Stability analysis of real-time dynamic substructuring using delay differential equation models

M. I. Wallace[1,*], J. Sieber[1], S. A. Neild[1], D. J. Wagg[1] and B. Krauskopf[1]

[1] *Bristol Laboratory for Advanced Dynamics Engineering, University of Bristol, Queen's Building, Bristol BS8 1TR, UK*

SUMMARY

Real-time dynamic substructuring is an experimental technique for testing the dynamic behaviour of complex structures. It involves creating a hybrid model of the entire structure by combining an experimental test piece — the substructure — with a numerical model describing the remainder of the system.

In this paper we focus on the influence of delay in the system, which is generally due to the non-instantaneous nature of the involved transfer systems (actuators). This naturally gives rise to a delay differential equation (DDE) model of the substructured system. With the case of a substructured system consisting of a single mass-spring oscillator we demonstrate how the DDE model can be used to understand the influence of the response delay of the actuator. Specifically, we describe a number of methods for identifying the critical delay time above which the system becomes unstable, which is characterized by positive exponential growth.

It is a typical situation in dynamic substructuring that the response time of the actuators exceeds the critical delay time. Therefore, additional (control) techniques need to be implemented in practice. We demonstrate with an adaptive delay compensation technique that the substructured mass-spring oscillator system can be stabilized successfully in an experiment. The approach of DDE modelling allows us to determine the dependence of the critical delay on the parameters of the delay compensation scheme. In this way it is possible to develop specific testing strategies that ensure stable operation of the substructured system. Finally, we describe an over-compensation method that is particularly suited to ensure stable testing of structures with very low damping.

KEY WORDS: Substructuring; hybrid testing; delay differential equations; delay compensation.

1. INTRODUCTION

The hybrid numerical-experimental testing technique known as *real-time dynamic substructuring* allows one to observe at full scale the behaviour of a critical element under dynamic loading. The whole system under consideration — the emulated system — is split up into an experimental test piece — the substructure — and a numerical model (or a class

*Correspondence to: M. I. Wallace, **max.wallace@bristol.ac.uk**

of models) describing the remainder of the structure. The challenge is to ensure that the substructure and the numerical model together behave in the same way as the emulated system. An overview of substructuring and how it relates to the field of experimental laboratory testing of structures can be found in Williams and Blakeborough [1]. So far the technique has been developed successfully by using expanded time scales, known as pseudo-dynamic testing [2–8], with the limitation that dynamic and hysteresis forces must be estimated. Implementing the substructuring process in real-time eliminates the need for these estimations. This is why real-time dynamics substructuring has been the subject of much recent research [9–15].

More specifically, to carry out a substructuring test the component of interest is identified as the substructure and fixed into an experimental test system. To link the substructure to the numerical model, a set of *transfer systems* (that act on the substructure) are controlled to follow the appropriate output from the numerical model [16–21]. A transfer system is typically a single (electric or hydraulic) actuator, but may also be a more complex test facility, such as a shaking table. At the same time, the forces between the transfer systems and the substructure are fed back as inputs to the numerical model. This entire process must take place in real-time. The feedback forces are treated as an external influence (or forcing) on the numerically modelled part of the system, which can then be described by a set of ordinary differential equations (ODEs). The advantage of this approach is that it is simple and fast to integrate the ODE numerically, such that real-time control can be achieved. Additionally, the dynamics may be decoupled when the substructured system has more than one transfer system [20].

The focus of this paper is on the fundamental principles behind substructuring. Our aim is to develop an understanding of the *effect of delay errors* that are always present in a substructured system. Delays arise naturally, because it is not possible for any (controlled) transfer system to react instantaneously to a change of state as prescribed by the numerical model. In fact, there are a number of different delays which combine together to give the overall delay of the transfer system, including data acquisition, computation, digital signal processing and the actuator delay itself. In some situations the transfer system delay may be so small as to be negligible, but the typical situation in substructuring is that this delay is large enough to have a significant influence on the overall dynamics of the substructured system.

We present here a technique that allows one to consider the stability of a substructured system dependant on the delay(s) and other relevant system parameters. To this end we model the substructured system with *delay differential equations* (DDEs), which are derived from the ODE model of the system by including explicitly the delay(s) due to the transfer system(s). A DDE model is a system of differential equations that depend on the current state of the system and on the state of the system some fixed time τ ago. (We restrict from now on to the case of a single fixed delay τ, but it is also possible in this framework to consider several delays.) As a general reference to the theory of DDEs see, for example, Diekmann *et al.* [22] or Stépan [23]. The advantage of DDE modelling is that we can use powerful analytical and numerical methods to determine the stability of the DDE model and, hence, of the substructured system. Specifically, the loss of stability as a function of increasing delay is typically observed in substructured systems by the onset of oscillations. Because this corresponds in the DDE model to a pair of complex conjugate eigenvalues with zero real part, it is possible to determine the critical delay time τ_c, above which the system is unstable. Depending on the system under consideration, this can be done either by considering the characteristic equation for the eigenvalues of the DDE [22, 23] or with the numerical tool DDE-BIFTOOL [24] for the stability analysis of DDEs.

In this paper we demonstrate with the case study of a mass-spring-damper system (introduced in Section 2) how DDE modelling can be used to realize the following general approach to real-time dynamic substructuring. Given a substructured system, the first step is to determine its critical delay τ_c with the stability analysis of the respective DDE model. If the delay in the transfer system is less than this critical delay the substructured system is stable. However, the typical situation is that the delay of the transfer system is larger than the critical delay — the substructured system is unstable and it develops oscillations exponentially increasing in amplitude. It is an important observation that the knowledge of τ_c and how it depends on other system parameters can be used to deal with this instability. In a second step one needs to devise a control scheme in such a way that the controlled system is stable and reproduces the dynamics of the emulated system. Whatever the choice of the controller, the controlled system is again modelled by a DDE, generally by modifying the DDE model of the substructured system, which now contains additional parameters describing the controller. The stability of this DDE model can be analysed efficiently for the dependence on the controller parameters. This knowledge can then be used to help develop suitable control strategies.

To illustrate this general approach we introduce in Section 2 the example of a substructured single mass-spring oscillator that we use throughout. We identify the origin of the delay in terms of the transfer system and show how it appears in the feedback force from the substructure. In this example, the emulated system is modelled by a linear DDE to create the substructured system. Nevertheless, this simple example already allows us to demonstrate the complex nature and effect of delay errors that are present in a substructured system. As the first step, we determine in Section 3 the critical delay τ_c of the system. Because the system is linear and quite simple, it is possible to perform an analytical as well as a numerical stability analysis. In Section 3.1 we explain how a perturbation analysis of the characteristic equation can be used (under the assumption of small delay) to compute an approximate expression of the critical time delay τ_c. For the simple mechanical system at hand it is even possible to compute explicit expressions for τ_c by computing the purely imaginary complex roots directly from the characteristic equation. We then demonstrate in Section 3.2 how the stability regions can be computed efficiently with the mathematical tool DDE-BIFTOOL. Because this software does not require any special property of the equation, the section demonstrates how one can find τ_c in a general situation of more complex and nonlinear DDEs. In Section 3.3 we demonstrate with numerical simulation what the stability boundary given by τ_c means in terms of the behaviour of the substructured system.

The overall result of Section 3 is that the substructured system will be unstable if the delay of the transfer system is indeed larger than the critical delay τ_c. Therefore, we consider and analyse in Section 4 a specific experimental implementation, where the substructure is connected to a servo-mechanical actuator and the overall system is stabilized by a delay compensation technique. Delay compensation is a well known issue for real-time substructuring; a number of single step forward prediction approaches have already been presented by Horiuchi *et al.* [9] and Darby *et al.* [12] along with other compensation techniques, such as Horiuchi and Konno [25] which have shown to improve accuracy. The technique used in this paper constitutes a more generic approach to delay compensation. It was presented by Wallace *et al.* [20] and is discussed in Section 4.1 in relation to the DDE analysis. Our experiment serves as an authenticating case study of the DDE modelling approach to real-time dynamic substructure testing that is proposed in this paper. We use further numerical stability analysis with DDE-BIFTOOL to identify the important trade-off between increased accuracy

of the numerical model and overall stability. In Section 4.2 we present the experimental results of the substructure test. Finally, in Section 4.3 we use information gained from the DDE modelling to propose a method of over-compensation that can be used to help attain a successful real-time dynamic substructure test for a dynamical system with very low damping. Concluding remarks can be found in Section 5.

2. THE SUBSTRUCTURED SYSTEM

We consider the example of a single mass-spring oscillator system, shown in Figure 1,with one excitation point for the emulated system. This well known linear system will allow us to demonstrate the fundamental problems associated with the occurrence of delay in a substructuring algorithm. The general equation of motion for the system can be written as,

$$M\ddot{z}^* + D\dot{z}^* + Kz^* = Sr(t), \tag{1}$$

where, M, D and K are the mass, damping and stiffness scalars, respectively, and $Sr(t)$ is the support excitation. The state of the system is represented by z^*, where $(.)^*$ is used to indicate that these dynamics are based on the 'complete' dynamics of the emulated system and not those of the numerical model of the substructuring algorithm. In this sense, Equation (1) is a good test case; note that for more complex or nonlinear systems it is generally not possible to calculate the emulated dynamics in this way. In this paper Equation (1) is used to assess the performance of the substructuring algorithm and highlight the effect of the delays in the system.

In order to create a substructured model of the system shown in Figure 1, the spring k_s is isolated and taken to be the substructure. The remainder of the structure, the excitation wall and the mass-spring-damper unit, is modelled numerically. This decoupling results in the substructured system shown schematically in Figure 2.

