常用结构二次幂零矩阵的构造

张承平

（海南热带海洋学院海洋信息工程学院，海南三亚 572022）

摘要：结构二次幂零矩阵构造的迭代法迭代简单、收敛速度快，且具有良好的并行性，但结构二次幂零矩阵有很多种，其形式和结构复杂，对于不同的线性方程组采用不同的结构二次幂零矩阵，迭代收敛速度可能不同。为选择合适的结构二次幂零矩阵，本文给出了一些常见的结构二次幂零矩阵的结构形式，以供实际计算中挑选合适的结构二次幂零矩阵。对于形式复杂的线性方程组的矩阵，当挑选比较困难时，本文给出了查找结构二次幂零矩阵的算法及 Matlab 程序。

关键词：计算数学；线性方程组；结构二次幂零矩阵；并行迭代法

中图分类号：O241.6　　**文献标识码**：A　　**文章编号**：1674-2850(2018)19-1927-07

Constructing of common structural 2-nilpotent matrices

ZHANG Chengping

(*School of Marine Information and Engineering*, *Hainan Tropical Ocean College*, *Sanya*, *Hainan* 572022, *China*)

Abstract: The iterative method constructed by structural 2-nilpotent matrices is a simple iteration which has fast convergence and good parallelism. But there are many kinds of structural 2-nilpotent matrices, and their forms and structures are complex. So different structural 2-nilpotent matrices are adopted for different linear equations, and the convergence rates of the iterations may be different. In order to choose the best structural 2-nilpotent matrix, some common structures of the structural 2-nilpotent matrices are given in this paper to select a suitable structural 2-nilpotent matrix for the actual calculation. When the matrix of a complex linear equations group is difficult to be selected, the algorithms for searching structural 2-nilpotent matrices and Matlab programs are also given in this paper.

Key words: computational mathematics; linear equations; structural 2-nilpotent matrix; parallel iterative method

0 引言

文献[1]～[3]给出了关于结构二次幂零矩阵的迭代方法和收敛定理。文献[4]对 Laplace 方程五点差分格式形成的线性方程组，构造其较好的对应结构二次幂零矩阵，通过数值实验，将其迭代法与雅可比迭代法比较，显示其良好的收敛效果。文献[5]指出矩阵与向量相乘可以采用并行计算编程，文献[6]给出了矩阵与向量相乘的并行计算编程方法，文献[7]对结构二次幂零矩阵迭代方法进行并行编程，数据实验结果表明其具有良好的并行性。文献[3]中的数据实验说明，不同的结构二次幂零矩阵构造的迭代法收敛速度可能不一样，因此构造合适的结构二次幂零矩阵是非常重要的。本文主要研究如何构造二次幂零矩阵，给出一些常用的结构二次幂零矩阵的结构形式，以及在一个矩阵中找到一个结构二次幂零矩阵的算法。

1 结构二次幂零矩阵的性质

结构二次幂零矩阵的条件为 $\boldsymbol{S}=(s_{ij})_{N\times N}$ 满足：对于 $\forall s_{ij}\neq 0$，则一定有 $s_{jk}=0$，且 $s_{ki}=0$，其中 $k=1,2,\cdots,N$.

作者简介：张承平（1976—），男，副教授，主要研究方向：数值数学与应用数学. E-mail: godloved@126.com

性质 1 $\boldsymbol{S}$ 为结构二次幂零矩阵，则 $\boldsymbol{S}^{\mathrm{T}}$ 也为结构二次幂零矩阵。

性质 2[3] $\boldsymbol{S}$ 为结构二次幂零矩阵，则 $(\boldsymbol{D}+\boldsymbol{S})^{-1}=\boldsymbol{D}^{-1}(\boldsymbol{D}-\boldsymbol{S})\boldsymbol{D}^{-1}$，其中 $\boldsymbol{D}$ 为与 $\boldsymbol{S}$ 同阶的可逆对角矩阵。

注意：当 $\boldsymbol{S}$ 为纯 k 阶子块的结构二次幂零矩阵、$\boldsymbol{D}$ 为同阶的纯 k 阶子块的可逆分块对角矩阵时，性质 2 也成立。

2 结构二次幂零矩阵的构造

根据线性方程组的结构特点，人工构造合适的结构二次幂零矩阵（注意：按照子块分类）。

2.1 k 阶子块

1）0 阶子块：全零矩阵 $\boldsymbol{S}=0$（迭代法简化为雅可比迭代法）.

