

Windows 环境中 C 语言对 Matlab 的编程

魏会贤¹, 刘洪坤²

(1. 空军工程大学 电讯工程学院, 陕西 西安 710077; 2. 海军工程大学, 湖北 武汉 430033)

摘要:阐述了 Windows 环境中 C 语言对 Matlab 编程的原理。提出了 C 使用引擎及 ActiveX 自动化技术开发 Matlab 程序的方法思路。实现了多应用软件开发中 Matlab 的连接方法和步骤, 有关思路和方法也适用于其它多应用程序的开发。

关键词:引擎; ActiveX 自动化; Matlab; CB

中图分类号: TP311.54 **文献标识码:** A **文章编号:** 1009-3516(2001)03-80-82

Matlab 是美国 MathWorks 公司开发的用于数值计算及图形处理的科学计算环境。特点是在生成、使用矩阵对象时, 不要求显示维数说明, 与利用 C 或其它如 Fortran 语言做数值计算的程序相比, 利用 Matlab 可以节省大量的编程时间, 因此, 在工程科技界, Matlab 被用来解决大量的实际课题及数学模型的分析、验证。

虽然就 Matlab 本身而言, 可以说已经包含较完整的数据处理、程序编写和调试环境, 但对有些低级操作却无能为力, 如: 实现 A/D、D/A 或其它硬件卡的访问、Windows 资源访问及与其它 Windows 应用交换数据等。但是, 如将完成此功能的 C(或 Fortran)链进 Matlab 可以间接实现(通过 .dll 形式的 mex 文件); 另外, 对于需要多应用的领域, 或者有些用户, 纯粹只是用到 Matlab 的处理计算功能, 而其主要程序却是 C(或 Fortran)语言编写的, 这就存在如何将 C(或 Fortran)程序与 Matlab 程序相链接的现实问题。Matlab 允许用户将自己的 Matlab 程序与 C(或 Fortran)程序相互链接, 也就是说, 允许 C(或 Fortran)程序链接 Matlab 或 Matlab 链接 C(或 Fortran)。本文就 C 程序如何链接 Matlab 程序的机理及方法给予了深入的分析并给出了完整的链接 Matlab 的程序, 而且, 与 Matlab 中给出的 DOS 下例程不同, 作者经过探索提出了如何在 Windows 环境下用 CB4 实现将 Matlab 链接进 C 程序的方法, 对于拓展 Matlab 的应用领域, 开发多应用程序软件提供了可能的方法。

1 使用引擎 C 对 Matlab 的编程

1.1 Matlab 的引擎技术

Matlab 引擎是 Matlab5.x 语言提供的供 C(Fortran)访问 Matlab 环境的一种机制, 它属于 C/S 模式的服务器, C 程序可以把此引擎当作函数库来使用, 它在 Matlab 中以 libeng 动态库的形式存在。但它又有别于一般的函数库, 最重要的是: 首先, C 程序必须遵从一定的顺序操作引擎, 即打开引擎、引擎函数使用、关闭引擎; 其次, Matlab 引擎与 C 程序之间数据的交换是通过一种 Matlab 专门定义的对象类型——Matlab 数组进行的。而对此数组操作的函数库又定义在 libmx.dll 动态库中, 因此, C 程序编译时, 必须链入这两个动态库。由于使用 Matlab 引擎在实现 C 对 Matlab 的操作时, 使用的绝大多数库函数位于这两个动态库中, 因此, 动态库以静态使用为好。

1.2 使用引擎 C 对 Matlab 的编程

为了在 Windows 环境下用 CB4 实现 C 对 Matlab 的编程, 首先, 必须将 libeng.dll 及 libmx.dll 动态库的输入库 libeng.lib 及 libmx.lib 加入到要开发的应用中(使用 CB4 中 Project/Add to Project 菜单, .Lib 输入库的生成要使用 CB4 提供的 Implib.exe 程序); 其次, 应用的头文件中要包含 libeng.h 及 libmx.h, 即应用的程序

收稿日期: 2000-08-29

作者简介: 魏会贤(1962-), 男, 陕西宝鸡人, 讲师, 主要从事光纤通信、网管技术研究。

中加入: #include " < Matlab > extern libeng. h" 及 #include " < Matlab > extern mex. h"; 最后, 就可以在 CB4 控件的方法、事件中使用 Matlab 引擎库的函数了。如下程序是在按钮控件的单击事件中使用 Matlab 引擎的一个较完整过程, 程序绘制的是一条指数增长曲线, 注释完整展示了 Matlab 引擎的使用过程及方法。

```
void __fastcall TForm1::UseMatlabEngineClick(TObject * Sender)
{
    Engine * ep; //使用 Matlab 定义的对象类型定义指针
    mxArray * T; //使用 Matlab 定义的对象类型产生 Matlab 数组
    mxArray * result = NULL;
    double time [ 10 ] = { 0. 0, 1. 0, 2. 0, 3. 0, 4. 0, 5. 0, 6. 0, 7. 0, 8. 0, 9. 0 };
    ep = engopen( NULL ); //使用 Matlab 函数 engOpen 打开 Matlab 函数产生一维数组
    mxSetName( T, " T" ); //Matlab 函数
    memcpy( ( void * ) mxGetPr( T ), ( void * ) time, sizeof( time ) );
    engPutArray( ep, T ); //使用 Matlab 函数置变量 T 到 Matlab 工作组
    engEvalString( ep, " D = . 5. * ( - 9. 8) . * T. ^ 2; " ); //使用 Matlab 函数计算结果并画图
    engEvalString( ep, " plot( T, D ); " ); //画图
    engEvalString( ep, " title( ' Position vs. Time for a falling object ' ); " );
    engEvalString( ep, " xlabel( ' Time( seconds ) ' ); " );
    engEvalString( ep, " ylabel( ' Position( meters ) ' ); " );
    engEvalString( ep, " close; " ); //使用 Matlab 函数释放内存、关闭 Matlab 引擎。
    mxDestroyArray( T );
    mxDestroyArray( result );
    engClose( ep );
    return;
}
```

