Improvement on Adaptive Forward Prediction Controller Using A Direct-Compensation Technique

WD. Hxiao
Department of Power Mechanical Engineering
National Tsing Hua University
Hsinchu 300, Taiwan
E-mail: qoo76730@gmail.com

JY. Tu/ CY. Chen
Department of Power Mechanical Engineering
National Tsing Hua University
Hsinchu 300, Taiwan
E-mail: jytu@pme.nthu.edu.tw

***Abstract*—Hybrid testing method of dynamically substructured system is used for performance evaluation of engineering systems. During the tests, an entire engineering system is decomposed into software simulation and physical hardware substructures. Linear and well-understood parts are simulated numerically, and uncertain specimens are tested physically. The success of the tests relies on a high-quality controller to cancel the unwanted dynamics introduced by actuators within the physical substructures, thus synchronizing the numerical and physical responses at the interface. Delay compensation techniques are commonly used in dealing with the substructuring control issues. This paper proposes a new direct-compensation strategy to define the initial condition of an adaptive delay-prediction controller, without using *a priori* information about the tested components' parameters. Meanwhile, singular value decomposition method is applied to the control synthesis procedure, in order to reduce the numerical sensitivity of the controller, providing with the possibility of achieving a high-level accuracy of the tests. Experimental results are included, which favorably verify the proposed improvement strategies.**

Keywords—adaptive forward prediction*; *substructuring*; *delay compensation*; *least-squares polynomial*; *singular value decomposition

I. INTRODUCTION

Experimental techniques of dynamically substructured systems (DSS), which are hybrid testing methods combining numerical simulation and physical process, are applied to performance evaluation of a wide range of engineering systems, e.g. structural [1] and automotive [2] systems. Since only critical and new components of a system require prototype construction, DSS provides with the advantages of saving experimental costs, space, and preparation time over conventional full-scale testing, while the physical realism is still preserved. In many cases, DSS methods can also be classified as an actuator-based hardware-in-the-loop simulation technique, where supplemental actuator systems are required in order to transfer mechanical motion between numerical/ software and physical/hardware parts.

A substructured framework including typical elements and signals within a DSS is displayed in Fig. 1 [3], where DSS divides a complete, tested engineering system (emulated system, Σ_E) into one numerical (Σ_{N1}) and one physical (Σ_{P2}) substructures. Critical and uncertain components (Σ_2) are physically tested at full size. The remainder, linear and well-identified parts (Σ_1) are simulated numerically. Additional transfer systems (G_{TS}), such as electric motors or hydraulic actuators, are installed within Σ_{P2} in order to interface Σ_1 with Σ_2. The resulting interaction force between G_{TS} and Σ_2 is measured and fed back to Σ_{N1}, acting as a constraint signal and denoted as y_i.

During the testing process, a numerical excitation signal, d_N, such as earthquake forces or road disturbances, is imposed on Σ_{N1} in order to fully excite the $\{\Sigma_1, \Sigma_2\}$ dynamics. Thus, on receiving the $\{d_N, y_i\}$ signals, the displacement output of Σ_{N1} can be computed, which is labeled z_N and sent to G_{TS}. Ideally, the G_{TS}'s output, z_P, should follow z_N perfectly. In other words, z_N and z_P must be in synchronization at the substructured interface, ensuring that the substructured error, $x_e = z_N - z_P$, approaches zero. Nevertheless, in practice G_{TS} inevitably includes unwanted dynamics related to transport lag (time delay), gain modulation, and phase lag, which lead to inaccurate synchronization and sometimes even unstable tests. As a result, a successful DSS test requires a high-quality control signal, u, to cancel and compensate for the G_{TS}'s dynamics, in order to ensure the synchronization accuracy and testing stability.

In structural engineering literature, delay compensation techniques are widely applied to the DSS control system design, e.g. [4], [5], and [6], which collectively model the G_{TS}'s dynamics as *time-varying delays*, as indicated in Fig. 1. Reference [6] proposes an adaptive forward prediction (AFP) technique to cancel the delay dynamics based on a curve-fitting concept as follows: (a) applying the least-squares polynomial

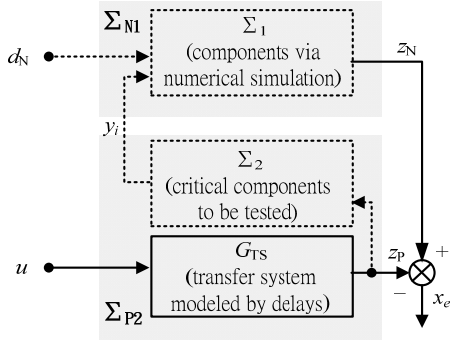


Figure 1. The substructured framework of [3].

