A Output-Based Approach for the Control of Dynamically Substructured Systems with Real-Time Finite Element Substructures

J.Y. Tu & J.Y. Jiang

Advanced Control and Dynamic Test Laboratory, Department of Power Mechanical Engineering, National Tsing Hua University, Taiwan



SUMMARY:

Structural tests using dynamic substructuring methods require a robust controller to synchronise the numerical and physical responses, and a stable and accurate algorithm to simulate the numerical components. The proposed new substructuring strategies use real-time finite element methods to guarantee modelling accuracy and output-based controllers to achieve robust synchronisation. Output-based controller design is generic and requires no *a priori* information of the specimen's dynamics. A beam-spring-damper system is developed as an example for illustration of the concepts, where the beam is simulated numerically and the spring-damper is tested physically. Iterative design procedure is required in order to achieve a trade-off between mesh density and sampling time. The result of this work offers the prospect of testing engineering systems with large-dimension or multiple finite element components in the future.

Keywords: substructuring, finite element methods, state-space techniques, disturbance cancellation

1. INTRODUCTION

World-wide interest in advanced dynamic testing techniques has been expedited by issues and concerns about efficiency, cost-reduction, safety, and sustainability of a broad range of engineering systems. Hybrid experimental method of dynamically substructured systems (DSS) or dynamic substructuring is proposed for structural testing (Horiuchi et al., 1999). During a DSS process, the critical (nonlinear) specimens are physically tested, and together with additional actuators (G_{TS}) are called physical substructures (Σ_P). The remaining and well-understood components are simulated in real time as numerical substructures (Σ_N). Thus, DSS testing environment enables the Σ_P design to be rectified and optimised, resulting in deeper insights into critical parameters, before the products are actually manufactured and implemented.

To pursue the success of DSS tests, a high-fidelity and robust controller is required to synchronise the Σ_N and Σ_P outputs, and a fast, stable, and accurate numerical algorithm is essential to model the numerical components. As a result, this work proposes relatively new substructuring strategies, with the synchronisation robustness being assured by output-based (OB) controllers and the simulation accuracy by the finite element methods (FEM).

Finite element (FE) analysis is widely applied in the simulation studies of structural, automotive, and aircraft systems. In the analysis, a component is meshed by a number of elements, and thus a continuous equation of motion is solved and transformed into discretised equations. Simulation accuracy can be promoted by refining the mesh diagram or selecting higher-order polynomials. However, using the FEM for dynamic substructuring may cause the following concern: (1) the mesh density related to simulation accuracy relies on computational capacity; (2) computer-aided engineering tools, such as ANSYS, cannot be implemented in a real-time process. These issues render the FEM is rarely discussed in the DSS literature, except Shing et al. (2004), Wang et al. (2006), and Stauffer et al. (2007). Since the issue of computation load related to mesh density can be mostly solved by using higher-level computation technologies, such as super computers and Mathworks XPC

target, in this phase, we emphasise the research work on the technical and theoretical issues of integrating the FEM and OB controllers in a real-time process.

The content of this paper is structured as follows. In Section 2, the innovative OB strategies are presented based on a substructured framework, including the synthesis of OB substructured dynamics and controllers. Control systems can be designed in transfer-function (TF) or state-space (SS) forms, depending on the implementation condition. Section 3 introduces a beam-spring-damper (BSD) structure as an example to construct the FE emulation in ANSYS and in Matlab/Simulink. The BSD system is decomposed into two substructures in Section 4, where the beam is modelled numerically using the FEM and the spring-damper device is tested physically, followed by the associated dynamics and control synthesis. Section 5 compares the results of the ANSYS simulation v.s. the real-time FE and OB substructuring methods. Finally, conclusion and future work of a new FE substructuring scheme are drawn in Section 6.

