

Chapter 5

Efficient computation of large-scale transfer function dominant zeros

Abstract. This chapter describes efficient algorithms for the computation of dominant zeros of large-scale transfer functions. The transfer function zeros are demonstrated to be equal to the poles of a new inverse system, which is valid even for the strictly proper case. This is a new finding, which is important from practical as well as theoretical viewpoints. Hence, the dominant zeros can be computed as the dominant poles of the inverse transfer function by the recent efficient SADPA and SAMDP algorithms. The importance of computing dominant zeros and the performance of the algorithms are illustrated by examples from practical power system models.

Key words. transfer function zeros, small-signal stability, poorly-damped oscillations, power system dynamics, transfer function, transfer function residues, pole-zero map, model reduction, large-scale systems, sparse eigenanalysis, modal equivalents, modal analysis, multivariable systems, inverse systems.

5.1 Introduction

The zeros of a transfer function are important in several applications such as the analysis and design of multivariable feedback control systems. The coordinated design of power system stabilizers, for damping electromechanical oscillations in large interconnected power systems, involves numerous decentralized control loops and greatly benefits from the knowledge of scalar as well as multivariable zeros.

There are several methods for the computation of all zeros of a transfer function, see [135, 100, 164] and references therein. However, all these methods are only applicable to moderately sized systems, while in practice one is often interested only in the most dominant zeros of large-scale transfer functions.

The SADPA (SAMDP) algorithms for the computation of dominant poles of (multivariable) transfer functions that have been developed in [123, 122] (see Chapters 3 and 4, respectively), can be used, when adapted, for several other applications. This chapter is concerned with variants that can be used to compute the dominant zeros of transfer functions of large-scale dynamical systems. In fact, it will be shown that the SADPA (SAMDP) algorithms can be used without adaptation, by using the relationship between the zeros of the transfer function and the poles of the corresponding inverse transfer function.

The outline of the chapter is as follows. Definitions of poles and zeros of general transfer functions are given in Section 5.2, as well as the motivation for finding dominant zeros of scalar and multivariable transfer functions. In Section 5.3, algorithms will be described to compute a dominant zero of a large-scale SISO or MIMO transfer function. It will be shown that the zeros of a transfer function are equal to the poles of the inverse transfer function. Hence, the SADPA [123] and SAMDP [122] algorithms can be used to compute the dominant zeros via the dominant poles of the inverse transfer function, as will be described in Section 5.4. All algorithms are illustrated by large-scale examples and applications from practical power system models in Section 5.5. Section 5.6 concludes.

5.2 Transfer functions, poles, and zeros

Throughout this chapter, the dynamical systems (E, A, B, C, D) are of the form

$$\begin{cases} E\dot{\mathbf{x}}(t) &= A\mathbf{x}(t) + B\mathbf{u}(t) \\ \mathbf{y}(t) &= C^*\mathbf{x}(t) + D\mathbf{u}(t), \end{cases} \quad (5.2.1)$$

where $A, E \in \mathbb{R}^{n \times n}$, E singular, $B \in \mathbb{R}^{n \times m}$, $C \in \mathbb{R}^{n \times p}$, $\mathbf{x}(t) \in \mathbb{R}^n$, $\mathbf{u}(t) \in \mathbb{R}^m$, $\mathbf{y}(t) \in \mathbb{R}^p$ and $D \in \mathbb{R}^{p \times m}$. The transfer function $H(s) : \mathbb{C} \rightarrow \mathbb{C}^{p \times m}$ is defined as

$$H(s) = C^*(sE - A)^{-1}B + D, \quad (5.2.2)$$

with $s \in \mathbb{C}$. If $m = p = 1$, the system (5.2.1) is called a single-input single-output (SISO) system, and $\mathbf{b} = B, \mathbf{c} = C \in \mathbb{R}^n$ are vectors and $d = D \in \mathbb{R}$ is a scalar. Otherwise it is a multi-input multi-output (MIMO) system.

The eigenvalues $\lambda_i \in \mathbb{C}$ of the matrix pencil (A, E) are the poles of transfer function (5.2.2). An eigentriplet $(\lambda_i, \mathbf{x}_i, \mathbf{y}_i)$ is composed of an eigenvalue λ_i of (A, E) and the corresponding right and left eigenvectors $\mathbf{x}_i, \mathbf{y}_i \in \mathbb{C}^n$:

$$\begin{aligned} A\mathbf{x}_i &= \lambda_i E\mathbf{x}_i, & \mathbf{x}_i &\neq 0, \\ \mathbf{y}_i^* A &= \lambda_i \mathbf{y}_i^* E, & \mathbf{y}_i &\neq 0. \end{aligned}$$

Assuming that all eigenvalues are distinct, the right and left eigenvectors corresponding to finite eigenvalues can be scaled so that $\mathbf{y}_i^* E\mathbf{x}_i = 1$. Furthermore, it can be shown that left and right eigenvectors corresponding to distinct eigenvalues

are E -orthogonal: $\mathbf{y}_i^* E \mathbf{x}_j = 0$ for $i \neq j$. The transfer function $H(s)$ can be expressed as a sum of residue matrices $R_i \in \mathbb{C}^{p \times m}$ over the $\tilde{n} \leq n$ finite first-order poles [78]:

$$H(s) = \sum_{i=1}^{\tilde{n}} \frac{R_i}{s - \lambda_i} + R_\infty + D,$$

where the residues R_i are

$$R_i = (C^* \mathbf{x}_i)(\mathbf{y}_i^* B),$$

and R_∞ is the constant contribution of the poles at infinity (often zero).

5.2.1 Dominant poles

Although there are different indices of modal dominance [4, 68, 159], the following will be used in this chapter.

Definition 5.2.1. *A pole λ_i of $H(s)$ with corresponding right and left eigenvectors \mathbf{x}_i and \mathbf{y}_i ($\mathbf{y}_i^* E \mathbf{x}_i = 1$) is called the dominant pole if $\hat{R}_i = \|R_i\|_2 / |\operatorname{Re}(\lambda_i)| = \sigma_{\max}(R_i) / |\operatorname{Re}(\lambda_i)| > \hat{R}_j$ for all $j \neq i$.*

A pole λ_j that corresponds to a residue R_j with large $\sigma_{\max}(R_j) / |\operatorname{Re}(\lambda_j)|$ is called a dominant pole, i.e. a pole that is well observable and controllable in the transfer function. This can also be observed from the corresponding σ_{\max} -plot [89, 122, 17] of $H(s)$, where peaks occur at frequencies close to the imaginary parts of the dominant poles of $H(s)$ (in the SISO case this plot is called the Bode plot, see also Fig. 5.1). An approximation of $H(s)$ that consists of $k \ll n$ terms with $\|R_j\|_2 / |\operatorname{Re}(\lambda_j)|$ above some value, determines the effective transfer function behavior [144] and will be referred to as transfer function modal equivalent:

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

Because a residue matrix R_j is the product of a column vector and a row vector, it is of unit rank. Therefore at least $\min(m, p)$ different poles (and corresponding residue matrices) are needed to obtain a modal equivalent with nonzero $\sigma_{\min}(\omega)$ plot [93, 114].

