

MATLAB 和 VC++ 联合编程的 COM 研究^①

A COM Research on Mixed Programming of MATLAB and VC++

吴宪传 张向文 (桂林电子科技大学 计算机与控制学院 广西 桂林 541004)

摘要: VC++ 在实现矩阵运算、数值分析、工程计算等方面比较复杂,应用 VC++ 与 MATLAB 混合编程方法可以充分发挥 VC++ 和 MATLAB 各自优势,提高仿真和开发效率。着重介绍了利用组件对象模型(COM)技术的 VC++ 与 MATLAB 的联合编程方法,阐述了该方法实现过程,并通过两个具体应用实例加以验证。实验仿真结果表明,采用联合编程的 COM 技术能快速直观地得到满意结果,对求解矩阵运算、数值分析、工程计算等问题是行之有效的。

关键词: VC++ MATLAB 联合编程 组件对象模型

1 引言

MATLAB 是美国 MathWorks 公司开发的一种高性能的面向工程实践和科学运算的交互式仿真计算软件,它集科学计算、信号处理、系统仿真与图形可视化于一体,具有极高的编程效率,非常有利于工程技术人员方便地解决各种问题^[1];然而由于 MATLAB 是一种解释性语言开发平台,这就决定了它的代码执行效率比较低。VC++ 是 Windows 平台主要的应用程序开发环境之一,它既具有 C++ 语言的灵活性、高效性,又具有可视化编程的便利性、界面友好的特点,但在矩阵运算、数值分析、工程计算等实现功能稍嫌不足。因此,把 MATLAB 在数值计算、算法设计以及数据可视化等领域的优势与 VC++ 应用系统相结合,进行联合编程,实现优势互补,不仅可满足系统在数据运算与表现方面需求,而且还可提高系统处理效率和稳定性,从而能够在代码执行速度和编程效率上找到一个最佳平衡点^[2]。

本文结合 VC++ 和 MATLAB 各自优点,以 VC++ 图形界面作为前台框架, MATLAB 作为后台进行数值运算,利用组件对象模型(COM)技术作为媒介,通过实例实现对 MATLAB 和 VC++ 联合编程的 COM 研究。

2 MATLAB和VC++联合编程方法

2.1 常用的联合编程方法

目前,常用的 VC++ 与 Matlab 混合编程的方法主要有以下几种^[3-5]:(1)使用 Matlab 提供的 mcc 编译器。(2)利用 ActiveX 自动化技术,调用 Matlab 的服务引擎库。(3)使用第三方软件 Matcom 来实现。(4)使用 Matlab Visual Studio Add-in 工具。(5)基于 COM 组件技术。本文主要利用基于 COM 组件技术的 VC++ 与 Matlab 联合编程方法。

2.2 COM 组件技术

COM(组件对象模型)是微软公司为了计算机工业的软件生产更加符合人类的行为方式,开发的一种新的软件开发技术,是 Windows 对象的二进制标准,为 Windows 提供了统一的、面向对象的、可扩充的通讯协议^[6]。Matlab 的 COM Builder 能够帮助用户把 Matlab 编写的 M 程序文件快速地做成 COM 组件,供使用其他语言的客户端调用。该方法几乎可以调用 Matlab 的任何函数,而且实现简单,通用性强,程序间无需环境切换,代码执行速度快,因此可获得较高的执行效率。

^① 基金项目:国家自然科学基金(60804059);广西自然科学基金(桂科自 0640172)
收稿时间:2008-11-24

2.3 Matlab 和 VC++ 联合编程的实现

2.3.1 创建 COM 组件

首先, 确保安装了 Matlab COM Builder 组件; 其次, 使用 `mbuild -setup` 命令设置了编译环境。COM 组件的创建过程如下:

①启动 Matlab COM Builder。在 Matlab 中输入命令 `com tool` 或者单击 `Start-> Matlab-> Matlab COM builder`, 运行 COM Builder 所需的编译环境。

