

Application of mode superposition to hybrid simulation using real time finite element method

W. MUCHA*, W. KUŚ**

*Institute of Computational Mechanics and Engineering, Silesian University of Technology, ul. Konarskiego 18A, 44-100 Gliwice, Poland, E-mail: waldemar.mucha@polsl.pl

**Institute of Computational Mechanics and Engineering, Silesian University of Technology, ul. Konarskiego 18A, 44-100 Gliwice, Poland, E-mail: waclaw.kus@polsl.pl

crossref <http://dx.doi.org/10.5755/j01.mech.23.5.14642>

1. Introduction

Hybrid simulation is a method for investigating dynamic material and structural properties of mechanical systems by performing an experiment and a numerical analysis simultaneously. It applies to mechanical systems with components that are difficult to model numerically. The essence of the method is building two models that are dependent on each other: analytical model and experimental model. The components that are difficult to model numerically are physically attached to actuators and measuring devices and consist the experimental model. The analytical model is the numerical representation of the rest of the tested mechanical system. The two built models closely cooperate during the simulation. The aim is that the physical components are strained exactly as they would be as the part of the whole system. The general algorithm of hybrid simulation is presented in Fig. 1 [1–3].

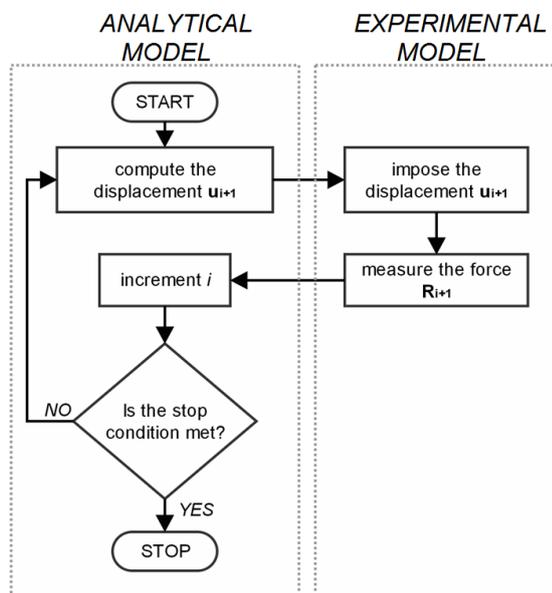


Fig. 1 General algorithm of hybrid simulation

Finite Element Method (FEM) is used in the following paper for building the numerical representation for its great advantages like versatility [4, 5], adaptivity, stability and flexibility (possibility to model inhomogeneity, complex geometries and complex boundary conditions) [6].

Because the actuators that strain the physical components have to be controlled in a closed loop in real

time (in order to obtain reliable results), all the FEM computations in the analytical model must also be performed in real time. Therefore Real Time Finite Element Method (RTFEM) is used to simulate the non-physical part of the tested system. As FEM is very computationally demanding method, using it in real time is not an easy challenge [7, 8]. In dynamic computations which are used in hybrid simulation, consecutive states of the system are calculated with a constant time step. It is obvious that the time of computations for a single step should not exceed the value of the time step itself, otherwise there is no real time. The value of the time step should be adjusted to the time of computations, however setting too big time step causes decrease of accuracy [3, 6]. Therefore hybrid simulations with complex finite element models require introducing techniques for speeding-up computations.

The following paper focuses on such a technique - mode superposition. Mode superposition introduced to the analytical model enables to significantly reduce the model order and, consequently the time step of real-time computations with only unnoticeable decrease of accuracy (if the reduction is performed correctly).

2. Mode superposition

Introducing mode superposition to the analytical model of the hybrid simulation consists of transforming the finite element displacement coordinates to the modal coordinates. In order to transform coordinates, free vibration mode shapes need to be used. Therefore the procedure requires the solution of the generalized eigenproblem [6, 9]. Solving it is a computationally demanding task, however the most important thing is that, when some requirements are met, the eigenproblem can be solved offline, before performing the hybrid simulation. That way the computation time of solving the eigenproblem does not matter, because its results can increase the speed of real time computations during hybrid simulation.

