High-order computation of burning propellant surface and simulation of fluid flow in solid rocket chamber

D. Gueyffier, F.X. Roux, Y. Fabignon, G. Chaineray, N. Lupoglazoff, F. Vuillot, F. Alauzet

▶ **To cite this version:**

D. Gueyffier, F.X. Roux, Y. Fabignon, G. Chaineray, N. Lupoglazoff, et al.. High-order computation of burning propellant surface and simulation of fluid flow in solid rocket chamber. 50th AIAA/ASME/SAE/ASEE Joint Propulsion Conference, Jul 2014, Cleveland, United States. hal-01068675

HAL Id: hal-01068675
https://onera.hal.science/hal-01068675

Submitted on 26 Sep 2014

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

High-order computation of burning propellant surface and simulation of fluid flow in solid rocket chamber

D. Gueyffier[1], F.X. Roux[2], Y. Fabignon[3],
ONERA—The French Aerospace Lab, Palaiseau, France

G. Chaineray[4], N. Lupoglazoff[5], F.Vuillot[6]
ONERA—The French Aerospace Lab, Châtillon, France

and F. Alauzet[7]
INRIA Paris-Rocquencourt, 78153, Le Chesnay, France

In this paper, we present a numerical approach for predicting fluid flows in solid rocket motor (SRM) chambers. We use a novel high-order technique to track the burning grain surface. Spectral convergence toward the exact burning surface is achieved thanks to Fourier differentiation. In addition, we make use of a body-fitted mesh deforming with the burning surface and present a method to avoid manual remeshing. We describe several methods to deform the volume mesh and to keep good mesh element quality during the computation. We then couple the surface and volume approaches. The resulting coupled method is able to handle the formation of geometric singularities on the burning surface while keeping constant surface and volume mesh topology. This geometrical approach is integrated into a complex code for compressible, multi-species, turbulent flow simulations. Applications to the simulation of the internal flow in realistic solid rocket motors with complex grain geometry are then presented.

I. Introduction

Propellant grains currently in use in solid rocket motors (SRM) have all sorts of complex shapes: cylinders, stars, finocyl, anchors and multiple fins. The grain shape is designed to obtained constant or time dependent thrust during the rocket flight. For example star grains are used to maintain constant thrust during most of the rocket propulsion phase. Various techniques have been used in the past to predict the time evolution of the burning grain surface. A number of analytical methods have been developed to study simple 2-D geometries [1-3]. For more complex grain shapes or for 3-D geometries, CAD software has been used by some authors [4, 5] while others resolve a Hamilton-Jacobi problem for the time evolution of the burning surface [6-8]. The level set method was used in [9, 11-13] to predict grain burning. The level set method has also been used to simulate burning grain at small scale [10].
A complex coupled problem needs to be solved in order to compute the time dependent flow in solid rocket chambers while the grain is burning. This coupled approach can be used to study combustion instabilities and pressure oscillations in rocket chambers [14-17]. Other applications include optimal grain design, or computation of

[1] Scientist – Rocket Propulsion team – AIAA Member
[2] Team Leader – High Performance Computing team
[3] Team Leader – Rocket Propulsion team
[4] Scientist – Computational Fluid Dynamics team
[5] Scientist – Computational Fluid Dynamics team
[6] Team Leader – Computational Fluid Dynamics team
[7] Researcher – GAMMA3 Team

the multiphase flow in the chamber [18, 19]. For these applications, the method has to compute the flow in a time dependent domain and the method has to track the grain surface with a burning rate depending on local flow properties. This coupled problem is more difficult than only predicting the burning surface evolution.
In order to treat deformations of the computational domain, most codes make use of an ALE method [20]. But difficulties arise when the mesh undergoes dramatic deformations during the regression of star-shaped or finocyl grains. The most problematic deformations occur in regions where the grain surface is initially convex toward the gas. In such regions, a geometric singularity (caustic) forms in finite time on the grain surface. As shown on figure 1, if no special action is taken the neighboring mesh will become invalid. Our approach prevents such problems by moving the element vertices away from the singularity. In addition most CFD codes do not handle addition or deletion of mesh elements during computations. Our method ensures that the mesh keeps the same number of elements at all times, i.e. the mesh is always homeomorphic to the initial mesh.

The contributions of this paper are: (1) a novel spectral method to accurately track the burning grain surface and deform the surface mesh, (2) a method which prevents volume mesh elements to become invalid in regions where singularities appear on the grain surface, (3) a technique which constrains the volume mesh to keep the same topology during the entire burning phase, (4) this novel methodology can later be used to show the dynamic effect of grain regression on the internal flow. We expect flow simulations with dynamic grain regression to differ from static simulations (CFD in the chamber taking into account a constant burning surface) described for example in [15, 16]. In particular, this dynamic effect may have an influence on pressure oscillations.

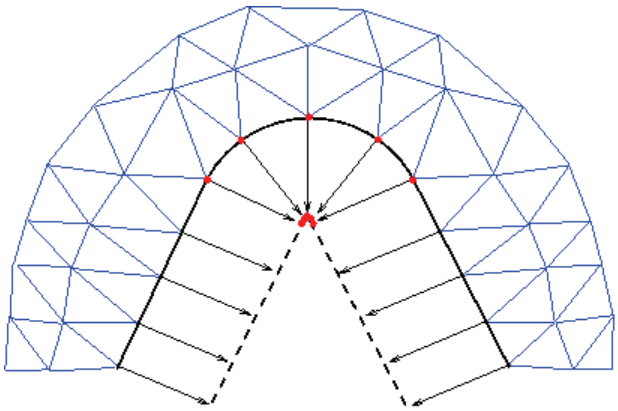

Figure 1: if the grain surface is convex toward the gas, a geometric singularity (caustic) appears in finite time and mesh vertices can become clustered. The body-fitted mesh can rapidly become invalid.

The paper is organized as follows: section II defines the problem under study. Section III describes the spectral algorithm for tracking the burning front surface and deforming the surface mesh. Tests and validations are also presented. Section IV details several volume mesh deformation algorithms which have been coupled with the surface tracking algorithm. Computations of the flow inside a realistic SRM chamber are presented in section V.

II. Problem definition

Eikonal equation. We assume that the grain surface burning velocity follows the classical Vieille or Saint Robert law [1, 21]

$$\mathbf{V} = A\,P^{\beta}\,\mathbf{n}\,,$$

with $\mathbf{n}$ the exterior normal to the grain surface and P the local static pressure at each surface location. A and β are parameters which are determined experimentally and depend on the propellant composition. In the following, we will note $\alpha = A\,P^{\beta}$. We suppose that the grain surface is defined parametrically by

$$\mathbf{S}(t) = \mathbf{x}(u,v,t)\,,$$

where $\mathbf{x}(u,v,t) \in \mathbf{R}^3$ is the position vector on the burning surface, t is time and (u,v) are parameters in the $\mathbf{R}^2$ plane. Then the position vector x is solution of the following nonlinear PDE

$$\frac{\partial \mathbf{x}}{\partial t} = \alpha \mathbf{n} = \alpha \frac{\mathbf{x}_u \otimes \mathbf{x}_v}{\left|\mathbf{x}_u \otimes \mathbf{x}_v\right|}, \quad (1)$$

where $\otimes$ is the cross-product in $\mathbf{R}^3$, $\mathbf{x}_u = \frac{\partial \mathbf{x}}{\partial u}$ and $\mathbf{x}_v = \frac{\partial \mathbf{x}}{\partial v}$.

