1. Introduction

Experimental dynamically substructured system (DSS) techniques, often referred to as real-time hybrid simulation, which combine numerical simulation and
Figure 1. DSS framework and example from the literature: (a) the substructured framework for the nominal case [6,7] and (b) the emulated MSDP system [8,9].

A physical process, are applied to investigate the dynamic behaviour of a wide range of structural systems (e.g. [1–5]). In order to help present the DSS concepts and synthesize the associated dynamics in a systematic manner, a nominal substructured framework is displayed in figure 1a [6,7], in which the original, entire engineering system to be tested is called an emulated system. The emulated system is broken down into at least two heterogeneous subsystems. The critical, new, uncertain and nonlinear component, denoted as $\Sigma_2$, is tested at full size within a physical substructure ($\Sigma_{P2}$). The remaining linear and well-identified part, $\Sigma_1$, is simulated as a numerical substructure ($\Sigma_{N1}$). Additional transfer systems ($G_{TS}$), such as a one-axis hydraulic actuator or electric motors, are installed within $\Sigma_{P2}$ in order to interface $\Sigma_{N1}$ with $\Sigma_{P2}$.

During the DSS tests, a numerical excitation signal, $d_N$, such as seismic accelerations, is imparted to $\Sigma_{N1}$ in order to fully excite the $\{\Sigma_1, \Sigma_2\}$ dynamics. Often, the displacement outputs of $\Sigma_{N1}$ and $G_{TS}$ are selected as the synchronized variables. In figure 1a, the $y_i$ signal represents the dynamic constraint, which is the reaction force between $\Sigma_2$ and $G_{TS}$; it is fed back to $\Sigma_{N1}$. Successful DSS tests require that the numerical output, $z_N$, and the $G_{TS}$ output, $z_P$, are synchronized in real time, ensuring that the substructured error, $x_e = z_N - z_P$, approaches zero. However, $G_{TS}$ inevitably includes unwanted dynamics related to transport lag (time delay), phase lag and gain modulation, which degrade the synchronization accuracy and sometimes destabilize the tests. As a result, the design of a high-quality control signal, $u$, is important in order to properly deal with the $G_{TS}$ dynamics.

Because only parts of an emulated system require prototype construction, dynamic tests using DSS techniques provide the advantages of saving preparation time, cost, experimental space and actuation energy, compared with conventional full-size testing methods. In addition, important parameters within $\Sigma_2$ can be rectified more focally and efficiently without the loss of physical realism. Sometimes, the DSS method is also analogous to an actuator-based hardware-in-the-loop simulation (HILS) technique, where actuation devices are required in order to transfer mechanical motion between software and hardware components. In this article, the authors focus on the modelling and control issues related to DSS literature.

DSS methods have been widely studied by mechanical and civil engineering communities, and research teams from the University of Bristol, UK, have contributed some important insights in regard to the progress of DSS application, dynamics analysis and control design. More specifically, their work can be divided into two tracks: dynamics- and geometry-based approaches, which are summarized in table 1. Dynamics-based methods are defined as using...
Table 1. Literature comparison of dynamically substructured system modelling and control approaches.

<table>
<thead>
<tr>
<th>types</th>
<th>mathematical expression</th>
<th>control design examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>dynamics-based</td>
<td>ODE (transfer-function and state-space)</td>
<td>linear substructuring controller [10], $H_\infty$ controller [3,11], adaptive controller [10–12], linear controller [4,6,13–15], etc.</td>
</tr>
<tr>
<td>geometry-based</td>
<td>DDE (interpolation, extrapolation, etc.)</td>
<td>AFP [8,16,17], delay compensators [1,18–22], DDE analysis [23–26], etc.</td>
</tr>
</tbody>
</table>

Transfer-function and state-space frameworks for the control design and modelling the $G_{\text{TS}}$ dynamics via ordinary differential equations (ODEs), as shown in the $G_{\text{TS}}$ block in figure 1a. Accordingly, classic control theories can provide a basis for solving the synchronization problems. Geometry-based methods approximate the $G_{\text{TS}}$ dynamics as delays based on delay differential equations (DDEs); thus, curve-fitting and advanced numerical algorithms are developed for control signal predictions. A comparison of ODE and DDE methods would be important in order to justify their relative benefits and effectiveness. Wu et al. [18] present an introduction to the two types of control methods, where the feed-forward and lead compensators can be considered as dynamics-based controllers. Their delay compensation technique [18] can be defined as a geometry-based compensator, as seen in figs 4 and 5 of [18], and is relevant to the compensation scheme of [16,17].

Delay compensation strategies are widely considered in DSS control literature. In [17–19,23,27–29], the authors collectively modelled $G_{\text{TS}}$ dynamics as fixed or time-varying delays, represented by $\tau$ and $\tau(t)$, respectively, causing the de-synchronization of $z_N$ and $z_P$ responses. Therefore, Wallace et al. [17] proposed an adaptive forward prediction (AFP) algorithm based on curve-fitting concepts, as follows: (a) adopt the least-squares polynomial functions to approximate the previous $z_N$ curve backward; (b) anticipate future output $z_N$ to be used as current control signal $u$; (c) consider an adaptive mechanism for the sake of modifying the polynomial coefficients due to significant delay variations; and (d) apply an over-compensation method [16] to parametrize the initial conditions of the controller. Section 2 will introduce the AFP strategy in more detail.