The abovementioned requirements for introducing hybrid simulation as described is that the part of the tested mechanical system that constitute the analytical model must be linear with proportional damping. It is a significant limitation for using mode superposition in hybrid simulation, however for models that meet the requirements, the results are very satisfying.

The model order reduction is achieved because not all mode shapes have to be used to transform the displacement coordinates to modal coordinates. Good approximate solutions can be obtained by superposition with only

first few free vibration mode shapes [6, 10].

If the tested mechanical system was modeled using FEM, the general equation of motion for time step i would be in form:

$$\mathbf{M}\ddot{\mathbf{u}}^i + \mathbf{C}\dot{\mathbf{u}}^i + \mathbf{K}\mathbf{u}^i = \mathbf{F}^i, \quad (1)$$

where \mathbf{M} is the mass matrix; \mathbf{C} is the damping matrix; \mathbf{K} is the stiffness matrix; \mathbf{u}^i is the displacement vector and \mathbf{F}^i is the excitation force vector. The superscript i denotes the time dependence [3, 10].

Considering that the mechanical system is divided into analytical model and experimental model, Eq. (1) can be partitioned in the following way:

$$(\mathbf{M}_A + \mathbf{M}_E)\ddot{\mathbf{u}}^i + (\mathbf{C}_A + \mathbf{C}_E)\dot{\mathbf{u}}^i + (\mathbf{K}_A + \mathbf{K}_E)\mathbf{u}^i = \mathbf{F}^i, \quad (2)$$

where the matrices with subscript A represent the properties of the analytical part and those with subscript E the experimental part of the tested system. As the forces related to the experimental model are measured during the simulation, the Eq. (2) can be rewritten in a more applicable form [3, 10]:

$$\mathbf{M}_A\ddot{\mathbf{u}}^i + \mathbf{C}_A\dot{\mathbf{u}}^i + \mathbf{K}_A\mathbf{u}^i = \mathbf{F}^i - \mathbf{R}_E^i, \quad (3)$$

where:

$$\mathbf{R}_E^i = \mathbf{M}_E\ddot{\mathbf{u}}^i + \mathbf{C}_E\dot{\mathbf{u}}^i + \mathbf{K}_E\mathbf{u}^i. \quad (4)$$

The eigenproblem that must be solved offline takes the form:

$$(\mathbf{K}_A + \mathbf{K}_E^0)\boldsymbol{\Phi} = \boldsymbol{\omega}^2(\mathbf{M}_A + \mathbf{M}_E)\boldsymbol{\Phi}, \quad (5)$$

where the solution is diagonal matrix $\boldsymbol{\omega}$ of eigenvalues and square matrix $\boldsymbol{\Phi}$ of vertical eigenvectors. \mathbf{K}_E^0 is the initial stiffness matrix of the experimental part. For most tested mechanical systems where the experimental part is significantly smaller than the analytical part, this approximation (solving eigenproblem offline with assumed constant stiffness) is usually acceptable, however in other cases the mode superposition may cause significant errors, which cannot be avoided as determining the stiffness matrix of the experimental part every iteration can be difficult or even impossible and, most of all, the necessity of solving eigenproblem every iteration misses the whole point of using mode superposition in hybrid simulation.

If the level of approximation is denoted as α , $\boldsymbol{\Phi}_\alpha$ will be a matrix built of α first columns of matrix $\boldsymbol{\Phi}$.

The equation of motion (5) can be now reduced to only α DOFs:

$$\hat{\mathbf{M}}_A\ddot{\mathbf{x}}^i + \hat{\mathbf{C}}_A\dot{\mathbf{x}}^i + \hat{\mathbf{K}}_A\mathbf{x}^i = \hat{\mathbf{F}}^i - \hat{\mathbf{R}}_E^i, \quad (6)$$

where:

$$\begin{cases} \hat{\mathbf{M}}_A = \boldsymbol{\Phi}_\alpha^T \mathbf{M}_A \boldsymbol{\Phi}_\alpha; \\ \hat{\mathbf{C}}_A = \boldsymbol{\Phi}_\alpha^T \mathbf{C}_A \boldsymbol{\Phi}_\alpha; \\ \hat{\mathbf{K}}_A = \boldsymbol{\Phi}_\alpha^T \mathbf{K}_A \boldsymbol{\Phi}_\alpha; \\ \hat{\mathbf{F}}^i = \boldsymbol{\Phi}_\alpha^T \mathbf{F}^i; \\ \hat{\mathbf{R}}_E^i = \boldsymbol{\Phi}_\alpha^T \mathbf{R}_E^i. \end{cases} \quad (7)$$

In order to solve the reduced FEM equation of motion (6) step by step, explicit or implicit integration schemes can be utilized. Both approaches vary in difficulty and application possibilities a lot, therefore both are discussed in detail below.

