13th World Conference on Earthquake Engineering
Vancouver, B.C., Canada
August 1-6, 2004
Paper No. 2529

USING ADAPTIVE CONTROL FOR DYNAMIC SUBSTRUCTURING TESTS

C.N.Lim[1], S.A.Neild, D.P.Stoten, C.A.Taylor and D.Drury

SUMMARY

Dynamic substructuring, or real-time hybrid testing, is a technique that allows the experimental testing of critical elements of a structure within the context of the overall structure. The emulated structure is split into two parts: the region of interest containing the critical element, which is physically tested and the remainder of the structure, which is numerically modelled. For an accurate representation of the overall structure, it is crucial that the two parts interact, via a controller, in real-time with minimal errors at the interfaces between them. For example, the displacements at the interface may be passed from the numerical model to the controller, which imposes them on the physical substructure using hydraulic actuators (the transfer system). The forces required to impose these displacements may be in turn passed back to the numerical model.

The adaptive minimal controller synthesis (MCS) controller has been implemented for substructuring of a one degree-of-freedom mass-spring-damper test and evaluation system, Neild [1, 2, 3], where a proportion of the mass was taken as the experimental substructure, and the remainder of the system numerically modelled. The potential advantage of such a controller over traditional linear controller is that it will be capable of adapting to unknown dynamics, disturbances and non-linearities in the experimental specimen. The work presented in this paper described the use of the MCS controller on a more complex substructuring system, consisting of a two degree-of-freedom mass-spring-damper system.

INTRODUCTION

Real-time substructuring (or real-time hybrid testing) is a dynamic testing technique, which involves experimentally testing a part of a structure within the context of the overall structure. This is achieved by splitting the structure to be emulated into two parts, which interact together in real-time. One part consists of typically the non-linear region, or any other region that is difficult to accurately model (e.g. complex structural element within a building), and is tested experimentally (the *physical substructure*). The other part is the remainder of the structure (e.g. building excluding the complex structural element) and is modelled numerically (the *numerical substructure*). The displacement of the shared degrees of freedom are calculated using the numerical substructure and are then imposed on the test specimen using

[1] Department of Mechanical Engineering, University of Bristol, Queen's Building, University Walk, Bristol, BS8 1TR, UK. Email: michael.lim@bristol.ac.uk

actuators and the forces required to impose them are passed back into the model. For these two parts to accurately represent the dynamics of the complete structure (the *emulated system*) they must operate in parallel with minimal differences at the interfaces between them. The potential advantage of this test technique over traditional experimental methods is that, since only a portion of the structure is experimentally tested, it can be tested at full scale or at a larger scale than would otherwise be possible.

One of the critical elements necessary for the test specimen and numerical substructure to replicate the behaviour of the emulated system is accurate control of the actuators, which apply the displacements to the test specimen. Most attempts to improve the control of the actuator system in order to obtain better substructuring performance have followed the approach proposed by Horiuchi [4].

In this paper, the concept of substructuring is demonstrated in an example where there is ideal actuator control and no actuator dynamics present, such that perfect desired interface response is obtained. Subsequently, it is shown that substructuring is not properly implemented if a realistic actuator or transfer system dynamics is considered, unless a suitable controller is used to compensate for this. The results of substructuring implementation in this paper were obtained via simulation.

An adaptive controller is potentially a more attractive solution over conventional linear scheme, since it has the ability to control a non-linear physical substructure (the inner-loop controller plus the actuator system and test specimen). Wagg [5] developed a modified version of the minimal controller synthesis (MCS) adaptive control algorithm suitable for substructuring and demonstrated that the system was stable in continuous time. The benefits of using the adaptive MCS controller over a linear controller in a substructuring context were demonstrated in simulation by Stoten [6]. The ability of the MCS controller to operate satisfactorily for a variety of numerical and physical substructure combinations was also demonstrated. Early experiments implementing substructuring with the MCS adaptive controllers on a six-degree of freedom shaking table at the University of Bristol (Neild [1]) were performed for a single degree-of-freedom mass-spring-damper system.

THE SUBSTRUCTURED SYSTEM CONFIGURATION

The emulated system is a two degree-of-freedom mass-spring-damper system with two points of excitation (r_1 and r_2) at either end as shown in Figure 1. The system was split into the numerical model, consisting of the mass *B* and the spring *b* and the physical substructure consisting of spring *c*, mass *A* and spring *a*. The experimental substructure that was used for the testing of the substructuring system is shown in Figure 2. It comprises of two hydraulic actuators each placed at the extremes of the rig, and between which a linkage of springs and masses are positioned to represent the physical substructure.

The numerical model computes the displacement at the numerical model-physical substructure interface x_{c1m} by utilising the interface force information fed back form the physical substructure. The interface force was measured within the physical substructure at the interface. Actuator *a* was used to impose the excitation r_2 on the physical substructure. In the substructure system, actuator *b* generates the displacement at the interface between the numerical model and the physical substructure, x_{c1}. Ideally if the dynamics of actuator *b* are compensated for exactly using the controller then $x_{c1}=x_{c1m}$ and there will be no errors at the interface. The control of this actuator is critical to the successful emulation of the whole system within the substructuring framework.