2. THE OUTPUT-BASED DYNAMICALLY SUBSTRUCTURED SYSTEMS



Figure 1. The proposed output-based substructured framework

A substructured framework is shown in Fig. 1, which displays the essential signal flows and components within DSSs. Signals $\{z_N, z_P\}$ in Fig. 1 indicate the outputs of $\{\Sigma_N, \Sigma_P\}$ at the substructured interface, to be synchronised via the action of a outer-loop DSS control signal, labelled by *u*. A robust DSS controller can ideally drive the substructured error to zero ($x_e = z_N - z_P, x_e \rightarrow 0$). Note that Σ_{N1} and/or Σ_{P2} can contain more than one substructure, i.e. multi-input-multi-output systems, so that the signals $\{d_N, d_P, u, y_i, x_e, z_O, z_N, z_P\}$ can be vectors. The detailed introduction to the substructured framework can be referred to Tu and Jiang (2012).

The OB framework (Tu, 2012) proposes that only the nominal model of G_{TS} and the outputs of $\{\Sigma_N, \Sigma_P\}$ are considered in the synthesis procedure of DSS dynamics and controllers, regardless of the parameters and inputs related to $\{\Sigma_1, \Sigma_2\}$. Thus, $\{\Sigma_1, \Sigma_2, d_N, d_P, y_i, z_0\}$ are depicted by dotted lines in Fig. 1, assumed to be completely *unknown* by the OB controllers, and $\{\Sigma_1, \Sigma_2\}$ can contain nonlinear components. It is to be noted that the OB control scheme has been adopted implicitly in some delay-compensation research, e.g. Darby et al. (2002) and Wallace et al. (2005).

Since the OB controllers do not require knowing the dynamic expression for Σ_1 , it is advantageous for testing Σ_1 with complexity and is adopted in this work for FE dynamic substructuring. In the following sections, the OB substructured dynamics will be derived and fit into the framework in Fig. 1, using TF and SS descriptions. To be noted that, although the TF and SS dynamics and controllers are derived from an identical linear model of G_{TS} , the resulting gain synthesis and implementation conditions are different and one representation cannot be fully transformed into the other.

2.1. The OB substructured framework and control system in TF form

First, the generalised models and outputs of $\{\Sigma_{N1}, \Sigma_{P2}\}$ can be expressed by (Tu, 2012)

$$\Sigma_{\rm N1}: \qquad z_{\rm N}(s) = \underbrace{\left[G_{d\rm N}(s) - G_{i\rm N}(s)\right]}_{\Sigma_{\rm N} = \Sigma_{\rm r}} \begin{bmatrix} d_{\rm N}(s) - y_{i}(s) \end{bmatrix}^{T} = G_{d\rm N}(s)d_{\rm N}(s) + G_{i\rm N}(s)y_{i}(s)$$
(2.1)

$$\Sigma_{\rm P2}: \qquad y_i(s) = \underbrace{\left[G_{d\rm P}(s) \quad G_{\rm P}(s)\right]}_{\Sigma_2} \begin{bmatrix} d_{\rm P}(s) \quad z_{\rm P}(s) \end{bmatrix}^T = G_{d\rm P}(s) d_{\rm P}(s) + G_{\rm P}(s) z_{\rm P}(s) \tag{2.2}$$

$$z_{\rm P}(s) = G_{\rm TS}(s)u(s) \tag{2.3}$$

where $\{G_{dN}(s), G_{iN}(s), G_{dP}(s), G_{P}(s)\}\$ are the input TFs associated with $\{d_{N}, y_{i}, d_{P}, z_{P}\}\$, respectively, and z_{P} is the G_{TS} output to be synchronised with z_{N} . The signal y_{i} is fed back from Σ_{P2} to Σ_{N1} in real time, in order to compute the next time step response of z_{N} .

The nominal models of $\{\Sigma_{N1}, \Sigma_{P2}\}$ are expressed in Eqns. 2.1-2.3. However, referring to the OB scheme and Fig. 1, the OB substructured error dynamics can be derived irrespective of the $\{\Sigma_1, \Sigma_2\}$ parameters. Simply subtracting Eqn. 2.3 from $z_N(s)$ results in the OB error dynamics

$$x_{e}(s) = z_{N}(s) - z_{P}(s) = z_{N}(s) - G_{TS}(s)u(s)$$
(2.4)

where the signal z_N is always measurable from the simulation loop of Σ_{N1} , and z_P is essentially measurable via sensor devices.

