



A Space-Time Finite Element Method for Elastodynamics Problems with a Moving Loading Zone

S. Dumont¹ and F. Jourdan²

¹LAMFA, CNRS UMR 6140, Université de Picardie Jules Verne, France

²LMGC, CNRS UMR 5508, Université Montpellier 2, France

Abstract

We present a space-time finite element method (STFEM) for linearised elastodynamics in three dimensions in space and one in time. For that purpose, we have developed a technique of four-dimensional (4D) mesh generation adapted to space-time remeshing. This original technique does not require coarse-to-fine and fine-to-coarse mesh transfer operators and does not increase the size of linear systems to be solved, compared to the traditional finite element method. Space-time meshes are composed of simplex finite elements. Computations are realized in the context of the continuous Galerkin method. Our technique of mesh adaptation is applied to a problem of mobile loading, where the evolutionary mesh is able to follow the mobile loading zone.

Keywords: finite elements, space-time, four-dimensional mesh generation, 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. The first studies date from the late sixties [6, 16] and currently, several approaches exist. One can quote for example the Large Time INcrement method (LATIN [15]), the Discontinuous Galerkin method [1, 9, 10, 14], and the present method which is a Continuous Galerkin method. In most publications on the Discontinuous Galerkin method, like in [7], the interpolation functions are assumed to be the product of functions of space variables and functions of time variables. We will see in this paper that a special attention will be paid with the non separation of the space and time variables, because the final objective of this study (not presented in this paper) is the remeshing, and this type of interpolation is well-suited to mesh adap-

tation. The space-time mesh generation we develop here permit to construct meshes 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 of what can be done 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.

Let us note that one of the drawbacks of the STFEM such as it is defined in the works of [5] and [11] is the size of the linear systems to be solved, because the whole space-time problem has to be solved simultaneously. The use of a laminated mesh allows us to assemble not the total matrix of the problem, but only the 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 “classic” finite element method to solve the space problem.

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. Numerical results are presented and discussed in section 4.

2 Principle of the method

Let us consider the motion of an elastic body within the small perturbations theory. Let Ω be the set taken up by the body and $[0, T]$ a time interval. The body is under 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 : Find the displacement u and the Cauchy stress tensor σ such that

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

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 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 $[0, T]$. Like in the Discontinuous Galerkin method [8, 9, 13] and in the LARge Time INcrement (LATIN) method [12, 15], 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 follow:

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

$$\int_0^T \int_{\Omega} (\rho \ddot{u}v + a\varepsilon(u) : \varepsilon(v) - f_d v) dx dt = \int_0^T \int_{\partial_1 \Omega} F_d v ds dt, \quad \forall v \in U_{ad}^0 \quad (2)$$

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 in the left hand side of the equation (2) is integrated by part in time in order to bring up the first derivative of u and to let appear the initial velocity. It comes

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

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

In our space-time interpolation, classical Lagrange polynomial are used. The finite elements are isoparametric. On a space-time finite element E_e (figure 1) the displacement verifies

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

where n_e is the total number of nodes for the element E_e , φ_i^e are the interpolation functions and u_i^e the nodal displacements.

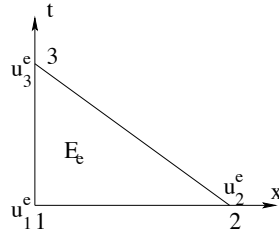


Figure 1: 2D space-time finite element

Using a matrix notation, one has

$$u(x, t) = N_e(x, t) U_e \quad \text{where} \quad U_e = (u_1^e, \dots, u_{n_e}^e)^T \quad (5)$$

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

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

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

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

where

$$M_e = \int_0^T \int_{E_e} \rho \frac{\partial N_e^T}{\partial t} \frac{\partial N_e}{\partial t} dx dt \quad (8)$$

