任意组合,动态增减,用游标读取数据点;在观察一定时间范围内两个参数(如频率和幅值)的变化时,需要能采用不同视角对曲线旋转的三维瀑布图。

1 基本思路

控件界面如图1所示。二维曲线控件要求使用简单,能够动态创建、销毁,最小的内存占用,最快的曲线显示速度。数据点和曲线的数量取决于硬件限制。具备以下功能:

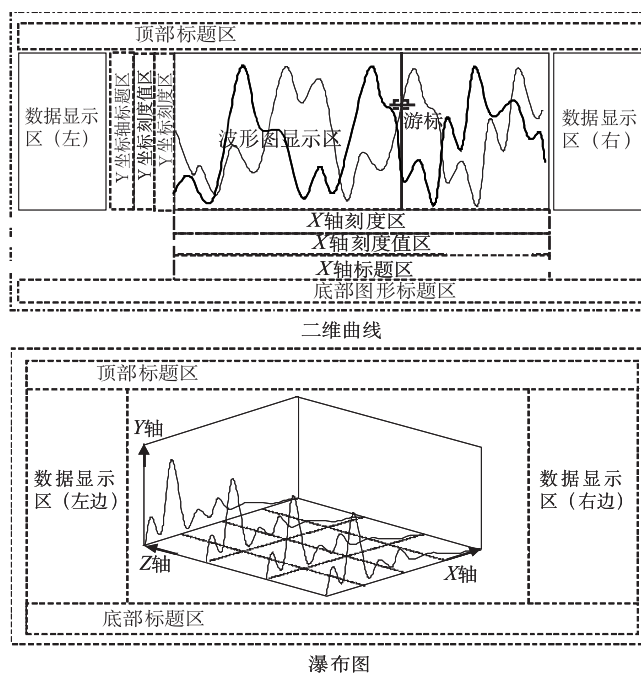


图1 控件界面
Fig.1 The control interface

收稿日期: 2017-01-18; 退修日期: 2017-02-17; 录用日期: 2017-02-27

* 通讯作者. Tel.: 022-59800361 E-mail: 280946919@qq.com

引用格式: WU Tangzhen, LIN Bin, ZENG Jiuhai. Control exploitation of analyzing measured helicopter flight load measured data [J]. Aeronautical Science & Technology, 2017, 28 (04): 57-62. 吴堂珍, 林斌, 曾玖海. 直升机飞行载荷测量数据分析软件的控制开发 [J]. 航空科学技术, 2017, 28 (04): 57-62.

能用鼠标和键盘对曲线和游标进行灵活操作;能对图形各属性详细设置;能根据字体、刻度、控件大小自动调整显示;能动态增减曲线;每条曲线采用独立 Y 坐标,并以颜色区分,通过双击 Y 坐标区进行坐标切换;能实现多个控件之间的协调显示;能显示多个游标,可直接用游标选读单点数据;图像可复制到剪贴板;曲线数据可提取保存到文件;能实现曲线打印、回放、警戒线显示等。

三维瀑布图控件要求能用鼠标对图形进行三维旋转;能对图形各属性详细设置;快速的图形显示,曲线打印,图像可复制到剪贴板。

控件编程采用 Microsoft VC++6.0^[2]。主要采用了如下 C++ 类:

二维曲线控件类 CChart,瀑布图曲线控件 CWtfChart,均派生于 MFC 的 CWnd 类,可在任意视图或窗口的某一个矩形区上动态创建、销毁,这也是采用 CWnd 类派生最大的优点,可以灵活应用到软件中,方便鼠标和键盘消息的处理。负责图形区标题、刻度线、刻度值、曲线的绘制,游标显示,鼠标、键盘的响应^[3]。

