

A parallel spectral element method for dynamic soil-structure interaction problems

F. Maggio^a, L. Massidda^a, R. Paolucci^b and M. Stupazzini^c

^a CRS4, Parco Scientifico e Tecnologico, Polaris, Edificio 1, 09010 Pula - Italy

^b Department of Structural Engineering - Politecnico di Milano - Italy

^c Geophysics Department of Earth and Environmental Sciences - Munich University - Germany

Abstract

We illustrate the capability of a spectral element method to efficiently handle the numerical analysis of complex dynamic soil-structure interaction problems. Spectral elements are naturally suited for this class of applications: they provide high accuracy, are CAD-oriented, enjoy the flexibility of finite elements and allow the implementation of optimized parallel algorithms. Furthermore, spectral accuracy is the key to reduce the number of degrees of freedom and efficiently include in the 3D computational model both the structure and the surrounding soil. Therefore the need of sub-structuring techniques to deal with this type of problems is avoided. We illustrate the implementation of the method and provide numerical tests about its parallel performance. Finally, we show how our spectral element method is suitable for handling a challenging computational problem, such as the 3D analysis of building vibrations induced by underground train passage.

1 INTRODUCTION

In spite of the enormous computational progress in recent years, problems involving the interaction of structures with the surrounding soil under dynamic loading, such as earthquake, wind, or other vibration sources, are still extremely challenging from the analytical and numerical point of view. These problems are typically defined by three major elements: the source, the propagation path and the structure itself. The accurate modelling of each of such elements is a hard task: the source may not be well defined, such as in the earthquake case, or it may involve a very large frequency spectrum, such as for traffic-induced vibrations; the propagation path, in terms of spatial variability of dynamic soil properties, is seldom well constrained by suitable geophysical/geotechnical prospecting; finally, the dynamic behaviour of the structure and the supporting soil may be strongly affected by nonlinear effects, the influence of which can be assessed reliably only in few cases.

The simultaneous presence of all of these elements in the same numerical model makes the problem extremely difficult to be handled from the computational point of view, mainly due to the different scale dimensions of the various elements, that may range from fractions of a meter for the structure, up to hundreds or thousands of meters for the propagation path.

Furthermore, it is well known that dynamic soil-structure interaction (DSSI) problems face another major difficulty, since closed-form transparent boundary conditions for time-domain formulations are available only in approximate form (see, e.g., the Clayton and Engquist paraxial conditions [7]). Therefore, the analyst is generally forced to artificially enlarge the computational domain to prevent spurious signals originated from the boundary to affect the region of interest.

To overcome such problems, engineers and researchers dealing with DSSI problems are accustomed to sub-structuring techniques. One of the best known examples of such techniques is the replacement of the soil supporting a vibrating structure by a suitable set of elastic springs and dashpots, calibrated to model the soil stiffness, the radiation of waves outside the foundation and the internal dissipation of materials [15]. Although this approach had an enormous and beneficial impact on practical DSSI applications, it is afflicted with important theoretical limitations, such the frequency-dependence of the equivalent springs and dashpots, their availability for simple soil-foundation geometries only, and the inaccurate treatment of nonlinear effects (e.g., internal dissipation in the soil and/or sliding or uplifting at the soil-foundation interface).

A recent and quite promising sub-structuring technique is the “domain reduction method” [2], where the problem of coupling a “large” domain (including the earthquake source and the propagation path) and a “local” domain (the structure with the supporting soil) is solved by rigorously calculating the nodal forces at the boundary between both domains. The main problem with this technique, that is general enough to be applied either in a finite element or in a spectral element framework [12], is that for complex 3D geometries the coupling procedure itself may become a demanding task.

Hybrid approaches, widely used in recent years in various DSSI problems, may be also viewed as particular sub-structuring techniques, where different numerical methods are coupled to fully exploit their capabilities in different domains for seismic wave propagation and soil-structure interaction analyses. Examples of such hybrid approaches can be found in [37] for coupling boundary elements and finite elements, [26] for finite elements and finite differences, [5] for spectral elements and finite elements.

However, the enormous progress of supercomputing with huge computational power on one side, and the possibility of constructing low-cost PC clusters, on the other side, has encouraged a notable effort to exploit in parallel computer architectures the capabilities of well established and versatile numerical codes, such as finite elements (FEM,[38]) and finite differences (FD, [36], [21]).

All of these methods suffer of the well-known limit affecting numerical simulation of wave phenomena, that is the need to fix the accuracy of the method based on the frequency content of travelling signals, in order to prevent the onset of non-physical, spurious effects, commonly referred to as *numerical dispersion*. The obvious solution to this problem is grid refinement, a choice at the base of popular methods - the so-called *h-methods* - like standard finite elements and finite differences. In the research practice they are used with rules of thumb linking the number of grid-points per wavelength and the maximum

frequency which can be effectively simulated. FEM, in particular, are widely adopted for soil-structure interaction analyses also because of their capability to model physical domains with complex shape. Unfortunately, effective grid-refinement significantly increases the number of degrees of freedom, leading, especially for 3D DSSI analyses, to exceedingly large storage requirement and computing time.

On the other hand, it is possible to improve the quality of numerical simulations by expressing the solution in terms of polynomial functions of high degree: this is the idea behind the so-called *p-methods*. The possibility to merge advantages of both approaches is clearly appealing: one may tune the balance in favour of either the *h*- or the *p*-approach, taking into account the problem features (geometry complexity, regularity of the solution, etc.). This idea gave rise to the so-called *hp-methods*, including Spectral Element Methods (SEM), the technique adopted throughout this work.

After the pioneering work of Kosloff and Baysal [19], who used a Fourier decomposition of the displacement field, spectral methods for elastic wave propagation evolved first with the introduction of Chebyshev polynomials [20] and subsequently of Lagrange polynomials coupled with Legendre-Gauss-Lobatto quadrature formulas [11],[18]. Spectral element algorithms for the analysis of large scale soil problems have been proposed in [13], [16], [34].

The starting point of this work is the spectral element method originally proposed by [11] and subsequently applied in its sequential version in several works dealing with seismic wave propagation problems in complex geological configurations [29], [28], [35].

After a brief outline of the method, we show first how its main features fit well with the needs of complex DSSI problems and the criteria for the implementation in a parallel computer architecture. Subsequently, the parallel efficiency of the numerical code is assessed through suitable benchmark problems. Finally, a challenging example of application will be illustrated, for the evaluation of building vibrations due to underground train passage.

2 OUTLINE OF THE METHOD

We limit to materials following the linear Hooke constitutive behaviour, filling a domain Ω with regular boundary, described within a small displacement frame. As for finite elements, our method stems from the variational formulation of the elastodynamic problem: for each time $t \in (0, T]$ find $\mathbf{u}(t) \in [H^1(\Omega)]^3$ satisfying initial and boundary conditions and such that

$$\int_{\Omega} \rho \ddot{\mathbf{u}} \cdot \mathbf{v} \, d\Omega + \int_{\Omega} \boldsymbol{\sigma}(\mathbf{u}) : \boldsymbol{\epsilon}(\mathbf{v}) \, d\Omega + \int_{\Gamma_N} \mathbf{t} \cdot \mathbf{v} \, d\Gamma + \int_{\Gamma_{NR}} \mathbf{t}_{NR} \cdot \mathbf{v} \, d\Gamma = \int_{\Omega} \mathbf{f} \cdot \mathbf{v} \, d\Omega \quad \forall \mathbf{v} \in [H_0^1(\Omega)]^3 \quad (1)$$

where \mathbf{u} is the unknown displacement, \mathbf{v} the generic admissible displacement, \mathbf{f} the external volume force applied in Ω and ρ the density; $\boldsymbol{\sigma}$ and $\boldsymbol{\epsilon}$ denote usual stress and small-strain tensor, respectively, while vertical double dots denote tensor inner product: $A : B = \sum_{i,j=1}^3 A_{ij} B_{ji}$. The space $H^1(\Omega)$ consists of

functions which are square-integrable on Ω , whose gradient components are also square-integrable on Ω (see, for instance, [33]); $H_0^1(\Omega)$ is the subset of $H^1(\Omega)$ whose functions vanish on Γ_D . The boundary of Ω consists of three regions: Γ_0 , where displacements are prescribed; Γ_N , subject to known external tractions \mathbf{t} (vanishing in the free-surface case); Γ_{NR} , where artificial non-reflecting conditions are set to simulate propagation of waves in infinite domains (we adopt the Stacey boundary conditions, see [11]). The spectral element discretization is based on the decomposition of Ω in a family of non-overlapping open hexahedra Ω_k , such that

