Alma Mater Studiorum Università di Bologna
Archivio istituzionale della ricerca

Theoretical Study on the Influence of Debondings on Solid Rocket Motor Performance

(Article begins on next page)

12 September 2024

Theoretical study on the Influence of Debondings on Solid Rocket Motor Performance

Fabrizio Ponti

Full Professor, Department DIN, via Fontanelle 40, fabrizio.ponti@unibo.it

Stefano Mini

Research Fellow, Department DIN, via Fontanelle 40, stefano.mini3@unibo.it

Luca Fadigati

Ph.D. Candidate, Department DIN, via Fontanelle 40, luca.fadigati2@unibo.it

Adriano Annovazzi

Senior Engineer, AVIO Space Propulsion Design Department, Rome, adriano.annovazzi@avio.com

Enrico Corti

Associate Professor, Department DIN, via Fontanelle 40, enrico.corti2@unibo.it

Davide Moro

Full Professor, Department DIN, via Fontanelle 40, davide.moro@unibo.it

Abstract

In solid rocket motors, propellant debondings are very dangerous since they could cause an increment of the burning surface area and anticipate the exposure of case-insulating thermal protection material. Therefore, when a debonding internal zone is discovered, it is of primary importance to guarantee that the solid rocket motor is still able to accomplish the mission within its operational requirements. From a numerical point of view, debondings cannot be evaluated in an analytical, closed form due to their variety of shapes. The present study is aimed at assessing the impact of debondings through the adoption of advanced computer graphic techniques like the offsetting of the solid rocket motor burning surface discretized as a dynamic three-dimensional triangular mesh. Mesh handling algorithms are discussed in detail. Numerical results are obtained through an in-house simulation software which has been developed based on the aforementioned methods.

Key Words

Thermal protection, propellant debonding, propellant combustion simulation

Nomenclature

Latin

a = burning rate experimental factor, $m/s\ Pa^n$

A_p = port area, m^2

A_{mesh} = initial burning surface area calculate by the mesh, m^2

A_{ref} = reference initial burning surface area, m^2

C_v = specific heat capacity at constant volume, $J/(Kg\ K)$

e_A = relative area error between the calculated area and the reference one

F_{anis_i} = anisotropy correction factor of the i-th node of the burning surface mesh

H_f = grain combustion products enthalpy per unit mass, $J/(Kg\ K)$

n = burning rate experimental exponent

p = combustion chamber gas pressure, Pa

p_{0_i} = combustion chamber pressure at burning surface mesh i-th node, Pa

P_b = propellant burning perimeter, m

N = number of web values considered in the comparison between the SPP and ROBOOST burning surface areas

r_{b_i} = burning rate belonging to the burning surface mesh i-th node, ms^{-1}

R = specific gas constant, $J/(Kg\ K)$

$S_{b_{ROBOOST}}$ = burning surface area evaluated by ROBOOST, m^2

$S_{b_{SPP}}$ = burning surface area evaluated by the SPP, m^2

t = time, s

t_k = generic time instant at the time step k, s

T = combustion chamber gas temperature, K

u = combustion chamber gas velocity, m/s

w_i = web value, m

x_c = case curvilinear coordinate, m

x_i^* = burning surface area normalized error

$\bar{x}_i^*$ = mean burning surface area normalized error

Greek

$\phi(\vec{x})$ = implicit surface

θ = azimuthal coordinate, rad

Δs_i = displacement of the burning surface mesh i-th node, m

Δt = simulation's time interval, s

Δx = 1D ballistic model cell width along the motor axis, m

ρ = combustion chamber gas density, Kg/m^3

ρ_{pr} = propellant density, Kg/m^3

Acronyms

CAD   = Computer-Aided Design

SRM   = Solid Rocket Motors

SPP   = Solid Performance Program

PIBAL   = Propulsion and Internal BALlistic software

ROBOOST = ROcket BOOst Simulation Tool

AABB   = Axis Aligned Bounding Box

Z9   = Zefiro 9

1   Introduction

The use of numerical tools is important for SRM (Solid Rocket Motors) design because it allows to reduce expensive full-scale tests. Their use is fundamental to predict the performance of a motor under nominal conditions and becomes essential when flaws like inclusions or debondings are detected. The probability that propellant debondings or cavities may form is higher for motors built through multiple castings, which solidify in a single grain. The manufacturing process for those motors usually consists of the following steps:

- Once the thermal protection and the motor case are assembled, the surface of the thermal protection is carefully cleaned and an adhesive material (liner) is distributed on it.

- After the application of the liner, and before its polymerization, the propellant is cast into the motor.

- Propellant and liner are cured together guaranteeing a good adhesion of the propellant on the thermal protection surface.

If the process fails, debonding may form, whose presence should be considered as an influencing factor in the performance evaluation. The debonding could be generated mainly in two ways:

1. Despite the thermal protections being cleaned, some pollutant, like grease, could contaminate their surface decreasing the adhesive strength between the liner and the thermal protection (Martin et al., 2017). Therefore, after the casting, the liner could detach itself from the thermal protection. This type of debonding is not dangerous because the propellant surface is inhibited by the liner and the debonding surface could not be ignited.

2. The adhesive interface between the propellant and the liner could be weak and, for this reason, the debonding could rise, especially in the highly stressed regions. This weakness is usually related to poor mechanical properties of the interface, but in some rare case it could be due to the pollutants on the liner surface.

The second type of debonding is the most dangerous because the propellant surface is not inhibited by the liner, and consequently the defect starts growing when reached by the burning surface. This occurrence will be the main focus of this work, since it may be the cause of motor failures or the source of significative deviations with respect to its nominal performance.

Usually, the adhesion failure is detected through ultrasonic inspection, X-ray radiography, or thermography (Kumar and Selvaray, 2011) (Chen et al., 2015) (Tiu et al., 2015). When the hot gases of combustion chamber reach the debonding inhibited surface, the combustion process propagates through it and therefore the amount of burning propellant surface increases rapidly. The thermal protection material, which coats the case, is designed to withstand a limited thermal energy flux for a limited time interval. At the debonding zone, it is exposed sooner to the hot gases produced by the combustion of the grain, therefore it could be completely consumed before the end of the combustion of the propellant and the case would be directly exposed to the combustion chamber flow. This could even result in the failure of the launch.

In the literature, there are many methods to model and track the regression of the burning surface. For example, SPP (Solid Performance Program) (Coats et al., 1987) (Dunn and Coats, 1997) (Coats et al., 1975), a software designed for predicting the evolution of solid rocket motors, consists of boolean geometry operations to combine cones, cylinders, spheres, tori, and prisms in order to create

the most common propellant configurations: finocyl design, tapered star design, tubular, rod, and tube. An analogous approach has been used by the Société Nationale des Poudres et des Explosifs to develop a propulsion and internal ballistics simulator called PIBAL (Propulsion and Internal BALlistic software) (Dauch and Ribéreau, 2002). Although it is effective in simulating a great variety of geometries, it is not able to evaluate the influence of grain flaws.

Besides that, in recent years a set level method has been developed in order to improve the handling of the burning surface. Such methods have been firstly introduced by Oscher and Sethian (Oscher and Sethian, 1988). The level set method represents the surface as the zero-level of an auxiliary function $\phi(\vec{x})$. A certain surface Γ is defined through the Equation (1):

$$\Gamma = \{\vec{x} \in \mathcal{R}^3 \mid \phi(\vec{x}) = 0\} \quad (1)$$

The level set function is usually the Euclidian distance to the interface (Gibou et al., 2018) and its application does not allow to take into account non-homogeneities and flaws in the propellant like inclusions and debonding.

The following study deals with an in-house internal ballistic simulation software namely ROBOOST (ROcket BOOst Simulation Tool). It relies on a dynamic three-dimensional triangular mesh discretization to represent the burning surface regression; this approach allows to simulate any abnormal burning surfaces that may arise when propellant defects are introduced into the simulation. The main advantage of the above-mentioned solution, with respect to the Boolean geometric approach used by SPP, is the possibility to reproduce generic CAD-modeled geometries with only a little penalty due to the accuracy of the finite mesh discretization, and to apply a non-constant local burning rate at every mesh node, while SPP can only represent propellant shapes that are a combination of primitive solids (Dunn and Coats, 1997) (Peterson et al., 1968). The approach adopted in ROBOOST also allows to handle grain propellant flaws like cavities or debonding, while this is not possible in

SPP. The moving mesh method used in ROBOOST requires a re-meshing procedure (Bertacin et al., 2013) in order to preserve the quality and coherence of the mesh. Moreover, self-intersections between triangles could be identified during the motion of the mesh and they should be conveniently removed.

This paper presents the numerical procedure that was developed to incorporate a generic debonding into the main burning surface. The procedure has been integrated into ROBOOST, allowing to overcome the limitations of the previous solutions.

2   Code overview

2.1   ROBOOST code structure

Figure 1: ROBOOST modules interaction shows the software structure of ROBOOST. The grain regression module tracks the burning surface regression with a 3D triangular mesh: this strategy allows to handle a large variety of geometries. At each time step the mesh nodes are displaced according to Equations (2) and (3):

$$\Delta S_i = r_{b_i} \Delta t \qquad (2)$$

$$r_{b_i} = F_{anis_i} \, a \, p_{0_i}^n \qquad (3)$$

F_{anis_i} is a local correction factor that allows to take into account the local burning rate accordingly to the local grain anisotropy. The mesh nodes are displaced along the local burning propellant surface normal unit vector, for the i-th node, it is calculated as weight average between the normal of the triangles which contain the i-th node. This feature, combined with the 3D triangular mesh, implies a more effective description of the combustion process. During the simulation, the mesh triangles could become deformed due to nodes displacements. Thus, a re-meshing procedure needs to be performed in order to remove collapsed triangles and to split triangles edges larger than the user-defined range.

The ballistic module consists of 1D unsteady fluid-dynamics model (Shapiro, 1953) describing the evolution of the combustion chamber flow, it is based on mass conservation equation, the momentum conservation equation, the energy conservation equation and the ideal gas law (Equation (4-7)).

$$\frac{\partial \rho}{\partial t} + \frac{\partial}{\partial x}(\rho u) = \frac{r_b P_b}{A_p}\rho_{pr} \tag{4}$$

$$\frac{\partial(\rho u)}{\partial t} + \frac{\partial}{\partial x}(\rho u^2 + p) = \frac{p}{A_p}\frac{\partial A_p}{\partial x} - \frac{r_b P_b}{A_p}\rho u \tag{5}$$

$$\frac{\partial}{\partial t}\left[\rho\left(C_v T + \frac{u^2}{2}\right)\right] + \frac{\partial}{\partial x}\left\{u\left[\rho\left(C_v T + \frac{u^2}{2}\right)\right] + p\right\} = \frac{r_b P_b}{A_p}\rho_{pr}H_f \tag{6}$$

$$p = \rho RT \tag{7}$$

A constant pressure, density and temperature is set along the motor axis as initial condition, while for the velocity field a zero value is imposed. On the igniter side as boundary condition, a fixed mass flow rate and temperature is imposed to emulate the igniter effect. When the igniter effect ends, those boundary conditions are changed in zero mass flow rate and adiabatic wall. On the nozzle side, the boundary conditions are set according to the one dimensional, isentropic equation of the nozzle. The differential equation systems are solved applying a finite difference discretization for the space derivative, therefore the motor axis is divided in cells. A value of density, temperature, pressure and velocity is associated to central point of each cell. The above-mentioned parameters are the unknowns of the discretized version of Equations (4-7). This set of equations takes as input the propellant burning perimeter length (P_b) contained in the i-th cell. It can be evaluate dividing the burning surface (A_b) contained in the i-th cell with the cell width Δx (Equation(8)).

$$P_b = \frac{A_b}{\Delta x} \tag{8}$$

A_b is calculate intersecting the propellant burning surface mesh with the boundary of the i-th cell and then summing the area of the selected triangles. A_p is the port area and it is evaluated dividing the

combustion chamber volume contained inside the i-th cell with the cell width (Δx). This volume

bounded by the propellant burning surface can be calculated through Pappus-Guldinus theorem.