The objective of substructuring control is to drive the error dynamics in Eqn. 2.4 to zero $(x_e \rightarrow 0)$, ensuring that $\{z_N, z_P\}$ are robustly synchronised. Accordingly, the OB linear substructuring controller (O-LSC) is proposed as

$$u(s) = K_{\rm N}(s)z_{\rm N}(s) + K_{eo}(s)x_{e}(s)$$
(2.5)

where $K_N(s)$ and $K_{eo}(s)$ are the feedforward and feedback gain matrices, respectively. Substituting Eqn. 2.5 into Eqn. 2.4 yields the following closed-loop error dynamics and the $K_N(s)$ design

$$x_{e}(s) = \left[I + G_{\text{TS}}(s)K_{eo}(s)\right]^{-1} \left[I - G_{\text{TS}}(s)K_{\text{N}}(s)\right] z_{\text{N}}(s)$$
(2.6)

$$K_{\rm N}(s) = G_{\rm TS}^{-1}(s) \tag{2.7}$$

Eqn. 2.6 reflects two parts of control policy, with the first part of the solution, $K_N(s)$, for disturbance cancellation and the second, $K_{eo}(s)$, for closed-loop stability. Ideally, $K_N(s)$ can cancel the unwanted error dynamics, if the G_{TS} parameters are exactly known. In the presence of parameter variations within G_{TS} , $K_{eo}(s)$ needs to be designed, for example, using the roots' loci method (Stoten and Hyde, 2006). Eqns. 2.4-2.7 explicitly use a deductive approach to verify that the primary objective of open-loop and OB substructuring control is to negate the additional G_{TS} dynamics using a model-inversion controller.

2.2. The OB substructured framework and control system in SS form

Complete SS representation of $\{\Sigma_{N1}, \Sigma_{P2}\}$ are addressed in Tu (2012), and this section focuses on using the OB strategy to derive the SS models and controllers. The parameters and signals associated with $\{\Sigma_1, \Sigma_2\}$ are not required to be known by the controllers; as a result, the SS equation for G_{TS} , or the output dynamics of Σ_{P2} is expressed by

$$\Sigma_{P2}(G_{TS} \text{ dynamics}): \begin{bmatrix} \dot{x}_{P1} \\ \dot{x}_{P2} \end{bmatrix} = \underbrace{\begin{bmatrix} A_{P11} & A_{P12} \\ A_{P21} & A_{P22} \end{bmatrix}}_{A_{P}} \begin{bmatrix} x_{P1} \\ x_{P2} \end{bmatrix} + \underbrace{\begin{bmatrix} B_{Pu1} \\ B_{Pu2} \end{bmatrix}}_{B_{Pu}} u$$
(2.8)

$$z_{\rm P} = x_{\rm P1} = \begin{bmatrix} I & 0 \end{bmatrix} \begin{bmatrix} x_{\rm P1} \\ x_{\rm P2} \end{bmatrix}$$
(2.9)

where Eqn. 2.8 is a SS model including the G_{TS} parameters only, x_{Pl} is the synchronised state, and x_{P2} contains the remaining state of G_{TS} , irrespective of Σ_2 . Therefore, \dot{x}_{P1} can be extracted from Eqn. 2.8 as follows

$$\dot{x}_{p_1} = A_{p_{11}} x_{p_1} + A_{p_{12}} x_{p_2} + B_{p_{u1}} u \tag{2.10}$$

Subtracting Eqns. 2.8 from \dot{x}_{N1} yields the following OB error dynamics

$$\dot{x}_{e} = \dot{x}_{N1} - \underbrace{\left(A_{P11}x_{P1} + A_{P12}x_{P2} + B_{Pu1}u\right)}_{\dot{x}_{P1}} = A_{P11}x_{e} - A_{P11}x_{N1} - A_{P12}x_{P2} + \dot{x}_{N1} + \left(-B_{Pu1}\right)u \tag{2.11}$$