5.2.2 Dominant zeros

There are many different approaches to the definition of multivariable system zeros, see for instance [135] for an extensive overview. In the SISO case, $z_0 \in \mathbb{C}$ is called a transmission zero if $H(z_0) = \mathbf{c}^*(z_0 E - A)^{-1} \mathbf{b} + d = 0$. More generally, there is a distinction between transmission zeros and invariant zeros. This chapter will focus on the computation of transmission zeros.

The Rosenbrock system matrix [127] corresponding to (5.2.1) is

$$\Sigma = \begin{bmatrix} sE - A & B \\ -C^* & D \end{bmatrix}, \tag{5.2.3}$$

and is of importance, amongst others, because it can be transformed by basic operations to the (block Schur) form

$$\begin{bmatrix} I & 0 \\ 0 & H(s) \end{bmatrix}, \quad (5.2.4)$$

from which it can be seen that $\text{rank}(\Sigma) = n + \text{rank}(H(s))$. The definition of a transmission zero is given in Def. 5.2.2.

Definition 5.2.2. *A number $z_0 \in \mathbb{C}$ is called a transmission zero if it satisfies*

$$\text{rank}(H(z_0)) < \max_s \text{rank}(H(s)).$$

Note that with this definition it is in the MIMO case not sufficient (and not necessary) to have one entry of $H(z_0)$ equal to zero, and not necessary to have $H(z_0) = 0$. The transmission zeros are a subset of the invariant zeros of a system, that are defined as follows.

Definition 5.2.3. *A number $z_0 \in \mathbb{C}$ is called an invariant zero if it satisfies*

$$\text{rank} \begin{bmatrix} z_0 E - A & B \\ -C^* & D \end{bmatrix} < \text{rank}(\Sigma).$$

If the system (5.2.1) is minimal, the transmission zeros and invariant zeros coincide, and are simply called zeros.

If Σ , and hence $H(s)$ by (5.2.4), have full column rank ($m \leq p$), the vectors \mathbf{x} and \mathbf{u} corresponding to z_0 that satisfy

$$\begin{bmatrix} z_0 E - A & -B \\ C^* & D \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ \mathbf{u} \end{bmatrix} = 0, \quad \mathbf{x} \in \mathbb{C}^n, \mathbf{u} \in \mathbb{C}^m,$$

are called the *right zero state* and *state zero input*, respectively [135]. Dually, if Σ has full row rank ($m \geq p$), the vectors \mathbf{y} and \mathbf{v} corresponding to z_0 that satisfy

$$[\mathbf{y}^* \quad \mathbf{v}^*] \begin{bmatrix} z_0 E - A & -B \\ C^* & D \end{bmatrix} = 0, \quad \mathbf{y} \in \mathbb{C}^n, \mathbf{v} \in \mathbb{C}^p,$$

are called the *left zero state* and *state zero input*, respectively [135].

The problem of finding the invariant zeros of a square system can also be formulated as the generalized eigenvalue problem

$$A_z \mathbf{x}_z = \lambda E_z \mathbf{x}_z, \quad \mathbf{x}_z \in \mathbb{C}^{n+m}, \quad (5.2.5)$$

where

$$A_z = \begin{bmatrix} A & B \\ -C^* & -D \end{bmatrix}, \quad E_z = \begin{bmatrix} E & 0 \\ 0 & 0 \end{bmatrix}, \quad \text{and } \mathbf{x}_z = \begin{bmatrix} \mathbf{x} \\ \mathbf{u} \end{bmatrix}.$$

There are also definitions of input decoupling zeros and output decoupling zeros, and system zeros [135], but these are beyond the scope of this chapter.

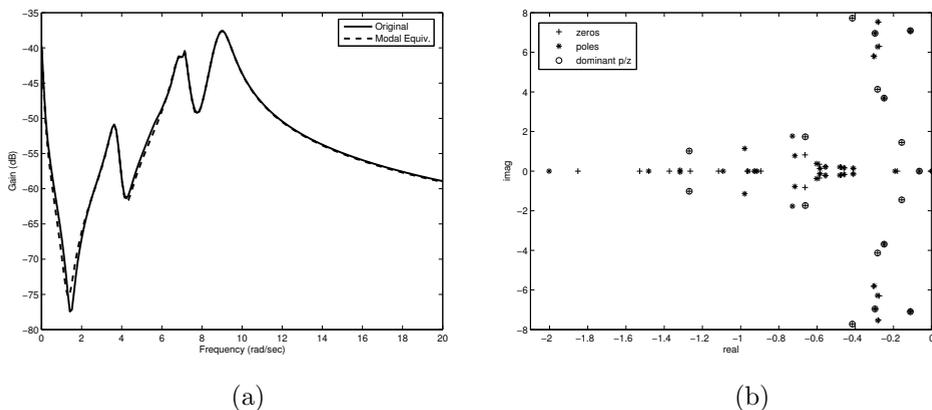


Figure 5.1: (a) Bode plot of transfer function $W_{36}/Pmec_{36}$ of the New England test system (66 states, see also [91]), together with a 11th order modal equivalent. (b) Part of corresponding pole (*) - zero (+) plot. The dominant zeros (poles), marked by encircled +s (*-s), cause the dips (peaks) in the Bode plot.

Whereas a dominant pole causes a peak in a Bode plot of the transfer function, a dominant zero cause a dip, see also Fig. 5.1. In other words, the dominant zeros are the zeros close to the imaginary axis, that ‘stand out’ in the pole-zero plot, see also Fig. 5.1. In Section 5.3.3, it will be shown that the zeros are the poles of the inverse system, if it exists. Then, a zero is called dominant if it is a dominant pole of the inverse system (by Definition 5.2.1). It must be stressed that in different applications different definitions of the dominance of a zero can be used. In stabilization studies, for example, the rightmost and lowest damped zeros are the dominant zeros.

5.2.3 Practical need for the computation of dominant transfer function zeros

Theoretical considerations

Consider the block diagram of a typical control system depicted in Fig. 5.2. Let $n_G(s)$, $d_G(s)$, $n_K(s)$ and $d_K(s)$ be polynomials of s defining scalar proper rational functions

$$G(s) = \frac{n_G(s)}{d_G(s)} \quad \text{and} \quad K(s) = \frac{n_K(s)}{d_K(s)},$$

where $G(s)$ represents the system dynamics to be controlled and $K(s)$ is the designed controller. It is easily verified that the closed-loop transfer function $H(s)$ is

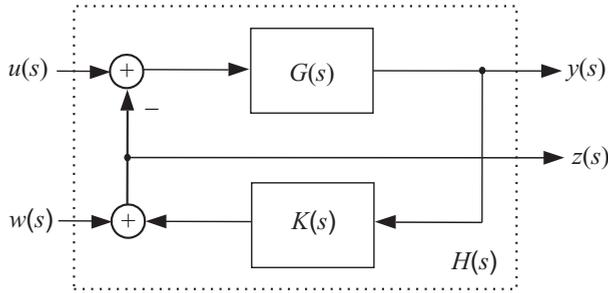


Figure 5.2: Closed-loop system.

