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

Serge Dumont, Franck Jourdan, Tarik Madani

RESEARCH

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

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

[*]Correspondence: fjourdan@um2.fr
[1]LMGC, CNRS UMR 5508, Université Montpellier 2, Place Eugene Bataillon, 34095 Montpellier, France
Full list of author information is available at the end of the article
[†]Equal contributor

Abstract

Background: In this article we propose a Space-Time Finite Element Method (STFEM) for the resolution of mechanical problems involving three dimensions in space and one in time. A special attention will be paid with the non separation of the space and time variables because this kind of interpolation is well-suited to mesh adaptation.

Methods: 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.

Results: We have tested the method on a linearised 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 is studied and compared with existing methods.

Conclusion: This work is a first implementation of 4D space-time remeshing. A stability criteria for the method is established, as well as a convergence rate of about 2. 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 Background

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 [19]), the Discontinuous Galerkin method [12, 13, 18], and our method which is a Continuous Galerkin method. In most publications on the Discontinue Galerkin method, like in [6], 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 with 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 [4] and [15] is the size of the linear systems to be solved. The use of a laminated mesh does not allow us to assemble 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 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 organised 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. 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: 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 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 $[0,T]$. As in the discontinuous Galerkin method [11, 12, 17], and the LArge Time INcrement (LATIN) method [16, 19], 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

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

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 on the left hand side of the equation (2) is integrated by part 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) has been proposed in [5, 25]. Their discretisation use 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 [17, 20], 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 [12] has been developed on unstructured meshes. It also employs the discontinuous Galerkin method in time and incorporates stabilising terms of 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 isoparametrics. 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 for the element E_e, φ_i^e are the interpolation functions and u_i^e the nodal displacements. Using a matricial notation, one has

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

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

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

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

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

where

$$M_e = \int \int_{E_e} \rho \frac{\partial N_e^T}{\partial t} \frac{\partial N_e}{\partial t} dx dt \tag{8}$$

is the elementary matrix relative to the inertia forces. One can notice that M_e is symmetric. Concerning the discretisation 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 discretised by

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

and

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

where the elementary matrix K_e relative to internal forces is

$$K_e = \int \int_{E_e} B_e^T a B_e dx dt \tag{14}$$

and the elementary vector F_e relative to external forces is

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

This space-time discretisation 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 conduced in such a way that all nodes in a same stratum have close numbers, then the left-hand side of the system (16) verifies

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

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

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

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

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 precise enough and it is necessary to use the discontinuous Galerkin, 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.

— 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. In fact, 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.

— 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 systems 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 a 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 penalising 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 [2] is of the Gauss Seidel nonlinear type, asymmetries do not affect computing time.

3 4D mesh and remeshing

In order to be able to propose a remeshing technique it is necessary to be able to build 4D meshes. Obtaining only one 4D finite element does not pose real problems, except for some difficulties in graphic representation (see figure 3). On the other hand, 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). But in the general case, and in particular the problem of remeshing which is what 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 generalisation of connectivities for the 4D mesh.

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

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

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

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

This 4D mesh is constituted by 4 hypertetrahedrons[1]. 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. But 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 diagram at the bottom 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. A way of doing this is to arrange the nodes of each element 3D in the 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 8 finite elements, against 2 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 bottom 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).

3.2 Remeshing technique

In this paragraph we present our technique of space-time mesh adaptations. In the literature, many articles on mesh adaptations, [7], [9], [12, 13], [23, 24], [8], [10] and [22], can be found. Among these papers a large number deals with space-time mesh adaptations. They use the discontinuous Galerkin method. In the majority, the approach is incremental, i.e. remeshing is carried out at given steps of times. 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

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

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 article [3] 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 [2]). In addition, we developed a nonincremental technique of mesh adaptation, based on non-structured space-times meshes.

Some teams have already worked on this problem. One can quote the works of Hugues and Hulbert [12, 13], Tezduyar et al. [23, 24], Idesman et al. [14, 15]. 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 [15] and [4].

In the context of the continuous Galerkin method, we suggested, in [1] and [3], a nonincremental solution, which substitutes the concept of step of time by that of a "space-time front". Erickson et al. [8] 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. [21] in their multi-field space-time discontinuous Galerkin method, for $d = 1$ and 2 in linearised 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 paragraph. This technique is aimed at application in simulating problems of wear between two bodies in contact.

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 [3] and compared with the Newmark integration scheme. Here we resume the main results. Let δ and θ be the parameters of the Newmark integration scheme, they verify

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

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 [3] 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 stable for all time step values.

- 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 stable under conditions on the time step.

- for higher dimensions (3D and 4D) no direct relationship between the STFEM and the Newmark method has been established. But we noted that our method required a very fine space-time slabs "small time step of discretization", of the same order as 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 change in space-time dimension do not affect the method.

In the present study a specific numerical investigation has been carried out to estimate the conditions of stability 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 square section of 0.01×0.01 m^2 (see figure 6). The Young modulus E was equal to 1,000 Pa, the Poisson's ratio ν was equal to 0.3 and the density ρ was equal to 680 kg/m^3.

We built boundaries conditions in order to respect 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 Neuwman conditions were imposed. On the other 4 faces (Σ_c) Dirichlet conditions were imposed in order to satisfy the analytic solution (20). The volume external force density f_d was assumed to be null, but for that, the wave velocity c must verifies

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

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

Linear fitting suggests that the stability criteria 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 is like the Newmark scheme, for 1D, and 2D. For the convergence analysis of the STEFEM in 4D, we used the previous example. The time discretisation Δt is scaled with respect to h. It is deduced from the stability criteria 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. Theesults are plotted in figure 8.

We can note that the convergence rate for the method is of about 2.

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 source model with 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 7,800 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 1,752 elements, and a fine mesh which contains 6,476 nodes and 25,856 elements (see figure 13).

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. 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-times 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 (see figure 10).

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 $t = 2.10^{-5}$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. But 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.

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 computational time is 6.6 times faster using the coarse mesh.

6 Conclusion

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. Conceptually this is not a new idea, but this work is a first implementation of 4D space-time remeshing.

However, to demonstrate the entire 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 articles [2] and [3] or in [8] and [21].

Competing interests
The authors declare that they have no competing interests.

Author's contributions
FJ, SD and TM worked on the algorithms, on the computation, drafted some parts of the manuscript and carried out detailed revisions. All authors read and approved the final manuscript.

Author details
[1]LMGC, CNRS UMR 5508, Université Montpellier 2, Place Eugene Bataillon, 34095 Montpellier, France. [2]LAMFA, CNRS UMR 6140, Université de Picardie Jules Verne, 33, rue Saint-Leu, 80039 Amiens, France.

