

# 基于 ActiveX 的 Matlab 与 VB 集成的 多目标优化系统开发

Development of Multi - Objects Optimal System with Matlab  
and VB Based on ActiveX

荆 平 (天津师范大学城市与环境科学学院 300074)  
(南开大学环境科学与工程学院 天津 300074)  
朱 坦 (南开大学环境科学与工程学院 天津 300074)

**摘要:** MATLAB 具有强大的科学计算功能,含有大量多目标优化分析函数。依据 ActiveX 技术将 MATLAB 函数在 VB 环境下进行调用,进行多目标优化集成系统开发,实现数据的自动输入和输出功能,编制了 MATLAB 与 VB 的数据接口代码,提出程序设计时应注意的问题。系统中的分析计算功能由 MATLAB 函数实现,非常方便用户使用,对结果数据的二次分析提供了条件。

**关键词:** ActiveX MATLAB VB 系统开发

对于多目标优化问题的求解,可以运用 MATLAB 工具软件来实现。但在具体应用过程中, MATLAB 语言有特定的输入输出要求,对于不熟悉该语言的用户来说,存在一定的应用困难,而且为了分析不同条件下的各种方案的优化结果,约束条件不断发生变化,需要在 MATLAB 环境下依据格式规定重新输入数据,给用户带来一些麻烦。因此有必要开发多目标优化分析系统,进行 VB 与 MATLAB 的系统集成,运用 VB 编程实现数据的输入输出功能,而多目标优化分析由 MATLAB 来完成,用户只需改变相关的输入数据,就可获得计算结果。

## 1 MATLAB 多目标优化函数简介

多目标优化模型的目标函数至少由两个以上函数组成,约束条件由函数变量构成,对于目标优化模型的求解,主要采用评价函数法,依据几何或应用中的直观背景,构造评价函数,从而将多目标优化问题转化为单目标优化问题,再利用单目标优化的求解方法求出最优解<sup>[1]</sup>,在实际应用中,通常由计算机软件来实现。对

于区域环境与经济协调发展的多目标规划而言, MATLAB 函数 fgoalattain 非常适用,本文在建立基于 ActiveX 的 Matlab 与 VB 集成的多目标优化系统时,以该函数作为研究的主要对象。该函数对应的多目标优化模型的标准形式及函数格式如下<sup>[2]</sup>:

$$\begin{aligned} & \text{minimize } \gamma \text{ such that } F(x) - \text{weight} \cdot \gamma \leq \text{goal} \\ & c(x) \leq 0 \\ & \text{ceq}(x) = 0 \\ & A \cdot x \leq b \\ & \text{Aeq} \cdot x = \text{beq} \\ & \text{lb} \leq x \leq \text{ub} \end{aligned}$$

$x = \text{fgoalattain}(\text{fun}, x_0, \text{goal}, \text{weight}, A, b)$

$x = \text{fgoalattain}(\text{fun}, x_0, \text{goal}, \text{weight}, A, b, \text{Aeq}, \text{beq})$

$x = \text{fgoalattain}(\text{fun}, x_0, \text{goal}, \text{weight}, A, b, \text{Aeq}, \text{beq}, \text{lb}, \text{ub})$

$[x, \text{fval}] = \text{fgoalattain}(\dots)$

$[x, \text{fval}, \text{attainfactor}, \text{exitflag}] = \text{fgoalattain}(\dots)$

$[x, \text{fval}, \text{attainfactor}, \text{exitflag}, \text{output}] = \text{fgoalattain}(\dots)$

在应用 MATLAB 工具软件进行计算时,应先将模型转化成 MATLAB 的标准应用格式,建立表达式对应的 m 函数文件,在命令窗口输入初始条件,运行求解函数的相应表达式获得计算结果。式中参量的详细说明可参见 MATLAB 帮助文件。

## 2 基于 ActiveX 的 VB 对 MATLAB 的调用

Microsoft 的 ActiveX 自动化技术是一种允许一个应用程序(控制端)去控制另一个应用程序(服务器端)的协议。VB 支持 ActiveX 自动化控制端协议,而 MATLAB 支持 ActiveX 自动化服务器端协议,这样就可以在 VB 中调用 MATLAB 引擎(Engine),利用 ActiveX 通道与 Object. Execute 建立联系,直接使用 MATLAB 的函数库和图形库。