At each time step t_k, the propellant regression module estimates the amount of propellant burned,

then the ballistic module assesses the thermodynamic parameters within the combustion chamber.

Starting from those values, a new local burning rate is computed for each mesh node of the burning

surface (Equation (3)) in order to obtain the new spatial position at the time step t_{k+1}, where $t_{k+1} =$

$t_k + \Delta t$. Subsequently, the remaining volume of the propellant at t_{k+1} is determined. If it differs from

the volume of the propellant estimated at the time t_{k+1} and its value is below a user-defined tolerance,

the simulation can proceed to the next time step, otherwise the previous method is repeated until the

tolerance criterion is satisfied (Bertacin et al., 2013). The distribution of pressure along the

combustion chamber could be non-uniform, implying a different local burning rate for every domain

section perpendicular to the motor axis. The regression module can deal with this non-uniform grain

regression, achieving a better local description of the combustion process. Moreover, the ROBOOST

software includes the igniter dynamics (Ponti et al, 2019b), the nozzle erosion effect (Acharya et al.,

2018), the ablation contribution due to case-insulating thermal protection material (Ponti et al., 2021)

(Schoner, 1970) (Rindal, 1968). The ability of the code to handle a non-uniform regression (for

instance due to the effect of oxidizer particle orientation on burning rate (Hasegawa et al., 2017))

allows to also take into account a variation of propellant properties close to the case wall (Friedman

Curl effect) (Ponti et al., 2019a), while the ability to deal with complex geometries allows to consider

propellant inclusions in the SRM simulation (Ponti et al., 2020). Finally, the debonding module

allows to take in account the effect of debonding on the propellant combustion. The debonding is

introduced in ROBOOST as an additional inhibited surface, which does not regress until it is reached

by the burning surface. The debonding mechanical behavior is not considered and therefore its surface

does not change due to the pressure within the combustion chamber. This phenomenon has been ignored because the input debonding shape already takes into account uncertainties on its size.

If the pressure inside the motor is almost constant along the axial direction, it is possible to change the simulation representation from a time-based one to a web-based one because when the pressure is constant along the combustion chamber axis also the combustion rate becomes constant on the surface. This type of simulations are called geometrical grain regression and their results are independent from the combustion rate.

2.2 *Self-intersections removal algorithm*

The regression of the burning surface described through a non-structured mesh, as well as the incorporation of the debonding surface into the burning one, can give rise to self-intersections between the different components of the mesh. These self-intersections should be identified and removed in order to avoid anomalies that would be generated by the progression of the surface position affected by self-intersection issues. A dedicated algorithm has then been developed to address this problem. The self-intersections removal algorithm (Figure 2) adopted in this work is inspired to (Jung et al., 2004) and could be summed up in the following steps. First, mesh nodes are displaced along their normal direction according to the local burning rate (Raw offset triangular mesh block). Second, triangles of low-quality aspect ratio are removed in order to avoid numerical instabilities (Remove degenerate triangles block). This re-meshing procedure performs five controls:

- *minimum triangles edges*: the triangles with an edge length shorter than a user-defined length are deleted. The edge that does not comply with this requirement is collapsed in one of its extreme points.

- *maximum triangles edges*: the triangles with an edge length greater than a user-defined length are split.

- *squeezed triangles*: the triangles that have a bad aspect ratio are removed by collapsing one of their edges.

- *mesh coherence*: this procedure deletes any isolated points which do not belong to any triangles and solves the triangles with non-manifold edges.

- *mesh smoothing*: this procedure smooths the sharp edges of the mesh which do not satisfy a user-defined tolerance on the angle between the normal vectors of the triangles.

Third, the self-intersection points of the triangles are computed through geometrical primitive-primitive intersection functions using the Moller algorithm (Moller, 1997). To speed up the self-intersection identification process, the mesh is partitioned into subsets using an Axis-Aligned Bounding Box (AABB) procedure. Fourth, if at least one self-intersection has been detected, the physical portion of the burning surface is identified through a growing region algorithm while the non-physical one is removed.

2.2.1   *Growing region algorithm*

Once the triangle-triangle intersections have been computed and at least one self-intersection has been detected, the triangular mesh is classified into three subsets of triangles:

- *valid region*: it contains triangles which have to be kept (yellow triangles in Figure 3) because they represent part of the physical surface (solid propellant surface).

- *invalid region*: it contains triangles which have to be deleted because they belong to a non-physical surface.

- *partially valid region*: it contains triangles crossed by the intersection line (violet, light blue, and green triangles in Figure 3) which splits them in two parts: one of these parts has to be kept, while the other one has to be deleted.

Actually, only the valid region is computed by applying a growing region algorithm. This algorithm requires an initial triangle called seed triangle which has to belong to the valid region. All triangles having the three vertices on the convex hull of the mesh could be chosen as seed triangle because

they always belong to the valid region. Figure 4 shows how the growing region algorithm works in case the burning surface reaches the region where the debonding is located: the burning surface is depicted in white in Figure 4a, while the debonding surface is orange. As Figure 4b and Figure 4c show, the growing region algorithm expands the valid region starting from the seed triangle (the red triangle in Figure 4b) including the ones surrounding it if and only if they do not reside in the partially valid subset1. In Figure 4b and 4c, the growing region expands including the $\triangle BDC$ and $\triangle BFD$ triangles because they are near the growing region triangles and they do not belong to the invalid region. $\triangle CDE$ is a partially valid triangle because it is crossed by the intersection line depicted in red in Figure 4.

When the growing region algorithm reaches a triangle belonging to the partially valid region ($\triangle CDE$ in Figure 4c), it stops growing. The partially valid triangle reached by the algorithm has to be first divided in sub-triangles, as shown in Figure 4d. Then, the growing region incorporates only the valid ones, as shown in Figure 4e. The remaining part of $\triangle CDE$ belongs to the invalid region, therefore the algorithm must cross over the intersection line to move into an unexplored region of the mesh. This process is identified as crossing-the-river method and it is highlighted in Figure 5. The first step consists in dividing the other intersecting triangle in sub-triangles. Then, the crossing-the-river algorithm starts from $t1$, which is a valid sub-triangle of $\triangle CDE$, and it tries to understand which one among the sub-triangles of $\triangle A'B'C'$ belongs to the valid region. In particular, in Figure 5, the algorithm has to choose between the triangles $t2$ and $t3$. This choice is made following a geometric criterion, as shown in Figure 5b. The unit vector $\hat{e}_n$ is normal to the plane in which the triangle $\triangle CDE$ lies, while the two unit vectors $\hat{e}_2$ and $\hat{e}_3$ are perpendicular to the intersection line (red line in Figure 5a) and they belong to the $\triangle A'B'C'$ triangle: $\hat{e}_2$ points to the half-plane containing B', while

---

[1] A triangle near the valid region could not belong to the invalid region because it is separated from the valid region by the partially valid region.

$\hat{e}_3$ faces the opposite direction. If $\hat{e}_2 \cdot \hat{e}_n > 0$, the $t2$ triangle is included in the growing region, while if $\hat{e}_3 \cdot \hat{e}_n > 0$ the triangle $t3$ would be chosen.

Figure 6 shows an example of intersection between an elliptical inclusion (red surface) and the solid rocket motor burning surface (yellow surface). The red section of the ellipse shown in Figure 6d contains triangles belonging to the invalid region: they will therefore be removed because they do not represent a physical surface anymore. In the same figure, the yellow part of the ellipsoidal surface contains triangles belonging to the valid region, therefore they have been kept and merged with the burning surface mesh.

The self-intersection algorithm is quite slow hence it is run every n time steps in order to reduce the global execution time. When a self-intersection is detected, the simulation is re-winded of $n - 1$ time steps to search the time at which the first self-intersection is risen in order to correct the mesh. Then, the simulation continues from this point.

2.3   Debonding integration procedure

In ROBOOST, the debonding is implemented as an additional disabled surface added inside the propellant domain and discretized with a triangular mesh. The grain regression is not active on this surface until it has been reached by the propellant burning one. At every n time step, the occurrence of intersections between these two surfaces is checked and each time an intersection is detected, the self-intersection removal algorithm removes the invalid region as described in Section 2.2. On the contrary, if no intersections are found, the burning surface mesh remains unchanged: this means that the burning surface has not reached the debonding region yet. To facilitate the debonding-burning surface integration, a local mesh refinement is performed near the debonding region: the mesh of the burning propellant surface is locally refined splitting the triangles until they have a similar size with the one forming the debonding mesh, thus improving the union of the two meshes when the burning propellant one reaches the debonding one. Similar triangle edge dimensions of the two meshes ensure the avoidance of an ill-conditioned numerical problem in evaluating the intersection points of the

triangles. The same procedures described in Section 2.2 are used to perform the local refinement. Self-intersections can occur not only due to propellant cavities or debonding but also due to portions of the propellant burning surface which consume the same propellant volume coming from opposite directions.

3   Simulations settings

For all the following simulations, the solid propellant surface has been meshed starting from a CAD representation of the surface using an open-source software (Geuzaine and Remacle, 2009). All the following simulations have been obtained as geometrical regression because, for the geometries considered in this paper, the propellant combustion rate is almost constant along the motor axis. Consequently the debonding effect can be analyzed as a function of the web, which is the coordinate that represents the combustion progression and displayed independently of the motor regression rate. In order to investigate the acceptance of the motor a regular constant burning rate simulation of the actual geometry with the debonding has to be run and compared with the nominal one. One of the ROBOOST main outcomes is the thermal protection exposure web map, which displays how long the thermal protections have been exposed to hot gases of the combustion chamber. The exposure increase map of the case-insulating thermal protection material is expressed in length units instead of time units, whereas the length is identified by the amount of web difference between the standard SRM simulation and the one including the debonding geometry in which the thermal protections are discovered earlier. In particular, the web parameter at a specified time is obtained by computing the integral of the burning rate-time profile up to the time instant of interest. Thus, the web difference between the burning surface with and without the debonding geometrically represents the distance displaced by the burning surface from the first point of contact with the debonding and the case itself. Hence, with the above-mentioned strategy, web difference maps no longer depend on the burning rate, but they are suitable for general values of burning rate.

4 Reliability and robustness of the self-intersection removal algorithm

A challenging test geometry has been designed to test the self-intersection algorithm in a critical scenario where the self-intersections arise in a large region of the propellant burning surface. This geometry consists of four separated lobes that evolve until they meet each other. Figure 7 shows the evolution of the burning surface. When the four lobes consume the cross-shaped central region of the propellant shortly after a web value of $10\ mm$, the self-intersection arises. Figure 8a,b show a comparison between the simulation where the self-intersections have not been removed (red lines) and the one where they are deleted (blue lines). From these two figures, it is possible to notice that when the self-intersections are not removed, the burning surface area increases more than the actual one and the consumed propellant starts to diverge from the true solution, leading to completely wrong results. The simulation where the self-intersection was not removed was stopped close to the web value of $26\ mm$ because it lost any physical meaning, therefore it was not necessary to complete it. Figure 9a to c show the propellant burning surfaces at different web values (points 1a, 2a and 3a in Figures 8a and 8b) obtained from the simulation with the self-intersection removal algorithm enabled in a detection-only mode in which the self-intersections are detected but not removed.

