6th International Conference on Advances in Experimental Structural Engineering
11th International Workshop on Advanced Smart Materials and Smart Structures Technology
August 1-2, 2015, University of Illinois, Urbana-Champaign, United States

Development of Hybrid Simulation System for Multi-Degree-of-Freedom Large-Scale Testing

M.A. Moustafa[1], K.M. Mosalam[2]

1 *Assistant Professor, Dept. of Civil and Environmental Engineering, University of Nevada, Reno, United States*
E-mail: mmoustafa@unr.edu
2 *Professor, Dept. of Civil and Environmental Engineering, University of California, Berkeley, United States*
E-mail: mosalam@berkeley.edu

ABSTRACT
Hybrid Simulation (HS) is a fast-growing computational/physical testing technique that can replace shaking table tests using an online computational substructure to update the input signal based on the feedback from the physical substructure. This paper presents the development and validation of a new Hybrid Simulation System (HSS) and communication scheme. The developed HSS utilizes readily available laboratory data acquisition (DAQ) systems along with inexpensive TCP/IP-Ethernet protocols to establish multi-degree-of-freedom communication between the physical and computational substructures. Several single and double free actuators, i.e. not attached to specimens, were used to verify the new system communication loop. Moreover, a repaired reinforced concrete bridge column-bent cap-box girder subassemblage was used to conduct a large-scale bidirectional HS trial tests using the developed system for further verification. The verification tests are discussed herein where the new communication scheme and the entire HSS were shown to be accurate.

KEYWORDS: *Hybrid Simulation, Large-Scale Testing, Reinforced Concrete Bridges*

1. INRODUCTION

Hybrid Simulation (HS) was first introduced by Takanashi et al. (1975), who referred to the method as “online test”. The essence of HS is to use an online computational substructure to update the earthquake input signal at each time step based on the force feedback from the physical substructure. In the last three decades, there were significant development efforts in different HS areas that included, but not limited to, development of suitable integration methods [e.g. Nakashima and Masaoka, 1999; Magonette, 2001], study of the effect of experimental errors [e.g. Mosqueda, 2003 and Elkhoraibi and Mosalam, 2007], and real-time HS [e.g. Mosalam and Günay, 2014]. Large portion of the previous studies that involved HS focused more on the development side and robustness of the testing method. However, several studies utilized HS directly in different applications [e.g. Mosalam et al., 1998; Terzic and Stojadinovic, 2013].

The main objective of this study was to develop a practical HSS that utilizes readily available laboratory DAQ systems along with inexpensive TCP/IP-Ethernet connections to establish the communication between the physical and computational substructures. The HSS was primarily developed to test a ¼-scale retrofitted bridge subassemblage and compare its behavior to an as-built identical specimen tested under cyclic loading as discussed in details in [Moustafa and Mosalam, 2015]. However, the first specimen that was tested under cyclic loading only was repaired and used in a HS trial test, as discussed in this paper, to verify the developed HSS before using it for testing the second specimen. The main development in this study was the Pacific Instruments (PI) DAQ interface to communicate with OpenFresco (2006) from the computational side, through the inexpensive Ethernet connection to replace expensive shared memory communication cards such as SCRAMNet, and a Digital Signal Processing (DSP) card from the experimental side to control the laboratory hardware and receive the physical substructure feedback. Another development was implementing a new test setup component in OpenFresco to perform geometric transformations between the computational model global degrees-of-freedom (DOFs) and the actuators local DOFs for the command displacements and force feedbacks.

2. HYBRID SIMULATION SYSTEM AND DEVELOPMENTS

To perform a HS test, several key components of software and hardware are necessary. The components of the specific HSS utilized in this study and the new developments in the system are discussed in this section.

