中国研究生创新实践系列大赛
“华为杯”第十八届中国研究生
数学建模竞赛

学　校		南京工业大学
参赛队号		21102910019
队员姓名	1.	庄清来
	2.	陆寅
	3.	许丽凤

中国研究生创新实践系列大赛

“华为杯”第十八届中国研究生
数学建模竞赛

题 目　　相关矩阵组低复杂度计算和存储模型

摘　　要：

本文设计相关模型以对矩阵在某些维度上存在的关联性进行了研究，分析了各数据矩阵组之间的关联性，建立模型降低计算复杂度并节省存储开销。

矩阵 W 的近似分析模型（问题一）本问题的目标是计算关联矩阵大的奇异值对应的右奇异向量拼接矩阵的广义逆。该计算有两部分构成：一个 8*8 矩阵的逆矩阵和 384 个分块矩阵的右奇异向量。考虑到逆矩阵计算的矩阵规模不大，具有对称性，且计算一次，我们直接采用 Cholesky 分解法。我们工作的重点放在分块矩阵右奇异向量的计算上。右奇异向量的计算一般采用 Golub-Kahan[1]的基于双对角化和 QR[2]分解（这里称作 GKSVD）。如果每一个分块矩阵的右奇异向量都用 GKSVD 计算，计算复杂度高。为了减少计算量，需要利用关联矩阵的特性降低右奇异向量的计算复杂度。通过分析我们发现矩阵 $H_{(j,\ k)}$两个最大右奇异向量的计算可以转化为矩阵 $H^*_{(j,\ k)}H_{(j,\ k)}$的两个最大特征值对应的特征向量。而乘幂法(Power Method)和降阶法（Deflation techniques）是用于计算上述特征向量的迭代算法，在迭代初始向量合理选择的前提下，收敛速度很快。由于我们的问题中分块矩阵之间有很强的关联性，矩阵 $H_{(j,\ k)}$的两个右奇异向量可以作为 $H_{(j,\ k+1)}$右奇异向量迭代的初始向量。对题目的数据进行计算发现，绝大多数情况下，上述迭代只需要 2~3 次， 但少部分数据要想达到精度要求需要更多次迭代。为此，我们在乘幂法和降阶法中增加了精度约束[3]。整个程序运行结果表明，我们设计的算法对于题目给定的相关矩阵组具有低复杂度的特性。

压缩及解压缩模型（问题二）为了降低存储复杂度，我们首先对原始数据矩阵组 H 进行压缩。考虑到题目涉及的问题背景：视频信号中的单帧图像可视为一个矩阵，连续的多帧图像则反映在矩阵间的相关性上。我们建立了关联矩阵的线性变化模型。但考虑到存储性，线性模型中采用了稀疏矩阵表示。上述存储模型同样可用于矩阵 W 的存储建模。不过，由于精确要求为-30dB，上述方案难以满足。考虑到题目数据中行数据具有非常高的行相关性，我们用大矩阵的每一行构成一个 384*64 的矩阵。针对该矩阵利用奇异值分解，保留贡献率大于 99%的主成份，实验表明在满足精度的情况下，压缩比可以达到 65%。

相关矩阵组的整体处理（问题三）我们直接使用压缩后的矩阵 $\tilde{H}$代替 H，再用乘幂法和降阶法计算右奇异向量。这样在降低存储复杂度的同时，可以降低计算复杂度。

关键词：乘幂法；关联矩阵；计算复杂度；降阶法

目　录

一、问题背景和问题重述

1.1 问题背景

相关矩阵是指矩阵通常在某些维度上存在关联性。关联性在我们的生活中随处可见，在一些电影中也会利用这一特性，比如王家卫导演在他作品中经常使用的抽帧手法。而在计算机视觉、相控阵雷达、声呐、射电天文、无线通信等领域中的信号通常以矩阵形式表现。例如，视频信号中的单帧图像可视为一个矩阵，连续的多帧图像组成了相关矩阵组，而相邻图像帧或图像帧内像素间的关联性则反映在矩阵间的相关性上。本文对矩阵在某些维度上存在的关联性进行了研究，分析了各数据矩阵组之间的关联性，降低计算复杂度并节省存储开销。

1.2 问题重述

基于上述研究背景，题目提供了6组矩阵数据，每组矩阵数据分别给定三组复数矩阵 $H=\left\{H_{j,\ k}\right\}$，$V=\left\{V_{j,\ k}\right\}$，$W=\left\{W_{j,\ k}\right\}$，本文需要解决以下问题：

问题一：矩阵W的近似分析模型的构建

基于给定的所有矩阵数据H，分析其数据间的关联性，计算关联矩阵大的奇异值对应的右奇异向量拼接矩阵的广义逆，建立矩阵W的近似分析模型。在满足最低建模精度的情况下，使得总计算复杂度最低。

问题二：压缩及解压缩模型的构建

基于给定的所有矩阵数据H和W，分析各自数据间的关联性，分别设计相应的压缩$P_1(\cdot)$、$P_2(\cdot)$和解压缩$G_1(\cdot)$、$G_2(\cdot)$模型。在满足最低误差的情况下，使得存储复杂度和压缩与解压缩的计算复杂度最低。

问题三：相关矩阵组的整体处理

基于给定的所有矩阵数据H，分析其数据间的关联性，在满足最低建模精度的情况下，设计低复杂度计算和存储的整体方案，完成从矩阵输入信号H到近似矩阵输出信号$\widehat{W}$的端到端流程。

二、模型假设和基本符号说明

2.1 模型假设

假设一：在第一问中假设模型精度的优先级略高于降低计算复杂度的优先级。

假设二：当前 L 个奇异值远大于其它奇异值时，收敛很快。

2.2 基本符号说明

符号	符号含义
S	稀疏矩阵
GKSVD	Golub-Kohn 奇异值分解
$D = diag(d_1，\ d_2，\ \cdots，\ d_M)$	对角矩阵
J	双对角矩阵
$\hat{V}$	矩阵 V 的近似分析模型
$\hat{V}1$	最大特征值向量构成的近似分析模型
$\hat{V}2$	次大特征值向量构成的近似分析模型
$H = \{H_{j，\ k}\}$	j 行 k 列给定矩阵 H
V=$\{V_{j，\ k}\}$	由$H_{j，\ k}$的最大、次大右奇异向量构成的矩阵
$X = \{X_{j，\ m}\}$	j 行 m 列矩阵 X
$\tilde{U}j，\ m$	H 矩阵关联性中的 j 行 m 列近似矩阵 U
$\tilde{S}j，\ m$	H 矩阵关联性中的 j 行 m 列近似矩阵 S
$\tilde{V}j，\ m$	H 矩阵关联性中的 j 行 m 列近似矩阵 V
$U_{j，\ m}$	H 矩阵关联性中的 j 行 m 列矩阵 U
$S_{j，\ m}$	H 矩阵关联性中的 j 行 m 列矩阵 S
$V_{j，\ m}$	H 矩阵关联性中的 j 行 m 列矩阵 V
$\tilde{U}j,nw$	W 矩阵关联性中的 j 行 nw 列近似矩阵 U
$\tilde{S}j,nw$	W 矩阵关联性中的 j 行 nw 列近似矩阵 S
$\tilde{V}j,nw$	W 矩阵关联性中的 j 行 nw 列近似矩阵 V
$U_{j,nw}$	W 矩阵关联性中的 j 行 nw 列矩阵 U
$S_{j,nw}$	W 矩阵关联性中的 j 行 nw 列矩阵 S
$V_{j,nw}$	W 矩阵关联性中的 j 行 nw 列矩阵 V
$\widehat{W}$	矩阵 W 的近似分析模型
m	矩阵的行数
n	矩阵的列数
K0	迭代次数
K_0	$\tilde{U}j,w$的列数
K_{0w}	$\tilde{U}j,nw$的列数
lamb1	最大特征值
V01	最大特征向量
lanbd	次最大特征值
V02	次最大特征向量

$V_{d(k-1)}$	优化乘幂法计算得到的第 K-1 次迭代结果
V_{dk}	优化乘幂法计算得到的第 K 次迭代结果
V_k	拼接之后的矩阵
Ves	矩阵 V 的近似分析模型
WWD	矩阵 W 的近似分析模型
γ	相关性分析得出的相关性系数
ROW1	最大特征向量的模型估计精度
ROW2	次大特征向量的模型估计精度
N_s	稀疏矩阵中非零元素的个数
ρ1	最大特征向量的模型估计精度系数
ρ2	次大特征向量的模型估计精度系数

三、问题一的建模与求解

3.1 任务一问题分析

基于给定的所有相关矩阵H，分析其数据间的关联性，计算关联矩阵大的奇异值对应的右奇异向量拼接矩阵的广义逆，建立矩阵W的近似分析模型。在满足最低建模精度的情况下，使得总计算复杂度最低。

首先利用 MATLAB 自带的相关性分析函数分析了矩阵数据 H 间的关联性。再根据问题一中的两个小问题降低计算复杂性的要求，可以通过直接计算并且降低 SVD 的计算复杂性来达到目的。利用 Golub-Kahan 双对角化或 QR 分解的方法求 SVD 分解(图 3.1)，考虑到 QR 分解的计算量过大故舍去，最终使用 Golub-Kahan 双对角化加上乘幂法计算最大特征值及其对应的特征向量，降阶法计算次大特征值及其对应的次大特征向量(图 3.2)。再根据三对角矩阵特征向量与原矩阵奇异向量之间的关系求右奇异向量，在此过程中为了降低计算复杂度且矩阵具有对称性，利用了 Rayleigh 熵加速特征值收敛效率。另外，上述算法中主要涉及乘法运算所以在计算复杂性讨论时，我们主要考虑乘法。