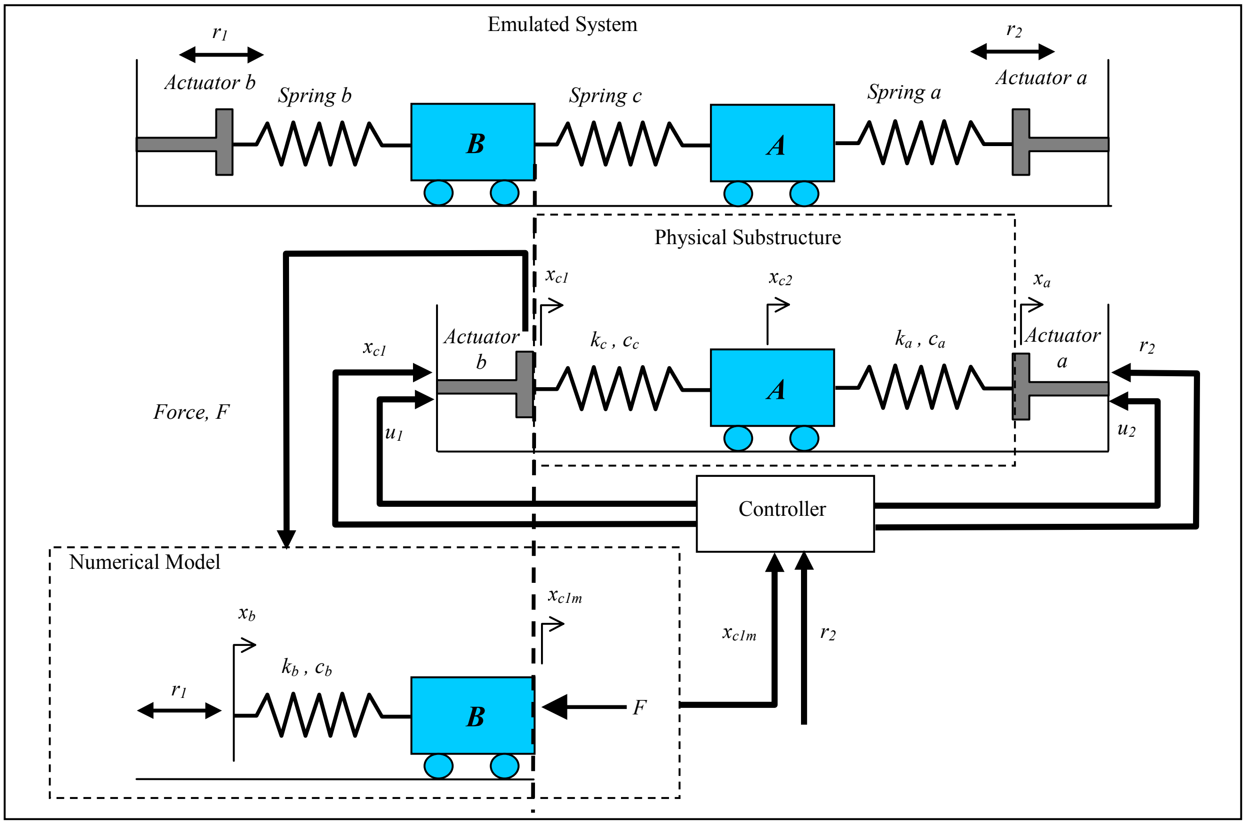


Figure 1: Schematic representation of the substructuring experiment with segmentation of emulated system into the numerical model and physical substructure.

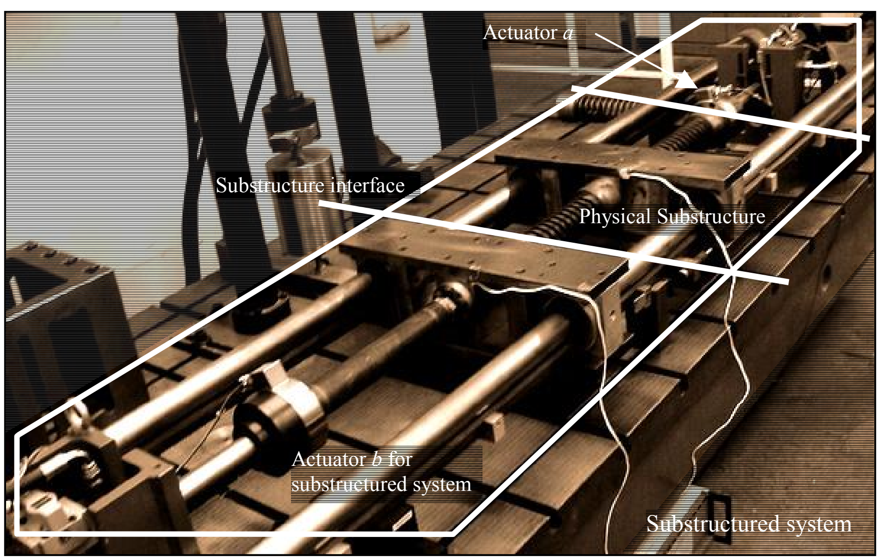


Figure 2: Photograph of the experimental arrangement of the substructured system. The physical substructure, substructure interface and the actuators are indicated.

Plant Dynamics

The plant is defined as the physical substructure portion of the whole substructured system including the transfer system, which accepted the substructuring interface control signal u_1 as its input and returned the force acting at the substructure interface as its output, F. Viscous friction in each of the springs *a, b* and *c* was included in the simulation so that the plant dynamics will be consistent with that of the experimental counterpart. The emulated system was modelled alongside the substructuring implementation during the tests to compare the output displacements of the masses *A* and *B* obtained from the tests. Assuming no interface error, the equation of motion for displacement of mass *B*, x_{c1} was found to be:

$$x_{c1} = \frac{(c_b s + k_b)x_b + (c_c s + k_c)x_{c2}}{m_B s^2 (c_b + c_c)s + (k_b + k_c)} \tag{1}$$

and for mass *A*, x_{c2}:

$$x_{c2} = \frac{(c_a s + k_a)x_c + (c_c s + k_c)x_{c1}}{m_A s^2 (c_a + c_c)s + (k_a + k_c)} \tag{2}$$

where s is the Laplace transform variable. The force exerted at the numerical model-physical substructure interface by spring c, acting onto the interface with numerical model can be obtained indirectly from the displacement and velocity measurement of mass *A* and *B* via

$$F = c_c(\dot{x}_{c2} - \dot{x}_{c1}) + k_c(x_{c2} - x_{c1}) \tag{3}$$

Demonstration of substructuring implementation

To demonstrate the concept of substructuring implementation, issues with control and transfer system dynamics were neglected and they were assumed to behave in an ideal fashion. This allows the displacement at the interface calculated by the numerical model to correspond exactly with that predicted by the emulated system.

(a)

(b)

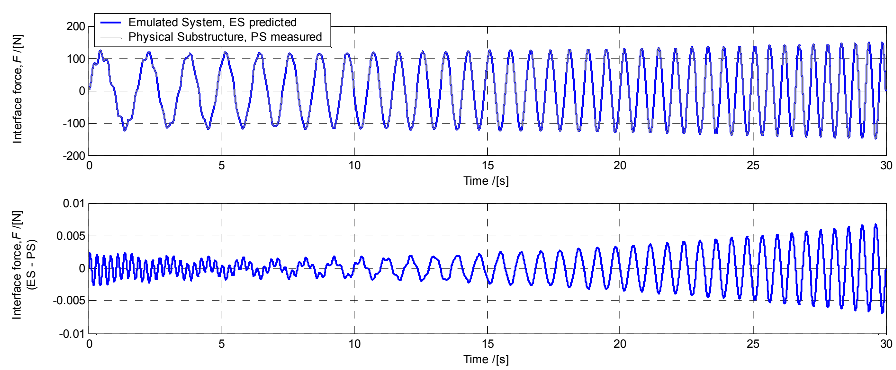


Figure 3: Simulated –results for (a) interface force, *F* predicted by the emulated system and measured from the physical substructure; (b) the difference of *F* between emulated system and physical substructure.

