



## TEST REHEARSAL METHODS FOR QUASI STATIC AND REAL-TIME HYBRID SIMULATIONS

S. You<sup>(1)</sup>, A. Schellenberg<sup>(2)</sup>

<sup>(1)</sup> Staff Engineer, MTS Systems, shawn.you@mts.com

<sup>(2)</sup> Principal, ASES - Advanced Structural Engineering Solutions, andreas.schellenberg@gmail.com

### Abstract

Over the last decade, the popularity of the hybrid simulation testing method has significantly increased and the testing method has been extensively adopted by the research community to conduct experimental earthquake and multi-hazard studies. While hybrid simulation provides a versatile, realistic and cost-effective method for simulating seismic response experimentally, test specimens and test setups have also become increasingly more complicated and therefore more costly. During a hybrid test, the command for the test control system is provided by a numerical analysis model based upon experimental feedback signals from the test specimen. Due to the nature of servo hydraulic test systems, which inherently include system delays and sensor noise, there is always discrepancy between command and feedback signals. Because these experimental errors, especially the systematic errors, are fed back to the numerical analysis model, they can accumulate during the solution process and cause a hybrid test to go unstable. As a consequence, the costly test specimens and possibly even the test setups can be damaged due to these incorrect loading histories. Thus, it is critical to have test methods available that can rehearse, meaning virtually simulate the different portions of a hybrid simulation, prior to the real hybrid test.

This paper introduces several test rehearsal methods for quasi-static and real-time hybrid simulation, where an increasing number of parts of the overall hybrid testing system are being included in the virtual simulation.

#### 1) FE adapter element method

In this method, the FEAdapter experimental control feature in OpenFresco is being used to connect the hybrid analysis model with a test specimen that is being simulated in a Finite Element Analysis (FEA) software. A wide range of FEA software can be used to simulate the test specimen and it can be used to simulate both quasi static and real-time hybrid simulations. This test rehearsal method provides the means to verify that the hybrid model has been setup correctly, appropriate analysis parameters have been selected, and all of the OpenFresco components have been correctly defined.

#### 2) Calculation channels in MTS Flextest software

The MTS Calculation Channel feature can be used to model the specimen, servo valve behavior, system delay, and sensor noise. The resulting signals from these calculations can be used in control and also output as valve drive signals. The virtual system runs in real-time according to the system clock rate. Combined with a hybrid model, the Calculation Channel functionality provides an effective way to simulate a hybrid test. This method is typically used for quasi static hybrid simulations.

#### 3) Model test system in Simulink and run it on target PC

Simulink is used to model the physical test system. The model is compiled and downloaded into a target PC and then run in real-time. Since Simulink is a powerful analysis tool, complicated specimen properties, system delay, sensor noise, and other experimental errors can be modeled accurately. This method can be used in both quasi static and real-time hybrid simulations.

Each test rehearsal method is illustrated through an example. By using proper test rehearsal methods, real hybrid tests have a much better opportunity to be successful. Incorporating such rehearsal techniques in a testing program can save valuable time and money by mitigating risk and improving overall accuracy.

**Keywords:** test rehearsal, hybrid simulation, calculation channel, Simulink, co-simulation.



## 1. Introduction

Hybrid simulation is a dynamic experimental testing method where a system is divided into numerically simulated and experimentally tested subassemblies. The simulation is then based on a step-by-step numerical integration of the equations of motion; where, for each analysis step, the numerical model calculates command signals, sends them to the experimental test system, and then collects the resulting response from instrumentation devices. Due to physical system and communication delays, sensor noise, joint friction, and other experimental errors, hybrid dynamic simulations can lose accuracy or even go unstable as a test progresses. The accumulation of these errors during a test causes the numerical subassembly to send unrealistic command signals to the experimental test system.

Due to the increasingly more sophisticated and complex hybrid simulations that are being conducted worldwide, many of these tests consist of expensive and fragile test specimens, complicated lengthy and thus expensive test setups, and complicated control algorithms and error compensation techniques. It is possible to cause damage to the specimen when a hybrid model is not setup correctly. Also, a hybrid test can go unstable because control algorithms and error compensation techniques have not been tuned correctly. Therefore, hybrid simulations can be risky if testing commences without previously verifying that all the employed testing components are set up and operating correctly. For this reason, an increasing number of researchers are starting to virtually simulate hybrid tests prior to their actual execution to make sure that everything will work as expected. In addition to mitigating risk, the virtual simulation of a hybrid test (virtual testing) can also be utilized to pre-tune control and error compensation parameters, which will lead to increased accuracy and confidence during the actual hybrid test.

Typical structural test specimens are load resisting and/or energy dissipating devices. With loads being applied by actuators, the structural specimen deforms and a dynamic response (displacement, velocity, and acceleration) is being generated. The relationship between the load and the dynamic response can be characterized and mathematical models can be created to mimic the behavior of the test specimen. In addition to the resistance, the energy dissipation and inertia of the specimen can also be modeled. In this way, a virtual specimen is simulated to provide the response for a given command. The virtual specimen can be as simple as a linear-elastic spring or as complicated as a non-linear dynamic finite element model.

To make the test rehearsal more realistic, test fixture, joint friction, sensor noise, system delay and roll-off, hydraulic components, cross-coupling effects, and control and error compensation algorithms can also be modeled and included in the virtual testing system. Given this virtual testing method, it is possible to find and rectify potential issues in the hybrid model and test system prior to conducting the actual test. Thereby, it is possible to greatly increase the success rate for hybrid simulation experimental tests.

