

第一章 前言

MXTOOL 是一个建立在矩阵运算基础上的多功能组合软件。

在现代科学计算中，矩阵运算起着越来越重要的作用。例如，在控制系统设计、数据处理、系统辨识等领域 或广而言之 对一切线性系统以及某些可用线性系统来逼近的非线性系统的分析与设计中，其基本运算乃是矩阵运算。

矩阵运算要比一般标量运算与矢量运算复杂得多。它除了一般的加、减、乘、数乘运算外，更重要的还有求逆、解线性方程组、求特征值与特征矢量等运算。近年来，发现矩阵的一些特殊分解使矩阵的性质表现得更为明确，其中特别重要的有 QR 分解与奇异值分解等。

一般说来，数字计算机是很适合于用来处理矩阵运算的。很多计算机配置的高级语言（例如 FORTRAN 语言和 C 语言）也都有能力处理矩阵运算。但是，目前情况下，要在数字计算机上运用矩阵运算来解决问题并不方便，这主要反映在下述几个方面：

(1) 在进行数值运算时，最方便的是使用计算器，几乎人人都会使用。即使对于较为复杂的数值运算，对一般非专业人员而言，使用可编程计算器也比使用计算机方便得多。但目前还没有矩阵计算器可供使用。

(2) 使用高级语言进行矩阵运算时，用户必须了解高级语言本身。而高级语言的学习及熟练使用，一般说来需要相当长时间的专门性训练。

(3) 即使采用了人机对话技术，矩阵的编辑工作，包括输入、显示、修改、查找、保存等 始终是不方便的。这主要受到计算机屏

幕大小的限制，很难将整个矩阵同时显示在一幅屏幕上。

鉴于上述情况，需要开发一种可类似于计算器那样使用的矩阵工具包（或称矩阵运算环境、矩阵计算器等）其主要特点应具有以下能力：（1）以矩阵元素为基础的全屏幕编辑；（2）用户不必了解任何高级语言，在解不太复杂的问题时，应使用通常的数学符号；（3）允许用户方便地扩充自己定义的数学符号（或称为函数）。MXTOOL 就是具有这些功能的一种以矩阵为运算对象的多功能组合软件。除了上述这些功能以外，对于熟悉程序语言的用户，MXTOOL 同时又提供了与一般高级语言类似的编程能力，因而它实际上又是一种“矩阵语言”。

§ 1.1 MXTOOL 的运行环境

运行 MXTOOL 需要有带有硬盘和图形监控器的 IBM-PC/XT 或 AT 或其兼容机。内存要在 512k 以上。MXTOOL 可以在有 8087 协处理器或没有 8087 协处理器的情况下工作，但没有 8087 协处理器时速度将明显减慢。因而推荐的工作方式是配有 8087 协处理器。

MXTOOL 要求 DOS-3.1 以上版本的软件支持。如果用户需要在屏幕上绘图，则应配置图形驱动卡。MXTOOL 适应的图形驱动卡种类有 CGA, MCGA, EGA, EGA64, EGAMONO, VGA 和 HERCMONO。

运行 MXTOOL 时，系统配置应保证能同时打开 25 个文件。为保证运行速度应设置 10 个以上用于输入输出的缓冲区。

§ 1.2 MXTOOL 的功能简介

MXTOOL 的基本功能包括基本矩阵运算、数值分析、绘图、控制系统的分析和设计、数字信号处理，以及系统辨识等。

基本矩阵运算包括两大类：结构运算与数学运算。结构运算包括矩阵的输入、显示、修改、加载、存储以及矩阵的合并、析取等。数学运算包括了线性代数中有关矩阵的大部分内容。例如，矩阵的求和、求积、数乘、求逆、求秩和范数、解线性代数方程组、求特征值和特征向量，以及多种矩阵分解，其中包括 LU 分解、CHOLESKY 分解、QR 分解、奇异值分解等。为保证这些基础运算的程序质量，这些基础程序参考了经过考验的“中国控制系统计算机辅助设计工程化软件系统”的线性代数库。

