Implementation issues in distributed hybrid simulation *

Stathis Bousias, Oh-Sung Kwon, Nikos Evangeliou, Anastasios Sextos and Olympia Taskari

Abstract— The attractiveness of the pseudodynamic testing method has triggered its continuous development into the form known as hybrid simulation. This evolution has been accelerated by the increasing interest focusing on the advantages of substructured testing for the seismic evaluation of structural systems. Nonetheless, the main feature of hybrid simulation, i.e. networking capabilities, is no available in many old controllers operated in structural laboratories. The research performed focused on this issue, employing two approaches for carrying out hybrid simulation of a three-span bridge. Several issues related to the implementation of the approaches are stressed and experimental results are presented and commented.

I. INTRODUCTION

Hybrid simulation is a rapidly expanding testing method owing to its versatility, capacity to cope with member, subassembly or full-scale structural testing (pseudodynamic or real-time) and its flexibility in combining the experimental/computational potential of individual laboratories. In all cases, a simulation-controlling component ("coordinator") is entrusted with the time-stepping integration algorithm and the coordination of the communication between substructures. Following the terminology recently established [Nakata et al., 2014], two configurations may be employed: the coordinating component is a designated software (e.g. *SimCor [1], [3]*) which relies on external finite element codes and physical testing for the numerical and the experimental substructures, respectively, while fully undertaking the task of communicating the deformation vector to all substructures and receiving the respective measured deformation/resistance vectors from them. The intense network communication is particularly problematic for structures with many substructures and/or degrees-of-freedom and may even lead to process halting. A second approach is that adopted in OpenFresco/Opensees platform [4]: the analysis of the numerical substructures is performed within the finite element software and the only network communication required is that with the laboratory-tested component(s). This feature is particularly advantageous for the hybrid simulation of structures with large number of DOFs, as it keeps network communication to the minimum.

Regardless of the selection of the simulation-controlling component, what is ultimately required is that computed deformations at physical-numerical substructure boundaries be imposed on the structure. This paper focuses on this very issue, i.e. the options available for realizing a two-way communication path between the simulation coordination software and the control system employed in the laboratory in which the physical component resides. It is noted that this issue is not only present when geographically distributed hybrid simulation is performed, but also when both the analytical and the experimental components of the simulation are executed within the same laboratory. Two options are examined for resolving the problem – a hardware/software and a software-only based approach. The former is a rather generic one and uses a feature available in most control systems, even old ones, i.e. that of receiving an external analog signal as the signal to command the actuators with. In recent hybrid simulations, controller boards (or analog I/O boards) from dSPACE or National Instruments were used. The alternative approach is entirely software based and is tailored to a specific controller used in the host laboratory - the controller was developed by JRC and employs the dual-RAM approach [5]. Along the same line, some of MTS controllers have enabled real-time communication functionality through SCRAMNET. Both approaches were implemented for the assessment of the response of a three-span RC bridge, discretized into five modules and with an elastomeric isolator being the only physically tested component.

II. SCHEMES FOR COMMUNICATION WITH THE CONTROLLERS

Network communication between the coordinating software and each substructure is a vital issue for hybrid simulation. While such communication presents no difficulties when it concerns the numerical tools performing the analysis for the emulated substructures, when it comes to the analogous communication to the hardware of the experimental components many difficulties are met. The standard approach adopted in hybrid simulation is that computed displacements become target values for the controller managing the experimental substructures, when they are passed (in digital form) to the actuator-controlling software via network. Due to the variety of controllers met in structural testing laboratories, no uniform solution exists and, unless a modern controller with networking capabilities is used (termed later on as "controller-specific" procedure, Fig. 1(a)), ad-hoc solutions have to be devised and implemented. The latter approach (termed later on as "analog-input")

*Support provided by 7th Framework Programme of European Commission - PIRSES-GA-2009-247567-EXCHANGE-SSI.

S. N. Bousias is with the Dept. of Civil Eng., University of Patras, 26500, GREECE (e-mail: sbousiar@upatras.gr).

O-S Kwon is with the Dept. of Civil Eng., University of Toronto, CANADA (e-mail: os.kwon@utoronto.ca).

