Lagrange Multiplier State-Space Substructuring

Lagrange Multiplier State-Space Substructuring

R S O Dias[1,*], M Martarelli[1] and P Chiariotti[2]

[1] Department of Industrial Engineering and Mathematical Sciences, Università Politecnica delle Marche, 60100 Ancona, Italy

[2] Department of Mechanical Engineering, Politecnico di Milano, 20156 Milan, Italy

[*] Corresponding author: R S O Dias, r.dasilva@staff.univpm.it

Abstract. In this article a new state-space coupling formulation, named Lagrange Multiplier State-Space Substructuring (LM-SSS), is presented. This method represents an evolution of the classical state-space substructuring method (classical SSS). Likewise the well-known Lagrange Multiplier Frequency Based Substructuring (LM FBS) method, the LM-SSS is achieved by using a dual assembly formulation, which means that the full set of degrees of freedom (DOFs) is retained. The equilibrium and compatibility conditions are established by directly considering the connecting forces. The LM-SSS method has demonstrated to be simple to understand and to implement in a computational environment. Furthermore, since the formulation uses a mapping matrix to enforce the coupling conditions, several substructures can be coupled at same time. Similar to classical SSS, this method is not able to directly compute a minimal-order coupled state-space model. However, two post-processing procedures are presented in order to compute a minimal realization of the obtained coupled state-space models by using these techniques.

By coupling numerical data is found that when dealing with state-space models estimated from noise-free data the same solutions are obtained by using the classical SSS or the LM-SSS. However, by comparing both methods is clear that the LM-SSS may have advantages over the classical method when dealing with state-space models estimated from experimental data, since it requires the inversion of only one matrix, thus reducing the instability of the associated numerical problems.

Keywords: Dynamic Substructuring, State-Space Substructuring, Dual Assembly Formulation, State-Space Models

1. Introduction

To characterize the dynamic behaviour of large and complex structures is convenient to consider them as an assembly of several components. This approach is, commonly, addressed as dynamic substructuring and entails several advantages:

- The possibility to exponentially reduce the complexity and the cost inherent to experimental tests;
- The possibility to perform deeper analyses of each component, which may result in a more effective modelling and optimization.

Currently, there are two groups of dynamic substructuring techniques that have been intensively investigated. One of them is the group of modal synthesis methods, where the components to be coupled are generally characterized in the modal domain. These techniques are more suitable to be used with numerical data, e.g. data obtained from finite element models [1].

The other major group of techniques is clustered under the Frequency Based Substructuring (FBS) label. In FBS, each component to be coupled is described by its frequency response functions (FRFs), hence it is quite typical to have these methods exploited when dealing with experimental data [2].

A new class of techniques labelled State-Space Substructuring (SSS) has grown attention in the last years. In SSS the components to be coupled are described by state-space models, hence these methods are suitable to be exploited both on numerically simulated and experimental data. Su and Juang proposed the first SSS method in [3], hence it is here labelled as classical SSS. This method is based on the construction of a diagonal uncoupled state-space model by using the acceleration state-space models (state-space models that present an output vector, whose elements are accelerations) of each component. Then, by using a coupling matrix, the coupling conditions are enforced and the coupled state-space model is obtained. This method has some limitations though: firstly it requires the inversion of two different matrices to compute the coupled state-space model, secondly it is unable to compute the state-space model in a minimal-order form. This means that the obtained coupled models will contain redundant states. The presence of such states does not affect the quality of the input-output transfer function that can be computed from the coupled state-space model [3]. However, their presence yields some drawbacks: the decrease of the elegance of the model, the probable increase of the condition number of its state matrix and the increment of the computational effort when performing calculations with the computed coupled state-space model.

This paper aims at presenting a novel SSS approach named Lagrange Multiplier SSS (LM-SSS). The approach wants to tackle the drawback of classical SSS without increasing the complexity of the approach. To achieve this goal, LM-SSS exploits a dual assembly formulation.

The LM-SSS method is described in section 2. In section 3 the procedure to obtain a minimal-order coupled state-space model is presented, then the performance of LM-SSS method is evaluated by coupling numerical data (section 4). Lastly, the discussion and the conclusions are presented in sections 5 and 6, respectively.

