COMPDYN 2013
4th ECCOMAS Thematic Conference on
Computational Methods in Structural Dynamics and Earthquake Engineering
M. Papadrakakis, V. Papadopoulos, V. Plevris (eds.)
Kos Island, Greece, 12–14 June 2013

EXTERNAL COUPLING OF FE CODES FOR STRUCTURAL DYNAMICS

M. Brun[1]*, A. Gravouil[2], A. Combescure[2], A. Limam[1]

[1]INSA-Lyon, Laboratoire de Génie Civil en Ingénierie Environnementale, Bât. Coulomb, 34 avenue des Arts, 69621, Villeurbanne, France; Michael.brun@insa-lyon.fr

[2] INSA-Lyon, Laboratoire de Mécanique des Contacts et des Structures, Bât. Jean d'Alembert, 18-20 rue des Science, 69621, Villeurbanne, France

Keywords: Subdomain decomposition, Structural Dynamics, Coupling software, multi-time step explicit/implicit co-computations.

Abstract. *Two external coupling software, whose purpose is enabling to conduct multi-time step explicit/implicit co-computations for structural dynamics problems, are set up. They are based on the coupling algorithms proposed by Gravouil and Combescure (GC method) and Prakash and Hjelmstad (PH method). The salient features of the multi-time step partitioning methods are presented: they involve non-overlapping partitions, follow a dual Schur approach by ensuring the velocity continuity at the interface with Lagrange multipliers. The main difference between the two methods lies in the time scale at which the interface problem is solved: micro time scale for the GC-algorithm and macro time scale for the PH-algorithm. During the multi-time step co-computations involving two Finite Element codes (explicit and implicit FE codes), the tasks carried out by the coupling software PH-CPL, based on a variant of the PH algorithm, are illustrated and compared to the coupling software GC-CPL based on the GC-algorithm. The advantage of the new coupling software PH-CPL is highlighted in terms of parallel capabilities. In addition, the coupling software PH-CPL alleviates the dissipative drawback of the GC method at the interface between the subdomains. Academic cases are investigated to check the accuracy order for the GC and PH algorithms. Finally, explicit/implicit multi-time step co-computations with GC-CPL and PH-CPL software are conducted under the assumption of linear elastic material for a reinforced concrete frame structure under blast loading striking its front face.*

1 INTRODUCTION

The increasing complexity of numerical models for engineering systems, seeking to account for an ever finer mesh, accurate material models and multiphysics phenomena with very different space and time scales, boosted the development of partitioning approaches. In structural dynamics such as a structure subjected to localised impact loads, a small part of the structure is expected to experience strong non linear phenomena on a short duration with high frequency features, whereas the remaining of the structure would be mainly concerned with linear low-frequency vibration on a much longer duration. In this case, the choice of the subdomain decomposition of the whole mesh is driven by computational consideration: taking different time-step and time-integration schemes (explicit or implicit) depending on the parts of the structure is much more interesting in terms of computation times than a unique time-step imposed by the smallest element of the mesh in the case of conditionally stable time-integration schemes (explicit).

The GC method proposed by Combescure and Gravouil [1] has been built in the framework of the FETI method proposed by Farhat and Roux [2-3] in the beginning of the nineties. The kinematic continuity of quantities at the interface is prescribed by means of Lagrange multipliers. The GC method requires a three step resolution: one solves first the free solution on each subdomain, then one ensures the velocity continuity at the interface of the subdomains through the resolution of a small interface problem at the fine time scale, and finally one computes a linked solution on each subdomain resulting from the interface forces expressed in terms of the previous Lagrange multipliers. The authors showed that the stability of the subdivided problem is ensured for any Newmark time-integration schemes by prescribing the velocity continuity at the interface: using the energy method (Hughes [4]), the authors also proved that the multi-time heterogeneous partitioning solution is stable as soon as the time step size satisfies the stability criterion of each subdomain. Nonetheless, a small amount of energy is dissipated at the interface when very different time steps are considered. As a result, when using Newmark time-integration schemes of the second order accuracy, the accuracy order of the coupling method is equal to two in the case of the same time step, and is reduced by one order in the case of different time steps. Furthermore, the GC method proposed for Newmark schemes in linear dynamics has been extended to non linear dynamics (Gravouil and Combescure [5]), explicit non linear dynamics with non matching meshes at the interface (Herry *et al.* [6]), and coupling of subdomains described according to a modal approach (Faucher *et al.* [7]).

More recently, Prakash and Hjelmstad [8] dealt with the energy dissipation problem at the interface by improving the GC method. The main difference lies in the time scale at which the interface problem is solved: contrary to the GC method whose interface problem is solved at the fine time scale, the interface problem is considered at the large time scale. The coupling scheme turned out to be stable and non dissipative at the interface in the sense that the interface pseudo-energy involved in the energy method remains equal to zero. The partitioning method remains of second order accuracy when Newmark second order accurate time integration schemes are considered. A comparison of coupling GC and PH algorithms with heterogeneous time integration schemes has been conducted by Mahjoubi *et al.* [9]. One must mention here that the kinematic continuity is not ensured in this case for each small time step.