二维曲线类 CCurve,派生于 MFC 的 CObject 类,可用于数据链表结构,能动态添加和删除。负责数据的处理、运算、曲线绘制^[4]。

三维曲线类 C3DCurve,派生于 CCurve,同样可以动态添加和删除。负责三维曲线的旋转、显示。

游标类 CSlider,派生于 MFC 的 CObject 类,可用于数据链表结构,能动态添加和删除。负责游标的显示^[2-4]。

长方体类 CCuboid,用于三维坐标轴的旋转控制、显示。

2 关键技术

2.1 速度优化

对于速度优化,可以采取 3 个措施,减少内存占用,减少绘图量和使用缓存。

措施 1: 利用数据等时间间隔特点,减少一般内存占用。

通常需要给每一个数据点的 X 、 Y 两个坐标值分配内存,由于数据点是按同样的采样频率采集,每一个数据点的时间间隔一样,实际上只需给 Y 坐标分配内存,因为只要知道一段数据的开始时刻 t (s) 和采样频率 p (Hz),就可以根据数据点的个数 n 算出 X 坐标值 (s),如式 (1) 所示,这样内存的消耗就可以减少一半。

$$x = t + \frac{n}{p} \quad (1)$$

措施 2: 通过绘图优化算法减少大量的重复线段绘制。

对于绘图量,假设一条曲线有 n 个数据点,则需要绘制 $n-1$ 个线段,由于用 GDI 函数绘图的速度很慢,导致绘图速度慢,能否对此进行优化是关键。假如绘图区域的宽度为 m 个像素,那么每一列像素将有 n/m 个数据点。当数据量很大时,在一列像素上重复绘制了很多线段。经分析,可以简化为 4 个点进行绘图,如图 2 所示。依次将上一列像素离开点,当前列像素进入点、最大点、最小点、离开点,下一列像素进入点连成线段,得到的图形与原始绘图效果是完全一致,总的绘图点数为 $m \times 4$,线段数 $m \times (4-1)$,且不论曲线放大到单个点还是缩小到整个绘图区均不会有任何失真。例如,有 1000 万个数据点需要绘制在 1000 列宽像素区域,原始方法需要绘制 1000 万减 1 条线段,优化后只需要绘制 3999 条线段,省掉了 99.96% 的绘图量,速度提升 2000 多倍,数据量越大优势越明显,代价是绘图之前需要计算每一列像素的 4 个点,由于 CPU 运算速度快,计算用的时间只需几毫秒,基本可以忽略,而传统方法需要 10 多秒时间。