2. Lagrange Multiplier State-Space Substructuring

Let us consider the assembled structure composed by two different components shown in figure 1.

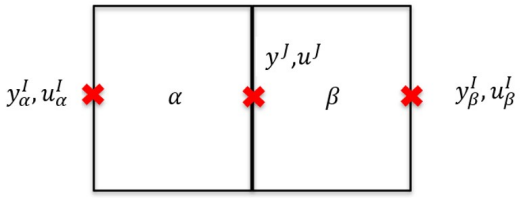

Figure 1: Assembled structure composed by two different components.

If we separate both components, two connecting forces ($\{g^J_\alpha(t)\}$ and $\{g^J_\beta(t)\}$), which are equal in intensity and opposite in direction, will appear at the interface of substructures α and β ($\{g^J_\alpha(t)\} = -\{g^J_\beta(t)\}$) (see figure 2).

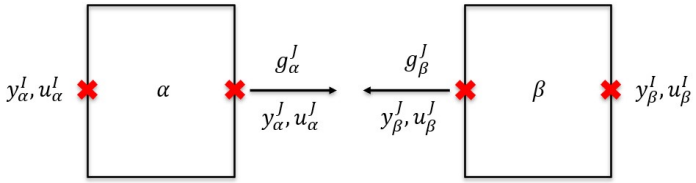

Figure 2: Separated components.

We may then establish a coupled state-space model by using the uncoupled models of each substructure. In fact, the connecting forces are applied at the interface of each component, thus, to achieve the coupled state-space model the connecting forces must be added to the interface inputs of each substructure. To construct such model, we must compute a diagonal state-space model from the models of each component and add the connecting forces to the respective interface inputs, as follows

$$\begin{Bmatrix} \{\dot{x}_\alpha(t)\} \\ \{\dot{x}_\beta(t)\} \\ \vdots \end{Bmatrix} = \begin{bmatrix} A_\alpha & & \\ & A_\beta & \\ & & \ddots \end{bmatrix} \begin{Bmatrix} \{x_\alpha(t)\} \\ \{x_\beta(t)\} \\ \vdots \end{Bmatrix} + \begin{bmatrix} B_\alpha & & \\ & B_\beta & \\ & & \ddots \end{bmatrix} \left(\begin{Bmatrix} \{u_\alpha(t)\} \\ \{u_\beta(t)\} \\ \vdots \end{Bmatrix} + \begin{Bmatrix} \{g_\alpha(t)\} \\ \{g_\beta(t)\} \\ \vdots \end{Bmatrix} \right)$$

$$\begin{Bmatrix} \{\ddot{y}_\alpha(t)\} \\ \{\ddot{y}_\beta(t)\} \\ \vdots \end{Bmatrix} = \begin{bmatrix} C_\alpha & & \\ & C_\beta & \\ & & \ddots \end{bmatrix} \begin{Bmatrix} \{x_\alpha(t)\} \\ \{x_\beta(t)\} \\ \vdots \end{Bmatrix} + \begin{bmatrix} D_\alpha & & \\ & D_\beta & \\ & & \ddots \end{bmatrix} \left(\begin{Bmatrix} \{u_\alpha(t)\} \\ \{u_\beta(t)\} \\ \vdots \end{Bmatrix} + \begin{Bmatrix} \{g_\alpha(t)\} \\ \{g_\beta(t)\} \\ \vdots \end{Bmatrix} \right) \tag{1}$$

where $\{x(t)\} \in R^{n\times 1}$ represents the state vector, $\{u\} \in R^{ni\times 1}$ represents the input vector, whose elements are forces, and $\{\ddot{y}\} \in R^{no\times 1}$ represents the acceleration output vector. The constants n, ni and no denote the number of states, inputs and outputs, respectively. $(\dot{\bullet})$ represents first order time derivate and subscripts α and β denote variables related to substructures α and β, respectively. The vectors $\{u_\alpha(t)\}$, $\{u_\beta(t)\}$, $\{g_\alpha(t)\}$ and $\{g_\beta(t)\}$ are given as follows

$$\{u_\alpha(t)\} = \begin{Bmatrix} u_\alpha^I(t) \\ u_\alpha^J(t) \end{Bmatrix} \quad (2) \qquad \{u_\beta(t)\} = \begin{Bmatrix} u_\beta^I(t) \\ u_\beta^J(t) \end{Bmatrix} \quad (3)$$

$$\{g_\alpha(t)\} = \begin{Bmatrix} 0 \\ g_\alpha^J(t) \end{Bmatrix} \quad (4) \qquad \{g_\beta(t)\} = \begin{Bmatrix} 0 \\ g_\beta^J(t) \end{Bmatrix} \quad (5)$$

Note that, the construction of the model presented on equation (1) does not impose any limit to the number of substructures to couple.