References

1. L. Adelaide, 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.
2. L. Adelaide, F. Jourdan, C. Bohatier, Frictional contact solver and mesh adaptation in Space-Time Finite Element Method, European Journal of Mechanics A/solids, vol. 22, 633-647, 2003
3. L. Adelaide, F. Jourdan, C. Bohatier, Méthode des éléments finis espace-temps et remaillage, Revue Européenne des éléments finis, Vol. 12 - no. 4, pp 427-459, 2003
4. M. Anderson, J.-H. Kimn, A numerical approach to space-time finite elements for the wave equation, Journal of Computational Physics, Volume 226, Issue 1, Pages 466-476, 2007
5. J.H Argyris and D.W. Scharpf, Finite elements in time and space, Nucl. Engrg. Design 10 (1969), pp. 456-464.
6. 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, Volume 39, Issue 7, Pages 561-580, 2003
7. R. Donea, A. Huerta, J. Sarrate, Arbitrary Lagrangian-Eulerian for fluid-rigid body interaction, Comput. Methods Appl. Mech. Engrg., Vol. 190, pp. 3171-3188, 2001
8. J. Erickson, D. Guoy, J.M. Sullivan, A. Üngör, Building spacetime meshes over arbitrary spatial domains, in: Proc. of the 11th International Meshing RoundTable, Sandia, pp. 391-402, 2002
9. C. Farhat, C. Degand, A three-dimensional torsional spring analogy method for unstructured dynamic meshes, Computers and Structures, Vol. 80, pp. 305-316, 2002
10. Y. T. Feng, D. Peric, A time-adaptive space-time finite element method for incompressible Lagrangian flows with free surfaces : computational issues, Comput. Methods Appl. Mech. Engrg., Vol. 190, pp. 499-518, 2000
11. H. Huang, F. Costanzo, On the use of space-time finite elements in the solution of elasto-dynamic problems with strain discontinuities, Comput. Methods Appl. Mech. Engrg., Vol. 191, Issue 46, pp. 5315-5343, 2002
12. T. J. R. Hughes, G. M. Hulbert, Space-time finite element methods for elastodynamics: formulations and error estimates, Comput. Methods Appl. Mech. Engrg. Vol. 66, pp 339-363, 1988
13. G. M. Hulbert, T. J. R. Hughes, Space-time finite element methods for second-order hyperbolic equations, Comput. Methods Appl. Mech. Engrg., Vol. 84, Issue 3, pp. 327-348, 1990
14. A. Idesman, R. Niekamp, E. Stein, Continuous and discontinuous Galerkin methods with finite elements in space and time for parallel computing of viscoelastic deformation, Computer Methods in Applied Mechanics and Engineering, Volume 190, Issues 8-10, Pages 1049-1063, 2000
15. A. V. Idesman, Solution of linear elastodynamics problems with space-time finite elements on structured and unstructured meshes, Comput. Methods Appl. Mech. Engrg, Vol. 196, Issues 9-12, pp. 1787-1815, 2007
16. F. Jourdan, P. Bussy, Large time increment method in dynamic regularization : sheet cutting simulations, Comput. Methods Appl. Mech. Engrg., Vol. 190, pp. 1245-1259, 2000
17. L. Karaoglan, A. K. Noor, Space-time finite element methods for sensitivity analysis of contact/impact response of axisymmetric composite structures, Comput. Methods Appl. Mech. Engrg., Vol. 144, pp. 371-389, 1997

18. 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, Comput. Methods Appl. Mech. Engrg., Vol. 197, Issues 13-16, pp. 1323-1342, 2008
19. P. Ladevèze, Non linear computational structural mechanics, new approaches and non incremental methods of calculation, Springer Verlag, 1999
20. X.D. Li, N.-E. Wiberg, Implementation and adaptivity of space-time finite element method for structural dynamics, Comput. Methods Appl. Mech. Engrg. 156 (1998), pp. 211-229.
21. S.T. Miller, B. Kraczek, R.B. Haber, D.D. Johnson, Multi-field spacetime discontinuous Galerkin methods for linearized elastodynamics, Computer Methods in Applied Mechanics and Engineering, Volume 199, Issues 1-4, Pages 34-47, 2009
22. R. Mohr, A. Menzel, P. Steinmann, A consistent time FE-method for large strain elasto-plasto-dynamics, Comput. Methods Appl. Mech. Engrg., Vol. 197, Issues 33-40, pp. 3024-3044, 2008
23. T. E. Tezduyar, S. Sunil, Enhanced-discretization space-time technique (EDSTT), Comput. Methods Appl. Mech. Engrg., Vol. 193, Issues 15-16, pp. 1385-1401, 2004
24. T. E. Tezduyar, S. Sathe, R. Keedy, K. Stein, Space-time finite element techniques for computation of fluid-structure interactions, Comput. Methods Appl. Mech. Engrg., Vol. 195, Issues 17-18, pp. 2002-2027, 2006
25. O.C. Zienkiewics and C.J. Parekh, Transient field problems-two and three dimensional analysis by isoparametric finite elements, Int. J. Numer. Methods Engrg. 2 (1970), pp. 61-71.

Figures

Figure 1 2D space-time finite element

Figure 2 2D regular space-time mesh

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

Figure 4 3D initial mesh (scheme at the top); 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 bottom)

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

[height=6cm]Maillage-poutre.pdf

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

Figure 7 Time step discretisation size Δt necessary for stability versus average size h of the 3D finite elements

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 logarithmic scale

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

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

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 radian.

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 radian.

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)

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

Tables

Additional Files

Additional file 1 — Figure 1
Additional file 2 — Figure 2
Additional file 3 — Figure 3
Additional file 4 — Figure 4
Additional file 5 — Figure 5
Additional file 6 — Figure 6
Additional file 7 — Figure 7
Additional file 8 — Figure 8
Additional file 9 — Figure 9
Additional file 10 — Figure 10
Additional file 11 — Figure 11
Additional file 12 — Figure 12
Additional file 13 — Figure 13
Additional file 14 — Figure 14

