

Development, implementation and verification of a user configurable platform for real-time hybrid simulation

Ali Ashasi-Sorkhabi^a and Oya Mercan^{*}

Department of Civil Engineering, University of Toronto, 35 St. George Street, Toronto, ON, M5S 1A4, Canada

(Received April 20, 2014, Revised July 20, 2014, Accepted August 5, 2014)

Abstract. This paper presents a user programmable computational/control platform developed to conduct real-time hybrid simulation (RTHS). The architecture of this platform is based on the integration of a real-time controller and a field programmable gate array (FPGA). This not only enables the user to apply user-defined control laws to control the experimental substructures, but also provides ample computational resources to run the integration algorithm and analytical substructure state determination in real-time. In this platform the need for SCRAMNet as the communication device between real-time and servo-control workstations has been eliminated which was a critical component in several former RTHS platforms. The accuracy of the servo-hydraulic actuator displacement control, where the control tasks get executed on the FPGA was verified using single-degree-of-freedom (SDOF) and 2 degrees-of-freedom (2DOF) experimental substructures. Finally, the functionality of the proposed system as a robust and reliable RTHS platform for performance evaluation of structural systems was validated by conducting real-time hybrid simulation of a three story nonlinear structure with SDOF and 2DOF experimental substructures. Also, tracking indicators were employed to assess the accuracy of the results.

Keywords: real-time hybrid simulation; performance evaluation; experimental substructure; analytical substructure; FPGA; phase error; amplitude error

1. Introduction

Large scale dynamic testing is vital to acquire the necessary system level behaviour to enable safer yet cost-effective designs of next generation structures. Real-time hybrid simulation (RTHS) combines physical testing of critical components with computational model of the remaining structure in real-time; and thus offers an economical and practical way to address this need.

Hybrid simulation has started as an extension to pseudodynamic (PSD) testing method (Takanashi *et al.* 1975), where the test is carried out at a very slow loading rate. During a PSD test, an integration algorithm is used to directly solve the equations of motion (in Eq. (1)) to obtain the command displacements; which are then applied to the test structure by hydraulic actuators. The measured restoring forces from the deformed structure are fed back and used by the integration algorithm in the generation of the subsequent command displacements (Mahin and Shing 1985).

The equations of motion that are solved by the integration algorithm during a PSD test are

^{*}Corresponding author, Ph.D., E-mail: oya.mercan@utoronto.ca

^a Ph.D. Candidate, E-mail: ali.ashasisorkhabi@mail.utoronto.ca

given by

$$M\ddot{x}(t) + C\dot{x}(t) + R(x, \dot{x}, \ddot{x}, t) = F(t) \quad (1)$$

where M is the mass matrix, C is the damping matrix (modeling the inherent viscous damping), R is the restoring force vector, F is the vector of externally applied or effective forces, and \dot{x} and \ddot{x} are the velocity and acceleration vectors, respectively. When the structure is subjected to ground acceleration \ddot{x}_g , the external (i.e., effective) force vector is $F(t) = -M\Gamma\ddot{x}_g(t)$, where Γ is the mass influence vector.

In hybrid simulation, the test structure is divided into two parts: the components of the test structure for which a reliable analytical model is not available are isolated and tested physically in the laboratory, while the rest of the system is modeled analytically in a computer (Dermitzakis and Mahin 1985). These are known as experimental substructure and analytical substructure, respectively. When the experimental substructure has load-rate dependent vibration characteristics, the hybrid simulation needs to be carried out in real-time (Nakashima *et al.* 1992, Horiuchi *et al.* 1999, Mercan and Ricles 2009). This requires efficient and robust computational resources as well as a well-synchronized data communication platform (Mercan and Ricles 2009). RTHS became an important tool to experimentally capture the rate-dependent vibration characteristics of complex structural systems and example applications of which can be seen in the recent literature (Christenson *et al.* 2008, Carrion *et al.* 2009, Karavasilis *et al.* 2011, Wu *et al.* 2013, Chae *et al.* 2013b). In RTHS, as only the critical components of the test structure need to be constructed and tested physically and remaining parts are modeled analytically, a wide range of influential parameters and loading cases could be considered in a timely and cost-effective manner (Ashasi-Sorkhabi *et al.* 2013). Since the accuracy of the real-time simulation depends on the accuracy of the actuator control, the design of improved control laws remains an active and challenging interdisciplinary research area. Current direction of research to improve the RTHS results include robust (Gao *et al.* 2013), adaptive control by inverse compensation of the system dynamics (Chen and Ricles 2010, Chen *et al.* 2012, Liu *et al.* 2013, Chae *et al.* 2013a, Chen and Tsai 2013 and model based control (Phillips and Spencer 2011). This recent focus on the servo-hydraulic actuator control is well placed on the way to enhance RTHS as an emerging testing technology.

The implementation of RTHS involves challenges in accurate control of experimental substructure and synchronization of the command and measured signals during the test to assure the accuracy and stability of the real-time hybrid simulation (Mercan *et al.* 2007, Mercan and Ricles 2008). In a real-time PSD or hybrid simulation the restoring force feedback to the integration algorithm may contain amplitude error and/or time delay which are due to control errors in tracking the command displacement within the specified time to accommodate a real-time test, dynamic behaviour of the servo-hydraulic system, and any latency caused by the digital controller in receiving and executing commands (Mercan and Ricles 2007). Several researchers in real-time testing area have noticed that time delay in the measured restoring force has detrimental effects on the outer loop dynamics. Wallace *et al.* (2005) determined that when the time delay in the restoring force in the system exceeds a critical value, that system becomes unstable. Wu *et al.* (2006) investigated the stability of an SDOF system in presence of actuator delay during a real-time test. Horiuchi *et al.* (1999) studied the effects of actuator delay in a real-time test, by means of energy balance approach and showed that a delay by an actuator causes increase in energy content of the system. The increase in energy was shown to be the same as that caused by a

negative damping; and the system becomes unstable when the negative damping exceeds the inherent damping. Another challenge in the implementation of the RTHS is the existence of random errors in restoring force feedback resulting from electrical noise that is usually inevitable in testing systems. Since the restoring force feedback is used in command generation, RTHS suffers from propagation of the errors influencing the accuracy and in some cases the stability of the test results. Ashasi-Sorkhabi and Mercan (2013) presented the effects of the electrical random noise in the restoring force feedback on the RTHS results.

Several delay compensation methods have been proposed to minimize the effect of actuator delay in RTHS results. Horiuchi *et al.* (1999) developed a method based on forward extrapolation of the command displacement to compensate for the actuator delay. In 2001, Horiuchi and Konno proposed another compensation method that was based on an assumption of linear acceleration of the structure. Carrion and Spencer (2006) implemented actuator time delay/lag compensation through model based response prediction. Some of other methods used to compensate for the actuator delay comprise the Smith Predictor (Shao *et al.* 2006), Inverse compensation method (Chen 2007) and the virtual coupling method (Christenson *et al.* 2008). There are other delay compensation methods introduced to RTHS such as lead compensator (Zhao *et al.* 2003) and velocity feedforward (Jung and Shing 2006, Mercan and Ricles 2009) in which control theory techniques are employed to minimize the control delay.

This paper introduces a new user-configurable computational/control platform that is recently developed at the University Toronto to carry out RTHS. Using the field programmable gate array (FPGA) technology, this platform not only enables the user to implement user-defined control laws to control the experimental substructures, but also provides ample computational resources to handle the integration algorithm and analytical substructure state determination algorithms in real-time. As will be explained in detail in the following sections, both the real-time controller and the FPGA reside on the same chassis and get synchronized through the centralized software, developed in LabVIEW. As such, unlike most of the previously developed RTHS platforms, there is no more need for SCRAMNet, (Shared Common Random Memory Access Network) in this new configuration. SCRAMNet is a fiber optic communication device that shares memory and enables time synchronization between the components of RTHS platforms. It has been used in most of the former RTHS platforms such as the NEES real-time Multi-directional experimental laboratory at the ATLSS engineering center, Lehigh university (Mercan and Ricles 2007) and the real-time hybrid system at University of Buffalo's Structural Engineering and Earthquake Simulation Laboratory (SEESL) (Reinhorn *et al.* 2004.). In both cases the RTHS platform was comprised of a real-time controller and a real-time simulation workstation (xPC kernel). The RTHS platform at the Newmark Structural Engineering Laboratory (NSEL), University of Illinois at Urbana-Champaign employs a D Space board, DSP card, and a host computer with fiber optic communication in between (Phillips and Spencer 2011). Recently, Nakata (2011) developed an earthquake simulator at the John Hopkins University, where the need for SCRAMNet was eliminated through the use of dual core NI controllers.

By eliminating the SCRAMNet the complexity of the required programming to enable data exchange decreases; the synchronization and timing among the servo-control loop, real-time simulation loop and data acquisition units become more straightforward. Also, a significant reduction in the cost of the platform may be obtained. As the user-configurable computational control platform introduced here is completely transparent to the user, it will enable easy implementation and verification of advanced control techniques and will help enhance RTHS as an emerging testing technology.