In (a), the AFP algorithm arranges a number of polynomial equations into a matrix form; inverting large-dimensional matrices is necessary in each sampled interval. Therefore, the DSS dynamics and control problems, i.e. compensating for the phase lag and gain modulation in the classic control sense, involve more concerns, such as the computational speed and numerical accuracy of the matrix inversion. Furthermore, in (d), information of $\Sigma_2$ and $G_{\text{TS}}$ models is usually required [16] in order to precisely parametrize the controller and to guarantee settling stability; this may degrade the benefit of using adaptive strategies. These two observations motivated the research into the efficacy of delay compensation and AFP techniques.

The purpose of this study is to re-examine the AFP design, which is taken as an example to make a cross-literature comparison of DDE/geometry and ODE/dynamics-based controllers, leading to a discussion on the AFP control and $G_{\text{TS}}$ modelling issues. In the remainder of this paper, the content is structured as follows. The AFP algorithm is summarized in §2, following the notation system of figure 1a; this makes it possible to reinterpret the AFP algorithm in a more systematic manner, and to offer common ground for further comparison of the DSS control strategies in §3c. In §3, new direct-compensation and singular value decomposition (SVD) methods are introduced to improve AFP performance [7]. In order to demonstrate the concepts involved, implementation studies of two DSSs are presented in §3c as examples, and the AFP controller is compared with a linear dynamics-based controller in [6]. Based on the results derived from §2 and §3, the AFP numerical problems, least-squares methods and $G_{\text{TS}}$ modelling policies related to the efficacy of DSS tests are discussed in §4. Finally, the conclusion is drawn in §5.
2. Reinterpretation and general form of the adaptive forward prediction controller

The principal AFP design in [16,17] was reviewed and divided into four parts: (a) the backward approximation using least-squares polynomials; (b) the single time-step feed-forward prediction method; (c) an adaptive algorithm; and (d) an over-compensation scheme. To begin with, in the \( i \)th step, the \( G_{TS}/\Sigma_{N1}/\text{controller output signal} \) is denoted as \( z_{P(i)}/z_{N(i)}/u_{(i)} \), where \( i \) is equated to \( h \), specifically indicating the current time step; the amount of sampled data, the sampling interval and the order of the least-squares polynomial equation are represented by \( n \), \( \Delta t \) and \( N \), respectively. The associated controller notation and parameters are referred to in figure 1a and table 2.

(a) Backward approximation using least-squares polynomials

Figure 2 shows the delay and AFP compensation schemes where the black and grey plots correspond to \( z_N \) and \( z_P \) trajectories, and the horizontal and vertical axes represent time history and \( z_N \) amplitudes, respectively. The first step of the AFP design involves approximation and curve-fitting of the previous \( z_N \) data points. Thus, the current sampled point, \( z_{N(h)} \), can be approximated by a least-squares polynomial of order \( N \) as in

\[
z_{N(h)} = a_0 + a_1 h \Delta t + a_2 [h \Delta t]^2 + \ldots + a_N [h \Delta t]^N,
\]

where \( h \Delta t \) is the current time and \( \{a_0, a_1, a_2, \ldots, a_N\} \) represent the polynomial coefficients (see eqn (3.1) in [17]). As a result, \( n \) sampled points yielding \( n \) polynomial equations are arranged into a matrix form

\[
\begin{pmatrix}
z_{N(i=0)} \\ \vdots \\ z_{N(i=-n+2)} \\ z_{N(i=-n+1)} \\ \vdots \\
z_{N(i=-1)} \\
z_{N(i=0)} \\
 \end{pmatrix}
= \begin{pmatrix}
1 & 0 & 0 & \cdots & 0 \\
\vdots & \vdots & \vdots & \ddots & \vdots \\
1 & (-n+1) \Delta t & [(-n+1) \Delta t]^2 & \cdots & [(-n+1) \Delta t]^N \\
1 & (-n) \Delta t & [(-n) \Delta t]^2 & \cdots & [(-n) \Delta t]^N \\
\end{pmatrix}
X_{(n,N)}
\begin{pmatrix}
a_0 \\
a_1 \\
\vdots \\
a_N \\
\end{pmatrix},
\]

where the \( \tilde{z}_N \) vector contains the sampled \( z_N \) amplitudes, \( \tilde{a}_{(N)} \) is the polynomial coefficient vector to be solved later and \( X_{(n,N)} \) is called a backward approximation matrix in this paper. The concept of time shifting is applied to equation (2.2) by defining \( h = 0 \), with reference to eqn (3.9) in [17], such that \( i \) equates 0 to \( -n + 1 \), and the expression of \( X_{(n,N)} \) is independent from \( h \). Accordingly, \( i = 0 \) becomes the current step, and \( i < 0 \) and \( i > 0 \) correspond to the past and future steps. In order to resolve the \( N + 1 \) unknowns within \( \tilde{a}_{(N)} \), pre-multiplying both sides of equation (2.2) by \( X_{(n,N)}^T \) gives the solution of \( \tilde{a}_{(N)} \) as follows (see eqn (3.4) in [17]):

\[
\tilde{a}_{(N)} = (X_{(n,N)}^T X_{(n,N)})^{-1} X_{(n,N)}^T \tilde{z}_N.
\]

In summary, point C in figure 2 is envisaged as the current step, \( z_{N(0)} \), and the \( n \) sampled data points of \( z_{N(i)} \) before point C are used to compute equations (2.1)–(2.3), and to determine \( \tilde{a}_{(N)} \). This section approximates the \( z_N \) response as polynomial functions and leads to the second part of the AFP design, i.e. the prediction of \( z_{N(1)} \).