In these figures, the invalid region is depicted in red. Figure 9d to f show the propellant burning surfaces at different web values (points 1b, 2b, and 3b in Figure 8a and b) obtained from the simulation with the completely enabled self-intersection removal algorithm: the self-intersections are detected and removed. From these figures, it is possible to notice that when the self-intersection is not removed, the burning surface area could increase due to the growth of non-physical surface, identified by the red triangles in Figure 9a to c. This surface does not enclose any propellant; therefore, it leads to an erroneous evaluation of the burning surface area and the propellant burning rate. Hence, it is crucial to ensure the detection and removal of the self-intersections to avoid erroneous results. Figure 9a to c show that the algorithm is able both to identify the triangles belonging to the invalid region and to cut them out also for this complex test case. Figure 10 shows the thermal

15

protection exposure map from which it is possible to recognize the initial four lobes where the thermal

protection case has the longest exposure to the hot gases in the combustion chamber.

5   Validation

The ROBOOST software has been used to simulate the third stage of a Vega launcher, namely Z9. This launcher is able to transport small payloads – up to 1,500 kg in low Earth orbits. It consists of four stages: the first three are solid propellant-based, while the last one is a liquid propellant engine. Z9 is $3.5\ m$ tall and has a diameter of about $2\ m$. It weighs $11500\ kg$. It is filled with the following composite propellant (Al-HTPB-AP): aluminum particles (Al), hydroxyl-terminated-polybutadiene (HTPB), and ammonium perchlorate (AP). The burning surface evaluated by ROBOOST has been compared with the one obtained by SPP, which can be considered a reference in the literature. Figure 11 shows the burning surface obtained from the two aforementioned softwares at different web values. Both axes have been divided by their maximum value in order to obtain dimensionless data. The Equations (4) and (5) have been used to calculate the mean error between the two simulations' results. Since it amounts to about 0.14 %, ROBOOST can be considered validated on this geometry.

$$x_i^* = \frac{S_{b_{SPP}}(w_i) - S_{b_{ROOBOST}}(w_i)}{max_{i\in[1,N]}S_{b_{SPP}}(w_i)} \tag{9}$$

$$\bar{x}_i^* = \frac{1}{N}\sum_{i=1}^{N} x_i^* \tag{10}$$

N is the number of web values considered for the comparison and w_i is the i-th web value. The ROBOOST validation with SPP cannot be carried out on a geometry with debonding because SPP is not able to evaluate their effects. Therefore ROBOOST has also been validated on a simple geometry with a debonding shown in FIGURA on which is possible to perform analytical calculation. Figure 12Figure 12: ROBOOST - Analytical burning surface validation shows the comparison between the

simulation and the analytical solution of the geometrical regression. The error remains low than 0.4% in the central part of the simulation while it increases only at the end of the combustion reaching a percentual error lower than 0.8%, therefore the verification can be considered satisfactory.

6   Debondings simulation results

Since the ROBOOST validation has been performed on a Z9 solid rocket motor, the following simulations have been computed on the same geometry. In these cases, the size of the triangular edges of the generated meshes ranges between $5\ mm$ and $30\ mm$: the smallest triangles are located close to the debonding site because of the mesh's local refinement. The surface of the case is examined through the coordinate system (x_c, θ) shown in Figure 13: x_c is the magnitude of the curvilinear coordinate moving along the case from the igniter to the nozzle, θ is the tangential coordinate, and $\hat{e}_{x_c}$ is the local curvilinear coordinate direction on the case itself.

Five debondings with different shapes and/or locations have been studied by performing geometrical regression simulations. Their shape, size, and position have been chosen according to actual debonding detected by radiographies of the propellant grain. In the first three simulations, the debondings are located along the surface of the case, as in Figure 14: the debonding 1 is depicted in orange, while the debondings 2 and 3, represented respectively with the colors red and green, have been obtained by shifting the debonding 1 along the case curvilinear coordinate x_c. The last two simulations have been computed by setting the debondings 4 and 5 with same location and same tangential width as the debonding 1 but with a different size along the curvilinear coordinate x_c. They are depicted in light blue and blue respectively in Figure 14b and c. Figure 15 shows the width and position along the case curvilinear coordinate of the five debondings.

Figure 16 shows the evolution of the burning surface of the simulation without any debonding compared with the one with the debonding 2: the red circle highlights the regions influenced by the debonding. When the star tip reaches the debonding surface, the hot gases inside the combustion chamber flow into the debonding volume, igniting the propellant in the region highlighted by the red

17

circle. Figure 17 shows a comparison between Z9 simulations with and without the propellant debonding in terms of burning surface area: until the combustion reaches the debonding region, the simulation results are identical, while they start to differ from a non-dimensional web equal to 0.1. When the grain burning surface reaches the debonding, its area increases (Figure 17a) rapidly due to the instantaneous exposure of the debonding surface. Figure 18 shows the Z9 thermal protection exposure web map variation. It has been obtained as the difference between the thermal protection exposure web map of the simulation with the debonding 2 minus the thermal protection exposure web map of the simulation without any debondings. Only in the region close to the debonding 2 site there is an appreciable variation in the thermal protection exposure web difference, implying that the thermal protection's case-insulating layer is prematurely exposed to the combustion chamber environment of a web distance of approximately $79 \ mm$.

The same analysis performed on the debonding 2 could be repeated for the other ones. Figure 19 shows the thermal protection exposure web difference between the simulation with the debonding 3 and the simulation with no debonding. Also in this case, the effect of the debonding on the thermal protection is localized around the debonding position leading to an early exposure, in terms of web difference, of the thermal protection layer equal to $75 \ mm$. The debonding 5 has the largest area on the case's surface, therefore it will lead to higher thermal protection web advance. Figure 20 shows the comparison between the Z9 standard simulation and the one with the debonding 5. The red circle highlights the region influenced by the debonding. In this region, the debonding effect which causes a different shape of burning surfaces is evident. Figure 21 shows the thermal protection exposure web difference between the simulation with the debonding 5 and the simulation without debondings. In this case, the maximum thermal protection exposure web difference is of approximately $161 \ mm$.

The effects of the five debondings analyzed here can be summed up by evaluating the maximum thermal protection web difference in every performed simulation and displaying this result against the debonding sizes and positions. Figure 22 shows how the x_c position of the debonding influences

the maximum thermal protection web exposure difference, while Figure 23 shows how the maximum thermal protection web exposure difference is influenced by the size of the debonding along the curvilinear coordinate direction. From these two plots, it can be seen that both an increase in the debonding position along the case curvilinear coordinate and an increase in the debonding width along the same direction generate an increment of thermal protection web exposure.

The mesh size has been chosen in order to avoid mesh-dependent results. A simple procedure has been adopted to choose the triangles edge length which allows to match this goal limiting the computational time. The triangle edges size of the burning surface mesh is directly connected with the error on the calculated increment of <u>thermal</u> protection exposure in term of web because the advance of the thermal protection exposure is related to the web value at which the burning propellant surface reaches the debonding one. Smaller triangle edges length allows to discretize better the surface portions with low local curvature radius involving a better evaluation of the burning propellant surface positions. Z9 has a star shape geometry in the portion close to the nozzle and the simulated debonding are located close to this region, therefore the star tips requires small triangles size to be represented well or otherwise they will be shortened by the discretization implying that the debonding surface will be reached at higher web value than the true one. Therefore to limit the previous described error, it is important to have a good discretization of the curved surface (in particular the star tips for the geometry studied in this paper). This can be checked comparing the actual area of the initial surface (A_{ref}), evaluated by CAD software, with the one calculated from the triangular discretization (A_{mesh}): triangular meshes always underestimate the total area, this error intuitively is related to the curve portion of the surface, therefore the goodness of the discretization can be measured as the error between the true initial burning surface area (A_{ref}) with the one obtained summing the area of every triangles contained into the mesh (A_{mesh}). Figure 24 shows the percentage error of eleven tested meshes evaluated with Equation (11). Decreasing the mean triangles edge size, the error decreases but at the same time the number of triangles contained inside the mesh increased.

The mesh used in this paper has been chosen taking the mesh with lowest number of triangles which has an error lower than fixed error threshold (0.15%, dashed line in Figure 24).

$$e_A = \frac{A_{ref} - A_{mesh}}{A_{ref}} 100 \tag{11}$$

All simulations previously discussed have been carried out with the ROBOOST software installed on a calculator with the following features: 16 GB of RAM on an Intel Core i7-7th generation CPU machine with 3.10 GHz and a Nvidia Quadro M1200 graphics card. Each one of them took about 19 hours to be completed.

7   Conclusion

ROBOOST has been successfully validated with respect to SPP on Z9 SRM with a maximum burning surface mean error of 0.14%. The self-intersection algorithm is of fundamental importance to manage the displacement of the 3D triangular mesh representing the burning surface. It has been shown that it is effective also with complex geometries which generate a great number of self-intersections. It could be also applied for the phenomenon of debonding, which generates issues of similar effort; for this reason, the presence of five debondings was simulated. These simulations were performed on Z9 SRM and they show that ROBOOST is able to handle debondings with different sizes and locations. The tool described in this work could be used to investigate actual debondings in order to verify whether the thermal protection layer is completely consumed during the combustion of the propellant due to the additional time in which it was exposed to the hot gases. Among the reported simulations, the last one is the most critical. If a constant propellant burning rate equal to $6.5\ mm/s$ is assumed, the thermal protection layer will be exposed to the combustion chamber's hot gases for $24.9s$ longer than the case without the debonding.

References

Acharya. R., Evans, B., Pitt, J., Costanzo, F., Kuo, K., K., Coupling of Transient Thermal and Mechanical Stresses Computations in Graphite Nozzle Materials, *International Journal of Energetic*

Materials and Chemical Propulsion, Vol. 16(2) pp. 175-195, 2018.   DOI: 10.1615/IntJEnergeticMaterialsChemProp.2018024875

Bertacin, R., Ponti, F., Corti, E., Fedele, D., and Annovazzi, A., Numerical Simulation of the Zefiro 9 Performance Using a New Dynamic SRM ballistic Simulator, *49th AIAA/ASME/SAE/ASEE Joint Propulsion Conference*, San Jose, CA, USA, 2013

Chen, L., Gui-e, L., Jinyong, J., Qiang, G., Shaoguang, W., and Bin, W., Rewiev on the test method of adhesive failure in solid rocket motor, *Joint International Mechanical, Electronic and Information Technology Conference,* Chongqiong, China, 2015

Coats, D., Cohen, N., Levine, J., Nickerson, G., Prics, C., and Tyson, T., A computer program for the prediction of solid propellant rocket motor performance, Air Force Rocket Propulsion Laboratory Technical Note , 1975

Coats, D., Nickerson, G., Dang, A., Dunn, S., and Kehtarnavaz, H., Solid performance program (spp), *23rd Joint Propulsion Conference,* San Diego, CA, USA, 1987. DOI: 10.2514/6.1987-1701

Dauch,F., and Ribéreau, D., A software for SRM grain design and internal ballistic evaluation PIBAL (Propulsion and Internal BALlistic), *38th AIAA/ASME//SAE/ASEE Joint Propulsion Conference & Exhibit*, Indianapolis, Indiana, USA, 2002. DOI: 10.2514/6.2002-4299.

