XLI CILAMCE
XLI Ibero-Latin American Congress on Computational Methods in Engineering
16-19 November, 2020 FOZ DO IGUAÇU PARANÁ - BRAZIL

Transient hyperbolic analyses considering a fully adaptive explicit time-marching formulation

Lucas Ruffo Pinto[1], Delfim Soares Jr.[2], Webe João Mansur[1]

[1]*COPPE/Federal University of Rio de Janeiro*
CEP 21941-611, Rio de Janeiro, Brazil
lucas.ruffo@engenharia.ufjf.br, webe@coc.ufrj.br
[2]*Structural Engineering Department, Federal University of Juiz de Fora*
CEP 36036-330, Juiz de Fora, MG, Brazil
delfim.soares@ufjf.edu.br

Abstract. This work presents an explicit formulation with adaptive time integrators, enhanced by sub-cycling, for time domain analyses of hyperbolic models. The method is based on single-step displacement-velocity relations, describing a truly self-starting, easy to implement technique. Its stability limit is the same as of the central difference method and it provides adaptive controllable numerical dissipation. Since the technique is explicit, it does not need to consider any solver routine, standing as a very efficient methodology. Subdomain decomposition procedures, associated to multiple time-steps and sub-cycling, are also considered herein to improve the performance of the formulation. In this case, a generic methodology is discussed, in which subdomain divisions and local time-step values are automatically evaluated. At the end of the paper, numerical results are presented in comparison to those of the central difference method and the explicit generalized α method, illustrating the effectiveness of the discussed approach.

Keywords: Time-marching, Explicit analysis, Adaptive parameters

1 Introduction

Time dependent hyperbolic equations are challenging to be solved and their analytical resolution is often unfeasible. So, in order to solve these equations, numerical methods are commonly used to find approximate solutions. These methods usually employ step-by-step time integration algorithms, solving initial value problems considering a given temporal discretization. Numerical methods are essentially divided into two groups: explicit methods [1–6], whose main advantage is that there is no need to treat systems of equations, making them very effective in terms of computational effort, yet with stability restrictions; and implicit methods [7–11], which may provide unconditional stability, but are considerably more computationally expensive per time step (for a comprehensive review, see [12]).

In this paper, an explicit formulation with adaptive time integrators is studied, considering the implementation of sub-cycling techniques to improve the efficiency and accuracy of the proposed time integration algorithm. The explicit method developed by Soares [13] is considered here as the key time-marching framework for the proposed formulation. This method is based on single-step displacement-velocity relations; is truly self-starting; presents the same stability limit as the central difference method (CDM); and, as an explicit approach, does not need to consider any solver routine. In this work, as a further development to this solution procedure, subdomain divisions and local time-step values are considered, also taking into account automated adaptive evaluations. Thus, more efficient and accurate analyses may be enabled.

The adopted time integration procedure is based on an adaptive α parameter that focuses on providing an effective numerically dissipative algorithm, aiming to eliminate the influence of spurious modes and to reduce amplitude decay errors. In this sense, the α time integrators are adaptively computed taking into account the local physical / geometrical properties of the spatial discretization, the local time-step value, and local previous time steps results. Thus, by introducing different time-steps into the analysis (considering subdomain divisions and sub-cycling techniques), the performance of the adaptive α evaluation may be also enhanced.

The techniques discussed in this work can be used to solve problems of different nature, however, here,

acoustic analyses and geophysical applications are focused. In geophysics, it is often necessary to directly analyse very heterogeneous domains. In this sense, automatic sub-cycling techniques become very attractive, since that different layers/media may be efficiently analysed considering proper subdomain divisions.

This article is divided into five sections, the first being this introduction. In the second section, the equations that govern the time integration strategy are presented. In the third section, a generic automatic methodology for sub-cycling is discussed. In the fourth section, two numerical applications are considered, illustrating the good performance of the proposed technique (in this case, the obtained results are compared to those of the CDM and of the explicit generalized α method (EG-α) [1], as well as with analytical solutions, whenever available). In the fifth and final section, conclusions are presented.