N. Evangeliou is with the Dept. of Electrical and Computer Eng., University of Patras, 26500, GREECE (e-mail: evangeliou@ece.upatras.gr).

A. Sextos is with the Dept. of Civil Eng., Aristotle University of Thessaloniki, GREECE (e-mail: asextos@civil.auth.gr).

O. Taskari is with the Dept. of Civil Engineering, Aristotle University of Thessaloniki, GREECE (e-mail: taskari@civil.auth.gr).

procedure, Fig. 1(b)), however, except for the practical difficulties entailed, it poses certain concerns related to the features of the intervening software, the accuracy of measured deformations and restoring forces fed back to the coordinating component and the associated latency with which reference deformations are applied. The above play a dominant role in the overall reliability and accuracy of hybrid simulation. In the following, two approaches are presented regarding communication of designated coordinating software (*SimCor*, in this case) with the experimental components of the hybrid simulation; results are compared and commented with particular emphasis on issues affecting the quality of hybrid simulation.



Figure 1. Generalc configuration of the communication schemes examined: (a) controller-specific scheme, (b) analog-input scheme

A. Controller-specific procedure

As a first approach the host controller incorporating network communication capabilities and supported by the associated software was selected, as the least difficult case for application of hybrid simulation. The only concern is then related to the fact that this controller is enclosed within the local laboratory network as a security measure against possible risks from exposing it the public network (through which the reference signals are expected during hybrid simulation). Thus, any scheme to implement hybrid simulation should deal with the problem of riskless introduction of reference signals from the public to the local network.

The architecture of the control system employed in the host laboratory is shown in Fig. 1(a): it is composed of the main computer (master) with a number of controller slave boards (maximum 4 per master controller) and equal number of control units linked one-to-one to the slave boards in master. The main computer operates under a real-time operating system. Each slave control board is equipped with a dual port RAM and includes three main onboard components (signal processing, data acquisition and data exchange) with the capacity to control the motion of the associated actuator in real-time (time sampling of 1 or 2ms). The control units function independently while communicating with the master via high-speed synchronous parallel communication. The control units can directly receive exernal analog signals for controlling the respective actuators – this option was used for the implementation of the analog-input approach presented in the following. The control software reflects the architecture of the hardware (Fig. 2): there is one master program that communicates with several slave programs. The master application uses memory blocks in the dual RAM to update reference values destined for each control unit – any updating of the reference signal is instantly seen by the control unit which then drives the respective actuator to apply the required displacement (or force) - this feature is similar to SCRAMNET. An application (*RemoteControl*) - included in the controller software - allows to view and modify the real-time system internal database through a DCOM object (Distributed Component Object Model - proprietary technology for communication among software components distributed across networked computers.).

Based on the latter feature of the system software, a parenthetic application (StrulabAPI) was built in Matlab, to communicate with the master application (Fig. 2), making it thus possible to access and modify the internal memory blocks seen by the slave controllers and, ultimately, by the control units. This parenthetic application runs on a machine on the public network, but has to communicate with both the remote server running SimCor and the master computer in the laboratory. To accommodate these needs the machine is equipped with two network cards: one belonging to the local laboratory network and the other in the public network. The application receives (in digital form) the reference signals (displacements or forces) from SimCor through the public network and accesses, through the local network, the controller memory blocks to update command displacements. This latter is facilitated by the *RemoteControl* application. Reference signals may previously undergo modifications within the application: after been mapped to the appropriate control unit and actuator, they are scaled (if appropriate) and transformed to the respective signals along actuator coordinate system (if necessary). The controller application is undertaking the task of displacement-ramp generation for the actuators to apply command signals – the ramp is based on a maximum velocity of 20mm/sec with minimum ramp time of 100ms and a fixed hold time of 50ms, while another 20ms buffer-time is added by the network card of the master controller. In addition, it offers other necessary services (management of alarms, offset, etc.), so no attention for treating these issues is necessary in developing the StrulabAPI script. Displacement and force values measured from the physical component are converted in digital form and stored in the dual RAM; they are thus made directly available to the Matlab application (again through RemoteControl) for processing (scaling, transformation to the global coordinate system) and delivery to SimCor.