Dunn, S., and Coats, D., 3-d grain design and ballistic analysis using the spp97 code, *33rd Joint Propulsion Conference and Exhibit*, Seattle, WA, USA, 1997. DOI: 10.2514/6.1997-3340

Geuzaine, C., and Remacle, J., F., Gmsh: A three-dimensional finite element mesh generator with built-in pre- and post-processing facilities, *International Journal for Numerical Methods in Engineering*, Vol. 79 (11), pp. 1309-1331, 2009. DOI: 10.1002/nme.v79:11

Gibou, F., Fedkiw, R., and Oscher, S., A review of level-set methods and some recent applications, *Journal of Computational Physics*, Vol. 353(15), pp. 82–109, 2018. DOI: 10.1016/j.jcp.2017.10.006

Hasegawa, H., Fukunaga, M., Kitagawa, K., and Shimada, T., Effect of Oxidizer Particle Orientation on Burning Rates of Composite Propellants, *International Journal of Energetic Materials and Chemical Propulsion,* Vol. 15(4), pp. 285-304, 2017.   DOI: 10.1615/IntJEnergeticMaterialsChemProp.2016014195

Jung, W., Shin, H., and Choi, B., Self-intersection removal in triangular mesh offsetting, *Computer-Aided Design and Applications,* Vol. 1, pp. 477-484, 2004. DOI: 10.1080/16864360.2004.10738290

Kumar, K. N. V. S. N. , and Selvaraj, P., Stresses at the interface between the propellant and insulation using linear visco elastic analysis, *Proceedings of the National Seminar & Exhibition on Non-Destructive Evaluation,* NDE, India, 2011

Liu, T., Zhang, W., and Yan, S., A novel image enhancement algorithm based on stationary wavelet transform for infrared thermography to the de-bonding defect in solid rocket motors, *Mechanical Systems and Signal Processing,* Vol. 62-63, 2015.DOI: 10.1016/j.ymssp.2015.03.010

Martin, H., T., Cortopassi, A., C., and Kuo, K., K., Assessment of the Performance of Ablative Insulators under Realistic Solid Rocket Motor Operating Conditions, *International Journal of Energetic Materials and Chemical Propulsion*, Vol. 16(1), pp. 1-22, 2017.   DOI: 10.1615/IntJEnergeticMaterialsChemProp.2017021828

Moller, T., A fast triangle-triangle intersection test, *Journal of Graphic Tools,* Vol. 2, pp. 25-30, 1997. DOI: 10.1080/10867651.1997.10487472

Oscher, S., and Sethian, J., Fronts propagating with curvature-dependent speed: Algorithms based on Hamilton-Jacobi formulations, *Journal of Computational Physics*, Vol. 79(1), pp. 12–49, 1988. DOI: 10.1016/0021-9991(88)90002-2

Peterson, E., Nielsen, C., Johnson, W., Cook, K., and Barron, J., Generalized coordinate grain design and internal ballistic evaluation program, *3rd Solid Propulsion Conference*, Atlantic City, NJ, USA, 1968. DOI: 10.2514/6.1968-490

Ponti, F., Mini, S., and Annovazzi, A., A simplified approach to predict Friedman curl effect in a solid rocket motor using ROBOOST simulation tool, *AIAA Propulsion and Energy Forum*, Indianapolis, IN, USA, 2019. DOI: 10.2514/6.2019-3960

Ponti, F., Mini, S., and Annovazzi, A., Numerical evaluation of the effects of inclusions on solid rocket motor performance, *AIAA Journal*, Vol. 58(9), pp. 4028-4036, 2020. DOI: 10.2514/1.J058735

Ponti, F., Mini, S., Fadigati, L., Corti, E., and Annovazzi, A:, Influence of Nozzle Radiation on Solid Rocket Motors Tail-off Thrust, *International Journal of Energetic Materials and Chemical Propulsion*, 2021. DOI: 10.1615/IntJEnergeticMaterialsChemProp.2021038491

Rindal, R., A., An analysis of the coupled chemically reacting boundary layer and charring ablator, NASA CR-1063, 1968

Schoner, R., J., User's Manual Aerotherm Charring Material Thermal Response and Ablation Program, 3 ed., Air Force Rocket Propulsion Laboratory Director of Laboratories, 1970.

Shapiro, A. H., *The Dynamics and Thermodynamics of Compressible Flow*, The Roland Press Company, pp. 3-22, 1953

Images and tables
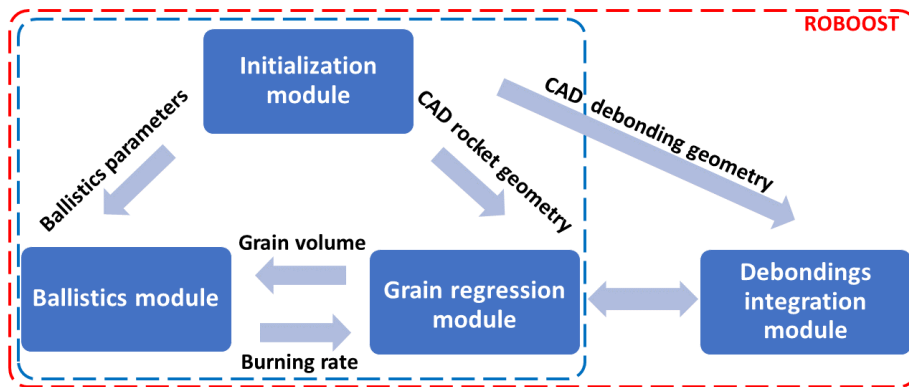
Figure 1: ROBOOST modules interaction
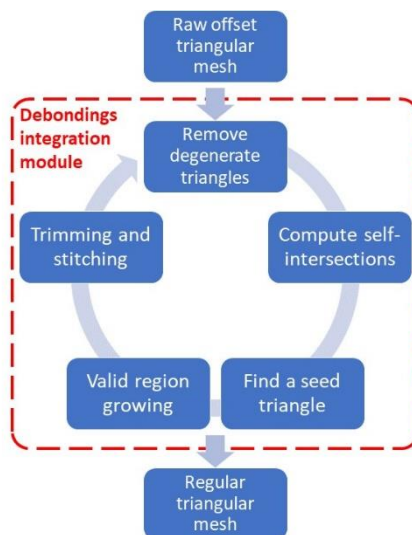
Figure 2: ROBOOST debonding integration procedure
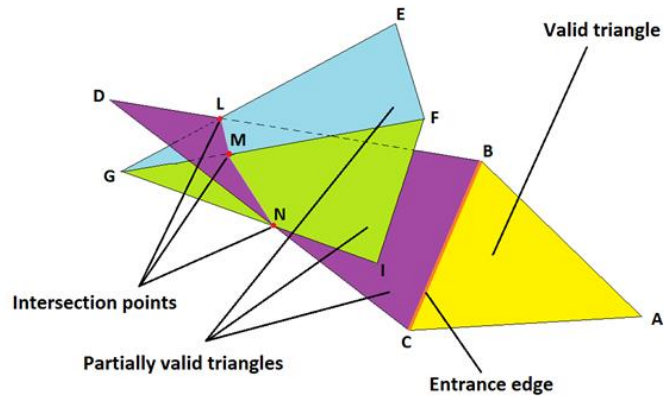
Figure 3: Triangles subsets

a)

b)

c)

d)