为方便用户输入、检查、修改数据，MXTOOL 配备了以矩阵元素为基础的全屏幕矩阵编辑器。

为直观地显示运算结果，MXTOOL 具有相当丰富的二维和三维绘图功能，并在必要时可在打印机上输出这些图形。

数值分析方面包括高次代数方程求根，插值和曲线拟合，数值积分、微分方程求解等。

控制系统的分析和设计方面，包括系统模型的变换和化简，系统的串联、并联和反馈连接，系统的脉冲响应、阶跃响应及对任意输入的响应，求根轨迹、Bode 图、乃氏图等，并具有相应的绘图功能。在现代控制理论方面，可以求系统的可控判别阵和可测判别阵，进行极点配置和线性系统二次指标最优调节器的设计等。

信号处理方面，包括求均值、方差、均方差、相关矩阵以及功率谱等数据的统计特性。可对信号进行 FFT 或逆 FFT，并具有用各种方法设计数字滤波器的功能。

系统辨识方面，可在输入-输出数据的基础上对系统参数进行辨识。建立系统的 ARMA 模型、输出-误差模型、线性模型等。

MXTOOL 除了具有上述基本功能外，还是一种具有开放性的矩阵开发环境。它提供了类似于高级语言的编程能力，用户可以在 MXTOOL 环境下很方便地用“矩阵语言”来扩充功能。

在 MXTOOL 环境下，为了使用户了解系统的运行情况，给出了关于系统运行情况的提示；在用户指令有错误时，给出包括错误

种类及错误位置的信息。

MXTOOL 可以和其他高级语言编写的程序相连接，在 MXTOOL 环境下，它允许用户执行所有的 DOS 命令，这样就极大地方便了用户在运算过程中了解外设的情况。并允许用户在 MXTOOL 环境下运行外存中的任一扩展名为 com 或 exe 的文件。在 C 语言或 PASCAL 语言中，也可以相当方便地执行 MXTOOL 语句。

最后，MXTOOL 还提供了在线 HELP，并给出了一系列示范程序供用户参考。

可以说，MXTOOL 是一个具有良好输入输出界面的可编的多功能矩阵计算器，或者说，是一种使用方便的矩阵开发工具。

§ 1.3 MXTOOL 的实现方案

MXTOOL 的功能十分广泛，要把所有这些功能所使用的程序同时放入内存对 PC 机说来几乎是不可能的。

在 MXTOOL 的实现方案中，把上述功能分三大类。第一类是其核心部分，主要是基本矩阵运算，它们的使用频率最高，因而要求运算速度快。这类程序是用 C 语言编制的，并在运行过程中常驻内存。第二类程序，如 MXTOOL1 程序、HELP 程序、EDIT 程序等，这类程序也是用 C 语言编制的，并以扩展名为 exe 的文件形式放在外存中，只在需要时才调入内存，用完后就从内存中删去。第三类程序是一些专用程序，例如控制系统分析与设计程序、复杂的矩阵运算、系统辨识等。它们主要是通过调用基本矩阵运算程序并配合一些控制语句（如 if 语句及各种形式的循环语句）来实现一些特定的功能。这类程序在 MXTOOL 中属外部命令或外部函数。它们以可读的形式放在外存中，只在第一次需要时才被调入内存，进行编译，并把编译结果储存在内存中，再次需要时就直接执行存在内存中的编译结果了。这类程序的执行速度比第一类

程序慢。

§ 1.4 MXTOOL 的主要文件

MXTOOL 的主要运行文件是 MXTOOL.EXE,它包含了上述第一类程序。另外有几个辅助运行文件,它们是 MXTOOL1 程序、HELP 程序、文本编辑程序等。