There are a range of test rehearsal methods that become increasingly more complex as more components of the test system are being included in the virtual testing. This paper discusses three virtual testing methods and illustrates them through examples.

The first test rehearsal method is based on co-simulation (co-operative simulation) where the experimental test specimen is modeled and simulated in one finite element analysis software while the numerical portion of the hybrid model is simultaneously analyzed in a second finite element analysis software. This method provides the means to confirm that the hybrid model has been setup correctly and appropriate analysis parameters have been selected. It can be used to rehearse quasi-static as well as real-time hybrid simulations.

The second test rehearsal method is to use the “Calculation Channel” functionality which is part of the MTS Flextest 793 software to model the test system and specimen. The advantage of this method is that no additional software and/or hardware are needed besides the test control software. However, this method is only used for simulating quasi-static or slow hybrid simulations and typically the simulated test specimen is fairly simple.

When more accurate virtual testing results are desired, the dynamic behavior of the hydraulic test system needs to be modeled as well. The third test rehearsal method uses Simulink to create the plant model. The

Simulink model can then be downloaded to a target PC in order for the model to be run in real-time. As a result, this method can be used to rehearse quasi-static as well as real-time hybrid simulations.

## 2. FE Adapter Element Method

The adapter element method is a simulation approach that couples two or more displacement-based structural FEA programs together [1]. During a simulation, all the FEA programs run concurrently and continuously exchanging data at the interfaces of their domains. Oftentimes such an approach is also referred to as co-simulation (co-operative simulation). The FE adapter element method can readily be employed to rehearse a hybrid simulation by using one FEA software to model and analyze the numerical portion of the hybrid model while the experimental test specimen is being modeled and simultaneously analyzed in a second FEA software.

### 2.1 Concept

In this co-simulation approach, the OpenFresco (Open-source Framework for Experimental Setup and Control) middleware [2, 3] is being employed to connect the different FEA programs in a modular and highly structured manner. The exchange of data between the coupled programs takes place through the combination of OpenFresco with special adapter and generic elements that are added to each FEA program using their published programming interfaces. Adapter and generic elements have been implemented for a range of FEA and technical computing software, such as OpenSees, Abaqus, LS-DYNA, ANSYS, Matlab, and Simulink.

This test rehearsal method can be used to simulate both quasi static and real-time hybrid simulations. The method provides the means to verify that the hybrid model has been set up correctly, appropriate analysis parameters have been selected, and all the OpenFresco components have been defined correctly. However, the method does not include any of the actual control system components in the test rehearsal and can thus not simulate any effects due to control system dynamics.

Whenever multiple FEA codes are coupled together, one of the programs needs to be selected to act as the master, solving part of a system and driving the analysis, while the other linked programs model and analyze different structural subassemblies, and thus, act as slaves. Each slaved subassembly is acting as a super-element and is connected to the master program via the exchange of a set of response variables at the interface degrees of freedom (see Fig.1). During an analysis, the master program imposes boundary conditions on all the subassemblies and the slave programs return the corresponding work conjugates and possibly their stiffness or flexibility matrices.

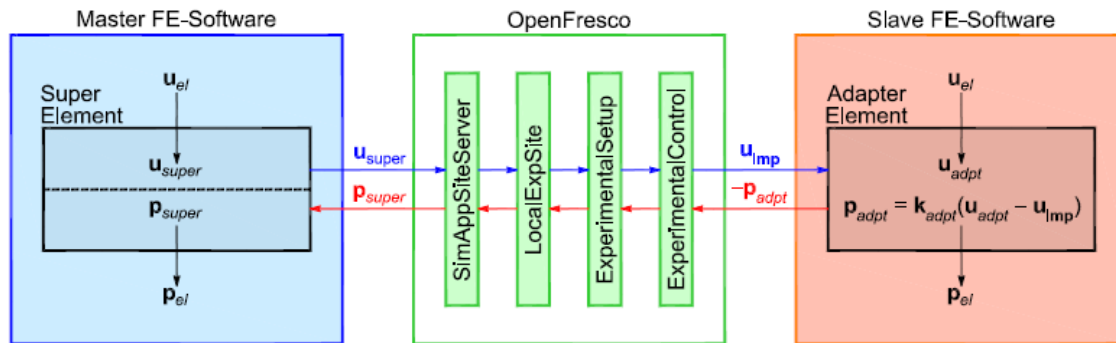


Fig. 1 – Components and data exchange for adapter element method

Applying this concept to conduct virtual hybrid simulations, the master program is set up as the analysis driver solving the numerical portion of the hybrid model in the exact same way as it is done for a real hybrid test. Next, the test specimen is modeled in the slave FEA code where the specimen model can be as simple or as complex as desired. Furthermore, the slave software can be a completely different FEA code than the master software or it can simply be another instance of the same master software. Identical to a displacement-controlled hybrid simulation, the boundary conditions that are being sent to the slave program are command displacements



and forces at the interface degrees-of-freedom in the slave program are being returned to the master program. As can be seen from Fig.1, the interaction with the slave program in terms of imposing displacements and acquiring forces at the interface degrees-of-freedom is made possible through the adapter element which is being added to the FE model of the specimen in the slave program.

To guarantee that the interface degrees of freedom move by the prescribed displacement values, the linear-elastic stiffness (penalty stiffness) of the adapter element should be two to three orders of magnitude larger than the stiffness of the simulated specimen. This concept is comparable to a hybrid simulation where a transfer system (actuator) with a large stiffness (high oil-column stiffness) is imposing boundary conditions (displacements) on an experimental subassembly.