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

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

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

$$\Lambda_e = \left[\int_{E_e \cap \Omega_T} \rho N_e^T \dot{u}_T dx - \int_{E_e \cap \Omega_0} \rho \dot{u}_0 dx \right] \quad (10)$$

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

$$\varepsilon(u(x, t)) = B_e(x, t)U_e, \quad (11)$$

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

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

and

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

where the elementary matrix K_e relative to internal forces is

$$K_e = \int_0^T \int_{E_e} B_e^T a B_e dx dt \quad (14)$$

and the elementary vector F_e relative to external forces is

$$F_e = \int_0^T \int_{E_e} N_e^T f_d dx dt + \int_0^T \int_{E_e \cap \partial_1 \Omega} N_e^T F_d ds dt. \quad (15)$$

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

$$([\widetilde{M}_u] + [\widetilde{K}_u])\{U\} = \{F_u\} + \{\Lambda\} \quad (16)$$

where $[\widetilde{M}_u]$ is the assembled matrix relative to the inertia forces, $[\widetilde{K}_u]$ is the assembled matrix relative to 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 notice that the matrices $[\widetilde{M}_u]$ and $[\widetilde{K}_u]$ are symmetric, and because we have in mind

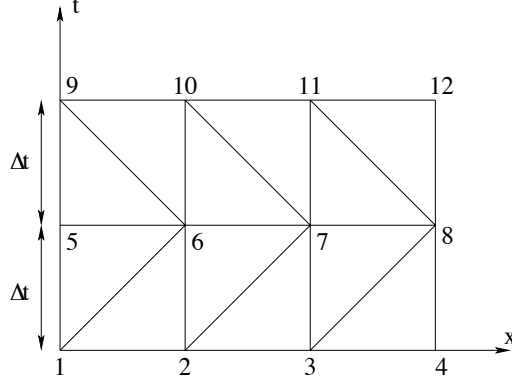


Figure 2: 2D regular space-time mesh

to make computations incrementally in time, the meshes are built to be stratified in time. In order to use efficiently this property, the node numbering is made such a way that all nodes in a same stratum have close numbers, for example like in the simple example presented in figure 2.

In this example, the unknown $\{U\}$ can be reorganized, defining

$$\{U_0\} = \begin{pmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{pmatrix}, \quad \{U_1\} = \begin{pmatrix} u_5 \\ u_6 \\ u_7 \\ u_8 \end{pmatrix} \quad \text{and} \quad \{U_2\} = \begin{pmatrix} u_9 \\ u_{10} \\ u_{11} \\ u_{12} \end{pmatrix}.$$

Using this writing, the left hand side of the system (16) can be organized as follow:

$$([\widetilde{M}_u] + [\widetilde{K}_u])\{U\} = [T]\{U\} =$$

$$\begin{pmatrix} [T_{0,0}] & [T_{0,1}] & 0 & 0 & 0 & 0 \\ [T_{1,0}] & [T_{1,1}] & [T_{1,2}] & 0 & 0 & 0 \\ 0 & [T_{2,1}] & [T_{2,2}] & [T_{2,3}] & 0 & 0 \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ 0 & 0 & 0 & [T_{n-1,n}] & [T_{n-1,n-1}] & [T_{n-1,n}] \\ 0 & 0 & 0 & 0 & [T_{n,n-1}] & [T_{n,n}] \end{pmatrix} \begin{pmatrix} \{U_0\} \\ \{U_1\} \\ \cdot \\ \cdot \\ \cdot \\ \{U_n\} \end{pmatrix}.$$

One can notice that with this numbering, the total matrix $[T]$ and the sub-matrices $[T_{ij}]$ are band, and this technique can be generalized in case of unstructured meshes in space, using the notion of front (see for example [2]).