(b) Single time-step feed-forward prediction

In the second part of the AFP design, the single time-step forward prediction of \( z_{N(1)} \) is considered. According to the delay time, a linear forward prediction (extrapolation) vector,
where $P$ is an integer. Thus, $\text{Ci n figure 2, equations (2.4) and (2.5) model the synchronized error, (2.4) is parametrized by which refers to eqn (3.11) in [17] a n di sa n $N$ is one of the initial conditions of the AFP algorithm to be assigned and is not necessarily an integer. Thus, $z_{N(1)}$ is a forward forecast by (see eqn (3.10) in [17])

$$z_{N(1)} = X_{Pd(N)} \equiv u(0).$$

The $z_{N(1)}$ response is sent to $G_{TS}$ as the current control signal, denoted as $u_{(0)}$. As shown at point C in figure 2, equations (2.4) and (2.5) model the synchronized error, $x_{(1)} = z_{N(1)} - z_{P(1)}$, as pure

Table 2. AFP controller and MSDP DSS parameters.

<table>
<thead>
<tr>
<th>components</th>
<th>parameters</th>
<th>notation</th>
<th>values</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\Sigma_1$</td>
<td>numerical mass</td>
<td>$m_1$</td>
<td>10 (kg)</td>
</tr>
<tr>
<td></td>
<td>damping coefficient</td>
<td>$c$</td>
<td>30 (Ns m$^{-1}$)</td>
</tr>
<tr>
<td></td>
<td>stiffness coefficient</td>
<td>$k$</td>
<td>1910 (N m$^{-1}$)</td>
</tr>
<tr>
<td>$\Sigma_2$</td>
<td>physical mass</td>
<td>$m_2$</td>
<td>1.5 (kg)</td>
</tr>
<tr>
<td></td>
<td>pendulum mass</td>
<td>$m_p$</td>
<td>1 (kg)</td>
</tr>
<tr>
<td></td>
<td>pendulum length</td>
<td>$l$</td>
<td>0.25 (m)</td>
</tr>
<tr>
<td>$G_{TS}$</td>
<td>numerator coefficient</td>
<td>$a$</td>
<td>7.6 (s$^{-1}$)</td>
</tr>
<tr>
<td></td>
<td>denominator coefficient</td>
<td>$b$</td>
<td>9.7 (s$^{-1}$)</td>
</tr>
<tr>
<td>AFP controller</td>
<td>control sampling interval</td>
<td>$\Delta t$</td>
<td>$2 \times 10^{-3}$ (s)</td>
</tr>
<tr>
<td></td>
<td>order of polynomial equation</td>
<td>$N$</td>
<td>6</td>
</tr>
<tr>
<td></td>
<td>number of sampled data</td>
<td>$n$</td>
<td>10</td>
</tr>
<tr>
<td></td>
<td>forward prediction parameters</td>
<td>$\rho$</td>
<td>32, 41, 62</td>
</tr>
<tr>
<td></td>
<td>adaptive weights</td>
<td>$\alpha, \beta, \gamma$</td>
<td>$2 \times 10^5, 5 \times 10^5, 2$</td>
</tr>
</tbody>
</table>

$X_P$, is proposed as

$$X_P = [1 \ \tau \ \tau^2 \ \cdots \ \tau^N], \quad (2.4)$$

which refers to eqn (3.11) in [17] and is an $N + 1$ by 1 vector. Here, the estimated delay in equation (2.4) is parametrized by

$$\tau = P \cdot \Delta t, \quad (2.5)$$

where $P$ is one of the initial conditions of the AFP algorithm to be assigned and is not necessarily an integer. Thus, $z_{N(1)}$ is a forward forecast by (see eqn (3.10) in [17])

$$z_{N(1)} \equiv X_{Pd(N)} \equiv u(0).$$

Figure 2. The delay and AFP compensation schemes.
amplitude differences due to a fixed-time delay. However, in practice, \( G_{TS} \) dynamics inevitably involve a time-varying delay. Thus, equations (2.4) and (2.5) are modified to

\[
XP(t) = [1 \quad \tau(t) \quad \tau(t)^2 \quad \ldots \quad \tau(t)^N] \quad (2.7)
\]

and

\[
\tau(t) = P(t) \cdot \Delta t = (P + \rho(t)) \cdot \Delta t. \quad (2.8)
\]

Equation (2.7) is called a nonlinear feed-forward prediction vector in this paper; \( \tau(t) \) and \( P(t) \) vary with time, and \( P \) is considered the initial value of \( P(t) \). The tuning of \( P(t) \) is achievable by adding the first adaptive gain, \( \rho(t) \), and the control signal of equation (2.6) is thus modified to

\[
u(0) = z_{N(1)} + (z_{N(0)} - z_P(0)) = z_{N(1)}(k_d + \sigma(t)), \quad (2.9)
\]

where \( \sigma(t) \) is the second adaptive gain for improving the amplitude accuracy; \( z_{N(0)} - z_P(0) \) indicates the current substructured error \( x_{e(0)} \) and \( k_d \) is typically set as a unity. Equations (2.8) and (2.9), incorporating the adaptive mechanism, are designed in the third step.