Figure 2. Control system software architecture

B. Analog-input procedure

This approach is based on the fact that the majority of controllers employed in structural testing support the option of accepting an external analog signal as the reference signal to be applied. Usually, this signal represents the updated displacement signal to be applied and, although force may also in general be considered as the controlled physical quantity, displacement control will be assumed in the following, without violating the generality of the approach. Also, the simulation coordination platform *SimCor* has been used for the work presented herein, as it allows the desired flexibility and can easily cope with the selected test structure (with few degrees of freedom, [6]).

Fig. 3 presents a schematic of the overall architecture: reference displacements resulting from the analysis performed at the node running *SimCor* are communicated via network (TCPIP protocol) to the local host machine running a parenthetic script in Labview (referred to as Network Interface for Controllers – *NICON* – and developed at the University of Toronto [2], [13]). Except for communicating with *SimCor*, the parenthetic script is furnished with other functionalities, such as ramp generation, displacement-limit and force-limit checks, accommodation of any initial displacement and force offsets, and the capability to perform coordinate system transformations. The script, based on a preset value of velocity, creates a ramp of appropriate displacement increments directed to a D/A (digital-to-analog) converter. The resulting voltage signal is hard-wired to the controller as an analog-input signal for driving the actuator under the inner PID-control loop. To minimize the duration of displacement application, while maintaining equal velocity per step, a velocity value close-to-maximum system velocity was used for determining the duration of each ramp. In the opposite direction, i.e. when measured quantities (displacement and force) are to be conveyed to *SimCor*, these are first digitized in the A/D (analog-to-digital) converter and, through *NICON*, are sent to *SimCor* for the integration procedure to advance.



Figure 3. Layout of the analog-input approach

The scheme above is a flexible approach, unaffected by the lack of network communication characteristics of old controllers, while allowing to perform hybrid simulation at minor additional cost for the (off-the-self) additional hardware. However, it involves successive DAC/ADC operations, which may introduce noise to both the reference and measured quantities. The use of 16-bit A/D-D/A cards, as well as the introduction of low-pass filters can decrease signal distortion (at the expense of introducing some lag). Nevertheless, the latter is not very important as long as no hard real-time testing is performed.

Despite how appealing the approach may be, it should be noted that, depending on the controller used, it may completely by-pass all or some of the controller software functionalities (except those of the inner control loop) and consequently, limits related to hardware (displacement, force), initial offsets, data validation, ramp generation, start/stop/pause options, graphics associated with control quantities, signal filtering, control error, network communication and others are no longer provided. In

that case, any attempt to implement hybrid simulation via the "analog-input" approach relies on implementing all the necessary functionalities in the parenthetic software operating in tandem to the selected analog input/output hardware.

A common advantage in the methods employed is that, because they are based on the development of parenthetic software, they allow for eventual transformations (correction, modification or treatment) of input/output signals to be included - this is usually not allowed by standard controller software (especially in older controller versions). In addition, if multiple actuators are used to control a multi-degree of freedom systems, the parenthetic software can take care of the conversion from the numerical model's coordinate system to actuator's strokes.

III. TEST STRUCTURE

The two approaches described above were implemented and employed for the hybrid simulation of an actual structure, a three-span, 99m-long reinforced concrete bridge. The structure was sub-structured (Fig. 4) into five modules (deck, left pier, right pier, left bearing, right bearing) with the elastomeric bearing on the left support being experimentally tested (Fig. 5(a)) – more details and results on the test performed can be found in [10].



Figure 4. Hybrid simulation scheme for the substructured bridge [10]





Figure 5. Test execution (a) test setup at local site, (b) remote-site screen (coordination-site)

(a)

The structure considered was subjected to the acceleration record from the El Centro event. Fig. 6 compares the displacement (Fig. 6(a)) and the force-displacement responses (Fig. 6(b)) obtained from each approach: the "analog-in" (*NICON*) and the Matlab script (*StrulabAPI*). It is shown that displacements obtained by the two approaches practically coincide and force-displacement loops compare very well. However, what is not depicted in these figures is that steps are completed faster in the "analog-input" (*NICON*) approach. Fig. 7 compares the total time required by each approach to complete each step (including the numerical integration, forward/backward communication to/from modules, analysis of numerical substructures and displacement application in the experimental substructure), showing a slight advantage of the "analog-input" approach (*NICON*) over the controller-specific (*StrulabAPI*) one. This lead is the result of a number of reasons: first, *NICON* is a Labview-based script and is thus a multi-thread application with all the advantages resulting from this. System time (elapsed time) is better estimated in *NICON* as timing of signals is assigned when the respective value is available in the memory. In contrary, Matlab-based *StrulabAPI* script determines time in a serial manner, i.e. signals are time-stamped when the software proceeds to signal saving, making *NICON* a possibly better candidate when it comes to the hybrid simulation of MDOF structures.