除了运行文件外, MXTOOL 还包含外部命令文件、外部函数文件、数据文件以及图形传送文件等。

外部命令文件以扩展名 p 为标志,以后也被称为“.p 文件”。实际上它就是一串命令,用户可以自己编写。调入运行时起的是宏命令的作用,与主程序之间不进行数据传送。

外部函数文件以扩展名 f 为标志,以后也被称为“.f 文件”。它也可由用户自己编写。一般一个 .f 文件对应一种功能。它包含函数样本,函数体与返回等几个部分。它运行时针对的是局部变量,因而对其他变量没有影响,返回后这些局部变量也不再有意义。外部函数可以与内部函数一样使用,同样可以进入表达式。这是 MXTOOL 具有开放性的主要手段。外部函数与内部函数的主要区别是执行速度较慢,其优点在于用户可以用大致与数学语言类同的表示方法来直接书写。且由于平时存放于外存,只在需要时调入,因而几乎不受内存容量的限制。

数据文件以扩展名 mat 为标志,它用来存储运行过程中产生的需要保存起来以备后用的各种变量。

图形传送文件以扩展名 pic 为标志。它用来存储有关图形的数据。

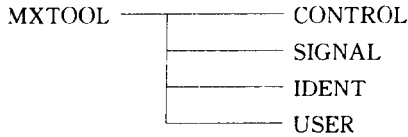
§ 1.5 MXTOOL 的屏幕设置

MXTOOL 的主要运行过程是在 25×80 的文字屏幕下进行

的。此时前 20 行为显示窗口 ;21 到 24 行为命令窗口 以高亮度标志 第 25 行为提示窗口。在显示窗口与命令窗口之间以 ESC 键切换。在要求输入命令时,将在命令窗口中出现光标提示。否则,表明系统处于显示状态。

§ 1.6 MXTOOL 的安装

完整的 MXTOOL 系统存贮在一张光盘或三寸软盘上。此盘上的目录结构如下:



在 CONTROL 子目录下装着有关控制系统分析与设计的外部函数 共 6 个 .p 文件和 56 个 .f 文件。

在 SIGNAL 子目录下装有关信号处理的外部函数,共 6 个 .p 文件和 38 个 .f 文件。

在 IDENT 子目录下装着有关系统辨识的外部函数共 8 个 .p 文件和 34 个 .f 文件。共 6 个 .p 文件和 34 个 .f 文件。

在 MXTOOL 目录下有 3 个可执行文件,3 个 C 语言接口文件,3 个 FORTRAN 语言接口文件,以及有关基本外部函数的 16 个 .p 文件和 70 个 .f 文件。3 个可执行文件是:

mxtool.exe.

mxtool1.exe (仅限于 1.5 版以上),

③ help.exe;

C 及 FORTRAN 语言的接口文件是:

① savemat.c.

loadmat.c.

③ testls.c.

④ savemat. for .

⑤ loadmat. for .

⑥ testls. for .

安装的具体方法是，在 C 盘的根目录下（一定要在 C 盘的根目录下）建立相应的目录结构并把全部文件复制到相应的目录下，安装即完成。（若可使用 Windows 的文件管理器，只要将目录 MXTOOL 复制到 C 盘的根目录下 安装即完成。）

在安装完毕后，用户还应把自己熟悉的文件编辑器以 edit.com 或 edit.exe 的名称复制到 MXTOOL 目录下 否则不能使用 edit 命令。

MXTOOL 的简单版本就是 mxtool.exe 将它放在任何目录下 或就放在软盘上 均可直接运行。但此时不能使用外部函数 也不能使用 help 命令。

§ 1.7 MXTOOL 的启动

在完成了 MXTOOL 的安装后 用户可进入 \MXTOOL\USER 子目录。再运行 MXTOOL 目录下的文件 MXTOOL.EXE 即键入 \mxtool\mxtool<CR> 就可启动 MXTOOL 了。此时屏幕的显示窗口的正中间显示信息

```
MXTOOL
copyright by Ye Qingkai
Peking University
Ver 2.0 NO 93011400
```