978-1-4673-5769-2/13/$31.00 ©2013 IEEE

fitting method to interpolate backward the previous z_N data; (b) using the polynomial function to forward extrapolate/anticipate the z_N's output; (c) considering an adaptive mechanism to be triggered in predefined conditions for the sake of modifying the polynomial coefficients due to significant delay variations; (d) adopting an over-compensation technique [7] to parameterize the initial values of the polynomial coefficients for preserving stability in a settling state. Section II will introduce the AFP strategy in more detail.

It is noticed that in the fourth step, the over-compensation analysis depends on complete information of the $\{\Sigma_1, \Sigma_2, G_{TS}\}$ parameters. However, in practice Σ_2 may contain new, complicated, and uncertain dynamics. In addition, the AFP algorithm incorporating the inversion of large-dimension matrices causes the concern about ill-conditioned matrices and numerical sensitivity. These issues could render the AFP method unsuitable for engineering application in terms of practicability, stability, and accuracy. Thus, this paper proposes a direct-compensation strategy to initiate the AFP controller in a more straightforward manner than that given by [7], requiring no the Σ_2 parameters in prior. Meanwhile, singular value decomposition is added to the AFP algorithm, in order to reduce the numerical error and thus enhance the control accuracy.

The content of this paper is arranged as follows. In Section II, the AFP and over-compensation techniques are reviewed and summarized, together with the discussion about the implementation issues. A new direct-compensation and singular value decomposition method are introduced to the AFP synthesis in Section III. For demonstration of the concepts, a mass-spring-damper DSS is presented as an example in Section IV, including the associated dynamics and control synthesis, and followed by the implementation results. Finally, conclusions are drawn in Section V.

II. Introduction to Adaptive Forward Prediction Techniques

This section reviews and summarizes the work of [6] and [7], with new substructured notation and appropriate terminology applied. Adopting new notation would enable the AFP algorithm to be fitted into the substructured framework of Fig. 1 in a consistent manner for further analysis and discussion about DSS control strategies, e.g. the feasibility studies and control comparisons in [8].

A. Adaptive Forward Prediction Algorithm

The major equations and steps of developing the AFP algorithm in [6] and [7] are presented in this section; the associated notation is summarized in Table I and is introduced as follows. First, $z_{P(i)}/z_{N(i)}/u_{(i)}$ represents the G_{TS}/Σ_{N1}/controller output signal at the i^{th} step, $i = h$ specifically indicates the current time step, Δt denotes the sampling interval, n is the amount of sampled data, and N is the order of the least-squares polynomial equation. The first step of AFP involves a backward interpolation of the previous $z_{N(i)}$ data. Considering that $z_{N(h)}$ can be approximated by a polynomial equation of degree N

$$z_{N(i=h)} = a_0 + a_1 h\Delta t + a_2 \left[h\Delta t\right]^2 + \cdots + a_N \left[h\Delta t\right]^N \qquad (1)$$

accordingly, for $i = (h - n)$ to h, the n polynomial equations of previous $z_{N(i)}$ data can be arranged into a matrix form as follows

$$\underbrace{\begin{bmatrix} z_{N(h)} \\ \vdots \\ z_{N(h-n+1)} \\ z_{N(h-n)} \end{bmatrix}}_{\underline{z}_N} = \underbrace{\begin{bmatrix} 1 & h\Delta t & \left[h\Delta t\right]^2 & \cdots & \left[h\Delta t\right]^N \\ \vdots & \vdots & \vdots & \cdots & \vdots \\ 1 & (h-n+1)\Delta t & \left[(h-n+1)\Delta t\right]^2 & \ddots & \left[(h-n+1)\Delta t\right]^N \\ 1 & (h-n)\Delta t & \left[(h-n)\Delta t\right]^2 & \cdots & \left[(h-n)\Delta t\right]^N \end{bmatrix}}_{X_{(n,N)}} \underbrace{\begin{bmatrix} a_0 \\ a_1 \\ \vdots \\ a_N \end{bmatrix}}_{\underline{a}_{(N)}} \qquad (2)$$

