A Space-Time Finite Element Method for elastodynamics problems : elementary examples of 4D remeshing using simplex elements

Serge Dumont, Franck Jourdan, Tarik Madani

Mathematical and Computational Applications

MDPI

Article

4D Remeshing Using a Space-Time Finite Element Method for Elastodynamics Problems

Serge Dumont [1], Franck Jourdan [2,*] and Tarik Madani [2]

[1] Institut Montpelliérain Alexander Grothendieck, University Montpellier, CNRS, 34090 Montpellier, France; serge.dumont@umontpellier.fr

[2] Laboratory of Mechanics and Civil Engineering, University Montpellier, CNRS, 34090 Montpellier, France; tarik.madani@umontpellier.fr

* Correspondence: franck.jourdan@umontpellier.fr; Tel.: +33-467-149-633

Received: 10 March 2018; Accepted: 22 May 2018; Published: 25 May 2018

Abstract: In this article, a Space-Time Finite Element Method (STFEM) is proposed for the resolution of mechanical problems involving three dimensions in space and one in time. Special attention will be paid to the non-separation of the space and time variables because this kind of interpolation is well suited to mesh adaptation. For that purpose, we have developed a technique of 4D mesh generation adapted to space-time remeshing. A difficulty arose in the representation of 4D finite elements and meshes. This original technique does not require coarse-to-fine and fine-to-coarse mesh-to-mesh transfer operators and does not increase the size of the linear systems to be solved, compared to traditional finite element methods. Space-time meshes are composed of simplex finite elements. Computations are carried out in the context of the continuous Galerkin method. We have tested the method on a linearized elastodynamics problem. Our technique of mesh adaptation was validated on elementary examples and applied to a problem of mobile loading. The convergence and stability of the method are studied and compared with existing methods. This work is a first implementation of 4D space-time remeshing. A stability criterion for the method is established, as well as a convergence rate of about two. Using simplex elements, it is possible to develop a technique of mesh adaptation able to follow a mobile loading zone.

Keywords: finite elements; space-time; elastodynamics; mesh adaptation

1. Introduction

The STFEM (Space-Time Finite Element Method) can be regarded as an extension of the classical finite element method, applied to a boundary problem resulting from a non-stationary problem. Currently, several approaches exist. One can quote for example the Large Time INcrement method (LATIN [1]), the discontinuous Galerkin method [2–4] and our method, which is a continuous Galerkin method [5,6]. In most publications on the discontinuous Galerkin method, like in [7], the interpolation functions are assumed to be a product of functions of space variables and functions of time variables. We will see in this paper that special attention will be paid to the non-separation of the space and time variables. The reason for this choice is not motivated by the accuracy of the numerical results, but rather by what constitutes the aim of our study: remeshing. We will see that this kind of interpolation is well suited to mesh adaptation. The space-time mesh adaptation we developed is based on a method of mesh generation not structured in space and time. The construction of 4D meshes collides with the limits of representation. To overcome this difficulty, we propose an automatic method of construction inspired by what can be achieved in 2D and 3D. Our technique of mesh adaptation was applied to a problem of mobile load like contact forces. Our approach makes possible the building of an evolutionary mesh able to follow the clamping zone.

Moreover, this technique does not require a mesh-to-mesh transfer operator and allows the preservation of the exact sizes of the linear systems on each space-time slab.

Let us note that one of the drawbacks of the STFEM such as defined in the works of [8,9] is the size of the linear systems to be solved, since it is necessary to solve the full 4D problem at once. The use of a laminated mesh allows us to avoid the assembling of the total matrix of the problem and permits us to consider only submatrices. This drastically reduces the size of the systems to be solved. The size of these linear systems is exactly the same as that obtained in the case of approaches coupling an incremental method of finite differences type to solve time integration, with the "classical" finite element method being used to solve the space problem.

Another large group of methods is based on semi-discretisation, whereby finite elements are used in space and finite differences are used in time. Even if this well-known technique is simpler to use in a classical framework, the remeshing required is expensive due to the necessity of the construction of interpolation/restriction operators between the grids.

The paper is organized as follows. In Section 2, the elastodynamics problem is formulated, and the space-time finite element method is developed. The 4D mesh generation is presented in Section 3, and a paragraph is specially devoted to adaptive mesh refinement. Numerical results are presented and discussed in Section 4.

2. Principle of the Method

We consider the motion of an elastic body within the small perturbations hypothesis. Let Ω be the set taken up by the body and $[0, T]$ a time interval. The body is submitted to volume force density f_d, boundary force density F_d on its boundary part $\partial_1\Omega$ and imposed displacements u_d on its boundary part $\partial_0\Omega$ ($\partial\Omega = \partial_0\Omega \cup \partial_1\Omega$, $\partial_0\Omega \cap \partial_1\Omega = \emptyset$). The dynamic problem is: seek the displacement u and the Cauchy stress tensor σ such that:

$$\begin{cases} div(\sigma(x,t)) + f_d(x,t) = \rho\ddot{u}(x,t) & \forall (x,t) \in \Omega \times]0,T[\\ \sigma(x,t)n(x,t) = F_d(x,t) & \forall (x,t) \in \partial_1\Omega \times [0,T] \\ u(x,t) = u_d(x,t) & \forall (x,t) \in \partial_0\Omega \times [0,T] \\ u(x,t) = u_0(x) & \forall (x,t) \in \Omega \times \{0\} \\ \dot{u}(x,t) = \dot{u}_0(x) & \forall (x,t) \in \Omega \times \{0\} \\ \sigma(x,t) = a\varepsilon(x,t) & \forall (x,t) \in \Omega \times [0,T] \end{cases} \tag{1}$$

where ρ is the specific mass, $\ddot{u}$ is the second derivative of the displacement with respect to time, u_0 is the initial displacement, $\dot{u}_0$ is the initial velocity, a is the Hooke tensor and ε is the infinitesimal linear strain tensor. The aim of this study is to use a finite element method. Then, the previous dynamic problem has to be considered as a boundary problem on the time interval $[0, T]$. For that purpose, as in the cases of the discontinuous Galerkin method [2,10,11] and the LArge Time INcrement (LATIN) method [1,12], the variational formulation is written on the whole space-time domain $\Omega \times [0, T]$. The variational formulation of the previous boundary problem can be written as follows:

Find $u \in U_{ad}$ such that:

$$\int_0^T \int_\Omega (\rho\ddot{u}v + a\varepsilon(u) : \varepsilon(v) - f_d v)\, dxdt = \int_0^T \int_{\partial_1\Omega} F_d v dsdt, \tag{2}$$

$$\forall v \in U_{ad}^0$$

where U_{ad} is the set of displacements, regular enough, which verifies the boundary kinematic conditions and the initial conditions, v is the virtual displacement and U_{ad}^0 is the set of virtual displacements, regular enough, which verify boundary kinematic conditions only.

The first term on the left-hand side of Equation (2) is integrated by parts in time in order to determine the first derivative of u and the initial velocity. It gives:

$$\begin{aligned}\int_0^T \int_\Omega \rho \ddot{u} v dx dt = & -\int_0^T \int_\Omega \rho \dot{u} \dot{v} dx dt \\ & + \int_\Omega [\rho \dot{u}_T(x) v(x,T) - \rho \dot{u}_0(x) v(x,0)] dx\end{aligned} \tag{3}$$

where $\dot{u}_T(x)$ is the velocity at time $t = T$.