a (2×2) transfer matrix:

$$\begin{bmatrix} y(s) \\ z(s) \end{bmatrix} = \underbrace{\begin{bmatrix} H_{yu}(s) & H_{yw}(s) \\ H_{zu}(s) & H_{zw}(s) \end{bmatrix}}_{H(s)} \begin{bmatrix} u(s) \\ w(s) \end{bmatrix},$$

with

$$\begin{aligned} H &= \begin{bmatrix} G & -G \\ GK & 1 \end{bmatrix} \frac{1}{1+GK} \\ &= \begin{bmatrix} n_G d_K & -n_G d_K \\ n_G n_K & d_G d_K \end{bmatrix} \frac{1}{d_G d_K + n_G n_K}. \end{aligned} \quad (5.2.6)$$

The focus here is on the main diagonal elements of the numerator matrix of H shown in (5.2.6). While the transfer function $H_{yu}(s)$ represents the actual closed-loop control channel, the input-output pair (w, z) defines a synthetic disturbance channel $H_{zw}(s)$ used here only for analysis purposes. It is clear from (5.2.6) that the set of zeros of H_{yu} and H_{zw} can be split in two subsets:

$$\begin{aligned} \{\text{zeros of } H_{yu}\} &= \{\text{zeros of } G\} \cup \{\text{poles of } K\}, \\ \{\text{zeros of } H_{zw}\} &= \{\text{poles of } G\} \cup \{\text{poles of } K\}. \end{aligned} \quad (5.2.7)$$

As the controller K is determined by the designer and is, in general, of small order, the subset of closed-loop zeros corresponding to the poles of K are already known and not relevant to the current analysis. The remaining closed-loop zeros are either open-loop zeros (for the control channel (H_{yu})) or open-loop poles (for the disturbance channel (H_{zw})) of possibly large-scale transfer functions.

The case where the signals u , y , w and z are vectors of compatible dimensions is also of interest. In this case, the MIMO closed-loop transfer functions for the control and disturbance channels are, respectively:

$$\begin{aligned} H_{yu} &= (I + GK)^{-1}G = G(I + KG)^{-1}, \text{ and} \\ H_{zw} &= (I + KG)^{-1}. \end{aligned} \quad (5.2.8)$$

Considering a right and a left coprime factorization for G and K , respectively, as

$$G = N_G D_G^{-1} \text{ and } K = D_K^{-1} N_K,$$

(N_G, D_G, N_K, D_K polynomial matrices, with D_G, D_K nonsingular) and substituting them in (5.2.8) yields

$$\begin{aligned} H_{zw} &= (I + D_K^{-1} N_K N_G D_G^{-1})^{-1} \\ &= D_G (D_K D_G + N_K N_G)^{-1} D_K \end{aligned} \quad (5.2.9)$$

that pre-multiplied by $N_G D_G^{-1}$ becomes

$$H_{yu} = N_G (D_K D_G + N_K N_G)^{-1} D_K. \quad (5.2.10)$$

Since D_G and N_G are numerators in (5.2.9) and (5.2.10), respectively, one can draw similar conclusions regarding closed-loop zeros for MIMO systems (even for non-square G and K).

Practical applications

- a. The first practical application comes from the fact that the closed-loop transfer functions H_{zw} and H_{yu} have some zeros on the two extreme points of relevant root locus branches. In other words, some branches of interest start at the zeros of H_{zw} (null loop gain) and end at the zeros of H_{yu} (infinite loop gain). Hence, an efficient algorithm for the computation of dominant transfer function zeros, applied to adequately chosen closed-loop channels, is useful to the stability analysis of large-scale control systems.
- b. There are at least two other interesting uses of the property (5.2.7) in the dominant zeros context. Some important features of the dominant open- and closed-loop poles can be caught from a single frequency response diagram, making the control analysis easier. For instance, given a stabilizing controller K , the Bode plot of the synthetic disturbance channel H_{zw} provides good estimates for the frequency and damping ratio of the open- (dips) and closed-loop (peaks) poles.
- c. Dominant transmission zeros properties are applied to identify the best candidate locations for additional stabilizers to most effectively damp electromechanical modes in large power systems, as described in the closure of [92] and in [24].
- d. Dominant zeros of driving point impedances are computed and effectively shifted, by use of eigenvalue sensitivity coefficients, in the harmonic distortion analysis of industrial power systems [160].

5.3 Computing transfer function zeros

The algorithms for the computation of zeros in this chapter are in fact Newton schemes and have many similarities with the dominant pole algorithms DPA [91],

SADPA [123] and SAMDP [122]. Algorithms that compute a single zero are shown in Section 5.3.1 for the SISO case and in Section 5.3.2 for the MIMO case. In Section 5.3.3 it is shown that the zeros are poles of the inverse transfer function. Consequently, the SADPA and SAMDP algorithms can be used to compute the dominant zeros via the poles of the inverse transfer function, as will be described in Section 5.4.

Earlier work in this direction was done in [92]. Full space methods based on the QZ algorithm [101, 64] to compute *all* zeros of a system are described in [44, 100]. These methods are of complexity $O((n + \max(m, p))^3)$ and hence only applicable to moderately sized systems. The methods described in this chapter are intended for very large systems, where one is specifically interested in the dominant zeros.

5.3.1 SISO Dominant Zero Algorithm (DZA)

Newton's method can be used to compute the zeros $z_i \in \mathbb{C}$ of the objective function

$$f : \mathbb{C} \longrightarrow \mathbb{C} : s \longmapsto \mathbf{c}^*(sE - A)^{-1}\mathbf{b} + d. \quad (5.3.1)$$

The derivative of the inverse of a square matrix valued function $Y(s)$ is given by

$$\frac{d[Y^{-1}(s)]}{ds} = -Y^{-1}(s)Y'(s)Y^{-1}(s),$$

and it follows that the derivative of the objective function (5.3.1) is

$$f'(s) = -\mathbf{c}^*(sE - A)^{-1}E(sE - A)^{-1}\mathbf{b}. \quad (5.3.2)$$

Using (5.3.1) and (5.3.2), the Newton scheme for the computation of a zero becomes

$$\begin{aligned} s_{k+1} &= s_k - \frac{f(s_k)}{f'(s_k)} \\ &= s_k + \frac{\mathbf{c}^*(s_k E - A)^{-1}\mathbf{b} + d}{\mathbf{c}^*(s_k E - A)^{-1}E(s_k E - A)^{-1}\mathbf{b}} \\ &= s_k + \frac{\mathbf{c}^*\mathbf{v} + d}{\mathbf{w}^*E\mathbf{v}}, \end{aligned}$$

where

$$\mathbf{v} = (s_k E - A)^{-1}\mathbf{b}, \text{ and } \mathbf{w} = (s_k E - A)^{-*}\mathbf{c}.$$

An implementation of this scheme is shown in Alg. 5.1. In the neighborhood of a solution the algorithm converges quadratically. A different, more difficult formulation that works with Σ (5.2.3) can be found in [92].