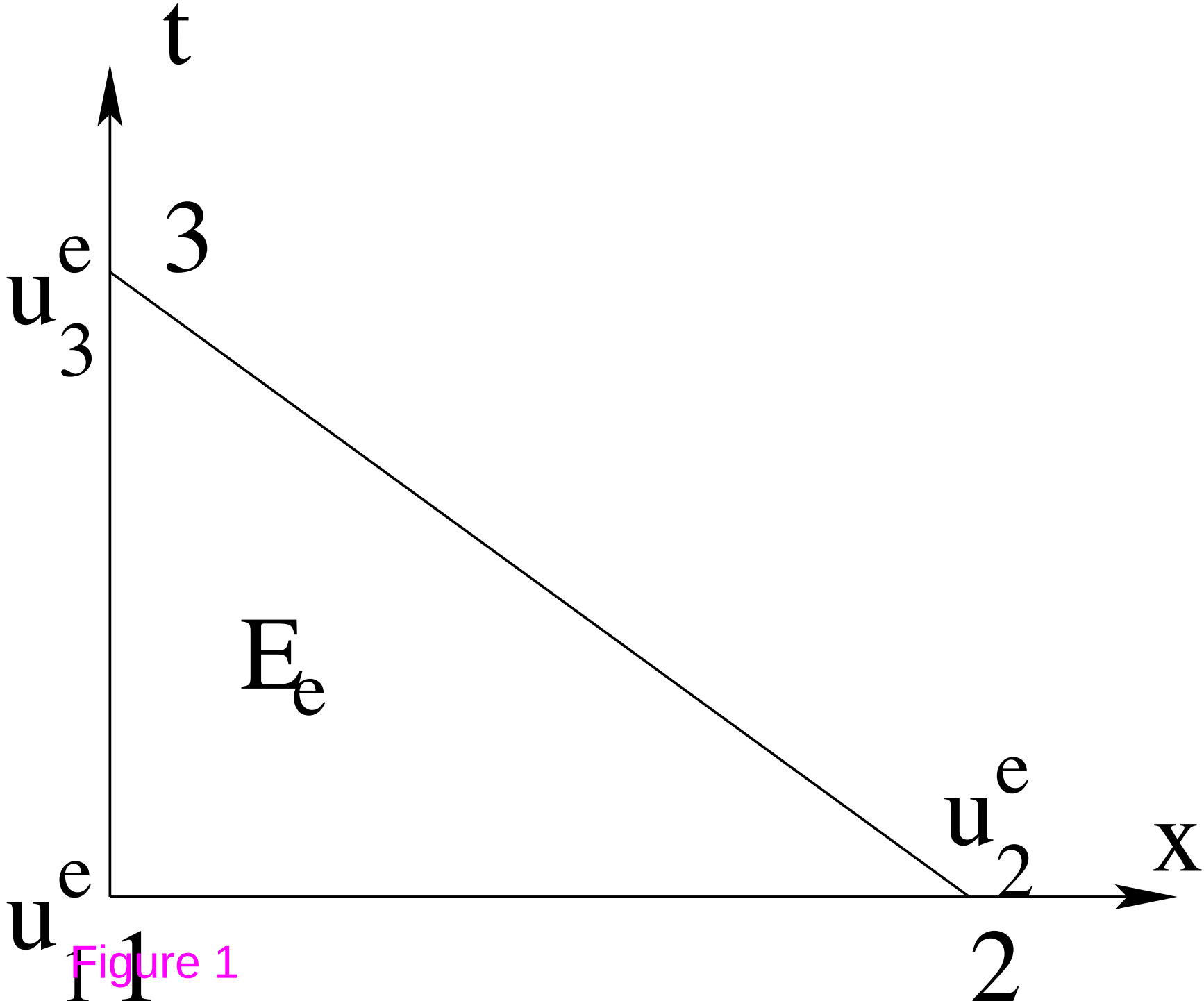


Figure 1

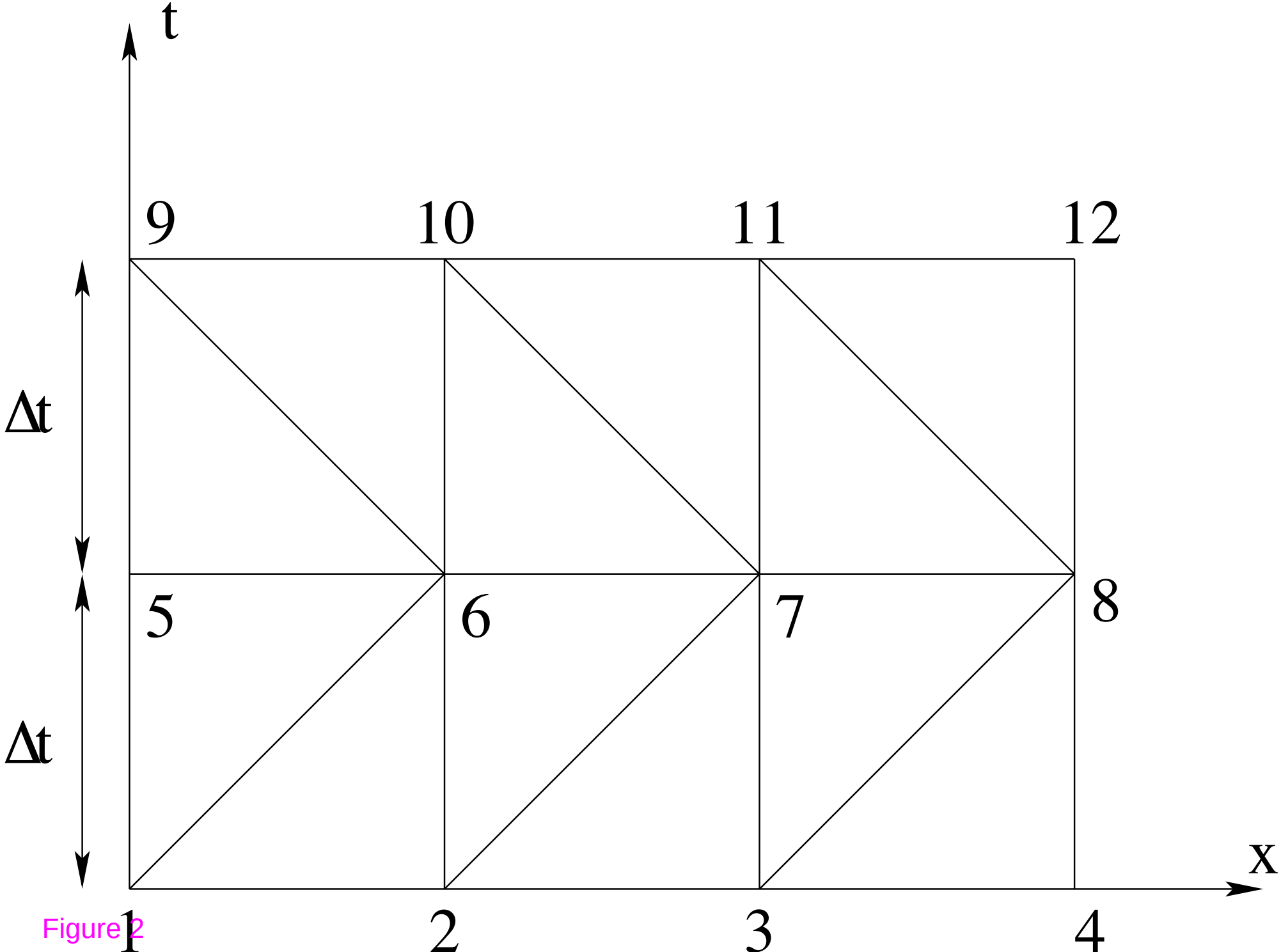
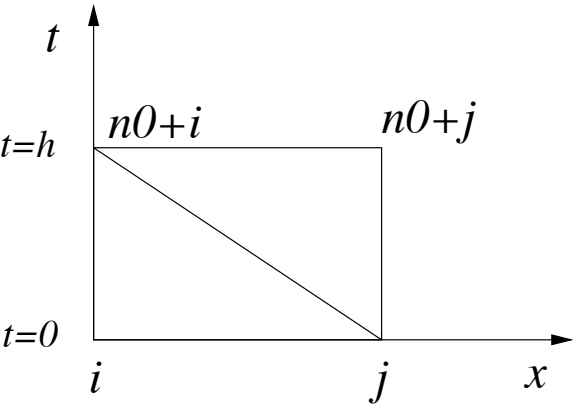
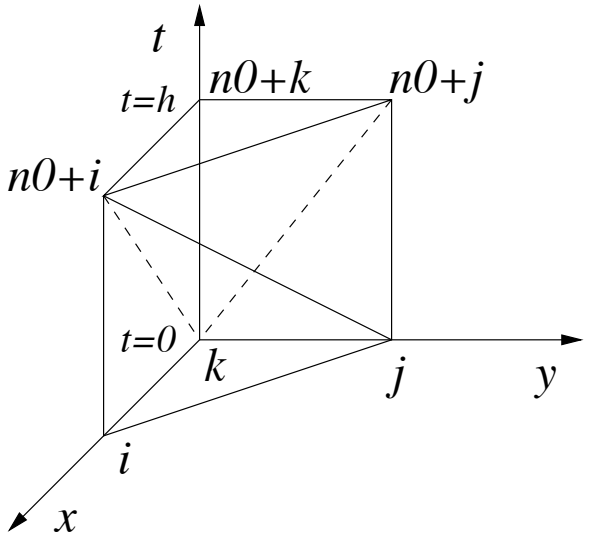