The heterogeneous time integrators multi-time step algorithm, labelled as the GC method proposed by Gravouil and Combescure, is first reminded before presenting the Prakash and Hjelmstad method. A coupling software is set up on the basis of the PH algorithm and is validated for a split oscillator and a reinforced structure under blast loading.

2 COUPLING CONTINUOUS IN TIME SUBDOMAINS

Both GC and PH coupling methods are based on a Dual Schur complement approach by imposing the continuity of velocities across the interface between subdomains by means of Lagrange multipliers. The Lagrange multipliers represent the interface forces between the two subdomains Ω_A and Ω_B. The continuous in time kinematic constraint related to the velocity continuity across the interface is expressed as:

$$L_A \dot{u}^A(t) + L_B \dot{u}^B(t) = 0 \tag{1}$$

where L_A and L_B are Boolean connectivity matrices for matching meshes at the interface. Velocity constraint at the interface is ensured by means of Lagrange multipliers $\Lambda(t)$. Therefore, the semi-discretized equilibrium equation (discrete in space and continuous in time) in the case of linear elastic material can be written as:

$$\begin{cases} M_A \ddot{u}^A(t) + K_A u^A(t) = f_{ext}^A(t) - L_A^T \Lambda(t) \\ M_B \ddot{u}^B(t) + K_B u^B(t) = f_{ext}^B(t) - L_B^T \Lambda(t) \end{cases} \tag{1.1}$$

where M is the symmetric definite-positive mass matrix, K is the semi-definite positive stiffness matrix and $u(t)$ denotes the nodal displacements. A superposed dot over a quantity denotes its time derivative. The subscripts A and B are used for denoting the subdomains for the matrix quantities, whereas the superscripts A and B are used for vector quantities.
On the right side of the above equations, the interface forces, $-L_A^T \Lambda(t)$ and $-L_B^T \Lambda(t)$, acting on subdomains Ω_A and Ω_B, are defined by the product of the transpose of the connectivity matrices with the Lagrange multipliers $\Lambda(t)$. It can be noted that they act on a given subdomain as a field of external forces.

3 COUPLING DISCRETE IN TIME SUBDOMAINS ACCORDING TO THE GC METHOD

Let us now consider two time integration schemes for the subdomains Ω_A and Ω_B from the Newmark family, characterised by the parameters γ_A , γ_B and β_A , β_B. One of the main interests of the subdomain decomposition method is to deal with a problem exhibiting very different time scales, by choosing appropriate time steps depending on subdomains. Thus, two time scales are considered, a macro time scale with the size $h_A = [t_0, t_m]$ for the subdomain Ω_A and a micro time scale with the size $h_B = [t_{j-1}, t_j]$ for the subdomain Ω_B. The discretized equilibrium equation becomes at the end of the macro time-step and at the end of the micro time-steps:

$$\begin{cases} M_A a_m^A + K_A u_m^A = f_{ext,m}^A - L_A^T \Lambda_m \\ M_B a_j^B + K_B u_j^B = f_{ext,j}^B - L_B^T \Lambda_j \ , \ j = 1, \ldots, m \end{cases} \tag{1.2}$$

The discrete continuity on velocities is imposed at each micro time t_j :

$$L_A v_j^A + L_B v_j^B = 0 \ , \ \ j = 1, \ldots, m \tag{1.3}$$

where velocities v_j^A at the micro times t_j are linearly interpolated from the velocities at the beginning and at the end of the macro time step $h_A = [t_0, t_m]$, namely v_0^A and v_m^A.
By introducing the approximate formula for Newmark schemes, the system of equilibrium equations over the macro time step can be rewritten as:

$$\begin{cases} \tilde{M}_A \, a_m^A = f_{ext,m}^A - K_A \, {}^p u_0^A - L_A^T \Lambda_m \\ \tilde{M}_B \, a_j^B = f_{ext,j}^B - K_B \, {}^p u_{j-1}^B - L_B^T \Lambda_j \ , \ j = 1, \ldots, m \end{cases} \tag{1.4}$$

where the predictors ${}^p u_0^A$ and ${}^p u_{j-1}^B$ depending only on quantities known at the beginning of the macro and micro time steps, respectively. The dynamic operators at the left hand side are defined by:

$$\begin{cases} \tilde{M}_A = M_A + \beta_A h_A^2 K_A \\ \tilde{M}_B = M_B + \beta_B h_B^2 K_B \end{cases} \tag{1.5}$$

The splitting procedure proposed by Combescure and Gravouil for Newmark time integration schemes consists in splitting the kinematic quantities (displacements, velocities, and accelerations) into free and linked quantities. A linear assumption over the macro time step for the velocities in the subdomain Ω_A is required in order to express the velocity continuity at the micro times t_j as a small-size symmetric dual problem in the form:

$$H_{GC} \, \Lambda_j = b_j \tag{1.6}$$

where the left hand side matrix H_{GC} denotes the interface operator for the GC method, and the right hand side vector b_j is only dependent on free velocities in subdomains Ω_A and Ω_B. They are defined by:

$$\begin{cases} H_{GC} = \gamma_A h_A L_A \tilde{M}_A^{-1} L_A^T + \gamma_B h_B L_B \tilde{M}_B^{-1} L_B^T \\ b_j = L_A v_{free,j}^A + L_B v_{free,j}^B \end{cases} \tag{1.7}$$

The above interface problem in (1.6) has to be solved at all the micro times t_j so as to derive the Lagrange multipliers Λ_j and then the linked quantities at the micro time scale.
It has to be underlined that the GC method is completely staggered: macro subdomain first for free quantities followed by micro subdomain for free quantities and again, macro subdomain first for linked quantities followed by micro subdomain for linked quantities. Thus, the free and link computations cannot be done in a concurrent way for both subdomains. The Prakash and Hjelmstad algorithm provides an improvement in terms of parallel capabilities with respect to its GC progenitor method.