$$\overline{\Omega} = \cup_k \overline{\Omega_k} \quad \text{and} \quad \Omega_i \cap \Omega_j = \emptyset \quad \text{if} \quad i \neq j$$

where overline denotes the union of the domain and its boundary. An admissible hexahedron is obtained from the cube $\widehat{\Omega} = (-1, +1)^3$ (the *master element*), through a mapping T_k defined as follows:

$$T_k : \begin{cases} x^{(k)} &= \alpha_1^{(k)} \hat{x} + \beta_1^{(k)} \hat{y} + \gamma_1^{(k)} \hat{z} + \lambda_1^{(k)} \hat{x}\hat{y} + \mu_1^{(k)} \hat{x}\hat{z} + \nu_1^{(k)} \hat{y}\hat{z} + \sigma_1^{(k)} \hat{x}\hat{y}\hat{z} + \delta_1^{(k)} \\ y^{(k)} &= \alpha_2^{(k)} \hat{x} + \beta_2^{(k)} \hat{y} + \gamma_2^{(k)} \hat{z} + \lambda_2^{(k)} \hat{x}\hat{y} + \mu_2^{(k)} \hat{x}\hat{z} + \nu_2^{(k)} \hat{y}\hat{z} + \sigma_2^{(k)} \hat{x}\hat{y}\hat{z} + \delta_2^{(k)} \\ z^{(k)} &= \alpha_3^{(k)} \hat{x} + \beta_3^{(k)} \hat{y} + \gamma_3^{(k)} \hat{z} + \lambda_3^{(k)} \hat{x}\hat{y} + \mu_3^{(k)} \hat{x}\hat{z} + \nu_3^{(k)} \hat{y}\hat{z} + \sigma_3^{(k)} \hat{x}\hat{y}\hat{z} + \delta_3^{(k)} \end{cases} \quad (2)$$

where coordinates $\{x^{(k)}, y^{(k)}, z^{(k)}\}$ and $\{\hat{x}, \hat{y}, \hat{z}\}$ are associated to Ω_k and $\widehat{\Omega}$, respectively. While, in general, transformation (2) describes hexahedra with curved quadrilateral faces, in practice mesh generators produce elements with planar faces. Therefore, the physical domain Ω is approximated by $\widetilde{\Omega}$, whose boundary is the union of planar quadrilaterals; Γ_0 , Γ_N , and Γ_{NR} , are replaced by $\widetilde{\Gamma}_0$, $\widetilde{\Gamma}_N$, and $\widetilde{\Gamma}_{NR}$, respectively. For each hexahedron Ω_k , the eight parameters $\alpha_i^{(k)}, \beta_i^{(k)}, \gamma_i^{(k)}, \delta_i^{(k)}, \lambda_i^{(k)}, \mu_i^{(k)}, \nu_i^{(k)}, \sigma_i^{(k)}$ ($i = 1, 2, 3$) are determined exploiting the 1-to-1 mapping between corner points of Ω_k and $\widehat{\Omega}$.

On the reference element $\widehat{\Omega}$ we introduce $\mathbf{Q}_n(\widehat{\Omega})$, the space of polynomial function with degree less than or equal to n with respect to each variable: the generic element of $\mathbf{Q}_n(\widehat{\Omega})$ reads

$$\widehat{\psi} = \sum_{i,j,k=0}^n a_{ijk} \hat{x}^i \hat{y}^j \hat{z}^k \quad (3)$$

where the $\{a_{ijk}\}$ are real coefficients and n is called *spectral degree*. A finite dimensional approximation $V_{h,n}$ of $H_0^1(\Omega)$ can then be defined considering the continuous functions obtained by mapping the $\{\widehat{\psi}\}$'s element-by-element on the hexahedra:

$$V_{h,n} = \left\{ v \in C^0(\Omega) : v = 0 \text{ on } \widetilde{\Gamma}_0 \text{ and } v|_{\Omega_k} = \widehat{\psi} \circ T_k^{-1}, \widehat{\psi} \in \mathbf{Q}_n(\widehat{\Omega}) \right\} \quad (4)$$

where $\widehat{\psi} \circ T_k^{-1}$ is the mapping of $\widehat{\psi}$ from $\widehat{\Omega}$ to Ω_k . It can now be understood that the generic T_k introduced in (2) is an element of $[\mathbf{Q}_1(\widehat{\Omega})]^3$: thus, when $n > 1$ the mapping is *sub-parametric*, meaning that its degree is lower than the spectral degree. This choice is essentially motivated by a practical consideration: mesh generators produce 8-points hexahedra, while n -order hexahedra need $(n+1)^3$

total points (and produce different grids for analyses with different spectral degree). Furthermore, the adopted sub-parametric mapping (2) is known to enjoy good mathematical properties (see, for instance, [9]). Nevertheless, if high order hexahedra are desirable (for instance, when dealing with domains with curved boundary which can be described in terms of geometrical primitives, or for large displacement/deformation formulations), they can be incorporated in the spectral element frame with small additional effort [22].

The next step is the introduction of the Legendre-Gauss-Lobatto (LGL) nodes in $\widehat{\Omega} = [-1, +1]^3$. They are obtained via tensor product of the one-dimensional LGL nodes ζ_0, \dots, ζ_n defined over the interval $[-1, +1]$: it turns out that $\zeta_0 = -1$, $\zeta_n = +1$ and the intermediate points are the zeros of the first derivative of the Legendre polynomial of degree n [4]. The full spectral grid $\{\mathbf{a}_p\}_{p=1}^N$ can then be built mapping the LGL nodes over the hexahedra and eliminating duplicated points: as for finite elements, a global numbering is associated to the N grid-points. If iso-parametric mapping has to be used, this procedure is obviously inverted: for each element Ω_k , the $(n+1)^3$ spectral grid-points should be externally provided, in order to define the n -order hexahedron. Sub-parametric and iso-parametric mapping for a two-dimensional case are shown in figure 1.

An intuitive basis for $V_{h,n}$ is provided by $\{N_p(\mathbf{x})\}_{p=1}^N$, the Lagrange polyno-

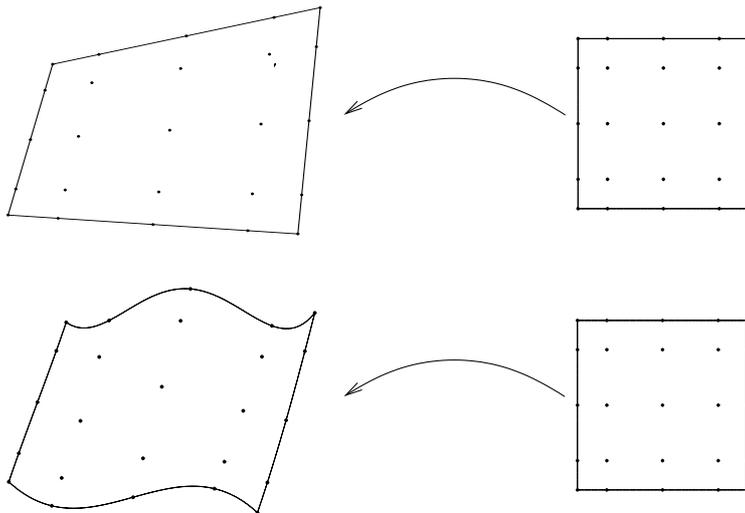


Figure 1: Examples of sub-parametric (*top*) and iso-parametric (*bottom*) mapping between the generic quadrilateral (hexahedron in 3D) and the reference element $\widehat{\Omega}$: spectral degree $n = 4$.

mials of order n defined over the spectral grid-points:

$$N_p \in V_{h,n} \quad \text{and} \quad N_p(\mathbf{a}_q) = \delta_{pq}$$

where δ_{pq} is the Kronecker delta. Then the semi-discrete approximation of (1) reads:

for each $t \in (0, T]$ find $\mathbf{u}(t) = \sum_{q=1}^N (u_q(t)\hat{\mathbf{e}}_1 + v_q(t)\hat{\mathbf{e}}_2 + w_q(t)\hat{\mathbf{e}}_3)N_q$, $\mathbf{u}(t) \in [H^1(\Omega)]^3$, with prescribed displacements on $\tilde{\Gamma}_0$ and such that, for $i = 1, 2, 3$:

$$\begin{aligned} & \sum_{q=1}^N \left[(\ddot{u}_q \hat{\mathbf{e}}_1 + \ddot{v}_q \hat{\mathbf{e}}_2 + \ddot{w}_q \hat{\mathbf{e}}_3) \cdot \hat{\mathbf{e}}_i \int_{\Omega} \rho N_p N_q d\Omega + \int_{\Omega} \sigma((u_q \hat{\mathbf{e}}_1 + v_q \hat{\mathbf{e}}_2 + w_q \hat{\mathbf{e}}_3)N_q) : \epsilon(N_p \hat{\mathbf{e}}_i) d\Omega \right] + \\ & \int_{\Gamma_N} \mathbf{t} \cdot (N_p \hat{\mathbf{e}}_i) d\Gamma_N + \int_{\Gamma_{NR}} \mathbf{t}_{NR} \cdot (N_p \hat{\mathbf{e}}_i) d\Gamma_{NR} = \int_{\Omega} \mathbf{f} \cdot (N_p \hat{\mathbf{e}}_i) d\Omega \quad \forall p = 1, \dots, N \end{aligned} \quad (5)$$

where $\hat{\mathbf{e}}_i$ is the unit vector of the i -th coordinate.