A level set function $T(\mathbf{x})$ can also be introduced in order to define the surface implicitly. Let $T(\mathbf{x})$ be a function such that surfaces $\mathbf{x} = \mathbf{S}(t)$ are solution of equation

$$T(\mathbf{x}) = t.$$

Using these definitions and following [22-24], equation (1) can be expressed as

$$\left|\nabla T\right|^2 = \frac{1}{\alpha^2}. \quad (2)$$

Equation (2) is the eikonal equation, which is a particular type of Hamilton-Jacobi equation. This equation is well known for its applications in geometrical optics and for various wave problems (electromagnetic waves, detonation waves, seismic waves). It also applies to the resolution of the Schrödinger and Klein-Gordon equations and to the computation of distance functions. Solutions of the eikonal equation exhibit geometric singularities (caustics) and can become multi-valued [25-27], i.e. the propagating surface forms loops.

III. Spectral method for burning front surface and surface mesh deformation

Goals of the method. In this section we present a novel technique to very precisely track the burning surface and to deform a surface mesh conforming to this propagating front. Our approach offers the following features:

i- the solution converges spectrally toward the exact solution of (1). We can therefore obtain a precise solution using few degrees of freedom,

ii- with this technique, curvilinear distances along the surface are computed precisely. Knowing the distance to the geometric singularities, one can move mesh vertices away from the caustics and prevent mesh elements from becoming singular. The surface mesh topology remains the same throughout the computation.

Spectral differentiation. The spectral method was inspired by prior work on Wilmore flows for surfaces which are topologically equivalent to a sphere or a torus [28-30]. Unlike eikonal flows, Wilmore flows tend to smooth out geometric singularities. We will see in the next paragraph how our approach is able to treat the formation of geometric singularities (caustics).

We assume that the surface is represented by a set of $N \times M$ markers or sampling points defined by

$$\mathbf{x}_{i,j} = \mathbf{x}(\theta_i, z_j)\ i \in \{1,..,N\}, j \in \{1,..,M\},$$

where the parameters are equispaced

$$\theta_i = 2\pi\ (\mathrm{i}-1)/\mathrm{N},\ i \in \{1,..,N\}, \qquad z_j = 2\pi\ (\mathrm{j}-1)/\mathrm{M},\ j \in \{1,..,M\}.$$

In order to calculate the local surface properties at each sampling point, we make use of spectral differentiation in Fourier space. To do so, we suppose that the surface can be represented by a mapping between $C = [0, 2\pi] \times [0, L]$ and $\mathbf{R}^3$ such that

$$M : C \to R^3$$

$$(\theta, z) \to \mathbf{x} = \sum_{k=-N/2}^{N/2-1} \sum_{l=-M/2}^{M/2-1} \hat{\mathbf{x}}(k,l) \exp(-ik\theta) \exp(-il\,2\pi\, z/L)$$

where M is the trigonometric interpolation polynomial between the $\mathbf{x}_{i,j}$ sampling points. We obtain the first tangent vector by differentiating along θ

$$\mathbf{x}_\theta = \sum_{k=-N/2}^{N/2-1} \sum_{l=-M/2}^{M/2-1} -ik\,\hat{\mathbf{x}}(k,l) \exp(-ik\theta) \exp(-il\,2\pi\, z/L),\ \theta \in [0, 2\pi]$$

In practice, we use an FFT to convert from $\mathbf{x}$ to $\hat{\mathbf{x}}$ and an IFFT (inverse FFT) to convert from $\hat{\mathbf{x}}$ to $\mathbf{x}$. The

$\mathbf{x}_\theta$ derivative is approximated by

$$\mathbf{x}_\theta \approx IFFT(-i\,c\,FFT(\mathbf{x})),\ c=[-\frac{N}{2},-\frac{N}{2}+1,...,\frac{N}{2}+1], \quad (3)$$

which converges spectrally with N toward the exact derivative.

Calculation of geometric quantities. Curvilinear distances in the two parametric directions are defined by

$$s(\theta)=\int_0^\theta |\mathbf{x}_\theta|\,d\theta\,, v(z)=\int_0^z |\mathbf{x}_z|\,dz\,. \quad (4)$$

The components of the 1st fundamental form (metric tensor)

$$E=\mathbf{x}_\theta.\mathbf{x}_\theta,\ G=\mathbf{x}_z.\mathbf{x}_z,\ F=\mathbf{x}_\theta.\mathbf{x}_z=\mathbf{x}_z.\mathbf{x}_\theta$$

are used to obtain the area element

$$\mathrm{dA}=\sqrt{\mathrm{EG}-\mathrm{F}^2}\,d\theta\,dz=|\mathbf{x}_\theta\otimes\mathbf{x}_z|\,d\theta\,dz\,.$$

Gaussian curvature at each point of the surface is obtained using the expression

$$\kappa=\frac{PR-Q^2}{EG-F^2},$$

where P, Q and R are the components of the second fundamental form. The unit outward normal reads

$$\mathbf{n}=\frac{\mathbf{x}_\theta\otimes\mathbf{x}_z}{|\mathbf{x}_\theta\otimes\mathbf{x}_z|}\,. \quad (5)$$

Then the local propagation velocity of each point of the surface follows equation (1), i.e.

$$\dot{\mathbf{x}}=\alpha\,\mathbf{n}\,,$$

The surface normal is computed spectrally using (3) and (5). Markers are transported in a lagrangian manner using equation (1) and using simple forward Euler explicit time stepping.

Although the markers do not stay uniformly spaced during their movement, we can still use a uniform FFT to perform the differentiation. In fact, in 2-D we make use of the Lagrange variable *u=θ(t=0)* which is uniformly spaced. We write (see also [28, 31]) that the tangent to the contour is the derivative of the position vector and using the chain rule

$$\mathbf{x}_\mathbf{s}=\mathbf{x}_\theta/s_\theta\,.$$

We obtain $\mathbf{x}_\theta$ and s_θ using the FFT differentiation described above.

Treatment of caustics. In the following, we detail how the method is able to naturally handle the formation of geometric singularities. We consider an 8-branch star grain in 2-D defined by

$$\mathbf{x}=(r(\theta)\cos(\theta),r(\theta)\sin(\theta)),\ r(\theta)=1+0.3\cos(8\theta)\,,$$

and let the front propagate in the local normal direction at constant velocity $\alpha=1$. A caustic is formed in finite time and the contour becomes multi-valued as predicted in [25-27], see figure 2. The curve remains C^0 and its derivative remains continuous (the curve is at least C^1). Hence no Gibbs phenomenon occurs and the algorithm remains accurate and stable. We simply avoid the geometric singularity by letting the solution become multivalued and letting the surface form a loop. Determining the caustic position is performed by detecting where the contour self-intersects.

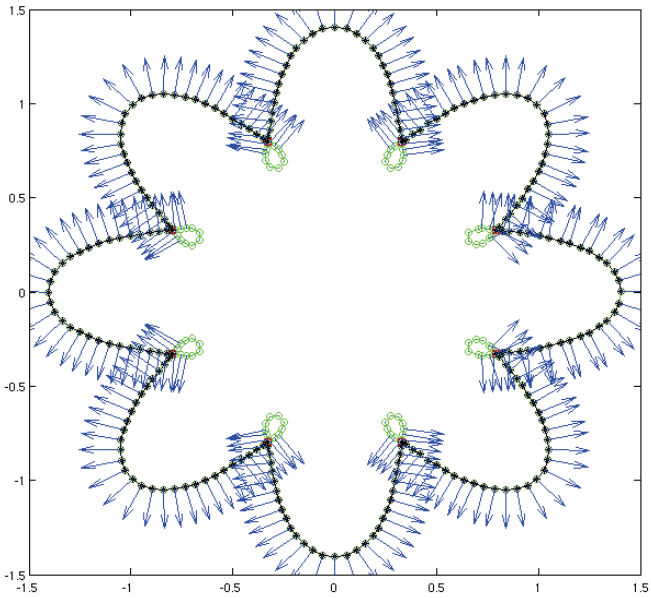

Figure 2: regression of an 8-branch star. The contour forms caustics and the solution becomes multi-valued. The green curve shows the interior part of the contour.