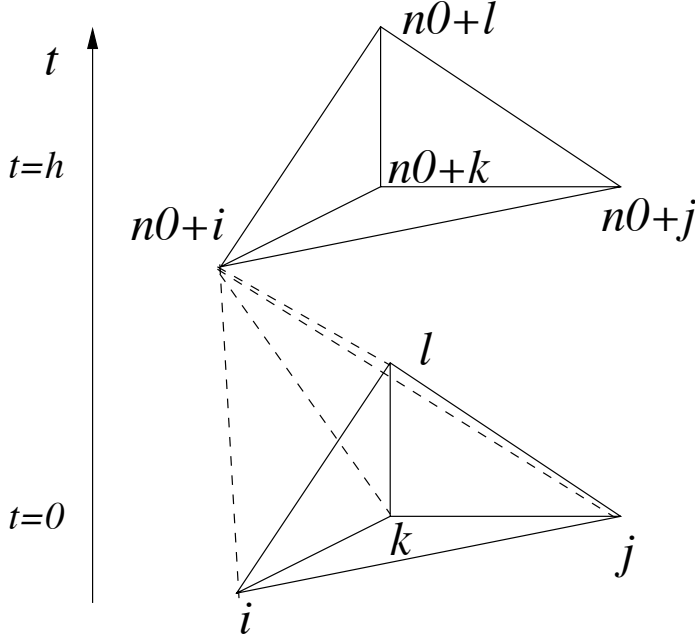
Figure 2



2D

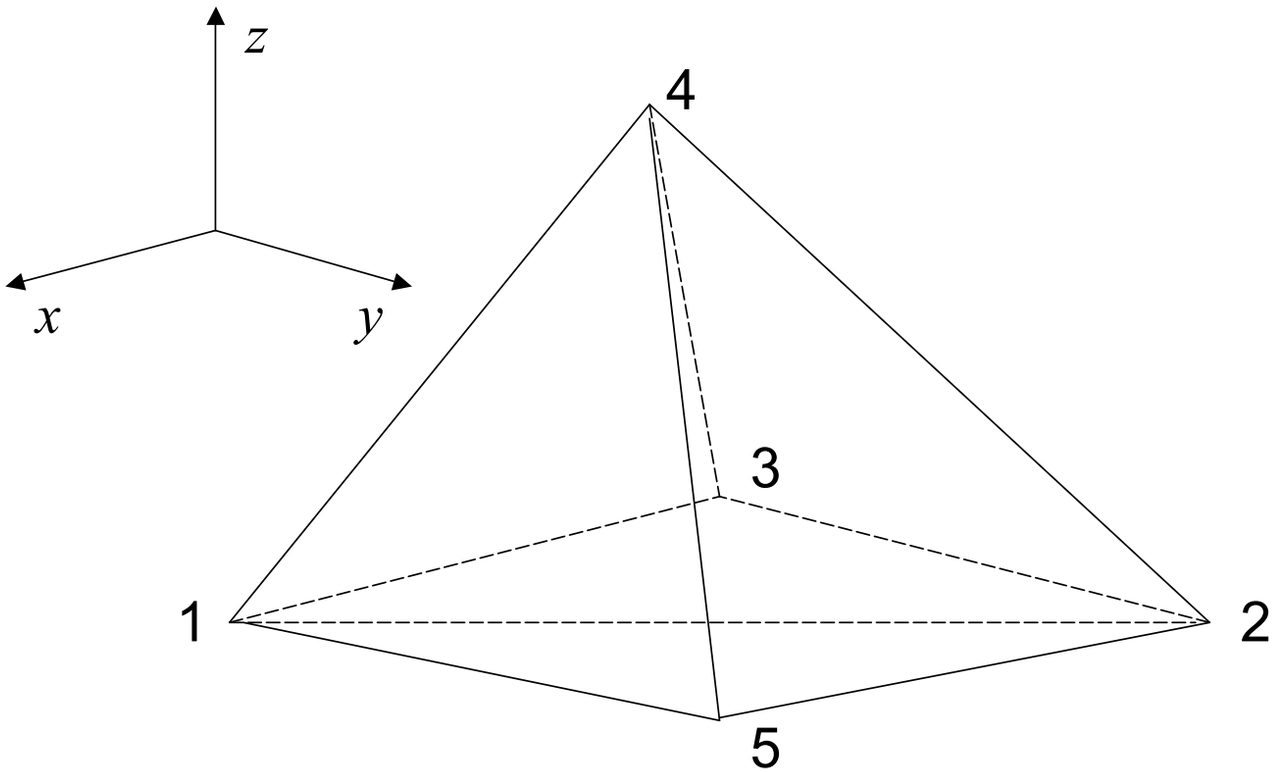


3D



4D

Figure 3



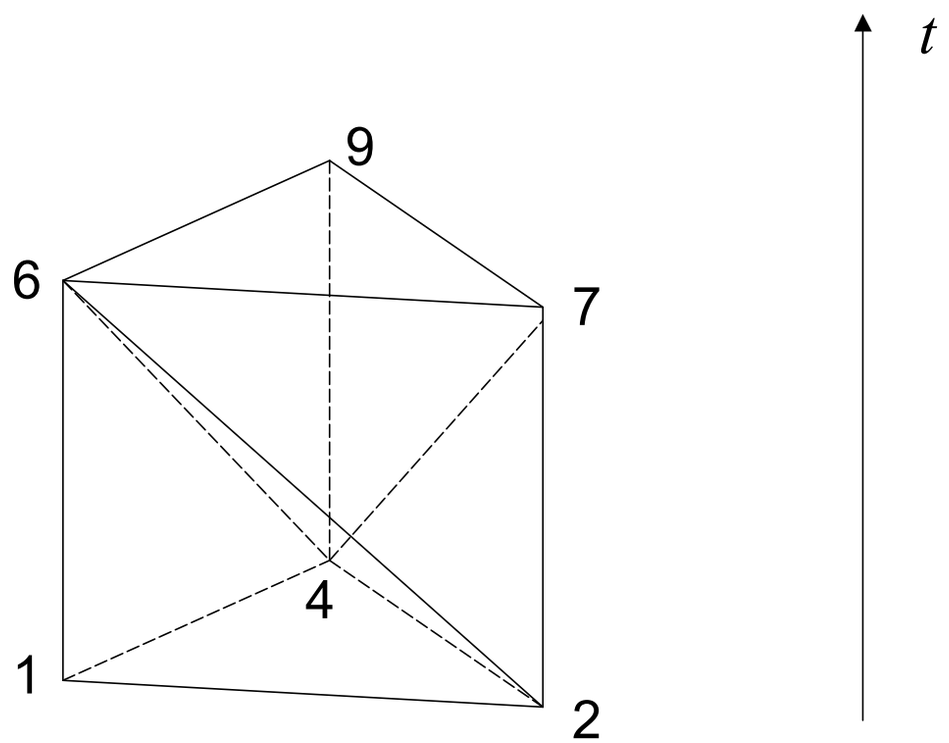


Figure 4

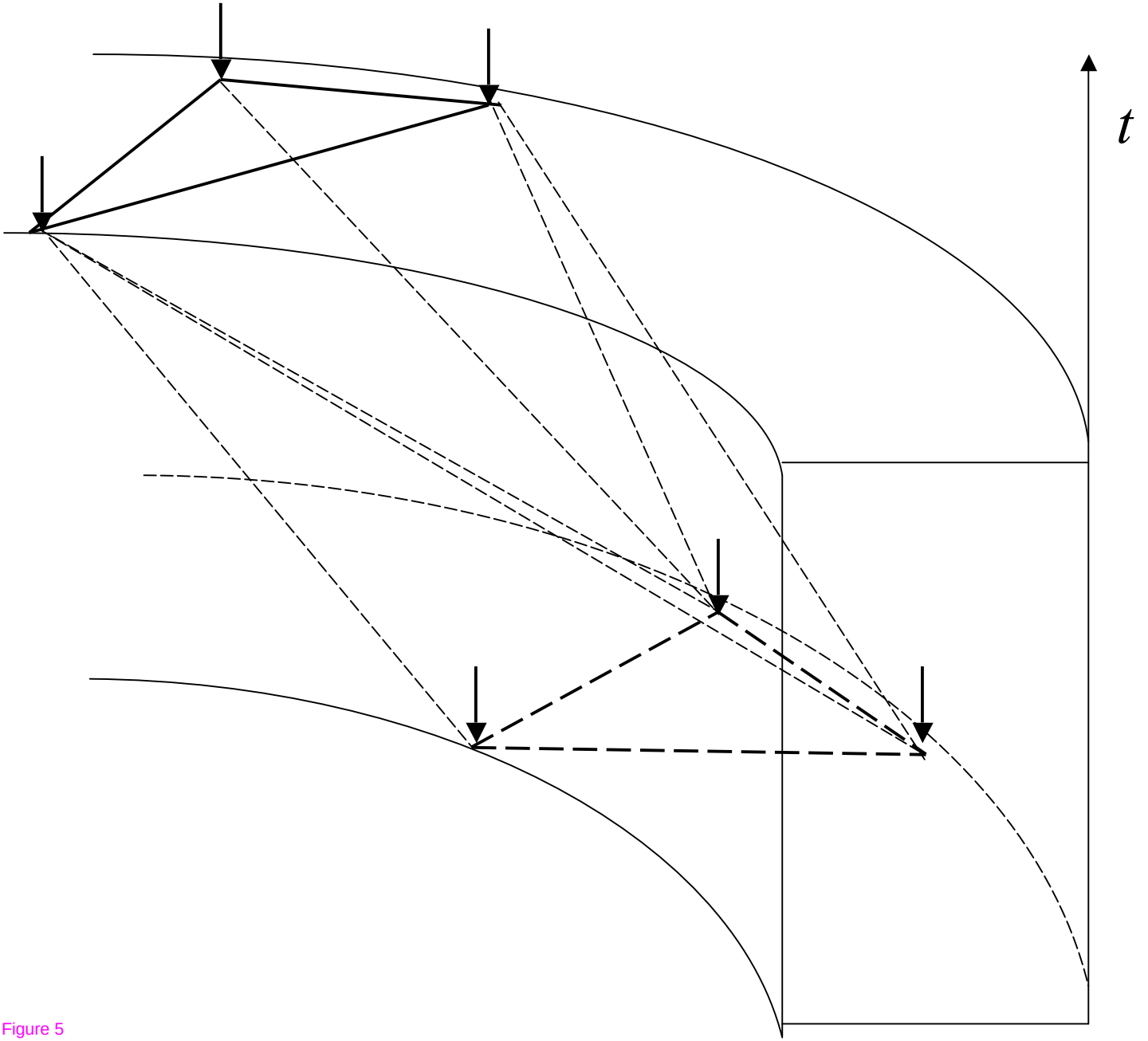
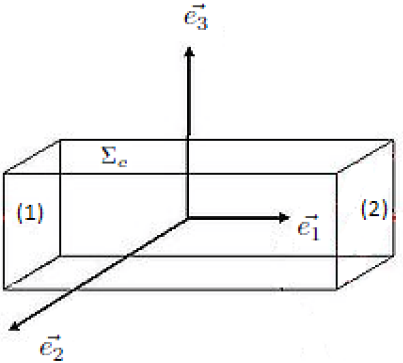


Figure 5


$\vec{e}_3$
Σ_e
(1)
(2)
$\vec{e}_1$
$\vec{e}_2$

