A Novel Approach to Model Order Reduction for Coupled Multiphysics Problems

Wil Schilders, Agnieszka Lutowska

A Novel Approach to Model Order Reduction for Coupled Multiphysics Problems

Wil H.A. Schilders and Agnieszka Lutowska

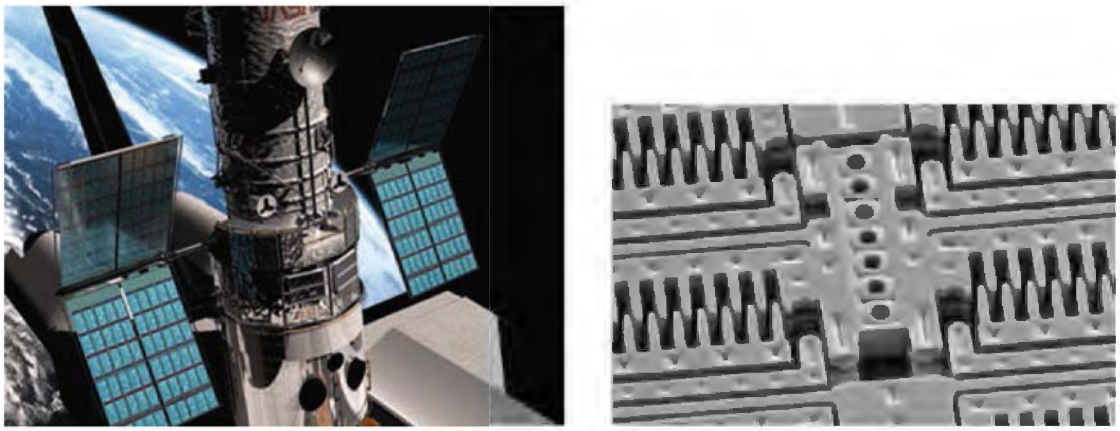
Abstract Model order reduction (MOR) has become an important tool in the design of complex high-tech systems. It can be used to find a low-order model that approximates the behavior of the original high-order model, where this low-order approximation facilitates both the computationally efficient analysis and controller design for the system to induce desired behavior. This chapter introduces MOR techniques that are designed especially for coupled problems, meaning that different physical phenomena are simulated in conjunction with each other. The method developed makes use of the reduction of the individual systems, and low rank approximations of the coupling blocks. This is done in such a way that existing software for industrial problems can be adapted in a straightforwward way. An industrial test case is described in detail, so as to demonstrate the effectiveness of the reduction technique.

1.1 Introduction

This chapter focuses on the development of a model reduction methodology for coupled multi-physical models to serve the efficient simulation-based design of the underlying coupled systems. Examples of coupled systems are larger systems such as magnetic resonance imaging (*MRI*) scanners, printers/copiers, precision motion stages, foldable solar panels of a space-telescope, down to very small systems such as very large scale integrated (*VLSI*) systems (see for instance [12, 21]) and micro-electromechanical systems (*MEMS*) (see for instance [15]). Figure 1.1 shows such examples.

W.H.A. Schilders
TU Eindhoven, Centre for Analysis, Scientific Computing and Applications
e-mail: w.h.a.schilders@tue.nl

A. Lutowska
TU Eindhoven, Centre for Analysis, Scientific Computing and Applications



(a) (b)

Fig. 1.1. Coupled systems. (a) Foldable solar panels (courtesy ESA); (b) A MEMS comb drive

The word *system*, which originates from the Greek word s'ustema and the Latin word *sustema*, stands for "a set of interacting or interdependent components forming an integrated whole". In this chapter, the integrated whole is called the *system* or *coupled system* and its individual components are called *sub-systems*. The word *model* as in "physical model" stands for a "representation" for the system under consideration, usually in terms of a set of physical quantities and relations. A coupled system's model consists of the coupled sub-systems' models. A *multi-physical model* is a model which is represented by multiple physical quantities such as temperature, structural mechanical displacements [7], electro-magnetic fields, and so forth. Simple systems in an insulated environment can often be described with few physical quantities and relations, while interacting systems frequently require more of such quantities and relations.

This chapter is about sub-systems which *interact*. When the interaction takes place inside a domain of interest or through the boundary which separates, such a domain of interest from the outside world such a system is called a *coupled system*. If the physical quantities interact through a discrete amount of *inputs* and *outputs* in space, then the system is said to be an *interconnected system* (see for instance [24]) rather than a *coupled system*.

To explain the envisioned reduction, first note that most physical models cannot be solved exactly with contemporary computers. To calculate an approximate solution, the involved physical quantities such as an electromagnetic field are first discretized, i.e., represented by a finite number of *degrees of freedom*, after which the physical equations are reformulated for the discretized physical quantities, leading to a *discrete system* of equations. This process is called *discretization* of the model. An accurate representation of physical quantities such as an electromagnetic field can require millions of degrees of freedom and consume a considerable amount of data storage and computation time. Therefore, an analysis of a coupled system's dynamic behavior can require excessive amounts of data storage and computation time.

We focus on state-of-the-art model order reduction techniques which *reduce the system as a whole based on available reduction techniques for the individual sub-systems*. Such methods are scarcely available and mostly in development. They have

an advantage that the individual sub-systems can be reduced in parallel (see [3]) with the method best suited for each of them. This can save a considerable amount of data storage and computational time since these systems are also smaller than the system as a whole. On the other hand, one must figure out how to couple the individually reduced models to a reduced model for the whole, i.e., need to figure out how to effectively deal with interior couplings/interconnections.

Our reduction methods are primarily for coupled time-invariant linear models. Time-dependent linear models, *affine models* (such as presented in [4]) and *nonlinear models* (see for instance [14, 23]) require other than the presented reduction techniques. Furthermore, we restrict ourselves to Krylov subspace projection techniques (see [11]).

In more detail, without loss of generality, we focus at systems which consist of two coupled subsystems. We suggest a method for the parallel reduction of the individual sub-systems, call it the *Separate Bases Reduction* algorithm (SBR), and show how to create a reduced model for the whole system based on the reduced parts. Furthermore, we show that this algorithm applied to coupled systems matches at least the same amount of moments as a standard method applied to the whole system would (see [24] for interconnected systems). We establish that a large amount of internal couplings leads to large and hence undesirable reduced models and show that this can be overcome with the use of a generalized singular value decomposition (GSVD) based reduction of the coupling blocks. However, the use of a GSVD-based approximation leads to an approximation of the moments – which as benchmark examples show can still be quite accurate.

The remainder of this chapter is focused on the presentation of the SBR algorithm and the GSVD reduction of the internal couplings. It is organized as follows. Section 1.2 describes Krylov subspace techniques, focusing on *coupled* and *interconnected* time-invariant linear systems. First, it shows what happens if standard techniques are applied to the coupled system as a whole – it shows that the *block structure* is lost. Next, it introduces existing techniques from the literature such as [1, 6, 9], still based on Krylov subspace methods for the coupled system as a whole, which preserve the block-structure and the number of matched moments. At the end of this chapter, we show an alternative method to efficiently calculate the second Krylov projector and extend the proof of [6] to a more general case, under assumptions.

In Sect. 1.3 we assume that Krylov subspace reduction methods are already available for the individual sub-systems and based thereon, we focus on the construction of a reduced-order model for the system as a whole. We show that this is possible (and also that moments are matched) in Theorem 1.2 and call the approach the Separate Bases Reduction algorithm (SBR). In Subsection 1.3.6 we show that the SBR algorithm also matches the standard double amount of moments if one uses two Krylov subspace projectors instead of one.

In Sect. 1.4 we show that the replacement of the coupling blocks by an explicitly rank-revealing GSVD based components leads to the same Krylov subspaces and hence matched moments. Approximations based on a few of the dominant modes lead to quite accurate moment approximation.

Finally, in Sect. 1.5 we apply the SBR algorithm to a benchmark system. The system under consideration is scaled in a specific manner such that it is numerically better conditioned. We conclude with some remarks and recommendations for further research in Sect. 1.6.

1.2 Block-Structure Preserving Model Order Reduction

Model order reduction is frequently based on Krylov subspace projections. The starting point is a linear time-invariant system, that in the Laplace domain is given by (later we will also use small letters $\mathbf{x}$, $\mathbf{y}$ for unknowns in the Laplace domain)

$$\begin{aligned} s\mathbf{E}\mathbf{X}(s) &= \mathbf{A}\mathbf{X}(s) + \mathbf{B}\mathbf{U}(s) \\ \mathbf{Y}(s) &= \mathbf{C}^T\mathbf{X}(s). \end{aligned} \tag{1.1}$$

The left side of Fig. 1.2 represents a schematic model of an *interconnected system* which consists of four sub-systems and a number of interconnections. These interconnections can be realized in different ways, which will be focused on in Sect. 1.3. The right side of Fig. 1.2 shows the system matrix $\mathbf{A}$ which corresponds to the graph on the left. The matrix $\mathbf{A}$ has a visible block-structure. Each of the gray diagonal blocks corresponds to one *sub-system*. The off-diagonal blocks are related to the *interconnections*. The blue dots in the off-diagonal blocks show that the two corresponding sub-systems are *interconnected*. The empty off-diagonal blocks show that there is no coupling between the corresponding two sub-systems.

In general, a system of k components, can be described by a linear system

$$\begin{aligned} s\begin{bmatrix} \mathbf{E}_{11} & \cdots & \mathbf{E}_{1k} \\ \vdots & \ddots & \vdots \\ \mathbf{E}_{k1} & \cdots & \mathbf{E}_{kk} \end{bmatrix} \begin{bmatrix} \mathbf{X}_1 \\ \vdots \\ \mathbf{X}_k \end{bmatrix} &= \begin{bmatrix} \mathbf{A}_{11} & \cdots & \mathbf{A}_{1k} \\ \vdots & \ddots & \vdots \\ \mathbf{A}_{k1} & \cdots & \mathbf{A}_{kk} \end{bmatrix} \begin{bmatrix} \mathbf{X}_1 \\ \vdots \\ \mathbf{X}_k \end{bmatrix} + \begin{bmatrix} \mathbf{B}_1 \\ \vdots \\ \mathbf{B}_k \end{bmatrix} \mathbf{U} \\ \mathbf{Y} &= [\mathbf{C}_1^T, \cdots, \mathbf{C}_k^T] \begin{bmatrix} \mathbf{X}_1 \\ \vdots \\ \mathbf{X}_k \end{bmatrix}, \end{aligned} \tag{1.2}$$

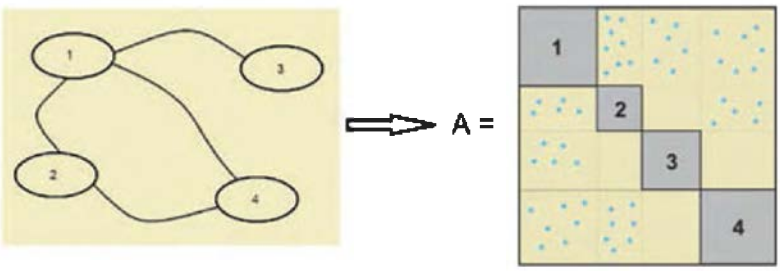


Fig. 1.2. Modeling of a coupled system

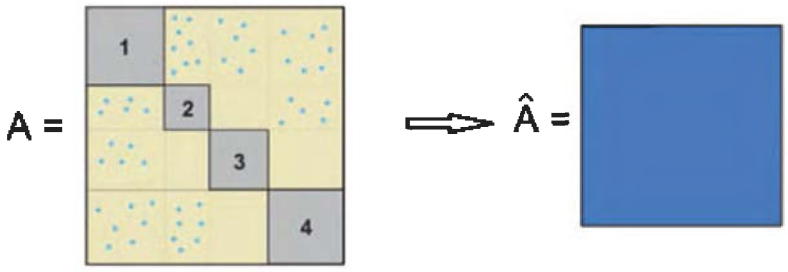


Fig. 1.3. Loosing of the structure in the reduced-order matrix $\hat{\mathbf{A}}$

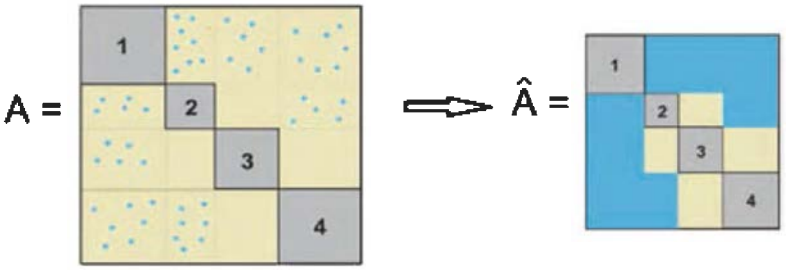


Fig. 1.4. Block structure preservation in the reduced-order matrix $\hat{\mathbf{A}}$

where the $\mathbf{X}_i \in \mathbb{R}^{N_i}$, $\mathbf{N}_i \in \mathbb{N}$, $i = 1, \ldots, k$, and the corresponding sub-blocks have compatible dimensions, where typically the off-diagonal blocks are not square. Naturally, we would like to still be able to recognize this type of *block-structure* in a reduced-order system matrix $\hat{\mathbf{A}}$. Unfortunately, if we apply a standard Krylov subspace reduction technique to the matrix $\mathbf{A}$ we unavoidably lose the block-structure and obtain a non-structured *dense* reduced-order matrix $\hat{\mathbf{A}}$ as shown in Fig. 1.3. In the next two subsections, we present a brief overview of Krylov-subspace based block-structure preserving reduction techniques. Such techniques applied to a structured matrix $\mathbf{A}$ result in a reduced-order matrix $\hat{\mathbf{A}}$ like the one shown in Fig. 1.4. Although the potential sparse nature of the interconnection off-diagonal blocks is lost, one can still recognize the system's general block-structure. The diagonal blocks still correspond to the reduced-order sub-systems and the zero blocks related to uncoupled sub-systems are preserved. The reduction techniques of this type are called block-structure preserving (BSP) methods (see for instance [9]). For more information about this type of technique the reader can consult for instance [18].

For the sake of simplicity assume that there are two coupled sub-systems ($k = 2$ in (1.2)). Then the system matrix has the block structure

$$\begin{bmatrix} \mathbf{A}_{11} & \mathbf{A}_{12} \\ \mathbf{A}_{21} & \mathbf{A}_{22} \end{bmatrix}$$

We call such a system an *interconnected system* if $\mathbf{A}_{12}$ and $\mathbf{A}_{21}$ are explicitly defined by means of their inputs and outputs, i.e., if for instance $\mathbf{A}_{12} = \mathbf{B}_3\mathbf{C}_4^{\mathrm{T}}$. Otherwise, if $\mathbf{A}_{12}$ and $\mathbf{A}_{21}$ are specified in unfactored form, we call the system a *coupled system*. However, it is reasonable to assume that even for the blocks specified in unfactored form there might be defined related input and output operators, i.e., that there can be

constructed $\mathbf{B}_3$ and $\mathbf{C}_4$ such that for instance $\mathbf{A}_{12} = \mathbf{B}_3\mathbf{C}_4^{\mathrm{T}}$. [13] considers possible construction methods for the input and output maps when $\mathbf{A}_{12}$ and $\mathbf{A}_{21}$ are specified in unfactored form.

1.2.1 Moment Matching Methods for the Coupled Formulations

We will begin with BSP methods that are directly applicable to coupled systems of the form (1.2)

$$s\begin{bmatrix}\mathbf{E}_{11} & \cdots & \mathbf{E}_{1k}\\ \vdots & \ddots & \vdots\\ \mathbf{E}_{k1} & \cdots & \mathbf{E}_{kk}\end{bmatrix}\begin{bmatrix}\mathbf{X}_1\\ \vdots\\ \mathbf{X}_k\end{bmatrix} = \begin{bmatrix}\mathbf{A}_{11} & \cdots & \mathbf{A}_{1k}\\ \vdots & \ddots & \vdots\\ \mathbf{A}_{k1} & \cdots & \mathbf{A}_{kk}\end{bmatrix}\begin{bmatrix}\mathbf{X}_1\\ \vdots\\ \mathbf{X}_k\end{bmatrix} + \begin{bmatrix}\mathbf{B}_1\\ \vdots\\ \mathbf{B}_k\end{bmatrix}\mathbf{U}$$

$$\mathbf{Y} = [\mathbf{C}_1^T, \cdots, \mathbf{C}_k^T]\begin{bmatrix}\mathbf{X}_1\\ \vdots\\ \mathbf{X}_k\end{bmatrix}.$$

This type of methods is studied in more detail in for instance [2,6,9]. These methods aim at the creation of a reduced-order model whose matrices exhibit the original block-structure and whose transfer function matches a number of moments of the transfer function of the original system. As for standard Krylov methods, the moment matching property is realized by projecting the original system matrices onto the appropriate input- and/or output-based Krylov subspaces by using the matrices $\tilde{V}$ and $\tilde{W}$ for a chosen expansion point $s_0 \in \mathbb{C}$. However, to preserve the block structure of the original system, the reduction bases also need to have a special shape. They are created by partitioning the matrices $\tilde{\mathbf{V}}$ and $\tilde{\mathbf{W}}$ into k sub-blocks (with k being the number of sub-systems)

$$\tilde{\mathbf{V}} = \begin{bmatrix}\mathbf{V}_1\\ \vdots\\ \mathbf{V}_k\end{bmatrix} \quad \text{and} \quad \tilde{\mathbf{W}} = \begin{bmatrix}\mathbf{W}_1\\ \vdots\\ \mathbf{W}_k\end{bmatrix},$$

where the number of rows in the blocks $\mathbf{V}_i$, $\mathbf{W}_i$, $i = 1,\ldots,k$ corresponds to the number of rows of the diagonal blocks $\mathbf{A}_{ii}$. Next, the blocks $\mathbf{V}_i$ and $\mathbf{W}_i$ are used to build block-diagonal reduction matrices $\mathbf{V}$ and $\mathbf{W}$

$$\mathbf{V} = \begin{bmatrix}\mathbf{V}_1 & & \\ & \ddots & \\ & & \mathbf{V}_k\end{bmatrix} \quad \text{and} \quad \mathbf{W} = \begin{bmatrix}\mathbf{W}_1 & & \\ & \ddots & \\ & & \mathbf{W}_k\end{bmatrix} \tag{1.3}$$

and the reduced-order system is obtained by projecting the original matrices

$$\hat{\mathbf{A}} = \mathbf{W}^T\mathbf{A}\mathbf{V}, \quad \hat{\mathbf{E}} = \mathbf{W}^T\mathbf{E}\mathbf{V}, \quad \hat{\mathbf{B}} = \mathbf{W}^T\mathbf{B}, \quad \hat{\mathbf{C}} = \mathbf{V}^T\mathbf{C}. \tag{1.4}$$

Note that since the splitting of the matrices $\tilde{\mathbf{V}}$ and $\tilde{\mathbf{W}}$ into sub-blocks may create linearly dependent columns, one needs to apply a *re-orthogonalization* of the matrices $\mathbf{V}$ and $\mathbf{W}$ to remove every possible linear dependence. Moreover, after re-orthogonalization, one has to assure, that the matrices $\mathbf{V}$ and $\mathbf{W}$ have the same number of columns. This can be done by adding the necessary number of random orthogonal columns to the matrix with the smallest amount of columns.

For the reduction bases created in the way described above, the following theorem holds.

Theorem 1.1 *Let* $\tilde{\mathbf{V}}$ *and* $\tilde{\mathbf{W}}$ *span the input- and output-based Krylov subspaces of the rth order around the expansion point* $s \in \mathbb{C}$ *for the system* (1.2)*. If*

$$colspan\tilde{\mathbf{V}} \subseteq colspan\mathbf{V} \qquad and \qquad colspan\tilde{\mathbf{W}} \subseteq colspan\mathbf{W},$$

then a reduced-order system computed as in (1.4) *has the transfer function that matches* $2p$ *moments of the transfer function of the original system* (1.2)*.*

There are several examples of methods that satisfy the foregoing. Paper [6] presents *SPRIM*, a structure preserving reduced order method for interconnect macromodeling. It focuses on an *RLC circuit* application, as model order reduction methods are of importance to *microchip* manufacturers since complex microchips such as processors contain many interconnected substructures. The relevant equations are (notation as in [6])

$$\mathscr{G}\mathbf{x} + \mathscr{C}\mathbf{x}' = \mathscr{B}u \tag{1.5}$$

with

$$\mathscr{G} = \begin{bmatrix} \mathbf{E}_g^{\mathrm{T}}\mathbf{G}\mathbf{E}_g & \mathbf{E}_l^{\mathrm{T}} \\ -\mathbf{E}_l & \mathbf{0} \end{bmatrix}, \quad \mathscr{C} = \begin{bmatrix} \mathbf{E}_c^{\mathrm{T}}\mathbf{C}\mathbf{E}_c & \mathbf{0} \\ \mathbf{0} & \mathbf{L} \end{bmatrix}, \quad \mathscr{B} = \begin{bmatrix} \mathbf{E}_i^{\mathrm{T}} \\ \mathbf{0} \end{bmatrix},$$

where $\mathbf{G}$, $\mathbf{C}$, and $\mathbf{L}$ are *symmetric positive definite* (square) matrices. The matrices $\mathbf{E}_g$, $\mathbf{E}_c$, $\mathbf{E}_l$ and $\mathbf{E}_i$ are parts of an adjacency matrix $\mathbf{E}$ which describes the connectivity of the electronic circuit, the subscripts g, c, l, i stand for branches containing resistors, capacitors, inductors and current sources. The SPRIM related Laplace domain transfer function $\mathbf{H}_{\mathrm{SPRIM}}$ is

$$\mathbf{H}_{\mathrm{SPRIM}}(s) = \mathscr{B}^{\mathrm{T}}\left(\mathscr{G} + s\mathscr{C}\right)^{-1}\mathscr{B}$$

where $\mathscr{B}$, $\mathscr{C}$ and $\mathscr{G}$ are re-written

$$\mathscr{C} = \begin{bmatrix} \mathbf{C}_1 & \mathbf{0} \\ \mathbf{0} & \mathbf{C}_2 \end{bmatrix}, \quad \mathscr{G} = \begin{bmatrix} \mathbf{G}_1 & \mathbf{G}_2^{\mathrm{T}} \\ -\mathbf{G}_2 & \mathbf{0} \end{bmatrix}, \quad \mathscr{B} = \begin{bmatrix} \mathbf{B}_1 \\ \mathbf{0} \end{bmatrix}.$$

The paper presents a reduction basis $\mathbf{V}$ of the type (1.3) in [6, (21)] and proves in [6, Theorem 3] that it ($\mathbf{W} = \mathbf{V}$) preserves $2p$ moments, double the amount preserved by *PRIMA*.