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 precise enough and it should be required to use

the Discontinuous Galerkin method, which amounts to writing the derivative of velocity within the theory of distributions. In our study, we will use a Continuous Galerkin method and 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, one can notice that the same approximation is classically used in space since with a Lagrange interpolation, local displacements are continuous, whereas the global deformation is discontinuous. Others similar studies on STEM has been developed on unstructured meshes [1, 9]. However, they employ the Discontinuous Galerkin method in time and incorporate stabilizing terms of least squares type. The space and time discontinuities of all variables are taken into account. The question of stabilization of inertia effects is out of purpose of the present paper.

- The advantage of using a laminated mesh such as defined here is that it becomes possible to assemble only the sub-matrices $[T_{ij}]$ rather than assembling the total matrix $[T]$. This considerably reduces the size of systems to be solved. In fact, the size of these linear systems is exactly the same as that obtained in the case of approaches based on coupling finite incremental differences in time with finite elements in space.

More precisely, if we suppose in a sake of simplicity that the nodal vector relating to the boundary conditions with velocity $\{\Lambda\}$ is written as

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

where $\{\Lambda_0\}$ is given starting from conditions of initial velocity while $\{\Lambda_n\}$ is unknown (in the general case, the final velocity is not necessarily only in the last stratum), then the system (16) can be solved as follow:

- The first system of equations,

$$[T_{0,0}]\{U_0\} + [T_{0,1}]\{U_1\} = \{F_0\} + \{\Lambda_0\} \quad (17)$$

provides $\{U_1\}$ from $\{U_0\}$.

- The systems of equations

$$[T_{i,i-1}]\{U_{i-1}\} + [T_{i,i}]\{U_i\} + [T_{i,i+1}]\{U_{i+1}\} = \{F_i\} \quad 1 \leq i \leq n-1, \quad (18)$$

provides the nodal displacements $\{U_{i+1}\}$ from $\{U_{i-1}\}$ and $\{U_i\}$.

- Finally, the last system of equations

$$[T_{n,n-1}]\{U_{n-1}\} + [T_{n,n}]\{U_n\} = \{F_n\} + \{\Lambda_n\}, \quad (19)$$

provides the final nodal velocities $\{\Lambda_n\}$ if it is necessary.

- The matrices appearing in the linear systems to be solved $[T_{i,i+1}]$ are generally non-symmetric, even if the total matrix $[T]$ is symmetric. Thus, for the algorithm presented above, 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 [3] is of the Gauss Seidel nonlinear type, asymmetries do not affect computing time.

3 4D mesh generation

Obtain only one 4D finite element does not pose real problems, except some difficulties of drawing representation (see figure 3). But build a 4D mesh, even 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). But in the general case, and in particular the problem of remeshing, the meshing remains a difficulty.

We present in this paragraph a technique to mesh a 4D domain, using a generalization of what can be done for 2D or 3D domains, when the total number of node remains constant at each time slab.

3.1 4D mesh generation

In figure 3 are represented 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 that the total number of nodes remains the same at time $t = h$.

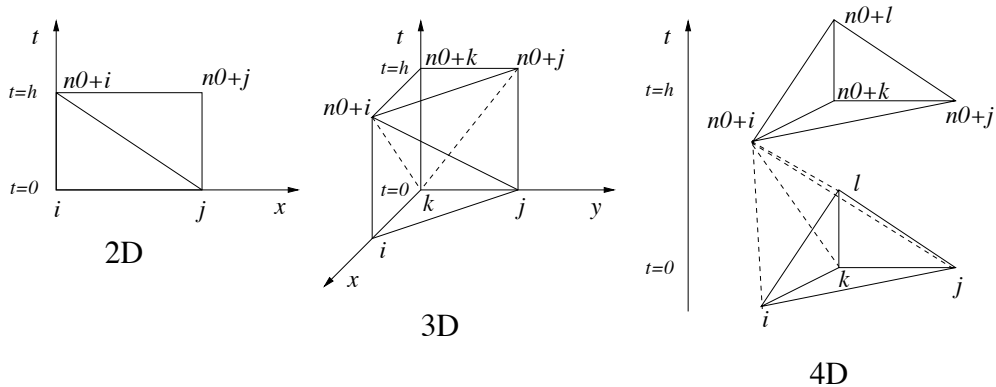


Figure 3: 2D, 3D and 4D space-time mesh

For the 2D mesh, the connectivities are

$$\begin{aligned} & i, j, n_0 + i \\ & j, n_0 + i, n_0 + j. \end{aligned}$$

For the 3D mesh, the connectivities are

$$\begin{aligned} & i, j, k, n_0 + i \\ & j, k, n_0 + i, n_0 + j \\ & k, n_0 + i, n_0 + j, n_0 + k. \end{aligned}$$