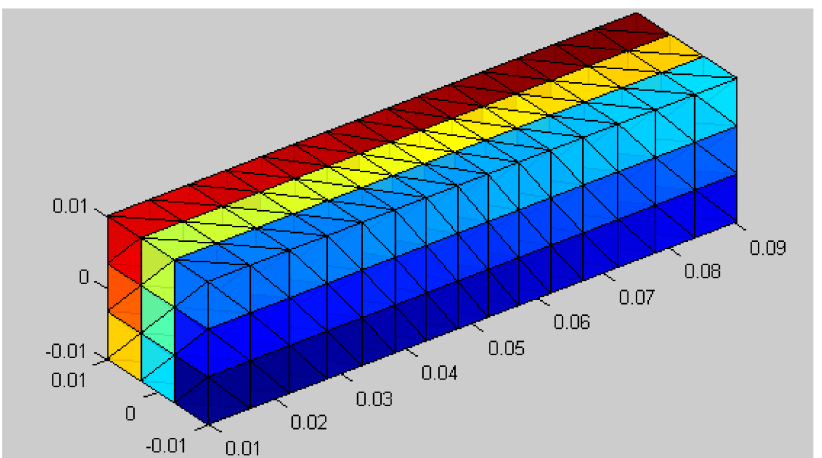

0.01
0
-0.01
0.01
0
-0.01
0.01
0.02
0.03
0.04
0.05
0.06
0.07
0.08
0.09

Figure 6

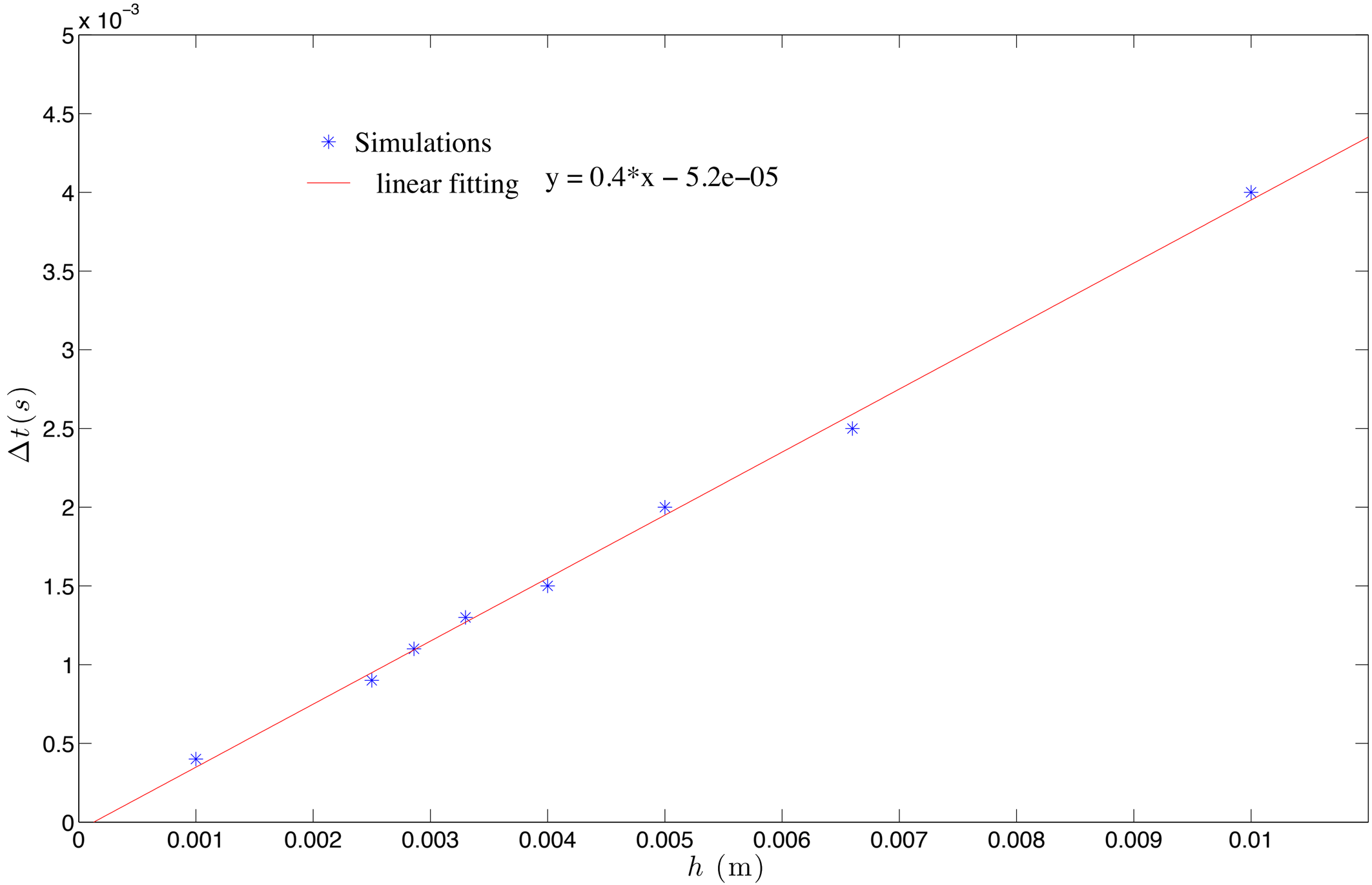

x 10^−3
5
4.5
4
3.5
3
2.5
2
1.5
1
0.5
0
Simulations
linear fitting
y = 0.4*x − 5.2e−05
Δt (s)
0
0.001
0.002
0.003
0.004
0.005
0.006
0.007
0.008
0.009
0.01
h (m)