e)
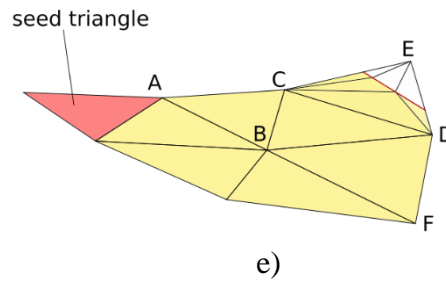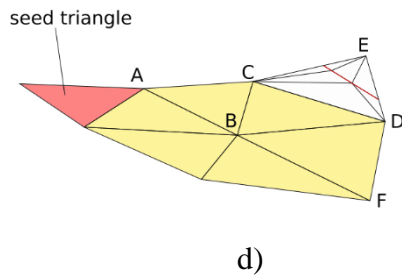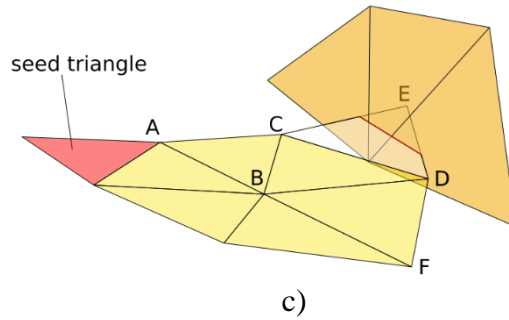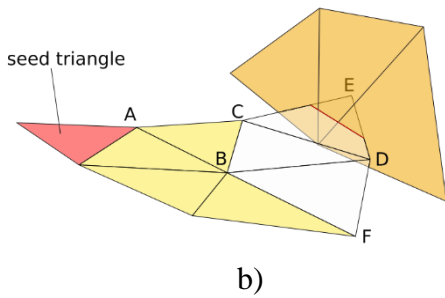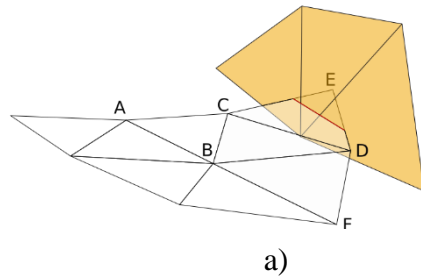
Figure 4: Growing region algorithm
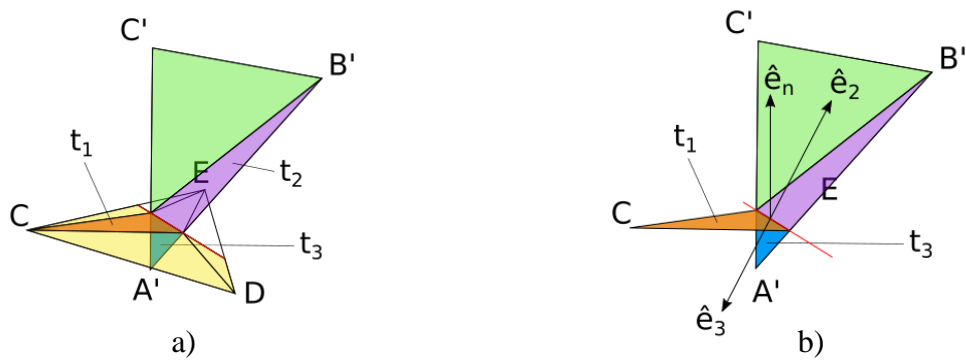
Figure   5:   Crossing-the-river procedure
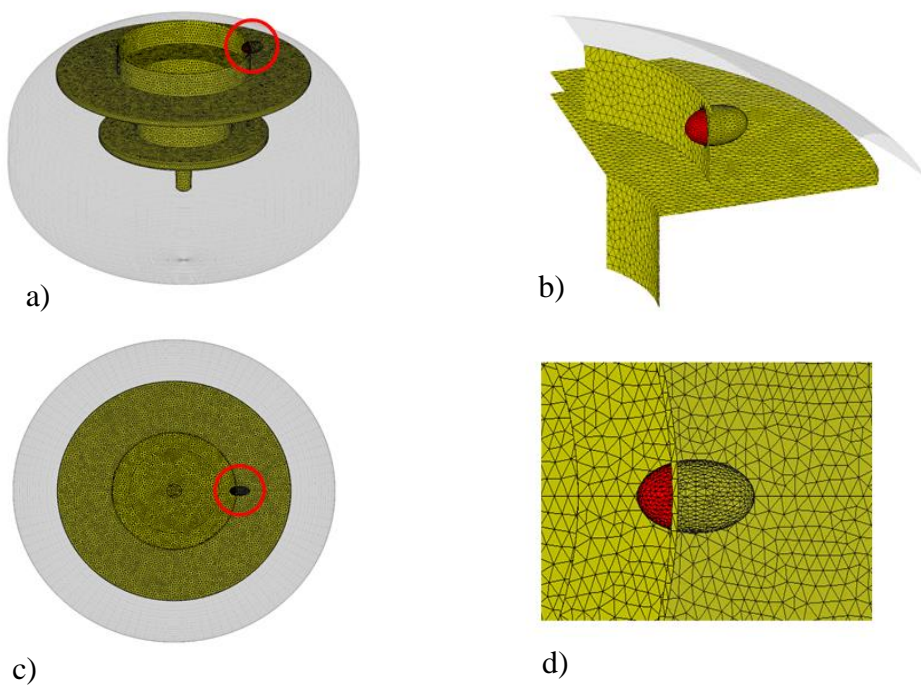
Figure   6:   Elliptical   cavity intersection
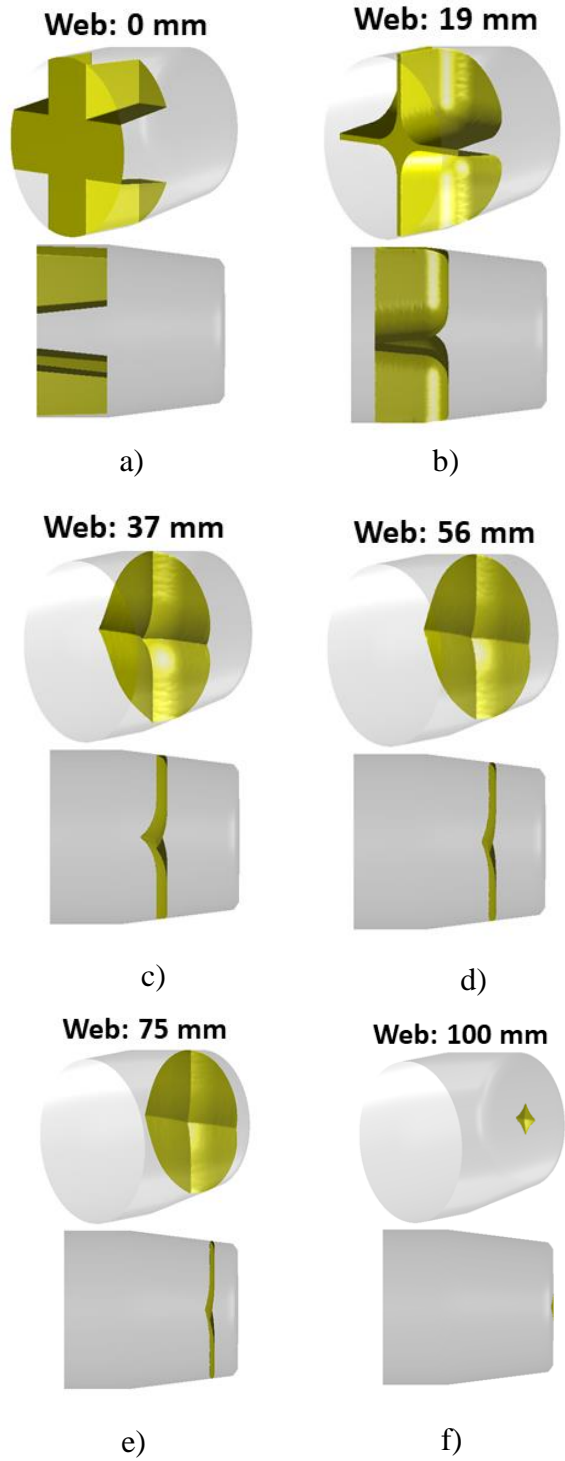
Figure 7: Test geometry burning surface regression
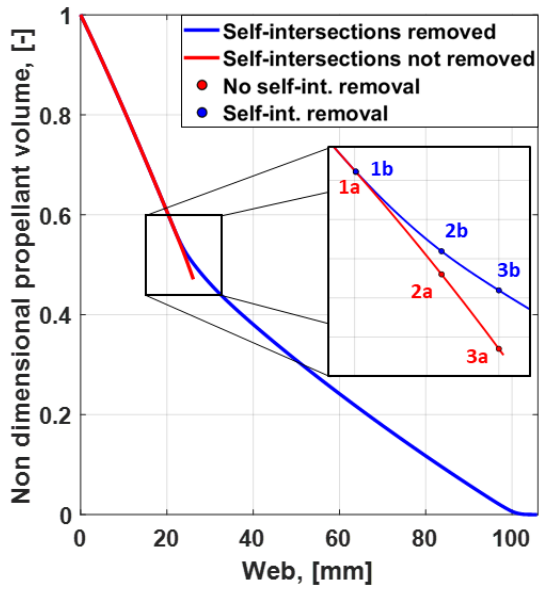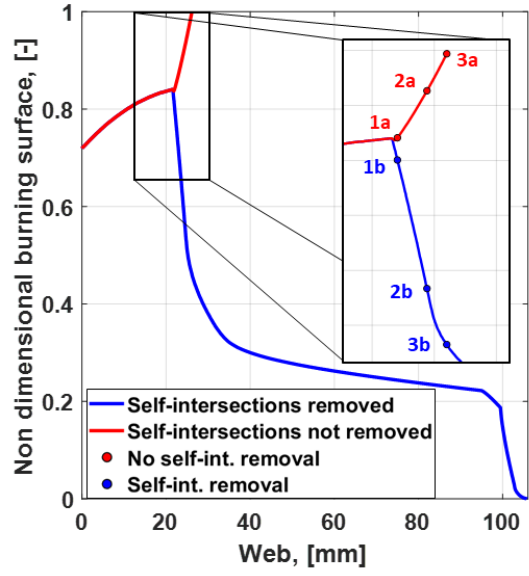
Figure 8: Burning surface and propellant volume time
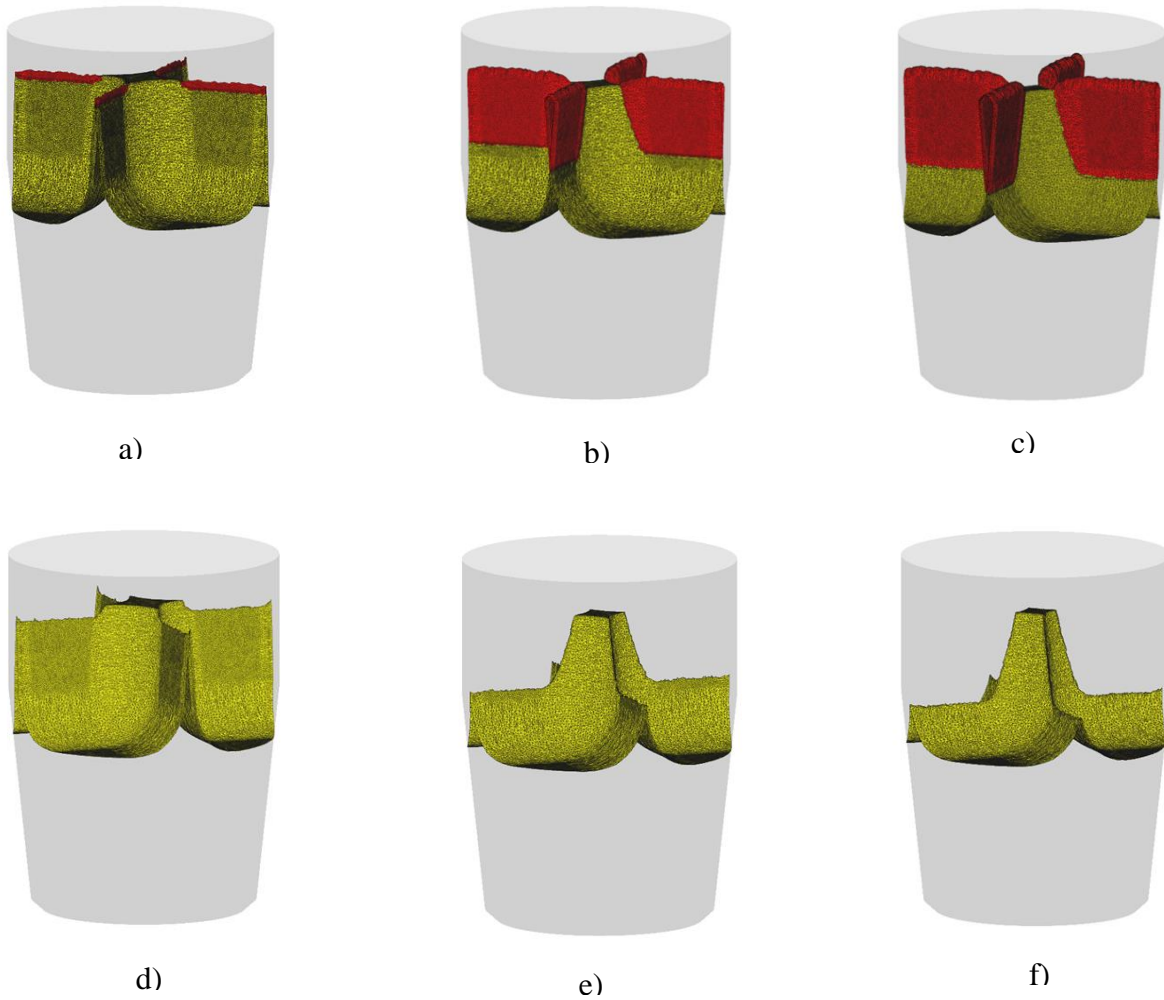
Figure 9: The first row shows the propellant burning surface obtained by the simulation with the self-intersection removal algorithm set in detection-only mode, while the second one shows the simulation result with the fully enabled self-intersection removal algorithm
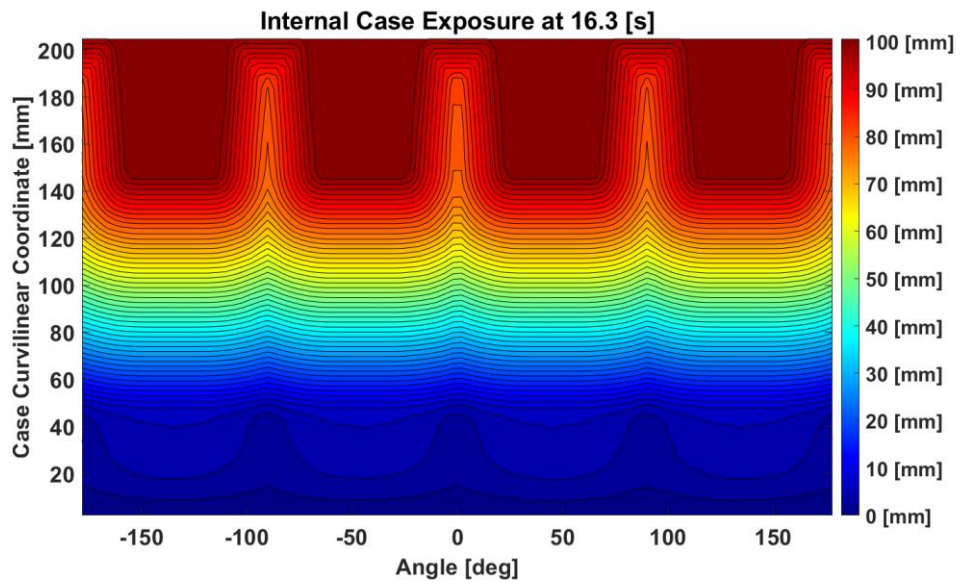
a)
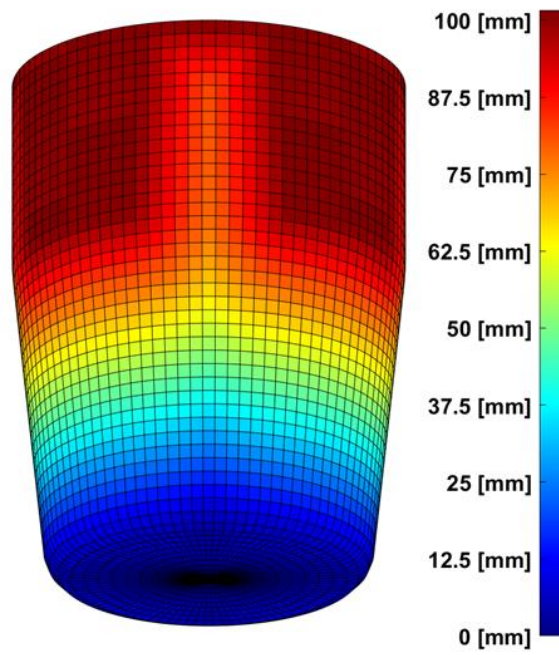
b)

