Numerical simulation of the friction stir welding process

Simon Guerdoux

To cite this version:

Simon Guerdoux. Numerical simulation of the friction stir welding process. Mechanics [physics.med-ph]. École Nationale Supérieure des Mines de Paris, 2007. English. NNT : 2007ENMP1518 . tel-00271234

HAL Id: tel-00271234

https://pastel.hal.science/tel-00271234

Submitted on 8 Apr 2008

MINES PARIS
ParisTech

ED n°364 : Sciences Fondamentales et Appliquées

N°attribué par la bibliothèque
|__|__|__|__|__|__|__|__|__|__|

T H E S E

pour obtenir le grade de
Docteur de l'Ecole des Mines de Paris
Spécialité " Mécanique Numérique "

présentée et soutenue publiquement par
M. Simon GUERDOUX

le 13 décembre 2007

SIMULATION NUMERIQUE DU SOUDAGE PAR FROTTEMENT-MALAXAGE
NUMERICAL SIMULATION OF THE FRICTION STIR WELDING PROCESS

Directeur de thèse : M. Lionel FOURMENT

Jury :

M. Han HUETINK	Examinateur
M. Jean-Philippe PONTHOT	Rapporteur
M. Guillaume RACINEUX	Rapporteur
M. Pierre VILLON	Examinateur
M. Michael MILES	Examinateur

A ma fille aussi…

ABSTRACT

Despite considerable interests in the Friction Stir Welding (FSW) technology in past decades, the basic physical understanding of the process is still insufficient. Clearly, the complete understanding of the material flow around the rotating tool is crucial to the optimization of FSW parameters (including tool rotation rate, traverse speed, spindle tilt angle, and target depth) and design of tool geometry. Numerical simulation, conducted on these aspects of the process, can so contribute to the increase in weld quality and productivity.

This work presents the development of a numerical tool. An Arbitrary Lagrangian Eulerian (ALE) formulation is implemented in the 3D FORGE3® F.E. software to simulate the different stages of the Friction Stir Welding (FSW) process. A splitting method is utilized: a) the material velocity/pressure and temperature fields are calculated, b) the mesh velocity is derived from the domain boundary evolution and an adaptive refinement criterion provided by error estimation, c) nodal and P0 variables are remapped. Different velocity computations and remap techniques are investigated, providing significant advantages with respect to more standard approaches. Improvement is also brought to the contact algorithm through a tool smoothing procedure. These proposed enhancements have been tested and applied on industrial cases. They allow for the entire FSW process simulation.

Steady state welding, but also transient welding phases are simulated, exhibiting good robustness and accuracy of the developed ALE formulation. On the first hand, friction parameters are identified using Eulerian steady welding state simulations by comparison with experimental results. On the second hand, one major interest of the ALE model being the possibility to simulate void formation at the tool/workpiece interface, the transient plunge and welding phases are modeled. Their simulations can thus help to better understand the mechanisms of the deposition process that occurs at the trailing edge of the probe in order to obtain sound and defect-free welds. Finally, the flexibility and robustness of the model allows the investigation of new tooling designs influence in the deposition process.

RESUME

Bien que le soudage par frottement malaxage ait suscité un intérêt croissant ces dix dernières années, les phénomènes physiques qui sont à la base de ce procédé sont encore mal connus. Clairement, la compréhension du flux de matière autour du pion de l'outil en rotation est crucial pour l'optimisation des paramètres du procédé (comprenant la vitesse de rotation, la vitesse d'avance, l'angle d'inclinaison de l'outil et sa profondeur de pénétration) et la conception de nouvelles géométries d'outils. La simulation numérique, portant sur ces aspects du procédé, peut ainsi contribuer à l'amélioration de la qualité des soudures et de leur productivité.

Ce travail présente le développement d'un outil numérique. Une formulation arbitrairement lagrangienne-eulérienne (ALE) est implémentée dans le logiciel 3D éléments finis FORGE3® pour simuler les différentes étapes du procédé de soudage par frottement-malaxage (FSW). Une méthode découplée est utilisée : a) les champs de vitesses, pressions et températures du matériau sont calculés, b) la vitesse de maillage est calculée à partir de l'évolution des frontières du domaine et d'un critère de raffinement adaptatif procuré via une estimation d'erreur, c) les variables nodales et P0 sont transportées. Différentes techniques de calcul de la vitesse de maillage et de transport des variables sont étudiées, apportant des avantages significatifs par rapport à des approches plus standard. L'algorithme de contact a également été enrichi par une procédure de lissage d'outil. Ces améliorations ont été testées et appliquées sur des cas industriels. Elles permettent la simulation du procédé complet de soudage FSW.

L'état stationnaire de soudage, tout comme les phases transitoires, sont simulés, montrant une bonne robustesse et une bonne précision de la formulation ALE développée. Dans un premier temps, la simulation de la phase de soudage stationnaire permet d'identifier, par comparaison avec des résultats expérimentaux, les paramètres de frottement. Dans un second temps, un des intérêts majeurs du modèle ALE étant la possibilité de simuler la formation de vide à l'interface outil/matière, la phase de plongée et des phases transitoires sont modélisées. Leurs simulations peuvent ainsi aider à mieux appréhender les mécanismes du phénomène complexe de déposition de matière qui doit avoir lieu à l'arrière du pion de façon à obtenir un joint de soudure correct, sans défaut. La flexibilité et la robustesse du modèle permettent enfin d'étudier l'influence de nouvelles formes d'outil sur ce phénomène.

REMERCIEMENTS

On y est…enfin! Après trois années, plus une bien chargée, riches en rencontres et en émotions… après un mariage… un bébé… deux déménagements… un nouveau travail… 27 rallyes et une finale… 3 pièces de théâtre… 2 groupes de musiques et 18 boeufs… 53 plombs… 12 randonnées… 36 agachons… 86 jibes…

Je garderai un excellent souvenir de mon passage au Cemef et je remercie tous ceux que j'ai eu la chance de côtoyer pendant ces quatre dernières années : merci pour tout ce que j'ai appris ou tout ce que j'ai pu recevoir!

Je voudrais particulièrement te remercier Lionel pour m'avoir accueilli, m'avoir fait confiance tout au long de ma thèse, m'avoir permis de participer à des congrès (parfois « paradisiaques »), m'avoir encadré avec toute l'humanité dont tu sais faire preuve, avec ta patience et ton humilité. Sans ton encadrement, ma date de soutenance serait encore loin d'être arrêtée...

Je voudrais ensuite remercier Lisebeth, qui nous a bénévolement encadré et mis en scène pendant plus de trois ans (et qui continuera longtemps je l'espère) et à la direction de l'école qui encourage ce genre d'initiative. Merci à mes amis « des planches », Mariane, Julien, Véro, Aurélie et Tof, Mathilde, Marc, Julia, Alexis, Lionel et tous ceux qui ont passé du temps, pour faire vivre ces ateliers « théâtre »…

Merci également à mes secrétaires préférées Marie-Françoise et Sylvie pour leur gentillesse et leur bonne humeur, à tous les permanents du Cemef, je pense notamment à Patrick pour son accueille et sa disponibilité, et merci à tout le staff enseignant pour leur temps qu'il consacre aux étudiants.

Merci aux anciens thésards (Olga, Josué, Sorin, Benjamin, Ramzy, Mehdi, Fredéric, …), et particulièrement à Julien, Hughes, et Louisa qui répondent toujours avec gentillesse et efficacité quand on les sollicite.

Merci aussi aux nouveaux thésards (Eli, Seb, Sabine, David, Makhlouf,…) que j'ai eu la chance de croiser et au personnel de Transvalor (Christine et Etienne) avec qui j'ai partagé les joies du développement et notamment à Sophie qui m'a permis de faire un rallye de plus...

Merci aux techniciens, Francis, Marc, et Gilbert pour leur système D et les sorties « vélo » dans la Valemasque, merci à Manu pour sa patience, son humour et sa jeunesse d'esprit. Merci à Evelyne, Suzanne et Monique, pour leur gentillesse, leurs conseils et leur encadrement lors de mon DEA.

Merci à Fabien, Omar, et tous les fouteux avec qui je me suis défoulé de temps à autre entre midi et deux…

Un grand merci à mes principaux collègues de bureau, Nadia, Rodolphe, Gregory, et Cyril, pour leur bonne humeur, les cafés et les clopes…et à ceux avec qui on les partageait : Manu, Sam, Christelle, Alex, Jésus La Grimpe, Ben, Isa, Magali.

I would like to thank also all the Brigham Young University students and staff (Mike, Carl, Stacy,…) for their reception in Provo, the trust that they brought to me and the snorkeling session in Kauaii.

Merci à toutes les personnes m'ont aidée de près ou de loin, parfois sans le savoir, en m'apportant leur conseil ou juste leur sourire.

Pour terminer je souhaite remercier ma famille, ainsi que ma belle famille préférée pour leur soutien permanent surtout pendant cette dernière année chargée… Merci à vous aussi : les Volpe, Méla, Antoine, tous les potos du cercle, Julien et Lann, sur qui je sais que je peux également compter!

Enfin merci à toi surtout, Amandine, d'être à mes côtés pour me soutenir dans les périodes difficiles, d'être là simplement pour partager tout ça avec moi. Je suis heureux d'être maintenant ton mari, et le père de notre fille !

TABLE OF CONTENTS

TABLE OF ILLUSTRATIONS

Chapter I : Introduction: The FSW process

1 *Process description*

1.1 Friction welding processes: advantages and potential uses

Friction Welding (FW) converts mechanical energy into heat and material deformation to create a weld. Currently, the parent material is not melt. Joining occurs in the solid state, avoiding possible metallurgical complications such as porosity, cracking or detrimental metallurgical changes.

One of the main advantages of the friction-based technologies is that thus enable joining material combinations which were previously difficult to weld. Furthermore, compared to traditional arc welding, FW produces little or no weld distortion, porosity or fume, and has excellent mechanical properties.

Rotary friction welding (Figure 1a) has been used for about fifty years in industry and is still the most widely used of the friction welding technologies. Non-rotary welding (linear, orbital and angular reciprocating motions -Figure 1b) has been a first major extension. It permits the joining of noncircular shapes such as squares and rectangular bars, which are very difficult to weld with rotary technology providing correct alignment. But the Friction Stir Welding (FSW) process has enlarged the application field of friction-based technologies since its invention in 1991 at The Welding Institute in England (Figure 1c).

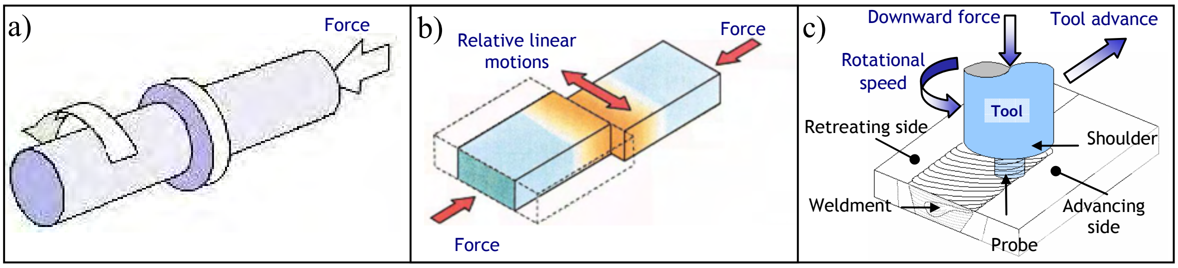


Figure 1: Friction welding processes: a) Rotary Friction Welding, b) Linear Friction Welding, c) Friction Stir Welding.

FSW has been successful in welding a wide range of materials such as plastics, metal matrix composites, aluminium, copper, steel, stainless steel, nickel alloys, and titanium. FSW has already been successfully implemented in various applications involving aluminium and copper in many industries including marine, aerospace, and railways.

The evolution of FSW equipment over the last decade has expended the potential of this innovative manufacturing technology and new developments in process application, such as underwater FSW of steel or stainless linepipes, have been proposed. The machines, which were originally converted milling machines, are currently designed with force control in mind. Improvements in the wear resistance, fracture toughness, and design of FSW tools continue to enhance the feasibility of FSW in a large variety of materials.

This recent friction stir technology can also be exploited to consolidate or refine the microstructure of a material. This kind of use also involves application of friction and pressure but as it does not imply creation of bond between two pieces of material, it is called Friction Stir Process (FSP).

In this manuscript we shall abusively use the abbreviations FSP or FSW without any distinction. The numerical modelisation of the FSW process does not take into account the interface between the two pieces of material to weld: therefore the simulation is basically equivalent to FSP simulation. The experimental weldings discussed and compared to numerical simulations in the last part of this manuscript have also been performed on a single plate.

1.2 Friction Stir Welding Process

The FSW process combines frictional heating and stirring motion to soften and mix the interface between two workpieces, yielding a solid, fully consolidated weldt.

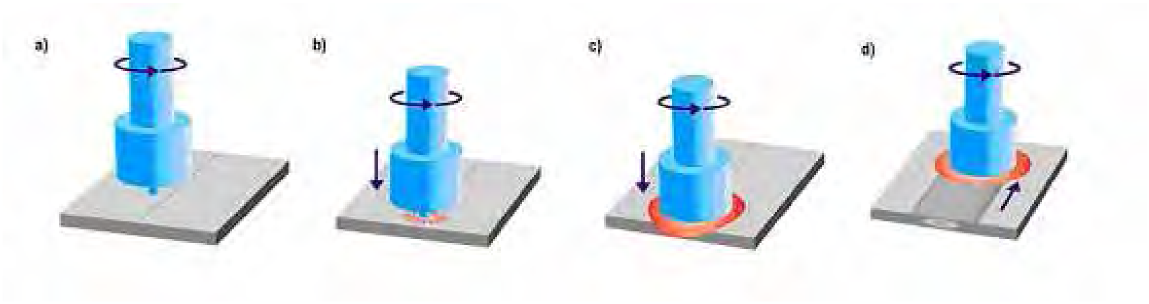


Figure 2 : Friction Stir Welding phases: a) Initial state b) the plunge, c) dwelling phase, d) welding phase

First, parts are mated together, rigidly clamped (Figure 2a). A rigid cylindrical spinning tool, composed by a protruding pin and a larger concentric shoulder, is plunged into the joint line until the shoulder rests on the surface: this is the plunging phase (Figure 2b). The dwelling phase (Figure 2c) follows and rapidly softens the material through frictional heating. Then the tool is advanced along the joint line: the material is plastically deformed, stirred around the tool pin in a complex movement depending on forces involved and tool design (Figure 2d). This welding phase provides a solid, fully consolidated weldment between the matted parts.

Temperatures in the tool and workpiece are often close to the solidus temperature of the workpiece (3-8 kW of mechanical power are converted into heat during each weld of aluminium alloys). A complete understanding of the FSW process through thermo-mechanical simulation requires carefully considering both workpiece and tool.

The thermo-mechanical process involved under the tool results in different microstructural regions (see Figure 3). Some are common to all forms of welding, others are unique to the technique.

The *stir zone* (also called nugget, the dynamically recrystallised zone) is a region of heavily deformed material that roughly corresponds to the location of the pin during welding. The grains within the stir zone are roughly equiaxed and often an order of magnitude smaller than the grains in the parent material. A unique feature of the stir zone is the common occurrence of several concentric rings, which has been referred to as an ‘onion-ring’ structure. The precise origin of these rings has not been firmly established, although variations in particle number density, grain size and texture have all been suggested.

The *flow arm* is on the upper surface of the weld and consists of material that is dragged by the shoulder from the retreating side of the weld, around the rear of the tool, and deposited on the advancing side.

The *thermo-mechanically affected zone* (TMAZ) occurs on either side of the stir zone. In this region, the strain and temperature are lower and the effect of welding on the microstructure is correspondingly smaller. Unlike in the stir zone, the microstructure is recognizably that of the parent material, albeit significantly deformed and rotated. Although the term TMAZ technically refers to the entire deformed region, it is often used to describe any region not already covered by the terms stir zone and flow arm.

The 'heat affected zone' (HAZ) is common to all welding processes. As indicated by the name, this region is subjected to a thermal cycle but is not deformed during welding. The temperatures are lower than those in the TMAZ but may still have a significant effect if the microstructure is thermally unstable. In fact, in age-hardened aluminium alloys this region commonly exhibits the poorest mechanical properties.

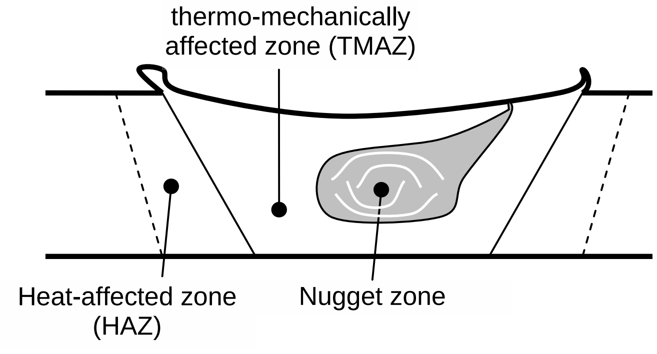


Figure 3 : Illustration of the different microstructural regions in a transverse cross section of the weldment

1.2.1.1 Process parameters

For FSW (Figure 4), two parameters are very important: tool rotation rate (ω, rpm) in clockwise or counterclockwise direction and tool traverse speed (v, mm/min) along the line of joint. Rotation of the tool results in stirring and mixing of material around the rotating pin, and the translation of tool moves the stirred material from the front to the back of the pin, and finishes the welding process. Higher tool rotation rates generate higher temperature because of higher friction heating, and result in more intense stirring and mixing of material. However, it should be noted that friction between tool and workpiece governs the heating, and that the coefficient of friction at the interface will change with increasing tool rotation rate. So, a monotonic increase in heating with increasing tool rotation rate is not expected.

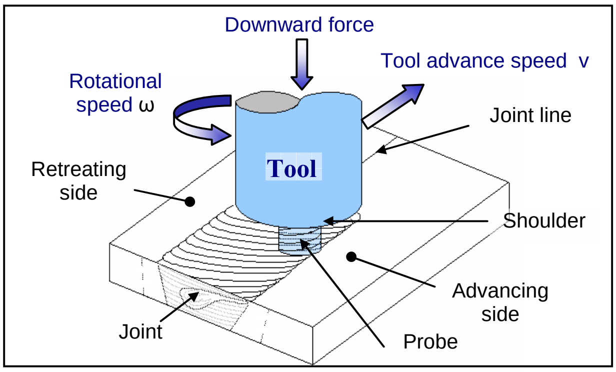


Figure 4 : Illustration of the Friction Stir Welding phase

In addition to the tool rotation rate and traverse speed, another important process parameter is the insertion depth of the pin into the workpiece (also called target depth). This parameter is related to the pin length. When it is not enough, the tool shoulder does not correctly contact the original workpiece surface. Thus, it cannot move the stirred material efficiently from the front to the back of the pin, resulting in generation of welds with inner channels or surface grooves. When the insertion depth is too deep, the tool shoulder plunges into the workpiece creating excessive flashes. In this case, a concave weld is produced, with a local thinning of the plates in the welded zone.

Further, the angle of the spindle, or tool tilt, with respect to the workpiece surface is important for producing sound welds with smooth tool shoulders. A suitable tilt of the spindle towards trailing direction ensures that the shoulder of the tool holds the stirred material by the threaded pin and moves material efficiently from the front to the back of the pin. It should be noted that the recent development of 'scrolled' tool shoulder allows FSW without tool tilt (see illustration in Figure 5). Such tools are particularly preferred for curved joints.

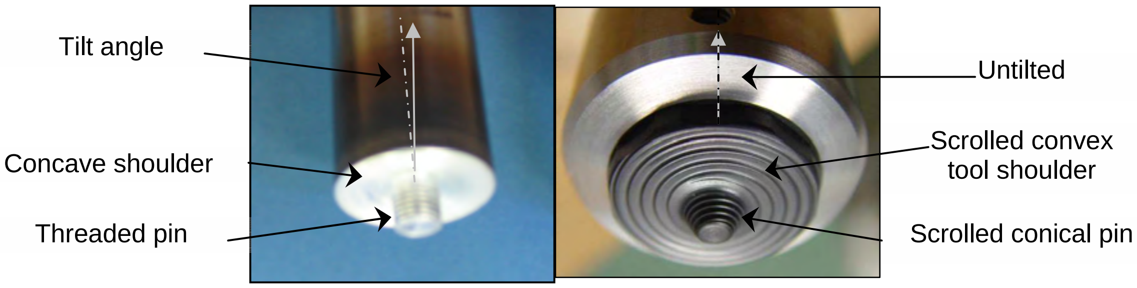


Figure 5 : Illustration of two different tool shapes.

1.2.1.2 Tool geometry

Tool geometry is the most influential aspect of process development. The tool geometry plays a critical role in material flow and consequently governs the traverse rate at which FSW can be conducted. As mentioned earlier, the FSW tool has two primary functions: to produce a localized heating and a material flow.

In the initial stage of the tool plunge, the heating mainly results from the friction between the pin and the workpiece. Additional heat is produced by the deformation of the material. The tool is plunged till the shoulder touches the workpiece. The friction between the shoulder and workpiece results in the largest component of heating. From the heating standpoint, the relative size of pin and shoulder is the most important, and the other design features are not critical. The shoulder also provides confinement for the heated volume of material.

The second function of the tool is to ‘stir’ and ‘move’ the material. The uniformity of microstructure and properties, as well as process loads, are governed by the tool design. At present, concave shoulder and threaded cylindrical pin are generally used.

With increasing experience and some improvements in understanding the material flow, the tool geometry has evolved significantly. New complex design features are regarded to (a) reduce welding force, (b) enable easier flow of plasticized material, (c) facilitate the downward augering effect, and (d) increase the interface between the pin and the plasticized material, thereby increasing heat generation.

The tool geometry has a significant effect on the metal flow. The correlations between the material flow and the resultant microstructure of welds varies with each tool design. The generalization of microstructural development and influence of processing parameters is difficult in absence of the tool information. Therefore, a critical need is to develop a systematic framework for tool design. In friction stir welding conferences, several companies have indicated internal R&D efforts but no open literature is available on such efforts and outcomes.

Computational tools, including finite element analysis (FEA), can be used to visualize the material flow and calculate axial forces. However any generalization should be treated carefully because the material flow is very significantly influenced by the tool design, while most studies do not report the utilized tool design and all process conditions. Therefore, the differences observed between various studies cannot be easily compared.

1.2.1.3 Microstructure and texture

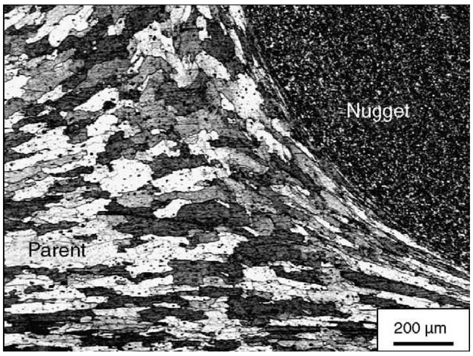


Figure 6: Example of microstructure observed in TMAZ and HAZ

The solid-state nature of FSW, combined with its unusual tool and asymmetric features, result in a highly characteristic microstructure. Some regions are common to all kinds of welds but some are quite unique to the technique.

Texture influences a variety of properties, including strength, ductility, formability and corrosion resistance. As mentioned earlier, the FSW material consists of distinct microstructural zones (see Figure 6), i.e., nugget, TMAZ, HAZ and base material. Each zone has a different thermo-mechanical history. What is even more complicated in FSW is that the nugget zone consists of several sub-domains. For example, the top layer undergoes a specific deformation by the shoulder after the pin has passed through it. On the other hand, depending on the tool rotation rate and traverse speed, the nugget region can have an onion ring pattern or other microstructural alterations. In the last decade, the use of orientation imaging microscopy (OIM) has shown to be a very valuable tool, not only for obtaining the texture information, but also for establishing the grain boundary misorientation distribution from same set of experiments.

1.2.1.4 Residual stresses

During fusion welding, complex thermal and mechanical stresses develop in the weld and the surrounding region due to the localized application of heat and resulting constraint. The residual stresses usually reach the yield strength value of the base material. On the other hand, it is generally considered that residual stresses are much lower in friction stir welds due to much lower temperatures. However, compared to rather compliant clamps used for fixing the parts in conventional welding, the very rigid clamping used in FSW exerts a much higher restraint on the welded plates. These restraints impede the contraction of the weld nugget and heat-affected zone during cooling in both longitudinal and transverse directions, thereby resulting in generation of longitudinal and transverse stresses. High values of residual stresses have a significant effect on the postweld mechanical properties, particularly the fatigue ones. Therefore, it is of practical importance to carefully investigate the residual stress distribution in the FSW welds.

2 *Motivation and problem statement*

2.1 Motivation and numerical objectives

FSW is still a recent and complex process. Sound welds are achieved only for specific combinations of the process parameters. The particular phenomena, which are responsible of the deposition of material that occurs at the trailing edge of the probe, are not completely understood yet. Many experimental investigations have already been conducted to adjust input parameters (spindle speed, feed rate, and tool depth), contrary to numerical investigations, which have been scarcely used for these issues.

Firstly computational tools could be helpful to better understand and observe the influence of input parameters on resulting phenomena. Visualization and analysis of the material flow, temperature map, stresses and strains involved during the process is much easier in simulations than in experiments.

Moreover, the process results in significant microstructural evolution, including grain size, grain boundary character, dissolution and coarsening of precipitates, break-up and redistribution of dispersoids, and texture. The numerical modelisation can provide mechanical and thermal histories of material particles, which are necessary to compute final microstructure of the weld. Thus the numerical tool can also be helpful for understanding and controlling the final microstructure and properties of welds.

Therefore and secondly, predictive simulations could help to adjust and optimize the process parameters and tool designs in order to achieve the best weld properties, increase welding rates and tool life, and enlarge the application field of the process by reducing stresses on FSW tool.

The main objective of this work is to develop a 3D robust numerical tool enabling the thermo-mechanical simulation of the whole FSP (plunging phase, dwelling phase and welding phase). As numerical simulation has to be as predictive as possible, the accuracy of the developed methods is crucial.

2.2 Difficulties and background literature

FSW results in intense plastic deformation and temperature increase within and around the stirred zone. It induces complex movement of material, large shear forces in the plastically deformed material, raising the temperature of the material to about 80% of the melting temperature [1]. Therefore, this process is highly thermo-mechanically coupled and presents a formidable challenge for researchers attempting to characterize these phenomena through various modelling techniques.

First models reported in the literature for FSW were analytical models addressing temperature analysis. Initially based on Rosenthal's equations, they describe the quasi-steady temperature field of a semi-infinite body on which is applied a surface heat source moving with constant velocity. The heat source really produced around the tool is replaced by simple analytical solutions [2]. McClure et al. [3], along with Gould and Feng [4] have incorporated in their equations the term of frictional heat assuming a constant uniform pressure between the tool and the part. Figure 7 shows the temperature field described by the analytical model of Gould and Feng. The traverse speed of the tool, as the rotational velocity ω and the loading force F are the three parameters of the model.

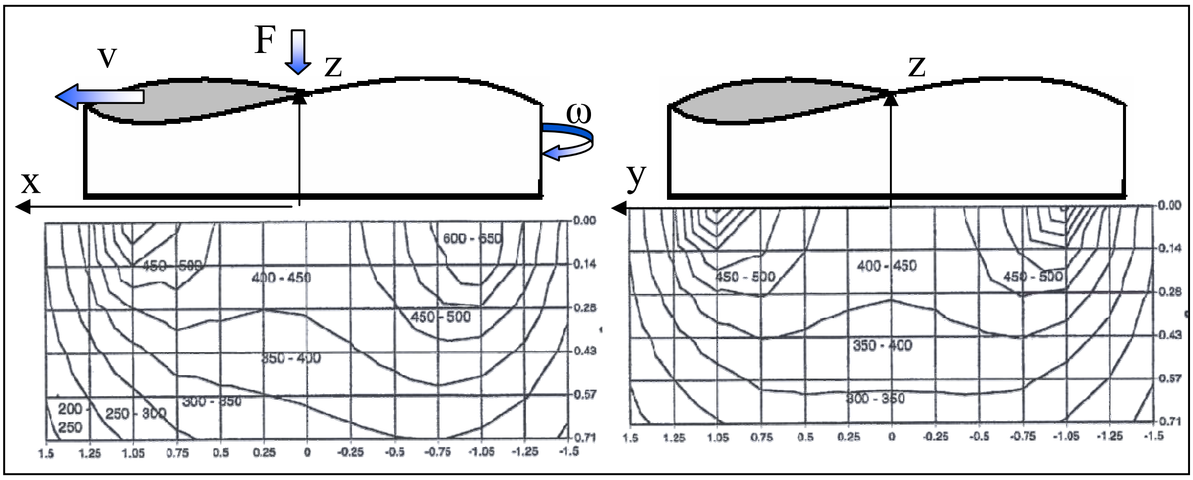


Figure 7: Longitudinal (leftside) and transverse (rightside) prints of the temperature field calculated with the analytical model of Gould and Feng [1].

An expression of the estimated analytical heat generation, developed by Schmidt et al [5] for more complex tool shapes is the following:

$$Q_{sliding,ana} = \frac{2}{3}\mu\omega\frac{F_{exp}}{R^2_{shoulder}}\left(\left(R^3_{shoulder} - R^3_{probe}\right)\left(1+\tan\alpha\right) + R^3_{probe} + 3R^2_{probe}H_{probe}\right) \qquad \text{(I-1)}$$

R_{probe} and $R_{shoulder}$ are respectively the radius of the probe and the shoulder of the tool. The pin length is given by H_{probe}. F_{exp} is the plunge force experimentally observed. μ is the Coulomb friction coefficient. α is the cone angle of the tool shape. Thereafter, the convective heat transfer due to the material flow in the shear layer have been taken into account (using sticking conditions in a simplified way) by prescribing a velocity boundary condition for the convective term in the energy equation. Kandkar *et al* [6] introduced a torque based heat input model in which the local heat flux $\dot{q}$ is linearly related to the distance to the tool axis r as follows:

$$\dot{q}(r) = \frac{r.M_{tot}.\omega}{\frac{2}{3}\pi R^3_{shoulder} + 2\pi R^2_{probe}H_{probe}} \quad \text{with } M_{tot} = \frac{P_{av}}{\omega} \qquad \text{(I-2)}$$

M_{tot} is the total torque and P_{av} is the average power.

All these models constitute first tools to approximate the temperature map or the heat source in FSW. However, they are based on strong assumptions concerning the contact (sliding or sticking conditions are assumed) and always need the adjustment of experimental coefficients to be predictive. Moreover, these models have low flexibility (tool design and transient phases of the FSP are not taken into account) and they only provide thermal informations. Their major interest is in feeding Finite Element (FE) solid-mechanics models with temperature fields.

Finite Element (FE) solid-mechanics models have been logically developed after analytical ones. The high distortion of the mesh, when Lagrangian formulation is used, is the main difficulty that Finite Element (FE) solid-mechanics models have to deal with. They occur under the tool shoulder, where high strains are logically observed during FSW. Therefore, the FSW tool is usually substituted by an analytical heat source. This approximation, widely used, allows for compution of residual stresses that are due to thermal distortions. Figure 8 shows the residual stresses provided by the model of Chen and Kovacevic in the code ANSYS [7], when the heat source is assumed to be symmetric. Lawrjaniec et al. [8] have developed two models using the thermomechanicaly coupled FE codes SYSWELD and MARC. 2D surface heat sources or 3D heat sources based on Russel's equations [9] are assumed taking into account the non-symmetrical character of the process. In WELDSIM, Chao et al. [10] have modelled the mechanical action of the tool as a uniform forging pressure. A Coulomb friction law is assumed and the frictional heat source due to the rotational movement of the tool is taken into account. But none of these FE solid-mechanics based models enable the material flow simulation.

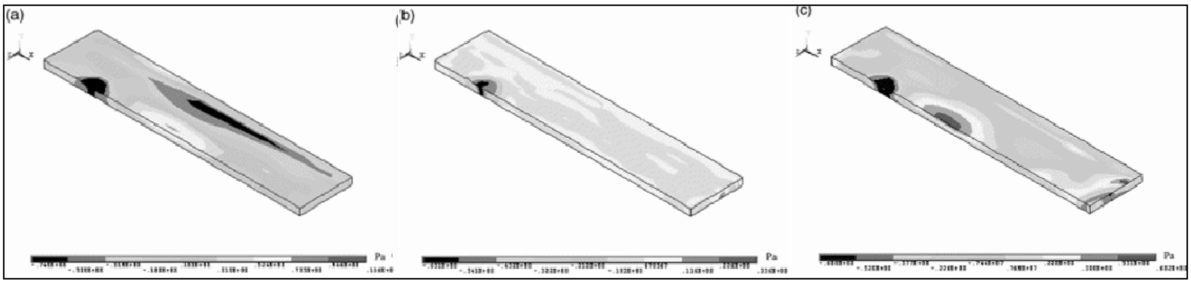


Figure 8: Stresses observed during welding with the FE model of Chen in the three main directions.

The simulation of material flow during FSW requires robust treatment of large deformations. Thus, at first it has been logically modelled using Computational Fluid Dynamics (CFD) models. Indeed, CFD Software can be easily adapted to FSW process: the material, regarded as a viscous fluid, flows across the Eulerian grid and interacts with the rotating tool. The inlet material velocity prescribed at the boundary of the domain corresponds to the traverse speed of the tool. Figure 9 shows the model developed by Ulysse [11]. The contact is assumed sticking and thus, the heat source is only due to the viscous dissipation of the laminar flow [12]. Seidel [9] has approximated the viscosity by a function of the material yield stress throughout the Zener-Hollomon parameter. Its model has been implemented in the 3D code FLUENT. Colegrove et al. have [13, 14] also used a CFD code to develop a global thermal model, in which the heat flow (applied at the interface FSW tool/workpiece) includes terms due to material shearing and friction. They have also developed a local model to visualize the influence of the screw on the material flow around the pin.

One of the major drawbacks of these CFD models is the approximation usually resulting on the material rheology, which does not allow for residual stresses prediction. In addition, simulation of transient phases of the process is hardly possible with such Eulerian description of the grid.

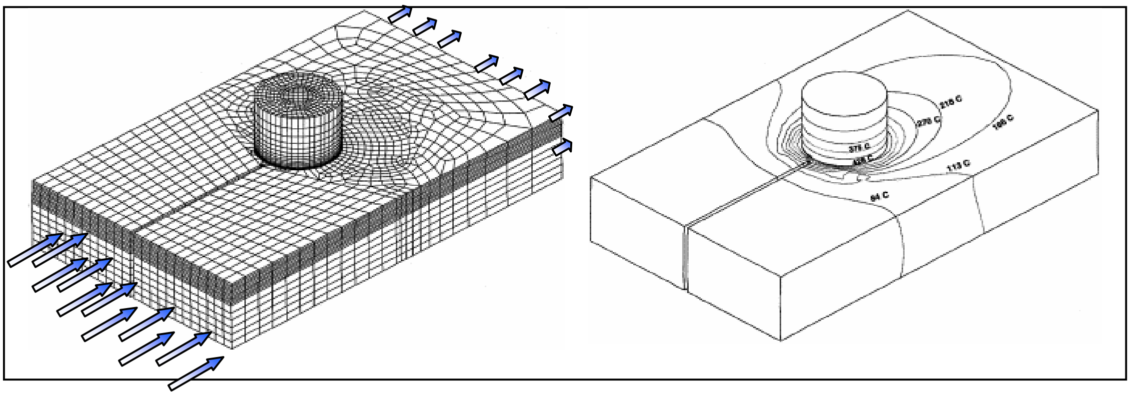


Figure 9: Ulysse's Eulerian model with coupled thermal computation in tools

Few models have actually taken into account the unilateral contact conditions [15]; they do enforce the contact condition between tool and workpiece, thus prohibiting void formation. The model of Schmidt et al. [16] is currently the only one in literature which handles this condition and so enables simulating under which conditions the material deposition process is successful. It has been developed within the Abaqus Explicit software, and utilizes an Arbitrary Lagrangian Eulerian (ALE) formulation. The remeshing algorithm provides an Eulerian character to the formulation that allows free surface movements to be modeled. The welding speed is prescribed as an inflow velocity at the inlet Eulerian surface which corresponds to the half-circumference of the discretized domain. The plunge phase is not simulated, but a spring force is applied to the tool. It corresponds to the experimental machine stiffness. The initially prescribed plunge depth is automatically provided by the spring back of the tool which settles at a value modelled close to the experimentally found one. The elasticity of the material is modeled by an elastic–plastic Johnson–Cook constitutive equation. The contact is bilateral during the period of time necessary to reach a steady state and unilateral (so enabling void formation) afterwards, in order to verify that the material deposition process correctly occurs. A Coulomb friction law is assumed in this local fully coupled 3D model.

2.2.1.1 Summary

This short bibliographic study highlights several difficulties to simulate FSW. In this process, heat sources result from two different phenomena. In the first hand, heat is generated by friction at the tool/workpiece interface, and in the second hand by energy dissipation during plastic deformation of the material under the tool. The thermo-mechanical coupling is strong so the numerical model has to accurately take into account these two terms. The heat source is three dimensional, asymmetrical, and most importantly, dependent of the contact area. Therefore analytical models can only provide a first approximation of the thermal field during the steady phase of the process. The description of the contact area, taking into account the complex tool shape, is crucial.

The thermal boundary conditions are also important to properly model the heat dissipation and the accuracy of the simulation logically requires coupled computations inside the FSW tool and in the backing plate.

The major difficulty of simulating the FSP transient phases consists in dealing with the large deformations, which are involved under the tool, while describing with enough accuracy little surface movements in the contact area. The complex material flow generates large distortions of the mesh when a Lagrangian description is utilized. Thus complex remeshing procedures are required. The Eulerian formulation more easily provides the history of the material flow during stationary welding. Nevertheless, the transient events, such as the phenomenon of filling or of prevailing cavity behind the probe of the tool, are hardly simulated because of the difficulty to track free surface movements.

The ALE description makes it possible to takes into account movements of free surfaces while reducing mesh distortions. This formulation looks procuring the best compromise between advantages and drawbacks of Lagrangian and Eulerian formulations.

Therefore, it seems to be the most adapted description for simulating steady and transient steps of the process. It enables studying one of the less understood aspects of FSW: the condition under which the deposition of material behind the tool is successful. This success depend on an adequate combination of the machining parameters: advancing speed, rotational speed, and depth of the tool inside workpiece. Thus, the model should adequately traduce these parameters in terms of frictional forces and forging pressure.

The keypoint of the present work is developing a robust ALE formulation in the code Forge3 (initially Lagrangian) in order to deal with the different aspects of FSW simulation which have just been presented.

Chapter II : Numerical Problem

In the present study, the FORGE3® code has been modified in order to simulate the different phases of FSP. This commercial software is extensively used to simulate hot, warm and cold forging of 3D geometry parts. This solid-mechanics code uses a Lagrangian finite element formulation to solve the thermal and mechanical equations at each time step of the simulated process. These equations are detailed in the first half of this chapter, which also presents global different constitutive equations and boundary conditions used to model the process. The second half is m4.1ore dedicated to their resolution.

1 Mechanical Equations

The mechanical problem is based on a set of two physical equations, which are numerically integrated.

1.1 Continuity Equation

The continuity equation states that mass cannot be lost or gained in time. This is mathematically expressed in equation (II-1).

$$\frac{dm}{dt} = 0 \tag{II-1}$$

m is the mass of a given volume Ω. The integration of equation (II-1) gives:

$$\frac{dm}{dt} = \frac{d}{dt}\left(\int_{\Omega} \rho(x,t)\,dV\right) = 0 \tag{II-2}$$

ρ is the density. Finally, the differentiation of equation (II-2) leads to the continuity equation:

$$\frac{\partial \rho}{\partial t} + \mathrm{div}(\rho\,v) = 0 \tag{II-3}$$

If we consider only rigid plastic materials (so neglecting elasticity effects), the time derivative can be considered as null and the incompressibility condition is then written for as:

$$\mathrm{div}(\mathbf{v}) = 0 \tag{II-4}$$

Other wise one have:

$$\mathrm{tr}(\dot{\varepsilon}_{pl}) = 0 \tag{II-5}$$

1.2 Motion Equation

The general form of the motion equation is expressed in equation (II-6) as a force balance, including dynamic forces (inertia), static forces, and gravity.

$$\rho.\boldsymbol{\gamma} = \mathbf{div}(\boldsymbol{\sigma}) + \rho.\mathbf{g} \tag{II-6}$$

$\boldsymbol{\gamma}$ is the acceleration, $\boldsymbol{g}$ is the gravity acceleration and $\boldsymbol{\sigma}$ the stress tensor of Cauchy.

Gravity and inertia are often neglected in most static force problems.

1.3 Boundary conditions

The boundary of the mechanical domain Ω is called ∂Ω. It can be divided into several distinct boundary conditions resulting in the definition (II-7) (see Figure 10).

$$\partial\Omega = \partial\Omega_C + \partial\Omega_L + \partial\Omega_V + \partial\Omega_T \tag{II-7}$$

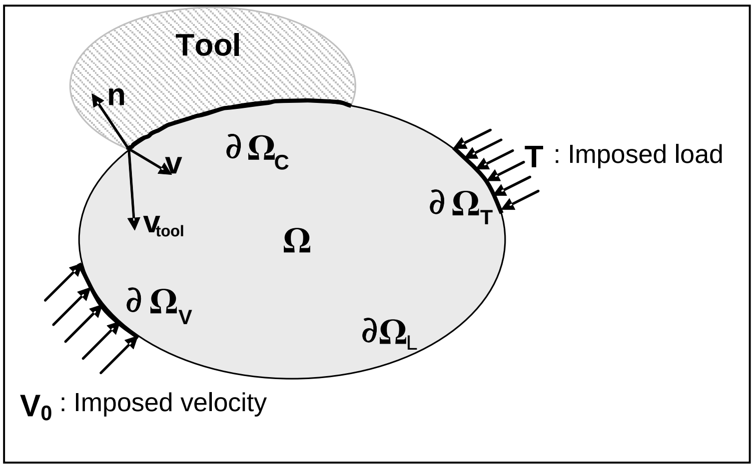


Figure 10: Mechanical boundary decomposition of the domain

- On $\partial\Omega_L$, the free surface conditions impose that the normal stress is null:

$$\boldsymbol{\sigma}.\mathbf{n} = \mathbf{0} \tag{II-8}$$

where n is the normal out warding the surface.

- On $\partial\Omega_T$, imposed loading conditions force to have equation (II-9).

$$\boldsymbol{\sigma}.\mathbf{n} = \mathbf{T} \tag{II-9}$$

- On $\partial\Omega_V$, the velocity is imposed, so we have equation :

$$\mathbf{v} = \mathbf{v}_0 \qquad \text{(II-10)}$$

- On $\partial\Omega_C$, two kinds of conditions are imposed due to contact:

- a non penetration condition in the normal direction, given by the Signorini equations:

$$\begin{aligned} (\mathbf{v} - \mathbf{v}_{\text{tool}}).\mathbf{n} &\leq 0 \\ \sigma_n &\leq 0 \\ \left[(\mathbf{v} - \mathbf{v}_{\text{tool}}).\mathbf{n}\right]\sigma_n &= 0 \end{aligned} \qquad \text{(II-11)}$$

$\mathbf{v}_{\text{tool}}$ is the displacement of the tool; $\sigma_n = (\boldsymbol{\sigma}.\mathbf{n}).\mathbf{n}$ is the contact pressure.

Note: The notation **.** is used both for matrix or scalar products.

- and a friction condition in the tangential direction, imposing the boundary shear stress $\boldsymbol{\tau}_f$:

$$\boldsymbol{\tau}_f = \boldsymbol{\sigma}.\mathbf{n} - \sigma_n.\mathbf{n} \qquad \text{(II-12)}$$

$\boldsymbol{\tau}_f$ depends on the friction equations introduced in paragraph 2.2.

2 *Modelling the problem*

2.1 Constitutive models

2.1.1 Definitions

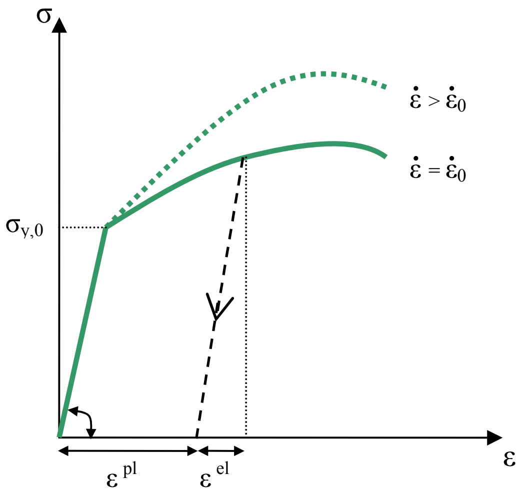


Figure 11: Example of stress-strain curves for a one-dimensional tensile test

The constitutive equations define the relation between stress $\boldsymbol{\sigma}$ and strain $\boldsymbol{\varepsilon}$. It is generally based on experimental observations. The type of employed constitutive model depends on the material under investigation, and the applied loads. It is also possible that the behaviour depends on the strain rate $\dot{\boldsymbol{\varepsilon}}$ (see Figure 11) and the temperature T. A general expression is given by the following equation:

$$\boldsymbol{\sigma} = \boldsymbol{\sigma}(\boldsymbol{\varepsilon}, \dot{\boldsymbol{\varepsilon}}, \mathbf{T}, \mathbf{P}) \quad \text{(II-13)}$$

P is a set of coefficients called rheological parameters which are used in the constitutive equation.