Figure 3 shows a generalised block diagram representing the substructuring algorithm for this single mass, single actuator system. An external wall excitation r is calculated/read from a file and, along with the force measurement F fed back from the substructure in the outer loop, is used to calculate the reference displacement z for the current time step in the numerical

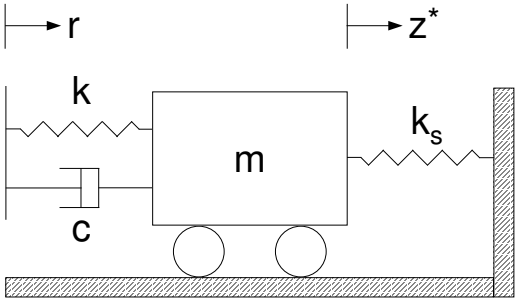

Figure 1. Schematic representation of the single mass-spring oscillator

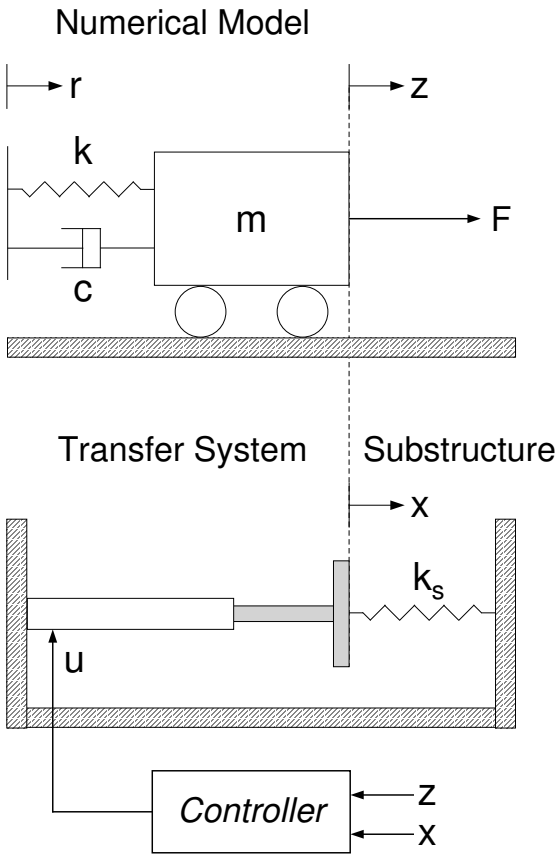

Figure 2. Schematic representation of a *substructured* system with one transfer system

model. The demand signal is fed into the transfer system, which consists of a actuator and an inner loop linear P controller. The inner loop controls the actuator state x (an experimental measurement rather than a numerical estimation) to track the desired interface displacement z. The substructure reacts according to this applied displacement and the resulting force F is measured for the following time step. This process is then iterated within hard real-time constraints (a sample rate of $1k$Hz was used for the experimental testing in Section 4).

The dynamics of the numerical model is governed by

$$m\ddot{z} + c(\dot{z} - \dot{r}) + k(z - r) = F, \tag{2}$$

where the feedback force F is the substructure response of $F = -k_s x$ (see Figure 2) and is treated as an external disturbance in the numerical model in order to simplify integration. As the transfer system has its own dynamics, it cannot react instantaneously to the change of state of the numerical model and thus introduces an inevitable time delay. This means that $x(t) = z(t - \tau)$ for some positive τ (here we ignore the additional effects of any physical

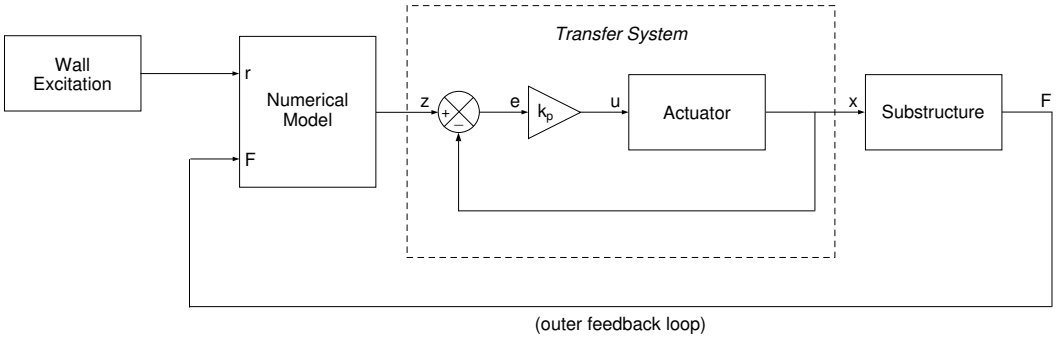

Figure 3. Block diagram for the substructuring algorithm. The inner loop is for actuator feedback control and the outer loop for the substructuring force feedback.

disturbances on the transfer system).

The delay τ introduces a systematic synchronisation error $z(t) - x(t) = z(t) - z(t-\tau)$ into the substructuring algorithm. Wallace *et al.* [20] have conjectured that, in general, the substructured system $z(t)$ approximates the emulated system $z^*(t)$ if this synchronisation error is small, that is, $z \to z^*$ if $x \to z$. Therefore, it is natural that the synchronisation error is a crucial measure for the accuracy of the substructuring experiment and, in fact, is the only explicit measurement of accuracy available for complex systems. We call the substructured system *unstable* if the synchronisation error grows exponentially in time and we call it *stable* if the synchronisation error remains bounded. As a result, when the synchronisation error is non-zero, the force is described by the delayed state z of the numerical model,

$$F = -k_s z(t-\tau). \tag{3}$$

The overall substructured system is then governed by equation (2) with (3), which constitutes a delay differential equation (DDE) that can be written as

$$m\ddot{z} + c\dot{z} + kz + k_s z(t-\tau) = c\dot{r} + kr\,. \tag{4}$$

3. STABILITY OF THE SUBSTRUCTURED SYSTEM

In this section we perform a detailed study of the substructured system Equation (4) in order to determine the critical delay τ_c above which the system is unstable. In Section 3.1 we present a complete explicit stability analysis by considering the characteristic equation of Equation (4). This is possible only for simple linear systems (like the single-mass system presented in this paper). We therefore demonstrate in Section 3.2 a numerical approach with the package DDE-BIFTOOL [24]. This numerical analysis is an essential tool when one studies complex or nonlinear substructured systems; it is used in Section 4.1 where we consider a delay compensation scheme.

3.1. Explicit Stability Analysis

From Equation (4) we obtain for zero external wall excitation r and with $x(t) = z(t-\tau)$ the complimentary equation

$$m\ddot{z} + c\dot{z} + kz + k_s x = 0. \tag{5}$$

It can be expressed with non-dimensionalized parameters in the form

$$\frac{d^2z}{d\hat{t}^2} + 2\zeta\frac{dz}{d\hat{t}} + z + px = 0, \tag{6}$$

where

$$\omega_n = \sqrt{\frac{k}{m}}, \qquad \hat{t} = \omega_n t, \qquad \hat{\tau} = \omega_n \tau, \qquad p = \frac{k_s}{k}, \qquad \zeta = \frac{c}{2\sqrt{mk}}.$$

(The values of these parameters are given in Section 4.2.)

The introduction of a delay term into a linear ordinary differential equation (ODE) has two effects. First, it changes the spectrum of the ODE by a perturbation of order τ. Second, it introduces infinitely many new modes. If the delay is small, the new modes are all strongly damped and the perturbation of the ODE spectrum can be expanded in the small parameter τ. This perturbation analysis is often easier to perform analytically than the search for all complex roots of the typically transcendental characteristic equation of the full DDE. For the parameters used in the experiments presented in Section 4.2 the assumption that τ is small is valid. Thus, we pursue both approaches and compare the perturbation analysis with the full root analysis of the characteristic equation of (6).

It is standard practice to search for solutions of the form $z = Ae^{\lambda\hat{\tau}}$. This leads to the characteristic equation for the system

$$\lambda^2 + 2\zeta\lambda + 1 + pe^{-\lambda\hat{\tau}} = 0. \tag{7}$$

The complex roots λ_i of Equation (7) are the system eigenvalues, the sign of their real parts determines the stability of the system. The majority of large structures are lightly damped thus ζ is small. We may assume that $\hat{\tau}$ is small and expand $e^{-\lambda\hat{\tau}}$ to first order as $1-\lambda\hat{\tau}$. Using this approximation, Equation (7) becomes

$$\lambda^2 + \lambda(2\zeta - p\hat{\tau}) + (1+p) = 0. \tag{8}$$

Solving for λ gives the roots

$$\lambda_{1,2} = -\frac{1}{2}(2\zeta - p\hat{\tau}) \pm \frac{1}{2}\sqrt{(2\zeta - p\hat{\tau})^2 - 4(1+p)}, \tag{9}$$

which govern the dominant eigenvalues for the DDE system given by Equation (4) when $\tau \ll 1$. When $\tau = 0$, Equation (9) reduces to the eigenvalue equation for a standard underdamped spring-mass-damper system, for which the eigenvalues are complex and stable for positive values of m, c, p. Additionally, we note that ζ, ω_n and τ are positive quantities. Thus, because $\hat{\tau}$ is small, we can make the assumption that the eigenvalues remain complex, i.e. $4(1+p) > (2\zeta - p\hat{\tau})^2$. Therefore, the real parts of the eigenvalues from Equation (9) determine the overall stability, such that the system is stable only if $p\hat{\tau} < 2\zeta$. Converted back to the original parameters this means that the system is stable if the delay τ is less than the critical value

$$\tau_c = \frac{2\zeta}{p\omega_n} = \frac{c}{k_s}. \tag{10}$$

This expression highlights that for lightly damped, stiff structures τ_c will be small and consequently the control algorithm must work harder to maintain stability. As the response delay, τ, increases past τ_c the real parts of the eigenvalues become positive and result in the transition of a pair of complex conjugate eigenvalues from the left to the right hand plane. This transition is called a Hopf bifurcation; it entails the creation of (small) oscillations. Indeed a Hopf bifurcation is a well know phenomenon in the context of DDEs [23, 26–28].

