Numerical Simulations of DSGE Models with MATLAB®
– Inside the Blackbox of Dynare –

Dipl.-Vw./Dipl.-Kfm. Sebastian Krug

Part of the Lecture on
Macroeconomic Dynamics and Optimal Monetary Policy

June 05th, 2013
Organizational Matters

- EMail: s.krug@economics.uni-kiel.de
- Syllabus:
  - 05.06.2013 (RBC/Jordan Decomposition)
  - 12.06.2013 (NKM (closed)/Schur)
  - 19.06.2013 (NKM (SOE)/QZ Factorization)
  - 26.06.2013 (Monetary Union/Aoki/QZ Factorization)
Motivation

Why do we need to learn how to apply matrix decomposition methods?

- RE Macro models typically involve (non-)predetermined variables
- solutions require numerical methods to find the RE equilibrium (or the optimal (monetary) policy)
- Most people use Dynare for numerical simulations of DSGE models
  - just specify the core of the model
    - endogenous variables (state variables)
    - exogenous variables (shocks)
    - parameters of the model
    - and its state equations
  - just run code ⇒ software package solves/simulates

⇒ only few people know exactly what happens to produce the output
- This is why Dynare is often called a “Blackbox”
- to provide an insight into the Blackbox we will shed light on the used matrix decomposition algorithms
Motivation

Why do we need to learn how to apply matrix decomposition methods?

General framework:

1. build a system of dynamic equations (state equations)
2. set up the state space representation (i.e. matrix notation of the dyn. eq. system)

\[ A \begin{pmatrix} w_{t+1} \\ v_{t+1} \end{pmatrix} = B \begin{pmatrix} w_t \\ v_t \end{pmatrix} \]

3. transform/decompose the dyn. eq. system such a way that one obtains two *subsystems* of which one contains
   - the predetermined variables (\(\rightarrow\) can be solved backward)
   - the non-predetermined variables (\(\rightarrow\) can be solved forward)

⇒ The eigenvalues of the system matrices determine the behavior of the dyn. system

4. to obtain the eigenvalues of the system matrices and in order to arrange the system in an appropriate way we have to apply a suitable matrix decomposition method
### Matrix Decomposition Methods: Overview

<table>
<thead>
<tr>
<th>Decomposition Method</th>
<th>Form of the System</th>
<th>Decomposition</th>
<th>Corresponding EV Problem</th>
<th>Preconditions</th>
</tr>
</thead>
</table>
| Jordan               | \[
\begin{pmatrix}
v_{t+1} \\
E_t v_{t+1}
\end{pmatrix} = C \begin{pmatrix} w_t \\ v_t \end{pmatrix}
\] with \( C = A^{-1} B \) | \( C = H \Lambda H^{-1} \) | ??? | ??? |
| Schur                | \[
\begin{pmatrix}
v_{t+1} \\
E_t v_{t+1}
\end{pmatrix} = C \begin{pmatrix} w_t \\ v_t \end{pmatrix}
\] with \( C = A^{-1} B \) | \( C = Z T Z' \) | ??? | ??? |
| QZ Factorization (Generalized Schur) | ??? | ??? | ??? | ??? |

**Tabelle:** Matrix Decomposition Methods in Macroeconomic Dynamics: An Overview
We got two state variables: $k_t$ (predetermined) and $c_t$ (non-predetermined).

In order to obtain stable solutions of the state equations we must decompose the dynamic equation system into 2 subsystems:

$$B \begin{pmatrix} E_t k_{t+1} \\ E_t c_{t+1} \end{pmatrix} = C \begin{pmatrix} k_t \\ c_t \end{pmatrix} + \begin{pmatrix} \tilde{d}_1 \\ \tilde{d}_2 \end{pmatrix} a_t$$

Therefore, apply the Jordan decomposition method to decompose the system matrix $A = B^{-1}C$ of the (explicit) state space representation:

$$A = H \Lambda H^{-1}$$

1. Precondition: $B$ has to be invertible.
Jordan Decomposition

MATLAB command:

\[
[H, \Lambda] = \text{jordan}(A)
\]