The space-time finite element method (STFEM) was firstly proposed in [5,6]. Their discretisation used structured space-time meshes obtained as the Cartesian product of spatial elements and a time interval, which is not generally suitable for space-time mesh adaptations. Since then, numerous papers on STFEM have been published. Most of them, like [11,13], deal with the discontinuous Galerkin method in time, but the discretisation also uses structured meshes obtained as the Cartesian product of spatial elements and a time interval. However, the STFEM proposed in [2] has been developed on unstructured meshes. It also employs the discontinuous Galerkin method in time and incorporates stabilizing terms of the least squares type. The space and time discontinuities of all variables are taken into account. In our study, we use a continuous Galerkin method. Classical Lagrange polynomials are used. The finite elements are isoparametric. On a space-time finite element E_e (Figure 1), the displacement verifies:

$$u(x,t) = \sum_{i=1}^{n_e} \varphi_i^e(x,t) u_i^e \tag{4}$$

where n_e is the total number of nodes of the element E_e, φ_i^e are the interpolation functions and u_i^e the nodal displacements. Using matrix notation, one has:

$$u(x,t) = N_e(x,t) U_e \text{ where } U_e = (u_1^e, ..., u_{n_e}^e)^T \text{ and } N_e(x,t) = (\varphi_1^e(x,t), ..., \varphi_{n_e}^e(x,t)) \tag{5}$$

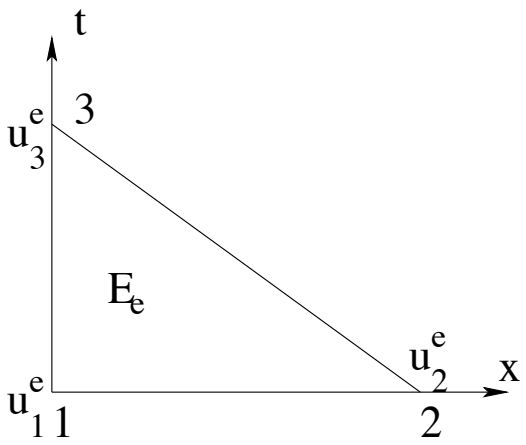

Figure 1. 2D space-time finite element.

The same interpolation is used for the virtual displacement v. Then:

$$v(x,t) = N_e(x,t) V_e \text{ where } V_e = (v_1^e, ..., v_{n_e}^e)^T \tag{6}$$

Let p be the total number of space-time elements; the previous discretization gives:

$$\int_0^T \int_\Omega \rho \ddot{u} \dot{v} dx dt = \sum_{e=1}^{p} V_e^T M_e U_e \tag{7}$$

where:

$$M_e = \int_0^T \int_{E_e} \rho \frac{\partial N_e^T}{\partial t} \frac{\partial N_e}{\partial t} dxdt \tag{8}$$

is the elementary matrix relative to the inertia forces. One can notice that M_e is symmetric. Concerning the discretization of the initial and final impulses' contributions, one has:

$$\int_\Omega [\rho \dot{u}_T(x) v(x,T) - \rho \dot{u}_0(x) v(x,0)] dx = \sum_{e=1}^{p} V_e^T \Lambda_e \tag{9}$$

where Λ_e is the elementary vector relative to the initial and final impulses. It is defined by:

$$\Lambda_e = [\int_{E_e \bigcap \Omega_T} \rho N_e^T \dot{u}_T dx - \int_{E_e \bigcap \Omega_0} \rho \dot{u}_0 dx] \tag{10}$$

where Ω_0 is the domain at time $t = 0$ and Ω_T is the domain at time $t = T$.

Similarly, let B_e be the matrix such that:

$$\varepsilon(u(x,t)) = B_e(x,t) U_e \,, \tag{11}$$

the virtual works of internal and external forces are respectively discretized by:

$$\int_0^T \int_\Omega a\varepsilon(u) : \varepsilon(v) dxdt = \sum_{e=1}^{p} V_e^T K_e U_e \tag{12}$$

and:

$$\int_0^T \int_\Omega f_d v dxdt + \int_0^T \int_{\partial_1 \Omega} F_d v ds dt = \sum_{e=1}^{p} V_e^T F_e \tag{13}$$

where the elementary matrix K_e relative to internal forces is:

$$K_e = \int_0^T \int_{E_e} B_e^T a B_e dxdt \tag{14}$$

and the elementary vector F_e relative to external forces is:

$$F_e = \int_0^T \int_{E_e} N_e^T f_d dxdt + \int_0^T \int_{E_e \bigcap \partial_1 \Omega} N_e^T F_d ds dt. \tag{15}$$

This space-time discretization leads to the following linear system:

$$([\widetilde{M_u}] + [\widetilde{K_u}])\{U\} = \{F_u\} + \{\Lambda\} \tag{16}$$

where $[\widetilde{M_u}]$ is the assembled matrix relative to the inertia forces, $[\widetilde{K_u}]$ is the assembled matrix relative to the internal forces, $\{F_u\}$ is the nodal vector of external forces, Λ is the nodal vector of impulses and $\{U\}$ is the nodal vector of displacements. One can note that the matrices $[\widetilde{M_u}]$ and $[\widetilde{K_u}]$ are symmetric. In order to have band matrices, and because we have in mind making computations incrementally in time, the meshes are built to be stratified in time, as in Figure 2. Moreover, the node numbering is conducted in such a way that all nodes in a same stratum have close numbers, then the left-hand side of the system (16) verifies.

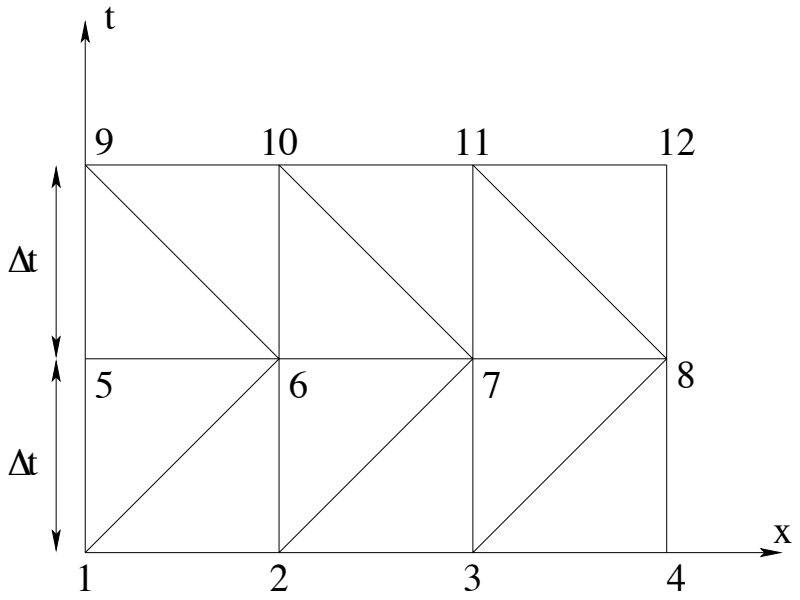

Figure 2. 2D regular space-time mesh.

$$([\widetilde{M_u}] + [\widetilde{K_u}])\{U\} = [T]\{U\} =$$

$$\begin{pmatrix} [T_{11}] & [T_{12}] & 0 & 0 & 0 & 0 \\ [T_{21}] & [T_{22}] & [T_{23}] & 0 & 0 & 0 \\ 0 & [T_{32}] & [T_{33}] & [T_{34}] & 0 & 0 \\ . & . & . & . & . & . \\ 0 & 0 & 0 & [T_{n/n-1}] & [T_{n/n}] & [T_{n/n+1}] \\ 0 & 0 & 0 & 0 & [T_{n+1/n}] & [T_{n+1/n+1}] \end{pmatrix} \begin{pmatrix} \{U_0\} \\ \{U_1\} \\ . \\ . \\ . \\ \{U_n\} \end{pmatrix}.$$

Using the space-time mesh described in Figure 2, one has:

$$\{U_j\} = \begin{pmatrix} u_{j+1} \\ u_{j+2} \\ u_{j+3} \\ u_{j+4} \end{pmatrix} \quad \text{for} \quad j = 0, ..., n.$$

With this numbering, the total matrix $[T]$ and the sub-matrices $[T_{ij}]$ are band matrices.

Comments:

- Choosing a Lagrange interpolation for displacements implies that displacements are continuous, but the velocities are discontinuous. As a consequence, integration by parts as in (3) is not totally rigorous, and it could be necessary to use the discontinuous Galerkin formulation, which amounts to writing the derivative of velocity within the theory of distributions. We will preserve the formulation in (3), knowing that the error here is of the same order as in the case of traditional finite elements in space. Indeed, with a Lagrange interpolation local displacements are continuous, whereas the global deformation is discontinuous.
- Even if it is not absolutely necessary, the advantage of using a laminated mesh such as defined here is that it becomes possible, rather than assembling the total matrix $[T]$, to only assemble the sub-matrices $[T_{ij}]$. This considerably reduces the size of the systems to be solved. More precisely, the size of these linear systems is exactly the same as that obtained in the case of approaches based on the coupling of finite incremental differences in time with finite elements in space. Moreover, the method is not limited to simplex elements, and the spatial position of each set of nodes can vary from one time plane to the other. It is one of the main advantages of the method.