The following example illustrates a simple test of substructuring where the physical substructure end, r_2, was excited in a swept sinusoidal fashion from 0.5 to 1.5 Hz at an amplitude of 0.02m over a period of 20s with linear frequency increment while the numerical model end was kept still. It can be seen in Figure 3(a) for the interface force, F and Figure 4(a) and Figure 4(b) for the displacement of the masses (x_{c1} and x_{c2}), that proper substructuring implementation has been carried out. The interface force and displacements modelled and measured from the numerical model and physical substructure closely matched the predicted values by the emulated system. The errors between them shown in Figure 3(b) and

Figure 4(b) are due to truncation of values during computation in the simulation, although a relatively more significant error can be observed (in Figure 3(b)) between the interface force predicted by the emulated system and that measured in the physical substructure. This error was due to the slight phase lag in outputting the force reading between the two model blocks in Simulink. The interface force, F and displacements, x_{c1} and x_{c2} were also gradually increasing with frequency. The errors between the predicted and the modelled values are also greater at higher excitation frequency.

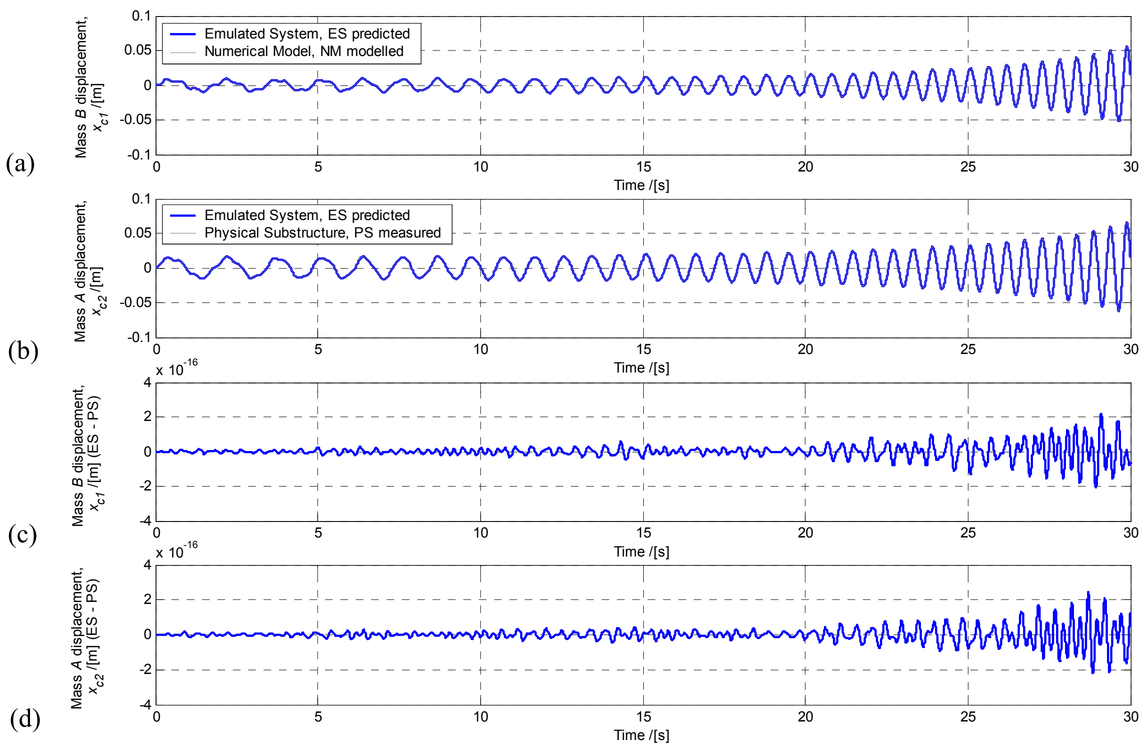


Figure 4: Simulated –results for (a) mass *B* displacement, x_{c1} (b) mass *A* displacement, x_{c2}, predicted by the emulated system and modelled by numerical model and the physical substructure respectively; (c) the difference of x_{c1} and (d) of x_{c2} between emulated system and physical substructure.

Substructuring problems related to transfer system issues

In practice, there will always be some form of transfer system dynamics present at the interface between the numerical model and the physical substructure as illustrated in Figure 5. Therefore, the correct substructuring model must contain the transfer system dynamics. Consequently, there is an increase in the phase lag of the feedback force from the physical substructure to the numerical model.

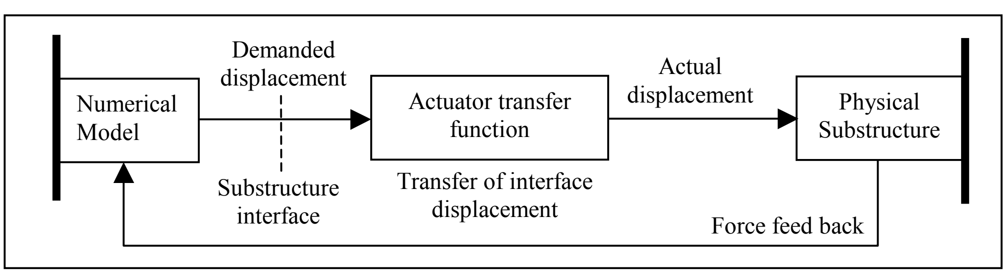


Figure 5: Diagram showing the presence of the transfer system and the inner loop control to compensate for its dynamics place between the numerical model and physical substructure.

In this case, both the actuators have an estimated transfer function of:

$$G_{TS} = \frac{20}{s+20} \tag{4}$$

The presence of the actuator dynamics in the substructured system complicates the transfer of interface displacement modelled by the numerical model onto the physical substructure. If there is a phase lag between the intended transfer of interface displacement and when it actually take place in the transfer system, the substructure may become unstable. The phase lag between the demanded displacement and the actual displacement leads to errors in the applied interface displacement and the measured corresponding interface force both of which were cumulative with time.