Testing spectral convergence. In order to test our approach, we compare solutions obtained using our method with solutions obtained by Kimmel [32] and Sethian [24] for eikonal flows. These authors have predicted that for constant and uniform velocity α along the normal, the perimeter L of a 2-D contour evolves according to the following law

$$L_t = 2\pi\alpha\,,$$

whose solution is

$$L(t) = L(0) + 2\pi\alpha t\,. \quad (6)$$

The curvature satifies the Riccati equation [31]

$$\kappa_t = -\alpha\kappa^2\,,$$

which admits the following solution

$$\kappa(\mathbf{x},t) = \kappa(\mathbf{x},0)/(1+\alpha\kappa(\mathbf{x},0)t)\,, \quad (7)$$

with $\kappa(\mathbf{x},t)$ the curvature at each location on the curve at time t and $\kappa(\mathbf{x},0)$ the curvature at time $t = 0$.

For the 8-branch star described in the previous paragraph, we have verified that the contour perimeter follows expression (6), see figure 3. This test is ran until the formation of caustics on the contour.

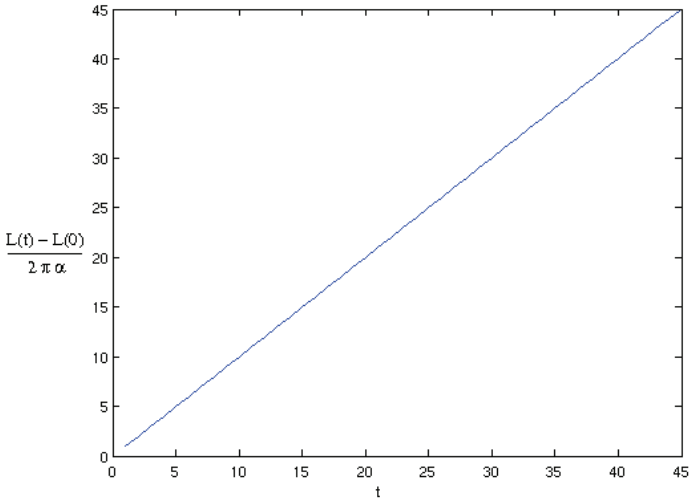

Figure 3: time evolution of the scaled contour perimeter (L(t)-L(0))/(2απ) for #DoF=320. Expression (6) is verified in our simulations.

We also verify that the curvature converges toward the solution (6) for curvature, see figure 4. This figure shows the L_2 error between our curvature calculation and expression (6).
The L_2 error on curvature is defined as

$$\sqrt{\sum_{i=1}^{\#DoF}\left|\frac{\kappa(\mathbf{x}_i,t_1)(1+\alpha\,\kappa(\mathbf{x}_i,0)t_1)}{\kappa(\mathbf{x}_i,0)}-1\right|^2}\ , \quad (8)$$

at time $t=t_1=8$ before caustic formation. #DoF is the number of degrees or freedom (=N in 2-D, = N x M in 3-D). For small and medium #DoF, convergence is indeed spectral. The error saturates around 10^{-10} for large #DoF (=640) because of the truncation error in the FFT-differentiation (see [29, 30]).

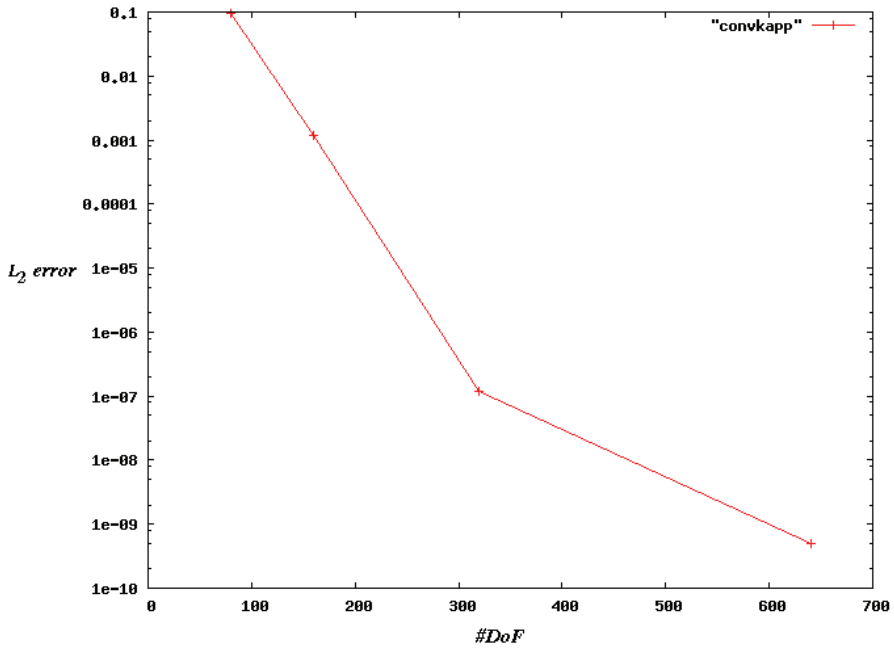

Figure 4: L_2 error between curvature predicted by our method and solution (7) as a function of #DoF.

We have also studied the convergence toward a unique profile after the caustic formation (t=50) for increasing #DoF (#DoF ranging from 80 to 640), see figures 5 and 6.

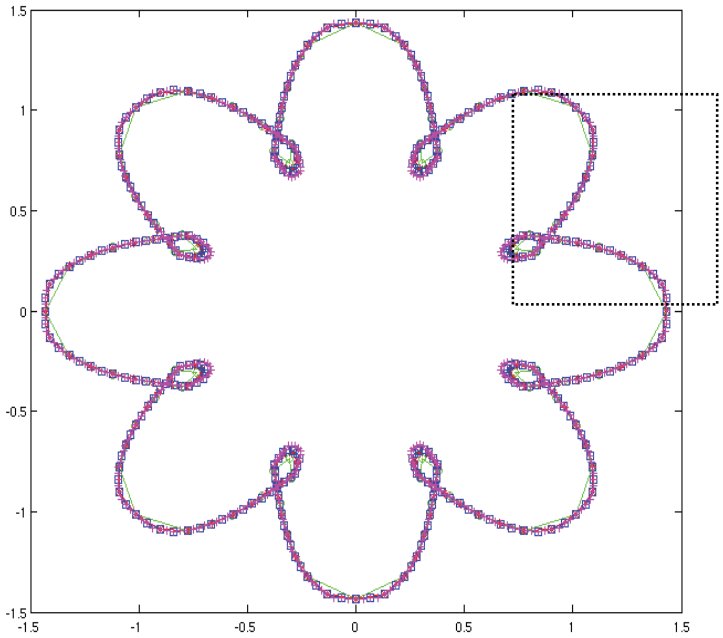

Figure 5: convergence toward unique contour for increasing number of markers (#DoF = 80, 160, 320, and 640). Time is t=50 after the caustic formation.

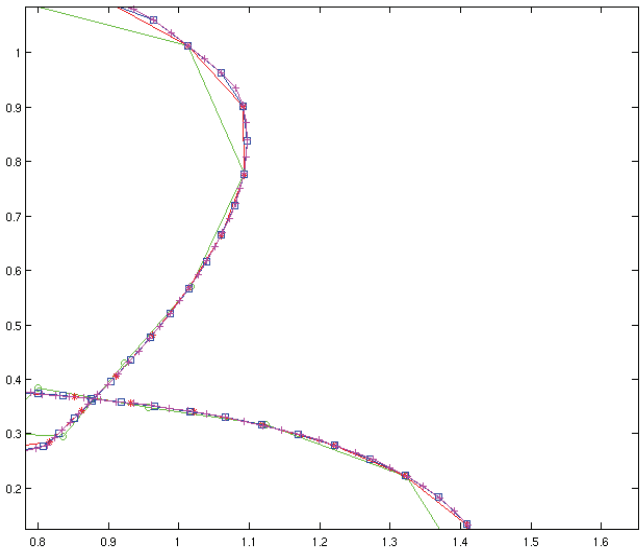

Figure 6: zoom on the upper right corner of figure 5. We obtain convergence toward a unique contour with increasing #DoF.