## 2.2 Implementation Details

During a test rehearsal, the following sequence of operations (shown in Fig. 1) is executed: Starting on the side of the master program, the super-element receives a vector of global trial displacements  $\mathbf{u}_{\text{super}} = \mathbf{u}_{\text{el}}$  for all its degrees-of-freedom from the master integration method. It then sends these displacements, using a TCP/IP socket, to the OpenFresco simulation application server. If necessary, the experimental site and setup modules are responsible for storing and transforming the response quantities. Next, the trial displacements are passed to the SimFEAdapter experimental control object that provides the connection to the adapter element also utilizing a TCP/IP socket. The adapter element then combines the received displacements  $\mathbf{u}_{\text{imp}}$  with its own element displacements  $\mathbf{u}_{\text{adpt}} = \mathbf{u}_{\text{el}}$  from the subassembly. Subsequently, the element force vector  $\mathbf{p}_{\text{el}} = \mathbf{p}_{\text{adpt}}$  of the adapter element is updated using the penalty stiffness  $\mathbf{k}_{\text{adpt}}$  and returned to the subassembly. Once the equilibrium solution process of the slave program has converged, the negative of the element force vector  $-\mathbf{p}_{\text{adpt}}$  is returned to the SimFEAdapter experimental control object across the TCP/IP socket. The experimental site and setup modules are again responsible for storing and transforming the response quantities. Next, the simulation application server returns the force vector through the TCP/IP socket to the super-element in the master program. Finally, the super-element saves them as element forces  $\mathbf{p}_{\text{el}} = \mathbf{p}_{\text{super}}$  and returns them to the master integration method, which is then capable to determine the new trial displacements and proceed to the next time step.

If OpenSees (Open System for Earthquake Engineering Simulation) is being employed as the master program, the super-element on the left side of Fig.1 can be replaced by any of the experimental elements available in OpenFresco. This has the advantage that instead of using a TCP/IP connection to communicate between the super-element and OpenFresco, it is possible to directly use function calls to communicate between the experimental element and the LocalExpSite in OpenFresco. This reduces communication delays and thus simulation times.

If this virtual simulation method is being used to rehearse real-time hybrid simulations in cases where test specimen exhibit rate dependent and/or dynamic behavior, the following important points need to be considered when setting up the test rehearsal.

Firstly, appropriate and compatible time integration methods need to be selected in the master and slave programs, so that time in both programs advances at the same rate. Explicit or Operator-Splitting methods, which only require one data exchange per integration time step, or the specialized constant iteration hybrid simulation integrators, which require a fixed number of data exchanges per time step, need to be employed for the dynamic analysis in the master program [3]. For the dynamic analysis in the slave programs, any transient integration method can be used as long as the time step sizes are compatible with the master integrator. However, methods that provide numerical energy dissipation, such as the class of Generalized Alpha methods, can eliminate high frequency acceleration chatter at the interface degrees of freedom and are thus recommended for the slave programs. On the other hand, if the behavior of the subassembly in the slave program is time independent (no inertia and energy dissipation effects), a static integrator should be employed in the slave program and any integration method can be utilized for the static or dynamic analysis in the master program.

Secondly, to prevent dynamic instabilities when the adapter element is imposing displacements, no mass should be assigned at the interface degrees of freedom in the slave program.



## 2.1 Example Application

To demonstrate the benefit of the adapter element method, an example application is presented next where the method is employed to rehearse a real-time hybrid simulation of a base isolated nuclear power plant. The example is based on a research program on seismic isolation of nuclear power plants that experimentally assessed the behavior of very large full-scale isolators through hybrid simulation. The adaptation of the Caltrans Seismic Response Modification Device (SRMD) testing facility and the implementation and execution of hybrid simulations using full-scale experimental bearings in a seismically isolated nuclear power plant are discussed in detail in [4, 5, 6].

The hybrid model that was used for these tests is based on a simplified, but realistic stick model of an Advanced Power Reactor 1400 (APR-1400) nuclear power plant designed by KEPCO Engineering and Construction. Several different isolation systems were being tested, including lead plug rubber bearings (LPRB), EradiQuake system bearings (EQSB), and friction pendulum system bearings (FPSB). It is important to note, that all of these isolators exhibit rate dependent behavior either due to their temperature dependence and/or velocity dependence of the materials and mechanical mechanisms involved. The isolation system was represented in the hybrid model by (1) a single test bearing representing all 486 bearings supporting the plant, (2) a single test bearing representing a group of bearings under the plant, or (3) a single test bearing representing one of the 486 bearings. The isolators that were not being tested experimentally were modeled numerically and the plant super structure, which was required to remain elastic, was also modeled numerically. The three different hybrid models consisted of (1) 456 free degrees of freedom, (2) 552 free degrees of freedom, and (3) 10752 free degrees of freedom. To conduct the actual hybrid simulations in real-time and capture the isolation bearing's rate dependency, it was necessary to analyze the hybrid model in OpenSeesSP (the parallel version of OpenSees) [7, 8] on an overclocked high performance multi-core machine.

Before the actual hybrid tests were being executed, virtual hybrid simulations using the adapter element method were conducted for each of the three isolator types and each of the three hybrid models. The reasons for performing these test rehearsals were to (a) verify that the OpenSees model had been setup correctly, (b) that the OpenSeesSP analysis parameters had been defined adequately, (c) that the OpenFresco components, including a newly developed experimental bearing element and the experimental setup, had been defined correctly, and (d) to obtain response estimates for the test specimens, so that appropriate limits and control parameters could be set in the MTS SRMD controller software.