2.1 HS Components

Four main components comprise a typical HSS. The first is a discrete model of the structure to be computationally analyzed under any static and the dynamic loading. The finite element (FE) method is used to discretize the problem spatially and a time-stepping integration algorithm is used for the solution of the equations of motion with time discretization. The second required component is a transfer system consisting of a controller and static or dynamic actuators, so that the incremental response (generally the displacements) determined by the time-stepping integration algorithm can be applied to the physical substructures. For slow tests such as the ones conducted in this study, quasi-static testing equipment can be used. The third major component of the HSS is the physical specimen that is being tested in the laboratory and a support system (e.g. reaction wall or frame and a strong floor) against which the actuators of the transfer system can react. The fourth component is a DAQ system including displacement transducers and load cells. This data acquisition system in this study is responsible for measuring the response of the test specimen and returning the resisting forces to the time-stepping integration algorithm to advance the solution to the next analysis step. A vital feature of HS is to connect the above-mentioned four components together to achieve reliable two-way communication for sending the displacement input and receiving the force feedback. The major components and their connectivity of the utilized HSS at the Structures Laboratory of the University of California, Berkeley are shown in Fig. 2.1. The main components identified in this figure are: (a) A computational platform where the numerical integration of the governing equations of motion is performed (OpenSees (2000) was used in this case), (b) OpenFresco (2008) generic middleware that communicates with the computational platform, (c) New interface software developed within the PI DAQ that communicates, in turn, with OpenFresco through TCP/IP connection, (d) DSP card that further complements the communication loop with the laboratory hardware, and (e) digital controllers that command the hydraulic actuators in displacement control.

Two multi-DOF computational models were considered in this study. The first model was a generic one with many DOFs that was used only for the verification tests that used the free actuators without any attached specimen. A multi-story multi-bay frame was used in this computational model where one of the first story columns was replaced by the experimental element. A simulation experimental element, available in OpenFresco and based on input material and geometric properties, was used instead of an actual experimental physical substructure. In this case, a multiplier (assumed stiffness) of the displacement of the free actuators in the HS verification tests was used as a virtual force feedback to the hybrid system to check the communication loop against the pure simulation results. The second computational model was used in the bridge subassemblage HS trial test and consisted of multi-DOF column with lumped mass at the top and defined damping ratio. The lumped mass was calibrated to reflect a representative segment of the prototype bridge used in the study (Moustafa and Mosalam, 2015). Damping was modeled as Rayleigh damping with coefficients determined using 5% damping ratio for periods corresponding to the transverse and longitudinal modes of vibration. The physical substructure used with the second computational model was a ¼-scale reinforced concrete (RC) column-bent cap-box girder bridge subassemblage. Moreover, the HSS verification tests used different ground motions with more cycles and harmonic nature, such as the El Centro record, and pulse-like nature, such as the Rinaldi record.

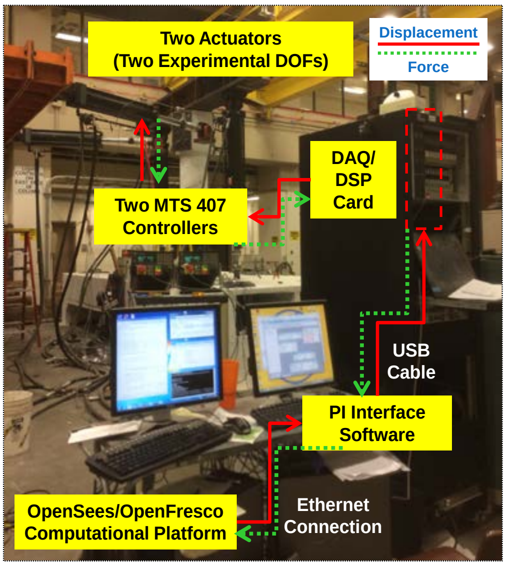


Figure 2.1 Overview of the main components and connectivity of the developed HSS

OpenSees (2000) was used as the FE software to analyze the computational substructure and solve the dynamic equation of motion for the displacement at each time step. Moreover, OpenSees was used along with the middleware OpenFresco (Schellenberg et al., 2006) to connect the FE model with the control and data acquisition software. OpenFresco is designed in an object-oriented structure that is similar to OpenSees and shares common classes, e.g. for element types and numerical integration methods. Therefore, OpenFresco is most conveniently used with OpenSees as the FE platform and a single OpenSees/OpenFresco input file (prepared using the Tool Command Language, TCL) to define the computational model and the communication settings.

To properly connect all the HSS components, a robust communication loop is indispensable. In general, the readily available OpenFresco software comprised the main part of the necessary middleware needed for connecting the FE software and the experimental control and DAQ systems. However, OpenFresco lacked the needed experimental setup to perform the specific geometric transformations between the global DOF and the local DOF of the lateral actuators per the required setup for the bridge subassemblage test (Moustafa and Mosalam, 2015). Thus, implementing a new experimental setup object in OpenFresco was the first development to achieve the sought HSS. On the other hand, to avoid using expensive shared-memory network cards, such as SCRAMNet, to communicate with the controllers, a practical use of the inexpensive TCP/IP Ethernet connection was another objective sough in this HSS. Although a generic TCP control is available in OpenFresco, an interface that utilizes such TCP connection to communicate the commands to the controllers was required. The PI DAQ software was modified to encompass a new module that can integrate the networking capabilities of the DAQ console along with the programmable DSP card to achieve the desired PI interface as the second development achieved in this study. More details about these two developments are presented in the following two subsections.