Definition

The Jordan decomposition decomposes system matrix \( A \) into its Jordan canonical form \( \Lambda \) and the similarity transform \( H \).

\[
\Lambda = \begin{pmatrix}
|\lambda_1| & < & 1 & 0 \\
0 & |\lambda_2| & > & 1
\end{pmatrix}
\]

contains the eigenvalues that solve the (standard) eigenvalue problem

\[
Av = \lambda v
\]

The matrix \( H \) contains the (generalized) eigenvectors of \( A \) as columns.
Jordan Decomposition
Limitations of the Jordan Decomposition (theoretical)

- Requirement for existence of a solution: \( H \) must be non-singular (i.e. its inverse exists)
- If \( A \) has multiple eigenvalues (\( A \) is said to be defective) the dimension of the associated eigenspace is reduced (number of linear independent eigenvectors < \( n \))
- in this case there are not sufficiently many independent eigenvectors to span the entire space

\[ \Rightarrow H^{-1} \text{ does not exist} \Rightarrow \text{No solution of the Jordan Decomposition} \]
- Solution: Use generalized eigenvectors which leads always to a full set of linear independent eigenvectors
Jordan Decomposition

Limitations of the Jordan Decomposition (numerical)

⇒ The theoretical limitation induces a numerical limitation due to the fact that \( \Lambda \) isn’t a diagonal matrix anymore

\[
\Lambda_{\text{ordinary}} = \begin{pmatrix}
  \lambda_i & 0 & 0 \\
  0 & \lambda_i & 0 \\
  0 & 0 & \lambda_i
\end{pmatrix} \quad \Rightarrow \quad \Lambda_{\text{generalized}} = \begin{pmatrix}
  \lambda_i & 1 & 0 \\
  0 & \lambda_i & 0 \\
  0 & 0 & \lambda_i
\end{pmatrix}
\]

• Even small perturbations of \( \Lambda \) can lead to the disappearance of its property of defectiveness

• Then the “ones” above the main diagonal also disappear (because there is no need of generalized eigenvalues)

⇒ no robust numerical results
## Aim of this Part of the Lecture

Matrix Decomposition Methods: Overview

<table>
<thead>
<tr>
<th>Decomposition Method</th>
<th>Form of the System</th>
<th>Decomposition</th>
<th>Corresponding EV Problem</th>
<th>Preconditions</th>
</tr>
</thead>
</table>
| Jordan               | \[
\begin{pmatrix}
    w_{t+1} \\
    E_t v_{t+1}
\end{pmatrix} = C \begin{pmatrix}
    w_t \\
    v_t
\end{pmatrix}
\]
with \( C = A^{-1} B \) | \( C = H \Lambda H^{-1} \) | \( C v = \lambda v \) | \( \cdot A \) has to be non-singular |
| Schur                | \[
\begin{pmatrix}
    w_{t+1} \\
    E_t v_{t+1}
\end{pmatrix} = C \begin{pmatrix}
    w_t \\
    v_t
\end{pmatrix}
\]
with \( C = A^{-1} B \) | \( C = Z T Z' \) | ??? | ??? |
| QZ Factorization (Generalized Schur) | ??? | ??? | ??? | ??? |

**Tabelle:** Matrix Decomposition Methods in Macroeconomic Dynamics: An Overview
Schur Decomposition
The Decomposition

MATLAB command:

\[
[Z, T] = \text{schur}(A, \ '\text{complex}')
\]

Definition

Any \( n \times n \) square matrix \( A \) (here with complex entries) can be decomposed into

\[
A = Z T \bar{Z}'
\]

where \( Z \) is a unitary matrix with

\[
Z \cdot \bar{Z}' = \bar{Z}' \cdot Z = I_{n \times n}
\]

and \( T \) is an upper triangular matrix, which is called the Schur form of \( A \). The decomposition solves also the standard eigenvalue problem

\[
A v = \lambda v
\]
Schur Decomposition

The Rearrangement of EV or the creation of two distinct subsystems
(Logical vector with cluster indices)