The technique proposed in [5] is motivated by the fact, that for some applications the single-point expansion does not give a sufficient approximation accuracy in the frequency range. On the other hand, using a multi-point expansion can result in excessively large models, especially for systems with many external inputs and outputs. The method proposed in the paper mentioned above, is based on creating a reduction space that consists of a number of sampling matrices $\mathbf{Z}_j,\ j = 1,\ldots,p$, computed for the system (1.2) for p sampling points s_j as follows

$$\mathbf{Z}_j = (s_j\mathbf{E} - \mathbf{A})^{-1}\mathbf{B}.$$

In other words, $\mathbf{Z}_j,\ j = 1,\ldots,p$ is a vector (or a matrix) that, after projecting the system (1.2) onto, will match the 0th moment around the point s_j of the original transfer function, since it consists of the input based starting matrix for the Krylov subspace for s_j. After computing p samples, the total sampling matrix $\mathbf{Z}$ is defined as

$$\mathbf{Z} = [\mathbf{Z}_1,\ldots,\mathbf{Z}_p].$$

Next, following the block-structure presented by the system matrices, matrix $\mathbf{Z}$ is split row-wise into k blocks $\tilde{\mathbf{V}}_i,\ i = 1,\ldots,k$

$$\mathbf{Z} = \begin{bmatrix} \tilde{\mathbf{V}}_1 \\ \vdots \\ \tilde{\mathbf{V}}_k \end{bmatrix}$$

and a block-diagonal projector is created

$$\tilde{\mathbf{V}} = \begin{bmatrix} \tilde{\mathbf{V}}_1 & & \\ & \ddots & \\ & & \tilde{\mathbf{V}}_k \end{bmatrix}.$$

Finally, the singular value decomposition (SVD) is performed on each of the blocks separately, to produce the orthogonal matrix $\mathbf{V}$

$$\mathbf{V} = \begin{bmatrix} \mathbf{V}_1 & & \\ & \ddots & \\ & & \mathbf{V}_k \end{bmatrix}$$

where $\mathbf{V}_i, i = 1,\ldots,k$ is an orthogonal basis for $\tilde{\mathbf{V}}_i$. At this point, further reduction in size is possible, by removing from the bases $\mathbf{V}_i, i = 1,\ldots,k$ the columns that correspond to to small singular values. Having the reduction bases $\mathbf{V}$, one can project the original system in the way defined in (1.4).

A noticeable advantage of the technique described above is, next to the block-structure preservation, the possibility of reducing different sub-systems with differ-

ent reduction ratio, determined for each sub-system separately, based on the singular values related to this sub-block as well as the importance of the considered sub-system in the total coupled system.

1.2.2 Two-Sided Structure Preserving Methods

In this section we will explain how the two-sided projection idea can be implemented in case of the block-structure preserving methods. A detailed explanation of the two-sided methods one can find for instance in [8]. Generally speaking, the use of a two-sided reduction method means, that the system is projected onto two subspaces, $\mathbf{V}$ and $\mathbf{W}$, based on input and output matrices, respectively. In case of the coupled system (1.10) (defined somewhat later), the reduction matrices $\mathbf{V}$ and $\mathbf{W}$, for an expansion point $s_0 \in \mathbb{C}$, are built according to the following algorithm:

1. Create matrix $\tilde{\mathbf{V}}$, whose columns span the nth Krylov subspace around $s_0 \in \mathbb{C}$

$$\tilde{\mathbf{V}} = \mathscr{K}_n(\mathbf{P}(s_0), \mathbf{R}(s_0)),$$

   where $\mathbf{P}(s_0)$ and $\mathbf{R}(s_0)$ are

$$\mathbf{P}(s_0) = (s_0\mathbf{E} - \mathbf{A})^{-1}\mathbf{E} \quad \text{and} \quad \mathbf{R}(s_0) = (s_0\mathbf{E} - \mathbf{A})^{-1}\mathbf{B}.$$

2. Create matrix $\tilde{\mathbf{W}}$, whose columns span the nth Krylov subspace around $s_0 \in \mathbb{C}$

$$\tilde{\mathbf{W}} = \mathscr{K}_n(\mathbf{S}(s_0), \mathbf{T}(s_0)),$$

   where $\mathbf{S}(s_0)$ and $\mathbf{T}(s_0)$ are

$$\mathbf{S}(s_0) = (s_0\mathbf{E} - \mathbf{A})^{-T}\mathbf{E}^T \quad \text{and} \quad \mathbf{T}(s_0) = (s_0\mathbf{E} - \mathbf{A})^{-T}\mathbf{C}.$$

3. Build the block-diagonal reduction matrix $\mathbf{V}$ with $N_1 + N_2 = N$ rows

$$\mathbf{V} = \begin{bmatrix} \mathbf{V}_1 & \mathbf{0} \\ \mathbf{0} & \mathbf{V}_2 \end{bmatrix},$$

   where $\mathbf{V}_1$ and $\mathbf{V}_2$ contain the first N_1 respectively last N_2 rows of the matrix $\tilde{\mathbf{V}}$.
4. Build the block-diagonal reduction matrix $\mathbf{W}$ with $N_1 + N_2 = N$ rows

$$\mathbf{W} = \begin{bmatrix} \mathbf{W}_1 & \mathbf{0} \\ \mathbf{0} & \mathbf{W}_2 \end{bmatrix},$$

   where $\mathbf{W}_1$ and $\mathbf{W}_2$ contain the first N_1 respectively last N_2 rows of the matrix $\tilde{\mathbf{W}}$.

Different algorithms lead to $\tilde{\mathbf{V}}$ and $\tilde{\mathbf{W}}$ (and hence $\mathbf{V}$ and $\mathbf{W}$) with different specific properties (such as orthogonality or bi-orthogonality). Some properties and their advantages and disadvantages are discussed in [17].

The described BSP algorithm results in a block-structured reduced order system and uses both inputs *and* outputs. Consequently, the BSP-based reduced order sys-

tem's transfer function matches twice as many moments of the original system's transfer function.

1.3 Separate Bases Reduction Algorithm

Model order reduction techniques, designed especially for coupled or interconnected systems, became a new field of research in recent years. The common feature of this type of methods is the use of a special block-diagonal form reduction basis $\mathbf{V}$

$$\mathbf{V} = \begin{bmatrix} \mathbf{V}_1 & & \\ & \ddots & \\ & & \mathbf{V}_k \end{bmatrix} \tag{1.6}$$

that results from the splitting a matrix $\tilde{\mathbf{V}}$ created by a Krylov method applied directly to the coupled system. This approach allows for preservation of the zero-blocks in the coupled system's coefficient matrix. Such blocks appear when two of the sub-systems are not coupled (interconnected) or the coupling holds only in one direction. An example of uni-directional coupling can be a case of a vibrating structure, where the movement of the structure causes acoustic noise, but there is no influence (feedback) of the acoustic behavior of the system on it's dynamics.

Due to the fact that the zero-blocks are preserved in the reduced system, such MOR techniques are called *block structure preserving* (*BSP*) model reduction methods. Their application usually results in a good approximation of the original model. For most of them one can prove the moment matching property. However, this type of methods also has three important drawbacks:

- Though $\mathbf{V}$ in (1.6) (possibly) matches the same (number of) moments as $\tilde{\mathbf{V}}$, it has k times more column vectors and therefore leads to a k times larger reduced system.
- The calculation of $\tilde{\mathbf{V}}$ requires (repeatedly) solving systems with the entire coupled system's coefficient matrix which can be computationally (time- and memory-wise) expensive.
- In practice, the reduction techniques based on an uncoupled formulation of the system (see e.g. [24]) are restricted to the case of interconnected systems with a limited number of interconnections. Otherwise, the reduction procedure is not very efficient, since the dimension of the reduction basis (hence, the reduced-order model) grows very fast. Moreover, such techniques assume that the inputs $\mathbf{B}$ and outputs $\mathbf{C}$ of the sub-systems are both explicitly available. In case of a coupled system these are not explicitly available, only their product $\mathbf{BC}$ is.

In the remainder of this chapter, we will focus on the second and third issue. We present a reduction algorithm suitable for systems, coupled through a large number of couplings. We introduce a reduction technique based on an uncoupled formulation of a coupled system, called *Separate Bases Reduction* (*SBR*) algorithm.

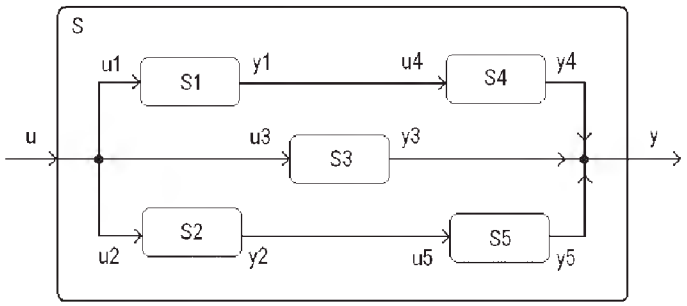


Fig. 1.5. Schematic representation of the interconnected system S

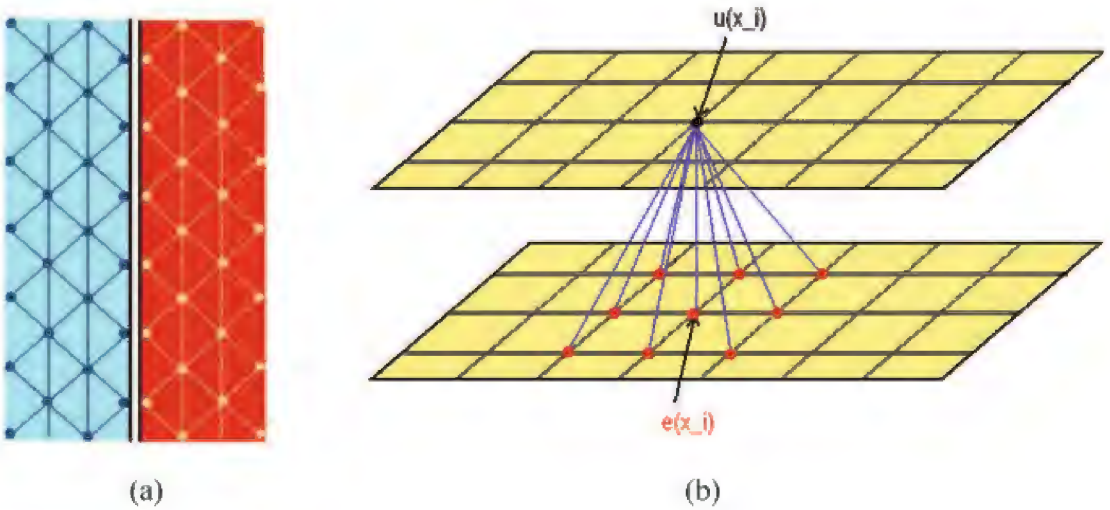


Fig. 1.6. Different types of strong coupling. (a) an interface coupling; (b) a strong coupling between different physical domains

It creates a reduction basis for each sub-system separately, hence is computationally cheaper compared to the reduction techniques that use a coupled formulation such as the BSP methods discussed in Sect. 1.2. However, the algorithm still suffers from the third point of the drawback list presented above. They can be easily applied to interconnected systems of a form shown in Fig. 1.5, where the sub-systems are not strongly interconnected (i.e. each sub-system exchanges information only with a small number of other sub-systems). We suggest a way to relax this limitation, and will also show how to apply the SBR algorithm to strongly coupled systems, i.e. to the systems, where many degrees of freedom of one sub-system are coupled to many degrees of freedom of other sub-systems and where the internal input and output matrices are not explicitly given in the system formulation. Examples of these types of coupled problems are shown in Fig. 1.6. Figure 6(a) presents a coupled system that consists of two sub-structures, for instance a solid body and a fluid. The coupling occurs at the interface, where all degrees of freedom of one sub-domain which are sufficiently close to the interface influence similar degrees of freedom of the second sub-domain and vice versa. A different type of *strong coupling* is shown in Fig. 6(b). This picture shows a situation, where all degrees of freedom related to both physical quantities u and e are located inside the same domain. Such situations appear for

instance in case of modeling of systems, where the dynamics of the structure is influenced by an electromagnetic field (and vice versa). In the depicted case the change of the velocity of the node $u(x_i)$ influences the electromagnetic field $x \mapsto e(x)$ at the node x_i, and at many nodes in the neighborhood of x_i.

1.3.1 Interconnected System – System Definition

In this subsection we introduce the family of linear *interconnected systems* to which the reduction algorithm is to be applied to. For the sake of simplicity, we focus on a system of two-subsystems where one sub-system's output is used as a part of the other sub-system's input and vice versa. However, the proposed method can easily be extended to systems composed of an arbitrary number of sub-systems.

1.3.1.1 The Uncoupled Formulation

The time domain behavior of each of the sub-systems S_1 and S_2 is modeled by a system of first order differential-algebraic equations after which the frequency domain behavior is obtained via Laplace transformation. For the two sub-system examples in Fig. 1.7, this procedure leads to the Laplace domain systems

$$S_1 : \begin{cases} s\mathbf{E}_{11}\mathbf{x}_1 = \mathbf{A}_{11}\mathbf{x}_1 + \mathbf{B}_1\mathbf{u}_1 + \mathbf{B}_3\mathbf{u}_3, \\ \mathbf{y}_1 = \mathbf{C}_1^T\mathbf{x}_1, \\ \mathbf{y}_3 = \mathbf{C}_3^T\mathbf{x}_1, \end{cases}$$

$$S_2 : \begin{cases} s\mathbf{E}_{22}\mathbf{x}_2 = \mathbf{A}_{22}\mathbf{x}_2 + \mathbf{B}_2\mathbf{u}_2 + \mathbf{B}_4\mathbf{u}_4, \\ \mathbf{y}_2 = \mathbf{C}_2^T\mathbf{x}_2, \\ \mathbf{y}_4 = \mathbf{C}_4^T\mathbf{x}_2. \end{cases}$$

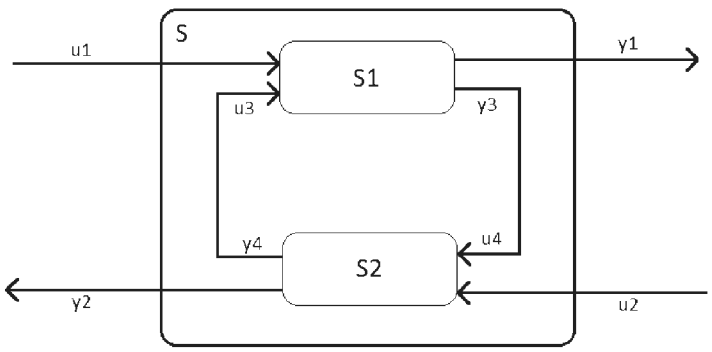


Fig. 1.7. Schematic representation of the interconnected system

Using matrix notation, the *system* S_1 and *system* S_2 can be described as

$$S_1 : \begin{cases} s\mathbf{E}_{11}\mathbf{x}_1 = \mathbf{A}_{11}\mathbf{x}_1 + [\mathbf{B}_1\ \mathbf{B}_3]\begin{bmatrix}\mathbf{u}_1\\ \mathbf{u}_3\end{bmatrix} \\ \begin{bmatrix}\mathbf{y}_1\\ \mathbf{y}_3\end{bmatrix} = \begin{bmatrix}\mathbf{C}_1^T\\ \mathbf{C}_3^T\end{bmatrix}\mathbf{x}_1, \end{cases} \tag{1.7}$$

$$S_2 : \begin{cases} s\mathbf{E}_{22}\mathbf{x}_2 = \mathbf{A}_{22}\mathbf{x}_2 + [\mathbf{B}_2\ \mathbf{B}_4]\begin{bmatrix}\mathbf{u}_2\\ \mathbf{u}_4\end{bmatrix} \\ \begin{bmatrix}\mathbf{y}_2\\ \mathbf{y}_4\end{bmatrix} = \begin{bmatrix}\mathbf{C}_2^T\\ \mathbf{C}_4^T\end{bmatrix}\mathbf{x}_2. \end{cases} \tag{1.8}$$

1.3.1.2 The Coupled System

When the output of S_1 is used as an input of S_2 and the output of S_2 is used as an input of S_1, equations (1.7) and (1.8) reduce to an interconnected Laplace domain system. Due to the design of the system depicted in Fig. 1.7 one has

$$\begin{cases} \mathbf{u}_3 = \mathbf{y}_4 = \mathbf{C}_4^T\mathbf{x}_2 \\ \mathbf{u}_4 = \mathbf{y}_3 = \mathbf{C}_3^T\mathbf{x}_1, \end{cases} \tag{1.9}$$

which in addition implies

$$\begin{cases} m_3 = p_4 \\ m_4 = p_3. \end{cases}$$

Using relation (1.9), the interconnected system (1.7) can be represented as a single *coupled system* **S** of equations

$$S : \begin{cases} s\mathbf{E}_{11}\mathbf{x}_1 = \mathbf{A}_{11}\mathbf{x}_1 + \mathbf{B}_1\mathbf{u}_1 + \mathbf{B}_3\mathbf{C}_4^T\mathbf{x}_2, \\ s\mathbf{E}_{22}\mathbf{x}_2 = \mathbf{A}_{22}\mathbf{x}_2 + \mathbf{B}_2\mathbf{u}_2 + \mathbf{B}_4\mathbf{C}_3^T\mathbf{x}_1, \\ \mathbf{y}_1 = \mathbf{C}_1^T\mathbf{x}_1, \\ \mathbf{y}_2 = \mathbf{C}_2^T\mathbf{x}_2 \end{cases}$$

and in matrix form

$$S : \begin{cases} s\begin{bmatrix}\mathbf{E}_{11} & \mathbf{0}\\ \mathbf{0} & \mathbf{E}_{22}\end{bmatrix}\begin{bmatrix}\mathbf{x}_1\\ \mathbf{x}_2\end{bmatrix} = \begin{bmatrix}\mathbf{A}_{11} & \mathbf{B}_3\mathbf{C}_4^T\\ \mathbf{B}_4\mathbf{C}_3^T & \mathbf{A}_{22}\end{bmatrix}\begin{bmatrix}\mathbf{x}_1\\ \mathbf{x}_2\end{bmatrix} + \begin{bmatrix}\mathbf{B}_1 & \mathbf{0}\\ \mathbf{0} & \mathbf{B}_2\end{bmatrix}\begin{bmatrix}\mathbf{u}_1\\ \mathbf{u}_2\end{bmatrix} \\ \begin{bmatrix}\mathbf{y}_1\\ \mathbf{y}_2\end{bmatrix} = \begin{bmatrix}\mathbf{C}_1^T & \mathbf{0}\\ \mathbf{0} & \mathbf{C}_2^T\end{bmatrix}\begin{bmatrix}\mathbf{x}_1\\ \mathbf{x}_2\end{bmatrix}. \end{cases} \tag{1.10}$$

Let $N = N_1 + N_2$, $m = m_1 + m_2$, $p = p_1 + p_2$ and define

$$\mathbf{E} = \begin{bmatrix} \mathbf{E}_{11} & \mathbf{0} \\ \mathbf{0} & \mathbf{E}_{22} \end{bmatrix}, \quad \mathbf{A} = \begin{bmatrix} \mathbf{A}_{11} & \mathbf{B}_3\mathbf{C}_4^T \\ \mathbf{B}_4\mathbf{C}_3^T & \mathbf{A}_{22} \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} \mathbf{B}_1 & \mathbf{0} \\ \mathbf{0} & \mathbf{B}_2 \end{bmatrix}, \quad \mathbf{C} = \begin{bmatrix} \mathbf{C}_1 & \mathbf{0} \\ \mathbf{0} & \mathbf{C}_2 \end{bmatrix}, \tag{1.11}$$

where $\mathbf{A}, \mathbf{E} \in \mathbb{R}^{N\times N}$, $\mathbf{B} \in \mathbb{R}^{N\times m}$, $\mathbf{C} \in \mathbb{R}^{N\times p}$. The matrices defined in (1.11) show a special block structure. The sub-systems' matrices $\mathbf{A}_{11}$ and $\mathbf{A}_{22}$ form the diagonal blocks of the system matrix $\mathbf{A}$ of S. The off-diagonal blocks are the products $\mathbf{B}_3\mathbf{C}_4^T$ and $\mathbf{B}_4\mathbf{C}_3^T$ of the internal input and output matrices of the sub-system. The input and output matrices $\mathbf{B}$ and $\mathbf{C}$ are block structured, as well as the matrix $\mathbf{E}$.

1.3.2 Transfer Functions of the Uncoupled and Coupled Systems

One of the questions arising at this point is the relation between the transfer functions of the sub-systems S1 and S2, and the transfer function of the coupled system. In this subsection we will study this issue. Let us begin with the uncoupled sub-systems. At $s \in \mathbb{C}$ the transfer function of sub-system S1 defined in (1.7) is given by

$$\begin{aligned} \mathbf{H}(s) &= \begin{bmatrix} \mathbf{C}_1^T \\ \mathbf{C}_3^T \end{bmatrix} (s\mathbf{E}_{11} - \mathbf{A}_{11})^{-1} [\mathbf{B}_1 \ \mathbf{B}_3] \\ &= \begin{bmatrix} \mathbf{C}_1^T (s\mathbf{E}_{11} - \mathbf{A}_{11})^{-1}\mathbf{B}_1 & \mathbf{C}_1^T (s\mathbf{E}_{11} - \mathbf{A}_{11})^{-1}\mathbf{B}_3 \\ \mathbf{C}_3^T (s\mathbf{E}_{11} - \mathbf{A}_{11})^{-1}\mathbf{B}_1 & \mathbf{C}_3^T (s\mathbf{E}_{11} - \mathbf{A}_{11})^{-1}\mathbf{B}_3 \end{bmatrix} = \begin{bmatrix} \mathbf{H}_{11}(s) & \mathbf{H}_{12}(s) \\ \mathbf{H}_{21}(s) & \mathbf{H}_{22}(s) \end{bmatrix}. \end{aligned} \tag{1.12}$$

For the sub-system S2 defined in (1.8), similarly

$$\begin{aligned} \mathbf{G}(s) &= \begin{bmatrix} \mathbf{C}_2^T \\ \mathbf{C}_4^T \end{bmatrix} (s\mathbf{E}_{22} - \mathbf{A}_{22})^{-1} [\mathbf{B}_2 \ \mathbf{B}_4] \\ &= \begin{bmatrix} \mathbf{C}_2^T (s\mathbf{E}_{22} - \mathbf{A}_{22})^{-1}\mathbf{B}_2 & \mathbf{C}_2^T (s\mathbf{E}_{22} - \mathbf{A}_{22})^{-1}\mathbf{B}_4 \\ \mathbf{C}_4^T (s\mathbf{E}_{22} - \mathbf{A}_{22})^{-1}\mathbf{B}_2 & \mathbf{C}_4^T (s\mathbf{E}_{22} - \mathbf{A}_{22})^{-1}\mathbf{B}_4 \end{bmatrix} = \begin{bmatrix} \mathbf{G}_{11}(s) & \mathbf{G}_{12}(s) \\ \mathbf{G}_{21}(s) & \mathbf{G}_{22}(s) \end{bmatrix}. \end{aligned} \tag{1.13}$$

At $s \in \mathbb{C}$ the transfer function of the coupled system (1.10) is

$$\begin{aligned} \mathbf{Z}(s) = \mathbf{C}^T (s\mathbf{E} - \mathbf{A})^{-1}\mathbf{B} &= \begin{bmatrix} \mathbf{C}_1^T & \mathbf{0} \\ \mathbf{0} & \mathbf{C}_2^T \end{bmatrix} \left(s \begin{bmatrix} \mathbf{E}_{11} & \mathbf{0} \\ \mathbf{0} & \mathbf{E}_{22} \end{bmatrix} - \begin{bmatrix} \mathbf{A}_{11} & \mathbf{B}_3\mathbf{C}_4^T \\ \mathbf{B}_4\mathbf{C}_3^T & \mathbf{A}_{22} \end{bmatrix} \right)^{-1} \begin{bmatrix} \mathbf{B}_1 & \mathbf{0} \\ \mathbf{0} & \mathbf{B}_2 \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{Z}_{11}(s) & \mathbf{Z}_{12}(s) \\ \mathbf{Z}_{21}(s) & \mathbf{Z}_{22}(s) \end{bmatrix}. \end{aligned} \tag{1.14}$$

Based on definitions Eqs. (1.12) to (1.14) we will express the components of the transfer function $\mathbf{Z}(s)$ in terms of the components of the transfer functions $\mathbf{H}(s)$ and $\mathbf{G}(s)$ in two manners. First we follow the typical approach used in the field of systems and control (more details can be found in for instance [19]). Secondly we use the Sherman-Morrison-Woodbury formula.

