

TeeChart控件在转台调试软件中的应用

苏长青, 张旭明, 董 恒

(九江精密测试技术研究所, 江西 九江 332000)

✉suchangqing6666@163.com; zhangxuming@163.com; dongheng@163.com



摘 要: 为满足转台调试过程中调试数据的图形化显示需求, 通过在调试软件中应用TeeChart绘图控件将采集到的数据以实时曲线的方式进行显示, 利用TeeChart的绘图事件接口, 通过自定义函数实现了曲线块的最大值、最小值、平均值、标准差等统计计算功能, 并给出相关函数的实现代码, 对数据的实时曲线显示有一定的参考价值。

关键词: TeeChart; 控件; 转台

中图分类号: TP319 **文献标识码:** A

Application of TeeChart Control in Turntable Debugging Software

SU Changqing, ZHANG Xuming, DONG Heng

(Jiujiang Precision Measuring Technology Research Institute, Jiujiang 332000, China)

✉suchangqing6666@163.com; zhangxuming@163.com; dongheng@163.com

Abstract: In order to meet the graphical display requirements of the debugging data during the debugging of the turntable, this paper proposes to apply the TeeChart drawing control in the debugging software and display the collected data in the form of real-time curves. The TeeChart drawing event interface and the custom function are used to realize the maximum value, minimum value, average value, standard deviation and other statistical calculation functions of the curve block. The implementation code of the relevant function is given, which has a certain reference value for the real-time curve display of the data.

Keywords: TeeChart; control; turntable

1 引言(Introduction)

在转台调试过程中, 经常需要分析调试过程中某些数据的变化, 如果调试数据能以图形化的方式进行显示, 并分析计算调试数据的最大值、最小值、平均值、标准差等统计特性, 往往能极大地方便调试过程^[1-2]。比如在转台控制系统PID参数的调试过程中, 通常是产生一个阶跃信号, 观察控制系统对该信号的响应情况, 这就需要调试软件能够将阶跃信号和阶跃响应以曲线的形式同时显示出来, 调试人员根据阶跃响应曲线的上升时间、调整时间、超调量等修改PID参数, 最终得到符合要求的阶跃响应曲线, 完成转台控制器的性能调试^[3]。

在设计转台调试软件时, 通过调用Windows系统自带绘

图函数的方法来实现实时曲线, 不但费时费力, 而且效果往往不尽如人意。而专门为图表应用开发的软件库, 往往通过简单的编程接口调用, 即可实现复杂的绘图功能, 使用起来极其简单, 并且在绘图效率上做过专门的优化, 非常符合实时曲线绘制的需求。因此, 转台调试软件选择基于ActiveX技术的图形控件TeeChart进行实时图形显示, 并利用开放的事件接口完成统计计算的功能。

2 TeeChart特性(TeeChart features)

TeeChart是Steema Software公司开发的图表图形组件, 其Windows版本基于ActiveX技术开发, 用户可以通过Visual Studio集成开发环境进行调用, 支持Visual Basic、Visual C++等编程语言, 可以方便地集成到用户程序中。TeeChart

图形类型支持线条图、棒形图、面积图、饼形图和其他形状图等,并且支持2D和3D风格的图形图表绘制,提供了图形化的用户配置界面及用户编程接口,用户甚至只需要几行代码就能完成复杂图表的显示,大大提升了图形化软件的开发效率^[4-8]。

TeeChart控件的主类是TChart,数据以Series的形式组织, Series通过TChart向用户展示,而Axes是TChart和Series之间的接口。TeeChart可对大多数值进行最佳估计,以尽量减少手动定义图表和系列方法的需要^[3]。下面简要介绍TChart类的一些数据成员。

TChart.Series: 序列(Series)是TeeChart中数据的组织形式,用户可以在图表中插入多个序列,并且每个序列可以有不同的图形类型,如线条图、棒形图、饼形图,等等。

TChart.Axes: 坐标轴(Axes)控制图表坐标轴的属性。TeeChart支持根据数据内容自动调整坐标轴的范围,也支持手工调整,利用坐标轴可以方便地实现图形的缩放和平移等操作。