Using the previous building of connectivities, obtained by circular permutations, we propose the following generalization of connectivities for the 4D mesh:

$$\begin{aligned} & i, j, k, l, n_0 + i \\ & j, k, l, n_0 + i, n_0 + j \\ & k, l, n_0 + i, n_0 + j, n_0 + k \\ & l, n_0 + i, n_0 + j, n_0 + k, n_0 + l. \end{aligned}$$

This 4D mesh is constituted by 4 hypertetrahedrons¹. We propose to build 4D space-time meshes, resulting from unspecified 3D space meshes, by applying the building technique developed above, at each 3D finite element of the 3D space mesh. But in this case, it must be checked that the total space-time volume is covered by the 4D mesh. For this purpose, we have numerically computed the sum of volumes of each hypertetrahedron of the 4D mesh and have successfully compared it with the total volume generated by the 3D domain Ω multiplied by the time interval $[0, T]$.

For a 3D mesh made up of tetrahedrons, the interfaces between the elements are triangles. In 4D, these interfaces are tetrahedrons (see diagram on the left of figure 4). So, 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. We have observed that a way of doing this is to arrange the nodes of each element 3D in the ascending order. Table 1 gives an example of table of connectivities for an elementary 4D mesh, resulting from the 3D space mesh represented by the left diagram of figure 4. Let us note that this 4D mesh contains 8 finite elements, against 2 for the 3D mesh source and that $n_0 = 5$. In this case, it is checked that the tetrahedra filling the space-time interface (diagram on the right of figure 4) are common to the adjoining elements. Indeed, elements 1 and 5 contain the tetrahedron (1; 2; 4; 6), elements 2 and 6 contain the tetrahedron (2; 4; 6; 7). Lastly, elements 4 and 7 contain the tetrahedron (4; 6; 7; 9).

¹The hypertetrahedron is the four-dimensional tetrahedron. Other names are simplex or pentatope.

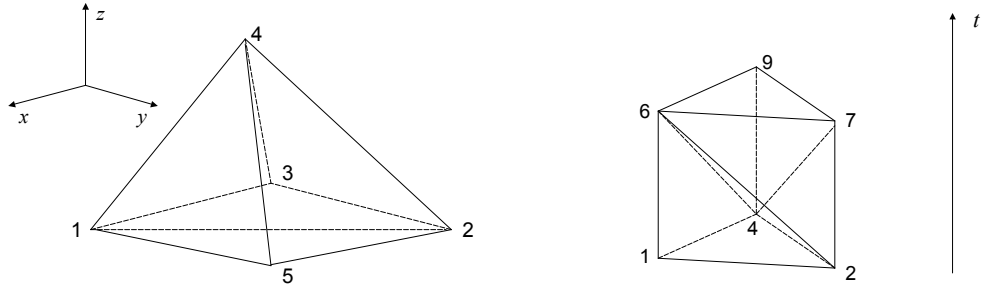


Figure 4: 3D initial mesh (left); part of the 4D mesh generated by the triangle (1;2;4) common at the two finite elements of the 3D initial mesh (right).

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

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

4 Numerical results

We present in this paragraph an example of mechanical problem that is a first tentative to simulate problems of wear between two bodies in contact.

We consider a brake disc subjected to a tightening on a part of the top face of the disc (see source model with coarse mesh in figure 5). The underside of the disc is blocked and the other parts of the boundary are free.