2.2 Development I: OpenFresco New Experimental Setup

The *ExperimentalSetup* is one of four main classes in OpenFresco. The transformation of the prescribed boundary conditions from the local or basic element DOF of the experimental elements into the actuator DOF of the transfer system is the first core task of the *ExperimentalSetup* class. Similarly, the transformation of the work conjugates measured by transducers and load cells back to the experimental element DOF is the second core task of the *ExperimentalSetup* class (Schellenberg et al., 2006). For the HS tests considered in this study, the two horizontal actuators used for applying the lateral load were arranged in a planer triangular configuration. A new *ExperimentalSetup* object was required in OpenFresco to perform the geometric transformation between the two model (global) DOFs, designated as *x* and *y*, and the two actuators (local) DOFs, designated as 1 and 2, as shown in Fig. 2.2. The sough transformation is applied to the computed displacements such that displacement command readily in each actuator DOF is delivered to the corresponding controller. Similarly, the received force feedback in each actuator DOF is transformed to the *x* and *y* DOFs before passing it to the FE software to proceed with the next time step calculations. The “TriangularActautors” object was successfully developed and implemented in an updated version of OpenFresco. The TCL syntax input for the new experimental setup is as follows:

expSetup TriangularActuators *$tag* -control *$ExpControltag $A1 $A2 $B1 $B2 $C1 $C2*

where *$ExpControltag* is the defined tag for the used experimental control object, which is the GenericTCP in this case, and *$A1*, *$A2*, *$B1*, *$B2*, *$C1*, and *$C2* are geometric input parameters that describe the relative locations of the two actuators as identified in Fig. 2.2.

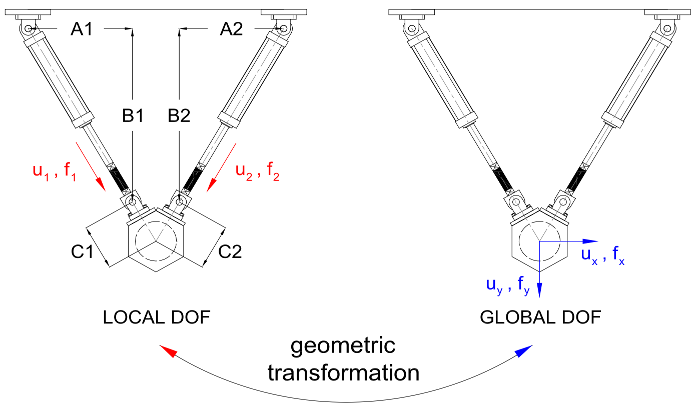


Figure 2.2 Input displacement and measured force feedback geometric transformation between the model global DOFs and the actuators local DOFs

2.3 Development II: New PI Interface

An important development achieved in this study is a practical interface between OpenFresco and the controllers. This interface is built into the PI DAQ system. The final interface consists of two parts: (1) Microsoft Windows application customized from the PI DAQ software, and (2) PI test file containing specific PI6042 DSP routines.

Microsoft Windows Custom Application: The application *PI660C UCB HybridSim* interface is a heavily modified version of the original *PI660C DAQ* program. The modifications include the addition of a TCP communications interface, an OpenFresco command interpreter, and a raw data format handler and translator. The main purpose of this PI interface is to exchange displacement and force vectors up to 5 DOFs with OpenFresco over an Ethernet TCP/IP connection. Thus, this application is responsible of receiving the displacement vector from OpenFresco and passing it through the DSP routines to the controllers. In addition, it receives the force feedbacks from allocated memory locations and send them back to OpenFresco. All the operations performed through this part of the interface utilize data in the actuators DOF. The geometric transformation to the global DOFs to solve the equations of motion takes place in OpenFresco through the new *ExperimentalSetup* class, as previously discussed. A screen shot of the *PI660C UCB HybridSim* Microsoft Windows application is shown in Fig. 2.3a. The figure shows the implemented module that handles the HS mode and sets its parameters. A set of parameters that can be assigned beforehand are shown in Fig. 2.3b. All the parameters are considered input for the DSP routines that are called through the PI application. Because the interface can exchange data from up to 5 DOFs, a span definition is required for each of these 5 DOFs for control purposes. The rate of loading, defined in terms of the maximum velocity, is one of the parameters input shown in Fig. 2.3b. A maximum velocity is defined instead of a constant one because based on the number of controlled DOFs, one actuator might have to slow its velocity to match other actuators motion. Finally, two additional options that are still under development, but were not needed for the tests conducted in this study, are the super pipeline mode and the pipeline predict. These modes aim at minimizing the communication delays for the prospect of real-time HS applications.