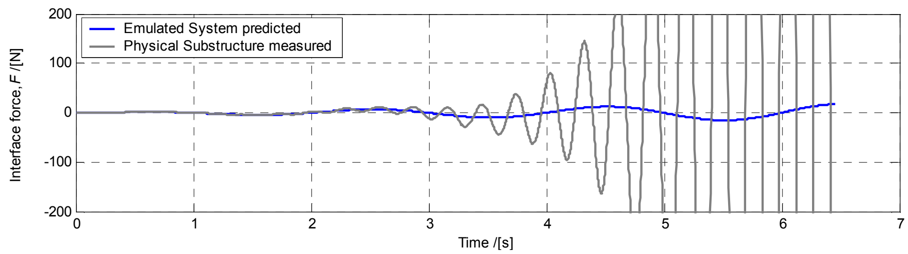
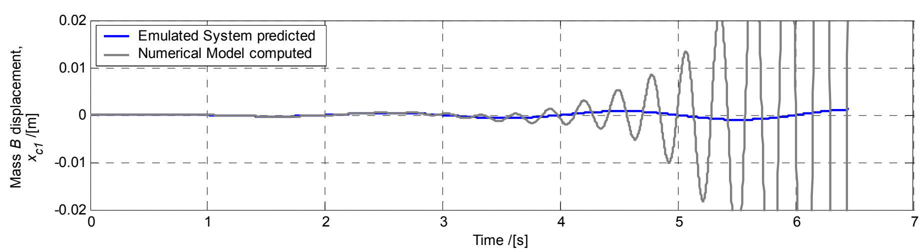


Figure 6: Simulated – Substructuring results for the interface force, *F* predicted by the emulated system and modelled by the physical substructure.

Figure 6 shows the resulting interface force and Figure 7 shows the resulting displacements, x_{c2} and x_{c1} of masses *A* and *B* respectively, when the dynamics of the transfer system was considered in the substructuring implementation. The force and displacement diverge from the values predicted by the emulated system when the instantaneous force fed back used to compute the response of interface displacement was applied to the wrong time step due to the phase lag, resulting in an inaccurate numerical model.

(a)



(b)

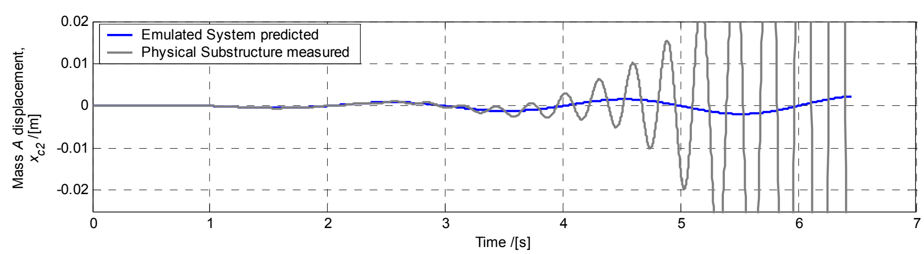


Figure 7: Simulated – substructuring results for (a) mass *B* displacement, x_{c1} (b) mass *A* displacement, x_{c2}, predicted by the emulated system and modelled by numerical model and the physical substructure respectively.

Furthermore, due to the lag in the actuator response, externally induced dynamics were also added into the substructure resulting in instability. These are evident with the mass displacements modelled by the numerical model and the physical substructure moving in opposite directions (In Figure 7, see numerical model modelled plot in (a) and physical substructure modelled plot in (b)) whilst they were predicted to transverse closely in phase between them and with the wall excitation at the given excitation frequency. When masses were moving in opposite directions, large induced forces caused by spring *c* acted on the substructuring interface. This erroneous measured force results in errors in the numerical model displacement that in the actual case should produce a small interface force. The subsequent computation produced the incorrect numerical model and corresponding interface displacement and this destabilise the whole system. Relative mass motion phase difference is observed to cause resonance where the second mode of vibration was introduced into the system. This was seen as extreme mass *A* and *B* displacements and vibrations in the system.

In substructuring, where the transfer system dynamics problem is evident, the force computed by the emulated system could be used to ensure the numerical model stability. This is possible because the use of emulated system predicted force omits the phase delay on the force signal fed back to the numerical model. However, this may not always be possible and also partially defeats the actual objective of the substructure testing technique, where the concept is to feed back the force measured from the complex physical substructure (that may not be easily modelled within the emulated system).

The following section describes the work carried out in developing an adaptive controller scheme to appropriately control the substructured system and compensate for the actuator dynamics. Two excitation cases were considered and the control performance was assessed for simulated substructuring implementation with excitation introduced in Case 1, at the physical substructure end (r_2) and in Case 2, at the numerical model end (r_1).

ADAPTIVE CONTROL

The MCS Controller

The adaptive controller considered for use with substructuring in based on the first order Minimal Control Synthesis (MCS) (Stoten [7, 8]) algorithm. The MCS algorithm is an adaptive model reference control strategy where the *a priori* knowledge of the substructure dynamics is not required when designing the controller while the controller is able to cope with parameter variations. It is an extension of the Model Reference Adaptive Controller (MRAC) that moderates some of the criteria required by MRAC, resulting in a simpler implementation of control scheme. This is particularly important when non-linear or ill-defined substructures are tested since the corresponding closed-loop dynamics will also be essentially non-linear.

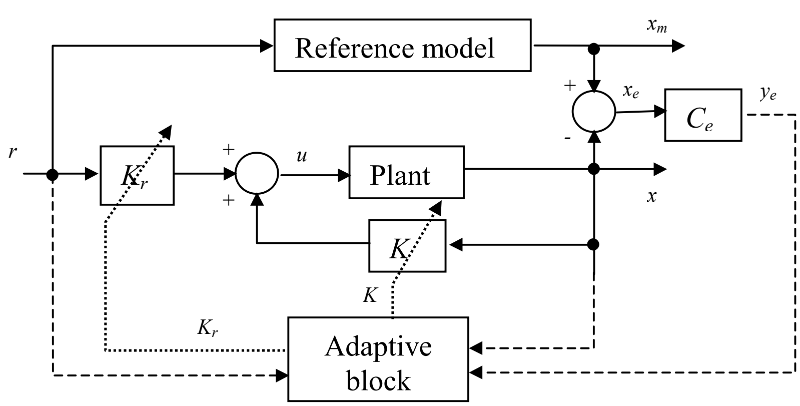


Figure 8: Block diagram representation of the MCS algorithm.

The form of the MCS controller is shown in Figure 8. The plant is assumed to have first order dynamics

$$x(s) = \frac{b}{s+a} u(s) \tag{5}$$

where b and a are the plant parameters. The demand r is fed into a reference model, which has the effect of limiting the bandwidth of the response of the plant. The MCS algorithm is designed to control the plant to this band-limited output from the reference model. For the first-order version of the controller the equation governing the reference model in Laplace form may be written as:

$$x_m(s) = \frac{b_m}{s+a_m} r(s) \tag{6}$$

where b_m and a_m are the reference model parameters and are based on the desired settling time of the closed loop system. It is usual for $b_m = a_m$ such that the reference model has unity steady-state gain. The control signal u is based on the demand, the plant displacement x and the adaptive gains K and K_r.

