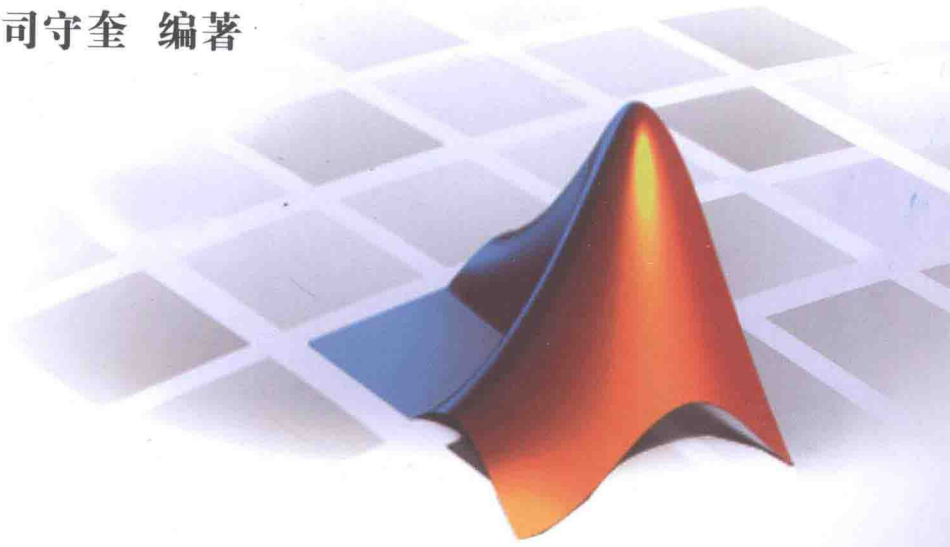
MATLAB
的工程数学应用

孙玺菁 司守奎 编著



国防工业出版社
National Defense Industry Press

第1章 线性代数

线性代数是处理矩阵和向量空间的数学分支，在很多实际领域都有应用。本科线性代数教学多偏重自身的理论体系，多强调基本概念，很少涉及应用定理及其证明，基本不涉及数值计算。对于线性代数中的基本概念和定理，参看同济大学数学系编写的《线性代数》(第六版)，本书中不再详述。特征值和特征向量一直是线性代数中应用广泛的一个关键内容，同时也是教学过程中的重点和难点，本章结合具体案例介绍了其在层次分析法、马尔科夫链、Pagerank 算法中的应用。本书没有涉及数值计算的相关理论，而是以课后习题为例题，结合 MATLAB 软件相关工具箱，讲解了线性代数相关问题的计算机实现，并补充了一些实际应用案例，从最基本和低年级学员最容易理解的角度学习 MATLAB 软件。

1.1 行 列 式

1.1.1 逆序数的计算

对于 n 个不同的元素，先规定各元素之间有一个标准次序(例如 n 个不同的自然数，可规定由小到大为标准次序)，于是在这 n 个元素的任一排列中，当某两个元素的先后次序与标准次序不同时，就说有 1 个逆序。一个排列中所有逆序的总数称为这个排列的逆序数。

下面介绍计算排列的逆序数的方法。

不失一般性，不妨设 n 个元素为 $1\sim n$ 的自然数，并规定由小到大为标准次序，设 $p_1p_2\cdots p_n$ 为这 n 个自然数的一个排列，考虑元素 $p_i(i=1,2,\cdots,n)$，如果比 p_i 大且排在 p_i 前面的元素有 t_i 个，就说 p_i 这个元素的逆序数是 t_i。全体元素的逆序数之总和就是这个排列的逆序数，即

$$t=t_1+t_2+\cdots+t_n=\sum_{i=1}^{n}t_i。$$

例 1.1 求排列 32514 的逆序数。

```
clc,clear,a=32514;
b=num2str(a);                    % 把数值型数据转换成字符型数据
for i=1:length(b);
    c(i)=str2num(b(i));
end
s=0;                             % 逆序数初始化
for i=2:length(b)
```

```
    s = s + sum(c(1:i-1) - c(i) > 0);
end
s                                    % 显示逆序数
```

求得排列 32514 的逆序数为 5。

1.1.2 行列式的计算及几何性质

MATLAB 计算行列式的命令为 det，该命令既可以计算数值行列式，也可以计算符号行列式的值。

例 1.2 计算下列行列式的值。

$$(1)\begin{vmatrix} 2 & 1 & 4 & 1 \\ 3 & -1 & 2 & 1 \\ 1 & 2 & 3 & 2 \\ 5 & 0 & 6 & 2 \end{vmatrix}; \quad (2)\begin{vmatrix} a & b & c \\ b & c & a \\ c & a & b \end{vmatrix}。$$

```
clc,clear
A1 = [2 1 4 1;3 -1 2 1;1 2 3 2;5 0 6 2];val1 = det(A1)
B = sym(A1),val2 = det(B)            % 转化为符号矩阵计算,结果精确为 0
syms a b c d
A2 = [a b c;b c a;c a b]
val = det(A2)
```

求得

$$\begin{vmatrix} 2 & 1 & 4 & 1 \\ 3 & -1 & 2 & 1 \\ 1 & 2 & 3 & 2 \\ 5 & 0 & 6 & 2 \end{vmatrix} = 0; \begin{vmatrix} a & b & c \\ b & c & a \\ c & a & b \end{vmatrix} = -a^3 - b^3 - c^3 + 3abc。$$

下面的应用给出了行列式的几何解释。

定理 1.1 若 $\boldsymbol{A}$ 是一个 2×2 矩阵，则由 $\boldsymbol{A}$ 的列确定的平行四边形的面积为 $|\det\boldsymbol{A}|$（这里 $\det\boldsymbol{A}$ 表示矩阵 $\boldsymbol{A}$ 的行列式），若 $\boldsymbol{A}$ 是一个 3×3 矩阵，则由 $\boldsymbol{A}$ 的列确定的平行六面体的体积为 $|\det\boldsymbol{A}|$。

证明：若 $\boldsymbol{A}$ 为二阶对角矩阵，定理显然成立。

$$\left|\det\begin{bmatrix} a & 0 \\ 0 & d \end{bmatrix}\right| = |ad| = \text{矩形的面积},$$

见图 1.1。若 $\boldsymbol{A}$ 不为对角情形，只需证 $\boldsymbol{A}=[\alpha_1,\alpha_2]$ 能变换成一个对角矩阵，同时既不改变相应的平行四边形面积又不改变 $|\det\boldsymbol{A}|$。当行列式的两列交换或一列的倍数加到另一列上时，行列式的绝对值不改变。同时容易看到，这样的运算足以能够使 $\boldsymbol{A}$ 变换成对角矩阵。由于列交换一点都不改变对应的平行四边形，所以只需证明下列在 $\mathbb{R}^2$ 和 $\mathbb{R}^3$ 中的向量的简单几何现象就足够了。