To ensure successful substructuring tests, the error dynamics must be driven to zero, via the action of OB controllers. Thus, the corresponding control law and gains are proposed as

$$u = K_{eo}x_e + K_{N1}x_{N1} + K_{N1d}\dot{x}_{N1} + K_{P2}x_{P2}$$
(2.12)

$$K_{\rm N1} = B_{\rm Pu1}^{-1} (-A_{\rm P11})$$
; $K_{\rm N1d} = B_{\rm Pu1}^{-1}$; $K_{\rm P2} = B_{\rm Pu1}^{-1} (-A_{\rm P12})$ (2.13)

called output-based state-space linear substructuring controller (O-SSLSC). The feedforward gain matrices, $\{K_{N1}, K_{N1d}, K_{P2}\}$, are designed to cancel the unwanted dynamics within x_e , where the states $\{x_{N1}, \dot{x}_{N1}\}$ are measurable from the Σ_{N1} simulation, and x_{P2} can be estimated using filter designs. Feedback gain, K_{eo} , is designed for closed-loop performance, using the robust eigenstructure assignment technique (Kautsky and Nichols, 1985), for example.

Compare Eqns. 2.5 and 2.7 with Eqns. 2.12 and 2.13, although the underlying G_{TS} models are identical in the O-LSC and O-SSLSC designs, the resulting gain synthesis are not fully equivalent. State feedback of { x_{N1} , \dot{x}_{N1} , x_{P2} } to O-SSLSC generates a richer feedback environment, such that O-SSLSC would provide a better synchronisation accuracy than O-LSC.

3. THE EMULATED BEAM-SPRING-DAMPER SYSTEM

This section introduces the dynamics and modelling of the emulated beam-spring-damper (BSD) structure (Σ_E). We use the FEM simulation results in ANSYS as the benchmark, called *numerically emulated responses*, in order to compare with the DSS test results in Section 5.

Parameter	Description	Value
L	Undeformed length	2 m
Ε	Young modulus	11 GPa
b	Width	0.2 m
h	Height	0.006 m
ρ	Density	900 kg/s^3
m _b	Mass	2.2 kg
k (uncertain)	Linearised spring stiffness coefficient	675 N/m
c (uncertain)	Linearised viscous friction coefficient	0.7 Ns/m
$\{a, b\}$	Nominal actuator numerator and denominator coefficients	$\{48.6, 49.2\}$ s ⁻¹

Table 1. Notation and parameters for the BSD DSS



Figure 2. The scheme of the emulated BSD system and its substructuring (The beam is simulated by the real-time FEM in Simulink and the absorber is physically tested.)

As shown in Fig. 2, the BSD structure comprises a cantilever beam with a rectangular cross-section. The beam's both ends are rigidly fixed, and the middle is coupled with a spring-damper device, which is mounted to a solid foundation and acts as a passive vibration absorber. Here, the isolated beam is modelled using the Euler-Bernoulli theory, assuming that the shear force and normal deformation are negligible (Fagan, 1997). Therefore, the fourth-order model of the BSD system is written as (Kyrychko et al., 2007)

$$EI\frac{\partial^4 w(x,t)}{\partial x^4} + m_b \frac{\partial^2 w(x,t)}{\partial t^2} + \left(c\frac{\partial^2 w(x,t)}{\partial t^2} + kw(x,t)\right)\delta(x-L) = \underbrace{A\sin(\omega t)\delta(x-\frac{3}{4}L)}_{d}$$
(3.1)

where w(x, t) describes the lateral deflection of the beam. The deflections are resulted by a lateral pointed force, *d*, which is modelled as a sinusoid and ω is the excitation frequency. Table 1 lists the associated parameters of the emulated system.