The stress tensor is split into a deviatoric part s, called deviatoric stress, and a spherical one pI, as follows:

$$\boldsymbol{\sigma} = \mathbf{s} - p\mathbf{I} \quad \text{(II-14)}$$

The hydrostatic pressure p is defined by the equation (II-15).

$$p = -\frac{1}{3}\mathrm{tr}(\boldsymbol{\sigma}) \quad \text{(II-15)}$$

The strain rate tensor is defined as:

$$\dot{\boldsymbol{\varepsilon}} = \frac{1}{2}\left(\nabla.\mathbf{v} + \mathbf{v}.\nabla \right) \quad \text{(II-16)}$$

One-dimensional representation of stress, and strain rate are respectively given by the equivalent Von Mises Stress tensor (equation (II-17)) and the equivalent strain rate (equation (II-18)).

$$\overline{\sigma} = \sqrt{\frac{3}{2}\mathbf{s} : \mathbf{s}} \quad \text{(II-17)}$$

$$\dot{\overline{\varepsilon}} = \sqrt{\frac{2}{3}\dot{\boldsymbol{\varepsilon}} : \dot{\boldsymbol{\varepsilon}}} \quad \text{(II-18)}$$

The equivalent strain is defined by the time integration of the equivalent strain rate:

$$\overline{\varepsilon} = \int_0^t \dot{\overline{\varepsilon}}\, dt \quad \text{(II-19)}$$

For hot processes, the elastic strain is often neglected and the behaviour of the metal is often modelled by a viscoplastic constitutive law. On the other hand, elastic strain can not be neglected in cold processes, therefore elastoplastic, elasto-viscoplastic, or pure elastic laws can be considered.

A strong gradient of temperature takes place in FSW process. The material is highly softened under the tool and has a viscoplastic behaviour in this area. Nevertheless, an elasto-viscoplastic constitutive law has to be considered in order to accurately simulate the plunging phase and the global matrix response in the far-field. This latter is crucial to observe loading effects and compute residual stresses, which appear during the cooling of the workpiece.

An additive Prantl-Reuss decomposition of the strain rate into elastic and plastic parts is assumed (equation (II-20) and Figure 11).

$$\dot{\boldsymbol{\varepsilon}} = \dot{\boldsymbol{\varepsilon}}^{\mathrm{el}} + \dot{\boldsymbol{\varepsilon}}^{\mathrm{pl}} \tag{II-20}$$

2.1.2 Elasticity

Elasticity characterizes a reversible linear behaviour. The Hooke's law (equation (II-21)) is representative of a linear elastic and isotropic material rheology. Its time derivative is given as:

$$\left|\begin{array}{l} \dot{\boldsymbol{\sigma}} = \mathbf{D}\dot{\boldsymbol{\varepsilon}}^{\mathrm{el}} = 2\,\mu\,\dot{\boldsymbol{\varepsilon}}^{\mathrm{el}} + \lambda\,\mathbf{tr}\left(\dot{\boldsymbol{\varepsilon}}^{\mathrm{el}}\right) \\ \mu = \dfrac{\mathrm{E}}{2(1+\nu)} \quad \mathrm{et} \quad \lambda = \dfrac{\mathrm{E}\,\nu}{(1+\nu)(1-2\nu)} \end{array}\right. \tag{II-21}$$

λ and μ are the Lamé coefficients, which are constant for homogenous material (and linearly time dependent). E is the Young modulus, ν is the Poisson coefficient, $\dot{\varepsilon}^{\mathrm{el}}$ is the elastic strain rate and $\dot{\sigma}$ is the time derivative of stress.

The inverse form of this Hooke law is written as follows:

$$\dot{\boldsymbol{\varepsilon}}^{\mathrm{el}} = \mathbf{D}^{-1}\dot{\boldsymbol{\sigma}} = \frac{1+\nu}{\mathrm{E}}\dot{\boldsymbol{\sigma}} - \frac{\nu}{\mathrm{E}}\mathrm{tr}(\dot{\boldsymbol{\sigma}})\mathbf{I} \tag{II-22}$$

2.1.3 Elastoplasticity

Elastoplasticity correctly models the rheology of metal in cold forming processes. As illustrated in Figure 11, the yield stress σ_y is defined as the minimal stress to create a plastic non-reversible strain in a given direction. A basic goal of phenomenological plasticity models is to replicate the one-dimensional tension test. To achieve this, a scalar function has to be defined as plasticity criterion:

$$\begin{array}{l} \mathrm{f}(\boldsymbol{\sigma},\sigma_{\mathrm{y}}) < 0 \ \Rightarrow \text{pure elastic behaviour} \\ \mathrm{f}(\boldsymbol{\sigma},\sigma_{\mathrm{y}}) = 0 \ \Rightarrow \text{plastic behaviour} \end{array} \tag{II-23}$$

The Von Mises criterion is commonly used for modelling isotropic metals. It is expressed in the principal stresses referential as follows:

$$f = (\sigma_I - \sigma_{II})^2 + (\sigma_{II} - \sigma_{III})^2 + (\sigma_I - \sigma_{II})^2 - 2\sigma_y^2 \qquad \text{(II-24)}$$

Using the splitting formulation (II-14) of the stress, (II-24) becomes:

$$f = 3\left(s_I^2 + s_{II}^2 + s_{III}^2\right) - 2\sigma_y^2 \qquad \text{(II-25)}$$

Note that the plastic flow occurs when:

$$\frac{2}{3}\sigma_y^2 = \mathrm{tr}(\mathbf{s}^2) = \mathbf{s} : \mathbf{s} \qquad \text{(II-26)}$$

So, using the equivalent Von Mises stress tensor definition of equation (II-17), the plasticity criterion can be finally written in the simple form:

$$f(\boldsymbol{\sigma}, \sigma_y) = 0 \Leftrightarrow \overline{\sigma} = \sigma_y \qquad \text{(II-27)}$$

Because the six-dimensional stress space is collapsed into a single number, many stress states may produce the same equivalent stress (they collectively define a surface in stress space). The particular collections of stress states that satisfy equation (II-27) define the yield surface. The plastic strain rate is assumed parallel to the normal of the yield surface. The intensity Λ^{pl} of the plastic strain rate obeys the flowing rule:

$$\dot{\boldsymbol{\varepsilon}}^{pl} = \Lambda^{pl} \frac{\partial f}{\partial \boldsymbol{\sigma}} \qquad \text{(II-28)}$$

Λ^{pl} is a proportional constant that must be determined to complete the plasticity model. Λ^{pl} is always greater than, or equal to zero, because a negative value would imply that the response is elastic.

So for a Von Mises criterion, the equation (II-28) becomes:

$$\dot{\boldsymbol{\varepsilon}}^{pl} = \Lambda^{pl} \frac{3}{2} \frac{\mathbf{s}}{\sigma_y} \qquad \text{(II-29)}$$

And because of the incompressibility of the plastic strain, we have:

$$\mathrm{tr}(\dot{\boldsymbol{\varepsilon}}^{pl}) = 0 \qquad \text{(II-30)}$$

So for a plastic material, the stress deviator is given by:

$$\mathbf{s} = \frac{2}{3}\sigma_y \frac{\dot{\boldsymbol{\varepsilon}}^{pl}}{\dot{\overline{\varepsilon}}^{pl}} \qquad \text{(II-31)}$$

The evolution of the yield stress σ_y may be described through a function of the equivalent strain, the temperature, the pressure…

$$\sigma_y = \sigma_y(\bar{\varepsilon}, T, p) \tag{II-32}$$

To resume, the equations of elastoplasticity are:

$$\begin{cases} \dot{\boldsymbol{\varepsilon}} = \dot{\boldsymbol{\varepsilon}}^{\mathrm{el}} + \dot{\boldsymbol{\varepsilon}}^{\mathrm{pl}} \\ \dot{\boldsymbol{\varepsilon}}^{\mathrm{el}} = \mathrm{D}^{-1}\dot{\boldsymbol{\sigma}} \\ \dot{\boldsymbol{\varepsilon}}^{\mathrm{pl}} = \Lambda^{\mathrm{pl}} \dfrac{\partial f}{\partial \boldsymbol{\sigma}} \\ f(\boldsymbol{\sigma}, \sigma_{\mathrm{y}}) = 0 \end{cases} \tag{II-33}$$

2.1.4 Viscoplasticity

The elastic component can legitimately be neglected when modelling material flows at hot temperatures, and a viscoplastic potential φ can be highlighted. This latter connects the stress deviator **s** to the plastic strain rate $\dot{\boldsymbol{\varepsilon}}$:

$$\mathbf{s} = \frac{\partial \varphi}{\partial \dot{\boldsymbol{\varepsilon}}} \tag{II-34}$$

In the Forge3® software, the Norton-Hoff viscoplastic potential can be used:

$$\varphi = \frac{\mathrm{k}}{\mathrm{m}+1}\left(\sqrt{3}\dot{\bar{\varepsilon}}\right)^{\mathrm{m}+1} \tag{II-35}$$

k is the consistency of the material; m is the sensitivity to strain rate and highlights the difference between plastic and viscoplastic behaviour.

The Norton-Hoff constitutive law is deduced from (II-18), (II-34), (II-35) :

$$\mathbf{s} = 2\mathrm{K}\left(\sqrt{3}\dot{\bar{\varepsilon}}\right)^{\mathrm{m}-1} \dot{\boldsymbol{\varepsilon}} = \mu(\dot{\bar{\varepsilon}})\, \dot{\boldsymbol{\varepsilon}} \tag{II-36}$$

$\mu(\dot{\bar{\varepsilon}})$ is the viscous component of the Norton-Hoff law.

m remains generally inferior to 0.3 for metals and we can verify that:

- m=0 leads to the expression (II-31) of a rigid plastic constitutive law.
- m=1 leads to the linear expression of a Newtonian behaviour.

The equivalent stress can be written as follows:

$$\bar{\sigma} = \mathrm{K}\sqrt{3}\,(\sqrt{3}\dot{\bar{\varepsilon}})^{\mathrm{m}} \tag{II-37}$$

The hardening is described in the expression of the consistency K. Equation (II-38) gives the power law expression for a thermo-strain hardenable material.

$$K(T,\bar{\varepsilon}) = K_0\,(\bar{\varepsilon}+\varepsilon_0)^n\,\exp\left(\frac{\beta}{T}\right) \qquad \text{(II-38)}$$

K_0 is a constant, n is the sensitivity to strain hardening, and β is the sensitivity to thermal hardening, ε_0 a constant regulation term.

2.1.5 Elasto-viscoplasticity

As for elastoplasticity, small strain rate is assumed and it is additively splitted into elastic and viscoplastic component:

$$\dot{\boldsymbol{\varepsilon}} = \dot{\boldsymbol{\varepsilon}}^{el} + \dot{\boldsymbol{\varepsilon}}^{vp} \qquad \text{(II-39)}$$

with:

$$\left|\begin{array}{l} \dot{\boldsymbol{\varepsilon}}^{el} = \mathbf{D}^{-1}\,\dot{\boldsymbol{\sigma}} \\ \dot{\boldsymbol{\varepsilon}}^{vp} = \dfrac{\sqrt{3}}{2}\left(\dfrac{\bar{\sigma}}{K\sqrt{3}}\right)^{\frac{1}{m}}\dfrac{\mathbf{s}}{\bar{\sigma}} \end{array}\right. \qquad \text{(II-40)}$$

Considering the expression of the equivalent stress II-37, the viscoplastic strain rate becomes:

$$\dot{\boldsymbol{\varepsilon}}^{vp} = \frac{3}{2}\dot{\bar{\varepsilon}}\frac{\mathbf{s}}{\bar{\sigma}} \qquad \text{(II-41)}$$

Introducing in (II-40) the plastic threshold (yield stress) σ_y, the strain rate becomes:

$$\begin{array}{ll} \text{if } \bar{\sigma} \geq \sigma_y & \dot{\boldsymbol{\varepsilon}}^{vp} = \dfrac{\sqrt{3}}{2}\left(\dfrac{\bar{\sigma}-\sigma_y}{K\sqrt{3}}\right)^{\frac{1}{m}}\dfrac{\mathbf{s}}{\bar{\sigma}} \\ \text{if } \bar{\sigma} < \sigma_y & \dot{\boldsymbol{\varepsilon}}^{vp} = 0 \end{array} \qquad \text{(II-42)}$$

The deviatoric strain decomposition leads to the following set of equation:

$$\begin{cases} \dot{\boldsymbol{\varepsilon}} = \dot{\mathbf{e}} + \dfrac{1}{3}\mathrm{tr}(\dot{\boldsymbol{\varepsilon}})\mathbf{I} \\ \dot{\boldsymbol{\varepsilon}}^{el} = \dot{e}^{el} + \dfrac{1}{3}\mathrm{tr}(\dot{\boldsymbol{\varepsilon}}^{el})\mathbf{I} \\ \dot{\boldsymbol{\varepsilon}}^{vp} = \dot{\mathbf{e}}^{vp} \quad ; \text{ with } \quad \mathrm{tr}(\dot{\boldsymbol{\varepsilon}}^{vp}) = 0 \\ \dot{\mathbf{e}} = \dot{\mathbf{e}}^{el} + \dot{\mathbf{e}}^{vp} \end{cases} \qquad \text{(II-43)}$$

By identity with the Hook's law (II-21) or (II-40), the deviatoric expression of the stress (II-14), and the elastic strain rate expression (II-22), one have:

$$\begin{cases} \dot{\mathbf{e}}^{\mathrm{el}} = \dfrac{\dot{\mathbf{s}}}{2\mu} = \dfrac{1+\nu}{\mathrm{E}}\dot{\mathbf{s}} \\ \dot{\mathrm{p}} = \dfrac{\mathrm{E}}{3\left(2\nu - 1\right)}\,\mathrm{tr}(\dot{\boldsymbol{\varepsilon}}^{\mathrm{el}}) \end{cases} \tag{II-44}$$

The viscoplastic equations are then written as follows:

$$\begin{cases} \dot{\boldsymbol{\varepsilon}} = \dot{\boldsymbol{\varepsilon}}^{\mathrm{el}} + \dot{\boldsymbol{\varepsilon}}^{\mathrm{vp}} \\ \dot{\boldsymbol{\varepsilon}}^{\mathrm{el}} = \dfrac{\dot{\mathbf{s}}}{2\mu} + \dfrac{2\,\nu - 1}{\mathrm{E}}\dot{p}\mathbf{I} \\ \dot{\boldsymbol{\varepsilon}}^{\mathrm{vp}} = \dfrac{\sqrt{3}}{2}\left\langle \dfrac{\overline{\sigma} - \sigma_{\mathrm{y}}}{\mathrm{K}\sqrt{3}} \right\rangle^{\frac{1}{\mathrm{m}}} \dfrac{\mathbf{s}}{\overline{\sigma}} \end{cases} \tag{II-45}$$

$\langle \cdot \rangle$ defines the positive part.

The resolution of this system is more complex than in the pure viscoplastic case. For each time step, the incremental Prandtl-Reuss resolution is used to compute the local stress state at the end of the increment. The principle of this incremental technique is briefly described below:

- At first, the method consists in a purely elastic calculation to test if the computed stress is effectively in the elastic domain. The elastic predictor s^{p} is then computed as follows.

$$\mathbf{s}^{\mathrm{p}} = \mathbf{s}^{\mathrm{t}} + 2\mu\Delta\mathbf{e} \tag{II-46}$$

s^{t} is the stress state at the beginning of the increment.

If the following inequality is true, the deformation is purely elastic.

$$\overline{\sigma}^{\mathrm{p}} = \sqrt{\frac{3}{2}\mathbf{s}^{\mathrm{p}} : \mathbf{s}^{\mathrm{p}}} < \sigma_{\mathrm{y}\,(\overline{\varepsilon})} \tag{II-47}$$

- If $\overline{\sigma}^{\mathrm{p}} \geq \sigma_{\mathrm{y}}$ the plastic surface is reached. The associated deviatoric stress tensor $\mathrm{s}^{\mathrm{t+\Delta t}}$ is then obtained through the following relationship:

$$\mathbf{s}^{\mathrm{t+\Delta t}} = \frac{\mathbf{s}^{\mathrm{p}}}{1 + 2\mu.\Delta\lambda} \qquad \text{with} \qquad \Delta\lambda = \frac{3}{2}\frac{\overline{\varepsilon}}{\sigma_{\mathrm{y}\,(\overline{\varepsilon} + \Delta\overline{\varepsilon})}} \tag{II-48}$$

$\overline{\varepsilon}$ is given in by equation (II-19). Substituting (II-48) in equation (II-26), one obtains:

$$\overline{\sigma}^{\mathrm{p}} = \left(1 + 2\mu.\Delta\lambda\right).\sigma_{\mathrm{y}\,(\overline{\varepsilon} + \Delta\overline{\varepsilon})} \tag{II-49}$$

Finally, introducing the complete expression $\Delta\lambda$ in (II-49), the following non-linear equation is obtained:

$$\sigma_{\mathrm{y}\,(\overline{\varepsilon}^{\mathrm{vp}} + \Delta\overline{\varepsilon}^{\mathrm{vp}})} + 3\mu.\Delta\overline{\varepsilon}^{\mathrm{vp}} - \overline{\sigma}^{\mathrm{p}} = 0 \tag{II-50}$$

The unknown $\Delta\bar{\varepsilon}$ is computed trough an iterative Newton method. Afterwards, the deviatoric stress tensor is updated for the next time increment using equation (II-48).
For more clarity, a simple viscoplastic constitutive law will be utilized in the next paragraphs.

2.1.6 Thermo-elasto-viscoplasticity

In order to take into account the thermal expansion of the material, the component $\dot{\varepsilon}^{th}$ is added to the expression of the strain rate:

$$\left|\begin{array}{l}\dot{\boldsymbol{\varepsilon}} = \dot{\boldsymbol{\varepsilon}}^{el} + \dot{\boldsymbol{\varepsilon}}^{vp} + \dot{\boldsymbol{\varepsilon}}^{th} \\ \text{with} \quad \dot{\boldsymbol{\varepsilon}}^{th} = \alpha \dot{T}\,\mathbf{I} \quad \text{and} \quad \mathrm{tr}(\dot{\boldsymbol{\varepsilon}}^{th}) = 3\alpha\dot{T}\end{array}\right. \tag{II-51}$$

α is the thermal expansion coefficient and $\dot{\mathrm{T}}$ is the cooling rate. More details on the thermal coupling are given in the next chapters.
This thermo-elasto-viscoplastic constitutive law will be retained for the modelisation of the FSP.

2.2 Friction law

As previously introduced, a friction law allows for friction to be modeled at the contact interface between two bodies which have a relative sliding movement. Several friction laws defining the interfacial shear stress (see equation (II-12)) are available in Forge3®.

2.2.1 The viscoplastic friction law

The viscoplastic friction law connects the frictional shear stress $\boldsymbol{\tau}_s$ to the sliding velocity Δv_s as follows:

$$\boldsymbol{\tau}_f = -\alpha_f\, K(T,\bar{\varepsilon}) \| \Delta\mathbf{v}_s \|^{q-1} \Delta\mathbf{v}_s \tag{II-52}$$

α_f is the friction coefficient; q is the sensitivity to the sliding velocity; and $\Delta\mathbf{v}_s$ is the sliding velocity defined by equation:

$$\Delta\mathbf{v}_s = (\mathbf{v} - \mathbf{v}^{tool}) - [(\mathbf{v} - \mathbf{v}^{tool}).\mathbf{n}]\mathbf{n} \tag{II-53}$$

This viscoplastic friction law is often used in hot forging processes under high contact pressures, and should provide a good friction model for the welding step of the FSP.

2.2.2 The Tresca friction law

The Tresca law comes from the rigid plastic constitutive laws. It connects the frictional shear stress $\boldsymbol{\tau}_s$ to the material flow stress σ_y as follows:

$$\boldsymbol{\tau}_f = -\overline{m}\,\frac{\sigma_y}{\sqrt{3}}\,\frac{\Delta \mathbf{v}_s}{\|\Delta \mathbf{v}_s\|} \quad \text{with} \quad 0 \le \overline{m} \le 1 \tag{II-54}$$

$\overline{m}$ is the Tresca friction coefficient.

2.2.3 The Coulomb friction law

With the coulomb law, the shear stress is dependent on the contact pressure $p_c=\sigma_n$:

$$\boldsymbol{\tau}_f = -\mu\,\sigma_n\,\frac{\Delta \mathbf{v}_s}{\|\Delta \mathbf{v}_s\|} \tag{II-55}$$

μ is the Coulomb friction coefficient.

2.2.4 Coulomb limited by Tresca law

This is a combination of the two previous laws:

$$\begin{aligned} &\text{if} \quad |\mu\,\sigma_n| < \overline{m}\frac{\sigma_y}{\sqrt{3}} \quad \text{then} \quad \boldsymbol{\tau}_f = -\mu\,\sigma_n\,\frac{\Delta \mathbf{v}_s}{\|\Delta \mathbf{v}_s\|} \\ &\text{else if} \quad |\mu\,\sigma_n| \ge \overline{m}\,\frac{\sigma_y}{\sqrt{3}} \quad \text{then} \quad \boldsymbol{\tau}_f = -\overline{m}\,\frac{\sigma_y}{\sqrt{3}}\,\frac{\Delta \mathbf{v}_s}{\|\Delta \mathbf{v}_s\|} \end{aligned} \tag{II-56}$$

2.3 Weak formulation of the mechanical problem

The strong form of the problem given by equation (II-4), and (II-6) is reminded below:

$$\begin{cases} \mathbf{div}(\boldsymbol{\sigma}) = \rho(\boldsymbol{\gamma} - \mathbf{g}) \\ \mathrm{tr}(\dot{\boldsymbol{\varepsilon}}_{pl}) = 0 \end{cases} \tag{II-57}$$

By considering a virtual velocity field **v*** with good properties of integrability and differentiability, and applying integration by parts on the domain Ω, one obtains the weak form (in velocity, pressure) of the equilibrium equation, also called virtual work principle :

$$\text{Find } (\mathbf{v}, p) \in (V^{ca}, P) \text{ such as } \forall (\mathbf{v}^*, p^*) \in (V_0^{ca}, P)$$

$$\begin{cases} \int_\Omega \mathbf{s} : \dot{\boldsymbol{\varepsilon}}(\mathbf{v}^*) d\omega - \int_\Omega p . \operatorname{div}(\mathbf{v}^*) d\omega - \int_{\partial\Omega_c} \boldsymbol{\tau} . \mathbf{v}^* \, ds - \int_\Omega \rho(\boldsymbol{\gamma} - \mathbf{g}) . \mathbf{v}^* d\omega = 0 \\ \int_\Omega p^* \left(\operatorname{tr}(\dot{\boldsymbol{\varepsilon}}) + \dfrac{3\dot{p}}{3\lambda + 2\mu} \right) d\omega = 0 \end{cases} \qquad \text{(II-58)}$$

V^{ca} (respectively V_0^{ca}) is the acceptable kinematically velocity space (respectively acceptable kinematically at the zero) and P is the pressure space.

Assuming V is a subspace of $L^2(\Omega)$:

$$V^{ca} = \{ \mathbf{v} \in V \ / \ (\mathbf{v} - \mathbf{v}^{tool}) . n \leq 0 \quad \text{on } \partial_c \Omega \}$$

$$V_0^{ca} = \{ \mathbf{v} \in V \ / \ \mathbf{v} . n \leq 0 \quad \text{on } \partial_c \Omega \}$$

$$P = L^2(\Omega)$$

Note that the unilateral condition is imposed through the admissible kinematic fields. The incremental penalty method used to satisfy the contact condition is detailed in paragraph 4.

3 *Thermal Equations*

The thermal computations are coupled to the mechanical ones, so the discretized form of the heat equation is solved at each time step. Heat generated by material deformation and by friction with the tool, thermal exchanges by conduction, convection and radiation, should all be taken into account.

3.1 Global Heat Equation

The transient thermal problem is governed by the well-known heat equation, which is written on a domain Ω:

$$\rho c \dot{T} = \operatorname{div}(\mathbf{q}) + E \qquad \text{(II-59)}$$

ρ and c are respectively the material density and heat capacity. E is possible internal heat source. The heat flux is governed by the isotropic conduction law of Fourier:

$$\mathbf{q} = -k \nabla T \qquad \text{(II-60)}$$

k is the thermal conductivity.

The heat source E is the sum of two types of contributions:

- power dissipated through plastic deformation:

$$\dot{q}_v = f\,\bar{\sigma}\dot{\bar{\varepsilon}} \qquad \text{(II-61)}$$

The coefficient f takes into account the fraction of energy which is converted into heat. It generally ranges between 0.9 and 1. For a Norton-Hoff viscoplastic material:

$$\dot{q}_v = f\,K\left(\sqrt{3}\dot{\bar{\varepsilon}}\right)^{m+1} \qquad \text{(II-62)}$$

- possible internal heat source Q, due to internal chemical reaction for example. This contribution is negligible in metal joining.

So equation (II-59) becomes:

$$\rho c\dot{T} + \mathrm{div}(k\nabla T) = \dot{q}_v \qquad \text{(II-63)}$$

3.2 Boundary Conditions

The boundary $\partial\Omega$ of the domain Ω can be divided into four distinct zones corresponding to different types of boundary conditions and resulting in the definition of equation (II-64) (see Figure 13).

$$\partial\Omega = \partial\Omega_C + \partial\Omega_{CR} + \partial\Omega_\varphi + \partial\Omega_T \qquad \text{(II-64)}$$

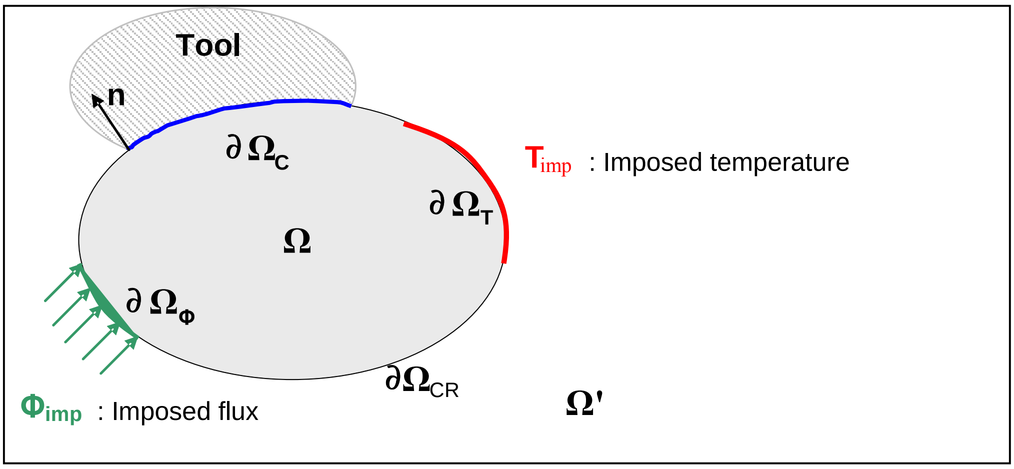


Figure 13: Thermal boundary decomposition of the domain

- On the contact surface $\partial\Omega_C$, a thermal exchange by conduction between the workpiece and the tool takes place. This is expressed by the following equation :

$$-k\nabla T \cdot \mathbf{n} = h_c\left(T - T_{tool}\right) \qquad \text{(II-65)}$$

h_c is the conduction coefficient which can depend on the temperature or on the contact pressure. T is the temperature of the considered point M of the surface $\partial\Omega_C$ and T_{tool} is the temperature of the point of the tool in contact with M.

The exchanges at the contact boundary can also result from a friction type dissipation phenomenon process. The dissipated power at the interface is “shared” between the two concerned bodies, relatively to their respective effusivity. So heat created by friction is taking into account through the additional term:

$$-k\nabla \mathrm{T}.\mathbf{n} = \frac{\mathrm{b}}{\mathrm{b}+\mathrm{b}_{\Omega'}}\boldsymbol{\tau}.\mathbf{v}_s \quad \text{with } \mathrm{b} = \sqrt{\mathrm{ck}\rho} \tag{II-66}$$

b is the effusivity of body **Ω**, and $b_{\Omega'}$ is the effusivity of body **Ω’**. In the particular case of the viscoplastic friction law we have:

$$\boldsymbol{\tau}.\mathbf{v}_s = \alpha_f K \left\|\Delta\mathbf{v}_s\right\|^q \tag{II-67}$$

- On free surfaces $\partial\Omega_{CR}$, or surfaces in contact with a non-solid body **Ω’**, convection and radiation happen. The thermal exchange due to convection is driven by the following relationship:

$$-k\nabla \mathrm{T}\cdot\mathbf{n} = \mathrm{h}_{conv}\left(\mathrm{T}-\mathrm{T}_{\Omega'}\right) \tag{II-68}$$

The radiation toward the ambient surroundings is expressed as follows:

$$-k\nabla \mathrm{T}\cdot\mathbf{n} = \varepsilon_r\sigma_r\left(\mathrm{T}^4-\mathrm{T}_{\Omega'}^4\right) \tag{II-69}$$

These two contributions (convection and radiation) can be merged into the single equation (II-70) using the global transfer coefficient h_{cr} defined as below:

$$\begin{aligned} -k\nabla \mathrm{T}\cdot\mathbf{n} &= \mathrm{h}_{cr}\left(\mathrm{T}-\mathrm{T}_{\Omega'}\right) \\ \mathrm{h}_{cr} &= \mathrm{h}_c + \varepsilon_r\sigma_r\left(\mathrm{T}^2+\mathrm{T}_{\Omega'}^2\right)\left(\mathrm{T}+\mathrm{T}_{\Omega'}\right) \end{aligned} \tag{II-70}$$

- Finally *Neumann* or *Dirichlet* conditions, which consist in respectively imposing a heating flux through the surface $\partial\Omega_\Phi$ or a temperature on the boundary $\partial\Omega_T$, can be enforced (see respectively equations (II-71) and (II-72)).

$$-k\nabla \mathrm{T}\cdot\mathbf{n} = \Phi_{imp} \tag{II-71}$$

$$\mathrm{T} = \mathrm{T}_{imp} \tag{II-72}$$

3.3 Weak formulation of the thermal problem

The weak form of the second term of the equation (II-63) is developed for any test function T^* as follows:

$$-\int_{\Omega} \mathrm{div}(k\nabla T).T^* d\omega = \int_{\Omega} k\nabla T.\nabla T^* d\omega - \int_{\partial\Omega} k\nabla T.\mathbf{n}.T^* d\omega \qquad \text{(II-73)}$$

Consequently, the integral formulation of the heat equation (II-63), enriched with previous boundary conditions, can be written for any test function T^*:

$$\text{Find } T \in \theta \text{ such as } \forall\ T^* \in \theta_0$$

$$\begin{cases} \int_{\Omega} \rho c \dot{T} T^* d\omega + \int_{\Omega} k\nabla T.\nabla T^* d\omega \\ + \int_{\partial\Omega_{CR}} h_{CR}(T - T_{\Omega'})T^* dS + \int_{\partial\Omega_{CR}} \left(h_C(T - T_{tool}) + \frac{b}{b + b_{\Omega'}} \boldsymbol{\tau}.\mathbf{v} \right) T^* dS \\ = \int_{\Omega} \dot{q}_v T^* d\omega + \int_{\partial\Omega_{\phi}} \Phi_{imp} T^* dS \end{cases} \qquad \text{(II-74)}$$

with $\theta = \{T \in \Omega \ / \ T = T_{imp} \text{ on } \partial\Omega_T\}$ and $\theta_0 = \{T \in \Omega \ / \ T = 0 \text{ on } \partial\Omega_T\}$

4 *Forge3 Finite element formulation*

4.1 Spatial discretization

The Finite Element Method (FEM) consists in dividing the whole domain Ω by an approximation Ω_h, using subdomains. Thus are defined elements and connectivity, or else FE mesh, noted T_h, of the domain Ω_h.

Ω_h^e is the volume of an element e of the mesh T_h, which is defined such as:

$$\Omega_h = \bigcup_{e \in T_h} \Omega_h^e \qquad \text{(II-75)}$$

For $i \neq j$ the intersection $\Omega_h^i \cap \Omega_h^j$ is empty or contains a node, a face or an edge.

V_h^{ca} and P_h are approximations of the spaces V^{ca} and P, containing the solution (v_h, p_h) of the discrete problem. Existence and unicity of this solution are assured by the Brezzi-Babuska compatibility condition. This condition also assures that the solution (v_h,p_h) converges towards the continuous solution (v,p) when the element size h tends towards 0.

Tetrahedral P1+/P1 type elements are used in the Forge3® software (see Figure 14). For more details on the choice of the formulation and the nature of the elements one can refer to [17]. Pressure, like temperature, are linearly interpolated from values computed at the four summits of each tetrahedron (called nodes of the element).

Velocity is also linearly interpolated from values computed at nodes, and is enriched by an additional linear contribution from the bubble function, which is defined on the 4 subtetrahedra consisting of the gravity centre of the tetrahedron and the nodes.

The bubble function, which is linear and continuous on the subtetrahedra of each facet of the tetrahedron, assures the Brezzi-Babuska condition.

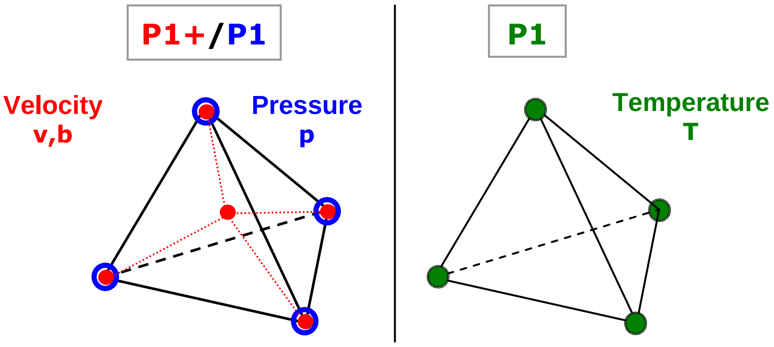


Figure 14: finite elements for velocity/ pressure and temperature interpolation

Approximation spaces are thus defined as follows:

$$
\begin{aligned}
P_h &= \left\{ p_h \in C^0(\Omega_h) / \forall e \in T_h, p_{h/\Omega_h^e} \in P1(\Omega_h^e) \right\} \\
V_h^{ca} &= L_h^{ca} \oplus B_h \\
L_h^{ca} &= \left\{ \mathbf{v}_h \in \left(C^0(\Omega_h)\right)^3 / \forall e \in T_h, \mathbf{v}_{h/\Omega_h^e} \in \left(P1(\Omega_h^e)\right)^3 \text{ and } \left(\mathbf{v}_h - \mathbf{v}^{tool}\right).\mathbf{n} \le 0 \text{ on } \partial_c\Omega \right\} \\
B_h &= \left\{ \mathbf{b}_h \in \left(C^0(\Omega_h)\right)^3 / \forall e \in T_h, \mathbf{b}_{h/\Omega_{h,i}^e} \in \left(P1(\Omega_{h,i}^e)\right)^3 \text{ and } \mathbf{b}_h = 0 \text{ on } \partial\Omega_i^e \right\}
\end{aligned}
\qquad \text{(II-76)}
$$

P1 is a linear continuous interpolation. $\Omega_{h,i}^e$ $(i = 1,4)$ are the sub-tetrahedra of element Ω_h^e.

Discreet velocity and pressure are interpolated at any point of the mesh:

$$
\begin{aligned}
&\mathbf{v}_h = \mathbf{v}_h^l + \mathbf{v}_h^b = \sum_{i=1}^{Nbnoe} \mathbf{v}_i^l N_i^l + \sum_{j=1}^{Nbelt} \mathbf{v}_j^b N_j^b \\
&p_h = \sum_{i=1}^{Nbnoe} p_i N_i^l
\end{aligned}
\qquad \text{(II-77)}
$$

Nbnoe and Nbelt respectively are the number of nodes and the number of elements of the mesh. N_i^l and N_j^b respectively are the linear shape function associated to the node i and the bubble shape function associated to the element j. N_i^l ranges between 1 at node i, and 0 at the other nodes of the element and outside the elements not containing i, whereas N_j^b ranges between 1 at the center of the element and 0 at the boundaries of the element and outside it.

Considering the discrete decompositions (II-77), the weak form of the viscoplastic problem (II-58) is written:

$$
\left\{
\begin{aligned}
&\forall k = 1,\ldots,\ nbnoe \quad \text{and} \quad \forall i = 1,\ldots,3 \\
&\int_{\Omega_h} 2K\left(\sqrt{3}\dot{\bar{\varepsilon}}\left(\mathbf{v}_h^l + \mathbf{v}_h^b\right)\right)^{m-1} \dot{\boldsymbol{\varepsilon}}\left(\mathbf{v}_h^l + \mathbf{v}_h^b\right) : \dot{\boldsymbol{\varepsilon}}\left(N_k^l \mathbf{e}_i\right) d\omega_h - \int_{\Omega_h} p \,.\, \operatorname{div}\,(N_k^l \mathbf{e}_i)\, d\omega_h \\
&\quad + \int_{\partial_c\Omega_h} \alpha_f K \left\| \mathbf{v}_{hs} \right\|^{q-1} \Delta\mathbf{v}_{hs} . \mathbf{e}_i\ N_k^l ds - \int_{\Omega_h} \rho\left(\boldsymbol{\gamma}\left(\mathbf{v}_h^l + \mathbf{v}_h^b\right) - \mathbf{g}\right) . \mathbf{e}_i N_k^l d\omega_h = 0 \\
&\\
&\forall e = 1,\ldots,\ nbelt \quad \text{and} \quad \forall i = 1,\ldots,3 \\
&\int_{\Omega_h^e} 2K\left(\sqrt{3}\dot{\bar{\varepsilon}}\left(\mathbf{v}_h^l + \mathbf{v}_h^b\right)\right)^{m-1} \dot{\boldsymbol{\varepsilon}}\left(\mathbf{v}_h^l + \mathbf{v}_h^b\right) : \dot{\boldsymbol{\varepsilon}}\left(N_k^b \mathbf{e}_i\right) d\omega_h \\
&\qquad\qquad - \int_{\Omega_h^e} p_h \,.\, \operatorname{div}\,(N_k^b \mathbf{e}_i)\, d\omega_h - \int_{\Omega_h^e} \rho(\boldsymbol{\gamma} - \mathbf{g}).\ \mathbf{e}_i N_k^l d\omega_h = 0 \\
&\\
&\forall k = 1,\ldots,\ nbnoe \\
&\int_{\Omega_h} N_k^l \left(tr\left(\dot{\boldsymbol{\varepsilon}}\left(\mathbf{v}_h^l + \mathbf{v}_h^b\right)\right) + \frac{3\dot{p}_h}{3\lambda + 2\mu} \right) d\omega = 0
\end{aligned}
\right.
\qquad \text{(II-78)}
$$

$\mathbf{e}_i$ is a unit vector in direction i of the space.

The friction component does not appear in the second equation because the bubble field is null on the boundaries of the elements. This formulation is written for a kinematically admissible field: i.e. verifying the contact equations. This restriction is tackled in the next chapter on contact improvement.

For more clarity, this set of equations is resumed as following:

$$\begin{cases} P^{l}\left(\mathbf{v}_{h}^{l},\mathbf{v}_{h}^{b},p_{h}\right)=0 \\ P^{b}\left(\mathbf{v}_{h}^{l},\mathbf{v}_{h}^{b},p_{h}\right)=0 \\ P^{p}\left(\mathbf{v}_{h}^{l},\mathbf{v}_{h}^{b},p_{h}\right)=0 \end{cases} \tag{II-79}$$

4.2 Time discretization

The system of equation (II-78) has to be satisfied at any moment. So the time space is also discretized in N increments defined by the time steps Δt^{i}:

$$T^{tot}=\bigcup_{i=1,N}\left[t^{i},t^{i}+\Delta t^{i}\right] \tag{II-80}$$

A θ method generally provides a suitable scheme for the temporal integration of any variable y:

$$y^{t+\Delta t}=y^{t}+\left((1-\theta).\dot{y}^{t}+\theta.\dot{y}^{t+\Delta t}\right)\Delta t \tag{II-81}$$

In Forge3, for the mechanical problem, the θ-parameter is set to 1. This is the Eulerian implicit scheme. Assuming that the configuration $\mathbf{\Omega}^{t}$ is known at the moment t, the solution $(v^{t+\Delta t},p^{t+\Delta t})$ satisfying the system (II-79), can be computed. Nodal locations **x** are updated at (t+Δt) from the known configuration at t through the implicit Eulerian scheme of equation (II-82). It so provides the temporal scheme of the updated Lagangian formulation.

$$\forall t\text{, (II-79)}\rightarrow\left(\mathbf{v}^{t+\Delta t},p^{t+\Delta t}\right)\rightarrow\ \mathbf{x}^{t+\Delta t}=\mathbf{x}^{t}+\mathbf{v}^{t+\Delta t}\Delta t \tag{II-82}$$

Another scheme is used in the Forge3® software: a second order Runge-Kutta (RK2) scheme. In this middle point scheme, two velocity fields have to be computed. An intermediary configuration is build at (t+Δt/2) as follows:

$$\mathbf{\Omega}^{t+\Delta t/2}=\mathbf{\Omega}^{t}+v^{t+\Delta t}\Delta t/2 \tag{II-83}$$

The velocity $v^{t+\Delta t/2}$ is computed on this intermediary configuration, and is then used to update the initial one:

$$\mathbf{\Omega}^{t+\Delta t}=\mathbf{\Omega}^{t}+v^{t+\Delta t/2}\Delta t \tag{II-84}$$

Figure 15 illustrates this updating scheme.

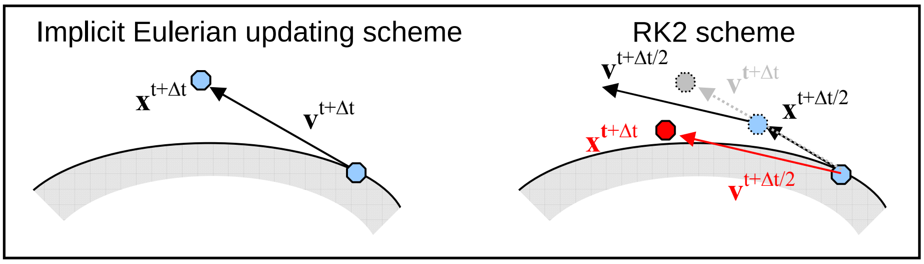


Figure 15: The two available updating scheme in Forge3

This RK2 scheme is significantly more accurate for rotational movements, limiting the numerical variations of the volume. Therefore, it is quite an interesting scheme for the simulation of the FSW process.

4.3 Mechanical Resolution

The problem (II-79) can be written as $\mathbf{P}_h(\mathbf{X}) = \mathbf{K}(\mathbf{X}).\mathbf{X} - \mathbf{F}(\mathbf{X}) = 0$, and is non-linear: **K** and **F** are both depending on the unknowns $\mathbf{X} = (v^l, v^b, p)$. It is solved by a Newton-Raphson algorithm which consists in a predictor and a corrector steps more precisely detailed in [18].

In the predictor step, $\mathbf{P}_h(\mathbf{X})$ is linearized with respect to **X** and subsequently $\Delta\mathbf{X}$ is calculated from this linearized system of equations. At the kth iteration, the following linear system is solved (see Figure 16):

$$\mathbf{P}_h(\mathbf{X}^{k-1} + \Delta\mathbf{X}^k) \approx \mathbf{P}_h(\mathbf{X}^{k-1}) + \frac{\partial \mathbf{P}_h(\mathbf{X}^{k-1})}{\partial \mathbf{X}}.\Delta\mathbf{X}^k = 0 \qquad \text{(II-85)}$$

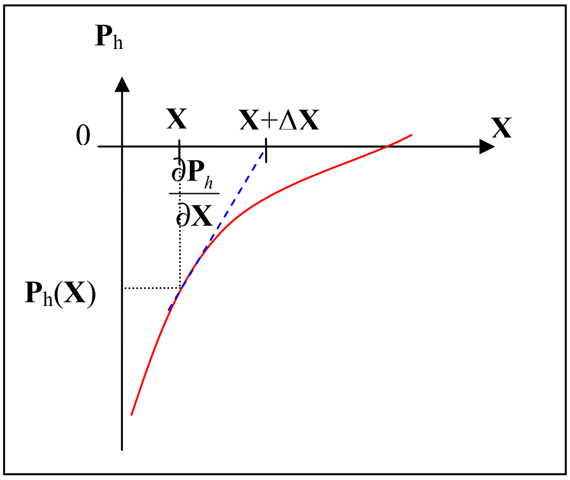


Figure 16: Schematic representation of the predictor step in Newton-Raphson iteration

The tangential stiffness matrix is given by:

$$\left(\frac{\partial \mathbf{P}_h}{\partial \mathbf{X}}\right)^{k-1} = \mathbf{K}^{k-1} \tag{II-86}$$

Note that:

$$\frac{\partial \mathbf{P}_p}{\partial p} = K^{pp} = 0 \tag{II-87}$$

Further more, the properties of the P1+/P1 element and its bubble function lead to the following simplification [19]:

$$K^{lb} = 0 \tag{II-88}$$

So (II-85) is more precisely written as:

$$\begin{pmatrix} K^{ll} & 0 & K^{lp} \\ 0 & K^{bb} & K^{bp} \\ K^{pl} & K^{pb} & 0 \end{pmatrix} \begin{pmatrix} \Delta v^{l} \\ \Delta v^{b} \\ \Delta p \end{pmatrix} = \begin{pmatrix} F'^{l}\left(v_h^l, v_h^b, p_h\right) \\ F'^{b}\left(v_h^l, v_h^b, p_h\right) \\ F'^{p}\left(v_h^l, v_h^b, p_h\right) \end{pmatrix}$$

$$K^{ij} = \frac{\partial R^i\left(x_1, x_2, x_3\right)}{\partial x_j} \quad \text{with} \quad \left(x_1, x_2, x_3\right) = \left(v_h^l, v_h^b, p_h\right) \tag{II-89}$$

(Δv^l, Δv^b, Δp) are the corrections of the solution (v^l, v^b, p) of the previous Newton-Raphson iteration. A static condensation technique simplifies the linear system by the elimination of the bubble term. It so provides a well conditioned stiffness matrix. Then, the system is solved in an iterative way by the pre-conditioned minimal conjugated residual algorithm.