The Systems and Control Approach

The starting point of this approach are two transfer functions $\mathbf{H}(s)$ and $\mathbf{G}(s)$ of the sub-systems 1 and 2, respectively. For each sub-system, its transfer function relates its inputs to outputs:

$$\begin{bmatrix}\mathbf{y}_1\\ \mathbf{y}_3\end{bmatrix} = \begin{bmatrix}\mathbf{H}_{11}(s)\ \mathbf{H}_{12}(s)\\ \mathbf{H}_{21}(s)\ \mathbf{H}_{22}(s)\end{bmatrix}\begin{bmatrix}\mathbf{u}_1\\ \mathbf{u}_3\end{bmatrix}, \quad \begin{bmatrix}\mathbf{y}_2\\ \mathbf{y}_4\end{bmatrix} = \begin{bmatrix}\mathbf{G}_{11}(s)\ \mathbf{G}_{12}(s)\\ \mathbf{G}_{21}(s)\ \mathbf{G}_{22}(s)\end{bmatrix}\begin{bmatrix}\mathbf{u}_2\\ \mathbf{u}_4\end{bmatrix}$$

and

$$\begin{bmatrix}\mathbf{y}_1\\ \mathbf{y}_2\end{bmatrix} = \begin{bmatrix}\mathbf{Z}_{11}(s)\ \mathbf{Z}_{12}(s)\\ \mathbf{Z}_{21}(s)\ \mathbf{Z}_{22}(s)\end{bmatrix}\begin{bmatrix}\mathbf{u}_1\\ \mathbf{u}_2\end{bmatrix}. \tag{1.15}$$

Systems (1.12) and (1.13) in combination with relation (1.9) lead to

$$\mathbf{y}_1 = \mathbf{H}_{11}(s)\mathbf{u}_1 + \mathbf{H}_{12}(s)\mathbf{y}_4 \tag{1.16}$$
$$\mathbf{y}_3 = \mathbf{H}_{21}(s)\mathbf{u}_1 + \mathbf{H}_{22}(s)\mathbf{y}_4 \tag{1.17}$$
$$\mathbf{y}_2 = \mathbf{G}_{11}(s)\mathbf{u}_2 + \mathbf{G}_{12}(s)\mathbf{y}_3 \tag{1.18}$$
$$\mathbf{y}_4 = \mathbf{G}_{21}(s)\mathbf{u}_2 + \mathbf{G}_{22}(s)\mathbf{y}_3. \tag{1.19}$$

Substituting $\mathbf{y}_4$ of (1.19) for $\mathbf{y}_4$ in (1.17) we obtain

$$\mathbf{y}_3 = \mathbf{H}_{21}(s)\mathbf{u}_1 + \mathbf{H}_{22}(s)\mathbf{y}_4 = \mathbf{H}_{21}(s)\mathbf{u}_1 + \mathbf{H}_{22}(s)[\mathbf{G}_{21}(s)\mathbf{u}_2 + \mathbf{G}_{22}(s)\mathbf{y}_3]$$

and hence

$$\mathbf{y}_3 = [\mathbf{I} - \mathbf{H}_{22}(s)\mathbf{G}_{22}(s)]^{-1}[\mathbf{H}_{21}(s)\mathbf{u}_1 + \mathbf{H}_{22}(s)\mathbf{G}_{21}(s)\mathbf{u}_2]. \tag{1.20}$$

With this result and (1.19), we can also express $\mathbf{y}_4$ in terms of $\mathbf{u}_1$ and $\mathbf{u}_2$

$$\begin{aligned}\mathbf{y}_4 = {} & \mathbf{G}_{21}(s)\mathbf{u}_2 + \mathbf{G}_{22}(s)\mathbf{y}_3 = \mathbf{G}_{21}(s)\mathbf{u}_2 \\ & + \mathbf{G}_{22}(s)[\mathbf{I} - \mathbf{H}_{22}(s)\mathbf{G}_{22}(s)]^{-1}[\mathbf{H}_{21}(s)\mathbf{u}_1 + \mathbf{H}_{22}(s)\mathbf{G}_{21}(s)\mathbf{u}_2].\end{aligned} \tag{1.21}$$

Using (1.20) and (1.21) in (1.16) and (1.18), we arrive at

$$\begin{aligned}\mathbf{y}_1 = {} & \mathbf{H}_{11}(s)\mathbf{u}_1 + \mathbf{H}_{12}(s)\mathbf{y}_4 = \mathbf{H}_{11}(s)\mathbf{u}_1 \\ & + \mathbf{H}_{12}(s)\Big(\mathbf{G}_{21}(s)\mathbf{u}_2 + \mathbf{G}_{22}(s)[\mathbf{I} - \mathbf{H}_{22}(s)\mathbf{G}_{22}(s)]^{-1} \\ & \qquad \times [\mathbf{H}_{21}(s)\mathbf{u}_1 + \mathbf{H}_{22}(s)\mathbf{G}_{21}(s)\mathbf{u}_2]\Big) \\ = {} & \Big(\mathbf{H}_{11}(s) + \mathbf{H}_{12}(s)\mathbf{G}_{22}(s)[\mathbf{I} - \mathbf{H}_{22}(s)\mathbf{G}_{22}(s)]^{-1}\mathbf{H}_{21}(s)\Big)\mathbf{u}_1 \\ & + \Big(\mathbf{H}_{12}(s)\mathbf{G}_{21}(s) + \mathbf{H}_{12}(s)\mathbf{G}_{22}(s)[\mathbf{I} - \mathbf{H}_{22}(s)\mathbf{G}_{22}(s)]^{-1}\mathbf{H}_{22}(s)\mathbf{G}_{21}\Big)\mathbf{u}_2\end{aligned}$$

and

$$\begin{aligned}\mathbf{y}_2 &= \mathbf{G}_{11}(s)\mathbf{u}_2 + \mathbf{G}_{12}(s)\mathbf{y}_3 = \mathbf{G}_{11}(s)\mathbf{u}_2 + \mathbf{G}_{12}(s)[\mathbf{I} - \mathbf{H}_{22}(s)\mathbf{G}_{22}(s)]^{-1}[\mathbf{H}_{21}(s)\mathbf{u}_1 \\ &\quad + \mathbf{H}_{22}(s)\mathbf{G}_{21}(s)\mathbf{u}_2] = \mathbf{G}_{12}(s)[\mathbf{I} - \mathbf{H}_{22}(s)\mathbf{G}_{22}(s)]^{-1}\mathbf{H}_{21}(s)\mathbf{u}_1 \\ &\quad + \Big(\mathbf{G}_{11}(s) + \mathbf{G}_{12}(s)[\mathbf{I} - \mathbf{H}_{22}(s)\mathbf{G}_{22}(s)]^{-1}\mathbf{H}_{22}(s)\mathbf{G}_{21}(s)\Big)\mathbf{u}_2.\end{aligned}$$

This shows that the components of $\mathbf{Z}(s)$, as defined in (1.15), are

$$\mathbf{Z}_{11}(s) = \mathbf{H}_{11}(s) + \mathbf{H}_{12}(s)\mathbf{G}_{22}(s)[\mathbf{I} - \mathbf{H}_{22}(s)\mathbf{G}_{22}(s)]^{-1}\mathbf{H}_{21}(s) \tag{1.22}$$

$$\mathbf{Z}_{12}(s) = \mathbf{H}_{12}(s)\mathbf{G}_{21}(s) + \mathbf{H}_{12}(s)\mathbf{G}_{22}(s)[\mathbf{I} - \mathbf{H}_{22}(s)\mathbf{G}_{22}(s)]^{-1}\mathbf{H}_{22}(s)\mathbf{G}_{21}(s) \tag{1.23}$$

$$\mathbf{Z}_{21}(s) = \mathbf{G}_{12}(s)[\mathbf{I} - \mathbf{H}_{22}(s)\mathbf{G}_{22}(s)]^{-1}\mathbf{H}_{21}(s) \tag{1.24}$$

$$\mathbf{Z}_{22}(s) = \mathbf{G}_{11}(s) + \mathbf{G}_{12}(s)[\mathbf{I} - \mathbf{H}_{22}(s)\mathbf{G}_{22}(s)]^{-1}\mathbf{H}_{22}(s)\mathbf{G}_{21}(s). \tag{1.25}$$

Computing the Transfer Function of the Coupled System Using the Sherman–Morrison–Woodbury Formula

The evaluation of the transfer function of the coupled system, as defined in (1.14), requires a computation of an inverse of a block matrix. For a system consisting of an arbitrary number of sub-systems, a suitable tool towards this end is the Sherman–Morrison–Woodbury formula (see for instance [10] and references therein). This formula allows for a computationally cheap matrix inversion, as long as the considered matrix can be easily expressed as a sum of a matrix for which an inverse is known (or easy to compute) and a (low rank) correction. Let $\mathbf{L}$ be non-singular and let matrices $\mathbf{J}$, $\mathbf{M}$, $\mathbf{N}$ be of compatible size. Then the formula of $\mathbf{K} = \mathbf{L} + \mathbf{MJN}^T$ is (after [10])

$$\mathbf{K}^{-1} = (\mathbf{L} + \mathbf{MJN}^T)^{-1} = \mathbf{L}^{-1} - \mathbf{L}^{-1}\mathbf{M}(\mathbf{J}^{-1} + \mathbf{N}^T\mathbf{L}^{-1}\mathbf{M})^{-1}\mathbf{N}^T\mathbf{L}^{-1}, \tag{1.26}$$

In our case, the matrix to be inverted can be decomposed into

$$\begin{aligned}\mathbf{K} &= s\begin{bmatrix}\mathbf{E}_{11} & \mathbf{0} \\ \mathbf{0} & \mathbf{E}_{22}\end{bmatrix} - \begin{bmatrix}\mathbf{A}_{11} & \mathbf{B}_3\mathbf{C}_4^T \\ \mathbf{B}_4\mathbf{C}_3^T & \mathbf{A}_{22}\end{bmatrix} = \begin{bmatrix}s\mathbf{E}_{11} - \mathbf{A}_{11} & -\mathbf{B}_3\mathbf{C}_4^T \\ -\mathbf{B}_3\mathbf{C}_3^T & s\mathbf{E}_{22} - \mathbf{A}_{22}\end{bmatrix} \\ &= \begin{bmatrix}s\mathbf{E}_{11} - \mathbf{A}_{11} & \mathbf{0} \\ \mathbf{0} & s\mathbf{E}_{22} - \mathbf{A}_{22}\end{bmatrix} - \begin{bmatrix}\mathbf{0} & \mathbf{B}_3\mathbf{C}_4^T \\ \mathbf{B}_4\mathbf{C}_3^T & \mathbf{0}\end{bmatrix} \\ &= \mathbf{L} - \begin{bmatrix}\mathbf{0} & \mathbf{B}_3\mathbf{C}_4^T \\ \mathbf{B}_4\mathbf{C}_3^T & \mathbf{0}\end{bmatrix}\end{aligned}$$

where $\mathbf{L}$ is a block-diagonal matrix, whose inverse can be calculated by computing the inverses of each sub-block separately and the correction matrix can be factored

$$\begin{bmatrix}\mathbf{0} & \mathbf{B}_3\mathbf{C}_4^T \\ \mathbf{B}_4\mathbf{C}_3^T & \mathbf{0}\end{bmatrix} = \begin{bmatrix}\mathbf{0} & \mathbf{B}_3 \\ \mathbf{B}_4 & \mathbf{0}\end{bmatrix}\mathbf{I}\begin{bmatrix}\mathbf{C}_3^T & \mathbf{0} \\ \mathbf{0} & \mathbf{C}_4^T\end{bmatrix} = \mathbf{MJN}^T.$$

Abbreviate $\mathbf{G}_i(s) = (s\mathbf{E}_{ii} - \mathbf{A}_{ii})^{-1}$, $\mathbf{P}_i(s) = \mathbf{G}_i(s)\mathbf{E}_{ii}$ and $\mathbf{R}_i(s) = \mathbf{G}_i(s)[\mathbf{B}_i \ \mathbf{B}_{2+i}]$, $i = 1,2$ and omit the argument s when possible. Note that $\mathbf{R}_i = [\mathbf{R}_{i1}, \mathbf{R}_{i2}] = [\mathbf{G}_i\mathbf{B}_i, \mathbf{G}_i\mathbf{B}_{2+i}]$ consists of two blocks. Substituting the formulas for $\mathbf{L}, \mathbf{M}, \mathbf{J}$ and $\mathbf{N}$ into the the Sherman–Morrison–Woodbury formula, we get

$$
\begin{aligned}
&\left(s\begin{bmatrix} \mathbf{E}_{11} & \mathbf{0} \\ \mathbf{0} & \mathbf{E}_{22} \end{bmatrix} - \begin{bmatrix} \mathbf{A}_{11} & \mathbf{B}_3\mathbf{C}_4^T \\ \mathbf{B}_4\mathbf{C}_3^T & \mathbf{A}_{22} \end{bmatrix} \right)^{-1} \\
&\quad = \left(\begin{bmatrix} s\mathbf{E}_{11} - \mathbf{A}_{11} & \mathbf{0} \\ \mathbf{0} & s\mathbf{E}_{22} - \mathbf{A}_{22} \end{bmatrix} - \begin{bmatrix} \mathbf{0} & \mathbf{B}_3 \\ \mathbf{B}_4 & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{C}_3^T & \mathbf{0} \\ \mathbf{0} & \mathbf{C}_4^T \end{bmatrix} \right)^{-1} \\
&\quad = \begin{bmatrix} \mathbf{G}_1 & \mathbf{0} \\ \mathbf{0} & \mathbf{G}_2 \end{bmatrix} + \begin{bmatrix} \mathbf{G}_1 & \mathbf{0} \\ \mathbf{0} & \mathbf{G}_2 \end{bmatrix} \begin{bmatrix} \mathbf{0} & \mathbf{B}_3 \\ \mathbf{B}_4 & \mathbf{0} \end{bmatrix} \circ \\
&\qquad \left(\mathbf{I} - \begin{bmatrix} \mathbf{C}_3^T & \mathbf{0} \\ \mathbf{0} & \mathbf{C}_4^T \end{bmatrix} \begin{bmatrix} \mathbf{G}_1 & \mathbf{0} \\ \mathbf{0} & \mathbf{G}_2 \end{bmatrix} \begin{bmatrix} \mathbf{0} & \mathbf{B}_3 \\ \mathbf{B}_4 & \mathbf{0} \end{bmatrix} \right)^{-1} \begin{bmatrix} \mathbf{C}_3^T & \mathbf{0} \\ \mathbf{0} & \mathbf{C}_4^T \end{bmatrix} \begin{bmatrix} \mathbf{G}_1 & \mathbf{0} \\ \mathbf{0} & \mathbf{G}_2 \end{bmatrix} \\
&\quad = \begin{bmatrix} \mathbf{G}_1 & \mathbf{0} \\ \mathbf{0} & \mathbf{G}_2 \end{bmatrix} + \begin{bmatrix} \mathbf{0} & \mathbf{R}_{12} \\ \mathbf{R}_{22} & \mathbf{0} \end{bmatrix} \circ \\
&\qquad \left(\mathbf{I} - \begin{bmatrix} \mathbf{0} & \mathbf{C}_3^T\mathbf{R}_{12} \\ \mathbf{C}_4^T\mathbf{R}_{22} & \mathbf{0} \end{bmatrix} \right)^{-1} \begin{bmatrix} \mathbf{C}_3^T\mathbf{G}_1 & \mathbf{0} \\ \mathbf{0} & \mathbf{C}_4^T\mathbf{G}_2 \end{bmatrix},
\end{aligned}
\tag{1.27}
$$

where the entries of $\mathbf{G}$ and $\mathbf{R}$ depend on s. Using this result and Eqs. (1.12), (1.13), (1.14), one can find the formula for the transfer function of the coupled system

$$
\begin{aligned}
\mathbf{Z}(s) &= \begin{bmatrix} \mathbf{C}_1^T & \mathbf{0} \\ \mathbf{0} & \mathbf{C}_2^T \end{bmatrix} \left(\begin{bmatrix} \mathbf{G}_1 & \mathbf{0} \\ \mathbf{0} & \mathbf{G}_2 \end{bmatrix} + \begin{bmatrix} \mathbf{0} & \mathbf{R}_{12} \\ \mathbf{R}_{22} & \mathbf{0} \end{bmatrix} \circ \right. \\
&\qquad \left. \left(\mathbf{I} - \begin{bmatrix} \mathbf{0} & \mathbf{C}_3^T\mathbf{R}_{12} \\ \mathbf{C}_4^T\mathbf{R}_{22} & \mathbf{0} \end{bmatrix} \right)^{-1} \begin{bmatrix} \mathbf{C}_3^T\mathbf{G}_1 & \mathbf{0} \\ \mathbf{0} & \mathbf{C}_4^T\mathbf{G}_2 \end{bmatrix} \right) \begin{bmatrix} \mathbf{B}_1 & \mathbf{0} \\ \mathbf{0} & \mathbf{B}_2 \end{bmatrix} \\
&= \begin{bmatrix} \mathbf{C}_1^T\mathbf{R}_{11} & \mathbf{0} \\ \mathbf{0} & \mathbf{C}_2^T\mathbf{R}_{21} \end{bmatrix} + \begin{bmatrix} \mathbf{0} & \mathbf{C}_1^T\mathbf{R}_{12} \\ \mathbf{C}_2^T\mathbf{R}_{22} & \mathbf{0} \end{bmatrix} \circ \\
&\qquad \left(\mathbf{I} - \begin{bmatrix} \mathbf{0} & \mathbf{C}_3^T\mathbf{R}_{12} \\ \mathbf{C}_4^T\mathbf{R}_{22} & \mathbf{0} \end{bmatrix} \right)^{-1} \begin{bmatrix} \mathbf{C}_3^T\mathbf{R}_{11} & \mathbf{0} \\ \mathbf{0} & \mathbf{C}_4^T\mathbf{R}_{21} \end{bmatrix} \\
&= \begin{bmatrix} \mathbf{H}_{11}(s) & \mathbf{0} \\ \mathbf{0} & \mathbf{G}_{11}(s) \end{bmatrix} + \begin{bmatrix} \mathbf{0} & \mathbf{H}_{12}(s) \\ \mathbf{G}_{12}(s) & \mathbf{0} \end{bmatrix} \circ \\
&\qquad \left(\mathbf{I} - \begin{bmatrix} \mathbf{0} & \mathbf{H}_{22}(s) \\ \mathbf{G}_{22}(s) & \mathbf{0} \end{bmatrix} \right)^{-1} \begin{bmatrix} \mathbf{H}_{21}(s) & \mathbf{0} \\ \mathbf{0} & \mathbf{G}_{21}(s) \end{bmatrix}.
\end{aligned}
\tag{1.28}
$$

It is easy to show, that the formulation (1.28) is equivalent to the formulation given by Eqs. (1.22) to (1.25). Moreover, (1.28) provides an elegant relationship between the components of the transfer functions of the sub-systems and the coupled system, that reveals the symmetry and the structure of the coupled system. In addition it

shows that the relation between the transfer functions is not straightforward. Since several sub-expressions such as $(s\mathbf{E}_{ii} - A_{ii})^{-1}$ reoccur frequently, we will introduce abbreviations in the upcoming sections.

Formula (1.28) reveals a structure which is more difficult to find in (1.22)–(1.25) and can be used to calculate the transfer function of the coupled system if the transfer functions of the individual sub-systems are available. The involved inverse is of a small matrix which means that calculation of the transfer function of the coupled system is relatively cheap.

1.3.3 Standard Block Structure Preserving Reduction

In this section we will recall the general ideas of the standard block-structure preserving methods.

A typical block structure preserving (*BSP*) model reduction method applied to the system (1.10) consists of the following three steps:

1. Create the matrix $\tilde{\mathbf{V}}$ whose columns span the nth Krylov subspace around $s_0 \in \mathbb{C}$

$$\tilde{\mathbf{V}} = \mathscr{K}_n(\mathbf{P}(s_0), \mathbf{R}(s_0)),$$

where $\mathbf{P}(s_0)$ and $\mathbf{R}(s_0)$ are

$$\mathbf{P}(s_0) = (s_0\mathbf{E} - \mathbf{A})^{-1}\mathbf{E} \in \mathbb{R}^{N\times N} \quad \text{and} \quad \mathbf{R}(s_0) = (s_0\mathbf{E} - \mathbf{A})^{-1}\mathbf{B} \in \mathbb{R}^{N}.$$

2. Build a the block-diagonal reduction matrix $\mathbf{V}$ with $N_1 + N_2 = N$ rows

$$\mathbf{V} = \begin{bmatrix} \mathbf{V}_1 & \mathbf{0} \\ \mathbf{0} & \mathbf{V}_2 \end{bmatrix},$$

where $\mathbf{V}_1$ and $\mathbf{V}_2$ contain the first N_1 respectively last N_2 rows of the matrix $\tilde{\mathbf{V}}$.
3. Project the original system onto a lower-dimensional space

$$\hat{\mathbf{E}} = \mathbf{V}^T\mathbf{E}\mathbf{V}, \quad \hat{\mathbf{A}} = \mathbf{V}^T\mathbf{A}\mathbf{V}, \quad \hat{\mathbf{B}} = \mathbf{V}^T\mathbf{B}, \quad \hat{\mathbf{C}} = \mathbf{V}^T\mathbf{C}.$$

When possible we write $\mathbf{P}$ and $\mathbf{R}$ rather than $\mathbf{P}(s_0)$ respectively $\mathbf{R}(s_0)$. The model reduction methods based on this idea are widely applied and popular due to a good accuracy of the reduced-order systems that they deliver. However, they have a few drawbacks, one of them being the high cost of the construction of the reduction basis. The main computational cost of this type of methods is related to evaluation of $\mathbf{x} \mapsto (s_0\mathbf{E} - \mathbf{A})^{-1}\mathbf{x}$, which involves solving a system of equations with a large coefficient matrix. In the next section we introduce an alternative structure preserving method which for some cases can significantly reduce the computational costs.

1.3.4 Separate Bases Reduction Algorithm

In the classical case, the reduction basis is built using the coupled formulation of the system (1.10). The construction of this basis requires repeated evaluations of

$\mathbf{x} \mapsto (s_0\mathbf{E} - \mathbf{A})^{-1}\mathbf{x}$ where $s_0\mathbf{E} - \mathbf{A}$ is an $N \times N$ matrix. For large N this procedure can be computationally very expensive or even unfeasible. In such cases one can try to make use of a natural block structure of the coupled system and for instance replace the evaluations involving $(s_0\mathbf{E} - \mathbf{A})^{-1}$ by evaluations involving $(s_0\mathbf{E}_{11} - \mathbf{A}_{11})^{-1}$ and $(s_0\mathbf{E}_{22} - \mathbf{A}_{22})^{-1}$, i.e., by evaluations involving only the coefficient matrices of both sub-systems. If N is large and for instance $N_1 = N_2 = N/2$ then the serial computation of $(s_0\mathbf{E}_{11} - \mathbf{A}_{11})^{-1}$ and $(s_0\mathbf{E}_{22} - \mathbf{A}_{22})^{-1}$ may be much faster that of $(s_0\mathbf{E} - \mathbf{A})^{-1}$. Further acceleration can be achieved through parallelism.

