Appendix 4.1: The Schur Form

Not all matrices can be diagonalized by a similarity transformation. Actually, an $n \times n$ matrix can be diagonalized if and only if it has $n$ independent eigenvectors. However, all $n \times n$ matrices are similar to an upper-triangular matrix containing a diagonal of eigenvalues. As argued in this appendix, given arbitrary $n \times n$ matrix $A$, there exists an $n \times n$ unitary $U$ ($U^* = U^{-1}$ or $U^*U = I$) such that $U^*AU$ is upper-triangular with a diagonal containing eigenvalues. An upper-triangular matrix, containing a diagonal of eigenvalues, is known as a Schur Form. There are many numerical algorithms that start by converting a supplied matrix to its Schur Form. In Appendix 4.2, we show that a diagonal matrix is obtained by reducing an Hermitian matrix to Schur form. In Chapter 4, we use this fact in the development of the matrix 2-norm.

**Theorem (Schur Decomposition)**

For any $n \times n$ matrix $A$, there is an $n \times n$ unitary matrix $U$ (i.e., $U^* = U^{-1}$ or $U^*U = I$) such that $T = U^*AU$ is upper-triangular (i.e., everything below the diagonal is zero). Furthermore, the eigenvalues of $A$ appear on the main diagonal of $T$.

**Proof:** To make this simple, assume that $A$ is $4 \times 4$. The general $n \times n$ result will be evident once the simple $4 \times 4$ is understood. Let $\lambda_1$ and $\bar{X}_1$ be an eigenvalue and eigenvector, respectively, of $A$ (in the “worse” case, $\lambda_1$ could be repeated 4 times). Assume that eigenvector $\bar{X}_1$ has been normalized so that $\|\bar{X}_1\| = 1$. Let $\tilde{X}_1$ be the first column of an $n \times n$ matrix $U_1$. Fill out the remaining three columns of $U_1$ in any way that makes all columns orthonormal and $U_1^*U_1 = I$ (for example, find three independent vectors that are independent of $\tilde{X}_1$, and apply the Gram-Schmidt process to make all four vectors orthonormal). Now, the product $U_1^*AU_1$ has its first column in the “right” form: $A\tilde{X}_1 = \lambda_1\bar{X}_1$ means that
where \( ? \) denotes a generally nonzero, unknown value. Basically, this procedure is repeated until the result is upper triangular. At the second step of the procedure, we work with the \( 3\times3 \) matrix that appears “partitioned off” in the lower right-hand corner of \( U_1^*AU_1 \). This \( 3\times3 \) matrix has an eigenvalue \( \lambda_2 \); and it has a \( 3\times1 \) normalized eigenvector \( \vec{X}_2 \) that can be made into the first column of a \( 3\times3 \) unitary matrix \( M_2 \). Now, use \( M_2 \) as a lower block, and define \( 4\times4 \) matrix \( U_2 \) as

\[
U_2 = \begin{bmatrix}
1 & 0 & 0 & 0 \\
0 & M_2 \\
0 & 0 & 0 \\
0 & 0 & 0 & 0
\end{bmatrix}
\]

so that

\[
U_2^*(U_1^*AU_1)U_2 = \begin{bmatrix}
\lambda_1 & ? & ? & ? \\
0 & \lambda_2 & ? & ? \\
0 & 0 & ? & ? \\
0 & 0 & 0 & ?
\end{bmatrix}
\]

Now, we work with the \( 2\times2 \) matrix that appears “partitioned off” in the lower right-hand corner of \( U_2^*(U_1^*AU_1)U_2 \). This \( 2\times2 \) matrix has an eigenvalue \( \lambda_3 \), and it has a \( 2\times1 \) normalized eigenvector \( \vec{X}_3 \) that can be made into the first column of a \( 2\times2 \) unitary matrix \( M_3 \). Now, use \( M_3 \) as a lower block, and define \( 4\times4 \) matrix \( U_3 \) as
so that

$$U_3^*(U_2^*(U_1^*AU_1)U_2)U_3 = \begin{bmatrix} \lambda_1 & ? & ? & ? \\ 0 & \lambda_2 & ? & ? \\ 0 & 0 & \lambda_3 & ? \\ 0 & 0 & 0 & ? \end{bmatrix}, \quad U_4 = \begin{bmatrix} \text{3 \times 3 identity block} & 0 \\ 0 & 0 \end{bmatrix}. \quad (4.1-5)$$

Finally, the last step. Clearly, a unitary $U_4$ (containing a upper $3 \times 3$ identity block) exists that can be applied to turn the previous result into an upper triangular matrix with the eigenvalues $\lambda_1, \lambda_2, \lambda_3$ and $\lambda_4$ on its diagonal. Then $U = U_4U_3U_2U_1$ is unitary (product of unitary matrices is unitary), and $U^*AU$ is upper triangular with eigenvalues on its diagonal (no restriction on the eigenvalues: they can be real, complex, zero, anything!).