3. Explicit scheme

Using Central Difference Method [6, 10] one can obtain the displacement for the time step $i+1$ by solving the Eq. (6) in the following form:

$$\hat{\mathbf{M}}_{eff} \mathbf{x}^{i+1} = \mathbf{B}^i, \quad (8)$$

where (if Δt denotes the timestep):

$$\begin{cases} \hat{\mathbf{M}}_{eff} = \frac{1}{\Delta t^2} \hat{\mathbf{M}}_A + \frac{1}{2\Delta t} \hat{\mathbf{C}}_A; \\ \mathbf{B}^i = \hat{\mathbf{F}}_A^i - \hat{\mathbf{R}}_E^i - \left(\hat{\mathbf{K}}_A - \frac{2}{\Delta t^2} \hat{\mathbf{M}}_A \right) \mathbf{x}^i - \\ - \left(\frac{1}{\Delta t^2} \hat{\mathbf{M}}_A - \frac{1}{2\Delta t} \hat{\mathbf{C}}_A \right) \mathbf{x}^{i-1}. \end{cases} \quad (9)$$

Only the right-hand side vector of the Eq. (8) must be computed every iteration, the main matrix – the effective stiffness matrix – can be calculated before iterations. To calculate the displacement in the first time step using Eq. (8), one needs to compute a fictitious vector \mathbf{x}_A^{i-1} :

$$\mathbf{x}_A^{i-1} = \left(\mathbf{x}_A^0 - \Delta t \dot{\mathbf{x}}_A^0 + \frac{\Delta t^2}{2} \ddot{\mathbf{x}}_A^0 \right), \quad (10)$$

where the modal coordinates initial conditions are calculated from the known initial conditions of the analytical part (displacement and velocity; acceleration is calculated from equation of motion (6)):

$$\begin{cases} \mathbf{x}_A^0 = \boldsymbol{\Phi}^T \mathbf{M}_{AA} \mathbf{u}_A^0; \\ \dot{\mathbf{x}}_A^0 = \boldsymbol{\Phi}^T \mathbf{M}_{AA} \dot{\mathbf{u}}_A^0; \\ \ddot{\mathbf{x}}_A^0 = \hat{\mathbf{F}}_A^0 - \hat{\mathbf{R}}_{EA}^0 - \hat{\mathbf{C}}_{AA} \dot{\mathbf{x}}_A^0 + \hat{\mathbf{K}}_{AA} \mathbf{x}_A^0. \end{cases} \quad (11)$$

The initial conditions of the experimental part should be imposed at the beginning of the hybrid simulation. In most cases of hybrid simulation all the initial conditions are equal to zero. It may be difficult to set nonzero initial conditions other than displacement in the experimental model.

In order to impose the displacement on the experimental model, at the end of each iteration a reverse transformation from modal coordinates should be performed:

$$\mathbf{u}^{i+1} = \boldsymbol{\Phi}_\alpha \mathbf{x}^{i+1}. \quad (12)$$

In the experimental model the displacement should be imposed on the boundary nodes and the forces should be also measured in these points. When there is an external force applied to the experimental model, it also must be taken into account. There is also possibility of

imposing displacements and measuring forces in less points than boundary nodes (if the loss of accuracy is acceptable), however, in such case, the values must be mathematically transformed to the boundary nodes.

The stability condition for the Central Difference Method described above, is given below:

$$\Delta t_{max} = \frac{T_\alpha}{\pi}, \quad (13)$$

where T_α means the oscillation period of the highest free vibration frequency used in the approximation. Eq. (13) determines the highest possible time step for the solution to be convergent. This is the biggest drawback of utilizing explicit schemes in hybrid simulation, because often (especially for system with many DOFs) the maximum time step is so low, that it makes real-time computations impossible as the computation time is much bigger. However when introducing modal superposition to the algorithm and approximating the solution with only first few vibration mode shapes, the maximum time step may increase enough to enable real-time computations.