The DZA scheme only differs a minus sign from the Newton scheme for finding dominant poles (DPA, see [91, 20, 21]). For DPA, the objective function is $1/f(s)$

Algorithm 5.1 Dominant Zero Algorithm (DZA)

INPUT: System $(E, A, \mathbf{b}, \mathbf{c}, d)$, initial zero estimate $s_1 \in \mathbb{C}$, tolerance $\epsilon \ll 1$

OUTPUT: Approximate zero λ and corresponding right and left zero states \mathbf{x} and \mathbf{y}

- 1: Set $k = 1$
- 2: **while** not converged **do**
- 3: Solve $\mathbf{v} \in \mathbb{C}^n$ from $(s_k E - A)\mathbf{v} = \mathbf{b}$
- 4: Solve $\mathbf{w} \in \mathbb{C}^n$ from $(s_k E - A)^*\mathbf{w} = \mathbf{c}$
- 5: Compute the new zero estimate

$$s_{k+1} = s_k + \frac{\mathbf{c}^*\mathbf{v} + d}{\mathbf{w}^*E\mathbf{v}}$$

- 6: The zero $\lambda = s_{k+1}$ with $\mathbf{x} = \mathbf{v}$ and $\mathbf{y} = \mathbf{w}$ has converged if $\|\mathbf{r}_k\|_2 < \epsilon$, where

$$\mathbf{r}_k = \begin{bmatrix} s_{k+1}E - A & -\mathbf{b} \\ \mathbf{c}^* & d \end{bmatrix} \begin{bmatrix} \mathbf{v} \\ 1 \end{bmatrix}$$

- 7: Set $k = k + 1$
- 8: **end while**

and the scheme becomes (see also [123])

$$\begin{aligned} s_{k+1}^{dpa} &= s_k^{dpa} - \frac{\mathbf{c}^*(s_k^{dpa} E - A)^{-1}\mathbf{b} + d}{\mathbf{c}^*(s_k^{dpa} E - A)^{-1}E(s_k^{dpa} E - A)^{-1}\mathbf{b}} \\ &= \frac{\mathbf{w}^*A\mathbf{v}}{\mathbf{w}^*E\mathbf{v}} \\ &\equiv \rho(A, E, \mathbf{v}, \mathbf{w}), \end{aligned}$$

where $\rho(A, E, \mathbf{v}, \mathbf{w})$ is called the two-sided Rayleigh quotient [110]. Similarly, it follows that in DZA, the new shift s_{k+1} can be written as a two-sided Rayleigh

quotient $\rho(A_z, E_z, \mathbf{v}_z, \mathbf{w}_z)$:

$$\begin{aligned}
 s_{k+1}^{dza} &= s_k^{dza} + \frac{\mathbf{c}^* \mathbf{v} + d}{\mathbf{w}^* E \mathbf{v}} \\
 &= \frac{\mathbf{w}^* (A + (s_k^{dza} E - A)) \mathbf{v}}{\mathbf{w}^* E \mathbf{v}} + \frac{\mathbf{c}^* \mathbf{v} + d}{\mathbf{w}^* E \mathbf{v}} \\
 &= \frac{\mathbf{w}^* (A \mathbf{v} + \mathbf{b}) + \mathbf{c}^* \mathbf{v} + d}{\mathbf{w}^* E \mathbf{v}} \\
 &= \frac{\begin{bmatrix} \mathbf{w}^* & 1 \end{bmatrix} \begin{bmatrix} A & \mathbf{b} \\ \mathbf{c}^* & d \end{bmatrix} \begin{bmatrix} \mathbf{v} \\ 1 \end{bmatrix}}{\begin{bmatrix} \mathbf{w}^* & 1 \end{bmatrix} \begin{bmatrix} E & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \mathbf{v} \\ 1 \end{bmatrix}} \\
 &= \frac{\mathbf{w}_z^* A_z \mathbf{v}_z}{\mathbf{w}_z^* E_z \mathbf{v}_z} \\
 &\equiv \rho(A_z, E_z, \mathbf{v}_z, \mathbf{w}_z),
 \end{aligned}$$

with

$$A_z = \begin{bmatrix} A & \mathbf{b} \\ \mathbf{c}^* & d \end{bmatrix}, \quad E_z = \begin{bmatrix} E & 0 \\ 0 & 0 \end{bmatrix}, \quad \mathbf{v}_z = \begin{bmatrix} \mathbf{v} \\ 1 \end{bmatrix}, \quad \mathbf{w}_z = \begin{bmatrix} \mathbf{w} \\ 1 \end{bmatrix}.$$

Like the poles can also be computed with two-sided Rayleigh quotient iteration (two-sided RQI) [110] applied to (A, E) , the zeros can be computed by applying two-sided RQI to (A_z, E_z) as well. However, as is discussed in [126] (see Chapter 2), DPA (DZA) tends to converge to dominant poles (zeros), while two-sided RQI tends to converge to poles (zeros) closest to the initial shift.

5.3.2 MIMO Dominant Zero Algorithm (MDZA)

The objective function for finding the zeros of a general MIMO transfer function (5.2.2) is

$$f : \mathbb{C} \longrightarrow \mathbb{R} : s \longmapsto \sigma_{\min}(H(s)). \quad (5.3.3)$$

Let $(\sigma_{\min}(s), \mathbf{u}(s), \mathbf{z}(s))$ be the minimal singular triplet of $H(s)$, i.e. $H(s)\mathbf{z}(s) = \sigma_{\min}(s)\mathbf{u}(s)$ and $H^*(s)\mathbf{u}(s) = \sigma_{\min}(s)\mathbf{z}(s)$. It follows that $H^*(s)H(s)\mathbf{z}(s) = \sigma_{\min}^2(s)\mathbf{z}(s)$. It is more convenient to work with the following objective function (equivalent to (5.3.3)):

$$f : \mathbb{C} \longrightarrow \mathbb{R} : s \longmapsto \lambda_{\min}(H^*(s)H(s)), \quad (5.3.4)$$

with $\lambda_{\min} = \sigma_{\min}^2$. Because $f(s)$ in (5.3.4) is a function from $\mathbb{C} \longrightarrow \mathbb{R}$, in general the derivative $df(s)/ds$ is not injective. A complex scalar $z = a + ib \in \mathbb{C}$ can be represented by $[a, b]^T \in \mathbb{R}^2$. The partial derivatives of the objective function (5.3.4) become

$$\begin{aligned}
 \frac{\partial f}{\partial a}(s) &= -\sigma_{\min}(s)(\mathbf{w}^* E \mathbf{v} + (\mathbf{w}^* E \mathbf{v})^*), \\
 \frac{\partial f}{\partial b}(s) &= -i\sigma_{\min}(s)(\mathbf{w}^* E \mathbf{v} - (\mathbf{w}^* E \mathbf{v})^*),
 \end{aligned}$$

where

$$\mathbf{v} = (sE - A)^{-1}B\mathbf{z}, \quad \mathbf{w} = (sE - A)^{-*}C\mathbf{u}.$$

The derivative of (5.3.4) then becomes

$$\nabla f = -2\sigma_{\min}(s)[\operatorname{Re}(\mathbf{w}^*E\mathbf{v}), -\operatorname{Im}(\mathbf{w}^*E\mathbf{v})],$$

where $\operatorname{Re}(a + ib) = a$ and $\operatorname{Im}(a + ib) = b$. The Newton scheme is

$$\begin{aligned} \begin{bmatrix} \operatorname{Re}(s_{k+1}) \\ \operatorname{Im}(s_{k+1}) \end{bmatrix} &= \begin{bmatrix} \operatorname{Re}(s_k) \\ \operatorname{Im}(s_k) \end{bmatrix} - (\nabla f(s_k))^\dagger f(s_k) \\ &= \begin{bmatrix} \operatorname{Re}(s_k) \\ \operatorname{Im}(s_k) \end{bmatrix} \\ &\quad + [\operatorname{Re}(\mathbf{w}^*E\mathbf{v}), -\operatorname{Im}(\mathbf{w}^*E\mathbf{v})]^\dagger \frac{\sigma_{\min}}{2}, \end{aligned}$$

where $Y^\dagger = Y^*(YY^*)^{-1}$ denotes the pseudo-inverse of a matrix $Y \in \mathbb{C}^{n \times m}$ with $\operatorname{rank}(Y) = n$ ($n \leq m$) [64].