PI6042 DSP routines: The DSP routines are responsible for the low-level, high-priority, and time-sensitive tasks. The main purpose of these routines is the motion interpolation and data generation tasks. They are also responsible for data acquisition hardware handling, e.g. sending and receiving analog signals via the USB data link interface to and from the computer where the new PI interface is running. The DSP program is uploaded via a USB link from the control computer to the PI6042 DSP cards residing in the PI6000 chassis and executed once per data acquisition scan. The PI6042 DSP routines were coded in a Reversed Polish Notation (RPN), close variant to the Assembly Language. The RPN routines are executed once per data acquisition scan cycle at the requested sampling rate. For proper operation, the sampling rate required to define the actuators path velocity was set to 10 msec for the HS trials and tests conducted in this study. The RPN routines, called by the *PI660C UCB HybridSim* interface, are executed on the PI6042 DSP card sequentially at every data acquisition scan. One of the main functions of these routines is to interpolate the final end-displacement at a given time step, as received from OpenFresco via the new PI interface, and deliver the interpolated calculated signal to the MTS 407 controllers. The physical connection for the interpolated signal transfer to the controllers is a standard BNC to BNC cable where one end is connected to the DSP card in the PI chassis, and the other end is connected to the controller.

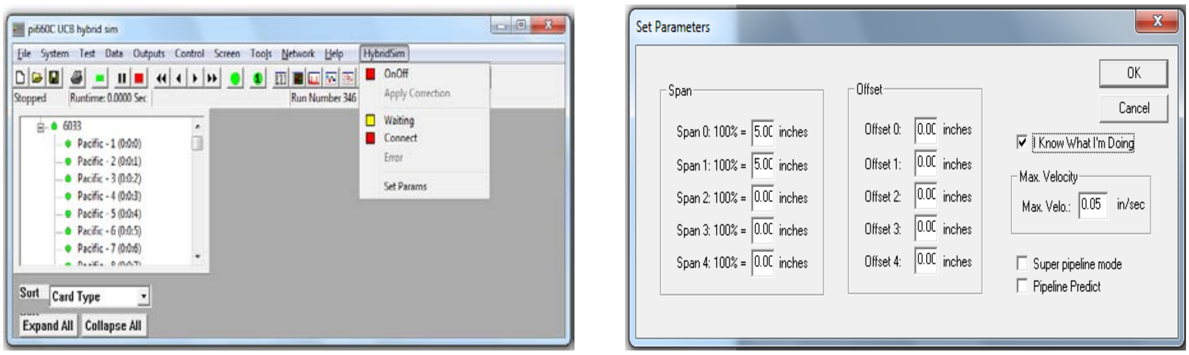


Figure 2.3 Screen shots of the new PI interface with HybridSim module (left) and HS parameters (right)

3. HSS VERIFICATIONS

To confirm the performance of the implemented developments and validate the HSS for testing, several trials and verification tests were conducted. The verification tests started with network protocol analysis, then utilized single and double free actuators, i.e. not attached to any physical specimens. Moreover, a full HS trial test that utilized a repaired bridge subassemblage to validate the whole HSS was conducted.