The first time-derivatives of unknowns $\{u_q, v_q, w_q\}$ appear also in the term \mathbf{t}_{NR} due to non-reflecting conditions. In order to time-discretize (5), we use the following 2^{nd} order *LF2-B2* scheme, which has been proven to be effective [25]:

$$\begin{aligned} \ddot{f}(t_n) &= [f(t_{n+1}) - 2f(t_n) + f(t_{n-1}))]/(\Delta t)^2 + \mathcal{O}((\Delta t)^2) \\ \dot{f}(t_n) &= [3f(t_n) - 4f(t_{n-1}) + f(t_{n-2}))]/(2\Delta t) + \mathcal{O}((\Delta t)^2) \end{aligned} \quad (6)$$

where $t_k = k\Delta t$, $k = 0, 1, 2, \dots$. Proceeding as for finite elements, one has to replace derivatives and integrals in (5) with suitable numerical approximations. Derivatives are evaluated in the reference element $\hat{\Omega}$ via the chain rule, where terms linking $\{x^{(k)}, y^{(k)}, z^{(k)}\}$ and $\{\hat{x}, \hat{y}, \hat{z}\}$ are obtained through (2), and derivatives in the reference element are computed via the matrix of the collocation derivative (see [4], chap. 2). Integrals are also computed in $\hat{\Omega}$:

$$\int_{\Omega} f d\Omega = \sum_k \int_{\Omega_k} f d\Omega = \sum_k \int_{\hat{\Omega}} \hat{f} J_k d\hat{\Omega} \simeq \sum_{r,s,t=0}^n \hat{f}(\hat{x}_r, \hat{y}_s, \hat{z}_t) J_k(\hat{x}_r, \hat{y}_s, \hat{z}_t) \hat{w}_r \hat{w}_s \hat{w}_t \quad (7)$$

where \hat{w}_l is the generic weight of the Gauss-Lobatto quadrature formula [4] and J_k is the Jacobian of transformation (2). Merging (5) and (6), and evaluating derivatives and integrals as described above, leads to the following algebraic problem

$$\mathbf{M}\ddot{\mathbf{u}} + \mathbf{C}\dot{\mathbf{u}} + \mathbf{K}\mathbf{u} = \mathbf{F} \quad (8)$$

where \mathbf{M} , \mathbf{C} , and \mathbf{K} are usually denoted as the *mass*, *damping* and *stiffness* matrix. It is understood that a 1-to-1 relation exists between local indices of the element $((r, s, t)$ in (7)) and global indices of arrays in (8); furthermore, an ordering has been chosen for solution array \mathbf{u} (for instance, the first N components for the u_q , then the v_q and finally the w_q). The size of the algebraic problem is $3N$.

3 MAIN FEATURES OF SEM FOR DYNAMIC SOIL-STRUCTURE INTERACTION PROBLEMS

We have already discussed in the introduction the challenging aspects of large 3D DSSI problems, mainly related to the need of simultaneously modelling small-scale and large-scale elements, on one side, and, on the other side, to the possible onset of nonlinearities, both within the structure and within the supporting soil. The treatment of nonlinear effects within the SEM framework is the subject of a recent study that has provided very promising results [35]. Here we show, instead, how the features of the SEM are suited to deal accurately with DSSI problems involving also complex shaped structures, like dams, bridges, subways, monumental or historical buildings, and portions of soil beneath.

A first important feature of SEM, that was shown to be a great help in design of seismic analyses [11], [18], [32], as well as in numerical simulation of general wave phenomena [24], [3], [23], is that they are capable - to some extent - to exploit the flexibility of standard finite elements in dealing with complex structures, and enjoy the possibility to play on the degree of piecewise polynomial functions to model high frequency waves, rather than using grid refinement. This is illustrated in figure 2: acoustic waves arising from a point source propagate through a 2D homogeneous domain with irregular shape. The snapshots, taken at the same instant, are computed using different spectral degrees. Those with lower degrees are clearly affected by numerical dispersion, which is practical absent in the cases with higher degree. This example highlights a major asset of *hp*-methods for wave applications: the possibility of changing the accuracy of the numerical solver - and therefore to deal with signals with higher frequency - without the need to modify the computational grid, an operation highly demanding for applications with complex geometry. While, in principle, it is possible to design FEM algorithms with shape functions of arbitrary degree, engineering implementations are usually limited to degree 2, due to practical constraints. On the opposite, the SEM can perform analyses with arbitrary degree, simply selecting it *at run-time*.

Furthermore, due to the adoption of sub-parametric mapping, the interaction with CAD-based models is straightforward. On the computational ground, the resulting algorithm can be highly optimized, thanks to a matrix-free implementation (memory allocation reduction) and the natural use of explicit time-marching schemes (CPU-time saving): as a consequence, the treatment of large-scale problems eases off. The implementation on massively parallel machines is effective, despite the more complicated data-structure with respect to low-order methods. Since the SEM have a very close relationship with finite elements, our method can easily inherit many technicalities developed for the latter, including physical models and numerical schemes. Last but not least, the algorithm behaves like a *black-box* requiring a minimum effort from the user, who simply has to provide the grid, the selected spectral degree and, possibly, the number of processors available for the parallel run. The Achilles' heel of

spectral elements is the need to use mesh of quadrilaterals and hexahedra. In recent years there have been a few attempts to include triangular elements in the SEM for elasticity problems [17], [5]: this gave rise to sophisticated and somehow elegant mathematical approaches which, in our view, are not completely suited for engineering applications (for instance, they can hardly be extended to complex 3D models and introduce additional computational effort). Based on our experience, both as developers and users of SEM, the limit concerning the use of pure quadrilateral elements mainly belongs to the past, since nowadays flexible and powerful tools for hexahedra mesh generation are available [6]. Fully automatic mesh generation with quadrilateral elements has still to come, but we believe that the supplementary effort to generate such numerical grids is well worthwhile, since hexahedra have proven to be more accurate than tetrahedra for structural and elastic analyses (at equal computational effort: see [1], [27] and references therein).

4 IMPLEMENTATION IN A PARALLEL COMPUTER ARCHITECTURE

Optimal implementation of a parallel spectral element solver may be a tricky task, since elements possess a high number of degrees of freedom with respect to standard FEM, thus requiring more sophisticated criteria for domain-decomposition and message passing organization.

4.1 Mesh partition

In our method, mesh partition is a fully automated procedure that divides the computational domain and the mesh among the processors, aiming at two main targets: balance the computational load on each CPU and minimize data exchange between the processors. The first task consists in dividing the computational domain in parts that require approximately the same computational effort. The load on a processor for an element-by-element explicit scheme is easily related to the number of elements assigned to that processor. Data exchange is associated with nodes and elements close to the interface between sub-domains, shared by different processors. Communication is minimized reducing data length and amount of message-passing per time step. Thus, mesh partitioning plays a crucial role by producing “optimal” interfaces among sub-domains, involving the minimum number of nodes and elements. We adopted Metis, a family of well-known multilevel partitioning algorithms [14]. Even though it was originally designed for dealing with finite element meshes, Metis essentially works on graphs, and can be fruitfully adapted to a mesh-based method like spectral elements. Graphs can be defined either using the nodes or the elements as vertexes: both these strategies may be adopted in an explicit element-by-element solution scheme, but we show in the following that the element based strategy fits better within a spectral element frame.