In the corrector step, the non-linearities of the problem are handled. The unknown variable is updated as:

$$\mathbf{X}^k = \mathbf{X}^{k-1} + \alpha \Delta \mathbf{X}^k \tag{II-90}$$

α is the line search coefficient. It is calculated by dichotomy in order to minimize the residue $\left\|\mathbf{R}_h(\mathbf{X}^k)\right\|$ which is computed as:

$$\mathbf{R}_h(\mathbf{X}^k) = \mathbf{F}^k - \mathbf{K}^k.\mathbf{X}^k \tag{II-91}$$

The iteration process has converged when the following criterion is satisfied for a small accuracy coefficient C_R:

$$\frac{\left\|\mathbf{R}_h(\mathbf{X}^k)\right\|}{\left\|\mathbf{K}^k.\mathbf{X}^k\right\|} \leq C_R \tag{II-92}$$

4.4 Thermal resolution: the Galerkin Method

At each time increment, the mechanical resolution is followed by the thermal resolution which is thus incrementally coupled. In the utilized Galerkin method, the temperature is the unique unknown variable. It is discretized by the finite element method using a linear interpolation function P1 as previously shown in Figure 14. At any time, the temperature is expressed at a given point of a considered element as a function of the nodal temperatures T_k:

$$\mathrm{T}(\mathbf{x}) = \sum_{k=1}^{nN} \mathrm{N}_k(\mathbf{x}).\mathrm{T}_k \tag{II-93}$$

nN is the number of nodes of the considered element and N(x) are its linear interpolation functions.

So the problem (II-74) can be expressed in the following matrix formulation:

$$\mathbf{C}.\frac{d\mathbf{T}}{dt} + \mathbf{K}.\mathbf{T} = \mathbf{Q} \tag{II-94}$$

Where **T** is the unknown vector of the nodal values, **C** is the heat capacity matrix, **K** is the conductivity matrix and **Q** is the mass matrix.Given the fact that these three matrixes are thermo-dependent, the system is non-linear.

$$\begin{aligned}
&\forall (i,j) \in [1,\dots,\mathrm{nbnodes}]^2 : \\
&C_{ij} = \int_{\Omega_h} \rho c \mathrm{N}_i \mathrm{N}_j \, d\omega \\
&K_{ij} = \int_{\Omega_h} k \nabla \mathrm{N}_i \nabla \mathrm{N}_j \, d\omega + \int_{\partial\Omega_{hCR}} h_{CR} \mathrm{N}_i \mathrm{N}_j \, dS + \int_{\partial\Omega_{hC}} h_C \mathrm{N}_i \mathrm{N}_j \, dS \\
&Q_{ij} = \int_{\partial\Omega_{hCR}} h_{CR} \mathrm{N}_i \mathrm{T}_{\Omega'} \, dS + \int_{\partial\Omega_{hC}} \left(h_C \mathrm{T}_{tool} - \frac{b}{b + b_{\Omega'}} \boldsymbol{\tau}.\mathbf{v} \right) \mathrm{N}_i \, dS \\
&+ \int_{\partial\Omega_{hC}} \Phi_{imp} \mathrm{N}_i \, dS + \int_{\Omega_h} f \boldsymbol{\sigma} : \dot{\boldsymbol{\varepsilon}} \, \mathrm{N}_i \, dS
\end{aligned} \tag{II-95}$$

The time integration is provided by a two time steps scheme presented in eq. (II-96) and (II-97).

$$\mathbf{T}^{*} = \alpha_1 \mathbf{T}_{t-\Delta t_1} + \alpha_2 \mathbf{T}_t + \alpha_3 \mathbf{T}_{t+\Delta t_2} \tag{II-96}$$

$$\frac{\partial \mathbf{T}^{*}}{\partial t} = (1-\gamma_2)\frac{\mathbf{T}_{t-\Delta t_1} + \mathbf{T}_t}{\Delta t_1} + \gamma_2 \frac{\mathbf{T}_{t+\Delta t2} + \mathbf{T}_t}{\Delta t_2} \tag{II-97}$$

Where $\alpha_1, \alpha_2, \alpha_3$ and γ_2 are constants which give the ability to choose different time integration schemes. $\Delta t_1, \Delta t_2$ are the backward and the forward time steps. $\mathbf{T}_{t-\Delta t_1}, \mathbf{T}_t, \mathbf{T}_{t+\Delta t_2}$ correspond to the temperature at those time steps.

All the following equations are first written at a virtual time step t*, which is expressed by:

$$t^{*} = \alpha_1(t-\Delta t_1) + \alpha_2 t + \alpha_3(t+\Delta t_2) \tag{II-98}$$

Equation (II-94) is then rewritten as:

$$\mathbf{C}^{*}\left[(1-\gamma_2)\frac{\mathbf{T}_{t-\Delta t1} + \mathbf{T}_t}{\Delta t_1} + \gamma_2 \frac{\mathbf{T}_{t+\Delta t_2} + \mathbf{T}_t}{\Delta t_2}\right] + \mathbf{K}^{*}\left[\alpha_1 \mathbf{T}_{t-\Delta t_1} + \alpha_2 \mathbf{T}_t + \alpha_3 \mathbf{T}_{t+\Delta t_2}\right] = \mathbf{Q}^{*} \tag{II-99}$$

$\mathbf{K}^{*}$ and $\mathbf{C}^{*}$ and $\mathbf{Q}^{*}$ are the conductivity and heat capacity matrix for $\mathbf{T}^{*}$.

The non-linearities of the heat equation are taken into account by the Zlàmal method [20], which can be classified as an explicit method:

$$\mathbf{K}^{*} = \mathbf{C}_1 \mathbf{K}_{t-\Delta t_1} + \mathbf{C}_2 \mathbf{K}_t \tag{II-100}$$

$$\mathbf{K}^{*} = \left(\alpha_1 - \alpha_3 \frac{\Delta t_2}{\Delta t_1}\right)\mathbf{K}_{t-\Delta t_1} + \left(\alpha_2 + \alpha_3\left(1 + \frac{\Delta t_2}{\Delta t_1}\right)\right)\mathbf{K}_t \tag{II-101}$$

Equation (II-99) is then solved with respect to $\mathbf{T}_{t+\Delta t_2}$, so providing a linear system to solve.

4.4.1 Thermal shock

When a linear interpolation is used in the classical Galerkin formulation, the difficulties arise to deal with thermal shock. It consists of a non-physical oscillation of the temperature wherever its gradient is too large to be absorbed during one time step (in other words, to cross the element within the time step, see Figure 17).

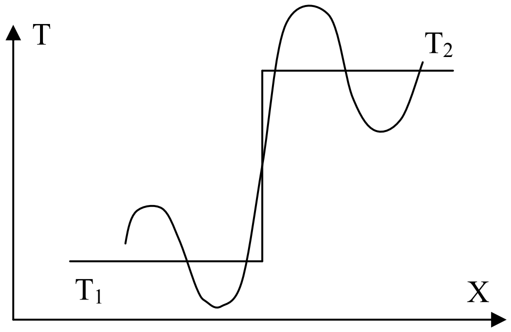


Figure 17: Temperature oscillations due to the thermal shock phenomenon.

In order to circumvent it, mechanical and temperature resolutions which are explicitly coupled are desynchronised, not using the same time step for both systems. The heat equation is solved with a larger one which is called the shock time step. It is calculated according to eq. (II-102) in order to guaranty that no numerical shock should appear. Then, a linear interpolation is used to extrapolate the temperature at the time step (II-103) that is common to the mechanical and thermal resolution.

$$\Delta t^{asyn} = \max_e\left(\frac{\rho C}{K}h^2\right) \tag{II-102}$$

$$T_{n+1} = T_n + \left(T_{n+1}^{asyn} - T_n\right)\frac{\Delta t}{\Delta t^{asyn}} \tag{II-103}$$

ρ is the density, Δt^{asyn} the asynchronous time step and h the element size.

4.4.2 Specific remapping step for rigid body movement

A specific transfer procedure has been implemented for the thermal update of tools having a rigid body movement. Chapter III : presents several remapping techniques for nodal variables such as temperature. But in case of a rigid body movement, as only the plunge velocity is used to update the tool grid position, a specific remapping procedure has been developed in order to better handle the tool rotation and consequently better transfer the temperature field. For a tool having a rigid body movement, the “ancestor” of any node or integration point can be easily found (see Figure 18). Having located this ancestor, the temperature (or any other nodal field) is interpolated at this point and then transferred to the parent node. It so provides an exact interpolation method that does not suffer from numerical problems caused by large rotations.

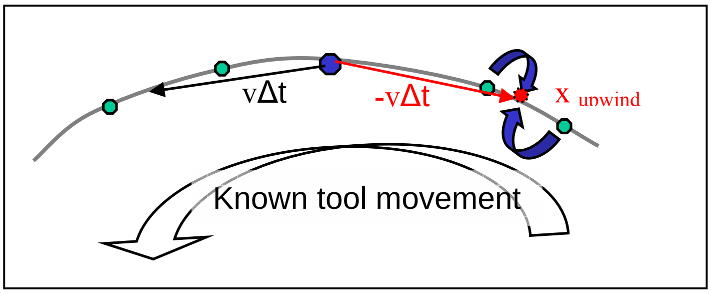


Figure 18: Illustration of the specific remapping procedure for nodal variable of rigid tools.

4.5 Remeshing procedure

The Lagrangian description utilized in Forge3 forces to deal with mesh distortion. So a specific remeshing procedure detailed in [21] for instance, generates unstructured meshes for the current complex geometries. The utilized meshing technique exploits the possibility to operate on the mesh topologies without considering any geometric constraint. A rather simple local algorithm is used to move from meshes to meshes. The nodes creation and deletion are implicitly included in the algorithm and the modifications of the surface and volume meshes is strongly coupled (see [22] for more details).

A mesh can also be adapted dynamically by deriving a new mesh from the old mesh taking into account a mesh size map. A posteriori estimation of the finite element discretization error allows for an optimal element size map compution, which minimizes the mesh dimensions for a prescribed accuracy (see Chapter IV : 2.5.2 for some more details).

Chapter III : Contact Improvement by tool smoothing

The contact modelisation is another major key point for the FSW simulation, as it plays a very important part in the problem solution. In fact, in the FSW process, the major part of heat is generated by friction. The quantity and location of this energy are very influent on the material rheological behaviour and condition the success or unsuccess of the deposition welding process. Therefore, the phenomena involved at the contact interface are crucial and have to be properly taken in account into the numerical model.

The contact algorithm has to tackle with many numerical difficulties which are introduced in this chapter. Some developments are then presented to improve this contact algorithm.

1 Contact Algorithm

1.1 Numerical Treatment of the Lagrangian Unilateral Contact

In the previous chapter, contact was simply introduced as a condition used in the definition of the space of the velocity solution field. Its numerical treatment is presented here.

In Forge3®, like in many finite element softwares, it is very convenient to discretise complex obstacles (tool boundaries) with segments (in 2D) or facets (in 3D). For each point **x** of the domain boundary, the signed distance $\delta(\mathbf{x})$ to the closest of these segments or facets can be easily computed with the following convention:

- $\delta(\mathbf{x})>0$ if the point is outside the tool
- $\delta(\mathbf{x})=0$ if the point is on the tool boundary
- $\delta(\mathbf{x})<0$ means that the point has penetrated into the tool

A nodal approach is used in Forge3®: the contact condition is exactly imposed only at the nodes of the domain boundary. This condition (corresponding to the first equation of the set (II-11)) can be written as:

$$\forall k \in \partial\Omega_h : \quad \delta_k \geq 0 \qquad \text{(III-1)}$$

with $\delta_k = \delta(\mathbf{x}_k)$.

As the time is also discretized, this condition is written at t+Δt:

$$\forall k \in \partial\Omega_h : \quad \delta_k^{t+\Delta t} \geq 0 \qquad \text{(III-2)}$$

It traduces that no node are authorized to penetrate the obstacle at the end of the time increment. The distance δ is expressed with the assumption that the tool can locally be approximated by its tangent plane (flat surface):

$$\delta(\mathbf{x},t) = \left(\mathbf{x}(t) - \Pi(\mathbf{x},t)\right).\mathbf{n}(\mathbf{x},t)$$

$$\frac{d\delta\,(\mathbf{x},t)}{dt} = \left(\frac{d\mathbf{x}(t)}{dt} - \left(\frac{\partial\Pi}{\partial\mathbf{x}}\frac{\partial\mathbf{x}(t)}{\partial t}\right) + \frac{\partial\Pi(\mathbf{x},t)}{\partial t}\right).\mathbf{n}(\mathbf{x},t) + \left(\mathbf{x}(t) - \Pi(\mathbf{x},t)\right).\left(\frac{\partial\mathbf{n}}{\partial t} - \frac{\partial\mathbf{n}}{\partial\mathbf{x}}\frac{d\mathbf{x}}{dt}\right) \qquad \text{(III-3)}$$

The following first order expansion (illustrated by Figure 19) is then utilized at t+Δt:

$$\begin{aligned}\delta_k^{t+\Delta t} &= \delta_k^t + \frac{d\delta_k^t}{dt}\Delta t \\ &= \delta_k^t + \left(\mathbf{v}_{tool} - \mathbf{v}_k\right).\mathbf{n}_k^t\Delta t + O\left(\Delta t^2\right)\end{aligned} \qquad \text{(III-4)}$$

$\mathbf{v}_{tool}$ is the velocity of the tool at π_k^t, the orthogonal projection of node k at time t on the tool. $\mathbf{n}_k^t$ is the unit vector defined as:

$$\mathbf{n}_k^t = \frac{\overrightarrow{x_k^t\Pi_k^t}}{\left\|\overrightarrow{x_k^t\Pi_k^t}\right\|} \qquad \text{(III-5)}$$

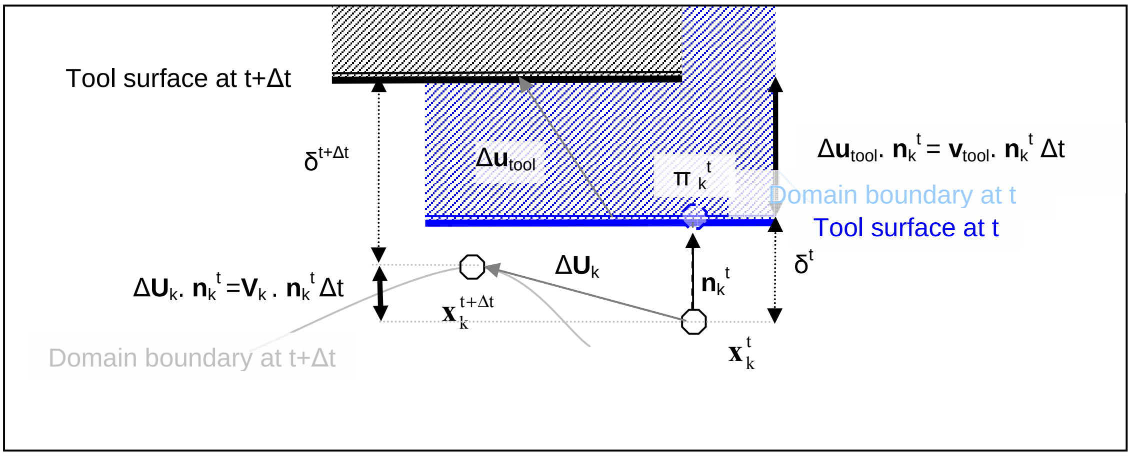


Figure 19: Illustration of the contact condition (where the tool is locally approximated by its tangent plane π_k^t.

So the condition of unilateral contact (III-2) is written as:

$$\forall k \in \partial\Omega_h :\ h\left(\mathbf{v}_k\ ,\mathbf{v}_{tool}\right) = \left(\mathbf{v}_k - \mathbf{v}_{tool}\right).\mathbf{n}_k^t - \frac{\delta_k^t}{\Delta t} \leq 0 \qquad \text{(III-6)}$$

Then the contract potential is expressed by under its discrete form as:

$$\phi_{contact}\left(\mathbf{v}_k\ ,\mathbf{v}_{tool}\right) = \frac{1}{2}\rho_c\sum_{k=1}^{nbnoe} S_k\left[h\left(\mathbf{v}_k\ ,\mathbf{v}_{tool}\right)\right]^{+2} \qquad \text{(III-7)}$$

With the notation of the positive part: $[x]^+ = \frac{x+|x|}{2}$. ρ_c is a penalty coefficient taken large enough. S_k is the surface associated to node k. It allows weighting the nodal contact condition with the surface area of the element containing the node. It is defined as follows:

$$S_k = \int_{\partial_c\Omega_h} N_k ds \qquad \text{(III-8)}$$

The contact potential is added to the Lagrangian of the mixed velocity/pressure problem, which is minimized, leading to the following set of equations which so provide the complete form of (II-78):

$$\begin{cases}
\forall k = 1,\ldots,\text{nbnoe} \ \text{ and } \ \forall i = 1,\ldots,3 \\
\int_{\Omega_h} 2K\left(\sqrt{3}\dot{\bar{\varepsilon}}\left(\mathbf{v}_h^l + \mathbf{v}_h^b\right)\right)^{m-1} \dot{\boldsymbol{\varepsilon}}\left(\mathbf{v}_h^l + \mathbf{v}_h^b\right) : \dot{\boldsymbol{\varepsilon}}\left(N_k^l \mathbf{e}_i\right) d\omega_h - \int_{\Omega_h} p_h \,.\, \text{div}\,(N_k^l \mathbf{e}_i)\, d\omega_h \\
\quad - \int_{\Omega_h} \rho\left(\boldsymbol{\gamma}\left(\mathbf{v}_h^l + \mathbf{v}_h^b\right) - \mathbf{g}\right). \mathbf{e}_i N_k^l d\omega_h + \int_{\partial_c\Omega_h} \mathbf{1}_{\partial\Omega_{Fh}}(k)\, \alpha_f K \left\| \mathbf{v}_{hs} \right\|^{q-1} \Delta\mathbf{v}_{hs} . \mathbf{e}_i \, N_k^l ds \\
\quad + \rho_c \mathbf{1}_{\partial\Omega_{Ch}}(k) S_k \left(\left(\mathbf{v}_k - \mathbf{v}_{tool}\right).\mathbf{n}_k^t - \frac{\delta_k^t}{\Delta t} \right). \mathbf{n}_k^t = 0 \\
\\
\forall e = 1,\ldots,\text{nbelt} \ \text{ and } \ \forall i = 1,\ldots,3 \\
\int_{\Omega_h^e} 2K\left(\sqrt{3}\dot{\bar{\varepsilon}}\left(\mathbf{v}_h^l + \mathbf{v}_h^b\right)\right)^{m-1} \dot{\boldsymbol{\varepsilon}}\left(\mathbf{v}_h^l + \mathbf{v}_h^b\right) : \dot{\boldsymbol{\varepsilon}}\left(N_k^b \mathbf{e}_i\right) d\omega_h \\
\qquad - \int_{\Omega_h^e} p_h \,.\, \text{div}\,(N_k^b \mathbf{e}_i)\, d\omega_h - \int_{\Omega_h^e} \rho(\boldsymbol{\gamma} - \mathbf{g}).\ \mathbf{e}_i N_k^l d\omega_h = 0 \\
\\
\forall k = 1,\ldots,\text{nbnoe} \\
\int_{\Omega_h} N_k^l \left(\text{tr}\left(\dot{\boldsymbol{\varepsilon}}\left(\mathbf{v}_h^l + \mathbf{v}_h^b\right)\right) + \frac{3\dot{p}_h}{3\lambda + 2\mu} \right) d\omega = 0
\end{cases} \qquad \text{(III-9)}$$

Notice the introduction of the contact and friction indicator functions, which are respectively defined as:

$$\forall k \in \partial\Omega_h :$$

$$\begin{cases} \mathbf{1}_{\partial\Omega_{Ch}}(k) = 1 \quad \text{if} \quad h(\mathbf{v}_k, \mathbf{v}_{tool}) = (\mathbf{v}_k - \mathbf{v}_{tool}).\mathbf{n}_k^t - \dfrac{\delta_k^t}{\Delta t} > 0 \\ \mathbf{1}_{\partial\Omega_{Ch}}(k) = 0 \quad \text{else.} \end{cases} \qquad \text{(III-10)}$$

$$\begin{cases} \mathbf{1}_{\partial\Omega_{Fh}}(k) = 1 \quad \text{if} \quad \delta_k^t \le 0 \\ \mathbf{1}_{\partial\Omega_{Fh}}(k) = 0 \quad \text{else.} \end{cases}$$

The numerical resolution of this system has been presented in the previous chapter. The penalty terms are concentrated on the diagonal block of the tangent matrix, so its conditioning is not worsen for reasonable values of ρ_c. The particularity of this contact formulation and improvements which have been brought are going to be detailed thereafter.

1.2 Formulation improvement

The presented contact formulation, which is implemented in Forge3®, is very robust and has shown its ability to accurately simulate many processes. However, it is based on assumptions which could be at the origin of several problems such as convergence difficulties, numerical oscillations with exaggerated stresses, or unjustified loss of contact with coarse models, which are mainly due to:

- the normal $\mathbf{n}^t$ and the distance to the tool δ^t are computed at time t, and are assumed constant during the time step Δt. So, the contact obstacle can be regarded as “explicitly” discretized in time.
- the discretization of the tool boundaries with facets results in a C^0 continuity of the obstacle and in the discontinuity of its normals.

1.2.1 Bilateral and Eulerian Contact Formulations

The problem of numerical contact losses (illustrated in Figure 20) can be very problematic in the zone under the tool during FSW simulation. Two easy but crude ways to tackle this problem are using a bilateral Lagrangian or Eulerian contact formulation.

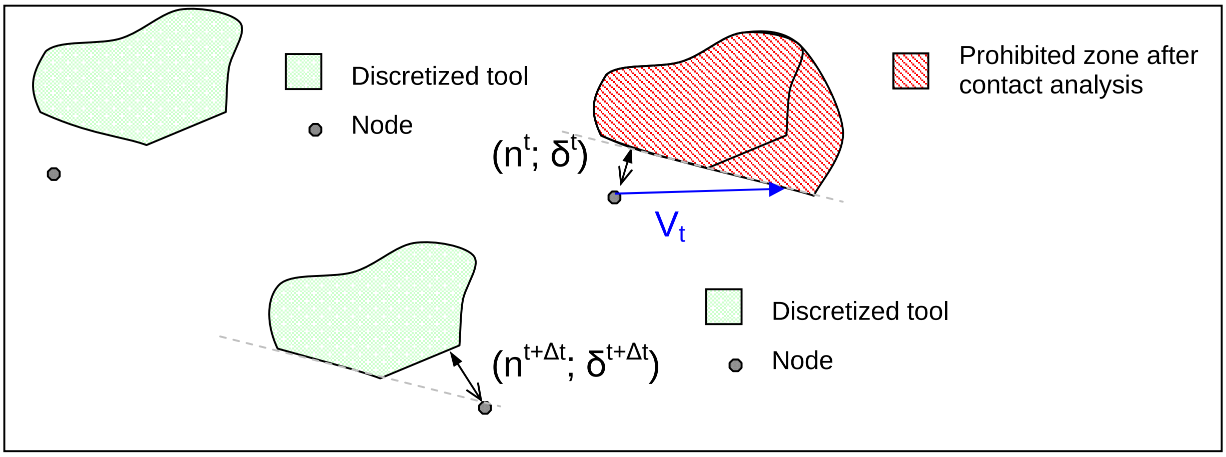


Figure 20: Illustration of the contact loss for a simple 2D convex angle.

In the case of bilateral contact, the nodes which are not in contact are allowed to get in contact, but the nodes which are in contact are not allowed to leave the contact. The contact inequality (III-6) becomes an equality condition for these nodes:

$$\begin{aligned}&\forall k \in \partial\Omega_h :\\ &\text{if} \quad \delta_k^t \le 0 \quad \text{then} \quad h(\mathbf{v}_k, \mathbf{v}_{tool}) = (\mathbf{v}_k - \mathbf{v}_{tool}).\mathbf{n}_k^t - \frac{\delta_k^t}{\Delta t} = 0\\ &\text{else} \quad h(\mathbf{v}_k, \mathbf{v}_{tool}) = (\mathbf{v}_k - \mathbf{v}_{tool}).\mathbf{n}_k^t - \frac{\delta_k^t}{\Delta t} \le 0\end{aligned} \tag{III-11}$$

In (III-9), it just leads to change the contact indicator function definition:

$$\forall k \in \partial\Omega_h :$$

$$\begin{cases} \mathbf{1}_{\partial\Omega_{Ch}}(k) = 1 \quad \Leftrightarrow \begin{cases} \delta_k^t \le 0 \\ \text{or} \ \ h(\mathbf{V}_k, \mathbf{v}_{tool}) = (\mathbf{V}_k - \mathbf{v}_{tool}).\mathbf{n}_k^t - \dfrac{\delta_k^t}{\Delta t} > 0 \end{cases} \\ \mathbf{1}_{\partial\Omega_{Ch}}(k) = 0 \quad \text{else.} \end{cases} \tag{III-12}$$

The friction indicator function is unchanged.

In the case of Eulerian simulations, the domain boundaries are stationary. The distance δ remains constant in time, and does not operate in the contact condition. The contacting nodes are a priori known, as the ones located at a maximum given distance d_E from the tool. The contact condition (III-6) becomes:

$$\forall k \in \partial\Omega_h \ / \ \delta_k < d_E : \quad (\mathbf{v}_k - \mathbf{v}^{tool}).n \ \le 0 \tag{III-13}$$

In the set (III-9), the first equation becomes:

$$
\begin{aligned}
&\forall \mathrm{k} = 1,\ldots,\mathrm{nbnoe} \ \text{ and } \ \forall \mathrm{i} = 1,\ldots,3 \\
&\int_{\Omega_h} 2K\left(\sqrt{3}\dot{\bar{\varepsilon}}\left(\mathbf{v}_h^l + \mathbf{v}_h^b\right)\right)^{m-1} \dot{\boldsymbol{\varepsilon}}\left(\mathbf{v}_h^l + \mathbf{v}_h^b\right) : \dot{\boldsymbol{\varepsilon}}\left(N_k^l \mathbf{e}_i\right) d\omega_h - \int_{\Omega_h} p_h \,.\, \mathrm{div}\,(N_k^l \mathbf{e}_i)\, d\omega_h \\
&\quad - \int_{\Omega_h} \rho\left(\boldsymbol{\gamma}\left(\mathbf{v}_h^l + \mathbf{v}_h^b\right) - \mathbf{g}\right) . \mathbf{e}_i N_k^l d\omega_h + \int_{\partial_c \Omega_h} \mathbf{1}_{\partial\Omega_{Fh}}(k)\, \alpha_f K \left\| \mathbf{v}_{hs} \right\|^{q-1} \Delta\mathbf{v}_{hs} . \mathbf{e}_i \, N_k^l ds \\
&\quad + \rho_c \mathbf{1}_{\partial\Omega_{Ch}}(k) S_k \left(\left(\mathbf{V}_k - \mathbf{v}_{tool}\right) . \mathbf{n}_k^t\right) . \mathbf{n}_k^t = 0 \\
&\text{with } \begin{cases} \mathbf{1}_{\partial\Omega_{Ch}}(k) = \mathbf{1}_{\partial\Omega_{Fh}}(k) = 1 & \Leftrightarrow \delta_k \le d_E \\ \mathbf{1}_{\partial\Omega_{Ch}}(k) = \mathbf{1}_{\partial\Omega_{Fh}}(k) = 0 & \text{else.} \end{cases}
\end{aligned}
\qquad \text{(III-14)}
$$

However, even though these two formulations tackle the problem of contact losses in a crude way, they force to make hypotheses (no relaxation of contact, a-priori known contact surface) that may result in exaggerated stresses or unphysical material flow (no possibility to model the formation of holes behind the pin, for instance).

1.2.2 Implicit formulation and tool smoothing

An implicit formulation for the tool surface allows taking into account possible evolutions of the contact conditions during the time step Δt, and should therefore overcome the drawbacks of the previous unilateral Lagrangian formulation.

One way to solve the resulting implicit equations and to avoid the restrictive hypotheses of (III-4) is to update the contact analysis at each iteration of the Newton Raphson algorithm. However, in some cases, the algorithm may not converge. It occurs when a node lies in a "corner" of the obstacle as presented in Figure 21. The corner normal is undefined, so the algorithm may oscillate between the two possible values of the normal.

Further more, when using an adapted remeshing algorithm based on error estimation and a coarse discretization of the obstacle, the refinement procedure tends to reduce the mesh sizes in the discontinuity zones. It so computes very high local efforts at surface angles, which have only a numerical nature. Therefore, the finite element method is not converging toward the problem solution, but amplifies a numerical defect.

In order to solve these issues, a tool smoothing procedure can be introduced as in Mocellin et al. [23] or in Chamoret et al. [24] with some further enhancements. This is the purpose of the following section.

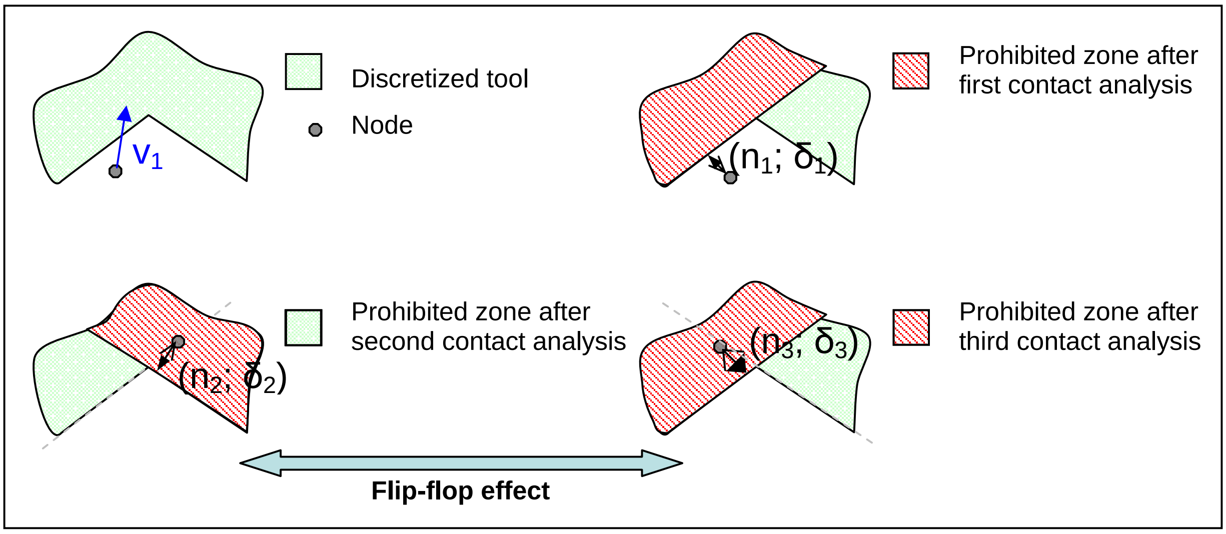


Figure 21: Illustration of the “flip-flop” effect for a simple 2D concave angle.

2 *Tool Smoothing Procedure*

2.1 Background

Several techniques have been proposed in literature to smooth contact obstacles. Some are based on Bezier surfaces [25] or on similar interpolations, while others use Gregory patches [26]. In 3D, and for very complex obstacles such as encountered in metal forming simulations, these methods are not very easy to implement into an existing finite element code. On the other hand, a much more simple method was proposed in Katia Mocellin’s thesis [27] but it was restricted to the smoothing of the concave parts of the obstacle. This approach is based on a specific transformation of the obstacle that results from a normal shift of the nodes of the facets. This transformation is accompanied by an associated modification of the distance function. Therefore, the contact with the transformed obstacle using the modified distance function produces the expected smoothing. The objective is to extend this approach to the convex parts of the obstacle, in order to smooth any kind of 3D surface with the aim of implementing an implicit contact formulation in the future.

2.2 Principle: 2D explanation

First, we consider the very simple case where the contact surface, C_A is a simple plane, π_A. Then, we consider C_B, the transformed contact surface of C_A. It is obtained by the translation of π_A, in the direction n_A (the exterior normal of π_A), by the distance δ_s (see Figure 22). C_B is so the plane π_B .

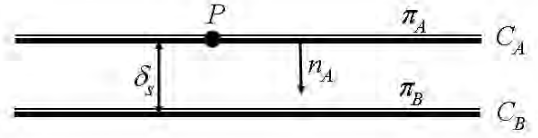


Figure 22: First case, the contact surface and the transformed contact surface.

For any point P, the signed distance of P to C_A is equal to the signed distance of P to C_B plus δ_s:

$$d(P,C_A) = d(P,C_B) + \delta_s \tag{III-15}$$

Then, the contact condition can be consequently transformed as follows:

$$d(P,C_A) = 0 \quad \Leftrightarrow \quad d(P,C_B) = -\delta_s \tag{III-16}$$

In the second case, the contact surface C_A is no longer a plane but a corner. The same transformation is applied to the two planes, π_A and $\pi_{A'}$ that constitute this corner (see Figure 23). The transformed contact surface C_B is the shifted corner that is made of the two planes π_B and $\pi_{B'}$, which respectively result from the translation of π_A in the direction n_A and the translation of $\pi_{A'}$ in the direction $n_{A'}$, by the distance δ_s.

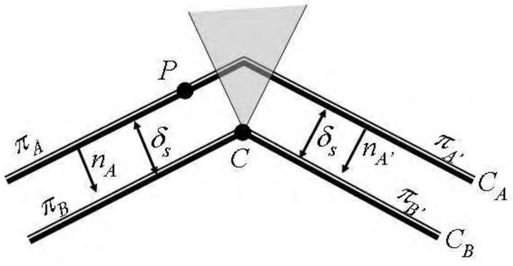


Figure 23: Second case, the contact surface consists in two planes.

In this case, equation (III-15) is correct almost everywhere, except in the shaded zone of Figure 23. In this zone, the equality (III-16) is not true, so we introduce the new distance function $\tilde{d}$ as:

$$\tilde{d}(P,C_A) = d(P,C_B) + \delta_s \tag{III-17}$$

$\tilde{d}$ is different from *d* only in the shaded zone, which thus becomes a smoothing zone. Any point *P* belonging to this zone is orthogonally projected onto the transformed contact surface C_B at point *C* (see Figure 23). Practically, this property is used to define the smoothing zone. Therefore, $\tilde{d}(P,C_A) = 0$ entails that *P* belongs to the sphere of radius δ_S centred in *C*. It results that the location of points such that $\tilde{d}(P,C_A) = 0$ consist of this sphere, inside the smoothing zone, and of C_A, outside it. We so obtain a smoothed contact surface, as presented in Figure 24.

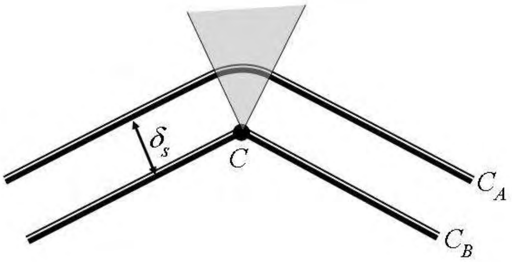


Figure 24: Smoothing of the contact surface.

Therefore, the smoothing procedure is quite simple: first construct the transformed contact surface by translation, and then use the new distance function. When the contact surface is made of several planes, the same transformation is applied to each plane, so the transformed contact surface is easily constructed.

The above description is only valid for concave surfaces. For convex surfaces, a symmetric procedure is utilized. First, we also consider that C_A is made of only two planes, as in Figure 25. The transformed contact surface C_C is constructed a similar way like C_B, but the transformation by the distance δ_S of π_A and $\pi_{A'}$ is carried out in the opposite directions, $-n_A$ and $-n_A$'.

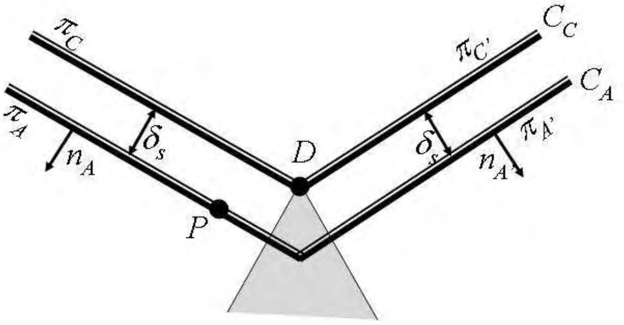


Figure 25: Third case of a convex angle.

In this case, the new distance function is defined as:

$$\tilde{\tilde{d}}(P,C_A)=d(P,C_C)-\delta_s \tag{III-18}$$

Figure 26 shows that the same kind of smoothing is obtained by replacing d by $\tilde{\tilde{d}}$.

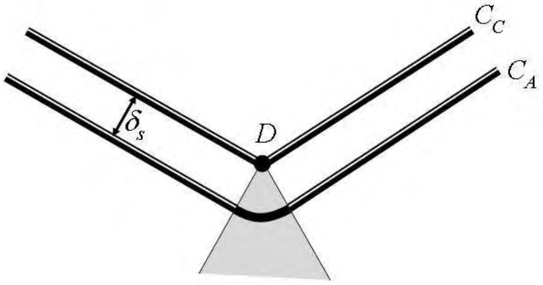


Figure 26: Smoothing of the contact surface in the convex case.

More generally, discretized 2D surfaces are made of both concave and convex angles. Therefore, we have to construct the two transformed contact surfaces, C_B and C_C and use the two new distance functions, $\tilde{d}$ and $\tilde{\tilde{d}}$. Outside the concave and convex smoothing zones, we have:

$$\tilde{\tilde{d}}(P,C_A)=\tilde{d}(P,C_A)=d(P,C_A) \tag{III-19}$$

So any distance function can be used. In 2D, an angle is either concave or convex, so a point cannot belong simultaneously to both smoothing zones. If it belongs to the concave smoothing zone, the $\tilde{d}$ function is used, whereas the $\tilde{\tilde{d}}$ one is preferred if it belongs to the convex one.

Remark 1: The smoothing radius provided by this method is exactly equal to δ_s. Therefore, the smoothing intensity is controlled by the value of δ_s. However, δ_s cannot take any value, as illustrated in Figure 27. It is limited by a portion of the smallest edge length in order that the transformation of a segment remains a segment with the same orientation.

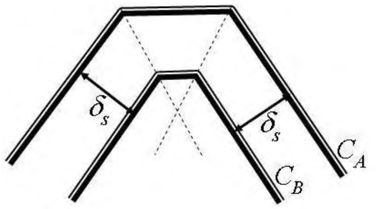


Figure 27: Limit of the contact surface transformation.

Remark 2: C_A is generally defined by a surface mesh, which so consists of facets and nodes. C_B and C_C are similar. They only differ from C_A by the position of their nodes, which are calculated by the transformation presented above. The topologies of C_A , C_B and C_C are identical. Therefore, the contact analyses that are necessary to evaluate the $\tilde{d}$ and $\tilde{\tilde{d}}$ functions, are exactly the same as for the d function. The same algorithm can be used. The present approach requires two contact analyses instead of one, although the additional two can be efficiently carried out using data from the first one.

2.3 The 3D problem

2.3.1 Transformed contact surfaces in 3D

For actual 3D configurations, the method is slightly more complex, both for the construction of the transformed surfaces C_B and C_C, and for the definition of the new distance function.

Ideally, each facet of C_B and C_C is parallel to the original facet of C_A, as it is the case in 2D. However, a node P of C_A is generally connected to more than 3 facets, which generally have different normal directions, as shown in Figure 28. Therefore, when the planes associated to each facet are translated by the distance δ_s in the directions n_A or $-n_A$, they do not intersect at a single point, as it is the case in 2D or when they are only 3 different normal directions. They are several intersection points, each of which is the location of the transformation of P on C_B or C_C for a certain combination of 3 facets with different normal directions. In order to build C_B and C_C, we select the barycentre of these points as the location of the transformation of *P*. It results that each facet of C_B and C_C is not strictly parallel to its original facet of C_A.

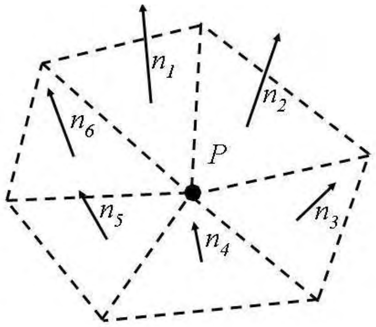


Figure 28: Facets of C_A in actual 3D for a surface discretization with triangles.

2.3.2 Distance function for general 3D surfaces

Consequently, equation (III-15) is not strictly verified outside the smoothing zones. It is no longer strictly equivalent to use the $\tilde{d}$ function or the $\tilde{\tilde{d}}$ one. The distance function is then modified into the $\bar{d}$ function outside the smoothing zones, as follows, by the average of these two functions:

$$\bar{d}(P,C_A) = \frac{1}{2}\tilde{d}(P,C_A) + \frac{1}{2}\tilde{\tilde{d}}(P,C_A) \qquad \text{(III-20)}$$

In 3D, the smoothing zones are also more complex. They are defined as the parts of space for which the orthogonal projection of any point *P* on the surfaces C_B or C_C belongs to an edge. It results that *P* can simultaneously belong to a smoothing zone of C_B and C_C. In other words, the intersection of the smoothing zones is not void as in 2D. To summarize, in the concave smoothing zones, we want to have $\bar{d} = \tilde{d}$. In the convex smoothing zones, we expect $\bar{d} = \tilde{\tilde{d}}$. And in the mixed smoothing zones, we are looking for a combination of both:

$$\bar{d} = \alpha\tilde{d} + (1-\alpha)\tilde{\tilde{d}} \qquad \text{(III-21)}$$

where α is a positive coefficient smaller than 1. α allows favouring the $\tilde{d}$ or $\tilde{\tilde{d}}$ function, for instance according to the intensity of smoothing provided by each of them. On the other hand, the $\bar{d}$ function must be continuous all over the space, throughout all smoothing zones.

In the smoothing zone, the intensity of smoothing is measured by the angle θ as presented in Figure 29. For any point *P* belonging to a concave smoothing zone, *P'* and *C* respectively are the orthogonal projections of P onto C_A and C_B. Then, θ_B is the angle between $\overrightarrow{P'P}$ and $\overrightarrow{PC}$, this is to say between the directions of the normal projections provided by d and $\tilde{d}$. θ_C is defined is a similar way in the convex smoothing zones.

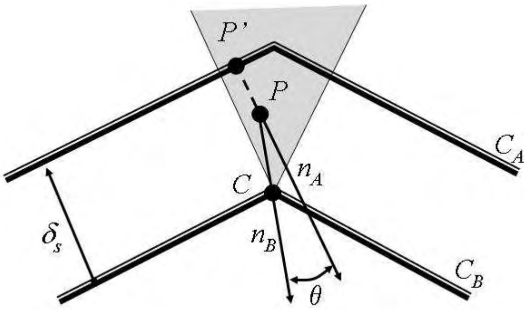


Figure 29: Definition of the θ_B angle.

The $\bar{d}$ function is then defined by:

$$\bar{d} = \frac{\beta_B}{\beta_B + \beta_C}\tilde{d} + \frac{\beta_C}{\beta_B + \beta_C}\tilde{\tilde{d}} \qquad \text{(III-22)}$$

where the β_B and β_C coefficients are defines as follows:

$$\beta_B = max\left(\frac{\lambda_B}{\lambda_B + \lambda_C};\frac{\lambda_{lim} - \lambda_C}{\lambda_{lim}}\right) \text{ and } \beta_C = max\left(\frac{\lambda_C}{\lambda_B + \lambda_C};\frac{\lambda_{lim} - \lambda_B}{\lambda_{lim}}\right) \qquad \text{(III-23)}$$
$$\text{where: } \lambda_B = 1 - cos\,\theta_B \text{ and } \lambda_C = 1 - cos\,\theta_C$$

The λ_{lim} coefficient is introduced to allow for a better continuity between the non smoothing and smoothing zones. If P does not belong to any smoothing zone, $\theta_B = \theta_C = 0$, $\lambda_B = \lambda_C = 0$, $\beta_B = \beta_C = 1$, and then $\bar{d} = \frac{1}{2}\tilde{d} + \frac{1}{2}\tilde{\tilde{d}}$. If P belongs to a single smoothing zone, for instance the concave one, $\theta_B \neq 0$ and $\theta_C = 0$, $\lambda_B = 1 - cos\,\theta_B$ and $\lambda_C = 0$. If $\lambda_B < \lambda_{min}$, then $\beta_B = 1$ and $\beta_C = \frac{\lambda_{lim} - \lambda_B}{\lambda_{lim}}$, so $\bar{d} = \frac{\lambda_{lim}}{2\lambda_{lim} - \lambda_B}\tilde{d} + \frac{\lambda_{lim} - \lambda_B}{2\lambda_{lim} - \lambda_B}\tilde{\tilde{d}}$. If now $\lambda_B \geq \lambda_{min}$, then $\beta_C = 0$, and then $\bar{d} = \tilde{d}$.

2.4 Remaining 3D difficulties

The formulation presented above is very simple to implement and is very efficient for geometries which are not too complex. However, there are some limitations due to the following facts:

- As mentioned in remark 1 of section 2.1, the translating distance δ_S is limited by the size of the smallest edge length, which can be locally very small with an unstructured mesh, so limiting the smoothing intensity.
- The translating distance δ_S is the only parameter governing the smoothing. The smoothing can not be local or in a specified direction. Therefore, preserving an exact angle (not due to space discretization) of a tool surface is not possible, which is limiting the field of application of this smoothing procedure.