Following this idea, we introduce a new model reduction algorithm, called Separate Bases Reduction (*SBR*) algorithm. Here the Krylov subspaces that create the reduction bases correspond to the uncoupled sub-systems (as defined in (1.7) and (1.8)) rather than to the coupled system (1.10). The procedure is as follows:

1. Create two matrices $\mathbf{V}_1$ and $\mathbf{V}_2$, one for each sub-system:
   - For the sub-system S_1, build a matrix $\mathbf{V}_1$, whose columns span the n_1th Krylov subspace around $s_0 \in \mathbb{C}$

     $$\mathbf{V}_1 = \mathscr{K}_{n_1}(\mathbf{P}_1(s_0), \mathbf{R}_1(s_0)),$$

     where $\mathbf{P}_1(s_0)$ and $\mathbf{R}_1(s_0)$ are

     $$\mathbf{P}_1(s_0) = (s_0\mathbf{E}_{11} - \mathbf{A}_{11})^{-1}\mathbf{E}_{11} \quad \text{and} \quad \mathbf{R}_1(s_0) = (s_0\mathbf{E}_{11} - \mathbf{A}_{11})^{-1}[\mathbf{B}_1\ \mathbf{B}_3].$$

     Matrix $\mathbf{V}_1$ has N_1 rows.
   - For the sub-system S_2, build a matrix $\mathbf{V}_2$, whose columns span the n_2th Krylov subspace around $s_0 \in \mathbb{C}$

     $$\mathbf{V}_2 = \mathscr{K}_{n_2}(\mathbf{P}_2(s_0), \mathbf{R}_2(s_0)),$$

     where $\mathbf{P}_2(s_0)$ and $\mathbf{R}_2(s_0)$ are

     $$\mathbf{P}_2(s_0) = (s_0\mathbf{E}_{22} - \mathbf{A}_{22})^{-1}\mathbf{E}_{22} \quad \text{and} \quad \mathbf{R}_2(s_0) = (s_0\mathbf{E}_{22} - \mathbf{A}_{22})^{-1}[\mathbf{B}_2\ \mathbf{B}_4].$$

     Matrix $\mathbf{V}_2$ has N_2 rows.
2. Build the block-diagonal reduction matrix $\mathbf{V}$ with $N_1 + N_2 = N$ rows

   $$\mathbf{V} = \begin{bmatrix} \mathbf{V}_1 & \mathbf{0} \\ \mathbf{0} & \mathbf{V}_2 \end{bmatrix}.$$

3. Project the original system onto a lower-dimensional space

   $$\hat{\mathbf{E}} = \mathbf{V}^T\mathbf{E}\mathbf{V}, \quad \hat{\mathbf{A}} = \mathbf{V}^T\mathbf{A}\mathbf{V}, \quad \hat{\mathbf{B}} = \mathbf{V}^T\mathbf{B}, \quad \hat{\mathbf{C}} = \mathbf{V}^T\mathbf{C}.$$

In the sequel, when possible without causing confusion, we omit the argument s_0 of $\mathbf{P}_i$ and $\mathbf{R}_i$, $i = 1, 2$. In the next subsection, we will compare the SBR algorithm with a standard BSP reduction method, by examining their most important properties.

1.3.5 Separate Bases Reduction Algorithm – Properties

In this subsection we will discuss the differences and similarities between Separate Bases Reduction algorithm and standard block structure preserving model reduction methods.

Block-Structure Preservation

As described in subsection 1.3.4, the SBR algorithm uses reduction matrices of the block-diagonal form

$$\mathbf{V} = \begin{bmatrix} \mathbf{V}_1 & \mathbf{0} \\ \mathbf{0} & \mathbf{V}_2 \end{bmatrix}.$$

Therefore, its application preserves the block structure of the coupled system matrices.

Rank and Orthogonality

The sub-blocks $\mathbf{V}_1$ and $\mathbf{V}_2$ of the projector $\mathbf{V}$ are constructed separately, using one of the Krylov basis building algorithms. Hence, both of them have a full column rank and, as a result, the matrix $\mathbf{V}$ also has a full column rank. If the sub-blocks $\mathbf{V}_1$ and $\mathbf{V}_2$ have orthogonal columns then also matrix $\mathbf{V}$ has (automatically) orthogonal columns, i.e., no explicit orthogonalization has to be applied.

Computational Cost

The difference between the computational costs for a standard block structure preserving method and the Separate Bases Reduction algorithm comes from the fact, that the SBR algorithm computes the reduction bases for the set of uncoupled systems instead of using the coupled formulation of the system. This approach can significantly reduce the computational time and storage requirements needed during the model reduction process.

The main cost of the Krylov basis construction lies in the evaluation of the matrix pencil inverse function $\mathbf{x} \mapsto (s_0\mathbf{E} - \mathbf{A})^{-1}\mathbf{x}$. For coupled models with many degrees n of freedom this evaluation may be unfeasible. But for sub-problems of smaller size evaluation may be possible. The amount of computational work required for the solution of $(s_0\mathbf{E} - \mathbf{A})\mathbf{x} = \mathbf{d}$ depends on the employed solution method which at its turn relies on specific properties of the matrix $s_0\mathbf{E} - \mathbf{A}$ (symmetry, monotone, positive definite, etc.). Different methods lead to different amounts of computational work: The minimal amount of work of $O(n)$ operations is usually achieved by multigrid methods (see [25]), other methods such as GMRES, PCG, CGS and BiCGstab(l) (see [16, 20, 22]) are more expensive. Classical fixed point methods such as Jacobi, Gauss-Seidel and matrix-splitting based methods are usually even slower.

Size of the Reduction Space

Another difference with respect to the standard BSP reduction methods is the size of the reduction matrix $\mathbf{V}$ and, as a result, dimension of the reduced order model.

Let us consider the coupled system (1.10) and assume, for simplicity, that there is no need for deflation (all columns turn out to be linearly independent) while building the matrix $\mathbf{V}$. We will apply a typical reduction procedure like described in subsection 1.3.3 and the SBR algorithm. In both cases, we will build a Krylov subspace of order n and estimate the size of the reduction space and reduced order model.

We begin with the analysis of the standard structure preserving algorithm. The nth Krylov subspace built for the coupled system for the starting matrices as defined in subsection 1.3.3 will be of the form

$$\tilde{\mathbf{V}} = \mathscr{K}_n\Big(\mathbf{P}, \mathbf{R}\Big) = \text{colspan}\{\mathbf{R}, \ldots, \mathbf{P}^{n-1}\mathbf{R}\}$$

where $\mathbf{P} = (s_0\mathbf{E} - \mathbf{A})^{-1}\mathbf{E}$ and $\mathbf{R} = (s_0\mathbf{E} - \mathbf{A})^{-1}\mathbf{B}$. Since $\mathbf{B} \in \mathbb{R}^{N\times m}$, each of the components $\mathbf{P}^j\mathbf{R}$ of the matrix $\tilde{\mathbf{V}}$ has m columns. Thus, for a degree n Krylov space, assuming no deflation, the size of $\tilde{\mathbf{V}}$ is $N \times (nm)$. Next, the block-diagonal reduction matrix $\mathbf{V}$ is created by splitting the rows of $\tilde{\mathbf{V}}$ according to the dimensions of the sub-problems. In our case, the coupled system consists of two sub-systems, so the final size of the reduction matrix $\mathbf{V}$ is $N \times (2nm)$. This leads to a reduced model of order $2nm$.

Next, we will focus on the *SBR* algorithm. In this case two matrices $\mathbf{V}_1$ and $\mathbf{V}_2$, are built separately and we assume that each of them corresponds to an nth degree Krylov subspace based on the appropriate matrices (for $i = 1,2$ define $\mathbf{G}_i(s_0) = (s_0\mathbf{E}_{ii} - \mathbf{A}_{ii})^{-1}$, $\mathbf{P}_i(s_0) = \mathbf{G}_i\mathbf{E}_{ii}$ and $\mathbf{R}_i(s_0) = \mathbf{G}_i[\mathbf{B}_i\ \mathbf{B}_{2+i}]$ and observe that $\mathbf{R}_i = [\mathbf{R}_{i1}, \mathbf{R}_{i2}]$ where $\mathbf{R}_{i1}$ and $\mathbf{R}_{i2}$ are $\mathbf{G}_i\mathbf{B}_i$, respectively $\mathbf{G}_i\mathbf{B}_{2+i}$). For the sub-system S_1, we create the matrix $\mathbf{V}_1$

$$\mathbf{V}_1 = \mathscr{K}_n\Big(\mathbf{P}_1,\ \mathbf{R}_1\Big).$$

Here, $\mathbf{R}_1, [\mathbf{B}_1\ \mathbf{B}_3] \in \mathbb{R}^{N_1\times(m_1+m_3)}$, so each component $\mathbf{P}_1^j\mathbf{R}_1$ of the matrix $\mathbf{V}_1$ has (m_1+m_3) columns whence $\mathbf{V}_1$ has $n\times(m_1+m_3)$ columns.
For the sub-system S_2, we create

$$\mathbf{V}_2 = \mathscr{K}_n\Big(\mathbf{P}_2,\ \mathbf{R}_2\Big).$$

Similarly, since $\mathbf{R}_2, [\mathbf{B}_2\ \mathbf{B}_4] \in \mathbb{R}^{N_2\times(m_2+m_4)}$, every component $\mathbf{P}_2^j\mathbf{R}_2$ of the matrix $\mathbf{V}_2$ has (m_2+m_4) columns, and matrix $\mathbf{V}_2$ has $n\times(m_2+m_4)$ columns.

Next, matrices $\mathbf{V}_1$ and $\mathbf{V}_2$ are used as diagonal blocks of the reduction matrix $\mathbf{V}$, resulting in a reduced model of order

$$n\times(m_1+m_3) + n\times(m_2+m_4) = n\times(m+m_3+m_4).$$

This result shows that the SBR algorithm creates a smaller reduced order model than standard BSP methods if $(m_3+m_4) < m$. This is for instance the case for coupled systems for which the number of internal inputs is not larger than the number of

external inputs. If there are many more internal inputs than external ones, the size of the SBR algorithm based reduction matrix will grow very fast compared to the size of the BSP reduction matrix. However, this problem can be avoided for the category of systems for which the internal input matrices $\mathbf{B}_2$ and $\mathbf{B}_4$ can be approximated by only a small number of dominant components. This approach will be explained in more detail in the next section.

The Moment Matching Property

In order to assess the *SBR moment matching* properties we compare the column-spaces of the BPS and SBR reduction matrices. For simplicity, without loss of generalization, we focus on the *SISO* case (the coupled system is SISO) where in addition $\mathbf{B}_i$, $\mathbf{C}_i$, $i = 1,\ldots,4$ related to the sub-systems are column-vectors which implies that all products $\mathbf{C}_i^T(\ldots)\mathbf{B}_j$, $i,j = 1,\ldots,4$, are scalars. A similar analysis is possible for the *MIMO* case (a MIMO coupled system with sub-system matrices $\mathbf{B}_i$, $\mathbf{C}_i$).

Theorem 1.2 *Let the coupled system be as in Fig. 1.7, described by* (1.7) *and* (1.8). *Assume that all inputs and outputs are column-vectors, i.e.,* $m_i = p_i = 1$, $i = 1,2,3,4$. *Then the SBR reduced-order model transfer function matches at least the same (number of) moments as the BSP reduced-order model transfer function.*

Proof First, we examine the reduction space built by a standard BSP method. To match the first k moments at $s_0 \in \mathbb{C}$, of the coupled system of the form (1.10), one has to construct the Krylov space

$$\tilde{\mathbf{V}} = \mathscr{K}_k(\mathbf{P},\mathbf{R}),$$

where

$$\mathbf{P} = (s\mathbf{E} - \mathbf{A})^{-1}\mathbf{E} \quad \text{and} \quad \mathbf{R} = (s\mathbf{E} - \mathbf{A})^{-1}\mathbf{B}.$$

The i^{th} Krylov step for the BSP method adds to the reduction basis the column span of the following matrix $\mathbf{V}_{\text{BSP}}^{(i)}$

$$\mathbf{V}_{\text{BSP}}^{(i)} = \begin{bmatrix} \mathbf{V}_{11}^{(i)} & \mathbf{V}_{12}^{(i)} & 0 & 0 \\ 0 & 0 & \mathbf{V}_{21}^{(i)} & \mathbf{V}_{22}^{(i)} \end{bmatrix} \tag{1.29}$$

with blocks of the form

$$\begin{aligned} \tilde{\mathbf{V}}_{\text{BSP}}^{(i)} &= \begin{bmatrix} \mathbf{V}_{11}^{(i)} & \mathbf{V}_{12}^{(i)} \\ \mathbf{V}_{21}^{(i)} & \mathbf{V}_{22}^{(i)} \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{P}_1^{i-1}\mathbf{R}_{11} + \sum_{j=0}^{i-1} \alpha_j \mathbf{P}_1^j \mathbf{R}_{12} & \sum_{j=0}^{i-1} \gamma_j \mathbf{P}_1^j \mathbf{R}_{12} \\ \sum_{j=0}^{i-1} \beta_j \mathbf{P}_2^j \mathbf{R}_{22} & \mathbf{P}_2^{i-1}\mathbf{R}_{21} + \sum_{j=0}^{i-1} \delta_j \mathbf{P}_2^j \mathbf{R}_{22} \end{bmatrix}. \end{aligned} \tag{1.30}$$

By (1.27) there exist scalars a, b, c, d and by construction (induction) there exist

coefficient vectors $\alpha = [\alpha_1, \ldots, \alpha_j] \in \mathbb{R}^{i-2}$, $\beta, \gamma, \delta \in \mathbb{R}^{i-2}$ such that

$$
\begin{aligned}
\tilde{\mathbf{V}}_{\mathrm{BSP}}^{(i)} &= \mathbf{P}\tilde{\mathbf{V}}_{\mathrm{BSP}}^{(i-1)} \\
&= \left(s \begin{bmatrix} \mathbf{E}_{11} & \mathbf{0} \\ \mathbf{0} & \mathbf{E}_{22} \end{bmatrix} - \begin{bmatrix} \mathbf{A}_{11} & \mathbf{B}_3\mathbf{C}_4^T \\ \mathbf{B}_4\mathbf{C}_3^T & \mathbf{A}_{22} \end{bmatrix} \right)^{-1} \begin{bmatrix} \mathbf{E}_{11} & \mathbf{0} \\ \mathbf{0} & \mathbf{E}_{22} \end{bmatrix} \tilde{\mathbf{V}}_{\mathrm{BSP}}^{(i-1)} \\
&\underset{\text{induction}}{\overset{(1.27)}{=}} \left(\begin{bmatrix} \mathbf{G}_1 & \mathbf{0} \\ \mathbf{0} & \mathbf{G}_2 \end{bmatrix} + \begin{bmatrix} \mathbf{0} & \mathbf{R}_{12} \\ \mathbf{R}_{22} & \mathbf{0} \end{bmatrix} \begin{bmatrix} a & b \\ c & d \end{bmatrix} \begin{bmatrix} \mathbf{C}_3^T\mathbf{G}_1 & \mathbf{0} \\ \mathbf{0} & \mathbf{C}_4^T\mathbf{G}_2 \end{bmatrix} \right) \begin{bmatrix} \mathbf{E}_{11} & \mathbf{0} \\ \mathbf{0} & \mathbf{E}_{22} \end{bmatrix} \tilde{\mathbf{V}}_{\mathrm{BSP}}^{(i-1)} \\
&= \left(\begin{bmatrix} \mathbf{P}_1 & \mathbf{0} \\ \mathbf{0} & \mathbf{P}_2 \end{bmatrix} + \begin{bmatrix} \mathbf{0} & \mathbf{R}_{12} \\ \mathbf{R}_{22} & \mathbf{0} \end{bmatrix} \begin{bmatrix} a & b \\ c & d \end{bmatrix} \begin{bmatrix} \mathbf{C}_3^T\mathbf{P}_1 & \mathbf{0} \\ \mathbf{0} & \mathbf{C}_4^T\mathbf{P}_2 \end{bmatrix} \right) \\
&\quad \begin{bmatrix} \mathbf{P}_1^{i-2}\mathbf{R}_{11} + \sum_{j=0}^{i-2} \alpha_j \mathbf{P}_1^j \mathbf{R}_{12} & \sum_{j=0}^{i-2} \gamma_j \mathbf{P}_1^j \mathbf{R}_{12} \\ \sum_{j=0}^{i-2} \beta_j \mathbf{P}_2^j \mathbf{R}_{22} & \mathbf{P}_2^{i-2}\mathbf{R}_{21} + \sum_{j=0}^{i-2} \delta_j \mathbf{P}_2^j \mathbf{R}_{22} \end{bmatrix} \\
&= \begin{bmatrix} \mathbf{P}_1^{i-1}\mathbf{R}_{11} + \sum_{j=1}^{i-1} \alpha_j \mathbf{P}_1^j \mathbf{R}_{12} & \sum_{j=1}^{i-1} \gamma_j \mathbf{P}_1^j \mathbf{R}_{12} \\ \sum_{j=1}^{i-1} \beta_j \mathbf{P}_2^j \mathbf{R}_{22} & \mathbf{P}_2^{i-1}\mathbf{R}_{21} + \sum_{j=1}^{i-1} \delta_j \mathbf{P}_2^j \mathbf{R}_{22} \end{bmatrix} + \\
&\quad \begin{bmatrix} \mathbf{0} & \mathbf{R}_{12} \\ \mathbf{R}_{22} & \mathbf{0} \end{bmatrix} \begin{bmatrix} a & b \\ c & d \end{bmatrix} \begin{bmatrix} \mathbf{C}_3^T\mathbf{P}_1 & \mathbf{0} \\ \mathbf{0} & \mathbf{C}_4^T\mathbf{P}_2 \end{bmatrix} \\
&\quad \begin{bmatrix} \mathbf{P}_1^{i-2}\mathbf{R}_{11} + \sum_{j=0}^{i-2} \alpha_j \mathbf{P}_1^j \mathbf{R}_{12} & \sum_{j=0}^{i-2} \gamma_j \mathbf{P}_1^j \mathbf{R}_{12} \\ \sum_{j=0}^{i-2} \beta_j \mathbf{P}_2^j \mathbf{R}_{22} & \mathbf{P}_2^{i-2}\mathbf{R}_{21} + \sum_{j=0}^{i-2} \delta_j \mathbf{P}_2^j \mathbf{R}_{22} \end{bmatrix} \\
&\underset{\mathbf{C}_i^T(\ldots)\mathbf{B}_j \in \mathbb{C}}{=} \begin{bmatrix} \mathbf{P}_1^{i-1}\mathbf{R}_{11} + \sum_{j=1}^{i-1} \alpha_j \mathbf{P}_1^j \mathbf{R}_{12} & \sum_{j=1}^{i-1} \gamma_j \mathbf{P}_1^j \mathbf{R}_{12} \\ \sum_{j=1}^{i-1} \beta_j \mathbf{P}_2^j \mathbf{R}_{22} & \mathbf{P}_2^{i-1}\mathbf{R}_{21} + \sum_{j=1}^{i-1} \delta_j \mathbf{P}_2^j \mathbf{R}_{22} \end{bmatrix} + \\
&\quad \begin{bmatrix} \mathbf{0} & \mathbf{R}_{12} \\ \mathbf{R}_{22} & \mathbf{0} \end{bmatrix} \begin{bmatrix} a & b \\ c & d \end{bmatrix} \begin{bmatrix} * & * \\ * & * \end{bmatrix} \\
&= \begin{bmatrix} \mathbf{P}_1^{i-1}\mathbf{R}_{11} + \sum_{j=1}^{i-1} \alpha_j \mathbf{P}_1^j \mathbf{R}_{12} & \sum_{j=1}^{i-1} \gamma_j \mathbf{P}_1^j \mathbf{R}_{12} \\ \sum_{j=1}^{i-1} \beta_j \mathbf{P}_2^j \mathbf{R}_{22} & \mathbf{P}_2^{i-1}\mathbf{R}_{21} + \sum_{j=1}^{i-1} \delta_j \mathbf{P}_2^j \mathbf{R}_{22} \end{bmatrix} + \\
&\quad \begin{bmatrix} \mathbf{0} & \mathbf{R}_{12} \\ \mathbf{R}_{22} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mu_1 & \mu_2 \\ \mu_3 & \mu_4 \end{bmatrix} \\
&= \begin{bmatrix} \mathbf{P}_1^{i-1}\mathbf{R}_{11} + \sum_{j=1}^{i-1} \alpha_j \mathbf{P}_1^j \mathbf{R}_{12} & \sum_{j=1}^{i-1} \gamma_j \mathbf{P}_1^j \mathbf{R}_{12} \\ \sum_{j=1}^{i-1} \beta_j \mathbf{P}_2^j \mathbf{R}_{22} & \mathbf{P}_2^{i-1}\mathbf{R}_{21} + \sum_{j=1}^{i-1} \delta_j \mathbf{P}_2^j \mathbf{R}_{22} \end{bmatrix} + \\
&\quad \begin{bmatrix} \mu_1\mathbf{R}_{12} & \mu_2\mathbf{R}_{12} \\ \mu_3\mathbf{R}_{22} & \mu_4\mathbf{R}_{22} \end{bmatrix} \\
&= \begin{bmatrix} \mathbf{P}_1^{i-1}\mathbf{R}_{11} + \sum_{j=0}^{i-1} \hat{\alpha}_j \mathbf{P}_1^j \mathbf{R}_{12} & \sum_{j=0}^{i-1} \hat{\gamma}_j \mathbf{P}_1^j \mathbf{R}_{12} \\ \sum_{j=0}^{i-1} \hat{\beta}_j \mathbf{P}_2^j \mathbf{R}_{22} & \mathbf{P}_2^{i-1}\mathbf{R}_{21} + \sum_{j=0}^{i-1} \hat{\delta}_j \mathbf{P}_2^j \mathbf{R}_{22} \end{bmatrix}
\end{aligned} \tag{1.31}
$$

where $\hat{\alpha} = [\mu_1, \alpha]$, $\hat{\beta} = [\mu_3, \beta]$, $\hat{\gamma} = [\mu_2, \gamma]$, $\hat{\delta} = [\mu_4, \delta]$, and the matrix with '*' is a full matrix. Now it is easy to see that the column span of the matrix constructed from the matrix $\tilde{\mathbf{V}}_{\mathrm{BSP}}^{(i)}$ by splitting its rows, has the same column span as the matrix defined in (1.29). Finally, the reduction basis $\mathbf{V}_{\mathrm{BSP}}$ after k steps of the BSP algorithm has the following form

$$\mathbf{V}_{\mathrm{BSP}} = [\mathbf{V}_{\mathrm{BSP}}^{(1)}, \ldots, \mathbf{V}_{\mathrm{BSP}}^{(k)}]. \tag{1.32}$$

Now we will examine the SBR reduction space algorithm. Let $\mathbf{P}_i, \mathbf{R}_i = [\mathbf{R}_{i1}, \mathbf{R}_{i2}]$, $i = 1,2$ be as defined before. For $s \in \mathbb{C}$ SBR builds two Krylov subspaces

$$\mathbf{V}_1 = \mathscr{K}_k(\mathbf{P}_1, \mathbf{R}_1), \quad \text{and} \quad \mathbf{V}_2 = \mathscr{K}_k(\mathbf{P}_2, \mathbf{R}_2).$$

One can easily prove, that the i^{th} step of the Krylov iteration within the SBR algorithm adds to the reduction basis the column span of the following matrix $\mathbf{V}_{\mathrm{SBR}}^{(i)}$

$$\mathbf{V}_{\mathrm{SBR}}^{(i)} = \begin{bmatrix} \mathbf{V}_1^{(i)} & 0 \\ 0 & \mathbf{V}_2^{(i)} \end{bmatrix}, \tag{1.33}$$

where

$$\mathbf{V}_1^{(i)} = \left[\mathbf{P}_1^{i-1}\mathbf{R}_{11}, \mathbf{P}_1^{i-1}\mathbf{R}_{12}\right]$$

and

$$\mathbf{V}_2^{(i)} = \left[\mathbf{P}_2^{i-1}\mathbf{R}_{21}, \mathbf{P}_2^{i-1}\mathbf{R}_{22}\right].$$

Finally, the reduction basis $\mathbf{V}_{\mathrm{SBR}}$ after k steps of the SBR algorithm has the following form

$$\mathbf{V}_{\mathrm{SBR}} = [\mathbf{V}_{\mathrm{SBR}}^{(1)}, \ldots, \mathbf{V}_{\mathrm{SBR}}^{(k)}]. \tag{1.34}$$

Comparing (1.30) and (1.33), we observe that

$$\mathrm{colspan}\mathbf{V}_{\mathrm{BSP}} \subset \mathrm{colspan}\mathbf{V}_{\mathrm{SBR}}.$$

Because the dimensions of the spaces are equal for our case (SISO external and column-vectors $\mathbf{B}_i, \mathbf{C}_i$ for the sub-systems) one finds that in addition

$$\mathrm{colspan}\mathbf{V}_{\mathrm{BSP}} = \mathrm{colspan}\mathbf{V}_{\mathrm{SBR}}. \tag{1.35}$$

Because $\mathrm{colspan}\mathbf{V}_{\mathrm{BSP}} \subset \mathrm{colspan}\mathbf{V}_{\mathrm{SBR}}$ the SBR reduced-order model transfer function matches (at least) the same (number of) moments as the BSP reduced-order model transfer function which at its turn (Theorem 2, [6]) matches the same (number of) moments as the original coupled system's transfer function. For the more general case where $\mathbf{B}_i$, $\mathbf{C}_i$, $i = 1, \ldots, 4$ are matrices one should also obtain

$$\mathrm{colspan}\mathbf{V}_{\mathrm{BSP}} \subseteq \mathrm{colspan}\mathbf{V}_{\mathrm{SBR}} \tag{1.36}$$

which is sufficient to prove the moment matching property of the SBR reduced-order system.