A previous analysis of the effect of time delay in substructuring [9] used an energy analysis of periodic orbits to equate the time delay to a form of negative damping. Equation (9) clearly demonstrates how this negative damping manifests itself. In fact, the equivalent negative damping term can be expressed as $c_{neg} = -k_s\tau$, with instability occurring at the point of sign change for the damping of the overall system.

The second approach to determining the stability boundaries of Equation (6) is to search for points in the parameter space where the characteristic Equation (7) has purely imaginary solutions, that is, just undergoes a Hopf bifurcation. This analysis is valid not just for small τ but for any value of τ; see Gilsinn *et al.* [27] and Larger and Goedgebuer *et al.* [28] for similar approaches. Specifically, the stability boundaries are found in the parameter space by searching for solutions of the form $z = Ae^{j\omega t} = Ae^{j\hat{\omega}\hat{t}}$ where $\hat{\omega} = \frac{\omega}{\omega_n}$ is a positive real number (0 cannot be a characteristic root in this case). We insert $\hat{\omega}$ into the characteristic equation to obtain

$$-\hat{\omega}^2 + 2\zeta\hat{\omega}j + 1 + pe^{-j\hat{\omega}\hat{\tau}} = 0. \tag{11}$$

Splitting Equation (11) into real and imaginary parts gives a system of two real equations:

$$0 = 1 - \hat{\omega}^2 + p\ \cos(\hat{\omega}\hat{\tau}) \qquad \text{for the real part,} \tag{12}$$

$$0 = 2\zeta\hat{\omega} - p\ \sin(\hat{\omega}\hat{\tau}) \qquad \text{for the imaginary part.} \tag{13}$$

We use the Equations (12) and (13) to express the parameters as functions of $\hat{\omega}$. In this way, we can identify all points in the parameter space where the DDE has purely imaginary eigenvalues and, thus, changes stability (at a Hopf bifurcation). Dividing Equation (12) by (13) we get

$$\cot(\hat{\omega}\hat{\tau}) = \frac{\hat{\omega}^2 - 1}{2\zeta\hat{\omega}}. \tag{14}$$

The cot function is periodic, therefore

$$\hat{\tau} = \frac{1}{\hat{\omega}}\ \text{arccot}\left(\frac{\hat{\omega}^2 - 1}{2\zeta\hat{\omega}}\right) + \frac{\mathrm{n}\pi}{\hat{\omega}}, \tag{15}$$

where n is an integer, is satisfied on the stability boundary. If arccot is to be taken between 0 and π then n has to be non-negative since $\hat{\tau}$ is positive. Squaring and adding the Equations (12) and (13) and rearranging for p, taking into account that p must be positive, we get

$$p = \sqrt{(\hat{\omega}^2 - 1)^2 + 4\zeta^2\hat{\omega}^2}. \tag{16}$$

Figure 4(a_1) shows the curves for $n = 0$ to $n = 7$ (up to the limit of $\hat{\tau} = 30$) of the infinite solution set for the critical parameter pairs $(\hat{\tau}, p)$ in the $(\hat{\tau}, p)$-plane with ζ fixed at 0.1066 (see Section 4.2 for parameter values). These curves are parameterized by $\hat{\omega}$ running from 0 to $+\infty$ using the Equations (15) and (16). Along these curves the system has a pair of purely imaginary eigenvalues and, hence, gains one additional unstable mode. Along the line

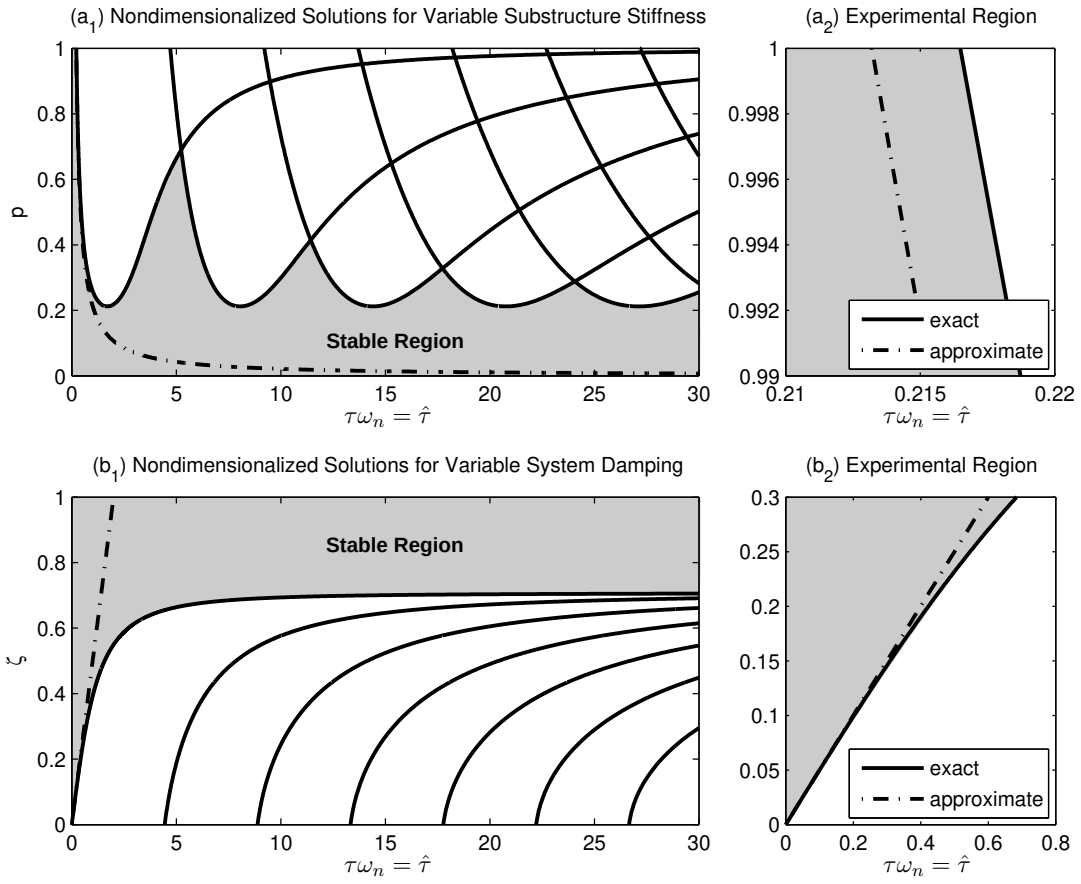

Figure 4. Nondimensionalized complex root solutions of the DDE analysis for variable response delay.

$\hat{\tau} = 0$ the system is stable. Consequently, always the lowest parts of the curves define the stability boundary; the grey area is the region of stability. For comparison we have inserted into Figure $4(a_1)$ as a dashed curve the stability boundary (10) obtained from the perturbation analysis. As stated earlier, it can be seen that the approximation only holds for small values of the delay. Figure $4(a_2)$ shows an enlargement of the region where the ratio of the spring constants has a value of $p = 1$ for which the experimental testing is performed in Section 4.2. The perturbation analysis (10) is shown to be a slight underestimation of the critical delay as the higher order terms are not included in the approximation. The nondimensionalized critical value $\hat{\tau}_c$ from the perturbation analysis (10) can be read from Figure $4(a_2)$ as $\hat{\tau}_c = 0.2132$ from which the critical time delay can be computed as $\tau_c = 6.67\,\text{ms}$. Comparing this to the value obtained for the complex root solution we see $\hat{\tau_c} = 0.2165$ which give a critical value of $\tau_c = 6.77\,\text{ms}$.

To obtain the critical delay $\hat{\tau}_c$ and ζ for fixed p as parametric curves in the $(\hat{\tau}, \zeta)$-plane we

rearrange Equation (12) for $\hat{\tau}$ and (16) for ζ, thus expressing the critical $\hat{\tau}$ and ζ as functions of $\hat{\omega}$ and p (taking into account that $\hat{\tau}$ and ζ must be positive):

$$\hat{\tau} = \frac{1}{\hat{\omega}} \arccos\left(\frac{\hat{\omega}^2 - 1}{p}\right) + \frac{2n\pi}{\hat{\omega}}, \tag{17}$$

$$\zeta = \frac{1}{2\hat{\omega}}\sqrt{p^2 - (\hat{\omega}^2 - 1)^2}, \tag{18}$$

where $\hat{\omega}$ runs from 0 to $\sqrt{1+p}$, and n is any non-negative integer if arccos takes values between 0 and π. Figure $4(b_1)$ shows the stability region (grey) and the critical values of $\hat{\tau}$ and ζ for fixed $p = 1$ using the curves defined parametrically by (17) and (18). The primary curve with $\hat{\tau}$ for $n = 0$ is always the stability boundary in the $(\hat{\tau}, \zeta)$-plane. Again, we have inserted into Figure $4(b_1)$ as a dashed curve the approximate stability boundary given by (10) that was obtained from the perturbation analysis. This curve is only accurate for systems which are lightly damped with a maximum of approximately 15% damping for this structure, which can be seen from the enlargement of the experimental region in Figure $4(b_2)$.

Figures $4(a_1)$ and $4(b_1)$ also highlight that there are 'stable' parameter regions. These are regions where the system is stable regardless of the delay $\hat{\tau}$. We can compute the boundaries of these regions by rearranging (16) for $\hat{\omega}$,

$$\hat{\omega}^2 = 1 - 2\zeta^2 \pm \sqrt{(1 - 2\zeta^2)^2 + p^2 - 1}. \tag{19}$$

The right-hand-side of Equation (19) is never real if $p < p_{\min} = 2\zeta\sqrt{1-\zeta^2} = 0.212$ (discriminant negative), giving rise to the stable region in Figure $4(a_1)$. Moreover, the right-hand-side cannot be positive if $p \leq 1$ and $\zeta > \sqrt{2}/2 = 0.7071$, which accounts for the stable region in Figure $4(b_1)$ for the specific case $k_s = k$ ($p = 1$).