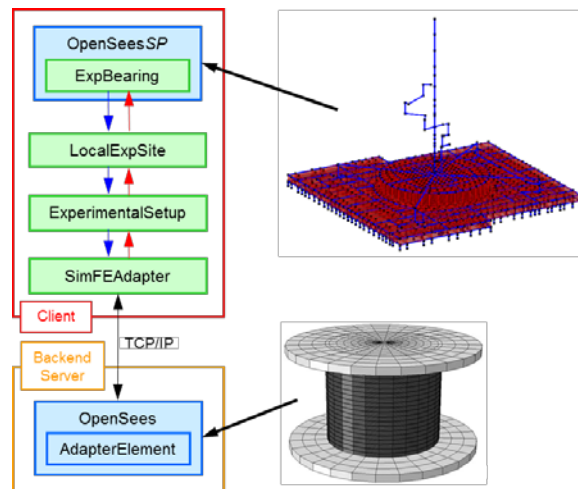


Fig. 2 – Components of FE Adapter test rehearsal method

The hybrid models for these test rehearsals, which were analyzed in OpenSeesSP, were identical to the ones used in the subsequent actual hybrid tests. They consisted of the entire nuclear power plant super structure and all the isolators except for the single isolator that was being experimentally tested in the SRMD machine. Because OpenSees was being used for these tests, it was possible to directly represent the tested isolator



specimen through an experimental bearing element without having to use a generic element in the master program (see Fig. 2). For the slave program, the sequential version of OpenSees was employed to model and analyze the isolator test specimen. As mentioned above, all three isolator types exhibited rate dependencies and it was therefore necessary to run dynamic analyses in the slave program. The AlphaOSGeneralized integration method with  $\rho_{inf} = 0.7$  was selected for the master program and, to avoid high frequency acceleration chatter, the HHTGeneralized integration method with  $\rho_{inf} = 0.0$  was selected for the slave program. The integration time step size for both the master and the slave program were set to 0.01 seconds. The adapter element that was being added to the slave program consisted of three degrees of freedom in order to control the two horizontal deformations and the vertical deformation of the isolator test specimen. The penalty stiffness values of the adapter element were set to 1.5E5 kN/mm for the two horizontal degrees of freedom and due to the higher axial test specimen stiffness to 1.5E8 kN/mm for the vertical degree of freedom. It is essential to keep in mind that very large penalty stiffness values could cause numerical problems and penalty stiffness values that are too small will reduce the accuracy of the imposed displacements. For the LPRB isolator, the specimen was modeled using the LeadRubberX element [9]. For the EQSB isolator, the RJWatsonEqsBearing element was used. And, for the FPSB isolator, the TripleFrictionPendulum element [10] was used. The properties for the numerical isolator elements were estimated based on material and geometric properties provided by the bearing manufactures or, if available, they were estimated based on isolator hysteresis loops from manufacturer quality control tests.

It is important to note that for this specific problem, gravity loads were applied in the master program before running the dynamic earthquake analysis. If the vertical degree of freedom is controlled in the hybrid simulation, the experimental specimen is loaded in axial direction during the gravity load analysis. If only the two horizontal degrees of freedom are controlled, the gravity load is manually applied through the controller set points before starting the hybrid simulation. For the test rehearsals discussed here, the vertical degree of freedom was always controlled from the master program. Thus, the vertical load was applied to the simulated test specimen in the slave program by imposing vertical displacements through the adapter element.

In contrast, there are other hybrid simulations, such as hybrid shake table tests [11], where the gravity load on the test specimen is always present and can thus not be applied through the control system. To virtually simulate such conditions in the slave program, it is necessary to perform a gravity load analysis in the slave program before adding the adapter element. Once the gravity loads are applied to the simulated test specimen, the adapter element is added which will then connect to the master program and impose displacements on the test specimen at the controlled interface degrees of freedom. Furthermore, since it is still necessary to apply gravity loads to the numerical portions of the hybrid model, displacement commands that are being sent to the control system or the slave program need to be relative to the state at the end of the gravity load analysis in the master program. This feature, to utilize relative instead of absolute trial displacements, is directly built into OpenFresco, so that users do not have to make any special changes when performing hybrid shake table tests.