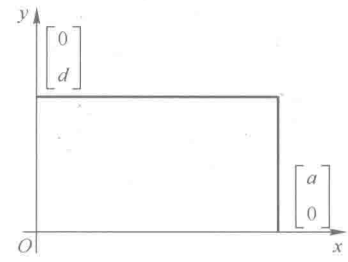


图 1.1 面积 $=|ad|$

引理 1.1 设 α_1 和 α_2 为非零向量，则对任意数 c，由 α_1 和 α_2 确定的平行四边形的面积等于由 α_1 和 $\alpha_2+c\alpha_1$ 确定的平行四边形的面积。

为了证明这个结论，可以假设 α_2 不是 α_1 的倍数，否则这个平行四边形将退化成面积为 0。若 L 是通过 O 和 α_1 的直线，则 α_2+L 是通过 α_2 且平行于 L 的直线，$\alpha_2+c\alpha_1$ 在此直线上，见图 1.2，点 α_2 和 $\alpha_2+c\alpha_1$ 到直线 L 具有相同的垂直距离，因此图 1.2 中的两个平行四边形具有相同的底边，即由 O 到 α_1 的线段，所以这两个平行四边形具有相同的面积，这就完成了$\mathbb{R}^2$的情形的证明。

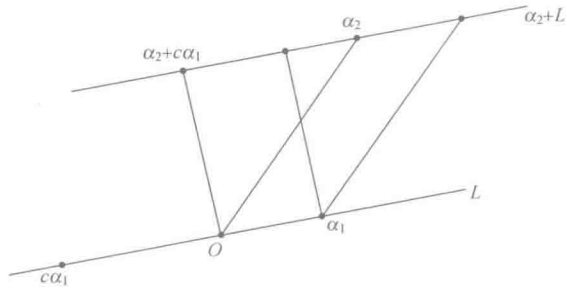


图 1.2　两个等面积的平行四边形

类似地可证明$\mathbb{R}^3$的情形。

例 1.3 计算由点(-2, -2)，(0,3)，(4, -1)和点(6,4)确定的平行四边形的面积。

解：先将此平行四边形平移到使原点作为其一顶点的情形。例如，将每个顶点坐标减去顶点(-2, -2)坐标。这样，新的平行四边形面积与原平行四边形面积相同，其顶点为(0,0)，(2,5)，(6,1)和(8,6)。

此平行四边形由 $\boldsymbol{A}=\begin{bmatrix}2 & 6\\ 5 & 1\end{bmatrix}$ 的列所确定，由于 $|\det\boldsymbol{A}|=|-28|=28$，所以所求的平行四边形的面积为 28。

计算的 MATLAB 程序如下：

```
clc,clear
a=[-2 -2;0 3;4 -1;6 4];          % 输入原来顶点坐标组成的矩阵
b=a-repmat(a(1,:),size(a,1),1)% 把其中的一个顶点平移到坐标原点
s=abs(det(b([2,3],:)))           % 计算面积
```

行列式可用于描述平面和$\mathbb{R}^3$中线性变换的一个重要几何性质。若 $\boldsymbol{T}$ 是一个线性变换，$\boldsymbol{S}$ 是 $\boldsymbol{T}$ 的定义域内的一个集合，用 $\boldsymbol{T}(\boldsymbol{S})$ 表示 $\boldsymbol{S}$ 中点的像集。那么，$\boldsymbol{T}(\boldsymbol{S})$ 的面积(体积)与原来的集合 $\boldsymbol{S}$ 的面积(体积)相对比有何变化呢?

定理 1.2 设 $\boldsymbol{T}:\mathbb{R}^2\to\mathbb{R}^2$ 是由一个 2×2 矩阵 $\boldsymbol{A}$ 确定的线性变换，若 $\boldsymbol{S}$ 是$\mathbb{R}^2$中一个平行四边形，则

$$\boldsymbol{T}(\boldsymbol{S})\text{的面积}=|\det\boldsymbol{A}|\cdot\boldsymbol{S}\text{的面积}。\tag{1-1}$$

若 T 是一个由 3×3 矩阵 $\boldsymbol{A}$ 确定的线性变换，而 $\boldsymbol{S}$ 是$\mathbb{R}^3$中的一个平行六面体，则

$$\boldsymbol{T}(\boldsymbol{S})\text{的体积}=|\det\boldsymbol{A}|\cdot\boldsymbol{S}\text{的体积}。\tag{1-2}$$

注 1.1 定理 1.2 的结论对$\mathbb{R}^2$中任意具有有限面积的区域或$\mathbb{R}^3$中具有有限体积的区域均成立。

例 1.4 若 a,b 是正数,求由方程$\frac{x_1^2}{a^2}+\frac{x_2^2}{b^2}=1$ 确定的椭圆为边界的区域 E 的面积。

解:可以肯定,E 是单位圆盘 D 在线性变换 $\boldsymbol{T}$ 下的像。这里 $\boldsymbol{T}$ 由矩阵 $\boldsymbol{A}=\begin{bmatrix}a & 0\\ 0 & b\end{bmatrix}$ 确定,这是因为若 $\boldsymbol{u}=\begin{bmatrix}u_1\\ u_2\end{bmatrix}$,$\boldsymbol{x}=\begin{bmatrix}x_1\\ x_2\end{bmatrix}$,且 $\boldsymbol{x}=\boldsymbol{A}\boldsymbol{u}$,则

$$u_1=\frac{x_1}{a},u_2=\frac{x_2}{b}。$$

从而把区域 E 映射到 $D:u_1^2+u_2^2\leqslant 1$。所以

$$\text{椭圆的面积}=\boldsymbol{T}(D)\text{的面积}=|\det\boldsymbol{A}|\cdot D\text{ 的面积}=ab\pi 1^2=\pi ab。$$

1.1.3 克拉默法则

克拉默法则:含有 n 个未知数 $x_1,x_2,\cdots,x_n$ 的 n 个线性方程的方程组

$$\begin{cases}a_{11}x_1+a_{12}x_2+\cdots+a_{1n}x_n=b_1,\\ a_{21}x_1+a_{22}x_2+\cdots+a_{2n}x_n=b_2,\\ \qquad\qquad\vdots\\ a_{n1}x_1+a_{n2}x_2+\cdots+a_{nn}x_n=b_n。\end{cases}\tag{1-3}$$

若线性方程组(1-3)的系数行列式不等于零,即

$$D=\begin{vmatrix}a_{11} & \cdots & a_{1n}\\ \vdots & \ddots & \vdots\\ a_{n1} & \cdots & a_{nn}\end{vmatrix}\neq 0,$$

