

文章编号:1000-582X(2007)07-0026-04

MATLAB 与 VC ++ 混合编程在系统仿真中的应用

陶桂宝,郭少波

(重庆大学机械工程学院,重庆 400030)

摘要:通过讨论 MATLAB 与 VC ++ 之间的接口方案,介绍了 MATCOM 与 MATLAB 的 m 文件的链接方式以及 MATCOM 的使用方法和技巧,以单自由度振动系统为对象,通过对有阻尼状态下的振动模型的分析,对仿真算法进行了混合编程。仿真结果表明:系统仿真的执行速度得到了提高。

关键词:系统仿真;MATLAB;VC ++;单自由度振动;阻尼

中图分类号: TP311.1;TP391.9

文献标志码: A

MATLAB 是美国 MATHWORKS 公司自 20 世纪 80 年代中期推出的数学软件,它集数值分析、矩阵运算、信号处理和图形显示于一体,是建立在向量数组和矩阵的基础上,具有简洁的人机界面、丰富的图形用户界面 GUI 开发功能和极高的编程效率的数学工具,受到越来越广泛的重视和应用,但是这些功能只能在它本身所提供的平台上使用,且具有代码执行速度慢、安全保密性差的特点^[1]。

VC ++ 是 Microsoft 推出的可视化编程环境,它是目前综合性最高、最强大、也是最复杂的软件开发工具之一,应用极为广泛,在工程计算方面和 MATLAB 相比,编程就显得复杂得多^[2]。把两者相结合,无疑是一种有效的编程方法,将给应用程序系统本身及其开发带来很好的改善。通过混合编程,可以大大提高代码的执行速度以及安全保密性。本文介绍 VC 调用 MATLAB 的主要方案,并就单自由度振动系统,采用了混合编程方式,在脱离 MATLAB 环境条件下实现 VC ++ 对 m 文件调用,从而在 VC 环境下实现系统振动仿真的方案^[3]。

1 MATLAB 与 VC ++ 混合编程接口

根据 MATLAB 是否运行,一般将 MATLAB 与 VC 接口分为 2 大类:需 MATLAB 在后台运行的混合编程接口和可以脱离 MATLAB 环境运行的独立应用程序接口^[4]。

第 1 类,需 MATLAB 在后台运行的混合编程接口;MEX 文件、engine 应用程序和 MAT 文件。由于 VC 的内存管理及类库函数比 C 更加灵活方便,因此,对这 3 种接口方式在 VC 中的实现,主要基于内存管理和类库函数应用方面。

第 2 类,可以脱离 MATLAB 环境运行的独立应用程序接口 MCC 和 MATCOM 编译器。MCC 是 MATLAB 中经过优化的编译器。使用 MCC,用户可将 MATLAB 数学库、图形库和界面的 MATLAB 程序转化为独立于 MATLAB 的 EXE 应用程序和 DLL 动态链接库。

MATCOM 是 MATHWORKS 公司推出的第一个由 MATLAB 到 C ++ 的编译软件开发平台,其最新版本为 MATCOM4.5,其集成调试编译环境为 MIDEVA。通过 MATCOM 连接 MATLAB 的 m 文件有以下 3 种方法:

1) 经过简单设置后,由 MIDEVA 将 m 源文件转化为 C/C ++,然后添加到 MSVC 工程中。

2) 由 MIDEVA 直接生成 EXE 文件,然后在 VC 中通过 Shell 调用。这种方法简单方便,但运行时出现一个控制台窗口,而且由于 VC 和 MATLAB 之间不能交互,通用性差,仅适用于 VC 调用 MATLAB 实现图形显示的场合。

3) 通过 Visual MATCOM 工具条,使用 Add-in,这种方法提供了一个 MATLAB 和 VC 直接集合的途径,而且可以快速集成 m 文件到 VC 工程中创建独立的

收稿日期:2007-03-14

基金项目:重庆市自然科学基金资助项目(CSTS,2006BB3015)。

作者简介:陶桂宝(1967-),男,重庆大学副教授,博士,主要从事机电一体化、汽车电子、网络化制造等方面的研究,
(E-mail) ghtao@cqu.edu.cn。

C/C++应用程序、C、MEX、DLL,在调试过程中可以查看矩阵变量的值,可直接修改m源文件而不是修改生成的C/C++文件。因此,本文中单自由度系统的振动仿真采用Add-in方法实现。