Following eq. (8) and the approximation (6), $\mathbf{u}^{(n+1)}$, the nodal displacement at t_{n+1} , may be calculated as follows:

$$\mathbf{u}^{(n+1)} = \Delta t^2 \mathbf{M}^{-1}(\mathbf{f}_{ext}^{(n)} - \mathbf{f}_{int}^{(n)}) + 2\mathbf{u}^{(n)} - \mathbf{u}^{(n-1)} \quad (9)$$

where the term $\mathbf{f}_{int}^{(n)}$ includes contributions from stiffness and damping matrix. Each processor should be able to calculate its part of the nodal unknowns, storing part of the nodal displacement vectors $\mathbf{u}^{(n)}$ and $\mathbf{u}^{(n-1)}$, and its portion of the mass matrix and of the external force vector $\mathbf{f}_{ext}^{(n)}$. The internal elastic force vector $\mathbf{f}_{int}^{(n)}$ at time t_n should also be evaluated. This is done by summing on the interface nodes the results of an integration over the spectral elements: an easy task for nodes that are internal to the computational domain, and a critical one when nodes and/or elements are close to the interface between sub-domains.

4.2 Node based partitioning

In this approach spectral grid-points are divided in subsets associated 1-to-1 with the processors of the parallel machine: as a result, elements crossed by the interface are split in two or more parts. For the sake of simplicity we focus on two processors P_1 and P_2 , the corresponding sub-domains Ω_1 and Ω_2 , and their interface Γ_{12} ; in figure 3 elements from $E1$ to $E4$ and elements from $E7$ to $E10$ are assigned to P_1 and P_2 respectively, while $E5$ and $E6$ are crossed by the interface and shared between the two processors.

A further division can be made, defining *internal*, *border* and *external* nodes for each processor. The internal nodes can be updated without the need of communication among processors. Updating border nodes requires information on the nodal unknowns belonging to other sub-domains. Finally, external nodes are updated by other sub-domains and processors, but their nodal unknowns are required for the border nodes update.

Concerning processor P_1 , the knowledge of nodal variables at the time t_n for the internal and border nodes (marked with \circ and \otimes , respectively) allows to calculate deformation, stresses, and to integrate over the internal elements from $E1$ to $E4$. Border nodes also need the contribution from the elements crossed by the interface: in order to perform the integration over elements $E5$ and $E6$, processor P_1 should have at disposal the nodal unknowns of the external nodes that belong to processor P_2 (marked with ∇): communication between processors is required. Similarly processor P_2 can integrate on elements from $E7$ to $E10$, while to complete the integration on the interface elements, needs the data relative to its external nodes.

The main advantage of this approach is that the communication between processor is required only once per time step; a major drawback is the fact that the operations on the interface elements have to be calculated by each processor, and the amount of data exchange may be high since it involves all the nodes belonging to the interface elements.

4.3 Element based partitioning

This approach stems from a different partition of the mesh that is now divided on an element basis rather than on a node basis: nodes “follow” element partition and some of them are shared. The mesh is divided among the different processors assigning to each of them a group of elements of approximately the same size; the interface between sub-domains is located on the edge or face between the elements, therefore no *interface elements* are considered, and *interface nodes* are defined instead.

We again use the two processor example. In figure 4 we see that elements from $E1$ to $E4$ belong to the sub-domain Ω_1 and are assigned to processor P_1 , while elements from $E5$ to $E10$ are relative to the sub-domain Ω_2 and to processor P_2 . Nodes lying on the interface Γ_{12} are shared by the two sub-domains and processors: they are assigned to one or the other processor on a load balance basis. It turns out that that the internal nodes for a processor are those grid-points not lying on the interface; border nodes belong to the interface and are assigned to the processor at hand, while the external nodes belong to the interface but are assigned to a different processor: the value of their nodal unknowns is needed to update the degrees of freedom defined on the internal and border nodes.

At each time step each processor updates a portion of the vector of nodal displacements $\mathbf{u}^{(n+1)}$. Focusing on processor P_1 in figure 4, the nodal displacements $\mathbf{u}^{(n)}$ and $\mathbf{u}^{(n-1)}$ for internal and border nodes are known, as well as the external force vector $\mathbf{f}_{ext}^{(n)}$; the processor has to calculate stresses and perform the integration over its elements (from $E1$ to $E4$), and clearly needs to receive the information of the nodal displacement of the external node (marked with ∇). Similarly processor P_2 should receive from processor P_1 the data related to the displacements of the external nodes (marked with \otimes). A first communication is then required at this stage of the time step. After that, each processor can perform the integrations over its sub-domains.

While the coefficients of the force vector at the internal nodes can be fully calculated by the corresponding processor, computation at the interface nodes depends also from the contribution of elements belonging to different sub-domains, and communication is required once again. Processor P_1 receives the contribution to $\mathbf{f}_{int}^{(n)}$ due to elements $E5$ and $E6$ to their own external nodes. These data are summed to the contribution evaluated by processor P_1 , due to elements $E3$ and $E4$, in order to complete the computation of the force vector. Processor P_2 , in turn, receives from P_1 the result of the integration over element $E4$ for its external nodes (marked with ∇).

All in all, the main advantage of this approach is that the load on each processor is well balanced, since the operations on elements and nodes may be exactly distributed between the different processors; moreover, the amount of data to be communicated is minimum. On the other hand, the main drawback is that communication is required twice per time step.

4.4 Spectral element strategy

The node based approach is a standard choice for linear finite element codes in which the stiffness and mass matrix are assembled, since in that case the requirement of a single communication per time step is particularly appealing. In the spectral element case, even if both approaches are feasible, the element based technique is preferable.

As previously illustrated, the number of nodes of a spectral element depends on n , the spectral degree adopted: there are $n + 1$ nodes per edge, $(n + 1)^2$ nodes per face, and $(n + 1)^3$ nodes per hexahedron. Let us denote by n_Γ the number of elements crossed by an interface between two sub-domains. In the node based approach all the nodal unknowns belonging to the interface elements have to be exchanged between the two processors considered, once per time cycle; this is approximately equivalent to exchange $3n_\Gamma(n + 1)^3$ real numbers in 3D (the exact amount depends on the interface shape). In the element based partitioning the interface is stuck to the element faces, the nodal displacements belonging to the interface nodes are exchanged twice per time cycle, corresponding to $6n_\Gamma(n + 1)^2$ reals.

Therefore, the ratio of data exchange between the node based and the element based partitioning is approximately equal to $(n + 1)/2$ - making the two approaches equivalent for $n = 1$ (finite elements) - and increases with the spectral degree adopted. For instance, in the case $n = 5$ the amount of communication is three times smaller for the element based partitioning. For very low values of the spectral degree, the necessity to synchronize processors twice per cycle in the element based approach may lead to worse results with respect to node partitioning; but as mesh complexity and spectral degree increases, the advantages of the element based partitioning become fundamental to get a good parallel efficiency.

5 PARALLEL PERFORMANCE

The parallel efficiency of our code was checked with a set of benchmarks performed on two different computer architectures located at CILEA (Consorzio Interuniversitario Lombardo per l'Elaborazione Automatica, Segrate, Italy). The computer architectures for our tests are summarized in Table 1.

The parameters considered for the performance tests are the following:

$$\begin{aligned}
 &T(i)_{SEQ}, \text{ CPU time for the } i\text{-th time step in the sequential run;} \\
 &T(i)_{PAR}, \text{ CPU time for } i\text{-th time step in the parallel run;} \\
 &T_{SEQ} = \sum_{i=1}^m \frac{T(i)_{SEQ}}{m}, \text{ average } T(i)_{SEQ} \text{ over } m \text{ time steps;} \\
 &T_{PAR} = \sum_{i=1}^m \frac{T(i)_{PAR}}{m}, \text{ average } T(i)_{PAR} \text{ over } m \text{ time steps;} \\
 &PE = \frac{T_{SEQ}}{N_{CPU}T_{PAR}}, \text{ parallel efficiency, where } N_{CPU} \text{ is the number of CPUs.}
 \end{aligned}$$

| Machine Name | Golgi | Avogadro |
|------------------|------------------------------|----------------------------|
| # CPUs | 48 | 256 |
| CPU type | AMD Opteron 64 bit 2.2 GHz | Intel Xeon 32 bit 3.06 GHz |
| # CPUs/node | 4 | 2 |
| RAM/node | 4Gb | 2Gb |
| Connections | Gigabit Ethernet, Infiniband | Myrinet (fiber optic) |
| Operative system | Linux Fedora core 3 | Linux Red Hat 9.0 |

Table 1: Characteristics of the two clusters adopted for parallel efficiency tests

In all comparisons we used $m = 1\,000$, while N_{CPU} ranges from 1 up to 128 for the “Avogadro” cluster, and up to 32 for the “Golgi” cluster. The results obtained on the two architectures are similar, so the presentation will be limited to the Avogadro cluster. Our benchmark is the wave propagation induced by a point force inside an homogeneous medium. The computational models are cubes with side length of 1 km (“small” case), 2.155 km (“medium” case) and 4.642 km (“large” case), respectively. Both structured and unstructured grids were considered. In the structured case the element size is 100 m, corresponding to 68 921, 704 969 and 6 331 625 spectral nodes, respectively, using a spectral degree $n = 4$. The allocated memory is approximately 100 Mb, 1 Gb and 10 Gb, for the three cases. For the unstructured mesh, the number of spectral nodes ranges from 68 429 (“small”) up to 5 201 645 (“large”) with $n = 4$; in the latter case the elements size ranges from 15 up to 500 meters.