在显示窗口的右上角显示当前的时间。此时命令窗口中出现光标，表明系统处于命令状态，等待用户键入任何命令。

上述状态我们称为 MXTOOL 屏幕的初始状态。MXTOOL 在运行过程中，如果产生了在第八章 § 8.1 出错信息对照表中列出的错误，屏幕均将显示此初始状态。

§ 1.8 MXTOOL 的版本

MXTOOL 的简单版本或称教学版只有一个 `mxtool.exe` 文件,功能比较简单。MXTOOL 的早期版本(1991年版)已具有较完整的功能。MXTOOL 的当前版本(1993年版)改正了 1991年版中的某些错误并增加了一些重要的函数,它还增大了数据库容量,以便适应求解更复杂的课题。

第二章 数据的输入与修改

MXTOOL 中使用的基本数据结构是矩阵，标量和行、列矢量均作为特殊矩阵来处理。

MXTOOL 中可用多种方式输入与修改矩阵。在输入十分简单的矩阵，例如 3×3 的实矩阵时，方便的方法是由内部命令产生。在输入比较复杂的矩阵（行数与列数均小于 40 的实或复的矩阵）时，最方便的方法是使用显示窗口的矩阵编辑功能。当输入更为复杂的矩阵时，还可由执行外部命令文件或装入外部数据文件来得到。若要使用一些特殊矩阵，MXTOOL 还提供了产生零矩阵、全 1 矩阵、单位矩阵、对角矩阵 以及随机矩阵等特殊矩阵的内部函数。

§ 2.1 显示窗口的矩阵编辑

MXTOOL 的显示窗口可用来编辑和修改矩阵数据。显示窗口的大小为 20 行 80 列。头两行的显示内容为：

可用内存量： 提示信息：

矩阵名： 矩阵大小： 矩阵属性：

接着 16 行用来显示相应的矩阵的一部分。共显示 15 行 6 列矩阵元素，这些元素在矩阵中的位置可由用户指定。用高亮度的光标表明正在处理的元素的位置。光标位置可用各种移位键来移动。当光标位置移出屏幕之外时，MXTOOL 将自动调整屏幕上所显示的矩阵元素以使光标位置仍保留在屏幕上。

MXTOOL 的光标移位键如下：

CR：光标向右移动一个元素

→: 光标向右移动一个元素

←: 光标向左移动一个元素

↑: 光标向上移动一行

↓: 光标向下移动一行

pgup: 光标向上移动一页 (15 行)

pgdown: 光标向下移动一页 (15 行)

ctr →: 光标向右移动一页

ctr ←: 光标向左移动一页

Home: 光标移至第 1 行第 1 列

End: 光标移至所编辑矩阵的最后一个元素

F1: 光标移至下一行的第一个元素

F2: 显示上述各移位键功能

显示窗口的第 19 行为提示行。在正常输入状态下它提示三个信息:

(1) 光标所在位置。例如 (10,7) 表明当前光标处于所编辑矩阵的第 10 行第 7 列元素。

(2) 光标所在处的内容。若已输入过数据则显示此数据的实际数值 (16 位有效数字) 而在光标处显示的则是 5 位有效数字的近似值, 否则显示 Empty。

(3) 正在编辑的矩阵的编号, 即它是 MXT00L 这次运行中所使用的第几个矩阵。MXT00L 一次运行共可使用 256 个矩阵。编号为 200 ~ 255 的矩阵为全局变量, 应尽可能少用。

显示窗口的第 20 行为输入行。在正常输入状态下用来编辑光标所示位置处矩阵元素的数值。它可以是 MXT00L 的任一表达式。当按下回车键后, MXT00L 计算此表达式, 并将计算结果 (为一矩阵) 的第一行第一列元素的值作为光标所在处矩阵元素的值。此时可用的编辑功能键有:

→: 光标向右移动一位

←: 光标向左移动一位

Del: 删除光标处的字符

Ins: 改变为插入状态, 或退出插入状态

BS: 退格, 即删除光标左边的字符

Home: 光标移至行首

End: 光标移至行末

Enter: 结束输入

MXTOOL 的矩阵编辑器还具有 9 种复合编辑功能。它们是:

(1) “/n”: 更改所编辑的矩阵。此时提示行上出现 Enter the new name of the matrix, 要求输入需编辑的矩阵的名称。若此矩阵已经存在, 则将此矩阵显示出来供进一步编辑; 否则, 产生一个新矩阵等待输入。

(2) “/a”: 确定所编辑矩阵的属性。此时在提示行上出现 Enter the new attribute of the matrix, 要求输入相应属性。R 表示实矩阵, C 表示复矩阵。编辑复矩阵时, 复数的表示方法为(实部 虚部)。

(3) “/r”: 行编辑。此时提示行上出现 Insert Delete。按 d 键将删去光标所在行, 按 i 键将在光标所在行插入一空行, 以供进一步编辑。

(4) “/c”: 列编辑。此时提示行上出现 Insert Delete。按 d 键将删去光标所在列, 按 i 键将在光标所在列插入一空列, 以供进一步编辑。

(5) “/g”: 转移光标到指定位置。此时提示行上出现 Enter the cell to go to:, 要求输入所需转移去的位置的行号与列号。执行结果使光标移至指定位置并将其显示于屏幕的适当位置处。输入方式应为 行号 列号。

(6) “/s” 将所编辑的矩阵存入 MXTOOL 的数据库中。

(7) “/l” 以所编辑的矩阵的名字为标志, 将 MXTOOL 的数据库中的相应矩阵重新显示出来以供编辑。

(8) “/e”: 清除显示窗口中各矩阵元素的值以便重新编辑。

(9) “/q”退出矩阵编辑 返回至 MXTOOL 的初始状态 但不破坏当时的数据。

最后,ESC 键用来退出矩阵编辑状态,回到命令状态。

假定需要输入一个名为 ABC 的 3×3 实矩阵:

$$\begin{bmatrix} 1 & 2 & 3 \\ 5 & 6 & 8 \\ 4 & 3 & 7 \end{bmatrix}。$$

首先,观察命令窗口中是否出现光标。若有,按<CR>键(即回车键)使其转入矩阵编辑状态。在按了“/n”键后输入 ABC<CR>,此时显示的矩阵名为 ABC,表明显示窗口中编辑器处于等待输入 ABC 矩阵的状态,高亮度光标位于第一行第一列处。

现在开始输入矩阵数据。输入 1<CR>后第一行第一列元素显示 1,同时高亮度光标移至第一行第二列处。接着输入的次序为:2<CR> 3<CR> F1.5<CR> .6<CR> .8<CR> F1.4<CR> .3<CR>,7<CR>。这样就完成了矩阵数据的输入。所显示的矩阵大小为 $3 \times$

最后,按“/s”键将所输入的 ABC 矩阵存入 MXTOOL 的数据库中以备后用。

如果在矩阵数据输入后,发现所输入的数据有错误,可以利用所提供的编辑功能改正错误后再存入数据库。

在编辑一个实矩阵时,若输入了一个复元素,则认为要输入的元素是该复数的实部。

在编辑一个复矩阵时,若输入了一个实元素,则认为要输入的复元素的虚部为零。

§ 2.2 用内部命令来输入矩阵

在 MXTOOL 中,允许进行矩阵结构运算,即可以把两个矩阵合并为一个矩阵,或从一个矩阵中取出一块来作为一个独立的

矩阵等。

在 MXTOOL 中用 [...] 表示矩阵结构运算。它们主要有三类。

1. 两个矩阵合并为一个矩阵

当两个矩阵用“;”分开时表明两个矩阵是按水平方向合并的。例如表达式 [A,B] 的结果为矩阵 [A B] 此时要求矩阵 A, B 有相同的行数。因而, 若

$$A = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}, \quad B = \begin{bmatrix} 5 & 6 \\ 7 & 8 \end{bmatrix},$$