4 COUPLING DISCRETE IN TIME SUBDOMAINS ACCORDING TO THE PH METHOD

The method proposed by Prakash and Hjelmstad [8] permits to eliminate the interface pseudo-energy in the sense of the energy method (Hughes, [4]). The key difference from the GC method is the enforcement of the velocity continuity by Lagrange multipliers at the macro-time scale rather than at the micro-time scale as done in the GC method. The discrete velocity continuity at the end of the macro time step (at time t_m) becomes:

$$L_A v_m^A + L_B v_m^B = \mathbf{0} \tag{1.8}$$

As for the GC method, the system of equilibrium equations for both subdomains is decomposed into free and linked quantities as follows:

- the free problem:

$$\begin{cases} M_A a_{free,m}^A + K_A u_{free,m}^A = f_{ext,m}^A \\ M_B a_{free,j}^B + K_B u_{free,j}^B = f_{ext,j}^B \ , \ j = 1,\ldots,m \end{cases} \tag{1.9}$$

- the link problem:

$$\begin{cases} M_A a_{link,m}^A + K_A u_{link,m}^A = -L_A^T \Lambda_m \\ M_B a_{link,j}^B + K_B u_{link,j}^B = -L_B^T \Lambda_j \ , \ j = 1,\ldots,m \end{cases} \tag{1.10}$$

In order to express the interface problem, the interpolated quantities in subdomain Ω_A are required at the micro time t_j, written above as:

$$\begin{cases} w_{free,j}^A = \left(\mathbf{1} - \dfrac{j}{m}\right) w_0^A + \left(\dfrac{j}{m}\right) w_{free,m}^A \\ w_{link,j}^A = \left(\dfrac{j}{m}\right) w_{link,m}^A \end{cases} \tag{1.11}$$

where the quantities *w* denote for the displacements, velocities and accelerations in the subdomain Ω_A.

Then, the authors introduced an unbalanced free interface reaction and an unbalanced linked interface reaction defined by:

$$\begin{cases} S_j = L_A \left(f_{ext,j}^A - M_A a_{free,j}^A - K_A u_{free,j}^A \right) \\ T_j = L_A \left(-M_A a_{link,j}^A - K_A u_{link,j}^A \right) - \Lambda_j \end{cases} \tag{1.12}$$

The above interface forces can be viewed as the disequilibrium of the free and link problem for the macro subdomain Ω_A expressed at the micro time scale. This disequilibrium comes from the assumed interpolated quantities from the Eq. (1.11), projected to the interface of the

subdomain Ω_A.

The two disequilibrium terms must be counterbalanced at each micro time t_j as:

$$T_j + S_j = L_A\left(f_{ext,j}^A - M_A a_j^A - K_A u_j^A - L_A^T \Lambda_j \right) = \mathbf{0} \tag{1.13}$$

Using the second equation of (1.11), the projected disequilibrium of the linked quantities can be expressed as:

$$T_j = -\left(\frac{j}{m}\right) L_A\left(M_A a_{link,m}^A + K_A u_{link,m}^A\right) - \Lambda_j \tag{1.14}$$

In addition, incorporating the expression (1.10) for subdomain Ω_A into the above equation, the projected disequilibrium of the linked quantities can be written as:

$$T_j = \left(\frac{j}{m}\right) L_A\left(L_A^T \Lambda_m\right) - \Lambda_j \tag{1.15}$$

And finally, by using Eq. (1.13) at the micro time, a relationship linking the Lagrange multipliers at the micro time scale to the unbalanced free interface forces and to the Lagrange multipliers at the macro time step, is derived as:

$$\Lambda_j = S_j + \left(\frac{j}{m}\right)\Lambda_m \tag{1.16}$$

The above relationship permits to express the Lagrange multipliers Λ_j at the micro time scale as a function of the unbalanced free interface reactions S_j and the Lagrange multipliers Λ_m at the end of the macro-time step. The unbalanced term S_j are first computed by solving the system of free equilibrium equations. Then, the relationship (1.16) is incorporating into the equilibrium equations for the subdomain Ω_B given in Eq. (1.2):

$$M_B\, a_j^B + K_B u_j^B = f_{ext,j}^B - L_B^T S_j - \frac{j}{m} L_B^T \Lambda_m \;\;,\; j = 1,\ldots,m \tag{1.17}$$

Without entering in further details, it can be shown that the global system composed of the equilibrium equation in subdomain Ω_A and the m equilibrium equations (1.17), can be solved in three stages by splitting the equations into free and link problems:

- Solve the global free problem only accounting for the external forces $f_{ext,j}^B$ and the unbalanced terms $-L_B^T S_j$ at each micro time t_j (from $j = \mathbf{1}$ to $j = m$).