1.3.6 Two-Sided Separate Bases Reduction Algorithm

The two-sided projection technique introduced in the previous section can be adapted to similarly improve the moment matching properties of the SBR algorithm, where we assume, as in the previous section, thath the B_i and C_i are column vectors. With the uncoupled formulation (1.7) and (1.8) in mind we define the reduction algorithm as follows.

1. For the sub-system S_1, create two matrices:
   - Matrix $\mathbf{V}_1$, whose columns span the n_1th Krylov subspace around $s_0 \in \mathbb{C}$

     $$\mathbf{V}_1 = \mathscr{K}_{n_1}(\mathbf{P}_1(s_0), \mathbf{R}_1(s_0)),$$

     where $\mathbf{P}_1(s_0)$ and $\mathbf{R}_1(s_0)$ are

     $$\mathbf{P}_1(s_0) = (s_0\mathbf{E}_{11} - \mathbf{A}_{11})^{-1}\mathbf{E}_{11} \quad \text{and} \quad \mathbf{R}_1(s_0) = (s_0\mathbf{E}_{11} - \mathbf{A}_{11})^{-1}[\mathbf{B}_1\ \mathbf{B}_3].$$

     Matrix $\mathbf{V}_1$ has N_1 rows.
   - Matrix $\mathbf{W}_1$, whose columns span the n_1th Krylov subspace around $s_0 \in \mathbb{C}$

     $$\mathbf{W}_1 = \mathscr{K}_{n_1}(\mathbf{S}_1(s_0), \mathbf{T}_1(s_0)),$$

     where $\mathbf{S}_1(s_0)$ and $\mathbf{T}_1(s_0)$ are

     $$\mathbf{S}_1(s_0) = (s_0\mathbf{E}_{11} - \mathbf{A}_{11})^{-T}\mathbf{E}_{11}^T \quad \text{and} \quad \mathbf{T}_1(s_0) = (s_0\mathbf{E}_{11} - \mathbf{A}_{11})^{-T}[\mathbf{C}_1\ \mathbf{C}_3].$$

     Matrix $\mathbf{W}_1$ has N_1 rows.
2. For the sub-system S_2, create two matrices:
   - Matrix $\mathbf{V}_2$, whose columns span the n_2th Krylov subspace around $s_0 \in \mathbb{C}$

     $$\mathbf{V}_2 = \mathscr{K}_{n_2}(\mathbf{P}_2(s_0), \mathbf{R}_2(s_0)),$$

     where $\mathbf{P}_2(s_0)$ and $\mathbf{R}_2(s_0)$ are

     $$\mathbf{P}_2(s_0) = (s_0\mathbf{E}_{22} - \mathbf{A}_{22})^{-1}\mathbf{E}_{22} \quad \text{and} \quad \mathbf{R}_2(s_0) = (s_0\mathbf{E}_{22} - \mathbf{A}_{22})^{-1}[\mathbf{B}_2\ \mathbf{B}_4].$$

     Matrix $\mathbf{V}_2$ has N_2 rows.
   - Matrix $\mathbf{W}_2$, whose columns span the n_2th Krylov subspace around $s_0 \in \mathbb{C}$

     $$\mathbf{W}_2 = \mathscr{K}_{n_2}(\mathbf{S}_2(s_0), \mathbf{T}_2(s_0)),$$

     where $\mathbf{S}_2(s_0)$ and $\mathbf{T}_2(s_0)$ are

     $$\mathbf{S}_2(s_0) = (s_0\mathbf{E}_{22} - \mathbf{A}_{22})^{-T}\mathbf{E}_{22}^T \quad \text{and} \quad \mathbf{T}_2(s_0) = (s_0\mathbf{E}_{22} - \mathbf{A}_{22})^{-T}[\mathbf{C}_2\ \mathbf{C}_4].$$

     Matrix $\mathbf{W}_2$ has N_2 rows.

3. Build two block-diagonal reduction matrices $\mathbf{V}$ and $\mathbf{W}$ with $N_1 + N_2 = N$ rows

$$\mathbf{V} = \begin{bmatrix} \mathbf{V}_1 & \mathbf{0} \\ \mathbf{0} & \mathbf{V}_2 \end{bmatrix} \qquad \text{and} \qquad \mathbf{W} = \begin{bmatrix} \mathbf{W}_1 & \mathbf{0} \\ \mathbf{0} & \mathbf{W}_2 \end{bmatrix}.$$

4. Project the original system onto the lower-dimensional space

$$\hat{\mathbf{E}}_{\text{SBR}} = \mathbf{W}^T\mathbf{E}\mathbf{V}, \quad \hat{\mathbf{A}}_{\text{SBR}} = \mathbf{W}^T\mathbf{A}\mathbf{V}, \quad \hat{\mathbf{B}}_{\text{SBR}} = \mathbf{W}^T\mathbf{B}, \quad \hat{\mathbf{C}}_{\text{SBR}} = \mathbf{V}^T\mathbf{C}.$$

Again, different algorithms lead to $\mathbf{V}_1$, $\mathbf{V}_2$ and $\mathbf{W}_1$, $\mathbf{W}_2$ with different properties. Also the above SBR algorithm results in a block-structured reduced order system and uses all of the inputs *and* outputs. Consequently, also the above SBR-based reduced order system's transfer function matches twice as many moments of the original system's transfer function as the only inputs based one in Sect. 1.3.3 (the moment matching property follows from the BSP algorithm, Theorem 1 and Theorem 2).

1.4 Low-Rank Approximations Based SBR Algorithm

In Sect. 1.3 we presented the Separate Bases Reduction algorithm – a block-structure preserving model reduction method for coupled systems. As discussed in that section, one of the SBR method's disadvantages is that the sizes of the its Krylov subspaces increase very fast for systems with a large number of internal inputs and outputs. Hence, the use of the SBR algorithm was recommended for the cases, in which the number of internal inputs and outputs was considerably smaller than the dimension of the system or comparable to the number of the external inputs and outputs. In this section, we approximate the internal inputs (outputs) by their GSVD-based dominant parts. This improves the efficiency of the SBR method. In addition we will prove that both the SBR algorithm and its low-rank based variant can be applied to coupled systems for which the internal input and output operators $\mathbf{B}$ and $\mathbf{C}$ are not explicitly availabe.

1.4.1 Implicitly Defined Couplings

In Sect. 1.3, we introduced the interconnected system (1.10) as a result of the coupling of the two sub-systems, (1.7) and (1.8). Here, the coupling blocks are given by the explicit products of the internal inputs and outputs of the two sub-systems, namely $\mathbf{B}_3\mathbf{C}_4^T$ and $\mathbf{B}_4\mathbf{C}_3^T$. Having such a formulation at our disposal, we can apply the SBR algorithm in a straightforward way. However, for some applications it may be impossible to obtain matrices $\mathbf{B}_3$, $\mathbf{B}_4$, $\mathbf{C}_3$ and $\mathbf{C}_4$. In the following sections we propose a way of transforming an interconnected system with *implicitly defined*

couplings of a form

$$S: \begin{cases} s\begin{bmatrix} \mathbf{E}_{11} & \mathbf{0} \\ \mathbf{0} & \mathbf{E}_{22} \end{bmatrix}\begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \end{bmatrix} = \begin{bmatrix} \mathbf{A}_{11} & \mathbf{A}_{12} \\ \mathbf{A}_{21} & \mathbf{A}_{22} \end{bmatrix}\begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \end{bmatrix} + \begin{bmatrix} \mathbf{B}_1 & \mathbf{0} \\ \mathbf{0} & \mathbf{B}_2 \end{bmatrix}\begin{bmatrix} \mathbf{u}_1 \\ \mathbf{u}_2 \end{bmatrix} \\ \begin{bmatrix} \mathbf{y}_1 \\ \mathbf{y}_2 \end{bmatrix} = \begin{bmatrix} \mathbf{C}_1^T & \mathbf{0} \\ \mathbf{0} & \mathbf{C}_2^T \end{bmatrix}\begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \end{bmatrix} \end{cases}, \tag{1.37}$$

with

$$\mathbf{E} = \begin{bmatrix} \mathbf{E}_{11} & \mathbf{0} \\ \mathbf{0} & \mathbf{E}_{22} \end{bmatrix}, \quad \mathbf{A} = \begin{bmatrix} \mathbf{A}_{11} & \mathbf{A}_{12} \\ \mathbf{A}_{21} & \mathbf{A}_{22} \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} \mathbf{B}_1 & \mathbf{0} \\ \mathbf{0} & \mathbf{B}_2 \end{bmatrix}, \quad \mathbf{C} = \begin{bmatrix} \mathbf{C}_1 & \mathbf{0} \\ \mathbf{0} & \mathbf{C}_2 \end{bmatrix} \tag{1.38}$$

into a form that can be reduced using the SBR algorithm. Our goal is to find decompositions (factorizations) of the two coupling blocks

$$\mathbf{A}_{12} = \tilde{\mathbf{B}}_3\tilde{\mathbf{C}}_4^T \qquad \text{and} \qquad \mathbf{A}_{21} = \tilde{\mathbf{B}}_4\tilde{\mathbf{C}}_3^T \tag{1.39}$$

that provide a good (with respect to the corresponding Krylov subspaces) approximation of the original internal inputs and outputs of the coupled system (1.37). A factorization of the type $\mathbf{A} = \mathbf{BC}$ is not be unique. The next section shows how to deal with this.

1.4.2 Decomposition Theorem

In this section, related to (1.37), first we show that a factorization $\mathbf{A} = \mathbf{BC}$ is not unique and next we prove that if $\mathbf{A}_{12} = \mathbf{B}_1\mathbf{C}_1$ and simultaneously $\mathbf{A}_{12} = \mathbf{B}_2\mathbf{C}_2$ then $\mathscr{K}_p(\mathbf{A}_{11}, \mathbf{B}_1) = \mathscr{K}_p(\mathbf{A}_{11}, \mathbf{B}_2)$ if $\mathbf{C}_1$ and $\mathbf{C}_2$ are of *full column rank*. The proofs will be for the input-based Krylov subspaces. Similar theory applies to the output-based Krylov subspaces.

First, a factorization of the type $\mathbf{A} = \mathbf{BC}$ is not unique since $\mathbf{A} = \mathbf{IA}$ and $\mathbf{A} = \mathbf{AI}$ are two different factorizations. Even a QR factorization $\mathbf{A} = \mathbf{QR}$ is not unique since if $\mathbf{A} = \mathbf{QR}$ then $\mathbf{A} = (\mathbf{Q}\overline{\mathbf{S}})(\mathbf{SR})$ for all complex valued diagonal matrices $\mathbf{S}$ with unit-length diagonal elements ($\overline{\mathbf{S}}$ denotes the complex conjugate of $\mathbf{S}$). Also other factorizations such as Gaussian-elimination based $\mathbf{A} = \mathbf{LU}$ exist.

Since we aim at the use of $\mathbf{B}$ for the generation of a Krylov subspace $\mathscr{K}_p(\mathbf{A}_{11}, \mathbf{B}_1)$ we will next show that the non-uniqueness does not need to be an issue. To this end we prove the following Lemma 1.1 and Theorem 1.3.

Lemma 1.1 *Let* $\mathbf{B} \in \mathbb{R}^{n\times p}$, $\mathbf{C} \in \mathbb{R}^{p\times m}$, $m, n, p \in \mathbb{N}$. *Then*

$$\text{rank}(\mathbf{C}) = p \implies \textit{colspan}\,\mathbf{BC} = \textit{colspan}\mathbf{B}.$$

Proof Matrix $\mathbf{C}$ has rank p which implies $p \leq m$ and that $\mathbf{C}$ has p linearly independent columns of length p. Thus based on

$$\text{colspan}\mathbf{C} = \{\mathbf{Cx}\colon \mathbf{x} \in \mathbb{R}^m\} \tag{1.40}$$

one finds

$$\operatorname{colspan}\mathbf{C} \underset{(1.40)}{=} \{\mathbf{C}\mathbf{x}\colon \mathbf{x} \in \mathbb{R}^m\} \underset{p\leq m}{=} \mathbb{R}^p \tag{1.41}$$

whence

$$\operatorname{colspan}\mathbf{B}\mathbf{C} \underset{(1.40)}{=} \{\mathbf{B}\mathbf{C}\mathbf{x}\colon \mathbf{x} \in \mathbb{R}^m\} \underset{(1.41)}{=} \{\mathbf{B}\mathbf{y}\colon \mathbf{y} \in \mathbb{R}^p\} \underset{(1.40)}{=} \operatorname{colspan}\mathbf{B}.$$

Note: The condition that $\mathbf{C}$ has full column rank is sufficient but not necessary. It can be relaxed: If for instance $\mathbf{B}$ has only $2 \leq p$ linearly independent columns, e.g. the ith and the jth column, then a sufficient condition is $\operatorname{colspan}\mathbf{C} = \operatorname{colspan}\{\mathbf{e}_i, \mathbf{e}_j\} \subset \mathbb{R}^p$.

Theorem 1.3 *Let* $\mathbf{B}_1, \mathbf{B}_2 \in \mathbb{R}^{n\times p}$, $\mathbf{C}_1, \mathbf{C}_2 \in \mathbb{R}^{p\times m}$ *and* $m, n, p \in \mathbb{N}$.
If

$$\operatorname{rank}(\mathbf{C}_1) = \operatorname{rank}(\mathbf{C}_2) = p \quad \textit{and} \quad \mathbf{B}_1\mathbf{C}_1 = \mathbf{B}_2\mathbf{C}_2$$

then

$$\mathit{colspan}\mathbf{B}_1 = \mathit{colspan}\mathbf{B}_2.$$

Proof Observe that

$$\operatorname{colspan}\mathbf{B}_1 \underset{\text{Lem.1.1}}{=} \operatorname{colspan}\mathbf{B}_1\mathbf{C}_1 = \operatorname{colspan}\mathbf{B}_2\mathbf{C}_2 \underset{\text{Lem.1.1}}{=} \operatorname{colspan}\mathbf{B}_2.$$

Next we prove that certain Krylov subspaces are identical.

Theorem 1.4 *Let* $\mathbf{A} \in \mathbb{R}^{n\times n}$ *is non-singular and* $\mathbf{B}_1, \mathbf{B}_2 \in \mathbb{R}^{n\times m}$, $n, m \in \mathbb{N}$. *Then*

$$\mathit{colspan}\mathbf{B}_1 = \mathit{colspan}\mathbf{B}_2 \implies \mathscr{K}_p(\mathbf{A}, \mathbf{B}_1) = \mathscr{K}_p(\mathbf{A}, \mathbf{B}_2).$$

Proof Note that

$$\begin{aligned}
\operatorname{colspan}\mathbf{B}_1 &= \operatorname{colspan}\mathbf{B}_2 \iff \\
\{\mathbf{B}_1\mathbf{x}\colon \mathbf{x} \in \mathbb{R}^m\} &= \{\mathbf{B}_2\mathbf{x}\colon \mathbf{x} \in \mathbb{R}^m\} \iff \\
\{\mathbf{A}\mathbf{B}_1\mathbf{x}\colon \mathbf{x} \in \mathbb{R}^m\} &= \{\mathbf{A}\mathbf{B}_2\mathbf{x}\colon \mathbf{x} \in \mathbb{R}^m\} \iff \\
\operatorname{colspan}\mathbf{A}\mathbf{B}_1 &= \operatorname{colspan}\mathbf{A}\mathbf{B}_2,
\end{aligned}$$

which, repeatedly applied, shows that $\operatorname{colspan}\mathbf{A}^k\mathbf{B}_1 = \operatorname{colspan}\mathbf{A}^k\mathbf{B}_2$ for all $k \geq 0$ whence $\mathscr{K}_p(\mathbf{A}, \mathbf{B}_1) = \mathscr{K}_p(\mathbf{A}, \mathbf{B}_2)$.

Theorem 1.3 in combination with Theorem 1.4 show that every factorization of an off-diagonal block of the form $\mathbf{A}_{12} = \mathbf{B}\mathbf{C}^{\mathrm{T}}$ with $\mathbf{C}$ of *full column rank* leads to the same krylov space $\mathscr{K}_p(\mathbf{A}_{11}, \mathbf{B})$. The following sections show how to use this property for the application of the SBR method to an arbitrary coupled system (1.37).

1.4.3 Decomposition Theorem – Numerical Example

In Sect. 1.4.2 we showed that the Krylov space does not depend on the factors of the decomposition $\mathbf{A}_{12} = \mathbf{B}\mathbf{C}^{\mathrm{T}}$ when these factors are of maximal column rank. To illustrate this numerically, we calculate these factors of $\mathbf{A}_{12}$ with different factorization

techniques, based on a *QR factoration* and *LU factorization*. For simplicity, we use a one-sided variant of the SBR method. The system used for the test is a linear beam coupled to a controller. Only the beam system has an external input and external output. Hence, the considered system is of a form

$$S: \begin{cases} s\begin{bmatrix} \mathbf{I}_{11} & \mathbf{0} \\ \mathbf{0} & \mathbf{I}_{22} \end{bmatrix}\begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \end{bmatrix} = \begin{bmatrix} \mathbf{A}_{11} & \mathbf{B}_3\mathbf{C}_4^T \\ \mathbf{B}_4\mathbf{C}_3^T & \mathbf{A}_{22} \end{bmatrix}\begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \end{bmatrix} + \begin{bmatrix} \mathbf{B}_1 \\ \mathbf{0} \end{bmatrix}\mathbf{u}_1 \\ \mathbf{y}_1 = [\mathbf{C}_1^T\ \mathbf{0}]\begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \end{bmatrix}. \end{cases} \tag{1.42}$$

Let $\mathbf{A}_{12} = \mathbf{B}_3\mathbf{C}_4^T$ and $\mathbf{A}_{21} = \mathbf{B}_4\mathbf{C}_3^T$. Here, the full coupled system has 80 degrees of freedom, 40 for each sub-system. Both of the sub-systems have 5 internal inputs and 5 internal outputs. It means, that the coupling blocks $\mathbf{A}_{12}$ and $\mathbf{A}_{21}$ are of rank 5. For all cases, the same number of Krylov iterations is performed and the reduced-order systems are of the order 55 (originally 80). The first sub-system was reduced from order 40 down to 30 and the second from order 40 down to 25.

To reduce the original system, we will build three reduction matrices involving an nth-order Krylov sub-space as follows:

- **Reduction matrix based on the original internal input blocks**
  The diagonal sub-blocks of the reduction matrix span the Krylov subspaces

  $$\mathbf{V}_1 = \mathscr{K}_n(\mathbf{P}_1, \mathbf{R}_1),$$

  where

  $$\mathbf{P}_1 = (s\mathbf{I}_{11} - \mathbf{A}_{11})^{-1} \qquad \text{and} \qquad \mathbf{R}_1 = (s\mathbf{I}_{11} - \mathbf{A}_{11})^{-1}[\mathbf{B}_1\ \mathbf{B}_3]$$

  and

  $$\mathbf{V}_2 = \mathscr{K}_n(\mathbf{P}_2, \mathbf{R}_2),$$

  where

  $$\mathbf{P}_2 = (s\mathbf{I}_{22} - \mathbf{A}_{22})^{-1} \qquad \text{and} \qquad \mathbf{R}_2 = (s\mathbf{I}_{22} - \mathbf{A}_{22})^{-1}\mathbf{B}_4.$$

  The block-diagonal reduction matrix $\mathbf{V}$ is of the form

  $$\mathbf{V} = \begin{bmatrix} \mathbf{V}_1 & \mathbf{0} \\ \mathbf{0} & \mathbf{V}_2 \end{bmatrix}.$$

- **Reduction matrix based on a QR decomposition of the coupling blocks**
  Based on a QR decomposition of the coupling matrices $\mathbf{A}_{12}$ and $\mathbf{A}_{21}$, we get

  $$\mathbf{A}_{12} = \mathscr{Q}_1\mathscr{R}_1 \qquad \text{and} \qquad \mathbf{A}_{21} = \mathscr{Q}_2\mathscr{R}_2.$$

  We use an *rank-revealing* version of the QR algorithm, i.e., $\mathscr{Q}_1$, $\mathscr{Q}_2$, $\mathscr{R}_1^{\mathrm{T}}$, $\mathscr{R}_2^{\mathrm{T}}$ are of full column rank. Hence, the matrices $\mathscr{Q}_1$ and $\mathscr{Q}_2$ used to build the Krylov

subspaces have the same rank (and most likely amount of columns) as $\mathbf{B}_3$ and $\mathbf{B}_4$. Next, the reduction sub-blocks are created

$$\mathbf{V}_1^{QR} = \mathscr{K}_n(\mathbf{P}_1^{QR}, \mathbf{R}_1^{QR}),$$

where

$$\mathbf{P}_1^{QR} = (s\mathbf{I}_{11} - \mathbf{A}_{11})^{-1} \qquad \text{and} \qquad \mathbf{R}_1^{QR} = (s\mathbf{I}_{11} - \mathbf{A}_{11})^{-1}[\mathbf{B}_1 \ \mathscr{Q}_1]$$

and

$$\mathbf{V}_2^{QR} = \mathscr{K}_n(\mathbf{P}_2^{QR}, \mathbf{R}_2^{QR}),$$

where

$$\mathbf{P}_2^{QR} = (s\mathbf{I}_{22} - \mathbf{A}_{22})^{-1} \qquad \text{and} \qquad \mathbf{R}_2^{QR} = (s\mathbf{I}_{22} - \mathbf{A}_{22})^{-1}\mathscr{Q}_2.$$

The block-diagonal reduction matrix $\mathbf{V}^{QR}$ is of the form

$$\mathbf{V}^{QR} = \begin{bmatrix} \mathbf{V}_1^{QR} & \mathbf{0} \\ \mathbf{0} & \mathbf{V}_2^{QR} \end{bmatrix}.$$

- **Reduction matrix based on the LU decomposition of the coupling blocks**
  Based on the LU decomposition of the coupling matrices $\mathbf{A}_{12}$ and $\mathbf{A}_{21}$, we get

$$\mathbf{A}_{12} = \mathscr{L}_1\mathscr{U}_1 \qquad \text{and} \qquad \mathbf{A}_{21} = \mathscr{L}_2\mathscr{U}_2.$$