3 *Benchmark test and application*

This smoothing procedure has been implemented and in order to visualize its effect, we study the indentation of a cube with different tools that have very sharp angles. During the process, the material comes in contact with the tool. It is finely discretized with refined meshes that are updated by an automatic remeshing procedure. Its shape reflects the contact surface that has been perceived by the algorithm.

3.1 Concave angle smoothing

Figure 30 presents the indentation of a cube by a pyramid. Its second picture (up-right) shows that the tip of the pyramid is perceived as a hemisphere. The other pictures show the intensity of the smoothing.

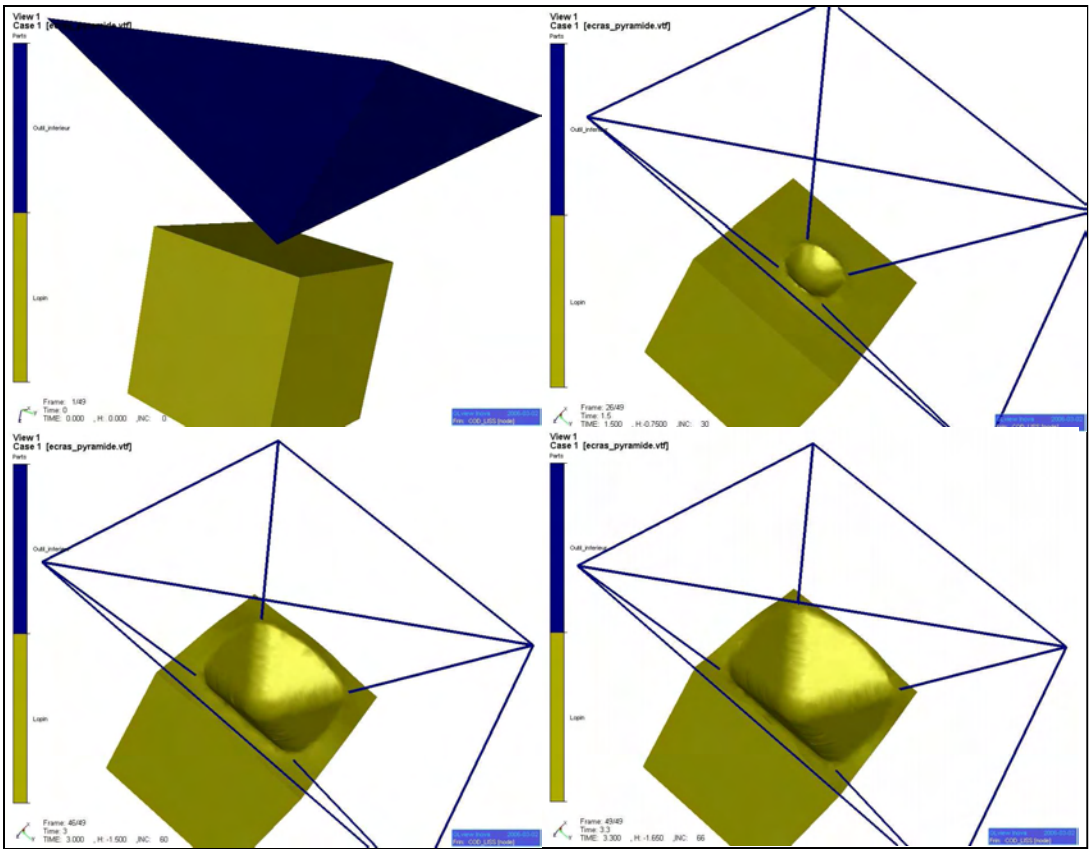


Figure 30: Indentation of a cube with a pyramid.

3.2 Convex angle smoothing

In Figure 31, the tool is also convex. A cylinder is very coarsely discretized by few facets. However, the material deformation seems to result from the indentation of a virtually perfect cylinder.

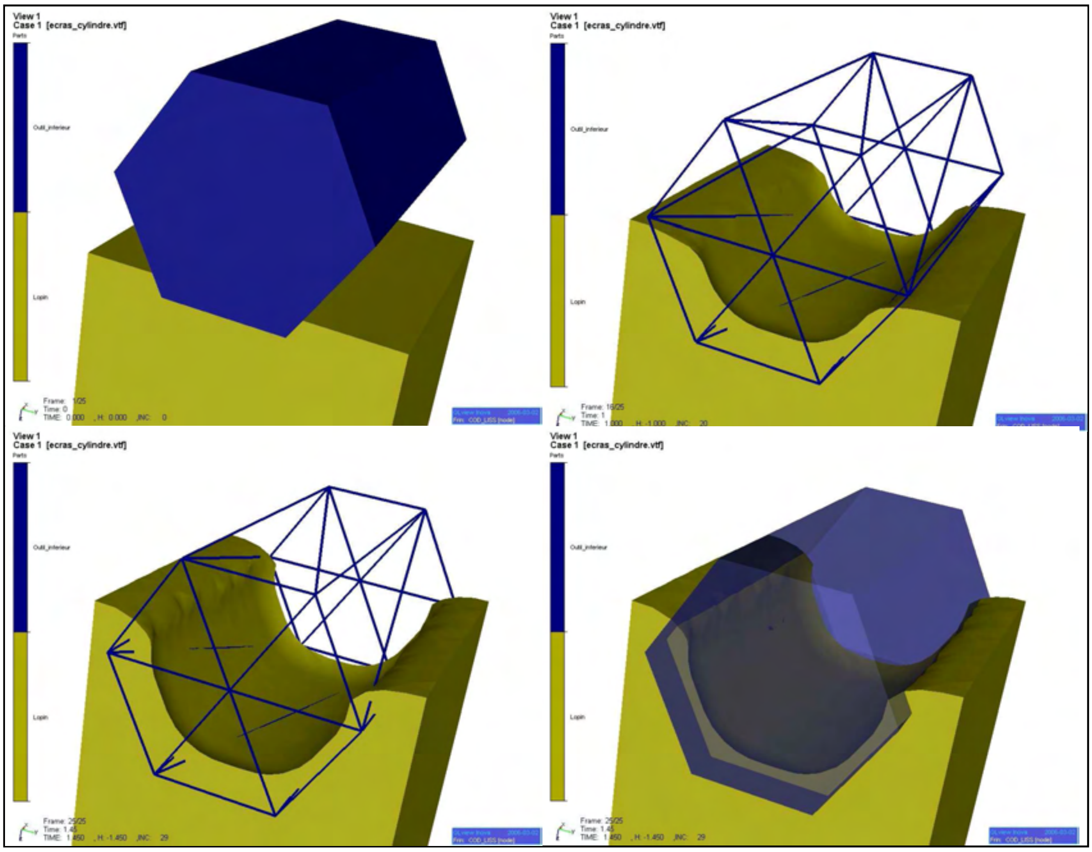


Figure 31: Indentation of a cube with a facetized cylinder.

3.3 Concave-convex angle smoothing

The tool utilized in Figure 32 combines convex and concave parts. It shows how the combination of the two smoothing procedures (the concave and convex ones) produces a continuous surface.

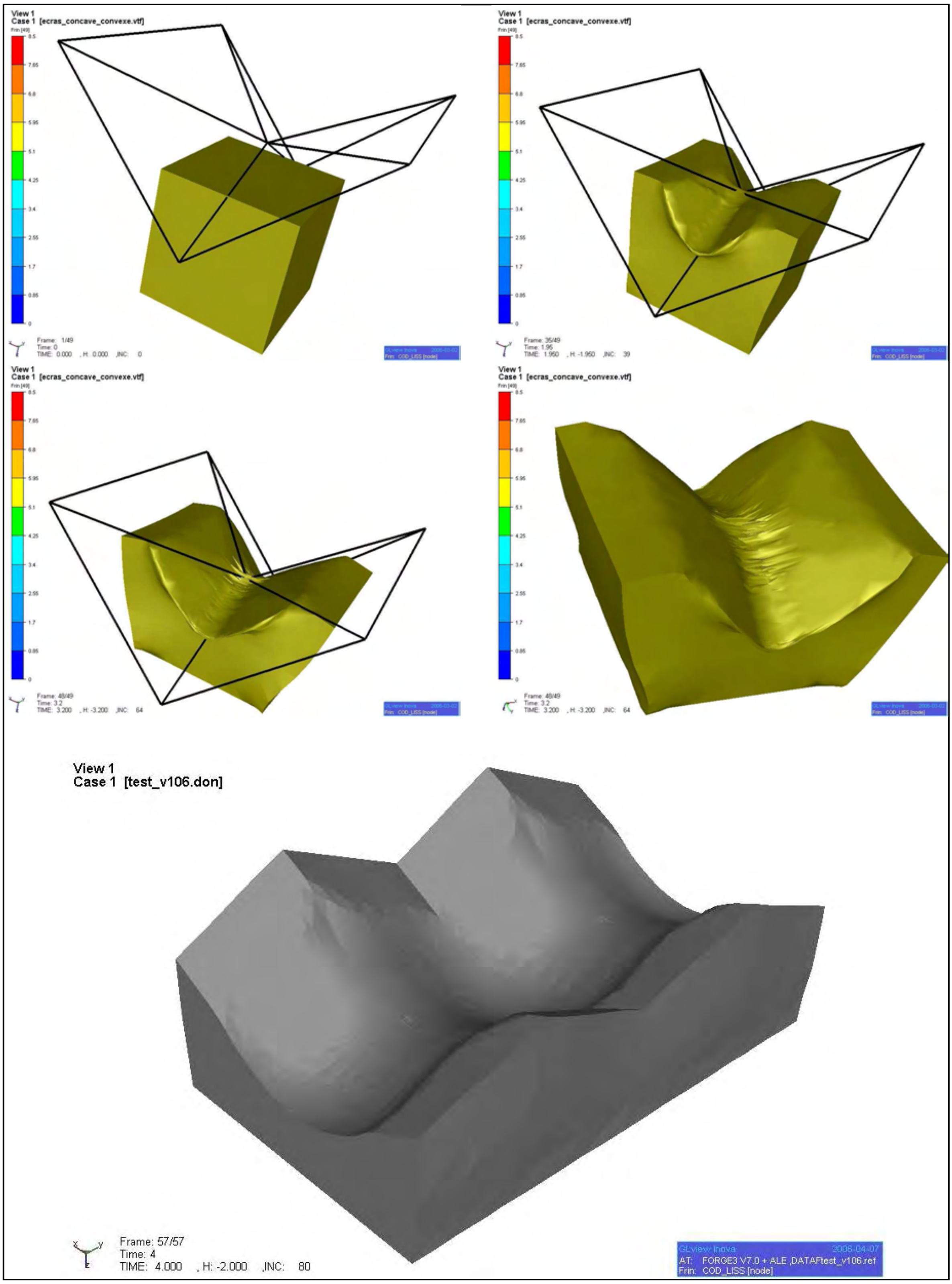


Figure 32: Indentation of a cube by a tool that combines concave and convex facets (on top). Indentation of the complementary surface of the same previous tool design with symmetry plan on 2 lateral faces.

3.4 First application

Figure 33 shows an actual application of the smoothing procedure to a metal forming problem. In orthogonal cutting, the tool radius plays a critical role on the cutting force and more generally on the chip formation [28]. In order to simplify the design of the tool, the cutting angle is simply defined by two intersecting planes. The tool radius only results from the smoothing procedure. It effects are clearly seen in the pictures of Figure 33. The resulting distribution of the equivalent stain rate is very similar to the one obtained with a more classical discretization of the tool.

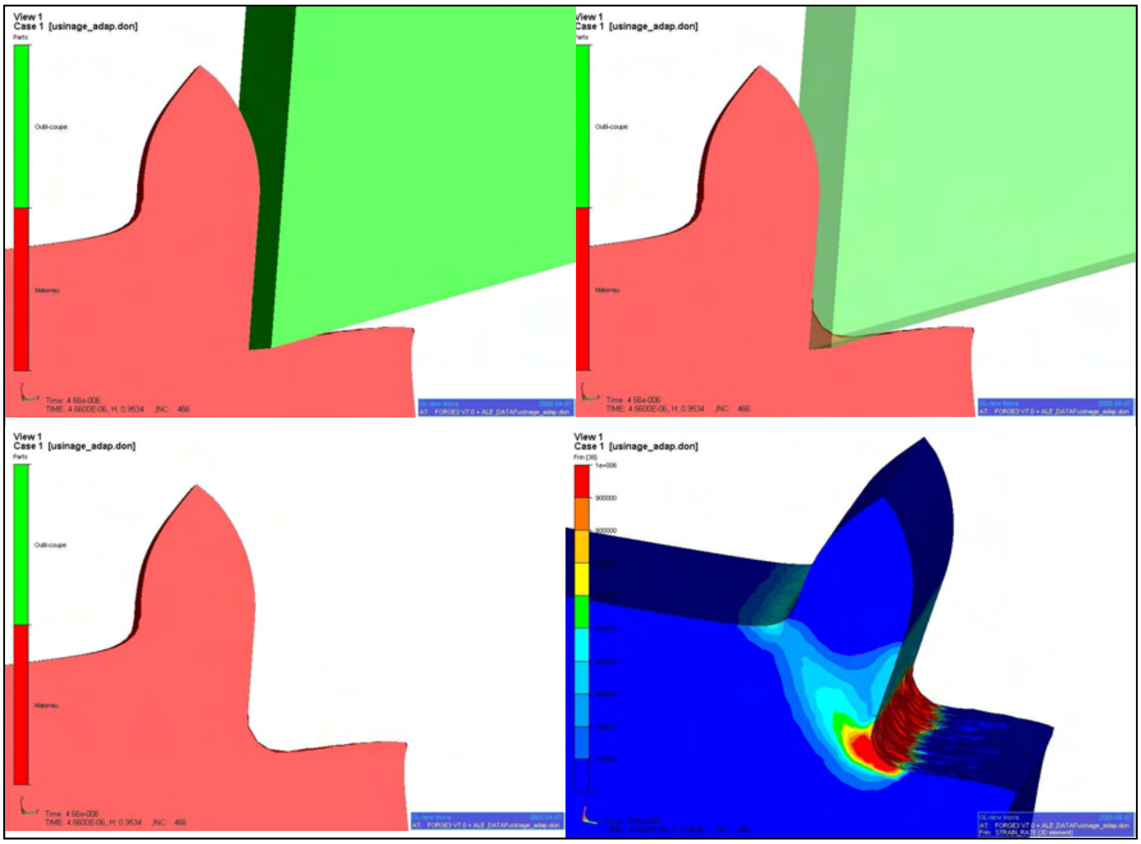


Figure 33: Simulation of the orthogonal cutting process. The smoothing procedure is utilized to introduce the cutting radius of the tool. Visualization of the equivalent strain rate (low-right).

Other applications are carried out in FSW in chapter 5.

As a conclusion, this smoothing procedure is both simple and efficient when the obstacle is discretized by a uniformly refined surface mesh. Its interest is not as clear, under its present implementation, when this mesh exhibits local refinements.

Chapter IV : ALE formulation

1 Background

The choice of an appropriate grid description is fundamental when developing a computer code. The 3D numerical simulation of the FSW requires coping with strong distortions of the material while allowing for a clear modelling of free surfaces and structure interfaces. The algorithms of non linear mechanics usually make use of two classical descriptions of motion: the Lagrangian and the Eulerian ones. The Arbitrary Lagrangian Eulerian description, which is the subject of this chapter, was developed in an attempt to combine the advantages of the above classical descriptions, while minimizing their respective drawbacks as far as possible.

1.1 Lagrangian, Eulerian and ALE description

Three domains can be used to describe motion in continuum mechanics:

- the material domain R_X, associated to material particles $\mathbf{X}$ and regarded as fixed in time,
- the spatial domain R_x, which is associated to spatial points $\mathbf{x}$,
- the referential domain R_χ, which is associated to grid nodes $\boldsymbol{\chi}$.

The Lagrangian description consists in following the material particles during their motion. Each individual node of the computational mesh follows the same permanently connected material particle. Their locations are updated using the computed material velocity $\mathbf{v}$ defined by:

$$\mathbf{v} = \mathbf{v}(\mathbf{X}, t) = \left.\frac{\partial \mathbf{x}}{\partial t}\right|_{\mathbf{X}} \qquad \text{(IV-1)}$$

with $\left.\right|_{\mathbf{X}}$ meaning “holding the material coordinate $\mathbf{X}$ fixed”.

The domain boundaries are then automatically tracked by the grid. Figure 34 illustrates the evolution of a Lagrangian mesh configuration M_{LAG} for a time step Δt. This description, which is widely used in structural mechanics, also facilitates the treatment of materials with history-dependent constitutive relations. Its weakness is its inability to follow large distortions in the meshed domain without recourse to frequent and often complex remeshing operations, and associated transfer.

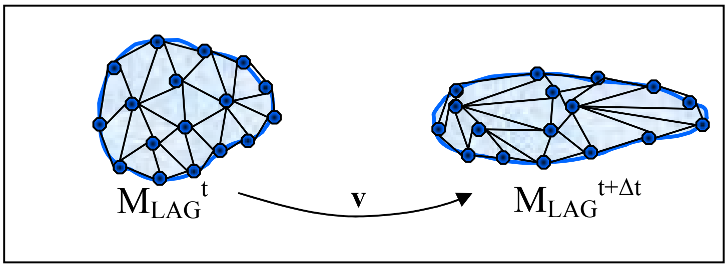


Figure 34: schematic representation of an updated Lagrangian description.

The difficulties caused by an excessive distortion of the finite element grid are overcome in Eulerian description, which is widely used in fluid dynamics. The computational mesh is fixed and the material domain moves through the grid. The velocity **v** is consequently expressed with respect to the fixed-element mesh without any reference to the initial configuration of the material coordinates **X**:

$$\mathbf{v} = \mathbf{v}(\mathbf{x}, t) \tag{IV-2}$$

It thus allows for large distortions to be handled with relative ease. But since the mesh velocity is null, it is commonly different from the velocity of the material particles and this relative motion forces to deal with convective effects which appear. So, specific solvers are usually developed to tackle with nonsymmetric character of convection operators. Furthermore, serious difficulties are found in following deforming material interfaces and mobile boundaries if the flow is not confined. Figure35 illustrates the fact that continuum boundaries are not inevitably coinciding with the Eulerian mesh configuration M_{EUL}.

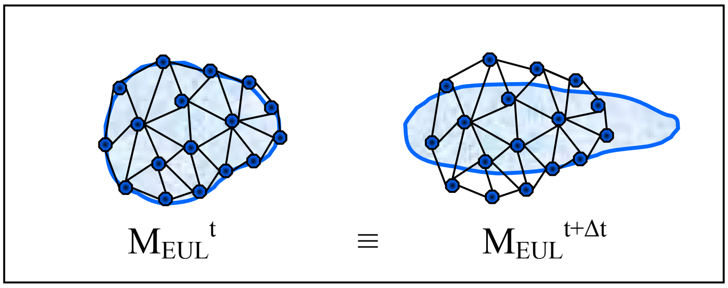


Figure 35: schematic representation of an Eulerian description.

The ALE formulation succeeds, to a certain extent, in combining the best features of both Lagrangian and Eulerian approaches. Neither the material configuration $R_{\mathbf{X}}$, nor the spatial configuration $R_{\mathbf{x}}$ is taken as the reference. The nodes can be moved with the material as in the normal Lagrangian description, or be held fixed in Eulerian manner, or, be moved in some arbitrary specified way to give a continuous rezoning capability. The mesh velocity is thus defined by:

$$\mathbf{w} = \left.\frac{\partial \mathbf{x}}{\partial \mathrm{t}}\right|_{\chi} \tag{IV-3}$$

Because of this freedom in moving the computational mesh offered by the ALE description, greater distortions of the material can be handled, while following boundaries of the domain. Figure 36 shows the evolution of an Arbitrary Lagrangian Eulerian mesh configuration M_{ALE} for a time step Δt, and compared to Figure 34 illustrates the possibility of mesh rezoning. The mesh velocity can also be computed in order to concentrate elements in zones of steep solution gradient.

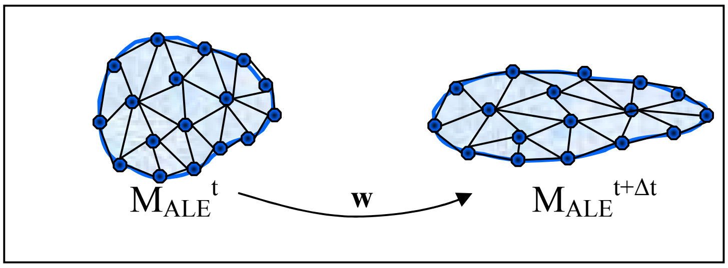


Figure 36: schematic representation of an Arbitrary Lagrangian Eulerian description.

But the ALE formulation also combines the computational difficulties of the Lagrangian and Eulerian approaches. The treatment of convective terms, common in an Eulerian formulation, and the determination of mesh updating procedures, which is a problem usually encountered when remeshing is necessary in a Lagrangian description, are the two main difficulties that ALE formulation has to tackle with.

The convective terms appear in all time derivative values. They are due to the difference between the material **v** and mesh **w** velocities. Equation (IV-4) gives the expression of the time derivative value for a scalar physical quantity f. One can refer to [29] for a complete mathematical introduction of the material, spatial, and referential time derivatives.

$$\begin{cases} \dfrac{\mathrm{d}f}{\mathrm{dt}} = \dfrac{\mathrm{d}_g f}{\mathrm{dt}} + \mathbf{c}.\nabla f \\ \mathbf{c} = \mathbf{v} - \mathbf{w} \end{cases} \tag{IV-4}$$

$\frac{\mathrm{d_g}f}{\mathrm{dt}}$ is called the grid derivative.

One can verify that:

- in the Lagrangian description, the time derivative is equal to the grid derivative (see equation (IV-5)). Since the material points coincide with the same grid points during the whole motion, there are no convective effects in the Lagrangian calculations.

$$\begin{cases} \frac{\mathrm{d}f}{\mathrm{dt}} = \frac{\mathrm{d}_g f}{\mathrm{dt}} \equiv \frac{\partial f}{\partial t} \\ \mathbf{w} = \mathbf{v} \end{cases} \tag{IV-5}$$

- in the Eulerian description, convective terms exist and the convective velocity **c** is equal to the material velocity (see equation (IV-6)).

$$\begin{cases} \frac{\mathrm{d}f}{\mathrm{dt}} = \frac{\mathrm{d}_g f}{\mathrm{dt}} + \mathbf{v}.\nabla f \\ \mathbf{w} = 0 \end{cases} \tag{IV-6}$$

1.2 ALE method in literature

In literature, two approaches are commonly used for the resolution of the ALE formulation:

- the direct approach (generally used in CFD softwares), which consists in solving the convection diffusion equations.

- the uncoupled or splitting approach (generally used in solid-mechanics softwares), in which the convective terms are treated separately in a different way.

Note that none of these methods is known to compute in a strictly coupled way the material and grid velocities. For computational efficiency and stability reasons, this latter is always calculated through a mesh regularization or adaptation procedure, which is uncoupled with the resolution of the mechanical problem providing the material velocity.

1.2.1 Direct Method: resolution of convection diffusion equations

The direct method derives from the resolution of the well-known Eulerian problem in which convective terms always exist. The differential forms are obtained by introducing the time derivative (IV-4) into the conservation equations of mass (II-3), momentum (II-6), and energy (II-59). It leads to the following problem:

$$\begin{cases} \dfrac{d_g \rho}{dt} + \mathbf{c}.\nabla \rho + \rho \,\mathrm{div}\, v = 0 \\ \rho \dfrac{d_g v}{dt} + \rho\, \mathbf{c}.\nabla v = \mathrm{div}\, \sigma + f \\ \rho\, c \dfrac{d_g T}{dt} + \rho\, \mathbf{c}.\nabla T - \mathrm{div}\,(k \nabla T) = E \end{cases} \tag{IV-7}$$

One can refer to [30] for a bibliographical study of many resolution techniques of the convection-diffusion equations, which were developed in literature. Among them can be quoted the SUPG (Streamline Upwind Petrov-Galerkin) method, the Taylor/Galerkin method, the *characteristics*/Galerkin method, the ST/SGS (Space Time/ Subgrid Scale) methods, or ELLAM methods (Eulerian-Lagrangian Localized Adjoint Method) more recently developed in multi-dimension [31]. Note that the Discontinuous Galerkin method is only available for pure convective equations. These methods will not be detailed in this manuscript.

1.2.2 Splitting Method

In the uncoupled approach, the treatment of convective terms is splitted from the global system resolution. The main advantage, with respect to the coupled approach, is the simplification it provides to the matricial system resolution. Since the stiffness matrix does not contain convective terms, the standard Lagrangian method can be used for solving the system. Thus, the uncoupled ALE method can be adapted to Lagrangian codes with relative ease.

Three main steps results from this split:

- a Lagrangian step: the weak form of the equilibrium and thermal equations are solved (as seen in Chapter II :).
- the mesh velocity **w** is computed.
- a remapping step.

This latter step is another manner to treat the convective terms due to the difference between grid and material velocities. The values of the computed state variables are remapped on the ALE updated mesh $M_{ALE}^{t+\Delta t}$. Integrating the grid time derivative (see equation (IV-8)), or interpolating the values of the updated Lagrangian configuration $M_{L}^{t+\Delta t}$ on the new ALE configuration $M_{ALE}^{t+\Delta t}$ are two different approaches for remapping.

$$f^{t+\Delta t} = f^{t} + \int_{\Delta t} \left(\frac{df}{dt} - \mathbf{c}.\nabla f \right) dt \tag{IV-8}$$

Finally, the splitting ALE method is equivalent to an incremental remeshing step in which the topology of the mesh remains the same.

The two main difficulties of such a method are:

- the calculation of the mesh velocity. Adaptation, regularization and volume conservation may be three conditions difficult to combine.
- the remapping step. It has to be accurate enough to not disturb the equilibrium equations and not introduce too much numerical diffusion.

Procedures developed to overcome these two difficulties are discussed in the next two sections.

2 *Mesh velocity computation*

2.1 Background

Three types of adaptivity can be distinguished: the R-adaptivity (modification of the nodal locations while preserving the topology of the grid), the P-adaptivity (increase in the polynomial degrees of the interpolation), and the H-adaptivity (change of size of the mesh by refinement-unrefinement). The ALE method, which requires the calculation of the grid velocity, is equivalent to a R-adaptation. Since the connectivity is preserved, this procedure is much faster than a complete remeshing. The calculation of the grid velocity can explicitly take into account geometrical criteria aiming at preserving the quality of the elements. It may also be computed in an adaptative way using criteria based on error estimation. Different existing methods are detailed in [32].

2.1.1 Mesh regularization

Mesh regularization requires that the updated nodal coordinates be specified at each step of the calculation, either through their displacements, or from the current mesh velocities **w**.

2.1.1.1 Implicit Approach: discretized Laplace Methods

The most popular implicit approach consists in minimizing the gradient of the velocity or the gradient of the nodal density of the grid. The rezoning of the mesh nodes may consist in solving a Laplace equation for each component of the node velocity or position, so that on a logically regular region the mesh forms lines of equal potential:

$$\Delta \mathbf{w} = 0 \quad \text{on} \quad \Omega_{\text{ALE}} \quad \text{with} \quad \mathbf{w} = \mathbf{w}^0 \quad \text{on} \quad \partial\Omega_{\text{ALE}} \qquad \text{(IV-9)}$$

A harmonic function (solution of the Laplace equation) is framed by its boundary values (principle of maximum), therefore the velocity solutions of (IV-9) are framed by the velocities at the boundaries of the computational domain. Consequently the nodes which are moved according to the solution displacements cannot cross these boundaries except if the computational domain is convex (else there is no guaranty). Techniques to preclude this pitfall either increase the computational cost or introduce new terms in the formulation, which are specific to each geometry.

A FE based discretization method of the Laplace equation consists in minimizing the following potential *I*:

$$I = \int_{\Omega} \frac{1}{2} (\nabla \mathbf{w}) : (\nabla \mathbf{w}) \, \mathrm{d}\omega + \int_{\partial\Omega} \lambda (\mathbf{v} - \mathbf{w}).\mathbf{n} \, \mathrm{d}S \qquad \text{(IV-10)}$$

Boundary conditions are then imposed through the Lagrange multiplier λ which is null inside the domain. The minimization leads to a linear system of equation providing **w** [18].

$$\begin{cases} \int_{\Omega} (\nabla \mathbf{w}) : (\nabla \mathbf{w}) \, \mathrm{d}\omega - \int_{\partial\Omega} \lambda \, \mathbf{n}.\mathbf{v}^* \mathrm{d}S = 0 & \forall \mathbf{v}^* \\ \int_{\partial\Omega} (\mathbf{v} - \mathbf{w}).\mathbf{n} \, \lambda^* \mathrm{d}S = 0 & \forall \lambda^* \end{cases} \qquad \text{(IV-11)}$$

2.1.1.2 Explicit Approach

In fact, it is possible to use any mesh-smoothing algorithm that is designed to improve the shape of the elements for a fixed topology. The centring method, applied to coordinates or velocities, is a simple and fast method to regularize the grid. In the “geometrical splitting” variant, the position of each interior node is chosen as close as possible to the gravity centre of the polygon joining its neighbour nodes. In the “kinematical splitting” variant, the velocity of a node is respectively chosen equal to the average velocities of its neighbour nodes. The method can be regarded as a minimization of the potential *I*:

$$\underset{\mathbf{w}}{\mathrm{Min}}\, I = \underset{\mathbf{w}}{\mathrm{Min}} \sum_{\mathrm{m}=1,\mathrm{Nbnoe}} \left\| \mathbf{w}_{\mathrm{m}} - \overline{\mathbf{w}}_{\mathrm{m}} \right\|^2$$

$$\text{with} \quad \left| \begin{array}{l} \overline{\mathbf{w}}_{\mathrm{m}} = \dfrac{1}{|\Gamma_{\mathrm{m}}|} \sum_{\mathrm{n}\in\Gamma_{\mathrm{m}}} (\mathbf{w}_{\mathrm{n}}) \\ \mathbf{w} = \mathbf{w}^0 \ \text{ on } \partial\Omega_{\mathrm{ALE}} \end{array} \right. \qquad \text{(IV-12)}$$

$\overline{\mathbf{w}}_{\mathrm{n}}$ is the average mesh velocity of the neighbour nodes (belonging to Γ_{m}) of the node m. Γ_{i} is the set of the nodes connected to the node i ; $|\Gamma_{\mathrm{i}}|$ is its cardinal.

This minimization can be solved through an iterative procedure. This approach is actually equivalent to a discretization of the Laplace equation. On a uniform regular 2D grid, one obtains:

$$\Delta \mathbf{w} = \frac{\partial^2 \mathbf{w}}{\partial \mathrm{x}^2} + \frac{\partial^2 \mathbf{w}}{\partial \mathrm{y}^2} \approx \frac{(\mathbf{w}_{\mathrm{i-1,j}} - 2\,\mathbf{w}_{\mathrm{i,j}} + \mathbf{w}_{\mathrm{i+1,j}}) + (\mathbf{w}_{\mathrm{i,j-1}} - 2\,\mathbf{w}_{\mathrm{i,j}} + \mathbf{w}_{\mathrm{i,j+1}})}{\mathrm{h}^2} = 0$$

$$\Leftrightarrow \quad \mathbf{w}_{\mathrm{i,j}} = \frac{1}{4} \left(\mathbf{w}_{\mathrm{i-1,j}} + \mathbf{w}_{\mathrm{i+1,j}} + \mathbf{w}_{\mathrm{i,j-1}} + \mathbf{w}_{\mathrm{i,j+1}} \right) \qquad \text{(IV-13)}$$

And we see that $\mathbf{w}_{\mathrm{i,j}}$ is the average of its four neighbour nodes.

2.2 Mesh adaptation

When the ALE description is used as an adaptive technique, the objective is to optimize the computational mesh to achieve the best possible accuracy, for the lowest computational cost. Mesh refinement is typically carried out by moving nodes towards zones of strong solution gradient. The ALE algorithm then includes an indicator of the error, and the mesh is modified to obtain an equi-distribution of the error over the entire computational domain.

The strategy, which has been developed in this work, tries to combine mesh adaptation and mesh regularization.

2.3 Boundary conditions

In the ALE formulation, the volume conservation is the only coupling condition between the grid velocity and the material velocity: the grid evolves with respect of the boundaries of the computational domain at any moment.

$$\forall t: \quad \partial\Omega^t_{ALE} \cong \partial\Omega^t_L \tag{IV-14}$$

Thus, the computation of **w** can be splitted in two: at the boundary and in the volume of the domain. Once the velocity or the position of the boundary nodes is prescribed under the constraint (IV-14) that no particle cross the material boundary, the velocity or the position of the volume nodes can be calculated in a regularizing or adaptive way as seen in the previous paragraph.

The simplest method to enforce this condition consists in imposing the material velocity at boundary nodes:

$$\forall m \in \partial\Omega_h, \mathbf{w}_m = \mathbf{v}_m \tag{IV-15}$$

However, it does not allow adapting or regularizing the mesh at the domain surface, which is essential for most of the processes where boundaries are often crucial flow zones. In the FSP, high tangential distortions are involved in surface elements. They are due to frictional or sticking conditions between the workpiece and the tool shoulder. Therefore, this method is very limiting, and corollary, it is very important to regularize the surface mesh.

Traoré et al. [33] have developed a “geometrical” splitting method consisting in moving the grid on a parameterized B-spline surface: it is a quasi-stationary parametric reactualization.

First, the surface S of the updated Lagrangian mesh $M_L^{t+\Delta t}$ is interpolated by a B-splines cubic tensorial technique with a particular treatment for C^0 edges. Then the nodes of the initial mesh M_{ALE}^t are projected on S in such a way that their displacement is minimized. A schematic representation is given in Figure 37, where the parameter governing the projection is the angle between a prescribed plane of the part and nodes. The mesh velocity is thus calculated in an indirect way:

$$\mathbf{w}^{t+\Delta t}(x) = \frac{\mathbf{x}_{ALE}^{t+\Delta t} - \mathbf{x}_{ALE}^{t}}{\Delta t} \qquad \text{(IV-16)}$$

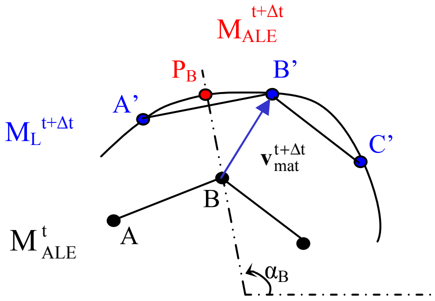


Figure 37: Geometrical splitting method: projection on B-Spline surface

Once again, this technique cannot be applied to FSW simulation. This surface smoothing procedure is hardly extendable to problems with complex geometries.

2.3.1 Consistent normal

The conservation of the domain shape must be locally satisfied for any part Γ of the boundary surface. This condition can be written:

$$\forall \Gamma \supset \partial\Omega_{ALE}^t \quad \int_\Gamma (\mathbf{v}-\mathbf{w}).\mathbf{n}\, dS = 0 \quad \Leftrightarrow \quad \int_\Gamma \mathbf{v}.\mathbf{n}\, dS = \int_\Gamma \mathbf{w}.\mathbf{n}\, dS \qquad \text{(IV-17)}$$

After discretization and considering that Γ is as small as possible, one can write, for each boundary node m, and introducing a consistent normal $\tilde{\mathbf{n}}_m$:

$$\forall t \text{ and } \forall m \in \partial\Omega_{ALE}^t : \qquad \mathbf{w}_m^{t+\Delta t}.\tilde{\mathbf{n}}_m = \mathbf{v}_m^{t+\Delta t}.\tilde{\mathbf{n}}_m \qquad \text{(IV-18)}$$

The expression of the consistent normal $\tilde{\mathbf{n}}_m$ results directly from the flux conservation equation (IV-17). Bellet et al. [34] has shown that for linear elements this consistent normal is the average of the normals $\mathbf{n}_f$ of the facets f contiguous to the node m, balanced by their respective surface S_f:

$$\tilde{\mathbf{n}}_m = \frac{1}{\left\| \sum_{f\in\Gamma_m} S_f \mathbf{n}_f \right\|} \sum_{f\in\Gamma_m} S_f \mathbf{n}_f \tag{IV-19}$$

The first reflex is to enforce the mesh velocity only in the consistent normal direction:

$$\forall t \text{ and } \forall m \in \partial\Omega^t_{ALE} : \mathbf{w}_m^{t+\Delta t} = \left(\mathbf{v}_m^{t+\Delta t} \,.\, \tilde{\mathbf{n}}_m\right)\tilde{\mathbf{n}}_m \tag{IV-20}$$

This condition satisfies the condition (IV-18), but eliminates all tangential movements of regularization. Further more, if tangential component of the material velocity is not negligible, such a simple kinematics leads to instabilities (see illustration in Figure 38) [35]

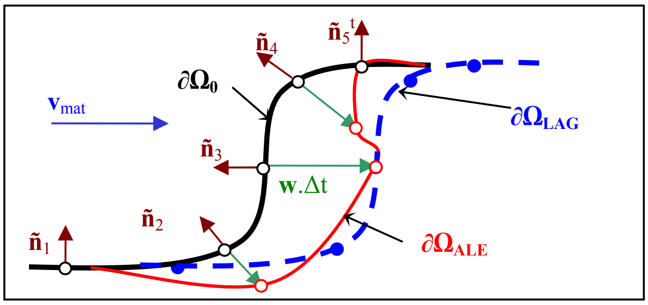


Figure 38: Schematic representation of instabilities generated by tangential movements of the surface.

If the tangential degrees of freedom are kept, they can be defined by the classical regularization procedure:

$$\forall t \text{ and } \forall m \in \partial\Omega^t_{ALE} : \quad \begin{cases} \mathbf{w}_m^{t+\Delta t} \,.\, \tilde{\mathbf{n}}_m = \mathbf{v}_m^{t+\Delta t} \,.\, \tilde{\mathbf{n}}_m \\ \mathbf{w}_m^{t+\Delta t} \,.\, \mathbf{t}_{im} = \dfrac{1}{|\Gamma_m|} \sum_{n\in\Gamma_m} (\mathbf{w}_m^{t+\Delta t} \,.\, \mathbf{t}_{im}) \quad ; \ i=1,2 \end{cases}$$

$$\text{with} \quad \begin{cases} \mathbf{t}_{1m} \wedge \mathbf{t}_{2m} = \dfrac{\tilde{\mathbf{n}}_m}{\|\tilde{\mathbf{n}}_m\|} \\ \mathbf{t}_{1m} . \mathbf{t}_{2m} = 0 \end{cases} \tag{IV-21}$$

But although this tangential criterion may reduce the instabilities, it does not guarantee complete volume conservation. The discretization of (IV-17) does not allow properly preserving the volume everywhere and partiticularly on edges and in corners.

Therefore, a specific algorithm has been developed, in order to define, for each boundary node, the directions in which it is crucial to impose the material velocity.

2.3.2 Specific algorithm for consistent normal computation

This algorithm has been developed with the aim of reducing the instabilities and avoiding numerical smoothing of edges. It consists in reducing the number of degrees of freedom of the grid velocity for "angular" nodes.

The condition (IV-18) forces the updated nodal position to be in a geometrical plane defined by the direction $\tilde{\mathbf{n}}$ (the consistent normal to the initial surface) and the signed distance $\mathbf{v}.\tilde{\mathbf{n}}$ to the initial nodal position (see Figure 39). So if the condition (IV-18) is applied for more than one normal, each direction (or "constraining" normal) will reduce by one the number of degrees of freedom in the tangential direction. If two different normals are utilized, the node automatically belongs to a line, which is the intersection of the two planes defined by the two normals. Keeping three different normals in 3D (two in 2D), means that the Lagrangian velocity is imposed to the node n as in (IV-15).

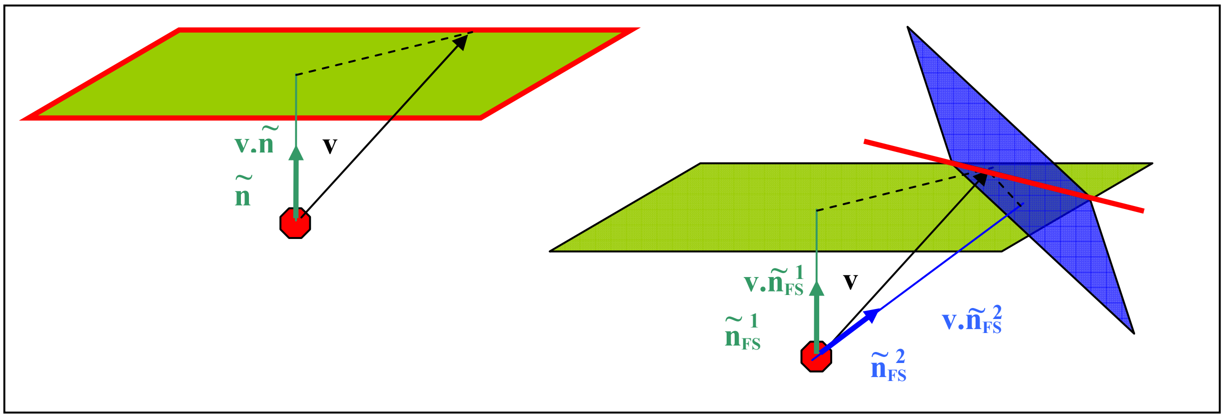


Figure 39: Graphic representation of possible locations of a node after the imposition of the condition (IV-18) for one or two normals

In 2D, the number of constraining normals can be easily determined trough a simple angular parameter α, defined by:

$$\alpha = \mathbf{n}_{FS}^1.\mathbf{n}_{FS}^2 \qquad \text{(IV-22)}$$

$\mathbf{n}_{FS}^1$ and $\mathbf{n}_{FS}^2$ are the normals of the two facets connected to the node.

- if $\alpha < \alpha_{\lim}$ the node is considered as angular and the condition (IV-18) has to be verified for the two normals,
- else the equation (IV-19) is used to compute a unique consistent normal.

But in 3D, the number of facets which are adjacent to a node is always higher than three, so the number of combination $\mathbf{n}_{FS}^{i}.\mathbf{n}_{FS}^{j}$ can be very large. Anyway, the main difficulty is in finding the best possible consistent normals that allow preserving the material flux and minimizing the number of imposed degrees of freedom. Therefore, a new procedure based on the minimization of the global flux across the surface of the computational domain $\partial\Omega$ has been developed.

The flux conservation condition (IV-17) is considered for each facets f of a surface patch Γ_m.

$$\forall f \in \Gamma_m \quad \int_f (\mathbf{v}-\mathbf{w}).\mathbf{n}\,dS = 0 \tag{IV-23}$$

Γ_m is the set of facets (represented in Figure 40) belonging to the domain boundary and containing the node m:

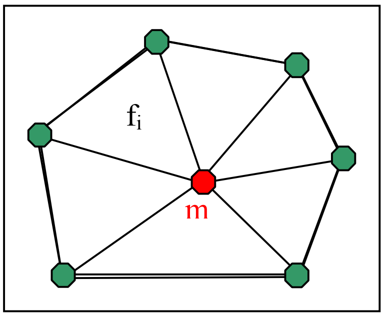


Figure 40: Nodal patch Γ_m

As the solution of the equation (IV-23) is not possible to obtain, the following expression is minimized:

$$\forall m \in \partial\Omega_{ALE}^{t} \quad \min_{\mathbf{w}_m} \Phi_m = \frac{1}{2}\sum_{f\in\Gamma_m}\left(\int_f (\mathbf{w}-\mathbf{v}).\mathbf{n}dS\right)^2 \tag{IV-24}$$

Note that if: $\Phi = \sum_{m\in\Omega_h}\Phi_m$ then $\dfrac{\partial\Phi}{\partial\mathbf{w}_m} = \dfrac{\partial\Phi_m}{\partial\mathbf{w}_m}$ (IV-25)

And so this local minimization is equivalent to a global one of the flux on all the facets of the domain.

Therefore, the grid velocity $\mathbf{w}_m$ of the node m has to nullify the following derivative:

$$\frac{\partial\Phi_m}{\partial w_{im}} = \sum_{f\in\Gamma_m}\int_f (\mathbf{w}-\mathbf{v}).\mathbf{n}dS\int_f n_i N_m dS = 0 \tag{IV-26}$$

This equation leads to solve the following system:

$$\mathbf{A}.\mathbf{W}_m = \mathbf{B} \quad \text{with}$$

$$\left|\begin{array}{l} \mathbf{A}_{ij} = \sum\limits_{f\in\Gamma_m} \int\limits_f n_i N_m dS \int\limits_f n_j N_m dS \\ \mathbf{B}_i = \sum\limits_{f\in\Gamma_m} \int\limits_f n_i N_m dS \int\limits_f \left(\sum\limits_{n\in f} \mathbf{v}_n N_n - \sum\limits_{\substack{n\in f \\ n\neq m}} \mathbf{w}_n^{(it-1)} N_n \right) .\mathbf{n} ds \end{array}\right. \qquad \text{(IV-27)}$$

This minimization problem is solved by a Jacobi type algorithm. At each iteration the neighboring nodes are regarded as leaving a prescribed grid velocity (from previous iteration). This local equation is over constraining the mesh velocity: no tangential degree of freedom is left. However, the invertible matrix A can be expressed in a new eigen vectors basis in order to classify and better select which degree of freedom have to be taken into account to satisfy the local flux conservation.

$$\mathbf{A}.\mathbf{W} = \mathbf{B} \;\Leftrightarrow\; \mathbf{A}'.\mathbf{W}' = \mathbf{B}' \quad ; \; \forall i = 1,..,3 \;\; w_i' = \frac{b_i'}{\lambda_i}$$

$$\text{with} \left|\begin{array}{l} \mathbf{A}' = \mathbf{P}.\mathbf{A}.\mathbf{P}^{-1} = \begin{bmatrix} \lambda_1 & 0 & 0 \\ 0 & \lambda_2 & 0 \\ 0 & 0 & \lambda_3 \end{bmatrix} ; \lambda_1 \geq \lambda_2 \geq \lambda_3 \\ \mathbf{B}' = \mathbf{P}.\mathbf{B} \\ \mathbf{W}' = \mathbf{P}.\mathbf{W} \\ \text{and} \;\; \mathbf{P} = [\mathbf{u}_1 \quad \mathbf{u}_2 \quad \mathbf{u}_3] \end{array}\right. \qquad \text{(IV-28)}$$

P is the basis of eigen vectors.