- Solve the interface problem derived from the velocity continuity given in (1.8) at the macro-time scale. As showed by the authors [8], the interface problem takes the form:

$$H_{PH}\, \Lambda_m = b_m \tag{1.18}$$

H_{PH} denoting the interface operator at the macro time scale, and the right hand side vector b_m

being only dependent on free velocities at the end of the macro time step:

$$\begin{cases} H_{PH} = L_A y_m^A + L_B y_m^B \\ b_m = L_A v_{free,m}^A + L_B v_{free,m}^B \end{cases} \tag{1.19}$$

The algorithm for building the PH interface operator is shown in Table 1. It depends on the matrices y_m^A and y_m^B, iteratively computed by looping from micro time step to micro time step.

(2A) Macro subdomain SD*A*	(2B) Micro subdomain *SDB*
$y_0^A = 0, \dot{y}_0^A = 0, \ddot{y}_0^A = 0$	$y_0^B = 0, \dot{y}_0^B = 0, \ddot{y}_0^B = 0$
	Loop for j=1…m
${}^p y_0^A = y_0^A + h_A \dot{y}_0^A + \left(\frac{1}{2} - \beta_A\right) h_A^2 \ddot{y}_0^A$	${}^p y_{j-1}^B = y_{j-1}^B + h_B \dot{y}_{j-1}^B + \left(\frac{1}{2} - \beta_B\right) h_B^2 \ddot{y}_{j-1}^B$
${}^p \dot{y}_0^A = \dot{y}_0^A + (1-\gamma_A) h_A \ddot{y}_0^A$	${}^p \dot{y}_{j-1}^B = \dot{y}_{j-1}^B + (1-\gamma_B) h_B \ddot{y}_{j-1}^B$
$\ddot{y}_m^A = \tilde{M}_A^{-1} \left(L_A^T - K_A \, {}^p y_0^A \right)$	$\ddot{y}_j^B = \tilde{M}_B^{-1} \left(\frac{j}{m} L_B^T - K_B \, {}^p y_{j-1}^B \right)$
$\dot{y}_m^A = {}^p \dot{y}_0^A + \gamma_A h_A \ddot{y}_m^A$	$\dot{y}_j^B = {}^p \dot{y}_{j-1}^B + \gamma_B h_B \ddot{y}_j^B$
$y_m^A = {}^p y_0^A + \beta_A h_A^2 \ddot{y}_m^A$	$y_j^B = {}^p y_{j-1}^B + \beta_B h_B^2 \ddot{y}_j^B$
(2') Computation of the interface operator	
$H_{PH} = L_A \dot{y}_m^A + L_B \dot{y}_m^B$	

Table 1: Computation of the interface operator for the PH method.

- Finally, once obtained the Lagrange multipliers Λ_m at the macro time scale, the linked quantities are obtained by solving the link problem only accounting for the terms $-\frac{j}{m} L_B^T \Lambda_m$ in Eq. (1.17).

It is interesting to note that the resolution of the free problem is staggered in the PH method. Indeed, the free quantities of the macro subdomain are first computed, then the unbalanced free interface forces S_j can be derived, and finally, the free quantities of the micro subdomain are obtained, from the first micro time step (end time t_1) to the last micro time step (end time t_m). Contrary to the free problem, the linked quantities for the two subdomains can be computed in parallel. Indeed, as soon as the Lagrange multipliers at the end of the macro time step are known, the linked quantities of both subdomains can be determined in a concurrent way. In comparison to the progenitor GC method which is completely staggered (in free and link computations), the parallelised features of the PH algorithm represent an improvement with respect to the GC method.

5 TWO EXTERNAL COUPLING SOFTWARE FOR EXPLICIT/IMPLICIT CO-COMPUTATIONS

Two external code coupling software have been set up in order to make a first finite element (FE) code based on an implicit time integration interact with a second FE code based on an explicit time integration.

In the following, we focus on the building of the PH coupling software, developed in C language, whereas the GC coupling software, presented in previous papers (Brun *et al.* [10,11]), is briefly reminded for comparison purpose. Two finite element codes are connected to the coupling software: CAST3M (implicit or explicit, [12]) and Europlexus (explicit, [13]).
Communications between processors use the pipe technology, enabling to send or receive data from one process to another process. The data exchange by means of pipes is very fast and does not require memory space in comparison to the classical writing and reading in classical files. Another important advantage in view of synchronizing the computations between the FE codes and the coupling software is the blocking feature of the reading function through a pipe.
The co-computation is carried out between the actors (the coupling software and the two FE codes for subdomains SDA and SDB) over the macro time step h_A. The tasks undertaken by the GC-CPL, that is the GC-based coupling software, are compared with the PH-CPL, that is the PH-based coupling software in Figure 1.