To avoid the partition of the input vectors in terms of internal and interface DOFs and for a easier implementation of the coupling technique, let us introduce the following relation [1],[2]

$$\{g(t)\} = -[B]^T\{\lambda(t)\} \quad (6)$$

where, $[B]$ is a mapping matrix and $\{\lambda(t)\}$ is the vector of Lagrange Multipliers that are associated to the constraints of our problem [4], which in this case represent the connecting forces [2]. Note that, the construction of $[B]$ matrix must guarantee the equality given in equation (6), which means that for the problem under analysis, the relation $\{g_\alpha^J(t)\} = -\{g_\beta^J(t)\}$ must be verified. This is the reason why, for each pair of connecting DOFs, the related coefficients of matrix $[B]$ must have an absolute value of 1 and be opposite in sign. This matrix must be constructed as presented in [2].

By using equation (6), the coupled state-space model given in equation (1) can be rewritten as follows

$$\begin{aligned} \{\dot{x}(t)\} &= [A_D]\{x(t)\} + [B_D](\{u(t)\} - [B]^T\{\lambda(t)\}) \\ \{\ddot{y}(t)\} &= [C_D]\{x(t)\} + [D_D](\{u(t)\} - [B]^T\{\lambda(t)\}) \end{aligned} \quad (7)$$

At this point, the coupling conditions (equilibrium and compatibility) must be established. The equilibrium condition states that at the interface, the external force applied must be equal to the sum of the internal forces [3]. This condition can be established by the output equation of the coupled state-space model given by equation (7). Solving this equation to find the value of {u(t)}, we obtain

$$\{u(t)\} = [D_D]^{-1}(\{\ddot{y}(t)\} - [C_D]\{x(t)\}) + [B]^T\{\lambda(t)\} \quad (8)$$

The compatibility condition establishes that at the interface the physical motion of the coupled substructures must be equal [3]. Thus, relative motion between the interfaces of the components to be coupled must not exist. This condition can be expressed by using the mapping matrix $[B]$ as follows [2]

$$[B]\{y(t)\} = \{0\} \quad (9)$$

By calculating the second order time derivate of equation (9), we obtain the following expression

$$[B]\{\ddot{y}(t)\} = \{0\} \tag{10}$$

Resuming the equilibrium condition and the second order time derivate of the compatibility condition, given by equations (8) and (10), respectively, and dropping $\{\bullet\}$, $[\bullet]$ and (t) for ease of readability, we can then write the two conditions as in equation (11).

$$\begin{cases} (D_D)^{-1}(\ddot{y} - C_D x) + B^T\lambda = u \\ B\ddot{y} = 0 \end{cases} \tag{11}$$

After performing some mathematical manipulations with the system of equations (11), we arrive to the following relations

$$\begin{cases} \lambda = (BD_DB^T)^{-1}(BC_Dx + BD_Du) \\ \ddot{y} = (C_D - D_DB^T(BD_DB^T)^{-1}BC_D)x + (D_D - D_DB^T(BD_DB^T)^{-1}BD_D)u \end{cases} \tag{12}$$

Inserting the value of $\{u(t)\}$ given by equation (8) in the state equation of the coupled state-space model (equation (7)), we obtain

$$\dot{x} = A_Dx + B_D((D_D)^{-1}(\ddot{y} - C_Dx) + B^T\lambda - B^T\lambda) \tag{13}$$

By using the bottom equation of the system of equations (12) and after some mathematical manipulations, equation (13) can be rewritten as follows

$$\dot{x} = (A_D - B_DB^T(BD_DB^T)^{-1}BC_D)x + (B_D - B_DB^T(BD_DB^T)^{-1}BD_D)u \tag{14}$$