Figure 10: thermal protection web exposure map
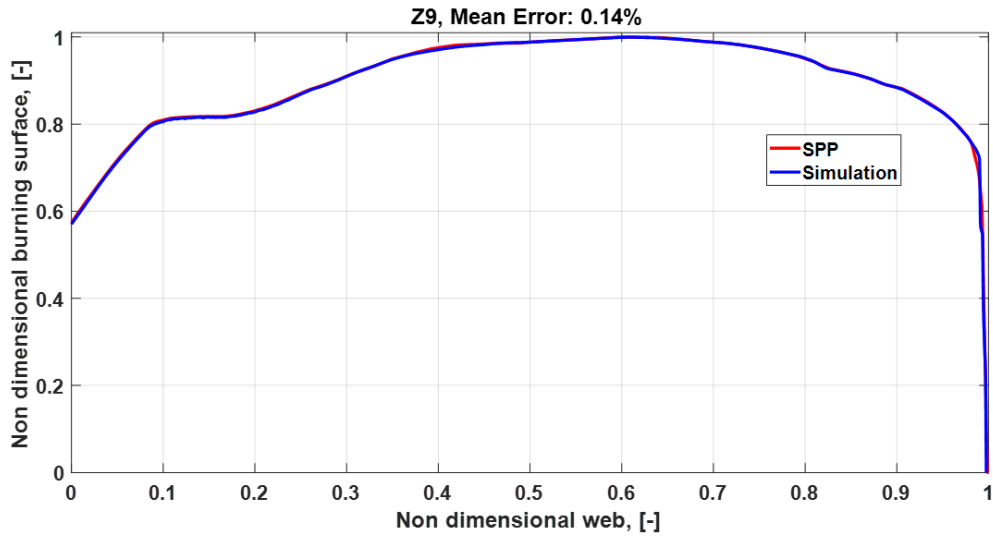
Figure 11: ROBOOST-SPP burning surface validation
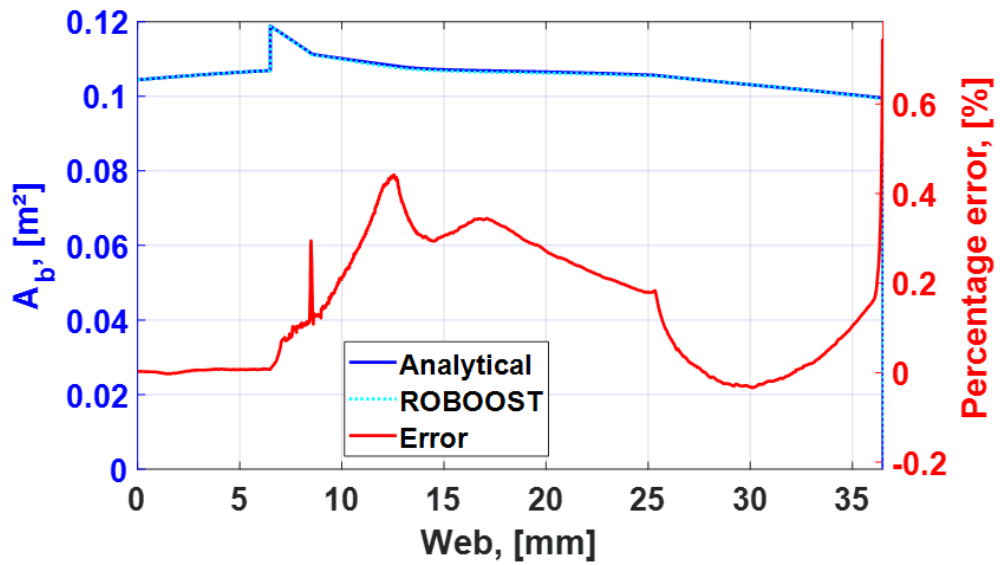
Figure 12: ROBOOST - Analytical burning surface validation
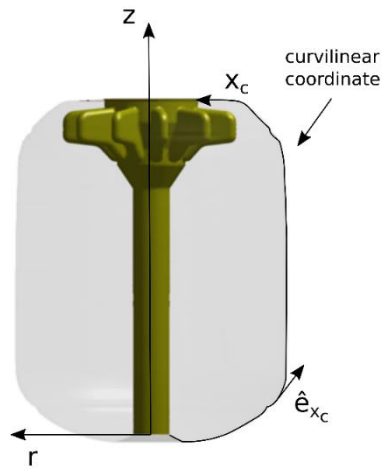
Figure 13: Curvilinear coordinate system
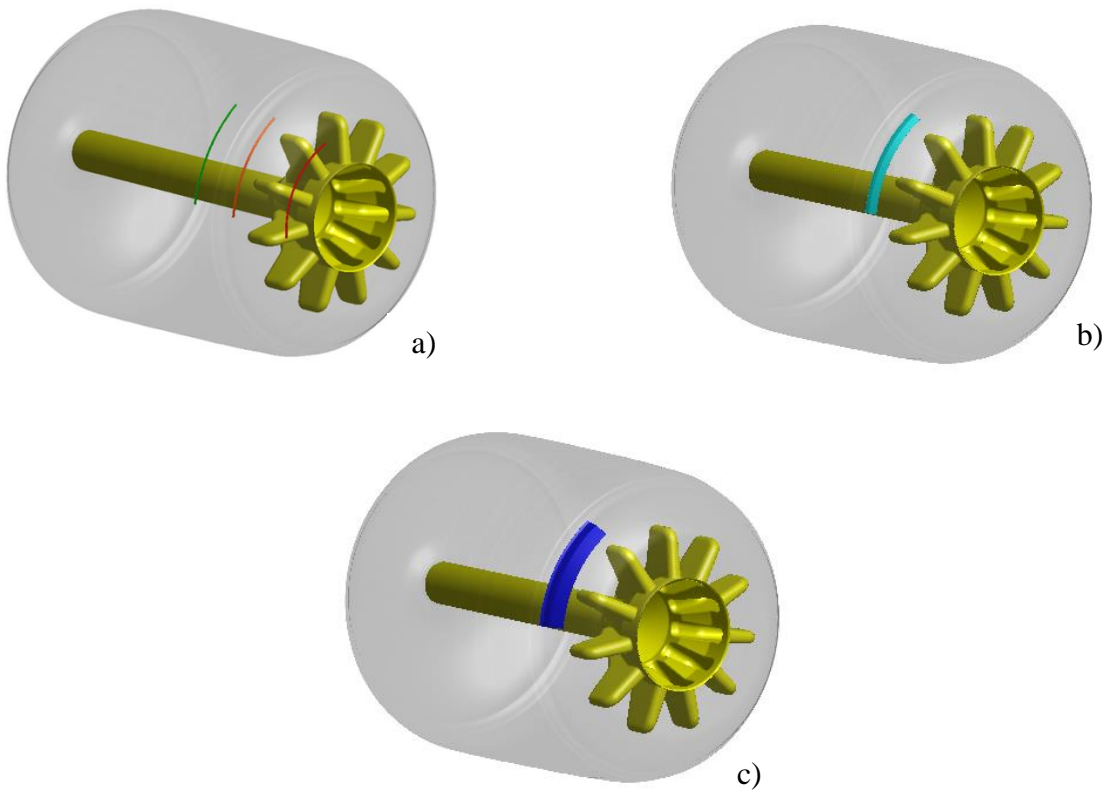
Figure 14: Z9 location of debondings
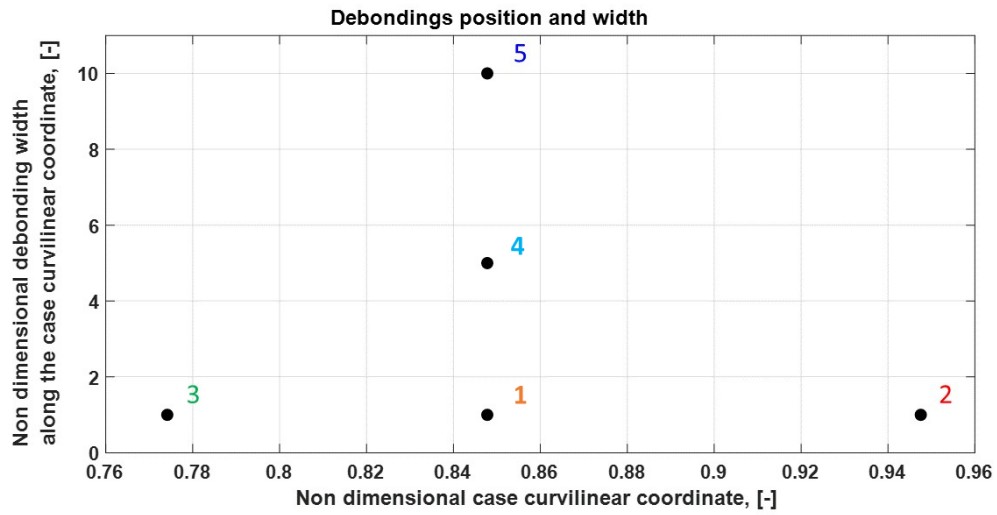
Figure 15: Debondings position and width