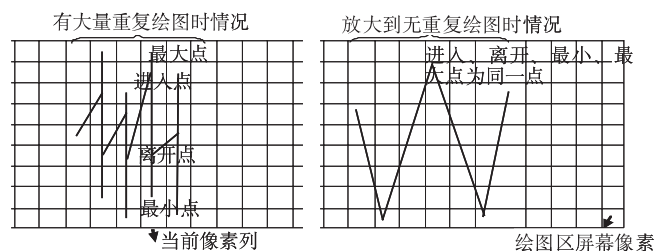


图 2 曲线绘图优化示意

Fig.2 The curve drawing optimization schematic

具体方法是先将数值点转换到绘图区屏幕像素点 (同时进行转换和优化效率更高),然后找出处于同一列像素的一组坐标点,得到该列像素的进入、离开、最大、最小点。

措施 3: 在内存中绘图。

另一个提高速度的常用方法是将图形先绘制到内存上,再一次性将图像拷贝到屏幕,由于在内存上绘图较快,可以提高绘图速度,尤其是可以避免绘图时的闪烁,代码如下所示。

```
void CChart::OnPaint()
{
    CRect rectCtr;
    GetClientRect(&rectCtr); // 得到绘图区
    CPaintDC dc(this); // 创建绘图设备指针
    CBitmap memBitmap; // 创建位图
    CDC memDC; // 创建绘图设备缓存指针
    memDC.CreateCompatibleDC(&dc);
    memDC.CreateCompatibleDC(&dc);
```

```

memBitmap.CreateCompatibleBitmap (&dc, rectCtr.
Width (), rectCtr.Height ());
memDC.SelectObject (&memBitmap);
memBitmap.DeleteObject ();
memDC.SetBkMode (TRANSPARENT);
DrawBackColor (memDC, rectCtr); // 绘制背景到内存
DrawAxis (memDC); // 绘制坐标轴到内存
DrawTitle (memDC); // 绘制标题到内存
DrawCurve (memDC); // 绘制曲线到内存
DrawSlider (memDC); // 绘制游标到内存
.....
dc.BitBlt (rectCtr.left, rectCtr.top, rectCtr.Width (),
rectCtr.Height (), &memDC, 0, 0, SRCCOPY); // 拷贝内存
图像到屏幕
memDC.DeleteDC ();
}

```

还有一个重要的问题,当视图中有许多控件需要滚动显示时,原来被遮住的控件曲线需要重绘,导致滚动时不顺畅,有卡滞的现象。解决的办法是将图像保存在内存中,当曲线没有发生改变时,只需要将内存中的图像拷贝到屏幕上即可,这样就可以流畅地进行滚动显示。

2.2 多曲线

采用 MFC^[4] 提供的 COBArray 类来管理曲线,可以很方便地对曲线动态增加和删除。

由于多条曲线的数值量纲可能不同,因此,每条曲线必须有独立的 Y 坐标,一种方法是多个 Y 坐标轴并列显示,但是会占用绘图区宽度,并且显得混乱;本文采用另一种方法,每次只显示当前曲线一个 Y 坐标轴,通过鼠标双击 Y 坐标轴区来切换显示不同的 Y 坐标轴,不同曲线及 Y 坐标用不同颜色区分。

2.3 游标

游标显示要解决游标与曲线的协调,若每次游标移动时,重绘曲线,将导致图形不停闪烁,速度较慢。可先将没有游标的图像保存在内存中,每次游标移动时,先将内存中图像拷贝到屏幕,再单独画游标。由于曲线和游标独立画图,且游标是数组中数据点换算到屏幕坐标,因此,可以用鼠标或键盘控制游标遍历所有的数据点,不会因为绘图量的优化而受影响。

游标的数量可以是 1 个或 2 个,其中只有一个游标是活动的(可以用 TAB 键或鼠标激活另一个游标),用键盘光

标键只能移动当前活动游标。

2.4 图形旋转和消隐

为了瀑布图的显示需要,构造了一个 C++ 类 Cline 代表的一条信号线,每个需要显示的瀑布图信号对应一个 Cline 对象^[5]。

瀑布图的关键一是旋转,二是消隐。

假设有三维坐标系下的一点 (x, y, z) , 绕 X 轴转 α 角, Y 轴旋转 β 角, Z 轴旋转 γ 角后的点为 (xr, yr, zr) , 应采用式 (2) 矩阵进行转换^[6]:

$$\begin{bmatrix} xr & yr & zr \end{bmatrix} = \begin{bmatrix} x & y & z \end{bmatrix} \times \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \alpha & \sin \alpha \\ 0 & -\sin \alpha & \cos \alpha \end{bmatrix} \times \begin{bmatrix} \cos \beta & 0 & -\sin \beta \\ 0 & 1 & 0 \\ \sin \beta & 0 & \cos \beta \end{bmatrix} \times \begin{bmatrix} \cos \gamma & \sin \gamma & 0 \\ -\sin \gamma & \cos \gamma & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (2)$$

经过实践,用鼠标在屏幕横向和纵向的移动量来控制 α, β 角大小,可以很好地实现瀑布图的简单旋转。

当瀑布图进行旋转时,长方体的绘图有 2 种方法,一种是画透视长方体(如图 3 所示),一种是只画 3 个面(如图 4 所示)。不管哪种方法,在旋转时均存在线面的消隐问题,线面会由可见转为不可见,或由不可见转为可见,并且坐标轴刻度及刻度值转到不可见位置时,应该重新在可见坐标轴上绘图。