$$u(t) = K(t)x(t) + K_r(t)r(t) \tag{7}$$

The gains adapt such that the error y_e is reduced and are governed by the equations:

$$K(t) = \alpha \int_0^t y_e(\tau)x(\tau)d\tau + \beta y_e(t)x(t) \tag{8}$$

$$K_r(t) = \alpha \int_0^t y_e(\tau)r(\tau)d\tau + \beta y_e(t)r(t) \tag{9}$$

where α and β are adaptive weightings and the initial gains are set to zero. Empirically it has been found that the ratio $\alpha = 10\beta$ works well and increasing α improved the rate of adaptation at the expense of increased noise propagation around the loop. In the case of a first order controller, y_e is proportional to x_e, the displacement error between the reference model and plant output, $y_e = C_e x_e$. The error dynamics for the system may be derived as:

$$\dot{x}_e = -a_m x_e + [a - a_m - bK]x + [b_m - bK_r]r \tag{10}$$

This equation may be reduced to:

$$\dot{x}_e = -a_m x_e \tag{11}$$

when the gains take the Erzberger values, Stoten [7], which are defined as:

$$K = K^E = \frac{a - a_m}{b} \ , \ K_r = K_r^E = \frac{b_m}{b} \tag{12}$$

The MCS controller must be altered for use with substructuring to incorporate the numerical model. Wagg [5] proposed that the reference model is replaced with the numerical model and the plant with the transfer system and physical substructure. There are two main consequences of the alteration. Firstly, since the numerical model is determined by the structure to be emulated using the substructuring technique, it is unlikely to take the same form as the reference model, i.e. the form of a first order transfer system. Secondly, there is an additional link from the plant to the numerical model providing the force feedback The modified structure is shown in Figure 9 and will be referred to as *substructuring MCS*.

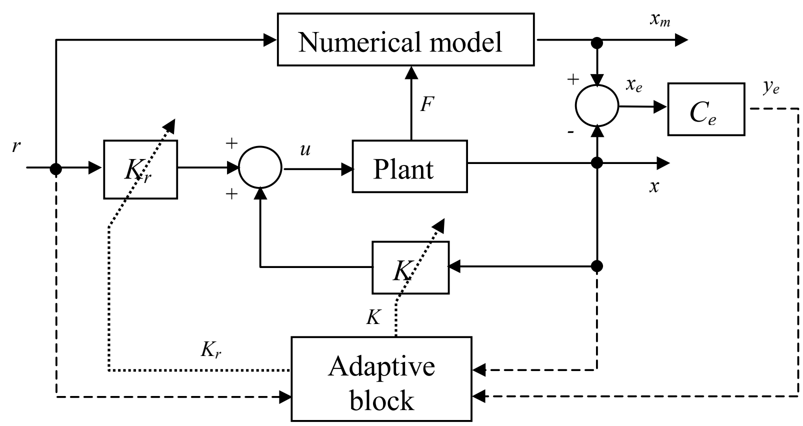


Figure 9: Block diagram representation of the MCS algorithm for substructuring

This form of the MCS algorithm for substructuring has been tested extensively on a simple one degree-of-freedom emulated system consisting of a mass connected to ground via both a spring and a damper, Neild [1, 2]. The system was split such that the physical substructure consisted of a proportion of the mass and the numerical model consisted of the spring, the damper and the remained of the mass. The system was excited through ground motion r within the numerical model. The resulting numerical model may be expressed as:

$$x_m(s) = \frac{cs + k}{ms^2 + cs + k} r(s) - \frac{1}{ms^2 + cs + k} F(s) \tag{13}$$

where x_m is the displacement of the mass within the numerical model, k, c, m are the spring stiffness, the damping coefficient and the mass within the numerical model respectively and F is the force feedback from the physical substructure. It was found that the second order transfer function between r and x_m had a deleterious effect of the performance of the controller, Neild [2].

To overcome this problem, Stoten originally proposed that the numerical model should be split into two parts, Neild [2]:

$$x_m(s) = \frac{b_m}{s + a_m} r'(s) - \frac{1}{ms^2 + cs + k} F(s) \tag{14}$$

and:

$$r'(s) = \frac{cs^2 + (a_m c + k)s + a_m k}{b_m (ms^2 + cs + k)} r(s) \tag{15}$$

By using r' rather than r as the demand to the MCS controller, i.e. by generating the control signal using:

$$u(t) = K(t)x(t) + K_{r'}(t)r'(t) \tag{16}$$

where the adaptive gain $K_{r'}$ is governed by:

$$K_{r'}(t) = \alpha \int_0^t y_e(\tau) r'(\tau) d\tau + \beta y_e(t) r'(t) \tag{17}$$

It can be seen that the relationship between the MCS demand r' and the numerical model output x_m is identical to the reference model relationship in (6) albeit with an additional term due to the force feedback. The substructuring results using this strategy were very good, with minimal errors between the substructured system and the system it was emulating. A similar strategy of rewriting the numerical model into a form that contains a intermediately signal *r'* which may be used as the MCS controller demand is employed here. We shall refer to this type of substructuring control strategy as *substructuring MCS with modified demand* (MCSMD).

MCS Substructuring for the Two DOF System

For the two degree-of-freedom system the numerical model may be expressed as:

$$x_{c1m}(s) = \frac{c_b s + k_b}{m_B s^2 + c_b s + k_b} r_1(s) - \frac{1}{m_B s^2 + c_b s + k_b} F(s) \tag{18}$$

Using the substructuring MCS strategy the MCS demand would be r_1 and the numerical model output, x_{c1m}, would be used in place of the reference model output. Ideally the controller would generate a control signal u_1, which would ensure that the interface actuator (actuator *b*) displacement x_{c1} follows x_{c1m} exactly.

Excitation Case 1

For the case where ground excitation is only present at the physical substructure end, i.e. at r_2 with $r_1 = 0$, the numerical model (18) simplifies to:

$$x_{c1m}(s) = -\frac{1}{m_B s^2 + c_b s + k_b} F(s) \tag{19}$$

The substructure MCS strategy is unsuitable for this numerical model. This is because the MCS demand r_1 and the plant displacement, x_{c1} are initially zero, therefore the control signal will remain zero for all time and hence the transfer system will not move regardless of the error at the interface between the numerical model and the physical substructure.

To use the MCSMD strategy the numerical model must be rewritten to incorporate the intermediary variable *r'*. This may be done by writing:

$$x_{c1m}(s) = \frac{b_m}{s + a_m} r'(s) \tag{20}$$

where:

$$r'(s) = -\frac{s + a_m}{b_m (m_B s^2 + c_b s + k_b)} F(s) = G_F(s) F(s) \tag{21}$$

Figure 10 shows a block diagram representation of the substructuring with MCSMD strategy. Figure 11 shows a comparison of displacements of the two masses using a simulation of the substructured system and a simulation of the emulated system. The simulation used $a_m = b_m = 40$ and $\alpha = 5$ and $C_e = 400$. The ground excitation imposed is a 0.5Hz sine wave with amplitude 0.02m, which is linearly ramped up from zero over the first 20s. It can be seen that after about 30s there is good agreement between the emulated and the substructured systems. However this is short lived, in the substructured system there is a rapid growth of the second mode followed by a gradual decay away. After 60s good control is achieved with the adaptive gains (see Figure 11(b)) stabilizing at approximately $K = -1$ and $K_{r'} = 2$. These correspond to the Erzberger gains for this plant and reference model.