MATLAB command:

\[
[Z_{re}, T_{re}] = \text{ordschur}(Z,T,[\text{logical vector}])
\]

\[
\begin{bmatrix}
1.1456 \\
1.3245 \\
\vdots
\end{bmatrix}
\begin{bmatrix}
1.3245 \\
0.8462 \\
\vdots
\end{bmatrix}
T_{12}
\begin{bmatrix}
T_{12} \\
0 \\
\vdots
\end{bmatrix}
= T
\]

\[
\begin{bmatrix}
[1 & 1 & 2] \\
[3 & 3 & 2] \\
\vdots
\end{bmatrix}
\begin{bmatrix}
[3 & 3 & 2] \\
[1 & 1] \\
\vdots
\end{bmatrix}
\]

\[
\begin{bmatrix}
0.4481 \\
0.4985 \\
\vdots
\end{bmatrix}
\begin{bmatrix}
0.4985 \\
0.8462 \\
\vdots
\end{bmatrix}
T_{12}
\begin{bmatrix}
T_{12} \\
0 \\
\vdots
\end{bmatrix}
\Rightarrow
\begin{cases}
|\lambda_1| < 1 \\
|\lambda_2| < 1 \\
|\lambda_3| < 1 \\
|\lambda_4| > 1 \\
|\lambda_5| > 1
\end{cases}
\]
Schur Decomposition
Algorithm to find the logical vector and cluster indices

Algorithm:
- Search for clusters of gen. EV which are already in ascending order
- Assign an index to each cluster (even though the cluster consists of only one element)

⇒ The cluster with the highest index ranks first (after the rearrangement)
- Thus, the position of an index within the logical vector assigns its value to the eigenvector at the corresponding position within the matrix $T$
- The value of an index determines the affiliation of the appropriate eigenvalue to a cluster and, therefore, its new position in $T$ after the rearrangement

Example:
The logical vector $[2 \ 3 \ 3 \ 1 \ 1 \ 1 \ 1 \ 2]$ contains 3 different clusters with indices 1, 2 and 3. After applying the `ordschur()`-command the rows of $T$ and $Z$ are rearranged in such a way that the clusters appear in the following order: $[3 \ 3 \ 2 \ 2 \ 1 \ 1 \ 1]$

(Note: The following logical vectors lead to the same ordering:

$[2 \ 3 \ 3 \ 1 \ 1 \ 1 \ 2] = [1 \ 2 \ 2 \ 0 \ 0 \ 0 \ 1] = [3 \ 4 \ 4 \ 2 \ 2 \ 2 \ 3] = [207 \ 439 \ 439 \ 77 \ 77 \ 77 \ 207]$)
# Matrix Decomposition Methods

## Overview

<table>
<thead>
<tr>
<th>Decomposition Method</th>
<th>Form of the System</th>
<th>Decomposition</th>
<th>Corresponding EV Problem</th>
<th>Preconditions</th>
</tr>
</thead>
</table>
| Jordan               | \[
\begin{pmatrix} w_{t+1} \\ E_t v_{t+1} \end{pmatrix} = C \begin{pmatrix} w_t \\ v_t \end{pmatrix} \\
\text{with } C = A^{-1} B
\] | \[ C = H \Lambda H^{-1} \] | \[ Cv = \lambda v \] | \bullet A has to be non-singular \bullet H has to be non-singular |
| Schur                | \[
\begin{pmatrix} w_{t+1} \\ E_t v_{t+1} \end{pmatrix} = C \begin{pmatrix} w_t \\ v_t \end{pmatrix} \\
\text{with } C = A^{-1} B
\] | \[ C = Z T \bar{Z}' \] | \[ Cv = \lambda v \] | \bullet A has to be non-singular |
| QZ Factorization (Generalized Schur) | \[
A \begin{pmatrix} w_{t+1} \\ E_t v_{t+1} \end{pmatrix} = B \begin{pmatrix} w_t \\ v_t \end{pmatrix}
\] | \[ A = \bar{Q}' S \bar{Z}' \]
| \[ B = \bar{Q}' T \bar{Z}' \] (generalized) | \[ Bv = \lambda Av \] | – |