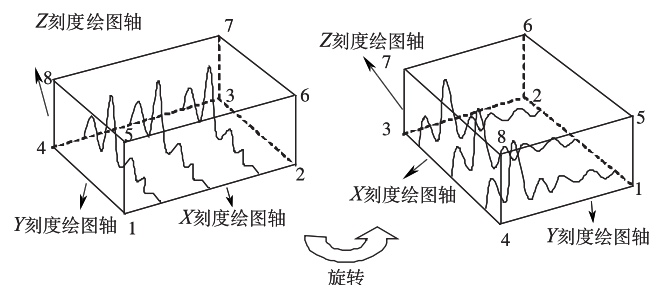


图 3 透视长方体瀑布图的旋转与消隐

Fig.3 The rotation and blinding of perspective rectangular waterfall figure

采用只画 3 个面的方法,更加符合曲线可视的实际情况。当瀑布图处于图 4 中左边位置时,为便于曲线观察,长方体应该剖开,只需画出 3 个面(1-2-3-4, 2-3-7-6, 3-4-8-7),线段 1-5、5-6、5-8 处于视线的前端,不应该画图;当瀑布图处于图 4 中右边位置时,只需画出 3 个面(1-2-3-4, 2-3-7-6,

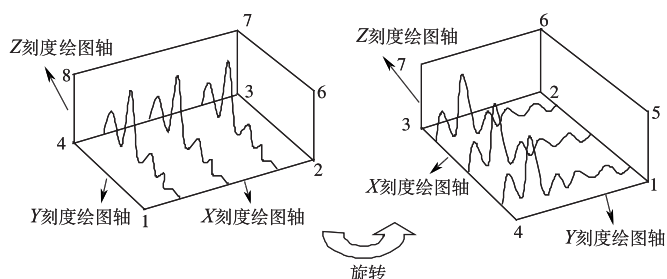


图4 3面瀑布图的旋转与消隐

Fig.4 The rotation and blinding of the 3 faces waterfall chart

1-2-6-5), 线段4-8、7-8、5-8处于视线的前端, 不应该画图。判断的方法是根据长方体8个坐标顶点的 z 坐标相对大小来进行, 从而改变长方体线面及3个坐标轴的绘图。

2.5 控件之间的协同

当视图中同时显示多个控件曲线时, 一般 X 轴为时间轴, 同步缩放, 游标移动也需要同步, 这时需要采用自定义消息, 向父窗口发送 X 轴或游标改变的消息, 当某个控件的曲线 X 轴或游标改变时, 父窗口就可以收到消息, 从而可以实现对所有控件曲线进行同步。

控件类中添加自定义消息如下:

```
#define WM_XAXIS_CHANGED WM_APP+100
```

```
//X轴改变
```

```
#define WM_SLIDER_CHANGED WM_APP+101
```

```
//游标移动
```

```
::PostMessage(GetParent()->GetSafeHwnd(), WM_
```

```
XAXIS_CHANGED, (long) GetSafeHwnd(), 0);
```

```
::PostMessage(GetParent()->GetSafeHwnd(), WM_
```

```
SLIDER_CHANGED, (long) GetSafeHwnd(), 0);
```

在父窗口类相应位置添加如下代码:

```
afx_msg void OnXChange(WPARAM wParam, LPARAM lParam);
```

```
//X轴改变
```

```
afx_msg void OnSliderChange(WPARAM wParam, LPARAM lParam);
```

```
//游标移动
```

```
ON_MESSAGE(WM_XAXIS_CHANGED, OnXChange)
```

```
ON_MESSAGE(WM_SLIDER_CHANGED, OnSliderChange)
```

3 应用效果

二维曲线控件大约完成代码3500行, 函数160个, 其

中, 公共接口函数75个。瀑布图控件大约完成代码1800行, 函数89个, 其中, 公共接口函数38个。

3.1 二维曲线

图5显示多曲线显示及 Y 坐标切换效果, 通过鼠标双击 Y 坐标区, 坐标轴及刻度的改变, 游标的指向也相应发生改变。图6显示应用到软件(视图和对话框)中的效果。