TChart.Legend: 图例(Legend)控制图标的图例标注。图例通过在图形中增加一个矩形区域,可以对序列进行图例标注,也可以控制序列的显示与隐藏。

TChart.Panel: 面板(Panel)可以控制图表的背景。TeeChart支持用户使用背景图片或者渐变色彩等作为图形绘制的背景。

TChart.Canvas: 画布(Canvas)是TeeChart开放给用户的自定义绘图接口,与Windows自带的绘图接口类似,支持TextOut、LineTo、Arc等图形编程接口。

3 TeeChart的使用(TeeChart usage)

根据数据源的生成方式不同,曲线显示可以采用实时曲线和历史曲线的显示方式。实时曲线显示时,数据是实时生成的,通常需要将数据存储至一个数组中,随着数据更新,平移数组中的数据,保证数组中的数据是最新的,并通过数组中的数据更新曲线。历史曲线显示时,数据源是已经生成的,通常需要将数据从数据库中读取并显示出来即可,不存在曲线的更新问题。在转台调试中,调试软件通常实时显示采集到的调试数据,因此仅讨论实时曲线显示形式。

3.1 控件的导入

3.1.1 导入控件

TeeChart控件的导入与其他ActiveX控件类似,在Visual Studio开发环境中的导入过程分为以下几步:

(1)在项目中选择“工具”,打开“选择工具箱项”对话框;

(2)弹出的“选择工具箱项”对话框如图1所示,选择“COM组件”标签页,勾选“TeeChart Pro Activex control v2010”并点击“确定”,即可添加至当前开发环境;

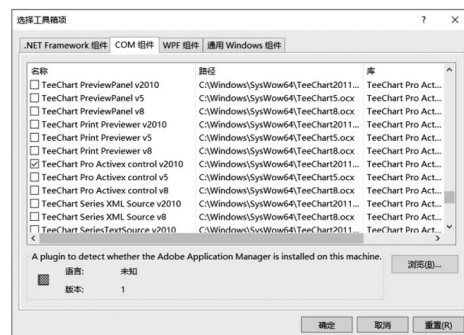


图1 TeeChart控件的导入

Fig.1 Import of TeeChart control

(3)添加成功后,开发环境中在控件面板上会显示TeeChart控件的图标,将其拖曳到当前项目中即可。

3.1.2 设置控件

在添加好的控件上点击右键,选择编辑控件,即可打开TeeChart控件的属性编辑对话框,如图2所示。点击左侧列表框中的Chart可以设置其数据成员的属性,包括坐标轴、标题、图例、面板等;点击左侧列表框中的Series可以设置添加序列。图表设置的大部分工作都可以在对话框中完成,也可以根据实际需要,通过代码来动态修改。为了程序的可移植性,采用代码的形式来添加序列,具体方法见3.2所述。

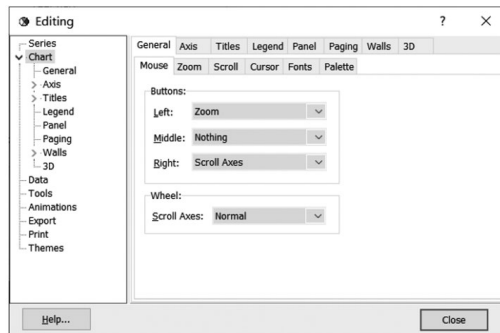


图2 TeeChart图形化属性设置界面

Fig.2 TeeChart graphical attribute setting interface

3.2 初始化

TeeChart控件的初始化主要完成控件颜色、图例、边距等外观的设置,以及图形缩放、按键等属性的设置,比较简单,具体代码如下。在实时曲线显示时,为了提高绘图效率,选择FastLine作为曲线类型。

```
Void InitTeeChart()
{
    //设置Margin
    m_chart.GetPanel().SetMarginTop(0);
    //背景色为淡蓝色,取消背景透明
    m_chart.GetWalls().GetBack().
SetColor(RGB(192,192,255));
    m_chart.GetWalls().GetBack().SetTransparent(FALSE);
    //取消panel颜色渐变
    m_chart.GetPanel().GetGradient().SetVisible(FALSE);
}
```

```

//取消Legend显示
m_chart.GetLegend().SetVisible(TRUE);
m_chart.GetLegend().SetCheckBoxes(TRUE);
m_chart.GetLegend().SetColor
(RGB(192,192,255));
//设置标题
m_chart.GetHeader().SetVisible(FALSE);
m_chart.GetHeader().SetCaption(_T("pretty
much"));
//取消3D外观
m_chart.GetAspect().SetView3D(FALSE);
//设置坐标轴
m_chart.GetAxis().GetLeft().GetAxisPen().
SetWidth(1);
m_chart.GetAxis().GetBottom().GetAxisPen().
SetWidth(1);
m_chart.GetAxis().GetBottom().GetLabels().
SetRoundFirstLabel(TRUE);
//添加Series0为FastLine,并设定名称
m_chart.AddSeries(scFastLine);
m_chart.Series(0).SetTitle(_T("轴0-4:用户位置
(°)"));
.....
//添加Series8为FastLine,并设定名称
m_chart.AddSeries(scFastLine);
m_chart.Series(6).SetTitle(_T("轴6-4:用户位置
(°)"));
}