Time steps when using explicit schemes are usually low enough that provide smooth and continuous paths of the actuators when updating the displacements values every time step [3, 10]. If not, one should introduce extrapolation and interpolation techniques described in more detail in the implicit scheme section.

4. Implicit scheme

Implicit integration schemes for solving reduced equation of motion step-by-step in hybrid simulation should only be used when explicit scheme cannot be utilized due to too strict stability condition. For calculating the next state $i + 1$ implicit schemes, unlike explicit schemes, require not only known data from current step i but also data from step $i + 1$, which in hybrid simulation cannot be completely known yet. Therefore the biggest drawback of implicit schemes in hybrid simulation is the necessity for numerical modeling the whole tested system, including the experimental part, in order to predict the forces that will be developed in time step $i + 1$ before it comes. This enforces an iterative solution procedure. On the other hand, the main advantage of implicit (over explicit) integration schemes is that many of them are unconditionally stable [3, 6, 10].

The Newmark algorithm with constant-average-acceleration [3, 6, 9, 10] is used for solving Eq. (6) step-by-step. For iterative approach (where k is the iteration number), Eq. (6) takes the form:

$$\hat{\mathbf{K}}_{eff} \Delta \mathbf{x}_{(k)}^{i+1} = \mathbf{B}_{(k)}^{i+1}, \quad (14)$$

where

$$\begin{cases} \hat{\mathbf{K}}_{eff} = \frac{4}{\Delta t^2} \mathbf{I} + \frac{2}{\Delta t} (\hat{\mathbf{C}}_A + \hat{\mathbf{C}}_E) + (\mathbf{K}_A + \mathbf{K}_E^0); \\ \mathbf{B}_{(k)}^{i+1} = \hat{\mathbf{F}}^{i+1} - \hat{\mathbf{M}}_A \left[\frac{4}{\Delta t^2} (\tilde{\mathbf{x}}_{(k)}^{i+1} - \mathbf{x}^i) - \frac{4}{\Delta t} \dot{\mathbf{x}}^i - \ddot{\mathbf{x}}^i \right] - \\ - \hat{\mathbf{C}}_A \left[\frac{2}{\Delta t} (\tilde{\mathbf{x}}_{(k)}^{i+1} - \mathbf{x}^i) - \dot{\mathbf{x}}^i \right] - \hat{\mathbf{K}}_A \tilde{\mathbf{x}}_{(k)}^{i+1} - \tilde{\mathbf{R}}_{E(k)}^{i+1}. \end{cases} \quad (15)$$

In every time step there is a constant number of iterations, in order to make real time possible. The number of iterations in a single time step should be adjusted for the solution to be convergent every time step. $\tilde{\mathbf{x}}_k^{i+1}$ is the predicted displacement for time step $i + 1$. For the first iteration it should be predicted by extrapolation from previous time steps (quadratic extrapolation is usually sufficient), for further iterations it is calculated from simple formula: $\tilde{\mathbf{x}}_k^{i+1} = \mathbf{x}^i + \Delta \mathbf{x}_{k-1}^{i+1}$. $\tilde{\mathbf{R}}_{E(k)}^{i+1}$ is the predicted force developed in the physical part in time step $i + 1$:

$$\tilde{\mathbf{R}}_{E(k)}^{i+1} = \hat{\mathbf{R}}_E^i + \hat{\mathbf{K}}_{E\,eff} (\tilde{\mathbf{x}}_k^{i+1} - \mathbf{x}^i), \quad (16)$$

where $\hat{\mathbf{K}}_{E\,eff}$ is the portion of the effective stiffness matrix described in Eq. (15) that contain only the properties of the experimental part. $\hat{\mathbf{R}}_E^i$ is the vector of measured forces in the previous iteration. As one can see, the forces must be measured in every node of the experimental model, or measured in smaller number of points and transformed to finite element nodes. This is another drawback of implicit integration schemes (in explicit schemes forces need to only be measured on boundary nodes of analytical and experimental part).