This generalization of the Newton method is also known as the normal flow method [163] and can be shown, under certain conditions, to converge (super) linearly. The algorithm for systems with full column rank transfer functions $H(s)$ ($p \geq m$) is shown in Alg. 5.2. If $p = m$, both right and left zero states are returned. If $p > m$, the right zero states are returned. If $p < m$, the left zero states can be computed by calling the algorithm with the dual system $(E^*, A^*, -C, -B, D^*)$.

For SISO systems, the algorithm simplifies to Alg. 5.1. For square MIMO systems, it is advised to follow an approach similar to the MDPA algorithm for square systems in [122]: if the objective function

$$f : \mathbb{C} \longrightarrow \mathbb{C} : s \longmapsto \lambda_{\min}(H(s))$$

is taken, the scheme simplifies to a standard Newton scheme, where no pseudo-inverses are needed and convergence is asymptotically quadratic.

5.3.3 Zeros as the poles of the inverse transfer function

In this section realizations of inverse systems are given. An inverse system is a system whose transfer function is the inverse of the transfer function of the original system. The zeros of the original transfer function become poles of the inverse transfer function. This fact can be used to compute the zeros via the poles of the inverse transfer function, using the SADPA [123] and SAMDP [122] algorithms, as is shown in Section 5.4. In principle, a modal equivalent of the inverse transfer function can be build from the dominant poles and corresponding residues of the inverse system.

SISO transfer functions

It is well known that for SISO systems, the poles of the high gain output feedback closed-loop system approach the zeros of the open-loop system $\Sigma = (E, A, \mathbf{b}, \mathbf{c}, d)$.

Algorithm 5.2 MIMO Transmission Zero Algorithm (MDZA)

INPUT: System (E, A, B, C, D) with $p \geq m$, initial pole estimate s_1 , tolerance $\epsilon \ll 1$

OUTPUT: Approximate zero λ and corresponding right and/or left state zero directions, inputs \mathbf{v} , \mathbf{u} and \mathbf{w} , \mathbf{v} .

- 1: Set $k = 1$
- 2: **while** not converged **do**
- 3: Compute singular triplet $(\sigma_{\min}, \mathbf{z}, \mathbf{u})$ of $H(s_k)$
- 4: Solve $\mathbf{v} \in \mathbb{C}^n$ from $(s_k E - A)\mathbf{v} = B\mathbf{z}$
- 5: Solve $\mathbf{w} \in \mathbb{C}^n$ from $(s_k E - A)^*\mathbf{w} = C\mathbf{u}$
- 6: Compute the new pole estimate

$$\begin{aligned} \begin{bmatrix} \operatorname{Re}(s_{k+1}) \\ \operatorname{Im}(s_{k+1}) \end{bmatrix} &= \begin{bmatrix} \operatorname{Re}(s_k) \\ \operatorname{Im}(s_k) \end{bmatrix} \\ &+ [\operatorname{Re}(\mathbf{w}^* E \mathbf{v}), -\operatorname{Im}(\mathbf{w}^* E \mathbf{v})]^\dagger \frac{\sigma_{\min}}{2}, \end{aligned}$$

- 7: The zero $\lambda = s_{k+1}$ with $\mathbf{x} = \mathbf{v}$ and $\mathbf{y} = \mathbf{w}$ has converged if $\|\mathbf{r}_k\|_2 < \epsilon$, where

$$\mathbf{r}_k = \begin{bmatrix} s_{k+1}E - A & -B \\ C^* & D \end{bmatrix} \begin{bmatrix} \mathbf{v} \\ \mathbf{z} \end{bmatrix}$$

- 8: Set $k = k + 1$
- 9: **end while**

This result is used in some algorithms for the computation of transmission zeros [39, 92]. In the following it will be shown how, given a system $\Sigma = (E, A, \mathbf{b}, \mathbf{c}, d)$, an inverse system $\Sigma_z = (E_z, A_z, \mathbf{b}_z, \mathbf{c}_z, d_z)$ can be defined, so that the transfer function $H_z(s)$ of Σ_z is the inverse of the transfer function $H(s)$ of Σ . The important consequence of these results is that the dominant zeros become the dominant poles of the inverse transfer function, so that existing algorithms such as DPA and SADPA can be used to compute the dominant zeros via the dominant poles of the inverse system.

The following theorem defines a system $\Sigma_z = (E_z, A_z, \mathbf{b}_z, \mathbf{c}_z, 0)$, with transfer function $H_z(s)$, such that $H_z(s)H(s) = H(s)H_z(s) = 1$, where $H(s)$ is the transfer function of $\Sigma = (A, \mathbf{b}, \mathbf{c}, 0)$. Existing literature that mentions such a system (for $d = 0$) is not known to the authors.

Theorem 5.3.1. *Let*

$$y(s) = H(s)u(s) = [\mathbf{c}^*(sE - A)^{-1}\mathbf{b}] u(s)$$

be the transfer function of the descriptor realization $\Sigma = (E, A, \mathbf{b}, \mathbf{c}, d = 0)$, and let

$$y_z(s) = H_z(s)u_z(s) = [\mathbf{c}_z^*(sE_z - A_z)^{-1}\mathbf{b}_z] u_z(s)$$

be the transfer function of the augmented descriptor realization

$$\Sigma_z = (E_z, A_z, \mathbf{b}_z, \mathbf{c}_z, d_z = 0),$$

where

$$A_z = \begin{bmatrix} A & \mathbf{b} \\ -\mathbf{c}^* & 0 \end{bmatrix}, \quad E_z = \begin{bmatrix} E & 0 \\ 0 & 0 \end{bmatrix},$$

$$\mathbf{b}_z = \begin{bmatrix} \mathbf{b} \\ 1 \end{bmatrix}, \quad \mathbf{c}_z = \begin{bmatrix} \mathbf{c} \\ 1 \end{bmatrix}, \quad d_z = 0.$$

Then $H_z(s) = H^{-1}(s)$.