2）1 阶子块：

第 1 列开始（偶数行的奇数列），

$$\boldsymbol{S}=\begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & \cdots \\ * & 0 & * & 0 & * & 0 & \cdots \\ 0 & 0 & 0 & 0 & 0 & 0 & \cdots \\ * & 0 & * & 0 & * & 0 & \cdots \\ 0 & 0 & 0 & 0 & 0 & 0 & \cdots \\ * & 0 & * & 0 & * & 0 & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \end{pmatrix}_{N\times N}. \tag{1}$$

第 1 行开始（奇数行的偶数列），

$$\boldsymbol{S}=\begin{pmatrix} 0 & * & 0 & * & 0 & * & \cdots \\ 0 & 0 & 0 & 0 & 0 & 0 & \cdots \\ 0 & * & 0 & * & 0 & * & \cdots \\ 0 & 0 & 0 & 0 & 0 & 0 & \cdots \\ 0 & * & 0 & * & 0 & * & \cdots \\ 0 & 0 & 0 & 0 & 0 & 0 & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \end{pmatrix}_{N\times N}. \tag{2}$$

注：*表示元素可以不取零，行和列对应互为转置。

3）2 阶子块：

第 1 列开始，

$$\boldsymbol{S}=\begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots \\ * & * & 0 & 0 & * & * & 0 & 0 & * & * & \cdots \\ * & * & 0 & 0 & * & * & 0 & 0 & * & * & \cdots \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots \\ * & * & 0 & 0 & * & * & 0 & 0 & * & * & \cdots \\ * & * & 0 & 0 & * & * & 0 & 0 & * & * & \cdots \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \end{pmatrix}_{N\times N}. \tag{3}$$

第 1 行开始，

$$
S=\begin{pmatrix}
0 & 0 & * & * & 0 & 0 & * & * & 0 & 0 & \cdots \\
0 & 0 & * & * & 0 & 0 & * & * & 0 & 0 & \cdots \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots \\
0 & 0 & * & * & 0 & 0 & * & * & 0 & 0 & \cdots \\
0 & 0 & * & * & 0 & 0 & * & * & 0 & 0 & \cdots \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots \\
0 & 0 & * & * & 0 & 0 & * & * & 0 & 0 & \cdots \\
0 & 0 & * & * & 0 & 0 & * & * & 0 & 0 & \cdots \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots &
\end{pmatrix}_{N\times N} . \qquad (4)
$$

注：行和列对应互为转置。

4）3 阶子块：

第 1 列开始，

$$
S=\begin{pmatrix}
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots \\
* & * & * & 0 & 0 & 0 & * & * & * & 0 & 0 & 0 & \cdots \\
* & * & * & 0 & 0 & 0 & * & * & * & 0 & 0 & 0 & \cdots \\
* & * & * & 0 & 0 & 0 & * & * & * & 0 & 0 & 0 & \cdots \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots \\
* & * & * & 0 & 0 & 0 & * & * & * & 0 & 0 & 0 & \cdots \\
* & * & * & 0 & 0 & 0 & * & * & * & 0 & 0 & 0 & \cdots \\
* & * & * & 0 & 0 & 0 & * & * & * & 0 & 0 & 0 & \cdots \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots &
\end{pmatrix}_{N\times N} . \qquad (5)
$$