  We use a *rank-revealing* version of the LU algorithm, i.e., $\mathscr{L}_1$, $\mathscr{L}_2$, $\mathscr{U}_1^{\mathrm{T}}$ and $\mathscr{U}_2^{\mathrm{T}}$ are of full column rank. Hence, the matrices $\mathscr{L}_1$ and $\mathscr{L}_2$ used to build the Krylov subspaces have the same rank (and most likely amount of columns) as $\mathbf{B}_3$ and $\mathbf{B}_4$. Next, the reduction sub-blocks are created

$$\mathbf{V}_1^{LU} = \mathscr{K}_n(\mathbf{P}_1^{LU}, \mathbf{R}_1^{LU}),$$

  where

$$\mathbf{P}_1^{LU} = (s\mathbf{I}_{11} - \mathbf{A}_{11})^{-1} \qquad \text{and} \qquad \mathbf{R}_1^{LU} = (s\mathbf{I}_{11} - \mathbf{A}_{11})^{-1}[\mathbf{B}_1 \ \mathscr{L}_1]$$

  and

$$\mathbf{V}_2^{LU} = \mathscr{K}_n(\mathbf{P}_2^{LU}, \mathbf{R}_2^{LU}),$$

  where

$$\mathbf{P}_2^{LU} = (s\mathbf{I}_{22} - \mathbf{A}_{22})^{-1} \qquad \text{and} \qquad \mathbf{R}_2^{LU} = (s\mathbf{I}_{22} - \mathbf{A}_{22})^{-1}\mathscr{L}_2.$$

  The block-diagonal reduction matrix $\mathbf{V}^{LU}$ is of the form

$$\mathbf{V}^{LU} = \begin{bmatrix} \mathbf{V}_1^{LU} & \mathbf{0} \\ \mathbf{0} & \mathbf{V}_2^{LU} \end{bmatrix}.$$

Figure 1.8 shows the magnitude plots with respect to the frequency of the frequency response functions of the three reduced-order systems, created using original, QR-,

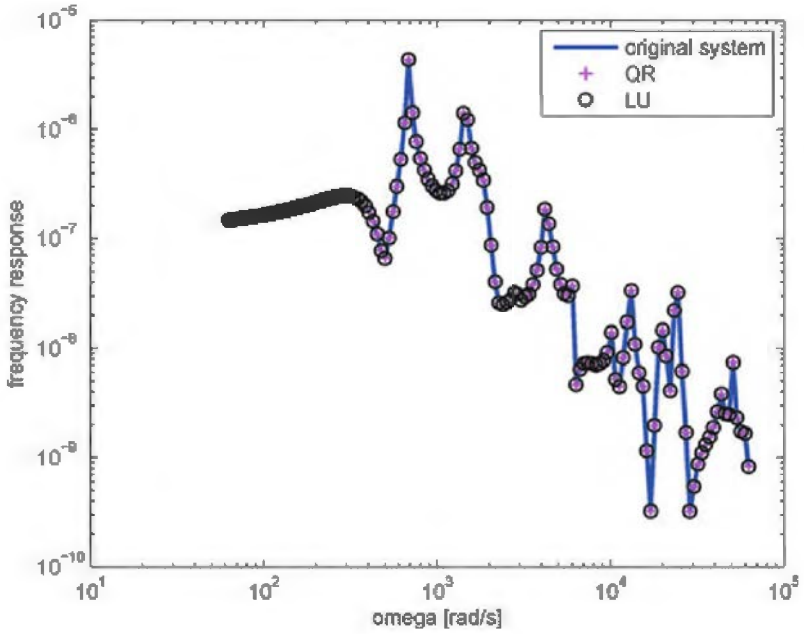


Fig. 1.8. Magnitude plots of the frequency response functions of the reduced-order systems based on different decompositions of the coupling blocks

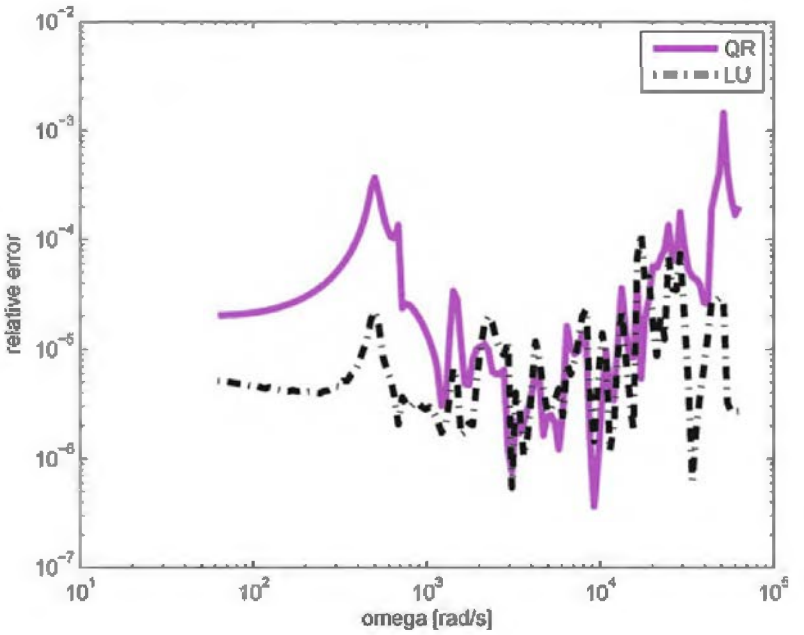


Fig. 1.9. Magnitude plots of the relative errors of the reduced-order frequency response functions based on different decompositions of the coupling blocks with respect to the reduced-order frequency response function based on the original input and output matrices

and LU-decomposition based input matrices. The plots are almost identical, which is confirmed in Fig. 1.9, that shows the relative errors between the reduced-order frequency response function of the original system and the frequency response functions computed based on both decompositions. The small differences between the three frequency response functions should be caused by round-off errors.

The next section shows how Theorems 1.3 and 1.4 in combination with GSVD can be used to improve the performance of the SBR algorithm applied to coupled systems with a high number of couplings (or interconnections).

1.4.4 Low-Rank Approximations Based SBR Algorithm

For coupled systems it is not always necessary to take into account all of the coupling components. Sometimes only a small number of them determines the behavior of the system and the rest can be neglected without much loss of accuracy. This section extends the application of the SBR algorithm to coupled (or interconnected) systems characterized by a high number of couplings of which only a small percentage is relevant to obtain an accurate solution.

Section 1.3 pointed out that the standard SBR method should be applied only for the systems with a relatively small number of internal inputs and outputs. That is, only for coupled systems where few degrees of freedom of one sub-system (related to one physical domain or to a physcial quantity) are coupled/connected to the other sub-system, which implies that the coupling blocks $\mathbf{A}_{12}$ and $\mathbf{A}_{21}$ of the system (1.37) are of low rank. Otherwise, the SBR method produces reduction bases which increase in size too fast with respect to the number of Krylov iterations. However, if only a part of the components of the high rank coupling blocks is relevant, we can decrease the growth speed of the reduction bases. To do so, we first need to determine, which components of the coupling are important and should be kept, and which ones can be neglected. One of the ways to make this decision, is to apply the generalized singular value decomposition (GSVD) to the coupling matrices $\mathbf{A}_{12}$ and $\mathbf{A}_{21}$. The GSVD should be applied to the pairs $(\mathbf{A}_{11}^T, \mathbf{A}_{12}^T)$ and $(\mathbf{A}_{22}^T, \mathbf{A}_{21}^T)$. One then has

$$\mathbf{A}_{12}^T = \mathbf{V}_1 \mathbf{S}_1 \mathbf{X}_1^T \quad \text{and} \quad \mathbf{A}_{21}^T = \mathbf{V}_2 \mathbf{S}_2 \mathbf{X}_2^T$$

which results in the expressions for the coupling blocks

$$\mathbf{A}_{12} = \mathbf{X}_1 \mathbf{S}_1^T \mathbf{V}_1^T \tag{1.43}$$

$$\mathbf{A}_{21} = \mathbf{X}_2 \mathbf{S}_2^T \mathbf{V}_2^T. \tag{1.44}$$

Note, that here the matrices $\mathbf{C}_1$ and $\mathbf{C}_2$ are not used to denote external output matrices, but components of the GSVD. Assuming that the coupling blocks are of the form (1.39), since $\mathbf{S}_1$ and $\mathbf{S}_2$ are real-valued non-negative diagonal, we can define the input and output matrices as following products

$$\tilde{\mathbf{B}}_3 = \mathbf{X}_1 \mathbf{S}_1^{1/2}, \qquad \tilde{\mathbf{C}}_4 = \mathbf{V}_1 \mathbf{S}_1^{1/2} \tag{1.45}$$

$$\tilde{\mathbf{B}}_4 = \mathbf{X}_2 \mathbf{S}_2^{1/2}, \qquad \tilde{\mathbf{C}}_3 = \mathbf{V}_2 \mathbf{S}_2^{1/2}. \tag{1.46}$$

Since $\mathbf{S}_1$ and $\mathbf{S}_2$ are diagonal matrices with non-negative entries their square roots are diagonal matrices with entries $\sqrt{[\mathbf{S}_1]_{ii}}$ and $\sqrt{[\mathbf{S}_2]_{ii}}$. Constructing the inputs and outputs as in (1.45) and (1.46), all of $\mathbf{B}_i$ and $\mathbf{C}_i$, $i = 3, 4$ are scaled by $\sqrt{\mathbf{S}_1}$ or $\sqrt{\mathbf{S}_2}$.

According to the Theorems 1.3 and 1.4, $\mathscr{K}_p(\mathbf{A}_{11}, \mathbf{B}_3\mathbf{C}_4^T) = \mathscr{K}_p(\mathbf{A}_{11}, \tilde{\mathbf{B}}_3\tilde{\mathbf{C}}_4^T)$ and $\mathscr{K}_p(\mathbf{A}_{22}, \mathbf{B}_4\mathbf{C}_3^T) = \mathscr{K}_p(\mathbf{A}_{11}, \tilde{\mathbf{B}}_4\tilde{\mathbf{C}}_3^T)$. Moreover, using a type of the decomposition that orders the components with respect to their importance has an additional benefit. It makes it possible to approximate the inputs and outputs leaving only the most relevant components and, as a result, reduces the dimensions of the blocks. In some cases, this reduction is sufficient to allow for an efficient application of the SBR algorithm.

Let us now compare the procedures of building the standard and GSVD-based Krylov subspaces. Here, we will limit the discussion to the case of creation of a Krylov space based on inputs of the sub-system (1.7), but a similar analysis applies to all the other cases, i.e. input-based Krylov subspace for (1.8) and output-based Krylov subspaces for both sub-systems, (1.7) and (1.8). As defined in Chap. 1.3, matrices $\mathbf{A}_{11} \in \mathbb{R}^{N_1 \times N_1}$, $\mathbf{B}_1 \in \mathbb{R}^{N_1 \times m_1}$ and $\mathbf{B}_3 \in \mathbb{R}^{N_1 \times m_3}$. Assume, that $\mathbf{B}_3$ has full column rank m_3 and that application of GSVD to the pair $(\mathbf{A}_{11}^T, \mathbf{A}_{12}^T)$ leads to

$$\tilde{\mathbf{B}}_3 = \mathbf{X}_1\mathbf{S}_1^{1/2} = [\tilde{\mathbf{b}}_1, .., \tilde{\mathbf{b}}_{m_3}] \in \mathbb{R}^{N_1 \times m_3}.$$

where both $\mathbf{X}_1$ and $\mathbf{S}_1$ in (1.45) are of full column rank. Next, let $\hat{\mathbf{B}}_3 = \mathbf{X}_1^{(k)}(\mathbf{S}_1^{(k)})^{1/2}$ approximate $\tilde{\mathbf{B}}_3$ with the use of k dominant components. Then

$$\hat{\mathbf{B}}_3 = [\tilde{\mathbf{b}}_1, .., \tilde{\mathbf{b}}_k] \in \mathbb{R}^{N_1 \times k}. \tag{1.47}$$

For simplicity, we assume that $m_1 + m_3$ is a multiple of $m_1 + k$ (this may not be the case in general), so there exists $\lambda \in \mathbb{N}$ such that

$$m_1 + m_3 = \lambda(m_1 + k). \tag{1.48}$$

The pth Krylov subspace created by the SBR algorithm for the sub-system (1.7) for $s_0 \in \mathbb{C}$ is

$$\mathscr{K}_p(\mathbf{P}_1, \mathbf{R}_1) = \text{colspan}\{\mathbf{R}_1, \mathbf{P}_1\mathbf{R}_1, \ldots, \mathbf{P}_1^{p-1}\mathbf{R}_1\},$$

where

$$\mathbf{P}_1 = (s_0\mathbf{E}_{11} - \mathbf{A}_{11})^{-1}\mathbf{E}_{11} \quad \text{and} \quad \mathbf{R}_1 = (s_0\mathbf{E}_{11} - \mathbf{A}_{11})^{-1}[\mathbf{B}_1\ \mathbf{B}_3]$$

and consists of $p(m_1 + m_3)$ columns (assuming that no linear dependence occurs).

Likewise,

$$\mathscr{K}_{\lambda p}(\mathbf{P}_1, \hat{\mathbf{R}}_1) = \text{colspan}\{\hat{\mathbf{R}}_1, \mathbf{P}_1\hat{\mathbf{R}}_1, \ldots, \mathbf{P}_1^{\lambda p-1}\hat{\mathbf{R}}_1\}, \tag{1.49}$$

where

$$\mathbf{P}_1 = (s_0\mathbf{E}_{11} - \mathbf{A}_{11})^{-1}\mathbf{E}_{11} \quad \text{and} \quad \hat{\mathbf{R}}_1 = (s_0\mathbf{E}_{11} - \mathbf{A}_{11})^{-1}[\mathbf{B}_1\ \hat{\mathbf{B}}_3]$$

consists also of $p(m_1 + m_3)$ columns, but *approximately* matches λ as many moments of the original transfer function.

Projecting system (1.7) onto a subspace $\mathscr{K}_{\lambda p}(\mathbf{P}_1, \hat{\mathbf{R}}_1)$ in (1.49) does not preserve the moments of the transfer function of this sub-system. However, if the column span

of the matrix $\hat{\mathbf{B}}_3$ gives a good approximation of the column span of the matrix $\mathbf{B}_3$ we can expect that the reduced-order system obtained by projection onto the space (1.49) will give an accurate approximation of the appropriate number of moments of the transfer function of the original system. Moreover, if the matrix $\mathbf{B}_3$ can be approximated by $\hat{\mathbf{B}}_3$ with a significantly smaller number of columns, λ times more steps may be used during the Krylov procedure (to approximate a higher number of moments) or one can use more expansion points, keeping the reduced-order model still relatively small.

In the next section, we present the results of some numerical tests that show the advantage of using the low-rank approximation based SBR algorithm for a system with a high order of coupling.

1.5 Numerical Examples

In this section, we present two examples of the application of the SBR method combined with low rank approximations for the coupling blocks. The first example is a simple and small example, yet exhibiting interesting behaviour as far as coupling is concerned. The second example is described in much more detail, as this is an industrial benchmark problem and needs some preliminary steps before the methods described in this chapter can be applied.

1.5.1 A Simple Example

In this section, we consider a simple example. The difficulty of this test case is that here the coupling blocks of the system are of rank 10 (the coupled system has 10 internal inputs and 10 internal outputs), while each of the sub-systems contains only 40 degrees of freedom (80 degrees of freedom in total). In this case, the standard SBR algorithm generates too many columns to be competitive. However, the use of low-rank approximations makes the SBR algorithm more competitive. Fig. 1.10 shows the magnitude plots with respect to the frequency of the original and reduced-order frequency response functions. In case of the two-sided BSP method and the two-sided SBR algorithm based reduced-order systems, the original system was reduced to 42 degrees of freedom. The low-rank approximation based two-sided SBR algorithm created the reduction bases for rank 3 approximations of the coupling blocks, i.e. the internal input and output matrices $\mathbf{B}_i, \mathbf{C}_i \in \mathbb{R}^{40\times 10}$, $i = 3,4$ were approximated by $\hat{\mathbf{B}}_i, \hat{\mathbf{C}}_i \in \mathbb{R}^{40\times 3}$, $i = 3,4$. Hence, every Krylov step was adding 4 new columns to the reduction basis (3 corresponding to $\hat{\mathbf{B}}_3$ or $\hat{\mathbf{C}}_4$ and 1 corresponding to $\mathbf{B}_1$ or $\mathbf{C}_1$) in case of the sub-system S_1 and 3 new columns (corresponding to $\hat{\mathbf{B}}_4$ or $\hat{\mathbf{C}}_3$) in case of the sub-system S_2. To construct the reduced-order system of dimension 42, the low-rank approximation based SBR algorithm performed 6 iterations for each sub-system (for both, input and output related bases). Figure 1.11 shows the magnitude plots of the relative errors of the reduced-order frequency response functions with respect to the original one. Note that the two-sided SBR algorithm

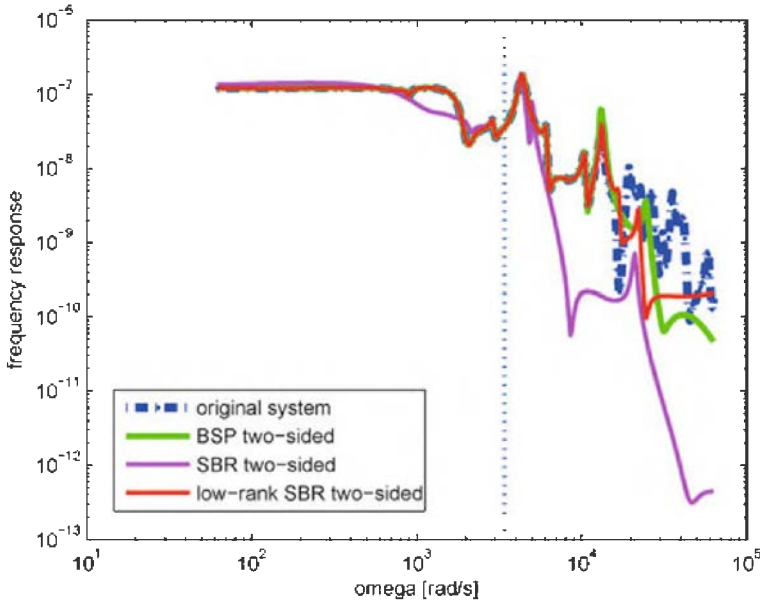


Fig. 1.10. Magnitude plots of the frequency response functions of the original and reduced-order systems

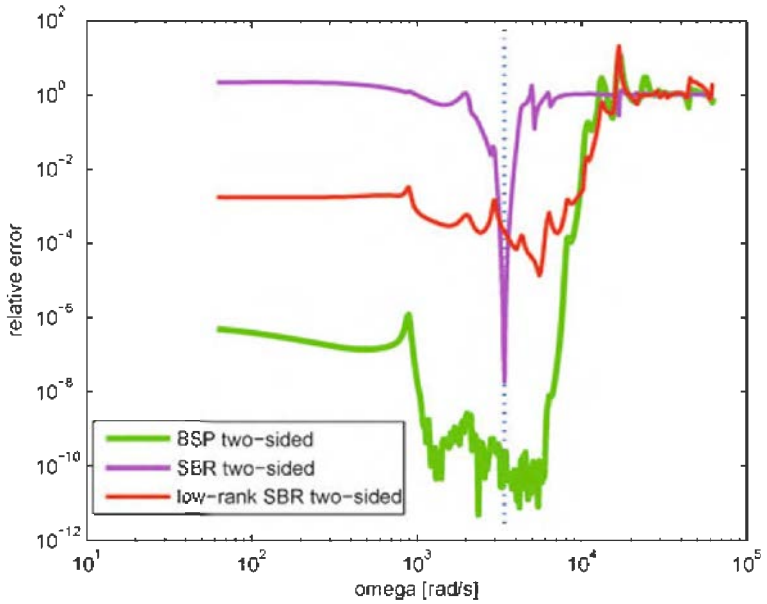


Fig. 1.11. Magnitude plots of the relative errors of the reduced-order frequency response functions with respect to the original frequency response function

based on low-rank approximations of the internal inputs and outputs leads to much better results than the SBR algorithm applied to the original coupling blocks. The two-sided low-rank based reduced-order transfer function $\mathbf{H}_{\text{low-rank-SBR}}$ approximates $\mathbf{H}$ less accurate than the standard two-sided BSP transfer function but in the neighborhood of the expansion point s the relative error is still below 2%. Table 1.1 shows that not only the first 6 derivatives are matched but also the 7^{th} one is well approximated.

Table 1.1. Derivatives of the original and low-rank approximation based reduced-order transfer functions for the expansion point (s) for the second example, multiplied by 10^7

i	$\partial^i \mathbf{H}(s)$	$\partial^i \mathbf{H}_{\text{low-rank-SBR}}(s)$
0	−0.349984611544531	−0.349975323605725
1	0.000580754070987	0.000580770193275
2	−0.000001928114532	−0.000001928787960
3	0.000000012770698	0.000000012766510
4	−0.000000000067912	−0.000000000068062
5	0.000000000000859	0.000000000000859
6	−0.000000000000014	−0.000000000000014
7	0.000000000000000	0.000000000000000
8	-0.000000000000000	−0.000000000000000
9	0.000000000000000	0.000000000000000

1.5.2 Industrial Benchmark Problem

The benchmark system treated in this chapter is a model of a printhead delivered by Océ Technologies B.V. in the Netherlands. It is a MEMS (micro-electro-mechanical-system) based design, containing a large number of individual channels integrated into a single chip. A schematic overview of a single channel (a side and bottom view) is shown in Fig. 1.12. The dotted line depicts the ink flow; the ink, coming from the reservoir, enters through a restriction (1), from which it flows into the actuation

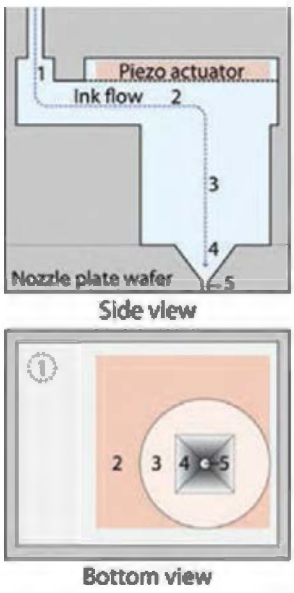


Fig. 1.12. A schematic overview of a single channel (courtesy of Herman Wijshoff)

chamber (2). Below the actuation chamber, a 300 μm long feed-through is placed (3), after which the nozzle plate is reached. The nozzle plate is 75 μm thick and consists of a pyramid shaped funnel (4) and a nozzle (5) with a radius of 11 μm.

The main goal is to suppress acoustic pressure waves, which can be generated in a number of ways, such as the non-continuous ink supply by many thousands of ink channels, residual vibrations at the inlet of the ink channels, fast movement of the printhead, resonance of the whole structure, etc.

The models of such devices used for simulations can reach large dimensions, hence application of the model order reduction techniques is often required, to decrease the simulation time. In this chapter, we study the application of the GSVD based approximations for the coupling blocks in the model of the printhead.