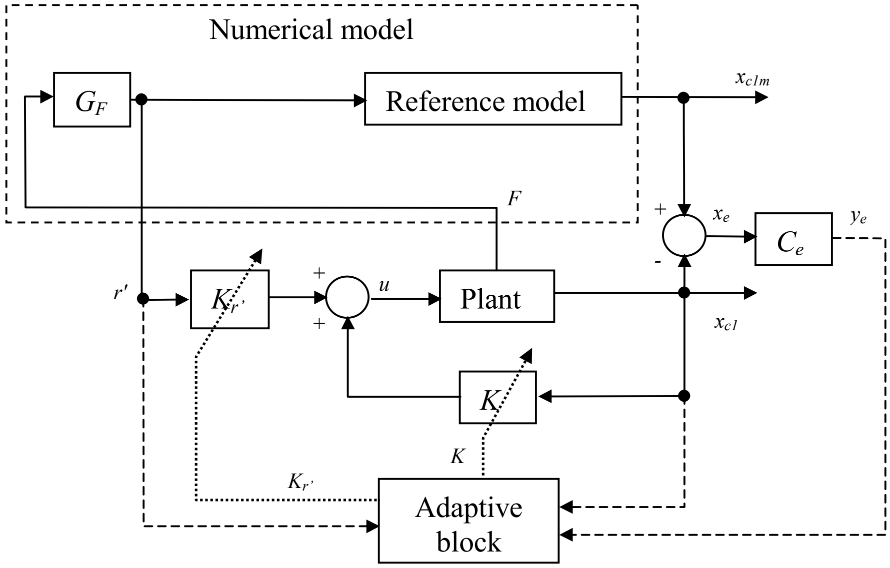
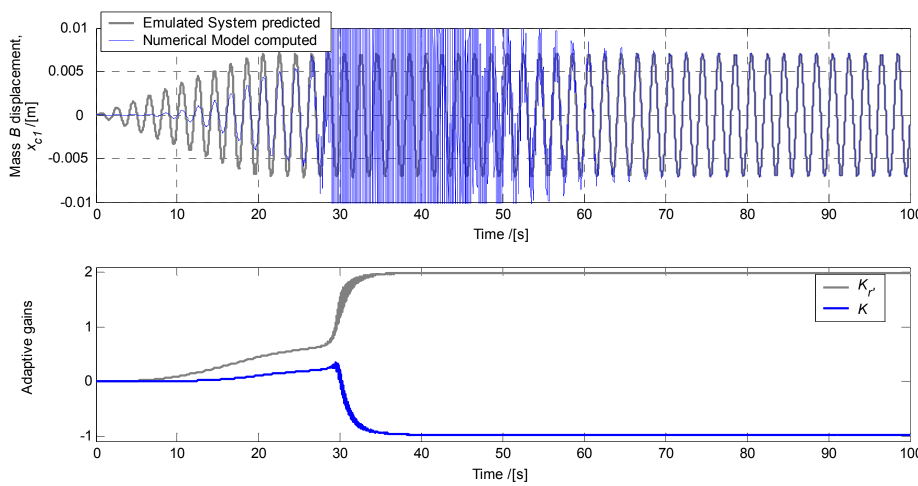


Figure 10: Block diagram representation of the MCSMD algorithm for the two degree-of-freedom system, Case 1, the plant contains both the interface actuator *b* and the physical substructure and has a second input r_2, which for clarity is not shown.

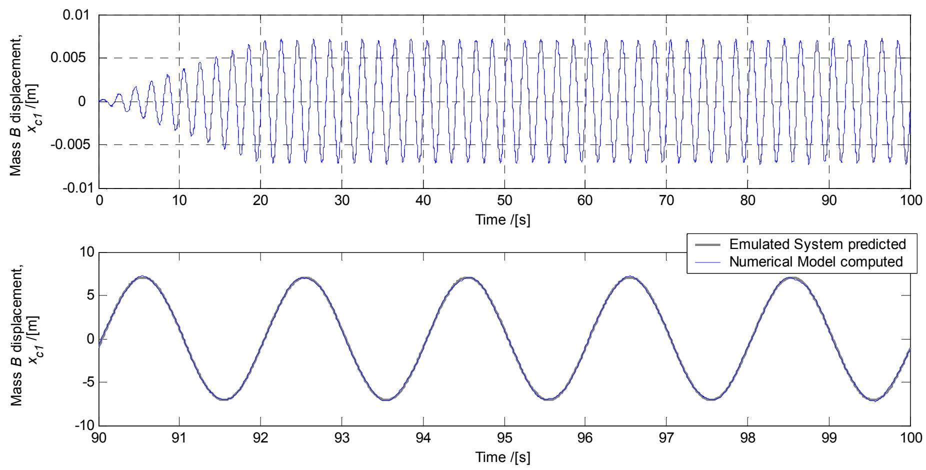


(a)

(b)

Figure 11: Substructuring using MCSmd, Case 1, mass *B* displacement, x_{c1} and MCSMD gains (K, $K_{r'}$)

To avoid the undesirable behaviour between 30 and 60s, it is necessary to use initial gains that are close to the Erzberger gain values rather than using zero initial gains. In simulation this is easily achieved as both the plant and the reference model parameters are known, however in practice the plant parameters will not be known exactly so the gain values will not be set exactly. Setting the initial gains to the Erzberger gains results in near perfect control of the substructured system in simulation. However a more realistic simulation is shown in Figure 12. Here the plant setting time used in calculating the Erzberger gains in 10% too great, resulting the initial gains being set to $K(0)$ = -1.2 and $K_{r'}(0)$ = 2.2, and white noise has been added to both the displacement and the force feedback from the physical substructure. For both the force and the displacement, the amplitude of the noise is approximately 5% of the signal amplitude. It should be noted that for ease of comparison the physical substructure interface displacement plotted is that before the noise has been added. The control of the displacement was excellent with the numerical model computed displacement perfectly overlaps that predicted by the emulated system.



