Computation of Large Scale Transfer Function Dominant Poles

Joost Rommes
Utrecht University
rommes@math.uu.nl
http://www.math.uu.nl/people/rommes

Introduction

The behavior of a large scale dynamical system can often be described by a relatively small number of its dominant modes. A reduced order model, called the modal equivalent, can be obtained by projecting the state space on the subspace spanned by the dominant modes. Modal approximation has been successfully applied to transfer functions of large scale power systems and electrical circuits, with applications such as stability analysis and controller design.

The dominant modes, and the corresponding dominant poles of the system transfer function, are specific eigenvectors and eigenvalues of the state matrix. Because the systems are very large in practice ($n > 10000$ variables), it is not feasible to compute all modes. Instead, specialized eigenvalue methods are required that compute the most dominant poles and corresponding modes. In this work the subspace accelerated dominant pole algorithm (SADPA) [3, 2] is presented, that computes the dominant poles of a large scale transfer function in an efficient and effective way.

Dominant poles

The transfer function of a linear dynamical system

$$\begin{cases} \dot{\mathbf{x}}(t) = A\mathbf{x}(t) + \mathbf{b}u(t) \\ y(t) = \mathbf{c}^*\mathbf{x}(t), \end{cases}$$

where $A \in \mathbb{R}^{n\times n}$, $\mathbf{x}(t), \mathbf{b}, \mathbf{c} \in \mathbb{R}^n$ and $u(t), y(t) \in \mathbb{R}$, is defined as

$$H(s) = \mathbf{c}^*(sI - A)^{-1}\mathbf{b},$$

where $I \in \mathbb{R}^{n\times n}$ is the identity matrix and $s \in \mathbb{C}$.

The eigenvalues $\lambda_i \in \mathbb{C}$ of A are the poles of $H(s)$. An eigentriplet $(\lambda_i, \mathbf{x}_i, \mathbf{y}_i)$ is composed of an eigenvalue λ_i of A and the corresponding right and left eigenvectors $\mathbf{x}_i, \mathbf{y}_i \in \mathbb{C}^n$. The transfer function $H(s)$ can be expressed as a sum of residues R_i over first order poles:

$$H(s) = \sum_{i=1}^{n} \frac{R_i}{s - \lambda_i},$$

where the residues R_i are

$$R_i = (\mathbf{c}^*\mathbf{x}_i)(\mathbf{y}_i^*\mathbf{b}).$$

A pole λ_i of $H(s)$ with right and left eigenvectors $\mathbf{x}_i$ and $\mathbf{y}_i$ ($\mathbf{y}_i^*\mathbf{x}_i = 1$) is called the *dominant* pole if $\widehat{R}_i = |R_i|/|\mathrm{Re}(\lambda_i)| > \widehat{R}_j$, $j \neq i$. In the Bode magnitude plot of $H(s)$, peaks occur at frequencies close to the imaginary parts of the dominant poles of $H(s)$ (see figure 1).

A *modal equivalent* $H_k(s)$ is an approximation of a transfer function $H(s)$ that consists of $k < n$ terms:

$$H_k(s) = \sum_{j=1}^{k} \frac{R_j}{s - \lambda_j} + d.$$

If X and Y are $n \times k$ matrices having dominant left and right eigenvectors $\mathbf{y}_i$ and $\mathbf{x}_i$ of A as columns (with $Y^*AX = \Lambda = \mathrm{diag}(\lambda_1, \ldots, \lambda_k)$ and $Y^*X = I$), then the corresponding (complex) reduced system follows by setting $\mathbf{x} = X\tilde{\mathbf{x}}$:

$$\begin{cases} \dot{\tilde{\mathbf{x}}}(t) = \Lambda\tilde{\mathbf{x}}(t) + (Y^*\mathbf{b})u(t) \\ \tilde{\mathbf{y}}(t) = (\mathbf{c}^*X)\tilde{\mathbf{x}}(t) + du(t). \end{cases}$$

Because the order of the reduced system is much smaller than the order of the original system ($k \ll n$), the reduced system can efficiently be used for simulation and controller design.

Computing dominant poles

The poles of $H(s)$ are the $\lambda \in \mathbb{C}$ for which $\lim_{s\to\lambda} |H(s)| = \infty$, or, with $G(s) = 1/H(s)$, $\lim_{s\to\lambda} G(s) = 0$. In other words, the poles are the roots of $G(s)$, and these roots can be computed with Newton's method. This leads to the Dominant Pole Algorithm [1]:

> **while** $(s, \mathbf{v}, \mathbf{w})$ not converged **do**
> Solve $\mathbf{v}$ from $(sI - A)\mathbf{v} = \mathbf{b}$
> Solve $\mathbf{w}$ from $(sI - A)^*\mathbf{w} = \mathbf{c}$
> Set new estimate $s = s - (\mathbf{c}^*\mathbf{v})/(\mathbf{w}^*\mathbf{v})$
> **end while**

The Subspace Accelerated Dominant Pole Algorithm (SADPA) is a generalization of DPA to compute more than one dominant pole and has better convergence.