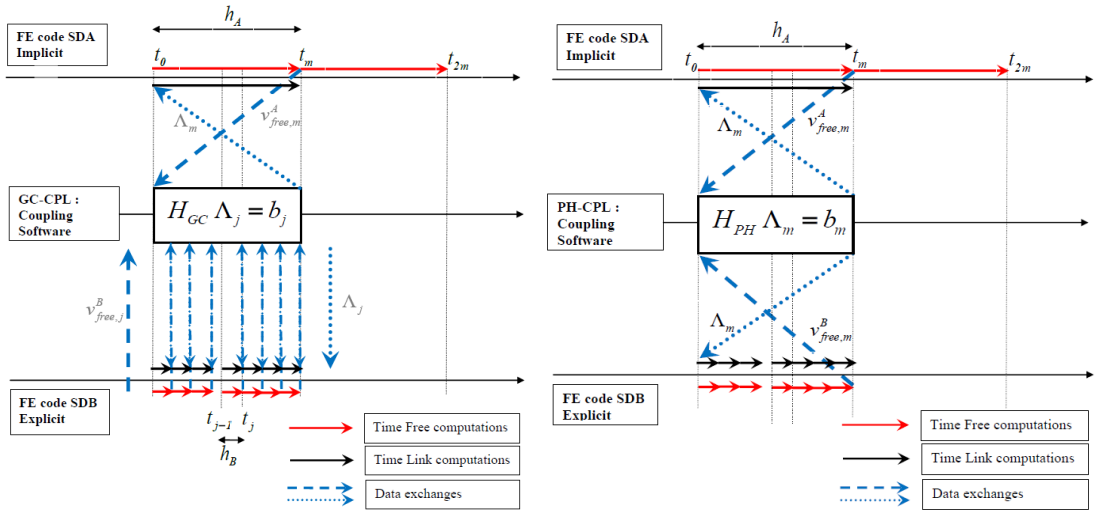

Figure 1: Time-computations and data exchanges over a macro time step between FE codes (subdomains SDA and SDB) and the external coupling software according to the GC algorithm (on left side) and to the PH algorithm (on the right side).

Over a given macro time step associated with the subdomain Ω_A, the following tasks undertaken by the PH-CPL are detailed below:

- Computing the free quantities by the FE code SDA over the macro-time step h_A and sending the free velocities $v^A_{free,m}$ to the PH-CPL.
- Solving the series of free problems in the subdomain Ω_B at the micro time scale by the FE code SDB, by looping over the macro-time step h_A, composed of m micro-time step h_B (for $j = 1,\ldots,m$). The last free velocities $v^B_{free,m}$ in subdomain Ω_B is sent from the FE code SDB to the PH-CPL.

- The interface problem can be set up. The PH interface operator has been previously built in the PH-CPL. The right hand side b_m is composed of the free velocities at the end of macro time step coming from both FE codes SDA and SDB.
- Solving the interface problem by computing the Lagrange multipliers Λ_m at the end of the macro-time step and sending the Lagrange multipliers to the FE codes SDA and SDB.
- Computing the linked quantities by the FE code SDA on the macro-time step h_A and updating the quantities (free+linked)
- Solving the series of linked problems in the subdomain Ω_B at the micro time scale by the FE code SDB, by looping over the macro-time step h_A, composed of m micro-time step h_B (for $j = \mathbf{1},\ldots,m$) and updating all the quantities at the micro time scale.

From Figure 1, it can be checked that no data exchange occurs directly between the two FE codes. Apart from the initial building of the PH interface operator, the mission of the PH-CPL exclusively consists in receiving free velocities from the FE codes and sending back the computed Lagrange multipliers. The coupling software completely ensures the interface between the two FE codes. In addition, the co-computations can be conducted without interfering in the parallel computing strategies employed by both the FE codes.
Contrary to the building of the interface operator which is much time consuming in the case of the PH algorithm, the requirements in terms of data exchanges in comparison to the GC algorithm, depicted on the left side of Figure 1, are clearly alleviated. Indeed, the GC coupling software has to communicate $\mathbf{2}m$ times over the macro-time step to solve the quantities in subdomain Ω_B: at each micro time step, one way for reading the free velocities $v^B_{free,j}$ from the FE code SDB, and one way for sending back the computed Lagrange multipliers Λ_j at the fine time scale. In the PH algorithm, only $\mathbf{2}$ data exchanges are involved concerning the computation in the FE code SDB: the PH-CPL reads the last free velocities $v^B_{free,m}$ from the FE code SDB and sends back the computed Lagrange multipliers Λ_m at the end of the macro-time step. It can be concluded that the PH algorithm is far more effective than the GC algorithm in terms of exchange data.

6 NUMERICAL EXAMPLES

6.1 Split oscillator

The first academic example is a single degree of freedom problem involving an oscillator with a mass and a stiffness split into two masses and two springs linked together through a Lagrange multiplier. Accuracy order will be checked for the GC and PH algorithms.
The equation of motion, continuous in time, of the undamped oscillator characterised by the angular frequency ω, under free vibrations is:

$$a(t) + \omega^2 u(t) = \mathbf{0} \tag{1.20}$$

The oscillator angular frequency is defined by the mass m and the stiffness k as: $\omega = \sqrt{\frac{k}{m}}$. Initial conditions are noted as: $u(t=0)=u_0$, $v(t=0)=v_0$. In the following, the analytical solution of the undamped oscillator under free vibration will be used as reference results so as to assess the accuracy order of the multi-time step algorithms.