②选择 `File-> New Project`, 输入 COM 组件名并修改类名, 如果调用了 Matlab 绘图操作, 则需要选中 "Use Handle Graphics Library" 以便让 Matlab 配置 C/C++ 图形函数库。

③选择 `Project-> Add File`, 添加生成 COM 组件的函数形式的 M 文件。

④单击 "Build" 按钮, 编译完成后在 Matlab 工作目录下的工程目录中生成以 COM 组件命名的目录, 里面包含两个文件夹: `distrib` 目录中是以动态链接库形式存在的 COM 组件, `src` 目录中则是 COM 组件的源代码。

2.3.2 在 VC++ 中调用 COM 组件

在 VC++ 中调用 COM 组件的方法主要有两种: 一种方法是将 `src` 目录中生成的 `.h` 和 `.cpp` 文件复制并添加到工程、添加头文件并编写调用代码; 另一种方法更加简单, 通过 DLL 来调用 COM 组件。后者的详细步骤为: ① 在 VC++ 中新建一个 MFC (exe) 类型基于对话框的工程并添加相应控件; ② 打开 `ClassWizard-> Add class-> From a type library class`, 选中在 (1) 中生成的 DLL 的 COM 组件 (.dll), 按照提示完成后, 系统会自动生成 `.h` 和 `.cpp` 文件, 并增加了一个新类。③在需要调用 COM 组件函数的首部添加头文件 `# include ".h"`, 并在调用处添加代码。

2.3.3 发布 COM 组件

MATLAB COM Builder 提供了 COM 打包程序, 在 MATLAB COM Builder 中, 使用菜单选项 `Component-> Package Component`, 这个命令将在 `distrib` 文件夹下生成一个自解压的 `exe` 安装文件。将该文件复制到没有安装 Matlab 的机器上执行,

输入解压目录并设置系统搜索路径后, 系统会自动注册该 COM 组件和所需的 `.dll` 通用组件, 从而能够实现完全脱离 Matlab 环境执行。

利用 COM 组件能够发挥 VC++ 和 Matlab 两种语言的长处, 同时具有操作简单、代码高效的特点; 除了 COM 组件的部分限制外(例如: 不支持 Object 变量), 近乎是一种 VC++ 和 Matlab 混合编程的完美选择。

2.4 在 VC++ 中调用 COM 组件的语句

VC++ 中调用 COM 组件的具体语句步骤如下:

- (1) 初始化 COM 库;
- (2) 得到 COM 对象的 CLSID;
- (3) 创建一个 COM 对象的实例;
- (4) 使用 COM 对象;
- (5) 退出 COM 库。

//初始化 COM 库

```
HRESULT CoInitalize(LPVOID pvReserver);
```

//Windows 保留参数, 必须设为 NULL

//根据返回的 HRESULT 值来判断 COM 库的初始化是否成功

.....

```
HRESULT hr;
```

```
Hr=CoInitalize(NULL);
```

```
If ( FAILED(hr))
```

```
{
    return false;
}
```

.....

//得到 COM 对象的 CLSID

```
CLSID CLSID_MyInterface;
```

```
HRESULT hr;
```

```
Hr=CLSIDFromProgID(L"Myinterface.Myinterface1.0",& CLSID_MyInterface);
```

```
If(FAILED(hr))
```

```
{
    MessageBox("CLSIDFromProgID 调用失败!"); return false;
}
```

```
//创建一个 COM 对象的实例
MyInterface *pMyinterface
Hr=CoCreateInstance(CLSID_MyInterface,N
ULL,CLSCTX_ALL,IID_ImyInterface,(void*)&pMy
Inface); // IID_ImyInterface 接口 ImyInterface 的
IID
If(FAILED(hr))
{
    MessageBox("创建 ImyInterface 接口失
败! ");
return false;
}
//使用创建的 COM 对象
.....
退出 COM 库
CoUninitialize();
```