```

3.3 绘制曲线

在FastLine的曲线绘制过程中, TeeChart通过用折线拟合曲线的方式来提升绘图效率, 曲线数据的添加通过CSeries类来完成, 该类提供了Add、AddXY、AddArray等数据接口, 分别用于点单以及数组数据的添加。当序列中的数据更新后, TeeChart控件自动将序列中的数据以折线连接并显示在图表中。

(1)向Series中添加一个点:

```
m_chart.Series(0).AddXY(x, y, "Lab", RGB (255, 0, 0));
```

(2)以数组的形式添加一组数据:

```
m_chart.Series(0).AddArray(200000, YValues, XValues);
```

该接口通常用于历史曲线显示时, 数据大量添加的情况, 序列数据添加完成后才进行一次图形的更新, 相比AddXY接口, 其效率有极大提高。

(3)修改数据:

```
m_chart.Series(0).GetValueLists().GetItems(0).
SetValue(index,newValue);
```

(4)删除数据:

```
m_chart.Series(0).Delete(x);
```

4 统计特性的实现(Implementation of statistical characteristics)

对TeeChart显示的曲线进行统计分析, 需要利用TeeChart开放的绘图事件接口来实现, 实现步骤如下:

(1)增加鼠标按下事件处理函数OnMouseDownTchartCurve, 在鼠标左键按下的时刻, 记录当前的坐标位置, 并设置bMouseFlag变量, 表示进入框选模式。

(2)增加鼠标移动事件处理函数OnMouseMoveTchartCurve, 当处于框选模式时, 移动鼠标时框选出需要统计分析的曲线块。

(3)增加鼠标按键弹起事件处理函数OnMouseUpTchartCurve, 在鼠标左键弹起的时刻, 清除bMouseFlag变量, 退出框选模式, 并记录下此时的坐标位置。利用两个坐标点确定矩形框, 并将两个坐标点的屏幕坐标转换到曲线显示的坐标系, 得到数据块起始、结束的横坐标, 以此范围计算数据块的统计信息。

(4)增加绘图完成事件处理函数AfterDrawTchart, 在该函数中增加计算完成曲线块的统计信息功能, TeeChart在绘图结束后将自动调用该函数。

几个关键函数的具体实现代码如下所示:

```
Void OnMouseDownTchartCurve(long Button, long
Shift, long X, long Y)
```

```

{
    if(Button==mbLeft)
    {
        //左键按下, 设置标识
        FirstPointX=SecondPointX=X;
        FirstPointY=SecondPointY=Y;
        bMouseFlag=true;
    }
}

```

```
Void OnMouseMoveTchartCurve(long Shift, long X,
long Y)
```

```

{
    if(bMouseFlag)
    {
        //鼠标左键按下标识
        SecondPointX=X;
        SecondPointY=Y;
        m_chart.GetCanvas().GetBrush().
SetStyle(bsClear);
        m_chart.GetCanvas().GetPen().
SetColor(RGB(255,255,255));
        m_chart.Repaint();
        m_chart.GetCanvas().Rectangle(FirstPointX,