Figure 6. Experimental component: (a) displacement, and (b) force-displacement response

The second cause of time-step lead of "analog-input" approach, and maybe the most influential one, lies in the fact that the machine running the StrulabAPI operates two network cards – except for the doubling of the network latency time per step, the operating software needs also to identify the appropriate network card to communicate with at the beginning and the end of each step (it initially communicates to the public network to receive reference signals, then to the local network to make the signals available to the controller software and, finally, back to the public network to supply restoring forces to SimCor). It should be mentioned that because NICON is in control of displacement ramp generation, the hold period and the stp duration can be decreased considerably

The time required for the complete execution of an accelerogram ranges between 0.5 and 4 secs, which implies that the simulation is performed at an expanded time scale of 50 to 400 times that of the actual excitation. For rate sensitive devices this time expansion is of course not acceptable – in the present case, though, the elastomeric bearing tested bear low effect of the rate of deformation o their properties. It is mentioned that, although real-time applications can be developed for the

experimental substructure based on analog I/O methods or Real-Time Target (instead of Matlab), this might not be very advantageous for the overall test velocity as most numerical simulation software cannot perform analysis within theninelastic range in-real time, anyway. Real-time simulation is applicable for mostly elastic systems with very well-defined structural components such as elastic building with MR damper, base-isolation system, etc. Consequently a simpler approach was adopted her for the specific case of low-damping elastomeric devices used: a special procedure that has been examined in the past [12] has been employed: measured forces undergo modification before been communicated to the coordinating software, to compensate for the otherwise neglected rate-effects. This underlines also the issue that in many testing campaigns in general, it is required that certain modifications/calculations have to be performed before/after the signals are passed to/received from the controller – the existence of a external script with which these operations can be realized outside the controller, is very important and t is fulfilled in both approaches examined herein.

Fig. 8 depicts the per step time used exclusively by the experimental sub-structure: it includes the time upon receiving the reference signals until communicating the measured displacements and forces back to the simulation coordination. It is realized that for the selected structure and sub-structures, the experimental part is by far the major contributor to the overall per step duration. The difference, i.e. the time for all rest substructures (difference of signals shown in Fig. 8 from those shown in Fig. 7), is presented in Fig. 9, showing that the approaches are almost equivalent, with the exception of some random cases in which the "analog-input" approach shows unexpected delays – the issue is still under investigation. However, and setting aside the long duration of the experimental part, it is seen that the rest processes require more than 0.5secs representing a time scale expansion of 50 if compared to real-time. Obviously, real-time hybrid simulation cannot be performed with the tools employed in this study.



Figure 7. Total time duration for step completion



Figure 8. Duration of time step in the experimental sub-structure



Figure 9. Duration of time step, excluding the time for the experimental sub-structure

IV. CONCLUSION

Two approaches for implementing hybrid simulation in different control systems, has been presented. In the first, a fully featured controller was employed, while in the second the "analog-input" approach was selected. Hybrid simulation of a three-span bridge composed of 5 sub-structures, showed that satisfactory results can be obtained by both approaches, at the expense of building parenthetic software. However, this software should be supplied with controller-related features because the use of "analog-input" signals approach bypasses, in this case, the associated software. The fact that no appreciable precedence of one approach to the other was recognized implies that for laboratories equipped with controllers lacking networking features, there exists a viable solution for performing hybrid simulation.