In the following two sections the details of the hardware/software development will be provided. Then, the accuracy of the servo-hydraulic actuator displacement control, where the control tasks get executed on the FPGA, will be verified. This verification includes an SDOF experimental substructure, a 2DOF uncoupled and a 2DOF coupled experimental substructures. Additionally, to validate the RTHS capabilities of the platform, RTHS results of a three story nonlinear analytical substructure coupled with SDOF and 2DOF experimental substructures will be provided, respectively. In all these cases the experimental substructure remains linear elastic. As such, the accuracy of the RTHS results will be assessed by comparing the RTHS results with the "true solution" that is obtained through time history analysis of the same structure. To assess the tracking performance of the controller in the inner loop and also functionality of the outer loop as a reliable real-time hybrid simulator Phase and Amplitude Error Indices (PAEI) developed by Hessabi and Mercan (2012) will be used. These indicators can quantify the phase and amplitude errors between any given two signals, and thus will be used here to assess the accuracy of the RTHS results.

2. Real-time hybrid simulation hardware

2.1 Mechanical test setup

An overview of the RTHS test setup that was used in this study is shown in Fig. 1. This test setup includes a hydraulic service manifold (HSM), rated at 120 gpm continuous flow, that regulates the supply oil pressure at 210 bar (3000 psi), two fatigue rated hydraulic actuator each with stroke length of ± 127 mm (± 5 inch) and maximum force capacity of 33 kN ($\pm 7,500$ lbf) driven by an electro servo-valve with flow capacity of 16.5 gpm. A built-in ± 127 mm (± 5 inch) AC LVDT and a dynamic load cell with a capacity of 50 kN ($\pm 12,500$ lbf) are used to obtain the displacement and force feedbacks from each experimental substructure.

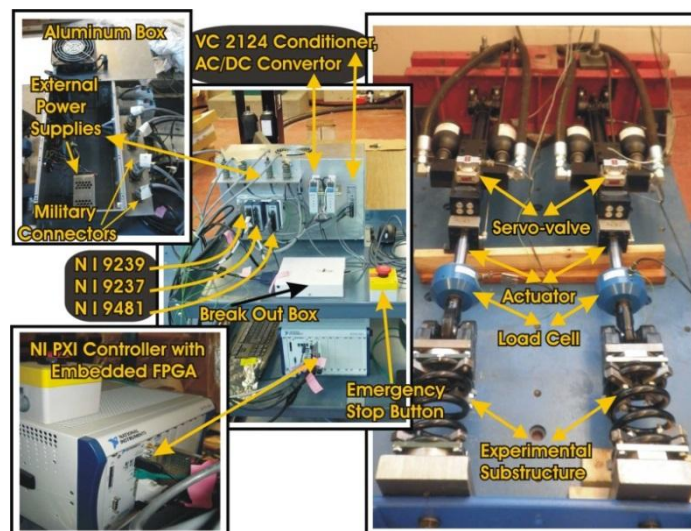


Fig. 1 The components of the real-time testing system

2.2 Computational/control platform

An NI controller with an onboard 2.26 GHz quad-core processor (here referred to as real-time controller) and an FPGA, both residing on an NI PXI chassis are the key components that form the RTHS computational/control platform. When programmed properly, the FPGA is a hardware that can conduct flexible I/O (input/output) operations as well as on-board processing. The architecture of the designed platform and all related signal routings are shown in Fig. 2.

As shown in Fig. 2 the real-time hybrid simulation process includes two nested operational loops. In the outer loop, all the tasks related to time integration, analytical substructure state determination, and input/output file manipulations are carried out by the real-time controller. During a test for a given time interval (step), the command displacements are calculated by the numerical integration algorithm and then issued to the servo-hydraulic system in the inner loop, which are then imposed to the experimental substructure. The measured restoring forces from the deformed experimental substructure are fed back to the integration algorithm for next step command generation. In the current setup the execution rate of the outer (real-time) loop is set to 1 kHz (i.e., integration time step = 0.001 sec) and can be increased up to 5 kHz.

In processor based platforms, multiple abstraction layers are included to assist with task scheduling and to allocate available resources among multiple processes. Hardware resources are controlled by the driver layer while the memory and the processor's bandwidth are handled by the operating system. Only one instruction per each processor core could be executed at a time. If programmed considering proper priority hierarchy, processor based real-time operating systems can decrease jitter to a finite maximum. Unlike the processor based systems, the facts that there is no operating system utilized in FPGAs and that all the logics are compiled to a physical hardware (FPGA), minimize the reliability concerns and as a result true parallel execution with deterministic hardware assigned to every task could be obtained (i.e., high reliability and high determinism). In addition, in an FPGA parallel pieces of code are implemented as parallel circuits in the hardware and therefore, the number of the code pieces that could be executed concurrently is not limited by the processor cores (i.e., true parallelism). Lastly, because the FPGA code is executed directly on the FPGA hardware without incorporating any overarching operating system, any code that is downloaded to the flash memory of the FPGA, starts running the instant that the PXI chassis is electrically powered (i.e., instant boot up). (Compact Rio Developers Guide 2009). With all the advantages listed, in the RTHS platform developed in this study, the inner loop tasks are executed on FPGA. Currently, the FPGA is configured to an execution rate of 25 kHz. In other words, for each loop-iteration of the outer loop there are 25 iterations of the inner loop. The inner and outer loops exchange data through direct memory access (DMA) option at the outer loop's execution rate (i.e., 1 kHz).

The controller output signal is issued to the servo-valve through the SCB-68 also known as the breakout box. The break-out box is a shielded connector block that has 68 screw terminals for I/O signal conditioning and is able to issue only voltage signals within $\pm 10\text{V}$. This appears as a challenge in the hardware as only current signals within $\pm 40\text{ mA}$ can drive the servo-valves. To resolve this issue, a DC voltage to current converter (VC 2124) is introduced to the circuit between the breakout box and the servo-valve which can transform voltage to current linearly within the required range when configured properly. The FPGA issues signals to the HSM as well as it acquires data from the displacement and force transducers through NI 9481, NI 9239, and NI 9237 modules, respectively.

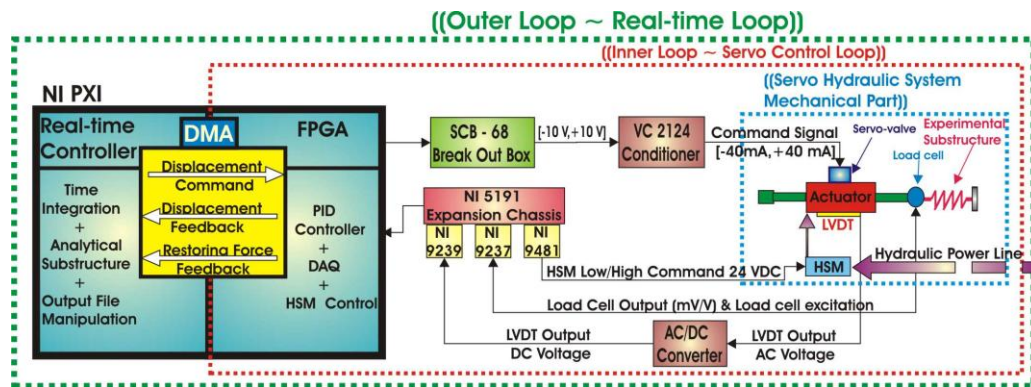


Fig. 2 RTHS facility integrated control system architecture

The NI 9481, the relay module, with a 10-terminal detachable screw terminal connector provides the connection for four electromechanical relay channels (NI 9481 Operating Instructions and Specifications 2008) to safely operate HSM low/high pressure states. NI 9239, the LVDT module, is a 4-channel, 24 bit analog input module that reads the LVDT output voltage at 50 kS/s (kilo sample per second) sampling rate per channel and then performs the analog to digital conversion internally (NI 9239 User Guide and Specifications 2007). NI 9239 is configured to receive DC voltage signals; however the built-in LVDTs of the actuators output AC voltage signals. To deal with this compatibility issue, as shown in Figs. 1 and 2, a linear AC/DC adapter is introduced. NI 9237, the simultaneous bridge module, with 4-channel, $\pm 25\text{mV/V}$ analog inputs and 24-Bit resolution. It includes all the signal conditioning required to energize and measure up to four bridge-based sensors concurrently at 50 kS/s of sampling rate for each channel (NI 9237 User Guide and Specifications 2007). All of these three modules reside on an NI 5191 expansion chassis that is directly connected to the FPGA.

There are also external power supplies necessary to energize the HSM solenoids, displacement transducers and signal conditioners. To ensure the safety of the personnel as well as the equipment, all of these external power supplies are secured inside an aluminium case with a cooling fan (see Fig. 1). The wiring from the I/O modules with screw-terminals is strain relieved using military connectors secured on the side of the aluminium case. As the military connectors are much stronger than the wire terminals, the probability of having catastrophic failures as a result of accidental tripping over of wires in the laboratory is reduced. Also, for emergency situations, an E-Stop button has been wired to the solenoids of the HSM.

3. Real-time hybrid simulation software development

3.1 Introduction to Lab VIEW

LabVIEW (short for Laboratory Virtual Instrumentation Engineering Workshop) is a system design platform that provides visual programming environment for code development from National Instruments. LabVIEW could be used for control and measurement applications where all

required logic and conditions are implemented using graphical icons connected by wires (LabVIEW Tutorial Manual 1996).