(c) Adaptive algorithm

With reference to [16,17], the adaptive laws of \( \sigma(t) \) and \( \rho(t) \) in equations (2.8) and (2.9) are expressed by

horizontal delay correction: \( \rho_{(j+1)} = \rho_{(j)} \pm \alpha x_{e(0)}^{\gamma} \) \quad (2.10)

and

vertical amplitude correction: \( \sigma_{(j+1)} = \sigma_{(j)} \pm \beta x_{e(0)}^{\gamma} \) \quad (2.11)

where \( j \) denotes the adaptive gain updating step, and \( \{ \alpha, \beta, \gamma \} \) are the adaptive weights to be determined prior to the tests. Only when the following conditions are met does the AFP controller update the adaptive gains: (i) the sign of \( z_{N(i)} \) is changed or \( z_{N(i)} = 0 \), as shown at points B and E in figure 2; (ii) the slope of \( z_{N(i)} \) encounters a sign change, or \( \dot{z}_{N(i)} = 0 \), such as the sinusoid peaks at points A and D in figure 2.

(d) An over-compensation method

An over-compensation method is proposed in [16] to define the value of \( P \), which importantly dominates the settling performance of DSS tests. It is noted that \( P = 0 \) implies that \( P(0) \) and \( \tau(0) \) in equations (2.7) and (2.8) are assumed to be zero, leading to an erroneous \( XP(0) \), and degrading the prediction accuracy and control stability. In order to ensure a settling performance, the numerical and analytical methods based on the DDE model are introduced to search for the lower and upper limits of \( P \). As depicted in figure 3 [16], assigning \( P \) within the permissible region produces a stable and over-compensated AFP controller.
3. Discussion and improvement of the adaptive forward prediction design

Based on the review in the preceding section, two major implementation concerns for the AFP controller are discussed in §3a: the numerical computation accuracy of equation (2.3) and the tuning of $P$ in equation (2.8). To address these two issues and thereby improve the AFP performance, SVD and direct-compensation methods are introduced in §3b [7]. The implementation results, based on two mechanical systems, are presented in §3c to verify the proposed improvement strategies.

(a) Discussion of adaptive forward prediction implementation issues

The numerical condition of matrix $X_{(n,N)}$ in equation (2.3) can be evaluated using the condition number technique, which is denoted by $\text{cond}(X_{(n,N)})$, and written as [30]

$$\text{cond}(X_{(n,N)}) = \|X_{(n,N)}^T X_{(n,N)}\|_2 \|(X_{(n,N)}^T X_{(n,N)})^{-1}\|_2 .$$

A large condition number or an ill-conditioned matrix often implies computational sensitivity to input uncertainties. Wallace et al. [16] point out that the order of $N$ is constrained by the noise that is fed back from the load cell. More specifically, a larger $N$ increases the $\text{cond}(X_{(n,N)})$ and exaggerates the propagation of computational errors while inverting $X_{(n,N)}^T X_{(n,N)}$; small perturbations (noise) within the input signals, $z_{N(i)}$ and $y_{(i)}$, thus significantly change the solution vector $\tilde{\mathbf{a}}_{(N)}$, yielding inaccurate and divergent polynomial fittings. This numerical problem is further addressed in §4.

In terms of parameter tuning, if $P$ is designated as null, i.e. $\tau = 0$, the AFP controller is uncompensated in the settling state, as shown in the $(0, 0.25)s$ interval in figure 2. In this case, erroneous $P(0)$ as well as $X_{P(0)}$, leading to divergent and unstable responses, can easily occur (see fig. 8a, b in [16]). Accordingly, attention must be paid to the tuning of $P$. However, a better and precise way to define $\{P, N, n\}$ based on the over-compensation concept [16] usually requires that the $\Sigma_1$, $\Sigma_2$ and $\tau$ parameters are known; sometimes this may not be feasible because $\Sigma_2$ may contain uncertain and unknown dynamics.

(b) Singular value decomposition and direct-compensation methods

Section 3a discussed the computation and stability issues related to $X_{(n,N)}$ and $P(t)$. The SVD method is, therefore, applied to reduce numerical sensitivity around the matrix operation, and a direct-compensation method is proposed to parametrize $P$ in a relatively optimal and straightforward manner [7] without the prior use of the $\Sigma_1$ and $\Sigma_2$ parameters.

First, the SVD method [30] is adopted for the synthesis of equation (2.3), where the SVD of $X_{(n,N)}^T X_{(n,N)}$ yields

$$X_{(n,N)}^T X_{(n,N)} = U \Sigma V^* .$$

The eigenvalues of $X_{(n,N)}^T X_{(n,N)}$ are placed on the diagonal entries of $\Sigma$; the small and negligible eigenvalues are set to zero. The directions of input and output vectors related to $X_{(n,N)}^T X_{(n,N)}$ are normalized, corresponding to $U$ and $V^*$, and the asterisk symbol denotes conjugate transposing. Substituting equation (3.2) into (2.3) gives

$$\tilde{\mathbf{a}}_{(N)} = (V \Sigma^{-1} U^*) X_{(n,N)}^T z_{N}. $$

Using equation (3.3) reduces the computational error while inverting the matrix, such that $N$ and $n$ can be prompted further, achieving a higher level prediction and synchronization accuracy.