3.1 TCP/IP Network Stack

Numerous performance and characterization tests were performed on the TCP/IP performance between the OpenFresco/OpenSees platform and the new *PI660C UCB HybridSim* interface. These characterization tests were performed directly using the Wireshark network protocol analyzer program, which attaches directly to the network software stack and records all the Ethernet packets traversing the Ethernet interface, commonly referred to as “sniffing”. By looking at the timestamps and decoding the packet payloads, the traffic flow and timing were understood. The Ethernet TCP/IP network transactions flowing through a preliminary established connection between the OpenFresco platform and the new PI interface was analyzed. The timing data from the Ethernet transactions first indicated a latency of approximately 216 msec. In order to reduce latency, the transmit buffer of OpenFresco was resized to be an integer multiple of the payload size of the Ethernet frame, i.e. the OpenFresco variable *OF_Network_dataSize* was modified from 256 to 365 such that on every network transaction, two totally filled Ethernet frames were utilized. Adjusting the OpenFresco packet size reduced the latency to 70 msec. Due to the slow loading rate of the test in this study, 70 msec latency was insignificant to alter the HS communication.

3.2 Single Actuator Tests

A single actuator was used along with a large multi-DOF computational model to test the communication loop between all components of the HSS. Two-way communication is necessary in HS; one way is for sending the displacement command and the other is for receiving the force feedback. A free actuator that is not attached to any specimen will report zero force feedback or only the load cell noise. Thus, for the free actuator trials, a multiplier (stiffness) of the displacement command was fed back to the DSP and DAQ as a virtual force feedback. The constant multiplier reflected the stiffness of a hypothetical linear force-displacement relationship. The advantage of this virtual feedback is to compare with pure simulation results where an elastic element with a constant stiffness replaces the actuator displacement/force feedback virtual experimental element. A multiplier of 2 was chosen for the displacement feedback to the DSP to reflect a hypothetical elastic element with 2 kip/in. stiffness, which was comparable to other columns in the hybrid model to compare with the pure simulation case.

The obtained displacement and force histories from the pure simulation were compared to those from the HS recorded data at both OpenSees and the PI DAQ in Fig. 3.1. OpenSees recorded the displacements obtained from the solution of the equation of motion at each time step along with the discrete force feedback at solution time steps only when received through the new PI interface. Meanwhile, the PI DAQ recorded the actual command data, i.e. actuator motion, and its multiplier when received at the DSP card. In addition, the force-displacement relationships are plotted for all cases as shown in Fig. 3.2 to demonstrate the constant stiffness used for the model and the case with the hypothetical feedback. The comparison shows the perfect match between the simulation and the HS tests. Despite the perfect match in the displacement amplitudes, the progression with time was different from the actual actuator motion and the OpenSees command, or pure simulation case. This is expected and attributed to the constant velocity or rate of loading used for commanding the actuator. The DSP routines were used to interpolate the received displacement command and apply it smoothly to the controller to pass it to the actuator with a constant velocity. The linear actuator displacements shown in Fig. 3.2 reflect a constant slope, i.e. constant actuator velocity as required. Therefore, the good comparison between the HS tests that used a hypothetical feedback of a constant multiplier applied to the actual command, and the pure simulation provided confidence on the accuracy of the communication loop among the different HSS components.

3.3 Double Actuators Tests

Similar trial tests were conducted using two free actuators setup with the actual computational model for the bridge subassemblage test specimen, i.e. utilizing the newly implemented geometric transformation setup (the *TriangularActuators ExperimentalSetup* class) in the OpenFresco/OpenSees input file. The same concept of feeding back a hypothetical force that is 2 times the actual displacement command was used in these trials. These tests aimed at verifying the correctness of the newly implemented geometric transformation and the DSP routines in interpolating the displacement command for two actuators simultaneously. To verify the geometric transformation, the input of the OpenFresco “*TriangularActuator*” command was set up in a way that rendered each of the actuators inclined with a 45^{o} angle, e.g. *A1* and *B1* identified in Fig. 2.2 were set to similar values. In this geometry, if a global transverse-only motion (u_x in Fig. 2.2) is required, the two actuators should have identical input along the local DOFs. On the other hand, if a longitudinal-only motion (u_y in Fig. 2.2) is required, the two actuators should have same magnitude but opposite direction local DOFs input. This anticipated geometric transformation was accurately verified from the two actuators displacement history shown in Fig. 3.3

for the transverse-only and longitudinal-only motions. Fig. 3.3a shows an identical linear actuator displacement motion of the two actuators in the transverse-only case, which indicates a constant velocity and verifies the capability of the DSP routines to interpolate the command for two actuators simultaneously. Fig. 3.3b shows that the two actuators had similar input along the local DOFs but with opposite direction (sign). This implies that the two components of the actuators motion in the transverse direction cancelled the effect of each other and forced the actuators along a longitudinal path as intended. Moreover, Fig. 3.3 compares the OpenSees displacement command for the two actuators with the actual PI DAQ recorded data. The comparable peak values confirm that the computed displacements form OpenSees were successfully achieved by the actuators. One final trial test used the two free actuators to conduct a generic bidirectional test to further verify the DSP interpolation routines rather than the geometric transformation but is not shown here for brevity. However, full details of the HSS verification tests are available in Moustafa (2014).