To solve Eqn. 3.1 using the FEM, the beam is meshed with *m* elements, as shown in Fig. 2. Hermite shape functions (Rogers and McCulloch, 1994) are applied to approximate the nodal displacements of each element, yielding the local stiffness and mass matrices. These local matrices are assembled to form the global mass and stiffness matrices $\{M, K\}$ as follows

$$M\ddot{X} + KX = F \tag{3.2}$$

where $X = [x_1, x_2, ..., x_{2i-1}, x_{2i}, ..., x_{2m-1}, x_{2m}]^T$ is a vector with x_{2i-1} $(i = 1 \sim m)$ being the nodal displacement and x_{2i} being the slope of nodal rotation. In addition, the forces applied to each element are arranged in a global vector, labelled as *F*. In this example, *F* includes the sinusoid loading (*d*) and the interaction force from the absorber (*f*), which is expressed by

$$f = -kw(\frac{1}{2}L,t) - c\dot{w}(\frac{1}{2}L,t)$$
(3.3)

Accordingly, the solution vector typically yields

$$X = \left(-\omega^2 M + K\right)^{-1} F \tag{3.4}$$

Fixed boundary conditions of the first and m^{th} element are applied to the relevant entries in Eqn. 3.4, with $x_i = 0$ and $dx_i/dt = 0$ (i = 1 and m). A large stiffness value is imposed to the (m - 1)th node following the penalty approach, to ensure only the unidirectional motion (Rahman and Davies, 1984; Dussault, 1995).

In the ANSYS emulation, the beam's aspect ratio is set to 0.1 according to the Euler-Bernoulli theory assumption, and the beam is meshed by *m* segments, modelled by the BEAM3 element. The vibration absorber is simulated using the COMBIN14 element, where the linearised $\{k, c\}$ coefficients are obtained from a primary system identification test prior to the DSS experiments. Boundary conditions of zero-degree-of-freedom are assigned to the first and last nodes, and the *y*-direction displacements are permitted for the remaining nodes. As a result, the maximum displacement of each node can be extracted from the "time history variables" window and that completes the ANSYS modelling.

4. THE DEVELOPMENT OF THE OB BSD DSS

According to Fig. 2, at the substructured interface where the beam and vibration absorber are connected, the BSD system is decomposed into two subcomponents: the beam (Σ_1) and the vibration absorber (Σ_2). The beam's dynamics are relatively well-understood, thus computed by the FEM in Simulink as the numerical substructure (Σ_{N1}). A numerical excitation force (d_N) is imposed on Σ_{N1} . Critical and uncertain spring-damper vibration absorber requires physical experiments; its one end is fixed and the other mounted to an actuator with a load cell. The absorber together with the electric-mechanical actuator (G_{TS}) is denoted as Σ_{P2} . The interaction force ($f_P = y_i$) measured by the load cell is fed back from Σ_{P2} to Σ_{N1} , acting as the constraint signal. Fig. 5 in Tu and Jiang (2012) displays the Σ_{P2} test rig. In this manner, the FE substructure (Σ_{N1}) in Simulink calculates real-time nodal responses, subject to d_N and y_i . The nodal displacement of Σ_{N1} at the interface, z_N , needs to be synchronised with the G_{TS} output, z_P , by the action of the OB controllers.

4.1. The synthesis of substructured dynamics in TF and SS forms

To gain a comprehensive understanding of the DSS dynamics and to develop the FE simulation in Simulink, the Σ_{N1} model of the entire BSD DSS is derived first, leading to the synthesis of OB dynamics and controllers. With reference to Eqns. 3.2 and 3.4, the expressions for Σ_{N1} are given by

$$M\ddot{X}_{N} + KX_{N} = F_{N} + F_{P} \quad ; \quad X_{N} = \left(-\omega^{2}M + K\right)^{-1}F_{N} + \left(-\omega^{2}M + K\right)^{-1}F_{P}$$
(4.1)

where $X_N = [x_{n1}, x_{n2}, ..., x_{n(2i-1)}, x_{n(2i)}, ..., x_{n(2m-1)}, x_{n(2m)}]^T$ is the numerical solution vector, $x_{n(2i-1)}$ is the nodal displacement, and x_{n2i} is the slope of nodal rotation. In addition, F_N is the numerical force vector including d_N , and F_P is the physical force vector including the constraint signal fed back from Σ_{P2} . In terms of the Σ_{P2} expression, typically the actuator dynamics are approximated by a first-order model as