Tabelle: Matrix Decomposition Methods in Macroeconomic Dynamics: An Overview
QZ Factorization (Generalized Schur Decomposition)

NKM for a small open economy

MATLAB command:

\[ [S, T, Q, Z] = qz(A, B) \]

Definition

In the complex version of the QZ Factorization

\[ A = Q' S Z' \]
\[ B = Q' T Z' \]

the ratios of the diagonal elements of \( T \) to the corresponding diagonal elements of \( S \) build the the generalized eigenvalues

\[ \lambda_i = \frac{T_{ii}}{S_{ii}} \]

that solve the generalized eigenvalue problem

\[ Bv = \lambda Av \]
Calculus with complex numbers
A broad hint for the upcoming exam

How to compute the absolute value (i.e. the modulus) $|z|$ of a complex number $z$:

$$z = a + bi$$

$$|z|^2 = |a|^2 + |b|^2 = |3|^2 + |4|^2$$

$$|z| = \sqrt{|a|^2 + |b|^2} = \sqrt{3^2 + 4^2} = 5$$

$$|z| = \sqrt{|a|^2 + |b|^2} = \sqrt{3^2 + 4^2} = 5$$
Calculus with complex numbers
A broad hint for the upcoming exam

Properties of unitary matrices:

Every (regular) matrix $Q$ has the property that

$$Q \cdot Q^{-1} = Q^{-1} \cdot Q = I_{n \times n}$$

holds.

If the matrix $Q$ is also a unitary matrix it additionally has the property that

$$Q \cdot Q' = Q' \cdot Q = I_{n \times n}$$

holds.

If the matrix is unitary

$$Q' = Q^{-1}$$

holds.
Calculus with complex numbers
A broad hint for the upcoming exam

How to compute the conjugate transpose $\overline{Q}'$ of a matrix $Q$:

$$Q = \begin{pmatrix}
0 & 0.8868 - 0.0848i & -0.1435 + 0.3717i & 0.1858 + 0.0717i & 0.0085 + 0.0887i \\
0 & 0.0869 + 0.3771i & 0.8550 + 0.0571i & 0.2355 + 0.1086i & 0.2181 + 0.0331i \\
0 & -0.2313 & -0.3022 & 0.8541 & 0.3545 \\
0 & 0.0559 & -0.1250 & -0.4044 & 0.9043 \\
1.0000 & 0 & 0 & 0 & 0
\end{pmatrix}$$

Since the *conjugate complex* of a complex number $z = a + bi$ is equal to $\overline{z} = a - bi$ one can get the conjugate complex $\overline{Q}$ of matrix $Q$ by reversing the signs of all imaginary parts:

$$\overline{Q} = \begin{pmatrix}
0 & 0.8868 + 0.0848i & -0.1435 - 0.3717i & 0.1858 - 0.0717i & 0.0085 - 0.0887i \\
0 & 0.0869 - 0.3771i & 0.8550 - 0.0571i & 0.2355 - 0.1086i & 0.2181 - 0.0331i \\
0 & -0.2313 & -0.3022 & 0.8541 & 0.3545 \\
0 & 0.0559 & -0.1250 & -0.4044 & 0.9043 \\
1.0000 & 0 & 0 & 0 & 0
\end{pmatrix}$$

Now one just has to transpose the conjugate complex $\overline{Q}$ to arrive at the conjugate transpose $\overline{Q}'$:

$$\overline{Q}' = Q^{-1} = \begin{pmatrix}
0 & 0 & 0 & 0 & 1.0000 \\
0.8868 + 0.0848i & 0.0869 - 0.3771i & -0.2313 & 0.0559 & 0 \\
-0.1435 - 0.3717i & 0.8550 - 0.0571i & -0.3022 & -0.1250 & 0 \\
0.1858 - 0.0717i & 0.2355 - 0.1086i & 0.8541 & -0.4044 & 0 \\
0.0085 - 0.0887i & 0.2181 - 0.0331i & 0.3545 & 0.9043 & 0
\end{pmatrix}$$