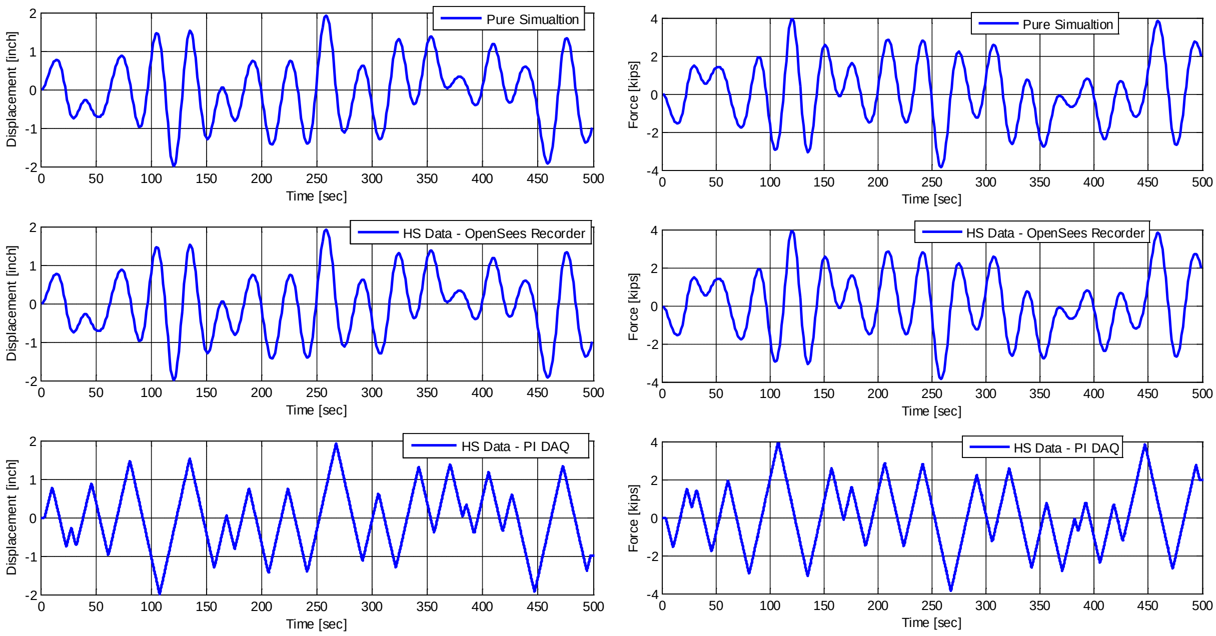


Figure 3.1 Comparison of the displacement (left) and force (right) history obtained from the pure simulation, the computed OpenSees command for HS, and the actual actuators motion from the single free actuator tests