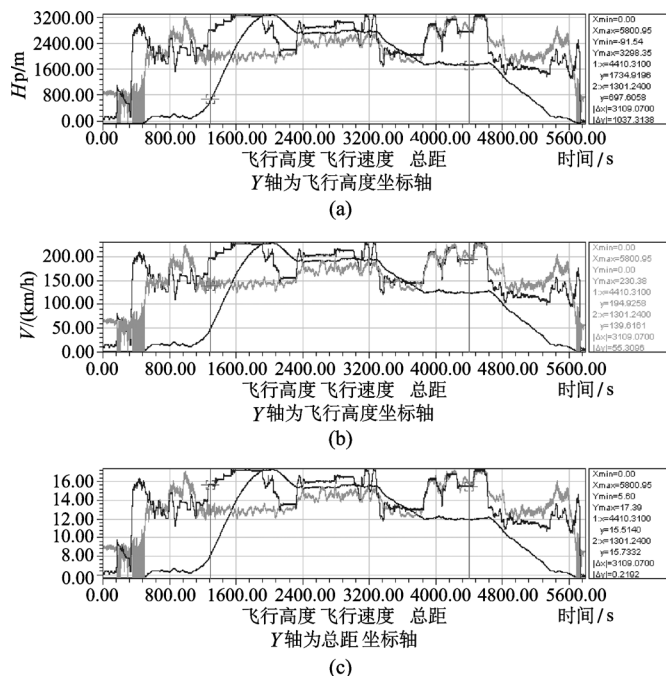
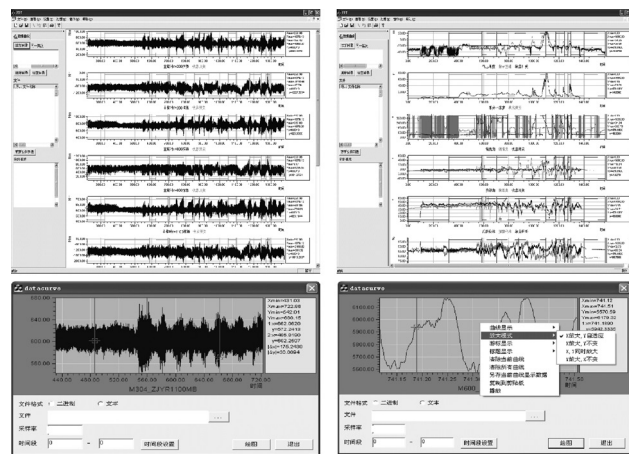
图5 多曲线显示及 Y 坐标切换效果Fig.5 Multi curves displaying and coordinate Y switching effect