在 VB 应用程序和 MATLAB 之间的 ActiveX 自动化连接后,就可以在 VB 应用程序中调用 MATLAB 的命令,向 MATLAB 传送或从 MATLAB 接收矩阵数组,实现数据的输入输出功能,主要的方法有以下三种:

### (1) BSTR Execute ([in] BSTR Command) 方法

BSTR 表示宽字符串类型,它与 VB 存储字符串所采用的数据格式相同。该方法接收字符串命令并在 Matlab 中执行,将结果以字符串形式返回。执行本方法将调用 MATLAB 执行一条 Command 字符串决定的 MATLAB 命令,同时返回一个字符串表示命令的执行情况。任何在 MATLAB 命令窗口可被执行的字符串,均可以包括在 Command 字符串中。

### (2) void GetFullMatrix ([in] BSTR Name, [in] BSTR Workspace, [in,out] SAFEARRAY(double) \* pr, [in

### ,out] SAFEARRAY(double) \* pi) 方法

该方法用来将 MATLAB 中的一个矩阵变量传送给 VB 程序的一个 1 维或 2 维数组。用户可以从指定的 MATLAB 的工作空间中获取一个完整的一维或二维的实型或虚结构体 mxArray 数据,其实部和虚部被分别存放两个单独的 Double 型数组中。

### (3) void PutFullMatrix ([in] BSTR Name, [in] BSTR Workspace, [in] SAFEARRAY(double) pr, [in] SAFEARRAY(double) pi) 方法

该方法用于向 MATLAB 的工作空间中输出一个指定名为 mxArray 结构体,将 VB 程序的一个 1 维或 2 维数组传送到 MATLAB 中的一个矩阵变量中,各参数的含义及调用方法与 GetFullMatrix 方法类似。

## 3 多目标优化系统开发

由于 MATLAB 的运行环境具有特定的格式要求,在输入数据和输出数据时,要求操作者严格遵守,而集成系统只能让用户输入数据即可,否则就没有必要在 VB 环境下进行系统集成。基于 ActiveX 的 Matlab 与 VB 集成的多目标优化系统开发,主要应用上面所述的三种方法来实现程序的交互作用,但在具体开发过程中,还必须注意一些具体细节,如输入输出数据的数组大小及数据类型的定义等,如不注意将导致程序无法正常运行,下面通过开发出的一个具体示例,分别对程序设计的方法和技巧进行说明。示例的模型形式<sup>[3]</sup>和约束条件、m 函数文件及在 MATLAB 环境下的输入初值参见表 1。

表 1 MATLAB 环境下示例模型的求解设置

多目标优化模型	m 函数文件	输入初值的形式	Matlab 分析计算函数
$\text{Min } f_1(x) = 2x_1 + 5x_2$ $\text{Min } f_2(x) = 4x_1 + x_2$ $x_1 \leq 5$ $x_2 \leq 6$ $x_1 + x_2 \geq 7$ $x_1, x_2 \geq 0$	youhua.m Function f=youhua(x) $f(1) = 2 * x(1) + 5 * x(2);$ $f(2) = 4 * x(1) + x(2);$	$\text{goal} = [20 \ 12];$ $\text{weight} = [20 \ 12];$ $\text{x0} = [2 \ 5];$ $\text{A} = [10; 0 \ 1; -1 \ -1];$ $\text{b} = [5 \ 6 \ -7];$	$[x, \text{fval}, \text{attainfactor}, \text{exitflag}, \text{output}, \text{lamda}] = \text{fgoalattain}(@\text{youhua}, \text{x0}, \text{goal}, \text{weight}, \text{A}, \text{b}, [], [], [], [])$

### 3.1 通用变量的声名

在进行应用软件设计前,将 Matlab 对象及输入输出数据声明为通用窗体变量。

Dim Matlab As Object 定义对象

Dim mreal(1) As Double 定义传入数据实部数组大小

Dim mimag() As Double 定义传入数据虚部数组

Dim myue(2, 1) As Double 定义约束条件的输入数据

Dim q(2) As Double 定义约束数据

Dim real(1) As Double'传出实部数据的大小

### 3.2 原始数据的输入

要进行系统集成,首先要通过 VB 实现数据的自动化输入,由应用软件实现数据的读入,然后利用 MATLAB 的 GetFullMatrix 方法,将已有数据转化为 MATLAB 的特定数据类型及格式,这样就能够实现外部数据被 MATLAB 调用的目的。对于用户而言,办公软件 EXCEL 的操作都比较熟悉,而 EXCEL 电子表格可通过数据库技术在 VB 中加以引用,只要参照 MATLAB 的多目标优化函数的要求,将优化模型的参数输入到 EXCEL 表格中,就实现了原始数据输入的功能(见图 1)。