We note that the vast majority of structures, especially in the civil engineering field, are lightly damped such that operating in the region of stability for all $\hat{\tau}$ would be extremely unlikely. However, when substructuring mechanical components, such as individual damper units, other parts of the stable region are likely to be accessible.

3.2. Numerical Stability Analysis

If we study more complex DDEs than Equation (6) it may become impossible to find stability regions as shown in the Figures $4(a_1)$ and $4(b_1)$ by the analytical calculations shown in Section 3.1. Even the perturbation analysis may become difficult or unrealistic. We can therefore move to a numerical approach to finding the stability regions when the substructured system is complex or nonlinear. In this section, we use a mathematical tool called DDE-BIFTOOL to demonstrate how this numerical analysis may be applied.

DDE-BIFTOOL [24] is a collection of Matlab routines for numerical bifurcation analysis of systems of DDEs with multiple fixed, discrete delays; it is freely available for scientific purposes [24]. The package can be used to compute branches of steady state solutions (equilibria) and Hopf bifurcations. Given an equilibrium, it approximates the right-most, stability determining roots of the characteristic equation. This stability information is then followed, or continued, when system parameters are changed. DDE-BIFTOOL detects when roots of the characteristic equation cross the imaginary axis, which corresponds to a change of stability of the equilibrium. In this way, a Hopf bifurcation is detected as two complex conjugate roots with zero real part.

It is then possible to continue this condition for the Hopf bifurcation point in two parameters, which gives a stability boundary in the parameter space. We remark that DDE-BIFTOOL is also able to compute branches of periodic solutions itself. Such a computation can be started from a Hopf bifurcation point or from numerical data of a periodic solution.

We used DDE-BIFTOOL to find the critical delay τ_c where the first Hopf bifurcations take place that destabilizes the substructured system. Figure 5(a) shows the real parts of the roots of the characteristic equation for the substructured system as the delay τ is increased. The system is stable when all roots are in the left half plane, that is, none of the curves are above

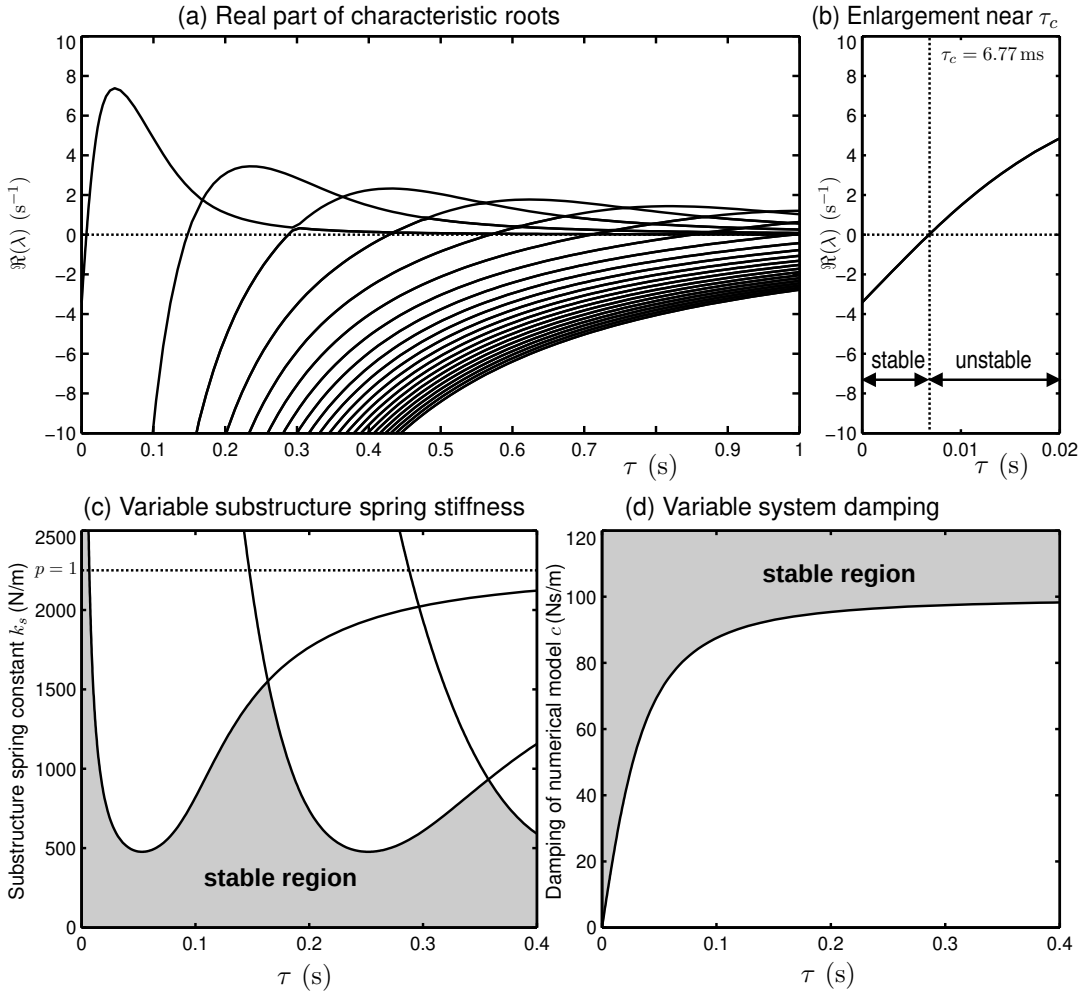

Figure 5. (a) Real eigenvalue components of the characteristic equation with enlargement of the critical region (b). Hopf bifurcation diagram showing the stability region for, (c) variable substructure spring stiffness, and (d) variable system damping.

zero. The first Hopf bifurcation takes place when the dominant branch crosses the dashed line where $\Re(\lambda) = 0$. As can be observed from the enlarged view in Figure 5(b), stability is maintained until the response delay reaches the critical value of $\tau_c = 6.77$ms. This agrees with the value found in the explicit stability analysis in Section 3.1.

The zero roots, that is the Hopf bifurcation, can be followed to see the effect of varying the structural parameters on the stability of the substructured system in terms of the critical delay. Figure 5(c) shows the stability boundary given by τ_c when the spring stiffness is varied. Indeed the boundary obtained with DDE-BIFTOOL agrees with that in Figure 4(a_1), which can be seen as validation of the continuation approach. For the response time of the actuator of the experiments presented in Section 4.2 we are dealing with a small delay, so that the first Hopf curve in Figure 5(c) is the relevant stability boundary. Note that the minimal value of the curve is observed at $k_s = 477$ such that $p = 0.212$. Furthermore, $\tau_c \to \infty$ as $p \to 1$.

Figure 5(d) shows the stability region, bounded by τ_c, for changing system damping of the numerical model. Again, we see a validation of the result found from the characteristic equation, shown in Figure 4(b_1), namely an agreement with the dominant eigenvalue branch. As the damping coefficient increases to $c = 99.5$ we see that the stability boundary (i.e. τ_c) reaches an asymptote corresponding to a system damping of $\zeta = 0.7071$.

The results in this section agree with the explicit stability analysis in Section 3.1. This clearly demonstrates that the numerical stability analysis with DDE-BIFTOOL is a straightforward and reliable tool, with the added advantage that it works also for more complex and nonlinear systems.

3.3. Simulation of the Substructured System

To demonstrate how the delay induced instability analysed in Section 3.1 manifests itself in an experiment, we now produce time domain and frequency response diagrams in simulation. The simulation allows us to idealize the dynamics of the transfer system as a pure constant time delay. This approach isolates the delay induced error caused by the instability from other parasitic effects, such as static friction, backlash and noise. This allows us to observe some of the characteristic features of the substructured system in isolation. It also gives us full control over the structural parameters and the size of the delay. In the simulations Equation (4) with the parameter values given in Section 4.2 are integrated by a fourth-order Runge-Kutta algorithm with a step size of 1ms.

Figure 6(a) shows a simulated substructure test where the response delay is $\tau = 7$ms, which is just larger than the critical delay of $\tau_c = 6.77$ms. When $\tau < \tau_c$ the numerical model error is small and bounded. However, when this critical value of $\tau_c = 6.77$ms is exceeded then the error is unbounded and grows exponentially — the overall system damping is effectively negative [9]. This situation is shown in Figure 6(b), ignoring the additional transient effects at the start of the simulation. Since the response delay is only just larger than τ_c in this instance, the growth coefficient is very small. We can observe the growth coefficient from Figure 5 by reading off the magnitude of the eigenvalue at the specific delay, in this case $\Re(\lambda) = 0.105$ at $\tau = 0.007s$.