2 Governing equations and time integration strategy

The governing system of equations describing a dynamic model is given by:

$$\mathbf{M}\ddot{\mathbf{U}}(t) + \mathbf{C}\dot{\mathbf{U}}(t) + \mathbf{K}\mathbf{U}(t) = \mathbf{F}(t), \tag{1}$$

where $\mathbf{M}$, $\mathbf{C}$, and $\mathbf{K}$ are mass, damping, and stiffness matrices, respectively; $\ddot{\mathbf{U}}(t)$, $\dot{\mathbf{U}}(t)$ and $\mathbf{U}(t)$ are acceleration, velocity, and displacement vectors, respectively; and $\mathbf{F}(t)$ stands for the force vector. The initial conditions of the model are given by: $\mathbf{U}^0 = \mathbf{U}(0)$ and $\dot{\mathbf{U}}^0 = \dot{\mathbf{U}}(0)$, where $\mathbf{U}^0$ and $\dot{\mathbf{U}}^0$ stand for initial displacement and velocity vectors, respectively.

In this work, the Finite Element Method (FEM) is used for the spatial discretization since geological problems take great advantage of its ability to work with irregular geometries. By considering the standard FEM, the domain of the problem is divided into elements, allowing the calculation of local matrices and vectors, which can then be assembled to generate the global matrices $\mathbf{M}$, $\mathbf{C}$ and $\mathbf{K}$, and vector $\mathbf{F}$.

For the time-domain solution of the system of equations (1), the following algorithm is proposed by Soares [13]:

$$\left(\mathbf{M}_e + \tfrac{1}{2}\Delta t\mathbf{C}_e\right)\dot{\mathbf{U}}_e^{n+1} = \int_{t_n}^{t_{n+1}} \mathbf{F}_e(t)\,\mathrm{d}t + \mathbf{M}_e\dot{\mathbf{U}}_e^n - \tfrac{1}{2}\Delta t\mathbf{C}_e\dot{\mathbf{U}}_e^n - \mathbf{K}_e\left(\Delta t\mathbf{U}_e^n + \tfrac{1}{2}\alpha_e^n\Delta t^2\dot{\mathbf{U}}_e^n\right), \tag{2a}$$

$$\mathbf{U}^{n+1} = \mathbf{U}^n + \tfrac{1}{2}\Delta t\dot{\mathbf{U}}^n + \tfrac{1}{2}\Delta t\dot{\mathbf{U}}^{n+1}, \tag{2b}$$

where Δt represents the time-step, and $\mathbf{U}^n$ and $\dot{\mathbf{U}}^n$ are the approximations of $\mathbf{U}(t^n)$ and $\dot{\mathbf{U}}(t^n)$, respectively. In eq. (2a), the subscript e indicates that a variable is locally defined, at an element level. Once eq. (2a) is assembled, the velocities of the model can be computed, and the displacements can then be evaluated following eq. (2b).

Considering the α parameter (see eq. (2a)), which controls numerical damping, the strategy is to adopt $\alpha > 1$ wherever and whenever numerical damping may be necessary, and $\alpha = 1$ otherwise. This is automatically carried out here based on an oscillatory criterion. In other words, if the computed displacement response of a degree of freedom of the model oscillates, numerical dissipation is introduced into the analysis. A φ parameter is calculated (at each time step and for each element) to control the local activation of the numerical damping. The calculation of this oscillatory parameter is given by: $\varphi_e^n = \sum_{i=1}^{d_e} \left|\left|u_i^n - u_i^{n-2}\right| - \left|u_i^n - u_i^{n-1}\right| - \left|u_i^{n-1} - u_i^{n-2}\right|\right|$, where d_e stands for the total amount of degrees of freedom of the element. Thus, when the oscillatory parameter is not null, at least one degree of freedom of the element is oscillating. In this case, the algorithm activates maximal numerical dissipation at the maximal sampling frequency of the element Ω_e^{max} and, consequently, dissipates its highest modes more effectively. So, when the φ_e^n parameter is different from zero, α_e^n assumes the value provided by the following expression:

$$\alpha_e = 2\left[\left(1 + \frac{\varsigma_e\Delta t}{2\rho_e}\right)\left(\frac{2}{\Omega_e^{max}}\right)^2\right]^{\frac{1}{2}} - 1 - \frac{\varsigma_e\Delta t}{2\rho_e}\left(\frac{2}{\Omega_e^{max}}\right)^2, \tag{3}$$

where ρ_e and ς_e stand for physical properties of the medium (mass density and viscous damping coefficient, respectively). For null φ_e^n values, $\alpha_e^n = 1$ is considered.