It is then possible to enforce only one or two conditions according to the intensity of the eigen values, in other words, to select only one or two free surface normals $\tilde{\mathbf{n}}_{FS}$ to enforce the flux conservation. A criterion can be based on the comparisons of the eigenvalues $\lambda_{i\,(i=1,3)}$:

If $\frac{\lambda_2}{\lambda_1} < \lambda_{crit}$ then the condition (IV-18) is enforced only for $\tilde{\mathbf{n}}_{FS} = \mathbf{u_1}$

Else if $\begin{cases} \frac{\lambda_2}{\lambda_1} \geq \lambda_{crit} \\ \frac{\lambda_3}{\lambda_1} < \lambda_{crit} \end{cases}$ then the condition (IV-18) is verified only for $\tilde{\mathbf{n}}_{FS}^1 = \mathbf{u_1}$ and for

$$\tilde{\mathbf{n}}_{FS}^2 = \frac{\mathbf{u}_2 - (\mathbf{u_1}.\mathbf{u}_2)\mathbf{u_1}}{\left\| \mathbf{u}_2 - (\mathbf{u_1}.\mathbf{u}_2)\mathbf{u_1} \right\|}$$

Else, $\begin{cases} \dfrac{\lambda_2}{\lambda_1} \geq \lambda_{\text{crit}} \\ \dfrac{\lambda_3}{\lambda_1} \geq \lambda_{\text{crit}} \end{cases}$ then the grid velocity is taken equal to the material velocity: $\mathbf{w} = \mathbf{v}$.

Note that $\mathbf{v}$ and $\mathbf{w}^{\text{it-1}}$ are not involved in the computation of the eigen values which is purely geometric. The tangential components of the mesh velocity must nevertheless be carefully computed: large tangential displacements of the boundary nodes should not be generated at the risk of obtaining an important volume diffusion.

Further investigations are still to be done in order to estimate the volume diffusion generated by the computed mesh velocity and thus correct this velocity to minimize the error. At first approach, two different global strategies for mesh regularization, described in the next sections, allow tangential rezoning of the boundary nodes in order that the volume diffusion be reduced by a simple projection procedure.

2.3.3 Projection procedure for free surface nodes

Material tangential movements of the free surface boundary nodes are often negligible whereas tangential rezoning criterion may lead to large tangential component of the mesh velocity. The projection procedure, detailed below, tends to reduce the volume diffusion due to this tangential difference.

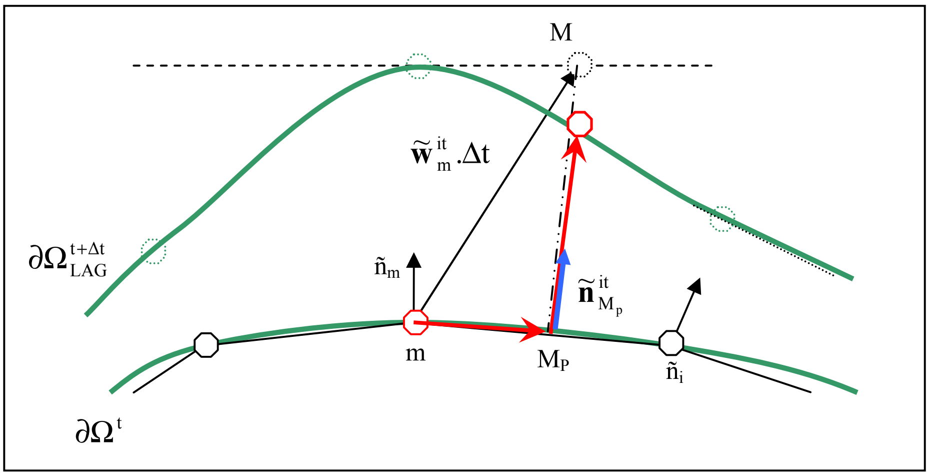


Figure 41: 2D schematic representation of the additional projection procedure for grid velocity computation

At each iteration of the mesh velocity computation, the previous normal criterion and a tangential criterion (detailed in next sections) provide a preliminary grid velocity $\tilde{\mathbf{w}}_{\mathrm{m}}^{\mathrm{it}}$ for the free surface boundary node m. Its coordinates are updated as follows:

$$\begin{aligned} &\mathbf{x}_{\mathrm{M}}^{\mathrm{it}} = \mathbf{x}_{\mathrm{m}}^{0} + \tilde{\mathbf{w}}_{\mathrm{m}}^{\mathrm{it}}.\Delta \mathrm{t} \\ &\text{with } \tilde{\mathbf{w}}_{\mathrm{m}}^{\mathrm{it}}.\tilde{\mathbf{n}}_{\mathrm{m}} = \mathbf{v}_{\mathrm{m}}.\tilde{\mathbf{n}}_{\mathrm{m}} \text{ for any active consistent normal } \tilde{\mathbf{n}}_{\mathrm{m}} \end{aligned} \tag{IV-29}$$

If only one consistent normal is active, the point M is orthogonally projected on the patch Γ_{m} (see Figure 41):

$$\mathbf{x}_{\mathrm{M_p}}^{\mathrm{it}} = \Pi_{\Gamma_{\mathrm{m}}}(\mathbf{x}_{\mathrm{M}}^{\mathrm{it}}) \tag{IV-30}$$

A new material velocity $\mathbf{v}_{\mathrm{M_p}}^{\mathrm{it}}$ and a new consistent normal $\tilde{\mathbf{n}}_{\mathrm{M_p}}^{\mathrm{it}}$ are computed at the node $\mathrm{M_p}$ by interpolating the nodal values of the nodes of the facet f containing the projected point $\mathrm{M_P}$ (under the condition that no more than one consistent normal $\tilde{\mathbf{n}}_n$ has been found for the nodes of f):

$$\begin{cases} \tilde{\mathbf{n}}_{\mathrm{M_p}}^{\mathrm{it}} = \dfrac{\sum\limits_{\mathrm{n}\in \mathrm{f}} N_{\mathrm{n}}(\mathbf{x}_{\mathrm{M_p}}^{\mathrm{it}}).\tilde{\mathbf{n}}_n}{\left\| \sum\limits_{\mathrm{n}\in \mathrm{f}} N_{\mathrm{n}}(\mathbf{x}_{\mathrm{M_p}}^{\mathrm{it}}).\tilde{\mathbf{n}}_n \right\|} \\ \mathbf{v}_{\mathrm{M_p}}^{\mathrm{it}} = \sum\limits_{\mathrm{n}\in \mathrm{f}} N_{\mathrm{n}}(\mathbf{x}_{\mathrm{M_p}}^{\mathrm{it}}).\mathbf{v}_n \end{cases} \tag{IV-31}$$

The resulting mesh velocity for the node m is then computed as follows:

$$\mathbf{w}_{\mathrm{m}}^{\mathrm{it}} = \frac{\mathbf{x}_{\mathrm{M}}^{\mathrm{it}} - \mathbf{x}_{\mathrm{M_p}}}{\Delta t} + \mathbf{w}_{\mathrm{M_p}} \quad \text{with} \quad \mathbf{w}_{\mathrm{M_p}} = (\mathbf{v}_{\mathrm{M_p}}^{\mathrm{it}}.\tilde{\mathbf{n}}_{\mathrm{M_p}}^{\mathrm{it}})\tilde{\mathbf{n}}_{\mathrm{M_p}}^{\mathrm{it}} \tag{IV-32}$$

In other words, the procedure allows preserving the tangential grid velocity as much as possible while the velocity in the normal direction is corrected to take into account the tangential displacement. Figure 42 shows the differences between free surfaces obtain with and without this projection procedure during FSW simulation. Note that free-surface oscillations are widely reduced.

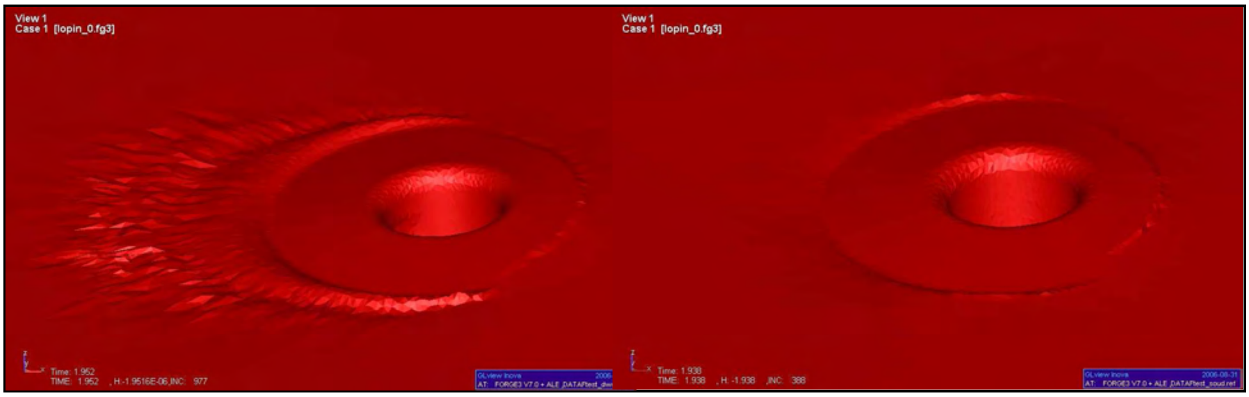


Figure 42: Free surface during an ALE welding simulation:
without the projection procedure (on the left side); with the projection procedure (on the right side)

2.3.4 Contact and symmetry boundary conditions

A good description of the limits between the surface of the workpiece which is in contact with the tool, and the surface of the workpiece which is not, is crucial in FSW simulation: it influences the frictional area and so the material flow. Figure 43 shows the two most critical zones in which the tangential movement of the contact limiting nodes must be properly taken into account.

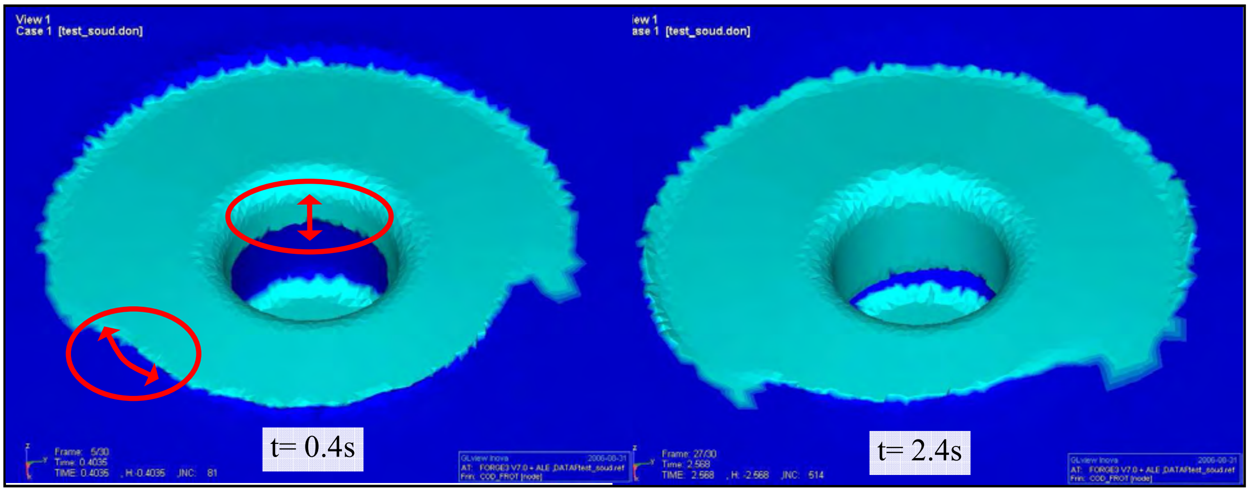


Figure 43: Critical zones of tangential movement of the contact limiting nodes for welding simulation.

If all nodes of the patch Γ_m (see Figure 40) are either in contact with another body or belong to a symmetry plane, the node m is considered as purely in contact or purely symmetric. Thus, no calculation of a consistent normal is necessary, and the condition (IV-18) is enforced only for the contact or symmetry normals:

$$\forall m \in \partial\Omega_{C/S} : \qquad \mathbf{w}_m^{t+\Delta t} \cdot \mathbf{n}_{C/S} = \mathbf{v}_m^{t+\Delta t} \cdot \mathbf{n}_{C/S} \tag{IV-33}$$

The material velocity **v** verifies the contact (III-6) or the symmetry condition, so does the grid velocity **w** through (IV-33).

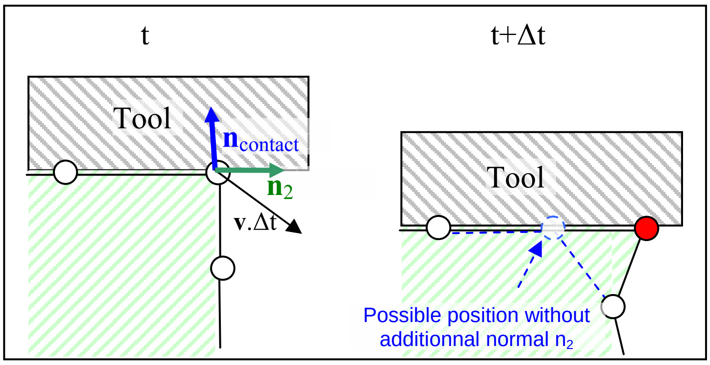


Figure 44: Motivation for adding penalizing free-surface normal for a node already in contact.

If one neighbouring node of the patch Γ_m is free, then the node m is not considered as purely in contact or symmetric. The condition (IV-33) is then enforced but the free-surface normal analysis is performed in order to compute another possible consistent normal $\tilde{\mathbf{n}}_{FS}$. If this last one is different from **n**, a new constraining normal $\mathbf{n}_2$ is taken into account:

$$\text{if} \quad \mathbf{n}_{C/S}.\tilde{\mathbf{n}}_{FS} < \alpha_{\lim} \quad \text{then} \quad \mathbf{n}_2 = \frac{\tilde{\mathbf{n}}_{FS} - (\mathbf{n}_{C/S}.\tilde{\mathbf{n}}_{FS}).\mathbf{n}_{C/S}}{\left\| \tilde{\mathbf{n}}_{FS} - (\mathbf{n}_{C/S}.\tilde{\mathbf{n}}_{FS}).\mathbf{n}_{C/S} \right\|} \tag{IV-34}$$

where $\alpha_{\lim}$ is an imposed limit cosinus angle.

$\mathbf{n}_2$ guarantees that diffusion is prevented from tangential movements of nodes delimiting the contact/symmetry area, as illustrated in Figure 44.

2.3.5 Material fold treatment

In FSP, the material flow may generate folds in some critical zones shown in Figure 45. These zones are neighbouring the critical zones of tangential surface movements shown in Figure 43. This phenomenon has to be numerically tackled in order to avoid mesh overlaps that could not be remeshed and consequently stop the simulation.

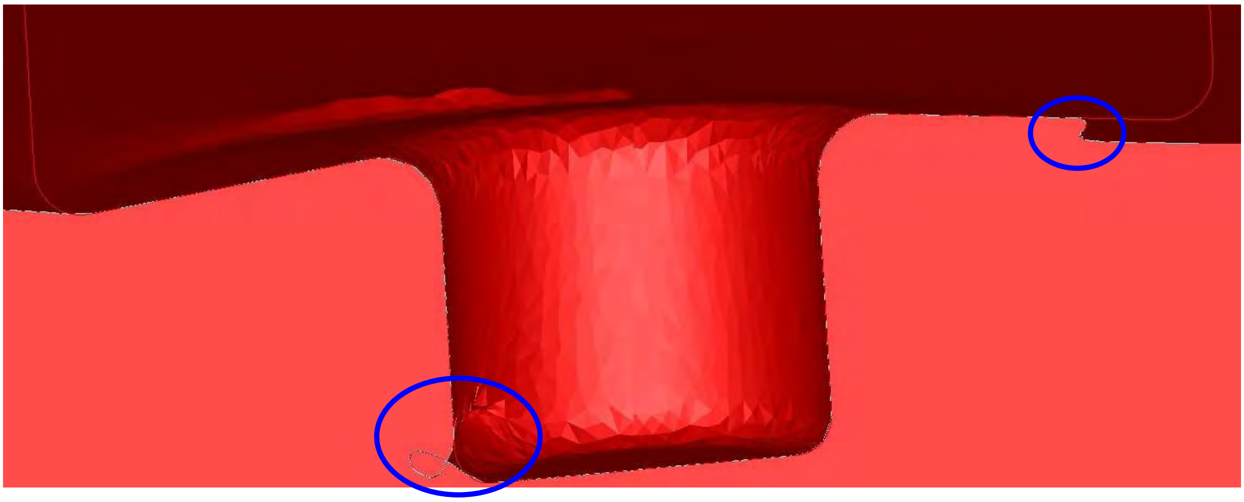


Figure 45: Possible folding zones during welding simulation

A simple procedure, which is schematized in Figure 46, has been introduced in the code. It is based on the fact that the equivalent strain rate is larger in folding zones:

- For each node m of the free-surface, the strain rate is averaged on its patch Γ_m.
- If this averaged value is abnormally high, the angles between each couple of facets that belong to the patch are calculated.
- If one of these angles is less than a prescribed critical angle, then the flux preservation condition (IV-32) is not taken into account. Consequently, the node is regularized like an internal node, without any boundary constraint, for providing good quality elements.

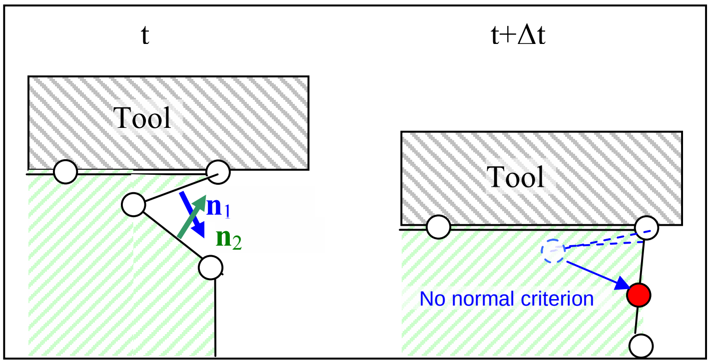


Figure 46: Schematic representation of folds treatment.

A self-contact procedure could be another way to tackle this problem. But it is more complex to implement, requires more computational time, and does not allow taking into account the fact that the material fold may collapse (a collapsing parameter, which should be dependent on temperature and contact pressure could be introduced and coupled to this complex remeshing procedure).

2.4 Non-adaptive formulation

A first and simple way to compute the mesh velocity has been implemented in Forge3®. It is based on an iterative algorithm in which the grid velocity of any node m is taken equal to the average of all its neighbour nodes velocities at the previous iteration (see illustration in Figure 47) within a Jacobi or Gauss Siedel (preferred) type algorithm.

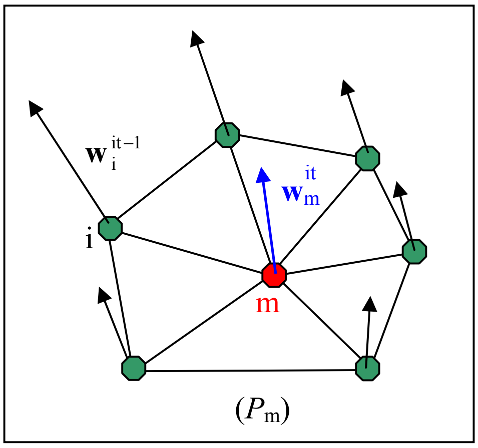


Figure 47: 2D illustration of the mesh velocity averaging

The mesh velocity is thus computed as follows (here for the Jacobi algorithm):

$$\forall \mathrm{m} \in \Omega_h : \quad \tilde{\mathbf{w}}_m^{it} = \frac{1}{|P_m|} \sum_{i \in P_m} \mathbf{w}_i^{it-1} \tag{IV-35}$$

*P*m is called the volumic nodal patch of m: it is the set of elements containing m.

This equation is applied to all nodes of the domain Ω. However, for boundary nodes, the boundary conditions have to be enforced using the consistent normals, so providing the grid velocity in the normal direction through equation (IV-36). Consequently, (IV-35) provides the grid velocity in the tangential/remaining directions.

$$\begin{aligned} &\forall \mathrm{m} \in \partial\Omega_h : \quad \mathbf{w}_m^{it} = \tilde{\mathbf{w}}_m^{it} - \sum_{\alpha=1}^{NBC} \left(\left(\mathbf{v}_m - \tilde{\mathbf{w}}_m^{it} \right) . \tilde{\mathbf{n}}_m^{\alpha} \right) \tilde{\mathbf{n}}_m^{\alpha} \\ &\forall \mathrm{m} \notin \partial\Omega_h : \quad \mathbf{w}_m^{it} = \tilde{\mathbf{w}}_m^{it} \end{aligned} \tag{IV-36}$$

where NBC is the number of boundary conditions.

The iterative algorithm converges when the maximum difference between grid velocities of two consecutive iterations is less than a prescribed percentage $R_{\%}$ of the maximum grid velocity norm, or if the maximum prescribed number of iteration it_{max} has been reached:

$$\text{Algorithm convergence} \Leftrightarrow \left| \begin{array}{l} \dfrac{\left\|\Delta_{\max}\mathbf{w}^{it}\right\|}{\left\|\mathbf{w}_{\mathbf{max}}\right\|} = \dfrac{\max\limits_{m\in\Omega}\left\|\mathbf{w}_{m}^{it}-\mathbf{w}_{m}^{it\text{-}1}\right\|}{\max\limits_{\substack{n\in\Omega\\ k=0,it}}\left\|\mathbf{w}_{n}^{k}\right\|} < R_{\%} \\ \\ \text{or} \quad it = it_{\max} \end{array} \right. \qquad \text{(IV-37)}$$

This algorithm is very simple and fast, without convergence problem. However, it has two main drawbacks:

- The formulation is based on a velocity averaging, not on a position averaging. Therefore, the element quality is dependent on the grid velocity gradient enforced at the boundary nodes, which is transmitted by averaging to volume nodes. The distortions of the elements are so slowed down but not totally under control.
- Such a kinematical formulation does not allow to incrementally adapt the grid density. Once again, the nodal positions are only dependent on the initial configuration and on the material velocity imposed at the boundary.

Therefore, a more sophisticated algorithm based on error-estimation has been developed and implemented.

2.5 Error-estimation and adaptive strategy

Error estimation allows computing an optimal mesh size map. The adaptation of the mesh consists in minimizing the number of elements for a prescribed accuracy θ^{imp} (classical adaptive remeshing strategy, which we call AST1), or, maximizing the accuracy for a prescribed number of elements $Nbelt^{imp}$ (strategy called AST2).

With a prescribed accuracy, the number of elements of the optimized mesh M^{opt} may increase rapidly. The computational cost becomes very high and there is a risk to exceed computer and mesh generator capabilities. So, if the computed number of elements resulting from the optimization AST1 is larger than an imposed maximum number of elements $Nbelt^{imp}$, the strategy AST2 is used. One may refer to [21] for more details.

In the ALE formulation, the mesh topology is preserved, so the maximum number of elements is equal to the current number of elements. Therefore, the utilized adaptive strategy corresponds to the AST2 one:

$$\mathrm{Nbelt}^{\mathrm{imp}} = \mathrm{Nbelt} \qquad \text{(IV-38)}$$

The utilized norm to express the error depends on the studied problem and on the variable that one wishes to control. Here the utilized norm is the energy one. This is the most adequate norm to measure error due to the finite element approximation in terms of material deformation energy and efforts applied to the system. It expresses the difference between the exact solution of the deviatoric stress and strain rate tensors $(s,\dot{\varepsilon})$ and the finite element ones $(s_h,\dot{\varepsilon}_h)$:

$$\theta = \left(\int_{\Omega} (s - s_{\mathrm{h}}) : (\dot{\varepsilon} - \dot{\varepsilon}_{\mathrm{h}}) \, d\omega \right)^{1/2} \qquad \text{(IV-39)}$$

Practically, the exact solutions $(s,\dot{\varepsilon})$ are approximated by recovered ones $(\tilde{s},\tilde{\varepsilon})$, which are continuous and computed from the FE solutions using a Patch Recovery Technique. The recovering technique is not specifically detailed in this document but introduced in the mapping techniques presented thereafter. One may refer to the work of Boussetta and Fourment [36], who has developed this error-estimator.

For each element e of the domain, its contribution to the global error θ is estimated, using the Zienkiewicz-Zhu approach:

$$\theta_e = \left(\int_{\Omega_e} (\tilde{s}_{\mathrm{h}} - s_{\mathrm{h}}) : (\tilde{\dot{\varepsilon}}_{\mathrm{h}} - \dot{\varepsilon}_{\mathrm{h}}) \, d\omega \right)^{1/2} \qquad \text{(IV-40)}$$

2.5.1 Adaptive strategies

The global estimated error θ is expressed from the local contribution θ_e of the element e as follows:

$$\theta = \left(\sum_{e}^{Nbelt} \theta_e^2 \right)^{\frac{1}{2}} \tag{IV-41}$$

The problem is to determine the size modification coefficient r_e for each element e:

$$r_e = \frac{h_e^{opt}}{h_e} \tag{IV-42}$$

h_e is the current size of the element e and h_e^{opt} its optimal size.

If we suppose that the convergence rate of the finite element method is uniform and equal to p on the whole domain, we can write:

$$r_e{}^{p} = \frac{\theta_e^{opt}}{\theta_e} \tag{IV-43}$$

Where θ_e^{opt} is the contribution of the optimal element e (which size is h_e^{opt}) to the global estimated error. For the utilized linear element the theoretical value of p is 1.

According to expression (IV-42), the number of new elements e' that are necessary to refine the element e of the old configuration M^{old} is equal to $r_e{}^{-d}$, where d is the space dimension.

The optimality condition involves that the error θ is uniformly distributed over all the elements e' of the new mesh M^{opt} with a θ^{uni} value. Applying equation (IV-41) to one element e, this condition can be written as:

$$\forall e, \quad \theta_e^{opt} = \left(\sum_{e' \in e}^{Nbelt} \theta_{e'}^2 \right)^{\frac{1}{2}} \Leftrightarrow \theta_e^{opt} = \left(\left(\theta^{uni} \right)^2 . r_e{}^{-d} \right)^{\frac{1}{2}} = \theta^{uni} . r_e{}^{-\frac{d}{2}} \tag{IV-44}$$

Replacing the expression (IV-44) in the equation (IV-43), one obtains:

$$r_e{}^{p} = \frac{\theta^{uni} . r_e{}^{-\frac{d}{2}}}{\theta_e} \Leftrightarrow r_e = \left(\frac{\theta^{uni}}{\theta_e} \right)^{\frac{2}{2p+d}} \tag{IV-45}$$

With the ALE formulation resulting in (IV-38), we have:

$$\sum_{e \in M^{old}} r_e{}^{-d} = Nbelt \tag{IV-46}$$

Remark: In the strategy AST1, which is utilized when the usual complete remeshing procedure is performed, the number of elements of the new mesh is not imposed, but a prescribed accuracy is enforced, so instead we have the equation:

$$\theta^{imp} = \left(\sum_{e}^{Nbelt} {\theta_e^{opt}}^2\right)^{\frac{1}{2}} = \left(\sum_{e}^{Nbelt} \left(\theta^{uni}\right)^2 . r_e^{-d}\right)^{\frac{1}{2}} = \theta^{uni}\left(\sum_{e}^{Nbelt} r_e^{-d}\right)^{\frac{1}{2}} \qquad \text{(IV-47)}$$

Replacing expression (IV-45) of the coefficient r_e into equation (IV-46) (respectively in equation (IV-47) for AST1 strategy), we obtain the new expression (IV-48) (respectively (IV-49)) for the uniform error.

$$\sum_{e\in M^{old}} \left(\frac{\theta^{uni}}{\theta_e}\right)^{\frac{-2d}{2p+d}} = Nbelt \Leftrightarrow \theta^{uni} = Nbelt^{-\frac{2p+d}{2d}} \left(\sum_{e\in M^{old}} \theta_e^{\frac{2d}{2p+d}}\right)^{\frac{2p+d}{2d}} \qquad \text{(IV-48)}$$

$$\theta^{imp} = {\theta^{uni}}^{\frac{2p}{2p+d}} \left(\sum_{e\in M^{old}} \left(\theta_e\right)^{\frac{2d}{2p+d}}\right)^{\frac{1}{2}} \Leftrightarrow \theta^{uni} = {\theta^{imp}}^{\frac{2p+d}{2p}} \left(\sum_{e\in M^{old}} \left(\theta_e\right)^{\frac{2d}{2p+d}}\right)^{-\frac{2p+d}{4p}} \qquad \text{(IV-49)}$$

The combination of (IV-43) and (IV-48) (respectively (IV-49) for AST1 strategy) provides the final expression (IV-50) (respectively (IV-51)) of the coefficient r_e.

$$r_e = \frac{h_e^{opt}}{h_e} = \frac{\left(Nbelt\right)^{-\frac{1}{d}} \left(\sum_{e\in M^{old}} \left(\left(\theta_e\right)^{\frac{2d}{2p+d}}\right)\right)^{\frac{1}{d}}}{\left(\theta_e\right)^{\frac{2}{2p+d}}} \qquad \text{(IV-50)}$$

$$r_e = \frac{h_e^{opt}}{h_e} = \frac{\left(\theta^{imp}\right)^{\frac{1}{p}}}{\left(\theta_e\right)^{\frac{2}{2p+d}} \left(\sum_{e\in M^{old}} \left(\theta_e\right)^{\frac{2d}{2p+d}}\right)^{\frac{1}{2p}}} \qquad \text{(IV-51)}$$

Thus, the error estimation allows computing an optimal mesh size for each element of the domain.

Note that equation (IV-51) is also available for the AST2 strategy where θ^{imp} is replaced by the estimated error of the optimal mesh θ^{opt}. So the combination of equations (IV-50) and (IV-51) gives:

$$\theta^{\text{opt}} = (\text{Nbelt})^{-\frac{p}{d}} \left(\sum_{e \in M^{\text{old}}} (\theta_e)^{\frac{2d}{2p+d}} \right)^{\frac{2p+d}{2d}} \qquad \text{(IV-52)}$$

2.5.2 Mesh improvement for remeshing

The usual remesher used in Forge3® is able to take into account an optimal size map resulting from the error estimation. The mesh generation algorithm is clearly different from other well established meshing engines using the Delaunay, frontal or octree methods. The algorithm exploits the possibility to operate on the mesh topologies without considering any geometric constraint. It consists in two main steps:

- Discretization of the surface of the domain, and generation of an initial topology. In order to do so, a staring operator connects one node to all the others, except those which belong to the same face as this latter. The resulting mesh is obviously not valid: elements may cross each others as well as the domain boundaries. The resulting topology is nevertheless sufficient to start generating a valid mesh.
- Simple local algorithms are used to derive one topology from another in order to progressively improve them and obtain an actual mesh. The nodes creation and deletion are implicitly included in the method, and modifications of surface and volume of the mesh are strongly coupled. Among the obtained candidates, the topology which minimizes the total volume is selected. Because several topologies may have the same volume, it is necessary to favour that which elements have better geometrical qualities. Therefore, this procedure consists in selecting the topology which tetrahedra maximize the following criterion C(e, h_e, h_e^{opt}):

$$C_f(e, h_e) = C_0 \frac{|\Omega_e|}{h_e^3} \qquad \text{(IV-53)}$$

Where Ω_e is the volume of the current element e, h_e is the average length of its edges, C_0 is a constant which makes the form factor equal to 1 for an equilateral tetrahedron. In order to take into account the mesh size map, this criterion is extended as follows:

$$C\left(e, h_e, h_e^{\text{opt}}\right) = C_f(e, h_e) \frac{\min\left(|\Omega_e|, \left|\omega(h_e^{\text{opt}})\right|\right)}{\max\left(|\Omega_e|, \left|\omega(h_e^{\text{opt}})\right|\right)} \qquad \text{(IV-54)}$$

$\left|\omega\left(h_e^{\text{opt}}\right)\right|$ is the theoretical optimal volume of the tetrahedron e for a prescribed size h_e^{opt}.

In the ALE formulation, the mesh management is completely different, and the connectivity is unchanged. Computing only the grid velocity is much faster than computing and selecting better topologies. But a common difficulty remains in combining a geometrical and an optimal size criteria.

2.6 Adaptive formulation

2.6.1 General formulation

In order to control the quality and size of the elements, the algorithm is based on the computation of new mesh coordinates. The mesh velocity is then indirectly computed.

In a classical position centring method, the equation (IV-35) of the non-adaptive iterative algorithm is replaced by the following ones:

$$\begin{cases} \mathbf{x}_{\mathrm{m}}^{\mathrm{it}} = \dfrac{1}{\left|\mathcal{P}_{\mathrm{m}}\right|} \displaystyle\sum_{\mathrm{e}\in \mathcal{P}_{\mathrm{m}}} \mathbf{x}_{\mathrm{g_e}}^{\mathrm{it-1}} \\ \tilde{\mathbf{w}}_{\mathrm{m}}^{\mathrm{it}} = \dfrac{(\mathbf{x}_{\mathrm{m}}^{0} - \mathbf{x}_{\mathrm{m}}^{\mathrm{it}})}{\Delta \mathrm{t}} \end{cases} \tag{IV-55}$$

$\mathbf{x}_{\mathrm{g_e}}^{\mathrm{it-1}}$ is the center of each element e which contains the node m.

Weight factors $\mathrm{C}_{\mathrm{e}}^{\mathrm{it-1}}$ are introduced in order to obtain an adaptive formulation. The main difficulty is to define them:

$$\mathbf{x}_{\mathrm{m}}^{\mathrm{it}} = \frac{1}{\left|\Gamma_{\mathrm{m}}\right|} \sum_{\mathrm{e}\in \Gamma_{\mathrm{m}}} \mathbf{x}_{\mathrm{g_e}}^{\mathrm{it-1}} \mathrm{C}_{\mathrm{e}}^{\mathrm{it-1}} \tag{IV-56}$$

2.6.2 Weight factors calculation

A first idea is to directly use the coefficient of size modification $\mathrm{r_e}$ (IV-50) :

$$\mathrm{C}_{\mathrm{e}}^{\mathrm{it-1}} = \mathrm{r}_{\mathrm{e}}^{\mathrm{it-1}} = \frac{\mathrm{h}_{\mathrm{e}}^{\mathrm{opt}}}{\mathrm{h}_{\mathrm{e}}^{\mathrm{it-1}}} \tag{IV-57}$$

The node is thus more attracted by the elements which optimal sizes are less than the current ones; the sizes of these elements should logically decrease after some iterations.

But such an algorithm has revealed some instabilities that result on distortions of elements. In fact, if the gradient of the element density is strong in a given direction, this algorithm provides a too strong anisotropy of the mesh.

To tackle with this problem, a geometrical quality factor has been added in the algorithm. Although the criterion of the usual remeshing procedure (IV-54) can not be directly used here (no maximization process), the idea is the same: to combine a geometrical form factor C_f and an adaptive factor C_a to look for the best quality adapted mesh.

$$\begin{aligned} C_{f\,e}^{it-1} &= C_0 \frac{\left[\Omega_e^{it-1}\right]^+}{\left(h_e^{it-1}\right)^3} \\ C_{a\,e}^{it-1} &= \left(\frac{h_e^{it-1}}{h_e^{opt}}\right)^3 \end{aligned} \qquad \text{(IV-58)}$$

C_0 is the adimensional constant previously presented.

For a given element size h_e, Figure 48 shows the evolution curves of C_f and C_a.

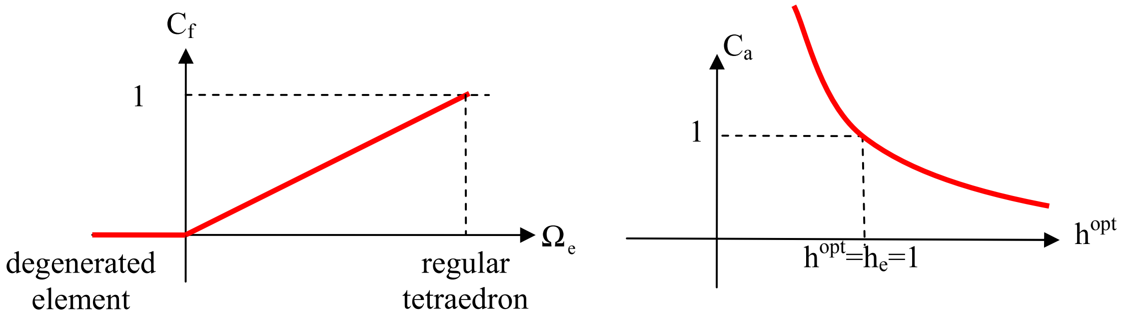


Figure 48: curves of the geometrical form coefficient and the adaptive coefficient for a given element size $h_e=1$

C_f is maximum and equal to 1 when e is a regular tetrahedron, and C_a is higher than 1 when $h_e>h^{opt}$. The weight factor C_e of equation (IV-56) is written as a combination of these two coefficients. As adaptivity is expected to govern the centring algorithm and as the quality criterion is only introduced to limit the element distortion, the following formula is used:

$$C_e^{it-1} = \xi(C_{f\,min}^{\,it-1}) \times C_{f\,e}^{it-1} + (1-\xi(C_{f\,min}^{\,it-1})) \times C_{a\,e}^{it-1} \qquad \text{(IV-59)}$$

ξ is a function of the minimal form coefficient $C_{f\,min}^{\,it-1}$ calculated among the contiguous elements of the considered node at iteration it-1:

$$C_{f\,min}^{\,it-1} = \min_{e\in P_m}\left(C_{f\,e}^{\,it-1}\right) \qquad \text{(IV-60)}$$

This ξ function is shown in Figure 49 and its expression follows:

$$\xi(C_{f\,min}^{it-1}) = \begin{vmatrix} 1-\exp\left(-\alpha\left(C_{f\,min}^{it-1} - C_{f\,crit}\right)^2\right) \text{ if } C_{f\,min}^{it-1} \leq C_{f\,crit} \\ 0 \text{ else} \end{vmatrix} \quad \text{(IV-61)}$$

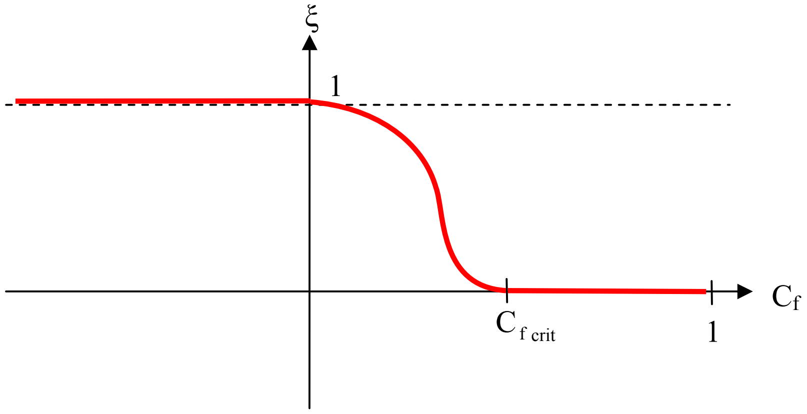


Figure 49: ξ function curve for combination of the geometrical form coefficient and the adaptive coefficient.

$C_{f\,crit}$ is a user value and α a parameter which controls the slope of the curve. If all the surrounding elements have a form factor, larger than $C_{f\,crit}$, the coefficient C_e^{it-1} is an adaptive one : $C_e^{it-1} = C_a^{it-1}$. If it is not the case, the form factor contributes to the weight. If there is a degenerated element in the nodal patch, the weighting coefficient C_e^{it-1} is the form one: $C_e^{it-1} = C_f^{it-1}$.

The parameter α has to be adjusted in order to minimize a flip-flop effect which may occur between these two situations. Its value depends on the coefficient $C_{f\,crit}$. Usual values for the couple $(C_{f\,crit};\alpha)$ range between (0.1; 100), and (0.5; 10). Flip-flop effect is often observed with this kind of approach, therefore the iteration algorithm for **w** does not converge in many cases and is consequently limited to a given number of iterations.

2.6.3 Rearranging procedure

However, although a geometrical form factor is used, the utilized centring algorithm does not fully guarenty the mesh quality. Some degeneration problems may occur for complex configurations, in particular close to the surface where the mesh velocity must satisfy restricting boundary conditions. In order to tackle possible mesh degeneration, an additional procedure has been implemented.

It consists in two steps: construction of an initial valid (non-degenerated) configuration, and research of a better backward position in case of unsuccess of the centring algorithm.

- In the first step (see illustration in Figure 50), we look for an initial nodal position which satisfies for any element of the current patch P_m:

  $$\forall e \in P_m : \; C^{i\ \ it}_{f\,e} > 0 \qquad \text{(IV-62)}$$

  If this condition is not satisfied the central node is moved. The fastest way to fulfil it consists in testing different positions of the central node, at the mid distance of its adjacent nodes. The position which maximizes $C^{i\ \ it}_{f\,m\,min} = \min_{e\in P_m} C^{i\ \ it}_{f\,e}$ is kept and the algorithm proceeds to a next iteration i until the condition (IV-62) is satisfied. It so provides an initial configuration without degenerated elements.

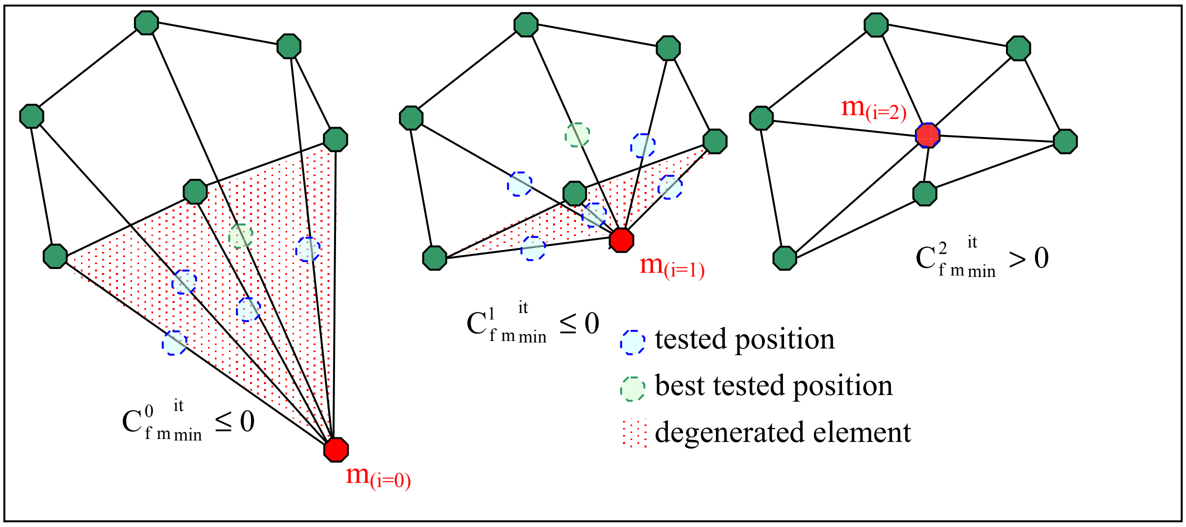


Figure 50: Schematic representation of the first step of the iterative initializing procedure

Then the centring method (IV-56) is applied using the weighting coefficients (IV-59), and provides a new value of the minimal form factor of the patch $C^{centred\ it}_{f\,m\ \ min} = \min_{e\in P_m} C^{centred\ it}_{f\ \ e}$.

- In a second step, if the condition (IV-63) is not satisfied after the centring algorithm, a new nodal position, located between the centred and the initial position is searched by dichotomy:

  $$C^{centred\ it}_{f\,m\ \ min} < \max\left(\max_{i}\left(C^{i\ \ it}_{f\,m\,min}\right),0\right) \qquad \text{(IV-63)}$$

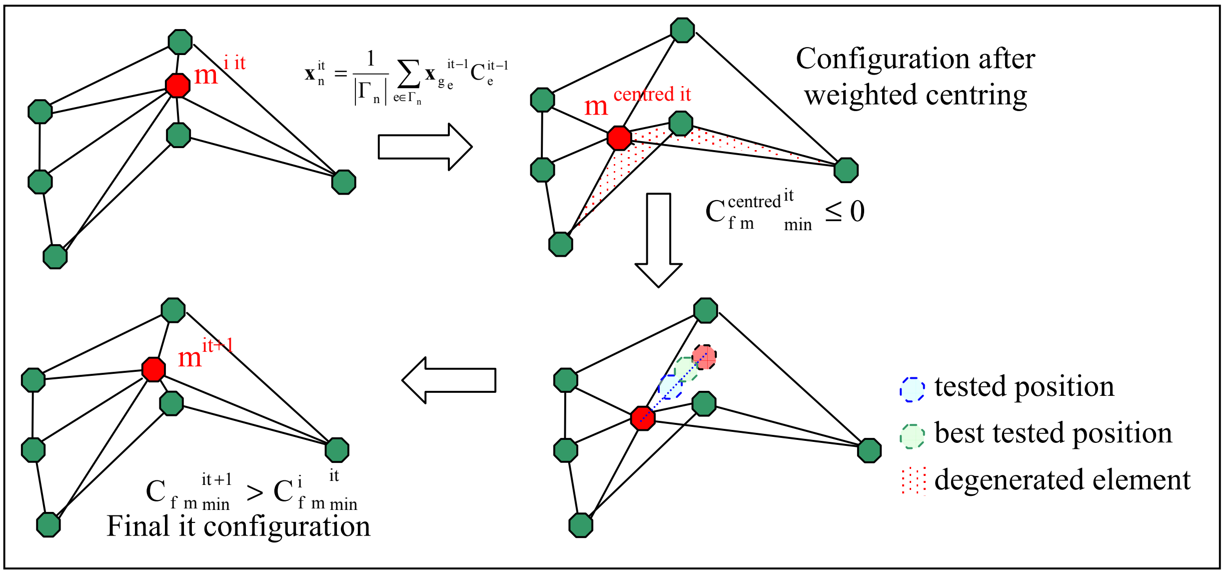


Figure 51: Schematic representation of the second step of the straightening procedure

This procedure requires using a Gauss-Siedel type algorithm (the actualisation of a nodal position is done immediately after its computation) instead of a Jacobi one (the actualisation is done after all nodal positions have been computed, at the end of the iteration; therefore the coefficients C_a and C_f are computed only once per iteration). The major drawback of this procedure is its computational cost. Therefore, the Gauss-Siedel algorithm coupled with this rearranging procedure could be used only for nodes n of the elements e which does not satisfy $C_{f\,e}^{centred} > 0$ at the end of the Jacobi algorithm.