LabVIEW programs/subroutines are called virtual instruments (VIs), due to their appearance and operation similar to physical instruments. Each LabVIEW VI has two components: a front panel and a block diagram. The front panel consists of controls and indicators. Controls are the input nodes where the user can provide the necessary information to the system. Indicators are the output nodes of the VI through which the results are displayed. The block diagram (also known as back panel) includes the graphical source code. All the objects on the front panel are represented on the block diagram as terminals connected by wires based on the required data flow. There are also various execution structures (e.g. timed loops, For loops, While loops, etc.) and functions available.

3.2 Software development for RTHS

Unlike a turn-key controller, the controller in the current platform must be configured by the user to run all the required tasks. Thus, in addition to the servo control tasks, several other tasks must be taken in to account to ensure safe start-up, satisfactory performance and safe/emergency shut-down of the system.

The main framework of the developed RTHS software is programmed in LabVIEW and includes a Host VI (also called as real-time VI), an FPGA VI, and several sub VIs (equivalent to sub functions) all coordinated by a LabVIEW project. However, the simulation core of the software that runs the time integration and analytical substructure state determination algorithms are programmed in MATLAB/Simulink. The Simulation Interface Toolkit (SIT) developed by National Instruments provides a link between LabVIEW and MATLAB/Simulink. The SIT toolbox is used to compile the aforementioned Simulink model files and generate the corresponding dynamic link library (DLL) files that are executable in LabVIEW environment. The File Transfer Protocol (FTP) is employed to transfer the DLL files to the real-time PXI over the Ethernet network. At each run of the simulation outer loop (real-time VI), using the restoring force and displacement measurements, the DLL file gets executed once and the new command displacements along with the state determination data of the analytical substructure become available.

The real-time and the FPGA VIs are designed to run specific tasks and are programmed to exchange data through the direct memory access (DMA). The DMA is the fastest available method to transfer large volumes of data between the FPGA (target) and the real-time (host) platforms. This is because in the DMA a separate piece of hardware is allocated to transfer data between the workstations and only one or two bus read/write cycles are needed per each packet of data transferred. Thus, the latency in data transfer could be minimized. Also, as DMA usually is intended to obtain maximum data transfer rate, it is beneficial for high speed data acquisition purposes (Harvey *et al.* 1991). In addition, DMA does not employ the real-time processor(s) during data transfer and thus, saves its resources to perform other calculations (LabVIEW Tutorial Manual 1996). In the current RTHS platform, DMA is utilized to send the displacement commands at each iteration of the integration algorithm from the real-time VI to the FPGA and also transfer the acquired force and displacement measurements back to the real-time VI to be used for next step command generation. The key-tasks, performed by each of real-time and FPGA VIs, are summarized in Fig. 3.

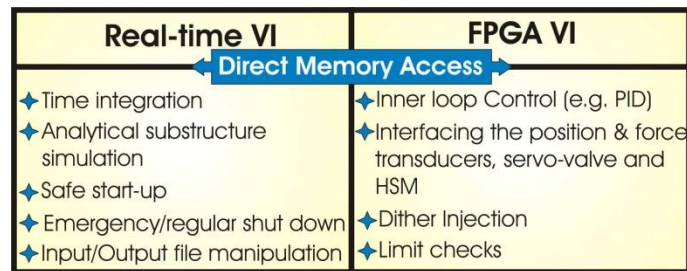


Fig. 3 Summary of the tasks executed the Host and the FPGA Vis

3.2.1 Real-time (Host) VI:

The real-time VI is planned and built based on state machine design. The state machine is one of the most effective tools available in Lab VIEW to handle applications with distinguishable states. State machines enable complicated decision making algorithms, summarized in a state flow diagram, to be implemented. In most applications with state machine architecture, the simulation starts with an initialization state and ends with a stop or shut down state that clears all actions undertaken by the system in the previous states. Each state could be followed by a single or multiple states. During the execution of an application, based on user inputs or in-state calculations, the state machine determines which state to go next.

The state flow diagram of the real-time VI designed for the current application is shown in Fig. 4. It includes 8 operational states with distinguishable tasks. Each application starts with the Initialization state that starts/resets the FPGA target and writes all user defined values for constants such as controller and sensors' calibration gains, as shown in Fig. 5. Then, the program enters the Start-up state and waits for the user to turn on the HSM low/high pressure and set the input and output file names and paths on the real-time controller hard drive (see Fig. 6).

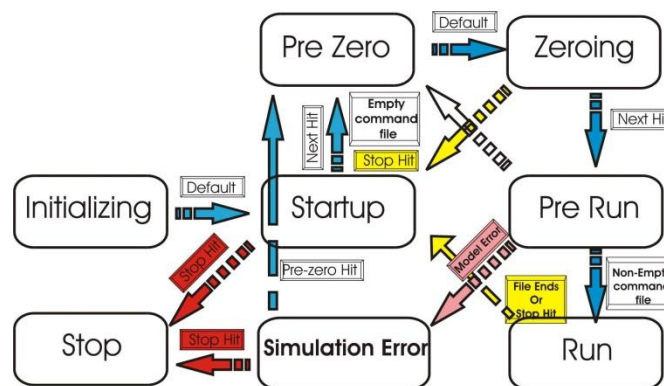


Fig. 4 State flow diagram of the Host VI for RTHS implementation

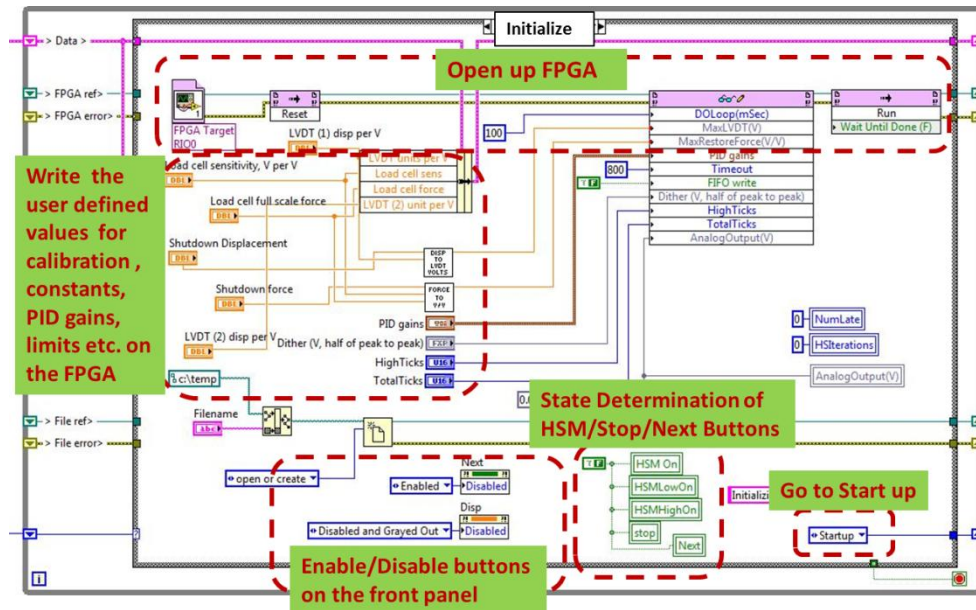


Fig. 5 Block diagram for the Initialization State

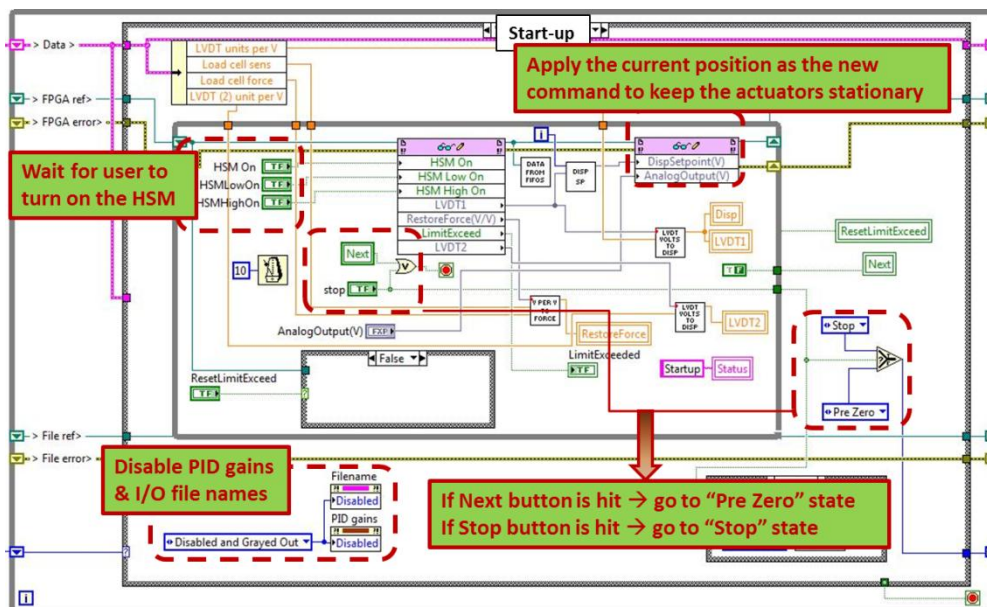


Fig. 6 Block diagram for the Start-up State

In Start-up state, if the Next button is hit, the state machine goes to an intermediate state called Pre-Zero where an analog output signal value of zero is issued to the servo-valve. This way no

erroneous valve command from a previous run that remained in the memory can be accidentally imposed. Then, the Zeroing state starts where the user is allowed to set any required initial position for the actuators before the real-time test starts (see Fig. 7). When the Next button is hit by the user, the state machine goes to the Pre-Run state, where the simulation DLL file is loaded, initialized and registered through SIT server. It is followed by the Run state.