From equation (14) and the bottom equation of the system of equations (11), we obtain the following coupled state-space model

$$\begin{aligned} \{\dot{x}(t)\} &= [\bar{A}]\{x(t)\} + [\bar{B}]\{u(t)\} \\ \{\ddot{y}(t)\} &= [\bar{C}]\{x(t)\} + [\bar{D}]\{u(t)\} \end{aligned} \tag{15}$$

where, the coupled state-space matrices can be computed from the diagonal coupled state-space model given by expression (7), as follows

$$\begin{aligned} [\bar{A}] &= A_D - B_DB^T(BD_DB^T)^{-1}BC_D \\ [\bar{B}] &= B_D - B_DB^T(BD_DB^T)^{-1}BD_D \\ [\bar{C}] &= C_D - D_DB^T(BD_DB^T)^{-1}BC_D \\ [\bar{D}] &= D_D - D_DB^T(BD_DB^T)^{-1}BD_D \end{aligned} \tag{16}$$

Analyzing equations (16), we can observe that we just need to perform the inversion of the matrix $[BD_DB^T]$ in order to compute the coupled state-space model, which might represent an important advantage over the classical SSS, when dealing with experimentally acquired data.

3. Minimal-Order Coupled State-Space Models

The classical SSS and LM-SSS have the limitation of not imposing any constraints to the states, which means that the obtained coupled state-space models will present redundant states. For each pair of connected DOFs, one of them should be eliminated from the coupled state-space model, and, thus, one state and its correspondent first derivative must be also eliminated. In this section, we will propose two different post-processing procedures to eliminate the extra states present in the obtained coupled state-space models.

Let us assume that the state-space models that are under analysis are written in physical coordinates (i.e. $\{x(t)\} = \{y(t)\}$). For this kind of models, the elimination of the redundant states can be performed by using a Boolean localization matrix, $[L_T]$, similar to the one presented in [1]. This matrix was already used by Gibanica in [5] to translate the state-space coupling method developed in [6] into the general framework presented by de Klerk et al. in [1]. To compute $[L_T]$, we must start by computing a mapping matrix, $[B_T]$, for which the following relation is true

$$[B_T]\{x(t)\} = \{0\} \tag{17}$$

Then, $[L_T]$ can be calculated as follows [5]

$$[L_T] = null(B_T) \tag{18}$$

By using $[L_T]$ we may establish the following relation between the state vector of the original coupled state-space model and the state vector of the minimal-order model (see [1]), as follows

$$\{x(t)\} = [L_T]\{z(t)\} \tag{19}$$

where, $\{z(t)\}$ represents the minimal realization of $\{x(t)\}$.

By using equation (19), a general non minimal-order coupled acceleration state-space model can be rewritten as follows

$$\begin{aligned} [L_T]\{\dot{z}(t)\} &= [A][L_T]\{z(t)\} + [B]\{u(t)\} \\ \{\ddot{y}(t)\} &= [C][L_T]\{z(t)\} + [D]\{u(t)\} \end{aligned} \tag{20}$$

By using the Moore-Penrose pseudoinverse of matrix $[L_T]$, the state equation of the state-space model (20) can be rewritten as follows

$$\begin{aligned} \{\dot{z}(t)\} &= [L_T]^{\dagger}[A][L_T]\{z(t)\} + [L_T]^{\dagger}[B]\{u(t)\} \\ \{\ddot{y}(t)\} &= [C][L_T]\{z(t)\} + [D]\{u(t)\} \end{aligned} \tag{21}$$

where, superscript † represents the pseudoinverse of a matrix.

The state-space model given by expression (21), represents the minimal realization of the original coupled state-space model. Even though, the use of the pseudoinverse of

matrix $[L_T]$ might be seen as a drawback for the presented procedure to eliminate the extra states, this is not the case, as we will prove hereafter.