则方程组(1-3)有唯一解

$$x_1=\frac{D_1}{D},x_2=\frac{D_2}{D},\cdots,x_n=\frac{D_n}{D}\tag{1-4}$$

其中,$D_j(j=1,2,\cdots,n)$是将行列式 D 中第 j 列的元素换成方程组右端的常数项 $b_1,b_2,\cdots,b_n$ 所得到的 n 阶行列式,即

$$D_j=\begin{vmatrix}a_{11} & \cdots & a_{1,j-1} & b_1 & a_{1,j+1} & \cdots & a_{1n}\\ \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots\\ a_{n1} & \cdots & a_{n,j-1} & b_n & a_{n,j+1} & \cdots & a_{nn}\end{vmatrix}。$$

例 1.5 解线性方程组

$$\begin{cases}2x_1+x_2-5x_3+x_4=8,\\ x_1-3x_2-6x_4=9,\\ 2x_2-x_3+2x_4=-5,\\ x_1+4x_2-7x_3+6x_4=0。\end{cases}$$

```
clc,clear
a=[2 1 -5 1;1 -3 0 -6;0 2 -1 2;1 4 -7 6];b=[8 9 -5 0]';
```

```
a1 = a;a1(:,1) = b;a2 = a;a2(:,2) = b;a3 = a;a3(:,3) = b;a4 = a;a4(:,4) = b;
for i = 1:4
    str = ['x',int2str(i),' = det(a',int2str(i),')/det(a)']
    % 上面是构造 xi = det(ai)/det(a)的字符串
    eval(str)          % 执行字符串对应的命令
end
```

例 1.6 λ 取何值时,齐次线性方程组

$$\begin{cases}(5-\lambda)x_1+2x_2+2x_3=0,\\ 2x_1+(6-\lambda)x_2=0,\\ 2x_1+(4-\lambda)x_3=0\end{cases}$$

有非零解?

如果该方程组有非零解,则它的系数行列式 $D=0$。而

$$\begin{aligned}D&=\begin{vmatrix}5-\lambda & 2 & 2\\ 2 & 6-\lambda & 0\\ 2 & 0 & 4-\lambda\end{vmatrix}\\ &=(5-\lambda)(6-\lambda)(4-\lambda)-4(4-\lambda)-4(6-\lambda)\\ &=(5-\lambda)(2-\lambda)(8-\lambda)\end{aligned}$$

由 $D=0$,解得 $\lambda=2$,$\lambda=5$ 或 $\lambda=8$。

```
clc,clear,syms t                % 方程组中的参数 lambda 用 t 表示
a = [5 - t 2 2;2 6 - t 0;2 0 4 - t];
D = det(a),DD = factor(D)       % 计算系数行列式的值,并进行因式分解
s = solve(D)                    % 求符号方程的根,也可以写作 s = solve(D == 0)
```

1.2 矩阵运算及线性变换

本节只简单地介绍矩阵的求逆、伴随矩阵及矩阵的一些变换操作,常用的一些命令见表 1.1。

表 1.1 矩阵求逆及一些变换操作命令

命令语法	功 能
inv(*A*)	求矩阵 *A* 的逆阵
pinv(*A*)	求矩阵 *A* 的 Moore - Penrose 伪逆
flip(*A*)	对矩阵 *A* 的行进行逆序变换,得到一个新矩阵
fliplr(*A*)	对矩阵 *A* 进行左右翻转
flipud(*A*)	对矩阵 *A* 进行上下翻转
rot90(*A*)	对矩阵 *A* 逆时针旋转 90°
tril(*A*)	提取矩阵 *A* 的下三角部分
triu(*A*)	提取矩阵 *A* 的上三角部分

（续）

命令语法	功　能
reshape(A,[m,n])	把矩阵 A 变换成 m 行 n 列的矩阵（变换前后矩阵的元素个数相同）
repmat(A,m)	把 A 作为一个子块，生成一个 $m \times m$ 分块矩阵（所有子块都是 A）
repmat(A,m,n)	把 A 作为一个子块，生成一个 $m \times n$ 分块矩阵（所有子块都是 A）
A'	求矩阵 A 的共轭转置矩阵
$A.'$	求矩阵 A 的转置矩阵

1.2.1　矩阵运算

1. 逆阵

例 1.7　设矩阵 A 和 B 满足关系 $AB = A + 2B$，已知 $A = \begin{bmatrix} 4 & 2 & 1 \\ 1 & 1 & 0 \\ 1 & 2 & 0 \end{bmatrix}$，求矩阵 B。

解：解矩阵方程，得 $B = (A - 2E)^{-1}A$，求得

$$B = \begin{bmatrix} 15/11 & 12/11 & 2/11 \\ 4/11 & 1/11 & 2/11 \\ 6/11 & -4/11 & 3/11 \end{bmatrix}。$$

```
clc,clear
a = [4,2,1;1,1,0;1,2,0];a = sym(a);     % 这里为了精确求解，转换为符号矩阵
b = inv(a - 2 * eye(3)) * a
```

2. 伴随矩阵

MATLAB 工具箱没有提供计算伴随矩阵的函数，可以利用伴随矩阵的定义和 Hamilton – Cayley 定理两种方法计算并自己编程求解。

例 1.8　求如下方阵 A 的伴随矩阵 A^*：

$$A = \begin{bmatrix} 3 & 1 & -1 & 2 \\ -5 & 1 & 3 & -4 \\ 2 & 0 & 1 & -1 \\ 1 & -5 & 3 & -3 \end{bmatrix}。$$

```
clc,clear
A = [3,1,-1,2;-5,1,3,-4;2,0,1,-1;1,-5,3,-3];
n = length(A);B = zeros(n);
for i = 1:n
    for j = 1:n
        Hij = A;Hij(i,:) = [];Hij(:,j) = [];
        B(j,i) = (-1)^(i + j) * det(Hij);
    end
end
disp('方阵 A 的伴随矩阵 B 如下所示:'),B
```

定理 1.3 （Hamilton－Cayley 定理）n 阶方阵 $\boldsymbol{A}$ 的特征多项式

$$f(\lambda)=|\lambda\boldsymbol{E}-\boldsymbol{A}|=\lambda^n+a_1\lambda^{n-1}+\cdots+a_{n-1}\lambda+a_n,$$

则

$$f(\boldsymbol{A})=\boldsymbol{A}^n+a_1\boldsymbol{A}^{n-1}+\cdots+a_{n-1}\boldsymbol{A}+a_n\boldsymbol{E}=0。\tag{1-5}$$

证明：设 $\boldsymbol{B}(\lambda)$ 为 $\lambda\boldsymbol{E}-\boldsymbol{A}$ 的伴随矩阵，则

$$\boldsymbol{B}(\lambda)(\lambda\boldsymbol{E}-\boldsymbol{A})=|\lambda\boldsymbol{E}-\boldsymbol{A}|\boldsymbol{E}=f(\lambda)\boldsymbol{E}。\tag{1-6}$$