2.6.4 Surface tangential movement limitation

As for the prior non adaptive algorithm, the grid velocity of boundary nodes is constrained through equation (IV-36). However, adaptivity may generate high tangential movements resulting in surface instabilities. Two strategies have been developed to avoid such movements.

The first one consists in controlling the adaptivity rate by limiting the value of the size modification coefficient r_e, and thus the value of the adaptive coefficient C_a. For each element, a maximum and a minimum size modification coefficients are prescribed in order that:

$$r_{\min}^{imp} \leq \frac{h_e^{opt}}{h_e} \leq r_{\max}^{imp} \qquad \text{(IV-64)}$$

Taking into account this limitation, an iterative algorithm has been implemented to modify the size modification coefficient:

1. Equation (IV-50) provides the iterative value of r_e as a function of $\mathrm{Nbelt}_{imp}^{it}$:

$$r_e^{it} = \frac{\left(\mathrm{Nbelt}_{imp}^{it}\right)^{-\frac{1}{d}} \left(\sum_{e \in M^{old}} \left((\theta_e)^{\frac{2d}{2p+d}} \right) \right)^{\frac{1}{d}}}{(\theta_e)^{\frac{2}{2p+d}}} \quad \text{with} \quad \mathrm{Nbelt}_{imp}^{0} = \mathrm{Nbelt} \tag{IV-65}$$

2. The limiting conditions are then enforced by projection:

$$\begin{array}{lll} \text{if} & r_e^0 < r_{min}^{imp} & \text{then } r_e^0 = r_{min}^{imp} \\ \text{else if} & r_e^0 > r_{max}^{imp} & \text{then } r_e^0 = r_{max}^{imp} \end{array} \tag{IV-66}$$

3. Equation (IV-46) provides a maximum and a minimum theoretical number of elements:

$$\begin{array}{l} \sum_{e \in M^{old}} \left(\left(r_{max}^{imp} \right)^{-d} \right) = \mathrm{Nbelt}_{min} \\ \sum_{e \in M^{old}} \left(\left(r_{min}^{imp} \right)^{-d} \right) = \mathrm{Nbelt}_{max} \end{array} \tag{IV-67}$$

4. The theoretical number of elements resulting from the first iteration is also computed as:

$$\sum_{e \in M^{old}} \left(\left(r_e^0 \right)^{-d} \right) = \mathrm{Nbelt}_{th}^{0} \tag{IV-68}$$

5. The optimal prescribed number of elements is then obtained by dichotomy. The new size modification coefficients are computed using equation (IV-65) with a newly imposed number of elements:

$$\begin{array}{l} \text{if} \quad \mathrm{Nbelt}^{it} < 0.95 \times \mathrm{Nbelt}, \text{ then } \mathrm{Nbelt}_{imp}^{it+1} = \dfrac{\mathrm{Nbelt}_{imp}^{it} + \min(\mathrm{Nbelt}_{imp}^{it-1}, \mathrm{Nbelt}_{max})}{2} \\ \text{else if} \quad \mathrm{Nbelt}^{it} > 1.05 \times \mathrm{Nbelt}, \text{ then } \mathrm{Nbelt}_{imp}^{it+1} = \dfrac{\mathrm{Nbelt}_{imp}^{it} + \max(\mathrm{Nbelt}_{imp}^{it-1}, \mathrm{Nbelt}_{min})}{2} \end{array} \tag{IV-69}$$

The resulting theoretical number of elements is re-evaluated using equation (IV-46) until it ranges between 95% and 105% of the current number of elements.

One the other hand, this limitation of the element size modification does not allow controlling the resulting mesh displacement: little modifications of element sizes may finally generate huge displacements (see illustration in Figure 52).

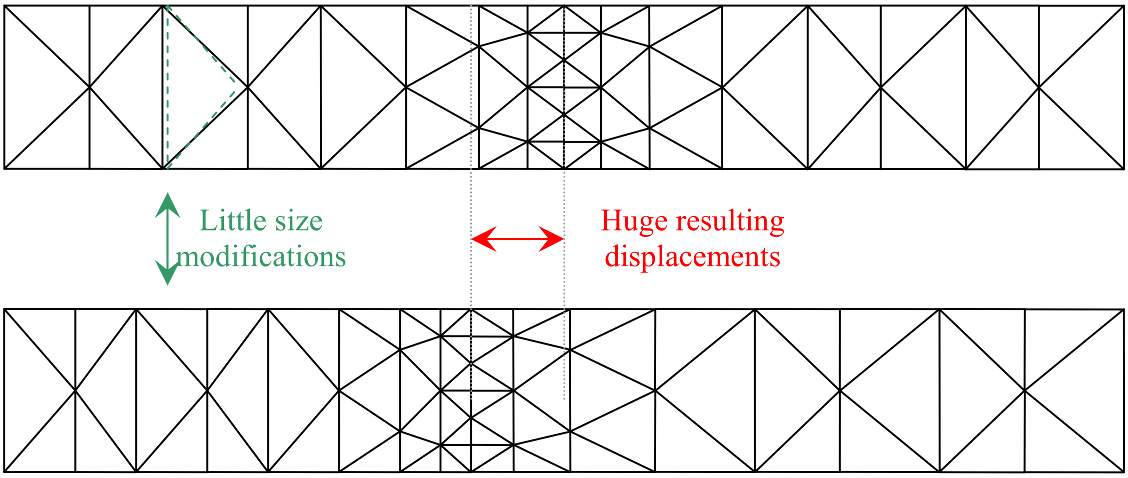


Figure 52: illustration of huge displacements generated by little size modifications.

Another enhancement consists in directly limiting the grid velocity of the surface nodes by a maximal value in the algorithm.

The maximum allowed grid velocity is computed for each of these nodes, as follows:

$$\forall m \in \partial\Omega_h : \left\| \mathbf{w}_m \right\|_{\max} = \frac{\min\limits_{\substack{n \in \Gamma_m \\ n \neq m}} \left\| \mathbf{x}_m^0 - \mathbf{x}_n^0 \right\|}{\Delta t} \qquad \text{(IV-70)}$$

This maximum velocity corresponds to the maximum displacement that the node can have without leaving its patch (see Figure 53). Thus, it guarantees that the normal projection of the node belongs to its patch, which is an important issue for the calculation of boundary condition of free surface nodes.

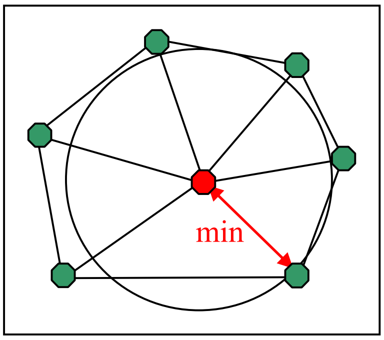


Figure 53: Maximal velocity allowed on surface patch Γ_m

However, the boundary conditions may force some nodes to have a grid velocity larger than the maximal one computed in (IV-70). This problem, illustrated in Figure 54, is tackled by introducing a correction vector $\tilde{\boldsymbol{\alpha}}^{it}_{m}$ .It is calculated at the beginning of each iteration as follows:

$$\forall m \in \partial\Omega_h \text{ and for } i=1,2,3 : \qquad \tilde{\alpha}^{it}_{i\,m} = \left|\begin{array}{ll} 1 & \text{for it} = 0 \\ \max\limits_{n\in\Gamma_m}\left(\dfrac{w^{\,it\text{-}1}_{i\,n}}{\left\|\mathbf{w}_n\right\|_{max}};1\right) & \text{for it} > 0 \end{array}\right. \tag{IV-71}$$

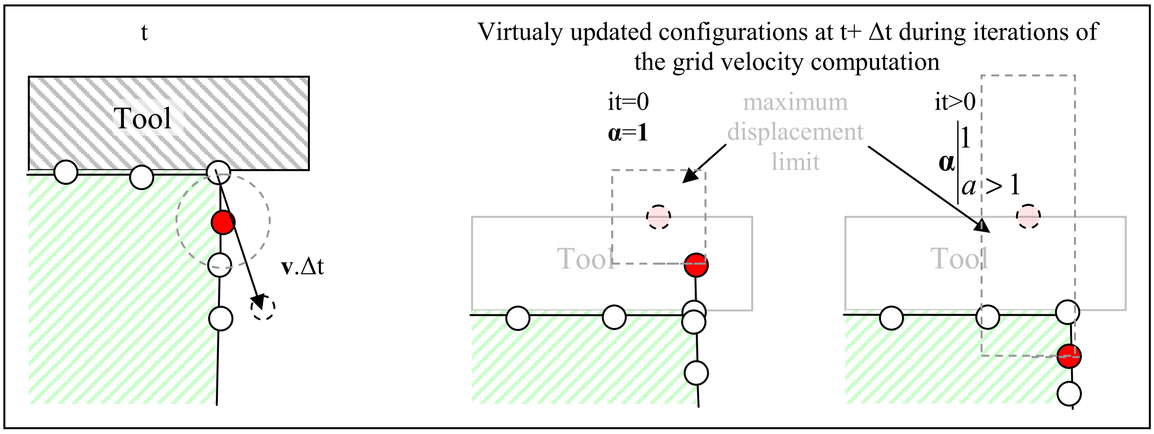


Figure 54: schematic representation of the limited grid displacement in a problematic case which shows the interest of the coefficient α

Thus, before applying the boundary conditions (IV-36) to the surface grid velocity, the following condition is enforced:

$$\forall m \in \partial\Omega_h \text{ and for } i=1,2,3 : \qquad \overline{w}^{it}_{i\,m} = \min\left(\tilde{w}^{it}_{i\,m};\tilde{\alpha}^{it}_{i\,m}\left\|\mathbf{w}_m\right\|_{max}\right) \tag{IV-72}$$

And $\overline{w}^{it}_{i\,m}$ is substituted to $\tilde{w}^{it}_{i\,m}$ in (IV-36).

2.6.5 Combination of mesh regularization and “topological remeshing”

This ALE formulation has been developed in order to regularize and adapt unstructured meshes of tetrahedral elements, however the classical remeshing procedure sometimes needs to be performed in order to modify the mesh topology after large deformations. It is coupled to the ALE formulation through two main criteria: a quality and an adaptative criterion.

After the grid velocity computation, the nodal positions are updated as follows:

$$\mathbf{x}^{t+\Delta t}_{m} = \mathbf{x}^{t}_{m} + \mathbf{w}^{t+\Delta t}_{m}\Delta t \qquad \text{(i.e. Chapter II : 4.2)} \tag{IV-73}$$

The quality Q_e and the size modification coefficient r_e of each updated element are computed and compared to the minimal admitted values Q_{min} and r_{min}, which are prescribed by the software user. If $Q_e < Q_{min}$ or $r_e < r_{min}$ for one of the elements, a topological remeshing is then carried out. In order to proceed with the adaptive remeshing, a new computation of the optimal element sizes is performed using the previously described adaptive strategies AST1 and AST2 (see paragraph 2.4.1).

Remark: If a topological remeshing is necessary, then a new Lagrangian updating step (equation (IV-74)) is carried out in order to avoid adding diffusions due to the ALE and topological remeshing remaps.

$$\mathbf{x}_m^{t+\Delta t} = \mathbf{x}_m^{t} + \mathbf{v}_m^{t+\Delta t}\Delta t \qquad \text{(IV-74)}$$

2.6.6 Schematized algorithm for grid velocity computation

The algorithm of grid velocity computation is schematized in Figure 55.

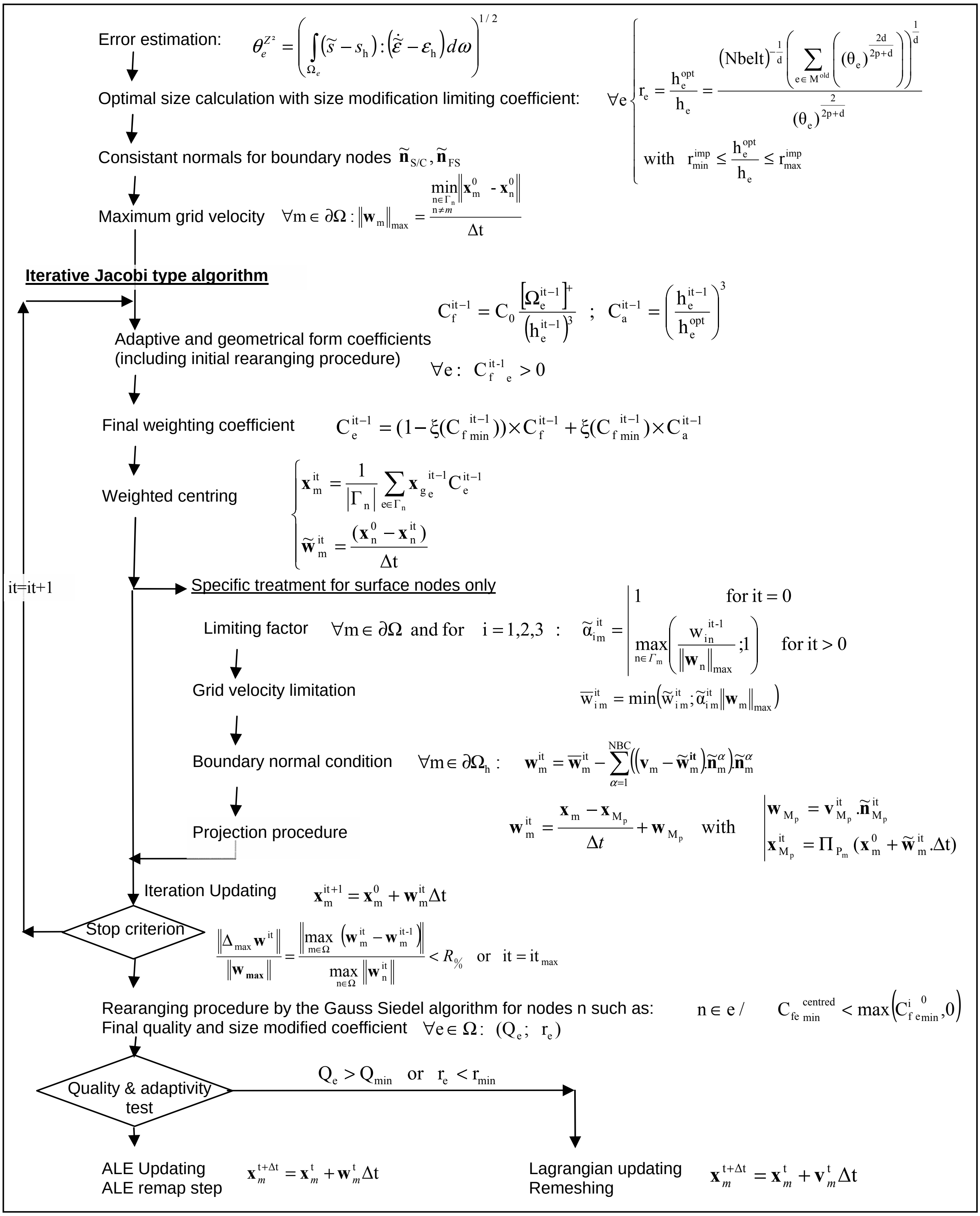


Figure 55: Schematic representation of the complete developed algorithm to compute the grid velocity

3 Remapping step

3.1 Background

As presented in the introduction, the second step of the ALE splitting method consists in remapping the variables on the new computed mesh. This remap needs to be accurately done in order to minimize the generated diffusion, which cumulated at each time increment, may lead to convergence problems or large numerical error.

Therefore, sophisticated techniques have to be developed. Two approaches are possible:

- the convective approach, which solves the grid time derivative equation,
- and the interpolation approach, which transfers the values obtained after the updated Lagrangian iteration.

Two variable types have to be remapped:

- nodal variables, such as temperature, pressure or velocity, which are P1 fields (linear and continuous per element), computed and stored at nodes,
- element piecewise variables, such as equivalent strain or stress, which are P0 (constant and discontinuous per element), computed and stored at integration point i.e. at the element center with the utilized scheme.

The techniques that are developed to remap P0 values can be significantly different from those for P1 values. P1 methods are hardly applicable to P0 variables because of their discontinuous characteristic.

P0 remapping is not easy. Discontinuity makes ineffective many methods developed for nodal variables. In the case of elasto-plastic, visco-elastic or elasto-visoplastic constitutive law, all P0 components of the stress tensor have to be remapped. The resulting diffusion must be limited in particular to avoid that the equilibrium equation be strongly unsatisfied at the beginning of the next time increment.

Different methods have been tested and implemented. Accuracy, consistency and robustness of each of them have been compared on specific tests. The computational time required by the method is also taken into account.

3.2 Nodal variables remapping

3.2.1 Classical P1 interpolation

The first method (illustrated in Figure 56) is used to remap nodal variables in the classical remeshing procedure of Forge3®. It requires building the ALE $M_{ALE}^{t+\Delta t}$ and the Lagrangian $M_{LAG}^{t+\Delta t}$ updated meshes. It consists in two steps for each node m of the new ALE grid:

- Identification of the element of $M_{LAG}^{t+\Delta t}$ containing the node m.
- Interpolation of the nodal values of the vertices at the position of m.

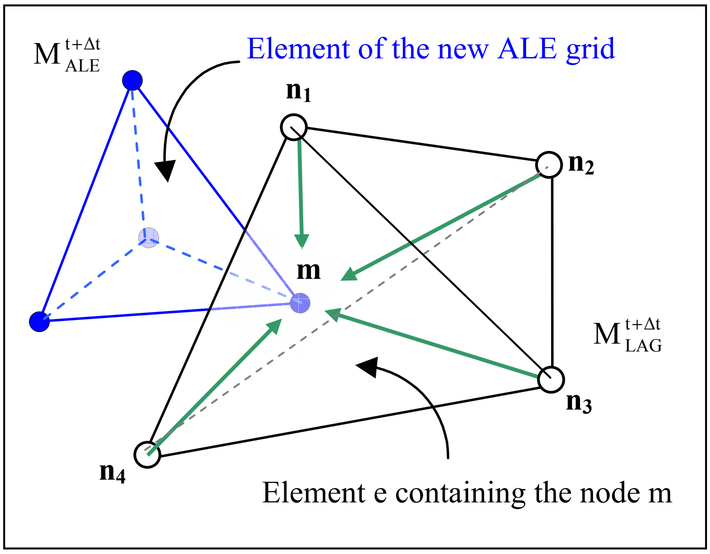


Figure 56: Classical P1 interpolation technique

The interpolation of a nodal value *f* is as follows:

$$\forall\, m \in M_{ALE}^{t+\Delta t}, f_m^{t+\Delta t} = \sum_{n \in e(m)} f_n^{t+\Delta t} N_n(m) \qquad \text{(IV-75)}$$

e(m) is the element of the Lagrangian updated mesh containing the node m. $(f_n)_{n=1,4}$ is the interpolation value of *f* at the Lagrangian updated node n. For a linear element, and using notations of Figure 56, the values of the weights or shape functions are calculated as:

$$\begin{aligned}
&N_2(m)=\frac{\det(\overrightarrow{n_1m},\overrightarrow{n_1n_3},\overrightarrow{n_1n_4})}{\Delta},\quad N_3(m)=\frac{\det(\overrightarrow{n_1n_2},\overrightarrow{n_1m},\overrightarrow{n_1n_4})}{\Delta}\\
&N_4(m)=\frac{\det(\overrightarrow{n_1n_2},\overrightarrow{n_1n_3},\overrightarrow{n_1m})}{\Delta},\quad N_1(m)=1-N_2(m)-N_3(m)-N_4(m)\\
&\text{with}\quad \Delta=\det(\overrightarrow{n_1n_2},\overrightarrow{n_1n_3},\overrightarrow{n_1n_4})
\end{aligned}\tag{IV-76}$$

This method is fast when the search algorithm for e(m) is well implemented, for instance using a tree structure. However, it requires building the updated Lagrangian configuration. So, this method can not handle excessive distortions of the elements.

3.2.2 Upwind technique for nodal variables

A convective approach can be preferred to avoid constructing the updated Lagrangian grid. It consists in a first order linearization of the grid time derivative. Using a finite difference scheme, the grid time derivative can be written as follows:

$$\underbrace{\frac{f^{t+\Delta t}_{ALE}(x^{t+\Delta t})-f^{t}(x^{t})}{\Delta t}}_{\frac{d_g f}{dt}}=\underbrace{\frac{f^{t+\Delta t}_{LAG}(x^{t+\Delta t}_{mat})-f^{t}(x^{t})}{\Delta t}}_{\frac{df}{dt}}-\nabla f^{t+\Delta t}.\left(\mathbf{v}^{t+\Delta t}_{m}-\mathbf{w}^{t+\Delta t}_{m}\right)\tag{IV-77}$$

So the first order linearization leads to the following equation [37]:

$$\forall m:\quad f^{t+\Delta t}_{ALE}=f^{t+\Delta t}_{LAG}+\nabla f^{t+\Delta t}.\left(\mathbf{w}^{t+\Delta t}-\mathbf{v}^{t+\Delta t}\right)\Delta t\tag{IV-78}$$

The first term $f^{t+\Delta t}_{LAG}$ corresponds to the value calculated by the Lagrangian iteration. The gradient value of f is P0 per element and then $\nabla f^{t+\Delta t}$ is not defined at node m. Several methods can be utilized to compute this nodal value but Stocker and Chenot [38] have shown that the best and most conservative scheme is obtained by taking the gradient value in the upwind element u, which is given by:

$$\nabla f=\sum_{n\in u}\nabla\left(N_{n}(m)\right)f_{n}\tag{IV-79}$$

The upwind element u is the first element located in the opposite direction of the convective velocity $\mathbf{c}^{t+\Delta t}_{m}=\mathbf{v}^{t+\Delta t}_{m}-\mathbf{w}^{t+\Delta t}_{m}$ (see Figure 57) of node m.

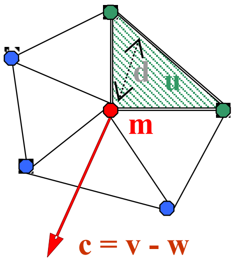


Figure 57: Upwind element

Because of the first order linearization, this method provides accurate results if that the following CFL condition is validated:

$$\Delta t \leq \frac{h_u}{\left|\mathbf{v}_m^{t+\Delta t} - \mathbf{w}_m^{t+\Delta t}\right|} \tag{IV-80}$$

In other words that the particle located in m at t+Δt is coming from the upwind element

In order to take this condition into account, an iterative algorithm has been implemented in the remapping the method:

- the minimal CFL time step is computed:

$$\Delta t_{\min}^{CFL} = \min_{m \in M} \frac{d}{\left|\mathbf{v}_m^{t+\Delta t} - \mathbf{w}_m^{t+\Delta t}\right|} \tag{IV-81}$$

  d is the distance from the node m to the boundary of its upwind element in the direction of the convective velocity c.

- the global time step is divided into several time steps satisfying the CFL condition: the nodal values are then iteratively updated for each time step using equation (IV-78) until the global time step is reached.

$$\Delta t \leftarrow \sum_i \Delta t_i \quad \text{where} \quad \forall i : \Delta t_i \leq \Delta t_{CFL} \tag{IV-82}$$

3.2.3 Nodal PR2 technique

In order to avoid the problem of satisfying CFL conditions, and to increase the accuracy of the transfer, another method has been implemented. As this method is adapted from P0 variable remapping techniques, it will be detailed in the next section.

3.3 Remapping of variables stored at integration points (P0 remapping)

The two prior methods cannot be used with P0 variables (mainly the stress field σ and the equivalent strain $\bar{\varepsilon}$) which are discontinuous per elements and are not defined at the nodes.

3.3.1 P0 remap

The first and simplest way to remap such variables consists in a direct P0 transfer. The value of the P0 variable is approximated by the value of the nearest updated Lagrangian integration point (see Figure 58). It is as good as and even better than an actual P0 interpolation. On the other hand, it is not limited by a possible distortion of the updated elements. Finally, the remapped stress tensor satisfies the balance equations.

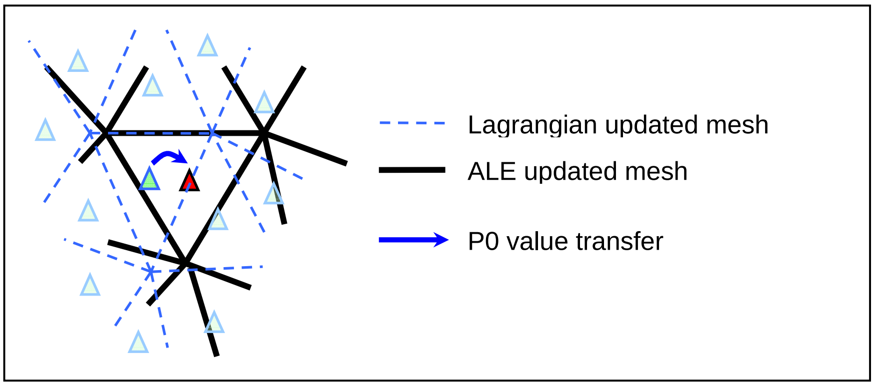


Figure 58: P0 remapping

This technique is very fast and rather efficient with refined mesh, however the resulting error is as large as the gradient of the remapped values for a given element size.

The particle method is another direct approximation, which allows improving this approach. It is based on the fact that the gradient provided by a layer of elements can not be preserved in a grid containing fewer elements. Therefore, particles are introduced to store and transport the P0 variables, in a larger number than the number of Gauss points (see Figure 59).

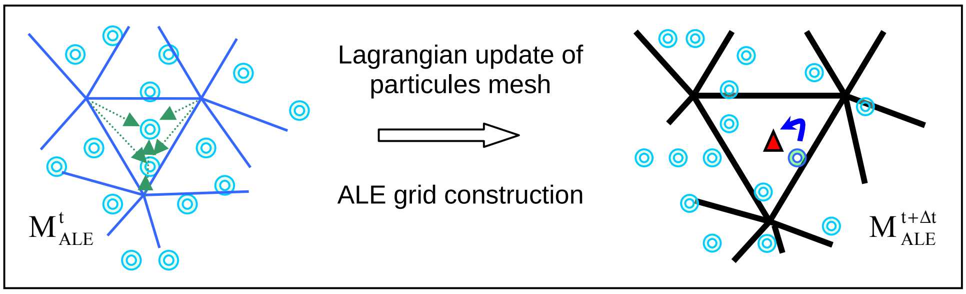


Figure 59: P0 particle remapping

The incremental variation of the variables to be transported is computed at time t, such as Δx^t or $\Delta\sigma^t$ or $\Delta\varepsilon^t$. Their values are extrapolated to all the particles contained in the elements before updating their positions in a Lagrangian way. The value of a newly considered ALE integration point is then approximated by the value of the nearest updated particle. Traoré [39] has shown that this method is much less diffusive than the previous one for the simulation of certain processes, provided that there is always one or more particles in an element.

However, it is more complicated to implement; controlling the number of particles and their space distribution in time requires implementing specific procedures, and a difficulty is to remove particles in order to prevent the method from exceeding the storage capacities.

3.3.2 From P0 to linear remapping

The techniques presented in this paragraph have not been implemented but illustrates well the process which has led us to new ALE remapping procedures.

In order to decrease the diffusion generated by a simple P0 transport, Srikanth et al. [40] have proposed a method mixing direct approximation and weighted average. As for a simple P0 remap, the updated Lagrangian mesh is built. If the nearest Lagrangian integration point is close enough to the considered ALE integration point, a direct approximation of the variable is performed. Else, a weighted average of the values stored at the integration points located at a selected maximal defined distance is performed (see following equation and Figure 60).

- if $\min\limits_{j\in M^{t+\Delta t}_{LAG}}(r_j) < R_{min}$: $\quad f^{t+\Delta t}_{iALE} = f^{t+\Delta t}_{jLAG}$
- else:

$$f^{t+\Delta t}_{iALE} = \left(\sum_{j\in M^{t+\Delta t}_{LAG}} \alpha_j . f^{t+\Delta t}_{jLAG} \frac{1}{\left|r_{ij}\right|}\right)\left(\sum_{j\in M^{t+\Delta t}_{LAG}} \alpha_j . \frac{1}{\left|r_{ij}\right|}\right)^{-1} \quad \text{with} \quad \alpha_j \left| \begin{array}{l} =0 \;\text{ if }\; r_j > R_{max} \\ =1 \;\text{ if }\; r_j \le R_{max} \\ >1 \;\text{ if }\; r_j = \min\limits_{j\in M^{t+\Delta t}_{LAG}}(r_j) \end{array} \right. \qquad \text{(IV-83)}$$

Where r_j is the distance from integration point i to integration point j.

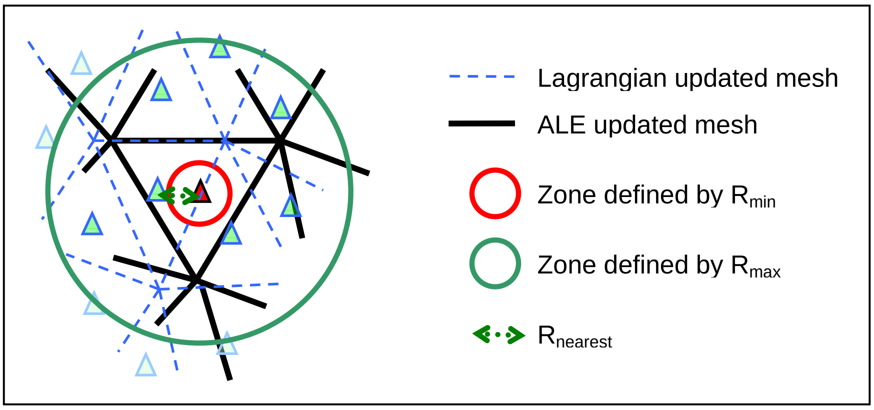


Figure 60: P0 averaging technique

The idea of this method is not far from the idea of meshless interpolation methods, such as Moving Least Square (MLS) or Radial Basis Function (RBF), which may provide a better accuracy.

In fact MLS provides a local approximation u^h of a function u at point x from the known values u_i in the vicinity of the point of co-ordinates $\mathbf{x}$:

$$u^h(\mathbf{x},t) = \sum_{j=1}^{m} p_j(\mathbf{x}).a_j(\mathbf{x},t) \equiv \mathbf{p}^T(\mathbf{x}).\mathbf{a}(\mathbf{x},t) \qquad \text{(IV-84)}$$

The basis functions $\mathbf{p}^T(\mathbf{x})$ are often polynomial. In 3D, we can choose the following linear basis: $\mathbf{p}^T(\mathbf{x}) = [1 \quad x \quad y \quad z]$.

$\mathbf{a}(\mathbf{x},t)$ are adjusted non-constant cœfficients. These coefficients are obtained by minimizing the weighted norms of the difference between the estimated values at their known location and their exact values, as follows:

$$J_x(\mathbf{a}) = \frac{1}{2}\sum_{i=1}^{n} w(\mathbf{x}-\mathbf{x}_i)\left[\mathbf{p}^T(x_i).\mathbf{a}(\mathbf{x},t) - u_i(t)\right]^2 \qquad \text{(IV-85)}$$

$w(\mathbf{x}-\mathbf{x}_\mathrm{i})$ is a weight function of the compact support (often called influence field). n is the number of points (known as "neighbours") whose influence field includes the point **x** i.e. for which $w_\mathrm{i}(\mathbf{x}-\mathbf{x}_\mathrm{i})>0$. The following definition of *w* is illustrated in Figure 61.

$$w(\mathrm{r})\begin{cases}\geq 0 & \text{for} \quad \mathrm{r}<1 \\ =0 & \text{for} \quad \mathrm{r}\geq 1\end{cases} \quad \text{with} \quad \mathrm{r}=\frac{|\mathbf{x}-\mathbf{x}_\mathrm{i}|}{\mathrm{r}_\mathrm{i}} \tag{IV-86}$$

Usually, $w(\mathrm{r})$ decreases monotonously with the normalised distance r. Details of the shape function computation can be found in [41, 42] .

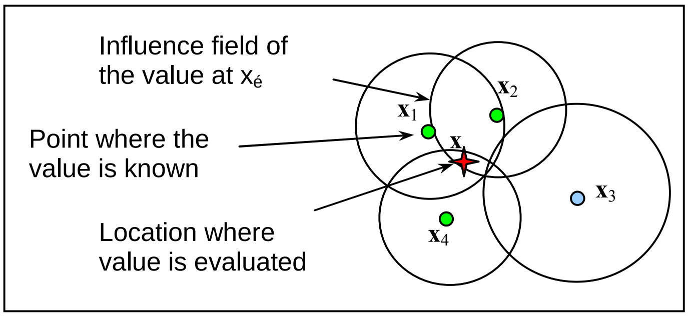


Figure 61: P0 averaging technique

The minimisation of the equation (IV-85) leads to the following system of linear equations:

$$\mathbf{A}(\mathbf{x})\,\mathbf{a}(\mathbf{x},\mathrm{t})=\mathbf{B}(\mathbf{x})\,\mathbf{u}(\mathrm{t})$$

$$\text{with} \quad \begin{cases}\mathbf{A}(\mathbf{x})=\sum_{\mathrm{i}=1}^{\mathrm{n}} w_\mathrm{i}(\mathbf{x}-\mathbf{x}_\mathrm{i})\,\mathbf{p}(\mathrm{x}_\mathrm{i})\,.\,\mathbf{p}^\mathrm{T}(\mathrm{x}_\mathrm{i}) \\ \mathbf{B}(\mathbf{x})=\left[w(\mathbf{x}-\mathbf{x}_1)\,\mathbf{p}(\mathrm{x}_1) \quad w(\mathbf{x}-\mathbf{x}_2)\,\mathbf{p}(\mathrm{x}_2) \quad \ldots \quad w(\mathbf{x}-\mathbf{x}_\mathrm{n})\,\mathbf{p}(\mathrm{x}_\mathrm{n})\right] \\ \mathbf{u}^\mathrm{T}(\mathrm{t})=\left[\mathrm{u}_1(\mathrm{t}) \quad \mathrm{u}_2(\mathrm{t}) \quad \cdots \quad \mathrm{u}_\mathrm{n}(\mathrm{t})\right]\end{cases} \tag{IV-87}$$

If **A** is invertible (a necessary but not sufficient condition is $\mathrm{n}\geq\mathrm{m}$), the coefficients are expressed as follows:

$$\mathbf{a}(\mathbf{x},\mathrm{t})=\mathbf{A}^{-1}(\mathbf{x})\,\mathbf{B}(\mathbf{x})\,\mathbf{u}(\mathrm{t}) \tag{IV-88}$$

So, by introducing Equation (IV-88) in Equation (IV-84), one obtains the following expression:

$$\mathbf{u}^\mathrm{h}(\mathbf{x},\mathrm{t})=\mathbf{p}^\mathrm{T}(\mathbf{x})\;[\;\mathbf{A}^{-1}(\mathbf{x})\,\mathbf{B}(\mathbf{x})\;]\;\mathbf{u}(\mathrm{t})\equiv\boldsymbol{\Phi}(\mathbf{x})\,\mathbf{u}(\mathrm{t}) \tag{IV-89}$$

$\boldsymbol{\Phi}(\mathbf{x})=\mathbf{p}^\mathrm{T}(\mathbf{x})\;[\;\mathbf{A}^{-1}(\mathbf{x})\,\mathbf{B}(\mathbf{x})\;]$ is called the shape function of the MLS method (note that $\sum_n \boldsymbol{\Phi}_n=1$)
The continuity of this function depends on the continuity of the basis functions $\mathbf{p}(\mathbf{x})$, as of the smoothing of the matrix **A** and **B**.

Considering our remapping problem, the main advantages of such a technique are:

- absence of connectivity between nodes and elements: the possible distortion of the Lagrangian updated grid is not an issue,
- the accuracy is formally better than with a P0 direct approximation.

The major shortcoming of these techniques is the difficulty to identify, for each integration point of the new ALE grid, the updated Lagrangian integration points that are included in the influence field. This research may be expensive in term of computational time and may increase the difficulties for the parallelisation of the algorithm.

Therefore, we have developed techniques which provide a similar accuracy of a polynomial expansion and in which the interpolated value depends only on the distance to the nearest Lagrangian updated point.

3.3.3 Patch Recovery (PR) /LO based Techniques

The developed techniques are based on the LO technique proposed by Liszka and Orkisz [43, 44] which is very similar to the MLS method and to the Superconvergent Patch Recovery (SPR) [45, 46] that can be regarded as a particular case of LO [36]. It consists in three steps:

- Construction of a “recovered” local continuous solution in the vicinity of each Lagrangian updated integration point,
- Research of the nearest Lagrangian updated integration point of each ALE updated integration point,
- Extrapolation of the solution to the ALE integration point.

In the following description, the stress components σ^{ij} will be used as an example for the variable which needs to be remapped.

3.3.3.1 Recovery equations

We are looking for a recovered solution $\tilde{\sigma}^{g,ij}$ that can be defined in the vicinity of the Lagrangian updated integration point g of coordinates $\mathbf{x}_g=(x_g,y_g,z_g)$.

Selecting a polynomial interpolation, $\tilde{\sigma}^{g,ij}$ can be written as:

$$\tilde{\sigma}^{g,ij}(\mathbf{x}) = \mathbf{P}^{g}(\mathbf{x}).\mathbf{a}^{g,ij} \qquad \text{(IV-90)}$$

Where $\mathbf{a}^{g,ij}$ is the vector of coefficients which corresponds to the gradient of the variable (for linear elements, these coefficients are constant); $\mathbf{P}^{g}$ is the polynomial basis and ij indicates the stress component. First and second order expansions have been implemented.

For the first order expansion, the method has been called PR1 (for Patch Recovery with 1st order expansion). The polynomial basis and the vector of constant coefficients are:

$$\left|\begin{array}{l}\mathbf{P}^g = (1, \Delta_g x, \Delta_g y, \Delta_g z) \\ \mathbf{a}^{g,ij} = (a_1^{g,ij}, a_2^{g,ij}, a_3^{g,ij}, a_4^{g,ij})^t \\ \text{with } \ \Delta_g x = x_g - x \quad , \ \Delta_g y = y_g - y \ \ , \ \Delta_g z = z_g - z \end{array}\right. \tag{IV-91}$$

For the second order expansion, the method has been called PR2 (for Patch Recovery of the 2nd order). The polynomial basis and the vector of constant coefficients are given by:

$$\left|\begin{array}{l}\mathbf{P}^g = (1, \Delta_g x, \Delta_g y, \Delta_g z, \Delta_g x^2, \Delta_g y^2, \Delta_g z^2, \Delta_g x.\Delta_g y, \ \Delta_g y.\Delta_g z, \Delta_g x.\Delta_g z) \\ \mathbf{a}^{g,ij} = (a_1^{g,ij}, a_2^{g,ij}, a_3^{g,ij}, a_4^{g,ij}, a_5^{g,ij}, a_6^{g,ij}, a_7^{g,ij}, a_8^{g,ij}, a_9^{g,ij}, a_{10}^{g,ij})^t \end{array}\right. \tag{IV-92}$$

The first coefficient of the vector $\mathbf{a}_g$ has to satisfy the consistency of the method. So we have:

$$\tilde{\sigma}^{g,ij}(\mathbf{x}_g) = \sigma_g^{h,ij} \Leftrightarrow a_1^{g,ij} = \sigma_g^{ij^h} \tag{IV-93}$$

So, there are respectively left 3 and 9 unknowns per integration point and per variable component for first and second order expansions.

The vicinity of the Lagrangian updated integration point $\mathbf{x}_g$ is a patch P_g that is centred at the integration point $\mathbf{x}_g$ as illustrated in 2D in Figure 62: P_g contains the elements that contain $\mathbf{x}_g$ and all neighbouring elements (that have a node in common with this element). The recovered expression $\tilde{\sigma}^{g,ij}$ is valid only in the vicinity of the integration point $\mathbf{x}_g$, e.i. on the patch centred on $\mathbf{x}_g$. $\tilde{\sigma}^{g,ij}(\mathbf{x}_f)$ (first order expansion is used for more simplicity) is written for each integration point $\mathbf{x}_f$ included in the patch P_g as:

$$\begin{aligned} \forall \mathbf{x}_f \supset P_g : \ \tilde{\sigma}^{g,ij}(\mathbf{x}_f) &= \mathbf{P}^g(\mathbf{x}_f).\mathbf{a}^{g,ij} \\ &= \sigma_g^{ij^h} + a_2^{g,ij}(x_g - x_f) + a_3^{g,ij}(y_g - y_f) + a_4^{g,ij}(z_g - z_f) = \sigma_f^{ij^h} \end{aligned} \tag{IV-94}$$

And we would like to have:

$$\tilde{\sigma}^{g,ij}(\mathbf{x}_f) = \sigma_f^{ij^h} \tag{IV-95}$$

It so provides one equation for each integration point included in the patch. In order to identify the coefficients, it is necessary (but not sufficient) that the number of integration points included in the patch be larger than or equal to the number of unknown coefficients a_i (3 for linear interpolations and to 9 for quadratic ones).

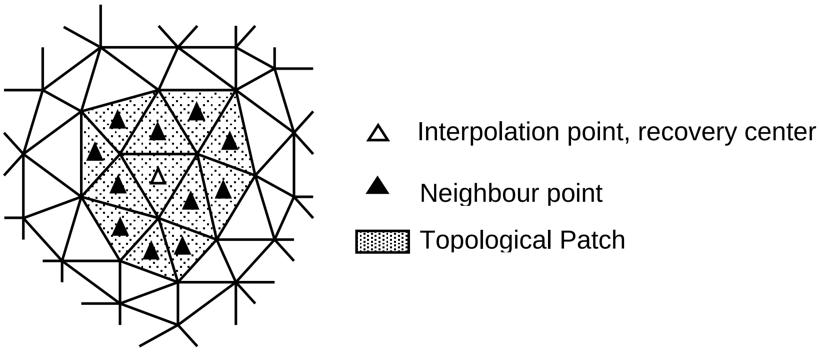


Figure 62: Patch centred on an element in 2D

When a first order interpolation is used, three elements per patch (the central element not being included) are enough for the method. Therefore, a Minimal Patch Recovery (MPR) method can be used for which the construction of patches requires less CPU time. The minimal patch includes only elements which share a facet with the central element. It is illustrated in 2D in Figure 63. For a volume element, the patch consists of 5 elements, and less for surface elements.