Proof. Let $u \equiv u(s)$, $y \equiv y(s) = \mathbf{c}^*\mathbf{x}(s)$, $\mathbf{x} \equiv \mathbf{x}(s) = (sE - A)^{-1}\mathbf{b}u(s)$, $u_z \equiv u_z(s)$, $y_z \equiv y_z(s) = \mathbf{c}_z^*\mathbf{x}_z(s)$, $\mathbf{x}_z \equiv \mathbf{x}_z(s) = (sE_z - A_z)^{-1}\mathbf{b}_z u_z(s)$, and the Rosenbrock system matrix [127] equation for Σ

$$\begin{bmatrix} sE - A & -\mathbf{b} \\ \mathbf{c}^* & 0 \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ u \end{bmatrix} = \begin{bmatrix} 0 \\ y \end{bmatrix}.$$

By adding $(\mathbf{b}y - \mathbf{b}y)$ to the left side of the equation $(sE - A)\mathbf{x} - \mathbf{b}u = 0$ and defining the error $(u - y)$ as a new algebraic variable, one obtains

$$\left[\begin{array}{cc|c|c} sE - A & -\mathbf{b} & & -\mathbf{b} \\ \mathbf{c}^* & 0 & & -1 \\ \hline & & 1 & \\ \hline & & & 0 \end{array} \right] \begin{bmatrix} \mathbf{x} \\ u - y \\ y \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ u \end{bmatrix}$$

or

$$\begin{bmatrix} sE_z - A_z & -\mathbf{b}_z \\ \mathbf{c}_z^* & 0 \end{bmatrix} \begin{bmatrix} \mathbf{x}_z \\ u_z \end{bmatrix} = \begin{bmatrix} 0 \\ y_z \end{bmatrix},$$

which is the Rosenbrock system matrix equation for Σ_z . Hence $u_z(s) = y(s)$, $y_z(s) = u(s)$ and $H_z(s) = H^{-1}(s)$. \square

A similar result can be obtained for the case $d \neq 0$, see for example [78, Ex. 2.2-20] and [79, 137], which is described below in the form of a theorem.

Theorem 5.3.2. *Let*

$$y(s) = H(s)u(s) = [\mathbf{c}^*(sE - A)^{-1}\mathbf{b} + d] u(s)$$

be the transfer function of the descriptor realization $\Sigma = (E, A, \mathbf{b}, \mathbf{c}, d \neq 0)$, and let

$$y_z(s) = H_z(s)u_z(s) = [\mathbf{c}_z^*(sE_z - A_z)^{-1}\mathbf{b}_z + d_z] u_z(s)$$

be the transfer function of the descriptor realization $\Sigma_z = (E_z, A_z, \mathbf{b}_z, \mathbf{c}_z, d_z \neq 0)$, where

$$\begin{aligned} A_z &= A - d^{-1}\mathbf{b}\mathbf{c}^*, & E_z &= E, \\ \mathbf{b}_z &= d^{-1}\mathbf{b}, & \mathbf{c}_z &= -d^{-1}\mathbf{c}, & d_z &= d^{-1}. \end{aligned}$$

Then $H_z(s) = H^{-1}(s)$.

Proof. Let $u \equiv u(s)$, $y \equiv y(s) = \mathbf{c}^*\mathbf{x}(s) + du(s)$, $\mathbf{x} \equiv \mathbf{x}(s) = (sE - A)^{-1}\mathbf{b}u(s)$, $u_z \equiv u_z(s)$, $y_z \equiv y_z(s) = \mathbf{c}_z^*\mathbf{x}_z(s) + d_z u_z(s)$, $\mathbf{x}_z \equiv \mathbf{x}_z(s) = (sE_z - A_z)^{-1}\mathbf{b}_z u_z(s)$, and the Rosenbrock system matrix equation for Σ

$$\begin{bmatrix} sE - A & -\mathbf{b} \\ \mathbf{c}^* & d \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ u \end{bmatrix} = \begin{bmatrix} 0 \\ y \end{bmatrix}. \quad (5.3.5)$$

The expression $u = -d^{-1}\mathbf{c}^*\mathbf{x} + d^{-1}y$ is obtained from the second equation of (5.3.5), which is substituted into the first equation of (5.3.5) to yield

$$\begin{bmatrix} sE - (A - d^{-1}\mathbf{b}\mathbf{c}^*) & -d^{-1}\mathbf{b} \\ -d^{-1}\mathbf{c}^* & d^{-1} \end{bmatrix} \begin{bmatrix} \mathbf{x}_z \\ y \end{bmatrix} = \begin{bmatrix} 0 \\ u \end{bmatrix},$$

or

$$\begin{bmatrix} sE_z - A_z & -\mathbf{b}_z \\ \mathbf{c}_z^* & d_z \end{bmatrix} \begin{bmatrix} \mathbf{x}_z \\ u_z \end{bmatrix} = \begin{bmatrix} 0 \\ y_z \end{bmatrix},$$

which is the Rosenbrock system matrix equation for Σ_z . Hence $u_z(s) = y(s)$, $y_z(s) = u(s)$ and $H_z(s) = H^{-1}(s)$. \square

In fact, DZA applied to $(E, A, \mathbf{b}, \mathbf{c}, d)$ and DPA applied to $(E_z, A_z, \mathbf{b}_z, \mathbf{c}_z, d_z)$ are equivalent processes and hence produce the same results, provided the initial shift is the same:

Theorem 5.3.3. *DZA($E, A, \mathbf{b}, \mathbf{c}, d, s_0$) and DPA($E_z, A_z, \mathbf{b}_z, \mathbf{c}_z, d_z, s_0$) are equivalent.*

Proof. First consider the case $d = d_z = 0$ (cf. Theorem 5.3.1 for notation). It can be verified, with $\mathbf{v} = (s_k E - A)^{-1}\mathbf{b}$ and $\mathbf{w} = (s_k E - A)^{-*}\mathbf{c}$, that

$$\mathbf{v}_z = \frac{1}{\mathbf{c}^*\mathbf{v}} \begin{bmatrix} \mathbf{v} \\ 1 - \mathbf{c}^*\mathbf{v} \end{bmatrix} \quad \text{and} \quad \mathbf{w}_z = \frac{1}{\mathbf{b}^*\mathbf{w}} \begin{bmatrix} -\mathbf{w} \\ 1 + \mathbf{b}^*\mathbf{w} \end{bmatrix}$$

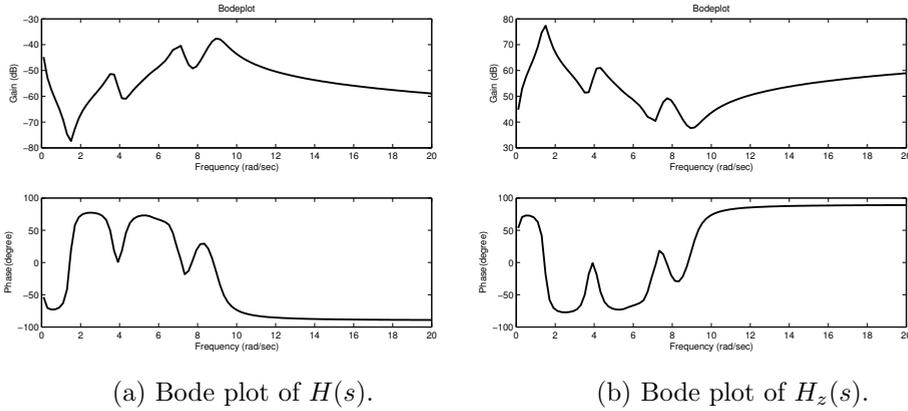


Figure 5.3: Bode plots of the New England test system (66 states, see also Fig. 5.1): (a) strictly proper transfer function $H(s) = W_{36}(s)/Pmec_{36}(s)$; (b) improper transfer function $H_z(s) = H^{-1}(s)$. The dominant zeros of $H(s)$ (dips) are the dominant poles of $H_z(s)$ (peaks).

are solutions of

$$(s_k E_z - A_z) \mathbf{v}_z = \mathbf{b}_z \quad \text{and} \quad (s_k E_z - A_z)^* \mathbf{w}_z = \mathbf{c}_z.$$

Since $(\mathbf{b}^* \mathbf{w})^* = \mathbf{c}^* \mathbf{v}$, it follows that if $s_k^{dza} = s_k^{dpa}$, then

$$s_{k+1}^{dpa} = s_k^{dpa} - \frac{\mathbf{c}_z^* \mathbf{v}_z}{\mathbf{w}_z^* E_z \mathbf{v}_z} = s_k^{dza} + \frac{\mathbf{c}^* \mathbf{v}}{\mathbf{w}^* E \mathbf{v}} = s_{k+1}^{dza}.$$

Because $s_0^{dza} = s_0^{dpa}$ it follows that $s_k^{dza} = s_k^{dpa}$ for all $k \geq 0$.