where $\underline{a}_{(N)}$ is a vector containing the polynomial coefficients to be solved later, $X_{(n, N)}$ is called backward interpolation matrix in this paper, and the expression of $X_{(n, N)}$ depends on the values of $\{h, n, N\}$. Thus, by using the concept of time shifting and by setting $h = 0$, (2) is re-written as

$$\underbrace{\begin{bmatrix} z_{N(i=0)} \\ \vdots \\ z_{N(i=-n+1)} \\ z_{N(i=-n)} \end{bmatrix}}_{\underline{z}_N} = \underbrace{\begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \cdots & \vdots \\ 1 & (-n+1)\Delta t & \left[(-n+1)\Delta t\right]^2 & \ddots & \left[(-n+1)\Delta t\right]^N \\ 1 & (-n)\Delta t & \left[(-n)\Delta t\right]^2 & \cdots & \left[(-n)\Delta t\right]^N \end{bmatrix}}_{X_{(n,N)}} \underbrace{\begin{bmatrix} a_0 \\ a_1 \\ \vdots \\ a_N \end{bmatrix}}_{\underline{a}_{(N)}} \qquad (3)$$

which is referred to Equation (3.9) in [6]. This time-shifting approach enables the expression of $X_{(n, N)}$ to be independent from h. As a result, $i = 0$ becomes the current step, and the size and the expression of $X_{(n, N)}$ only relate to n and N. Since N is required being smaller than or equating to $n - 1$ for solving the $N + 1$ unknowns within $\underline{a}_{(N)}$, $X_{(n, N)}$ in general is a non-square matrix [9]. Thus, (3) is pre-multiplied by the transpose of $X_{(n, N)}$ below

$$X_{(n, N)}^{T} \underline{z}_N = X_{(n, N)}^{T} X_{(n, N)} \underline{a}_{(N)} \qquad (4)$$

to yield the solution of $\underline{a}_{(N)}$ (refer to Equation (3.4) in [6])

$$\underline{a}_{(N)} = \left(X_{(n, N)}^{T} X_{(n, N)}\right)^{-1} X_{(n, N)}^{T} \underline{z}_N. \qquad (5)$$

According to the expression in (1), a linear forward prediction (extrapolation) vector, X_P, is proposed as (refer to Equation (3.11) in [6])

$$X_P = \begin{bmatrix} 1 & \tau & \tau^2 & \cdots & \tau^N \end{bmatrix} \qquad (6)$$

which is a $N + 1$ by 1 vector, and τ represents a fixed and predicted delay between $z_{N(0)}$ and $z_{P(1)}$, written as

$$\tau = P \times \Delta t. \qquad (7)$$

Here, P is the ratio between τ and Δt to be defined prior to the test.

Fig. 2 displays the delay compensation scheme underlying the AFP controller, where the black and grey curves are assumed to be the z_N and z_P responses, respectively, and Point C is considered as the current time step. Once $\underline{a}_{(N)}$ and X_P in (5) and (6) are obtained, the value of $z_{N(1)}$ at the next time step can be predicted as follows

$$z_{\mathrm{N}(1)} = X_P \underline{a}_{(N)} = u_{(0)} \tag{8}$$

which is used as the current control signal. The expressions of (6)-(8) imply that the synchronized error, $x_{e(i)} = z_{\mathrm{N}(i)} - z_{\mathrm{P}(i)}$, is interpreted as pure amplitude differences due to a *fixed delay*. However, in practice, the G_{TS}'s dynamics involve frequency-dependent phase lag and gain modulation, and thus the delay time of (7) should be modified to

$$\tau(t) = \Delta t \cdot P(t) = \Delta t \cdot \big(P + \rho(t)\big) \tag{9}$$

where $\tau(t)$ and $P(t)$ are time-varying parameters, and the tuning of $P(t)$ is accomplished by adding an adaptive gain, $\rho(t)$, to P. Furthermore, in order to improve the amplitude accuracy, the current control signal is corrected according to

$$u_{(0)} = z_{\mathrm{N}(1)} + \big(z_{\mathrm{N}(0)} - z_{\mathrm{P}(0)}\big) = z_{\mathrm{N}(1)}\big(k_a + \sigma(t)\big) \tag{10}$$

where $x_{e(0)} = z_{\mathrm{N}(0)} - z_{\mathrm{P}(0)}$ represents the current substructured error, k_a is typically set to be unity, and $\sigma(t)$ is the second adaptive gain for amplitude correction.