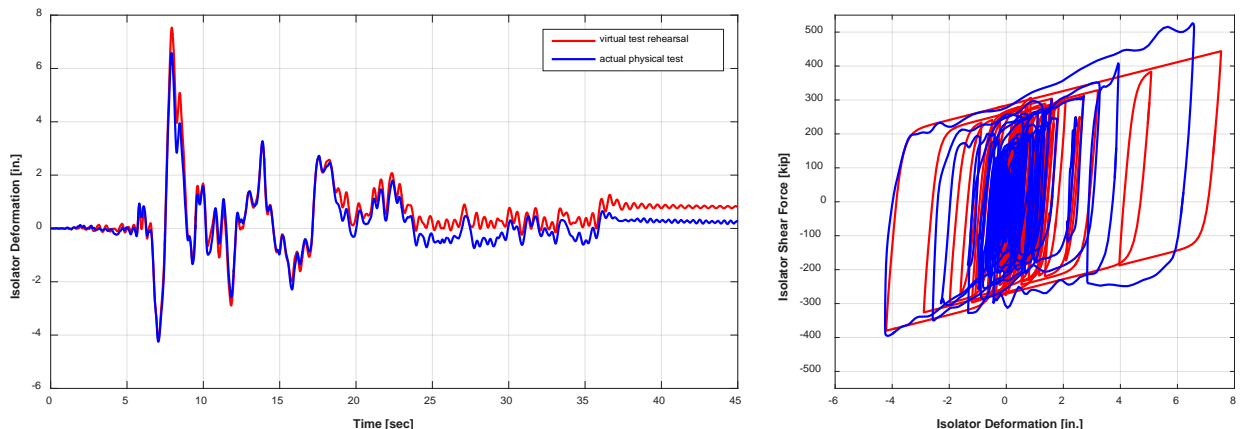


Fig. 3 – Comparison of isolator displacement histories and hysteresis loops

Fig. 3 shows the comparison of two lead plug rubber bearing hysteresis loops in the longitudinal direction for the nuclear power plant, where a single bearing was representing all 486 bearings supporting the plant. The hybrid model was subjected to one horizontal component of the 1979 Imperial Valley-06, Niland Fire Station earthquake record spectral-matched to the NRC RG1.60 (Nuclear Regulatory Commission - Regulatory Guide 1.60) target spectrum with a peak ground acceleration of 0.5g. The first hysteresis loop (in red) shows the result from the test rehearsal that was performed before running the actual hybrid simulation and the second hysteresis loop (in blue) shows the experimental result from the actual real-time hybrid simulation that was carried out subsequently. As can be seen, the test rehearsal was able to predict the overall isolator response quite accurately providing adequate displacement and force estimates to set controller limits for the actual physical test. In addition, the test rehearsal successfully confirmed that the OpenSees model and all the OpenFresco components were setup correctly and that OpenSeesSP was able to perform calculations fast enough to run the actual physical test in real time. Finally, the comparison of the hysteresis loops also shows that there is still some room to improve the analytical simulation of the lead plug rubber bearing, especially to better capture the shear resistance in the first large positive deformation cycle.

### 3. Calculation Channel Functionality in Flextest to Model the Test System

The MTS Flextest software allows loop output signals of the control loop to be converted to compensated servo valve commands. The feedback signals can then be calculated from the valve commands. Calculation parameters can be defined so that the calculation can be modified easily to fit different situations.

Through a Calculation Editor, the MTS Flextest software allows calculations to contain regular mathematical operations and functions as well as complex algorithms, based on the C programming language, containing elements such as variables, arrays, loops and conditions. All system signals, digital and analog, can be used for calculations. Calculations are initialized when loading the test configuration. They are active from that point on, unless the calculation itself defines a different behavior, e.g. activate only if high pressure is applied. As a result, the proper use of the Calculation Channel functionality becomes extremely powerful and can significantly extend the capability of the test system.

To show how Calculation Channels can be employed to virtually simulate hybrid tests, a real hybrid test example is presented. In this example, a test system similar to the one shown in Fig. 4 was used for a quasi-static hybrid simulation. This system controls three translational and three rotational Degrees Of Freedom (DOF). For each step, the FEA model (OpenSees model) sends DOF displacement and rotation commands to the test system and then collects DOF forces and torques as feedback signals. Due to swivel friction, measured actuator forces (and therefore the resulting DOF forces and torques) changed abruptly when the actuators reversed their direction of movement during the actual hybrid simulation. This sudden change caused the hybrid test to go unstable in one or two of the rotational degrees of freedom.

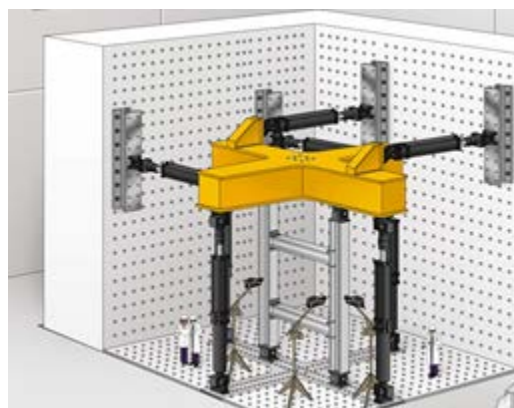


Fig. 4 – Multi-Axial Sub-assembly Test (MAST) system

To determine the root cause of this issue, a virtual test system was setup within the MTS Flextest 793 software using the Calculation Channel functionality to describe the relationship between the output of the control loop and global displacements, as well as the relationship between specimen deformation and force. To model the control loop output and global displacement relationship, a characterization test was conducted to better understand the system delay, square wave response, and sine wave response. Based upon the characterization test data, a filter was created to mimic the loop output to the DOF displacement/rotation relationship for each control channel. Sensor noise was also added to the displacement feedback using a high frequency signal with constant amplitude. Fig. 5 shows the setup of the physical and virtual hybrid test. Fig. 6 shows the Calculation Editor Window of the Flextest 793 software.