(a)

(b)

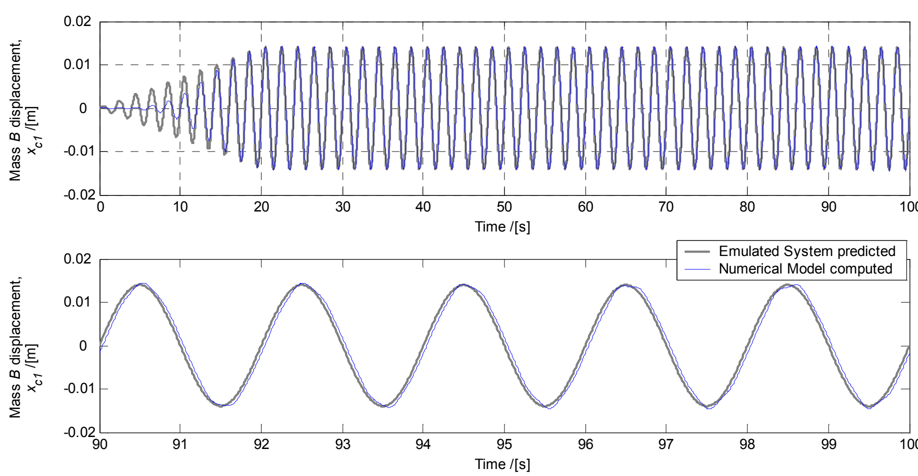
Figure 12: Substructuring using MCSMD with initial gains, Case 1, displacement of mass *B*, x_{c1}.

A similar technique of setting the initial gains to the Erzberger gains may not be used for the substructring MCS strategy since the reference model has been replaced with a higher order numerical model such that the Erzberger gains do not exist for the controller.

Excitation Case 2

Figure 13 shows the physical substructure interface displacement, x_{c1}, for a simulation using the substructuring MCS strategy with $\alpha = 5$ and $C_e = 400$ and a ground excitation consisting of a 0.5Hz sine wave with amplitude 0.02m, which is linearly ramped up from zero over the first 20s imposed at r_1 and with $r_2 = 0$. Slight phase lag could be seen between the displacement predicted by the emulated system and that computed by the numerical model. The figure shows that initially the controller performs well; however there is a gradual increase in the amplitude of the second mode of vibration at approximately 3.8Hz. This is particularly apparent for the displacement of mass *A*, x_{c2}, shown in Figure 14. The numerical model for this case is given by:

$$x_{c1m}(s) = \frac{c_b s + k_b}{m_B s^2 + c_b s + k_b} r_1(s) - \frac{1}{m_B s^2 + c_b s + k_b} F(s) \tag{22}$$



(a)

(b)

Figure 13: Substructuring using MCS, Case 2, displacement of mass *B*, x_{c1}.

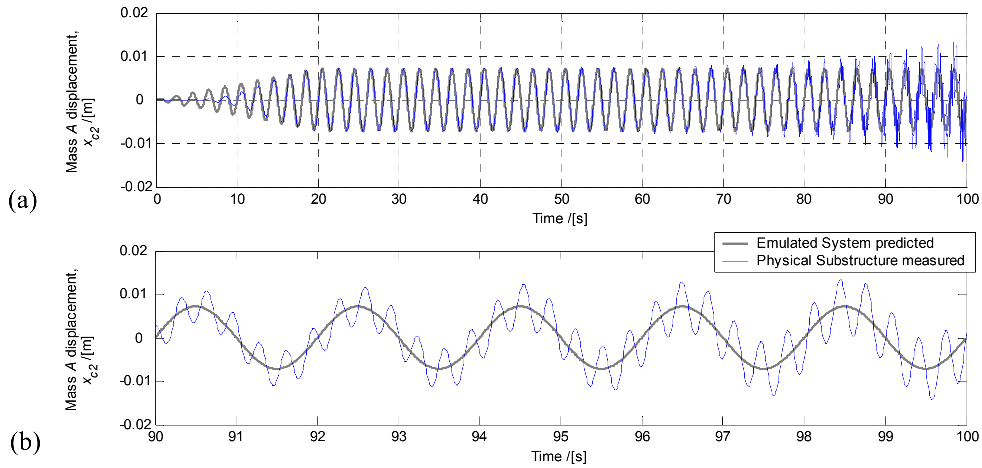


Figure 14: Substructuring using MCS, Case 2, displacement of mass *A*, x_{x2}.

Using the MCSMD strategy the numerical model may be rewritten as:

$$x_{c1m}(s) = \frac{b_m}{s + a_m} r'(s) \tag{23}$$

where the intermediary variable *r'* is given by:

$$\begin{aligned} r'(s) &= \frac{c_b s^2 + (c_b a_m + k)s + k_b a_m}{b_m (m_B s^2 + c_b s + k_b)} r_1(s) - \frac{s + a_m}{b_m (m_B s^2 + c_b s + k_b)} F(s) \\ &= G_r(s) r_1(s) + G_F(s) F(s) \end{aligned} \tag{24}$$

Figure 15 shows a block diagram of the MCSMD control strategy.

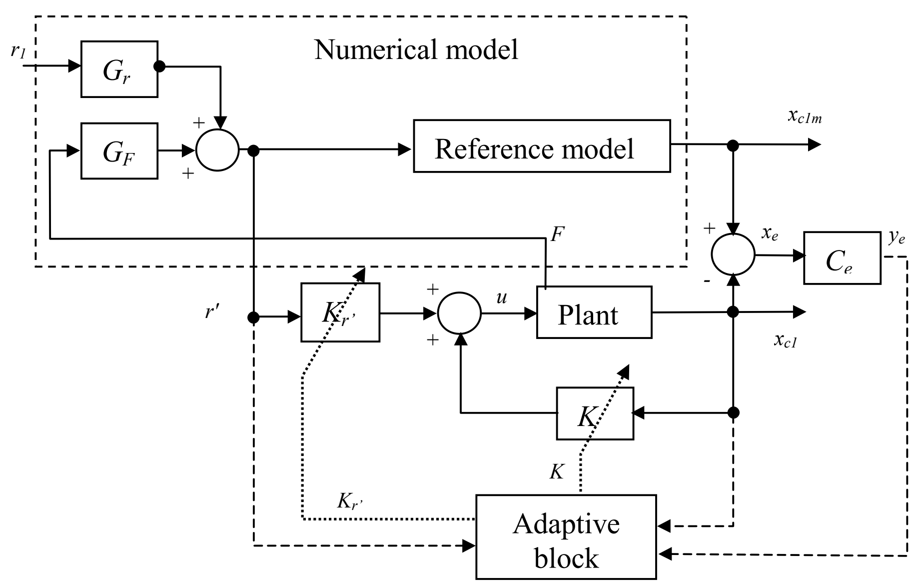


Figure 15: Block diagram representation of the MCSMD algorithm for the two degree-of-freedom system, Case 2