- We specify that the nodal vector relating the boundary conditions with velocity $\{\Lambda\}$ is written as:

$$\{\Lambda\} = (\{\Lambda_0\}, 0, ..., 0, \{\Lambda_n\})^T$$

where $\{\Lambda_0\}$ is given starting from conditions of initial velocity while $\{\Lambda_n\}$ is unknown. Consequently, the resolution of System (16) is the following:

The first system of equations,

$$[T_{11}]\{U_0\} + [T_{12}]\{U_1\} = \{F_0\} + \{\Lambda_0\} \tag{17}$$

provides $\{U_1\}$; the system of equations:

$$[T_{i/i-1}]\{U_{i-2}\} + [T_{i/i}]\{U_{i-1}\} + [T_{i/i+1}]\{U_i\} = \{F_{i-1}\} \quad 2 \leq i \leq n\,, \tag{18}$$

provides the displacements $\{U_i\}$; and the last system of equations,

$$[T_{n+1/n}]\{U_{n-1}\} + [T_{n+1/n+1}]\{U_n\} = \{F_n\} + \{\Lambda_n\}\,, \tag{19}$$

gives $\{\Lambda_n\}$.

- Finally, the matrices of resolution $[T_{i/i+1}]$ are generally non-symmetric, even if the total matrix $[T]$ is symmetric. Thus, for the algorithm presented above, a non-symmetrical solver should be used. This can appear penalizing in terms of computing time. However, since the final objective is to use this approach to deal with problems of contact with friction and since the nonlinear resolution we developed in [14] is of the Gauss–Seidel nonlinear type, asymmetries do not affect computing time.

3. 4D Mesh and Remeshing

In order to propose a remeshing technique, it is firstly necessary to be able to build 4D meshes. Obtaining only one 4D finite element does not pose real problems, even if some difficulties in graphic representation arise (see Figure 3). On the contrary, building a 4D mesh, even the most elementary is far from being commonplace, except in the case of regular meshes formed by finite elements of multiplexing type (functions of interpolation obtained as the products of functions of space by functions of time). However, in the general case and in particular with the problem of remeshing, which is what is of interest here, the meshing remains an issue.

3.1. 4D Mesh Generation

Figure 3 identifies parts of elementary 2D, 3D and 4D meshes with their node numbering. One denotes by n_0 the total number of nodes at time $t = 0$ of the entire space mesh, and we assume this total number is the same at time $t = h$.

For the 2D mesh, the connectivities are
$i, j, n_0 + i$
$j, n_0 + i, n_0 + j$.

For the 3D mesh, the connectivities are
$i, j, k, n_0 + i$
$j, k, n_0 + i, n_0 + j$
$k, n_0 + i, n_0 + j, n_0 + k$.

Using the previous building of connectivities, obtained by circular permutations, we propose the following generalization of connectivities for the 4D mesh.
$i, j, k, l, n_0 + i$
$j, k, l, n_0 + i, n_0 + j$

$k, l, n_0 + i, n_0 + j, n_0 + k$
$l, n_0 + i, n_0 + j, n_0 + k, n_0 + l.$

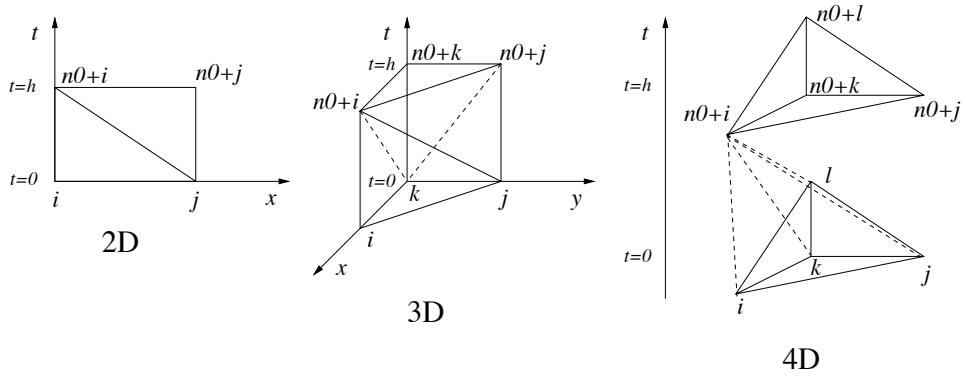

Figure 3. 2D, 3D and 4D space-time mesh.

This 4D mesh is constituted by four hypertetrahedrons (The hypertetrahedron is the four-dimensional tetrahedron. Other names of this element type are simplex or pentatope). We propose to build 4D space-time meshes, resulting from unspecified 3D space meshes, by applying the building technique developed above, for each 3D finite element of the 3D space mesh. However, in this case, it must be checked that the total space-time volume is covered by the 4D mesh. For this purpose, we computed the sum of the volumes of the hypertetrahedrons of the 4D mesh and compared it with the total volume generated by the 3D object multiplied by the time interval.

For a 3D mesh made up of tetrahedrons, the interfaces between the elements are triangles. In 4D, these interfaces are tetrahedrons (see the diagram at the bottom of Figure 4). Therefore, we must thus check that for our technique of mesh generation, all couples of adjoining 4D finite elements have a common tetrahedron. As we use a building technique containing circular permutations, it is necessary to respect a particular order in the numbering of the nodes of each 3D finite element. A way of doing this is to arrange the nodes of each 3D element in ascending order. Table 1 gives an example of a table of connectivities for an elementary 4D mesh, resulting from the 3D space mesh represented by the left-hand diagram of Figure 4. Let us note that this 4D mesh contains eight finite elements, against two for the 3D mesh source, and that $n_0 = 5$. It is observed that the connectivities are arranged in ascending order. In this case, it is checked that the tetrahedra filling the space-time interface (diagram at the right-hand of Figure 4) are common to the adjoining elements. Indeed, Elements 1 and 5 contain the tetrahedron (1; 2; 4; 6), and Elements 2 and 6 contain the tetrahedron (2; 4; 6; 7). Lastly, Elements 4 and 7 contain the tetrahedron (4; 6; 7; 9). Let us notice that the tetrahedron (1; 2; 3; 4; 5) at time t and the related tetrahedron (6; 7; 8; 9; 10) (not represented) at time $t + \Delta t$ could also have different shapes and could be localized at different places (see for example Figure 5). In this case, the triangles (1; 2; 4) and (6; 7; 9) in the right part of Figure 4 can be different.

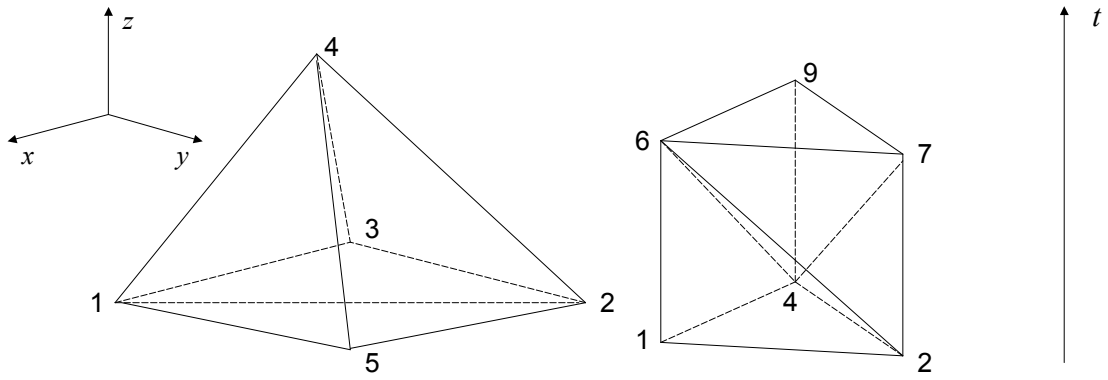

Figure 4. 3D initial mesh (scheme at the left); part of the 4D mesh generated by the triangle (1; 2; 4) common to the two finite elements of the 3D initial mesh (scheme at the right).