```

```

FirstPointY, SecondPointX, SecondPointY);
    }
}

Void OnMouseUpTchartCurve(long Button, long
Shift, long X, long Y)
{
    if((Button==mbLeft)&&(bMouseFlag))
    { //鼠标左键弹起, 清除标识
        bMouseFlag=false;
        CAxis Bottom=m_chart.GetAxis().GetBottom();
        CAxis Left=m_chart.GetAxis().GetLeft();
        //Calculate Coordination of the point
        FirstPtPosX=Bottom.CalcPosPoint
(FirstPointX);
        FirstPtPosY=Left.CalcPosPoint
(FirstPointY);
        SecondPtPosX=Bottom.CalcPosPoint
(SecondPointX);
        SecondPtPosY=Left.CalcPosPoint
(SecondPointY);
        CalcStatics();
        m_chart.Repaint();
    }
}

void CCurveDlg::AfterDrawTchart()
{
    int base, index;
    double sum, value, sumsquare;
    CSeries ser=m_chart.Series(ActiveSeries);
    //Calculate the X value index
    base=ser.GetXValues().Locate((int)(
        FirstPtPosX<SecondPtPosX?FirstPtPosX;
SecondPtPosX));
    //计算统计数据
    if(base !=-1)
    {
        Maximum=Minimum=ser.GetYValues().
GetValue(base);
        sum=0;
        sumsquare=0;
        for(index=0;index<(abs(SecondPtPosX-
FirstPtPosX));index++)
        {
            value=ser.GetYValues().GetValue
(base+index);
            Maximum=max(value, Maximum);

```

```

        Minimum=min(value, Minimum;
        MeanValue+=value;
        sum=sum+value;
        sumsquare=sumsquare+value*value;
    }
    MeanValue=sum/index;
    StdValue=sqrt((sumsquare-sum*sum/index)/
(index-1));
}
else
{
    Maximum=Minimum=StdValue=MeanValue=0;
}
}

```

基于TeeChart控件完成的转台调试软件, 其完成界面如图3所示, 在该图示区域的左上角, 输出了当前选定数据块的统计信息, 并且指示出了当前选定的Series, 以及选定数据块的范围。此时转台进行正弦运动, 图中区域实时显示了用户位置曲线, 可以看到该模式启动和停止时振幅的变化过程, 以及在平稳正弦运动中的一些统计信息, 极大地方便了转台调试过程。

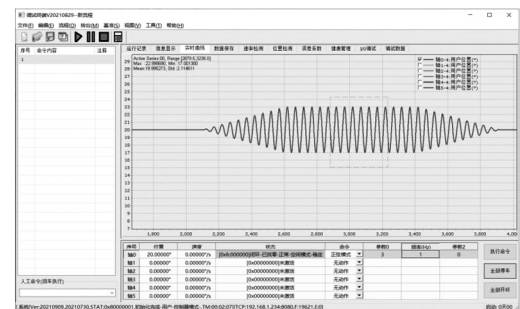


图3 利用TeeChart接口实现统计功能

Fig.3 Implementation of statistical function using TeeChart interface

5 结论(Conclusion)

针对转台调试测试软件调试数据的图形显示需求, 介绍了TeeChart控件在软件开发过程中的应用。利用TeeChart提供的属性和方法, 可实现各类数据的绘制, 并且通过自定义TeeChart事件接口函数的方法, 实现了曲线数据块的统计分析功能, 只要稍加修改并添加其他曲线分析算法, 就可以实现更加复杂、丰富的曲线分析功能, 可以显著缩短软件的开发周期和成本。基于TeeChart的开发应用并不局限于转台调试软件, 对于其他具有数据图形化显示需求的软件开发, 也有一定的参考价值。

参考文献(References)

- [1] 王绍伟, 盛小波, 赖欢, 等. 伺服转台调试软件的开发与应用[J]. 数字制造科学, 2017, 15(04): 186-191.
- [2] 龙治斌. P/T数字转台监控软件设计[J]. 沈阳工程学院学报(自

- 然科学版),2016,12(04):362-366.
- [3] 朱宁.仿真转台控制系统的设计与研究[D].哈尔滨:哈尔滨工业大学,2016.
- [4] 屈景辉.TeeChart应用技术详解[M].北京:中国水利水电出版社,2008:1-2.
- [5] 陈翔宇.基于TeeChart的无人机实时参数显示及告警系统[J].遥测遥控,2018,39(03):35-37.
- [6] 张利利,胡文东,马进.TeeChart控件在色觉检查系统中的应用[J].计算机技术与发展,2017,27(10):136-139.
- [7] 叶美玲,李智,王勇军.基于Teechart 8.0的测控系统数据分析处理软件设计与实现[J].桂林航天工业学院学报,2017,22(03):219-225.
- [8] 陈俊良,叶林,葛俊锋,等.TeeChart控件在实时检测系统上位机软件中的应用[J].工业控制计算机,2015,28(06):13-15.