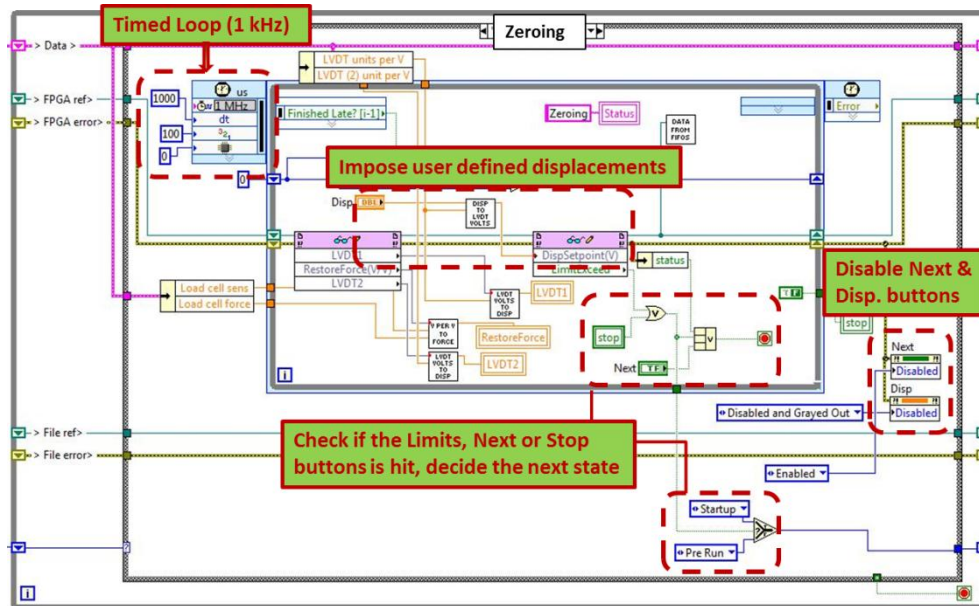


Fig. 7 Block diagram for the Zeroing state

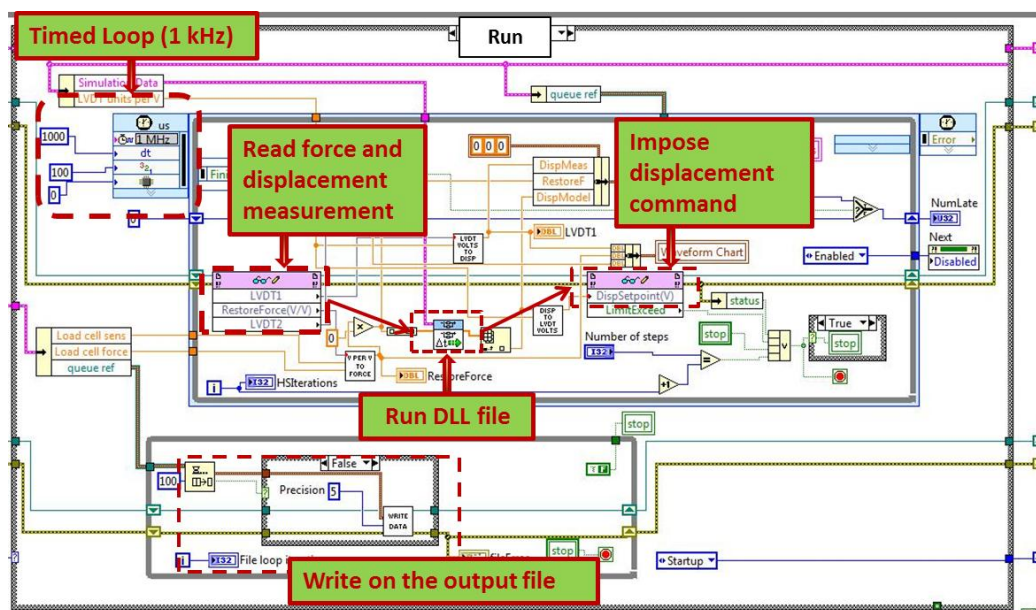


Fig. 8 Block diagram for the Run state

At each step of the Run state, the simulation DLL model is called to advance for one time step, where the integration algorithm computes next step command displacements based on measured and calculated restoring forces. These command displacements are then sent to the FPGA to be used as the displacement set point in the inner loop. When all the time steps get completed and the simulation finishes, the state machine moves to the Start-up state and waits for further input from the user. The graphical source code for the Run state is shown in Fig. 8.

In addition to all these, there is also the Simulation Error state built to check and display the error messages during the test. At any instance during the test, if the Stop button is hit, all the actions are stopped and the state-machine goes back to the Start-up state. If the Stop button is hit again while on the Start-up state, the state machine then goes to the Stop state. In the Stop state the simulation is terminated and the HSM high/low pressure solenoids are de-energized.

3.2.2 FPGA VI:

The FPGA VI shown in Fig. 9 handles the interface with the hardware, limit checks, control law implementation and dither injection (to prevent the servo-valve spool from sticking).

Controlled by the signal received from real-time VI, the first execution structure (while loop) controls the NI 9481 relay channels which in turn control the electrical signal from the external power supply that energizes the HSM solenoids. In case of emergency, when the state machine in the real-time VI is led to the Stop state, this FPGA digital loop shuts off the high pressure, low pressure, sequentially.

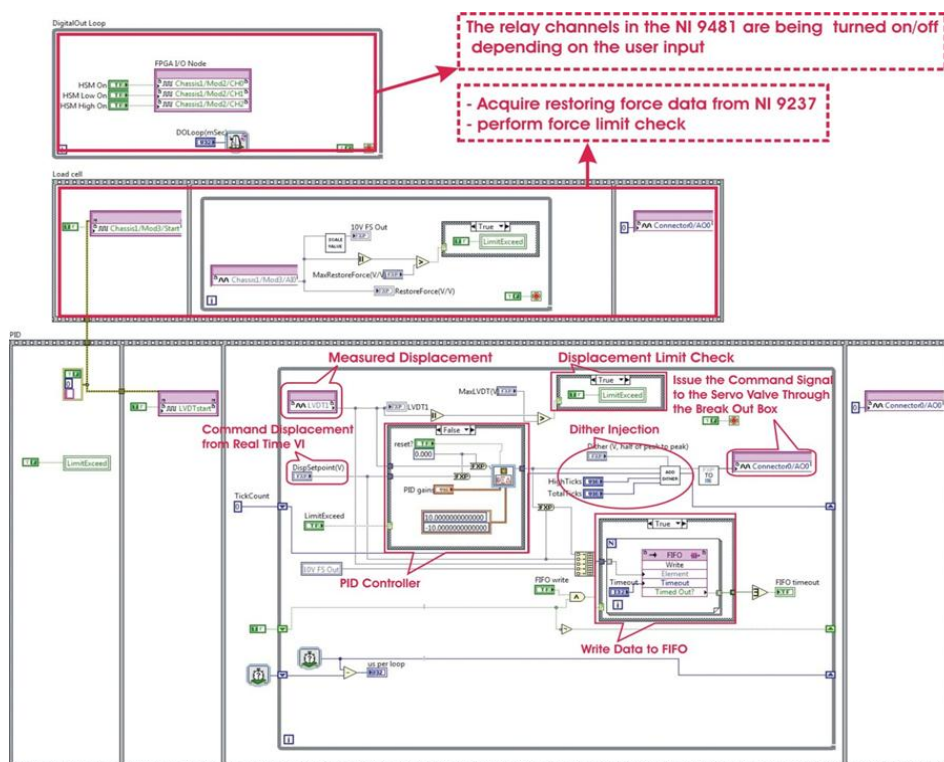


Fig. 9 FPGA block diagram

The second execution structure is designed to acquire signals from the load cells. It measures the load cell output through the NI 9237 module, and performs a limit check, and feeds the results back to the real-time VI. In the case of a limit hit, the real-time VI goes back to the Start-up state and waits for the process to be reset by the user. In the meantime, a control signal of zero volts is issued to the servo-valve, retracting the actuator to the neutral position.