3 Sub-cycling

Sub-cycling is a subdomain decomposition associated with computations at several time intervals. This technique allows a domain to be modelled considering different refinement levels without limiting its explicit time-marching solution to be restricted to its shortest critical temporal discretization, allowing greater time-steps values to be considered for different subdomains, enabling lower computational costs. However, this approach must be properly considered, once excessive subdivisions may provide deterioration in both accuracy and efficiency. Here, the following algorithm is considered to define the subdomain decomposition: (i) calculate the critical time-steps of all elements, finding the smallest Δt_e of the model (i.e., Δt_1, where $\Delta t_1 = min(\Delta t_e)$), which is the basic time-step for the controlled subdivision of the domain; (ii) with Δt_1 defined, calculate subsequent time-step values as multiple of the power of 2 of this minimal time-step value (i.e., calculate Δt_i, where $\Delta t_i = 2^{(i-1)}\Delta t_1$); (iii) associate each element to an computed time-step value (i.e., to Δt_i, where $\Delta t_i \leq \Delta t_e < \Delta t_{i+1}$ and i indicates the subdomain of that element); (iv) associate a time-step value (i.e., associate a subdomain) to each degree of freedom of the model considering the lowest time-step value of its surrounding elements.

Once the subdomains of the model are stablished, displacement and velocity values along the boundaries of these subdomains may be interpolated so that explicit time-marching solutions, considering different time-step values, can be regularly carried out along the subdivided model. In this work, the following expressions are adopted for these interpolations:

$$\mathbf{U}(t) = \tfrac{1}{2\Delta t}\left(\dot{\mathbf{U}}^{n+1} - \dot{\mathbf{U}}^{n}\right) t^2 + \dot{\mathbf{U}}^{n} t + \mathbf{U}^{n}, \tag{4a}$$

$$\dot{\mathbf{U}}(t) = \tfrac{1}{\Delta t}\left(\dot{\mathbf{U}}^{n+1} - \dot{\mathbf{U}}^{n}\right) t + \dot{\mathbf{U}}^{n}, \tag{4b}$$

where t is the current increment of time ($0 \leq t \leq \Delta t$) for the focused subdomain and Δt is the time-step value of the degree of freedom being interpolated, which is related to other subdomain.

4 Numerical applications

In this section, two numerical applications are considered, briefly illustrating the performance and potentialities of the adaptive explicit time-marching technique with subcycling. First, a homogeneous acoustic model is considered, and, subsequently, the propagation of acoustic waves in the Marmousi2 model by Martin et al. [14] are analysed. The computed results are compared to those of the CDM, EG-α and α adaptive method without subcycling. The EG-α is considered here since it is a very well-known dissipative method. In the following analyses, lumped mass and damping matrices are always considered.

4.1 Application 1

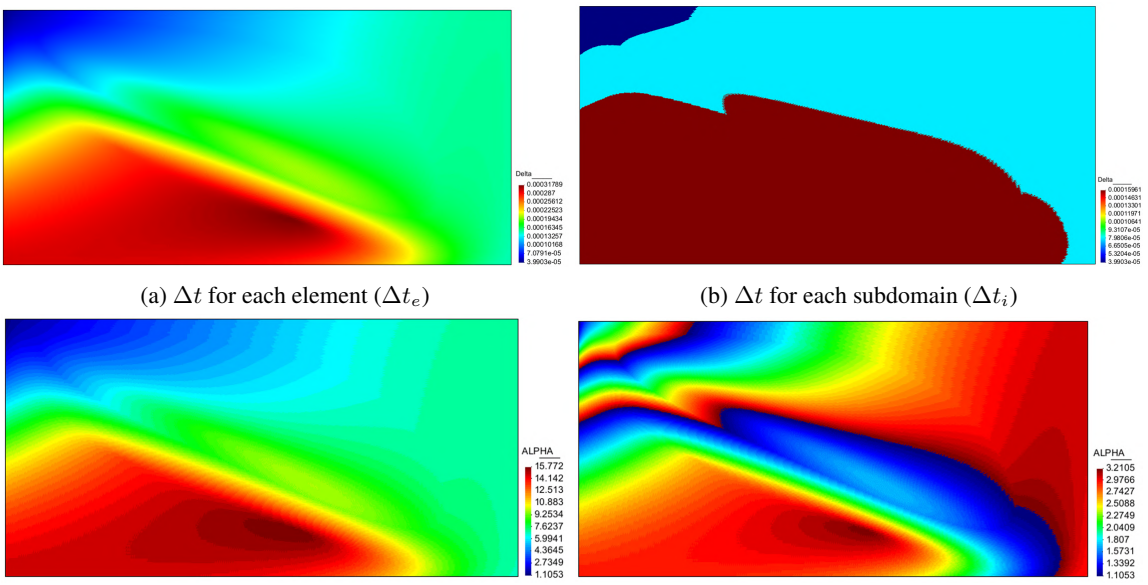
In this first example, a physically damped, acoustic model is analysed. A constant unit pressure is prescribed along the left boundary of the body and null pressure values are prescribed along the remaining boundaries. The model is defined by an unitary side length, mass density and viscous damping coefficient, as well as a wave propagation velocity of $10m/s$. The symmetry of the problem is regarded and only its upper half is spatially discretized by the FEM. The generated mesh is composed of 125000 elements and refinement towards the upper-left border of the model is considered. The exact solution for this application can be found in Soares [13].