Table 1. Table of connectivities of the 4D space-time mesh resulting from the elementary 3D space mesh of Figure 4.

Element Number	Node 1	Node 2	Node 3	Node 4	Node 5
1	1	2	3	4	6
2	2	3	4	6	7
3	3	4	6	7	8
4	4	6	7	8	9
5	1	2	4	5	6
6	2	4	5	6	7
7	4	5	6	7	9
8	5	6	7	9	10

3.2. Remeshing Technique

In this section, we present our technique of space-time mesh adaptations. In the literature, many articles on mesh adaptations [2,3,15–21] can be found. Among these papers, a large number deals with space-time mesh adaptations. They use the discontinuous Galerkin method. In most of them, the approach is incremental, i.e., remeshing is carried out at given steps of time. Generally, the values of the unknown of the new mesh are obtained by approximation or interpolation from those of the old mesh, which we will call "mesh-to-mesh transfer". Moreover, the interpolation used is of a multiplexing type; the function of interpolation is defined by the product of a function of space by a function of time.

In [22], we proposed an incremental technique for mesh adaptation, which does not require mesh-to-mesh transfer. It was coupled with problems of rubbing contact (see [14]). In addition, we developed a non-incremental technique of mesh adaptation, based on non-structured space-time meshes.

Some teams have already worked on this problem. One can quote the works of Hugues and Hulbert [2,3], Tezduyar et al. [20,21] and Idesman et al. [9,23]. They use the continuous or discontinuous Galerkin method. In these approaches, calculations are carried out on the whole space-time domain $\Omega \times [0, T]$. Thus, for a field Ω of dimension d and a total number N of nodes of the space-time mesh, the dimension of the linear problem to be solved is $d \times N$, which quickly becomes large when $d = 2$ or $d = 3$. A solution to decrease the computational time is to use parallel computations. This is the option chosen in [8,9].

In the context of the continuous Galerkin method, we suggested, in [22,24], a non-incremental solution, which substitutes the concept of a step of time by that of a "space-time front". Erickson et al. [16] have also proposed an advancing-front mesh generation, in the context of the discontinuous Galerkin method. This technique was successfully used by Miller et al. [25] in their multi-field space-time discontinuous Galerkin method, for $d = 1$ and 2 in linearized elastodynamics

applications. The advantage of this frontal resolution is that it decreases the size of the linear systems to be solved. Due to technical difficulties, the frontal resolution in the case of $d = 3$ has not yet been implemented. Nevertheless, we propose a particular incremental remeshing technique based on the construction of 4D space-time meshes that are able to follow an evolutionary loaded zone. This technique of mesh generation uses simplex finite elements. Figure 5 gives an illustration of the technique. The principle is to maintain the same number of nodes during the simulation, but to locate a sufficiently large number of them under the loaded area. In this particular case, it is possible to preserve the matrices $T_{i/i-1}$, $T_{i/i}$, $T_{i/i+1}$ identical for all i, which involves a reduction of the computational time. An example of mechanical application is provided in the following section. This technique is aimed at applications in simulating problems of wear between two bodies in contact.

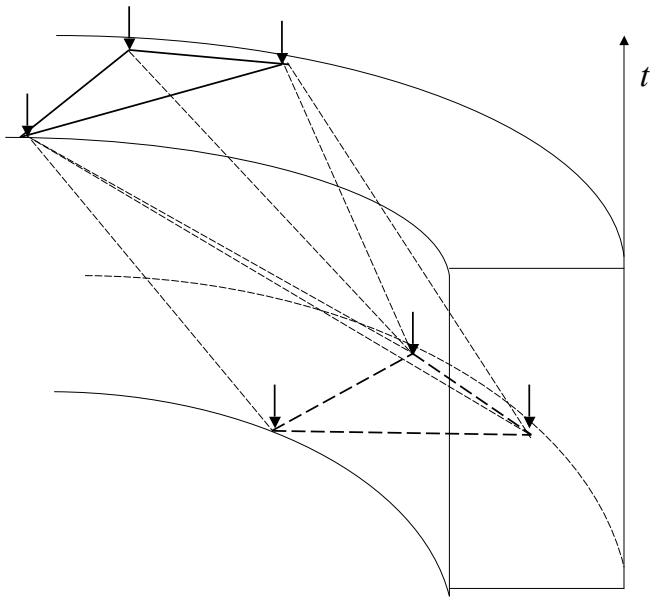

Figure 5. Space-time mesh generation by rotation of the loaded area. The loaded area is represented by arrows.

4. Numerical Analysis

Our space-time finite elements method was programmed using MATLAB software and was validated on elementary examples.

4.1. Stability

Preliminary results on the stability of the method have been established in [22] and compared with the Newmark integration scheme. Here, we summarize the main results. Let δ and θ be two real parameters; the Newmark integration scheme reads:

$$\begin{cases} \{\dot{U}_{i+1}\} = \{\dot{U}_i\} + \Delta t \left[(1-\delta)\{\ddot{U}_i\} + \delta\{\ddot{U}_{i+1}\}\right] \\ \{U_{i+1} = \{U_i\} + \Delta t\{\dot{U}_i\} + \Delta t^2 \left[\left(\frac{1}{2} - \theta\right)\{\ddot{U}_i\} + \theta\{\ddot{U}_{i+1}\}\right] \end{cases}$$

where Δt is the time step of integration, $\{\dot{U}_{i+1}\}$ and $\{\ddot{U}_{i+1}\}$ are respectively the assembled vector of nodal velocities and accelerations at time $(i+1)\Delta t$. We showed in [22] that:

- For 1D space-time elastodynamic applications, the use of the STFEM method with linear simplex elements is similar to the use of the implicit Newmark integration scheme with $\delta = 1/2$ and $\theta = 1/3$. The method is then unconditionally stable.
- For 2D space-time elastodynamic applications, the use of the STFEM method with linear simplex elements is similar to the use of the explicit Newmark integration scheme with $\delta = 1/2$ and $\theta = 0$.

The method is then conditionally stable. Classically, the time step has to verify the CFLcondition: $\Delta t \leq \min_J \frac{2}{\omega_J}$, where each ω_J is the frequency of a normal mode of vibration.

- For higher dimensions (3D and 4D), no direct relationship between the STFEM and the Newmark method has been established. Nevertheless, we noted that our method required sufficiently small space-time slabs, of the same order of the discretization time step necessary with explicit methods of integration.
- Furthermore, the use of the STFEM method with multiplex elements is similar to the use of the implicit Newmark integration scheme with $\delta = 1/2$ and $\theta = 1/3$, for 1D, 2D, 3D and 4D space-time applications. In this case, the method is unconditionally stable.

In the present study, a specific numerical investigation has been carried out to estimate the stability conditions for the STFEM method with linear simplex elements for 4D space-time elastodynamic applications. The stability was tested on a beam of length $L = 0.1$ m and a square section of 0.01×0.01 m^2 (see Figure 6). The Young modulus E was equal to 1000 Pa; the Poisson's ratio ν was equal to 0.3; and the density ρ was equal to 680 kg/m^3.