2 有阻尼单自由度自由振动系统

机械振动是工程中常见的物理现象。广泛地说,各种机器设备及其零部件和基础,都可以看成是不同程度的弹性系统,机械振动就是在一定的条件下,振动体在其平衡位置附近所作的往复性的机械运动^[5]。实际中的振动系统是很复杂的,为了便于分析研究和用数学工具进行计算,需要在满足工况要求的条件下,把实际的振动系统简化为力学模型。机械振动分析方法很多,对于简单的振动系统,可以直接求解其微分方程的通解。由于计算机进行数值计算非常方便,所以振动的微机仿真是一种最直接的方法。由于振动模型中尤其是多自由振动很方便用矩阵微分方程来描述,所以,MATLAB语言在振动仿真中体现出十分优越的特性^[6]。

图1所示为考虑了阻尼的单自由度振动系统模型。

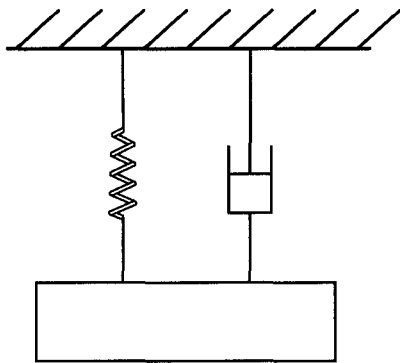


图1 振动系统模型

其运动微分方程为

$$m\ddot{x} + c\dot{x} + kx = 0, \quad (1)$$

$$\text{令 } \frac{c}{m} = 2n, \frac{k}{m} = \omega_n^2,$$

则

$$\ddot{x} + 2n\dot{x} + \omega_n^2 x = 0, \quad (2)$$

其通解为

$$x = e^{-nt} (c_1 e^{\sqrt{n^2 - \omega_n^2} t} + c_2 e^{-\sqrt{n^2 - \omega_n^2} t}), \quad (3)$$

式中 c_1, c_2 为积分常数,由振动初始条件确定。

令 $\frac{n}{\omega_n} = \varepsilon$,称为相对阻尼系数或阻尼率。则式(3)

可写为

$$x = e^{-\varepsilon\omega_n t} (c_1 e^{\omega_n t \sqrt{\varepsilon^2 - 1}} + c_2 e^{-\omega_n t \sqrt{\varepsilon^2 - 1}}). \quad (4)$$

由此可以讨论阻尼对系统的自由振动将会产生的影响。

1) 当时 $\varepsilon < 1$,称为弱阻尼状态

此时, $\varepsilon^2 - 1$ 为虚数,式(4)变为

$$x = e^{-\varepsilon\omega_n t} (c_1 e^{i\sqrt{\varepsilon^2 - 1}\omega_n t} + c_2 e^{-i\sqrt{\varepsilon^2 - 1}\omega_n t}), \quad (5)$$

利用欧拉公式,式(5)可写为

$$x = Ae^{-\varepsilon\omega_n t} [b\cos\sqrt{1 - \varepsilon^2}\omega_n t + a\sin\sqrt{1 - \varepsilon^2}\omega_n t], \quad (6)$$

括号内为两个简谐振动相加,即式(1-6)可写为

$$x = Ae^{-\varepsilon\omega_n t} \sin(\sqrt{1 - \varepsilon^2}\omega_n t + \varphi), \quad (7)$$

$$A = \sqrt{\frac{(V_0 + \varepsilon\omega_n x_0)^2 + x_0^2 \omega_n^2 (1 - \varepsilon^2)}{\omega_n^2 (1 - \varepsilon^2)}},$$

$$\varphi = \arctan\left(\frac{x_0 \omega_n \sqrt{1 - \varepsilon^2}}{V_0 + \varepsilon\omega_n x_0}\right).$$

2) 当 $\varepsilon > 1$ 时,称为强阻尼状态

此时,式(4)可写为

$$x = c_1 e^{(-\varepsilon + \sqrt{\varepsilon^2 - 1})\omega_n t} + c_2 e^{(-\varepsilon - \sqrt{\varepsilon^2 - 1})\omega_n t}, \quad (8)$$

$$c_1 = \frac{V_0 + (\varepsilon + \sqrt{\varepsilon^2 - 1})\omega_n x_0}{2\omega_n \sqrt{\varepsilon^2 - 1}},$$

$$c_2 = \frac{-V_0 + (-\varepsilon + \sqrt{\varepsilon^2 - 1})\omega_n x_0}{2\omega_n \sqrt{\varepsilon^2 - 1}}.$$