由于矩阵 $\boldsymbol{B}(\lambda)$ 的元素都是行列式 $|\lambda\boldsymbol{E}-\boldsymbol{A}|$ 中的元素的代数余子式，因而都是 λ 的多项式，其次数都不超过 $n-1$，故由矩阵运算性质，$\boldsymbol{B}(\lambda)$ 可以写成

$$\boldsymbol{B}(\lambda)=\lambda^{n-1}\boldsymbol{B}_0+\lambda^{n-2}\boldsymbol{B}_1+\cdots+\boldsymbol{B}_{n-1},\tag{1-7}$$

这里各个 $\boldsymbol{B}_i$ 均为 n 阶数字矩阵。因此有

$$\boldsymbol{B}(\lambda)(\lambda\boldsymbol{E}-\boldsymbol{A})=\lambda^n\boldsymbol{B}_0+\lambda^{n-1}(\boldsymbol{B}_1-\boldsymbol{B}_0\boldsymbol{A})+\cdots+\lambda(\boldsymbol{B}_{n-1}-\boldsymbol{B}_{n-2}\boldsymbol{A})-\boldsymbol{B}_{n-1}\boldsymbol{A}。\tag{1-8}$$

另外，显然有

$$f(\lambda)\boldsymbol{E}=\lambda^n\boldsymbol{E}+a_1\lambda^{n-1}\boldsymbol{E}+\cdots+a_{n-1}\lambda\boldsymbol{E}+a_n\boldsymbol{E}。\tag{1-9}$$

由式(1-6)，式(1-8)和式(1-9)即得

$$\begin{cases}\boldsymbol{B}_0=\boldsymbol{E},\\ \boldsymbol{B}_1-\boldsymbol{B}_0\boldsymbol{A}=a_1\boldsymbol{E},\\ \vdots\\ \boldsymbol{B}_{n-1}-\boldsymbol{B}_{n-2}\boldsymbol{A}=a_{n-1}\boldsymbol{E},\\ -\boldsymbol{B}_{n-1}\boldsymbol{A}=a_n\boldsymbol{E}。\end{cases}\tag{1-10}$$

以 $\boldsymbol{A}^n,\boldsymbol{A}^{n-1},\cdots,\boldsymbol{A},\boldsymbol{E}$ 依次右乘式(1-10)的第一式，第二式，…，第 $n+1$ 式，并将它们加起来，则左边变成零矩阵，而右边即为 $f(\boldsymbol{A})$，故有 $f(\boldsymbol{A})=0$。证毕。

由上面的证明过程和式(1-7)，可知

$$\boldsymbol{B}(0)=(-\boldsymbol{A})^*=(-1)^{n-1}\boldsymbol{A}^*=\boldsymbol{B}_{n-1},\tag{1-11}$$

以 $\boldsymbol{A}^{n-1},\cdots,\boldsymbol{A},\boldsymbol{E}$ 依次右乘式(1-10)的第一式，第二式，…，第 n 式，并将它们加起来，得

$$\boldsymbol{B}_{n-1}=\boldsymbol{A}^{n-1}+a_1\boldsymbol{A}^{n-2}+a_2\boldsymbol{A}^{n-3}+\cdots+a_{n-1}\boldsymbol{E},\tag{1-12}$$

从而由式(1-11)和(1-12)，得到 $\boldsymbol{A}$ 的伴随矩阵

$$\boldsymbol{A}^*=(-1)^{n-1}(\boldsymbol{A}^{n-1}+a_1\boldsymbol{A}^{n-2}+a_2\boldsymbol{A}^{n-3}+\cdots+a_{n-1}\boldsymbol{E})。\tag{1-13}$$

例 1.9 （续例 1.8）用式(1-13)再计算矩阵 $\boldsymbol{A}$ 的伴随矩阵 $\boldsymbol{A}^*$。

```
clc,clear
A=[3,1,-1,2;-5,1,3,-4;2,0,1,-1;1,-5,3,-3];
n=length(A);p1=poly(A)                    % 求 A 的特征多项式
p2=p1(1:end-1);                           % 构造新的多项式
B=(-1)^(n-1)*polyvalm(p2,A)               % 计算矩阵多项式，得到伴随矩阵
```

3. 矩阵变换

例 1.10 把矩阵 $\boldsymbol{A}=\begin{bmatrix}a_{11}&a_{12}&a_{13}\\a_{21}&a_{22}&a_{23}\\a_{31}&a_{32}&a_{33}\end{bmatrix}$ 逆时针旋转 90°。

```
clc,clear,A = sym('a% d% d',3)      % 构造符号矩阵
B = rot90(A)                        % 把矩阵 A 逆时针旋转 90°
```

1.2.2 齐次坐标、线性变换与图像的空间变换

1. 齐次坐标、线性变换

$\mathbb{R}^2$中每个点(x,y)可以对应于$\mathbb{R}^3$中的$(x,y,1)$。它们位于xy平面上方1单位的平面上。我们称(x,y)有齐次坐标$(x,y,1)$,例如,点$(0,0)$的齐次坐标为$(0,0,1)$。点的齐次坐标不能相加,也不能乘以数,但它们可以乘以3×3矩阵来做变换。

例1.11 形如$(x,y)\mapsto(x+h,y+k)$的平移可以用齐次坐标写成$(x,y,1)\mapsto(x+h,y+k,1)$,这个变换可用矩阵乘法实现,即

$$\begin{bmatrix}1&0&h\\0&1&k\\0&0&1\end{bmatrix}\begin{bmatrix}x\\y\\1\end{bmatrix}=\begin{bmatrix}x+h\\y+k\\1\end{bmatrix}。$$

例1.12 $\mathbb{R}^2$中的任意线性变换都可通过齐次坐标乘以3×3矩阵实现。典型的例子:

$$\begin{bmatrix}\cos\varphi&-\sin\varphi&0\\\sin\varphi&\cos\varphi&0\\0&0&1\end{bmatrix};\quad\begin{bmatrix}0&1&0\\1&0&0\\0&0&1\end{bmatrix};\quad\begin{bmatrix}s&0&0\\0&t&0\\0&0&1\end{bmatrix}。$$