求出第一问中的近似$\hat{V}$之后，根据题目中所给的公式，只需要再求出逆矩阵$(\mathrm{V}_k^H\mathrm{V}_k + \sigma^2\mathrm{I})^{-1}$，就能得出第二问中的近似$\widehat{W}$。在求该逆矩阵过程中，采用了 Gauss-Jordan 法与 Cholesky 分解法。

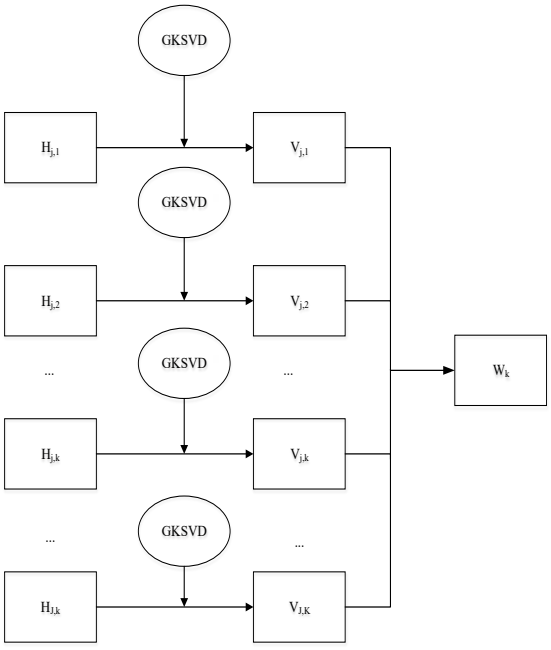


图 3.1 经典奇异值分解

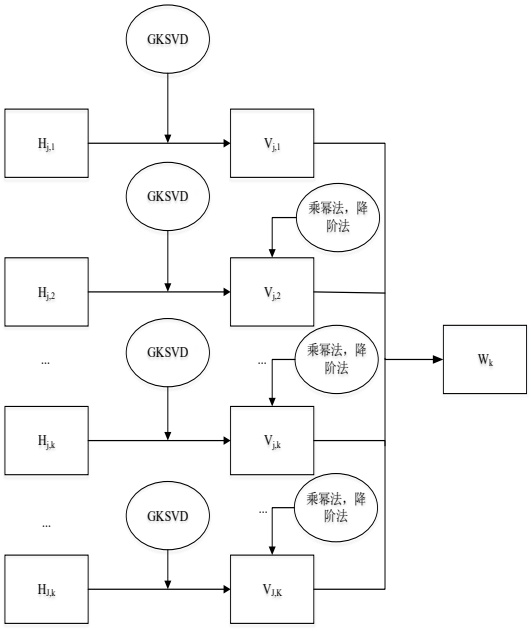


图 3.2 低复杂度算法

首先对给定的所有矩阵数据H，分析其数据间行向量的关联性。选择使用 MATLAB 中自带的相关性分析函数，该函数是用于返回输入矩阵中各列之间的两两线性相关系数矩阵。

由于矩阵 H 过于庞大，在 MATLAB 软件中打开时，将矩阵 H 分成了 384 块，每块的维度为 4 行*64 列，所以在分析其每行的相关性时数据是比较多的，但文章篇幅有限，在下表中只展示每行中相关性最低的数字，通过计算发现每个矩阵 H 间行向量的关联性都很高，如表 3.1 所示为六组数据每 j 行 $H_{j,k}$ 与 $H_{j,k+1}$ 相同行最小相关系数，如 DATA3 中的 $H=\{H_{3,\,k}\}$ 中最低的相关性系数都能达到 0.966389。

表 3.1 行向量相关性分析

DATA/J	1	2	3	4
DATA1	0.987073	0.991296	0.988607	0.979909
DATA2	0.984152	0.986603	0.983362	0.971361
DATA3	0.971158	0.976891	0.966389	0.943479
DATA4	0.993982	0.994046	0.992338	0.994492
DATA5	0.991965	0.991456	0.990765	0.988907
DATA6	0.990898	0.985689	0.99275	0.986134

显然，计算 $V_k=[V_{1,k}，……V_{j,k}，……V_{J,k}]$是需要大计算量的，因此我们不能计算每一个 $V_{j,k}$。而需要考虑拼接矩阵 V_k 中各矩阵 $V_{j,k}$ 之间的关联性。根据定义知 $V_{j,k}$ 是已知矩阵 $H_{j,k}$ 的奇异值分解的右奇异向量构成的矩阵。因为 $H_{j,k}$ 矩阵间是关联的，所以对应的 $V_{j,k}$ 一定也有关联性。

通过 MATLAB 代码分析数据相邻矩阵最大奇异值对应奇异向量之间以及次大奇异值和对应的奇异向量之间的相关性，可以清晰的看出关联性很高。其相关性可见图 3.3 和图 3.4。不过，在后续计算过程中我们发现有一些特殊位置关联性并不高，如图 3.5 这给我们的建模带来了不小的困难。

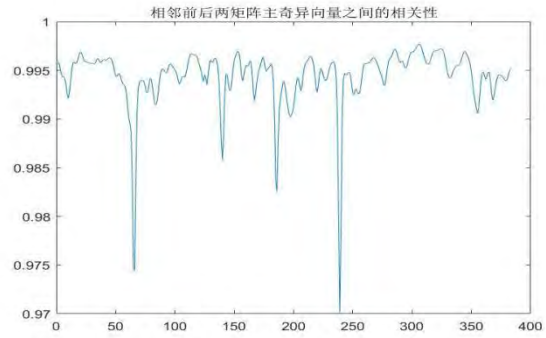


图 3.3 数据相邻矩阵最大奇异值对应奇异向量之间相关性

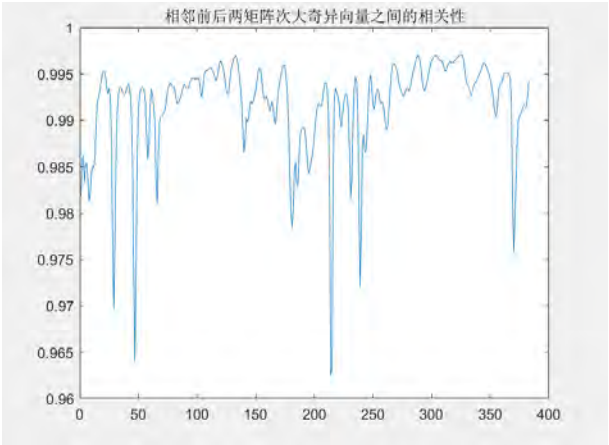


图 3.4 数据相邻矩阵次大奇异值对应奇异向量之间相关性

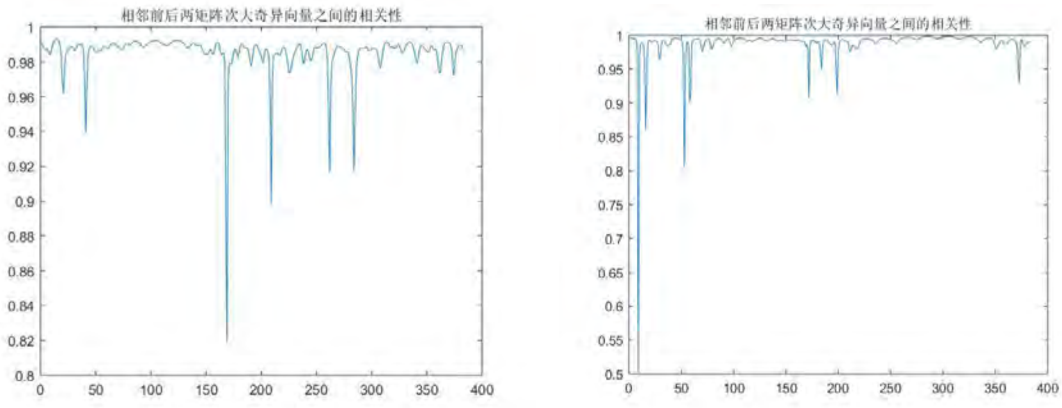


图 3.5 关联性不强的坏值位置

3.2 问题一方法

3.2.1 Golub-Kahan 双对角化与 QR 分解

利用 Golub-Kahan 双对角化的分解 SVD 的过程是假设 A 是一个 m*n 复矩阵，那么 $A = PJQ^*$，这里 P、Q 是酉矩阵，J 是 m*n 双对角矩阵。

$$J = \begin{bmatrix} \alpha_1 & \beta_1 & 0 & \cdots & \cdots & & & \\ & \alpha_2 & \beta_2 & 0 & \cdots & & & \\ & & \ddots & \ddots & & \cdots & & 0 \\ & & & & & & \ddots & \alpha_{n-1} & \beta_{n-1} \\ & & & & & & & \vdots & \alpha_n \\ & & & & & & & & \vdots \end{bmatrix}$$