With reference to [6] and [7], the adaptive laws of $\sigma(t)$ and $\rho(t)$ are proposed as:

$$\text{horizontal delay correction: } \rho_{(j+1)} = \rho_{(j)} \pm \alpha x_{e(0)}{}^{\gamma} \tag{11}$$

$$\text{vertical amplitude correction: } \sigma_{(j+1)} = \sigma_{(j)} \pm \beta x_{e(0)}{}^{\gamma} \tag{12}$$

where $\{\alpha, \beta, \gamma\}$ are the adaptive weights to be defined prior to the tests, and j denotes the sequence of the updated gains. By examining the sampled points, only when the following conditions are met, does the AFP controller update the adaptive gains:

- $z_{\mathrm{N}(i)} = 0$: the value of z_{N} has a sign change; as shown in Points B and E in Fig. 2.
- $\dot{z}_{\mathrm{N}(i)} = 0$: the slop of z_{N} has a sign change, e.g. the peaks of sinusoid at Points A and D in Fig. 2.

On the other hand, the tuning of $\{\alpha, \beta, \gamma\}$ requires a trade-off between the testing accuracy, stability, and sensitivity. Increasing $\{\alpha, \beta\}$ speeds up the error convergence and thus achieves higher accuracy. However, significant growth of the adaptive gains may promote the closed-loop gain and enable the DSS to become sensitive to noise effects, eventually leading to instability. In addition, [6] mentions that γ must be greater than or equal to one, and it was observed from our experimental work that excessive γ will slow down the x_e convergence speed, resulting in degraded accuracy. Thus, iterative procedure and considerable effort to find the optimal values of $\{\alpha, \beta, \gamma\}$ are required.

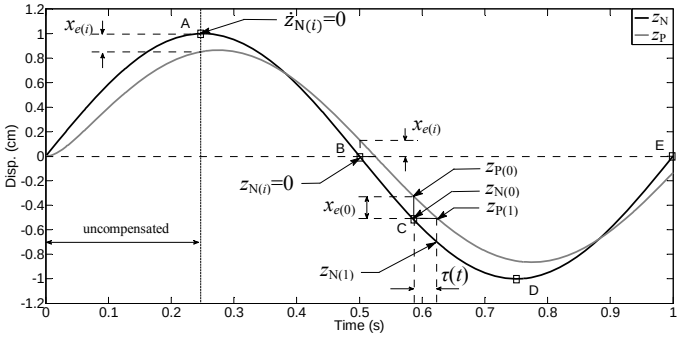


Figure 2. The delay and AFP compensation schemes.

B. An Over-Compensation Method

In addition to $\{\alpha, \beta, \gamma\}$, the value of P in (9) is also a critical parameter to be defined beforehand for ensuring settling accuracy and stability. When P is given null, the accuracy of $X_P(0)$ is not under any compensation, until either $z_{\mathrm{N}(i)} = 0$ or $\dot{z}_{\mathrm{N}(i)} = 0$ is detected. As shown in Fig. 2 in the interval (0, 0.25)s, this uncompensated duration and thus inaccurate $X_P(t)$ may lead to settling instability as well as the damage of Σ_2 in the existence of significant delay variations.

To cope with the settling performance, an over-compensation method for the stability analysis is proposed in [7], which tries to find an optimal value of P. First, an explicit stability analysis of the delay differential equation of DSS is conducted to determine the *fixed* critical delay, τ_{c}. Meanwhile, a transfer-function model of $G_{\mathrm{TS}}(s) = e^{-\tau s}$ is identified in order to obtain the value of τ. Thus, the lower limit of $P(t)$ can be obtained via the nominal parameters of $\{\tau_{\mathrm{c}}, \tau\}$ as follows

$$P(t) = (\Delta t)^{-1}(\tau - \tau_c). \tag{13}$$

Along this line of stability analysis, a DDE-BIFTOOL [10] is also used to define the upper and lower limits of $P(t)$. Therefore, the value of P can be assigned according to the permissible region of $P(t)$, that gives over-compensated and stable DSS tests. If one disregards the issues of the stability region, and overestimates or underestimates the value of P (or $\tau(t)$), x_e may grow up significantly, leading to instability in a settling state.