After last iteration (n), the state for time step $i + 1$ is obtained as follows:

$$\begin{cases} \mathbf{x}^{i+1} = \mathbf{x}^i + \Delta \mathbf{x}_n^{i+1}; \\ \dot{\mathbf{x}}^{i+1} = \frac{2}{\Delta t} (\mathbf{x}^{i+1} - \mathbf{x}^i) - \dot{\mathbf{x}}^i; \\ \ddot{\mathbf{x}}^{i+1} = \frac{4}{\Delta t^2} (\mathbf{x}^{i+1} - \mathbf{x}^i) - \frac{4}{\Delta t} \dot{\mathbf{x}}^i - \ddot{\mathbf{x}}^i. \end{cases} \quad (17)$$

In order to impose the displacement on the experimental model, at the end of each iteration a reverse transformation from modal coordinates should be performed exactly as in explicit schemes – Eq. (12).

As the described above procedure is computationally demanding (especially for models with many DOFs), the time steps are significantly greater than in explicit schemes, therefore sending displacement commands to actuators every time step may be insufficient for providing smooth and continuous path. In such case extrapolation and interpolation techniques should be utilized [10]. When performing computations in time step i for time step $i + 1$, the displacements values are not known yet (the computations are not finished), the displacements at the boundary nodes should be extrapolated for the time step $i + 1$ and displacement command sent to actuators should lie on the extrapolation curve. The moment the computations are done, an interpolation should be performed in order to replace the extrapolation curve [3]. The subsequent approximations of the solution for time step $i + 1$ in the Newmark iterative process can also be used in this process to generate more accurate commands for the actuators. To perform the signal generation task in the same time as performing the FEM computations, multicore approach is very helpful.

5. Numerical example 1

An aluminum truss (Fig. 2, dimensions in [cm], cross section 0,004 [m²]) with 49 DOFs is considered as the first numerical example. Analytical part consists of every member except *MN*, which is nonlinear and represents the experimental part. In the following example an external function simulates the force that is developed in the experimental model in response to imposed displacement (nonlinear elasticity). The aim of the example is to verify the described explicit scheme.

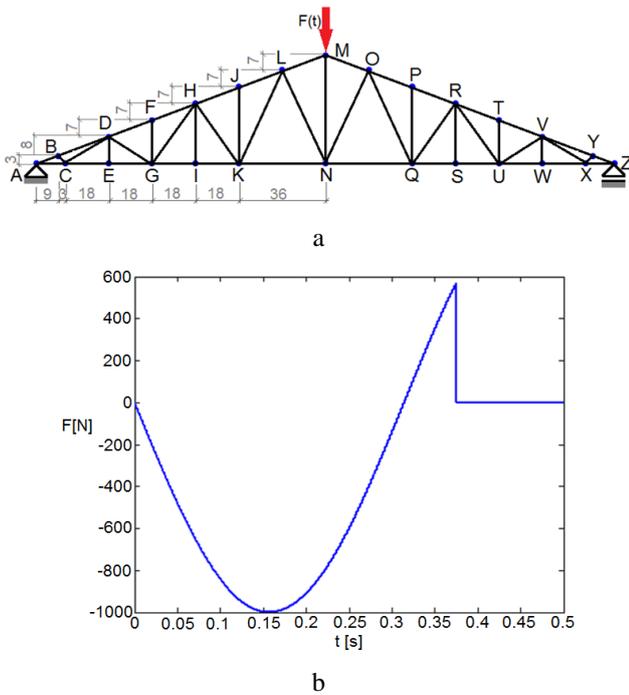


Fig. 2 Numerical example 1: a - model of the truss; b - excitation force $F(t)$

Fig. 3 presents the vertical displacement of node *N* during the simulation: the exact solution (without mode superposition) and the approximated solutions (with superposition, from 1 to 5 first modes).