**MatLab’s Schur Function**

Let $A$ denote an arbitrary $n \times n$ matrix. MatLab can be used to compute the Schur Decomposition of $A$; however, the process may require two steps. The MatLab syntax is

$$[U,T] = \text{schur}(A). \quad (4.1-6)$$

If $A$ is a complex-valued matrix, the above statement returns a unitary $U$ and the Schur form $T$. If $A$ is a real-valued matrix with real-valued eigenvalues, the above statement returns an orthogonal $U$ (i.e., $U^TU = I$) and the Schur form $T$. If $A$ is a real-valued matrix with complex-valued eigenvalues, the above statement returns an orthogonal $U$ (i.e., $U^TU = I$) and what MatLab calls
the *Real-Valued Schur Form* for \( T \) (the *Real-Valued Schur Form* is not discussed here). In this case, one additional step is required to get the Schur Form. Follow the \([U,T] = \text{schur}(A)\) statement with

\[
[U,T] = \text{rsf2csf}(U,T)
\]

(4.1-7)

to get a unitary \( U \) and the upper-triangular, eigenvalue-on-the-diagonal, Schur form \( T \).

**MatLab Example: Schur Decomposition of Real Matrix with Real Eigenvalues**

% Enter Matrix A

\[
>A = \begin{bmatrix}
1 & 2 & 3 \\
4 & 5 & 6 \\
7 & 8 & 9
\end{bmatrix}
\]

\[A = \begin{bmatrix}
1 & 2 & 3 \\
4 & 5 & 6 \\
7 & 8 & 9
\end{bmatrix}
\]

> \([U,T] = \text{schur}(A)\)

\[
U = \begin{bmatrix}
0.2320 & 0.8829 & 0.4082 \\
0.5253 & 0.2395 & -0.8165 \\
0.8187 & -0.4039 & 0.4082
\end{bmatrix} \quad T = \begin{bmatrix}
16.1168 & 4.8990 & 0.0000 \\
0.0000 & -1.1168 & 0.0000 \\
0.0000 & 0.0000 & 0.0000
\end{bmatrix}
\]

% \( U \) is orthogonal

\[
U^T U = \begin{bmatrix}
1 & 0 & 0 \\
0 & 1 & 0 \\
0 & 0 & 1
\end{bmatrix}
\]

% \( U^T A U \) is in Schur Form (note eigenvalues along diagonal)!

\[
U^T A U = \begin{bmatrix}
16.1168 & 4.8990 & 0.0000 \\
0.0000 & -1.1168 & 0.0000 \\
0.0000 & 0.0000 & 0.0000
\end{bmatrix}
\]

% The Schur Form; it is upper-triangular, and it has eigenvalues on its diagonal. Same as \( T \) above (as obtained from MatLab's \text{rsf2csf} function)
MatLab Example: Schur Decomposition of Real Matrix with Complex Eigenvalues

% Enter Real-Valued Matrix A

> A = [0 1 0 0; -1 0 0 0; 0 0 1 2; 0 0 3 4] 

A =

```
0 1 0 0
-1 0 0 0
0 0 1 2
0 0 3 4
```

% Matrix A has both real and complex valued eigenvalues

> eig(A)

(0 + 1.0000j), (0 - 1.0000j), -0.3723, 5.3723

% Compute what MatLab Calls the Real-Valued Schur Form (not the Schur Form)

> [U,T] = schur(A)

```
U =

1 0 0 0
0 -1 0 0
0 0 -0.8246 -0.5658
0 0 0.5658 -0.8246

T =

0 -1 0 0
1 0 0 0
0 0 -0.3723 -1
0 0 0 5.3723
```

% T is not upper-triangular; it is not what most folks call the Schur form (MatLab calls it the Real-Valued Schur Form). However, it can be converted to the upper-triangular Schur Form by using MatLab's function rsf2csf.

> [U,T] = rsf2csf(U,T)

```
U =

(0 + 0.7071j) -0.7071 0 0
-0.7071 (0 + 0.7071j) 0 0
0 0 -0.8246 -0.5658
0 0 0.5658 -0.8246

T =

(0 + 1.0000j) 0 0 0
0 (0 - 1.0000j) 0 0
0 0 -0.3723 -1
0 0 0 5.3723
```
% This is upper-triangular; this is what most folks call the Schur Form. HEY! Check it Out! Compute U*AU and make sure you get T given above!

\[
U^*AU = \begin{bmatrix}
(0 + 1.0000j) & 0 & 0 & 0 \\
0 & (0 - 1.0000j) & 0 & 0 \\
0 & 0 & -0.3723 & -1 \\
0 & 0 & 0 & 5.3723 
\end{bmatrix}
\]

% Same as T above. Upper triangular with eigenvalues on the diagonal - The Schur Form! ♥