3 应用实例

本节根据第二节介绍的方法，以两个具体实例利用组件对象模型(COM) 技术来实现 MATLAB 和 VC++联合编程的数值计算。

3.1 函数的数值定积分实例

采用差分法求函数 $f(a,x)=\int_a^x e^{-0.9t|\cos(t)} dt$ 在区间[a,x]上的数值积分，并显示出函数图形。

3.1.1 生成 COM 组件

在 Matlab 7.0 环境中编辑 M 文件，其主要程序语句为：

```
dt=0.01;
t=a:dt:x;
y=exp(-0.9*t.*abs(cos(t)));
sum_ax=dt*sum(y(1:length(t)-1));
%sum 积分
trapz_ax=trapz(t,y);           %trapz
积                             分
ff=inline('exp(-0.9*t.*abs(cos(t))','t');
q_ax=quad(ff,a,x);             %quad
积                             分
q8_ax=quad8(ff,a,x);
```

```
%quad8 积分
plot(t,y,'b');                %绘制函数
依照 2.3 节创建组件的方法，编译完成并生成
myjifen.dll 文件。
```

3.1.2 建立 VC++工程并设置必要的环境

建立一个 VC++ MFC (exe) 类型工程，生成一个基于对话框的应用程序。将自动生成的 myjifen.h 和 myfifen.c 文件拷贝到当前工程目录下并添加到工程中。同时在 VC++添加生成 COM 组件所需要的头文件和文件路径。

3.1.3 在 VC++工程中添加代码

在 myjifenDlg.cpp 添加必要的头文件和库文件，即添加以下代码：

```
#include "mwcomutil.h"
#import"D:\MATLAB70\bin\win32\mwcom
util.dll"raw_interface_only
#include "myjifen.h"
```

3.1.4 运行结果

在为界面添加所需控件和相应代码，调试程序，最后形成可脱离 MATLAB 环境运行的可执行文件 (函数的数值定积分实例 .exe), 在 f(x)积分区域[a,x] 上填写 a 为 0, x 为 7, 点击按钮“积分”，运行结果界面如图 1 所示，并生成该函数的函数图形，如图 2 所示。



图 1 函数 f(x)定积分数值计算图

从图 1 可知道， $f(a,x)=\int_a^x e^{-0.9t|\cos(t)} dt$ 在区间 [a,x]上的 sum, trapz, quad 和 quad8 是数值积分结果依次是 2.1963, 2.1914, 2.1913, 2.1913。

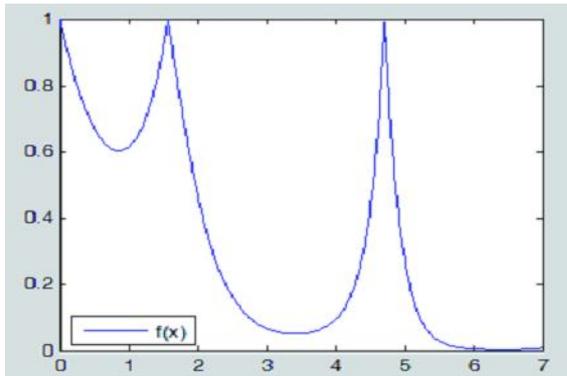


图 2 函数 f(x) 曲线图

3.2 气体分子运动速度分布律的数值计算实例

求 300 开尔文下氧分子运动的速度分布曲线, 并求速度在 260~500m/s 范围内的分子所占的比例。

麦克斯韦速度分布律为:

$$f = 4\pi \left(\frac{m}{2\pi kT}\right)^{3/2} v^2 \exp\left(\frac{-mv^2}{2kT}\right)$$

利用分布律绘制它们之间的关系图, 确定温度及分子量对速度分布曲线的影响。

Matlab 主要程序语句:

```
v=0:1500;
y=mksw(T,mu,v); %麦克斯韦分布
plot(v,y);
v1=260:500; %速度范围
y1=mxsw(T,mu,v1);
fill([v1,500,260],[y1,0,0],strcolor); %绘
```

出速度范围

```
guidata(hObject,handles); %保存数据
bili=trapz(y1); %计算比例
%手动鼠标标注曲线
gtext(['T=',num2str(T),'mu=',num2str(mu),'比例,num2str(bili)];
```