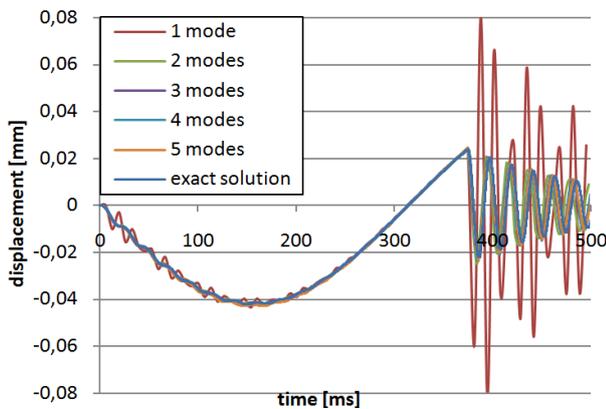


Fig. 3 Vertical displacement of node *N*—explicit scheme

Time steps for 6 different variants of the algorithm (exact solution and approximation for $\alpha = \{1, \dots, 5\}$) are presented in Table 1. As one can see, for the applied load, mode superposition using only 2 first free vibration mode shapes provided very good accuracy while allowing

to increase the time step 75 times and reducing the equation of motion from 49 to only 2 DOFs.

Table 1

Time step for different variants of the algorithm

| α | 1 | 2 | 3 | 4 | 5 | exact |
|-----------------|------|------|------|------|------|-------|
| Δt , ms | 6.70 | 3.00 | 1.66 | 1.34 | 0.92 | 0.04 |

This model has also been tested on a real-time operating system. The algorithm has been implemented to microcontroller: National Instruments myRIO (Processor Xilinx Z-7010, 667 Mhz, 2 cores; 512 MB DDR3 RAM, NI Linux Real-Time OS). The response of the experimental model was simulated using another microcontroller – devices were connected via analog inputs/outputs. myRIO was programmed in LabVIEW. For deterministic tasks (reading response of the experimental model from analog input, performing FEM computations, writing displacement to analog output) timed loop was used, all the matrices required for solving Eq. (8) were imported from MATLAB. Hybrid simulation without modal superposition has not been possible – the microcontroller could not perform the required computations in real time. However introducing mode superposition enabled real time computations – only if $\alpha \leq 4$. For load case as presented in Fig. 2, b, the results were matching those presented in Fig. 3.

6. Numerical example 2

The following numerical example is the same truss loaded the same way as in the previous numerical example (Fig. 2), but this time it is solved using an implicit integration scheme. The time step was set at 5 [ms] and the task was solved without modal superposition (understood here as the exact solution) and with modal superposition (5 approximated solutions, from 1 to 5 first modes applied).

The vertical displacement of node *N* is presented in Fig. 4 and computation times (time of performing computations for all 100 time steps, in MATLAB, processor Intel Core i3 540 3,07 GHz, 4GB RAM DDR3, average of 10 measurements) for each variant of the algorithm are compared in Table 2.

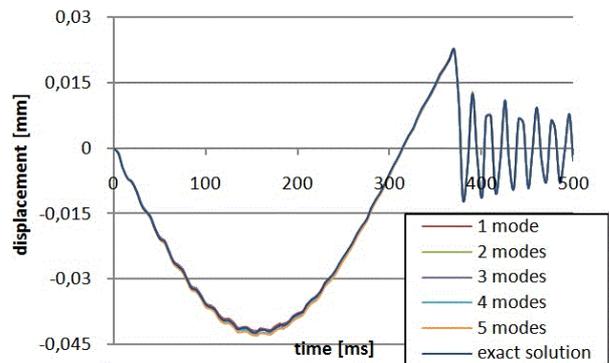


Fig. 4 Vertical displacement of node *N* - implicit scheme

Table 2

Computation time for different variants of the algorithm

| α | 1 | 2 | 3 | 4 | 5 | exact |
|----------|------|------|------|------|------|--------|
| t , ms | 0.52 | 0.82 | 1.24 | 1.30 | 1.79 | 112.45 |

For the quite large time step that has been set, the results for exact and approximated solutions are nearly identical. Therefore approximation $\alpha = 1$ for the applied load is sufficient and allows to reduce the computation time 216 times. For smaller time steps or different load-case, the level of approximation $\alpha = 1$ may not be sufficient, however even $\alpha = 5$ reduces the computation time over 6 times.

This model has also been tested on real-time operating system, with implemented implicit integration scheme. The hardware was identical as described in the first numerical example. Hybrid simulation of the presented truss without mode superposition required time step 26 [ms] (for two iterations per time step). By introducing mode superposition the time step could be decreased to 1.5 [ms] (the same for $\alpha = 1$ and $\alpha = 5$, as more consuming was the reverse transformation of displacements each iteration).