TF form:
$$z_{\rm P}(s) = G_{\rm TS}(s)u(s) = \left(\frac{b}{s+a}\right)u(s)$$
 (4.2)

SS form:
$$\dot{x}_{p_1} = -ax_{p_1} + bu$$
; $z_p = x_p$ (4.3)

where z_P is the G_{TS} output to be synchronised with z_N , and the values of $\{a, b\}$ are identified in Table 1. Accordingly, the synchronised output, $z_N = x_{n(m-1)} = x_{N1}$, at the substructured interface can be extracted from X_N . The resulting OB error dynamics in TF and SS forms are synthesised by subtracting Eqns. 4.2 and 4.3 from z_N and \dot{x}_{N1} , respectively, as follows

TF form:
$$x_e(s) = z_N(s) - z_P(s) = z_N(s) - \left(\frac{49.2}{s + 48.6}\right)u(s)$$
 (4.4)

SS form :
$$\dot{x}_e = \dot{x}_{N1} - \underbrace{\left(A_{P11}x_{P1} + B_{Pu1}u\right)}_{\dot{x}_{P1}} = \underbrace{\left(-48.6\right)}_{A_{P11}} x_e - \underbrace{\left(-48.6\right)}_{A_{P11}} x_{N1} + \dot{x}_{N1} + \underbrace{\left(-49.2\right)}_{B_u} u$$
 (4.5)

Eqns. 4.4 and 4.5 are used to design and synthesise the OB controllers in the next section.

4.2. The synthesis of OB controllers in TF and SS forms

Successful DSS tests require the $\{z_N, z_P\}$ signals at the interface to achieve exactly synchronised responses, ensuring that x_e is driven to zero. Thus, according to Eqns. 4.4 and 4.5, the control equations are proposed with reference to Eqns. 2.5 and 2.12. Considering the O-LSC design first, since G_{TS}^{-1} associated with Eqn. 4.4 represents a non-proper transfer function and is non-implementable, an inverse dynamics compensation method via real-time simulation loop (IDCS) method (Tu, 2006) is used to model G_{TS}^{-1} in a straightforward manner. The IDCS loop can be considered as a controller equivalent to $K_N(s)$, generating a noise-free feedfoward control signal, and the resulting control system is denoted as O-LSC-IDCS. On the other hand, the O-SSLSC control law is modified from Eqn. 2.12, where the K_{P2} gain is not required due to a reduced-order model of G_{TS} . Substituting Eqn. 2.12 into Eqn. 4.5, the homogeneous error equation is obtained via the following choice of gains

$$K_{\rm NI} = a / b = 0.99; \quad K_{\rm NId} = 1 / b = 0.02$$
 (4.6)

It is evident from Eqn. 4.6 that, the feedforward gain expressions are irrespective of the parameters in Σ_1 and Σ_2 , corresponding to the principle of OB substructuring strategies. The feedback gains of $K_{eo} = 0$ and $K_{eo} = 1.2$ are selected for implementation studies, in both the O-LSC-IDCS and O-SSLSC cases.