In Fig. 1, the obtained subdomain decomposition and active α values are depicted. In this context, in Fig. 1a and 1b , the Δt_e and Δt_i values are described, respectively, whereas, in Fig. 1c and 1d, the resulting active α values (see eq. 3) are depicted for analyses without and with this subdomain decomposition, respectively.

In Fig. 2, snap-shots of the results computed at time t = 0.125s and t = 0.25s are depicted, considering the selected different time-marching techniques. As can be observed, the CDM and EG-α do not provide appropriate results and spurious oscillations dominate the computed responses. On the other hand, appropriate results are provided by the $\alpha-$adap and $\alpha-$adap/sub, demonstrating their good performance.

Table 1 shows the CPU times and the relative errors (computed at the middle of the model) for each approach. It is noteworthy that the $\alpha-$adap/sub method runs with different time-steps per subdomain and its largest time-step value is provided in the table. As Table 1 indicates, the multiple time-step approach allows reducing the total CPU time of the analysis in approximately 37%. In addition, by considering multiple time-step values, more appropriate

α values can be computed (see Fig. 1d), allowing improving the accuracy of the analysis, as it is also described in Table 1 (results are obtained using an Intel Core i7 -9750H 2.60GHz processor, with the multiplications of the element stiffness matrices parallelized with OpenMP using 8 threads).



(a) Δt for each element (Δt_e)

(b) Δt for each subdomain (Δt_i)

(c) active α values for the $\alpha-$adap

(d) active α values for the $\alpha-$adap/sub

Figure 1. Subdomain decomposition and active α values for the first example.