2 ActiveX 自动化实现 C 对 Matlab 的编程

2.1 ActiveX 自动化

ActiveX 自动化是一个应用实现对另一个应用控制的机制, 它本质上也是 C/S 模式。ActiveX 自动化对象通常在文件扩展或格式化参数下描述用户层功能, 它允许用户越过应用程序直接编写程序(宏程序), 利用此机制, 使用者在自己的应用中使用宏程序, 而无须创建自己的宏语言。利用 ActiveX 自动化机制, 可以在应用中使用拖放技术、使用 ActiveX 自动对象、创建可被另一个应用控制的自动对象(ActiveX 自动服务器)及使用 ActiveX 服务器(自动客户)。

一个应用访问 ActiveX 自动化服务器时, 必须先打开服务器, 这可使用 CB4 的 oleCreateobject 或 Createobject 函数, 它是 Variannt 类的静态成员, 它的参数是 ActiveX 自动化服务器的 ProgID。

打开的 ActiveX 自动化服务器可以利用 Variannt 的四个成员函数访问 ActiveX 自动化服务器。oleProcedure 被用来访问服务器无返回值的函数, oleFunction 被用来访问服务器有返回值的函数, olePropertiSet 被用来设置服务器的某个属性值, olePropertiGet 被用来获取服务器的某个属性值。

但 oleProcedure 及 oleFunction 不支持以名字传递参数, 只支持以位置传递参数及省略参数传递。其第一个参数是函数名, 第二个及以后是位置序列的参数。

另外, 编译器不检查自动对象名、函数名、过程或参数, 必要时, 可将有关代码包含在 try——catch 块中。

2.2 使用 ActiveX 自动化 C 对 Matlab 的编程

Matlab5. x 支持 ActiveX 自动化, 可以被当作服务器使用。可以用 CB4 开发 Windows 下的基于 ActiveX 自动化的客户程序。具体自动化时, 可以对服务器进行动态或静态实例化。静态实例化时, 程序设计阶段需要链入 Matlab 中的输入库 mlapp. tlb(通过 Project/Import 菜单), 如下程序是 CB4 使用 ActiveX 自动化技术的一个较完整过程, /* */ 中的程序为静态自动化 Matlab 时的代码, 注释完整展示了 ActiveX 自动化技术的使用过程及方法。

```

void _fastcall TForm1::UseMatlabEngineClick(TObject * Sender)
{
    try {
        Variant oleMatlab; result; //declare Automation variant
        oleMatlab = Variant::CreateObject(" Matlab. Application"); //产生自动对象
        result = oleMatlab.Exec(Function(" Execute") << " a = { 5 6 7 8 }"); //送数、计算并返回结果
        Labell - > Caption = AnsiString( result); //在 Labell 显示结果
        oleMatlab.Exec( Procedure(" Quit") ); //退出 Matlab
        /* TCOMIMLApp p;
        DIMLAppDisp pp;
        pp. BindDefault();
        wchar_t * ip = L" "; //必须的占位符
        wchar_t * f3 = L" a = { 1 2 3 4}";
        result = pp.Execute(L" a = { 1 2 3 4}"); //宽字符串
        Labell - > Caption = AnsiString( result);
        Beep();
        pp.Quit(); //Quit Matlab */
    } catch( Exception &E)
        ShowMessage( E. Message);
}

```

3 结束语

通过上述两种方法(也可以通过其它方法,如:DDE),可实现 C 对 Matlab 的编程。当然,上述方法不能离开 Matlab 的集成开发环境,如果要脱离 Matlab 的集成开发环境,则要使用 Mediva 开发软件平台。总之,这些方法拓展了 Matlab 的使用范围,方便了多用途应用领域的软件开发,也符合现代软件开发思想。上面两段程序均是作者用 Matlab5.3 及 CB4 在 Windows98 环境下调试通过的关键代码。

参考文献:

- [1] 柳承茂. Matlab5. X 入门与应用[M]. 北京:科学出版社,1999.
- [2] 张 剑. C++ 与 Windows 高级编程[M]. 北京:清华大学出版社,1994.
- [3] 许冬岗. 周涤非, 聂文强. Borland C++ Builder 编程技巧与实例[M]. 西安:西安电子科技大学出版社,1998.

Studies on Developing MATLAB with C under Windows System

WEI Hui-xian¹, LIU Hong-kun²

- (1. The telecommunication Engineering Institute of the Air force Engineering University, Xi'an 710077, China;
- 2. Navy Engineering University, Wuhan 430033, China)

Abstract: This paper discusses the principle of developing Matlab with C under Windows system. It also raises two methods by which c can develop Matlab program through using engine and Active X unattended technology. The linking method and procedure in developing software of multi application are completed. This method and procedure can also be used in developing other multi application software programs.

Key words: engine; ActiveX automation; Matlab; CB