绕原点逆时针旋转角度φ　关于$y=x$的对称变换　x乘以s,y乘以t

复合变换相等于使用齐次坐标进行矩阵相乘。

例1.13 求出3×3矩阵,对应于先乘以0.3的倍乘变换,然后旋转90°,最后对图形的每个点的坐标加上$(-0.5,2)$后做平移。

解: 当$\varphi=\dfrac{\pi}{2}$时,$\sin\varphi=1$,$\cos\varphi=0$,由例1.11和例1.12,有

$$\begin{bmatrix}x\\y\\z\end{bmatrix}\xrightarrow{\text{缩小}}\begin{bmatrix}0.3&0&0\\0&0.3&0\\0&0&1\end{bmatrix}\begin{bmatrix}x\\y\\1\end{bmatrix}\xrightarrow{\text{旋转}}\begin{bmatrix}0&-1&0\\1&0&0\\0&0&1\end{bmatrix}\begin{bmatrix}0.3&0&0\\0&0.3&0\\0&0&1\end{bmatrix}\begin{bmatrix}x\\y\\1\end{bmatrix}$$

$$\xrightarrow{\text{平移}}\begin{bmatrix}1&0&-0.5\\0&1&2\\0&0&1\end{bmatrix}\begin{bmatrix}0&-1&0\\1&0&0\\0&0&1\end{bmatrix}\begin{bmatrix}0.3&0&0\\0&0.3&0\\0&0&1\end{bmatrix}\begin{bmatrix}x\\y\\1\end{bmatrix},$$

所以复合变换的矩阵为

$$\begin{bmatrix}1&0&-0.5\\0&1&2\\0&0&1\end{bmatrix}\begin{bmatrix}0&-1&0\\1&0&0\\0&0&1\end{bmatrix}\begin{bmatrix}0.3&0&0\\0&0.3&0\\0&0&1\end{bmatrix}=\begin{bmatrix}0&-0.3&-0.5\\0.3&0&2\\0&0&1\end{bmatrix}。$$

几何变换T把坐标(x,y)变换为坐标(X,Y),记作

$$(X,Y)=T(x,y),$$

具体数学表达式为

$$\begin{cases}X=a_0x+a_1y+a_2,\\Y=b_0x+b_1y+b_2。\end{cases}\tag{1-14}$$

写成矩阵形式

$$\begin{bmatrix} X \\ Y \\ 1 \end{bmatrix} = \begin{bmatrix} a_0 & a_1 & a_2 \\ b_0 & b_1 & b_2 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}。 \tag{1-15}$$

例 1.14 （1）求关于直线 $y=3x+5$ 对称的变换，对给定的圆 $(x-1)^2+y^2=1$，求其关于 $y=3x+5$ 的镜像曲线，并画出图形。

（2）利用求出的镜像曲线，利用反变换，求原来的曲线方程。

解：（1）设 $P_1(x_0,y_0)$ 是平面上的任意一点，它关于直线 $y=3x+5$ 的对称点为 $P_2(X,Y)$，则 P_1,P_2 的中点在直线 $y=3x+5$ 上，且 P_1P_2 与直线垂直，因而有

$$\begin{cases} \dfrac{Y+y_0}{2}=3\dfrac{X+x_0}{2}+5, \\ 3(Y-y_0)=-(X-x_0)。 \end{cases}$$

解得

$$\begin{cases} X=-\dfrac{4}{5}x_0+\dfrac{3}{5}y_0-3, \\ Y=\dfrac{3}{5}x_0+\dfrac{4}{5}y_0+1。 \end{cases} \tag{1-16}$$

式（1-16）即关于直线 $y=3x+5$ 对称的变换。

给定圆的参数方程为 $x=1+\cos t, y=\sin t, t\in[0,2\pi]$。把式（1-16）中的 x_0,y_0 分别代入 $x_0=1+\cos t, y_0=\sin t$，得到镜像曲线的参数方程为

$$\begin{cases} X=\dfrac{3}{5}\sin t-\dfrac{4}{5}\cos t-\dfrac{19}{5}, \\ Y=\dfrac{3}{5}\cos t+\dfrac{4}{5}\sin t+\dfrac{8}{5}。 \end{cases} \tag{1-17}$$

所画出的图形见图 1.3。（注：由于 x 轴、y 轴单位长度比并非 1∶1，故图形有所变形）

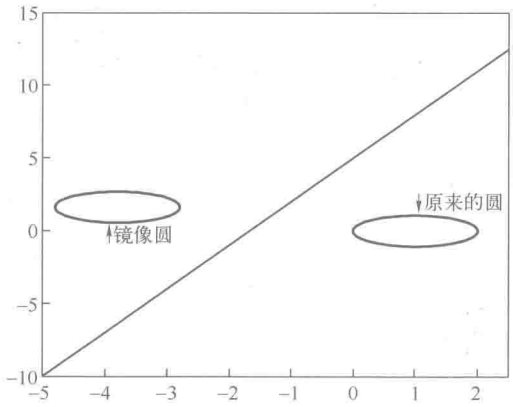


图 1.3　原来的圆与镜像圆

计算及画图的 MATLAB 程序如下：

```
clc,clear,axis square
```

```
syms x0 y0 X Y t
eq1 = (Y + y0)/2 == 3 * (X + x0)/2 + 5;
eq2 = 3 * (Y - y0) == - (X - x0);
[X,Y] = solve(eq1,eq2,X,Y)
X = subs(X,{x0,y0},{1 + cos(t),sin(t)})
Y = subs(Y,{x0,y0},{1 + cos(t),sin(t)})
t0 = 0:0.02:2 * pi;x = 1 + cos(t0);y = sin(t0);
plot(x,y,'k'),hold on,fplot(@ (x)3 * x + 5,[ - 5,2.5],'k')        % 画直线 y = 3x + 5
X = double(subs(X,t0));Y = double(subs(Y,t0));plot(X,Y,'k')      % 画镜像圆
text(1,2,'\downarrow 原来的圆','FontSize',12)
text( - 4,0,'\uparrow 镜像圆','FontSize',12)
```

(2) 对照式(1-15),变换(1-16)对应的变换矩阵

$$\boldsymbol{T}_1 = \begin{bmatrix} -\frac{4}{5} & \frac{3}{5} & -3 \\ \frac{3}{5} & \frac{4}{5} & 1 \\ 0 & 0 & 1 \end{bmatrix},$$

$\boldsymbol{T}_1$ 的逆矩阵

$$\boldsymbol{T}_2 = \boldsymbol{T}_1^{-1} = \begin{bmatrix} -\frac{4}{5} & \frac{3}{5} & -3 \\ \frac{3}{5} & \frac{4}{5} & 1 \\ 0 & 0 & 1 \end{bmatrix}。$$

$\boldsymbol{T}_2$ 对应的逆变换为

$$\begin{cases} x_0 = -\frac{4}{5}X + \frac{3}{5}Y - 3, \\ y_0 = \frac{3}{5}X + \frac{4}{5}Y + 1。 \end{cases} \tag{1-18}$$

把式(1-17)代入式(1-18),得到参数方程

$$\begin{cases} x_0 = 1 + \cos t, \\ y_0 = \sin t, \end{cases} t \in [0, 2\pi]。$$