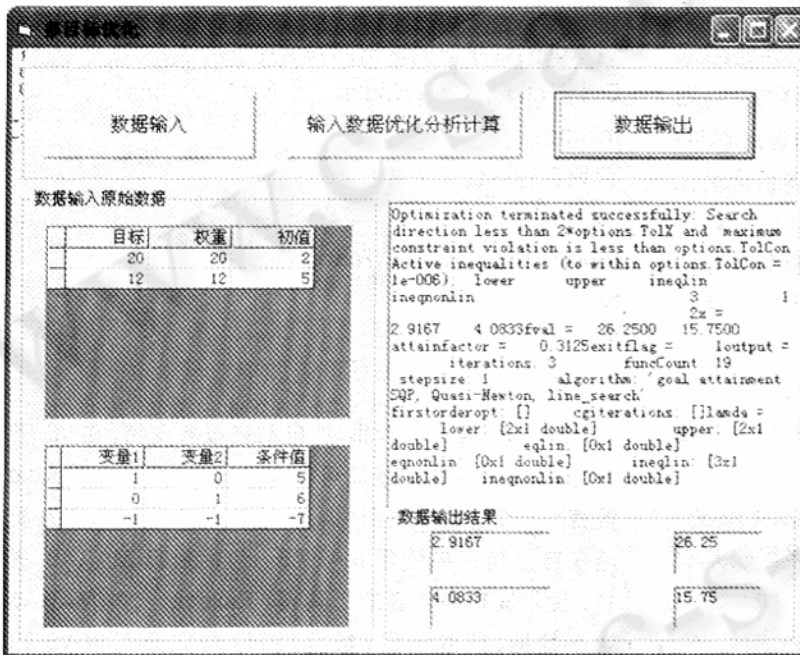


图 1 多目标优化模型集成系统

将 EXCEL 电子表格链接到 VB 中的具体操作为:在窗体中添加 Data1 控件和 MsFlexGrid1 控件,设置相关属性(见表 2)。完成相关设置后,就可以利用程序代码将 EXCEL 表格中的数据输入到 MATLAB 中,主要利用 ActiveX 技术,调用 MATLAB 自动化服务功能,结合 GetFullMatrix 方法来实现。在程序设计时,必须注意:在定义输入数据的数组大小时,如 mreal(1)等,必须和 EXCEL 中的传入数据个数绝对一致,而且数据类型必须定义为 Double,只有这样才能实现数据被 MATLAB 调用,否则程序将报错。

表 2 属性设置表

Data1	Connect	Excel 8.0;
	DatabaseName	C:\系数表.xls
	RecordSource	Sheet1\$
MsFlexGrid1	DataSource	Data1

实现数据输入功能的核心程序代码如下:

加载 Matlab 对象

```
Set Matlab = CreateObject("matlab.application")
```

读入目标初值和权重,因此例二者相同故读入一次

```
For i = 0 To 1
```

```
MSFlexGrid1.Col = 1
```

```
MSFlexGrid1.Row = i + 1
```

```
mreal(i) = Val(MSFlexGrid1.Text)
```

```
Next i
```

传入实部数据到 Matlab 中

```
Call Matlab.PutFullMatrix("goal", "base", mreal, mimag)
```

```
Call Matlab.PutFullMatrix("weight", "base", mreal, mimag)
```

传入变量数据初始值

```
For i = 0 To 1
```

```
MSFlexGrid1.Col = 3
```

```
MSFlexGrid1.Row = i + 1
```

```
mreal(i) = Val(MSFlexGrid1.Text)
```

```
Next i
```

传入变量数据初始值到 Matlab 中

```
Call Matlab.PutFullMatrix("x0", "base", mreal, mimag)
```

读取约束数据系数值

```
For i = 0 To 2
```

```
For j = 0 To 1
```

```
MSFlexGrid2.Col = j + 1
```

```
MSFlexGrid2.Row = i + 1
```

```
myue(i, j) = Val(MSFlexGrid2.Text)
```

```
Next j
```

```
Next i
```