Additionally, we observe that the frequency at which instability occurs is constant and independent of the excitation frequency. Figure 7 shows the magnitude of the numerical model response over the experimental range of frequencies for an increasing simulated response delay τ for a test duration of 5 seconds. The excitation frequency in this example is 3Hz at constant amplitude. Instability occurs at a frequency of approximately 7.1Hz and grows as expected

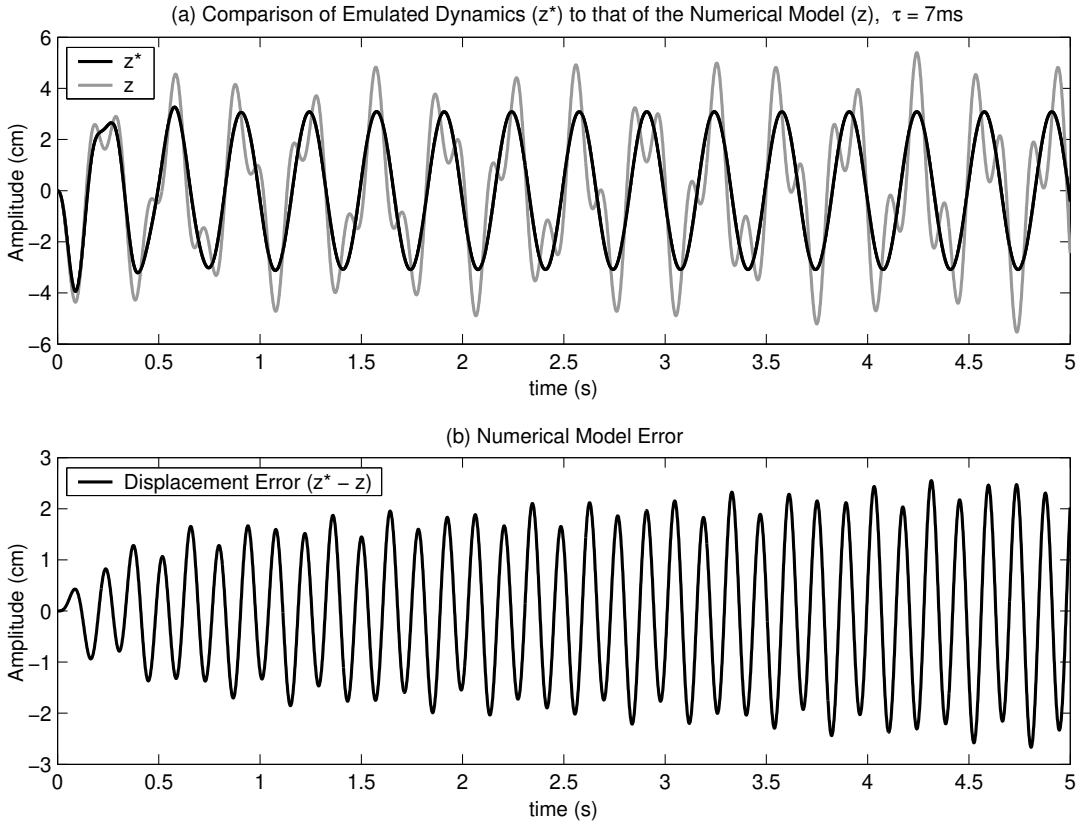

Figure 6. Simulation of numerical model accuracy caused by a transfer system delay of $\tau = 7$ms in the substructuring algorithm (where, $\tau_c = 6.77$ms).

with the exponential coefficients matching those of Figure 5 for varying τ. We can approximate the instability frequency, ω_I, from Equation (9) using the perturbation analysis at the point of instability, $p\hat{\tau} = 2\zeta$, such that

$$\lambda_i = j\sqrt{1+p} \tag{20}$$

Removing the non-dimensionalization,

$$\omega_I = \lambda_i \omega_n = \sqrt{\frac{k+k_s}{m}}, \tag{21}$$

which for this experimental setup $\omega_I = 7.198$Hz. We can find a more accurate value for this frequency from the complex root solutions at $n = 0$. From Equation (19) for the case where $p = 1$ and at the point of instability,

$$\omega_I = \omega_n\sqrt{(2-4\zeta^2)^2} = 7.116Hz. \tag{22}$$

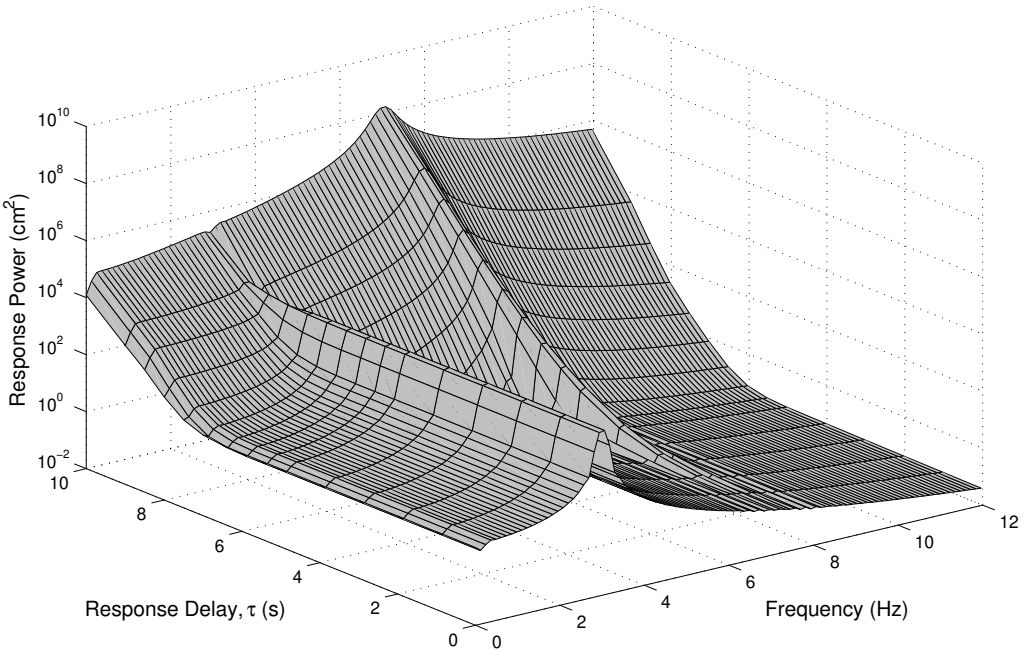

Figure 7. Exponential growth of instability (independent of the 3Hz excitation frequency).

4. EXPERIMENTAL SUBSTRUCTURE TESTING WITH DELAY COMPENSATION

As discussed in Section 2, when performing substructure tests with an experimental substructure, the response delay τ is a function of the real actuator dynamics and introduces the systematic synchronisation error $z(t) - x(t)$ into the substructuring algorithm. Typically, the delay τ is greater than the critical delay value τ_c for lightly damped systems, thus destabilizing the experiment as demonstrated in Section 3.3. Therefore, we must employ a control strategy to ensure the stability of the substructuring algorithm. A common strategy in real-time substructuring is delay compensation by extrapolation [9, 12]. In this section we show how a delay compensation scheme can be used for successful experimental stabilization of the substructured mass-spring system.

4.1. Delay compensation

The technique used in this paper is the Adaptive Forward Prediction (AFP) algorithm, presented by Wallace *et al.* [20]. The AFP algorithm is a generic approach to delay compensation as it allows non-integer multiples of the previous time step to be predicted. Furthermore, it adapts to changing plant conditions through self-tuning. Delay compensation is based on the idea of feeding a forward prediction z' of the state z of the numerical model into the transfer system. The AFP algorithm uses the prediction

$$z(t)' = (P_{N,n,\Delta}[z])(t + \rho) \tag{23}$$

where $P_{N,n,\Delta}[z]$ is the least squares fitting N^{th}-order polynomial through the n time-point pairs $(t, z(t))$, $(t-\Delta, z(t-\Delta))$,..., $(t-(n-1)\Delta, z(t-(n-1)\Delta))$. The time difference Δ is the sampling time step (1 ms if the sampling rate is 1 kHz) and ρ is the amount of forward prediction, which we do not know prior to experimentation. Thus, the quantity ρ is used to compensate for the delay τ generated by the control of the transfer system. The full AFP algorithm allows ρ to start from a set initial condition and includes an adaptive compensation for the amplitude inaccuracy and is described in detail in Wallace *et al.* [20].

A fundamental difficulty for substructuring is that it is only safe to start an experimental test from a region of stability, otherwise the unstable exponentially growth may make the test impossible if the controller cannot adapt quickly enough; see Section 4.3. Hence, it is a major concern for the performance of the AFP algorithm to find the interval of permissible ρ where the substructured system with delay compensation (23) is stable. Here, we demonstrate the stability restrictions imposed on the AFP algorithm by the DDE system of Equation (2) with (3). When we apply the delay compensation of the AFP algorithm, Equation (3) for the feed back force of the substructure changes to

$$F = -k_s(P_{N,n,\Delta}[z])(t+\rho-\tau). \tag{24}$$

System (2) with (24) is a DDE that depends on the values of z at the times $t-\tau$,..., $t-\tau-(n-1)\Delta$ in history. Thus, it contains n different delays which makes a complete analysis, as demonstrated in Section 3.1, impossible. However, DDE-BIFTOOL enables us to compute the interval of permissible ρ for the system parameters. For two pairs of algorithm parameters Figure 8 shows how the real part of the dominant eigenvalues of the DDE (2) with (24) varies with ρ. Figure 8(a) represents the stability of the AFP algorithm for a fitting polynomial $P_{N,n,\Delta}$ of order $N=2$ for $n=10$ previous values of z and Figure 8(c) corresponds to a polynomial of order $N=4$ fitted to $n=16$ previous values. Both prediction schemes are compared to the exact prediction (grey line) using

$$F = -k_s z(t+\rho-\tau) \tag{25}$$

for ρ within the interval from -20 ms to 45 ms. If the real part of the eigenvalue is positive, the DDE (2) with (24) is unstable. The vertical dashed lines highlight the parameter values $\rho=0$ (short dashes, no forward prediction, all curves coincide here) and $\rho=9.4$ ms (forward prediction equals the actual delay in the system, $\rho=\tau$). Figure 8 also shows the location of the dominant eigenvalue of system (2) with (24) for frequency vs. growth rate for the same algorithm parameter pairs, $N=2$, $n=10$ (Figure 8(b)), and $N=4$, $n=16$ (Figure 8(d)) and how it varies with ρ. The curves are traversed from top to bottom when changing ρ from -20 ms to 45 ms. The dashed circles point out the position of the dominant eigenvalues at $\rho=0$ (short dashes) and $\rho=\tau$ (long dashes). The grey curve shows how the corresponding eigenvalue of system (2) with the exact prediction (25) varies with ρ. We note that (2) with (25) has infinitely many eigenvalues close to $+\infty$, which are not approximated by the forward prediction for $\rho>\tau$.