The tests are meant to provide PE measurements for realistic problem size and engineering applications. This is the main reason why we fixed the total size of the problem, rather than the memory allocated to each CPU, as often done in similar tests [34]. Furthermore, we considered the average PE value over 1000 CPU time steps for each test, rather than its peak value. As shown in figure 5, in the “small” case the performance obtained for structured and unstructured mesh are similar, at least up to 56 CPUs, with an estimated average $PE=83.4\%$. For increasing CPUs, the PE fluctuates around an average value ranging from 60% to 70%. While a possible explanation for the average PE decay is the *latency time* - the communication between CPUs becomes predominant with respect to the computational time for solving the elastic equations in small subdomains -, major fluctuations may be due to odd mesh partition and corresponding unequal distribution of degrees of freedom among processors. Obviously, for problems with larger size, fluctuations are expected to be less evident, as the mesh partitioning automatically improves.

The PE curves for the “medium” and “large” cases are in good agreement, both for the structured or the unstructured domain discretization. Figure 6 shows that the PE value for the “large” case is nearly independent on the number of CPUs, with an average value of 74.5% both for the structured and the unstructured grids. As expected, the PE trend is much smoother than for the “small” problem of figure 5.

6 APPLICATION TO A LARGE DYNAMIC SOIL-STRUCTURE INTERACTION PROBLEM: THE CASE OF UNDERGROUND TRAIN INDUCED VIBRATIONS

The analysis of surface or underground train-induced ground vibrations and their effect on human beings and surrounding structures has recently become of paramount importance, especially for environmental impact studies of new high-speed train lines crossing urban areas. This problem is particularly challenging and demanding from a computational point of view, since it requires to handle the dynamic interaction of train-track-tunnel-soil-structure. It is not surprising that this problem has been mainly tackled in the past using empirical and/or simplified models, mainly through 2D finite element approximations that typically strongly underestimate radiation damping. One of the first examples of fully 3D approaches to such problems is the coupled boundary element/finite element procedure recently proposed by [8] for the numerical analysis of free-field vibrations due to a moving load in a tunnel.

In this section we show how our approach can handle within a reasonable computer time the propagation of waves from the track through the soil, and from the soil to a realistic structure. A validation of the spectral element approach with independent solutions based on the Betti-Rayleigh theorem has already been presented elsewhere [31] for the case of a train travelling at the surface of a layered halfspace.

A thorough study of this problem is beyond the scope of this work, and will be the object of a future paper. Instead, we address here the more relevant computational aspects, with emphasis on the performance of the numerical code. A summary of the more significant results is also provided, mainly with the aims of illustrating the potential practical applications of the numerical code.

A sketch of the sample problem considered in this study is shown in Figure 7. It consists of a eight-storey building close to a railway line. For the sake of comparison and to show the capability of handling different configurations, we have considered the following cases: (i) surface railway line (denoted by “*s*” in the following), (ii) underground railway line (denoted by “*u*”), and (iii) the intermediate situation where the line is partly underground and partly located in a trench delimited by a retaining wall (denoted by “*u – s*”). Our objective is to study the vibrations induced in the soil and in the building by the passage of a two-carriages train, moving with speed $c=70$ km/h= 19.4 m/s.

Details on the assumptions adopted to simulate the complex dynamic interaction between the moving train, the track and the soil, will be omitted for simplicity. The interested reader may refer to [30] and [31] for a thorough discussion of such assumptions.

The grid discretization is suitable to accurately propagate frequencies up to 10 Hz, that is the range of the Fourier spectrum of traveling load. Note that, owing to the strong difference of the mechanical properties of soil and concrete

and to the different geometrical details as well, the grid refinement needed in the various parts of the model is quite different. Therefore, a highly unstructured mesh is needed to minimize the number of elements, as shown in Figure 7 (the “ $u - s$ ” grid has 3 888 elements and, for spectral degree equal to two, 6 043 nodes and 18 129 degrees of freedom).

For the stability requirement of the time-advancing scheme (see [11] for details), the time step is $\Delta t = 0.9 \times 10^{-5} s$ (25% of the Courant-Friedrichs-Levy value for explicit time-advancing schemes). The total duration of the simulation is 14.5s, corresponding to around 1.6×10^6 time steps. Since the single time step takes $9.53 \times 10^{-5} s$ CPU time on a single CPU of the Avogadro cluster, the complete run requires 42 hours and 38 minutes. The corresponding run with 32 CPUs requires around 1 hour and 34 minutes.

Figure 8 displays the decay with distance of the vertical displacement at ground surface, calculated for the three cases analyzed and along the three transverse profiles indicated in the same figure. Similarly, Figure 9 illustrates the snapshots of vertical displacement for the “ $u - s$ ” case, at three different times. Although the discussion of these results is out of the scope of this work, we note that for relatively low train speeds, as is the case of this study, the ground motion amplitude decay with distance is fast, so that the building is practically unaffected by the train passage.

As shown in Figure 10, although the displacement amplitudes of the building are quite small, the numerical transfer functions, obtained by the Fourier spectral ratio of the response at a given level with respect to the free-field response, reproduce correctly the vibration modes of the building, computed with an independent numerical code [31].

7 CONCLUSIONS

The spectral element method (SEM) has already become one of the most popular and effective approaches for numerical wave propagation analyses in the seismological field. This paper demonstrates how it can efficiently deal also with challenging engineering problems such as complex 3D dynamic soil-structure interaction analyses, that are generally difficult to be handled from the computational point of view by more traditional numerical techniques, such as finite elements (FEM) and finite differences. As a matter of fact, the spectral accuracy of SEM allows to reduce the number of grid points required to propagate a signal with a given wavelength (3-4 nodal points per minimum wavelength according to [11] against around twice this number for FEM), so that the number of nodal points for large 3D numerical domains is considerably reduced.

The parallel implementation discussed in detail in this work provided satisfactory results, with parallel efficiency ranging from 70% to 80%, independent of the number of CPUs.

In comparison with well-established FEM, spectral element algorithms do not enjoy the wide availability of element libraries and nonlinear analysis capabilities. The first point has often been indicated as one of the main limitations of

the method at hand. Due to the significant development of grid generators, we believe that this problem actually belongs to the past: modern all-hexahedra mesh generators [6] have proven to be able to deal with complex geometries usually encountered in dynamic soil-structure interaction problems. Of course, fully automatic generation of good quality all-hexahedra grids has still to come, but the extra effort required is well worthwhile and eases off the solution of the differential problem (see for instance [1] and references therein). The lack of non-linear constitutive models and large displacement formulation algorithms which, on the other hand, are available in the FEM literature, can be explained with the relatively recent adoption of SEM among engineers and scientists working in the computational mechanic field. Actually, there are no theoretical reasons preventing the implementation of such features within a spectral element frame (see [10] and [22], respectively). Based on our experience of developers and users, spectral elements have given evidence to enjoy the same capabilities of FEM (apart the possibility to easily deal with triangular elements) in a wide range of applications, not necessarily limited to computational mechanics [23] [24]. The introduction of new features is under study, such as two-phase media, domain reduction method [2] for coupling numerical models of different size, other constitutive material models. We do hope that this work will stimulate other researchers to contribute to this promising numerical approach.