$[L_T]$ matrix is a rectangular matrix of dimensions $n \times n_z$, with $n > n_z$, whose $rank = n_z$ (being n the number of states of the original coupled state-space model and $n_z = n - rank(B_T)$ the number of states of the minimal realization of the same model). Thus, its pseudoinverse can be calculated as follows [7]

$$[L_T]^\dagger = (L_T^T L_T)^{-1} L_T^T \tag{22}$$

Where, $[L_T]^T[L_T]$ is a diagonal and invertible matrix, if $[L_T]$ is an orthogonal basis for the nullspace of $[B_T]$, as shown below [8].

$$[L_T]^T[L_T] = \begin{bmatrix} v_1^T \\ v_2^T \\ \vdots \end{bmatrix} \begin{bmatrix} v_1 & v_2 & \dots \end{bmatrix} = \begin{bmatrix} v_1 \bullet v_1 & & \\ & v_2 \bullet v_2 & \\ & & \ddots \end{bmatrix} \tag{23}$$

Note that, assuming that $[L_T]$ is an orthogonal basis $v_i \bullet v_j = 0,\ \forall\ i \neq j$, where $\bullet$ represents dot product and v_i represents the i^{th} column of $[L_T]$ matrix. For the particular case of this matrix be an orthonormal basis, $[L_T]^T[L_T] = [I]$.

Concerning what was presented, it is advised that $[L_T]$ be always computed to be an orthogonal basis for the nullspace of $[B_T]$. In this way, since the pseudoinverse of this matrix can be replaced by equation (22) and matrix $[L_T]^T[L_T]$ will always be invertible, no numerical problems will be faced due to the computation of $[L_T]^\dagger$, and the $[L_T]^\dagger$ calculated as in equation (22) will act as the left inverse, hence $[L_T]^\dagger[L_T] = [I]$. Moreover, since this matrix will always be diagonal, the computational effort required to compute $(L_T^T L_T)^{-1}$ is decreased. These facts suggest that this method might be a reliable choice to perform the elimination of the redundant states.

Other option to perform the states elimination is the use of a manual procedure, that mimics the function of $[L_T]$ matrix. For each pair of redundant sates we must proceed as follows:

- Sum the columns of $[\bar{A}]$ matrix that are being multiplied by the pair of redundant states in analyze, then perform the same for the columns of $[\bar{C}]$ matrix;
- Eliminate the row and column of matrix $[\bar{A}]$ related to one of those redundant states;
- Eliminate the same row and column of matrices $[\bar{B}]$ and $[\bar{C}]$, respectively;
- Repeat the procedure for the first derivative of the analyzed pair of redundant states.

By following this procedure, we arrive to the minimal realization of the coupled state-space model without the use of $[L_T]$ matrix.

4. Validation of LM-SSS method

In this section, the performance of the LM-SSS method is evaluated by coupling the two substructures presented in figure 3.

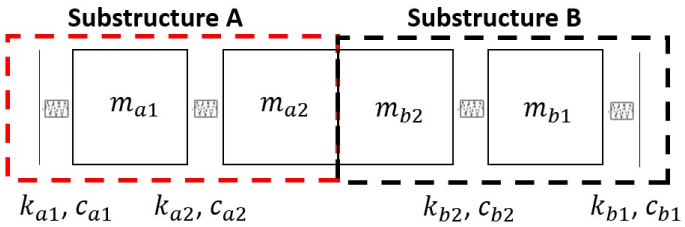

Figure 3: Assembled structure composed by two substructures.

The value of the physical parameters represented in figure 3, are given in table 1.

Table 1: Physical parameter values

i	m_i (kg)	c_i (Nsm^{-1})	k_i (Nm^{-1})
$a1$	9	40	4×10^7
$a2$	4	25	3×10^7
$b1$	5	50	5×10^7
$b2$	7	15	2×10^7

The results obtained by LM-SSS will be compared with the analytical solution and with the one obtained by using classical SSS. The exact FRFs of the assembled structure are going to be computed from its analytical state-space model. Therefore, the analytical state-space models of the assembled structure and of substructures A and B must be computed.

To calculate the analytical state-space models, we are going to use the Lagrange equations [9]. These equations permit the achievement of the stiffness, damping and mass matrices, needed to establish the acceleration state-space models written in physical coordinates. For simplicity, the damping matrix will be considered equal to the stiffness matrix, but instead of being composed by stiff therms it will be composed by damping ones.