The mass and the stiffness are split as follows: $m = m_A + m_B$ and $k = k_A + k_B$. The following values for the split system have been adopted: $m_A = 1.10^{-6}$, $k_A = 1.10^4$, $m_B = 1.10^{-6}$, $k_B = 1.10^4$ An initial displacement problem is considered: $u_0 = 1$, $v_0 = 0$. The macro subdomain Ω_A with a large time step Δt_A is integrated in time using the average acceleration scheme ($\gamma_A = \frac{1}{2}, \beta_A = \frac{1}{4}$), whereas the micro subdomain Ω_B with a fine time step Δt_B is integrated using the central difference scheme ($\gamma_B = \frac{1}{2}, \beta_B = 0$). The time step ratio between the two time steps is noted as m and is set to 100. The final time of the free vibration simulation is: $T_f = 2.10^{-4} s$.

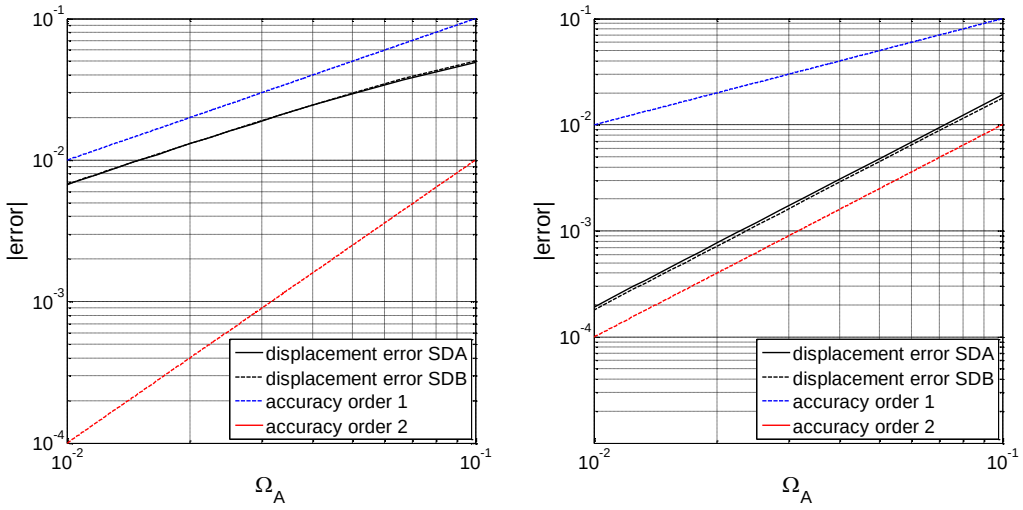

Figure 2: Convergence rate for the displacements of the multi-time step GC (on the left side) and PH (on the right side) algorithm for a time step ratio equal to 100.

In the case of the multi-time step GC algorithm, the errors in terms of displacements are plotted versus the reduced angular frequency of the subdomain SDA in Figure 2, along with slopes corresponding to the accuracy order equal to 1 or 2. It is shown that the convergence rate is reduced by one order when different time step sizes are involved as already highlighted in previous works (Mahjoubi *et al.* [9]). In the case of the multi-time step PH algorithm, the convergence rate remains equal to 2 for multi-time scales as shown on the right side of the Figure 6. The PH algorithm achieves a second order of accuracy, corresponding to the minimum of the orders of accuracy for the time integrators involved in the partitioned simulation.

6.2 Reinforced concrete frame structure under blast loading

The case of a non-earthquake resistant reinforced structure under blast loading is considered. The model of the structure has already been presented in previous works (Brun *et al.* [11]).

Briefly, the model of the frame structure is composed of multi-fibre Timoshenko beam elements, whose cross-section is discretised into material fibres for concrete and longitudinal reinforcement. The purpose is to assess the advantages of the PH-CPL coupling software in terms of energy dissipation at the interface in comparison to the GC-CPL coupling software. In Figure 3, the complete mesh of the structure is illustrated (3D visualisation for exhibiting the cross-section of the beam elements) as well as the partitioning assumed in the following co-computations: beam elements for columns and column-beam joints are gathered into the explicit partition, whereas the remaining of the mesh (the FE beams without the joints) are included in the implicit partition with the macro time step. Linear elastic behaviour is assumed for all the concrete and steel fibres with Young modulus equal to 25000 MPa and 200000 MPa, respectively. Dead and live loads are taken into account as well as the blast loads applied to the front face of the structure.