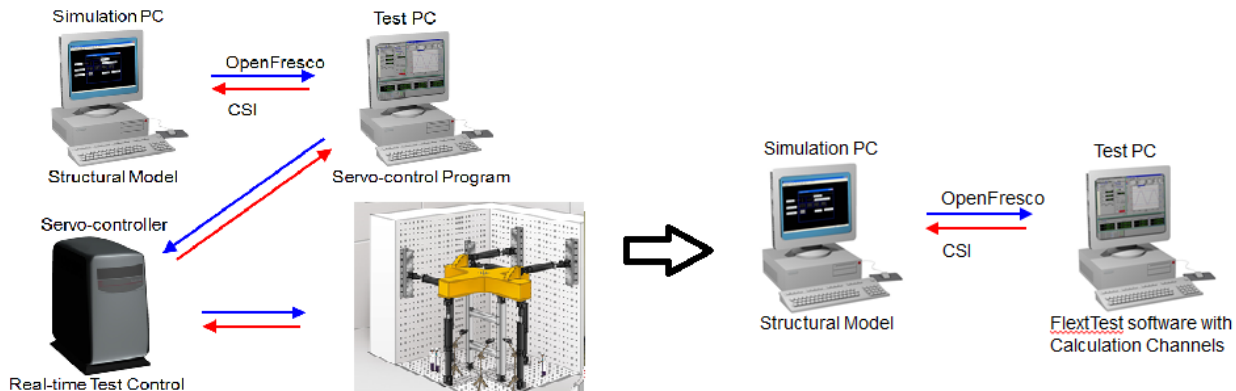


Fig. 5 – System setup of the physical and virtual hybrid test

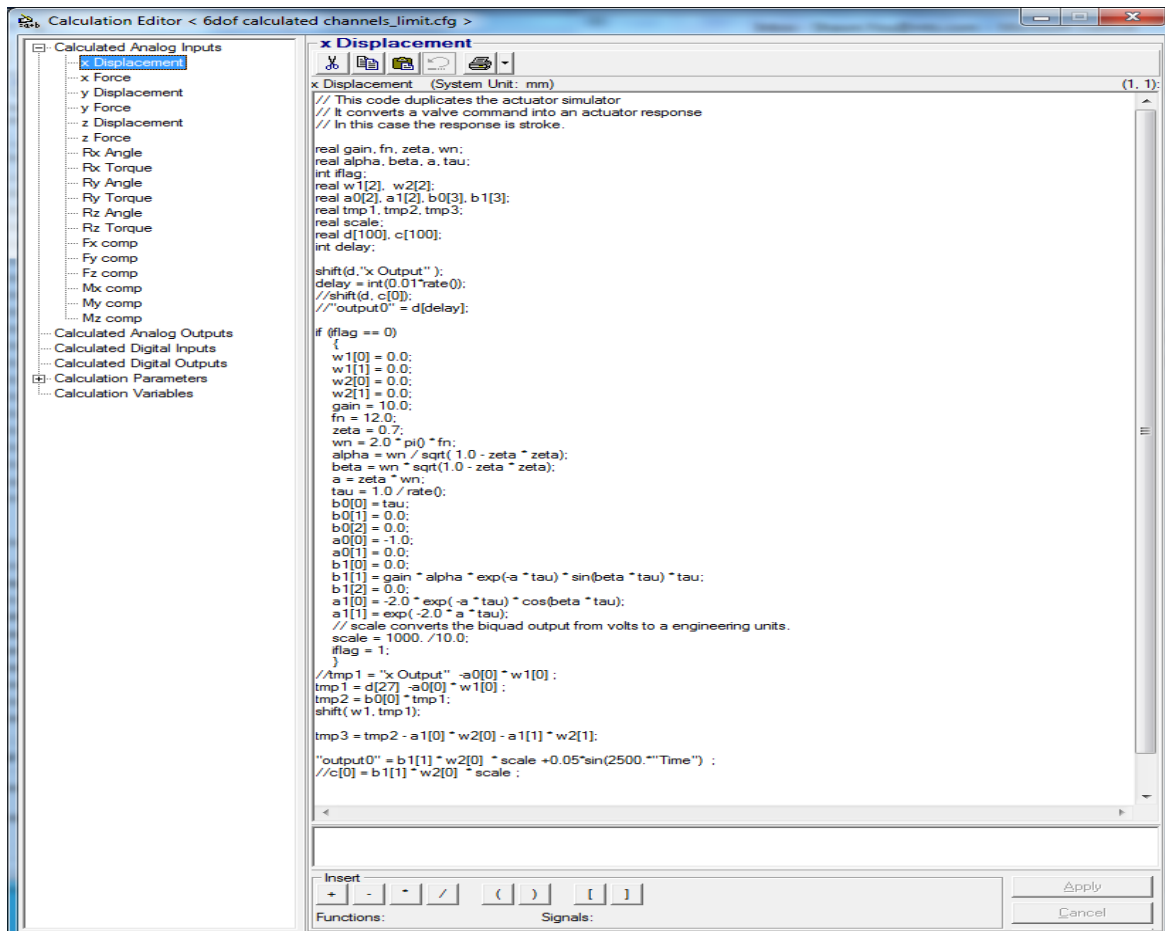


Fig. 6 – Calculation Editor Window of the Flextest 793 software



The relationship between specimen force and deformation was modeled by stiffness information from the test specimen. Constant linear-elastic stiffness values were assumed in order to simplify the implementation. Swivel friction and sensor noise were also modeled using the Calculation Channel functionality.

This virtual system was connected to the OpenSees model through the MTSCi (MTS Computer Simulation Interface) experimental control object available in OpenFresco similar to the real hybrid simulation [1, 2]. By using this virtual test system, the hybrid model was subjected to the same earthquake excitation that was selected for the actual physical hybrid test. The stability issue that was observed in the real hybrid test was successfully reproduced with this test rehearsal. Afterwards, what-if studies were conducted to find the root cause of the stability issue. Parameters defining system delay, friction level, noise level, and controller tuning were changed to find out the effect on system stability. At the end, it was found that the sharp changes of the two moment signals due to swivel friction caused the stability issue.

To solve this stability issue, different compensation techniques were investigated using this virtual test system. A response compensation method based on the rate of change of the measured force feedback was found to be very effective in stabilizing the hybrid simulation. This method was then employed in the real hybrid test and the stability issue was resolved.

#### 4. Using Simulink to Model the Test System

When the property of the physical specimen is rate dependent, hybrid simulations need to be executed in real time. In this case, the analysis model or a real-time coordinator model [12, 13, 14, 15] needs to be converted to C++ code and downloaded into a target PC. A low-latency network (such as SCRAMNet) is typically used to transfer data between real-time machines. Real-time hybrid simulations (RTHS) usually involve fast and powerful actuators and high speed loading profiles. Hence, it is very important to rehearse this kind of hybrid test to make sure that it will be smooth and stable.