5. IMPLEMENTATION STUDIES

5.1. Implementation results

Introduction to the testing rig can be referred to Section 4.2 in Tu and Jiang (2012). Real-time FEM simulation and DSS controllers were implemented via an outer-loop dSPACE® 1104 system. Selection of the mesh number required an iterative procedure to reach a satisfactory design that provided a trade-off between conflicting objectives of numerical accuracy, sampling time (t_s), and noise sensitivity. After a number of design iterations in ANSYS, Simulink, and dSPACE, the mesh number was ultimately selected as m = 8 for a preliminary investigation, resulting in $t_s = 0.003$ s. Compare with the fast hybrid testing by Stauffer et al. (2007) using OpenSees (m = 10 and $t_s = 0.01$ s), this experimental work provided faster sampling time, and the signals were transferred in real time between the computer, dSPACE, actuator, and sensor. Sinusoid excitation with an amplitude of 40 N, a frequency of 0.25 Hz, over a test span of 30 s, was chosen as d_N . The excitation was ramped by 3 s.

The benchmark of the numerically-emulated results given by ANSYS are plotted in Fig. 3 and labelled as z_{EA} , which exhibit the maximum amplitude of z_{EA} is ~1.23 cm. In addition, the DSS testing results are also displayed in Fig. 3, where the three plots { z_{EA} , z_N , z_P } have similar maximum magnitudes, implying the consistency of simulation and testing results. Furthermore, although the controller designs are not exactly the same, nearly perfect synchronisation responses and identical trajectory patterns are observed in Figs. 3(a)-(d).



Figure 3. Experiment results controlled by O-LSC-IDCS and O-SSLSC



Figure 4. Integral square error plots

5.2. Discussion

The integral square error (ISE) curves of Figs. 3(a)-(d) are depicted in Fig. 4, for a better comparison of the synchronisation accuracy. Although Figs. 3(a)-(d) show almost identical responses, Fig. 4 highlights that (1) considering the two O-LSC-IDCS-ISE curves, the addition of the feedback gain $K_{eo} = 1.2$ yielded a two-fold ISE increment over the feedforward case, and (2) the O-SSLSC outperformed the O-LSC-IDCS in both the feedforward and feedback-controlled cases. A preliminary investigation shows that since Σ_{N1} has fast eigenvalues (Tu and Jiang, 2012), the addition of a feedback gain could have increased the noise sensitivity and reduced the synchronised performance. This substructurability study will be further addressed in future work. Furthermore, the design of O-LSC-IDCS may not be able to ideally model G_{TS}^{-1} and result in control errors, while the O-SSLSC avoids the model-inversion problem and requires more state information related to $\{x_{N1}, \dot{x}_{N1}\}$, thus providing with better synchronisation accuracy. Additionally, it is found that the force signal, f_P , fed back from the load cell includes spurious noises due to electromagnetic interference. A notch filter design for noise suppression will be considered in future work.

6. CONCLUSIONS AND FUTURE WORK

6.1 Conclusion

Innovative output-based-finite-element dynamic substructuring strategies are proposed and verified in this work, using a beam-spring-damper dynamically substructured system. Output-based control designs require only the signals and parameters associated with the actuator, irrespective of the physical specimen's parameters. From this work, it is also observed that (1) the controller designs are relatively straightforward, (2) the finite element method has been widely accepted, and (3) fast computation technologies are commercially available, and thus the proposed substructuring strategies would be theoretically and technically implementable for testing large-dimension engineering systems. Experimental studies with changed sampling time, mesh density, penalty numbers comprise the ongoing work, in order to quantify an optimal and well-conditioned numerical substructure design. A basis on that will combine with substructurability analysis and advanced numerical integration algorithms which have been developed in the substructuring literature.

6.2 Future work

On-line calculation of the global matrices and solution vectors burdens the computation load, and thus constrains the permissible mesh density and model complexity. To improve the accuracy and speed of matrix computation, advanced numerical algorithms can be considered, such as the Gaussian elimination methods. On the other hand, based on the real-time dSPACE environment, we also propose a multi-finite-element-substructure approach in discrete domain to resolve these problems. This concept has been previously discussed, e.g. De Klerk et al. (2008), but is not in the real-time substructures, as illustrated in Fig. 2. The weak connections (or the constraint signals) between finite element substructures need to be reconstructed. In this manner, the model complexity, mesh density, and computation speed can be promoted, as the complete numerical component can be simulated by a number of small-size finite element models. This will be a topic of future work, in order to bring together the substructuring literature in real-time tests and in finite element analysis.