Surface mesh. After detecting self-intersections of the contour, it is easy to determine which markers are inside the multi-valued zone (loop) and which ones are on the exterior of the contour. We can then decide to locate the surface mesh vertices between the set of exterior marker points, including intersection points. To do so we dispose the surface mesh vertices at constant curvilinear distance from each other, see figure 7. We ensure that the surface mesh keeps the same topology over time by disposing a constant number (=N=*#DoF*) of surface mesh vertices at equispaced locations along the exterior contour.

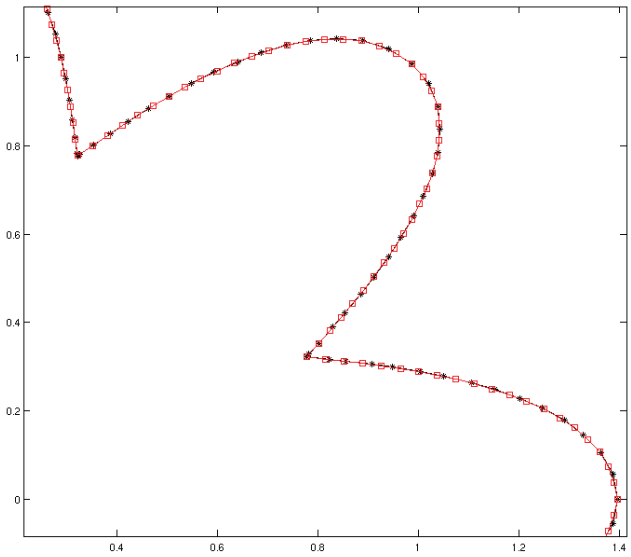

Figure 7: surface mesh vertices (in red) at equispaced locations along the exterior contour (in black).

IV. Volume mesh deformation

In the following, we present techniques to compute the initial mesh and to obtain time dependent deformations of the mesh enclosed by the burning grain contour.
In order for the volume mesh to retain the same topology as the initial mesh, we seek the following properties for the mesh deformation method:

i- the volume mesh vertices along the boundary must be the vertices of the surface mesh, the surface mesh remaining homeomorphic to the initial surface mesh,
ii- the volume mesh elements must keep good quality.

While the methods presented below satisfy property i) by construction, there exist several approaches to achieve property ii). A review of these methods can be found in [33].

Initial volume mesh. In order to compute the initial mesh, we have used MIT's DistMesh code [34]. This method takes as input boundary vertices locations and returns a 2-D triangulation with good element quality. Definition of element quality can be found in [34]. DistMesh first computes a Delaunay triangulation of the domain enclosed by the contour and then resolves an elasticity problem to adjust inner vertices positions while boundary vertices remain fixed. With DistMesh, we obtain excellent quality for the vast majority of elements and only a few elements reach average quality. In addition, no element has poor quality. The resulting volume mesh is shown on figure 8.

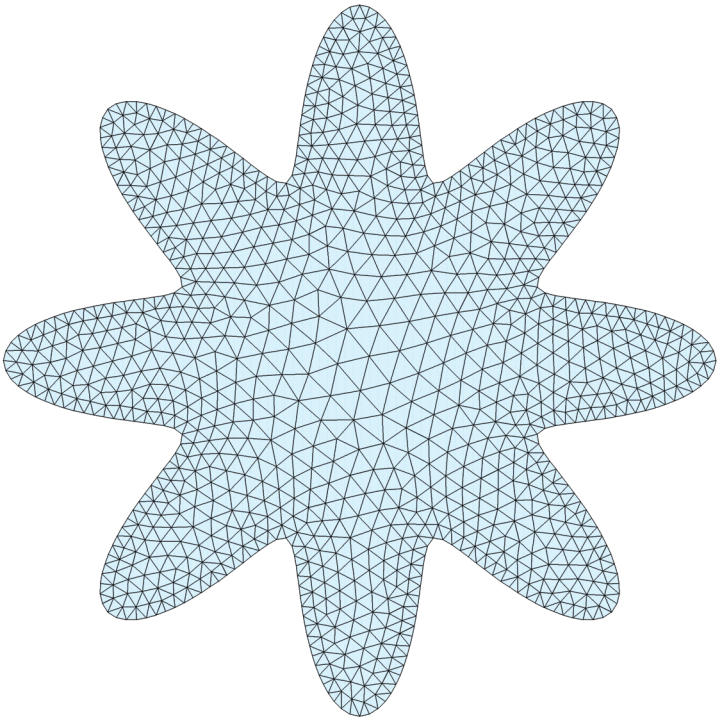

Figure 8: initial volume mesh obtained with DistMesh [34]. Quality is above 0.94 for 98% of the mesh elements.

Deformation using propagation functions. The first technique makes use of propagation functions to compute the displacement of each inner vertex of the volume mesh. Each inner vertex displacement depends on the displacement of boundary vertices through the expression

$$u(i) = \frac{1}{p}\sum_{k=1}^{p} F(i,k)u_S(k),$$

with $u(i)$ the displacement of the inner vertex whose index is i, p is the number of boundary vertices, $u_S(k)$ is the displacement of the boundary vertex whose index is k and $F(i,k)$ is the propagation function depending on the distance $d(i,k)$ between vertex i and vertex k. The displacement of boundary vertices $u_S(k), k = 1..p$ is obtained by resolving equation (1) using the method presented in section III. The propagation function is defined by

$$\begin{cases} F(i,k) = 1 \text{ for } 0 < d(i,k) < d_1 \\ F(i,k) = d_1/(d(i,k))^{\eta} \text{ for } d_1 < d(i,k) < d_2 \\ F(i,k) = 0 \text{ for } d_2 < d(i,k) \end{cases},$$

where η, d_1, d_2 are parameters which can be adapted to achieve good results. Figure 9 shows deformations of the volume mesh obtained using this approach. The 2-D mesh has been extruded to obtain a 3-D mesh with only one layer of cells.