Then, LM-SSS and classical SSS methods are used to estimate the coupled state-space model of the assembled structure (see figure 3). Afterwards, the presented procedures in section 3 are used to obtain a minimal realization of the original coupled state-space model, being found that both produce the same minimal-order model. It was found that the minimal-order state-space models obtained by using LM-SSS and classical SSS were exactly equal and close to the analytical one. Such a small mismatch

might be due to possible round-off errors. Accelerance FRFs were computed from the obtained coupled models and from the analytical one.

In figure 4, the exact FRF, whose output is the DOF related to m_{b1} and the input is the interface DOF, is compared to the same FRF computed from the calculated state-space models by using LM-SSS and classical SSS. The poles of the coupled state-space models obtained by using LM-SSS and classical SSS methods (identified poles) are compared with the exact poles of the assembled structure in table 2.

By observing figure 4 is possible to conclude that the 3 plotted FRFs are very well matching. Moreover, table 2 shows that the identified poles are closely matching the exact ones, which validates the LM-SSS method developed in section 2.

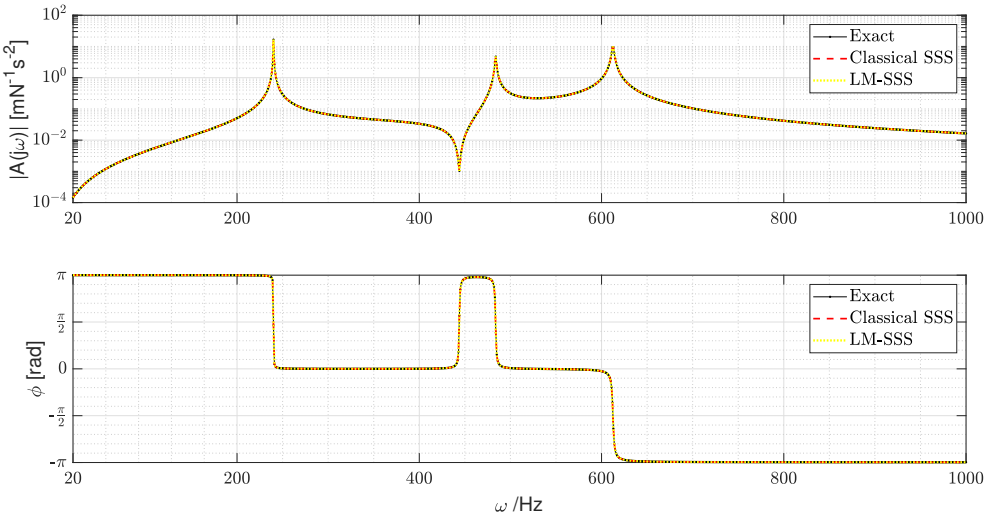

Figure 4: Accelerance FRF of the assembled system, whose output is the DOF related to m_{b1} and the input is the interface DOF.

Table 2: Comparison of the identified poles and exact ones of the assembled structure

LM-SSS	Classical SSS	Exact
$-6,7117 \pm 3,8484 \times 10^3 j$	$-6,7117 \pm 3,8484 \times 10^3 j$	$-6,7117 \pm 3,8484 \times 10^3 j$
$-4,1799 \pm 3,0396 \times 10^3 j$	$-4,1799 \pm 3,0396 \times 10^3 j$	$-4,1799 \pm 3,0396 \times 10^3 j$
$-1,0377 \pm 1,5078 \times 10^3 j$	$-1,0377 \pm 1,5078 \times 10^3 j$	$-1,0377 \pm 1,5078 \times 10^3 j$

5. Discussion

From section 4 we may conclude that both LM-SSS and classical SSS methods provide exactly the same results for coupling of state-space models identified from noise-

free data. However, is obvious that LM-SSS presents some advantages over the classical method. The most important advantages entailed by LM-SSS are the requirement of the inversion of just one matrix to obtain the coupled state-space model and the easy implementation, since it does not require the partition of the substructures state-space models in terms of internal and interface inputs and outputs.