These computations give conclusively show how the stability of the substructured system (2) with delay compensation (24) depends on ρ and the dynamical accuracy due the delay compensation. If the forward prediction could match $z(t-\tau+\rho)$ perfectly (grey line), all ρ greater than $\tau-\tau_c=2.63$ ms would result in a stable system. This observation corroborates the idea of thinking of the delay as negative damping [9]. We point out that this idea may be

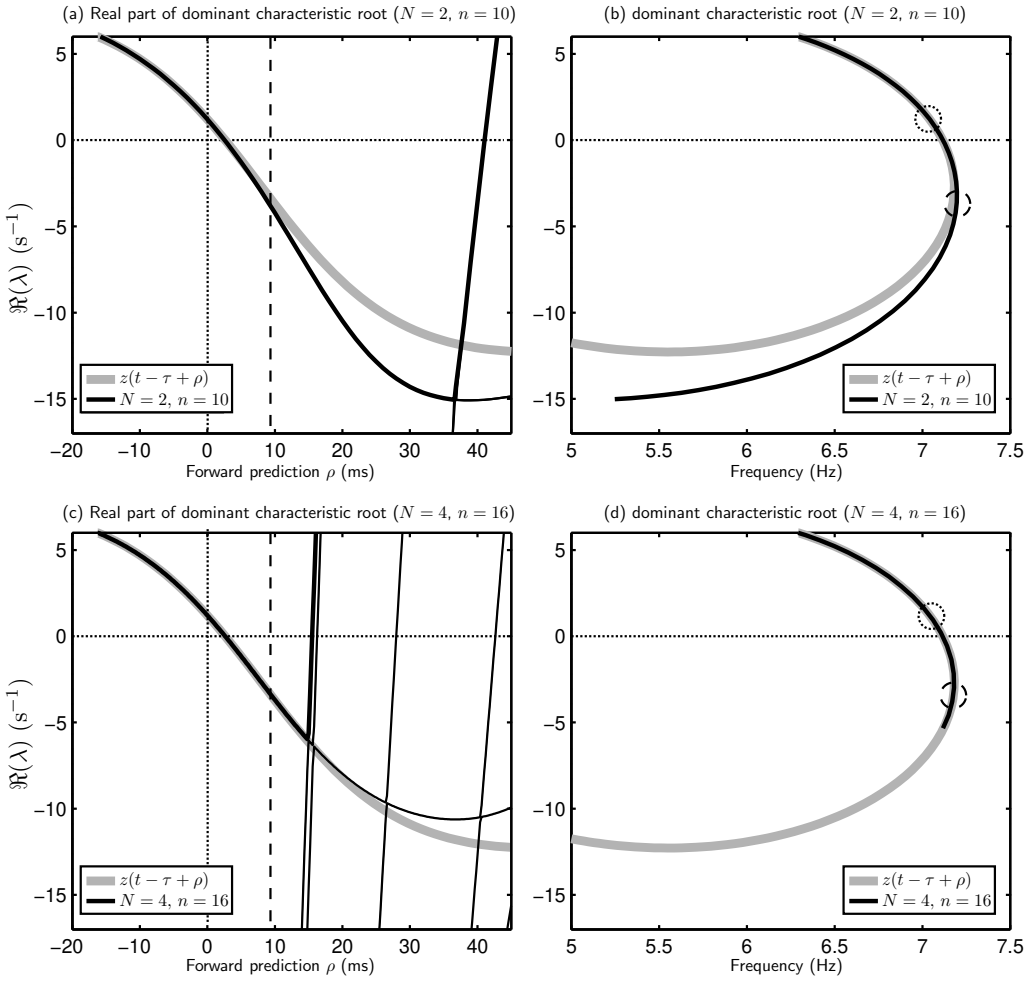

Figure 8. Eigenvalues for two delay compensation schemes with formula (23) compared to the forward prediction using the exact value $z(t - \tau + \rho)$ (grey line). Dominant eigenvalue is highlighted in bold.

misleading for a multi-mode system (as studied in [20]) because the delay can act as positive damping on the secondary modes of the system.

The polynomial forward prediction gives, in general, only a finite interval of stability for ρ. For low order schemes the interval of permissible ρ is large. Stability ranges from $\rho \approx \tau - \tau_c = 2.63\,\text{ms}$ to $\rho_{max} \approx 41\,\text{ms}$ for $N = 2$, $n = 10$ (as in Figure 8(a)). Thus, for low N the AFP algorithm can start with an initial guess for ρ that is substantially larger than the delay τ. Increasing the order N of the fitting polynomial improves the accuracy of the prediction but, in general, shrinks the range of forward prediction ρ that is permissible for stability. For example, Figure 8(c) shows that the maximal permissible ρ is at $\rho_{\max} \approx 15\,\text{ms}$ for $N = 4$, $n = 16$. Near $\rho_{\max}$ another eigenvalue of system (2) with (24) becomes dominant

and unstable. Figure 8(b) and (d) shows the dominant eigenvalue only for ρ up to $\rho_{\max}$ (thus, both curves stop early on their low end). Additionally, we note that the permissible order of N is limited by the noise fed back from the load transducer as well as $\rho_{max} > \tau - \tau_c$.

4.2. Experimental Results

To implement real-time substructuring experimentally we used a dSpace DS1104 R&D Controller Board running on hardware architecture of MPC8240 (PowerPC 603e core) at 250 MHz with 32 MB synchronous DRAM. This is fully integrated into the block diagram-based modelling tool MATLAB$^{\mathrm{TM}}$/Simulink$^{\mathrm{TM}}$which is used to build the substructured model. The dSpace companion software ControlDesk is used for online analysis, providing soft real-time access to the hard real-time application. The structural system parameters were found through system identification to be spring stiffness $k = k_s = 2250$N/m and damping ratio $c = 15$Ns/m. The constant mass of 2.2kg is connected via three parallel shafts that constrains its motion to one degree of freedom. The transfer system is a UBA (timing belt and ball screw configuration) linear Servomech actuator. Figure 9 shows the Experimental rig setup of the substructured system. It is noted that a damping of $\zeta \approx 10\%$ is relatively high for most large structures, but this reduces the sensitivity of the system to the response delay thus allowing us to highlight the characteristics of the DDE modelling approach to substructuring.

Figure 10 shows the experimental results for a wall excitation of 3Hz and constant delay compensation of $\rho = 9.4$ms with polynomial fitting of $N = 4$ and $n = 16$. It can be seen from Figure 10(a) that the numerical model dynamics z closely replicate those of the emulated system z^*, losing accuracy mainly at direction change for the actuator. Note that the transfer system dynamics are not shown on this plot but are represented by a synchronisation subspace plot [29] in Figure 10(b). Perfect synchronisation is represented by a straight diagonal line. A constant delay turns this straight line into an ellipse, as can be seen from the limit of stability

Figure 9. Experimental rig setup of substructured system.

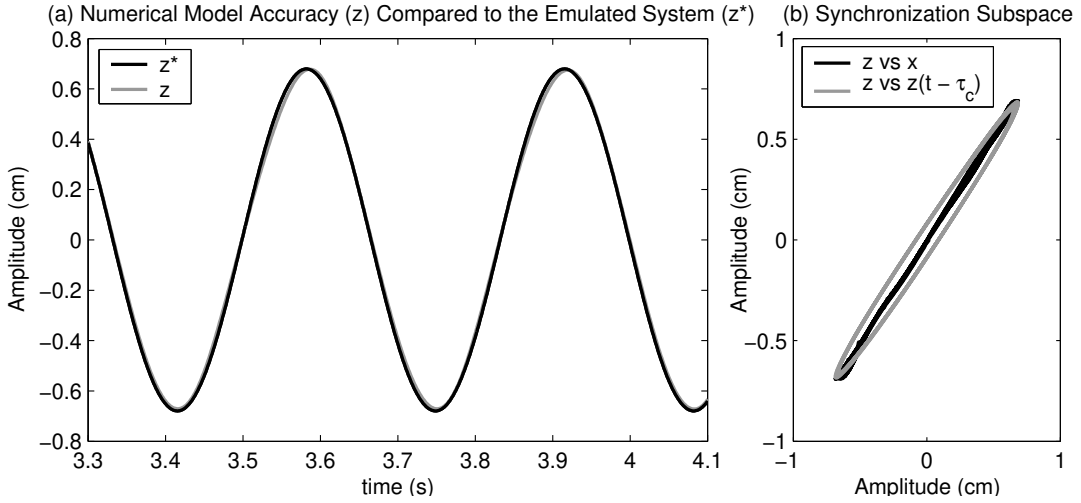

Figure 10. Experimental real-time substructure test with wall excitation of 3Hz with delay compensation of 9.4ms. Limit of stability is shown in (b) by the z vs $z(t-\tau_c)$ loop.

shown in grey representing z vs. $z(t-\tau_c)$. We can see from the subspace plot that there is generally a high level of synchronisation, well below the stable limit, apart from when the actuators change direction. Here we observe a region of loss in accuracy as the control signal must reach a certain level to overcome the static friction of the actuator mechanics before the piston will move. In fact, the algorithm verges into the unstable region at both limits. However, despite this, once the static friction is overcome, synchronisation is quickly regained. This shows that the instability shown by a substructured system (or this type of DDE) may not be not catastrophic. For instability to grow cumulatively, the synchronisation must remain in the unstable region for a longer period of time, that is a number of successive time steps. Therefore, if the control algorithm can recover more quickly than the exponential growth, given by the real part of the eigenvalue at the actual delay observed from Figure 5, then the system will be able to recover regardless of the disturbance to the synchronisation. We note that the lower the damping in the system, the smaller the critical limit of stability and, therefore, the higher $\Re(\lambda)$ will be for a given response delay. Thus, the harder the controller must work to achieve the same results for a given delay.