第 1 行开始，

$$
S=\begin{pmatrix}
0 & 0 & 0 & * & * & * & 0 & 0 & 0 & * & * & * & \cdots \\
0 & 0 & 0 & * & * & * & 0 & 0 & 0 & * & * & * & \cdots \\
0 & 0 & 0 & * & * & * & 0 & 0 & 0 & * & * & * & \cdots \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots \\
0 & 0 & 0 & * & * & * & 0 & 0 & 0 & * & * & * & \cdots \\
0 & 0 & 0 & * & * & * & 0 & 0 & 0 & * & * & * & \cdots \\
0 & 0 & 0 & * & * & * & 0 & 0 & 0 & * & * & * & \cdots \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots &
\end{pmatrix}_{N\times N} . \qquad (6)
$$

注：行和列对应互为转置。

5）3阶以上子块本文不再列举。

2.2 $k_1 \times k_2$ 子块

1）下三角型：

$$S = \begin{pmatrix} 0 & 0 \\ B_{(N-k)\times k} & 0 \end{pmatrix}, \tag{7}$$

其中，$B_{(N-k)\times k}$ 可以不取零，$k = 1, 2, \cdots, N-1$.

①$(N-k)\times k$ 型，如 $k=1$，即 $(N-1)\times 1$ 型，

$$S = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots \\ * & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots \\ * & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots \\ * & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots \\ * & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \ldots \\ * & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots \\ * & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots \\ * & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots \\ * & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots \\ * & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \end{pmatrix}_{N\times N}. \tag{8}$$

②$(N-k)\times k$ 型，如 $k=2$，即 $(N-2)\times 2$ 型，

$$S = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots \\ * & * & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots \\ * & * & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots \\ * & * & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \ldots \\ * & * & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots \\ * & * & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots \\ * & * & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots \\ * & * & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots \\ * & * & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \end{pmatrix}_{N\times N}. \tag{9}$$

③$(N-k)\times k$ 型，如 $k = \left[\frac{N}{2}\right]$，即 $\left(N - \left[\frac{N}{2}\right]\right)\times\left[\frac{N}{2}\right]$ 型，当 N=13 时，

$$
\boldsymbol{S}=\begin{pmatrix}
0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0\\
&&*&*&*&*&0&0&0&0&0&0&0\\
&&*&*&*&*&0&0&0&0&0&0&0\\
&&*&*&*&*&0&0&0&0&0&0&0\\
&&*&*&*&*&0&0&0&0&0&0&0\\
&&*&*&*&*&0&0&0&0&0&0&0\\
&&*&*&*&*&0&0&0&0&0&0&0\\
&&*&*&*&*&0&0&0&0&0&0&0
\end{pmatrix}_{13\times 13}. \tag{10}
$$

④ $(N-k)\times k$ 型，如 $k=N-1$，即 $1\times(N-1)$ 型，

$$
\boldsymbol{S}=\begin{pmatrix}
0&0&0&0&0&0&0&0&0&0&\cdots\\
0&0&0&0&0&0&0&0&0&0&\cdots\\
0&0&0&0&0&0&0&0&0&0&\cdots\\
0&0&0&0&0&0&0&0&0&0&\cdots\\
0&0&0&0&0&0&0&0&0&0&\ldots\\
0&0&0&0&0&0&0&0&0&0&\cdots\\
0&0&0&0&0&0&0&0&0&0&\cdots\\
0&0&0&0&0&0&0&0&0&0&\cdots\\
\vdots&\vdots&\vdots&\vdots&\vdots&\vdots&\vdots&\vdots&\vdots&\vdots&\\
0&0&0&0&0&0&0&0&0&0&\cdots\\
&&*&*&*&*&*&*&*&0&\cdots
\end{pmatrix}_{N\times N}. \tag{11}
$$

2）上三角型（与下三角型互为转置）：

$$
\boldsymbol{S}=\begin{pmatrix}0&\boldsymbol{B}_{k\times(N-k)}\\0&0\end{pmatrix}, \tag{12}
$$

其中，$\boldsymbol{B}_{k\times(N-k)}$ 可以不取零，$k=1,2,\cdots,N-1$.