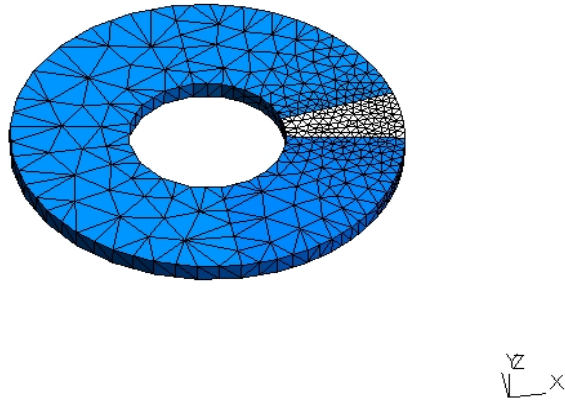


Figure 5: 3D initial coarse mesh: Mesh is finer under the tightening zone, displayed in white.

The disc is made of steel of Young modulus equal to 210 GPa, of Poisson's ratio equal to 0.3 and of density equal to 7800 kg/m³. 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 mesh have been tested: a coarse mesh which contains 634 nodes and 1,752 elements, and a fine mesh which contains 6,476 nodes and 25,856 elements (see figure 10).

The space-time finite elements method was programmed using MATLAB software and was validated on an elementary example of the type of figure 4. We noted that our method of 4D mesh generation required to have a very fine space-time slabs to be convergent, of same order as for explicit methods of integration. We have already noted this drawback for 2D and 3D meshes when the interpolation is linear, but this is not the case when the interpolation is quadratic (see [4]).

Tightening is modeled by a zone of constant pressure of 100 MPa. This zone is moving and course the circumference of the disc with a rotated velocity equal to $V = \frac{1}{2\pi R} \sqrt{\frac{E}{\rho}}$ where R is the average radius of the disc. We build an incremental

4D space-time mesh, which preserves the 3D mesh, but by imposing an axial rotation allowing him to keep its most refined zone under the loading area, as schematized in the figure 6.

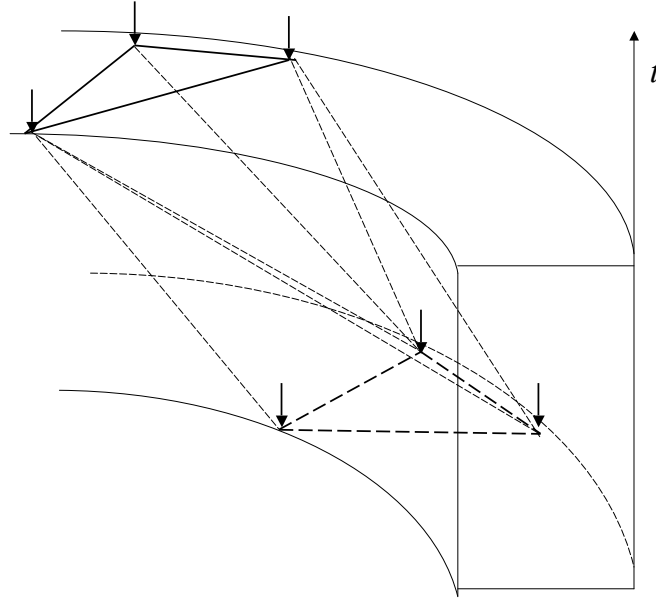


Figure 6: Space-time mesh generation by rotation of the loaded area. The loaded area is represented by arrows. Notice that the mechanical problem considered is 3D in space whereas the figure is presented in 2D in space for the sake of simplicity.

The principle is to keep the same number of nodes during the simulation, but to locate sufficiently large number of them under the loaded area. In this particular case, it is possible to preserve the matrices $T_{i,i-1}$, $T_{i,i}$ and $T_{i,i+1}$ identical for all i , which involves a reduction of the computational time.