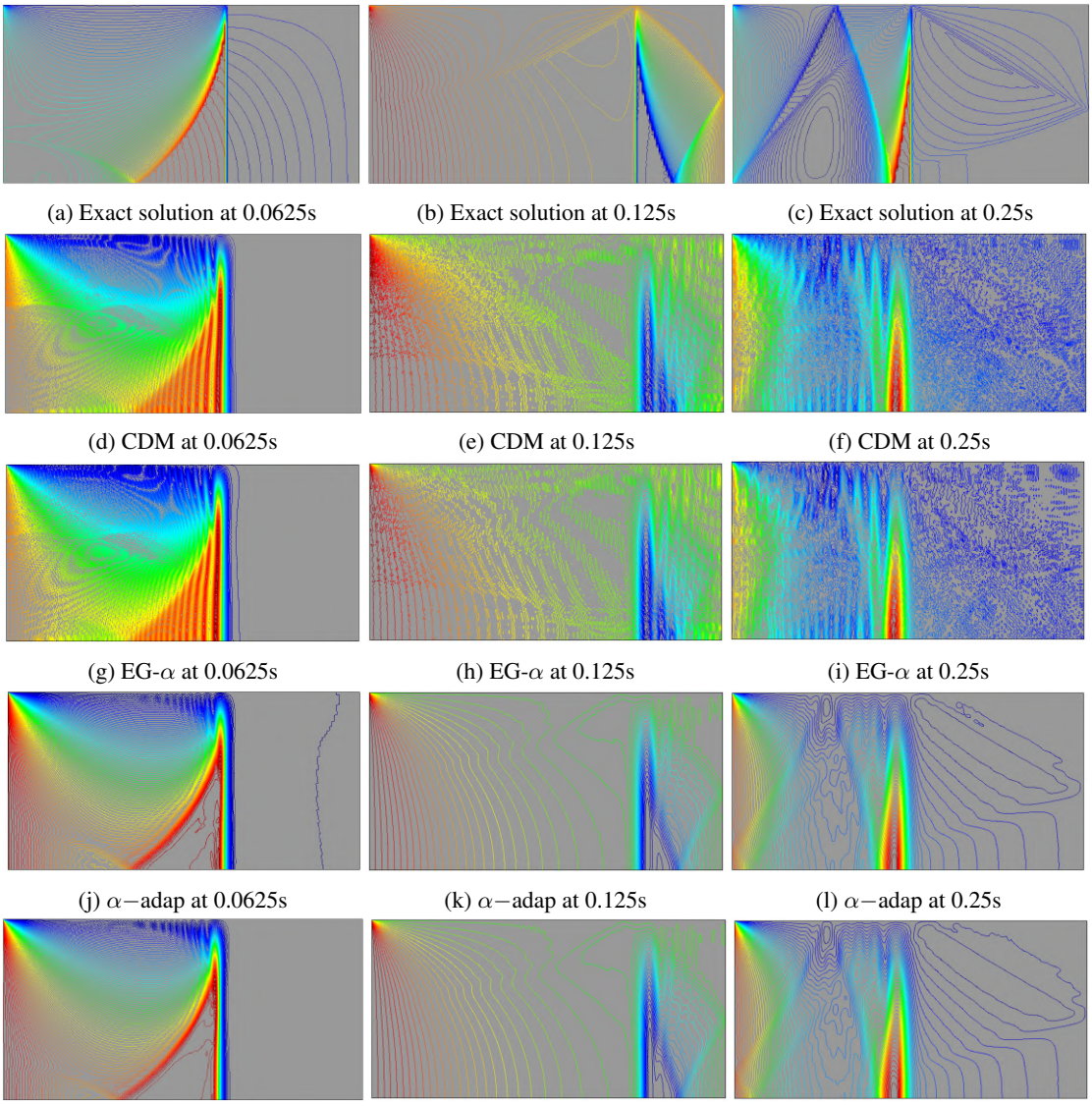
Table 1. Performance of the analyses for the first example

Method	$\Delta t(10^{-5}s)$	Steps	Error (10^{-1})	CPU time (s)
CDM	0.39903	6266	1.6369	274.570
EG-α	0.35963	6952	1.6195	288.130
$\alpha-$adap	0.39903	6266	1.2314	273.420
$\alpha-$adap/sub	1.59612	1567	1.1580	109.350

4.2 Application 2

In this second example, an extension of the original Marmousi model created by Martin et al. [14] is analysed. The model has a lateral extension of 17 km and a depth of 3.5 km and includes a total of 199 geological layers, as well as an extended water layer of 450 m deep. Here, the original finite difference synthetic data are transformed into a FEM mesh with 224731 nodes and 223672 linear square elements. Since this FEM mesh is created from a finite difference discretization, its elements are structured and they all have the same size. Thus, just the wave propagation velocity of each material controls the variability of the element time-steps. In Fig. 3, the obtained subdomain decomposition and active α values are depicted, analogously to Fig. 1.

Fig. 4 shows the computed fields along the model, taking into account the selected solution procedures, and Table 2 describes the performance of the analyses. As one can observe, the subdomain decomposition and sub-cycling techniques performed well for this complex heterogeneous model, allowing evaluating accurate solutions at lower computational costs.



(a) Exact solution at 0.0625s (b) Exact solution at 0.125s (c) Exact solution at 0.25s