3）混合型：

混合型结构复杂、样式多，本文没有办法一一列举，故给出程序得到一个结构二次幂零矩阵，

$$
\boldsymbol{S}=\begin{pmatrix}
0&0&*&0&*&*&0&*&*&0&*&0&\cdots\\
0&0&*&0&*&*&0&*&*&0&*&0&\cdots\\
0&0&0&0&0&0&0&0&0&0&0&0&\cdots\\
0&0&*&0&*&*&0&*&*&0&*&0&\cdots\\
0&0&0&0&0&0&0&0&0&0&0&0&\cdots\\
0&0&0&0&0&0&0&0&0&0&0&0&\cdots\\
0&0&*&0&*&*&0&*&*&0&*&0&\cdots\\
0&0&0&0&0&0&0&0&0&0&0&0&\cdots\\
0&0&0&0&0&0&0&0&0&0&0&0&\cdots\\
0&0&*&0&*&*&0&*&*&0&*&0&\cdots\\
0&0&0&0&0&0&0&0&0&0&0&0&\cdots\\
0&0&*&0&*&*&0&*&*&0&*&0&\cdots\\
\vdots&\vdots&\vdots&\vdots&\vdots&\vdots&\vdots&\vdots&\vdots&\vdots&\vdots&\vdots&
\end{pmatrix}_{N\times N}. \tag{13}
$$

注：混合型比较复杂，规律不明显，通常可根据结构二次幂零矩阵的条件，通过计算机编程寻找满足条件的矩阵。

3 计算机查找结构二次幂零矩阵的算法

自动选择结构二次幂零矩阵的算法如下：

步骤 1 设 $\boldsymbol{S}$ 为要在 $\boldsymbol{A}$ 矩阵中选择元素的同阶全 0 矩阵。

步骤 2 令 $\boldsymbol{B}=\boldsymbol{A}$，令 $\boldsymbol{B}$ 的对角元素全为 0.

步骤 3 查找 $\boldsymbol{B}$ 中满足条件的元素（例如元素离对角线最近或绝对值最大者），如果满足条件的元素不等于 0，满足条件的有多个，可任选其一，记录其行数 m_1 和列数 n_1，令 $S_{m_1n_1}=A_{m_1n_1}$，并令 $B_{m_1n_1}=0$，以及 $\boldsymbol{B}$ 的 m_1 列全部等于 0，$\boldsymbol{B}$ 的 n_1 行全部等于 0；如果满足条件的元素都等于 0，则转到步骤 4.

步骤 4 结束并输出结构二次幂零矩阵 $\boldsymbol{S}$.

4 算法的 Matlab 程序

程序 1：产生随机矩阵 $\boldsymbol{A}$ 并查找距离对角线最近优先的结构二次幂零矩阵 $\boldsymbol{A}$ 的 Matlab 程序[8]如下：

```
clear
clc
close all
n=16;
A=randn(n);
S=zeros(n);
B=abs(A-diag(diag(A)));
maxabs=max(max(B));
while maxabs>10^(-6)
    [m1,n1]=find(B>0);
    k=find(abs(m1-n1)==min(abs(m1-n1)));
    k1=k(1);
    S(m1(k1),n1(k1))=B(m1(k1),n1(k1));
    B(m1(k1),n1(k1))=0;
    B(n1(k1),:)=0;
    B(:,m1(k1))=0;
    maxabs=max(max(B));
end
S
```

程序 2：产生随机矩阵 $\boldsymbol{A}$ 并查找绝对值最大者优先的结构二次幂零矩阵 $\boldsymbol{A}$ 的 Matlab 程序如下：

```
clear
clc
close all
n=16;
A=randn(n);
S=zeros(n);
B=abs(A-diag(diag(A)));
maxabs=max(max(B));
while maxabs>10^(-6)
```

```
    [m1,n1]=find(B==maxabs);
    S(m1(1),n1(1))=A(m1(1),n1(1));
    B(m1(1),n1(1))=0;
    B(n1(1),:)=0;
    B(:,m1(1))=0;
    maxabs=max(max(B));
end
S
```