1.5.3 The Second and First Order System

The related system of equations is a second order system. Let $n_1, n_2 \in \mathbb{N}$ and $n = n_1 + n_2$. The *second order system* of interest is

$$\begin{cases} \mathbf{M}\mathbf{x}'' + \mathbf{K}\mathbf{x} = \mathbf{b} \\ \mathbf{y} = \mathbf{c}\mathbf{x} \end{cases} \tag{1.50}$$

with $(n_1 + n_2) \times (n_1 + n_2)$, 2×2 block-matrices

$$\mathbf{M} = \begin{bmatrix} \mathbf{M}_{11} & \mathbf{0} \\ \mathbf{M}_{22} & \mathbf{M}_{22} \end{bmatrix}, \quad \mathbf{K} = \begin{bmatrix} \mathbf{K}_{11} & \mathbf{K}_{12} \\ \mathbf{0} & \mathbf{K}_{22} \end{bmatrix} \tag{1.51}$$

and $\mathbf{M}_{21} = -\rho \mathbf{K}_{12}^{\mathrm{T}}$. The first sub-system corresponds to the displacement of the structure and the second sub-system describes the pressure of the fluid. The related *Laplace transformation*

$$\begin{cases} \tilde{w}^2 \mathbf{M}\mathbf{X} + \mathbf{K}\mathbf{X} = \mathbf{B} \\ \mathbf{Y} = \mathbf{c}\mathbf{X} \end{cases}$$

leads to transfer function

$$H(w) = \mathbf{c}(\mathbf{K} + \tilde{w}^2 \mathbf{M})^{-1} \mathbf{b}, \qquad \tilde{w} \in \mathbb{C}.$$

Searching for purely *oscillatory modes* implies that the related $\tilde{w}$ is purely imaginary, i.e., that one is interested in positive real values w of:

$$H(w) = \mathbf{c}(\mathbf{K} - w^2 \mathbf{M})^{-1} \mathbf{b}, \qquad w \in \mathbb{R}. \tag{1.52}$$

Let $\mathbf{x}_2 = \mathbf{x}_1'$. Then the *first order system* reformulation of (1.50) is

$$\begin{cases} \mathbf{x}_2 = \mathbf{x}_1' \\ \mathbf{M}\mathbf{x}_2' + \mathbf{K}\mathbf{x}_1 = \mathbf{b} \end{cases} \Longrightarrow \begin{cases} \mathbf{x}_1' - \mathbf{x}_2 = \mathbf{0} \\ \mathbf{M}\mathbf{x}_2' + \mathbf{K}\mathbf{x}_1 = \mathbf{b} \end{cases}$$

which implies

$$\begin{cases} \underbrace{\begin{bmatrix} \mathbf{I} & \\ & \mathbf{M} \end{bmatrix}}_{\mathbf{E}} \begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \end{bmatrix}' = \underbrace{\begin{bmatrix} \mathbf{I} \\ -\mathbf{K} \end{bmatrix}}_{\mathbf{A}} \begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \end{bmatrix} + \underbrace{\begin{bmatrix} \mathbf{0} \\ \mathbf{b} \end{bmatrix}}_{\mathbf{B}} \\ \begin{bmatrix} \mathbf{y}_1 \\ \mathbf{y}_2 \end{bmatrix} = \begin{bmatrix} \mathbf{c} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} + \begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \end{bmatrix}. \end{cases}$$

Its related transfer function is

$$H(s) = \mathbf{C}(s\mathbf{E} - \mathbf{A})^{-1}\mathbf{B}. \tag{1.53}$$

Solution of $\mathbf{FX} = \mathbf{B}$:

$$\begin{cases} s\mathbf{x}_1 - \mathbf{x}_2 = \mathbf{0} \\ s\mathbf{M}\mathbf{x}_2 + \mathbf{K}\mathbf{x}_1 = \mathbf{b} \end{cases} \Longrightarrow \begin{cases} s\mathbf{x}_1 - \mathbf{x}_2 = \mathbf{0} \\ s^2\mathbf{M}\mathbf{x}_1 + \mathbf{K}\mathbf{x}_1 = \mathbf{b} \end{cases} \Longrightarrow \begin{cases} \mathbf{x}_1 = \left(s^2\mathbf{M} + \mathbf{K}\right)^{-1}\mathbf{b} \\ \mathbf{x}_2 = s\mathbf{x}_1. \end{cases}$$

This implies that

$$\mathbf{y}_1 = \mathbf{c}\left(s^2\mathbf{M} + \mathbf{K}\right)^{-1}\mathbf{b}$$

is identical to $\mathbf{y}$ if and only if $s = iw$, $w \in \mathbb{R}$.

In the sequel we will examine the second order system.

1.5.4 Sparsity Patterns and Magnitudes of the Blocks of $\mathbf{M}$, $\mathbf{K}$

There are three available discretizations for the *OCE application*: *coarse*: `1188_1050`, *medium*: `4752_5304` and *fine*: `20748_35775`. The numbers relate to the amount of degrees of freedom as follows: Case `4752_5304` implies $n_1 = 4752$ and $n_2 = 5304$. Extracted from ANSYS, the blocks $\mathbf{M}_{11}, \mathbf{M}_{21}, \mathbf{M}_{22}, \mathbf{K}_{11}, \mathbf{K}_{12}, \mathbf{K}_{22}$ in (1.51) are very differently scaled: For instance, for the medium case their absolute value greatest resp. smallest entries (*magnitude*) are of the order

$$\begin{aligned} \mathbf{M} &= O\left(\begin{bmatrix} 10^{-10} & \mathbf{0} \\ 10^{-5} & 10^{-18} \end{bmatrix}\right), \quad \mathbf{K} = O\left(\begin{bmatrix} 10^{+8} & 10^{-8} \\ \mathbf{0} & 10^{-4} \end{bmatrix}\right), \\ \mathbf{M} &= O\left(\begin{bmatrix} 10^{-12} & \mathbf{0} \\ 10^{-6} & 10^{-20} \end{bmatrix}\right), \quad \mathbf{K} = O\left(\begin{bmatrix} 10^{-12} & 10^{-9} \\ \mathbf{0} & 10^{-6} \end{bmatrix}\right). \end{aligned} \tag{1.54}$$

For the calculation of the transfer function furthermore note that $w \in [0, 2\pi * 1500]$. Thus approximately, $w^2 \in [0, 10^8]$. The use of the standard `MATLAB` '\' operations to solve $(\mathbf{K} - w^2\mathbf{M})\mathbf{x} = \mathbf{b}$ leads to error messages and abortions, not to solutions. An alternative, the use of the `MATLAB` package `Factorize`, alleviates this problem, but (too) severe round-off remains. Furthermore, the '\' operation turns out to be very slow for this poorly scaled problem. Investigation shows that that $\mathbf{K}_{11}$ contains entries in $[10^{-12}, 10^{+8}]$. The use of standard double precision floating point *IEEE arithmetic* involved in matrix operations such as matrix multiplication is bound to round-away contributions of the smaller entries.

Further investigation shows that all diagonal blocks but $\mathbf{K}_{11}$ are symmetric. For the results shown in this chapter the slightly non-symmetric ANSYS block $\mathbf{K}_{11}$ has been used as is. The results would be the same if one had instead used its symmetric part $(\mathbf{K}_{11} + \mathbf{K}_{11}^{\mathrm{T}})/2$ (tested). It has also been shown that indeed $\mathbf{M}_{21} = -\rho\mathbf{K}_{12}^{\mathrm{T}}$ for all three examples, where $\rho = 1090$.

Observe that the determination of the smallest absolute value positive entry of a sparse `MATLAB` matrix with `MATLAB` is not trivial: The smallest entry of a sparse matrix usually is zero (since the default entry has value zero), MATHWORKS and other sources do not provide an on-the-shelf solution. To obtain the smallest non-zero entry we have written a `MATLAB` function `vfilter` which for a full or sparse matrix $\mathbf{X}$ writes all entries $\mathbf{X}_{ij}$ such that $|\mathbf{X}_{ij}| > \varepsilon \geq 0$ column-wise into a full vector. The use of this function applied to matrix `X` and $\varepsilon = 0$ in combination with `min` provides the smallest absolute value entry of `X`.

Naturally, small entries should only be discarded if they are not relevant to the system of interest, i.e., if the the system is properly scaled, which is the topic of discussion of the next subsection.

1.5.5 Scaling the Second Order System

We need to scale the matrices $\mathbf{K}$ and $\mathbf{M}$ ($\mathbf{E}$ and $\mathbf{A}$) to obtain a *numerically robust* solution of the system

$$\mathbf{F}(w)\mathbf{x} = \mathbf{b} \iff (\mathbf{K} - w^2\mathbf{M})\mathbf{x} = \mathbf{b} \iff \begin{bmatrix} \mathbf{K}_{11} - w^2\mathbf{M}_{11} & \mathbf{K}_{12} \\ \rho w^2 \mathbf{K}_{12}^{\mathrm{T}} & \mathbf{K}_{22} - w^2\mathbf{M}_{22} \end{bmatrix} \mathbf{x} = \mathbf{b},$$

which depends on w. For the problem of interest we expect symmetric blocks $\mathbf{M}_{11}, \mathbf{M}_{22}, \mathbf{K}_{11}$ and $\mathbf{K}_{22}$, and $\mathbf{M}_{21} = -\rho\mathbf{K}_{12}^{\mathrm{T}}$. This implies that this system could be scaled (preconditioned) into a symmetric one (*symmetry scaling*), for which efficient linear solvers exist. This can be done as follows: Observe that for a two by two matrix

$$\mathbf{A} = \begin{bmatrix} a & d \\ c & b \end{bmatrix}, \quad \mathbf{D}_1 = \begin{bmatrix} 1 & \\ & \sqrt{c/d} \end{bmatrix} \Longrightarrow \mathbf{D}_1^{-1}\mathbf{A}\mathbf{D}_1 = \begin{bmatrix} a & \sqrt{cd} \\ \sqrt{cd} & b \end{bmatrix}$$

can be scaled to a symmetric one. Hence, based on $c = \rho w^2$ and $d = 1$, define

$$\mathbf{D}_1 = \begin{bmatrix} \mathbf{I}_{n_1} & \\ & \sqrt{\rho w^2}\mathbf{I}_{n_2} \end{bmatrix}.$$

Furthermore, to better scale the entries inside and between blocks (create diagonal elements of magnitude 1), define

$$\mathbf{D}_2 = \mathrm{diag}(1/\sqrt{[\mathbf{D}_1^{-1}\mathbf{F}\mathbf{D}_1]_{11}}, \ldots, 1/\sqrt{[\mathbf{D}_1^{-1}\mathbf{F}\mathbf{D}_1]_{nn}}).$$

We now scale with a *diagonal scaling*:

$$\hat{\mathbf{M}} := \underbrace{\mathbf{D}_2\mathbf{D}_1^{-1}}_{\mathbf{Q}}\mathbf{M}\underbrace{\mathbf{D}_1\mathbf{D}_2}_{\mathbf{P}},$$
$$\hat{\mathbf{K}} := \mathbf{QKP}, \hat{\mathbf{b}} := \mathbf{Qb}, \hat{\mathbf{c}} := \mathbf{cP},$$

which, by invariance under inputs and outputs transformations means that

$$\hat{\mathbf{H}}(w) := \mathbf{c}(\hat{\mathbf{K}} - w^2\hat{\mathbf{M}})^{-1}\hat{\mathbf{b}}$$

is identical to $\mathbf{H}$ in (1.52) for all w. Obviously $\mathbf{D}_1$ is non-singular except for $w = 0$ and $\mathbf{D}_2$ exists and is non-singular when all diagonal entries of $\mathbf{D}_1^{-1}\mathbf{F}\mathbf{D}_1$ are non-zero.

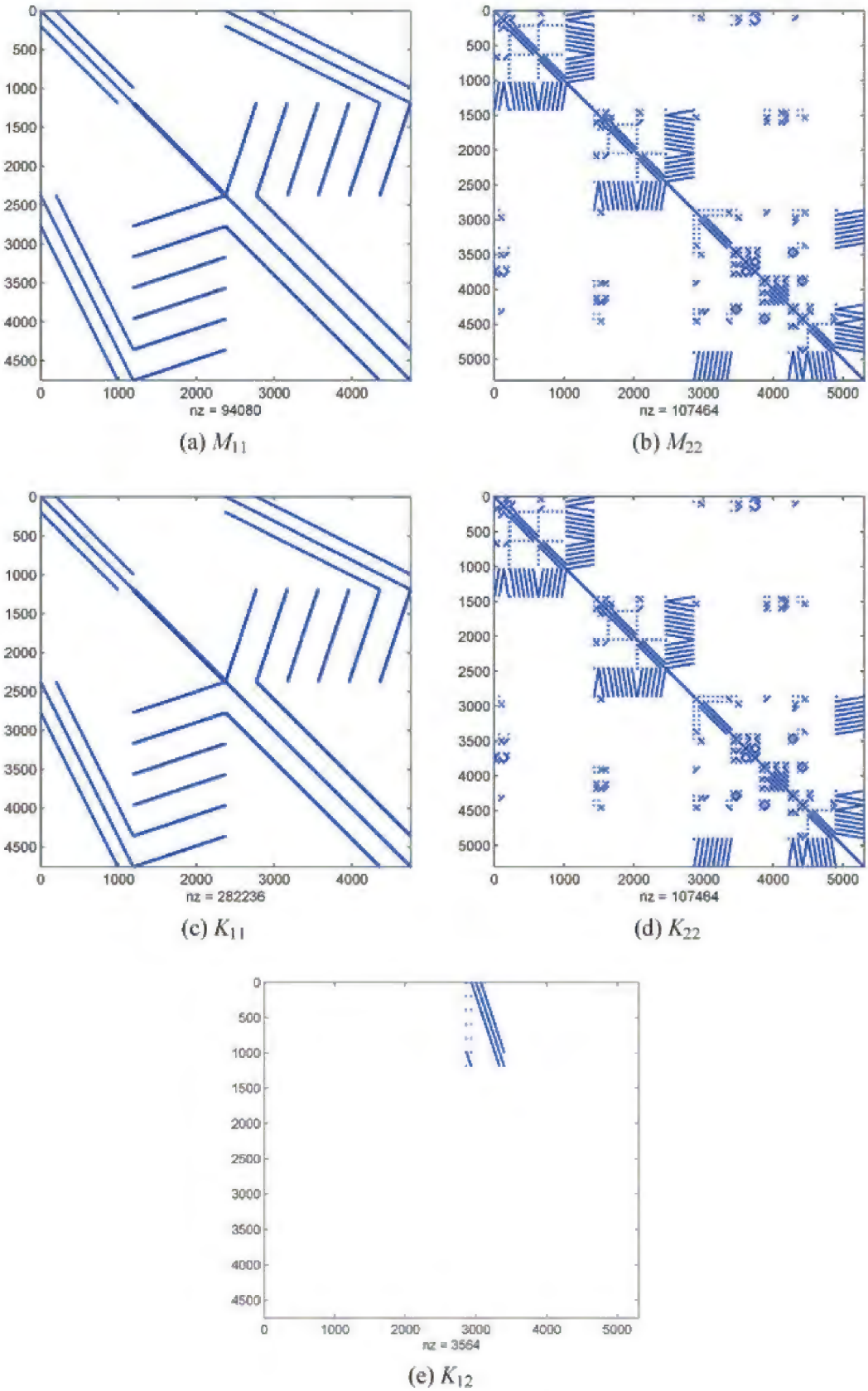
The factors $\mathbf{P} = \mathbf{P}(w)$ and $\mathbf{Q} = \mathbf{Q}(w)$ depend on w. This is fine for the construction of Krylov spaces to match moments. However, to plot the transfer function $\mathbf{H}$ one needs to evaluate $\mathbf{c}(\mathbf{K} - w_k^2\mathbf{M})$ for many $w_k \in [0, 10^8]$. Repeated calculation of $\mathbf{P}(w_k)$ and $\mathbf{Q}(w_k)$ would be (too) costly, so we decided to use the w-independent factors $\mathbf{P} := \mathbf{P}(\hat{w})$ $\mathbf{Q} := \mathbf{Q}(\hat{w})$ for all w where $\hat{w}$ is the average of all w_k. For the OCE example, to plot the transfer functions, we sample the provided region of interest: $w_k = 5\pi \cdot k$, $k = 0, \ldots, 600$. The value of $\hat{w}$ turns out to be w_{301} which is close to but not too close to a *pole* of $\mathbf{H}$ and such that all diagonal entries of $\mathbf{D}_1^{-1}\mathbf{F}\mathbf{D}_1$ are non-zero.

1.5.6 The Structure and the GSVD of $\mathbf{K}_{12}$

Here we briefly comment on the GSVD of the scaled $\mathbf{K}_{12}^{\mathrm{T}}$. Figure 1.13 and numerical investigation show that $\mathbf{K}_{12} \in \mathbb{R}^{1188\times 1050}$ is a *sparse matrix* which contains a small *non-zero sub-block* of size 295×175 (window $(3, \ldots, 297) \times (561, \ldots, 735)$). This is typical for applications where the different physical quantities are defined in bordering sub-domains and are coupled via the mutual boundary – if one numbers the degrees of freedom on the mutual boundary consecutively. Since $\mathbf{K}_{12}^{\mathrm{T}}$ has this structure it is of the required type. This means that also $\mathbf{V}$ has all its non-zero entries in the same sub-block, i.e., it only has possible non-zero entries from row 561 to 735. This information is of importance, because the standard GSVD implementations such as `MATLAB`'s do not use this information and generate $\mathbf{V}$ which contains round-off (non-zero) entries outside the window, as can be seen in Fig. 1.14. For the medium test case the results are worse, as to be expected: For $p = 5$ and $\varepsilon = 0$, $\mathbf{K}_{12}^{(p)}$ (definitions, see below) is a full matrix.

To work around this problem we have written a `MATLAB` function `spfilter` which for a full or sparse matrix $\mathbf{X}$ copies all entries $\mathbf{X}_{ij}$ such that $|\mathbf{X}_{ij}| > M(\mathbf{X}) \cdot \varepsilon$ into a sparse matrix $\mathbf{Y}$, where $M(\mathbf{X}) := \max\{|X|_{ij}\}_{i,j}$. This way, using $\varepsilon = 10^{-11}$, both $\mathbf{K}_{12} = \mathbf{X}\mathbf{S}^{\mathrm{T}}\mathbf{V}^{\mathrm{T}}$ and all of its dominant parts $\mathbf{K}_{12}^{(p)} := \mathbf{X}^{(p)}\mathbf{S}^{(p)}\mathbf{V}^{(p)}$ (for some $p \leq n$) have similar sparsity patterns.

In `MATLAB` there are different but equivalent manners for the filtering of entries from a matrix. However, most of them do not terminate or lead to out of memory

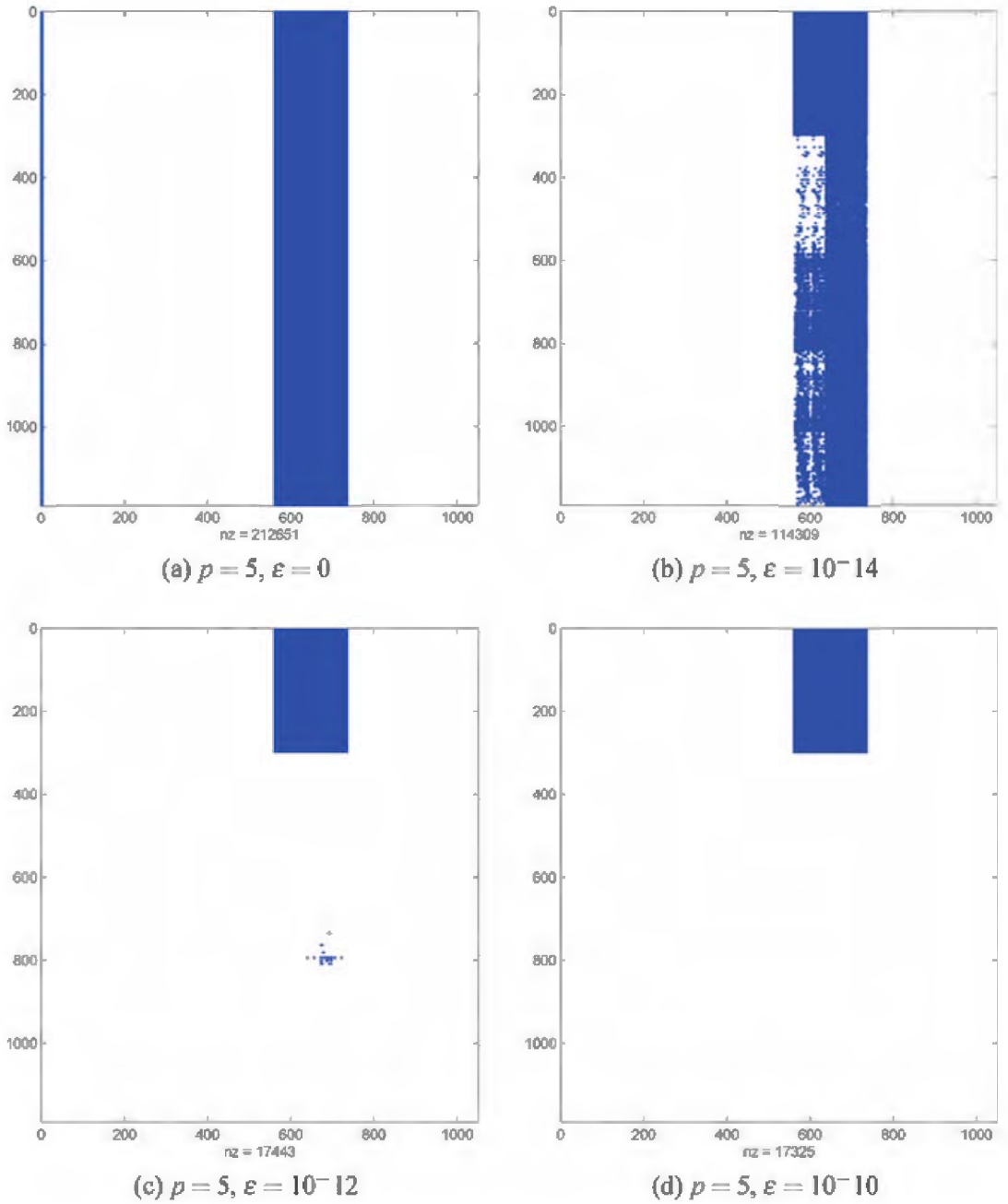


(a) M_{11} (b) M_{22}

(c) K_{11} (d) K_{22}

(e) K_{12}

Fig. 1.13. Sparsity pattern of matrix blocks



(a) $p = 5, \varepsilon = 0$ (b) $p = 5, \varepsilon = 10^{-}14$

(c) $p = 5, \varepsilon = 10^{-}12$ (d) $p = 5, \varepsilon = 10^{-}10$

Fig. 1.14. Entries of $\mathbf{K}_{12}^{(p)}$ greater than $\varepsilon \cdot M(\mathbf{K}_{12}^{(p)})$, small case

errors even for the small case. Functions `vfilter` and `spfilter` contain information on manners which somehow do not lead to the desired result.

Explicit multiplication with factors $\mathbf{X}^{(p)}$, $\mathbf{S}^{(p)}$ and $\mathbf{V}^{(p)}$ for the multiplication with $\mathbf{x} \mapsto \mathbf{K}\mathbf{x}$ is likely to be the more efficient then the use of multiplication with $\mathbf{K}_{12}^{(p)}$.