按照文献[1]的符号，记$A = A^{(1)}$，定义$A^{(\frac{3}{2})}$，$A^{(2)}$，…，$A^{(n)}$，$A^{(n+\frac{1}{2})}$如下：

$$A^{(k+\frac{1}{2})} = P^{(k)}A^{(k)}，k = 1，2，\cdots，n，\ \ldots\ldots\ldots\ldots\ldots\ldots(1)$$

$$A^{(k+1)} = A^{(k+\frac{1}{2})}Q^{(k)}，k = 1，2，\cdots，n-1\ \ldots\ldots\ldots\ldots\ldots\ldots(2)$$

其中$P^{(k)} = \mathrm{I} - 2\vec{x}^{(k)} \cdot \vec{x}^{(k)*}$，将其代入式(1)，得到式(3)

$$A^{(k+\frac{1}{2})} = A^{(k)} - \vec{x}^{(k)} \cdot 2\left(\vec{x}^{(k)*}A^{(k)}\right)\ldots\ldots\ldots\ldots\ldots\ldots (3)$$

这里令

$$s_k = \left(\sum_{i=1}^{m}\left|a_{i,k}^{(k)}\right|^2\right)^{1/2}\ldots\ldots\ldots\ldots\ldots\ldots(4)$$

$$x_i^{(k)} = 0，for\ i < k\ldots\ldots\ldots\ldots\ldots\ldots(5)$$

$$x_k^{(k)} = \left[\frac{1}{2}\left(1 + \frac{\left|a_{k,k}^{(k)}\right|}{s_k}\right)\right]^{1/2}\ldots\ldots\ldots\ldots\ldots\ldots(6)$$

$$c_k = \left(2s_k \frac{a_{k,\ k}^{(k)}}{\left| a_{k,\ k}^{(k)} \right|} x_k^{(k)} \right)^{-1} \quad \text{.....(7)}$$

$$x_i^{(k)} = c_k a_{i,\ k}^{(k)},\ \ for\ i > k \text{..... (8)}$$

$$A^{(k+1)} = A^{(k+\frac{1}{2})} - 2\left(A^{(k+\frac{1}{2})} \vec{y}^{(k)} \right) \cdot \vec{y}^{(k)*} \text{.....(9)}$$

这里

$$t_k = \left(\sum_{j=k+1}^{n} \left| a_{k,\ j}^{(k+\frac{1}{2})} \right|^2 \right)^{1/2} \quad \text{.....(10)}$$

$$y_j^{(k)} = 0,\ \ for\ j \le k, \text{.....(11)}$$

$$y_{k+1}^{(k)} = \left[\frac{1}{2} \left(1 + \frac{\left| a_{k,\ k+1}^{(k+\frac{1}{2})} \right|}{t_k} \right) \right]^{1/2} \text{.....(12)}$$

$$d_k = \left(2t_k \frac{a_{k,\ k+1}^{(k+\frac{1}{2})}}{\left| a_{k,\ k+1}^{(k+\frac{1}{2})} \right|} y_{k+1}^{(k)} \right)^{-1} \quad \text{.....(13)}$$

$$y_j^{(k)} = d_k \bar{a}_{k,\ j},\ \ for\ j > k+1 \text{.....(14)}$$

根据计算，上述算法对$k = 1, 2, \cdots, n$.对于每一个给定的k需要的乘除法次数大约：6nm。故总的乘除法次数为 $6n^2m$。显然，A 与 J 具有相同的奇异值，这些奇异值是 J*J 特征值的正平方根。不妨假设对应的奇异值为$\sigma_1 \ge \sigma_2 \ge \cdots \ge \sigma_n \ge 0$。

$$\Sigma = \begin{bmatrix} \sigma_1 & & & & \\ & \sigma_2 & & & \\ & & \ddots & & \\ & & & \cdots \quad \cdots & \sigma_n \\ & & & & \end{bmatrix}$$

令

$$J = X\Sigma Y^* \text{.....(15)}$$

这里X和Y是酉矩阵。故

$$A = U\Sigma V^* \text{.....(16)}$$

其中$U = PX$，$V = QY$

$$B \triangleq J^* J \text{.....(17)}$$

利用QR分解SVD的化解过程是将矩阵B分解为B=QR的形式，Q为正交矩阵，R为非奇异上三角矩阵，其中：

（1） 当R的对角元全为正数时，分解是唯一的；

（2） 分解过程实际上矩阵约化的过程，用正交变换QT使得QTA=R；

（3） 若分解结果中R的对角元不全为正数，取对角阵D[2]：

$$D=\begin{bmatrix} \frac{\lambda_0}{\|\lambda_0\|} & & \\ & \frac{\lambda_1}{\|\lambda_1\|} & \\ & & \frac{\lambda_{n-1}}{\|\lambda_{n-1}\|} \end{bmatrix}$$

此时 $B = QR$……(18)

记 B_1=B，对 B_1 作 QR 分解

$$B_1 = Q_1 R_1 \text{……(19)}$$

令

$$B_2 = R_1 Q_1 \text{……(20)}$$

然后 B_2 作 QR 分解

$$B_2 = Q_2 R_2 \text{……(21)}$$

再令

$$B_3 = R_2 Q_2 \text{……(22)}$$

一般地，设已得到 B_m，则对 B_m 作 QR 分解，即

$$B_m = Q_m R_m \text{……(23)}$$

再令

$$B_{m+1} = R_m Q_m \text{……(24)}$$

这样，可得到一个矩阵序列{B_m}。于是

$$B_{m+1} = Q_m^T B_m Q_m = \cdots = Q_m^T Q_{m-1}^T \cdots Q_1^T B_1 Q_1 \cdots Q_{m-1} Q_m \text{……(25)}$$

可以证明，在一定的条件下，B_m 的主对角线以下的元素当$m \to \infty$时，都趋于零。因此，当 m 足够大时，可将 B_m 主对角元作为矩阵 B 的特征值近似。

但考虑到我们的问题仅计算两个奇异值，且相较于 QR 化解，双对角矩阵很稀疏，计算量较小。所以选择对矩阵进行双对角化，再利用乘幂法计算最大特征值和其对应的特征向量。

3.2.2 乘幂法与降阶法

许多实际应用中，往往不需要计算矩阵A的全部特征值，而只要计算模数最大的特征值(称为主特征值)。乘幂法是计算矩阵的模数最大的特征值及其相应的特征向量的一种迭代法。设选定初始向量，这里$\vec{x}_1$为主特征值对应的特征向量。

$$\vec{v}_0 = a_1\vec{x}_1 + a_2\vec{x}_2 + \cdots + a_n\vec{x}_n \text{……(26)}$$

令

$$\vec{v}_k = A\vec{v}_{k-1}，\ k = 1，2，\cdots \text{……(27)}$$

通常的乘幂法迭代的初始向量是任意选取的。考虑到相邻矩阵的关联性，主特征值对应的特征向量具有高度的相关性。所以在已经得到前一个矩阵的右奇异向量后，在计算下一个关联的矩阵右奇异向量时，可以以前面的向量作为初始向量。

当初始向量$\vec{v}_0$选取接近于主特征值对应的特征向量，即系数满足

$$|a_1| \gg |a_i|，\ i = 2，3，\cdots，n \text{ ……(28)}$$

$$\vec{v}_k = A\vec{v}_{k-1} = a_1 A^k \vec{x}_1 + \sum_{i=2}^{n} a_i A^k \vec{x}_i = a_1 \lambda_1^k \vec{x}_1 + \sum_{i=2}^{n} a_i \lambda_i^k \vec{x}_i \text{…. (29)}$$

最大特征值与对应特征向量有如下摄动理论：

引理1 假设矩阵A的主特征值为λ_{max}，如果矩阵有摄动$A+O(\varepsilon)$，这里ε表示小量，则$A+O(\varepsilon)$的主特征值为$\lambda_{max}+O(\varepsilon)$。

证明：根据定义知，

$$\lambda_{max}=\max\frac{\vec{x}^T A\vec{x}}{\vec{x}^T\vec{x}}\text{……..……..…..(30)}$$

不难发现，$\max\frac{\vec{x}^T(A+O(\varepsilon))\vec{x}}{\vec{x}^T\vec{x}}=\lambda_{max}+O(\varepsilon)$…….……...…..…(31)

引理2. 假设矩阵A的主特征值对应的特征向量为$\vec{x}_{max}$，如果矩阵有摄动$A+O(\varepsilon)$，这里ε表示小量，则$A+O(\varepsilon)$主特征值对应的特征向量为$\vec{x}_{max}+O(\varepsilon)$.