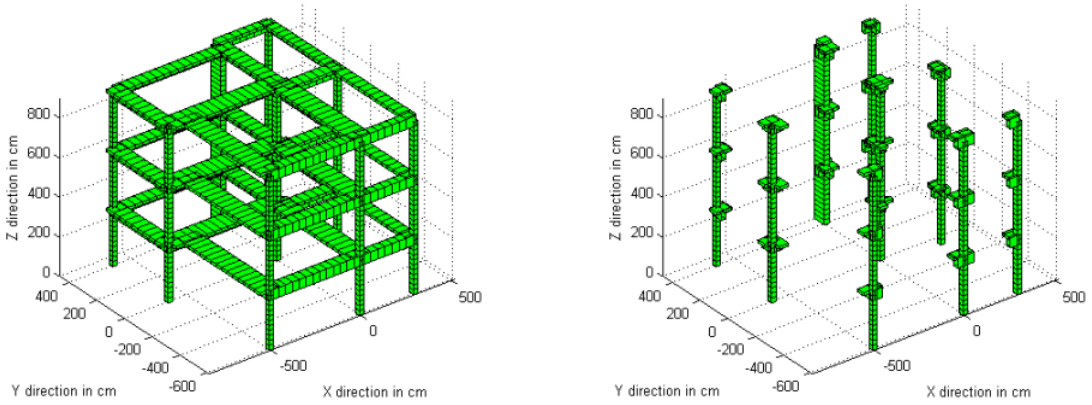

Figure 3: Mesh of the reinforced concrete beam-column structure and its partitioning (on the right side, the explicit subdomain)

Co-computations are carried out with both coupling software, the GC-CPL and the PH-CPL, by making interact two CAST3M processes, dealing with either the implicit or explicit partitions. The time-step ratio adopted between the macro time step and the micro time step is equal to 40. The reference results are provided by a full-explicit computation using the CAST3M code. Figure 4 shows the comparison of the investigated co-computations and the reference results in terms of displacements and velocities in the direction of the blast (axis Y) at the top of a column C5 of the structure. The almost perfect agreement between the results is highlighted.

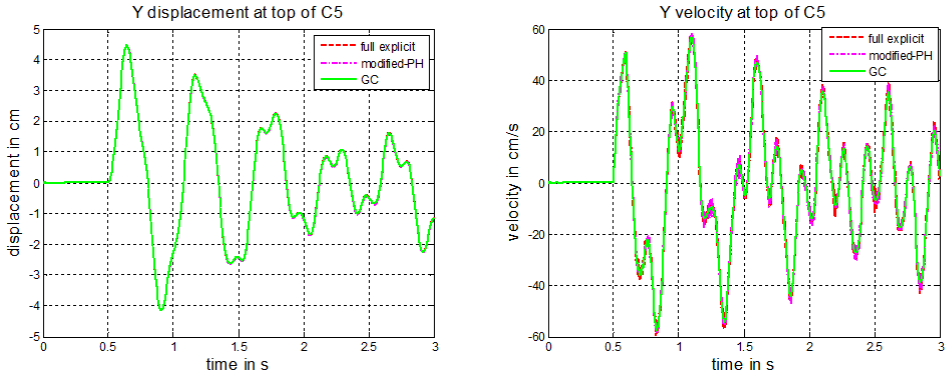

Figure 4: Time history Y-displacements (top graph) and Y-velocities (bottom graph) at the top of the C5 column predicted by full-explicit computation and co-computations according to the GC and PH algorithms

The unbalanced energy $W_{unbalanced}$, corresponding to the sum of the kinetic W_{kin}, internal W_{int}, complementary W_{comp} energies minus the external energy W_{ext} for both subdomains, are checked for the both coupling approaches. Note that the complementary energy is equal to zero when considered the implicit Newmark scheme with parameters $\gamma = \frac{1}{2}$ and $\beta = \frac{1}{4}$ and appears in the case of explicit Newmark subdomain with $\gamma = \frac{1}{2}$ and $\beta = 0$ (Hughes, [4]). This additional term with respect to the mechanical energy $W_{tot} = W_{kin} + W_{\text{int}}$ appearing in the balance energy equation is generated by the Newmark time discretization (Krenk, [14]). The interface energy coming from the coupling algorithms corresponds to the unbalanced energy when considered only energy conserving schemes. On the left of Figure 5, the dissipated energy created at the interface by the GC algorithm is highlighted as proved by the authors as soon as different time steps are adopted. As shown on the right of Figure 5, this spurious dissipative energy at the interface between subdomains is almost cancelled using the PH-based coupling software.