作者简介:

- 苏长青(1985-),男,硕士,高级工程师.研究领域:惯性测试技术,嵌入式软硬件开发.
- 张旭明(1984-),男,硕士,高级工程师.研究领域:惯性测试技术,自动化控制系统.
- 董恒(1984-),男,硕士,高级工程师.研究领域:上位机软件设计.

(上接第20页)

- [2] 赵记军,吴正强,董博.甘肃马铃薯产业现状与发展对策[J].甘肃农业科技,2021,52(04):77-82.
- [3] 张茜.消费者对电商平台新鲜农产品物流服务感知差异性研究[D].太原:山西财经大学,2019.
- [4] 朱希敏.基于关联分析的农产品销售管理系统的设计与实现[D].重庆:重庆三峡学院,2019.
- [5] 周正勇,阳光,钱凯凯.助力特色农产品销售的电商平台构建——以宁波九龙湖生态基地为例[J].商场现代化,2016(15):57.
- [6] 童颜.贵州省瓮安县农产品市场销售价格调查分析[J].农家参谋,2020(22):7,11.
- [7] 姚哲.基于关联分析模型的农产品价格趋势预测[J].浙江工商职业技术学院学报,2020,19(01):7-11.
- [8] 曾雷.关联规则挖掘中Apriori算法的研究[D].重庆:重庆交通大学,2016.
- [9] 林朋.两种改进的关键规则挖掘算法研究与应用[D].马鞍山:安徽工业大学,2017.
- [10] 魏坤.基于FP-GROWTH关联规则的MGFP-GROWTH改进算法的研究与应用[D].镇江:江苏科技大学,2020.
- [11] 马瑞敏,吴海霞.基于FP-growth算法的关联规则挖掘研究及应用[J].太原师范学院学报(自然科学版),2021,20(01):19-22.
- [12] 陈勇刚,孙向东,崔丽娟,等.基于关联规则挖掘的航空设备故障诊断研究[J].数学的实践与认识,2021(09):99-107.
- [13] KAMM M R, KUCKLICK J P, SCHNEIDER J, et al. Data mining for small shops: Empowering brick-and-mortar stores through BI functionalities of a loyalty program[J]. Information Systems Management, 2021, 38(4):270-286.

作者简介:

- 黄宇承(1980-),女,硕士生,讲师.研究领域:数据挖掘.
- 吴丽丽(1979-),女,博士,副教授.研究领域:数据挖掘.
- 本文通讯作者.

(上接第34页)

- [18] BIAN T, XIAO X, XU T, et al. Rumor detection on social media with bi-directional graph convolutional networks[C]// CONITZER V. Proceedings of the AAAI Conference on Artificial Intelligence. Palo Alto, USA: AAAI Press, 2020, 34(01):549-556.
- [19] 张仰森,彭媛媛,段宇翔,等.基于评论异常度的新浪微博谣言识别方法[J].自动化学报,2020,46(8):1689-1702.
- [20] SHU K, SLIVA A, WANG S, et al. Fake news detection on social media: A data mining perspective[J]. ACM SIGKDD Explorations Newsletter, 2017, 19(1):22-36.
- [21] DA SILVA F C D, VIEIRA R, GARCIA A C. Can machines learn to detect fake news? A survey focused on social media[C]// BUI T. Proceedings of the 52nd Hawaii International Conference on System Sciences. Hawaii: Hamilton Public Library, 2019:2763-2770.
- [22] POTTHAST M, KIESEL J, REINARTZ K, et al. A stylometric inquiry into hyperpartisan and fake news[DB/OL].(2017-02-18) [2021-07-16]. <https://arxiv.org/abs/1702.05638>.
- [23] PAKA W S, BANSAL R, KAUSHIK A, et al. Cross-SEAN: A cross-stitch semi-supervised neural attention model for COVID-19 fake news detection[J]. Applied Soft Computing, 2021, 107:107393.
- [24] 赵洁,何文娟,鲁晓阳.一种基于新闻学的微博事件特征选择方法[J].情报学报,2015,34(04):424-431.
- [25] WANG Y, MA F, JIN Z, et al. Eann: Event adversarial neural networks for multi-modal fake news detection[C]// Guo Y K. Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. New York, USA: Association for Computing Machinery, 2018:849-857.

作者简介:

- 李小艳(1997-),女,硕士生.研究领域:自然语言处理.