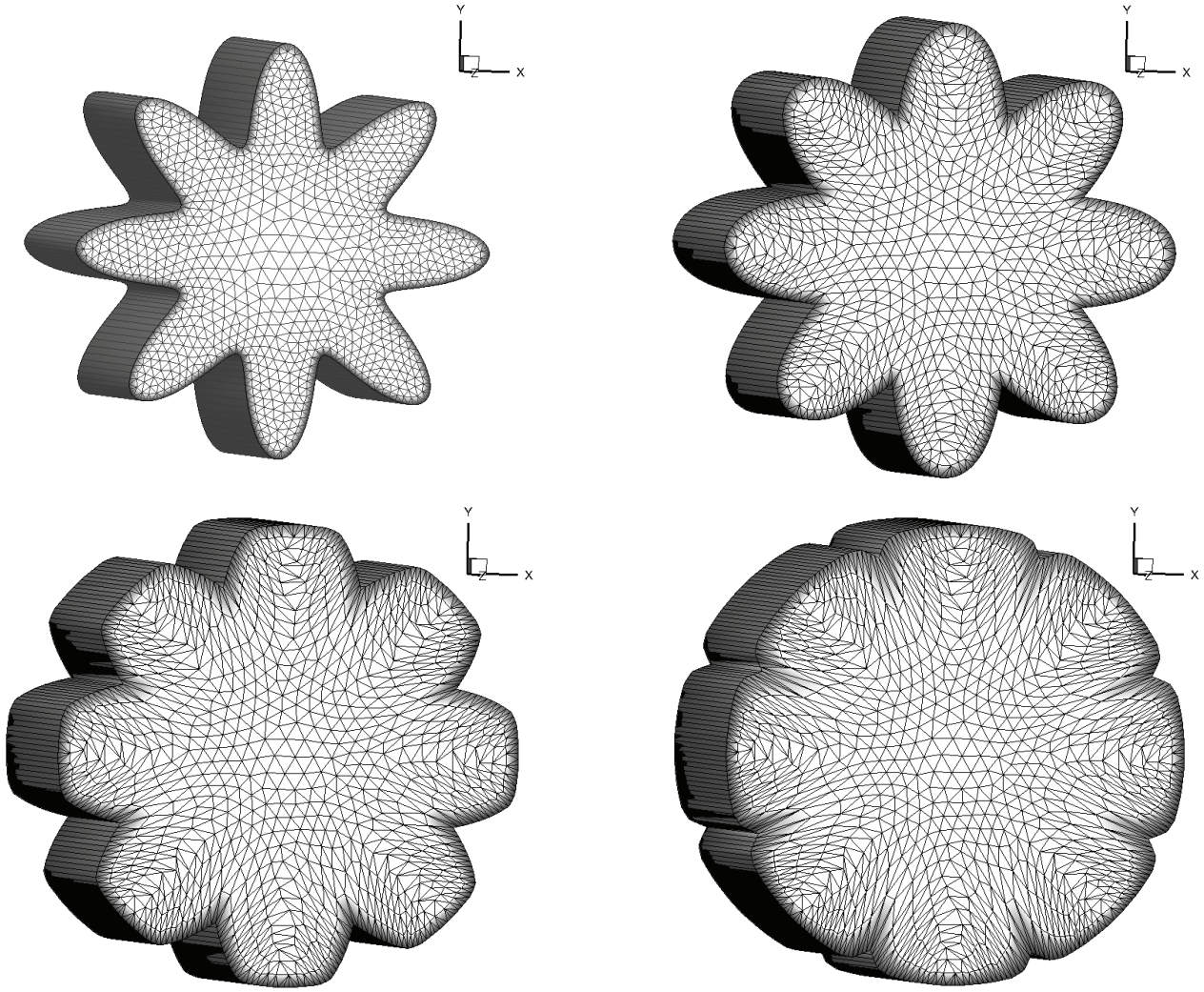

Figure 9: volume mesh deformations using propagation functions

Elastic deformations. With this second approach, mesh vertices displacements are obtained by resolving an elasticity problem using the FETI [35] finite element solver. We consider that the mesh is a structure submitted to displacements of the boundary. We calculate the response of this elastic structure to imposed boundary displacements governed by equation (1). Elementary stiffness matrices have variable coefficients. We adapt the coefficients to increase or decrease the rigidity of each mesh element in order to avoid compressing elements and deteriorating their quality. We use coefficients depending either on the mesh element area or on the eccentricity of the element. Element eccentricity is defined as the radius of the circumscribed circle divided by the radius of the inscribed circle.

In addition, we measure the distance of each element to the nearest boundary. We increase the rigidity of elements that are near the boundary. Results are shown on figure 10. On this figure, we observe that making the elements very rigid near boundaries also freezes them in elongated shapes

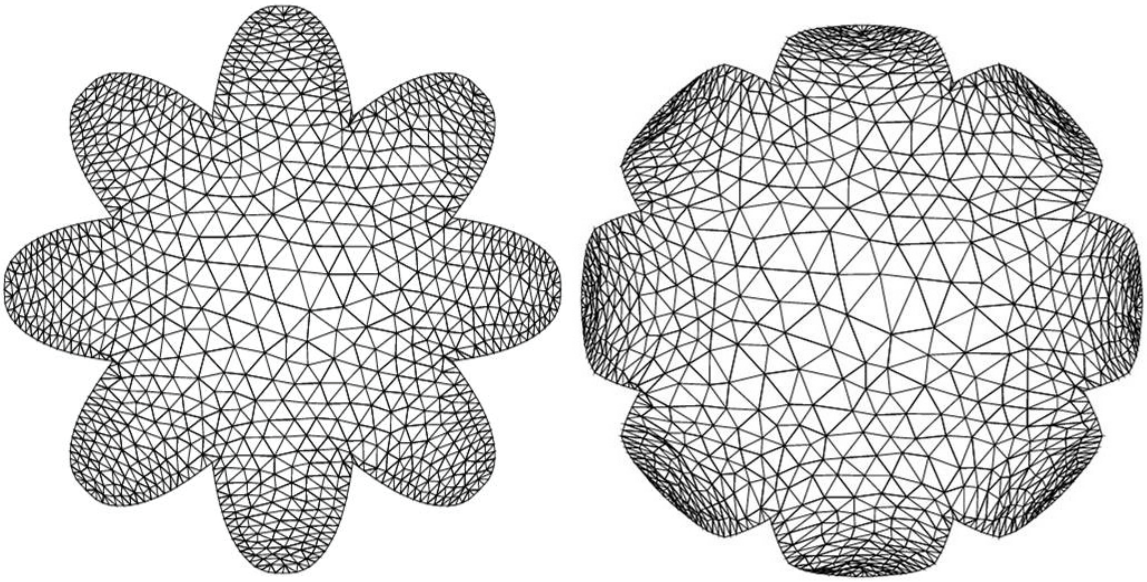

Figure 10: elastic deformations of volume mesh using finite element method.

Improving the elastic mesh strategy. In this paragraph, we briefly describe improvements to the linear elastic PDE approach. This work relies on a new body-fitted moving mesh technique [36] which has been proposed to handle efficiently and robustly large displacement of complex geometries inside unstructured meshes. This method is based on an enhanced PDE-based mesh deformation algorithm coupled with advanced mesh optimization using vertex displacements and mesh-connectivity changes.

The robustness and quality of the resulting deformed mesh is improved by

- modifying the elasticity system to stiffen small elements,
- supplying vertices with a high-order trajectory, i.e., vertices follow a curved path with a speed and an acceleration,
- rigidifying some inner regions by associating them with a rigid body.

Depending on the considered case, the efficiency can be improved by

- solving the elasticity system on only a part of the domain at a given topologic or algebraic distance of the moving object,
- solving the elasticity on a dedicated mesh - which is usually uniform and coarser than the computational mesh - and interpolating the result on the computational mesh,
- reducing the number of mesh deformation resolution which can be done because the mesh quality is preserved thanks to mesh optimization and vertices high-order trajectories.

To improve the whole algorithm and avoid global remeshing when the mesh is too distorted, the mesh deformation is coupled with advanced mesh optimization to maintain a good mesh quality during the entire process. To this end, vertices location is improved using vertex smoothing (re-location).

V. Computation of fluid flow in SRM chamber

Numerical Model. The mesh deformation algorithms presented in sections III and IV are integrated into CEDRE [37], a numerical tool designed to simulate supersonic reactive gas flows [38]. This tool has been used to simulate a number of problems involving liquid and solid propulsion systems [39-41].

Spatial discretization of the conservation equations is performed using second order finite volume approach for general unstructured grids.

Numerical Euler fluxes are computed using an upwind ODFI scheme which is similar to Roe's flux difference splitting. We use a Superbee slope limiter which is less dissipative than the Van Leer limiter. The solver performs calculations on unstructured meshes with cells of arbitrary geometry and we make use of an ALE approach [20] to compose the movement of mesh vertices with the eulerian definition of quantities and fluxes. Time integration is

first order implicit (backward Euler) and the nonlinear system of equations is linearized using Newton's method and the resulting jacobian is inverted using GMRES.

Preliminary calculations. We present a preliminary test with the CEDRE code using a cylindrical grain geometry. When the grain burns, the cylinder radius increases and the mesh deforms inside this region while in the nozzle region the mesh stays fixed. In the buffer zone between these two regions the mesh deforms with the magnitude of deformations decreasing to zero when approaching the nozzle region. On the grain boundary, we impose a gas flux entering the domain. Figure 11 shows the pressure field in the chamber at two separate instants during grain regression.