图6 控件在软件中的应用

Fig.6 The application of control in software

3.2 瀑布图

瀑布图不同旋转角度及坐标轴改变显示效果如图7所示。

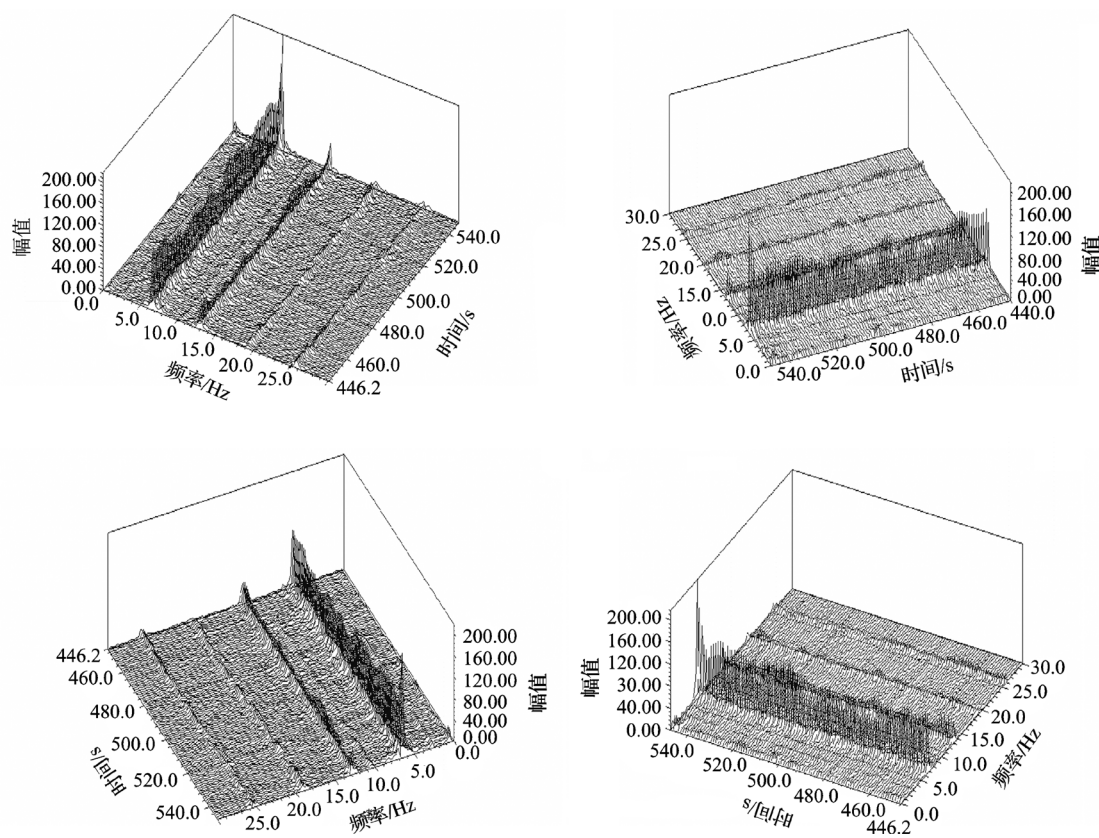


图7 瀑布图

Fig.7 The waterfall chart

3.3 组合应用

二维曲线控件和瀑布图结合应用如图8所示。使用了3个二维图形控件和1个瀑布图控件,第一个二维图形控件显示需要频率分析的测试数据,游标的移动会导致其它所有控件内容变化;第二个显示第三个控件游标位置(时间)一段时间的频率幅值图,会随第三个控件游标移动改变显示内容;第三个显示第二个控件游标位置(频率)下幅值随时间的变化,会随第二个控件游标移动改变显示内容;瀑布图控件显示第一个二维图形控件游标位置(时间)一段时间的时间-频率-幅值变化,可以通过改变第二个二维图形控件X坐标范围(频率)控制瀑布图的频率显示范围,通过改变第三个二维图形控件X坐标范围(时间)控制瀑布图的时间显示范围。

4 结束语

针对直升机旋转部件测试数据特点及分析需求,采取一种高效的绘图优化算法,并结合多种编程技术对数据曲线显示速度、内存占用进行了全方位的优化,实现了速度和效率几千倍的提升,完成了二维曲线显示控件的开发,在此基