为保证数值稳定，我们将乘幂法改写为：

$$\begin{cases}u_k=Av_{k-1}\\ v_k=u_k/m_k\end{cases},\ k=1,\ 2,\ \cdots\text{……...……(32)}$$

其中$m_k=\max(u_k)$表示u_k中绝对值最大的头一个分量。

在一般的乘幂法中迭代的初始向量通常是任意选取的，所以迭代的次数比较多。这里考虑到问题的特殊性，既然已知所需计算的特征向量的一个很好的近似，通过乘幂法可以提高逼近的程度。假设乘幂法迭代的次数为 K_0，这里$A=H_{j,\ k}^H H_{j,\ k}$，那么乘除法次数约为 $2K_0MN$。事实，我们在计算 Hj，k+1 的右奇异向量时，可以用 Hj，k 的右奇异向量的初始向量。

然而，乘幂法只能计算主特征值与对应的特征向量，这里还需要计算次大特征值所对应的特征向量，为此我们采用降阶算法(deflation techniques)。

降阶法是指假设 A 的模数最大特征值λ_1以及相应的特征向量$\vec{x}_1$已经得到。为方便记忆，记$A_1=A$。

$$A_2=S_1A_1S_1^{-1}=\begin{bmatrix}\lambda_1 & \vec{\omega}^T\\ 0 & B_2\end{bmatrix}\text{……..….(33)}$$

其中$\vec{\omega}$为 n-1 维向量，B_2是一个$(n-1)*(n-1)$阶矩阵。在记$\vec{x}_1=[x_1\quad x_2\quad\cdots\quad x_n]^T$，$x_1\neq 0$的假设下，$S_1$的选择如下

$$S_1=\begin{bmatrix}1 & 0 & \cdots & 0\\ -x_2/x_1 & 1 & \cdots & 0\\ \vdots & \ddots & \ddots & \vdots\\ -x_n/x_1 & 0 & \cdots & 1\end{bmatrix}\text{……..…… (34)}$$

考虑到数值稳定性，可采用选主元的方法，即把向量$\vec{x}_1$的模数最大的分量(设为x_p)与第一个分量x_1交换，这相当于用排列阵$I_{1,\ p}$左乘$\vec{x}_1$，即令

$$\vec{y}=[y_1\quad y_2\quad\cdots\quad y_n]^T=I_{1,\ p}\vec{x}_1\text{……..(35)}$$

这时

$$S_1=\begin{bmatrix}1 & 0 & \cdots & 0\\ -y_2/y_1 & 1 & \cdots & 0\\ \vdots & \ddots & \ddots & \vdots\\ -y_n/y_1 & 0 & \cdots & 1\end{bmatrix}\text{……..…….. (36)}$$

相应的，有

$$A_2 = S_1\left(I_{1,\ p}A_1I_{1,\ p}\right)S_1^{-1}\text{……..……(37)}$$

其中

$$S_1^{-1} = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ y_2/y_1 & 1 & \cdots & 0 \\ \vdots & \ddots & \ddots & \vdots \\ y_n/y_1 & 0 & \cdots & 1 \end{bmatrix}\text{……..……(38)}$$

为了计算矩阵A的次大特征值λ_2对应的特征向量$\vec{x}_2$，设$\vec{z}_2$是A_2与λ_2相应的特征向量，则

$$\begin{bmatrix} \lambda_1 & \vec{\omega}^T \\ 0 & B_2 \end{bmatrix}\vec{z}_2 = \lambda_2\vec{z}_2\text{…….……(39)}$$

另外，假设

$$B_2\vec{y}_2 = \lambda_2\vec{y}_2\text{……..……(40)}$$

显然，可以对上述矩阵B_2采用乘幂法计算其主特征值及其对应的特征向量。

令

$$\vec{z}_2 = \begin{bmatrix} a \\ \vec{y}_2 \end{bmatrix}\text{…….……..……..(41)}$$

于是

$$(\lambda_1 - \lambda_2)a + \vec{\omega}^T\vec{y}_2 = 0\text{……..……(42)}$$

即

$$a = -\frac{\vec{\omega}^T\vec{y}_2}{\lambda_1 - \lambda_2}\text{…….……..…(43)}$$

考虑到乘幂法是迭代算法，需要选择合适的初始向量。因为

$$S_1I_{1,\ p}A_1I_{1,\ p}S_1^{-1}\vec{z}_2 = \lambda_2\vec{z}_2\text{……..…….(44)}$$

因此，

$$\vec{x}_2 = I_{1,\ p}S_1^{-1}\vec{z}_2\text{…….……..…….(45)}$$

对于我们的问题，因为已经有了近似向量(上一个矩阵的右奇异向量)，不妨设其为$\tilde{x}_2$，故

$$\tilde{z}_2 = S_1I_{1,\ p}\tilde{x}_2\text{……..……(46)}$$

得到$\tilde{z}_2$后，利用式(41)，可以构造$\vec{y}_2$的近似$\tilde{y}_2$。

同样的，在计算$\vec{y}_2$时，我们采用乘幂法，而且乘幂法的初始向量可以由 $H_{j,\ k}$ 的第二个右奇异向量得到。

我们先固定乘幂法的迭代次数为3，经过3次迭代运行之后，可以得到所有的最大右奇异向量组$\hat{v}_1$，针对**DATA2**要计算得出了由最大特征向量构成的近似模型$\hat{v}_1$总共需要迭代4608次。利用公式(47)计算$\hat{V}$中两个向量的建模精度。

$$\rho_{l,\ j,\ k}(\boldsymbol{v}) = \frac{\left\|\hat{\boldsymbol{v}}_{l,\ j,\ k}^H\boldsymbol{v}_{l,\ j,\ k}\right\|_2}{\left\|\hat{\boldsymbol{v}}_{l,\ j,\ k}\right\|_2\left\|\boldsymbol{v}_{l,\ j,\ k}\right\|_2},\ l = 1,\ \ldots,\ L\text{…….…….. (47)}$$

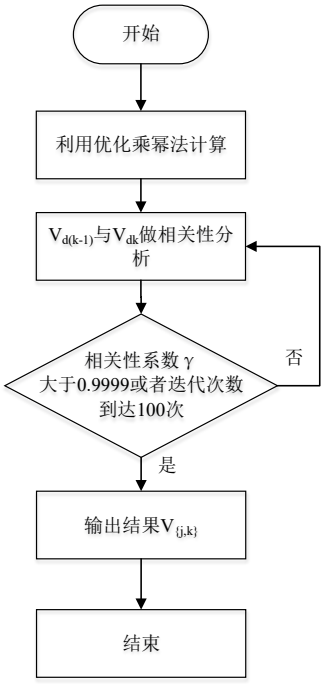


图 3.6 优化乘幂法流程

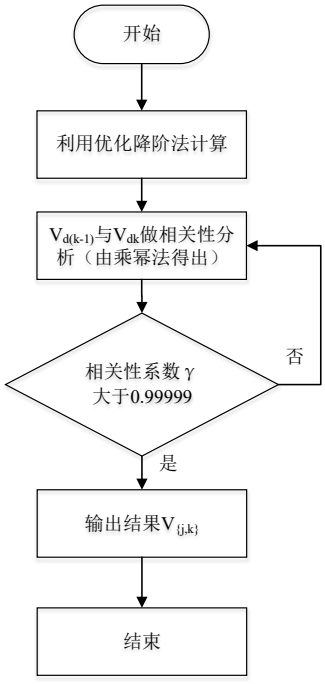


图 3.7 优化降阶法流程

通过计算发现此时的最大特征向量大部分都满足精度要求，但是在某些点处的精度达不到 0.99，甚至远远低于 0.99，为了提高精度，满足芯片使用要求，我们考虑提高迭代次数。但如果不加判断地增加乘幂法次数会提高计算复杂度，这里我们注意到乘幂法计算特征向量的收敛性可用相邻两次求得的特征向量的相关系数来表示。然而特征向量不唯一，为此我们通过对乘幂法的优化，利用乘幂法计算最大特征向量时，会将在第 K 次迭代完之后求出的结果与第 K-1 次的结果进行相关性计算，如果相关性系数大于 0.9999 或者迭代次数到达 100 次就终止迭代，并直接输出。通过优化我们将最大特征向量构成的$\hat{V}1$的精度 ROW1 均提高至 0.99 以上，满足题意。具体流程如图 3.6 所示。

同样的我们也对利用降阶法所求得的次大特征向量进行精度分析，通过分析发现次大特征向量的精度不满足题意，在第 3 行，278 列处的最低精度甚至只有 0.0123 远远低于 0.99。为了提高精度我们优化了降阶法，将乘幂法终值准则相关性系数阈值提高至 0.99999。经过优化最终输出的精度都得到了大幅度的提高，在最低处的精度提高至 0.9972，符合题意。具体流程如图 3.7 所示。

经过优化乘幂法和优化降阶法分别对 ROW1 和 ROW2 进行了提高，使其所有的模型估计精度系数 ρ1、ρ2 都达到了 0.99 以上，其中乘幂法的总体迭代次数也只提高了两倍为 8586 次，保证满足精度的条件下，最大程度的降低了计算复杂度。

3.2.3 Gauss-Jordan 法与 Cholesky 分解法

根据题目所给的公式：

$$V_{j,k} = svd\left(H_{j,k}\right), \text{or } H_{j,k} = U_{j,k}S_{j,k}\tilde{V}_{j,k}^{H}, V_{j,k} = \tilde{V}_{j,k}^{H}(:,1:L) \quad (48)$$

$$j = 1, \ \ldots, \ J; \ k = 1, \ \ldots, \ K$$

$$W_k = V_k(V_k^H V_k + \sigma^2 I)^{-1} \quad (49)$$

从公式(49)不难发现，W_k的计算由两部分构成，即计算V_k和求逆矩阵$(V_k^H V_k + \sigma^2 I)^{-1}$。这里我们计算$(V_k^H V_k + \sigma^2 I)^{-1}$。

另外，因为 $V_k=[V_{1,k}, \ldots\ldots V_{j,k}, \ldots\ldots V_{J,k}]$的维度$N * LJ$，所以$V_k^H V_k + \sigma^2 I$的维度为$LJ *$