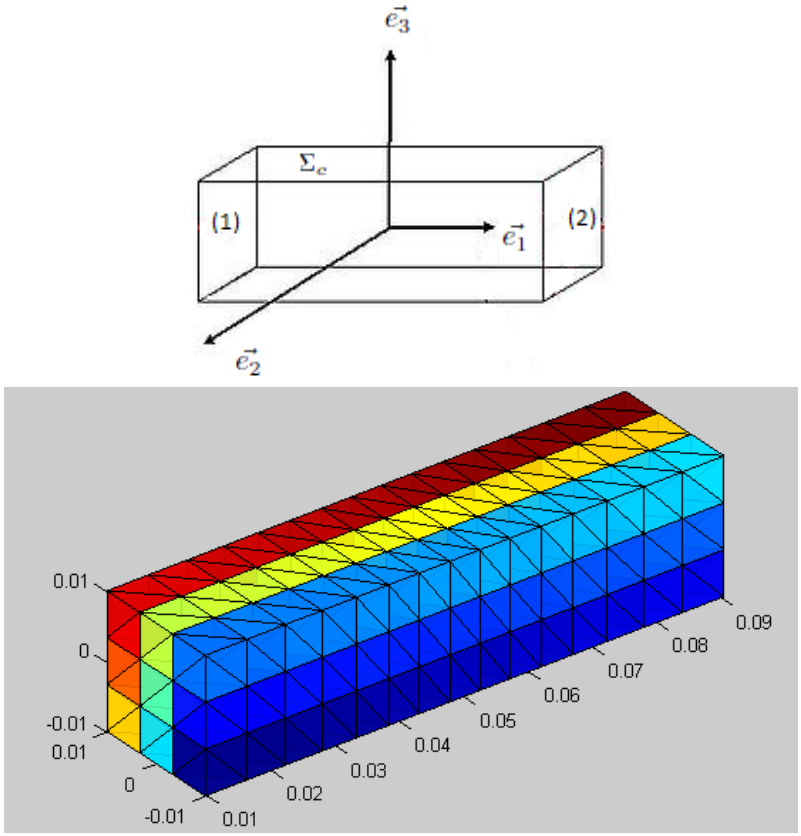

Figure 6. Geometry and 3D mesh of a beam with $h = 6.6 \times 10^{-3}$ m.

We built boundary conditions in order to obtain the following analytic solution:

$$u(x,t) = \cos(\frac{\pi x_1}{L}) \cos(\frac{\pi ct}{L}) e_1 \tag{20}$$

where c is the velocity of the wave propagation. On Faces (1) and (2), null Neumann conditions were imposed. On the other four faces (Σ_e), Dirichlet conditions were imposed in order to satisfy the analytic

solution (20). The volume external force density f_d was assumed to be vanishing. This imposes that the wave velocity c must verify:

$$c = \sqrt{\frac{E}{\rho(1+\nu)}(1+\frac{\nu}{\rho(1-2\nu)})}$$

Results plotted in Figure 7 give the dependence of the time discretization (size of the space-time slab) Δt on the average size h of the 3D finite elements, to ensure the stability of the method.

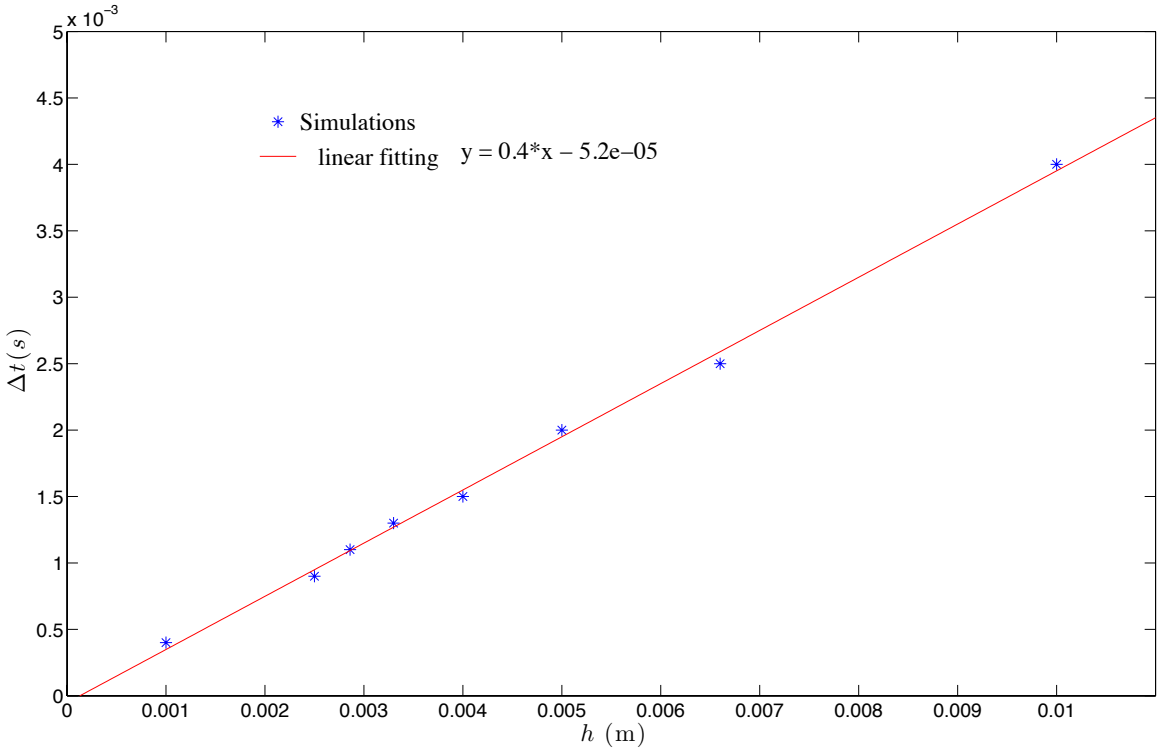

Figure 7. Size of the time step of discretization Δt necessary for stability with respect to the average size h of the 3D finite elements.

Linear fitting suggests that the stability criterion is:

$$\Delta t \leq \alpha h$$

with $\alpha \simeq 1/2c$. Indeed, in our example, $c = 1.407$ m/s.

4.2. Convergence

In the case of simplex finite elements, the convergence with the STFEM method is comparable to the convergence with the Newmark scheme, for the 1D and 2D problem. Concerning the convergence analysis of the STFEM in 4D, we used the previous example. The time step of discretization Δt is scaled with respect to h, using the stability criterion obtained in the previous paragraph. We computed the maximum error over space at the last time step between the analytic solution and the numerical solution for each mesh size h. The results are plotted in Figure 8.

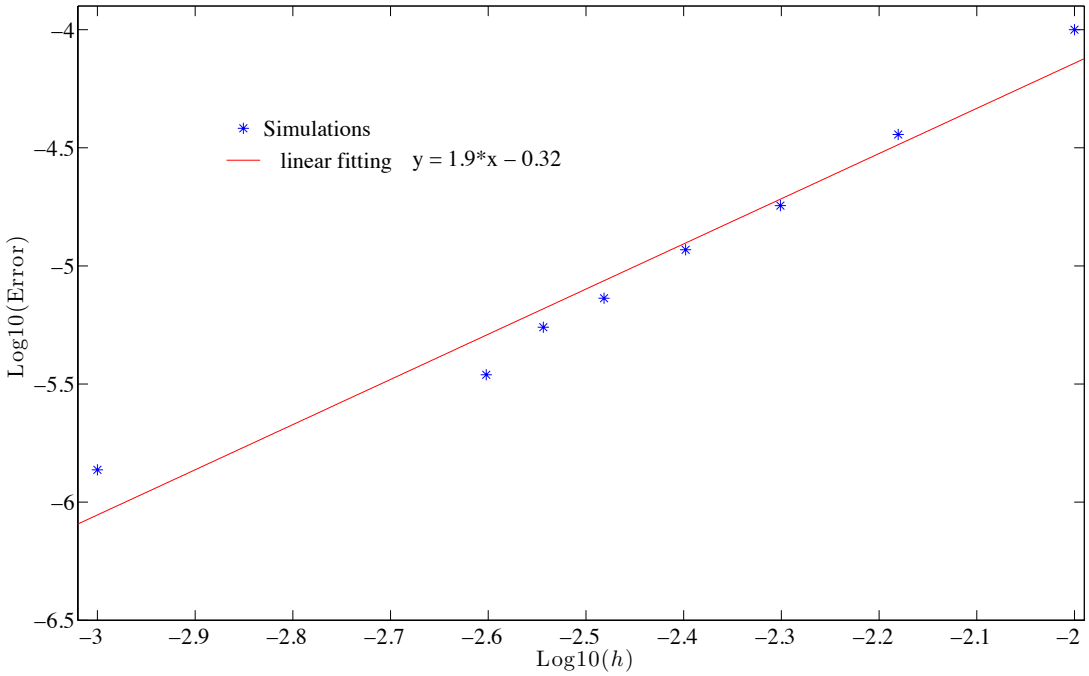

Figure 8. Maximum error over the space, at the last time step, between the analytic solution and the numerical solution for each mesh size h, in the logarithmic scale.