即为所求的原来的曲线方程。

计算的 MATLAB 程序如下:

```
clc,clear
syms x0 y0 X Y t
T1 = [ - 4/5,3/5, - 3;3/5,4/5,1;0,0,1];T1 = sym(T1);T2 = inv(T1)
X = 3/5 * sin(t) - 4/5 * cos(t) - 19/5;
Y = 3/5 * cos(t) + 4/5 * sin(t) + 8/5;
xy01 = T2 * [X;Y;1];
x0 = xy01(1),y0 = xy01(2)
```

2. 齐次三维坐标与透视投影

类似于二维情形，称$(x,y,z,1)$是$\mathbb{R}^3$中点(x,y,z)的齐次坐标。一般地，若$H\neq 0$，则(X,Y,Z,H)是(x,y,z)的齐次坐标，且

$$x=\frac{X}{H},y=\frac{Y}{H},z=\frac{Z}{H}。\tag{1-19}$$

$(x,y,z,1)$乘以一个非零标量都得到一组(x,y,z)的齐次坐标。例如，$(10,-6,14,2)$和$(-15,9,-21,3)$都是$(5,-3,7)$的齐次坐标。

例 1.15 给出下列变换的 4×4 矩阵。

（1）绕 y 轴旋转 30°（习惯上，正角是从旋转轴（本例中是 y 轴）的正半轴向原点看过去的逆时针方向的角）。

（2）沿向量 $\boldsymbol{p}=[-6,4,5]$ 的方向平移。

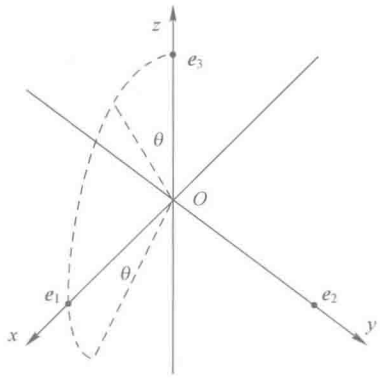


图 1.4 旋转变换示意图

解：（1）首先构造 3×3 矩阵表示旋转。如图 1.4 所示，向量 $\boldsymbol{e}_1$ 旋转到$[\cos 30°,0,-\sin 30°]=[\sqrt{3}/2,0,-0.5]$，向量 $\boldsymbol{e}_2$ 不变，向量 $\boldsymbol{e}_3$ 旋转到$[\sin 30°,0,\cos 30°]=[0.5,0,\sqrt{3}/2]$。这个旋转变换的标准矩阵为

$$\boldsymbol{A}=\begin{bmatrix}\sqrt{3}/2 & 0 & -0.5\\ 0 & 1 & 0\\ 0.5 & 0 & \sqrt{3}/2\end{bmatrix},$$

所以齐次坐标的旋转矩阵为

$$\boldsymbol{B}=\begin{bmatrix}\sqrt{3}/2 & 0 & -0.5 & 0\\ 0 & 1 & 0 & 0\\ 0.5 & 0 & \sqrt{3}/2 & 0\\ 0 & 0 & 0 & 1\end{bmatrix}。$$

（2）我们希望$(x,y,z,1)$映射到$(x-6,y+4,z+5,1)$，所求矩阵为

$$\begin{bmatrix}1 & 0 & 0 & -6\\ 0 & 1 & 0 & 4\\ 0 & 0 & 1 & 5\\ 0 & 0 & 0 & 1\end{bmatrix}。$$

三维物体在二维计算机屏幕上的表示方法是把它投影在一个可视平面上。为简单起见，设 xy 平面表示计算机屏幕，假设某一观察者的眼睛位置是$(0,0,d)$，透视投影把每个点(x,y,z)映射为点$(x^*,y^*,0)$，使这两点与观测者的眼睛位置（称为透视中心）在一条直线上，见图 1.5(a)。

xz 平面上的三角形画在图 1.5(b)中，由相似三角形知

$$\frac{x^*}{d}=\frac{x}{d-z},\quad x^*=\frac{dx}{d-z}=\frac{x}{1-z/d}。$$

类似地，有

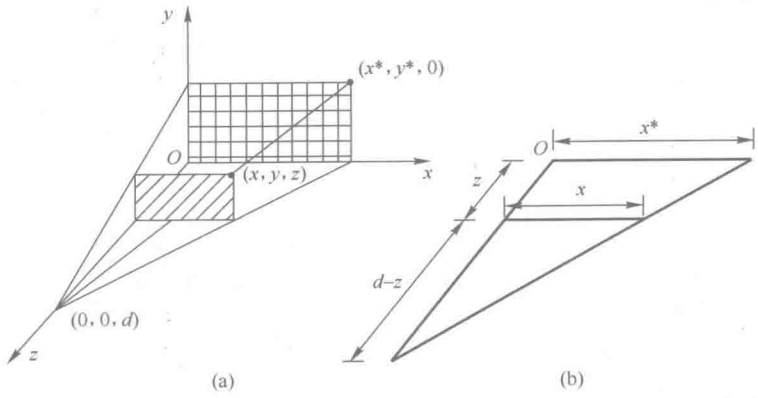


图 1.5　由(x,y,z)到$(x^*,y^*,0)$的透视投影

$$y^* = \frac{y}{1 - z/d}。$$

使用齐次坐标,可用矩阵表示透视投影,记此矩阵为 $\boldsymbol{P}$,$(x,y,z,1)$映射为

$$\left(\frac{x}{1-z/d}, \frac{y}{1-z/d}, 0, 1\right)。$$

把这个向量乘以 $1-z/d$,可用$(x,y,0,1-z/d)$作为齐次坐标的像,现在容易求出 $\boldsymbol{P}$。事实上,有

$$\boldsymbol{P}\begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & -1/d & 1 \end{bmatrix}\begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \begin{bmatrix} x \\ y \\ 0 \\ 1 - z/d \end{bmatrix}。$$

例 1.16　设 S 是顶点为$(3,1,5),(5,1,5),(5,0,5),(3,0,5),(3,1,4),(5,1,4),(5,0,4)$及$(3,0,4)$的长方体,求 S 在透视中心为$(0,0,10)$的透视投影下的像。

解:设 $\boldsymbol{P}$ 为投影矩阵,$\boldsymbol{D}$ 为用齐次坐标的 S 的数据矩阵,则 S 的像的数据矩阵为

$$\boldsymbol{PD} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & -1/10 & 1 \end{bmatrix}\begin{bmatrix} 3 & 5 & 5 & 3 & 3 & 5 & 5 & 3 \\ 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 \\ 5 & 5 & 5 & 5 & 4 & 4 & 4 & 4 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} 3 & 5 & 5 & 3 & 3 & 5 & 5 & 3 \\ 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0.5 & 0.5 & 0.5 & 0.5 & 0.6 & 0.6 & 0.6 & 0.6 \end{bmatrix}。$$