However, the over-compensation method for determining P requires knowing the nominal parameters of $\{\Sigma_1, \Sigma_2, G_{\mathrm{TS}}\}$ in advance, in order to draw the stability region precisely. In addition, an on-line pre-tuning process is essential in order to optimize the selection of N and n related to the stability analysis.

III. Improvement on the AFP Algorithm

From the discussion in the previous section, the issues of (i) stable over-compensation region related to P and $P(t)$, (ii) the stability analysis using the complete $\{\Sigma_1, \Sigma_2, G_{\mathrm{TS}}\}$ parameters, and (iii) optimal selection of N and n depending on real-time conditions, complicate the AFP design for practical application. For the sake of improving the AFP design, this section proposes a direct-compensation method for the tuning of P; this makes use of *a priori* identification result of phase lag of G_{TS}. Furthermore, condition number technique is also applied to analyze the numerical property of the backward interpolation matrix, $X_{P(n,N)}$. In the presence of ill-conditioned $X_{P(n,N)}$ with a large condition number, the singular value decomposition method is applied to compute the matrix inversion, such that numerical errors can be reduced, and N and n can be promoted to improve the control accuracy.

A. Development of Direct-Compensation Method

To ensure stable and accurate settling response of the AFP-controlled DSS, a direct-compensation method is introduced to parameterize the value of P. Similar to the over-compensation method, the direct strategy requires identification of the G_{TS} nominal model, whereas the G_{TS} dynamics are approximated as a first-order transfer function as follows

$$z_{\mathrm{P}}(s) = G_{\mathrm{TS}}(s)u(s) = \frac{b}{s+a}u(s) \tag{14}$$

and thus the frequency-dependent phase shift can be computed from (14), denoted by $\angle G_{\mathrm{TS}}(j\omega)$ in radians. Here, ω is known as the initial excitation frequency of d_{N}. Thus, the initial P can be approximated as

$$P = -\frac{1}{\Delta t}\omega^{-1}\angle G_{\mathrm{TS}}(j\omega). \tag{15}$$

The substitution of (15) into (9) provides a more practical, reliable, and straightforward approximation of P as well as $X_P(0)$, without using a nominal knowledge of $\{\Sigma_1, \Sigma_2\}$. The utilization of the DDE-BIFTOOL for analyzing the stability boundaries is therefore avoided.

B. Singular Value Decomposition Method

As stated in [7], the order of N is limited by the noises fed back from the load transducer. An ill-conditioned matrix of $X_{(n,N)}$ in (3) increases the numerical sensitivity to input uncertainty or reduces the accuracy of matrix inversion. Namely, small perturbations (noises) within the inputs of $\underline{z}_{\mathrm{N}}$ due to y_i can change the solution vector of $\underline{a}_{(N)}$ significantly. This unwanted property can be investigated by finding the condition number of $X_{(n,N)}$ as follows

$$\mathrm{cond}\left(X^T{}_{(10,6)}X_{(10,6)}\right) = \left\|X^T X\right\|\left\|(X^T X)^{-1}\right\| = 3.1853\times10^{31} \tag{16}$$

which takes $n = 10$ and $N = 6$ as an example. The condition number increases with the order of N and the number of n. A large condition number may imply accumulation and propagation of the numerical errors in each time step, resulting in unbounded numerical solutions and testing instability.

