MUS420
Introduction to Modal Representations

Julius O. Smith III (jos@ccrma.stanford.edu)
Center for Computer Research in Music and Acoustics (CCRMA)
Department of Music, Stanford University
Stanford, California 94305

February 5, 2019

State Space Modal Representation

Diagonal state transition matrix = *modal representation*:

$$\left[\begin{array}{c} x_1(n+1) \\ x_2(n+1) \\ \vdots \\ x_{N-1}(n+1) \\ x_N(n+1) \end{array}\right] = \left[\begin{array}{ccccc} \lambda_1 & 0 & 0 & \cdots & 0 \\ 0 & \lambda_2 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \lambda_{N-1} & 0 \\ 0 & 0 & 0 & 0 & \lambda_N \end{array}\right] \left[\begin{array}{c} x_1(n) \\ x_2(n) \\ \vdots \\ x_{N-1}(n) \\ x_N(n) \end{array}\right] + \left[\begin{array}{c} b_1 \\ b_2 \\ \vdots \\ b_{N-1} \\ b_N \end{array}\right] u(n)$$

$$y(n) = C\underline{x}(n) + Du(n)$$

The N complex modes are *decoupled*:

$$\begin{aligned} x_1(n+1) &= \lambda_1 x_1(n) + b_1 u(n) \\ x_2(n+1) &= \lambda_2 x_2(n) + b_2 u(n) \\ &\vdots \\ x_N(n+1) &= \lambda_N x_N(n) + b_N u(n) \\ y(n) &= c_1 x_1(n) + c_2 x_2(n) + \cdots + c_N x_N(n) + Du(n) \end{aligned}$$

That is, diagonal state-space system consists of N *parallel one-pole systems* (complex, in general).

Diagonalizing a State-Space Model

- To obtain a *modal representation*, we can *diagonalize* a state-space model
- The *similarity transformation* which diagonalizes the system is given by the *matrix of eigenvectors* of the state transition matrix A
- An eigenvector $\underline{e}_i$ of A satisfies, by definition,

  $$A\underline{e}_i = \lambda_i \underline{e}_i$$

  where $\underline{e}_i$ and λ_i may be complex
- In other words, a state-space model is diagonalized by a similarity transformation matrix $\mathbf{E}$ whose columns are given by the eigenvectors of A:

  $$\mathbf{E} = [\underline{e}_1 \ \cdots \ \underline{e}_N]$$

- A system can be diagonalized whenever the eigenvectors of A are *linearly independent*.
  - This always holds for *distinct* poles
  - May or may not hold for *repeated* poles

State Space Diagonalization

- Suppose we solve the equation $A\underline{e}_i = \lambda_i \underline{e}_i$ and find N linearly independent eigenvectors of A
- Form the $N \times N$ matrix $\mathbf{E} = [\underline{e}_1 \ldots \underline{e}_N]$ having these eigenvectors as columns.
- Since the eigenvectors are linearly independent, $\mathbf{E}$ is *full rank* and can be *inverted*. This means it is *one-to-one* and qualifies as a *linear coordinate transformation matrix*.
- As derived above, the *transformed* state transition matrix is given by

$$\tilde{A} = \mathbf{E}^{-1} A \mathbf{E}$$

- Since $A\underline{e}_i = \lambda_i \underline{e}_i$, we have

$$A\mathbf{E} = \mathbf{E}\Lambda$$

  where Λ is a diagonal matrix having the (complex) eigenvalues of A along its diagonal.

- It follows that

$$\tilde{A} = \mathbf{E}^{-1} A \mathbf{E} = \mathbf{E}^{-1}\mathbf{E}\Lambda = \Lambda.$$

  Thus, the new state transition matrix Λ is *diagonal* consisting of the eigenvalues of A.

- The transfer function of the diagonalized system is

$$\begin{aligned}\mathbf{H}(z) &= \tilde{D} + \tilde{C}\,(zI - \Lambda)^{-1}\,\tilde{B} \\ &= \tilde{D} + \frac{\tilde{c}_1 b_1 z^{-1}}{1 - \lambda_1 z^{-1}} + \frac{\tilde{c}_2 \tilde{b}_2 z^{-1}}{1 - \lambda_2 z^{-1}} + \cdots + \frac{\tilde{c}_N \tilde{b}_N z^{-1}}{1 - \lambda_N z^{-1}} \\ &= \tilde{D} + \sum_{i=1}^{N} \frac{\tilde{c}_i \tilde{b}_i z^{-1}}{1 - \lambda_i z^{-1}}\end{aligned}$$

  We see again that the diagonalized system (modal representation) consists of N *parallel one-pole systems*.

- Dynamic modes λ_i are *decoupled*
- Closely related to *partial-fraction expansion* of $\mathbf{H}(z)$:
  - *Residue* of the ith pole is $c_i b_i$
  - Complex-conjugate poles may be combined to form real second-order sections

Finding the Eigenvalues of A in Practice

Small problems may be solved by hand by solving the system of equations

$$A\mathbf{E} = \mathbf{E}\Lambda$$

The Matlab built-in function `eig()` may be used to find the eigenvalues of A (system poles).

