有限元计算前后期处理控件的设计与开发

王　琼

(南京师范大学数学与计算机科学学院, 210097, 南京)

[摘要]　在有限元计算中, 划分单元和显示矢量场是重要的前后期处理步骤, 涉及许多图形操作. 利用 ActiveX 控件技术, 设计、开发了网格控件和矢量场控件, 为工程计算人员提供了方便.

[关键词]　有限元方法, 单元划分, 矢量场, ActiveX 控件

[中图分类号]TP311.51,　[文献标识码]B,　[文章编号]1672-1292-(2003)02-0071-04

在有限元计算方法中, 前期处理主要是建立合理的单元划分. 虽然单元划分有一定的章法可循, 但在一些特殊部位, 例如不规则的边界、需要特别处理的区域, 单元划分存在一定的主观性, 其合理性需要进行计算检验. 由于单元划分属于空间图形操作, 若无可视化工具的辅助, 必然费时费力.

有限元计算的结果, 常常是海量的矢量场数据, 如流速场、应力场、位移场等, 对其进行可视化解读、评判是非常必要的. 许多数模专家借助一些零散的程序完成实现图形显示, 由于这些程序的数据、功能接口不规范, 程序的通用性较差, 造成了模型计算的重复工作.

笔者利用 ActiveX 技术, 结合计算机图形学算法, 开发了两个 ActiveX 控件: 网格控件用于可视化单元划分; 矢量场控件用于场数据的图形显示.

1　ActiveX 控件技术

1.1　ActiveX 控件简介

ActiveX 技术基于微软的构件对象模型 COM, 核心是构造可复用的独立软件单元. 这种软件单元独立于平台、语言, 通过规范的接口提供一组服务, 被称为构件. 从复用性角度看, 构件是质量、性能更为优秀的代码模块.

构件有两种形式: ActiveX 类和 ActiveX 控件, 后者具有可视界面. 由于有限元计算的前后处理都涉及图形操作, 故选择 ActiveX 控件技术.

1.2　VB 中 ActiveX 控件的技术

在 VB 中开发 ActiveX 控件的途径是创建 ActiveX Control 类型的项目. 项目中的"用户控件"模块是自定义控件的容器, 一个项目可实现多个自定义控件. 关键在于如何自定义属性、方法、事件.

属性定义的一种常用方法是: 用属性过程 Get、Let、Set 定义属性. 该方法可以通过编程实现属性的完整性、一致性检查, 并可以定义属性的读/写特性.

方法是控件的主要功能接口, 是在"用户控件"模块以 Public 关键字说明的函数或过程.

事件表达的是控件状态的变化, 其意义在于向控件用户提供进一步的应用开发途径. 定义一个事件, 需要两步工作:

(1) 在用户控件模块中定义事件, 形式如同定义变量. 例如:

```
Event FieldClick(X As Integer, Y As Integer)    // 定义事件 FieldClick
```

(2) 在某段代码中引发事件. 例如:

收稿日期: 2003-02-25.

作者简介: 王琼, 1968-, 南京师范大学数学与计算机科学学院讲师, 河海大学博士研究生, 主要从事计算机辅助设计、计算机图形学、工程数据库的教学与研究.

```
Private Sub Picture1_MouseDown(Button As Integer, Shift As Integer, X As Single, Y As Single)
    RaiseEvent FieldClick(X, Y)    //激发事件FieldClick
End Sub
```

2 网格控件的设计

网格控件的主要设计目的是: 根据用户指定的结点坐标文件, 绘制网格图; 查询结点的坐标; 在网格图上调整结点的坐标; 更新结点坐标文件. 为便于观察结点较多的网格图, 控件还应提供图形缩放、平移功能.

网格控件由3个基本控件组成: 1个PictureBox、1个HScrollBar、1个VScrollBar. 其中PictureBox用于显示网格图, 2个滚动条用于横向、纵向平移图形.

由于在控件设计、使用中, 频繁引用到结点坐标, 因此在控件接口中, 定义了公共结点类型Vertex:

```
Public Type Vertex
    X As Single, Y As Single    //二维坐标分量(x,y)
End Type
```

2.1 属性设计

控件的主要属性有:

FileName, 结点坐标文件名; Rows, 网格的行数; Cols, 网格的列数; VertexColor, 网格结点颜色; CellColor, 单元边框颜色; BackColor, 网格背景颜色.

在本文讨论的设计中, 结点坐标文件是一个文本文件, 包含Rows* Cols行数据, 每行数据对应一个结点坐标. 这是目前常用的数据存储形式.