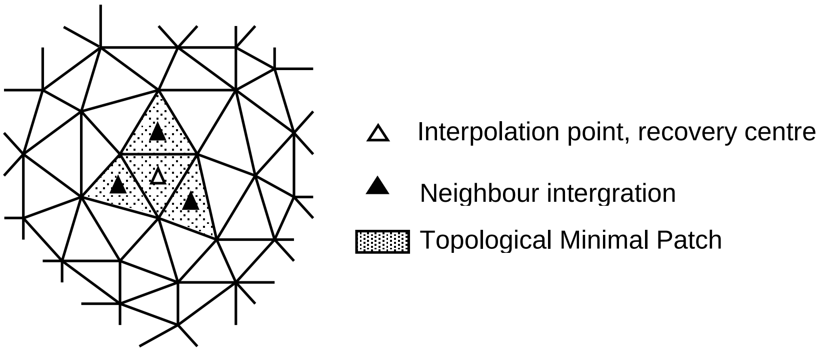


Figure 63: Patch centred on an element in 2D

3.3.3.2 Well conditioned system

In standard situations, the system is overconstrained, so the following least square expression is minimized for each component:

$$\Pi_g^{FD}(\mathbf{a}^{g,ij}) = \frac{1}{2}\sum_{f\in P_g}\left(\frac{O_{ij}\left(\Delta r_{gf}^2\right)}{\Delta r_{gf}^2}\right)^2 \qquad \text{(IV-96)}$$

$$\text{with} \quad \Delta r_{gf} = \sqrt{\Delta_{gf}x^2 + \Delta_{gf}y^2 + \Delta_{gf}z^2}$$

$((\Delta r_{lg}^2)^{-1}$ is a weighting term. The contribution of an integration point is all the larger than its distance to the patch center is smaller. From equation (IV-94), we can deduce:

$$O_{ij}(\Delta\,r_{gf}{}^{2}) = \sigma_{f}^{ij\,h} - \left(\sigma_{g}^{ij\,h} + a_{2}^{g,ij}(x_{g} - x_{f}) + a_{3}^{g,ij}(y_{g} - y_{f}) + a_{4}^{g,ij}(z_{g} - z_{f})\right) \qquad \text{(IV-97)}$$

So the coefficients a^{g} are obtained by minimizing the least square expression:

$$\Pi_{g}^{FD}(\mathbf{a}^{g,ij}) = \sum_{f\in P_{g}} \left(\frac{\sigma_{f}^{ij\,h} - \left(\sigma_{g}^{ij\,h} + a_{2}^{g,ij}(x_{g} - x_{f}) + a_{3}^{g,ij}(y_{g} - y_{f}) + a_{4}^{g,ij}(z_{g} - z_{f})\right)}{\Delta\,r_{gf}^{2}} \right)^{2} \qquad \text{(IV-98)}$$

By cancelling its derivatives:

$$\forall n = 2,..,4 \quad \frac{\partial \Pi_{g}^{ij\,FD}}{\partial a_{n}^{g,ij}} = 0 \qquad \text{(IV-99)}$$

It leads to the following system of 3x3 (9x9 with a 2nd order expansion) linear equations:

$$\mathbf{A}_{g}^{ij}.\mathbf{X}_{g}^{ij} = \mathbf{B}_{g}^{ij}$$

$$\text{with} \quad \left| \begin{array}{l} \mathbf{A}_{g}^{ij} = \displaystyle\sum_{j\in P_{g}} \frac{2}{\Delta\,r_{gf}^{4}} \begin{bmatrix} \Delta_{gf}x^{2} & \Delta_{gf}x.\Delta_{gf}y & \Delta_{gf}x.\Delta_{gf}z \\ \Delta_{gf}x.\Delta_{gf}y & \Delta_{gf}y^{2} & \Delta_{gf}y.\Delta_{gf}z \\ \Delta_{gf}x.\Delta_{gf}z & \Delta_{gf}y.\Delta_{gf}z & \Delta_{gf}z^{2} \end{bmatrix} \\ \mathbf{B}_{g}^{ij} = \displaystyle\sum_{j\in P_{i}} \frac{2}{\Delta\,r_{ij}^{4}} \begin{bmatrix} \Delta_{gf}x.(\sigma_{g}^{ij\,h} - \sigma_{f}^{ij\,h}) \\ \Delta_{gf}y.(\sigma_{g}^{ij\,h} - \sigma_{f}^{ij\,h}) \\ \Delta_{gf}z.(\sigma_{g}^{ij\,h} - \sigma_{f}^{ij\,h}) \end{bmatrix} \\ \mathbf{X}_{g}^{ij} = \begin{bmatrix} a_{2}^{g,ij} \\ a_{3}^{g,ij} \\ a_{4}^{g,ij} \end{bmatrix} \end{array} \right. \qquad \text{(IV-100)}$$

$$\Delta_{gf}x = x_{g} - x_{f} \quad , \quad \Delta_{gf}y = y_{g} - y_{f} \quad , \quad \Delta_{gf}z = z_{g} - z_{f} \;)$$

$$\text{and} \quad \Delta r_{gf} = \sqrt{\Delta_{gf}x^{2} + \Delta_{gf}y^{2} + \Delta_{gf}z^{2}}$$

In order for this system to be defined (A is invertible), it is necessary that the number of points $\mathbf{x}_{f}$ included in the patch be sufficient, in other words that the number of independent equations be larger than the number of unknown coefficients.

But it is not enough: when two neighbours $\mathbf{x}_{f}$ are aligned with the patch center, the equations that they generate are not independent. Their contributions relates to the same gradient direction. This can be easily illustrated in 2D with a 1st order expansion (see Figure 64): the number of unknowns is 2 but if the patch includes only two points which are aligned with the patch center, so there are not enough information to determine the gradient in the y direction.

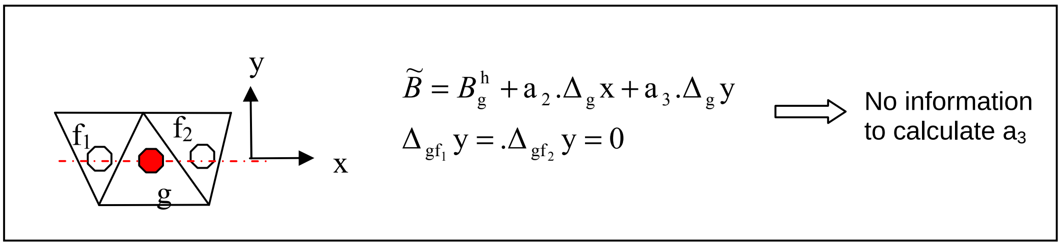


Figure 64: 2D illustration of non sufficient number of neighbours

3.3.3.3 Insufficient information

In case of a badly conditioned system (which is often the case for patches centred on boundary elements), two strategies are possible: either increase the size of the patch to get more information, or decrease the expansion order to decrease the number of unknowns.

In the MPR technique, even with a first order expansion, a patch centred on a boundary element never contains enough points. Thus, some virtual integration points are created, to take into account the boundary conditions as additional information:

- for symmetry boundary conditions, a virtual symmetric point is added (see Figure 65a).
- for free surface or imposed load facets, the normal stress condition (respectively σ.n=0 and $\boldsymbol{\sigma}.\mathbf{n} = \boldsymbol{\sigma}^{imp}.\mathbf{n}$) is written at the center of the element facet (see Figure 65b). This contribution is added to the least square expression (IV-98) and is written as:

$$\Pi_g^{imp}(\mathbf{a}^{g,ij}) = \sum_{f=1}^{Nfl}\left(\frac{\left(\sigma_f^{ij\,imp} - \left(\sigma_g^{ij\,h} + a_2^{g,ij}(x_g - x_f) + a_3^{g,ij}(y_g - y_f) + a_4^{g,ij}(z_g - z_f)\right)\right).n_j}{\Delta r_{gf}^2}\right)^2 \qquad \text{(IV-101)}$$

where Nfl is the number of free or loaded facets, $\mathbf{x}_f$ are their center and **n** their normals.

- for contact boundary conditions, the tangential and normal stresses conditions are described at the center of the element facet as follows:

$$\begin{cases} \left(\boldsymbol{\sigma}.\mathbf{n} - \boldsymbol{\sigma}_{\mathbf{n}}.\mathbf{n}\right).\mathbf{t}_n = \boldsymbol{\tau}^{cont}.\mathbf{t}_n, \ n = 1,2 \\ \boldsymbol{\sigma}_{\mathbf{n}}.\mathbf{n} = \boldsymbol{\sigma}^{cont}.\mathbf{n} \end{cases} \qquad \text{(IV-102)}$$

$\mathbf{t}_1$ and $\mathbf{t}_2$ are the two tangential directions of the facet. $\boldsymbol{\tau}^{cont}$ and $\boldsymbol{\sigma}^{cont}$ are respectively the friction shear and the normal contact stress. Both are computed at the center of the facets. Their contributions are added to the least square expression (IV-98) trough the following term:

$$\Pi^{\mathrm{cont}}=\sum_{\mathrm{f}=1}^{\mathrm{Nfc}}\left(\left(\boldsymbol{\sigma}_{\mathrm{f}}^{\mathrm{cont}}-\left(\tilde{\boldsymbol{\sigma}}^{g}(\mathbf{x}_{\mathrm{f}}).\mathbf{n}_{\mathrm{f}}\right)\right).\mathbf{n}_{\mathrm{f}}\right)^{2}+\sum_{\mathrm{f}=1}^{\mathrm{Nfc}}\left(\boldsymbol{\tau}^{\mathrm{cont}}.\mathbf{t}_{1}^{\mathrm{f}}-\left(\left(\tilde{\boldsymbol{\sigma}}^{g}(\mathbf{x}_{\mathrm{f}})-\left(\tilde{\boldsymbol{\sigma}}^{g}(\mathbf{x}_{\mathrm{f}}).\mathbf{n}_{\mathrm{f}}\right)\right).\mathbf{n}_{\mathrm{f}}\right).\mathbf{t}_{1}^{\mathrm{f}}\right)^{2}$$

$$+\sum_{\mathrm{f}=1}^{\mathrm{Nfc}}\left(\boldsymbol{\tau}^{\mathrm{cont}}.\mathbf{t}_{2}^{\mathrm{f}}-\left(\left(\tilde{\boldsymbol{\sigma}}^{g}(\mathbf{x}_{\mathrm{f}})-\left(\tilde{\boldsymbol{\sigma}}^{g}(\mathbf{x}_{\mathrm{f}}).\mathbf{n}_{\mathrm{f}}\right)\right).\mathbf{n}_{\mathrm{f}}\right).\mathbf{t}_{2}^{\mathrm{f}}\right)^{2} \qquad \text{(IV-103)}$$

with $\tilde{\boldsymbol{\sigma}}^{g}(\mathbf{x}_{\mathrm{f}})=\boldsymbol{\sigma}_{\mathrm{g}}^{\mathrm{h}}+\mathbf{a}_{2}^{\mathrm{g}}(\mathbf{x}_{\mathrm{g}}-\mathbf{x}_{\mathrm{f}})+\mathbf{a}_{3}^{\mathrm{g}}(\mathbf{y}_{\mathrm{g}}-\mathbf{y}_{\mathrm{f}})+\mathbf{a}_{4}^{\mathrm{g}}(\mathbf{z}_{\mathrm{g}}-\mathbf{z}_{\mathrm{f}})$

where Nfc is the number of facets in contact.

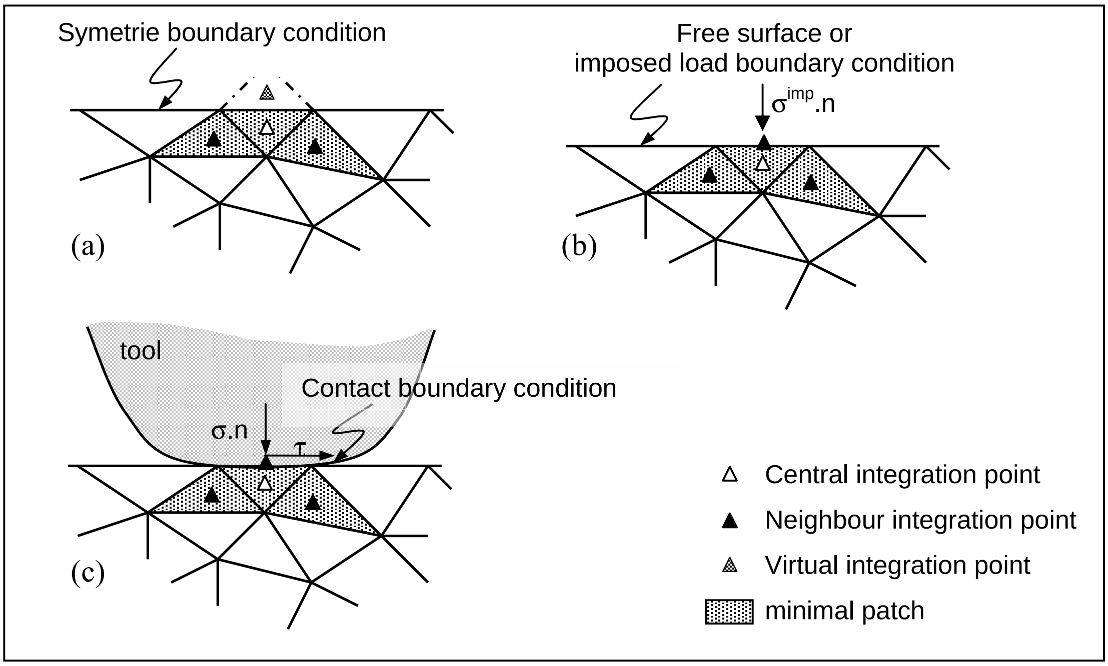


Figure 65: 2D illustration of the addition of points in minimal patches of boundary elements

If the normal stress is the only known data at the surface (no tangential information), the system remains insolvable. In this case, the full patch (illustrated on Figure 62) is necessarily used.

In the case of second order expansion necessarily on the full patch *P*$_{\mathrm{g}}$, the addition of virtual integration points is more complicated for boundary elements and do not always guarantee the good conditioning of the system. Further more, the enlargement of the second order patch by adding third order neighbours is not considered: the patch would be very huge and consequently provides too important diffusion. Therefore, in all cases of badly conditioned system, the strategy consists in decreasing the expansion order, and thus, the number of unknown. For a second order expansion, 9 unknowns have to be determined: $\mathbf{a}_2$ to $\mathbf{a}_{10}$. If the number of integration points is insufficient, a first order recovery is carried out (on the same patch), and thus only 3 unknowns have to be determined: $\mathbf{a}_2$ to $\mathbf{a}_4$ and $\mathbf{a}_5 = \ldots = \mathbf{a}_{10} = 0$. If the number of integration point is still insufficient, then a constant recovery is considered: $\mathbf{a}_1 = \boldsymbol{\sigma}_g^h$ and $\mathbf{a}_2 = \ldots = \mathbf{a}_{10} = 0$.

3.3.3.4 Equilibrated recovery

The loss of equilibrium caused by convection or remapping (i.e. the motion equation (II-6) is no longer satisfied) is generally not severe, and can be handled as extra residual forces in the next loading step. However it is better to perform an equilibrated remapping. In the literature several techniques have been developed for this purpose.

Srikanth et al. [40] have proposed a method which is applied to a remeshing procedure. Its different stages are schematized in Figure 66. It would be interesting to use it in R-adaptation but it is too expensive in terms of computational time, as it requires a second full resolution of the balance equations.

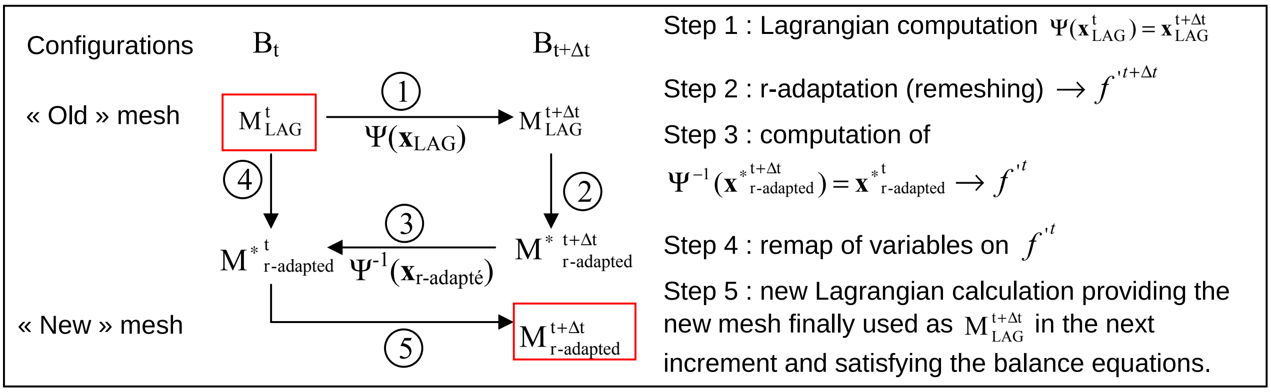


Figure 66: Method proposed by Srikanth et al. to ensure the respect of equilibrium after r-adaptation.

Brancherie and al. [47] have developed a transfer operator based on the diffuse approximation. The reconstruction of the stress field is carried out using operators that preserve the local equilibrium (div σ = 0 and stress admissibility). The recovered stress is approximated by $\boldsymbol{\sigma} = \mathbf{P}^T(\mathbf{x}).\mathbf{a}$, where the local polynomial basis $\mathbf{P}^T(\mathbf{x})$ is chosen in order to automatically satisfy div $\boldsymbol{\sigma}$ = 0, so ensuring the local equilibrium in a diffuse sense. Thus, the computation of the coefficients results in the resolution of the following minimization problem:

$$\min_{\mathbf{a}} \Pi_x(\mathbf{a}) \quad \text{with} \quad \Pi_x(\mathbf{a}) = \frac{1}{2} \sum_{g \in \Omega} w(\mathbf{x}_g - \mathbf{x}) \left\| \mathbf{P}^T(\mathbf{x}_g - \mathbf{x}).\mathbf{a} - \boldsymbol{\sigma}^{old}(\mathbf{x}_g) \right\| \qquad \text{(IV-104)}$$

where $w(\mathbf{x})$ is the weighting function of the diffusive approach.

Following the same idea, we have implemented two different methods in order to equilibrate the recovered stress transport.

The first approach consists in ensuring that the recovered stress satisfies the balance equation at the discrete level (see equation (II-6)):

$$\mathbf{div}(\tilde{\boldsymbol{\sigma}}_g(\mathbf{x})) + \rho(\mathbf{g} - \boldsymbol{\gamma}) = 0 \tag{IV-105}$$

This local equilibrium can be enforced through the recovering operation: the norm of equation (IV-105) is added to the expression (IV-98) as a simple penalizing term. The recovery of the stress $\tilde{\boldsymbol{\sigma}}_g$ at the integration point x_g thus becomes a minimization under constraint, with the following expression:

$$\Pi_g = \Pi_g^{FD} + \Pi_g^{E}$$
$$\text{with}\quad \left| \begin{array}{l} \Pi_g^{FD} = \displaystyle\sum_{f \in P_g} \left(\frac{\sigma_g^h - \tilde{\sigma}_g(f)}{\Delta r_{gf}^2} \right)^2 \\ \Pi_g^{E} = \dfrac{1}{2} \chi \displaystyle\sum_{f \in P_i} \left(\mathbf{div}\left(\tilde{\boldsymbol{\sigma}}_g(f)\right) + \rho(\mathbf{g} - \boldsymbol{\gamma}(f)) \right)^2 \end{array} \right. \tag{IV-106}$$

In order to make the additional term independent on the problem dimensions, the penalizing coefficient χ is defined as follows:

$$\chi = \frac{\chi_0}{(h_e)^2} \quad \text{with} \quad \chi_0 >> 1 \tag{IV-107}$$

where h_e is the characteristic size of the central element and χ_0 the penalizing constant.

The minimization leads to cancel the derivatives of Π_g:

$$\frac{\partial \Pi_g}{\partial a_n^{g,ij}} = \frac{\partial \Pi_g^{FD}}{\partial a_n^{g,ij}} + \frac{\partial \Pi_g^{E}}{\partial a_n^{g,ij}} = 0 \tag{IV-108}$$

If the first order Taylor expansion is utilized, then the divergence of $\tilde{\boldsymbol{\sigma}}_g$ is constant on the patch. On the other hand, the derivatives of Π_g^E provides a coupling between the different components ij of the recovered stress. It is no longer possible to solve the 6 systems 3x3 independently (to determine the 3 coefficients a_i for each of the 6 components ij of the recovered stress); in other words the global system 18x18 has extra-diagonal terms. This method is called PR1E (Patch Recovery with first order expansion and Equilibrium penalization).

This method is more difficult to implement when a 2nd order expansion is used. Alternatively, another method, REP2 (Recovery by Equilibrium in Patches with second order expansion), is preferred. This second approach has been introduced by Zienkiewicz [48-50] for patch centred on nodes whereas here it is applied on patch centred on elements and thus directly provides $\tilde{\boldsymbol{\sigma}}_g$ without any additional interpolation.

The principle consists in equilibrating the recovered stress in a weak sense on the patch. It is based on the weak form of the equilibrium equation (II-6):

$$\forall\, u_h^* \in U_0^{ca},\quad F_{\Omega_h} = \int_{\Omega_h} \sigma_h : \varepsilon_h(u_h^*)\,d\omega \;-\; \int_{\Omega_h} f\,.\,u_h^* \;-\; \int_{\partial_T\Omega_h} T^0\,.\,u_h^*\,ds \;=\; 0 \tag{IV-109}$$

where T^0 are loads imposed on the boundary $\partial_T\Omega_h$ of the considered FE domain Ω_h.

Considering a patch P_g of volume Ω_g, which is included in Ω_h, the integration in (IV-109) can be split in two: one term is integrated on the patch P_g and the other one is integrated on the remaining domain $\Omega_h - \Omega_g$.

$$\forall\, u_h^* \in U_0^{ca},\quad -F_{\Omega_h-\Omega_g} = \int_{\Omega_g} \sigma_h : \varepsilon_h(u_h^*)\,d\omega \;-\; \int_{\Omega_g} f\,.\,u_h^*\,d\omega \;-\; \int_{\partial_T\Omega_g} T^0\,.\,u_h^*\,ds \tag{IV-110}$$

$F_{\Omega_h-\Omega_g}$ are actions of the remaining domain Ω_h-Ω_g on the patch Ω_g.

Finally, we can write (IV-110):

$$\forall\, u_h^* \in U_0^{ca},\quad \int_{\Omega_g} \sigma_h : \varepsilon_h(u_h^*)\,d\omega = R_h \tag{IV-111}$$

where R_h represents the sum of volume forces and external loads. Considering the patch Ω_g as an isolated domain, the loads R_h contribute to the equilibrium (weak sense) of its stresses.

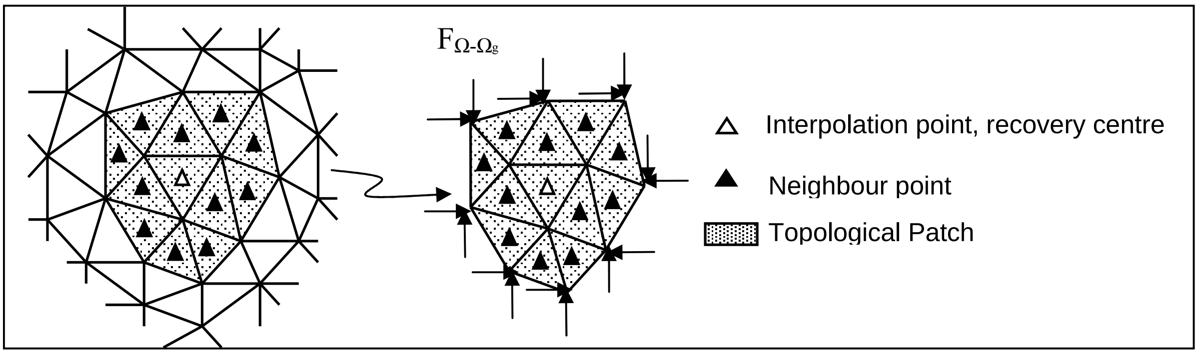


Figure 67: Patch for PR2E and REP2 techniques (schematized in 2D)

In order that the recovered stress $\tilde{\sigma}$ satisfies the local equilibrium on the patch in a weak sense, the following equation is to be verified:

$$\forall\, u_h^* \in U_0^{ca}, \quad \int_{\Omega_g} \tilde{\sigma}_h^g : \varepsilon_h\,(u_h^*)\,d\omega \;= R_h \tag{IV-112}$$

Where R_h is given by (IV-111), so we get:

$$\forall\, u_h^* \in U_0^{ca}, \quad \int_{\Omega_g} \left(\tilde{\boldsymbol{\sigma}}_h^g - \boldsymbol{\sigma}_h\right) : \boldsymbol{\varepsilon}_h\,(\mathbf{u}_h^*)\,d\omega = 0 \tag{IV-113}$$

which will be enforced in a least square sense [see (IV-115) and (IV-116)].

Considering that ε_h is written as:

$$\begin{gathered} \varepsilon_h = \sum_l \sum_\lambda \mathbf{B}_{l\lambda}.u_l^\lambda \\ \text{with } B_{l\lambda}^{ij} = \frac{1}{2}\left[\frac{\partial N_l}{\partial x_j}\delta_{i\lambda} + \frac{\partial N_l}{\partial x_i}\delta_{j\lambda}\right] \end{gathered} \tag{IV-114}$$

u_l^λ is the component λ (λ=1,..,3) of the displacement at node l. δ is the Kroenecker operator:

$$\begin{cases} \delta_{ij} = 1 \text{ if } i = j \\ \delta_{ij} = 0 \text{ if } i \neq j \end{cases}$$

Thus, the new following expression (IV-115) is minimized in order to compute the polynomial coefficients:

$$\Pi(\mathbf{a}^g) = \sum_{\lambda=1}^{3} \sum_{l\in\Omega_g} \left(\int_{\Omega_g} \left(\sum_{m=1}^{10} P_m^g.\mathbf{a}_m^g - \boldsymbol{\sigma}_h \right) : \mathbf{B}_{l\lambda}\ d\omega \right)^2 \tag{IV-115}$$

which leads to solve the system:

$$\forall\, n = 2,..,10\ ,\forall\, i = 1,..,3\ ,\forall\ j=1,..,3\ ,\ \frac{\partial \Pi(\mathbf{a}^g)}{\partial a_n^{g,ij}} = 0 \tag{IV-116}$$

With the suitable discrete form:

$$\begin{gathered} \forall\, n = 2,..,10\ ,\forall\, i = 1,..,3\ ,\forall\ j=1,..,3, \\ \sum_{\substack{i',j' \\ \in\{1,2,3\}}} \sum_{m=1}^{10} \left(\sum_{\lambda=1}^{3} \sum_{l\in\Omega_g} \int_{\Omega_g} P_n\, B_{l\lambda}^{ij}\, d\omega \int_{\Omega_g} P_m\, B_{l\lambda}^{i'j'}\, d\omega \right).a_m^{g,i'j'} = \\ \sum_{\substack{i',j' \\ \in\{1,2,3\}}} \sum_{\lambda=1}^{3} \sum_{l\in\Omega_g} \int_{\Omega_g} \sigma_h^{i'j'}\, B_{l\lambda}^{i'j'}\, d\omega \int_{\Omega_g} P_n\, B_{l\lambda}^{ij}\, d\omega \end{gathered} \tag{IV-117}$$

This system can be written as a matrix problem **A**.**x**=**B** of dimension 54×54:

$$\forall n=2,..,10\ ,\forall i=1,..,6$$

$$\begin{cases} \sum_{m=1}^{10}\sum_{i'=1}^{6} A_{nm}^{ii'}.x_m^i = b_n^i \\ x_m^i = a_m^{g,i} \\ A_{nm}^{ii'} = \sum_{\lambda=1}^{3}\sum_{l\in\Omega_g} \int_{\Omega_g} P_n\, B_{l\lambda}^{i}\, d\omega \int_{\Omega_g} P_m\, B_{l\lambda}^{i'}\, d\omega \\ b_n^i = \sum_{i'=1}^{6}\sum_{\lambda=1}^{3}\sum_{l\in\Omega_g} \int_{\Omega_g} \sigma_h^{i'}\, B_{l\lambda}^{i'}\, d\omega \int_{\Omega_g} P_n\, B_{l\lambda}^{i}\, d\omega \end{cases} \tag{IV-118}$$

To simplify the problem, the following writing convention for a symmetric tensor T of third order, has been used:

$$\begin{gathered} \forall\, i=1,..,3\ ,T^{ii}=T^{i} \\ T^{12}=T^{21}=T^{4},\ T^{23}=T^{32}=T^{5},\ T^{13}=T^{31}=T^{6} \end{gathered} \tag{IV-119}$$

Remark: the imposition of the local equilibrium introduces a coupling between the components of the recovered stress as for the PR1E technique.

3.3.3.5 Summary of developed methods

To summarise we have developed the PR1, PR2, MPR, PR1E, REP2 techniques:

- PR1, PR1E and MPR are first order methods
- MPR is based on minimal patch with additional boundary conditions
- PR1E and REP2 satisfy the balance equation, in a strong and weak sense respectively.

This is illustrated on Figure 68.

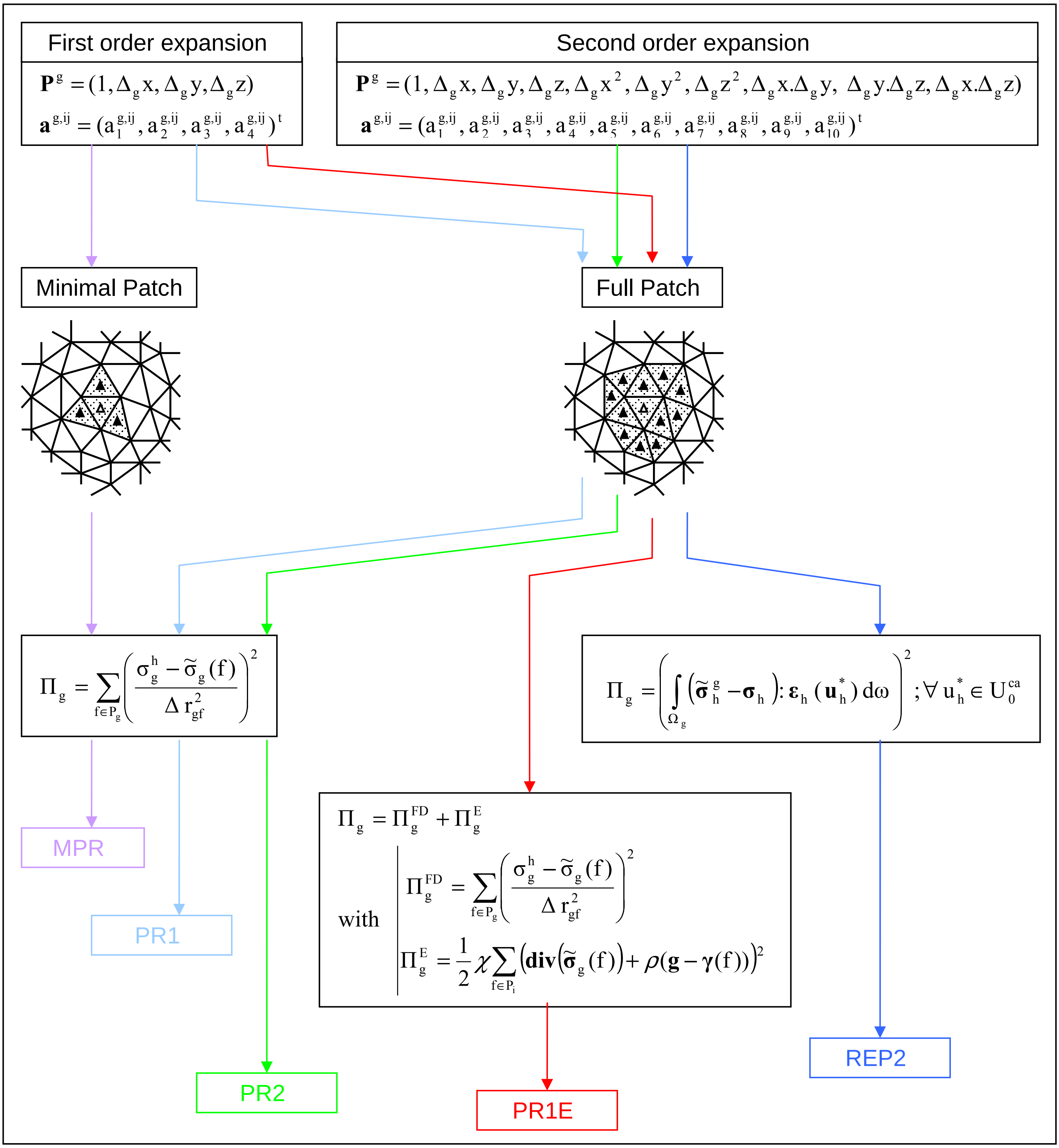


Figure 68: Schematization of the differences between the developed methods.

3.3.3.6 Extrapolation

For each of the presented recovery techniques, the polynomial factors defining the recovered variable are stored at the patch centers i.e. at each integration point of the Lagrangian updated configuration. The definition of the recovered variable is valid on the whole patch and can thus be used to extrapolate the values at the integration points of the ALE updated mesh that belong to this patch.

k being the ALE location of the integration point and g the nearest integration point of the Lagrangian updated configuration, the value of σ at k is then interpolated from the value of the recovered expression at k:

$$\sigma_{k\,ALE}^{ij\;t+\Delta t} = \tilde{\sigma}_g^{ij}(\mathbf{x}_k) = \sum_{m=1}^{n} P_m^g(\mathbf{x}_k).a_m^{ij,g} = \mathbf{P}^{kg}.\mathbf{a}^{ij,g}$$

$$\text{with}\left|\begin{array}{l}\mathbf{P}^{kg} = (1, \Delta_{kg}x, \Delta_{kg}y, \Delta_{kg}z, \Delta_{kg}x^2, \Delta_{kg}y^2, \Delta_{kg}z^2, \Delta_{kg}x.\Delta_{kg}y,\ \Delta_{kg}y.\Delta_{kg}z, \Delta_{kg}x.\Delta_{kg}z) \\ \mathbf{a}^{g,ij} = (a_1^{g,ij}, a_2^{g,ij}, a_3^{g,ij}, a_4^{g,ij}, a_5^{g,ij}, a_6^{g,ij}, a_7^{g,ij}, a_8^{g,ij}, a_9^{g,ij}, a_{10}^{g,ij})^t \ ;\ a_1^{g,ij} = \sigma_g^{h,ij}\end{array}\right. \qquad \text{(IV-120)}$$

To summarize, for one given point the knowledge of the nearest Lagrangian updated point and of its attached coefficients is sufficient to compute the new values. Possible distortions of the Lagrangian mesh do not influence the results because most of the proposed recovery techniques (except REP2) are only based on the position of the integration points at t+Δt in the Lagragian frame and do not require any finite elements integration.

Remark: several tests have shown that it is better to select the patch which center x_g is closer to the considered point x_k, rather than the patch which actually contains x_k (in other words, of which one element contains x_k) to carry out this extrapolation.

3.3.4 Nodal recovery technique

Another way to remap P0 elementary variables consists in using a nodal P1 intermediate projection: the P0 variables are first extrapolated to Lagrangian updated nodes, and then interpolated to new ALE integration points. Such a method is schematized in Figure 69.

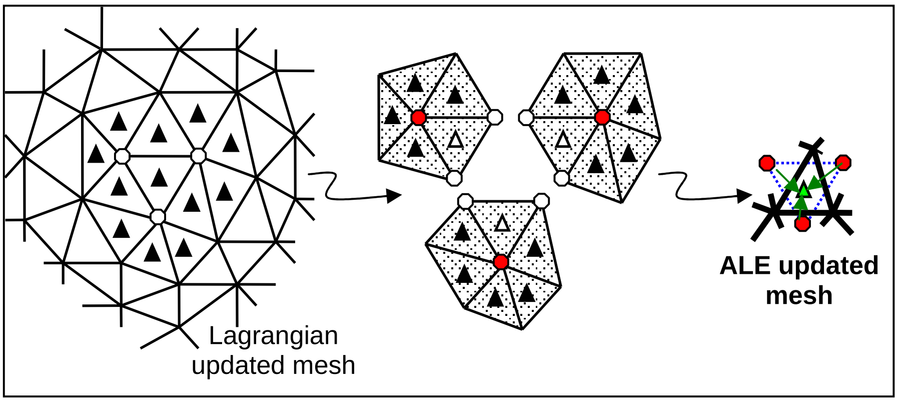


Figure 69: Nodal extrapolation technique for P0 variable remapping (schematized in 2D)

This method has the advantage of being easily implementable in the parallelized procedures of the code (construction of patches and then resolution of the systems of equations for each element). It makes it possible to have only one transfer procedure for P1 variables.

3.3.4.1 Classical nodal least square smoothing procedure

The nodal least square method is the projection method which is commonly used in the remeshing procedure of the code. The value of a P0 field can be written in any point **x**:

$$\sigma^{ij}(x) = \sum_{e=1}^{Nbelt} \sigma_e^{ij} 1^e(\mathbf{x}) \tag{IV-121}$$

where 1^e is a function which is equal to 1 on the element e, and null everywhere else.
This field is projected onto a nodal P1 field as follows:

$$\tilde{\sigma}^{ij}(x) = \sum_{k=1}^{Nbnoe} \tilde{\sigma}_k^{ij} N^k(\mathbf{x}) \tag{IV-122}$$

where $\tilde{\sigma}_k^{ij}$ is computed at node k by minimizing the following least square expression:

$$\Pi_k\left(\tilde{\sigma}_k^{ij}\right) = \sum_{e=1}^{Nbelt} V^e \left(\sum_{k=1}^{Nbnoe} \tilde{\sigma}_k^{ij} N^k(\zeta^e) - \sigma_e^{ij} \right)^2 \tag{IV-123}$$

where $\zeta^e = (\zeta^e_1, \zeta^e_2, \zeta^e_3)$ are the barycentric coordinates of the center of the element e with a volume V^e.

3.3.4.2 Super Convergent Patch Recovery (SPR)

These nodal values can also be recovered using the Liska Orkiz Finite Differences method, which has been described in section 3.3.3.1 for patch centred at integration points. The difference with the method used here lies on the form of the patch: they are centred at the nodes (see Figure 69 and 70).

This recovery technique, first proposed by [46], is based on the same principle as the previously described one:

- the variable is defined for any point **x** belonging to the patch centred on node k, by using a first order Taylor expansion (see equations (IV-90) and (IV-91)).
- the 4 unknown polynomial factors are identified by minimizing the following least square expression (which is slightly different from (IV-96)) for each component ij of the variable:

$$\Pi(\mathbf{a}_k^{ij}) = \sum_{g=1}^{NG} \left(\sigma_g^{ij\,h} - \tilde{\sigma}_k^{ij}(g)\right)^2 = \sum_{i=1}^{NG} \left(\sigma_g^{ij\,h} - \mathbf{P}^k(g).\mathbf{a}^{k,ij}\right)^2 \qquad \text{(IV-124)}$$

where NG is the number of integration points in the patch. Thus, the problem consists in solving the following system:

$$\forall\, n = 1..4, \quad \frac{\partial \Pi(\mathbf{a}^{k,ij})}{\partial a_n^{k,ij}} = 0 \qquad \text{(IV-125)}$$

It provides the nodal value: $\tilde{\sigma}_k^{ij} = a_1^{k,ij}$ that will be used in the interpolation (IV-122) (see 3.3.4.4).

If the number of elements in the patch is insufficient to obtain a well conditioned system, the patch is extended to second order neighbours (see Figure 70), and additional stabilizing terms are added to the expression (IV-124):

$$\Pi'(\mathbf{a}_k^{ij}) = \sum_{g=1}^{NGt}\left(\sigma_g^{ij\,h} - \tilde{\sigma}_k^{ij}(g)\right)^2 + \omega\sum_{g=1}^{NGs}\left(\sigma_g^{ij\,h} - \tilde{\sigma}_k^{ij}(g)\right)^2 \qquad \text{(IV-126)}$$

where NGt is the number of first order neighbouring integration points in the patch. NGs is the number of second order integration points. ω is the stabilizing coefficient such that: $0 \leq \omega < 1$

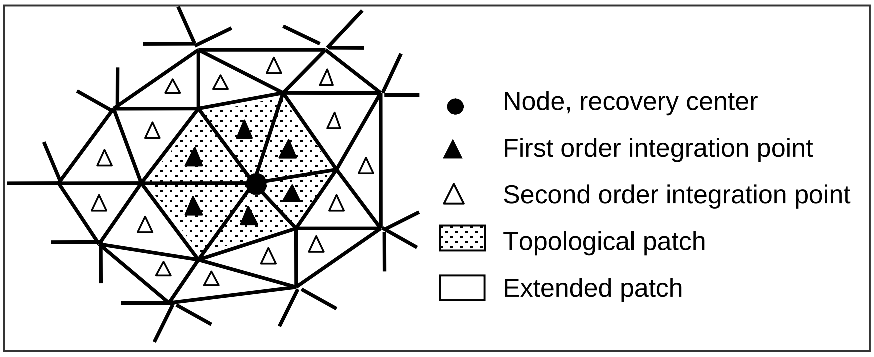


Figure 70: Patch centred on a node in 2D

3.3.4.3 PR2 technique for P1 fields

The Nodal PR2 technique introduced in section 3.2.3 has been extended from P0 to P1 fields:

- As in this later SPR method, a nodal continuous value $\tilde{f}_k$ is recovered on a patch centred on node k of the Lagrangian updated configuration. The difference comes from the fact that the information is provided by the nodes n rather than by the integration points g (see Figure 71). A second order expansion is used for accuracy reasons (the field has already a P1 accuracy so the recovery implies a higher order), so the expression to be minimized is the following one:

$$\Pi'(\mathbf{a}_k^{ij}) = \sum_{n=1}^{Nt}\left(f_n^h - \tilde{f}_k(n)\right)^2 + \omega\sum_{n=1}^{Ns}\left(f_n^h - \tilde{f}_k(n)\right)^2 \qquad \text{(IV-127)}$$

where Nt is the number of first order neighbouring nodes in the patch. Ns is the number of second order nodes in the patch.

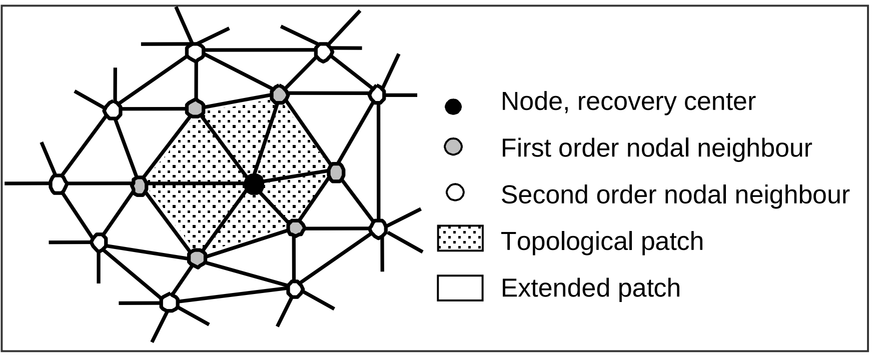


Figure 71: Patch centred on a node in 2D, with nodal values to recover.

- The value at a given ALE updated node m is then interpolated from its nearest Lagrangian updated node k through the second order approximation of the variable on the associated patch (as in the previously presented Patch Recovery techniques):

$$f_{m\,ALE}^{t+\Delta t} = \tilde{f}_k(m) = \sum_{i=1}^{10} P_i^k(m).a_i^k = \mathbf{P}^{mk}.\mathbf{a}^k$$

$$\text{with}\left|\begin{array}{l}\mathbf{P}^{mk} = (1, \Delta_{mk}x, \Delta_{mk}y, \Delta_{mk}z, \Delta_{mk}x^2, \Delta_{mk}y^2, \Delta_{mk}z^2, \Delta_{mk}x.\Delta_{mk}y,\ \Delta_{mk}y.\Delta_{mk}z, \Delta_{mk}x.\Delta_{mk}z) \\ \mathbf{a}^k = (a_1^k, a_2^k, a_3^k, a_4^k, a_5^k, a_6^k, a_7^k, a_8^k, a_9^k, a_{10}^k)^t \ ;\ a_1^k = f_k^h\end{array}\right. \quad \text{(IV-128)}$$

In the following section, it will be numerically observed that this technique provides less numerical diffusion during remapping than the classical nodal interpolation or the upwind techniques presented in the previous section 3.2.

3.3.4.4 Recovery by Equilibrium in Patches (REP)

The technique developed to recover locally equilibrated stresses (see section 3.3.3.4) can also be applied to patches which are centred at nodes (which was the original implementation of the method). It has been implemented with a first order expansion: the expression (IV-124) is then replaced by:

$$\Pi(\mathbf{a}^k) = \sum_{\lambda=1}^{3} \sum_{l\in\Omega_k} \left(\int_{\Omega_k} \left(\sum_{m=1}^{4} P_m^k(g).\mathbf{a}_m^k - \boldsymbol{\sigma}_g^h \right) : \mathbf{B}_{l\lambda} \ d\omega \right)^2 \quad \text{(IV-129)}$$

The minimization leads to solve the following 24×24 system:

$$\forall n = 1,..,4\ , \forall i = 1,..,3\ , \forall\ j=1,..,3\ ,\ \frac{\partial \Pi(\mathbf{a}^k)}{\partial a_n^{k,ij}} = 0 \quad \text{(IV-130)}$$

Which provides the nodal value: $\tilde{\sigma}_k^{ij} = a_1^{k,ij}$.

Note that contrary to section 3.3.3.4, there is no consistency condition to satisfy at the center of the patch, so the constant value $a_1^{k,ij}$ is not prescribed.

3.3.4.5 Summarize of Nodal recovery developed methods

Figure 72 illustrates the nodal recovery methods for P0 variables remapping.

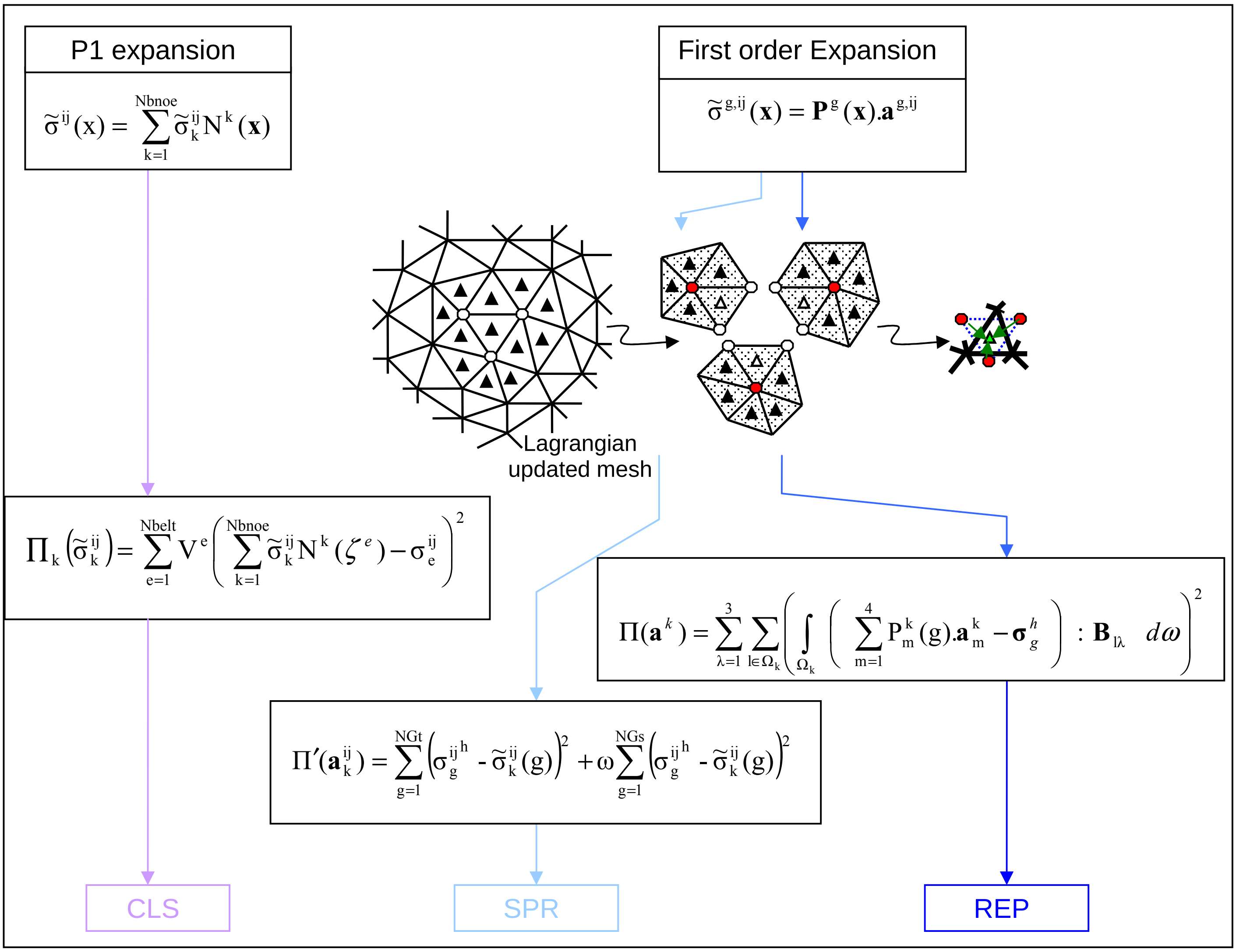


Figure 72: Schematization of the nodal recovery techniques for P0 variables remapping.