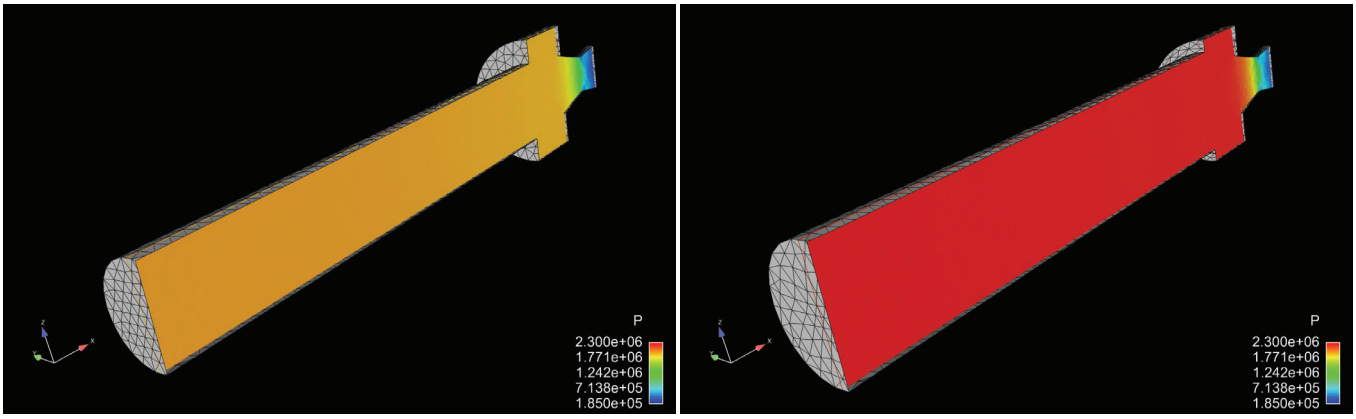

Figure 11: Pressure field in the SRM chamber obtained with CEDRE during cylindrical grain regression.

Test case with star shaped grain. We choose a similar geometry, although not precisely the same, to the one of a subscale LP9 solid rocket motor [42].
A rocket performance calculation is performed with the star shape shown on figure 8, in order to obtain a constant chamber pressure of about 70 MPa during the whole duration of the simulation.
We choose a JD94.04 [43] butalite propellant (without aluminum). For this propellant and using the targeted chamber pressure, the combustion gases temperature is 2299 *K*, the gases molar mass is 0.0222 *kg/mol*, and the specific capacity is 43.39 *J/K/mol*. The mean mass flux of combustion gases per unit grain surface is 17 $kg/m^2/s$. The resulting geometry for the star, chamber and nozzle is shown in table 1.

Nozzle throat radius	13.2 *mm*
Nozzle exit radius	38.2 *mm* (adapted nozzle exit)
Grain length	0.6 *m*
Initial surface of star grain	0.16 m^2
Chamber radius	0.036 *m*

Table 1 : rocket motor geometry

The initial computational mesh for this geometry is shown on figure 12. As in the preliminary test, the mesh within the grain region is assembled with a mesh for the cylindrical buffer region. The mesh in these two regions is made of triangular prisms. Two cross sections of the mesh are shown on figure 13. The first cross section (fig. 13, left) is located in the grain region while the second cross section (fig. 13, right) is located in the buffer region. The buffer region mesh is assembled with a fixed unstructured tetrahedral mesh for the nozzle region.

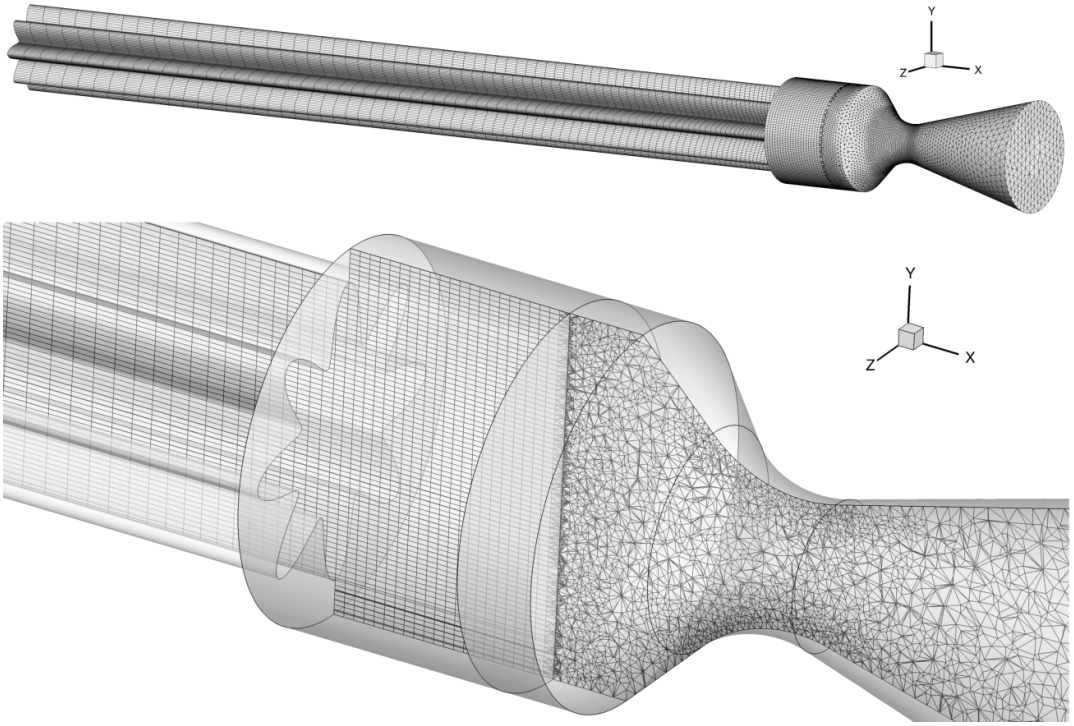

Figure 12: initial 3-D mesh for SRM test case

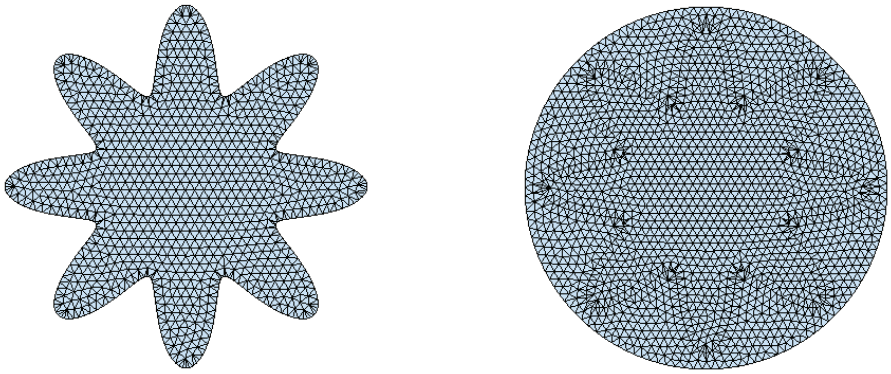

Figure 13 : two cross sections within the grain region (left) and within the buffer region (right)

In order to verify our design and our performance calculations, we perform a simulation keeping the grain surface fixed. The combustion gases mass flux through the star shape grain surface is 17 $kg/m^2/s$. Our simulations, see figure 14, show that

- the nozzle is adapted (pressure is 1MPa at the nozzle exit)
- the flow through the throat is sonic (Mach 1)
- the average chamber pressure is about 70 MPa