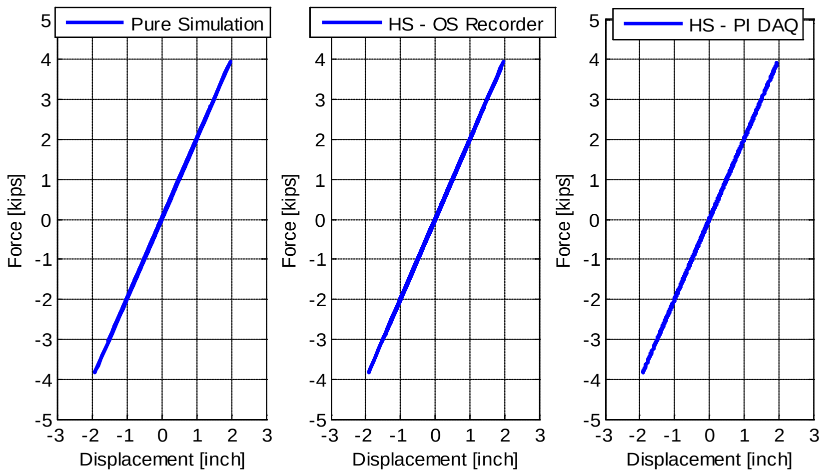
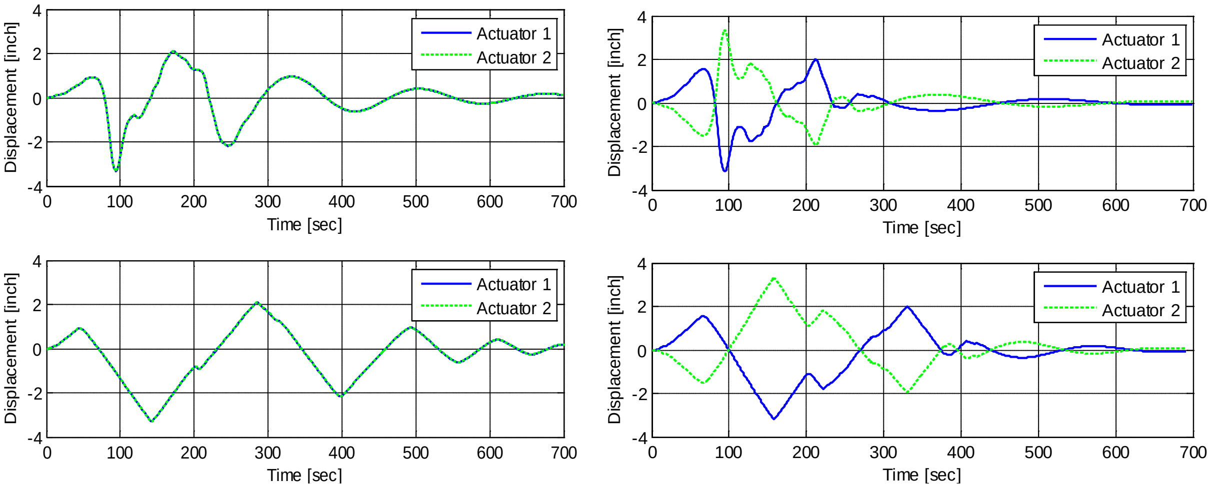


Figure 3.2 Force-displacement relationships from pure simulation and HS test data recorded at OpenSees (OS) and using the PI DAQ from HS tests using single free actuator

4. LARGE-SCALE BRIDGE SUBASSEMBLY HS TEST

To complete the validation of the integrated HSS and its new developments, a full specimen HS test was conducted using a repaired bridge subassemblage that was tested under quasi-static cyclic loads in another study

(Moustafa and Mosalam, 2015). The benefit of the full specimen HS tests was to validate the whole system using a true specimen with inelastic nonlinear behavior. The test setup and physical specimen (substructure) are shown in Fig. 4.1. Several HS trials with and without gravity load, and using different ground motion scales and components were conducted. Only sample results from an 80% bidirectional test that used Rinaldi ground motion and a gravity load of 10% of the column axial capacity is shown here and full details are in Moustafa (2014). To perform a final check of the newly implemented OpenFresco geometric transformation, the directly measured displacement response in the global *x* and *y* directions using wirepots from the test were compared to the intended OpenSees commands. The actuators load cells measured forces were transformed to the global *x* and *y* directions and were compared to the forces used by OpenSees to solve the equations of motion, i.e. the intended global computed displacement commands and received force feedbacks were compared against the actual test results. Fig. 4.2 summarizes the comparison through showing the overall force-displacement relationships from OpenSees versus that where the global force calculated from the local load cell measurements along with the actual displacements from the wirepots for both transverse and longitudinal directions. The good match of the nonlinear hysteretic force-displacement relationships in the global DOFs of the calculated and actual measured response validated the developed HSS in terms of the communication loop and geometric transformation.



(a) transverse-only (b) longitudinal-only

Figure 3.3 Actuators displacement history from the HS computed OpenSees signal (top) and actuators feedback from the DAQ (bottom)

Figure 4.1 Test setup used for conducting HS trial tests for the repaired SP1

5. CONCLUSIONS

Based on the verification tests from the free actuators and the full specimen HS test, it was concluded that the sought HSS for testing is reliable and performs as expected. In particular, the new *PI660C UCB HybridSim* application (interface) successfully communicates the displacement and force feedback vectors between OpenFresco and the experimental hardware. The associated DSP routines developed within the *PI660C UCB*

HybridSim successfully interpolated the commands for multi-actuators simultaneously, and communicated the DAQ actuators load cell measurements back to the PI interface. Finally, the newly implemented OpenFresco experimental setup object correctly performs displacement and force geometric transformation between the global DOFs and the two actuators in a triangular arrangement local DOFs.