Web: 11%   Web: 9.4%   Web: 15%

a)   b)   c)
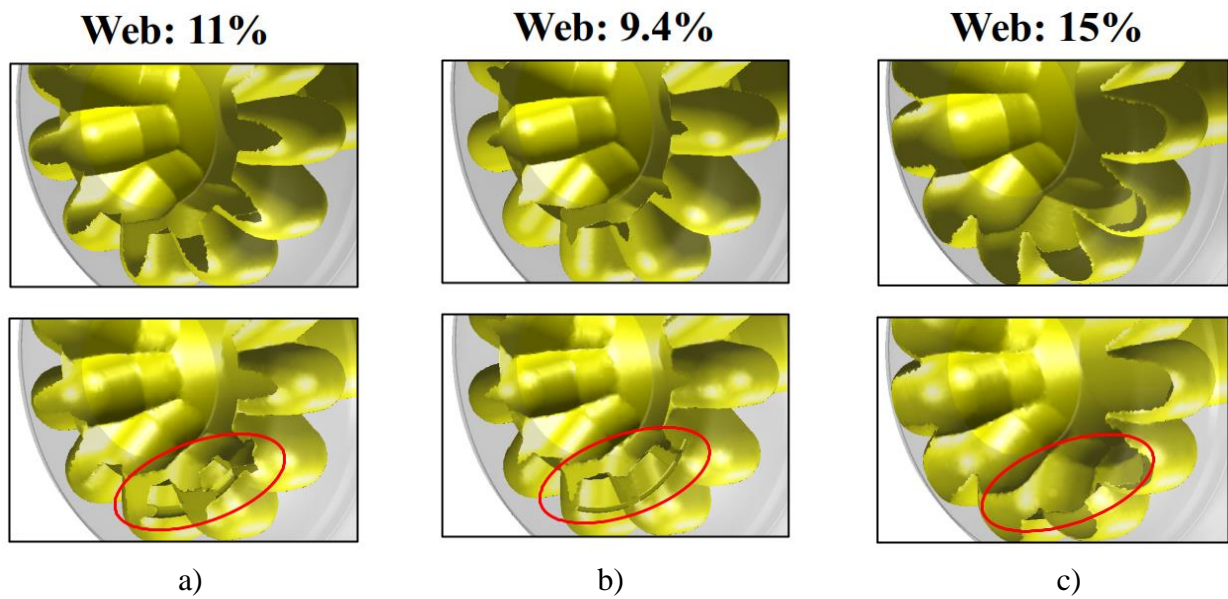
Figure 16: Comparison between the Z9 standard simulation and the simulation with debonding 2

a)

b)
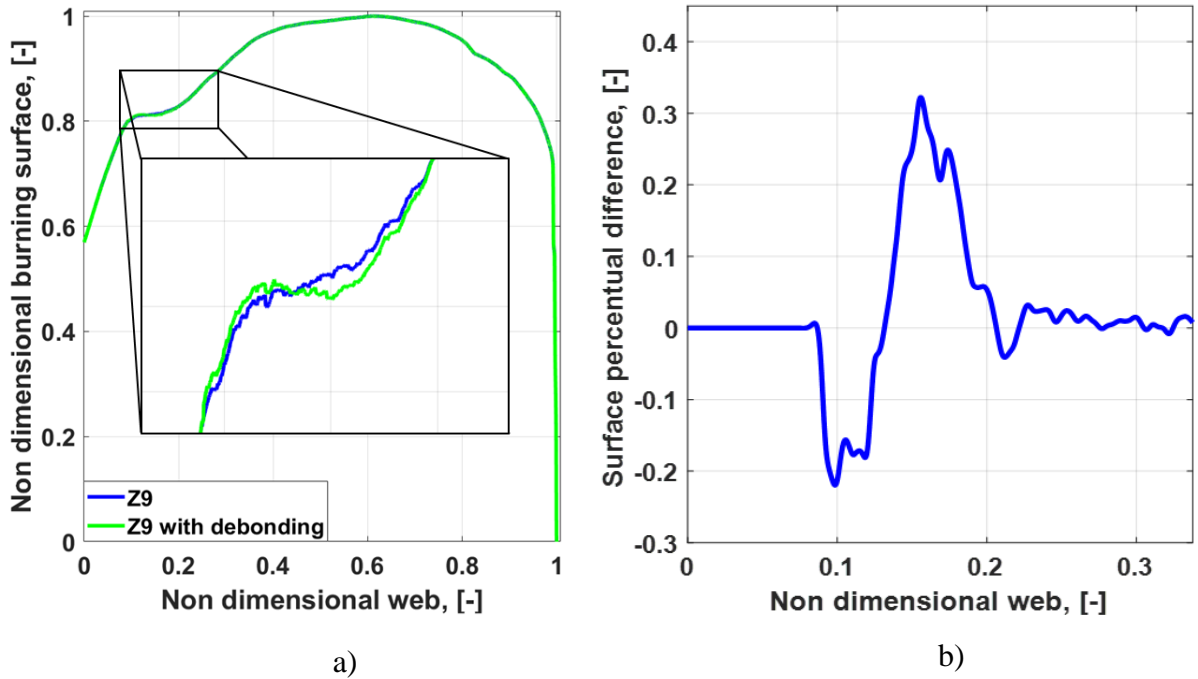
Figure 17: Simulation with debonding 2 compared with the standard simulation
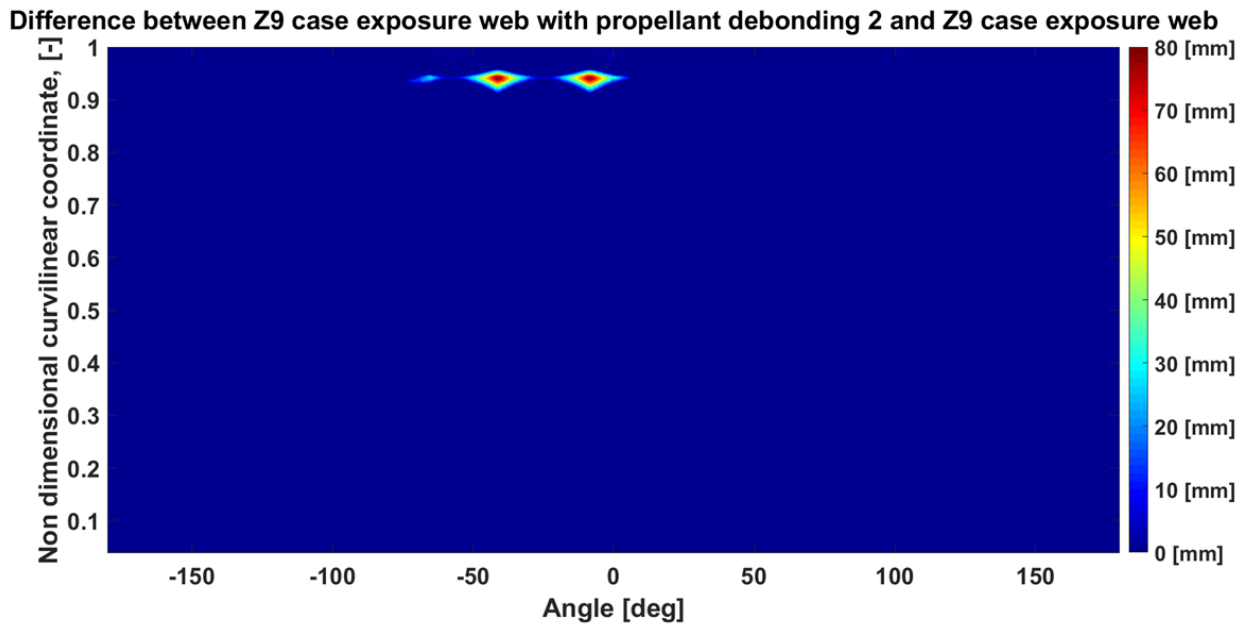
Figure 18: Difference in thermal protection exposure web in the debonding 2

34

Difference between Z9 case exposure web with propellant debonding 3 and Z9 case exposure web
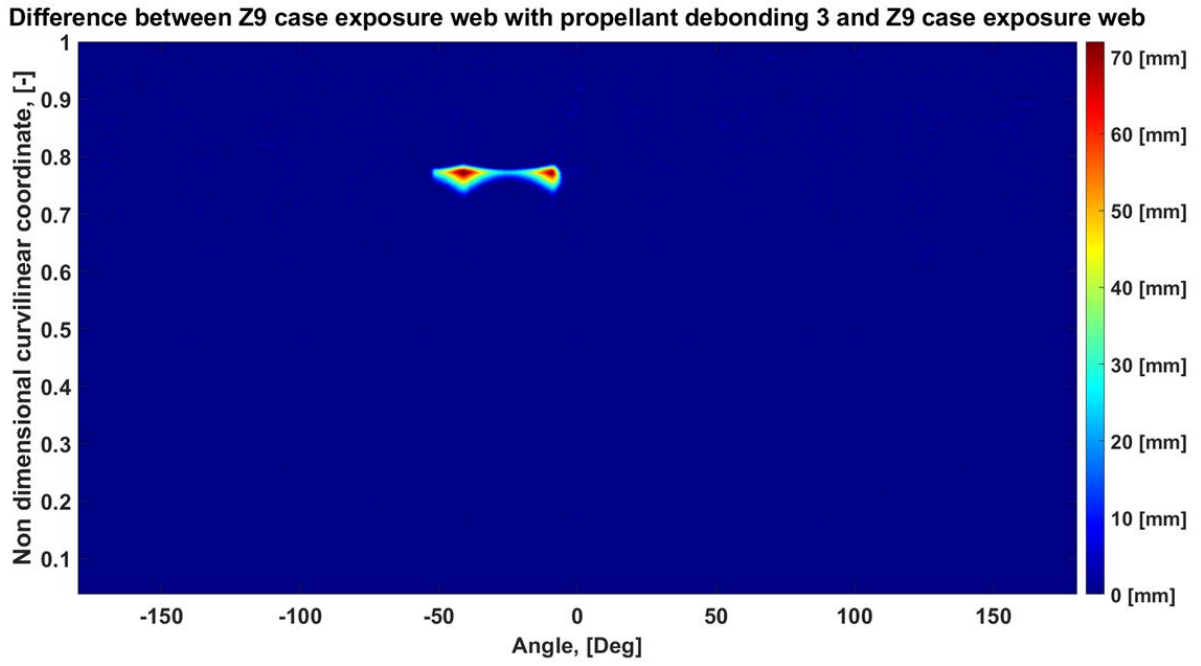
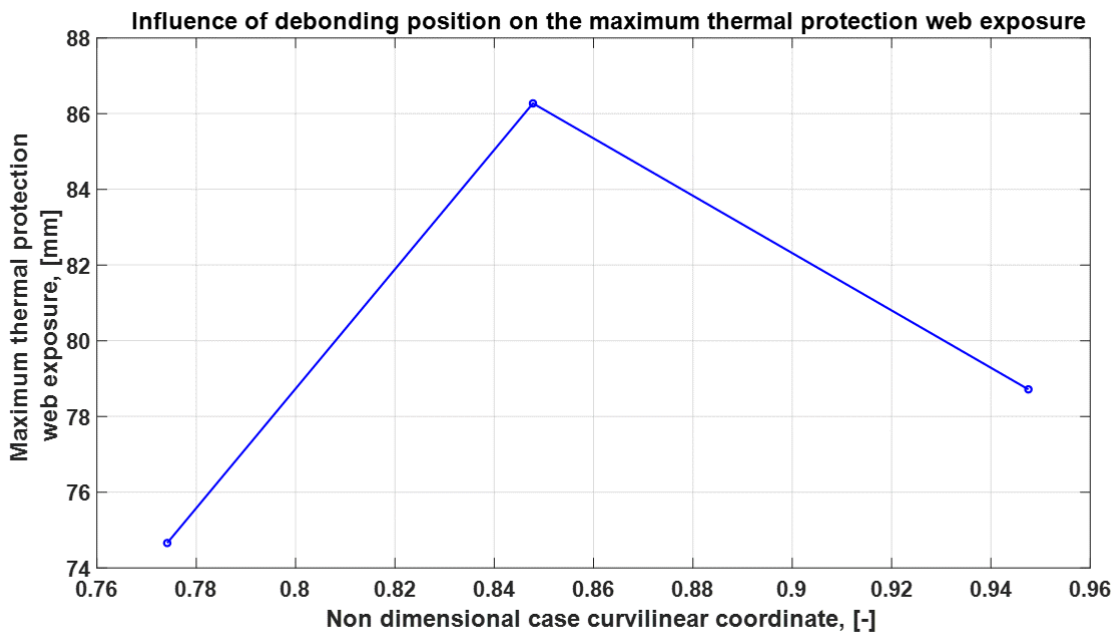**Figure 19: Thermal protection exposure web difference in debonding 3**