7. Conclusions

Presented algorithms for introducing mode superposition to hybrid simulation were verified and validated by the numerical examples.

They allow to significantly decrease the computation time for each time step. The first algorithm may allow to use explicit schemes in hybrid simulation, as normally the required time step is so small, that real-time computations are impossible – proven by the first numerical example. The second algorithm may significantly reduce the time step when using implicit integration schemes (which, in general, improves accuracy) – proven by the second numerical example.

The requirements for implementing the presented algorithms are strict (linear analytical part, proportional damping, necessity of at least rough estimation of stiffness of the experimental part), however if met, the results can be satisfying.

Acknowledgement

The research is partially financed from project BKM-547/RMT-4/2015 (10/040/BKM_15/0016).

References

1. **Bursi, O.S.** 2008. Computational Techniques for Simulation of Monolithic and Heterogeneous Structural Dynamic Systems. Modern Testing Techniques for Structural Systems. O.S. Bursi & D. Wagg (eds) 1-96: SpringerWienNewYork.
2. **Nakashima, M.; McCormick, J.; Wang T.** 2008. Hybrid simulation: A historical perspective. Hybrid Simulation: Theory, Implementation and Applications. V. Saouma & M. Sevaselan (eds), Taylor & Francis Group: 3-14.
3. **Mucha, W.** 2016. Real-time hybrid simulation using materials testing machine and FEM. Advances in mechanics: theoretical, computational and interdisciplinary issues. CRC Press/Balkema: 419-422. <https://doi.org/10.1201/b20057-90>.
4. **Ogierman, W.; Kokot G.** 2016. A study on fiber orientation influence on the mechanical response of a short fiber composite structure, Acta Mechanica 227(1): 173-183. <https://doi.org/10.1007/s00707-015-1417-0>.
5. **Dziatkiewicz, G.** 2016. Complex variable step method for sensitivity analysis of effective properties in multi-field micromechanics, Acta Mechanica 227(1): 11-28. <https://doi.org/10.1007/s00707-015-1419-y>.
6. **Zienkiewicz, O.C.; Taylor R.L.** 2000. The Finite Element Method, Vol. 1: The Basis, fifth ed. Butterworth-Heinemann, Oxford.
7. **Bro-Nielsen, M.; Cotin, S.** 1996. Real-time volumetric deformable models for surgery simulation using finite elements and condensation, Computer Graphics Forum 15(3): 57-66. <https://doi.org/10.1111/1467-8659.1530057>.
8. **Rock, S.; Pritschow, G.** 2007. Real-time capable Finite Element Models with closed-loop control: a method for Hardware-in-the-Loop simulation of flexible systems, Prod. Eng. Res. Devel. 1: 37-43. <https://doi.org/10.1007/s11740-007-0020-1>.
9. **Bathe, K.; Gracewski, S.** 1981. On nonlinear dynamic analysis using substructuring and mode superposition, Computers & Structures 13(5-6): 699-707. [https://doi.org/10.1016/0045-7949\(81\)90032-8](https://doi.org/10.1016/0045-7949(81)90032-8).
10. **Shing, P.B.** 2008. Integration schemes for real-time hybrid testing. Hybrid Simulation: Theory, Implementation and Applications. V. Saouma & M. Sevaselan (eds). Taylor & Francis Group, London: 25-34.

W. Mucha, W. Kuś

APPLICATION OF MODE SUPERPOSITION TO HYBRID SIMULATION USING REAL TIME FINITE ELEMENT METHOD

S u m m a r y

The following paper presents methods for application of mode superposition to the analytical part of the hybrid simulation, when Real Time Finite Element Method is used to model the analytical substructure. Introducing mode superposition can bring significant computation time decrease in real time computations, by performing some calculations offline. The paper describes requirements, methods and algorithms for introducing mode superposition to explicit and implicit integration schemes. In explicit schemes mode superposition allows to increase the minimum time step often enabling real time computations that were not possible before. In implicit schemes mode superposition allows to reduce the time step of the analysis (and therefore increase accuracy). Numerical examples proving the effectiveness of the presented algorithms are given.

Keywords: hybrid simulation, mode superposition, real time finite element method.

Received April 09, 2016

Accepted October 13, 2017