2.2 方法设计

控件的主要方法有:

Clear(), 清除网格图; Draw(), 读入数据, 计算结点坐标, 绘制网格图; GetVertex(vRow, vCol) As Vertex, 读取第vRow行第vCol列的结点坐标; SetVertex(vRow, vCol, vVertex), 将第vRow行第vCol列的结点坐标置为vVertex; DrawVertex(vRow, vCol, vColor), 以vColor色绘制第vRow行第vCol列的结点; DrawCell(vRow, vCol, vColor), 以vColor色绘制以第vRow行第vCol列结点为左下角的单元; Zoom(vScaleX, vScaleY), 将网格图按横向vScaleX、纵向vScaleY的比例缩放; Save(Filename), 将结点数据存入文件Filename.

控件中的重要数据结构有两个: 一是所有结点在世界坐标系中的坐标, 直接读自结点坐标文件; 二是所有结点在屏幕坐标系中的坐标. 控件用户始终面对的是世界坐标, 而作图使用的是屏幕坐标. 二者的转换由"窗口视区变换"实现, 对控件用户是不可见的.

方法GetVertex和SetVertex的作用是查询、修改各个结点的世界坐标.

方法DrawCell和DrawVertex的作用是, 控件用户可以对指定单元和结点加亮显示, 有利于构造一个界面良好的应用软件.

2.3 事件设计

控件的主要事件有:

Event SelectVertex (Row, Col), 用户用鼠标选择了第Row行第Col列的结点; Event SelectCell (Row, Col), 用户用鼠标选择了以第Row行第Col列结点为左下角的单元, 操作中, 需要同时按下Shift键.

上述两个事件是对PictureBox控件的MouseDown(Button, Shift, X, Y)事件的改装, 目的是判别鼠标对结点和单元的行为.

控件用户对这些事件的一般处理有: 在SelectVertex事件中调用GetVertex方法和SetVertex方法, 以实现选择结点、存取其坐标值的操作; 在SelectCell事件中调用DrawCell方法, 以实现选择单元、加亮显示的操作.

2.4 应用实例

网格控件作为单元划分的可视化工具,可在任何一种 OOPL 引用.引用后只需简单设置属性、调用方法,就可实现各结点坐标的可视化调整.

图 1 是网格控件应用于单元划分的实例.窗体左侧是网格控件,在指定 FileName、Rows、Cols 属性,调用 Draw 和 Zoom 方法后,其上显示放大后的网格图.窗体右上角 4 个文本框中的数据是 SelectVertex 事件驱动程序的运行结果.按钮"修改结点"的驱动程序的主体是调用 SetVertex 方法;按钮"保存文件"的驱动程序的主体是调用 Save 方法;按钮"放大"、"还原"的驱动程序的主体是调用 Zoom 方法.

用户首先按某种规则进行一个区域的单元划分,形成一个结点坐标文件.再使用类似图 1 的程序,完成各个特殊结点的坐标调整,为有限元计算作好准备工作.

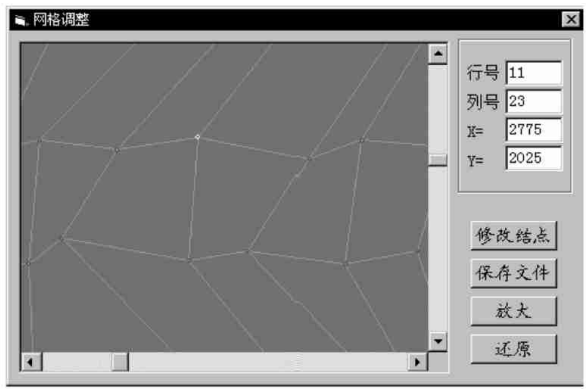

图 1 利用网格控件进行可视化的单元划分

3 矢量场控件的设计

矢量场控件的主要设计目的是:根据用户指定的矢量场数据文件,绘制箭头图(每个箭头表示一个矢量,箭头的方向和长度决定于矢量的水平、垂直分量);查询、修改任意矢量的数据;缩放、平移图形.

矢量场控件由 3 个基本控件组成:1 个 PictureBox、1 个 HScrollBar、1 个 VscrollBar.其中 PictureBox 用于显示箭头图,2 个滚动条用于横向、纵向平移图形.

由于在控件设计、使用中,频繁引用到矢量信息,因此在控件接口中,定义了公共矢量类型 Vector:

```
Public Type Vector
    X As Single, Y As Single   //矢量对应的点坐标(X,Y),为世界坐标系中的坐标值
    U As Single, V As Single   //矢量的水平分量 U、垂直分量 V
End Type
```

3.1 属性设计

控件的主要属性有:

VectorsFile,矢量场的数据文件名;BoundaryFile,边界线的端点坐标文件名;Rows,矢量场的行数;Cols,矢量场的列数;BackgroundColor,背景颜色;ArrowColor,箭头颜色;BoundaryColor,矢量场的边界颜色.