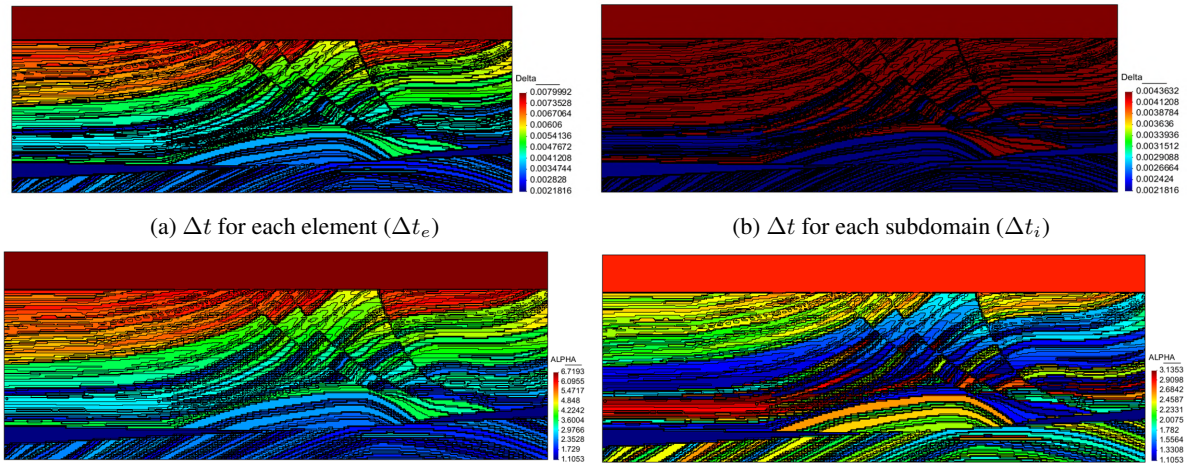
(d) CDM at 0.0625s (e) CDM at 0.125s (f) CDM at 0.25s

(g) EG-α at 0.0625s (h) EG-α at 0.125s (i) EG-α at 0.25s

(j) $\alpha-$adap at 0.0625s (k) $\alpha-$adap at 0.125s (l) $\alpha-$adap at 0.25s

(m) $\alpha-$adap/sub at 0.0625s (n) $\alpha-$adap/sub at 0.125s (o) $\alpha-$adap/sub at 0.25s

Figure 2. Exact and computed fields along the discretized domain, at t = 0.0625s, 0.125s and 0.25s.



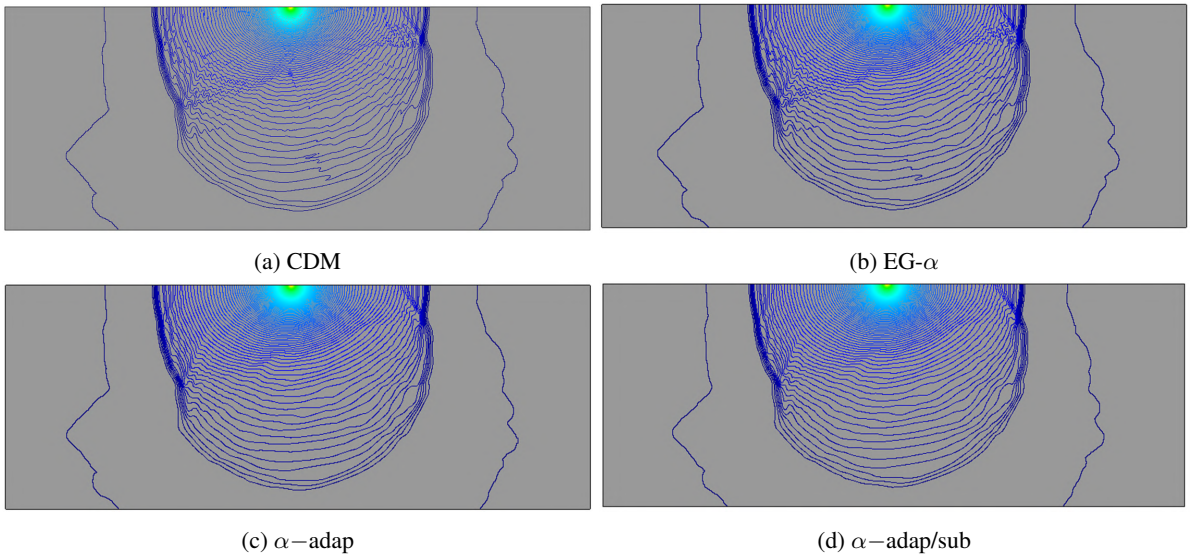
(a) Δt for each element (Δt_e) (b) Δt for each subdomain (Δt_i)

(c) active α values for the $\alpha-$adap (d) active α values for the $\alpha-$adap/sub

Figure 3. Subdomain decomposition and active α values for the second example.