In the third execution structure, currently, a PID controller is programmed to conduct the displacement control of the hydraulic actuator on the FPGA which gets the target displacements (from real-time VI), measured displacements (via NI 9239) and manipulates them based on the PID rules to generate the command(s) to the servovalve(s). Any other control law can be as easily introduced, provided that they can be broken to arithmetic operations to be programmed in LabVIEW FPGA. The user also needs to pay special attention to the fixed point math required by the FPGA operations. Since the FPGA has direct access to the hardware, control laws that need multiple states such as displacement, load, acceleration, etc. can be programmed in the FPGA VI. All the user needs to do is to replace the case structure indicated on the PID controller in Fig. 9 with their own control law. After the limit check and dither injection, the output of the control law is issued to the servo-valve as an electrical signal to drive the actuator to the desired position.

4. Verification tests and results

4.1 Displacement control of the servo-hydraulic actuators

The accuracy of the servo-hydraulic actuator displacement control executed on the FPGA was verified through three test cases. A PID controller was implemented in the FPGA to run these tests. Prior to the experiments, the PID gains were well tuned using step input.

In the first test case to control a single actuator several predefined displacement commands with harmonic and chirp type patterns were applied. As a sample result, Fig. 10(a) shows the command and measured displacements of the actuator compared over time for a chirp type displacement pattern with amplitude of 15 mm and frequency range of 0.5 to 4.5 Hz. It can be seen that there is a good agreement between the command and measured displacements, confirming the accuracy of the servo-hydraulic actuator displacement control performed by the FPGA for a single actuator. Figs. 10(b)-10(c) show the time plots for the phase and amplitude error indices (PAEI) that reveal the phase and amplitude errors between the command and measured displacements, respectively. PAEI was developed by Hessabi and Mercan (2012) as a tool to assess the accuracy of RTHS results by quantifying the amplitude and phase errors between command and measured signals. Figs. 10(b)-10(c) both confirm the above observation from Fig. 10(a), with the root mean square (RMS) of the phase and amplitude error are quantified as 0.106 rad 0.13 mm, respectively.

As the second test case, to verify the accuracy of the controller in controlling two uncoupled hydraulic actuators simultaneously, both actuators were subjected to several predefined displacement commands. As a sample set, Figs. 11(a) and 11(b) show the comparison between command and measured displacements of both actuators for a harmonic displacement pattern with variable amplitudes and frequencies. It can be seen that the controller is able to accurately track the command displacements in both actuators.

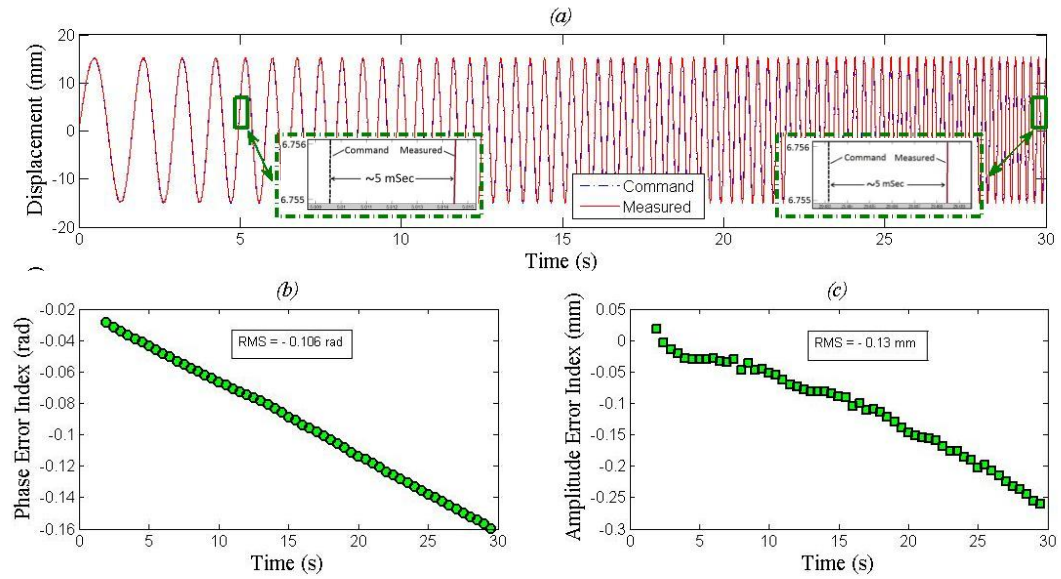


Fig. 10 Verification test results for the displacement control of a single actuator

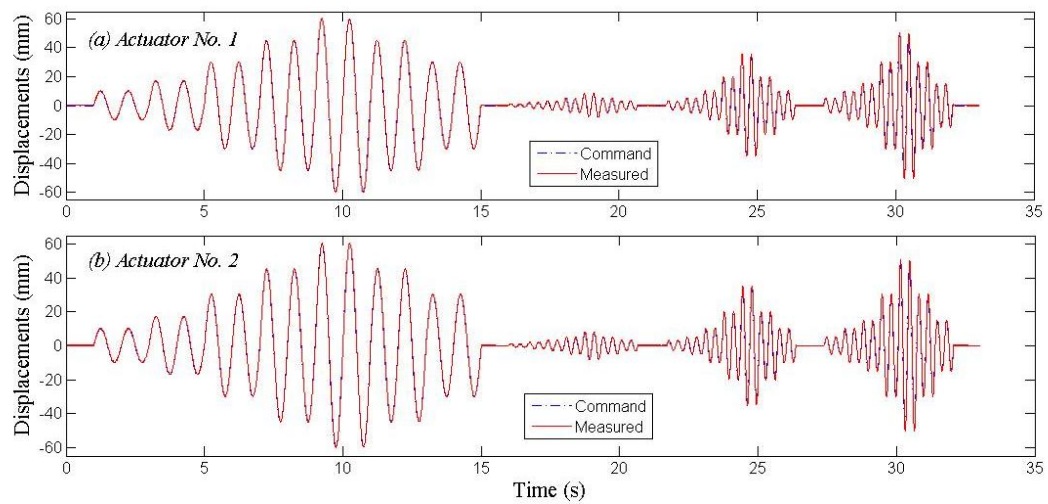


Fig. 11 Displacement tracking of two uncoupled actuators

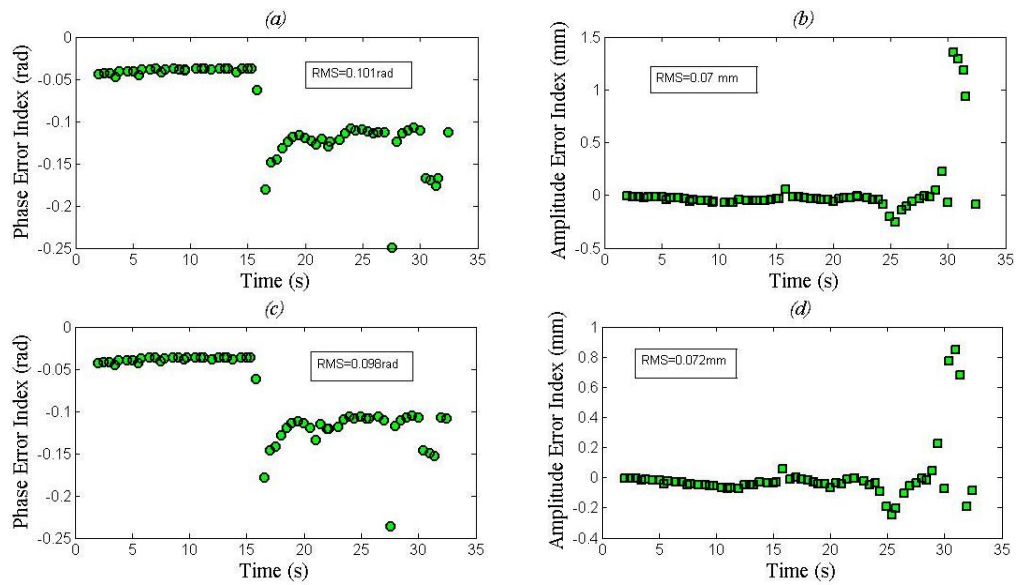


Fig. 12 Accuracy of the displacement control of two uncoupled actuators; (a) PEI for actuator No.1, (b) AEI for actuator No.1, (c) PEI for actuator No.2, (d) AEI for actuator No.2