Figure 7

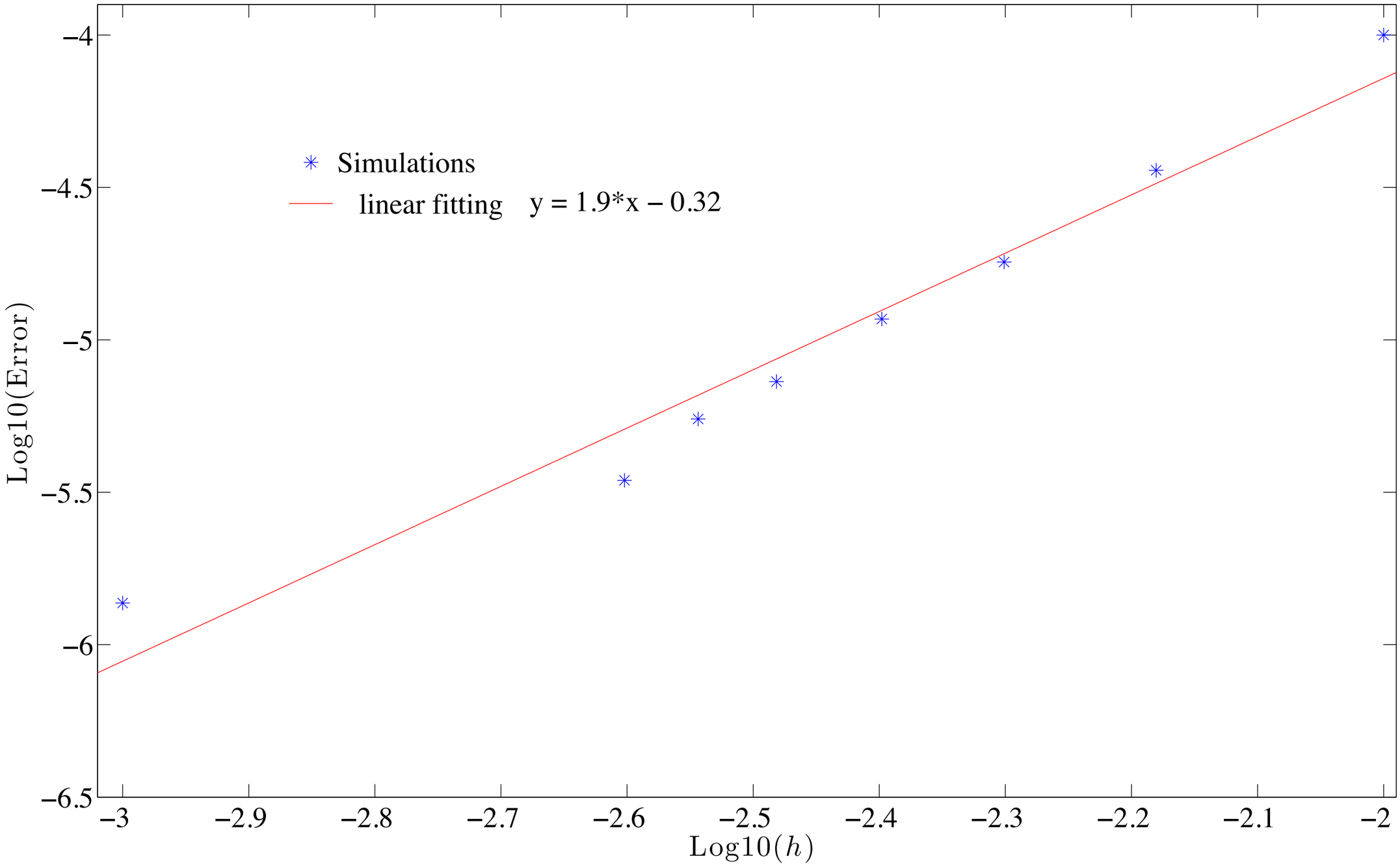

Simulations
linear fitting y = 1.9*x – 0.32
Log10(Error)
Log10(h)
−4
−4.5
−5
−5.5
−6
−6.5
−3
−2.9
−2.8
−2.7
−2.6
−2.5
−2.4
−2.3
−2.2
−2.1
−2