LJ，这里$L=2$，$J=4$。考虑到$\mathrm{V}_k^H\mathrm{V}_k$的计算复杂性(复数乘法)为$N(LJ)^2$，用Gauss-Jordan法(考虑对称性)求$\mathrm{V}_k^H\mathrm{V}_k+\sigma^2\mathrm{I}$逆矩阵的计算复杂性为$\mathrm{O}((LJ)^3)$。当算法结合了文献中提出的Coppersmith–Winograd 方法时[1]，理论上可以将上述复杂度降低到$O((LJ)^{2.376})$，似乎降低了计算复杂度，但因为本问题中$N=64\gg LJ$。这样做的意义不大。所以权衡利弊，这里我们采用 Gauss-Jordan 法。下面介绍 Gauss-Jordan 法和 Cholesky 法。

假设有 n 个线性方程组

$$A\vec{x}^{(p)}=\vec{b}^{(p)},\ p=1,\ 2,\ \cdots,\ n \quad\cdots\cdots(50)$$

其中$\vec{b}^{(p)}=\begin{bmatrix}0\\ \vdots\\ 0\\ 1\\ 0\\ \vdots\\ 0\end{bmatrix}$

其中A是一个可逆的$n*n$矩阵，考虑到这是对称矩阵，可以利用复数形式的 Cholesky 分解。

求A的逆矩阵等价于求解n个线性方程组(50)。

$$A=LL^H \quad\cdots\cdots(51)$$

需要复数乘除法次数$\frac{n^3}{6}-\frac{n}{6}$，加减法次数大约$\frac{n^3}{6}-\frac{n^2}{4}$。这时，解方程组(50)的问题转化为

$$L\vec{y}^{(p)}=\vec{b}^{(p)} \quad\cdots\cdots(52)$$

$$L^H\vec{x}^{(p)}=\vec{y}^{(p)} \quad\cdots\cdots(53)$$

这两个三角方程组求解大约需要乘除法：$2(n^2/2-n/2)$，加减法次数：$2(n^2/2-n/2)$。

故总的乘除法次数：$\frac{n^3}{6}+n^2-\frac{7n}{6}$ ，总的加减法次数：$\frac{n^3}{6}+\frac{3n^2}{4}-n$。

针对我们的特定问题，$n=LJ=8$，利用上述算法的计算逆矩阵总的乘除法为 140，而采用 Coppersmith–Winograd 方法时，总的乘除法次数大约为：$8^{2.376}$ =139.8756，几乎没有差别，所以这里我们采用基于 Cholesky 分解的求逆矩阵的方法。最终求得似分析模型$\widehat{W}$，并且利用公式(50)，分析了该模型的建模估计精度。

$$\rho_{l,\ j,\ k}(\boldsymbol{W})=\frac{\left\|\widehat{\boldsymbol{W}}_{l,\ j,\ k}^H\boldsymbol{W}_{l,\ j,\ k}\right\|_2}{\left\|\widehat{\boldsymbol{W}}_{l,\ j,\ k}\right\|_2\left\|\boldsymbol{W}_{l,\ j,\ k}\right\|_2},\ l=1,\ \ldots,\ L \quad\cdots\cdots(54)$$

3.3 问题一算法框图及结果总结

我们进行了以下的工作：

(1)我们利用 MATLAB 自带的相关性分析函数对所有矩阵数据 H 的关联性进行了分析，发现同一行相邻 H 的行向量之间的关联性都很高，最低都可以达到 0.96 以上。在文章中由于篇幅有限只展示了每行中相关性最低的数字，完整的表格可见附件中问题一的“H 关联性分析”；

(2)在设计相应的近似分析模型$\widehat{V}$、$\widehat{W}$时，我们根据公式(48)和公式(49)来进行算法设计，并且主要利用了乘幂法、降阶法等数学算法。程序代码可见附件中问题一的“FindVes10.m”。通过 MATLAB 程序最终输出了近似分析模型$\widehat{V}$、$\widehat{W}$的数值。完整的答案

表格可见附件中问题一的“近似分析模型 V 、W 值”；

(3)同时我们还对这两个模型进行了精度分析，算法设计参考公式(47)和(54)。公式中的$\|\cdot\|_2$表示矢量的欧几里得范数(也即 2 范数，对于列矢量a，$\|a\|_2=\sqrt{a^Ha}$)；$W_{l,\ j,\ k}$表示$W_{j,\ k}$的第l列。上式中，$\widehat{W}_{l,\ j,\ k}^{H}W_{l,\ j,\ k}$为复数标量，此处取其欧几里得范数即获取其模值。以 DATA1 的数据为例，分析以该数据生成的近似分析模型$\widehat{V}$、$\widehat{W}$，他们的精度分别为 0.992、0.999、0.994、0.990，满足题目中的$\rho_{\min}(V)\geq\rho_{th}=0.99$，$\rho_{\min}(W)\geq\rho_{th}=0.99$的要求。　　对于题目(1)中最关键的复杂度计算，我们将只利用 Golub 方法进行迭代的计算$\widehat{V}$的计算复杂度和利用乘幂法、降阶法进行迭代计算$\widehat{V}$的计算复杂度进行比较；

(4)乘幂法的迭代次数为 K_0 在绝大多数情况下取 2，极少数情况下取 K_0=3。

图 3.8 展示了基于乘幂法和降阶法的解题过程以及复杂度，逆矩阵表示$(V_k^HV_k+\sigma^2I)^{-1}$，小椭圆形中的公式表示复杂度。

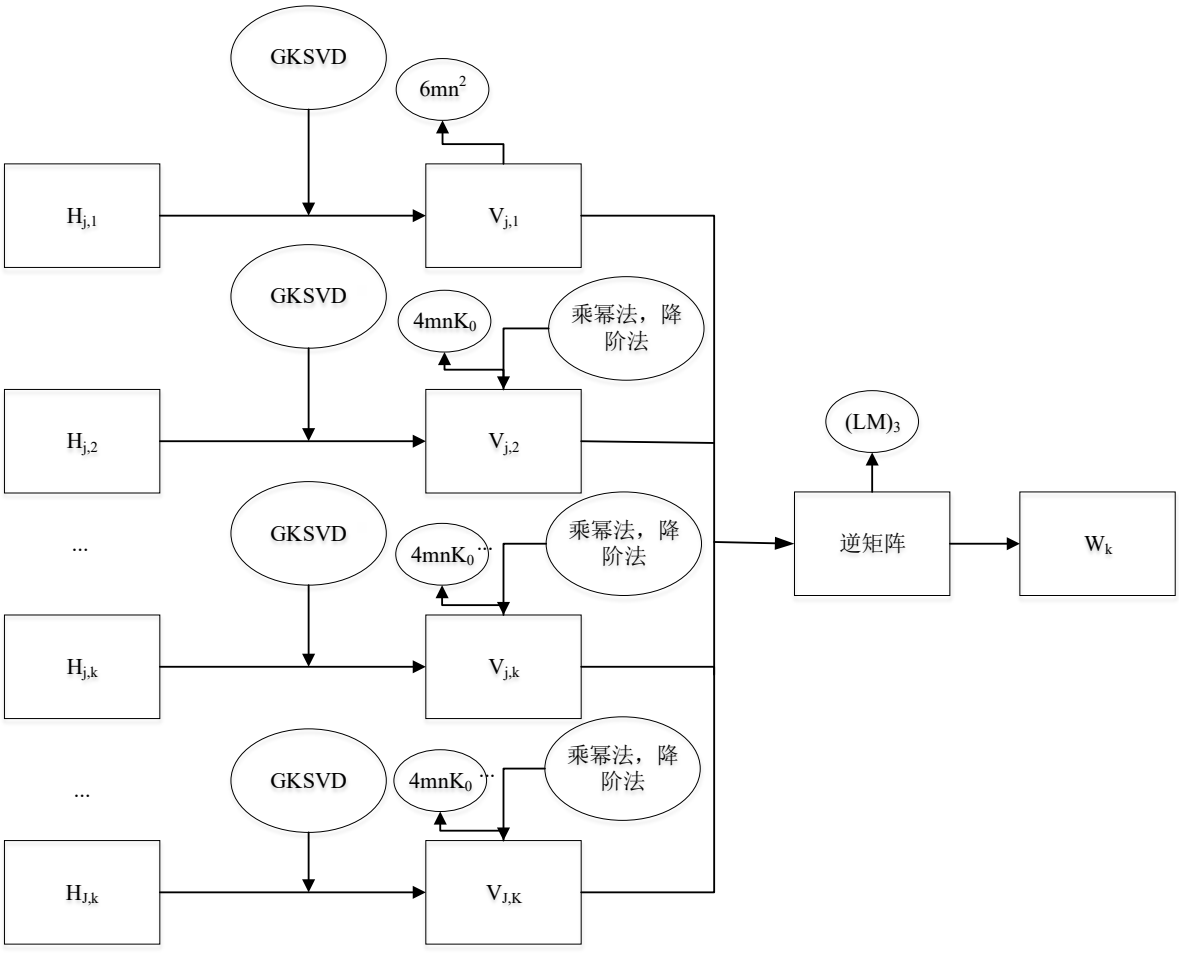


图 3.8 问题一算法框图及结果总结

以 DATA2 为例进行计算复杂度比较，针对 Golub 算法计算$\widehat{V}$，第一步对$k=1，2，\cdots，n$.对于每一个给定的k需要的乘除法次数大约：6nm，故总的乘除法次数为 $6n^2m$。根据表 3.3 提供的计算复杂度进行计算，迭代一次的复杂度为 $6*64^2*6*(25+3)=4.13E+06$。第二步本应使用 QR 分解，由于 QR 分解需要计算全部奇异向量，而乘幂法和降阶法只需要计算最大和其次的奇异向量。显然该选择乘幂法和降阶法。