为得到$\mathbb{R}^3$坐标,使用式(1-19)。把每一列的前 3 个元素除以第 4 行的对应元素,得

顶点

$$\begin{matrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \end{matrix}$$
$$\begin{bmatrix} 6 & 10 & 10 & 6 & 5 & 8.3 & 8.3 & 5 \\ 2 & 2 & 0 & 0 & 1.7 & 1.7 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

计算的 MATLAB 程序如下:

```
clc,clear
a = [3 1 5;5 1 5;5 0 5;3 0 5;3 1 4;5 1 4;5 0 4;3 0 4]';
b = [a;ones(1,length(a))]
P = eye(4);P(3,3) = 0;P(4,3) = -1/10
c = P * b                                    % 求像点的齐次坐标
cc = c([1:3],:)./repmat(c(4,:),3,1)          % 求出像点的坐标
```

3. 图像的空间变换

在 MATLAB 的图像处理工具箱中提供了一个专门的函数 imwarp，用户可以定义参数实现多种类型的空间变换，包括仿射变换（如平移、缩放、旋转、剪切）、投影变换等。函数 imwarp 具体的调用格式如下。

B = imwarp(A,tform)：该函数中 A 为待变换的图像矩阵；tform 为执行空间变换的所有参数的结构体；B 为按照 tform 参数变换后的图像矩阵。

在 MATLAB 中，利用函数 imwarp 实现图像的空间变换时，需要先定义空间变换的参数。对于空间变换参数的定义，MATLAB 提供了相应的函数 affine2d，affined3d，fitgeotrans 等，它们的作用是创建进行空间变换的参数结构体。affine2d 的具体调用方式如下。

tform = affine2d(C)：该函数返回一个 N 维的仿射变换参数结构体 tform，输入参数 C 是一个 $(N+1)\times(N+1)$ 的矩阵。

用户结合使用函数 affine2d 和函数 imwarp，就可以灵活实现图像的线性变换，而变换的结果和变换参数结构体密切相关。以二维仿射变换为例，原图像 $f(x,y)$ 和变换后图像 $g(X,Y)$，仿射变换中原图像中某个像素点坐标 (x,y) 和变换后该像素点坐标 (X,Y) 满足关系式(1-14)，写成矩阵形式即满足(1-15)。

例 1.17 利用函数 imwarp，实现图像的旋转和缩放。

```
clc,clear
a = imread('peppers.png');                   % MATLAB 工具箱的图像文件
tf1 = affine2d([cosd(30),-sind(30),0;sind(30),cosd(30),0;0 0 1]);% 创建旋转
参数结构体
ta1 = imwarp(a,tf1);
tf2 = affine2d([5 0 0;0 10.5 0;0 0 1]);  % 创建缩放参数结构体
ta2 = imwarp(a,tf2);                         % 实现图像缩放
subplot(131),imshow(a),subplot(132),imshow(ta1),subplot(133),imshow(ta2)
```

原图像及变换后的图像见图 1.6。

(a) 原图像　(b) 旋转30°后的图像　(c) 拉伸后的图像

图 1.6　原图像及变换后的图像

1.2.3 密码与破译

1. 古典密码的基本概念及理论

一个密码系统(Cryptosystem)是一个五元组(P,C,K,E,D),其满足条件:

(1) P 是可能的明文的有限集(明文空间)。

(2) C 是可能密文的有限集(密文空间)。

(3) K 是一切可能密钥构成的有限集(密钥空间),其中的每一个密钥 k 均由加密密钥 k_e 和解密密钥 k_d 组成,记为 $k=(k_e,k_d)$。

(4) E 为加密算法,它是一族由 P 到 C 的加密变换(对于每一个具体的 k_e,E 确定出一个具体的加密函数)。

(5) D 为解密算法,它由一族由 C 到 P 的解密变换(对于每一个具体的 k_d,D 确定出一个具体的解密函数)。

在这里,对每一确定的密钥 $k=(k_e,k_d)$,$c=E(l,k_e)$,$l=D(c,k_d)=D(E(l,k_e),k_d)$,其中 l 为明文,c 为密文。

对于正整数 m,记集合 $Z_m=\{0,1,2,\cdots,m-1\}$。

定义 1.1 对于一个元素属于集合 Z_m 的 n 阶方阵 $\boldsymbol{A}$,若存在一个元素属于集合 Z_m 的方阵 $\boldsymbol{B}$,使得

$$\boldsymbol{AB}=\boldsymbol{BA}=\boldsymbol{E}(\mathrm{mod}m),$$

称 $\boldsymbol{A}$ 为模 m 可逆,$\boldsymbol{B}$ 为 $\boldsymbol{A}$ 的模 m 逆矩阵,记为 $\boldsymbol{B}=\boldsymbol{A}^{-1}(\mathrm{mod}m)$。

$\boldsymbol{E}(\mathrm{mod}m)$ 的意义是,每一个元素减去 m 的整数倍后,可以化成单位矩阵。例如

$$\begin{bmatrix}27 & 52\\ 26 & 53\end{bmatrix}(\mathrm{mod}26)=\boldsymbol{E}。$$

定义 1.2 对 Z_m 的一个整数 a,若存在 Z_m 的一个整数 b,使得 $ab=1(\mathrm{mod}m)$,称 b 为 a 的模 m 倒数或乘法逆,记为 $b=a^{-1}(\mathrm{mod}m)$。

可以证明,如果 a 与 m 无公共素数因子,则 a 有唯一的模 m 倒数(素数是指除了 1 与自身外,不能被其他正整数整除的正整数),反之亦然。例如 $3^{-1}(\mathrm{mod}26)=9$。利用这点,可以证明下述定理。

定理 1.4 元素属于 Z_m 的方阵 $\boldsymbol{A}$ 模 m 可逆的充要条件是,m 和 $\det\boldsymbol{A}$ 没有公共素数因子,即 m 和 $\det\boldsymbol{A}$ 互素。

显然,所选加密矩阵必须符合该定理的条件。

2. $\mathbf{Hill}_2$ 密码的数学模型

一般的加密过程是这样的:

明文⇒加密器⇒密文⇒普通信道⇒解密器⇒明文,

其中的"⇒普通信道⇒解密器"这个环节容易被敌方截获并加以分析。

在这个过程中,运用的数学手段是矩阵运算,加密过程的具体步骤如下:

(1) 根据明文字母的表值,将明文信息用数字表示,设明文信息只需要 26 个英文大写字母(也可以不止 26 个,如还有小写字母、数字、标点符号等),通信双方给出这 26 个字母表值,见表 1.2。