Table 2. Performance of the analyses for the second example

Method	$\Delta t(10^{-3}s)$	Steps	CPU time (s)
CDM	2.07273	700	43.320
EG-α	1.86808	780	49.260
$\alpha-$adap	2.07273	700	43.060
$\alpha-$adap/sub	4.14545	350	25.400



(a) CDM (b) EG-α

(c) $\alpha-$adap (d) $\alpha-$adap/sub

Figure 4. Computed fields along the discretized domain, at t=1.4s.

5 Conclusions

This paper describes an explicit fully-adaptive time-marching formulation for hyperbolic models. In this approach, both time-step and time integrator values adapt to the properties of the discretized model, allowing providing a more efficient and accurate solution methodology. Two examples are discussed in this work, illustrating the good performance of the proposed approach. As the examples demonstrate, the proposed formulation allows obtaining better results than standard solution procedures, considering lower computational efforts. In the second example, a complex heterogeneous model is studied, highlighting the robustness of the proposed automated formulation for multiple time-step analyses.

Acknowledgements

The financial support by CNPq (Conselho Nacional de Desenvolvimento Científico e Tecnológico), CAPES (Coordenação de Aperfeiçoamento de Pessoal de Nível Superior) and PETROBRAS (project 21066) is greatly acknowledged.

Authorship statement

The authors hereby confirm that they are the sole liable persons responsible for the authorship of this work, and that all material that has been herein included as part of the present paper is either the property (and authorship) of the authors, or has the permission of the owners to be included here.

References

[1] Hulbert, G. M. & Chung, J., 1996. Explicit time integration algorithms for structural dynamics with optimal numerical dissipation. *Computer Methods in Applied Mechanics and Engineering*, vol. 137, n. 2, pp. 175–188.

[2] Soares, D., 2016. A novel family of explicit time marching techniques for structural dynamics and wave propagation models. *Computer Methods in Applied Mechanics and Engineering*, vol. 311, pp. 838–855.
[3] Zhang, H. & Xing, Y., 2019. Two novel explicit time integration methods based on displacement-velocity relations for structural dynamics. *Computers & Structures*, vol. 221, pp. 127–141.
[4] Soares, D., 2020. A novel time-marching formulation for wave propagation analysis: The adaptive hybrid explicit method. *Computer Methods in Applied Mechanics and Engineering*, vol. 366, pp. 113095.
[5] Kim, W., 2019. An accurate two-stage explicit time integration scheme for structural dynamics and various dynamic problems. *International Journal for Numerical Methods in Engineering*, vol. 120, n. 1, pp. 1–28.
[6] Tamma, K. K. & Namburu, R. R., 1990. A robust self-starting explicit computational methodology for structural dynamic applications: architecture and representations. *International Journal for Numerical Methods in Engineering*, vol. 29, n. 7, pp. 1441–1454.
[7] Hulbert, J. C. G. & Chung, J., 1993. A time integration algorithm for structural dynamics with improved numerical dissipation: The generalized-a method. *J. Appl. Math*, vol. 60, pp. 371–375.
[8] Hilber, H. M., Hughes, T. J., & Taylor, R. L., 1977. Improved numerical dissipation for time integration algorithms in structural dynamics. *Earthquake Engineering & Structural Dynamics*, vol. 5, n. 3, pp. 283–292.
[9] Wood, W., Bossak, M., & Zienkiewicz, O., 1980. An alpha modification of newmark's method. *International Journal for numerical methods in Engineering*, vol. 15, n. 10, pp. 1562–1566.
[10] Newmark, N. M., 1959. A method of computation for structural dynamics. *Journal of the engineering mechanics division*, vol. 85, n. 3, pp. 67–94.
[11] Soares, D., 2015. A simple and effective new family of time marching procedures for dynamics. *Computer Methods in Applied Mechanics and Engineering*, vol. 283, pp. 1138–1166.
[12] Tamma, K. K., Zhou, X., & Sha, D., 2000. The time dimension: a theory towards the evolution, classification, characterization and design of computational algorithms for transient/dynamic applications. *Archives of Computational Methods in Engineering*, vol. 7, n. 2, pp. 67–290.
[13] Soares, D., 2019. A model/solution-adaptive explicit-implicit time-marching technique for wave propagation analysis. *International Journal for Numerical Methods in Engineering*, vol. 119, n. 7, pp. 590–617.
[14] Martin, G. S., Wiley, R., & Marfurt, K. J., 2006. Marmousi2: An elastic upgrade for marmousi. *The leading edge*, vol. 25, n. 2, pp. 156–166.