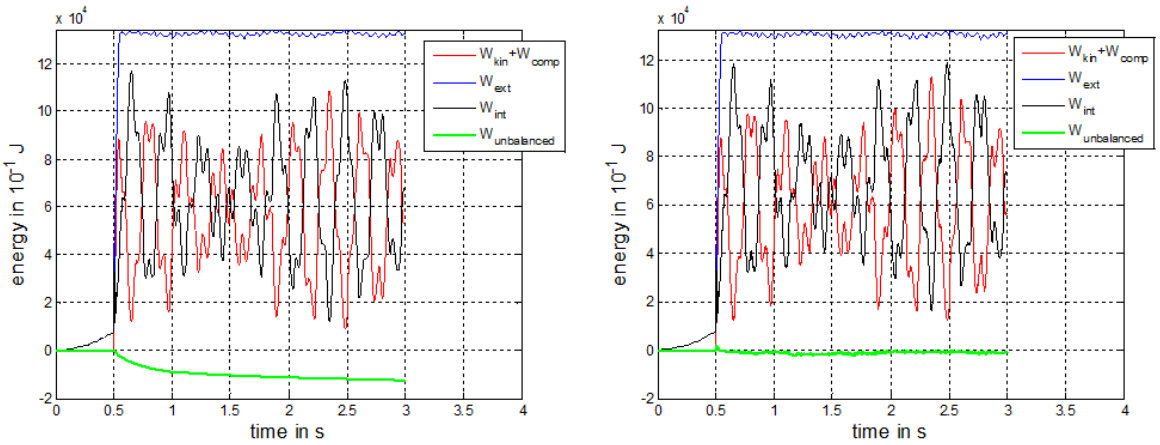

Figure 5: Time histories of kinetic (+complementary), internal, external energies and unbalanced energy from co-computations with GC-CPL (on the left) and PH-CPL (on the right) coupling software (time step ratio equal to 40).

7 CONCLUSIONS

This paper presents the set up of coupling software for structural dynamics. The coupling software makes interact in time Finite Element codes based on explicit and implicit time integration schemes. Two multi time step coupling algorithms are investigated: the GC algorithm, proposed by Gravouil and Combescure, and the PH algorithm, proposed by Prakash and Hjelmstad. The main difference between the two algorithms lies in the time scale at which the interface problem is solved: at the micro-scale for the GC algorithm and at the macro-scale for the PH-algorithm. The coupling software based on the PH algorithm, noted as PH-CPL, provides enhanced parallel capabilities and alleviates the energy dissipation drawback at the interface between subdomains.

The first numerical example is a simple split oscillator: using the modified-PH algorithm, the second-order of accuracy is achieved, whereas the GC algorithm only leads to first-order of accuracy when different time steps are considered depending on the subdomains.

The following application makes use of both coupling software GC-CPL and PH-CPL. It

deals with a reinforced concrete frame structure under blast loading striking its front face. Multi time step explicit/implicit co-computations with GC-CPL and PH-CPL software have been validated with respect to the full explicit computations. The two algorithms predict very close displacements and damage states. Nevertheless, the PH-CPL performs better with respect to the GC-CPL when one compares the spurious energy coming from the algorithms at the interface.

REFERENCES

[1] Combescure A, Gravouil A. A numerical scheme to couple subdomains with different time-steps for predominantly linear transient analysis. *Computer methods in applied mechanics and engineering* 2002; **191**:1129-1157.
[2] Farhat C, Roux F.X. A method of finite element tearing and interconnecting and its parallel solution algorithm. *International Journal for Numerical Methods in Engineering* 1991; **32**:1205-1227.
[3] Farhat C, Crivelli L, Roux F.X. Transient FETI methodology for large-scale parallel implicit computations in structural mechanics. *International Journal for Numerical Methods in Engineering* 1994; **37**:1945-1975.
[4] Hughes T.J.R. The Finite Element Method: Linear Static and Dynamic Finite Element Analysis. Prentice-Hall, Englewood Cliffs, NJ, 1987.
[5] Gravouil A, Combescure A. A multi-time-step explicit-implicit method for non-linear structural dynamics. *International Journal for Numerical Methods in Engineering* 2001; **50**:199-225.
[6] Herry B, Di Valentin L, Combescure A. An approach to the connection between subdomains with non-matching meshes for transient mechanical analysis. *International Journal for Numerical Methods in Engineering* 2002; **55**:973-1003.
[7] Faucher V, Combescure A. Local modal reduction in explicit dynamics with domain decomposition. Part 1: extension to sub-domains undergoing finite rigid rotations *International Journal for Numerical Methods in Engineering* 2004; **60**:2531-2560.
[8] Prakash A, Hjelmstad KD. A FETI-based multi-time-step coupling method for Newmark schemes in structural dynamics *International Journal for Numerical Methods in Engineering* 2004; **61**:2183-2204.
[9] Mahjoubi N, Gravouil A, Combescure A. Coupling subdomains with heterogeneous time integrators and incompatible time steps. *Computational mechanics* 2009; DOI 10.1007/s00466-009-0413-4.
[10] Brun M., Batti A., Limam A. and Combescure A. Implicit/Explicit multi-time step co-computations for predicting reinforced concrete structure response under earthquake loading. *Soil dynamics and Earthquake Engineering*;**33(1)**:19-37.
[11] Brun M., Batti A., Limam A., Gravouil A. Explicit/implicit multi-time step co-computations for blast analyses on a reinforced concrete frame structure. *Finite Elements in Analysis & Design 2012*;**52**:41-59.
[12] Verpeaux P, Charras T, Millard A, CASTEM 2000 une approche moderne du calcul de structures. In *Calcul de structure et intelligence artificielle*. Pluralis , Fouet JM, Ladeveze P., Ohayon R. (ed.), 1988, 261-271.
[13] EuroPlexus, A computer program for the finite element simulation of fluid-structure systems under dynamic loading, *User's manual*, CEA Saclay, CEA/DEN/SEMT/DYN, 2002.
[14] Krenk S. Energy conservation in Newmark based time integration algorithm. *Computer methods in applied mechanics and engineering* 2006; **195**:6110-6124.