表 1.2　明文字母的表值

A	B	C	D	E	F	G	H	I	J	K	L	M
1	2	3	4	5	6	7	8	9	10	11	12	13
N	O	P	Q	R	S	T	U	V	W	X	Y	Z
14	15	16	17	18	19	20	21	22	23	24	25	0

（2）选择一个二阶可逆整数方阵 $\boldsymbol{A}$，称为Hill_2 密码的加密矩阵，它是这个加密体制的“密钥”（是加密的关键，仅通信双方掌握）。

（3）将明文字母逐对分组。Hill_2 密码的加密矩阵为二阶矩阵，则明文字母每两个一组（可以推广到Hill_n 密码，则 n 个明文字母为一组）。若最后一组仅有一个字母，则补充一个没有实际意义的哑字母，这样使每一组都由两个明文字母组成。查出每个明文字母的表值，构成一个二维列向量 $\boldsymbol{\alpha}$。

（4）$\boldsymbol{A}$ 乘以 $\boldsymbol{\alpha}$，得一个新的二维列向量 $\boldsymbol{\beta}=\boldsymbol{A\alpha}$，由 $\boldsymbol{\beta}$ 的两个分量反查字母表值得到的两个字母即为密文字母。

以上 4 步即为Hill_2 密码的加密过程。

解密过程，即为上述过程的逆过程。

例 1.18　明文为“HDSDSXX”，$\boldsymbol{A}=\begin{bmatrix}1 & 2\\ 0 & 3\end{bmatrix}$，求这段明文的$\text{Hill}_2$ 密文。

解：将明文相邻字母每 2 个分为一组：HD SD SX XX，最后一个字母 X 为哑字母，无实际意义。查表 1.2 得到每对的表值，并构造二维列向量，即

$$\begin{bmatrix}8\\4\end{bmatrix},\begin{bmatrix}19\\4\end{bmatrix},\begin{bmatrix}19\\24\end{bmatrix},\begin{bmatrix}24\\24\end{bmatrix},$$

将上述 4 个向量左乘矩阵 $\boldsymbol{A}$，得到 4 个二维列向量为

$$\begin{bmatrix}16\\12\end{bmatrix},\begin{bmatrix}27\\12\end{bmatrix},\begin{bmatrix}67\\72\end{bmatrix},\begin{bmatrix}72\\72\end{bmatrix},$$

作模 26 运算（每个元素都加减 26 的整数倍，使其化为 0 ~ 25 的一个整数），得

$$\begin{bmatrix}16\\12\end{bmatrix}(\text{mod}26)=\begin{bmatrix}16\\12\end{bmatrix},\begin{bmatrix}27\\12\end{bmatrix}(\text{mod}26)=\begin{bmatrix}1\\12\end{bmatrix},$$

$$\begin{bmatrix}67\\72\end{bmatrix}(\text{mod}26)=\begin{bmatrix}15\\20\end{bmatrix},\begin{bmatrix}72\\72\end{bmatrix}(\text{mod}26)=\begin{bmatrix}20\\20\end{bmatrix}。$$

反查表 1.2，得到每对表值对应的字母为 PL AL OT TT，这就得到了“HDSDSXX”密文。

计算的 MATLAB 程序如下：

```
clc,clear
s ='HDSDSXX';
s = [s,'X']                    % 补充哑字母'X'
L = length(s);                 % 计算字符总数
num = double(s) - 64;          % 字母编码
num = mod(num,26)              % mod26,变换 Z 的编码
mm = reshape(num,[2,L/2]);     % 把行向量变成两行的矩阵
```

```
A = [1 2;0 3];                    % 输入密钥矩阵
mw = A * mm                       % 求密文的编码值
mw = mod(mw,26)                   % mod26
mw(mw == 0) = 26;                 % 变换 Z 的编码值
mw = reshape(mw,[1,L]) + 64       % 变换到字母的 ASCII 码值
mwzf = char(mw)                   % 转换成密文的字符
mwzf(end) = []                    % 删除最后一个字符
```

例 1.19 甲方收到与之有秘密通信往来的乙方的一个密文信息,密文内容:

```
WKVACPEAOCIXGWIZUROQWABALOHDKCEAFCLWWCVLEMIMCC
```

按照甲方与乙方的约定,他们之间的密文通信采用 Hill_2 密码,密钥为二阶矩阵 $\boldsymbol{A}=\begin{bmatrix}1 & 2\\ 0 & 3\end{bmatrix}$,问这段密文的原文是什么?

解:所选择的明文字母共 26 个,$m=26$,26 的素数因子为 2 和 13,所以 Z_{26} 上的方阵 $\boldsymbol{A}$ 可逆的充要条件为 $\det\boldsymbol{A}(\mathrm{mod}m)$ 不能被 2 和 13 整除。设 $\boldsymbol{A}=\begin{bmatrix}a & b\\ c & d\end{bmatrix}$,若 $\boldsymbol{A}$ 满足上述定理 1.4 的条件,不难验证

$$\boldsymbol{A}^{-1}(\mathrm{mod}26)=(ad-bc)^{-1}(\mathrm{mod}26)\begin{bmatrix}d & -b\\ -c & a\end{bmatrix}(\mathrm{mod}26),$$

式中:$(ad-bc)^{-1}(\mathrm{mod}26)$ 为 $(ad-bc)$ 的模 26 倒数。

显然,$(ad-bc)(\mathrm{mod}26)$ 是 Z_{26} 中的数。Z_{26} 中有模 26 倒数的整数及其倒数可见表 1.3。

表 1.3 模 26 倒数表

a	1	3	5	7	9	11	15	17	19	21	23	25
$a^{-1}(\mathrm{mod}26)$	1	9	21	15	3	19	7	23	11	5	17	25

表 1.3 可用下列程序求得。

```
clc,clear
m = 26;
for a = 1:m
    for i = 1:m
        if mod(a * i,m) == 1
            fprintf('The Inverse (mod %d) of number:%d is:%d\n',m,a,i)
        end
    end
end
```

利用表 1.3 可以反演求出 $\boldsymbol{A}^{-1}(\mathrm{mod}26)$ 如下:

$$\boldsymbol{A}^{-1}(\mathrm{mod}26)=3^{-1}(\mathrm{mod}26)\begin{bmatrix}3 & -2\\ 0 & 1\end{bmatrix}(\mathrm{mod}26)=9\begin{bmatrix}3 & -2\\ 0 & 1\end{bmatrix}(\mathrm{mod}26)=\begin{bmatrix}27 & -18\\ 0 & 9\end{bmatrix}(\mathrm{mod}26)$$

$$=\begin{bmatrix}1 & 8\\ 0 & 9\end{bmatrix}=\boldsymbol{B}$$