5 结论

同一个矩阵对应的结构二次幂零矩阵非常多，而选取不同的结构二次幂零矩阵，其迭代的收敛速度可能不一样，一般来说，包含非零元素的个数越多，收敛速度可能越快，因此需要根据实际情况来选择适当的结构二次幂零矩阵。对于结构有简单规律的矩阵，根据其结构特点，应尽量选择含非零元素个数比较多的结构二次幂零矩阵；对于结构比较复杂的矩阵，难以挑选合适的结构二次幂零矩阵，可根据本文算法采用计算机编程查找结构二次幂零矩阵。

[参考文献] (References)

[1] 张承平. 一类并行求解线性方程组的迭代方法[EB/OL]. 北京：中国科技论文在线 [2015-10-13]. http://www.paper.edu.cn/releasepaper/content/201510-66.
ZHANG C P. A class of parallel iterative methods to solve the linear equations[EB/OL]. Beijing: Sciencepaper Online[2015-10-13]. http://www.paper.edu.cn/releasepaper/content/201510-66. (in Chinese)

[2] 张承平. 关于一类并行求解线性方程组的迭代方法的补充[EB/OL]. 北京：中国科技论文在线[2015-10-26]. http://www.paper.edu.cn/releasepaper/content/201510-214.
ZHANG C P. The supplement of a class of parallel iterative methods to solve the linear equations[EB/OL]. Beijing: Sciencepaper Online[2015-10-26]. http://www.paper.edu.cn/releasepaper/content/201510-214. (in Chinese)

[3] 张承平. 基于结构二次幂零矩阵的迭代法构造[J]. 中国科技论文在线精品论文，2017，10（19）：2145-2150.
ZHANG C P. Structuring iterative methods on structural 2-nilpotent matrices[J]. Highlights of Sciencepaper Online, 2017, 10(19): 2145-2150. (in Chinese)

[4] 张承平. Laplace 方程五点差分格式的结构二次幂零矩阵迭代法[J]. 中国科技论文在线精品论文，2018，11（1）：11-16.
ZHANG C P. Structural 2-nilpotent matrix iterative method about the five points difference scheme of the Laplace equation[J]. Highlights of Sciencepaper Online, 2018, 11(1): 11-16. (in Chinese)

[5] 贺雨晴，张楠，李云东. 行列块不同划分机制下矩阵向量相乘的并行计算方法[J]. 电脑知识与技术，2015，11（20）：164-167.
HE Y Q, ZHANG N, LI Y D. Parallel computing method of matrix-vector multiplication with different partition for line, column and block[J]. Computer Knowledge and Technology, 2015, 11(20): 164-167. (in Chinese).

[6] 薛东川，朱振宇，杨俊，等. 用 MATLAB 实现波动方程多核并行计算[J]. 地球物理学进展，2015，30（6）：2841-2845.
XUE D C, ZHU Z Y, YANG J, et al. MATLAB solve wave equation on multi-core computer[J]. Progress in Geophysics, 2015, 30(6): 2841-2845. (in Chinese).

[7] 张承平. 结构二次幂零矩阵迭代法的并行计算[J]. 中国科技论文在线精品论文，2018，11（7）：698-704.
ZHANG C P. Parallel computation of iterative methods on structural 2-nilpotent matrices[J]. Highlights of Sciencepaper Online, 2018, 11(7): 698-704. (in Chinese)

[8] 张承平，马新文，林越，等. 数学实验与建模[M]. 昆明：云南科技出版社，2013.
ZHANG C P, MA X W, LIN Y, et al. Mathematical experiment and mathematical modeling[M]. Kunming: Yunnan Science and Technology Press, 2013. (in Chinese)