Algorithm 1 Subspace Accelerated DPA

INPUT: System $(A, \mathbf{b}, \mathbf{c})$, estimate s_1
OUTPUT: Dominant pole triplets $(\lambda_i, \mathbf{x}_i, \mathbf{y}_i)$

```
k = 1, p_found = 0, Λ = [ ]^{1×0}, X = Y[ ]^{n×0}
while p_found < #wanted poles do
    Solve v from (s_k I − A)v = b
    Solve w from (s_k I − A)^* w = c
    V = Expand(V, v)
    W = Expand(W, w)
    Select dominant approx. triplet (λ̂_1, x̂_1, ŷ_1)
    if ||A x̂_1 − λ̂_1 x̂_1||_2 < ε then
        Λ = [Λ, λ_1], X = [X, x̂_1], Y = [Y, ŷ_1]
        Deflate x̂_1 and ŷ_1 from b, c, V, W
        p_found = p_found + 1
        Set λ̂_1 = λ̂_2
    end if
    Set the new pole estimate s_{k+1} = λ̂_1
    Set k = k + 1
end while
```

The three additional ingredients of SADPA are:

1. Subspace acceleration: Keep the intermediate left and right eigenvector approximations $\mathbf{v}$ and $\mathbf{w}$ in orthogonal search spaces V and W.
2. Selection strategy: Every iteration, the $k \ll n$ eigentriplets $(\hat{\lambda}_i, \tilde{\mathbf{x}}_i, \tilde{\mathbf{y}}_i)$ of the pencil (W^*AV, W^*V) are used to form approximate triplets $(\hat{\lambda}_i, \hat{\mathbf{x}}_i = V\tilde{\mathbf{x}}_i, \hat{\mathbf{y}}_i = W\tilde{\mathbf{y}}_i)$ of A. Select the triplet with largest $|(\mathbf{c}^*\hat{\mathbf{x}}_i)(\hat{\mathbf{y}}_i^*\mathbf{b})|/|\mathrm{Re}(\hat{\lambda}_i)|$.
3. Deflation: If a dominant pole triplet has converged, deflate via $\mathbf{b}_d = (I - \mathbf{x}\mathbf{y}^*)\mathbf{b}$ and $\mathbf{c}_d = (I - \mathbf{y}\mathbf{x}^*)\mathbf{c}$. This avoids convergence to already found poles, since the corresponding residues are transformed to $R = (\mathbf{c}_d^*\mathbf{x})(\mathbf{y}^*\mathbf{b}_d) = 0$.

Example

The Brazilian Interconnected Power System (BIPS) is a year 1999 planning model. Fig. 1 shows the frequency response of the complete ($n = 13,251$) and the reduced model ($k = 41$) computed by SADPA, and the error. Both the magnitude and the phase plot are good matches of the exact transfer function.

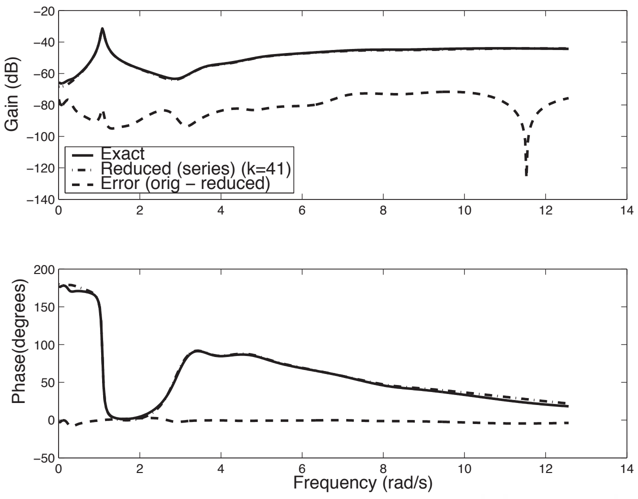

Figure 1: Bode plot of 41st order modal equivalent, complete model and error for BIPS.

References

[1] MARTINS, N., LIMA, L. T. G., AND PINTO, H. J. C. P. Computing dominant poles of power system transfer functions. *IEEE TPWRS 11*, 1 (Feb 1996), 162–170.

[2] ROMMES, J., AND MARTINS, N. Efficient computation of multivariable transfer function dominant poles using subspace acceleration. *IEEE TPWRS 21*, 4 (Nov 2006), 1471–1483.

[3] ROMMES, J., AND MARTINS, N. Efficient computation of transfer function dominant poles using subspace acceleration. *IEEE TPWRS 21*, 3 (Aug. 2006), 1218–1226.

Part of this research has been funded by the Dutch BSIK/BRICKS project MSV1.