我们对计算$\hat{V}$降低复杂度方法，第一步是对于乘幂法计算，假设乘幂法的次数为 K_0，那么乘除法次数约为 $2K_0MN$，所以通过乘幂法计算一次迭代的加法复杂度为 7.49E+04*1，乘法复杂度为 1.07E+06*3，迭代次数为 8586 次，所以总值为 2.81E+10。然后第二步再运用了一次除法，除以相关范数进行了相关性分析，此步骤的总复杂度是 2.43E+06。第三步降阶法计算，矩阵 A(维度为 m*n)*B(维度为 n*l)所需的运算次数为加法：(4n-2)ml，乘法：4mnl，按次序加法 1.63E+04 次、乘法 1.64E+04 次、除法 63 次、加法 4.16E+06 次、乘法 4.19E+06 次，所以迭代一次计算复杂度为 1.68E+07，总共的迭代次数为 8922 次，总值为 1.50E+11，将这三步的复杂度相加为 1.78E+11。

根据比较可以发现利用乘幂法进行迭代运算，计算复杂度可以大幅度降低。从表 3.2 的各类算法计算复杂度公式也可以看出乘幂法的计算复杂度公式是最简便的。

表 3.2 各类算法计算复杂度

	计算复杂度
GloubSVD	$O(mn^2)$
乘幂法+降阶法	$O(mn)$
QR	$O(mn^2+mn)$
Cholesky 分解	$O(n^3+n^2+n)$

表 3.3 实数基本运算的计算复杂度

运算类型	计算复杂度
加(减)法	1
乘法	3
倒数	25
平方根	25
自然指数	25
自然对数	25
正弦	25
余弦	25
其它	100

其他各数据计算$\hat{V}$的计算复杂度量级如表 3.4 所示：

表 3.4 各数据计算复杂度

数据	乘幂法复杂度	相关性分析复杂度	降阶法复杂度	复杂度总值
DATA1	2.48E+10	2.43E+06	1.39E+11	1.64E+11
DATA2	2.81E+10	2.43E+06	1.50E+11	1.78E+11
DATA3	3.21E+10	2.43E+06	1.85E+11	2.17E+11
DATA4	1.24E+10	2.43E+06	6.52E+10	7.76E+10
DATA5	1.85E+10	2.43E+06	9.05E+10	1.09E+11
DATA6	2.05E+10	2.43E+06	9.95E+10	1.20E+11

对于题目(2)中近似分析模型$\widehat{W}$的复杂度计算，我们采用 Cholesky 分解的求逆矩阵的方法，该方法总的乘除法次数：$\frac{n^3}{6}+n^2-\frac{7n}{6}$，总的加减法次数：$\frac{n^3}{6}+\frac{3n^2}{4}-n$。经过计算该方法的计算复杂度为 2.50E+07。总体而言，乘幂法可以降低计算复杂度。

四、问题二的建模与求解

4.1 任务二问题分析

基于给定的所有矩阵数据H和W，分析各自数据间的关联性，分别设计相应的压缩$P_1(\cdot)$、$P_2(\cdot)$和解压缩$G_1(\cdot)$、$G_2(\cdot)$模型，在满足误差的情况下，使得存储复杂度和压缩与解压缩的计算复杂度最低。

为了考虑存储复杂度，我们首先需要临接矩阵之间的关联性，并建立表示模型。事实上，为了节省存储开销，需要考虑对H和W分别进行压缩。首先，我们考虑对数据矩阵H进行压缩。初步的解题流程如图 4.1 所示。

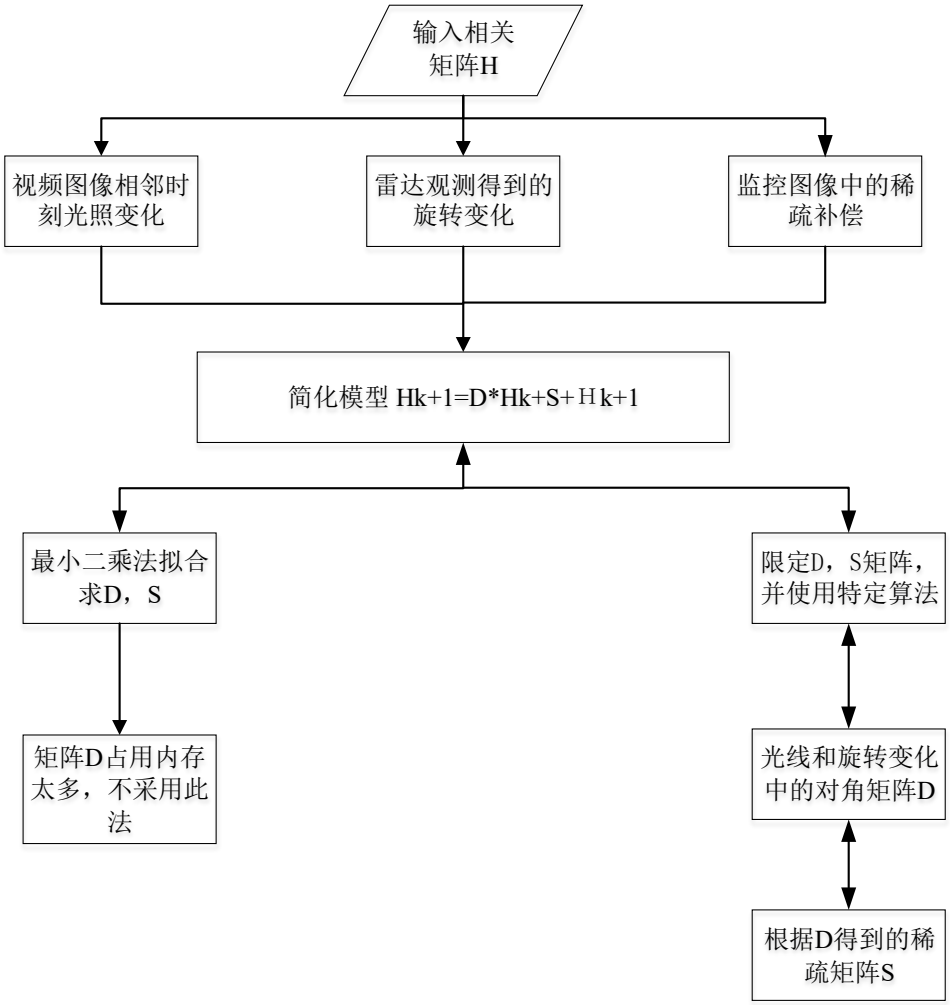


图 4.1 问题二初等模型示意图

4.2 矩阵相关性分析

H 的关联性我们在问题 1 中已经分析过了，故此小题不在赘述。

经分析同一行即 j 相同下的$W_{j,\ k}$与$W_{j,\ k+1}$之间有较强的相关性，而且是相同列之间的相关性很强。然而有少量$W_{j,\ k}$与$W_{j,\ k+1}$之间相关系数不高，第一列的相关系数如下图 4.2 所示和第二列的相关系数如下图 4.3 所示：