On the other hand, a direct-compensation method can be introduced to determine $P$. Assuming that the actuator loop gain has been shaped by an inner-loop controller, which reduces the
uncertainty and reaction force effects, the $G_{TS}$ dynamics, including the nominal dynamics related to the actuator and proprietary controller, can be approximated as a first-order transfer function as follows:

$$z_P(s) = G_{TS}(s)u(s) = \frac{b}{s+a}u(s),$$

where $a$ and $b$ are the coefficients to be identified. Thus, the frequency-dependent initial phase lag, denoted by $\angle G_{TS}(j\omega_i)$ in radians, yields an equivalent and initial time delay as in

$$\tau = -\omega_i^{-1} \angle G_{TS}(j\omega_i),$$

where $\omega_i$ is the known initial excitation frequency of $d_N$. Accordingly, $P$ can be assigned with reference to equations (2.5) and (3.5) as follows:

$$P = -\omega_i \Delta t^{-1} \angle G_{TS}(j\omega_i).$$

The direct approach of equations (3.4)–(3.6) uses the dynamics-based ODE model to approximate $P$ in a practical and straightforward manner, without using the nominal models of $\{\Sigma_1, \Sigma_2\}$. This also gives a nearly optimal solution to $P$, as will be verified in the following experiment results.

(c) Implementation studies

In order to verify and generalize the proposed AFP improvement methods, mass–spring–damper (MSD) [7] and MSD–pendulum (MSDP) [8,25] DSSs are presented in this section for implementation studies. First, the MSD DSS experiments with constant sinusoids, $d_N$, are discussed in figures 4 and 5. In the MSD DSS scheme, the entire mass is separated into two parts. A part of the mass, together with the spring–damper component, is simulated in $\Sigma_N$, while the remaining mass is tested physically. Both the MSD and MSDP DSSs used the same type of electromechanical actuator as the transfer system ($G_{TS}$). As shown in figure 6b, for the $G_{TS}$ set-up, a load

Figure 4. MSD DSS implementation results: comparison of the SVD and compensation methods ($N = 6, n = 10$ and $P$ is the initial value of $P(t)$). (a) Direct compensation with no SVD ($P = 10.3$), (b) Direct compensation with SVD ($P = 10.3$), (c) under-compensation with SVD ($P = 5$) and (d) over-compensation with SVD ($P = 25$).
cell is attached to the actuator in order to measure and give feedback on the force constraint signal $y_i$. The actuator and load cell are operated in the range of ±50 mm and ±450 N, respectively. An introduction to the MSD DSS and its associated parameters is referred to in [7]. The principal comparators were selected as $\{z_N, z_P\}$ and $x_e$; the control objective was to synchronize $z_N$ and $z_P$ and minimize $x_e$, despite the unexpected dynamics within $G_{TS}$.

In the tuning process, $N$ and $n$ were promoted to a maximum of 6 and 10, respectively, in order to obtain better accuracy, while preserving the stability. In this case, $\text{cond}(X_{10,6}^T X_{10,6}) = 3.2 \times 10^{31}$ was yielded, showing an ill-conditioned matrix. Comparing the integral-square-error (ISE) analyses of figure 4a,b, using the SVD method showed a 65% improvement in synchronization accuracy. Furthermore, the direct-, under- and over-compensation testing results, corresponding to the cases of $P$ equating to 10.3, 5 and 25, were examined, as shown in figure 4b–d.

**Figure 5.** ISE comparison with changed $P$.

**Figure 6.** (a) The scheme of the MSDP DSS and (b) the MSDP $\Sigma_p$ test rig. (Online version in colour.)
Here, $P = 10.3$ was determined from an a priori system identification process based on calculations from equations (3.4)–(3.6). In addition, we artificially changed $P$ from 5 to 30 with an increment of 5, and the error responses of each test were computed via the ISE equation in figure 5, where the $x$- and $y$-axes denote the $P$-values and the final ISE results, respectively. The comparisons of figure 4b–d and figure 5 demonstrate that the direct-compensation design provided the optimal solution to $P$, as well as the best synchronization accuracy and minimal ISE value. However, it is worth noting that the AFP controller tended to produce significant error and poor stability in the settling region of $(0, 4)s$, a phenomenon that will be investigated in the following MSDP DSS tests.

In terms of the second example, the emulated MSDP system is depicted in figure 1b, and table 2 summarizes the relevant parameters. The dotted line in figure 1b indicates the substructured interface, which divides the total mass into two parts in figure 6a. Here, $\{m_1, k, c\}$ are the mass, spring and damping components within $\Sigma_{N1}$, to be simulated numerically; the remaining mass and pendulum device, $\{m_2, m_p, l\}$, are denoted as $\Sigma_2$, to be physically tested within $\Sigma_{P2}$. In figure 1b, the pendulum oscillation angle is denoted as $\theta$; non-zero $\theta(t)$ indicates that the excitation energy of $m_1 + m_2$ is transferred to drive the pendulum oscillation, such that the $z_N/z_p$ vibration is reduced. Figure 6b shows the MSDP $\Sigma_{P2}$ test rig, and an introduction to MSDP dynamics is referred to in [8,9,25]. This section will focus on presenting the experimental setting and results.