3) 当 $\varepsilon = 1$ 时,称为临界阻尼状态

由于 $\varepsilon = \frac{n}{\omega_n} = 1, n = \omega_n$,则有

$$c_c = 2m\omega_n = 2m\sqrt{\frac{k}{m}} = 2km, \quad (9)$$

这里 c_c 为临界阻尼状态下的阻尼系数,称为临界阻尼系数,显然它是系统本身所具有的特性之一。

由 $\varepsilon = \frac{n}{\omega_n} = \frac{c}{2m\omega_n}$ 及 $c_c = 2m\omega_n$,有 $\varepsilon = \frac{c}{c_c}$ 。也就是说,相对阻尼系数(阻尼率) ε 反映了系统的实际阻尼与临界阻尼的关系。

在临界阻尼状态下,有

$$x = e^{-\omega_n t} (c_1 + c_2 t), \quad (10)$$

其中 $c_1 = x_0, c_2 = V_0 + \omega_n x_0$ 。显然,在这种状态下不能形成振动。

3 应用

3.1 根据式(7)、(8)、(10)编写如下程序

```
function vbt1(m,c,k,x0,v0,tf)
```

%m为质量;c为阻尼;k为刚度;x0为初始位移;v0为

初始速度;
%tf 为仿真时间;wn 为固有频率;A 为振动幅度;phi 为初相位

```
global wn A phi
wn = sqrt(k/m);
t = 0;tf/1000;tf;
if z < 1
    A = sqrt(((v0 + z * wn * x0)^2 + (x0 * wd)^2)/wd^2);
    phi = atan2(x0 * wd, z * wn * x0);
    x = A * exp(-z * wn * t) * sin(wd * t + phi);
else if z == 1
    a1 = x0;
    a2 = v0 + wn * x0;
    x = (a1 + a2 * t) * exp(-wn * t);
else
    a1 = ((-v0 + (-z + sqrt(z^2 - 1)) * wn * x0) * wn * x0) / 2 / wn / sqrt(z^2 - 1);
    a2 = (v0 + (z + sqrt(z^2 - 1)) * wn * x0) / 2 / wn / sqrt(z^2 - 1);
    x = exp(-z * wn * t) * (a1 * exp(wn * sqrt(z^2 - 1) * t) + a2 * exp(wn * sqrt(z^2 - 1) * t));
end
end
plot(t,x),grid;
xlabel('时间(s)');
ylabel('位移');
title('位移相对时间的关系');
```

3.2 创建 VC 工程“vibration”

运行 Visual C++ 并创建 vibration 工程,选取 MFC AppWizard (exe)。在 location 编辑框中输入欲创建过程的保存路径,在 Project name 编辑框中输入工程名“vibration”,注意要选取“Dialog based”。

3.3 转换 m 文件“vbt1.m”

运行 Visual C++,改变文件类型为 Add-ins (.dll),选定%MATCOM45%\bin\Mvcide.dll,确定。在 Visual C++ 的开发环境中可以看到 Visual MATCOM 工具条,单击 Visual MATCOM 工具条上的 m++ 图标,选择保存过的 MATLAB 文件 vbt1.m 进行转换。此时在 FileView 标签中多了 m-files、C++ files created from m-files、MATrix <lib> 和 External Dependencies 等文件夹。在相应的文件夹下增加了 vbt1.m、vbt1.h、vbt1.cpp、v450lv.lib 和 matlab.h 等文件,这表明 m 文件转化成功。在“vbt1.h”文件中添加头文件“matlib.h”。

3.4 创建如图 2 所示的对话框界面,并按表 1 设置各控件属性。

表 1 控件属性

控件属性	控制 ID	说明
Edit Box	IDC_EDIT_m	质量
Edit Box	IDC_EDIT_c	阻尼
Edit Box	IDC_EDIT_k	刚度
Edit Box	IDC_EDIT_x0	初始位移
Edit Box	IDC_EDIT_v0	初始速度
Edit Box	IDC_EDIT_tf	仿真时间
Edit Box	IDC_EDIT_wn	固有频率
Edit Box	IDC_EDIT_A	幅值
Edit Box	IDC_EDIT_phi	相位
Button	IDC_BUTTON_com	开始计算与仿真
Static Text	IDC_STATIC_simu	用于显示响应曲线