The transition to instability can be seen from Figure 11 for the experimental system. It is qualitatively similar to the simulated result shown in Figure 6 and occurs at a forward prediction of $\rho = 2.6$ms. The actual response delay of the transfer system is approximately $\tau = 9.4$ms for this excitation condition, giving an experimental limiting value of $\tau_c \approx 6.8$ms. The frequency at which instability is observed is shown in Figure 12; it again corroborates the results of Section 3.3 that $\omega_I \approx 7.1$Hz.

In addition to the delay error, the experimental tests were effected by approximately $5-7\%$ of noise on the feedback force signal, making it difficult to precisely determine the above values of τ_c and ω_I, unlike for the numerical simulations. This is a general issue for all experimental systems, nevertheless, the agreement with the DDE modelling remains excellent.

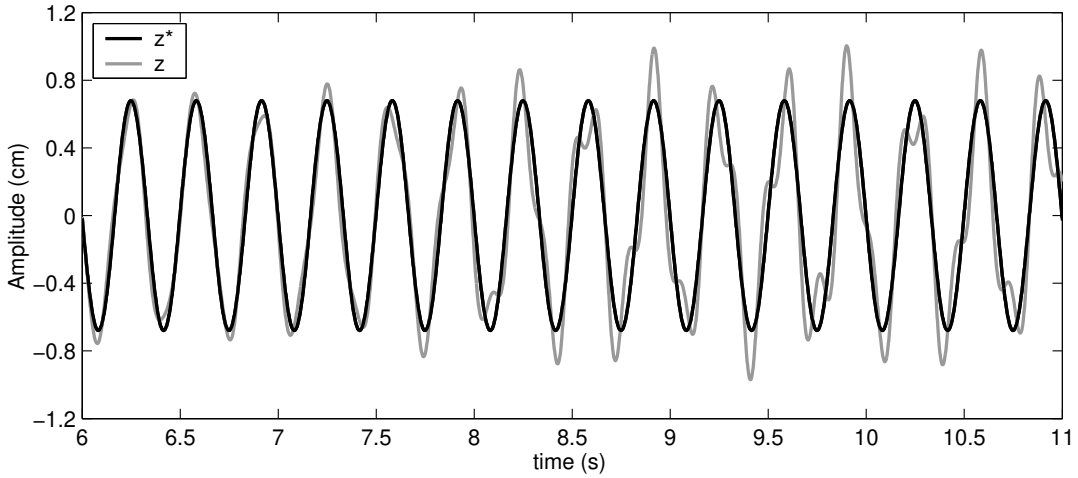

Figure 11. Transition to instability as the delay compensation is reduced on the experimental system, Forward Prediction $\rho \approx 2.6$ms.

4.3. An over compensation method

Section 4.1 discusses the permissable range of forward prediction ρ that generates a stable substructuring algorithm. We can use this information to our advantage: If we think of delay as adding negative damping in our system, then over compensating (predicting too far forward in time) will have the opposite effect of increasing the damping. If we control to a *shifted*

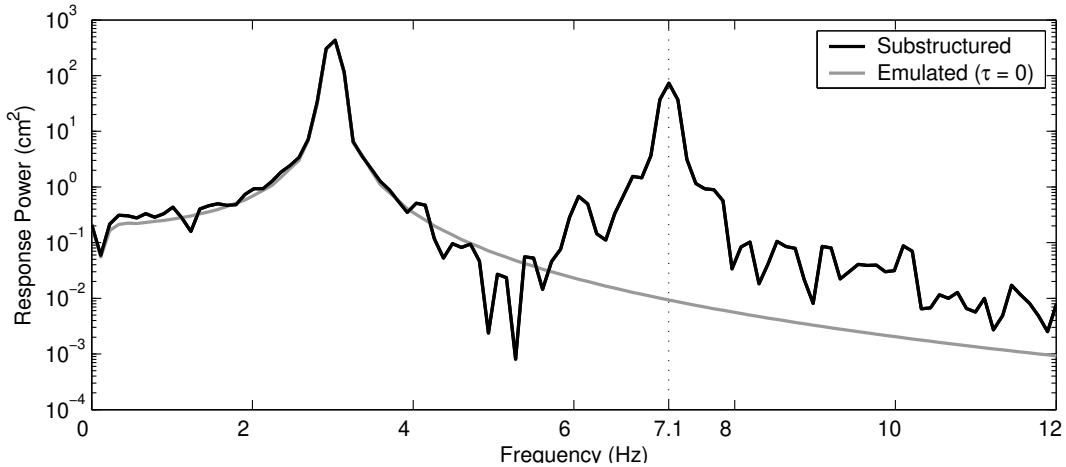

Figure 12. Frequency spectrum for the unstable experimental substructured response.

synchronisation origin, such that we now take $\tau = -0.5$ms as having zero synchronisation error, for example, this will have the effect of over-damping the dynamic response of the numerical model. Firstly, this makes the numerical model slower to react to sudden state changes, i.e. high frequency noise fed back from the substructure and, secondly, will mean that there is greater margin before the critical delay limit is reached.

However, there is a more fundamental reason why it is significant to be able to operate the substructuring algorithm in an over-compensated region as stated in Section 4.1. When starting a substructuring test when $\tau > \tau_c$, we can either initiate the test using a numerical estimation of the force (i.e. zero time delay) and switch over to the measured force when the control algorithm has achieved a high level of synchronisation, or ideally, just start from the measured force itself. As can be seen from Figure 8, if we start the test from zero compensation in this example, then the actual response delay of $\tau = 9.4$ms is greater than the critical delay of $\tau_c = 6.77$ms meaning that the test will be initiated in an unstable region. However, this may not necessarily lead to a catastrophic instability if the controller can respond faster than the unstable growth. It is preferable to start the test with the optimum level of compensation, but when we do not have a good understanding of the substructure characteristics, or of the transfer system(s) we are using, it could be very difficult to estimate this value. Therefore, if the delay τ is not known and expected to be larger than the critical delay of the substructured system, the AFP algorithm should start with a low order N to give a large range of stable forward prediction ρ and to over compensate the initial guess, as this will give the largest stable region as shown by Figure 8. Once the adaption algorithm is close to convergence, we then increase the prediction order N to improve the accuracy of the substructuring experiment. The permissible order N is limited because the maximal stable ρ shrinks below τ for increasing N. This can be seen from Figure 13, where panel (a) shows the over compensation method, (b) the zero initial conditions method and (c) the case of no delay compensation.

It is noteworthy that, although the zero initial conditions method managed to regain stability after approximately 0.6 seconds, the damping is relatively high for the example discussed in this paper ($\approx 10\%$). When this value is reduced the stability becomes increasingly more difficult to attain. For example, if the substructured system had a damping of 3%, which is more typical for large structures, then the critical limit of stability is reduced to $\tau_c = 1.3$ms. Sampling at $1k$Hz, as used in this example, a stable substructuring algorithm would be unattainable when initiating the test in this way. The over-compensation method is advantageous as the substructuring algorithm is always stable, under the maximum limit, and, therefore, has a correspondingly high numerical model accuracy throughout. A very high level of over-compensation would simply over damp the numerical model response until the optimum level had been reached adaptively. This is unlike the zero compensation method for which we see the expected exponential growth of the instability frequency. Additionally, if we remain in an over-compensated state by shifting the synchronisation origin as stated earlier, the margin to the critical stability value is extended, reducing the demands on the controller and, thus, increases the likelihood of a successful substructuring experiment, even for structures with very low damping.