ACKNOWLEDGMENT

The work carried out was funded by the 7th Framework Programme of the European Commission, under the PIRSES-GA-2009-247567-EXCHANGE-SSI grant (Experimental & Computational Hybrid Assessment Network for Ground-Motion Excited Soil-Structure Interaction Systems, <u>www.exchange-ssi.net</u>). Partial support to the first author was also provided by the European Union (European Social Fund – ESF) and Greek national funds through the Operational Program "Education and Lifelong Learning" of the National Strategic Reference Framework (NSRF) – Research Funding Program THALES - Investing in knowledge society through the European Social Fund.

REFERENCES

- [1] O.-S. Kwon, N. Nakata, A. Elnashai, and B. Spencer, "A framework for multi-site distributed simulation," J. Earthq. Eng., vol. 9, no. 5, pp. 741–753, 2005.
- [2] Kammula, V., Erochko, J., Kwon, O., and Christopoulos, C. "Application of hybrid-simulation to fragility assessment of the telescoping self-centering energy dissipative bracing system" Earthquake Engineering and Structural Dynamics, 2013, DOI: 10.1002/eqe.2374P.
- [3] Kwon, O., Elnashai, A. S., and Spencer, B. F. (2008). "A framework for distributed analytical and hybrid simulations." Structural Engineering and Mechanics, 30(3), 331–350.
- [4] Schellenberg, A. H., Mahin, S. A., and Fenves, G. L. (2009). Advanced Implementation of Hybrid Simulation. PEER Report 2009/104.
- [5] Pegon, P., Molina, F. J. and Magonette, G. [2008] "Continuous pseudo-dynamic testing at ELSA" in Hybrid Simulation, Theory, Implementation and Applications, ed. Saouma V.E. and Sivaselvan M. V. (Taylor & Francis/Balkema) 79-88.
- [6] Nakata, N, Dyke, S., Zhang, J., Mosqueda, G., Shao, X., Mahmoud, H., Head, M., Erwin M., Bletzinger, M., Marshall, G.A., Ou, G. and Song, C. Hybrid Simulation Primer and Dictionary", Network for Earthquake Engineering Simulation, NEES, 2014.
- [7] Taskari O. N. and Sextos, A. G. "Robustness, repeatability and resilience of intercontinental distributed computing for the purpose of seismic assessment of bridges", Computational Methods in Structural Dynamics and Earthquake Engineering (COMPDYN2013), Kos, June 2013.
- [8] Sextos, A. and Taskari, O. Single and multi-platform simulation of linear and non-linear bridge-soil systems", in Proceedings of the NATO Advanced Research Workshop on Coupled Site and Soil-Structure Interaction Effects with Application to Seismic Risk Mitigation, Tom Schanz and Roumen lankov (Eds), NATO Science for Peace Program. 2008.
- [9] Kwon, O-S., Elnashai, A. S., Spencer, B. F. and Park, K. "UI-SIMCOR : A global platform for hybrid distributed simulation," in 9th Canadian Conference on Earthquake Engineering, June, 2007, pp. 139–149.
- [10] Sextos, A., Bousias, S., Taskari, O., Evangeliou, N., Kwon, O-S., Elnashai, A., DiSarno, L. and Palios, X. Än intercontinental hybrid simulation experiment for the purposes of seismic assessment of a three-span R/C bridge", 10th US National Conf. on Earthquake Eng., Anchorage Alaska, 2014.
- [11] Molina, F. J., Verzeletti, G., Magonette, G., Buchet, P., Renda, V., Geradin, M., Parducci, A., Mezzi, M., Pacchiarotti, A., Frederici L., and Mascelloni S. (2002) "Pseudodynamic tests on rubber base isolators with numerical substructuring of the superstructure and the strain-rate effect compensation", Earthquake Engng Struct. Dyn, 31:1563-1582.
- [12] Palios, X., Molina, J., Bousias, S., Strepelias H. and Fardis, M. (2007) "Sub-structured pseudodynamic testing of rate-dependent bridge isolation devices", 2nd Int. Conf in Advances in Experimental Structural Engineering, Tongji Univ., Shanghai, China, Dec.4-6.
- [13] Kammula, V., Erochko, J., Kwon, O., and Christopoulos, C. (2012) "Performance assessment of the self centering energy dissipative (SCED) braced system using hybrid simulation" 15th World Conference on Earthquake Engineering, Lisbon, 2012.