调用 mksw 函数, 求对应温度、分子量和速度下的麦克斯韦速度分布, 代码如下:

```
function f=mksw(T,mu,v)
R=8.31; %气体常数
k=1.381*10^(-23); %波耳兹尔茨曼常数
```

```
Na=6.022*10^23; %阿伏伽德罗常数
m=mu/Na; %分子量
%麦克斯韦速度分布
f=4*pi*(m/(2*pi*k*T)).^(3/2).*exp(-m*v.^2./(2*k*T)).*v.*v;
```

依照 2.1 小节操作步骤和过程, 配置必要环境, 生成对应的 COM 组件, 添加相应的头文件和路径, 并为界面添加所需控件和相应代码, 调试程序, 最后形成可脱离 MATLAB 环境运行的可执行文件(气体分子运动速度分布计算实例 .exe), 在界面选择参数上填写温度为 300, 分子量为 32e-3, 点击按钮“计算”后运行结果界面如图 3 所示; 并生成分子运动的速度分布曲线图形, 如图 4 所示, 当用鼠标在图上单击时, 图上显示标注。

从图 3 和图 4 可知道, 当温度 T=300, 分子量 mu=32e-3 时, 速度在 260~500m/s 范围内分子所占比例为 0.47251。

3.3 结果分析



图 3 气体分子运动速度所占比例图

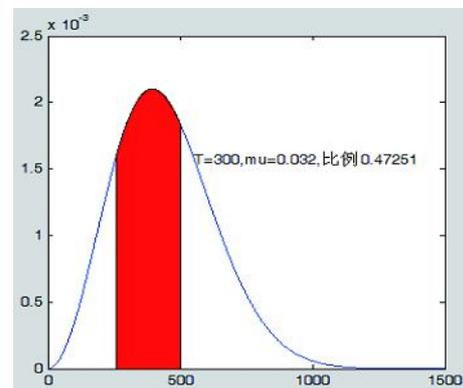


图 4 分子运动的速度分布曲线图

(下转第 194 页)

本节通过两个实例把 VC++与 MATLAB 的数值计算、图形显示功能进行了结合,既保持了前台 VC++良好用户界面和高执行效率,又保持将 MATLAB 作为后台一个功能十分强大的数学函数库,使 VC++和 MATLAB 各自优势能够得到充分发挥。在仿真实验中只用很少代码就实现了数值计算以及 MATLAB 和 VC++数据交换,取得了较为满意的结果。

4 结语

本文采用了基于 COM 的 MATLAB 与 VC++联合编程方法来实现数值计算,并用实例说明如何在脱离 MATLAB 运行环境的情况下,方便快速地运行并达到了预期效果。实践证明,这种方法不仅实现了 VC++可视化界面与 MATLAB 强大的数值分析和图形显示能力的有效结合,而且可完全脱离 MATLAB 环境运行,有效节省系统资源,缩短软件开发周期,提高软件可靠性,尤其在大型复杂系统

里,这优势将体现得更明显。笔者抛砖引玉,为信号处理和系统仿真等其他工程领域的应用提供了一种新渠道和新途径。

参考文献

- 1 杨高波,亓波.精通 MATLAB 7.0 混合编程.北京:电子工业出版社,2006.
- 2 黄金明,武玉强,邢西深.VC++与 MATLAB 的混合编程研究.电脑开发与应用,2008,(10):38-41.
- 3 李天昀,葛临东.综述 MATLAB 与 VC++ 的交互编程.计算机仿真,2004,21(9):193-196.
- 4 陶红,袁金荣.MATLAB 与 C/C++混合编程的实现.计算机工程与应用,2000,(10):100-101.
- 5 何晓涛,于春田.VC 调用 MATLAB 的方法.河北科技大学学报:自然科学版,2003,(1):35-39.
- 6 王世香.精通 MATLAB 接口与编程.北京:电子工业出版社,2007.