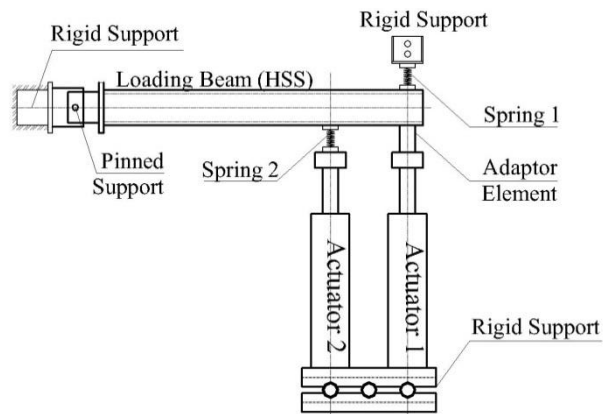


Fig. 13 Two degree of freedom coupled experimental substructure

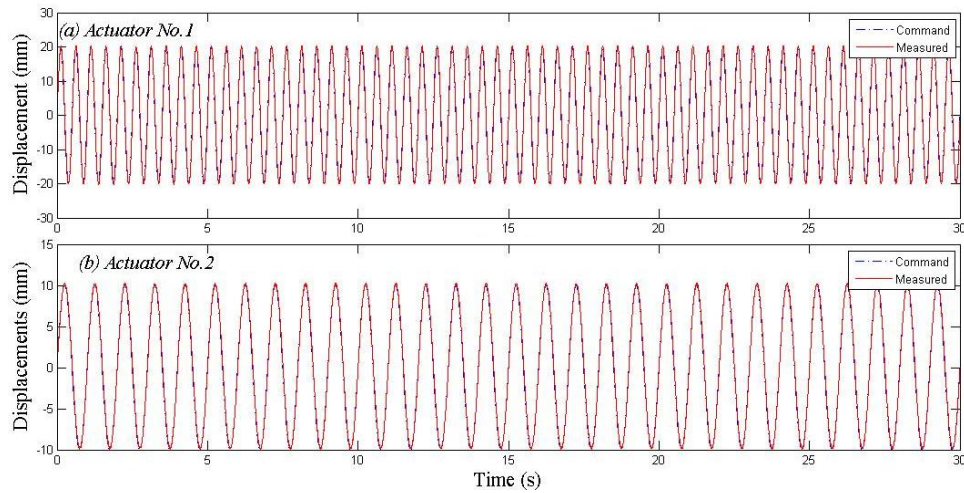


Fig. 14 Displacement tracking of two coupled actuators

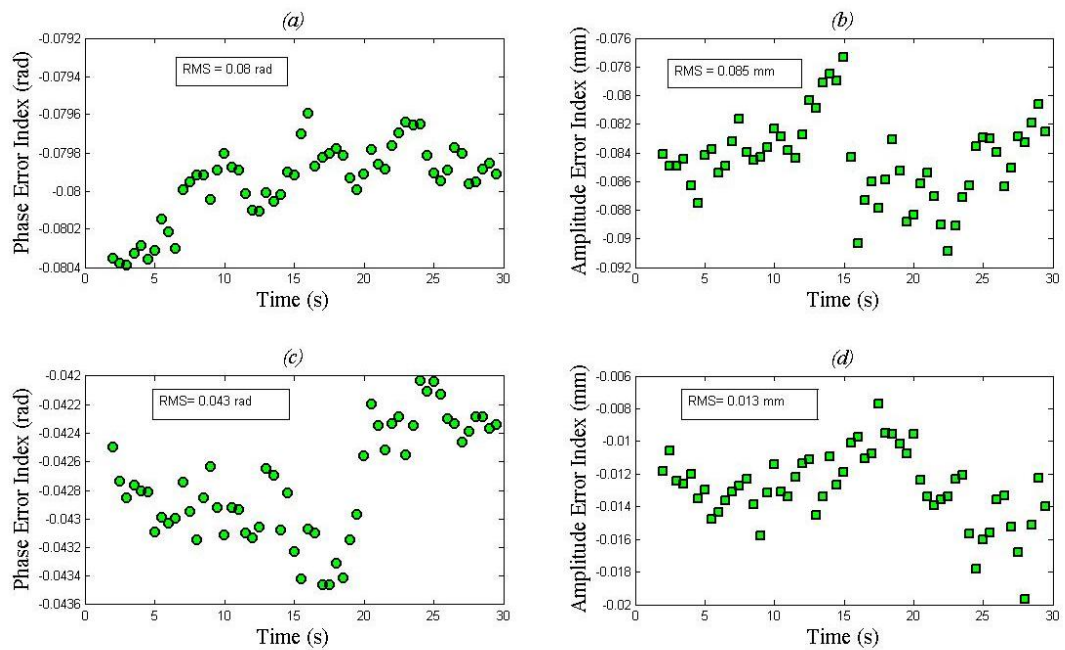


Fig. 15 Accuracy of the displacement control of two coupled actuators; (a) PEI for actuator No.1, (b) AEI for actuator No.1, (c) PEI for actuator No.2, (d) AEI for actuator No.2

In Fig. 12 the phase and amplitude errors for both actuators are quantified over the time using PAEI. As revealed by the indicators, for both actuators, the RMS values of the phase and amplitude errors are less than 0.101 rad and 0.072 mm, respectively. This shows an acceptable level of accuracy in terms of phase and amplitude while the controller is used to drive two uncoupled hydraulic actuators. It should be noted that although the two actuators are physically uncoupled, they are still hydraulically coupled through the hydraulic power unit they share during the test, which is shown not to affect their tracking performances.

As the third and last test case to verify the inner control loop accuracy, a 2DOF coupled experimental substructure, as shown in Fig. 13, was tested considering several predefined displacement histories.

As a sample result, Fig. 14 shows the displacement tracking of the actuators. Both actuators have sinusoidal displacement commands with amplitudes of 20 and 10 mm and frequencies of 2 and 1 Hz, respectively. There is a very good agreement between the command and measured displacements for both actuators. Fig. 15 shows the time plots for PEI and AEI for the inner loop with PID controller for both actuators. They confirm the above observations from Fig. 14, where the RMS values of the phase and amplitude error in both actuators are less than 0.08 rad, and 0.08 mm, respectively.

4.2 Real-time hybrid simulation experiments

To verify the RTHS functionality of the user programmable computational/control platform, RTHSs of a three story nonlinear structure with SDOF and 2DOF linear experimental substructures were conducted (see Fig. 16). For the RTHS with SDOF experimental substructure, the linear element was assumed to be located at the first floor level and for the test with 2DOF experimental substructures the linear elements were placed at the first and second floor levels.

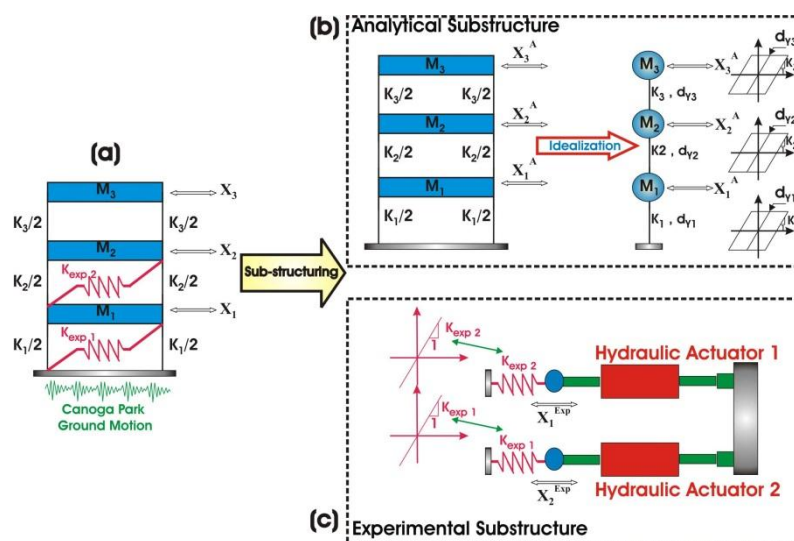


Fig. 16 Real-time Hybrid Test, (a) Structural System (b) Analytical Substructure (c) Experimental Substructure

Two linear springs (see Fig. 1) were selected and identified using the MTS loading machine in both tension and compression and then mounted to the hydraulic actuators to represent the experimental substructures during the RTHS. The stiffness coefficients of the first and second springs were identified as 210,000 N/m and 195,000 N/m, respectively. In all of these test cases the experimental substructures remained linear elastic.

An unconditionally stable implicit time-integration algorithm based on the α -method by Hilber *et al.* (1977) that is implemented into real-time PSD framework through a fixed number of iterations (Shing *et al.* 2002) was programmed in MATLAB/Simulink and the corresponding DLL files were downloaded to the real-time VI. The integration time step size for the verification tests was set to 0.001 sec. The N196E component of the 1994 Northridge earthquake ground motion recorded at Canoga Park was used as the seismic input in both RTHSs.

The structural properties of the analytical substructure for the RTHS with a single degree of experimental substructure are given in Table 1. As the experimental substructure remained linear during the RTHS, it was possible to get a "true solution" by numerical simulation; here it is referred to as Time history analysis. In figure 17a, the results of the RTHS are compared to the time history analysis. It can be seen that there is a very good agreement between the two, confirming the accuracy of the RTHS results.

In Figs. 17(b)-17(c) the command displacements from RTHS and Time history analysis are processed using the phase and amplitude error indices (PAEI) that reveal the phase and amplitude errors between them. Although these indices were initially introduced as a tool to assess the accuracy of RTHS results considering command and measured displacements of the experimental substructure, it could also be used to assess the accuracy of any two data sets against each other in terms of amplitude and phase errors. It can be seen that the root mean square values of phase and amplitude errors are less than 0.009 rad, and 0.177 mm, that confirms the observations in Fig. 17(a).

The RTHS tests with 2DOF experimental substructures were conducted considering the structural and modal properties listed in Table 2 for the analytical substructure. The simulation results for both experimental substructures are compared against the Time history analysis results in Fig. 18. Although a deterioration in the accuracy of the RTHS is observed there is still an acceptable agreement between the RTHS and Time history analysis.