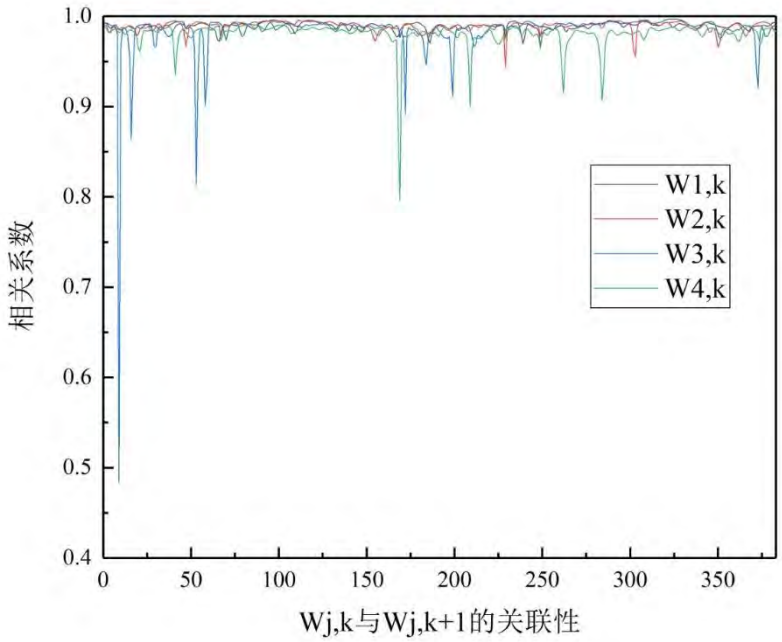


图 4.2 第一列相关系数

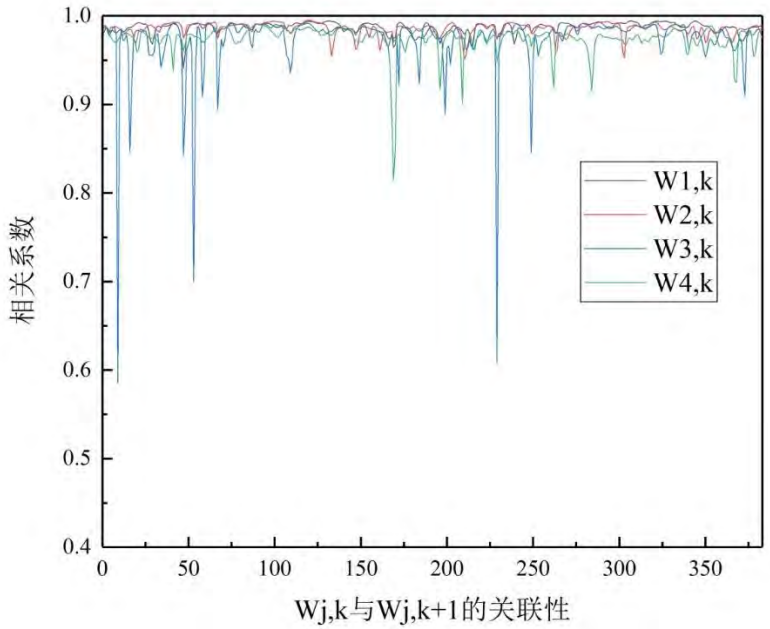


图 4.3 第二列相关系数

4.3 H 矩阵关联模型

首先，我们需要建立已知矩阵Y_k（m * n阶矩阵）之间关联性的模型。矩阵之间的关联性包括多种形式：一个矩阵通过变换（旋转、平移）可以得到另一个矩阵，或两个矩阵之间有微小的差别。比如相控阵雷达、视频信号中的单帧图像可视为一个矩阵，连续的多帧图像组成了相关矩阵组，而相邻图像帧或图像帧内像素间的关联性则反映在矩阵间的相关性上。如何建立模型表示？事实上，连续变化的图像帧之间的关联性体现在：

（1） 监控图像中有新的物体（目标）出现，一般为小的物体（比如一辆汽车进入镜头），即在一帧图像中占据很小的部分，所以不妨采用稀疏矩阵（稀疏补偿）表示：

$$H_{j,\ k+1} = H_{j,\ k} + S \text{.....(55)}$$

这里S表示稀疏矩阵。

（2） 视频图像在相邻时刻，由于光线的变化会造成图像的光照强度的细微变化，这可以通过一个对角阵相乘（因为本问题考虑行相关，所以考虑左乘）：

$$H_{j,\ k+1} = DH_{j,\ k} \text{.....(56)}$$

这里$D = diag(d_1,\ d_2,\ \cdots,\ d_M)$是对角阵，其中$d_i$表示对图像（矩阵）的第$i$行观测强度的的增益。

（3）而相控阵雷达（包括声呐）得到的观测矩阵，是相邻位置的观测，可以建立旋转变化模型，比如 Householder 变换（这一变换将一个向量变换为由一个超平面反射的向量。这可以近似表示摄像头改变角度）

$$H_{j,\ k+1} = H_{j,\ k}H_s \text{.....(57)}$$

这里$H_s = I - 2vv^T$表示 Householder 变换，其中v是单位向量。

很显然，通常连续图像的关联基本上包括上述三种情形，故总的模型

$$H_{j,\ k+1} = DH_{j,\ k}H_s + S + n_{j,\ k+1} \text{.....(58)}$$

这里$n_{j,\ k+1}$表示噪声，不妨假设为白噪声。

我们需要从数据中计算$D,\ H_s,\ S$。显然，不能直接利用最小二乘拟合，因为这样做会将问题转化为非线性形式，求解会带来巨大的计算量。另外，这是矩阵模型，也无法通过诸如取对数方法线性化。

为此，首先我们考虑简化的模型

$$H_{j,\ k+1} = DH_{j,\ k} + S + n_{j,\ k+1} \text{.....(59)}$$

但是上述模型并不是通常意义上的线性模型。考虑到模型的目标是减少存储，这里S是稀疏矩阵，D是一个对角矩阵。存储$H_{j,\ k+1}$变为存放D，S。因为对未知矩阵D，S做了限定，所以不能直接利用最小二乘法。为此，我们设计算法。考虑到光照和旋转是总体特征，具有优先性，而稀疏补偿是局部的。在先不考虑稀疏补偿的情况下，我们有

$$H_{j,\ k+1}H_{j,\ k+1}^T = DH_{j,\ k}H_{j,\ k}^T D^T \text{..... (60)}$$

$$D = H_{j,\ k+1}H_{j,\ k+1}^T \left(H_{j,\ k}H_{j,\ k}^T\right)^{-1} \text{.....(61)}$$

为了进一步较少存储，取D为对角阵。故，我们要求：

$$\sum_{n=1}^{N}\left(H_{j,\ k+1}(i,\ n) - d_i H_{j,\ k}(i,\ n)\right)^2 = \min \text{ .....(62)}$$

即

$$d_i = \frac{\sum_{n=1}^{N} H_{j,\ k+1}(i,\ n)H_{j,\ k}(i,\ n)}{\sum_{n=1}^{N} H_{j,\ k}^2(i,\ n)},\ i = 1,\ 2,\ \cdots,\ M \text{ .....(63)}$$

在确定了对角阵D后，我们再来估计稀疏矩阵S。记

$$C = H_{j,\ k+1} - DH_{j,\ k} \quad \text{.....(64)}$$

定义

$$C_{max} = \max_{\substack{1\le i\le M \\ 1\le j\le N}} |C_{ij}| \quad \text{.....(65)}$$

稀疏矩阵S定义如下：

$$s_{i,\ j} = \begin{cases} 0 \ \ if\ abs(s_{i,\ j}) \le TC_{max} \\ s_{i,\ j}, otherwise \end{cases} \quad \text{..... (66)}$$

其中T是一个阈值，这里我们取$T = 0.65$。也可以通过分析 3σ准则来取。 如果不限定矩阵D为对角阵，逼近效果会更好。上述矩阵的关联矩阵模型可以推广到存放W。

然而，通过计算我们发现上述方法难以达到题目的精度要求：-30dB。为此，需要进行改进。改进模型见图 4.4。

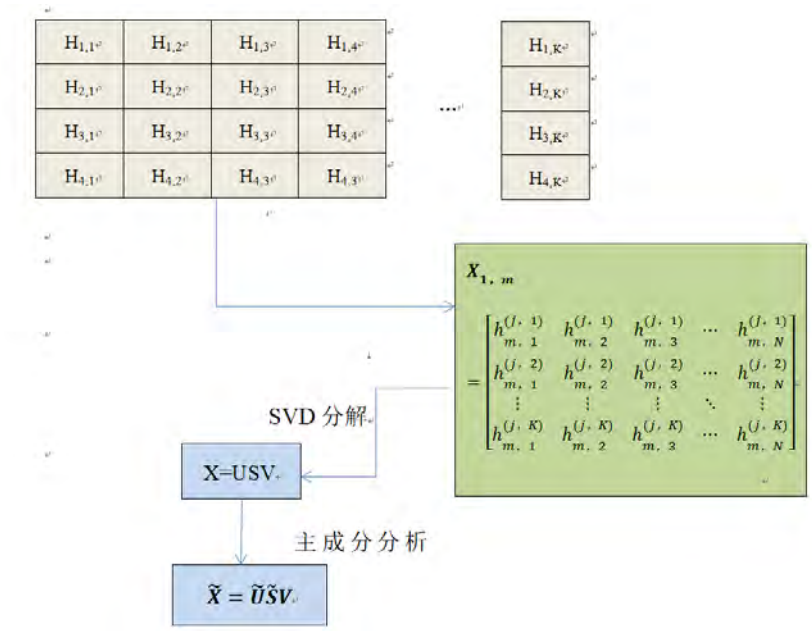


图 4.4 问题二改进模型示意图

考虑到分块矩阵的行具有高度的相关性，我们建立对矩阵组：

$$\left\{H_{j,\ 1},\ H_{j,\ 2},\ H_{j,\ 3},\ \cdots,\ H_{j,\ K}\right\} \quad \text{..... (67)}$$

在固定j时，以$H_{j,\ k}$的同一行构造矩阵，原矩阵$H = \left\{H_{j,\ k}\right\}$就转换成$X = \left\{X_{j,\ m}\right\}$

$$\boldsymbol{X_{j,\ m}} = \begin{bmatrix} h_{m,\ 1}^{(j,\ 1)} & h_{m,\ 2}^{(j,\ 1)} & h_{m,\ 3}^{(j,\ 1)} & \cdots & h_{m,\ N}^{(j,\ 1)} \\ h_{m,\ 1}^{(j,\ 2)} & h_{m,\ 2}^{(j,\ 2)} & h_{m,\ 3}^{(j,\ 2)} & \cdots & h_{m,\ N}^{(j,\ 2)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ h_{m,\ 1}^{(j,\ K)} & h_{m,\ 2}^{(j,\ K)} & h_{m,\ 3}^{(j,\ K)} & \cdots & h_{m,\ N}^{(j,\ K)} \end{bmatrix}$$

由于$X_{j,\ m}$是一个高度相关的矩阵，可以利用 PCA 或奇异值分解

$$X_{j,\ m} = U_{j,\ m} S_{j,\ m} V_{j,\ m} \quad (68)$$

其中奇异值$\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_{64}$，选定$k_0$满足

$$\frac{\sum_{i=1}^{k_0} \sigma_i}{\sum_{i=1}^{64} \sigma_i} \geq 99\% \quad (69)$$