A similar argument can be used for the case $d \neq 0$ (cf. Theorem 5.3.2), by noting that

$$\mathbf{v}_z = \frac{1}{\mathbf{c}^* \mathbf{v} + d} \mathbf{v} \quad \text{and} \quad \mathbf{w}_z = \frac{-1}{\mathbf{b}^* \mathbf{w} + d} \mathbf{w}$$

are solutions of

$$(s E_z - A_z) \mathbf{v}_z = \mathbf{b}_z \quad \text{and} \quad (s E_z - A_z)^* \mathbf{w}_z = \mathbf{c}_z.$$

□

This equivalence strengthens the motivation in Section 5.4 to use the SADPA and SAMDP algorithms for the computation of the dominant zeros.

The fact that $H_z(s) = H^{-1}(s)$ can also be observed from the symmetric Bode plots in Fig. 5.3, where it is clearly visible that the dominant zeros of $H(s)$ (dips) become the dominant poles of $H_z(s)$ (peaks). A dominant zero of $H(s)$ can now also be defined as a dominant pole of $H^{-1}(s)$.

Definition 5.3.4. A zero z_i of an invertible transfer function $H(s)$ is called dominant if it is a dominant pole of $H^{-1}(s)$ (by Definition 5.2.1).

MIMO transfer functions

Also for a square MIMO system $\Sigma = (E, A, B, C, D)$ with transfer function $H(s)$, a system Σ_z can be defined so that $H_z(s)$ is the inverse of $H(s)$. The result for systems with $D = 0$ is:

Theorem 5.3.5. *Let $H(s) = C^*(sE - A)^{-1}B$ be the transfer function of $\Sigma = (E, A, B, C, D = 0)$, and let $H_z(s) = C_z^*(sE_z - A_z)^{-1}B_z$ be the transfer function of $\Sigma_z = (E_z, A_z, B_z, C_z, D_z = 0)$, where*

$$A_z = \begin{bmatrix} A & B \\ -C^* & 0 \end{bmatrix}, \quad E_z = \begin{bmatrix} E & 0 \\ 0 & 0 \end{bmatrix}, \\ B_z = \begin{bmatrix} B \\ I_m \end{bmatrix}, \quad C_z = \begin{bmatrix} C \\ I_m \end{bmatrix}, \quad D_z = 0,$$

and $I_m \in \mathbb{R}^{m \times m}$ is the $m \times m$ identity matrix. Then $H_z(s) = H^{-1}(s)$.

Proof. Manipulation using the Schur complement of A in A_z and basic linear algebra operations show that $H(s)H_z(s) = I_m$. \square

For nonsingular D , a generalization of Theorem 5.3.2 can be obtained, see also [137]:

Theorem 5.3.6. *Let $H(s) = C^*(sE - A)^{-1}B + D$ be the transfer function of $\Sigma = (E, A, B, C, D \text{ non-singular})$, and let $H_z(s) = C_z^*(sE_z - A_z)^{-1}B_z + D_z$ be the transfer function of $\Sigma_z = (E_z, A_z, B_z, C_z, D_z)$, where*

$$A_z = A - BD^{-1}C^*, \quad E_z = E, \quad B_z = BD^{-1}, \quad C_z = -D^{-1}C, \quad D_z = D^{-1}.$$

Then $H_z(s) = H^{-1}(s)$.

Proof. The Sherman-Morrison-Woodbury formula [64, p. 50] gives

$$(sE - (A - BD^{-1}C^*))^{-1} \\ = \tilde{E}(I_n - BD^{-1}(I_m + C^*\tilde{E}BD^{-1})^{-1}C^*\tilde{E}),$$

where $\tilde{E} = (sE - A)^{-1}$, and using basic matrix-vector operations it follows that $H(s)H_z(s) = I_m$. \square

The important consequence of these results is that the MIMO dominant pole algorithm SAMDP can be used to compute the dominant zeros, via the dominant poles of the inverse system. In fact, MDZA(E, A, B, C, D, s_0) and MDPA($E_z, A_z, B_z, C_z, D_z, s_0$) are equivalent processes (cf. Theorem 5.3.3). Similar results may be obtained for non-square MIMO systems, using pseudo-inverses (see for instance [164]).

5.4 Computing more than one zero

It is not attractive to run (M)DZA for a number of different initial shifts to compute more than one dominant zero, since there is a big chance of recomputing an already found zero. Deflation, see for instance [132], is a well known technique to avoid this problem. For the computation of many dominant poles, deflation combined with subspace acceleration and a selection strategy has led to the efficient Subspace Accelerated DPA (SADPA) [123] and SAMDP [122].

Although there are efficient methods for the computation of few eigenvalues of large-scale generalized eigenvalue problems of the form (5.2.5), such as the Jacobi-Davidson QZ method [51], these methods are less suitable for the computation of dominant zeros, since they are designed to compute a few eigenvalues near a specific target.

The dominant zeros of $H(s)$ can be computed as the dominant poles of $H_z(s)$, by applying SADPA (or SAMDP) to the inverse systems as defined in Section 5.3.3. This is the most flexible approach, since it not only allows for the computation of the dominant zeros in the sense of Definition 5.3.4, but can also easily be adapted to compute the RHP zeros or lowest damped zeros (by using a different selection criterion, see [123, 122] for more details). The realizations of the inverse systems can be constructed efficiently following the results in Section 5.3.3, and because the structure of A_z and E_z can be exploited in the solves with $s_k E_z - A_z$, the increase in computational time will be very limited.

The use of subspace acceleration has also the advantage of avoiding convergence to (extraneous) zeros at infinity, since the selection criterion takes care of selecting the most dominant estimates every iteration. Convergence to zeros at infinity occasionally (and eventually) happens for the single zero DZA algorithm. Deflation by restriction ([111], see also chapter 3) can be used to compute more than one zero with the single zero DZA algorithm, with the advantage of not keeping a search space. However, also this variant eventually converges to zeros at infinity, and furthermore, since it is a Newton scheme, it may get stuck in stable period 2 cycle of the Newton scheme itself. Subspace acceleration also prevents this and therefore the most attractive approach is to compute the zeros via the poles of the inverse system.

Another advantage of using SADPA/SAMDP for the computation of the dominant zeros is that a single algorithm can be used for both the computation of the dominant poles and zeros, so that only one implementation is needed.