Table 1 Properties of the analytical substructure for the RTHS with SDOF experimental substructure

Floor	Story Stiffness, K (N/m)	Floor Mass, M (kg)	Yield Displacement, Yd (mm)	Mode	1	2	3
				Modal Period	1.31 sec	0.49 sec	0.37 sec
1	4,200,000	48393	10	Mode Shape	1.0	-1.0	1.0
2	4,200,000	48393	10		0.84	0.12	-1.03
3	3,500,000	24196	10		0.48	0.83	0.75

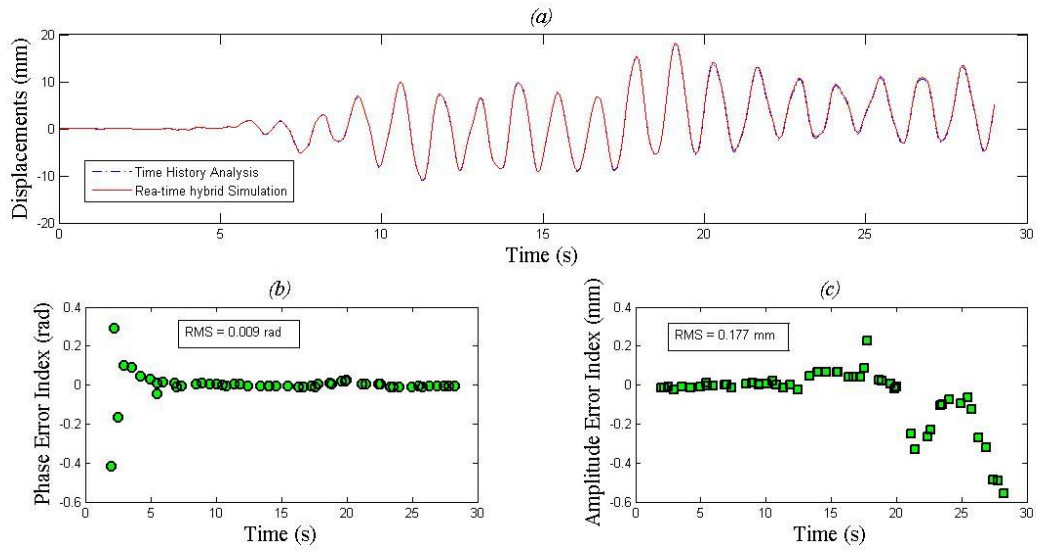


Fig. 17 RTHS results with single degree of experimental substructure (a) 1st floor RTHS results vs. time history analysis, (b) Phase error index, (c) Amplitude error index

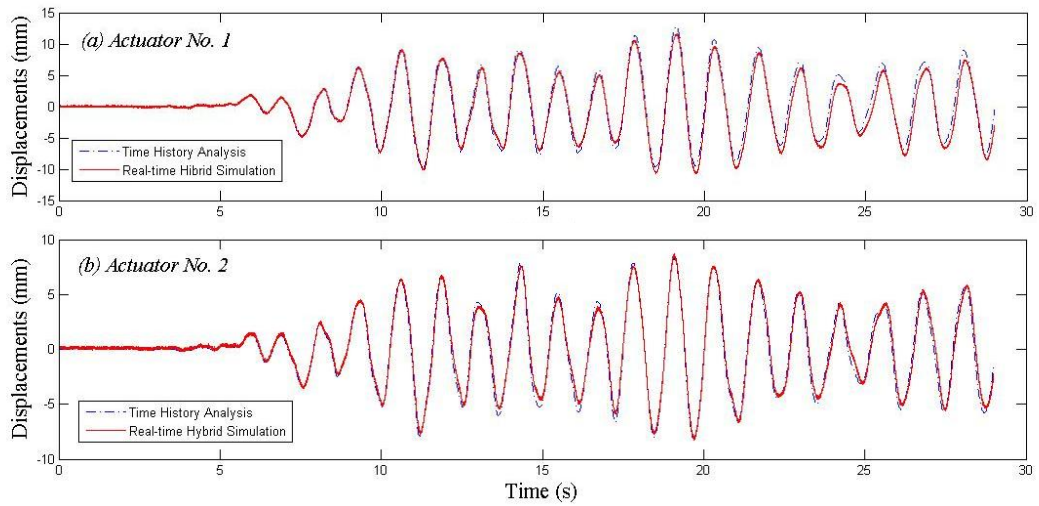


Fig. 18 RTHS results with 2DOF experimental substructures, (a) 1st floor RTHS results vs. Time history analysis, (b) 2nd floor RTHS results vs. Time history analysis

Table 2 Properties of the Analytical Substructure for RTHS with 2DOF Experimental Substructure

Floor	Story Stiffness, K (N/m)	Floor Mass, M (kg)	Yield Displacement, Yd (mm)	Mode	1	2	3
				Modal Period	1.31 sec	0.49 sec	0.37 sec
1	21,000,000	241960	10	Mode Shape	1.0 0.84 0.48	-1.0 0.12 0.83	1.0 -1.03 0.75
2	21,000,000	241960	10				
3	17,500,000	120980	10				

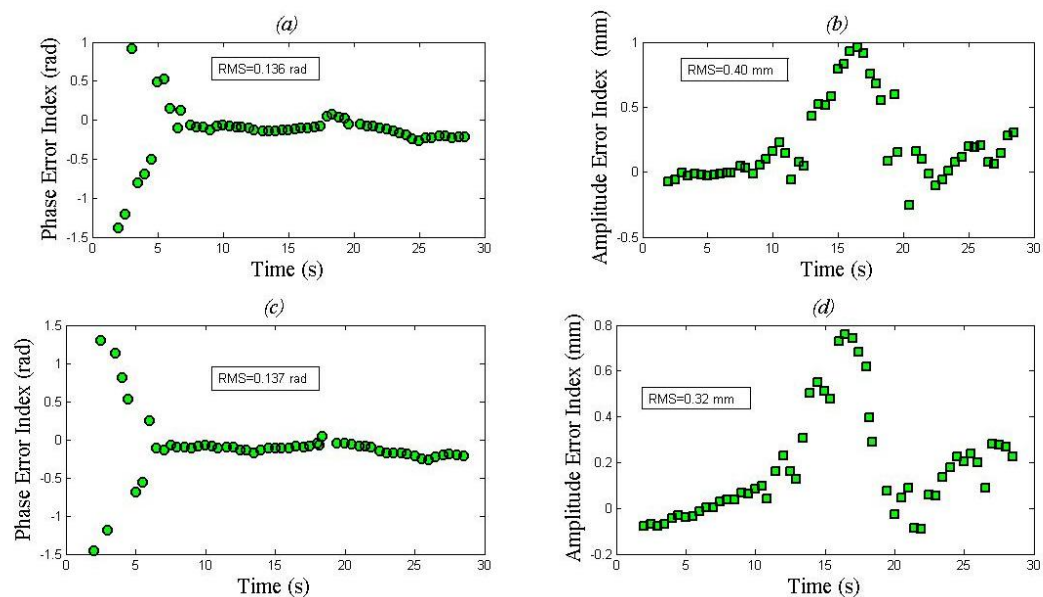


Fig. 19 RTHS with two degree of experimental substructure results; (a) PEI for actuator No.1, (b) AEI for actuator No.1, (c) PEI for actuator No.2, (d) AEI for actuator No.2

The results obtained from the RTHS are also processed using the PAEI and as shown in Fig. 19. It could be seen that in this test case the RMS values of the phase and amplitude errors for both degrees of freedom are less than 0.137rad and 0.4 mm, respectively.

5. Conclusions

This paper introduces a new computational/control platform using FPGA technology to implement real-time hybrid simulation method. A detailed explanation of the Lab VIEW codes developed on the real-time controller and FPGA sides of the platform are provided. The challenges

encountered in interfacing with the hardware and how they were handled are also explained. The verification tests presented show the success in displacement control of a single actuator as well as double actuators in both uncoupled and coupled configurations. Also, to verify the RTHS capabilities of this platform, RTHS of a three story nonlinear analytical substructure with SDOF and 2DOF experimental substructures were carried out. The results obtained from RTHS were compared with the time history analysis results. It was concluded that the RTHSs conducted using the user-programmable platform were able to capture the dynamic performance accurately for not only SDOF but also 2DOF experimental substructures and in the presence of nonlinearities in the analytical substructure. The accuracy of all the verification test cases were also confirmed using the phase and amplitude error indices.

This new computational/control platform is cost effective, very flexible and fully transparent to the user in terms of servo-hydraulic control law implementation. Another advantage of this platform is that there is no more need for a fiber optic communication device such as SCRAMNet to establish communication between control and real-time workstations. Furthermore, having displacement control loop of the servo-hydraulic system executed on the FPGA, the inner loop execution rate becomes independent of the CPU's rate and true determinism, tasks parallelism, and reliability are achieved in displacement control of the experimental substructure. With all these attributes, this platform will enable easy implementation and verification of advanced control techniques and will help enhance RTHS as an emerging testing technology.

Acknowledgments

The financial support for this study from NSERC Discovery (Grant 371627-2009) and NSERC RTI (Grant 374707-2009 EQPEQ) programs, as well as the start-up funds from the University of Toronto are gratefully acknowledged. Any opinions, findings, conclusions and recommendations expressed here are those of the authors and do not necessarily reflect the views of the sponsors.