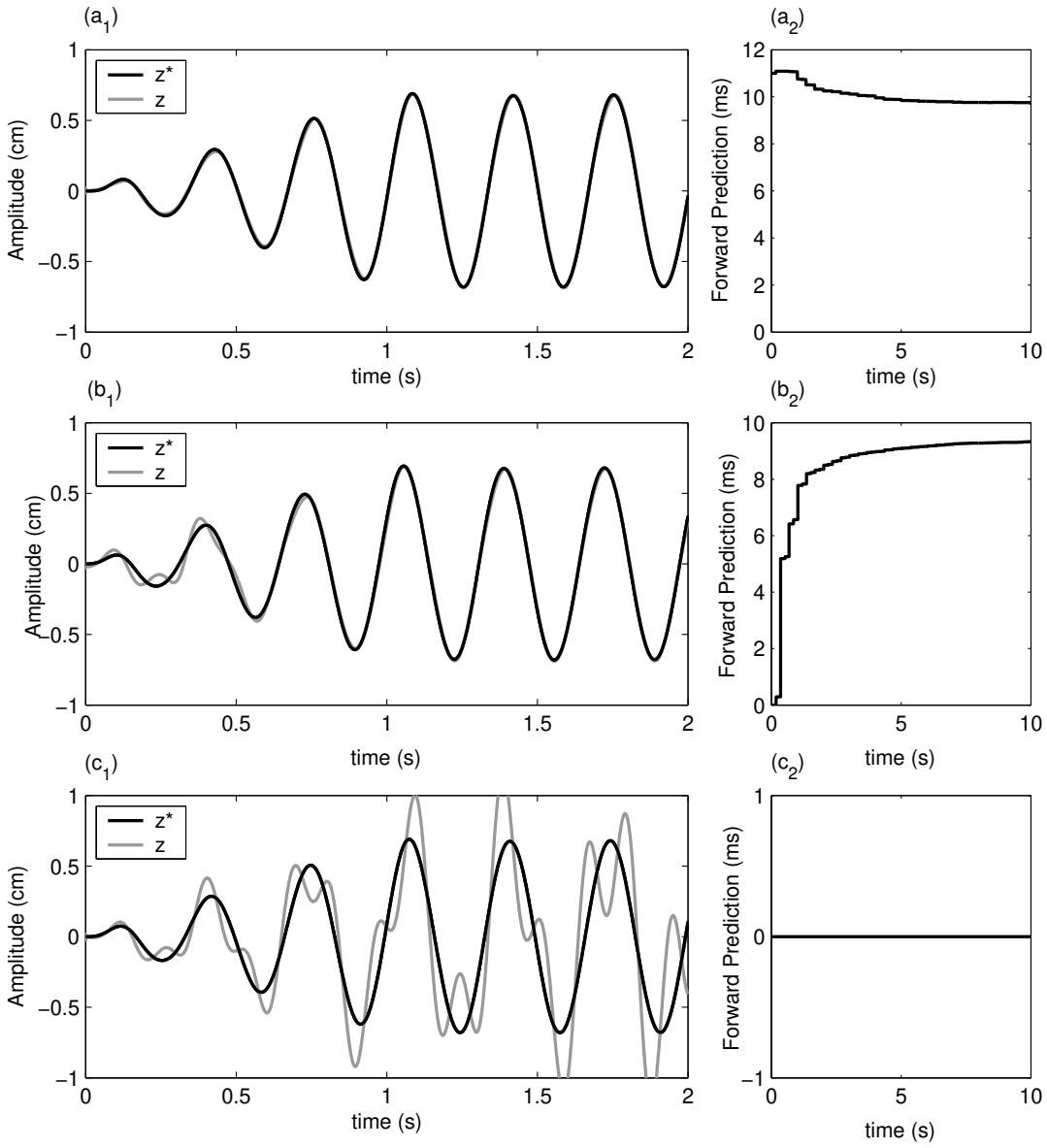

Figure 13. Experimental numerical model accuracy for differing control methodologies: (a) over compensation, (b) zero initial conditions, (c) no delay compensation. Controller adaption parameters: $\alpha = 75$, $\beta = 5$, $\gamma = 2$, $N = 4$, $n = 16$.

5. CONCLUSION

In this paper we have discussed the hybrid numerical-experimental technique of real-time dynamic substructuring. We proposed the new approach of representing the substructured

system as a delay differential equation (DDE) model. This allowed us to use well established techniques to determine the critical delay beyond which the substructured system is unstable. We used a well-known simple, linear example of a single mass-spring oscillator system to highlight the complex nature that delays play in the dynamical accuracy and stability of the substructuring algorithm. Specifically, we performed a perturbation analysis and calculated the exact complex roots of the characteristic equation of the DDE model. Furthermore, we demonstrated the simplicity of numerical stability analysis with the tool DDE-BIFTOOL.

In real-time dynamic substructuring, the delays arise through the control of the transfer systems, which is typically beyond the critical delay. Therefore, a strategy must be implemented to stabilize the substructured system, which leads to a new, more complicated DDE model. We demonstrated with the example of a delay compensation scheme how the stability of the overall DDE model can be analysed. This allowed us to evaluate and implement this delay compensation scheme, which was validated by experimental measurements. Finally, with further numerical analysis with DDE-BIFTOOL we proposed an over compensation method for substructuring that has a number of distinct benefits. Noise fed back from experimental load transducers is smoothed and we extend the margin to the critical limit of stability. Furthermore, by over-compensating at the start of the test we can initiate the test in a stable region of the substructuring algorithm. All these factors become increasingly important as the damping of the structure is reduced or the stiffness is increased.

The theoretical and experimental case study presented in this paper demonstrates the overall effectiveness of the DDE modelling approach. In future work this approach will be used in more complex substructuring scenarios. The overall goal is to realize real-time dynamic substructuring of realistic engineering components, such as cables of suspension bridges and sloshing tanks for high-rise buildings, as well as in mechanical systems, such as damper units for helicopter rotors.

ACKNOWLEDGEMENTS

The research of M.I.W. is supported by an EPSRC DTA and that of J.S. by EPSRC grant GR/R72020/01. D.J.W. and B.K. are EPSRC Advanced Research Fellows.

REFERENCES

1. M.S. Williams and A. Blakeborough. Laboratory testing of structures under dynamic loads: an introductory review. *Philosophical Transactions of the Royal Society A*, 359:1651 – 1669, 2001.
2. P.B. Shing and S.A. Mahin. Cumulative experimental errors in pseudodynamic tests. *Earthquake Engineering and Structural Dynamics*, 15:409–424, 1987.
3. M. Nakashima, H. Kato, and E. Takaoka. Development of real-time pseudo dynamic testing. *Earthquake Engineering and Structural Dynamics*, 21:779–92, 1992.
4. J. Donea, P. Magonette, P. Negro, P. Pegon, A. Pinto, and G. Verzeletti. Pseudodynamic capabilities of the elsa laboratory for earthquake testing of large structures. *Earthquake Spectra*, 12(1):163–180, 1996.
5. O.S. Bursi and P.B. Shing. Evaluation of some implicit time-stepping algorithms for pseudodynamic tests. *Earthquake Engineering and Structural Dynamics*, 25(4):333–355, 1996.
6. M. Nakashima. Development, potential, and limitations of real-time online (pseudo dynamic) testing. *Philosophical Transactions of the Royal Society A*, 359(1786):1851–1867, 2001.
7. A. Bonelli and O.S. Bursi. Generalized-a methods for seismic structural testing. *Earthquake Engineering and Structural Dynamics*, 33(10):1067–1102, 2004.

8. A.V. Pinto, P. Pegon, G. Magonette, and G. Tsionis. Pseudo-dynamic testing of bridges using non-linear substructuring. *Earthquake Engineering and Structural Dynamics*, 33:1125–1146, 2004.
9. T. Horiuchi, M. Inoue, T. Konno, and Y. Namita. Real-time hybrid experimental system with actuator delay compensation and its application to a piping system with energy absorber. *Earthquake Engineering and Structural Dynamics*, 28:1121–1141, 1999.
10. M. Nakashima and N. Masaoka. Real-time on-line test for mdof systems. *Earthquake Engineering and Structural Dynamics*, 28:393–420, 1999.
11. A. Blakeborough, M.S. Williams, A.P. Darby, and D.M. Williams. The development of real-time substructure testing. *Philosophical Transactions of the Royal Society of London A*, 359:1869–1891, 2001.
12. A.P. Darby, A. Blakeborough, and M.S. Williams. Real-time substructure tests using hydraulic actuator. *Journal of Engineering Mechanics*, 125(10):1133–1139, 2001.
13. M.I. Wallace, D.J. Wagg, and S.A. Neild. Multi-actuator substructure testing with applications to earthquake engineering: how do we assess accuracy? *Proc. of the 13th World Conf. Earthquake Engineering*, 2004.
14. P.J. Gawthrop, M.I. Wallace, and D.J. Wagg. Bond-graph based substructuring of dynamical systems. *Accepted to Earthquake Engineering and Structrural Dynamics*, 2004.
15. M.V. Sivaselvan, A. Reinhorn, Z. Liang, and X. Shao. Real-time dynamic hybrid testing of structural systems. *Proc. of the 13th World Conf. Earthquake Engineering*, 2004.
16. A.P. Darby, A. Blakeborough, and M.S. Williams. Improved control algorithm for real-time substructure testing. *Earthquake Engineering and Structural Dynamics*, 30:431–448, 2001.
17. D.J. Wagg and D.P. Stoten. Substructuring of dynamical systems via the adaptive minimal control synthesis algorithm. *Earthquake Engineering and Structural Dynamics*, 30:865–877, 2001.
18. S.A. Neild, D. Drury, D.J. Wagg, D.P. Stoten, and A.J. Crewe. Implementing real time adaptive control methods for substructuring of large structures. *Proc. of the 3rd World Conf. on Structural Control*, 2002.
19. C.M. Lim, S.A. Neild, D.P. Stoten, C.A. Taylor, and D. Drury. Using adaptive control for dynamic substructuring tests. *Proc. of the 3rd European Conf. on Structural Control*, 2004.
20. M.I. Wallace, D.J. Wagg, and S.A. Neild. An adaptive polynomial based forward prediction algorithm for multi-actuator real-time dynamic substructuring. *Submitted to Proceedings of the Royal Society A*, 2004.
21. M.I. Wallace, D.J. Wagg, and S.A. Neild. Use of control techniques for error analysis of real time dynamic substructure testing. *Proc. of the 3rd European Conf. on Structural Control*, 2004.
22. O. Diekmann, S. van Gils, S.M. Verduyn Lunel, and H.O. Walther. *Delay equations*, volume 110. 1995.
23. G. Stépan. *Retarded Dynamical Systems: Stability and Characteristic Functions*. 1989.
24. K. Engelborghs, T. Luzyanina, and D. Roose. Numerical bifurication analysis of delay differential equations using dde-biftool. *ACM Transactions on Mathematical Software*, 28(1):1–21, 2002.
25. T. Horiuchi and T. Konno. A new method for compensating actuator delay in real-time hybrid experiments. *Philosophical Transactions of the Royal Society A*, 359:1893 – 1909, 2001.
26. T. Kalmar-Nagy, G. Stepan, and F.C. Moon. Subcritical hopf bifurcation in the time delay equation model for machine tool vibration. *Nonlinear Dynamics*, 26:121–142, 2001.
27. D.E. Gilsinn. Estimating critical hopf bifurcation parameters for a second order delay differential equation with application to machine tool chatter. *Nonlinear Dynamics*, 30:103–154, 2002.
28. L. Larger and J. Goedgebuer. Subcritical hopf bifurcation in dynamical systems described by a scalar nonlinear delay differential equation. *Physical Review E*, 69(036210):1–5, 2004.
29. P. Ashwin. Non-linear dynamics, loss of synchronization and symmetry breaking. *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*, 212(3):183–187, 1998.