Figure 8

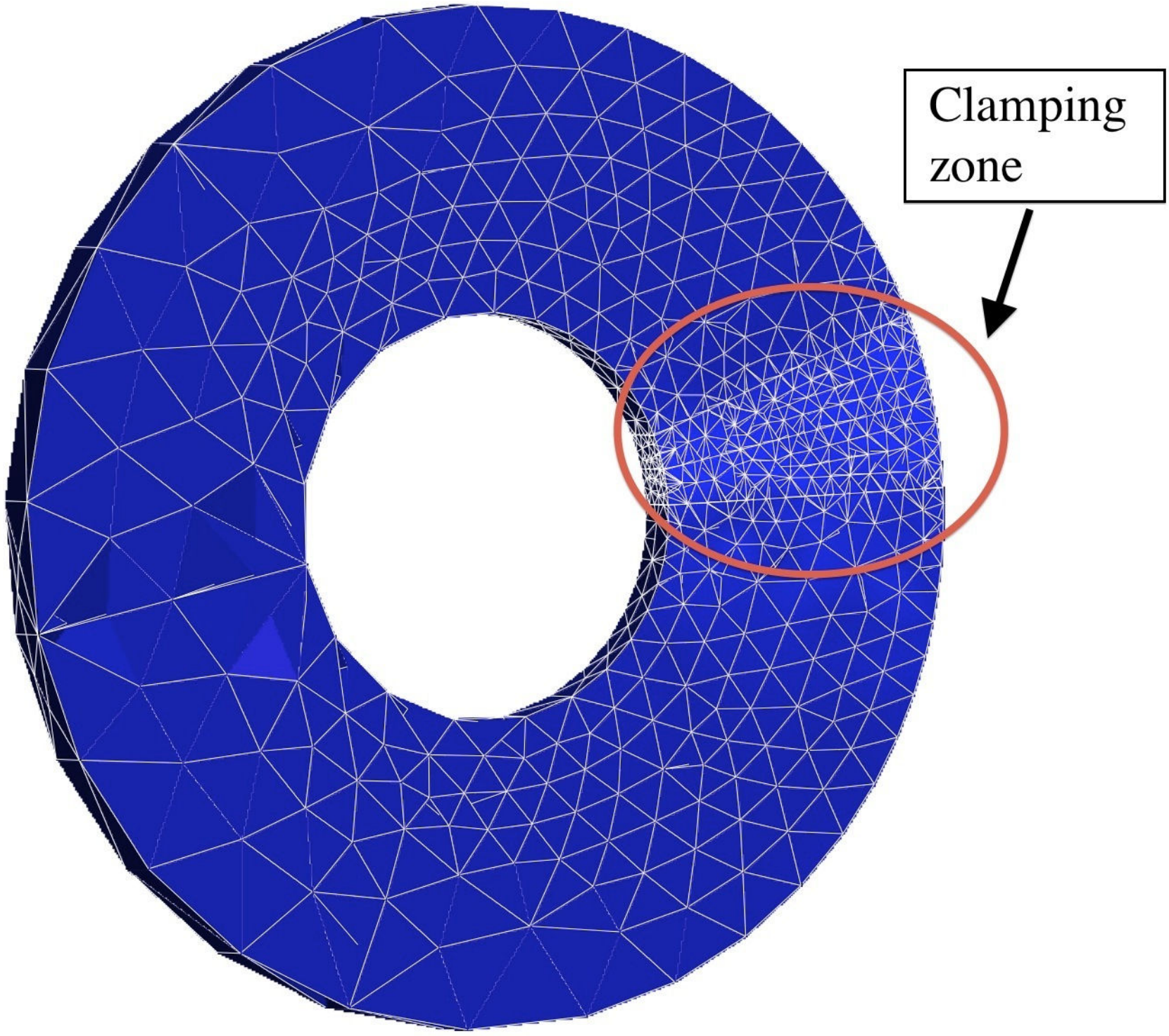


Figure 9

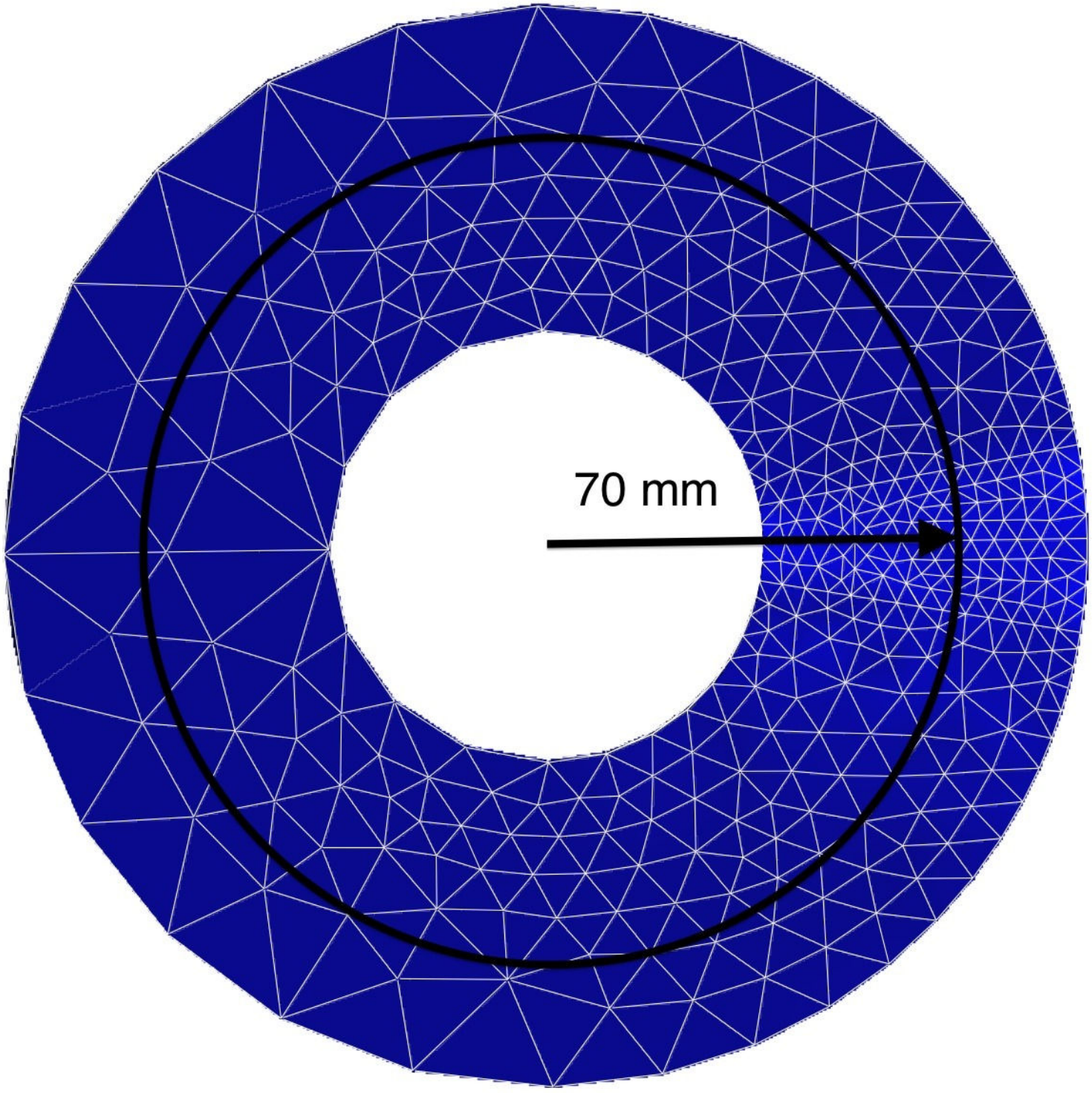


Figure 10

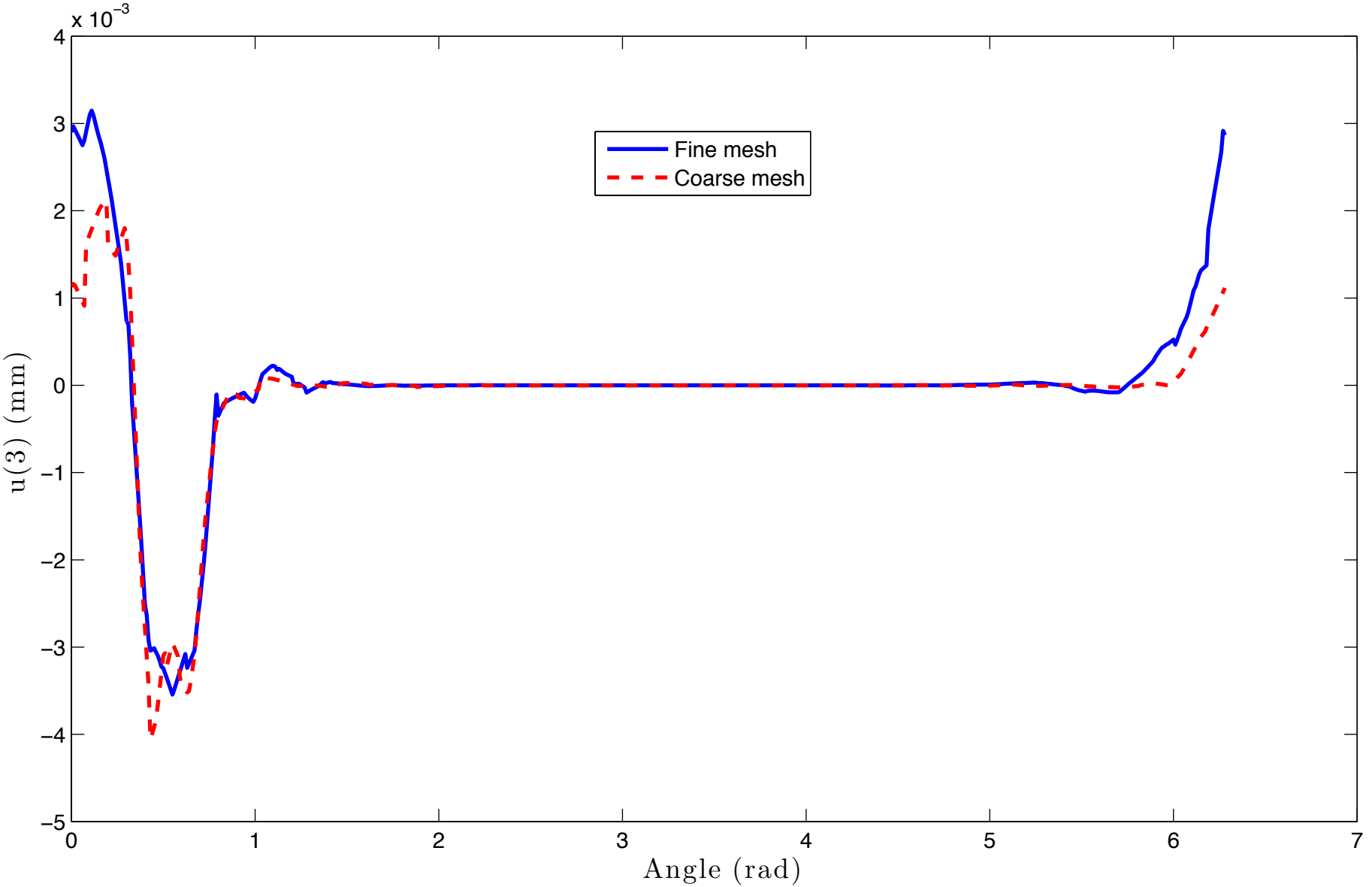

x 10^-3
4
3
2
1
0
−1
−2
−3
−4
−5
0
1
2
3
4
5
6
7
u(3) (mm)
Angle (rad)
Fine mesh
Coarse mesh