Tests were implemented via a dSPACE 1103 system with a control sampling interval of $\Delta t = 2 \times 10^{-3}$ s. Constant and swept sinusoidal waves of $d_N$ were considered in the experiments, with a fixed amplitude of 0.02 m and ramped by 1 s. In the following implementation studies, the AFP was designed based on the SVD and direct-compensation methods. The nominal parameters of $G_{TS}$ were identified as $\{a, b\} = [7.6, 9.7]$, and the analysis of equations (3.4)–(3.6) yielded $P$ equalling 41, while $\omega_i$ was $4\pi$ rad s$^{-1}$. The initial gains of $\rho(0) = \sigma(0) = 0$ were assigned, and other control parameters $\{N, n, \alpha, \beta, \gamma\}$ are summarized in table 2. Additionally, the linear numerical-substructure-based state-space controller (N-SSLSC) in [6] was used as a dynamics-based ODE design example in the experiments for the purpose of literature comparison. N-SSLSC synthesis requires prior possession of only the nominal ODE models of $\{\Sigma_{N1}, G_{TS}\}$, independent of the $\Sigma_2$ parameters; detail was obviated for the sake of brevity and can be seen in [6].

Figure 7 presents the AFP and N-SSLSC control performances in the left- and right-hand columns, respectively. The nominal and vibration-reduction cases were examined, corresponding to $\theta(0) = 0^\circ$ and $\theta(0) = 90^\circ$. In figure 7a–d, although the two controllers resulted in very similar $z_N$ and $z_p$ responses, the errors were uniformly bounded by N-SSLSC to near zero, whereas unwanted errors in the settling state were observed in the AFP-control case. Figure 7e,f extracts the $z_p$ responses from figure 7a–d in order to compare the control fidelity. The N-SSLSC-controlled DSS tests reliably emulated the vibration-reduction responses; however, the desired MSDP behaviour was slightly distorted by the AFP controller in figure 7c. In the presence of a significant phase lag in $G_{TS}$, which was artificially modelled by reducing the $G_{TS}$ built-in proportional control gains by 90% from 1 to 0.1 while the integral gain remained 0.1, figure 8 shows that N-SSLSC still sustained reasonable DSS responses, whereas unstable tests resulted in an AFP-controlled case.

Furthermore, swept frequencies from 3 Hz reducing linearly to 0.5 Hz, and from 0.5 Hz increasing to 3 Hz, were assigned to figure 9. Again, N-SSLSC still demonstrated excellent synchronization accuracy, whereas the AFP performances varied with excitation frequencies. At high frequencies, as shown in figure 9a,c, the errors were relatively small because the $\{z_N, z_p\}$ magnitudes were reduced, and the adaptive gains were updated more frequently. In the low-frequency region, particularly in figure 9c, the adaptive gains were unable to be promptly updated in reaction to the enlarged $x_o$, resulting in notable synchronization inaccuracy. In summary, the experimental studies verify that the ODE-based N-SSLSC, which explicitly considers frequency-dependent behaviour in the design, apparently outperformed the DDE-based AFP controller in terms of stability, accuracy and robustness.
Figure 7. MSDP DSS implementation results using AFP and N-SSLSC, with constant sine waves, $\omega_i = 4\pi$ rad s$^{-1}$, nominal $G_{TS}$, $P = 41$, and N-SSLSC feedback gain was 15. (a) AFP controller with $\theta(0) = 0^\circ$, (b) N-SSLSC with $\theta(0) = 0^\circ$, (c) AFP controller with $\theta(0) = 90^\circ$, (d) N-SSLSC with $\theta(0) = 90^\circ$; (e) the $z_p$ responses of (a, c) and (f) the $z_p$ responses of (b, d).

Figure 8. MSDP DSS implementation results using (a) AFP and (b) N-SSLSC, with constant sine waves, $\omega_i = 4\pi$ rad s$^{-1}$, $\theta(0) = 0^\circ$, changed $G_{TS}$, $P = 41$, and N-SSLSC feedback gain was 15.
Figure 9. MSDP DSS implementation results using AFP and N-SSLSC, with swept sine waves, $\theta(0) = 0^\circ$, nominal $G_{TS}$, and N-SSLSC feedback gain was 15. (a) AFP controller ($P = 32$, $3–0.5$Hz), (b) N-SSLSC ($3–0.5$Hz), (c) AFP controller ($P = 62$, $0.5–3$Hz) and (d) N-SSLSC ($0.5–3$Hz).

4. Further discussion on adaptive forward prediction control and modelling issues

It is noted from §3d that the AFP algorithm tended to yield unstable and inaccurate responses in the settling and transient states. Therefore, this section endeavours to address some fundamental design and implementation issues related to AFP, polynomial fitting and delay control concepts.

(a) Numerical issues within the substructured environment

Typically, numerical problems within a real-time substructured environment are attributed to the computation of $\Sigma_{N1}$ and DSS controller dynamics. Although the dynamic stability of $\Sigma_{N1}$ and the controller are ensured via their pole analysis, this does not guarantee numerical convergence in a real-time scenario. In addition, once the dynamics of $\Sigma_{N1}$ are determined, the numerical conditions of $\Sigma_{N1}$ are, for the most part, unchangeable and can only be marginally improved using advanced numerical algorithms (e.g. [31,32]).