Acknowledgments

The research at CRS4 has been supported by Sardinian Regional Authorities and Italian Ministry of Education, University and Research (MIUR, Project TAMC, Rif. n. 30 D.M. 1015 4/10/2001). The research at Politecnico di Milano has been partially funded by the Italian Ministry of Education, University and Research (MIUR Cofinanziamento 2003, Prot n. 2003080575.002). The authors are in debt with CILEA for the use of their computer resources. The authors are grateful to E. Faccioli and G. Fotia for fruitful discussion and suggestions.

References

- [1] Benzley SE, Perry E, Merkley K, Clark B, Sjaardema G. A comparison of all-hexahedral and all-tetrahedral finite element meshes for elastic and elasto-plastic analysis. *Proceedings, 4th International Meshing Roundtable, Sandia National Laboratories, October 1995*: 179-191.
- [2] Bielak J, Loukakis K, Hisada Y, Yoshimura C. Domain Reduction Method for Three-Dimensional Earthquake Modeling in Localized Regions. *Bulletin of the Seismological Society of America* 2003; 93: 817-824.
- [3] Massidda L. A Parallel Spectral Element Method for Industrial Applications: Dynamics of TCDS Collimator in the LHC. *Proceedings Of The 4th*

Joint Seminar On Industrial And Applied Mathematics, Gakuto International series, Mathematical Sciences and Applications, Gakko Tosho Pub. Co., 2005.

- [4] Canuto C, Hussaini MY, Quarteroni A, Zang TA. *Spectral methods in fluid dynamics*. Springer Verlag, New York, 1988.
- [5] Casadei F, Gabellini E, Fotia G, Maggio F, Quarteroni A. A mortar spectral/finite element method for complex 2D and 3D elastodynamic problems. *Computer Methods in Applied Mechanics and Engineering* 2002; 191: 5119-5148.
- [6] Cubit Home Page.
<http://cubit.sandia.gov/> [25 January 2006].
- [7] Clayton R, Engquist B. Absorbing boundary conditions for acoustic and elastic wave equations. *Bulletin of the Seismological Society of America* 1977; 67: 1529-1540.
- [8] Clouteau D, Arnst M, Al-Hussaini TM, Degrande G. Freefield vibrations due to dynamic loading on a tunnel embedded in a stratified medium. *Journal of Sound and Vibration* 2005; 283: 173-199.
- [9] Dautry R, Lions JL. *Analyse Mathematique et Calcul Numerique pour les Sciences et les Techniques (volume 6)*. Masson, 1988.
- [10] di Prisco C, Stupazzini M, Zambelli C. Non-linear SEM numerical analyses of dry dense sand specimens under rapid and dynamic loading. *International Journal for Numerical and Analytical Methods in Geomechanics*, 2005; submitted.
- [11] Faccioli E, Maggio F, Paolucci R, Quarteroni A. 2D and 3D elastic wave propagation by a pseudo-spectral domain decomposition method. *Journal of Seismology* 1997; 1: 237-251.
- [12] Faccioli E, Vanini M, Paolucci R, Stupazzini M. Comment on “Domain Reduction Method for Three-Dimensional Earthquake Modeling in Localized Regions, Part I: Theory,” by J. Bielak, K. Loukakis, Y. Hisada, and C. Yoshimura, and “Part II: Verification and Applications,” by C. Yoshimura, J. Bielak, Y. Hisada, and A. Fernandez. *Bulletin of the Seismological Society of America* 2005; 95: 763-769.
- [13] Furumura T, Kennett BLN, Koketsu K. Visualization of 3D Wave Propagation from the 2000 Tottori-ken Seibu, Japan, Earthquake: Observation and Numerical Simulation. *Bulletin of the Seismological Society of America* 2003; 93: 870-881.
- [14] Karypis G, Kumar V. A fast and high quality multilevel scheme for partitioning irregular graphs. *SIAM J. Sci. Comput.* 1998; 20 (1): 359-392.

- [15] Kausel E, Whitman RV, Morray JP, Elsabee F. The spring method for embedded foundations. *Nuclear Engineering and Design* 1978; 48: 377-392.
- [16] Komatitsch D, Liu Q, Tromp J, Süß P, Stidham C, Shaw JH. Simulations of ground motion in the Los Angeles basin based upon the spectral-element method. *Bulletin of the Seismological Society of America* 2004; 94: 187-206.
- [17] Komatitsch D, Martin R, Tromp J, Taylor MA, Wingate BA. Wave propagation in 2-d elastic media using a spectral element method with triangles and quadrangles. *Journal of Computational Acoustics* 2001; 9, 2: 703-718.
- [18] Komatitsch D, Vilotte JP. The spectral element method: an efficient tools to simulate the seismic response of 2D and 3D geological structures. *Bulletin of the Seismological Society of America* 1998; 88: 368-392.
- [19] Kosloff D, Baysal E. Forward modelling by the Fourier method. *Geophysics* 1982; 47: 1402-1412.
- [20] Kosloff D, Kessler D, Filho AQ, Tessmer E, Behle A, Strahilevitz R. Solutions of the equations of dynamics elasticity by a Chebyshev spectral method. *Geophysics* 1990; 55: 748-754.
- [21] Madariaga R, Olsen K, Archuleta R. Modeling dynamic rupture in a 3D earthquake fault model. *Bulletin of the Seismological Society of America* 1998; 88: 1182-1197.
- [22] Maggio F. Isoparametric mapping for Else: a feasibility study. *CRS4 Technical report 2002*
- [23] Maggio F, Massidda L, Siddi G, Vallascas M, Vicari C. A spectral element solver for large scale acoustic problems: application to external noise propagation. *CRS4 Technical report*.
- [24] Maggio F, Mazzarella G, Pitzianti C. Least squares spectral element method for 2D Maxwell equations in the frequency domain., *International Journal of Numerical Modelling: Electronic Networks, Devices and Fields* 2004; 17, 6: 509-522
- [25] Maggio F, Quarteroni A. Acoustic wave simulation by spectral methods. *East-West Journal of Numerical Mathematics* 1994; 2, 2: 129-150.
- [26] Moczo P, Bistricky E, Kristek J, Carcione J, Bouchon M. Hybrid modelling of P-SV seismic motion at inhomogeneous viscoelastic topographic structures. *Bulletin of the Seismological Society of America* 1997; 87: 1305-1323.
- [27] Owen SJ. *Non-Simplicial Unstructured Mesh Generation*. Ph.D. thesis: Carnegie Mellon University, Pittsburgh, U.S.A., 1999.
- [28] Paolucci R. Amplification of earthquake ground motion by steep topographic irregularities. *Earthquake Engineering and Structural Dynamics* 2002; 31: 1831-1853.

- [29] Paolucci R, Faccioli E, Maggio F. 3D response analysis of an instrumented hill at Matsuzaki, Japan, by a spectral method. *Journal of Seismology* 1999; 3: 191-209.
- [30] Paolucci R, Maffeis A, Scandella L, Stupazzini M, Vanini M. Numerical prediction of low-frequency ground vibrations induced by high-speed trains at Ledsgaard, Sweden. *Soil Dynamics and Earthquake Engineering* 2005; 23: 425-433.
- [31] Paolucci R, Spinelli D. Ground motion induced by train passage. *ASCE Journal of Engineering Mechanics* 2006; accepted for publication.
- [32] Priolo E. Earthquake ground simulation through the 2D spectral element method. *Journal of Computational Acoustics* 2001; 9, 4: 1561-1581.
- [33] Quarteroni A, Valli A. *Numerical approximation of partial differential equations*. Springer Verlag, Berlin, 1994.
- [34] Seriani G. 3-D large-scale wave propagation modeling by spectral element method on Cray T3E multiprocessor. *Computer Methods in Applied Mechanics and Engineering* 1998; 164: 235-247;
- [35] Stupazzini M. *A spectral element approach for 3D dynamic soil-structure interaction problems*. Ph.D. thesis: Politecnico di Milano, Italy, 2004.
- [36] Virieux J. P-SV wave propagation in heterogeneous media: Velocity-stress finite difference method. *Geophysics* 1986; 51, 889-901.
- [37] Von Estorff O, Kausel E. Coupling of boundary and finite elements for soil-structure interaction problems. *Earthquake Engineering and Structural Dynamics* 1989; 18: 1065-1075.
- [38] Xu J, Bielak J, Ghattas O, Wang J. Three-dimensional nonlinear seismic ground motion modeling in basins. *Physics of The Earth and Planetary Interiors* 2003; 137: 81-95.