The results of the computations presented here have been obtained for space-times slabs duration of 10^{-7} s (this is equivalent to use a time step equal to 10^{-7} s). Vertical displacements obtained with fine and coarse mesh have been compared for points located on a circle of check of radius 70 mm (see figure 7).

Numerical values are gathered in figure 8 for results at time $t = 2.10^{-5}$ s and in figure 9 for results at time $t = 2.10^{-5}$ s. Each point is located by its angle in polar coordinates.

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

Figures 10 and 11 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. On coarse mesh, we can observe that the refined zone really remains under the zone of tightening. The

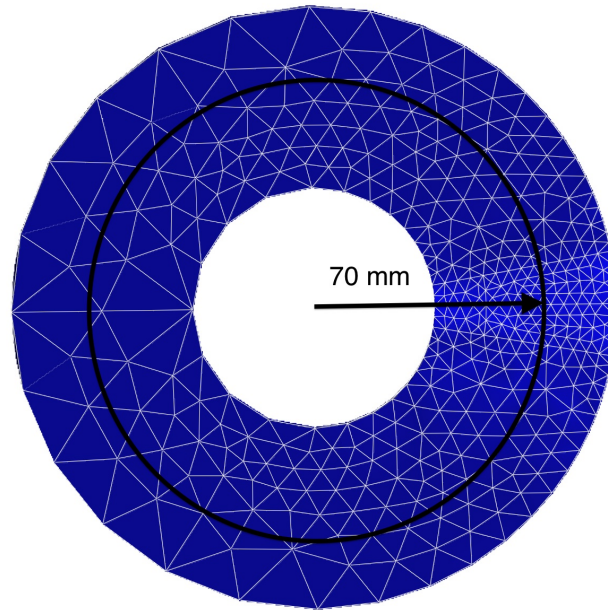


Figure 7: Location of displayed displacements: circle of radius equal to 70 mm at the top of the disk.

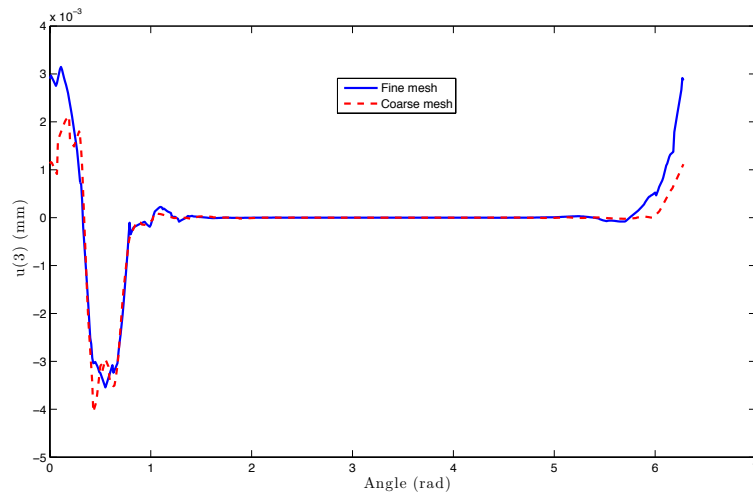


Figure 8: Comparison of vertical displacements for points located on the measurement circle, expressed in mm, at $t = 2.10^{-5}$ s. Each point is located by its angular coordinate, expressed in radian.