Conversely, we may also possibly point out a drawback of LM-SSS when compared with classical SSS. As previously stated, LM-SSS technique is obtained by using a dual assembly formulation, which means that the full set of interface DOFs is retained [2]. Therefore, for each pair of connected DOFs, we will obtain in the coupled state-space model, two inputs and outputs that represent exactly the same physical quantity. In order to remove these redundant outputs and inputs, for each pair of coupled DOFs we must remove a column of matrices $[B]$ and $[D]$ and a row of matrices $[C]$ and $[D]$ related to one of that DOFs. This procedure can easily be performed even when coupling experimentally acquired data, because it is always possible to understand where these DOFs will be placed in the coupled state-space model.

Is important to mention that to unleash the true potential of LM-SSS method, this technique must be evaluated by coupling state-space models, previously transformed into coupling form (see [6] and [10]). By transforming the state-space models before coupling them, and by using the post-processing procedures outlined in section 3, LM-SSS would be able to obtain minimal-order state-space models, when dealing with numerical or experimental data. In this way, LM-SSS would merge several advantages of different SSS methods. It turned out to be easy to implement, and capable to be used for coupling an unlimited number of substructures at the same time; moreover, it just requires the inversion of one matrix to compute the coupled state-space model and it makes it possible to get minimal-order models. The major drawback would be the necessity of performing the post-processing procedure described in section 3. However the procedure that uses the Boolean localization matrix, $[L_T]$, can be easily implemented and, probably, would require less effort than coupling several substructures step by step as required by the technique proposed in [6]. Even though, this procedure requires the inversion of $[L_T]$ matrix, this is not problematic and will not induce any numerical problems as proved in section 3.

6. Conclusion

A new state-space substructuring technique denominated LM-SSS was developed in this document. This method represents an evolution of classical SSS and has been developed by tacking LM-FBS method as the main source of inspiration. LM-SSS showed to be a promising technique, since it proved to be as efficient as classical SSS, with the further advantage of reducing the numerical complexity of the problem. However, the evaluation of its performance when dealing with more complex and experimental data is fundamental to better understand its potential.

Funding

This project has received funding from the European Union's Framework Programme for Research and Innovation Horizon 2020 (2014-2020) under the Marie Sklodowska-Curie Grant Agreement nº 858018.

References

[1] D. de Klerk, D.J. Rixen, and S.N. Voormeeren. General framework for dynamic substructuring: History, review, and classification of techniques. *AIAA J.*, 46(5):1169–1181, 2008.

[2] Dennis de Klerk, Daniel J. Rixen, and Jasper de Jong. The frequency based substructuring (fbs) method reformulated according to the dual domain decomposition method. *Proceedings of the XXIV International Modal Analysis Conference, St. Louis, MO*, 2006.

[3] Tzu-Jeng Su and Jer-Nan Juang. Substructure system identification and synthesis. *Journal of Guidance, Control and Dynamics*, 17(5):1087–1095, 1994.

[4] Daniel J. Rixen. *Substructuring and Dual Methods in Structural Analysis*. Université de Liège, Liège, Belgium, 1997.

[5] Mladen Gibanica, Anders T. Johansson, Anders Liljerehn, Per Sjövall, and Thomas Abrahamsson. Experimental-analytical dynamic substructuring of ampair testbed: A state-space approach.in: Allen M., Mayes R., Rixen D. (eds) Dynamics of Coupled Structures. *Conference Proceedings of the Society for Experimental Mechanics Series. Springer, Cham.*, 1:1–14, 2014.

[6] Per Sjövall and Thomas Abrahamsson. Component system identification and state-space model synthesis. *Elsevier*, pages 2697–2714, 2007.

[7] Marco A. M. Lasso. Linear vector space derivation of new expressions for the pseudo inverse of rectangular matrices. *Journal of Applied Research and Technology*, 5(3):150–159, 2007.

[8] Howard Anton and Chris Rorres. *Elementary linear algebra : applications version*. WILEY, 11^{th} edition, 2013.

[9] H.R. Harrison and T. Nettleton. *Advanced Engineering Dynamics*. Butterworth-Heinemann, 1^{st} edition, 1997.

[10] Matthew Allen, Daniel Rixen, Maarten van der Seijs, Paolo Tiso, Thomas Abrahamsson, and Randal Mayes. *Emerging Numerical and Experimental Techniques*. Springer, 2020.