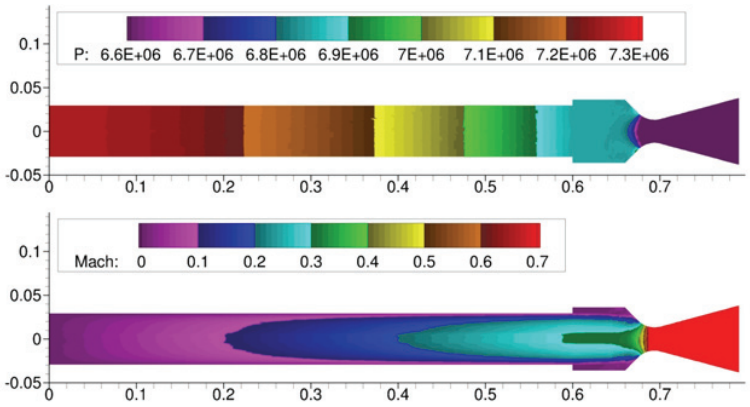

Figure 14 : simulations with fixed grain surface

We then make use of the surface and volume mesh deformation algorithms presented in this paper to let the star grain surface regress while computing the fluid flow with CEDRE. As in the preliminary test, the volume mesh deforms in the grain region, while in the buffer region the mesh deformation gradually decreases to zero until we reach the fixed mesh region containing the nozzle. Figure 15 shows three steps of the mesh deformation obtained within CEDRE.

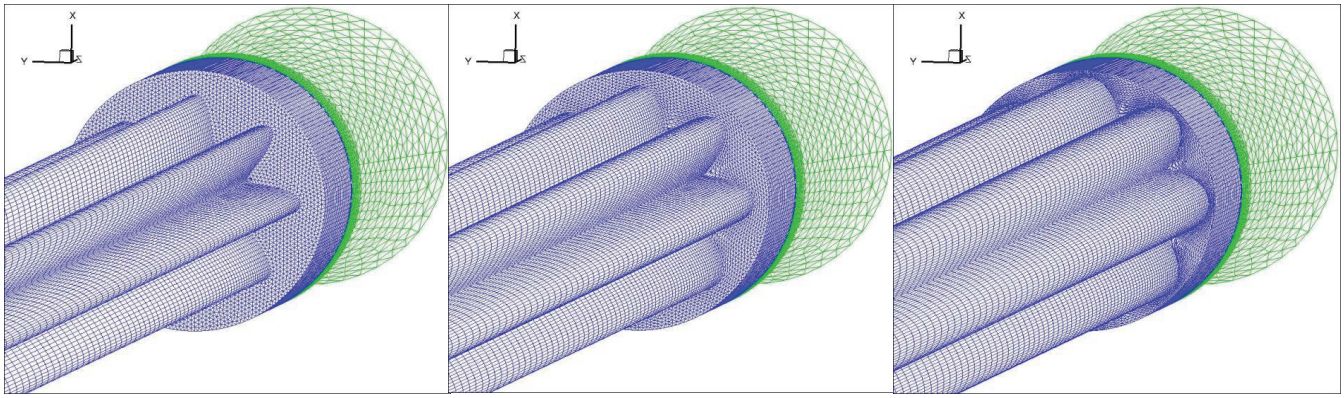

Figure 15: time evolution of mesh deformation

Figure 16 shows three steps of the time evolution of the pressure field in the chamber. It should be noted however that, in order to speed up the computation, we have increased the grain surface regression velocity to a value which is different from the actual burning rate. A physically realistic simulation will be presented in a future paper.

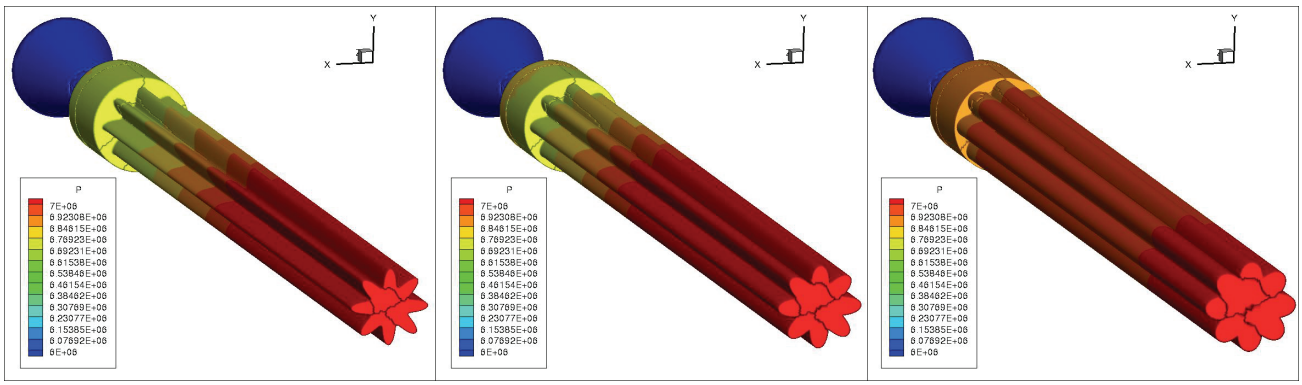

Figure 16 : time evolution of the pressure field

VI. Conclusion

We have designed a set of algorithms to accurately track the burning grain front and to deform the internal chamber mesh. A novel, spectrally accurate description of the burning surface has been proposed and validations with respect to prior theoretical work have been presented. We have coupled these geometrical techniques with a complex code for compressible, multi-species reactive gas flows. With this coupled approach, we are now able to simulate the dynamic response of the internal flow to the deformation of the grain surface. In the near future, we will perform simulations in order to quantify this dynamical effect on internal solid rocket motor flows. We are also working on a fully 3-D version of the front tracking algorithm and of the volume mesh deformation method in order to treat more complex grain geometries like 3-D finocyl grains. In addition, we plan to implement dynamic mesh topology changes to improve element quality during mesh deformation.

Acknowledgments

D.G. would like to thank B. Courbet and L. Matuszewski for their help on the design of meshing tools and with mesh format conversion. We would also like to thank J. Hijlkema, J.L. Estivalezes and G. Casalis for their input on the grain surface deformation algorithm. This research is supported by an ONERA grant (ONERA/MFE/ RGF/PR/2013/CYPRES).

References

[1]M. Barrère, Rocket Propulsion, Elsevier, Amsterdam, 1960

[2]H. Ray, R. Jenkins, J. Burkhalter and W. Foster, A review of analytical methods for solid rocket motor grain analysis, AIAA Paper 4506, 2003

[3]H. Ray, R. Jenkins, J. Burkhalter and W. Foster, Analytical methods for predicting grain regression in tactical solid-rocket motors, *Journal of Spacecraft and Rockets*, Vol. 41, No. 4, pp. 689-693, 2004

[4]A.Kamran and L. Guozhu. Design and optimization of 3D radial slot grain configuration. *Chinese Journal of Aeronautics* 23.4, pp. 409-414, 2010

[5]A. Kamran, L. Guozhu, A.F. Rafique and Q. Zeeshan, ±3-Sigma based design optimization of 3D Finocyl grain, *Aerospace Science and Technology*, Volume 26, Issue 1, April–May 2013, pp. 29-37

[6]M.A. Willcox et al., Solid propellant grain design and burnback simulation using a minimum distance function, *Journal of propulsion and power* 23.2, pp. 465-475, 2007

[7]F. Dauch and D. Ribereau, A Software for SRM Grain Design and Internal Ballistics Evaluation, PIBAL, AIAA paper 2002-4299, 2002

[8]S. Augoula and R. Abgrall, High order numerical discretization for Hamilton–Jacobi equations on triangular meshes, *Journal of Scientific Computing* 15, no. 2, pp. 197-229, 2000

[9]J.S. Sachdev, C. P. T. Groth, and J. J. Gottlieb, Parallel AMR scheme for turbulent multi-phase rocket motor core flows, *17th AIAA Computational Fluid Dynamics Conference*. 2005

[10]X. Wang, T. L. Jackson, and L. Massa, Numerical simulation of heterogeneous propellant combustion by a level set method, *Combustion Theory and Modeling* 8, no. 2, pp. 227-254, 2004

[11]Cavallini, E., Modeling and Numerical Simulation of Solid Rocket Motors Internal Ballistics, PhD. Dissertation, University of Rome La Sapienza, 2010

[12]Yildirim, C. and Aksel, M., Numerical Simulation of the Grain Burnback in Solid Propellant Rocket Motor, *41th AIAA/ASME/SAE/SEE Joint Propulsion Conference & Exhibit*, 2005, 10-13 July, Tucson, Arizona

[13]Qin F., Guoqiang, H., Peijin, L. and Jiang. L., Algorithm Study on Burning Surface Calculation of Solid Rocket Motor with Complicated Grain Based on Level Set Methods, *42th AIAA/ASME/SAE/SEE Joint Propulsion Conference & Exhibit*, 2006, 9-12 July, Sacramento, California.