则 [A,B] 为矩阵

$$\begin{bmatrix} 1 & 2 & 5 & 6 \\ 3 & 4 & 7 & 8 \end{bmatrix}。$$

又如表达式 [1,2,3,4] 产生一个 1×4 的矩阵, 即一个 4 维的行向量

$$[1 \ 2 \ 3 \ 4],$$

将其转置则可得到一个 4 维的列向量。

当两个矩阵用“;”分开时表明两个矩阵是按垂直方向合并的。例如, 表达式 [A;B] 的结果为矩阵

$$\begin{bmatrix} A \\ B \end{bmatrix}。$$

此时要求矩阵 A, B 有相同的列数。因而 若

$$A = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}, \quad B = \begin{bmatrix} 5 & 6 \\ 7 & 8 \end{bmatrix},$$

则 [A;B] 为矩阵

$$\begin{bmatrix} 1 & 2 \\ 3 & 4 \\ 5 & 6 \\ 7 & 8 \end{bmatrix}。$$

又如,表达式 $[1;2;3;4]$ 产生一个 4×1 的矩阵,即一个 4 维的列向量 将其转置可得到一个 4 维的行向量。

用这样的结构运算,下述表达式

$$[[1,2,3];[5,6,8];[4,3,7]]$$

或

$$[[1;5;4],[2;6;3],[3;8;7]]$$

或

$$[[1,5,4]';[2,6,3]';[3,8,7]']$$

均产生矩阵

$$\begin{bmatrix} 1 & 2 & 3 \\ 5 & 6 & 8 \\ 4 & 3 & 7 \end{bmatrix}。$$

使用此结构运算,可以用逐步构造的方式得到极其复杂的矩阵。

2. 从一个矩阵中取出一块来作为一个独立的矩阵

设 A 是一个 $n \times m$ 矩阵,则 $A[3]$ 代表 A 的第 3 行向量, $A[:,3]$ 代表 A 的第 3 列向量, $A[3][4]$ 代表 A 的第 3 行第 4 列元素。 $A[2:4][3:5]$ 代表 A 的第 2 至 4 行、第 3 至 5 列构成的 3×3 矩阵 当然 此时要求 $n \geq 4, m \geq 5$ 若要得到 A 的第 2 行以及第 4 行构成的 $2 \times m$ 矩阵 其表达式为

$$[A[2]; A[4]]$$

同样 要得到 A 的第 2 列以及第 4 列构成的 $n \times 2$ 矩阵 其表达式为

$$[A[:,2]; A[:,4]]$$

由此可见,利用上述两种结构运算,可以析出各种需要的子矩阵。

上述两种结构运算均适用于复矩阵。

3. 构造有规律的矩阵

在 `MXTOOI` 中 表达式

[1 : 5]

产生由 1 到 5 增量为 1 的行矢量

[1 2 3 4 5]

也可指定任何非 1 的增量 例如 表达式

[1 : 3 : 0.5]

产生行矢量

[1 1.5 2 2.5 3]

利用这种方法可以方便地造表。例如，下列三条 MXT00L 语句

$X = [0 : 1.41 : 0.2]'$;

$Y = \exp(-X) .* \sin(X)$;

$A = [X, Y]$;

产生的 A 矩阵为

$$\begin{bmatrix} 0 & 0 \\ 0.2 & 0.1627 \\ 0.4 & 0.2610 \\ 0.6 & 0.3099 \\ 0.8 & 0.3223 \\ 1.0 & 0.3096 \\ 1.2 & 0.2807 \\ 1.4 & 0.2430 \end{bmatrix},$$