We can note that the convergence rate for the method is nearly quadratic.

5. Numerical Results on Mesh Adaptation

To illustrate our technique of mesh adaptation, we consider the example of a brake disc subjected to the clamping of a plate on one of its faces (see the source model with the coarse mesh in Figure 9) and blocked on the opposite face. The disc is made of steel with a Young's modulus equal to 210,000 MPa, a Poisson's ratio equal to 0.3 and a density equal to 7800 kg/m^3. The internal radius of the disc is equal to 40 mm; its external radius is equal to 100 mm; and its thickness is 10 mm. Two 3D initial meshes have been tested: a coarse mesh, which contains 634 nodes and 1752 elements, refined only in the area of the clamping, and a fine mesh, which contains 6476 nodes and 25,856 elements and has a uniform mesh fineness over all the sample.

The clamping area is modeled by a constant pressure of 100 MPa. This area is moved along the circumference of the disc with a rotational speed equal to that of the propagating wave $V = \frac{1}{2\pi R}\sqrt{\frac{E}{\rho}}$, where $R = 70$ mm is the average radius of the disc. For both 3D initial meshes, we built an incremental 4D space-time mesh, which preserves the 3D mesh at each time step, by imposing an axial rotation to keep the finest zone of the 3D mesh under the loading area. An illustration is shown schematically in Figure 5.

The results of the calculations presented were obtained for space-time slabs of 10^{-7} s (this is equivalent to using a time step equal to 10^{-7} s). The vertical displacements obtained with fine and coarse meshes have been compared for points located on a circle of control, of radius R equal to 70 mm (see Figure 10).

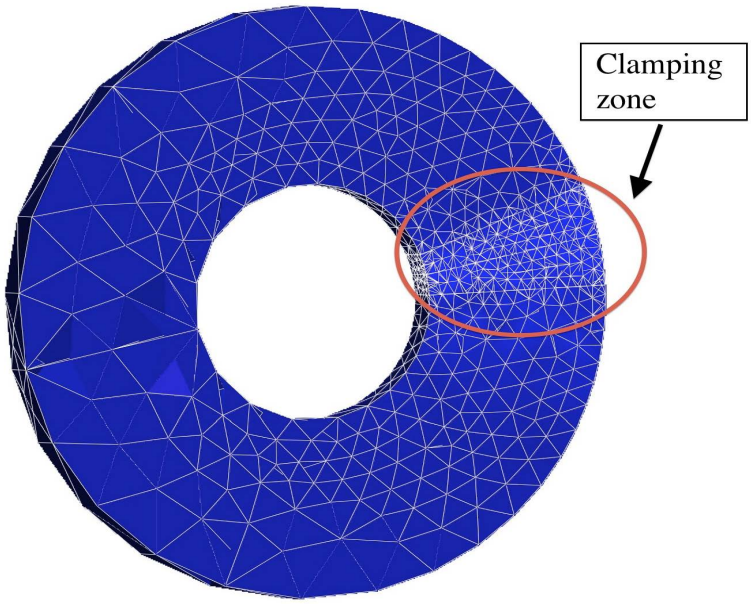

Figure 9. 3D initial coarse mesh: the mesh is finer under the clamping zone.

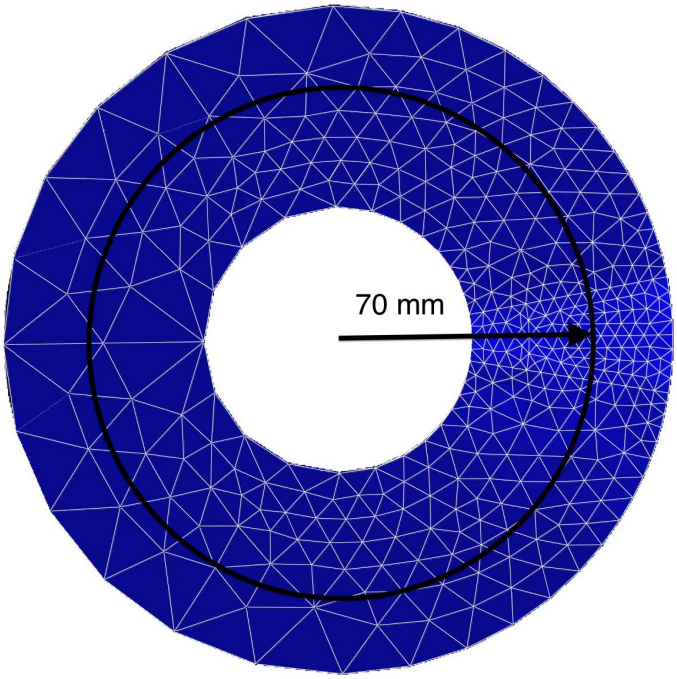

Figure 10. Location zone of checked displacements: circle of radius equal to 70 mm.

Numerical values are gathered in Figure 11 for the results at time $t = 2.10^{-5}$ s and in Figure 12 for the results at time $t = 4.10^{-5}$ s. Each check point is defined by its angle, in polar coordinates.

Numerical comparisons show that the coarse and fine meshes give similar results in the clamping zone. However, apart from this zone, the results are somewhat different. Let us note that nodal displacements cannot be identical because dynamic effects depend on the fineness of the mesh.

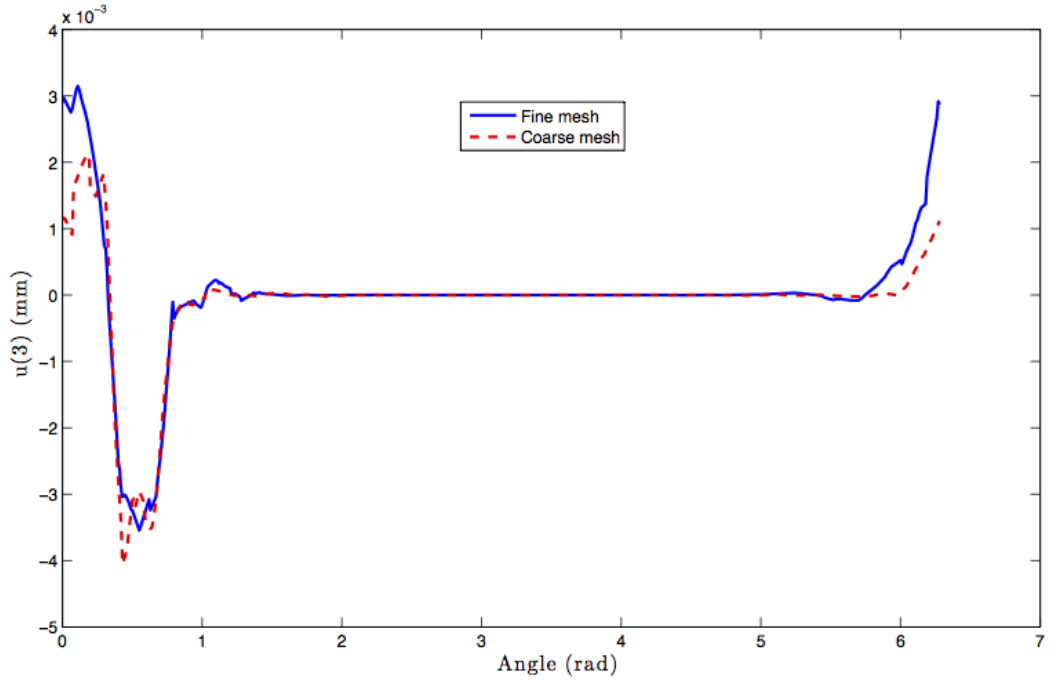

Figure 11. Comparison of vertical displacements for points situated on the circle of control, expressed in mm, at time $t = 2.10^{-5}$ s. Each check point is defined by its angular coordinate, expressed in radians.