Figure 11

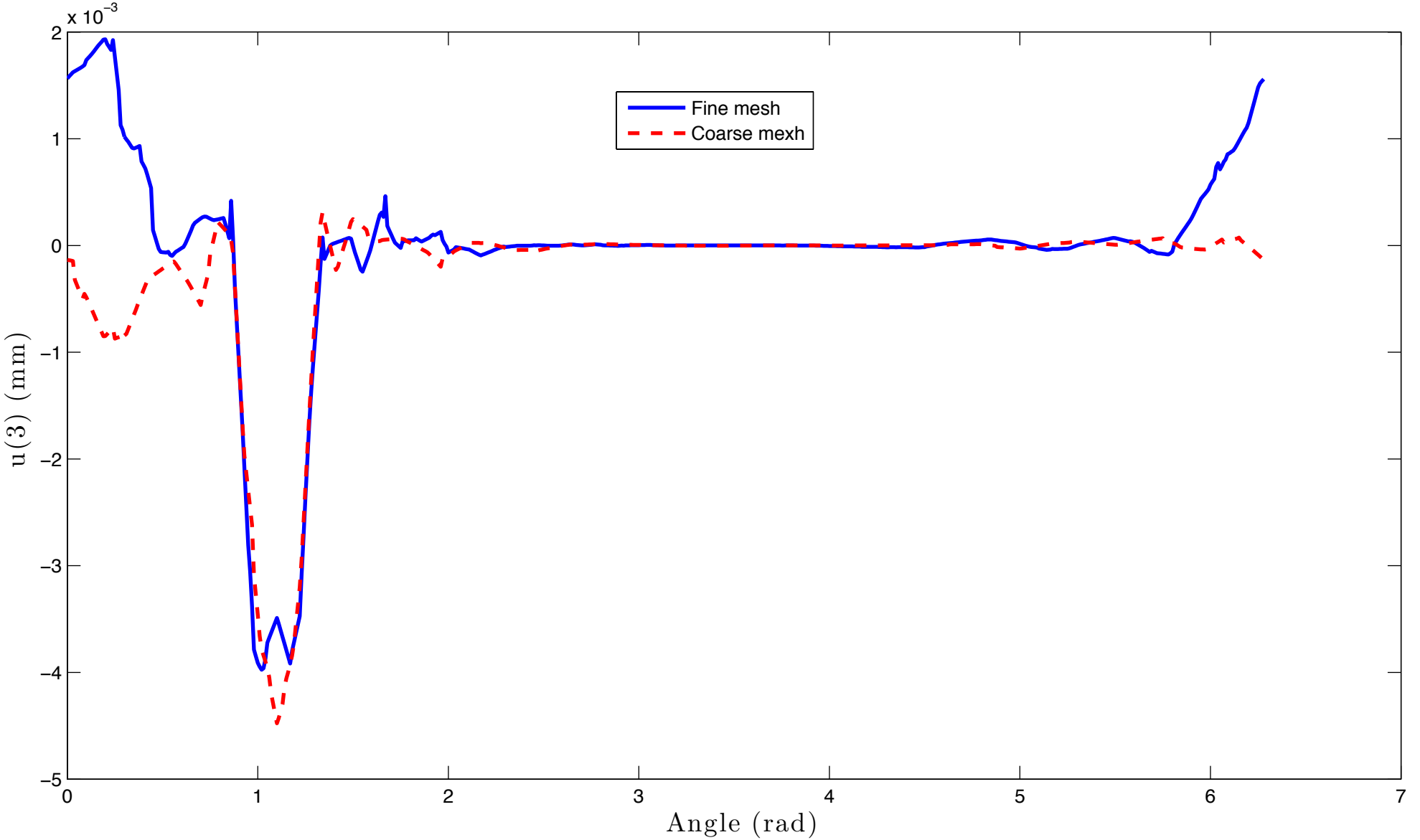
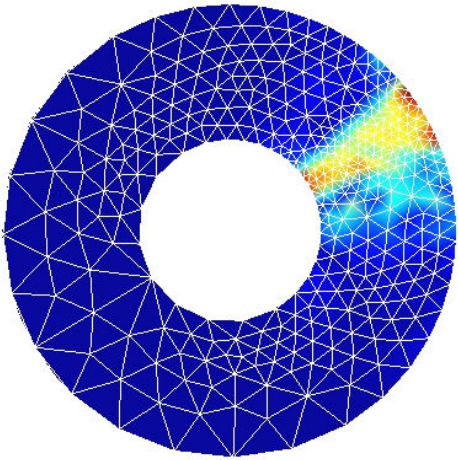

x 10^−3
2
1
0
−1
−2
−3
−4
−5
0
1
2
3
4
5
6
7
u(3) (mm)
Angle (rad)
Fine mesh
Coarse mexh

Figure 12



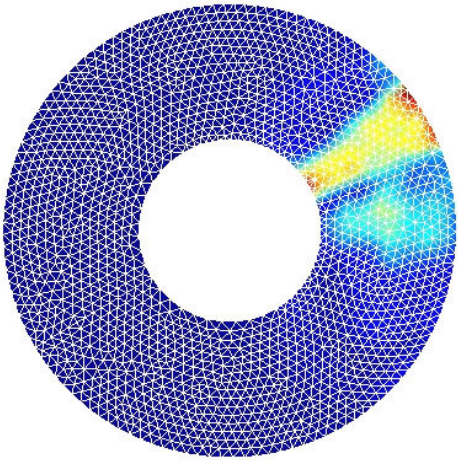
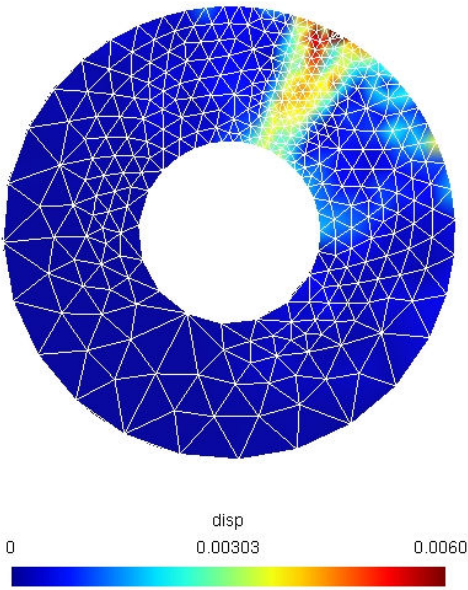


Figure 13



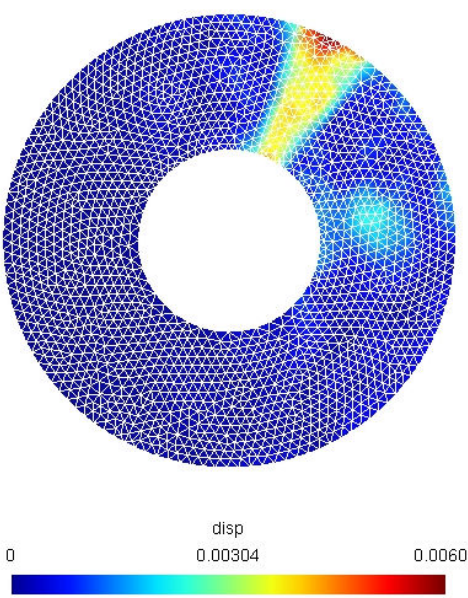


Figure 14