Web: 18.8%   Web: 25.6%   Web: 32%

a)   b)   c)
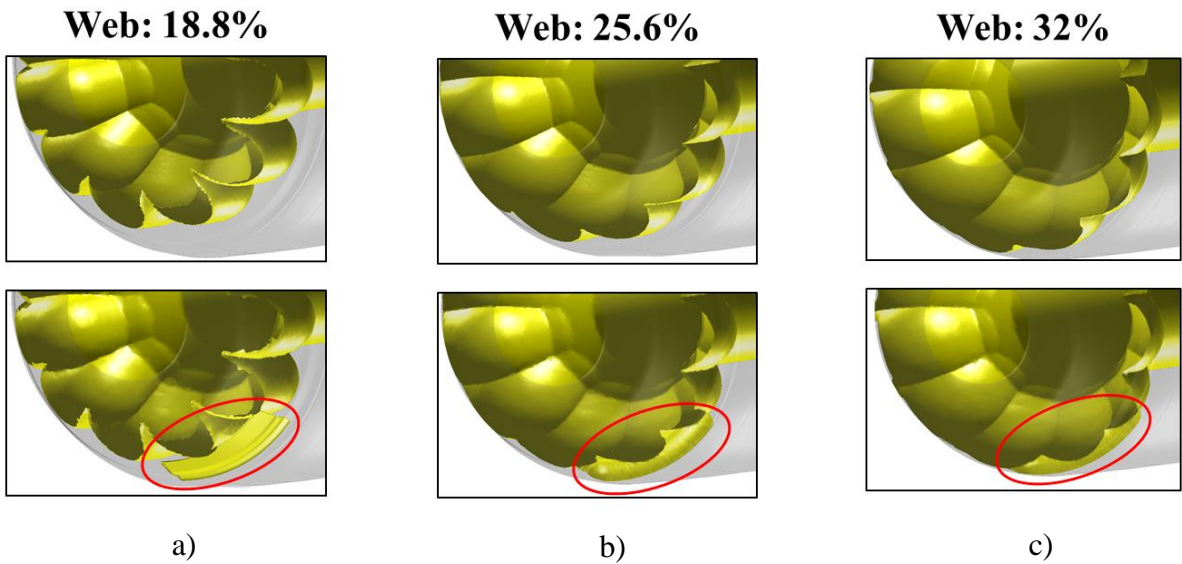
Figure 20: Comparison between the Z9 standard simulation and simulation with debonding 5
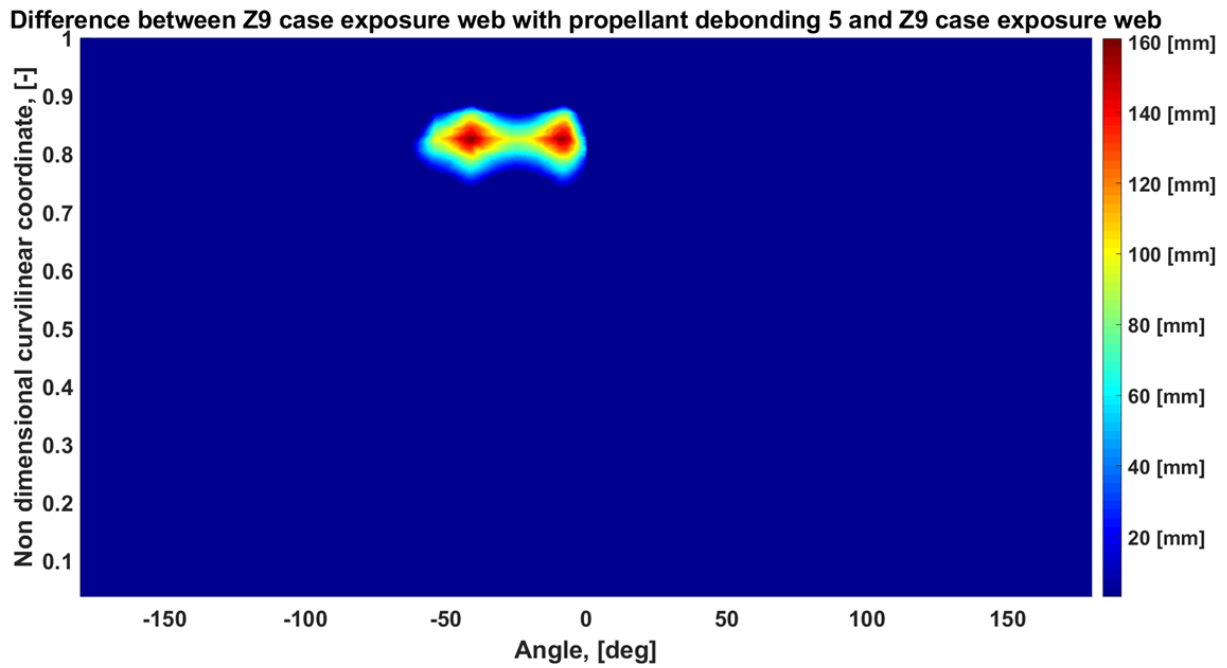
Figure 21: Thermal protection exposure web difference in debonding 5

Figure 22: Influence of the debonding position on the maximum thermal protection web exposure
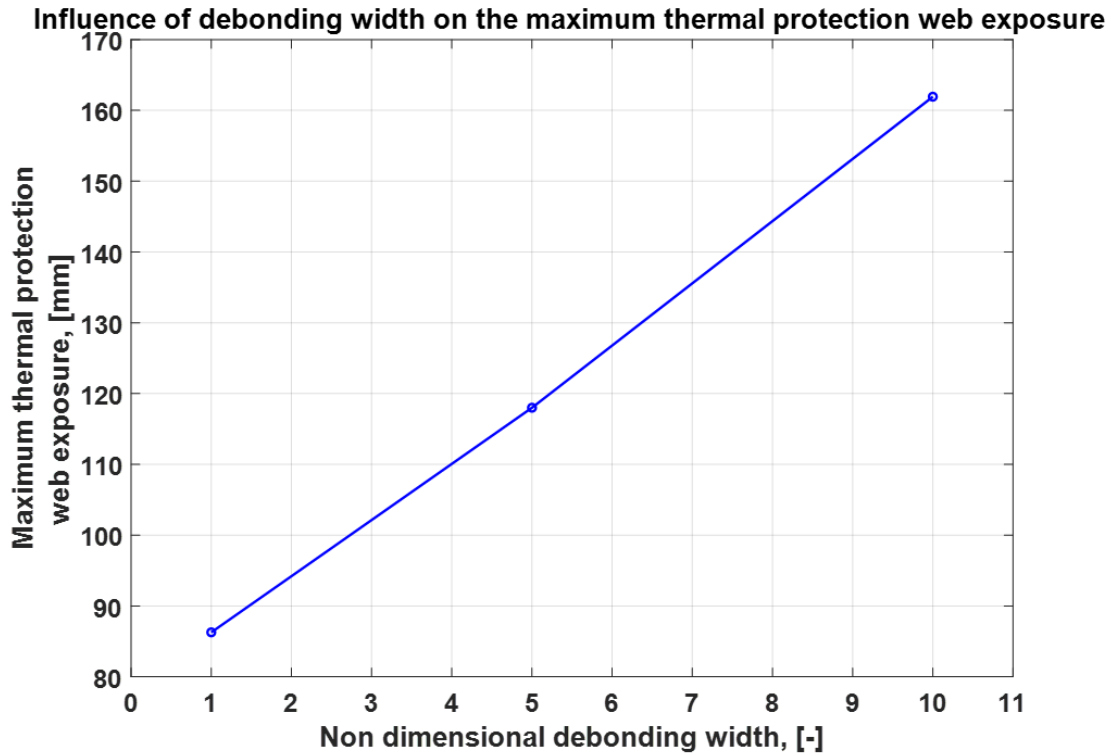
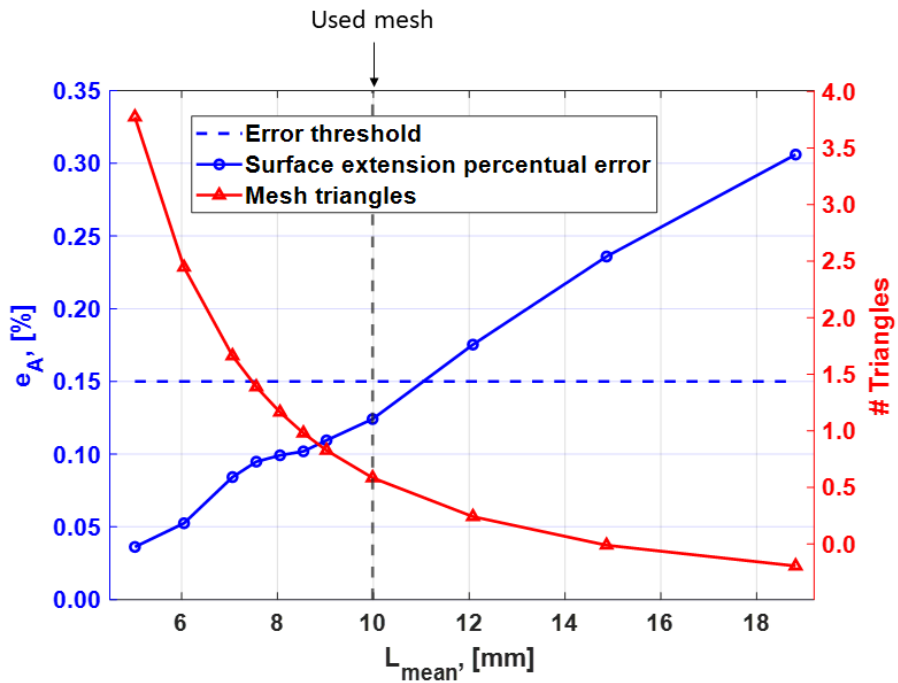**Figure 23: Influence of debonding width on the maximum thermal protection web exposure**

Figure 24: Initial burning surface area error