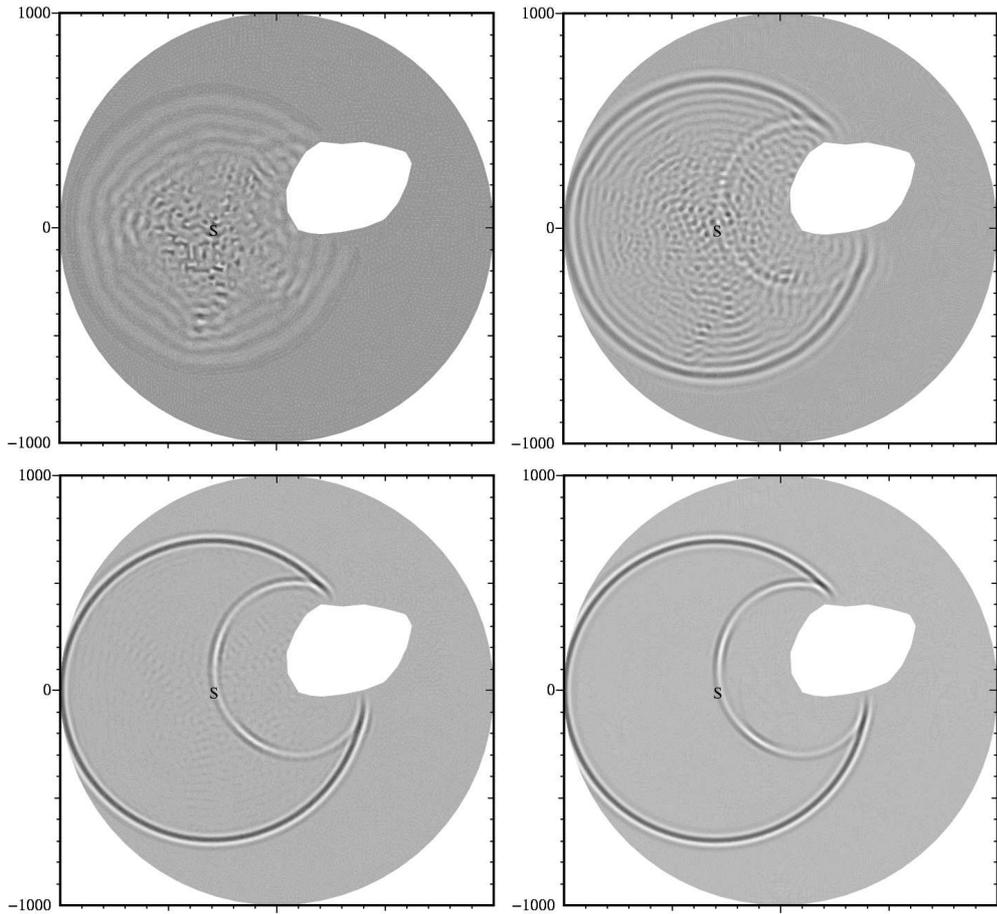


Figure 2: Snapshots of acoustic wave propagation from source S simulated for increasing spectral degrees: $n = 1$ (top-left), $n = 2$ (top-right), $n = 3$ (bottom-left) and $n = 4$ (bottom-right). Solutions provided by lower spectral degrees show significant numerical dispersion.

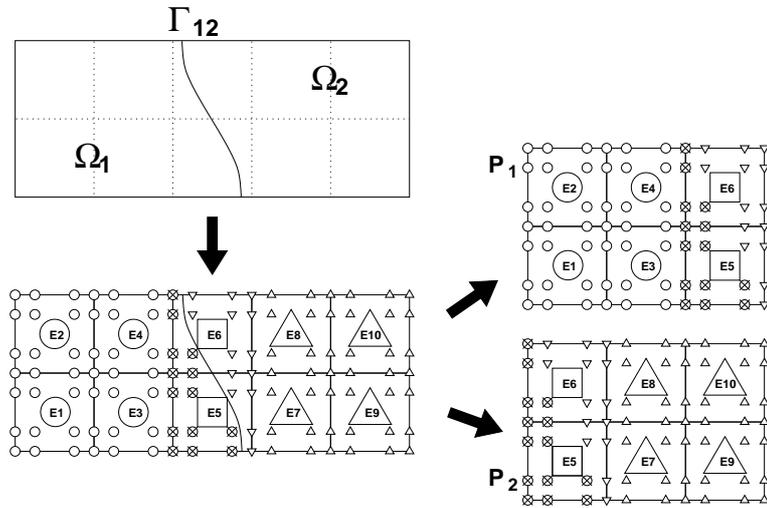


Figure 3: Node based partitioning on a 2D mesh with spectral degree $n = 3$. Processor P_1 : internal nodes are marked with \circ , border nodes with \otimes , external nodes with ∇ . Processor P_2 : internal nodes are marked with \triangle , border nodes with ∇ , external nodes with \otimes .

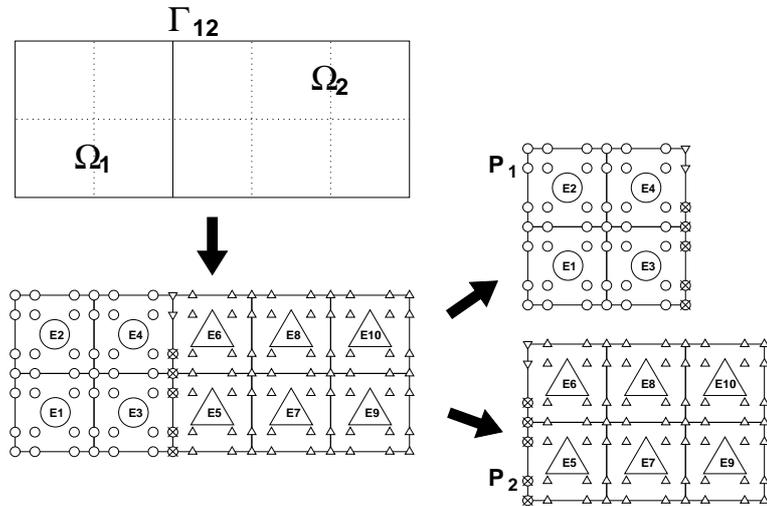


Figure 4: Element based partitioning on a 2D mesh with spectral degree $n = 3$. Processor P_1 : internal nodes are marked with \circ , border nodes with \otimes , external nodes with ∇ . Processor P_2 : internal nodes are marked with \triangle , border nodes with ∇ , external nodes with \otimes .

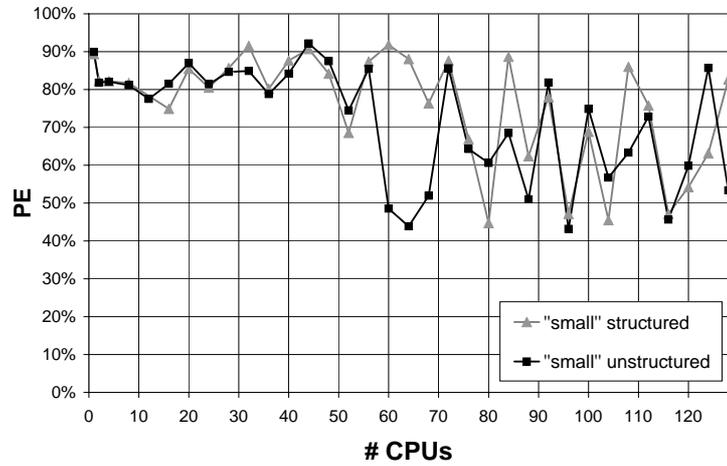


Figure 5: PE obtained on the Avogadro cluster at CILEA for the “small” benchmark case with structured (grey line) and unstructured (black line) grids.

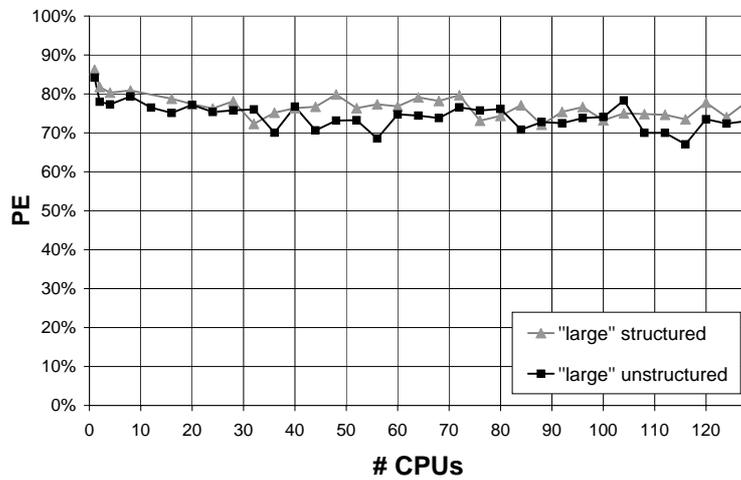


Figure 6: PE obtained on the Avogadro cluster at CILEA for the “large” benchmark case with structured (grey line) and unstructured (black line) grids.