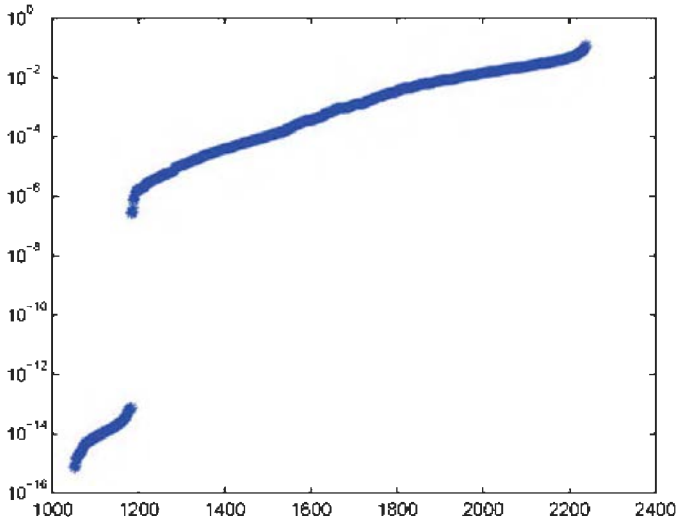


Fig. 1.15. Entries of $\mathbf{x}$ in (1.55), sorted

1.5.7 A GSVD-Based Approximation of $\mathbf{K}_{12}$

In this subsection we analyse how the GSVD based approximation of $\mathbf{K}_{12}$ influences the solution of the static problem

$$\begin{bmatrix} \mathbf{K}_{11} & \mathbf{K}_{12} \\ \mathbf{0} & \mathbf{K}_{22} \end{bmatrix} \begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \end{bmatrix} = \begin{bmatrix} \mathbf{b}_1 \\ \mathbf{b}_2 \end{bmatrix} \tag{1.55}$$

Based on the definition of $\mathbf{K}_{12}^{(p)}$ the approximation leads to system

$$\begin{bmatrix} \mathbf{K}_{11} & \mathbf{K}_{12}^{(p)} \\ \mathbf{0} & \mathbf{K}_{22} \end{bmatrix} \begin{bmatrix} \mathbf{y}_1^{(p)} \\ \mathbf{y}_2^{(p)} \end{bmatrix} = \begin{bmatrix} \mathbf{b}_1 \\ \mathbf{b}_2 \end{bmatrix}. \tag{1.56}$$

We intend to estimate

$$|| \left(|x_i - y_i^{(p)}| / |x_i| \right)_i ||_\infty \tag{1.57}$$

over the set of indices i for which x_i is non-zero (outside round-off region). To determine this set, we first solved (1.55) and made a log-plot of its sorted entries, shown in Fig. 1.15. Based on this plot we decided to omit all entries smaller than 10^{-7} and obtained the results in Table 1.2. The accuracy does not seem to be (very) sensitive to the amount of principal components used, which is due to the fact that the scaled $\mathbf{K}_{12}$ block is still of magnitude 10^5 smaller than the scaled diagonal blocks $\mathbf{K}_{11}$ and $\mathbf{K}_{22}$. However, Sect. 1.5.8 shows that different amounts of principal components do have a remarkable effect on the related transfer function.

1.5.8 The $\mathbf{K}_{12}{}^{(p)}$ *GSVD-Approximation Based Transfer Function*

The aim is to determine a principal component analysis (*PCA*) based *rank-revealing* factorization $\mathbf{K}_{12} \doteq \mathbf{B}\mathbf{C}^{\mathrm{T}}$ where $\mathbf{B}$ and $\mathbf{C}$ are constructed with the use of the first

Table 1.2. Relative errors due to use of the GSVD approximation

p	$\|\|\|x(i)-y^{(p)}(i)\|/\|x(i)\|\|\|_\infty$
1	3.750080647e-008
2	1.427208120e-007
3	1.119493657e-007
4	1.468582269e-007
5	1.500944068e-007

$p \leq n$ *principal components*, based on the scaled versions of $\mathbf{K}$ (and if needed $\mathbf{M}$) as constructed above.

To the scaled matrix $\mathbf{K}$ (which depends on $\hat{w}$) we apply a GSVD to $\mathbf{K}_{11}^{\mathrm{T}}$ and $\mathbf{K}_{12}^{\mathrm{T}}$ such that $\mathbf{K}_{11}^{\mathrm{T}} = \mathbf{U}\mathbf{C}\mathbf{X}^{\mathrm{T}}$ and $\mathbf{K}_{12}^{\mathrm{T}} = \mathbf{V}\mathbf{S}\mathbf{X}^{\mathrm{T}}$. Hence,

$$\mathbf{K}_{12} = \mathbf{X}\mathbf{S}^{\mathrm{T}}\mathbf{V}^{\mathrm{T}} = \underbrace{\mathbf{X}\sqrt{\mathbf{S}^{\mathrm{T}}}}_{\mathbf{B}}\underbrace{\sqrt{\mathbf{S}^{\mathrm{T}}}\mathbf{V}^{\mathrm{T}}}_{\mathbf{C}^{\mathrm{T}}}.$$

Figure 1.16 shows all of the diagonal values of the matrix S and Fig. 1.17 shows the first 1000 of them. Next, for $p = 1,\ldots,5$ we approximate $\mathbf{K}_{12}$ by the contribution of its p most dominant modes

$$\mathbf{K}^{(p)} = (\mathbf{X}^{(p)}\sqrt{\mathbf{S}^{(p)}})(\sqrt{\mathbf{S}^{(p)}}\mathbf{V}^{(p)})$$

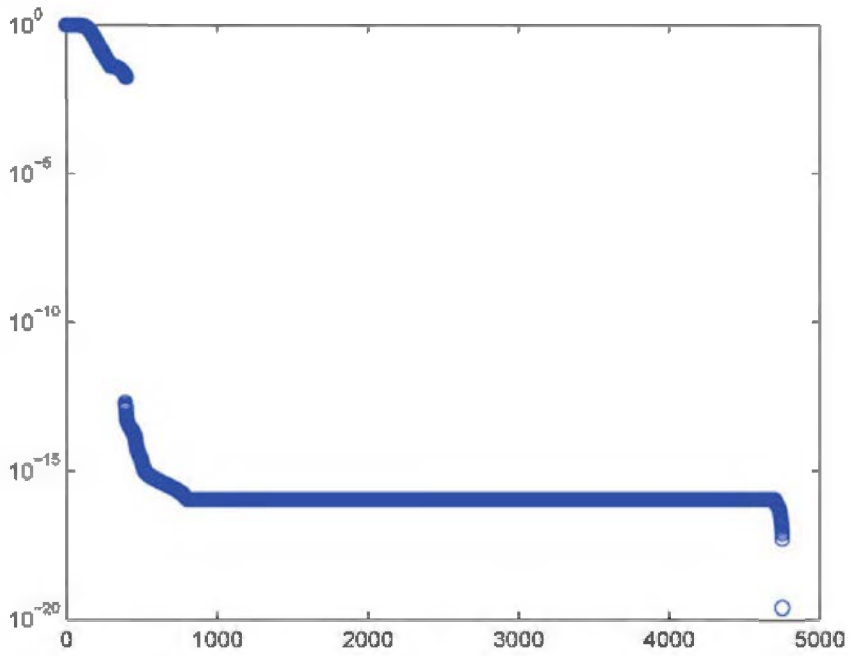


Fig. 1.16. Diagonal elements of S

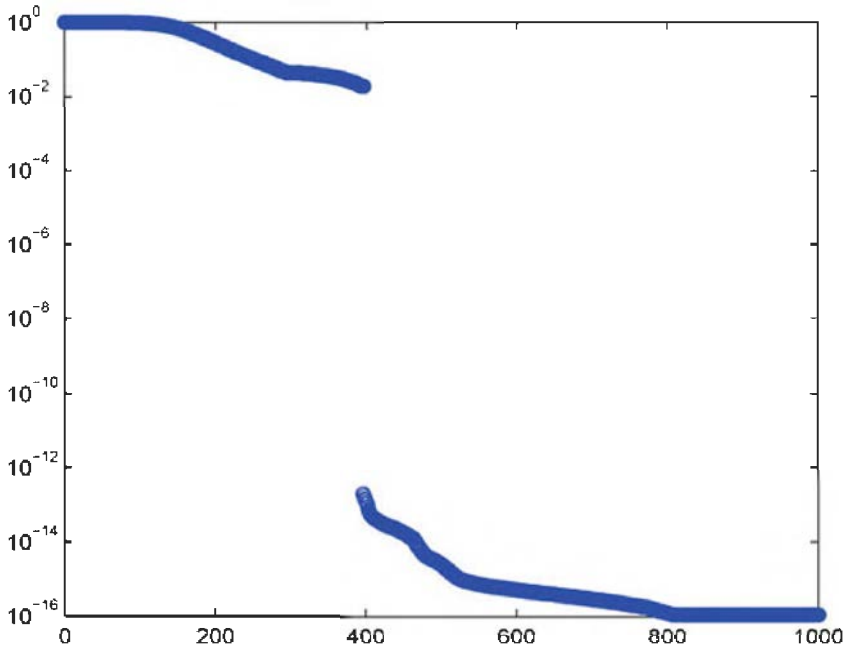


Fig. 1.17. First 1000 diagonal elements of S

and plot the related transfer functions, together with the transfer function related to $\mathbf{K}_{12}$ (blue) in Fig. 1.18. One can observe that the transfer function related to $\mathbf{K}^{(p)}$ closely approximates p peaks of the original transfer function (the one for $\mathbf{K}_{12}$).

1.5.9 The GSVD Approximation of $\mathbf{M}_{11}^{-1}\mathbf{K}_{12}$

In fact, we need to apply the GSVD to $\mathbf{M}_{11}^{-1}\mathbf{K}_{12}$ rather than $\mathbf{K}_{12}$. Fortunately, there is a straightforward relation between the GSVD of $(\mathbf{K}_{11}, \mathbf{K}_{12})$ and $(\mathbf{M}_{11}^{-1}\mathbf{K}_{11}, \mathbf{M}_{11}^{-1}\mathbf{K}_{12})$. To see this, abbreviate $\mathbf{K} := \mathbf{K}_{12}$ and $\mathbf{M} := \mathbf{M}_{11}$ and observe that

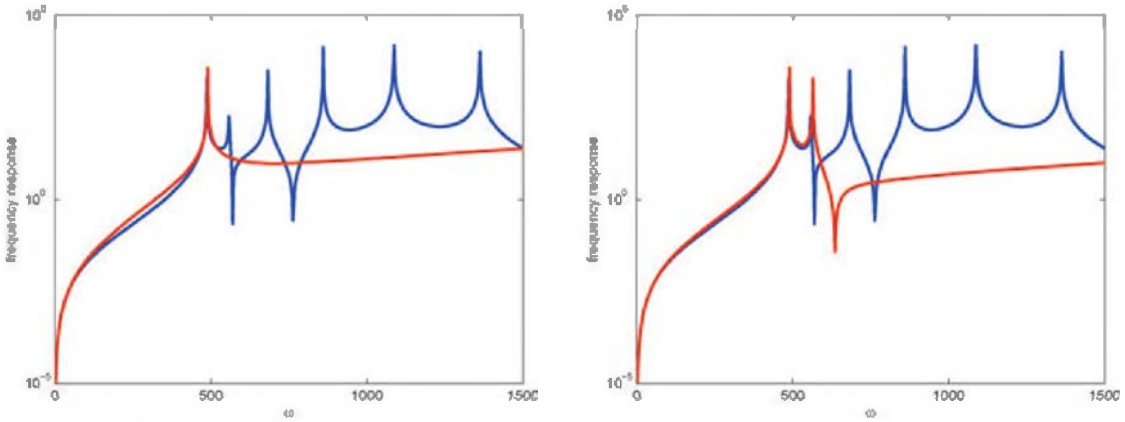
$$\begin{aligned}
\mathbf{K}^{\mathrm{T}} &= \mathbf{V}\mathbf{S}\mathbf{X}^{\mathrm{T}} \implies \\
\mathbf{K} &= \mathbf{X}\mathbf{S}^{\mathrm{T}}\mathbf{V}^{\mathrm{T}} \implies \\
\mathbf{M}^{-1}\mathbf{K} &= \mathbf{M}^{-1}\mathbf{X}\mathbf{S}^{\mathrm{T}}\mathbf{V}^{\mathrm{T}} \implies \\
\mathbf{M}^{-1}\mathbf{K} &= \underbrace{(\mathbf{M}^{-1}\mathbf{X})\sqrt{\mathbf{S}^{\mathrm{T}}}}_{\mathbf{Y}}\underbrace{\sqrt{\mathbf{S}^{\mathrm{T}}}\mathbf{V}^{\mathrm{T}}}_{\mathbf{Z}}
\end{aligned}$$

which leads to the principal component based approximation:

$$\mathbf{M}^{-1}\mathbf{K} \doteq \mathbf{M}^{-1}\mathbf{X}^{(p)}\mathbf{S}^{(p)}\mathbf{V}^{(p)}.$$

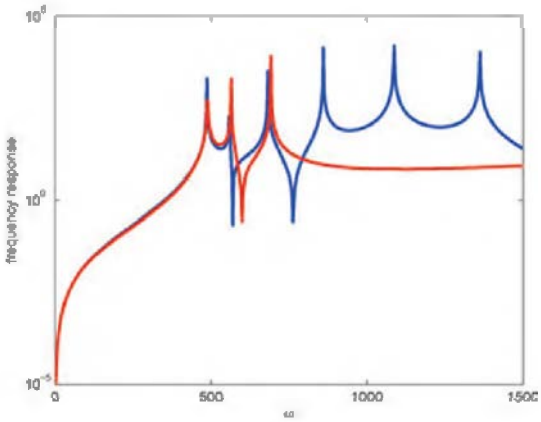
One first rewrites (1.53) to produce the term $\mathbf{sI}$, for instance as follows:

$$\begin{aligned}
H(w) &= \mathbf{c}(\mathbf{K} - w^2\mathbf{M})^{-1}\mathbf{b} \implies \\
H(w) &= \mathbf{c}(\mathbf{M}^{-1}\mathbf{K} - w^2\mathbf{I})\mathbf{M}^{-1}\mathbf{b} \implies \\
H(w) &= -\mathbf{c}(w^2\mathbf{I} - \mathbf{M}^{-1}\mathbf{K})\mathbf{M}^{-1}\mathbf{b}.
\end{aligned} \tag{1.58}$$

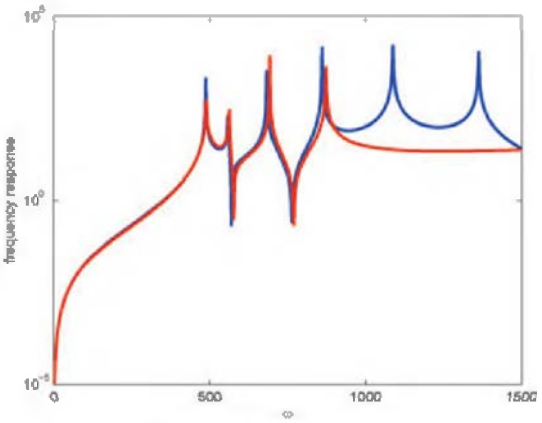


(a) rank 1 approx. of K_{12}

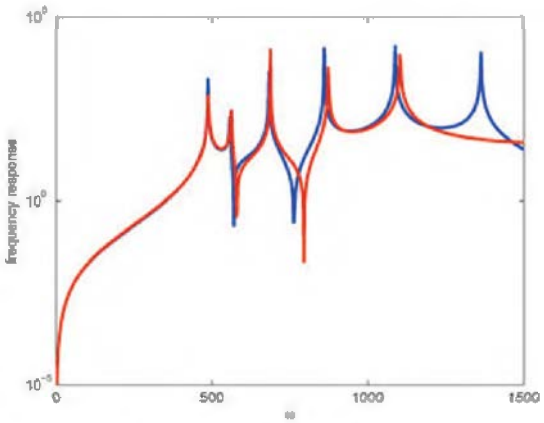
(b) rank 2 approx. of K_{12}



(c) rank 3 approx. of K_{12}



(d) rank 4 approx. of K_{12}



(e) rank 5 approx. of K_{12}

Fig. 1.18. Low-rank approximations of block K_{12}

Observe that the inverse of block-matrix $\mathbf{M}$ in (1.51) is

$$\mathbf{M}^{-1} = \begin{bmatrix} \mathbf{M}_{11}^{-1} & \mathbf{0} \\ -\mathbf{M}_{22}^{-1}\mathbf{M}_{21}\mathbf{M}_{11}^{-1} & \mathbf{M}_{22}^{-1} \end{bmatrix}$$

whence

$$\mathbf{M}^{-1}\mathbf{K} = \begin{bmatrix} \mathbf{M}_{11}^{-1}\mathbf{K}_{11} & \mathbf{M}_{11}^{-1}\mathbf{K}_{12} \\ -\mathbf{M}_{22}^{-1}\mathbf{M}_{21}\mathbf{M}_{11}^{-1}\mathbf{K}_{11} & -\mathbf{M}_{22}^{-1}\mathbf{M}_{21}\mathbf{M}_{11}^{-1}\mathbf{K}_{12} + \mathbf{M}_{22}^{-1}\mathbf{K}_{22} \end{bmatrix}$$

Now, SBR applied to the first row of this system leads to the approximation

$$\mathbf{M}^{-1}\mathbf{K} \doteq \begin{bmatrix} \mathbf{M}_{11}^{-1}\mathbf{K}_{11} & \mathbf{M}_{11}^{-1}\mathbf{X}^{(p)}\mathbf{S}^{(p)}\mathbf{V}^{(p)} \\ -\mathbf{M}_{22}^{-1}\mathbf{M}_{21}\mathbf{M}_{11}^{-1}\mathbf{K}_{11} & -\mathbf{M}_{22}^{-1}\mathbf{M}_{21}\mathbf{M}_{11}^{-1}\mathbf{K}_{12} + \mathbf{M}_{22}^{-1}\mathbf{K}_{22} \end{bmatrix}$$

which shows that one can use the GSVD-based approximation

$$H(w) \doteq \mathbf{c}(\mathbf{K}^{(p)} - w^2\mathbf{M})^{-1}\mathbf{b}$$

where

$$\mathbf{K} = \begin{bmatrix} \mathbf{K}_{11} & \mathbf{X}^{(p)}\mathbf{S}^{(p)}\mathbf{V}^{(p)} \\ \mathbf{0} & \mathbf{K}_{22} \end{bmatrix}.$$

1.6 Conclusions

We proposed a new model order reduction technique for coupled systems. Our method, called the Separate Bases Reduction (SBR) algorithm, belongs to the family of block-structure preserving (BSP) reduction techniques based on the uncoupled formulation of the coupled problem. However, unlike other reduction approaches dealing with the separate sub-system representation, the SBR algorithm can be applied to a wide category of coupled systems, including strongly coupled systems and interconnected systems with many interconnections. This is due to the fact that for such cases we avoid a too fast growth of the reduction bases and related reduced-order model, as long as the coupling can be well approximated by a relatively small number of GSVD principal components. Examples of such strongly coupled systems are systems with an interface coupling, for instance systems describing interactions between a fluid and a solid wall, or systems which for instance describe an electromagnetic-structural coupling in an electronic device. Another advantage of the proposed technique is that it is computationally cheaper than the more common BSP reduction methods which deal with the coupled formulation of the system.

For the initial version of the SBR algorithm (without low-rank approximations of the couplings), we proved the moment matching property. The GSVD based approximation of the couplings only approximates the moments, but numerical experiments show that taking a sufficient number of dominant components still results in accurately approximated moments. What makes the SBR algorithm universal, is the fact, that it can be applied even if the internal input and output matrices are not known explicitly. We show, that having at our disposal only the coupled system's matrices, external inputs and outputs, and the dimensions of the sub-systems, we are able to create appropriate Krylov subspaces for each sub-system. This property of the

reduction method is desirable when dealing with industrial problems for which the separate sub-systems' information may not be available.

The SBR method has been designed keeping in mind the practical use in an industrial environment. It is fairly straightforward to adapt existing software modules and make them suitable for application of SBR. This is certainly not the case for the BSP type methods. Although the reduced-order models obtained by application of the BSP methods frequently show a bit better approximation accuracy, the SBR algorithm is much more beneficial from the point of view of the computational time. This property is especially valuable in case of large industrial applications.

References

1. Bai, Z.: Krylov subspace techniques for reduced-order modeling of large-scale dynamical systems. Applied Numerical Mathematics **43**(1–2), 9–44 (2002)
2. Bai, Z., Li, B., Su, Y.: A unified Krylov projection framework for structure-preserving model reduction. In: Schilders, W.H.A., van der Vorst, H.A., Rommes, J. (eds) Model Order Reduction: Theory, Research Aspects and Applications. Mathematics in Industry, Vol. 13, pp. 75–93. Springer-Verlag, Berlin Heidelberg (2008)
3. Bisseling, R.H.: Parallel Scientific Computation, A Structured Approach using BSP and MPI. Oxford Scholarship Online (2007)
4. Doris, A., van de Wouw, N., Heemels, W.P.M.H., Nijmeijer, H.: A disturbance attenuation approach for continuous piecewise affine systems: Control design and experiments. J. Dynamic Systems, Measurement and Control 132(4), 044502-1– 044502-7 (2010)
5. Fernández Villena, J., Schilders, W.H.A., Miguel Silveira, L.: Order reduction techniques for coupled multi-domain electromagnetic based models. CASA report (2008)
6. Freund, R.W.: SPRIM: structure-preserving reduced-order interconnect macromodeling. In: Proceedings of the IEEE/ACM International Conference on Computer Aided Design, pp. 80–87 (2004). DOI 10.1109/ICCAD.2004.1382547
7. Géradin, M., Rixen, D.J.: Mechanical Vibrations and Structural Dynamics, 2nd ed. John Wiley and Sons, Chichester (1997)
8. Grimme, E.J.: Krylov projection methods for model reduction. PhD thesis, University of Illinois (1997)
9. He, L., Yu, H., Tan, S.X.D.: Block structure preserving model order reduction. BMAS – IEEE Behavioral Modeling and Simulation Workshop (2005)
10. Henderson, H.V., Searle, S.R.: On deriving the inverse of a sum of matrices. SIAM Review **23**(1), 53–60 (1981)
11. Heres, P.J., Deschrijver, P.J., Schilders, W.H.A., Dhaene, T.: Combining krylov subspace methods and identification-based methods for model order reduction. International Journal of Numerical Modelling: Electronic Networks, Devices and Fields **20**(6), 271–282 (2007)
12. Ionutiu, R.: Model Order Reduction for Multi-terminals Systems with Applications to Circuit Simulation. PhD thesis, Eindhoven University of Technology, Eindhoven, The Netherlands (2011)
13. Lutowska, A.: Model Order Reduction for Coupled Systems using Low-rank Approximations. PhD thesis, Eindhoven University of Technology (2012)

14. Pavlov, A.V., van de Wouw, N., Nijmeijer, H.: Uniform Output Regulation of Nonlinear Systems. Birkhäuser, Bostgon (2005)
15. Rochus, V., Rixen, D.J., Golinval, J.-C.: Electrostatic coupling of mems structures: transient simulations and dynamic pull-in. Nonlinear Analysis **63**(5–7), 1619–1633 (2005)
16. Saad, Y., Schulz, M.H.: GMRES: a generalized minimal residual algorithm for solving nonsymmetric linear systems. SIAM Journal on Scientific and Statistical Computing **7**, 856–869 (1986)
17. Salimbahrami, B., Lohmann, B.: Krylov subspace methods in linear model order reduction: Introduction and invariance properties. Sci. Rep. Inst. of Automation (2002)
18. Schilders, W.H.A., van der Vorst, H.A., Rommes, J.: Model Order Reduction: Theory, Research Aspects and Applications. Mathematics in Industry, Vol. 13. Springer-Verlag Berlin Heidelberg (2008)
19. Skogestad, S., Postlethwaite, I.: Multivariable feedback control, Analysis and Design. John Wiley and Sons, Philadelphia (2005)
20. Sonneveld, P.: CGS, a fast Lanczos-type solver for nonsymmetric linear systems. SIAM Journal on Scientific and Statistical Computing **10**, 36–52 (1989)
21. Ugryumova, M.V.: Model Order Reduction for Multi-terminals Systems with Applications to Circuit Simulation. PhD thesis, Eindhoven University of Technology, Eindhoven, The Netherlands (2000)
22. van der Vorst, H.A., Sleijpen, G.L.G., Fokkema, D.R.: Bicgstab(l) and other hybrid bi-cg methods. Numerical Algorithms **7**, 75–109 (1994)
23. van de Wouw, N., de Kraker, A., van Campen, D.H. Nijmeijer, H.: Non-linear dynamics of a stochastically excited beam system with impact. Int. J. Non-Linear Mech. **38**(5), 767–779 (2003)
24. Vandendorpe, A., Van Dooren, P.: Model reduction of interconnected systems. In: Schilders, W.H.A., van der Vorst, H.A., Rommes, J. (eds) Model Order Reduction: Theory, Research Aspects and Applications. of Mathematics in Industry, Vol. 13, pp. 305–321. Springer-Verlag, Berlin Heidelberg (2008)
25. Wesseling, P.: An Introduction to Multigrid Methods. John Wiley & Sons, Chichester (1992)