Example of State-Space Diagonalization

For the previous example

$$A \triangleq \begin{bmatrix} -\frac{1}{2} & -\frac{1}{3} \\ 1 & 0 \end{bmatrix} \qquad B \triangleq \begin{bmatrix} 1 \\ 0 \end{bmatrix} \qquad C \triangleq \begin{bmatrix} 3/2 \\ 8/3 \end{bmatrix} \qquad D \triangleq 1$$

we obtain the following in Matlab:

```
>> eig(A) % eigenvalues of state transition matrix

ans =
  -0.2500 + 0.5204i
  -0.2500 - 0.5204i

>> roots(den) % poles of transfer function \Hmtx(z)

ans =
  -0.2500 + 0.5204i
  -0.2500 - 0.5204i

% They are the same, as they must be.
>> abs(roots(den)) % check stability

ans =
    0.5774
    0.5774
```

The system is stable.

Complex-conjugate poles are typically combined to produce real, second-order (2×2) parallel sections in the modal representation. Thus, our second-order example is already in *real* modal form. However, to illustrate the computations, let's obtain the eigenvectors and compute the *complex* modal representation:

```
>> % Initial state space model from example above:
>> A = [-1/2, -1/3; 1, 0];
>> B = [1; 0];
>> C = [2-1/2, 3-1/3];
>> D = 1;

>> % Diagonalizing similarity transformation:
>> [E,L] = eig(A)  % [Evects, Evals] = eig(A)

E =

  -0.4507 - 0.2165i  -0.4507 + 0.2165i
        0 + 0.8660i        0 - 0.8660i



L =

  -0.2500 + 0.5204i        0
        0            -0.2500 - 0.5204i

>> A * E - E * L   % should be zero

ans =
```

```
  1.0e-016 *
        0 + 0.2776i             0 - 0.2776i
        0                       0
```

Now form the complete diagonalized state-space model (complex):

```
>> Ei = inv(E); % matrix inverse
>> Ab = Ei*A*E  % diagonalized state xition mtx

Ab =
  -0.2500 + 0.5204i   0.0000 + 0.0000i
  -0.0000            -0.2500 - 0.5204i

>> Bb = Ei*B   % new input "routing vector"

Bb =
   -1.1094
   -1.1094

>> Cb = C*E    % new output linear combination

Cb =
  -0.6760 + 1.9846i  -0.6760 - 1.9846i
```

```
>> Db = D       % feed-through term unchanged

Db =
     1
```

Verify that we still have the same transfer function:

```
>> [numb,denb] = ss2tf(Ab,Bb,Cb,Db)

numb =
   1      2 + 0i   3 + 0i

denb =
   1    0.5 - 0i   0.3333

>> num = [1, 2, 3]; % original numerator
>> norm(num-numb)

ans = 1.5543e-015

>> den = [1, 1/2, 1/3]; % original denominator
>> norm(den-denb)

ans = 1.3597e-016
```

Close enough.

Properties of the Modal Representation

- The modal representation is not *unique* since B and C may be scaled in compensating ways to produce the same transfer function. Also, the diagonal elements of A may be permuted.
- For oscillatory systems, the λ_i are *complex*.
- If mode i is oscillatory and *undamped* (lossless), the state variable $x_i(n)$ oscillates *sinusoidally* at some frequency ω_i, where

$$\lambda_i = e^{j\omega_i T}$$

- In the damped oscillatory case, we have

$$\lambda_i = R_i e^{j\omega_i T}$$

  where R_i is the pole (eigenvalue) radius. For stability, we must have $|R_i| < 1$.
- In practice, we often prefer to combine complex-conjugate pole-pairs to form a real, "block-diagonal" system in which A has two-by-two real matrices along its diagonal.
- Matlab function `cdf2rdf()` can be used to convert complex diagonal form to real block-diagonal form.

- The input vector $\tilde{B}$ in the modal representation specifies *how the modes are excited* by the input signal $u(n)$:

$$x_i(n) = \tilde{b}_i u(n)$$

- The output vector $\tilde{C}$ in the modal representation specifies *how the modes are mixed* in the output signal $y(n)$:

$$y(n) = \tilde{C}\underline{\tilde{x}}(n) = \tilde{c}_1\tilde{x}_1(n) + \tilde{c}_2\tilde{x}_2(n) + \cdots + \tilde{c}_N\tilde{x}_N(n)$$

Repeated Poles

For repeated poles λ_i. we have two cases:

- If the corresponding eigenvectors are *linearly independent*, the modes are independent and can be decoupled (system can be diagonalized)
- Otherwise, if λ_i corresponds to k linearly *dependent* eigenvectors, the diagonalized system will contain a *Jordan block* of order k corresponding to that mode.
- Same as repeated roots in a partial-fraction expansion
- Impulse response looks like $n\lambda^n$, $n^2\lambda^n$, etc.