It is noted that in the DSS literature, $\Sigma_{N1}$, including low damping and large mass, tends to cause unstable tests [14,16,33]. This phenomenon is rationalized herein, by considering the combined numerical and dynamic properties of $\Sigma_{N1}$. Non-ideal dynamics leading to an ill-conditioned plant matrix magnifies the uncertainty effects [30], causing numerical computation of $\Sigma_{N1}$ to be sensitive to input noise fed back from $y_i$; thus, significant variations in $z_N$ and $x_e$ finally destabilize the tests. The $\Sigma_{N1}$ computational sensitivity to the $y_i$ input noise has been observed from the test results of fig. 13 in [17] and fig. 20 in [26]: the control system was unchanged, whereas the substructured responses became unstable when $y_i$ was switched to be fully fed back from the load cell.

As $\Sigma_{N1}$ inevitably includes non-ideal dynamics leading to poor numerical properties, careful attention must be paid to the controller design. However, $X_{(n,N)}$ in equation (2.2) is known
as a Vandermonde matrix; its condition number increases with \( N \) and \( n \); it could be nearly singular, and its inversion is inherently difficult to compute with assured accuracy. In other words, equations (2.1)–(2.3), using the polynomial fitting techniques, essentially transform the dynamics and control design into a problem of solving numerical linear algebra related to the matrix operation. As a result, the global DSS stability is strongly affected by the numerical properties of the AFP controller, even though \( \Sigma_{N} \) contains only simple and well-conditioned dynamic equations. Many approaches (e.g. [34–36]) have been proposed to improve the numerical condition of the Vandermonde system; however, the properties of AFP-controlled error dynamics remain uncertain.

\[ (b) \text{ Error dynamics of the adaptive forward prediction controller} \]

This section further investigates the DSS error dynamics under AFP control, from the point of view of: (i) the least-squares curve-fitting theory and (ii) control tuning processes. First, the linear algebra underlying the AFP controller is discussed. Considering a single-input–single-output view of: (i) the least-squares curve-fitting theory and (ii) control tuning processes. First, the linear

\[ \text{DSS synchronization:} \quad \min_{u} x_{e(i)} = \min_{u} z_{N(i)} - z_{p(i)} \quad (4.1) \]

and

\[ \text{least-squares fitting:} \quad \min \| e(i) \| = \min \| z_N - X(u,N)(\hat{a}(N)) \|. \quad (4.2) \]

Equation (4.1) explicitly presents a closed-loop form problem which aims at finding a control signal \( u \) that minimizes \( x_{e(i)} \), whereas the least-squares method aims to minimize the residual \( e(i) \) between \( z_N \) and \( X(u,N)(\hat{a}(N)) \). The solution to equation (2.2) is further discussed in the following.

Equation (2.2) shows that \( X_{(n,N)} \) has full rank; assuming that \( z_N \) belongs to the column space of \( X_{(n,N)} \), expressed as \( z_N \in \text{col}(X_{(n,N)}) \), equation (2.3) gives exact and unique solutions only when (i) \( X_{(n,N)} \) is a square matrix, i.e. \( N = n + 1 \), or (ii) \( N < n + 1 \). Taking figure 10a as an example, the unique solution to \( \hat{a}(2) \) is obtainable when \( z_N \) happens to be the linear combination of the columns of \( X_{(2,2)} \); in this case, \( X_{(2,2)} \) is an exact interpolation matrix. Nevertheless, the two exact conditions are very rare in practice because \( z_N \) and the columns of \( X_{(n,N)} \) are associated with different physical quantities and units, i.e. metre and second; very often, equation (2.2) is an over-determined system. Accordingly, in these situations, \( z_N \notin \text{col}(X_{(n,N)}) \) and finding approximate solutions to \( \hat{a}(N) \), written as \( \hat{a}(N) \), which yield minimal residual norm, \( \| e(i) \| \), is called the least-squares solution, expressed by

\[ \| e(i) \| = \| z_N - X_{(n,N)}(\hat{a}(N)) \|. \quad (4.3) \]

As shown in figure 10b, the minimal \( \| e(i) \| \) exists when \( \| e(i) \| \) and \( \| X_{(n,N)} \| \) are orthogonal. Mathematically, multiplying \( z_N \) by a projection matrix \( B \) gives

\[ \hat{a}(N) = X_{(n,N)}(X_{(n,N)}^{T}X_{(n,N)})^{-1}X_{(n,N)}^{T}z_N. \quad (4.4) \]

Therefore, none of equations (4.2)–(4.4) guarantees that \( z_{N(1)} \) approaches \( z_{P(1)} \), because equation (2.3) only gives approximate solutions to \( \hat{a}(N) \).

In order to discuss the second control tuning issue, we envisage the least-squares polynomial fitting and prediction processes as an open-loop forward control part of the AFP algorithm, and the \( \sigma(t) \) and \( \rho(t) \) gains are treated as a closed-loop feedback control design. The technical tuning difficulties are summarized as follows:

1. A minimum of \( n \) experimental data points required for parametrizing the forward controller is unobtainable at the settling state, while the adaptive gains remain at zero and functionless; for instance, see the uncompensated interval in figure 2. Therefore,
Figure 10. The schemes of least-squares solutions: (a) exact solution: $z_N$ is the linear combination of the $X_{(2,2)}$ columns, and (b) approximate solution: the norms of $e_i$ and $X_{(2,2)}$ are orthogonal.

estimation errors in $a_{N(t)}$, $X_P(t)$ and $z_{N(t)}$ are inevitable and become significant if $G_{TS}$ has a large phase lag and $d_N$ has low frequencies.