传入约束数据系数值到 Matlab 的 A 中

```
Call Matlab.PutFullMatrix("A", "base", myue,
```

mimag)

读取约束数据限定值

```
For i = 0 To 2
```

```
    MSFlexGrid2.Col = 3
```

```
    MSFlexGrid2.Row = i + 1
```

```
q(i) = Val( MSFlexGrid2.Text)
```

```
Next i
```

传入约束数据限定值到 Matlab 的 b 中

```
Call Matlab.PutFullMatrix(" b", " base", q, mimag)
```

### 3.3 数据的分析计算

利用 ActiveX 技术调用 MATLAB 对象,进行数据的优化分析,系统必须依靠 MATLAB 应用软件,即系统运行的计算机主机必须安装 MATLAB 软件,此外还必须在 MATLAB 环境下,编写多目标模型的 M 函数文件,因此系统集成方式属于松散耦合,但能够成功调用 MATLAB 优化工具箱中的 fgoalattain 函数,进行多目标优化分析,便于用户应用。在进行系统集成时,将分析计算的不变部分采用代码一次性输入程序中,避免自行输入 MATLAB 的函数表达式,对用户而言,使用方便且不易出错。若按大多数文献所提到的通过文本框输入 MATLAB 代码,虽然软件适应性较强,但对用户的 MATLAB 语言有较高要求,而且容易因输入错误导致问题出现。

实现分析计算功能的核心程序代码如下:

```
Private Sub Command1 - Click ( ) ' 输入数据优化分析计算
```

```
    初始化对象
```

```
Set Matlab = CreateObject ( " matlab. application" )
```

```
    执行 Matlab 命令
```

```
Text2.text = Matlab. Execute ( " [ x, fval, attainfactor, exitflag, output, lamda ] =
```

```
fgoalattain ( @ youhua, x0, goal, weight, A, b, [ ], [ ], [ ], [ ] )"
```

```
End Sub
```

### 3.4 数据输出

在 MATLAB 中,以其特定的 x、fval 将多目标分析结果进行存储,因此在数据输出时,必须利用 MATLAB 的 GetFullMatrix 函数,将变量 x 和目标 fval 的结果数据输出给 VB 环境下的通用数组,再在 VB 环境下编制程序代码进行显示。在结果输出设计中,运用控件数组显

示数据,比较方面且易于用循环语句实现(见图1)。与数据输入的 PutFullMatrix 方法相似,运用 GetFullMatrix 方法时,获取结果数据的变量数组大小必须与数据结果绝对一致,类型必须定义为 Double,否则将导致错误出现。具体核心代码如下:

将 x 的结果数据输出到 real 中

```
Call Matlab. GetFullMatrix ( " x", " base", real, mimag)
```

读取变量的优化值并显示

```
For i = 0 To 1
```

```
Label1(i).Caption = Format(real(i), "#.####")
```

```
Next i
```

将 fval 的结果数据输出到 real 中

```
Call Matlab. GetFullMatrix ( " fval", " base", real, mimag)
```

读取目标的优化值并显示

```
For i = 0 To 1
```

```
Label2(i).Caption = Format(real(i), "#.####")
```

```
Next i
```

## 4 结束语

Matlab 实现了 ActiveX 自动化服务支持,在 VB 下通过 ActiveX 自动化接口可将 Matlab 作为 Visual Basic 语言的一个 ActiveX 部件调用,这样用户可以非常方便地在自己程序中使用 MATLAB,包括执行 MATLAB 命令,使用其功能丰富的工具箱 (Toolbox),向 MATLAB 输入数据,获取结果(数据,图形),充分利用 VB 和 MATLAB 的优点进行混合编程,可以编制出界面友好、计算及图形处理能力强大的软件,并在数据的输入输出方面,充分利用面向对象语言的特点,实现数据的自动化输入及输出结果的二次处理。

### 参考文献

- 1 飞思科技产品研发中心, MATLAB6.5 辅助优化计算与设计[M],北京 电子工业出版社,2003,47-51。
- 2 Matlab COM Builder User's Guide[M], MathWorks, 2002。
- 3 柯善军、冀小明,用 MATLAB 解决优化问题[J],西南民族大学学报,自然科学版,2005,31(2):294-298。