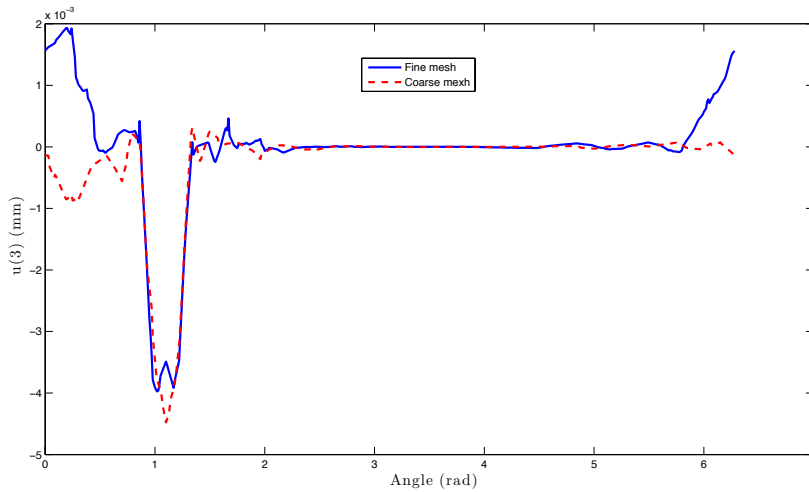


Figure 9: Comparison of vertical displacements for points located on the measurement circle, expressed in mm, at $t = 4.10^{-5}$ s. Each point is located by its angular coordinate, expressed in radian.

distribution of the norm of node displacements is similar for the two positions of the load. It is important to notice that they are incremental displacements and not total displacements.

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

5 Conclusion

The method of space-time mesh generation for 4D domains using simplex elements was presented in this paper made it possible to develop a technique of mesh adaptation able to follow a mobile loading zone. This original technique has been developed to ensure a minimal computational time and does not require coarse-to-fine and fine-to-coarse mesh transfer operators. However, due to technical difficulties, the frontal resolution in the case of $d = 3$ has not been implemented yet. Nevertheless, we propose a particular incremental remeshing technique based on the construction of 4D space-time meshes able to follow an evolutionary loaded zone. To show the entire capacities of this method, it will be necessary to go much further in mechanical applications (contact mechanics like in [13] and wear) and to propose a technique of 4D mesh adaptation and remeshing by frontal approach as presented in the articles [3] and [4].

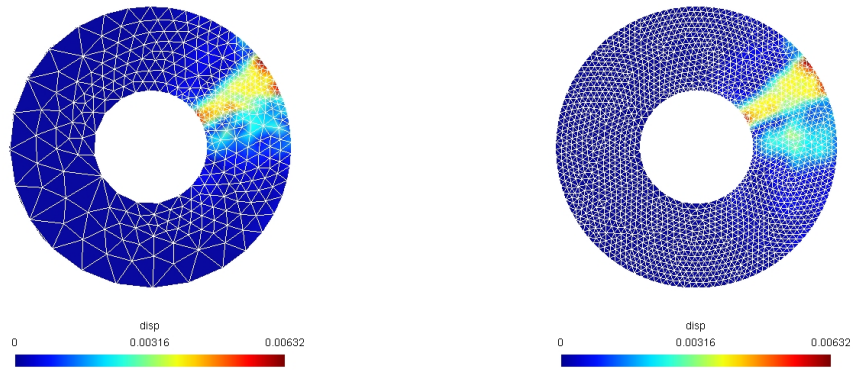


Figure 10: Isovalues of the norm of nodal displacements, expressed in mm, at $t = 2.10^{-5}$ s for the coarse mesh(left) and for the fine mesh (right).

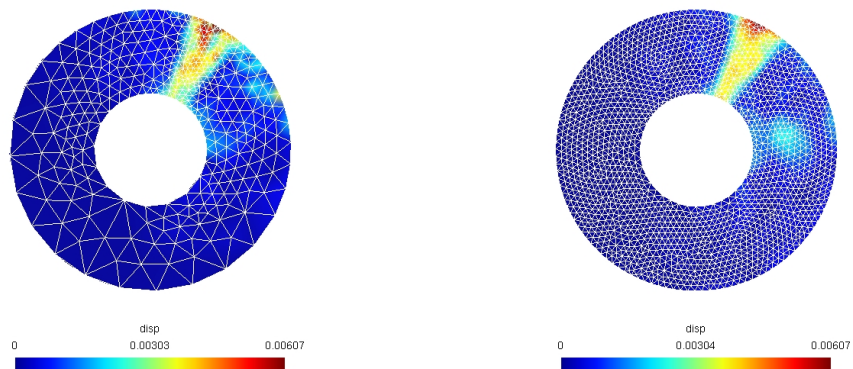


Figure 11: Isovalues of the norm of nodal displacements, expressed in mm, at $t = 4.10^{-5}$ s for the coarse mesh (left) and for the fine mesh (right).