(2) An iterative trade-off between numerical sensitivity, accuracy and stability for the selection of $[N, n, P]$ is required. Promoting $[\alpha, \beta, \gamma]$ may accelerate the error convergence; however, the growth of adaptive gains increases the closed loop gain and makes the test sensitive to noise effects.

(3) A higher $N$ provides better fitting accuracy but may result in round-off errors.

In addition, empirically, the efficacy of curve-fitting approaches depends on the data type. For example, the least-squares method is effective for approximating polynomial curves, but is insufficient to cope with random and non-smooth waves. Consequently, the AFP controller performed unsatisfactorily in figure 9a,c, in the presence of variations in the excitation signals.

In summary, the difficulties of using the static least-squares theory for dynamic estimation and using the AFP controller are: (i) the objectives of least-squares fitting and DSS synchronization differ, (ii) the least-squares forward controller only achieves approximate curve fitting, (iii) the initialization problem for polynomial fitting requires further improvement, and (iv) the adaptive gain updating is non-continuous and non-real-time. Therefore, estimation accuracy is not ensured at all times, and the error dynamics cannot be proved to be asymptotically stable under an AFP control action.

(c) Actuator modelling issues: homogeneous and heterogeneous systems

The $G_{TS}$ modelling strategies, which are divided into geometry-based DDE and dynamics-based ODE methods, are further discussed in this section. The AFP controller and many delay compensators in the literature model linear $G_{TS}$ as $e^{-\tau s}$. For example, eqn (2) in [16] expresses the overall DSS dynamics, including $\{\Sigma_1, \Sigma_2, G_{TS}\}$, as

$$m\ddot{z}_N + c(\dot{z}_N - \dot{d}_N) + k(z_N - d_N) + k_s z_N(t - \tau) = 0,$$

where $k_s$ is the spring constant in $\Sigma_2$. Equation (4.5) leads to a series of stability analyses in [16]. However, it is argued that a critical assumption underpinning equation (4.5) is: $G_{TS}$ has been made as an additional homogeneous component to $\Sigma_1$. Namely, $G_{TS}$ and $\Sigma_1$ have been made homogeneous or dynamically equivalent, having the same dynamic characteristics, such as eigenvalues and bandwidth, but differing in outputs by a delay $\tau$. This is true when $G_{TS}$ has been specifically chosen, or a DSS controller has been used and synthesized on the basis of the exact synchronization theory [37].
Essentially, $G_{TS}$ has its own dynamics, and $\tau$ is more related to frequency-dependent phase characteristics, as mentioned implicitly in some DSS literature [3,8,17,23]. Combining the delay and phase lag concepts, a more generalized model for $G_{TS}$ is suggested as

$$G_{TS}(s) = \frac{b(s)}{a(s)} e^{-\tau s}. \quad (4.6)$$

Here, $s$ denotes the Laplace variable, and $a(s)$ and $b(s)$ correspond to the denominator and numerator polynomials, respectively. The ODE model indicates phase dynamics as a result of non-ideal mechanics within $G_{TS}$, for example friction, backlashes and oil density variations. The time-shifting Laplace transform, $e^{-\tau s}$, is usually associated with the signal transport lag in signal processors, sensors and actuators (neglecting the sensor dynamics). Basically, $\tau$ should be small enough to be negligible and to reflect a real-time nature. Thus, it is expected that, when phase lag has significant variations, the DDE model, delay stability analysis and the AFP parameter tuning process, based on equation (4.5), may not work satisfactorily.

Finally, the authors emphasize again that $G_{TS}$ and $\Sigma_1$ are heterogeneous components. Using DDE or geometry-based concepts to model $G_{TS}$ and $\Sigma_1$ as a homogeneous system underestimates the influences of unwanted dynamic and numerical properties within $G_{TS}$, $\Sigma_1$ and the controller on overall DSS stability and accuracy. In addition, the control problems arising from undesired mechanical dynamics and signal processing delays should not be extensively combined. Insufficient problem definition and modelling strategies may lead to taking inefficient compensation approaches. As shown in the experimental results of §3c, the dynamics-based design, which explicitly and constructively takes the frequency-dependent magnitude and phase dynamics into account, could be more general and effective for practical DSS applications.

5. Conclusion

Successful DSS techniques for engineering testing depend on a reliable and effective synchronization controller design. The AFP technique, based on least-squares and delay compensation concepts for DSS control, was discussed in this work. Implementation results compared the synchronization performance using the AFP algorithm and a linear dynamics-based controller. Although the AFP controller has been refined by the proposed direct-compensation and SVD methods, the linear controller still outperformed AFP. The reasons for this are principally summarized as follows: (i) an ill-conditioned Vandermonde matrix leads to computational sensitivity and inaccuracy, (ii) there are inevitable errors in the least-squares polynomial fitting process, (iii) the adaptive gains are not continuously updated, and (iv) time delay or DDEs are insufficient for modelling heterogeneous systems. Detailed analysis of delay and negative damping effects (extensions to §4c) will constitute future work.

Funding statement. The authors gratefully acknowledge the support of the Taiwan National Science Council, under grant 100-2628-E-007-012-MY2 ‘Development of Signal-Based Control Approach for Dynamically Substructured Systems’, in the pursuance of this work.

References


30. Skogestad S, Postlethwaite I. 2005 *Multivariable feedback control: analysis and design*. Chichester, UK: John Wiley and Sons Ltd.