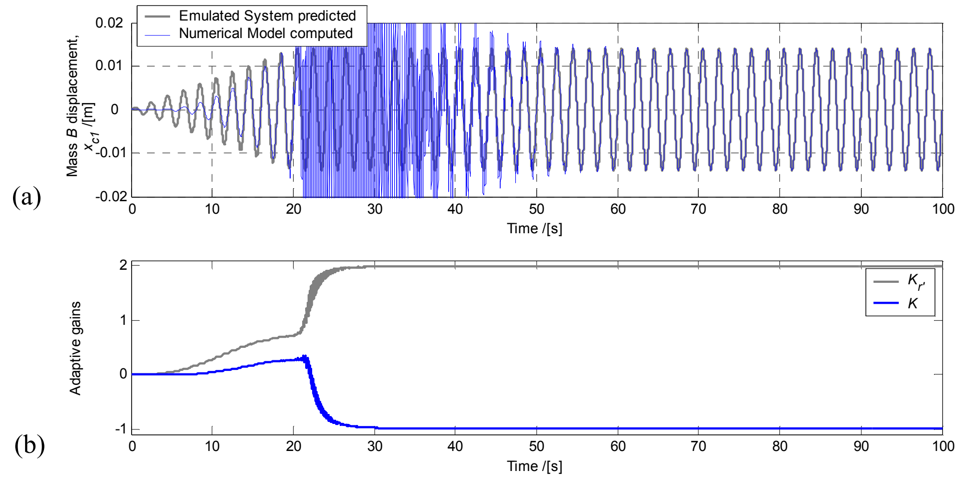


Figure 16: Substructuring using MCSMD, Case 2, displacement of mass *B*, x_{c1} and gains (K, $K_{r'}$)

As with the Case 1 example, there is a period of rapid growth in the amplitude of the second mode followed by a gradual decay away, see Figure 16. After about 50s the minimisation of the interface error using the MCSMD strategy is very good, with the adaptive gains settling to the Erzberger gains. To eliminate the period of adaptation where the second mode is excited the initial gain are set close to the Erzberger gains. As in the Case 1 example the initial gains are set slightly in error compared to the Erzberger gains, $K(0)$ = -1.2 and $K_{r'}(0)$ = 2.2, and white noise was added to both the displacement and the force feedback from the physical substructure. It can be seen from Figure 17 that this strategy results in excellent control.

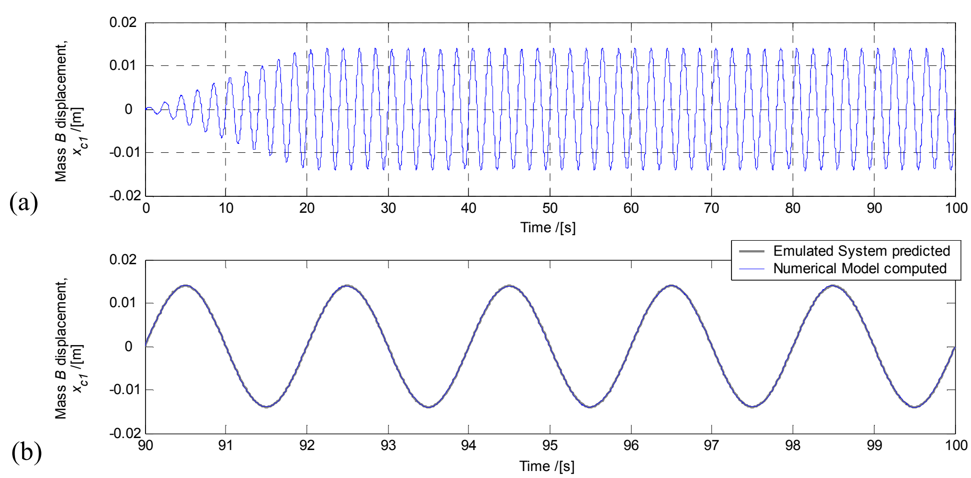


Figure 17: Substructuring using MCSMD with initial gains, Case 2, displacement of mass *B*, x_{c1}.

CONCLUSIONS

The effectiveness of the control of the transfer system is essential in ensuring accuracy of the entire substructuring implementation. This has been demonstrated in the simulations and the experiments conducted with a substructured system having a spring-mass-spring as the physical substructure and the remaining mass-spring section of the emulated system numerically modelled. The accuracy of controlling the transfer system is directly related to the subsequent accuracy of the numerical model such that if there is an error such as that induced by phase lag, the numerical model will be incorrect. This will propagate through to the next time step and the measured substructuring parameters further diverge away from those predicted by the emulated system thus leading to instability.

MCS was shown able to compensate for the transfer system dynamics where phase lag was presented as potential control problems in real-time substructuring. The tests carried out in simulation with the modified MCSmd controller scheme demonstrated good substructuring control. The next stage is to extend the implementation to the experimental rig. Future work involves developing the control algorithm further to cope with varying experimental substructuring test conditions and arrangements.

ACKNOWLEDGEMENTS

The authors gratefully acknowledge the support of the EPSRC grant: 'The Development of Multi-Axis Real-Time Substructuring Testing' and the EC grant: 'New fields of research in earthquake engineering experimentation' (HPRI-CT-2001-50023) in carrying out this work.

REFERENCES

1. 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. *Proceedings of the Third Conference in Structural Control*, Como, Italy, April 2002, Volume 2, 811-817
2. S.A. Neild, D.P. Stoten, D. Drury. Substructuring using the MCS adaptive controller. To be submitted to *Proceedings of the Institution of Mechanical Engineers, Part I, Journal of Systems and Control Engineers*.
3. S.A. Neild, D.P. Stoten, D. Drury and D.J. Wagg. Control issues relating to real-time substructuring, submitted to *Earthquake Engineering and Structural Dynamics*.
4. 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*, 1999; **28**: 1121-1141.
5. D. J. Wagg and D.P. Stoten. Substructuring of dynamic systems via the adaptive minimal control synthesis algorithm. *Earthquake Engineering and Structural Dynamics*, 2001; **30**: 865-877.
6. D.P. Stoten, S.A. Neild, D.J. Wagg and D. Drury. Dynamics and substructuring via the adaptive MCS algorithm. *Third World conference in Structural Control*, Como, Italy, April 2002, Volume 2, pp 819-825
7. D.P. Stoten. Model reference adaptive control of manipulators. *Mechanical Engineering Research Series*, Wiley and Sons, 1990.
8. D.P. Stoten and H. Benchoubane. Robustness of a minimal controller synthesis algorithm. *International Journal of Control*, 1990; **51**: 851-861
9. D.P. Stoten. An overview of the minimal control synthesis algorithm. *IMechE Conference on Aerospace Hydraulics and Systems* C474-033, London, 1993.