AKCNOWLEDGEMENT

The authors gratefully acknowledge the support of the Taiwan National Science Council, grant number 100-2628-E-007-012-MY2, Development of Signal-Based Control Approach for Dynamically Substructured Systems, and the Low Carbon Energy Research Center at National Tsing Hua University, in the pursuance of this work.

REFERENCES

- Darby, A. P., Williams, M. S. and Blakeborough, A. (2002). Stability and delay compensation for real-time substructure testing. *Journal of Engineering Mechanics* **128:12**, 1276-1284.
- De Klerk, D., Rixen, D. J. and Voormeeren, S. N. (2008). General framework for dynamic substructuring: History, review, and classification of techniques. *AIAA Journal* **46**:**5**, 1169-1181.
- Dussault, J.-P. (1995). Numerical Stability and Efficiency of Penalty Algorithms. *SIAM Journal on Numerical Analysis* **32**:**1**, 296-317.
- Fagan, M. J. (1997). Finite element analysis: theory and practice, Addison Wesley Longman.
- Horiuchi, T., Inoue, M., Konno, T. and Namita, Y. (1999). Real-time hybrid experimental system with actuator delay compensation and its application to a piping system with energy absorber. *Earthquake Engineering & Structural Dynamics* 28:10, 1121-1141.
- Kautsky, J. and Nichols, N. K. (1985). Robust pole assignment in linear state feedback. *International Journal of Control* **41** 1129-1155.
- Kyrychko, Y. N., Hogan, S. J., Gonzalez-Buelga, A. and Wagg, D. J. (2007). Modelling real-time dynamic substructuring using partial delay differential equations. *Proceedings of the Royal Society A: Mathematical*, *Physical and Engineering Science* 463:2082, 1509-1523.
- Rahman, B. M. A. and Davies, J. B. (1984). Penalty Function Improvement of Waveguide Solution by Finite

Elements. Microwave Theory and Techniques, IEEE Transactions on 32:8, 922-928.

- Rogers, J. M. and McCulloch, A. D. (1994). A collocation-Galerkin finite element model of cardiac action potential propagation. *Biomedical Engineering, IEEE Transactions on* **41**:**8**, 743-757.
- Shing, P. B., Wei, Z., Jung, R.-Y. and Stauffer, E. (2004). Nees fast hybrid test system at the university of Colorado. *13th World Conference on Earthquake Engineering*. Vancouver, B.C., Canada.
- Stauffer, E., Haussmann, G., Smith, K. and Wallen, R. (2007). Fast hybrid testing at CU NEES: benchmark tests. Boulder, Colorado: Center for Fast Hybrid Testing Department of Civil Environmental and Architectural Engineering University of Colorado.
- Stoten, D. P. and Hyde, R. A. (2006). Adaptive control of dynamically substructured systems: the single-input single-output case. *Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering* **220** 63-79.
- Tu, J. Y. (2006). Control of nonlinear system using the inverse dynamics compensation via simulation (IDCS) method. MSc, University of Bristol.
- Tu, J. Y. (2012). The development of numerical-substructure-based and output-based substructuring controllers. *Structural Control and Health Monitoring*, under review.
- Tu, J. Y. and Jiang, J. Y. (2012). A numerical-substructure-based approach to the synthesis of substructurability and exact synchronisation control. *15th World Conference on Earthquake Engineering*, Lisbon, Portugal. submitted.
- Wallace, M. I., Sieber, J., Neild, S. A., Wagg, D. J. and Krauskopf, B. (2005). Stability analysis of real-time dynamic substructuring using delay differential equation models. *Earthquake Engineering & Structural Dynamics* 34:15, 1817-1832.
- Wang, T., Nakashima, M. and Pan, P. (2006). On-line hybrid test combining with general-purpose finite element software. *Earthquake Engineering & Structural Dynamics* **35**:**12**, 1471-1488.