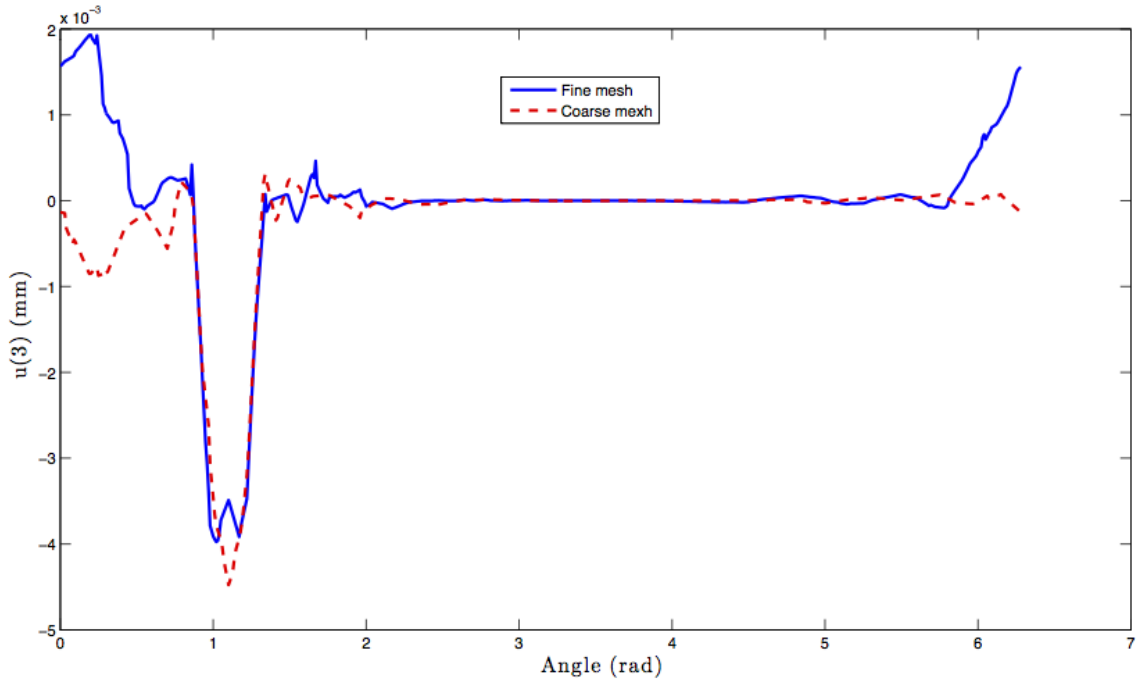

Figure 12. Comparison of vertical displacements for points situated on the circle of control, expressed in mm, at time $t = 4.10^{-5}$ s. Each check point is defined by its angular coordinate, expressed in radians.

Figures 13 and 14 show the norm of incremental displacements at time $t = 2.10^{-5}$ s and $t = 4.10^{-5}$ s for the coarse and the fine mesh, respectively. For the coarse mesh, we can observe that the refined zone really remains under the zone of clamping. The distribution of the norm of node displacements is similar for the two positions of the load. It is important to note that they are incremental displacements and not total displacements.

Finally, it must be noted that the computational time is 6.6-times faster using the coarse mesh.

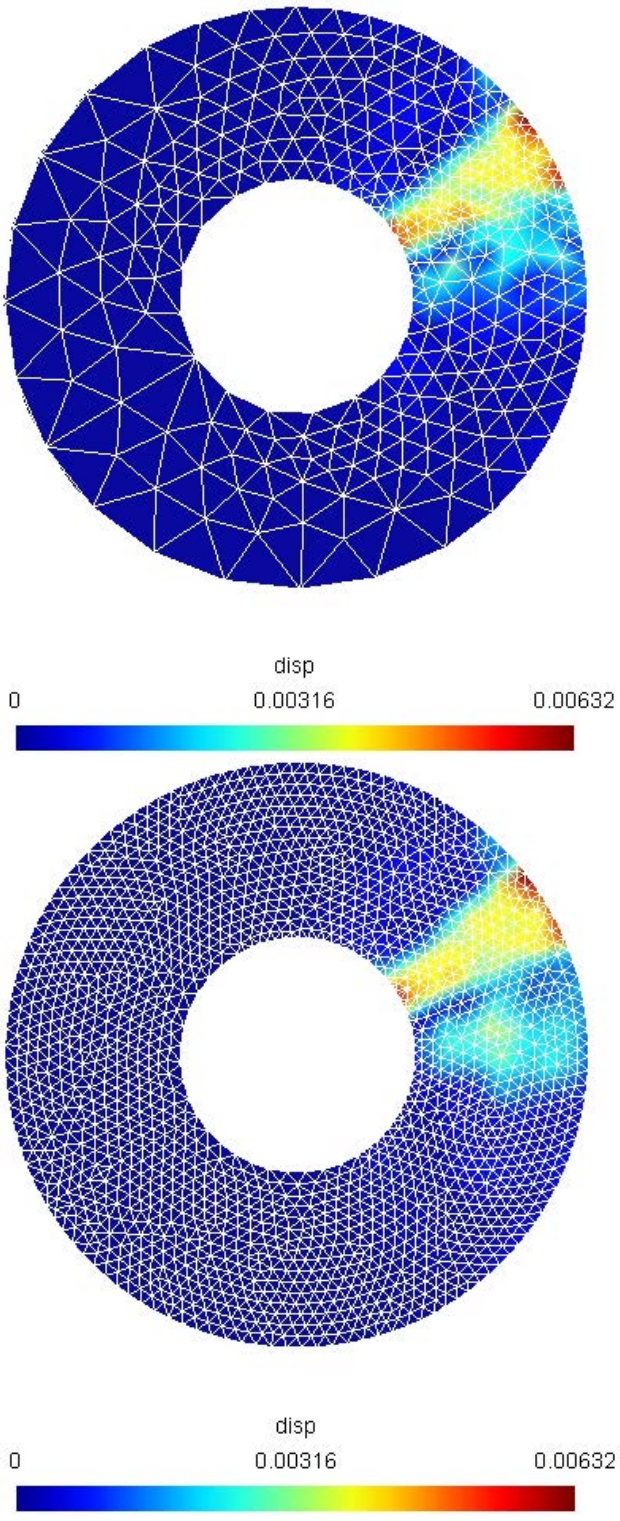

Figure 13. Isovalues of the norm of nodal displacements, expressed in mm, at $t = 2.10^{-5}$ s for the coarse mesh (image at the top) and for the fine mesh (image at the bottom).