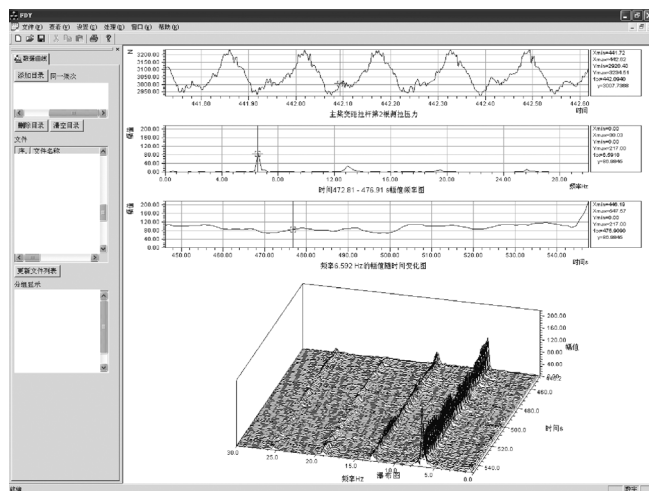


图8 瀑布图控件应用

Fig.8 The waterfall chart control application

础上完成了瀑布图控件的开发,解决了图形的旋转和消隐显示问题,可以很方便地分析一段时间载荷或频率的变化情况,在直升机飞行测试数据分析和飞行载荷测量软件上得到了很好的应用,满足了直升机载荷测试快速、便捷、使用的数据分析需求。

参考文献

- [1] 顾文标, 喻澹鉴, 潘春蛟, 等. 直升机飞行实测载荷有效性分析技术 [J]. 直升机技术, 2007 (4): 19.
GU Wenbiao, YU Jianjian, PAN Chunjiao, et al. Effectiveness analysis of helicopter flight measured load [J]. Helicopter Technology, 2007 (4): 19. (in Chinese)
- [2] Bruce Eckel. Thinking in C++ [M]. 碁峰出版社, 2002.
- [3] 北京博彦科技发展有限公司. Microsoft VC++ 开发教程 [M]. 北京: 清华大学出版社, 2000.
Beijing Beyondsoft Co. Ltd. Microsoft VC++ development tutorial [M]. Beijing: Tsinghua University Press, 2000. (in Chinese)
- [4] 侯俊杰. 深入浅出 MFC (第 2 版) [M]. 武汉: 华中科技大学出版社, 2001.
HOU Junjie. In MFC (Second Edition) [M]. Wuhan: Huazhong University of Science and Technology Press, 2001. (in Chinese)
- [5] 周建川, 李国红. 信号的瀑布分析技术与实现 [J]. 机电工程, 2004, 21 (12): 39-40.
ZHOU Jianchuan, LI Guohong. Waterfall analysis technology and realization of signal [J]. Mechanical Engineering, 2004, 21 (12): 39-40. (in Chinese)
- [6] 魏东, 李冬梅, 黄有群. 三维图形绕坐标轴连续旋转的算法与实现 [J]. 沈阳工业大学学报, 2007 (12): 697.
WEI Dong, LI Dongmei, HUANG Youqun. The algorithm and realization of 3D graphics rotation around axis [J]. Journal of Shenyang University of Technology, 2007 (12): 697. (in Chinese)

作者简介

吴堂珍 (1985—) 女, 硕士, 工程师。主要研究方向: 直升机疲劳强度设计。

Tel: 022-59800361

E-mail: 280946919@qq.com

Control Exploitation of Analyzing Measured Helicopter Flight Load Measured Data

WU Tangzhen^{1,*}, LIN Bin², ZENG Jiu hai¹

1. China Helicopter Research and Development Institute, Jingdezhen 333001, China

2. Navy Aviatric Representative Office in Jingdezhen, Jingdezhen 333001, China

Abstract: Flight test measured load is the key data for fatigue design and evaluation, because the helicopter measured data has many channels, high frequency and large quantities. It is very important of how to deal with the measured data efficiently and quickly. This paper gave a control program development technology used to analyze the helicopter flight load measured data, including the two-dimensional curve and three-dimensional waterfall chart controls. It was applied to display and analyze large data. It could realize the function of showing multiple two-dimensional curves side by side, reading cursor data points and rotating the three-dimensional waterfall charts in different visual angles. It is useful and high efficiency.

Key Words: helicopter; flight load; control exploitation; measured data

Received: 2017-01-18; Revised: 2017-02-17; Accepted: 2017-02-27

*Corresponding author. Tel.: 022-59800361 E-mail: 280946919@qq.com