References

- Ashasi-Sorkhabi, A., Malekghasemi, H. and Mercan, O. (2013), "Implementation and verification of real-time hybrid simulation (RTHS) using a shake table for research and education", *J. Vib. Control*, DOI: 10.1177/1077546313498616.
- Ashasi-Sorkhabi, A. and Mercan, O. (2013), "The effects of measurement errors in the restoring force feedback during real-time hybrid simulations", *Proceeding of the 9th International Conference on Earthquake Resistant Engineering Structures (ERES)*, A Coruña, Spain, July.
- Carrion, J.E. and Spencer Jr., B.F. (2006), "Real-time hybrid testing using model-based delay compensation", *Proceeding of the 4th International Conference on Earthquake Engineering*, Taipei, Taiwan.
- Carrion, J.E., Spencer Jr., B.F. and Phillips, B.M. (2009), "Real-time hybrid simulation for structural control performance assessment", *Earthq. Eng. Eng. Vib.*, **8**, 481-492.
- Chae, Y., Kazemibidokhti, K. and Ricles, J.M. (2013a), "Adaptive time series compensator for delay compensation of servo-hydraulic actuator systems for real-time hybrid simulation", *Earthq. Eng. Struct. D.*, **42**(11), 1697-1715.
- Chae, Y., Ricles, J.M. and Sause, R. (2013b), "Modeling of a large-scale magneto-rheological damper for seismic hazard mitigation. Part II: Semi-active mode", *Earthq. Eng. Struct. D.*, **42**(5), 687-703.

- Chen, C. (2007), *Development and numerical simulation of hybrid effective force testing method*, Ph.D. Dissertation, Lehigh University, Bethlehem, Pa.
- Chen, C. and Ricles, J.M. (2010), "Tracking error-based servohydraulic Actuator Adaptive Compensation for real-time hybrid simulation", *J. Struct. Eng. - ASCE*, **136**(4), 432-440.
- Chen, C., Ricles, J.M. and Guo, T. (2012), "Improved adaptive inverse compensation technique for real-time Hybrid simulation", *J. Eng. Mech. - ASCE*, **138**(12), 1432-1446.
- Chen, P.C. and Tsai, K.C. (2013), "Dual compensation strategy for real-time hybrid testing", *Earthq. Eng. Struct. D.*, **42**(1), 1-23.
- Christenson, R.E., Lin, Y.Z., Emmons, A.T. and Bass, B. (2008), "Large-scale experimental verification of semi-active control through real-time hybrid simulation", *J. Struct. Eng. - ASCE*, **134**(4), 522-535.
- CompactRio Developers Guide (2009), *National Instruments Corporation*, Austin, TX, USA.
- Dermitzakis, S.N. and Mahin, S.A. (1985), *Development of substructuring techniques for on-line computer controlled seismic performance testing*, Report UBC/EERC-85/04, Earthquake Engineering Research Center, University of California, Berkeley, CA.
- Gao, X., Castaneda, N. and Dyke, S.J. (2013), "Real time hybrid simulation: from dynamic system, motion control to experimental error", *Earthq. Eng. Struct. D.*, **42**(6), 815-832.
- Harvey, A.F. and Data Acquisition Division Staff (1991), *DMA Fundamentals on Various PC Platforms*, Application Note 011, National Instruments Corporation, Austin, TX, USA.
- Hessabi, R.M. and Mercan, O. (2012), "Phase and amplitude error indices for error quantification in pseudodynamic testing", *Earthq. Eng. Struct. D.*, **41**(10), 1347-1364.
- Hilber, H.M., Hughes, T.J.R. and Taylor, R.L. (1977), "Improved numerical dissipation for time integration algorithms in structural dynamics", *Earthq. Eng. Struct. D.*, **5**(3), 283-292.
- Horiuchi, T., Inoue, M., Konno, T. and Namita, Y. (1999), "Real-time hybrid experimental system with actuator delay compensation and its application to a piping system with energy absorber", *Earthq. Eng. Struct. D.*, **28**(10), 1121-1141.
- Horiuchi, T. and Konno, T. (2001), "A new method for compensating actuator delay in real-time hybrid experiment", *Philos. T. R. Soc. Lond.*, **359**(1786), 1893-1909.
- Jung, R.Y. and Shing, P.B. (2006), "Performance evaluation of a real-time pseudodynamic test system", *Earthq. Eng. Struct. D.*, **35**(7), 789-810.
- Karavasilis, T.L., Ricles, J.M., Sause, R. and Chen, C. (2011), "Experimental evaluation of the seismic performance of steel MRFs with compressed elastomer dampers using large-scale real-time hybrid simulation", *Eng. Struct.*, **33**(6), 1859-1869.
- LabVIEW Tutorial Manual(1996), *National Instruments Corporation*, Austin, TX, USA.
- Liu, J., Dyke, S.J., Liu, H.J., Gao, X.Y. and Phillips, B. (2013), "A novel integrated compensation method for actuator dynamics in real-time hybrid structural testing", *Struct. Control Health Monit.*, **20**(7), 1057-1080.
- Mahin, S.A. and Shing, P.B. (1985), "Pseudodynamic method for seismic testing", *J. Struct. Eng. - ASCE*, **111**(7), 1482-1503.
- Mercan, O. and Ricles, J.M. (2007), "Stability and accuracy analysis of outer loop dynamics in real-time pseudodynamic testing of SDOF systems", *Earthq. Eng. Struct. D.*, **36**(11), 1523-1543.
- Mercan, O., Ricles, J.M. and Sause, R. (2007), "Implementation of real-time hybrid pseudodynamic test method for evaluating seismic hazard mitigation measures", *Proceeding of the Structures Congress (ASCE)*, Long Beach, CA, USA.
- Mercan, O. and Ricles, J.M. (2008), "Stability analysis for real-time pseudodynamic and hybrid pseudo dynamic testing with multiple sources of delay", *Earthq. Eng. Struct. D.*, **37**(10), 1269-1293.
- Mercan, O. and Ricles, J.M. (2009), "Experimental studies on real-time pseudodynamic (PSD) and hybrid PSD testing of structures with elastomeric dampers", *J. Struct. Eng. - ASCE*, **135**(9), 1124-1133.
- Nakashima, M., Kato, H. and Takaoka, E. (1992), "Development of real-time pseudodynamic testing", *Earthq. Eng. Struct. D.*, **21**(1), 79-92.
- Nakata, N. (2011), "A multi-purpose earthquake simulator and a flexible development platform for actuator controller design", *J. Vib. Control*, DOI: 10.1177/1077546311421946.

- NI 9237 User Guide and Specifications (2007), *National Instruments Corporation*, Austin, TX, USA.
- NI 9239 User Guide and Specifications (2007), *National Instruments Corporation*, Austin, TX, USA.
- NI 9481 Operating Instructions and Specifications (2008), *National Instruments Corporation*, Austin, TX, USA.
- Phillips, B.M. and Spencer Jr., B.F.(2011), *Model-based feedforward-feedback tracking control for real-time hybrid simulation*, NSEL Report Series, Report No. NSEL-028.
- Reinhorn, A.M., Sivaselvan, M.V., Liang, Z. and Shao, X. (2004), "Real-time dynamic hybrid testing of structural systems", *Proceedings of the 13th World Conference on Earthquake Engineering (WCEE)*, Vancouver, B.C., Canada.
- Shao, X., Reinhorn, A.M. and Sivaselvan, M. (2006), "Real-time dynamic hybrid testing using force-based substructuring", *Proceeding of the Structures Congress (ASCE)*, Reston, Va.
- Shing, P.B., Spacone, E. and Stauffer, E. (2002), "Conceptual design of a fast hybrid test system at the University of Colorado", *Proceeding of the 7th US Conference on Earthquake Engineering*, Boston, MA, USA.
- Takanashi, K., Udagawa, K., Seki, M., Okada, T. and Tanaka, H. (1975), "Non-linear earthquake response analysis of structures by a computer actuator on-line system", *Bulletin of Earthquake Resistant Structure Research Center*, No. 8, Institute of Industrial Science, University of Tokyo, Tokyo, Japan.
- Wallace, M.I., Sieber, J., Nield, S.A., Wagg, D.J. and Krauskopf, B. (2005), "Stability analysis of real-time dynamic substructuring using delay differential equations", *Earthq. Eng. Struct. D.*, **34**(15), 1817-1832.
- Wu, B., Shi, P. and Ou, J. (2013), "Seismic performance of structures incorporating magnetorheological dampers with pseudo-negative stiffness", *Struct. Control Health Monit.*, **20**(3), 405-421.
- Wu, B., Xu, G., Wang, Q. and Williams, M.S. (2006), "Operator-splitting method for real-time substructuring testing", *Earthq. Eng. Struct. D.*, **35**(3), 293-314.
- Zhao, J., French, C., Shield, C., and Posbergh, T. (2003), "Considerations for the development of real-time dynamic testing using servohydraulic actuation", *Earthq. Eng. Struct. D.*, **32**(11), 1773-1794.