To show how this approach works, the previous hybrid test using the MAST test system was also used as an example. In the real test, a Predictor-Corrector algorithm, running on the target PC, synchronized test time and simulation time to ensure real-time execution of the hybrid test. The displacement/rotation commands generated by the OpenSees model were sent to the Predictor-Corrector algorithm through OpenFresco. Using interpolation and extrapolation, commands were sent to the controller in real-time. Because of actuator capacity limits, the earthquake was significantly slowed down so that the response could catch up with the command. Fig. 7 shows the hybrid test setup.

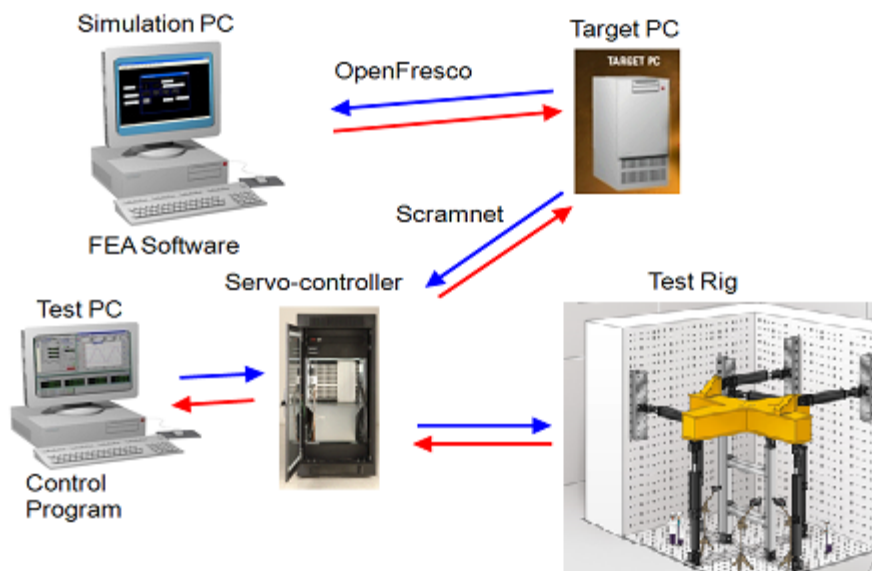


Fig. 7 – Fast hybrid simulation test setup

To simulate this test, a virtual Flextest station was created. This station still received displacement commands from the Predictor-Corrector algorithm, similar to the real hybrid test station. However, the force feedback signals that were sent back to the FEA model were no longer acquired from the force transducers. Instead, they were calculated from a Simulink model of the test specimen. When calculating the feedback signals, the force- displacement relationship of the specimen, system delay, and sensor noise were modeled. Fig. 8 shows the Simulink model that contains the Predictor-Corrector algorithm and the virtual test specimen.

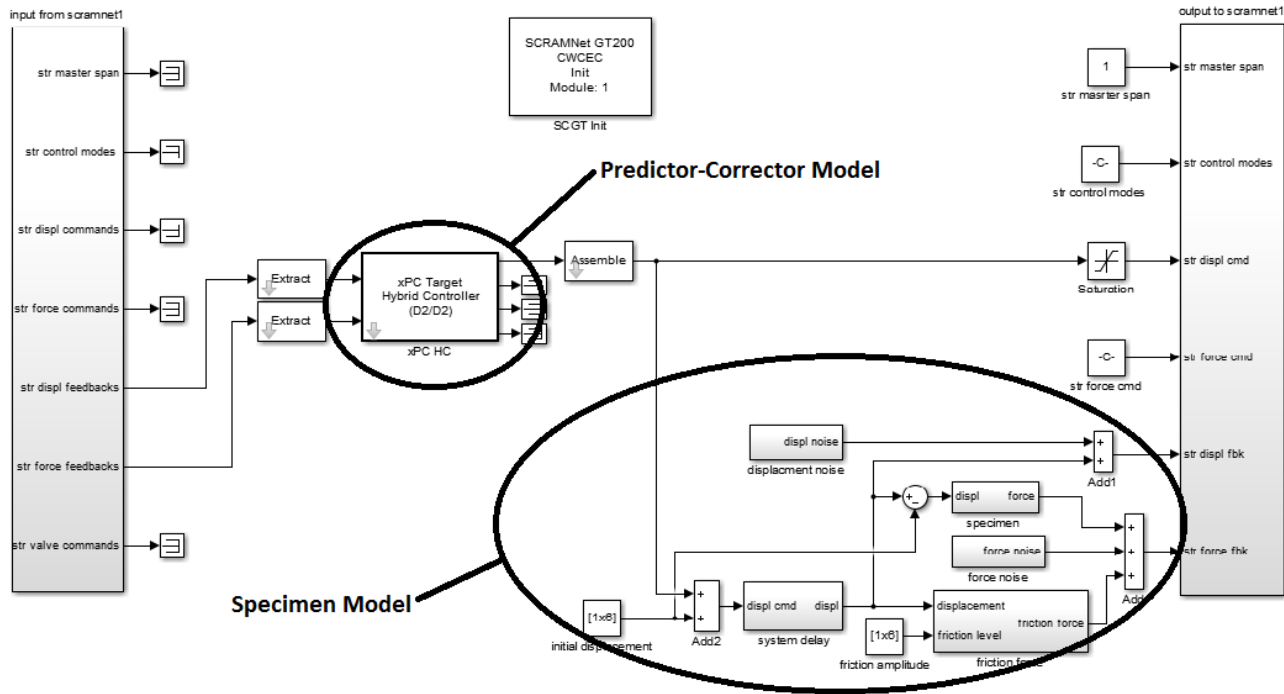


Fig. 8 – Simulink model including Predictor-Corrector algorithm and virtual test specimen

By using this virtual test system, the hybrid test explained in the previous section was simulated again. The stability issue that was observed in the real hybrid test was also duplicated with this test rehearsal. After applying the same response compensation technique mentioned in the previous section, the stability issue was again resolved.