Therefore, the singular value decomposition (SVD) method [11] is applied to the synthesis of the AFP controller, in order to reduce the numerical errors while inverting $X_{(n,N)}$ in (5). Firstly, $X_{(n,N)}{}^{\mathrm{T}}X_{(n,N)}$ is decomposed into

$$X^{\mathrm{T}}_{(n,N)}X_{(n,N)} = U\Sigma V^* \tag{17}$$

where Σ is a rectangular diagonal matrix with the eigenvalues of $X_{(n,N)}{}^{\mathrm{T}}X_{(n,N)}$ on the diagonal entries, and $\{U, V^*\}$ are unitary matrices related to the directions of input and output vectors. The symbol of $*$ denotes the conjugate transpose. Secondly, the diagonal entries of Σ with small and negligible eigenvalues are set to zero, and taking the inverse of the remaining diagonal elements, the modified matrix is denoted as $\tilde{\Sigma}$. Thus, the substitution of (17) and $\tilde{\Sigma}$ into (5) yields

$$\underline{\tilde{a}}_{(N)} = \left(V\tilde{\Sigma}U^*\right)X^{\mathrm{T}}_{(n,N)}\underline{z}_{\mathrm{N}}. \tag{18}$$

Using (18) reduces the numerical error in matrix inversion, such that the order of the polynomial equation can be increased further, yielding a higher-level prediction and synchronization accuracy.

IV. Implementation Studies

To illustrate the feasibility of the concepts in Sections II and III, a mass-spring-damper DSS is developed in this section for implementation studies. The associated dynamics and control synthesis is presented first, followed by the testing results and discussion.

A. The One-Mass-Two-Spring System

Fig. 3 shows the emulated system of a one-mass-two-spring (OMTS) device, where $\{k, k_s\}$ are the stiffness coefficients of the springs, and $\{m, c\}$ correspond to the mass and damping constants. Table I lists the associated parameters of the OMTS system. The dotted line indicates the substructured interface, which divides the OMTS system into Σ_{N1} and Σ_{P2} in Fig. 4. Here, $\{m, k, c\}$ are modeled numerically as Σ_{N1}, and k_s is the unknown specimen to be physically tested in Σ_{P2}. An electric-mechanical actuator (G_{TS}) is in series with Σ_2, and the measured force signal, f_{P}/y_i, is fed back to Σ_{N1} as a dynamic constraint. The test rig of Σ_{P2} is displayed in Fig. 5, where the actuator operated in the range ±50 mm, the load cell ±450 N.

To perform the DSS test, a numerical excitation signal, d_{N}, is sent to Σ_{N1}, and the control objective is to ensure that the numerical and physical outputs, z_{N} and z_{P}, are in perfect synchronization despite of the unwanted dynamics within G_{TS}. To resolve the synchronization problem, this section applies the AFP technique to control the OMTS DSS, based on the results of Sections II and III, and using the direct-compensation and SVD methods.

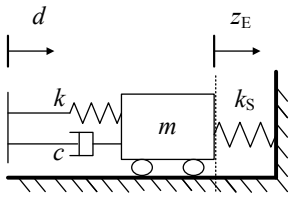


Figure 3. The emulated mass-spring-damper system (Σ_{E}).

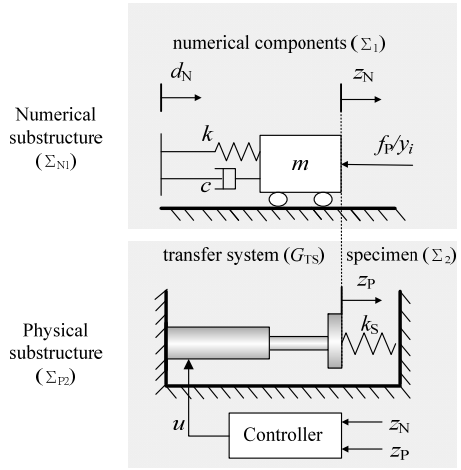


Figure 4. The OMTS substructured system.

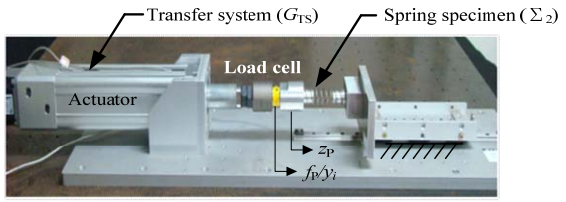


Figure 5. The test rig of Σ_{P2} showing the actuator and tested specimen.