其中第 1 列为自变量 x 的值 第 2 列为 $y = \exp(-x) \times \sin(x)$ 的对应值。

§ 2.3 产生特殊矩阵的内部函数

MXT00L 提供了产生特殊矩阵的 13 种内部函数。它们是^①。

^① 本书在叙述 MXT00L 提供的函数的功能时，若只用函数名 例如 ones，表明它可能有多种调用方式；若还给出参数，例如 ones(n)，则表示某一特定的调用方式。

(1) ones 产生实的全 1 阵。包括
ones(n) 给出一个 $n \times n$ 的全 1 阵；
ones(n, m) 给出一个 $n \times m$ 的全 1 阵；
ones(A) 给出一个与矩阵 A 同维的全 1 阵 此时 A 不能是标量。

(2) zeros 产生实的全 0 阵。包括
zeros(n) 给出一个 $n \times n$ 的全 0 阵；
zeros(n, m) 给出一个 $n \times m$ 的全 0 阵；
zeros(A) 给出一个与矩阵 A 同维的全 0 阵 此时 A 不能是标量。

(3) eye 产生实的单位阵。包括
eye(n) 给出一个 n 阶单位阵；
eye(n, m) 给出一个 $n \times m$ 矩阵 其主对角元素为 1 其他元素均为 0；

eye(A) 给出一个与矩阵 A 同维的单位阵 此时 A 不能是标量。

(4) rand 产生一个在 $[0, 1]$ 区间上均匀分布的实随机矩阵。包括

rand(n) 给出一个 $n \times n$ 随机方阵；

rand(n, m) 给出一个 $n \times m$ 随机矩阵；

rand(A) 给出一个与矩阵 A 同维的随机矩阵，此时 A 不能是标量。

(5) randn 产生一个正态分布 (均值为零 方差为 1) 的实随机矩阵。包括

randn(n) 给出一个 $n \times n$ 随机方阵；

randn(n, m) 给出一个 $n \times m$ 随机矩阵；

randn(A) 给出一个与矩阵 A 同维的随机矩阵，此时 A 不能是标量。

上述 5 种函数中的 n 与 m 均须大于或等于零；等于零时产生

空矩阵。

(6) $\text{diag}(A)$ 产生一个对角矩阵或取对角线。

当 A 为一行矢量或列矢量时，得到一个对角线元素为 A 的对角方阵；当 A 为一非矢量的矩阵时，得到以 A 的对角线元素组成的行矢量。例如：对于矩阵

$$A = \begin{bmatrix} 1 & 2 & 3 & 4 \\ 2 & 3 & 4 & 5 \\ 5 & 6 & 7 & 8 \end{bmatrix},$$

则 $\text{diag}(A)$ 为矩阵

$$[1 \quad 3 \quad 7];$$

而 $\text{diag}(\text{diag}(A))$ 为矩阵

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 3 & 0 \\ 0 & 0 & 7 \end{bmatrix}.$$

(7) $\text{mxtorow}(A)$ 将矩阵元素按行排列为一个行矢量。

例如 `MXTOOL` 语句

$$A = [[1 \ 2]; [3 \ 4]; [5, 6]];$$

$$B = \text{mxtorow}(A);$$

产生矩阵

$$A = \begin{bmatrix} 1 & 2 \\ 3 & 4 \\ 5 & 6 \end{bmatrix},$$

$$B = [1 \ 2 \ 3 \ 4 \ 5 \ 6].$$

(8) $\text{mxtocol}(A)$ 将矩阵元素按行排列为一个列矢量。

(9) $\text{rowtomx}(C, A)$ 将行矢量转换为一个矩阵。

设 A 为一个 $n \times m$ 矩阵。 C 为一个长度为 k 的行矢量。则 $\text{rowtomx}(C, A)$ 用 C 的前 $n \times m$ 个元素 (在 $k < n \times m$ 时以零补到 $n \times m$ 个元素) 按矩阵 A 的方式组成一个矩阵。例如 `MXTOOL` 语句。