[14]F. Vuillot, Vortex-shedding phenomena in solid rocket motors, *J. Propulsion and Power*, Vol. 11, Iss. 4, pp. 626–639, 1995

[15]Y. Fabignon, J. Dupays, G. Avalon, F. Vuillot, N. Lupoglazoff, G. Casalis, M. Prévost, Instabilities and pressure oscillations in solid rocket motors, *Aerospace Science and Technology*, Vol. 7, Iss. 3, pp. 191-200, 2003

[16]N. Lupoglazoff, F. Vuillot, J. Dupays, and Y. Fabignon Numerical simulations of the unsteady flow inside segmented solid-propellant motors with burning aluminum particles, *40th AIAA Aerospace Sciences Meeting & Exhibit*. 14-17 January 2002, Reno, NV

[17]G. Boyer, G. Casalis, and J-L. Estivalèzes, Stability analysis and numerical simulation of simplified solid rocket motors, *Physics of Fluids*, Vol. 25, Issue 8, 2013

[18]F. Doisneau, F. Laurent, A. Murrone, J. Dupays, and M. Massot. Eulerian multi-fluid models for the simulation of dynamics and coalescence of particles in solid propellant combustion, *Journal of Computational Physics*, 2012

[19]D. Gueyffier, S. Zaleski, Finger formation during droplet impact on a liquid film, *Comptes Rendus de l'Académie des Sciences Series IIB Mechanics Physics Astronomy*, Volume 326, Number 12, pp. 839-844, 1998

[20]C. W. Hirt, A. A. Amsden, and J. L. Cook. An arbitrary Lagrangian–Eulerian computing method for all flow speeds, *Journal of Computational Physics* 135, no. 2, pp. 203-216, 1997

[21]G.P. Sutton, Rocket Propulsion Elements, Wiley, New York, NY, USA, 6th edition, 1992
[22]P. Covello and G. Rodrigue, A generalized front marching algorithm for the solution of the eikonal equation, *Journal of Computational and Applied Mathematics*, Volume 156, Issue 2, 15 July 2003, Pages 371-388
[23]Osher, S. and Sethian, J.A., Fronts propagating with curvature-dependent speed: algorithms based on Hamilton-Jacobi formulations, *Journal of computational physics* 79.1 (1988): 12-49
[24]Sethian, J.A., Level set methods and fast marching methods: evolving interfaces in computational geometry, fluid mechanics, computer vision, and materials science, Vol. 3. Cambridge university press, 1999
[25]B. Engquist, O. Runborg, A.-K. Tornberg, High-frequency Wave Propagation by the Segment Projection Method, *Journal of Computational Physics*, Volume 178, 2, May 20, 2002, pages 373—390
[26]J.-D. Benamou, Big Ray Tracing: Multivalued Travel Time Field Computation Using Viscosity Solutions of the Eikonal Equation, *Journal of Computational Physics*, Volume 128, Issue 2, 15 October 1996, Pages 463-474
[27]H. Yu-Sing, and S. Leung. A Cell Based Particle Method for Modeling Dynamic Interfaces. Report U. California UCLA, 2013
[28]S. K. Veerapaneni, D. Gueyffier, D. Zorin, G. Biros, A boundary integral method for simulating the dynamics of inextensible vesicles suspended in a viscous fluid in 2D, *Journal of Computational Physics*, Volume 228, Issue 7, 20 April 2009, Pages 2334-2353
[29]S. K. Veerapaneni, D. Gueyffier, G. Biros, D. Zorin, A numerical method for simulating the dynamics of 3D axisymmetric vesicles suspended in viscous flows, *Journal of Computational Physics*, Volume 228, Issue 19, 20 October 2009, Pages 7233-7249
[30]S. K. Veerapaneni, A. Rahimian, G. Biros, D. Zorin, A fast algorithm for simulating vesicle flows in three dimensions, *Journal of Computational Physics*, Volume 230, Issue 14, 20 June 2011, Pages 5610-5634
[31]T. Y. Hou, J. S. Lowengrub, and M. J. Shelley Removing the Stiffness from Interfacial Flows with Surface Tension, *Journal of Computational Physics* 114, 312-338 (1994).
[32]Kimmel, Ron. Numerical geometry of images: Theory, algorithms, and applications, Springer, 2004.
[33]F. Alauzet, Efficient Moving Mesh Technique Using Generalized Swapping, Proceedings of the 21st International Meshing Roundtable, pp. 17-37, Springer Berlin Heidelberg, 2013
[34]P.-O. Persson, G. Strang, A Simple Mesh Generator in MATLAB. SIAM Review, Vol. 46 (2), pp. 329-345, June 2004
[35]C. Farhat and F. X. Roux, A method of finite element tearing and interconnecting and its parallel solution algorithm, *Int. J. Numer. Meth. Engrg.* 32, pp. 1205-1227, 1991
[36]F. Alauzet. A changing-topology moving mesh technique for large displacement. *Eng. w. Comp*., 30(2):175{200, 2014.
[37]Refloch, A., Courbet, B., Murrone, A., Villedieu, P., Laurent, C., Gilbank, P., Troyes, J., Tesse, L., Chaineray, G., Dargaud, J. B., Quemerais, E., and Vuillot, F., The CEDRE Software, *Aerospace Lab Journal*, Vol. 2, No. 2, pp. 1–10, 2011
[38]Chevalier, P., Courbet, B., Dutoya, D., Klotz, P., Ruiz, E., Troyes, J., and Villedieu, P., CEDRE: Development and Validation of a Multiphysics Computational Software, 1st European Conference for Aerospace Sciences (EUCASS), Paper 2.04.03, Moscow, Russia, 2005
[39]Troyes, J., Dubois, I., Borie, V., and Boischot, A.,Multi-Phase Reactive Numerical Simulations of a Model Solid Rocket Motor Exhaust Jet, 42nd AIAA/ASME/SAE/ASEE Joint Propulsion Conference and Exhibit, AIAA Paper 2006-4414, July 2006
[40]Troyes, J. and Vuillot, F., Numerical Simulations of Model Solid Rocket Motor Ignition Overpressure Waves, 44th AIAA/ASME/SAE/ASEE Joint Propulsion Conference & Exhibit, AIAA Paper 2008-5133, 2008
[41]Gueyffier, D., Fromentin-Denoziere, B., Simon, J, Merlen, A., Giovangigli, V., Numerical Simulation of Ionized Rocket Plumes, *Journal of Thermophysics and Heat Transfer*, 28, 2, 218-225, 2014
[42]Hijlkema, J., Prévost, M., Casalis, G., On the importance of reduced scale Ariane 5 P230 solid rocket motor models in the comprehension and prevention of thrust oscillations. *CEAS Space Journal*, Springer, 2011
[43]Dupays, J., Contribution à l'étude du rôle de la phase condensée dans la stabilité d'un propulseur à propergol solide pour lanceur spatial. PhD. Dissertation, 1997