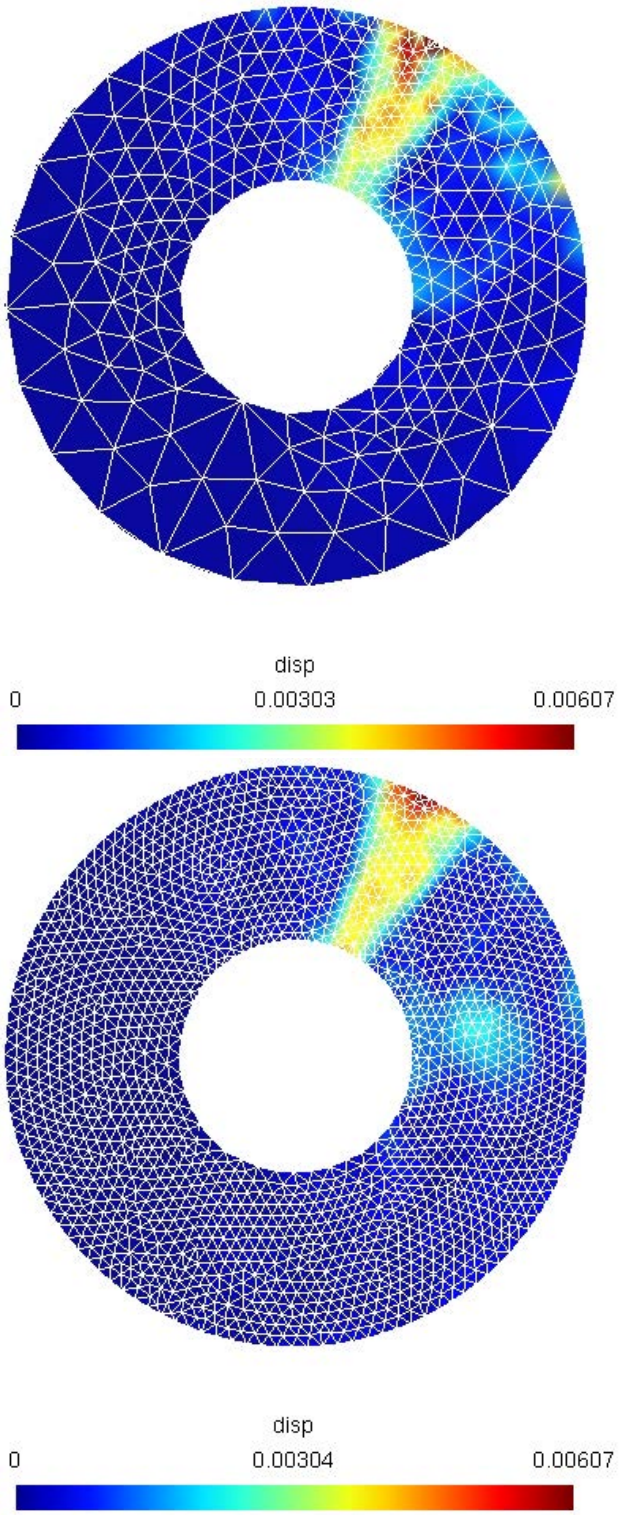

Figure 14. Isovalues of the norm of nodal displacements, expressed in mm, at $t = 4.10^{-5}$ s for the coarse mesh (image on the top) and for the fine mesh (image on the bottom).

6. Conclusions

The method we have presented for space-time mesh generation for 4D domains, using simplex elements, made it possible to develop a technique of mesh adaptation able to follow a mobile loading zone. This original technique has been carried out to ensure a minimal computational time and does

not require coarse-to-fine and fine-to-coarse mesh transfer operators. The convergence and stability of the method were studied and compared with existing methods. This approach opens the way to 4D remeshing. It allows, thanks to simplex elements, remeshing in both space and in time.

However, to demonstrate the all the capacities of this method, it is necessary to go much further in the mechanical applications and to propose a technique of 4D mesh adaptation using the frontal approach, as presented in [14,16,22,25].

Author Contributions: Conceptualization, F. Jourdan; Software, S. Dumont, F. Jourdan; Validation, T. Madani; Original Draft Preparation, S. Dumont, F. Jourdan and T. Madani.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Ladevèze, P. *Non Linear Computational Structural Mechanics, New Approaches and Non Incremental Methods of Calculation*; Springer: New York, NY, USA, 1999.
2. Hughes, T.J.R.; Hulbert, G.M. Space-time finite element methods for elastodynamics: Formulations and error estimates. *Comput. Methods Appl. Mech. Eng.* **1988**, *66*, 339–363. [CrossRef]
3. Hulbert, G.M.; Hughes, T.J.R. Space-time finite element methods for second-order hyperbolic equations. *Comput. Methods Appl. Mech. Eng.* **1990**, *84*, 327–348. [CrossRef]
4. Khalmanova, D.K.; Costanzo, F. A space-time discontinuous Galerkin finite element method for fully coupled linear thermo-elasto-dynamic problems with strain and heat flux discontinuities. *Comput. Methods Appl. Mech. Eng.* **2008**, *197*, 1323–1342. [CrossRef]
5. Argyris, J.H.; Scharpf, D.W. Finite elements in time and space. *Nucl. Eng. Des.* **1969**, *10*, 456–464. [CrossRef]
6. Zienkiewics, O.C.; Parekh, C.J. Transient field problems-two and three dimensional analysis by isoparametric finite elements. *Int. J. Numer. Methods Eng.* **1970**, *2*, 61–71. [CrossRef]
7. Chien, C.C.; Yang, C.S.; Tang, J.H. Three-dimensional transient elastodynamic analysis by a space and time-discontinuous Galerkin finite element method. *Finite Elem. Anal. Des.* **2003**, *39*, 561–580. [CrossRef]
8. Anderson, M.; Kimn, J.-H. A numerical approach to space-time finite elements for the wave equation. *J. Comput. Phys.* **2007**, *226*, 466–476. [CrossRef]
9. Idesman, A.V. Solution of linear elastodynamics problems with space-time finite elements on structured and unstructured meshes. *Comput. Methods Appl. Mech. Eng.* **2007**, *196*, 1787–1815. [CrossRef]
10. Huang, H.; Costanzo, F. On the use of space-time finite elements in the solution of elasto-dynamic problems with strain discontinuities. *Comput. Methods Appl. Mech. Eng.* **2002**, *191*, 5315–5343. [CrossRef]
11. Karaoglan, L.; Noor, A.K. Space-time finite element methods for sensitivity analysis of contact/impact response of axisymmetric composite structures. *Comput. Methods Appl. Mech. Eng.* **1997**, *144*, 371–389. [CrossRef]
12. Jourdan, F.; Bussy, P. Large time increment method in dynamic regularization: Sheet cutting simulations. *Comput. Methods Appl. Mech. Eng.* **2000**, *190*, 1245–1259. [CrossRef]
13. Li, X.D.; Wiberg, N.-E. Implementation and adaptivity of space-time finite element method for structural dynamics. *Comput. Methods Appl. Mech. Eng.* **1998**, *156*, 211–229. [CrossRef]
14. Adélaïde, L.; Jourdan, F.; Bohatier, C. Frictional contact solver and mesh adaptation in Space-Time Finite Element Method. *Eur. J. Mech. A/Solids* **2003**, *22*, 633–647. [CrossRef]
15. Donea, R.; Huerta, A.; Sarrate, J. Arbitrary Lagrangian-Eulerian for fluid-rigid body interaction. *Comput. Methods Appl. Mech. Eng.* **2001**, *190*, 3171–3188.
16. Erickson, J.; Guoy, D.; Sullivan, J.M.; Üngör, A. Building spacetime meshes over arbitrary spatial domains. In Proceedings of the 11th International Meshing RoundTable, Sandia, Ithaca, NY, USA, 15–18 September 2002; pp. 391–402.
17. Farhat, C.; Degand, C. A three-dimensional torsional spring analogy method for unstructured dynamic meshes. *Comput. Struct.* **2002**, *80*, 305–316.
18. Feng, Y.T.; Peric, D. A time-adaptive space-time finite element method for incompressible Lagrangian flows with free surfaces: Computational issues. *Comput. Methods Appl. Mech. Eng.* **2000**, *190*, 499–518. [CrossRef]
19. Mohr, R.; Menzel, A.; Steinmann, P. A consistent time FE-method for large strain elasto-plasto-dynamics. *Comput. Methods Appl. Mech. Eng.* **2008**, *197*, 3024–3044. [CrossRef]

20. Tezduyar, T.E.; Sunil, S. Enhanced-discretization space-time technique (EDSTT). *Comput. Methods Appl. Mech. Eng.* **2004**, *193*, 1385–1401. [CrossRef]
21. Tezduyar, T.E.; Sathe, S.; Keedy, R.; Stein, K. Space-time finite element techniques for computation of fluid-structure interactions. *Comput. Methods Appl. Mech. Eng.* **2006**, *195*, 2002–2027. [CrossRef]
22. Adélaïde, L.; Jourdan, F.; Bohatier, C. Méthode des éléments finis espace-temps et remaillage. *Rev. Eur. Elem. Finis* **2003**, *12*, 427–459. [CrossRef]
23. Idesman, A.; Niekamp, R.; Stein, E. Continuous and discontinuous Galerkin methods with finite elements in space and time for parallel computing of viscoelastic deformation. *Comput. Methods Appl. Mech. Eng.* **2000**, *190*, 1049–1063. [CrossRef]
24. Adélaïde, L.; Jourdan, F.; Bohatier, C. New results on mesh adaptation in space-time finite element method. In Proceedings of the ASME Engineering Technology Conference on Energy (ETCE 2002/STRUC-29042), Houston, TX, USA, 4–5 February 2002.
25. Miller, S.T.; Kraczek, B.; Haber, R.B.; Johnson, D.D. Multi-field spacetime discontinuous Galerkin methods for linearized elastodynamics. *Comput. Methods Appl. Mech. Eng.* **2009**, *199*, 34–47. [CrossRef]