The advantage of this method is that it can simulate real-time hybrid tests because the target PC and the SCRAMNet network are involved. It is much easier to model the test system in Simulink than in the previously described Calculation Editor where it is necessary to write code in the special syntax. The disadvantage of this test rehearsal method is that it requires access to a target PC and Simulink.

## 5. Conclusions

Like most experimental testing methods, hybrid simulations can be challenging and risky. There are many factors that can lead to inaccurate results or possibly even cause damage to the test specimen. Hence, it is essential to rehearse the hybrid simulation first to verify model and software setups, understand testing challenges, and determine appropriate solutions before conducting the real test. By using the proposed test rehearsal methods researchers are able to increase the overall success rate of hybrid simulations.

This paper presented three virtual testing methods that can be utilized to rehearse hybrid simulations. The first method is based on the co-simulation approach where the test specimen is modeled and analyzed in a slave FEA program. The hybrid simulation driver running in the master program exchanges data with the specimen in the slave program through the adapter element. This method can be used to rehearse quasi-static as well as real-time hybrid simulations, but it cannot include effects due to control system and sensor dynamics. The second method is to use the Calculation Channel functionality of the MTS Flextest software. This method does not need



additional software or hardware. It is very useful for simulating slow hybrid tests. The disadvantage is that the virtual test station needs to be programmed using specific syntax. The third method uses Simulink to create the virtual test system. Since Simulink is a powerful modeling tool, it is relatively easy to create the virtual station. With a target PC and a SCRAMNet network, the Simulink model can be run in real-time. Therefore, it is an effective way to rehearse real-time hybrid simulations.

## 7. References

- [1] Schellenberg A, Huang Y, Mahin SA (2008): Structural FE-Software Coupling through the Experimental Software Framework, OpenFresco. *Proceedings of the 14th World Conference on Earthquake Engineering*, Beijing, China.
- [2] Schellenberg A, Mahin SA, Fenves GL (2007): A Software Framework for Hybrid Simulation of Large Structural Systems. *Proceedings of the ASCE Structures Congress*, Long Beach, CA, USA.
- [3] Schellenberg AH, Mahin SA, Fenves GL (2009): Advanced Implementation of Hybrid Simulation. *Pacific Earthquake Engineering Research Center*, Report No. PEER 2009/104, Berkeley, CA, USA.
- [4] Lee SH, Schellenberg AH, Mahin S, Choi CS (2014): Real-Time Hybrid Simulation Test to Assess Performance of Seismically Isolated Nuclear Power Facilities. *Technical Innovation in Nuclear Civil Engineering – TINCE 2014*.
- [5] Schellenberg AH, et al. (2015): Hybrid Simulation of Seismic Isolation Systems Applied to an APR-1400 Nuclear Power Plant. *Pacific Earthquake Engineering Research Center*, Report No. PEER 2015/05, Berkeley, CA, USA.
- [6] Sarebanha A, Schellenberg A, Schoettler M, Mosqueda G, Mahin S (2017): Hybrid Simulation of Full-Scale Seismic Isolation Bearings Using the Caltrans SRMD Testing System. *Proceedings of the 16th World Conference on Earthquake Engineering*, Santiago, Chile.
- [7] McKenna FT (1997): Object-oriented finite element programming: frameworks for analysis, algorithms and parallel computing. *Ph.D. Thesis, University of California*, Berkeley, CA, USA.
- [8] McKenna F, Fenves GL (2007): Using the OpenSees Interpreter on Parallel Computers. *NEESit*.
- [9] Kumar M, Whittaker AS, Constantinou M (2014): An advanced numerical model of elastomeric seismic isolation bearings. *Earthquake Engineering & Structural Dynamics*, 43(13), 1955–1974.
- [10] Nhan DD, Ryan LK (2012): Seismic Response of a Full-scale 5-story Steel Frame Building Isolated by Triple Pendulum Bearings under 3D Excitations. *Ph.D. Thesis, University of Nevada*, Reno, NV, USA.
- [11] Schellenberg A, Becker TC, Mahin, SA (2014): Development of a Large Scale Hybrid Shake Table and Application to Testing a Friction Slider Isolated System. *Proceedings of the 10th U.S. National Conference on Earthquake Engineering*, Anchorage, AK, USA.
- [12] Wu B, Shi P, Wang Q, Guan X and Ou J (2011): Performance of an offshore platform with MR dampers subjected to ice and earthquake. *Struct. Control Health Monit*, 18, 682–697
- [13] Jung RY, Shing PB (2006): Performance evaluation of a real-time pseudodynamic test system. *Earthq Eng Struct Dyn*, 35(7), 789–810.
- [14] Chen C, Ricles JP, Entwistle J (2009): Analysis of actuator delay compensation methods for real-time testing. *Engineering Structures*, 31, 2643–2655.
- [15] Nakashima M, Kato H, Takaoka E (1992): Development of real-time pseudo-dynamic testing. *Earthquake Engineering and Structural Dynamics*, 21, 79–92.