在本文讨论的设计中,矢量场数据文件是一个文本文件,包含 Rows* Cols 行数据,每行数据对应一个矢量的数据.边界线是矢量场的外框,是多段折线的集合,BoundaryFile 文件中有序地存储了所有端点的坐标.这种数据存储形式也为目前许多数模专家所常用的.

3.2 方法设计

控件的主要方法有:

Clear(),清除箭头图;Draw(),读入数据,计算屏幕坐标,绘制箭头图;DrawVector(vRow, vCol, vColor),以 vColor 色绘制第 vRow 行第 vCol 列的矢量;Zoom(vScaleX, vScaleY),将箭头图按横向 vScaleX、纵向 vScaleY 的比例缩放;GetVector(vRow, vCol) As Vector,读取第 vRow 行第 vCol 列的矢量结构;SetVector(vRow, vCol, vVector),将第 vRow 行第 vCol 列的矢量置为 vVector.

控件中的重要数据结构有两个:一是矢量场结构,由 Vector 类型的数据元素组成,直接读自矢量场的数据文件,其中点坐标为世界坐标.二是箭头图结构,其中数据元素为箭头结构 Arrow,定义如下:

```
Private Type Arrow
```

```
    Startp  As  Vertex  //箭头的起点坐标
    Endp  As  Vertex  //箭头的终点坐标
    Endp1  As  Vertex, Endp2 As Vertex  //箭头终点旁左右端点的坐标
    Angle  As  Single  //箭头的方向角
    Len  As  Single  //箭头的长度
End Type
```

箭头结构中的点坐标为屏幕坐标, 由矢量场结构经"窗口视区变换"而得, 直接用于作图.

3.3　事件设计

虽然 GetVector 方法提供了对矢量数据的查询, 但指定参数有些不便. 为了便于控件用户设计图形化的查询界面, 可设计一个选择矢量的事件: Event SelectVector(Row As Integer, Col As Integer), 表示用户用鼠标选择了第 Row 行第 Col 列的矢量.

可视化查询矢量信息的典型途径是: 在 SelectVector 事件的驱动程序中使用 GetVector 方法.

3.4　应用实例

矢量场控件是可视化解读有限元计算结果的工具. 其使用方法与网格控件完全类似.

图 2 是矢量场控件的应用实例. 窗体上方是矢量场控件, 在指定 VectorsFile、Rows、Cols 属性, 调用 Draw 和 Zoom 方法后, 其上显示着放大后的箭头图. 为显示用户任意点选的矢量数据, 实例在 SelectVector 事件的驱动程序中, 调用了 GetVector 方法获取矢量数据, 并调用 DrawVector 方法重画所选矢量.

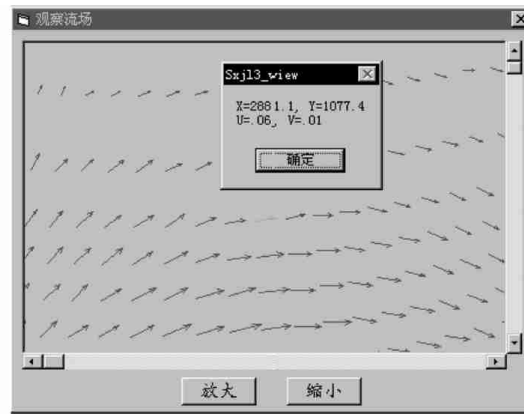

图 2　矢量场控件在后期处理中的应用

4　结束语

虽然上文给出的两个实例是独立的控件应用, 但控件用户可以将控件与有限元计算模块集成使用. 由于控件通过简单、规范的接口封装了有限元计算的前后期处理任务, 为有限元计算方法的灵活应用提供了较好的辅助工具.

[参考文献]

[1] 陈元琰, 张晓竞. 计算机图形学实用技术[M]. 北京: 科学出版社, 2000.
[2] Eric Brierley. Visual Basic 开发人员指南[M]. 北京: 机械工业出版社, 1999.
[3] Ash Rofail. COM 与 COM+ 从入门到精通[M]. 北京: 电子工业出版社, 2000.

Control's Design and Development In Pretreatment and Post-treatment of Finite Element Method

Wang Qiong

(College of Mathematics and Computer Science, Nanjing Normal University, 210097, Nanjing, PRC)

Abstract: In finite element method, the important pretreatment and post-treatment procedures are to create cells structure and visualize vectors field. Two ActiveX Controls are designed in this paper: net structure control and vector field control.

Key words: finite element method, cells structure, vectors field, activeX control

[责任编辑: 严海琳]