References

- [1] R. Abedi, B. Petracovici, R.B. Haber, “A space time discontinuous Galerkin method for linearized elastodynamics with element-wise momentum balance”, *Computer Methods in Applied Mechanics and Engineering*, 195, 3247-3273, 2006.
- [2] L. Adélaïde, F. Jourdan, C. Bohatier, “New results on mesh adaptation in space-time finite element method”, *ASME Engineering Technology Conference on Energy*, ETCE 2002/ STRUC-29042, Houston, Texas, 4-5 February, 2002.
- [3] L. Adélaïde, F. Jourdan, C. Bohatier, “Frictional contact solver and mesh adaptation in Space-Time Finite Element Method”, *European Journal of Mechanics A/solids*, 22, 633-647, 2003.
- [4] L. Adélaïde, F. Jourdan, C. Bohatier, “Méthode des éléments finis espace-temps et remaillage”, *Revue Européenne des éléments finis*, 12, 427-459, 2003.
- [5] M. Anderson, J.-H. Kimn, “A numerical approach to space-time finite elements for the wave equation”, *Journal of Computational Physics*, 226, 466-476, 2007.
- [6] J.H Argyris, D.W. Scharpf, “Finite elements in time and space”, *Nuclear Engineering and Design*, 10, 456-464, 1969.
- [7] C. C. Chien, C. S. Yang, J. H. Tang, “Three-dimensional transient elastodynamic analysis by a space and time-discontinuous Galerkin finite element method”, *Finite Elements in Analysis and Design*, 39, 561-580, 2003.
- [8] H. Huang, F. Costanzo, “On the use of space-time finite elements in the solution of elasto-dynamic problems with strain discontinuities”, *Computer Methods in Applied Mechanics and Engineering*, 191, 5315-5343, 2002.
- [9] T. J. R. Hughes, G. M. Hulbert, “Space-time finite element methods for elastodynamics: formulations and error estimates”, *Computer Methods in Applied Mechanics and Engineering*, 66, 339-363, 1988.
- [10] G. M. Hulbert, T. J. R. Hughes, “Space-time finite element methods for second-order hyperbolic equations”, *Computer Methods in Applied Mechanics and Engineering*, 84, 327-348, 1990.
- [11] A. V. Idesman, “Solution of linear elastodynamics problems with space-time finite elements on structured and unstructured meshes”, *Computer Methods in Applied Mechanics and Engineering*, 196, 1787-1815, 2007.
- [12] F. Jourdan, P. Bussy, “Large time increment method in dynamic regularization : sheet cutting simulations”, *Computer Methods in Applied Mechanics and Engineering*, 190, 1245-1259, 2000.
- [13] L. Karaoglan, A. K. Noor, “Space-time finite element methods for sensitivity analysis of contact/impact response of axisymmetric composite structures”, *Computer Methods in Applied Mechanics and Engineering*, 144, 371-389, 1997.
- [14] D.K. Khalmanova, F. Costanzo, “A space-time discontinuous Galerkin finite element method for fully coupled linear thermo-elasto-dynamic problems with strain and heat flux discontinuities”, *Computer Methods in Applied Mechanics and Engineering*, 197, 1323-1342, 2008.
- [15] P. Ladevèze, “Non linear computational structural mechanics, new approaches

and non incremental methods of calculation”, Springer Verlag, 1999.

- [16] O.C. Zienkiewics and C.J. Parekh, “Transient field problems-two and three dimensional analysis by isoparametric finite elements”, *International Journal for Numerical Methods in Engineering*, 2, 61-71, 1970.