5.5 Numerical experiments and applications

There are several applications where it is needed to compute some specific zeros. In the following subsections a number of SISO and MIMO transfer functions of the Brazilian Interconnect Power System (BIPS) are considered. See [123, 122] for more information on BIPS. The practical implementations operate on the sparse descriptor system model. All experiments were carried out in Matlab 7.2 on a SUN

Table 5.1: The 10 most dominant zeros and corresponding scaled residue matrix norms of the 2×2 transfer function of BIPS, sorted in decreasing $|R_i|/|\operatorname{Re}(\lambda_i)|$. Zeros found by SAMDP and/or SAMDZ are marked with 'yes'.

Modes		Residues	SAMDP	SAMDZA
Real	Imaginary	$\ R_i\ /\operatorname{Re}(\lambda_i)$	$s_0 = 10i$	$s_0 = 10i$
$-3.138 \cdot 10^{-2}$	± 4.356	1.010	yes	
$+4.484 \cdot 10^{-2}$	± 1.154	$1.761 \cdot 10^{-1}$	yes	
$-3.059 \cdot 10^{-1}$	± 3.667	$6.238 \cdot 10^{-2}$	yes	
$-6.203 \cdot 10^{-1}$	± 7.402	$3.709 \cdot 10^{-2}$	yes	yes
$-1.391 \cdot 10^{-1}$	± 0.2675	$3.305 \cdot 10^{-2}$	yes	
$-4.451 \cdot 10^{-1}$	± 3.015	$2.413 \cdot 10^{-2}$	yes	
-2.456		$2.031 \cdot 10^{-2}$		
-2.455		$1.711 \cdot 10^{-2}$		
$-7.478 \cdot 10^{-1}$	± 5.993	$1.269 \cdot 10^{-3}$		
-2.403		$8.678 \cdot 10^{-3}$		

Ultra 20 (AMD Opteron 2.8GHz), using codes based on the algorithms presented in [123, 122].

5.5.1 A comparison

The SADPA [123] and SAMDP [122] algorithms can be used to compute the dominant poles of the inverse system. To illustrate the importance of the selection criterion, SAMDP is compared to SAMDZ, subspace accelerated MDZA, with a (naive) selection criterion: select the estimates closest to the initial estimate s_0 . As an example, consider the 2×2 transfer function of BIPS, whose inputs are the voltage references at the excitation systems for generators 1107 and 5061, and whose outputs are the rotor speed deviations for the same two generators (see also [123]). Of interest here are the non-minimum phase zero (in the right half plane), and a number of most dominant zeros. The non-minimum phase (complex conjugate) zero $z \approx 4.48 \cdot 10^{-2} \pm 1.15i$ is exactly the unstable (complex conjugate) pole associated with the North-South mode in the absence of power system stabilizers at the two generating stations (see Section 5.2.3 for the theoretical explanation). SAMDP with $s_0 = 10i$ not only finds the RHP zero, but also computes many of the other dominant zeros: in Table 5.1, the 10 most dominant zeros are listed, and the zeros computed by SAMDP and SAMDZ, respectively, are marked with 'yes'. Both methods were allowed to compute 30 (pairs of) zeros. SAMDP needed 50 seconds (227 LU -factorizations), while SAMDZ needed 45 seconds (141 LU -factorizations). Clearly, SAMDP applied to the inverse system is better than SAMDZ. While SAMDP finds dominant zeros, SAMDZ finds zeros in the neighborhood of the initial shift. Providing a different initial shift does not help SAMDZ, while SAMDP still finds the dominant zeros.

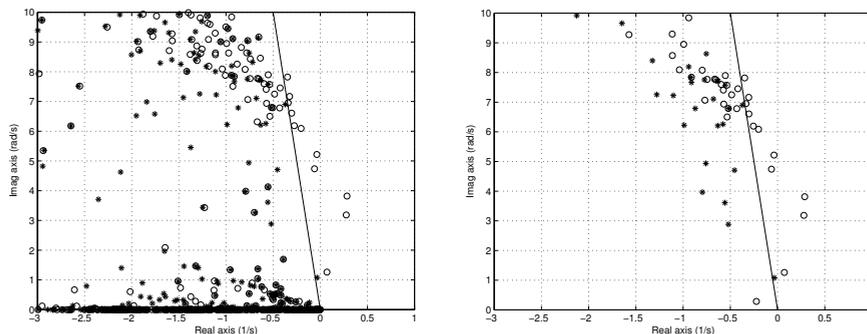


Figure 5.4: Pole-zero map of the 46×46 transfer matrix for disturbance channels (left), and poles and zeros found by SAMDP (right). Circles denote the transfer function zeros and stars denote the system poles. Complex zeros (and poles) appear in complex conjugate pairs, but here only the upper half of the complex plane is shown. Left half plane poles and zeros to the right of the solid line have damping ratio below 5%.

5.5.2 Computing dominant poles and zeros by SAMDP

Fig. 5.4 shows the pole-zero map for the 46×46 matrix transfer function $VPSS(s)/VPSS_d(s)$, where vector $VPSS_d$ contains input disturbance channels to each of the 46 PSSs in the system. The poles and zeros, and corresponding left and right eigenvectors, were obtained by use of the Matlab QR and QZ eigenroutines (`eig`, `schur`) in 146 seconds and 408 seconds, respectively. The zeros for this disturbance channel transfer function, as explained in Section 5.2.3, are identical to the system poles in the absence of PSSs. Note that the BIPS would be unstable at this operating point in the absence of PSSs: there are 3 unstable poles and 8 other poles (see the corresponding circles in Fig. 5.4 (left)) whose damping ratios are below 5%.

The SAMDP algorithm was applied to the inverse system in order to find rightmost dominant zeros for this 46×46 transfer matrix. The results are shown in Fig. 5.4 (right), where it is seen that the algorithm was able to capture the most relevant zeros. The 35 (complex conjugate) zeros, and corresponding left and right eigenvectors, were computed within 95 seconds (initial shift $s_0 = 1i$, 364 factorizations). Also the rightmost dominant poles computed by SAMDP (applied to the original system) are shown ($s_0 = 5i$, 35 (complex conjugate) triplets computed within 100 seconds and 408 factorizations). In order to select the rightmost zeros and poles, the selection criterion used in SAMDP was adapted to select the approximations with smallest damping ratio. Note that the CPU time required to compute the dominant poles and zeros is less than the time needed to compute all eigentriplets of the full state-space matrices (via QZ). For larger systems it is not feasible to compute all eigentriplets, while SAMDP can be applied to very large systems to compute the dominant poles and zeros (and corresponding left and right eigenvectors).

The results not only confirm that the dominant zeros can be computed via the dominant poles of the inverse system, but also show the flexibility of the SADPA/SAMDP algorithms when using different selection criteria.

5.6 Conclusions

This chapter described some control system applications where there is need for the computation of scalar as well as multivariable transfer function zeros. It was shown that dominant zeros can be computed as the dominant poles of the inverse transfer function, by the SADPA and SAMDP algorithms. A new inverse system formulation, which is valid even for strictly proper transfer functions, is described in the chapter. This is a new finding, which is important from practical as well as theoretical viewpoints.

Various uses for transfer function zeros are described, and numerical results are given for multivariable system models of realistic power systems, showing the practical value and flexibility of the algorithms presented in this chapter.

Addendum

This chapter is also available as [125]

Joost Rommes, Nelson Martins, and Paulo C. Pellanda, *Efficient computation of large-scale transfer function dominant zeros*, Preprint 1358, Utrecht University, 2006.

and has been submitted for publication.