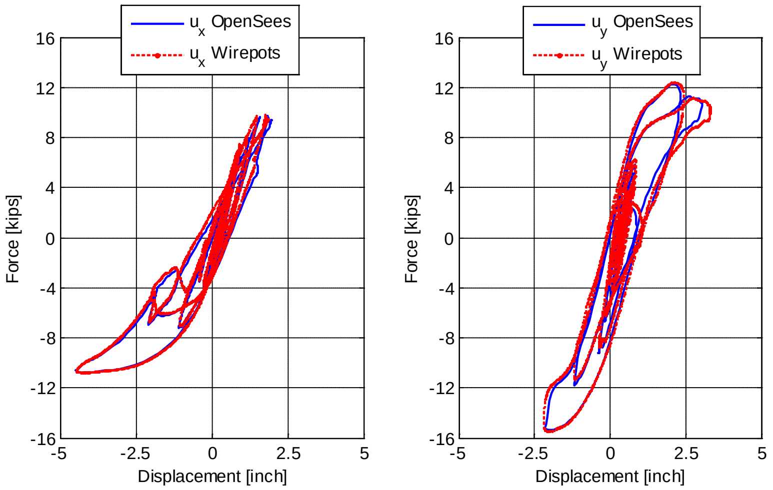


Figure 4.2 Comparison of force-displacement relationship in transverse (left) and longitudinal (right) directions from the recorded OpenSees data and actual load cells and wirepots DAQ data

AKCNOWLEDGEMENT

This work was supported in part by NSF EAGER award and by California Department of Transportation (Caltrans). Special thanks are due to Dr. Selim Günay and Mr. Philip Wong for their technical input, and the staff at the Structures Laboratory at UC Berkeley for their help throughout the setup and tests.

REFERENCES

1. Elkhoraibi, T., & K.M. Mosalam, (2007). Towards Error Free Hybrid Simulation Using Mixed Variables. Earthquake Engineering & Structural Dynamics, 36(11), 1497-1522.
2. Magonette, G. (2001). Development and application of large–scale continuous pseudo–dynamic testing techniques. Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences, 359 (1786), 1771-1799.
3. McKenna, F., G.L. Fenves, M.H. Scott, & B. Jeremic, (2000). Open System for Earthquake Engineering Simulation (OpenSees). PEER, University of California, Berkeley, CA.
4. Mosalam, K.M., R.N. White & G. Ayala (1998). Response of Infilled Frames Using Pseudo-Dynamic Experimentation. Earthquake Engineering & Structural Dynamics, 27, 589-608.
5. Mosalam, K. M., & S. Günay, (2014). Seismic performance evaluation of high voltage disconnect switches using real-time hybrid simulation: I. System development and validation. Earthquake Engineering & Structural Dynamics, 43(8), 1205-1222.
6. Mosqueda, G. (2003). Continuous hybrid simulation with geographically distributed substructures, Doctoral dissertation, University of California, Berkeley.
7. Moustafa, M.A. (2014). Structural Behavior of Bent Cap Beams in As-built and Retrofitted Reinforced Concrete Box-Girder Bridges. Ph.D. dissertation, University of California, Berkeley, Berkeley, CA.
8. Moustafa, M.A., & K.M. Mosalam, (2015). Seismic response of bent caps in as-built and retrofitted reinforced concrete box-girder bridges. Engineering Structures, 98, 59-73.
9. Nakashima, M., & Masaoka, N. (1999). Real-time on-line test for MDOF systems. Earthquake engineering & structural dynamics, 28(4), 393-420.
10. Schellenberg, A., H. Kim, Y. Takahashi, G. Fenves, & S. Mahin, (2006). OpenFresco Framework for Hybrid Simulation: Installation and Example Guide. PEER, University of California, Berkeley, CA.
11. Takanashi, K., Udagawa, K., Seki, M., Okada, T., & Tanaka, H. (1975). Nonlinear earthquake response analysis of structures by a computer-actuator on-line system. Bulletin of Earthquake Resistant Structure Research Center, 8, 1-17.
12. Terzic, V., & B. Stojadinovic, B. (2013). Hybrid simulation of bridge response to three-dimensional earthquake excitation followed by truck load. Journal of Structural Engineering, 140(8), A4014010.