即意味着保留的成分达 99%，即精度可达-30dB。

$$\tilde{X}j,\ m \approx \tilde{U}j,\ m\ \tilde{S}j,\ m\, Vj,\ m \quad (70)$$

而$\tilde{\boldsymbol{S}}$的奇异值为：$\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_{k_0} > 0$.这里的$\tilde{U}j,\ m$只是$\boldsymbol{U}$的前$k_0$列。

故可通过存储的$\tilde{U}j,\ m$，$\tilde{S}j,\ m$，$Vj,\ m$求出$\{\tilde{X}j,\ m\}$，再经过$\tilde{X}$矩阵的排列拼接还原出$\tilde{H}$。

$\tilde{U}j,\ m$为 384*k_0阶的复数矩阵，所需存储空间为 2.46E+04k0 比特；$V_{j,\ m}$为 64*64 阶的复数矩阵，所需存储空间为 2.62E+05 比特；$\tilde{S}j,\ m$只存储了k_0个非 0 实数，所需存储空间为 32k_0比特。

以 Data2 中的数据为例，k_0的值为如下表 4.1 所示：

表 4.1 DATA2K$_0$值

$X_{j,\ m}$对应的 k_0	m=1	m=2	m=3	m=4
j=1	29	27	28	30
j=2	29	33	26	30
j=3	29	28	29	25
j=4	29	29	24	30

故$\{X_{j,\ m}\}$对应的全部的$\tilde{U}$，$\tilde{S}$，V存储空间为 1.54E+07 比特，即压缩后的$\{H_{j,\ k}\}$存储空间为 1.54E+07 比特，而原矩阵$H = \{H_{j,\ k}\}$包含 4*384 个 4*64 阶的复数矩阵，存储空间为2.52E+07 比特，压缩后存储空间减少了 38.8%，故这种压缩方式非常节约芯片的存储空间。

经过压缩和解压缩之后得到的$\hat{H}$原始的H间的误差定义为

$$err_{\boldsymbol{H}} = 10 * \log_{10} \frac{E\left\{\left\|\hat{\boldsymbol{H}}_{j,\ k} - \boldsymbol{H}_{j,\ k}\right\|_F^2\right\}}{E\left\{\left\|\boldsymbol{H}_{j,\ k}\right\|_F^2\right\}} \quad (71)$$

六组数据 Data1、2、3、4、5、6 用此种压缩和解压方式，解压出来的$\tilde{H}$ 与原 H 间的误差如下表 4.2 所示：

表 4.2 解压出来的$\tilde{H}$ 与原 H 间的误差

Data1	Data2	Data3	Data4	Data5	Data6
-30.1391	-30.1408	-30.1708	-30.1458	-30.1349	-30.1652

由表可得$err_{\boldsymbol{H}}$ 均小于-30dB，由此可见，压缩及解压方法 d 得到的$\tilde{H}$准确性很高。

4.4 W 矩阵关联模型

W 的压缩及解压缩模型与 H 的类似，{W}是由 $J\times K\times Mw\times Nw = 4\times 384$ 个 64×2 阶的矩阵$W_{j,k}$ 组成，由于$W_{j,k}$与$W_{j,k+1}$相同列之间关联性很高，故在固定j时，以$W_{j,k}$的同一列构造矩阵,原矩阵$W=\{W_{j,k}\}$就转换成$Y=\{Y_{j,nw}\}$ (Nw=2,,故 nw=1,2)。

$$\boldsymbol{Y_{j,nw}}=\begin{bmatrix} w_{1,nw}^{(j,1)} & w_{1,nw}^{(j,2)} & w_{1,nw}^{(j,3)} & \cdots & w_{1,nw}^{(j,K)} \\ w_{2,nw}^{(j,1)} & w_{2,nw}^{(j,2)} & w_{2,nw}^{(j,3)} & \cdots & w_{2,nw}^{(j,K)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ w_{m,nw}^{(j,1)} & w_{m,nw}^{(j,2)} & w_{m,nw}^{(j,3)} & \cdots & w_{m,nw}^{(j,K)} \end{bmatrix}$$

由于$Y_{j,nw}$是一个高度相关的矩阵，可以利用 PCA 或奇异值分解。

$$Y_{j,nw}=U_{j,nw}S_{j,nw}V_{j,nw}\qquad(72)$$

其中奇异值$\sigma_{1w}\geq\sigma_{2w}\geq\cdots\geq\sigma_{64w}$，选定$k_{0w}$满足

$$\frac{\sum_{i=1}^{k_0}\sigma_{iw}}{\sum_{i=1}^{64}\sigma_{iw}}\geq 99\%\qquad(73)$$

即意味着保留的成分达 99%，即精度可达-30dB.

$$\tilde{Y}j,nw\approx\tilde{U}j,nw\ \tilde{S}j,nw\ Vj,nw\qquad(74)$$

而$\tilde{\boldsymbol{S}}$的奇异值为：$\sigma_{1w}\geq\sigma_{2w}\geq\cdots\geq\sigma_{k_{0w}}>0$.这里的$\tilde{U}j,nw$只是$\boldsymbol{U}$的前$k_0$列。

故可通过存储的$\tilde{U}j,nw$， $\tilde{S}j,nw$ ，Vj,nw求出出$\{\tilde{Y}j,nw\}$，再经过$\tilde{Y}$矩阵的排列拼接还原出$\tilde{W}$。

$\tilde{U}j,nw$为 64×k_{0w}阶的复数矩阵，所需存储空间为 4.10E+03k0 比特；$V_{j,nw}$ 为 384×384 阶的复数矩阵，所需存储空间为 9.44E+06 比特；$\tilde{S}j,nw$只存储了k_{0w}个非 0 实数，所需存储空间为 32k_{0w}比特。

以 Data6 中的数据为例，k_{0w}的值为如下表 4.3 所示：

表 4.3 DATA6k_{0w}值

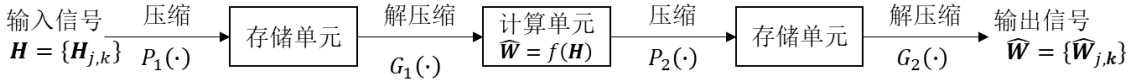
$Y_{j,nw}$对应的k_{0w}	nw=1	nw=2
j=1	37	43
j=2	40	43
j=3	37	42
j=4	37	38

可算得$\{Y_{j,nw}\}$对应的全部的$\tilde{U}j,nw$， $\tilde{S}j,nw$ ，Vj,nw存储空间为 7.68E+07 比特，即压缩后的$\{W_{j,k}\}$存储空间为 7.68E+07 比特，而原矩阵$W=\{W_{j,k}\}$包含 4×384 个 64×2 阶的复数矩阵，存储空间为 1.26E+07 比特。压缩后的$\tilde{W}$存储空间比原矩阵 W 的存储空间大，故此种压缩方式不适用于 W。

五、问题三的建模与求解

5.1 任务三问题分析

相关矩阵组的整体处理流程如下图所示。



先利用存储模型：

$$\widetilde{H}_{j,\ k+1}=DH_{j,\ k}+S \quad \text{.....(75)}$$

再利用乘幂法和降阶法计算矩阵$\widetilde{H}_{j,\ k+1}$的右奇异向量（代替计算$H_{j,\ k+1}$）。需要注意的是该过程中，每一步迭代需增加乘法次数：16+N_S

其中N_S表示稀疏矩阵中非零元素的个数。

六、总结

6.1 选择乘幂法与降阶法的原因

乘幂法的思想是简单而又重要的，由此可诱导出一些重要而更有效的方法，如同时迭代法等。同时它和它的变形特别适用于大稀疏矩阵（含有大量零元素的矩阵）。对于计算三对角阵或 Hessenberg 阵对应于一个给定特征值的特征向量，乘幂法的是最佳方法之一。

本题我们大胆运用乘幂法对前一个值进行迭代求出下一个值，来计算题目中的近似分析模型$\hat{V}$、$\hat{W}$，通过对复杂度的计算，我们发现乘幂法与降价法的复杂度公式是 m*n，公式非常简便，故此坚定了我们选择其来降低计算过程中产生的计算复杂度。

6.2 做题过程中的缺憾

我们在解决第一问时，发现乘幂法所求得最大特征向量的模型精度有部分点会低于 0.99，考虑到芯片的第一标准应该是高精度，所以我们将精度的阈值提高至 0.99999，这也导致了计算复杂度的提高。但是总体的计算复杂度还是低于利用 Golub-Kohn 奇异值分解方法的计算复杂度。在寻求高精度和低复杂度的平衡点上，我们依旧任重道远，希望本文的方法可以为精进中国芯片贡献出自己的力量。

参考文献

[1] Golub， Gene， and William Kahan. "Calculating the singular values and pseudo-inverse of a matrix." Journal of the Society for Industrial and Applied Mathematics， Series B: Numerical Analysis 2.2 (1965): 205-224.

[2] 马文 Marvin， 求解矩阵特征值的 QR 算法，https://blog.csdn.net/qq_40922398/article/details/112788453?utm_medium=distribute.pc_relevant.none-task-blog-2%7Edefault%7EBlogCommendFromBaidu%7Edefault-6.no_search_link&depth_1-utm_source=distribute.pc_relevant.none-task-blog-2%7Edefault%7EBlogCommendFromBaidu%7Edefault-6.no_search_link，2021/10/16

[3] Eterbity，模拟退火算法详细讲解（含实例 python 代码），https://blog.csdn.net/weixin_48241292/article/details/109468947，2021/10/16