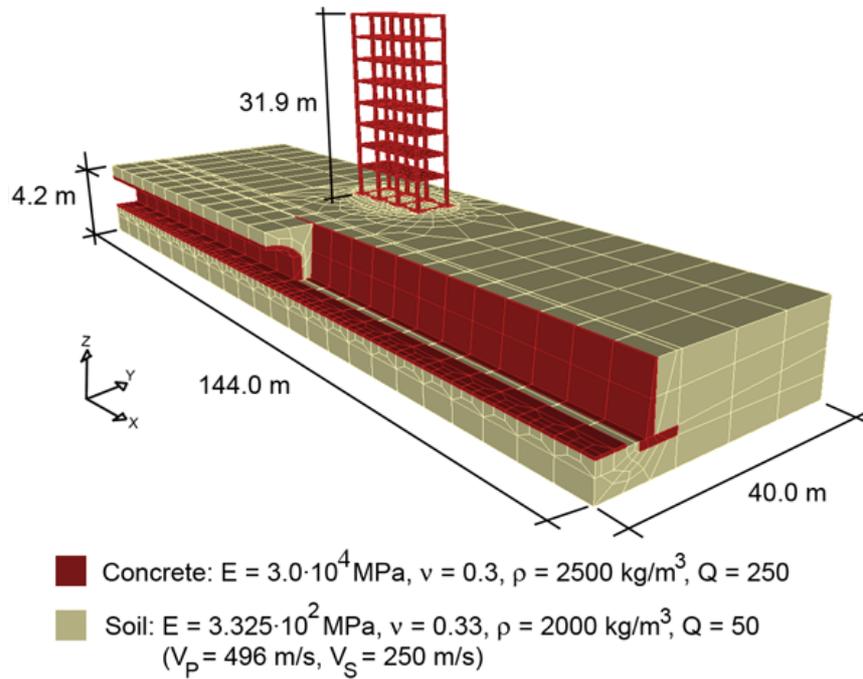


Figure 7: Spectral element mesh adopted for the simulation of train-induced vibrations in the “ $u - s$ ” case. E =Young’s modulus, ν = Poisson ratio, ρ = mass density, Q = quality factor = $1/2\xi$, ξ being the damping ratio. V_P and V_S are the P and S wave propagation velocities, respectively.

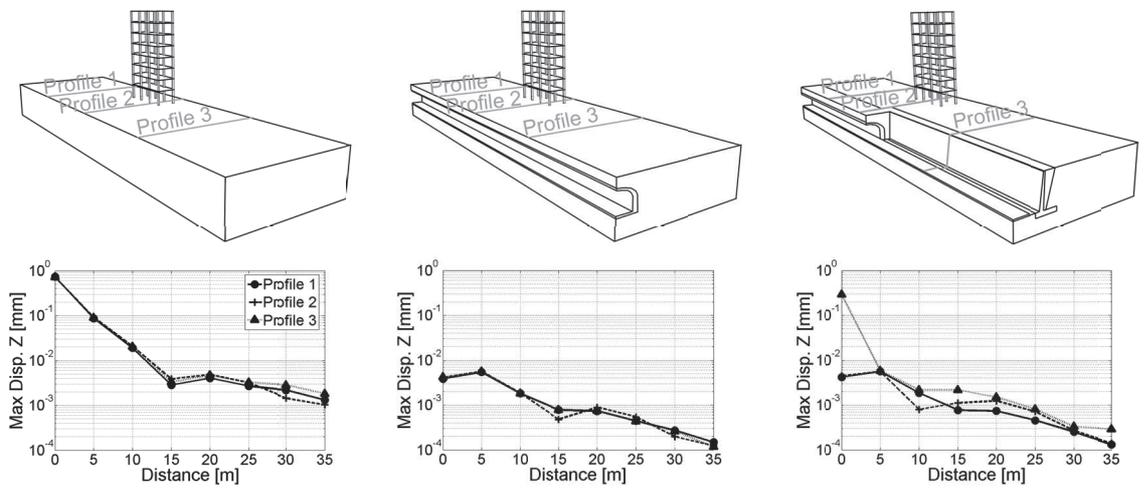


Figure 8: Top: sketch of the three configurations under study: (i) surface railway line, (ii) underground railway line, (iii) partially underground railway. Bottom: maximum vertical displacement for the three profiles.

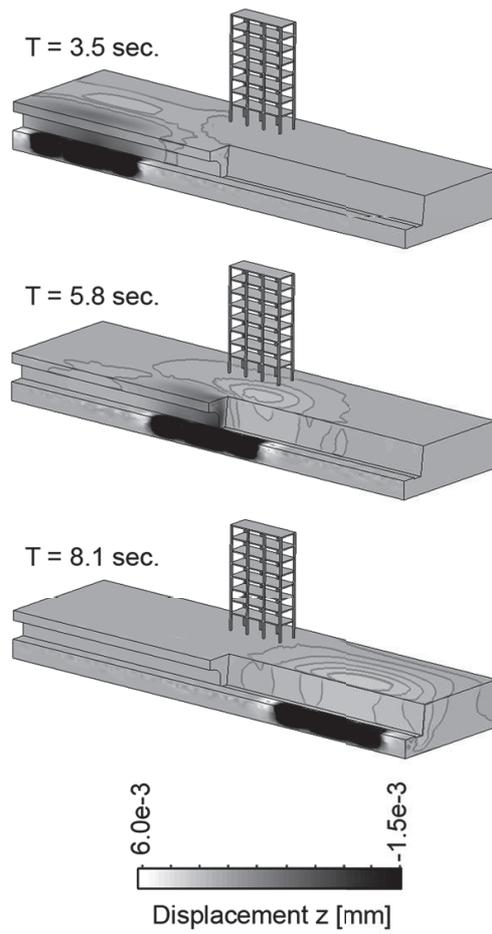


Figure 9: Snapshots of the vertical displacement at $T = 3.5, 5.8$ and 8.1 s. The gray scale is saturated in order to show the propagating wavefield induced by the traveling load.

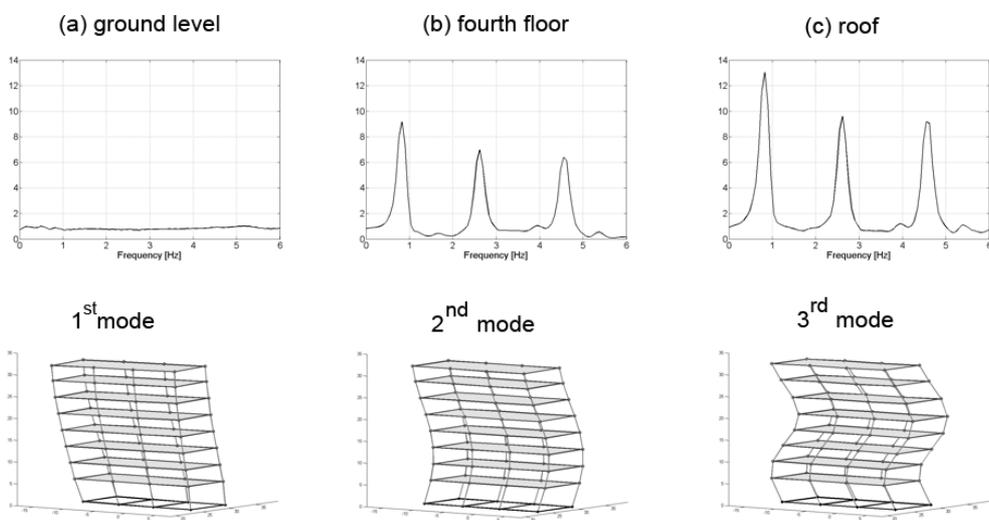


Figure 10: Top: spectral ratios with respect to free-field response at (a) ground level, (b) fourth floor and (c) roof. Bottom: vibration modes of the structure, deduced by band-pass filtering the calculated time response at the corresponding natural frequencies.