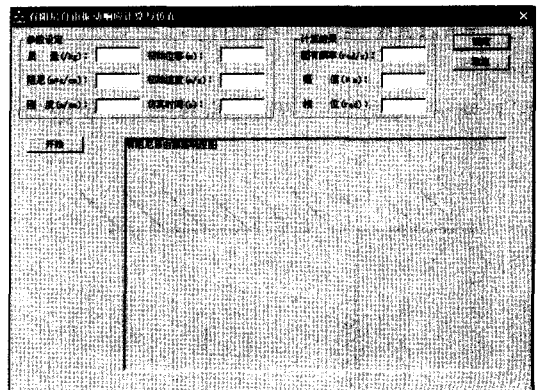


图 2 对话框界面

1) 添加头文件

```
include "vbt1.h"
include "matlib.h"
```

2) 在 vibrationDlg.cpp 中添加如下声明

```
extern Mm wn; extern Mm A; extern Mm phi;
```

3) 为 IDC_BUTTON_com 添加响应代码

```
void CTest4Dlg::OnBUTTONcom()
{
    //TODO: Add your control notification handler code here
    double x1,x2,x3,x4,x5,x6;
    UpdateData(TRUE);
    x1 = (double)m_m; x2 = (double)m_c; x3 = (double)m_k;
    x4 = (double)m_x0; x5 = (double)m_v0; x6 = (double)
    m_tf;
    CWnd * pvtb1 = (CWnd *) GetDlgItem ( IDC_STATIC_
    simu);
    Mm plothandl = winaxes(pvtb1 -> m_hWnd);
```

```

initM(MATCOM_VERSION);
vbt1(x1,x2,x3,x4,x5,x6);
m_wn = wn. r();
m_A = A. r();
m_phi = phi. r();
UpdateData(FALSE);
exitM();

```

3.5 编译链接 vibration 工程

按一般的 C++ 程序进行编译 vibration 工程, 由于 vbt1.h 函数定义文件是相应的 m 文件转换生成的, 若 vbt1.m 发生修改, VC++ 会询问用户是否重新加载这个文件, 此时, 在重新生成的 vbt1.h 文件中没有包含 "matlib.h" 头文件, 需手动添加, 若没有包含该文件, 编译器不能识别 Mm 数据类型而发生错误。

3.6 执行 vibration.exe

执行 vibration.exe 文件后, 得如图 3 所示对话框,

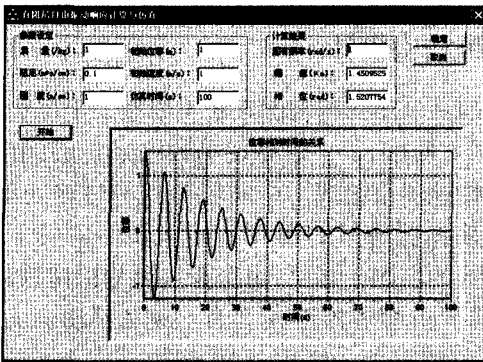


图 3 仿真结果

在参数设定栏中输入各参数, 单击“开始”按钮, 在左框中绘制了有阻尼的单自由度振动系统的响应曲线。

4 结 论

利用 MATCOM 的 Add-in 方法进行程序设计的规范, 不需要对于不同的应用程序做不同的接口连接代码, 只需要编写相应的 m 文件即可; 同时由于 MATLAB 强大的数值运算及绘图功能, 也可以简化工程算法及绘图问题, 是一种省时高效的系统仿真的实现方法, 对于 VC++ 并不熟练的工程技术人员来说是一个相当有效的选择, 使工程人员可以将更多的时间运用于工程原理上, 该程序在 VC++ 6.0 和 MATLAB 6.5 下测试通过。

参考文献:

- [1] 张宜华. 精通 MATLAB5[M]. 北京: 清华大学出版社, 1999.
- [2] KAET GREGORY. VisualC++ 5 开发使用手册[M]. 北京: 机械工业出版社, 1998.
- [3] 潘林峰, 谭立超. MATLAB 与 VC 混合编程的实现[J]. 武汉化工学院学报, 2003, 25(4): 74-75.
- [4] 苏金明, 刘宏, 刘波. MATLAB 高级编程[M]. 北京: 电子工业出版社, 2005.
- [5] 刘成俊, 周雄, 谭逢友, 等. 机械振动与数学的关系初探[J]. 重庆科技学院学报: 自然科学版, 2006, 8(4): 39-40.
- [6] 尚涛, 石瑞伟, 安宁, 等. 工程计算可视化与 Matlab 实现[M]. 武汉: 武汉大学出版社, 2002.

Application of Mixed Programming by MATLAB and VC++ in System Simulation

TAO Gui-bao, GUO Shao-bo

(College of Mechanical Engineering, Chongqing University, Chongqing 400030, China)

Abstract: After discussing the interface between MATLAB and VC++, the method of connecting MATCOM with the m-files, the usage and skills of MATCOM are introduced. Taking single-freedom vibration system as the object, through analyzing the vibration model with damping, the simulation algorithm is programmed. The simulation results show that the system simulation efficiency has been improved.

Key words: system simulation; MATLAB; VC++ ; single-freedom free vibration; damping

(编辑 张小强)