B. Synthesis of the AFP Control Systems

The AFP compensators were implemented via a dSPACE 1104 system, with a control sampling interval of $\Delta t = 2 \times 10^{-3}$ s, and the parameters of G_{TS} were identified as $\{a, b\} = \{49.46, 47.51\}$. Therefore, the direct-compensation strategy of (14)-(15) yielded the value of P as follows

$$P = -\frac{1}{0.002}(4\pi)^{-1} \angle G_{TS}(4\pi j) = 10.3 \qquad (19)$$

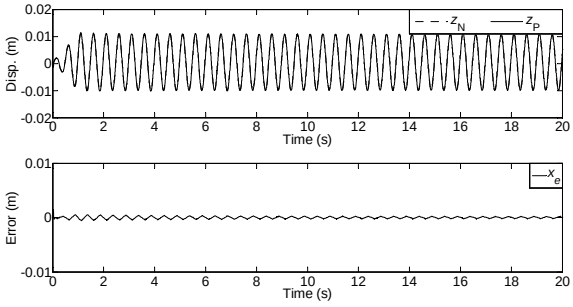
where the excitation frequency of d_N was known and given by $\omega = 4\pi$ rad/s with an amplitude of 0.02 m. This value of (19) was applied to the prediction vector in (9) as the initial condition of $P(t)$, assuming that $\tau(0)$ equates 2.06×10^{-2} s.

On the other hand, $n = 10$ was set for the AFP controller, and using the SVD method enabled the polynomial order to be promoted to $N = 6$, which was higher than that provided by [6] and [7]. Based on these setting, the initial gains of $\rho(0) = \sigma(0) = 0$ and the parameters of $\{\alpha, \beta, \gamma\} = \{30, 1, 2\}$ were applied.

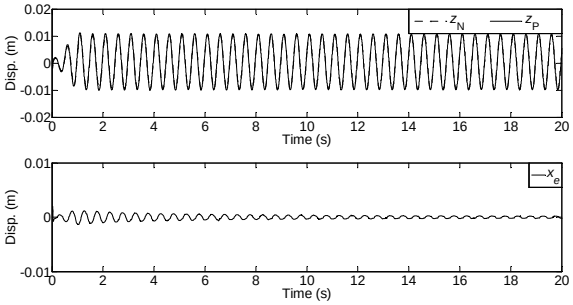
C. Implementation Results

Fig. 6(a) shows the AFP control results using the direct-compensation and SVD methods, where the responses of $\{z_N, z_P\}$ and x_e are depicted in the first and second subplots, respectively. The trajectories of $\{z_N, z_P\}$ were almost synchronous, and the magnitude of x_e was bounded within $\pm 6 \times 10^{-6}$ m, reflecting stable and accurate DSS performance, and verifying the proposed improvement methods.

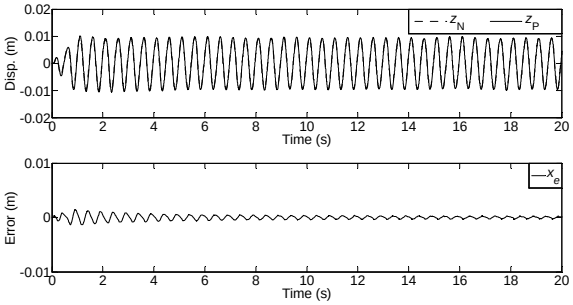
We further assumed that $P = 10.3$ was roughly the optimal prediction that separates the under from over-compensation regions. To prove that, we artificially increased and decreased P by 5, and the testing results are displayed in Figs. 6(b) and (c), respectively. As expected, the settling errors became relatively large in Figs. 6(b) and (c), due to the over or under-estimation of P. This comparison implies that if P is assigned with a large difference from the critical prediction value, significant settling errors leading to invalid tests may happen.



(a) Direct-compensation ($P = 10.3$)



(b) Over-compensation ($P = 15$)



(c) Under-compensation ($P = 5$)

Figure 6. Implementation results of the OMTS DSS with the SVD method, $N = 6$, and $n = 10$ applied.

In order to further investigate the synchronization accuracy related to P, a set of experiments were conducted, where P varied from 5 to 30 with an increment of 5. When P was set to 10.3 via the direct-compensation method, a minimal integration of square-error was obtained. Therefore, these experiments verify again that using the direct approach can determine P in an optimal and straightforward manner.

V. Conclusions And Future Work

Dynamically substructured systems (DSS) are widely applied to investigate dynamic behavior of engineering systems. The success of the test requires a high-quality controller to compensate for unwanted dynamics introduced by actuator systems. The adaptive forward prediction (AFP) technique is discussed and applied to cope with the DSS issues in this work. A direct-compensation strategy for the design of AFP controllers, which does not require knowing the nominal parameters of the numerical substructure and tested components, simplifies the control and stability analysis procedure. Singular value decomposition (SVD) method is considered to improve the numerical error of the AFP controller while inverting ill-conditioned matrices. Implementation studies of a one-mass-two-spring system demonstrate that the direct-compensation plus SVD methods provide a straightforward, reliable, and practical strategy for the tuning of the control parameters, as well as improve the achievable synchronization accuracy by increasing the order of the polynomial equations and the number of sampled data. In the future work, the robustness of the AFP controller in the presence of significant phase lags and realistic DSS problems will be considered.

APPENDIX

TABLE I.

Parameters	Notation	Values
Σ_1 and Σ_2		
Stiffness coefficient	k	1800 N/m
Damping coefficient	c	30 Ns/m
Mass	m	2.2 Kg
Stiffness coefficient	k_s	unknown
Transfer system (G_{TS})		
Numerator coefficient	a	47.51 s^{-1}
Denominator coefficient (b)	b	49.46 s^{-1}
AFP parameters		
Control sampling interval	Δt	2×10^{-3} s
Order of polynomial equation	N	6
Number of sampled data	n	10
Forward prediction parameters	P, $P(t)$	10.3
Excitation frequency	ω	4π rad/s
Adaptive weights	α, β, γ	30, 1, 2
Adaptive gains	$\sigma(t)$, $\rho(t)$	
Backward interpolation matrix	$X_{(n, N)}$	
Forward prediction vector	X_P	

ACKNOWLEDGMENT

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*', for the support in the pursuance of this work.

REFERENCES

[1] 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,* vol. 359, pp. 1869-1891, 2001.

[2] S. Oncu, S. Karaman, L. Guvenc, S. S. Ersolmaz, E. S. Ozturk, E. Cetin, and M. Sinal, "Robust yaw stability controller design for a light commercial vehicle using a hardware in the loop steering test rig," in *Proceedings of the 2007 IEEE Intelligent Vehicles Symposium*, Istanbul, Turkey, 2007, pp. 852-859.

[3] J. Y. Tu, "Development of numerical-substructure-based and output-based substructuring controllers," Structural Control and Health Monitoring, available online, 2012.

[4] C. Chen and J. M. Ricles, "Analysis of actuator delay compensation methods for real-time testing," Engineering Structures, vol. 31, pp. 2643-2655, 2009.

[5] A. P. Darby, M. S. Williams, and A. Blakeborough, "Stability and delay compensation for real-time substructure testing," Journal of Engineering Mechanics, vol. 128, pp. 1276-1284, 2002.

[6] M. I. Wallace, D. J. Wagg, and S. A. Neild, "An adaptive polynomial based forward prediction algorithm for multi-actuator real-time dynamic substructuring," *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Science,* vol. 461, pp. 3807-3826, 2005.

[7] M. I. Wallace, J. Sieber, S. A. Neild, D. J. Wagg, and B. Krauskopf, "Stability analysis of real-time dynamic substructuring using delay differential equation models," Earthquake Engineering & Structural Dynamics, vol. 34, pp. 1817-1832, 2005.

[8] C. Y. Chen, Y. C. Chen, W. D. Hxiao, and J. Y. Tu, "Preliminary feasiblity studies of real-time substructuring control strategies," submitted to Asian Control Conference, Istanbul, Turkey, 2012.

[9] E. Kreyszig, Advanced engineering mathematics, 10th ed. New York: Wiley, 1999.

[10] K. Engelborghs, T. Luzyanina, and D. Roose, "Numerical bifurcation analysis of delay differential equations using DDE-BIFTOOL," ACM Transactions on Mathematical Software, vol. 28, pp. 1-21, 2002.

[11] S. Skogestad and I. Postlethwaite, Multivariable feedback control: analysis and design, 2nd ed. Chichester: John Wiley & Sons Ltd, 2005.