3.3.4.6 Interpolation

The second step of the nodal recovery method consists in interpolating the recovered nodal field at the ALE integration points. It can be carried out using the shape functions of the Lagrangian updated element e which contains the ALE integration point:

$$\forall \mathrm{i}=1,..,3\ ,\forall\ \mathrm{j}=1,..,3\ ,\ \forall\, \mathrm{g}\in \mathrm{M}_{\mathrm{ALE}}^{\mathrm{t+\Delta t}},\quad \left(\tilde{\sigma}_{\mathrm{g}}^{\mathrm{ij}}\right)^{\mathrm{t+\Delta t}}=\sum_{\substack{\mathrm{k}\in \mathrm{e}\\ \mathrm{g}\supset \mathrm{e}}}\tilde{\sigma}_{\mathrm{k}}^{\mathrm{ij}}\ \mathrm{N}_{\mathrm{k}}\left(\varsigma_{g}^{e}\right) \tag{IV-131}$$

However, this approach does not satisfy the consistency of the remapping technique. If the ALE integration point g has the same coordinates as the Lagrangian updated integration point g’, we should have:

$$\left(\tilde{\sigma}_{\mathrm{g}}^{\mathrm{ij}}\right)^{\mathrm{t+\Delta t}}=\sigma_{\mathrm{g'}}^{\mathrm{ij}} \tag{IV-132}$$

The error δ resulting from the the interpolation (IV-131) can be written as:

$$\delta=\left(\sum_{\substack{\mathrm{k}\in \mathrm{e}\\ \mathrm{g}\supset \mathrm{e}}}\tilde{\sigma}_{\mathrm{k}}^{\mathrm{ij}}\ \mathrm{N}_{\mathrm{k}}\left(\varsigma_{g'}^{e}\right)\right)-\sigma_{\mathrm{g'}}^{\mathrm{ij}} \tag{IV-133}$$

It can then be eliminated from the interpolation procedure by introducing the following correction:

$$\begin{aligned}&\forall \mathrm{i}=1,..,3\ ,\forall\ \mathrm{j}=1,..,3\ ,\ \forall\, \mathrm{g}\in \mathrm{M}_{\mathrm{ALE}}^{\mathrm{t+\Delta t}}:\ \left(\tilde{\sigma}_{\mathrm{g}}^{\mathrm{ij}}\right)^{\mathrm{t+\Delta t}}=\left(\sum_{\substack{\mathrm{k}\in \mathrm{e}\\ \mathrm{g}\supset \mathrm{e}}}\tilde{\sigma}_{\mathrm{k}}^{\mathrm{ij}}\mathrm{N}_{\mathrm{k}}\left(\varsigma_{g}^{e}\right)\right)-\delta\\ &\Leftrightarrow\quad \left(\tilde{\sigma}_{\mathrm{g}}^{\mathrm{ij}}\right)^{\mathrm{t+\Delta t}}=\sigma_{\mathrm{g'}}^{\mathrm{ij}}+\sum_{\substack{\mathrm{k}\in \mathrm{e}\\ \mathrm{g}\supset \mathrm{e}}}\tilde{\sigma}_{\mathrm{k}}^{\mathrm{ij}}\left(\mathrm{N}_{\mathrm{k}}^{e}\left(\varsigma_{g}^{e}\right)-\mathrm{N}_{\mathrm{k}}^{e}\left(\varsigma_{g'}^{e}\right)\right)\end{aligned} \tag{IV-134}$$

However, and on the other hand this modification generates an error equal to δ if the ALE integration point g is close to a Lagrangian updated node k.

Therefore, another interpolation technique has been preferred by considering the sub-tetrahedra of an element which are built from the central integration point and lie on its facets. The P1 variable can thus be expressed as follows:

$$\forall \mathrm{i}=1,..,3\ ,\forall\ \mathrm{j}=1,..,3\ ,\ \forall\, \mathrm{g}\in \mathrm{M}_{\mathrm{ALE}}^{\mathrm{t+\Delta t}},\quad \left(\tilde{\sigma}_{\mathrm{g}}^{\mathrm{ij}}\right)^{\mathrm{t+\Delta t}}=\sum_{\substack{\mathrm{k}\in \mathrm{e'}\\ \mathrm{g}\supset \mathrm{e'}}}\tilde{\sigma}_{\mathrm{k}}^{\mathrm{ij}}\ \mathrm{N}_{\mathrm{k}}^{e'}\left(\varsigma_{g}^{e'}\right)+\sigma_{\mathrm{g'}}^{\mathrm{ij}}\ \mathrm{N}_{\mathrm{g'}}^{e'}\left(\varsigma_{g}^{e'}\right) \tag{IV-135}$$

Where e’ is the sub tetrahedron of e which contains the ALE point g and N

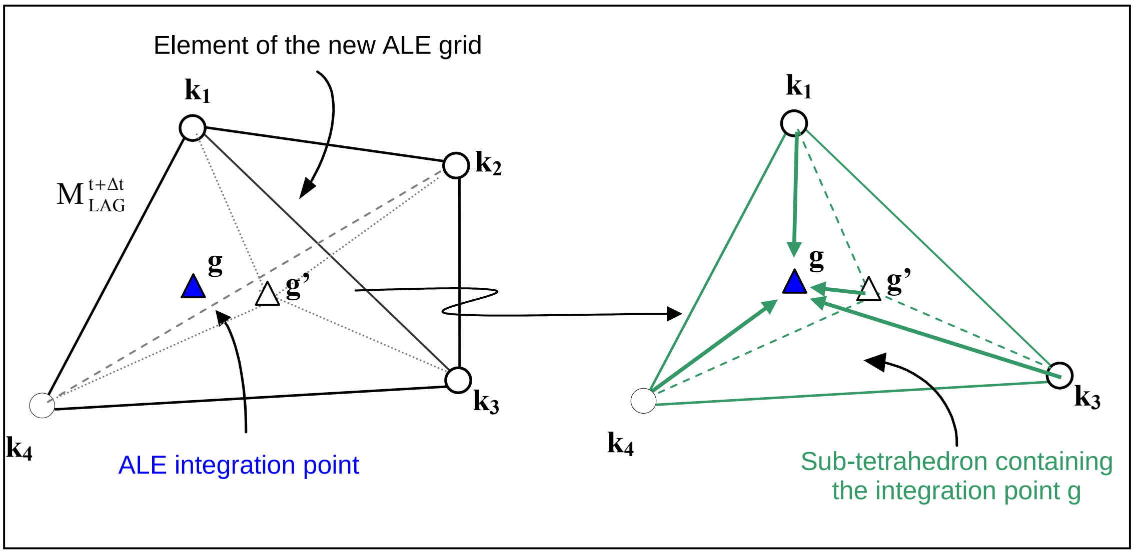


Figure 73: Sub-tetrahedron interpolation

3.4 Comparisons and benchmark tests

In order to compare the accuracy and efficiency of implemented techniques, two main tests have been performed.

3.4.1 Qualitative evaluation of the accuracy

A first simple test consists in considering the transport of a constant field of value throughout a fixed grid. This field is moved at a constant prescribed velocity, so only the remapping procedure of the ALE method is considered (it can represent a welding field of temperature).

The initial field is built by imposing the value of 100 inside a spherical box which is centred at mid-depth of the plate, as represented in Figure 74.

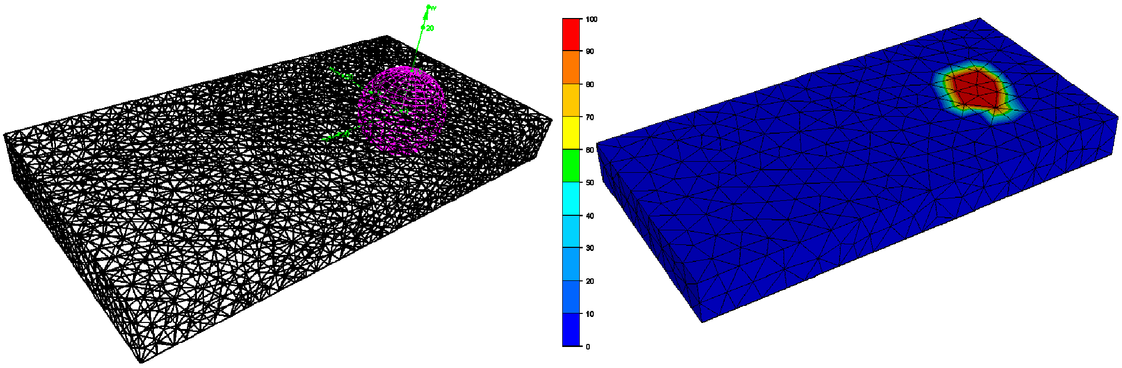


Figure 74: Construction of the constant field to remap on the grid A

Two refinements A and B of the Eulerian mesh have been utilized (see Figure 75). The two following sub-sections show the results which have been obtained with some of the more promising implemented techniques only. It is important to notice that this test is very severe for the remapping methods. In fact the gradient of the initial field is very large: the value decreases from 100 (maximal value in red) to 0 (minimum value in blue) within one element size and the number of elements or nodes which are set to the maximum value is very limited. The isovalues of 80 and 50 are respectively visualized for nodal and element values in the following.

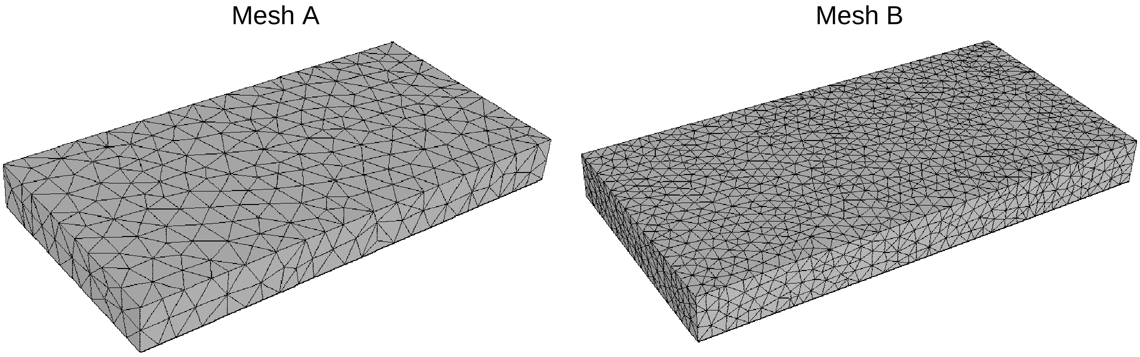


Figure 75: Two mesh refinements utilized for testing the remapping efficiency

3.4.1.1 Nodal Field (P1 variables)

Figure 76 shows that the Nodal Upwind technique (presented in section 3.2.2) provides results which are similar to those obtained with the classical interpolation technique (presented in section 3.2.1): they are both too diffusive. The mesh should be adapted according to the gradient variable in order that both methods be conservative. On the other hand, the Nodal Upwind technique has the advantage of avoiding the construction of the Updated Lagrangian configuration, which makes it possible to tackle with the problem of possible element distortions.

The Nodal PR2 technique (section 3.3.4.2) highly improves the accuracy and the conservativity of the remap step. It is visualized in Figure 77 for two different refinements. Contrary to the two previous techniques, the isovalue of 80% of the maximum value is well conserved throughout the remap. On the coarsest grid, the technique is slightly diffusive:
dragging values appear and the global spot do not reach the end of the plate after 35 remapping step. This is mainly due to the fact that the minimal number of neighbours that is necessary to perform a second order expansion is not always achieved and consequently that a lower order expansion is used, of first or even zero order (constant transfer). This drawback can be improved by replacing (or combining) the PR2 Nodal technique by (or with) the Upwind Nodal technique when the nodal patch does not contain enough neighbours.

In conclusion, the nodal PR2 technique always provides the best accuracy which is very satisfying on the fine mesh (which is still coarse with respect to the gradient used in the test).

3.4.1.2 Element Field (P0 variables)

Figure 78 compares two direct techniques: the constant (P0) transfer and the first order transfer (PR1). The comparison is done on the coarsest grid A which clearly reveals the total inefficiency of the constant transfer. On the other hand, the higher order transport is much less diffusive and rather conservative.

Figure 79 compares two nodal techniques: the classical nodal least square smoothing method and the superconvergent Patch Recovery (SPR) method. They are significantly better than the P0 transfer, but they are both quite diffusive, much more than the PR1 transfer.

The PR2 (Patch recovery with second order interpolation) technique looks to be the less diffusive of the developed and classical methods. The results obtained on the coarse grid are slightly better than with the PR1 technique and those obtained on the fine grid are very accurate and satisfying (see Figure 80).

In conclusion, the patch recovery techniques, (which are discontinuous per patch so per elements), provide a better accuracy which is dependent of the expansion order: the higher the expansion order the better the accuracy is. Satisfying results have been observed in this very severe test case (strong gradient, coarse grids) and validate the use of the most efficient developed techniques for process simulation.

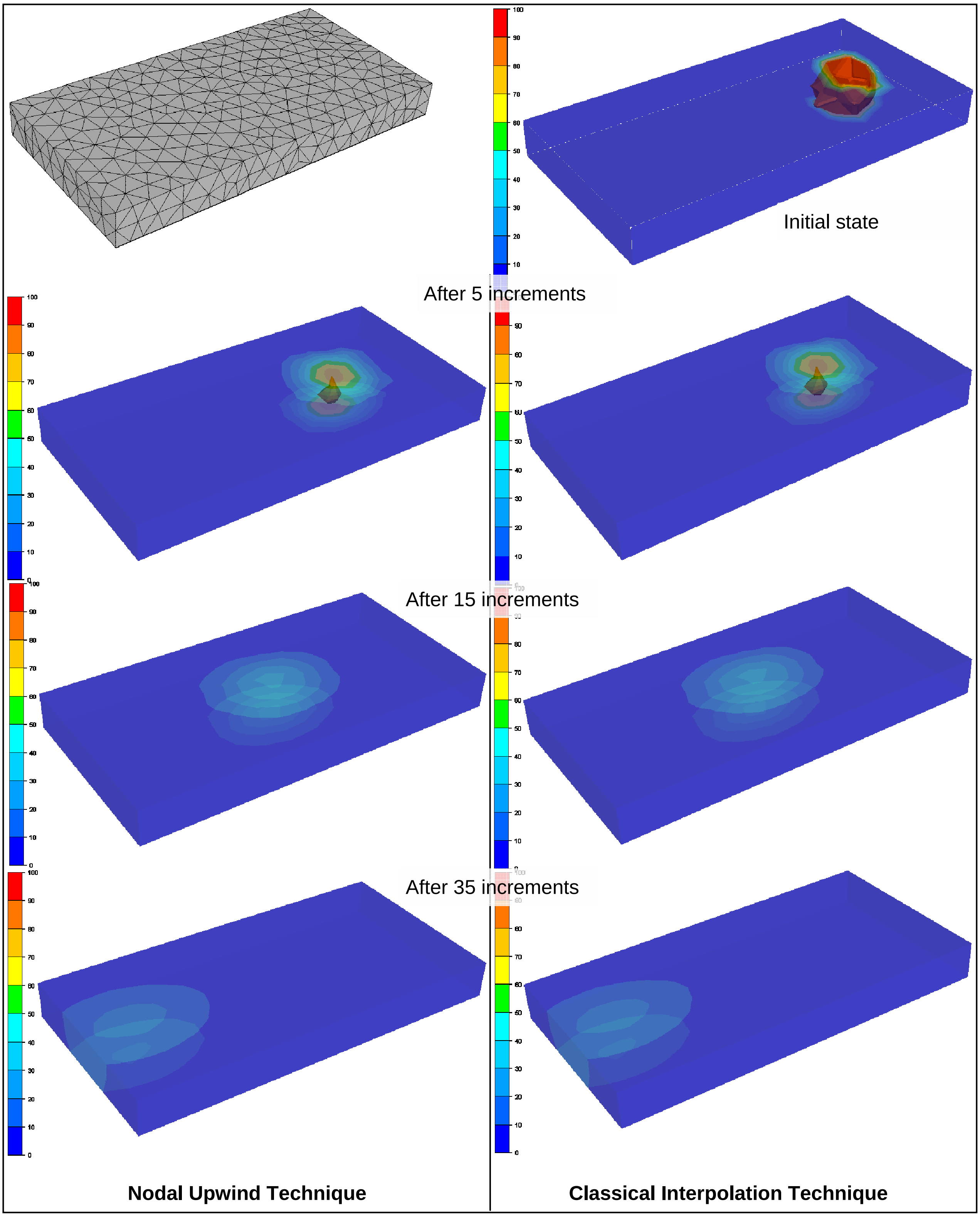


Figure 76: Comparison between Nodal Upwind technique and Classical P1 Interpolation technique for nodal variable remapping

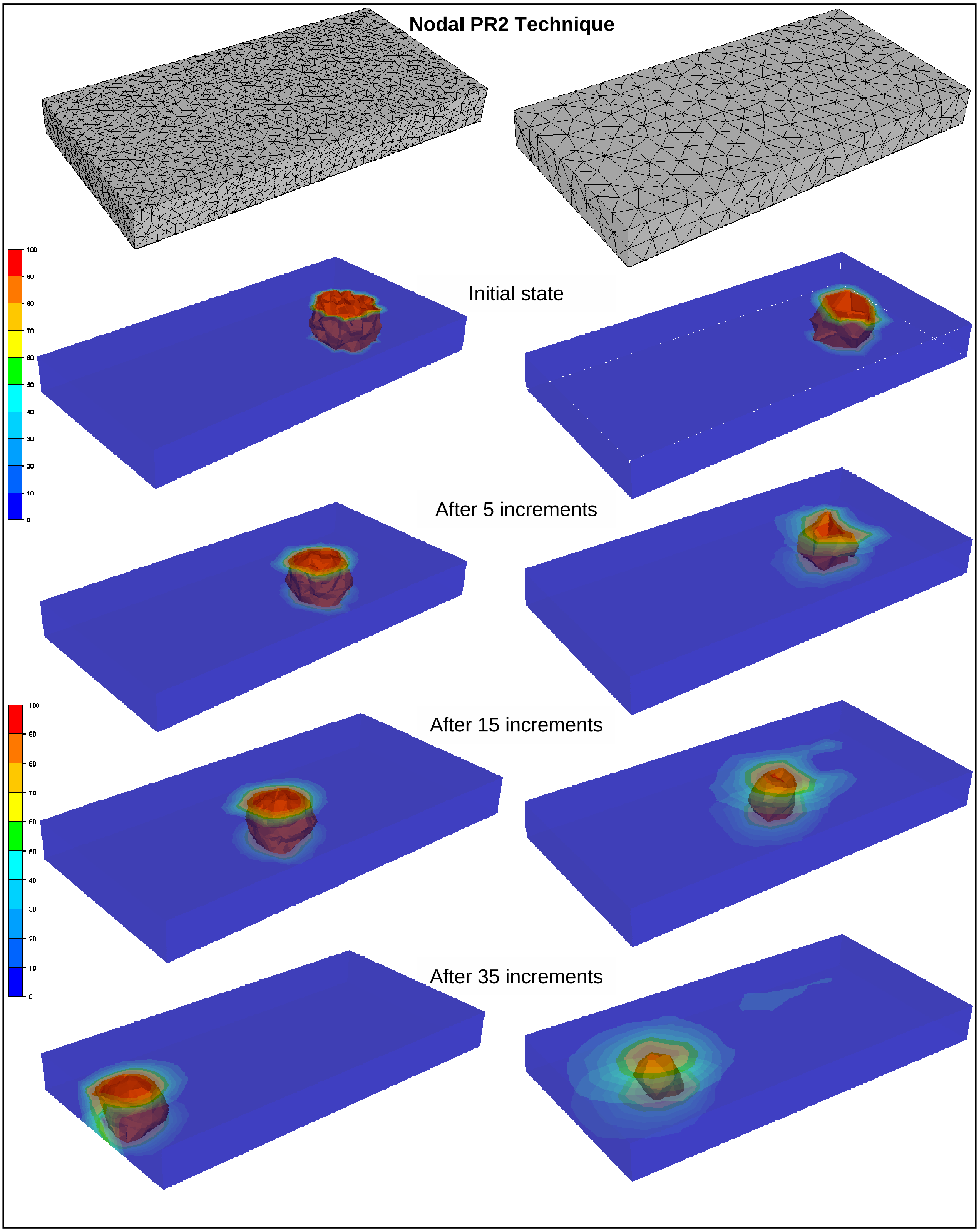


Figure 77: Results obtained with the PR2 technique for nodal variable remapping with two different grid refinements.

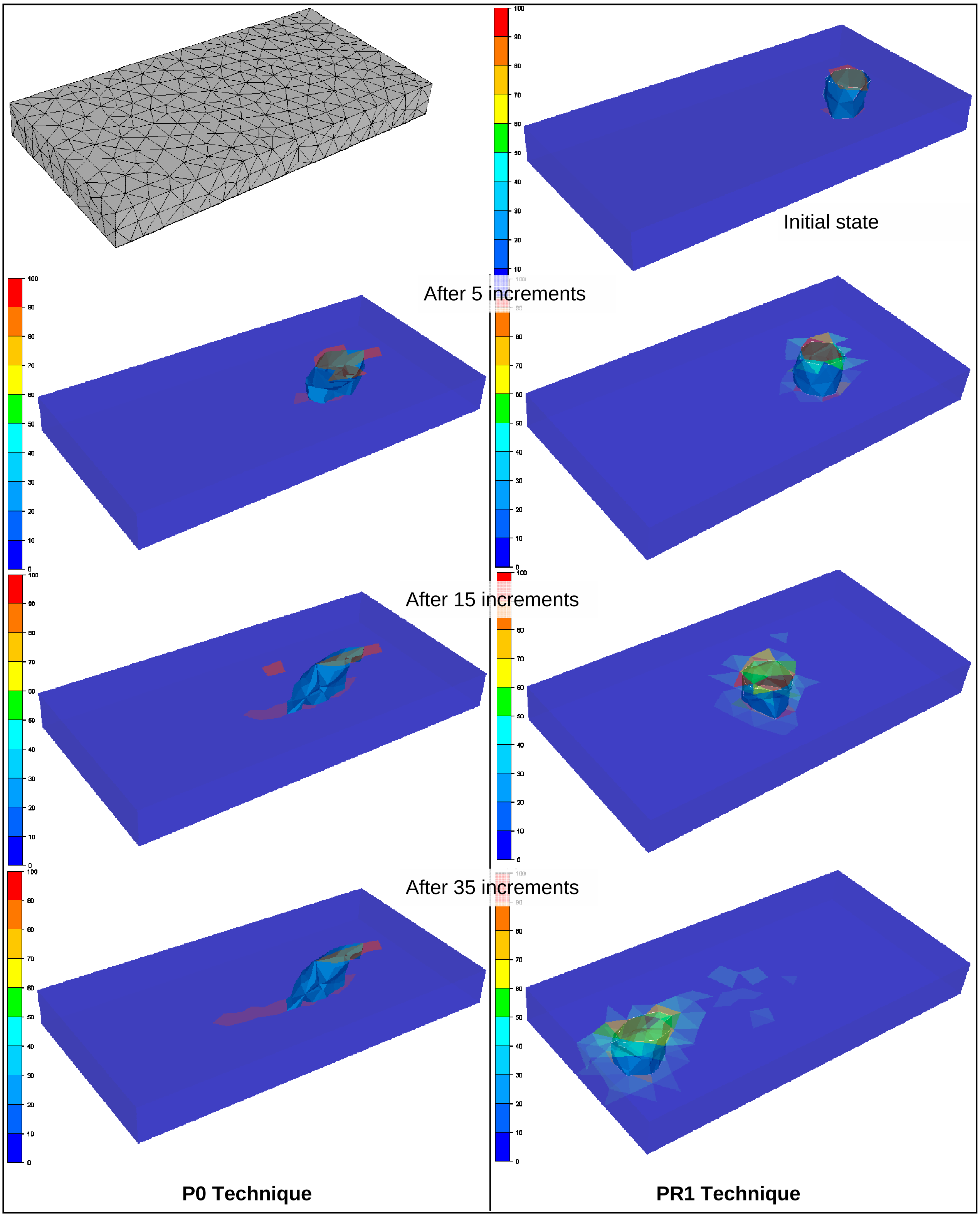


Figure 78: Comparison between the P0 and PR1 techniques for an element variable remapping.

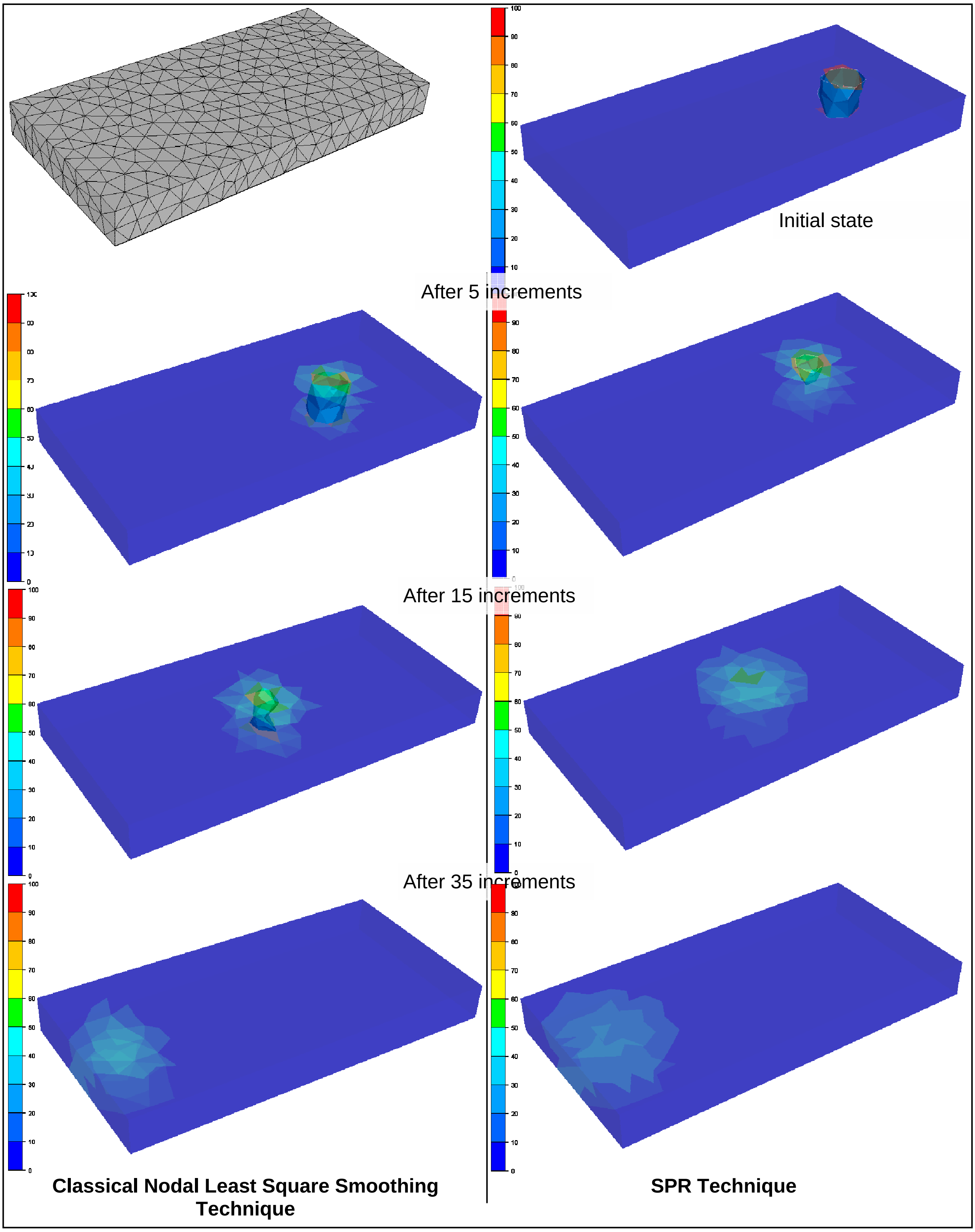


Figure 79: Comparison between the Nodal Least Square Classical smoothing technique and SPR technique for an element variable remapping.

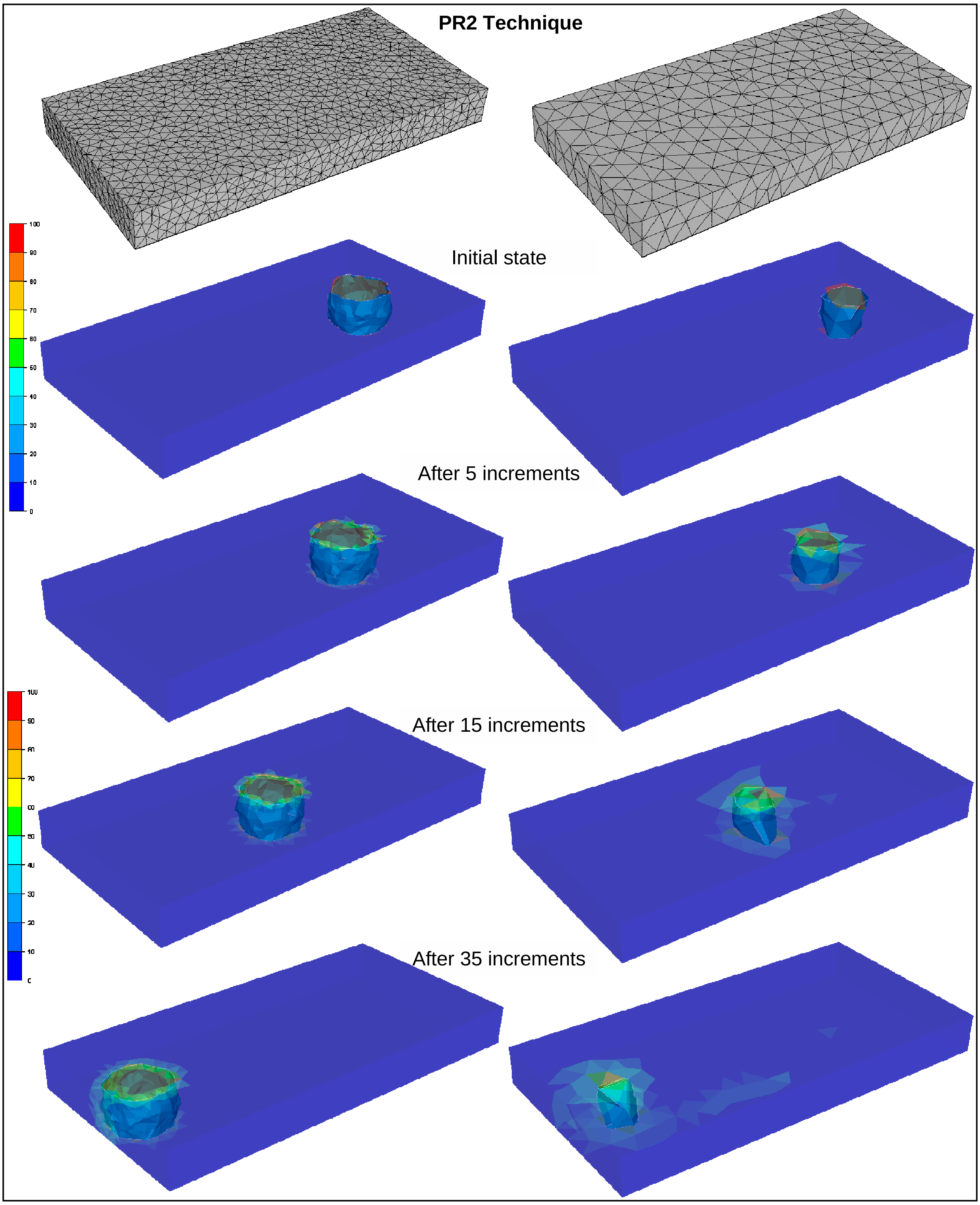


Figure 80: Results obtained with the PR2 technique for element variable remapping with two different grid refinements.

3.4.2 Quantitative evaluation

To evaluate the accuracy of the remapping methods, it is necessary to quantify the error it generates. Therefore a benchmark test has been selected. It is a case for which the analytic exact solution is known: the uni-axial traction of an infinite plate with centred hole (see Figure 81a). For symmetry reasons (in the directions Ox and Oy), calculations are performed only on the part delimited by (a,b,c,d,e,e',a',b',c',d') (see Figure 81b). In order to satisfy the plane strains conditions, symmetry conditions are also applied on the upper and lower faces.

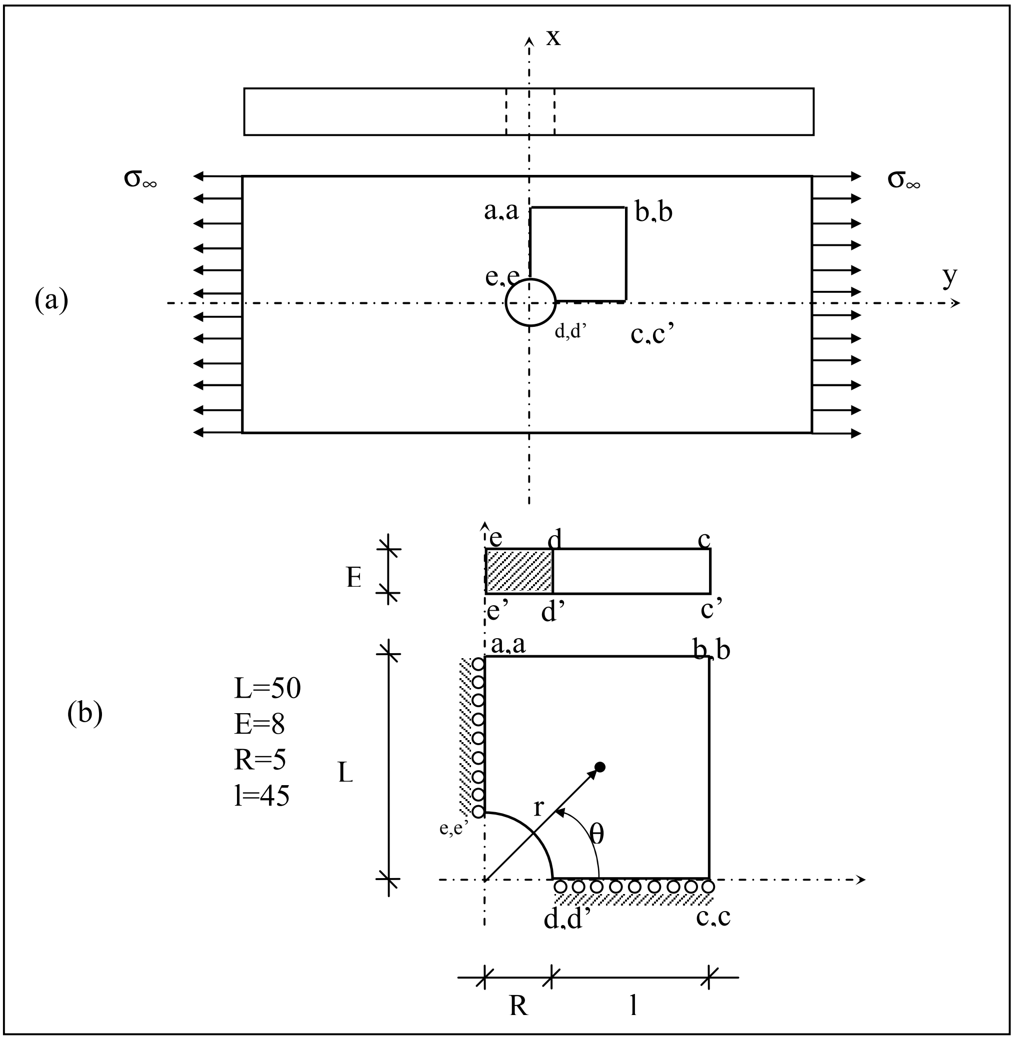


Figure 81: Holed plate for the benchmark test

To determine the error generated by remapping only, the initial stress field is calculated from the exact solution given in equations (IV-136) at each integration point, which lie at the center of the elements.

$$\frac{\sigma_{rr}}{\sigma_\infty}=\frac{1}{2}\left(1-\frac{R^2}{r^2}\right)-\frac{1}{2}\left(1-4\frac{R^2}{r^2}+3\frac{R^4}{r^4}\right)\cos 2\theta$$

$$\frac{\sigma_{r\theta}}{\sigma_\infty}=\frac{1}{2}\left(1+2\frac{R^2}{r^2}-3\frac{R^4}{r^4}\right)\sin 2\theta \qquad \text{(IV-136)}$$

$$\frac{\sigma_{\theta\theta}}{\sigma_\infty}=\frac{1}{2}\left(1+\frac{R^2}{r^2}\right)+\frac{1}{2}\left(1+3\frac{R^4}{r^4}\right)\cos 2\theta$$

$$\sigma_{zz}=\nu\,(\sigma_{rr}+\sigma_{\theta\theta}) \text{ and } \sigma_{\theta z}=\sigma_{rz}=0$$

where (r,θ) are the polar coordinates of any point of the domain, R is the radius of the central hole. σ_∞ is the imposed load, equal to 100 MPa. The elastic parameters are arbitrarily chosen: Young modulus E=2.10^5 MPa, Poisson coefficient ν=0.3.

For each element, the stress field at the center of the element is remapped at each of the four gauss points g'$_i$ of the element e as shown in Figure 82. These remapped values are compared to the analytical ones, and the global error is computed according to equation (IV-137).

$$\xi_{glob}=\frac{\left\|\tilde{\boldsymbol{\sigma}}-\boldsymbol{\sigma}_{exa}\right\|_\Omega}{\left\|\boldsymbol{\sigma}_{exa}\right\|_\Omega}=\frac{\left(\sum_{e\in\Omega}\sum_{i=1}^{4}\left(\tilde{\boldsymbol{\sigma}}(g_i')-\boldsymbol{\sigma}_{exa}(g_i')\right):\left(\tilde{\boldsymbol{\sigma}}(g_i')-\boldsymbol{\sigma}_{exa}(g_i')\right)\right)^{1/2}}{\left(\sum_{e\in\Omega}\sum_{i=1}^{4}\boldsymbol{\sigma}_{exa}(g_i'):\boldsymbol{\sigma}_{exa}(g_i')\right)^{1/2}} \qquad \text{(IV-137)}$$

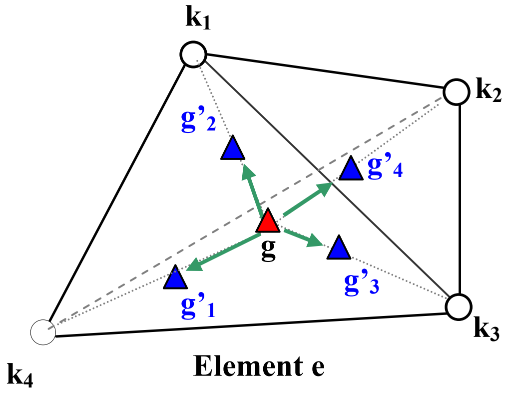


Figure 82: Four gauss points remapping for global error computation

The following graphics (Figures 83, 84, 85) summarize the results obtained with the different implemented techniques. The error has been evaluated for three different mesh refinements from coarse to very fine (see meshes in Figures 83, 84, 85).

Concerning nodal extrapolation techniques, this test shows that the SPR and REP developed methods:

- provide better results than the reference least square smoothing procedure (from the Forge3® code) except the REP method on the medium mesh (a rather amazing and unexplained behaviour to be further investigated).
- converge faster with mesh refinement.

REP does not improve SPR. In other words, considering the balance equation does not increase the accuracy (as it was noticed in error estimation) but might provide more “stable” remapped stresses (in particular for non linear constitutive equations).

These nodal techniques are as or more accurate than the direct one using a minimal patch: MPR and MPRE. However, the remapping error decreases when the size of the patch is enlarged to first and then to second order neighbours or/and when the interpolation order is increased ($\xi_{P0}>\xi_{MPR}>\xi_{PR1}>\xi_{PR2}$.). The convergence rate is higher for the developed methods:

- the error vary in O(h) for the least square smoothing reference method
- the error vary in $O(h^{\alpha})$ with $1<\alpha<2$ for the other developped methods ($\alpha=2$ for the PR2 technique).

This is very interesting to notice that taking into account the equilibrium condition in a strong sense by a local penalizing method increases the accuracy because it shows that enforcing the balance equation provides a better solution closer to the exact one whereas it minimizes the least square expression in a worse manner ($\xi_{MPR}>\xi_{MPRE}$; $\xi_{PR1}>\xi_{PR1E}$). With a second order expansion, this technique should also provide better results ($\xi_{PR2}>\xi_{PR2E}$), but the generated coupling, between the different stress components would increase the computational time of the recovery (whereas the expected accuracy improvement does not justify it).

One can notice that the differences between the errors provided by the different direct techniques logically decrease when the mesh becomes coarser. Indeed, the number of elements in patches may be insufficient to solve high order recovery equations. The interpolation order is therefore automatically decreased until the P0 technique is used, when necessary.

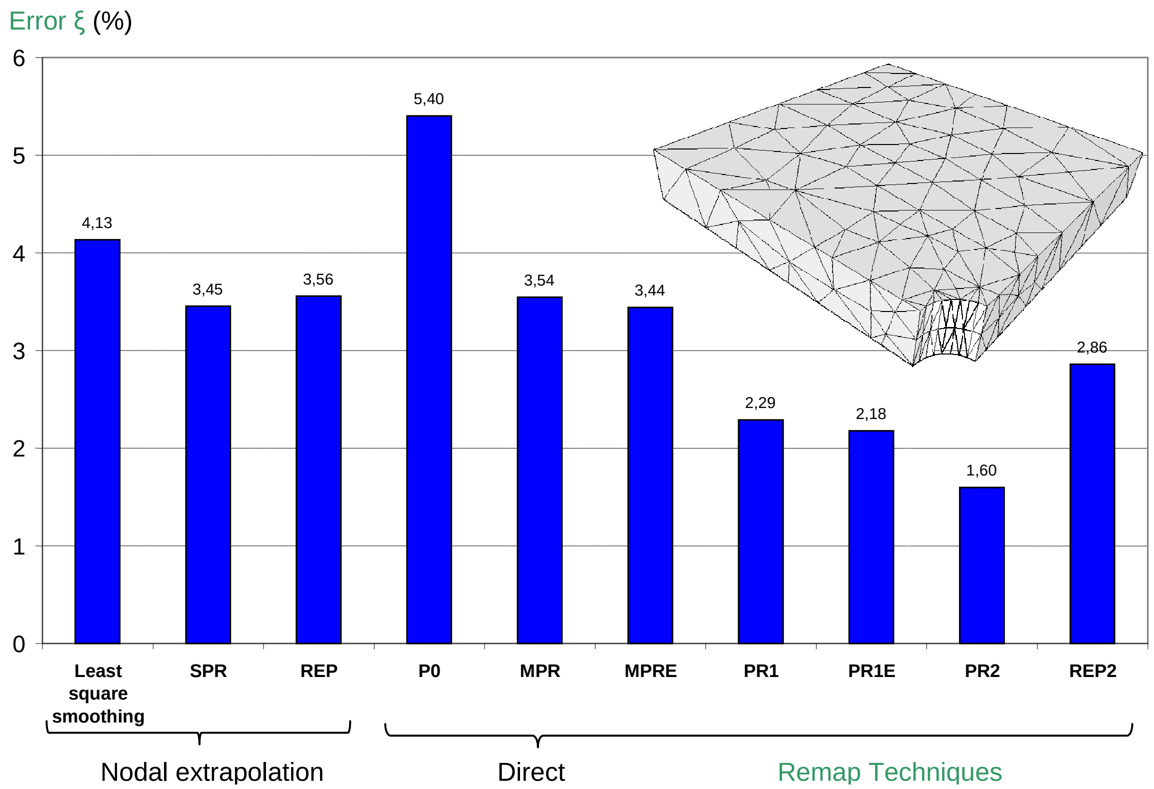


Figure 83: Evaluated error for the different implemented remapping techniques on the coarsest mesh.

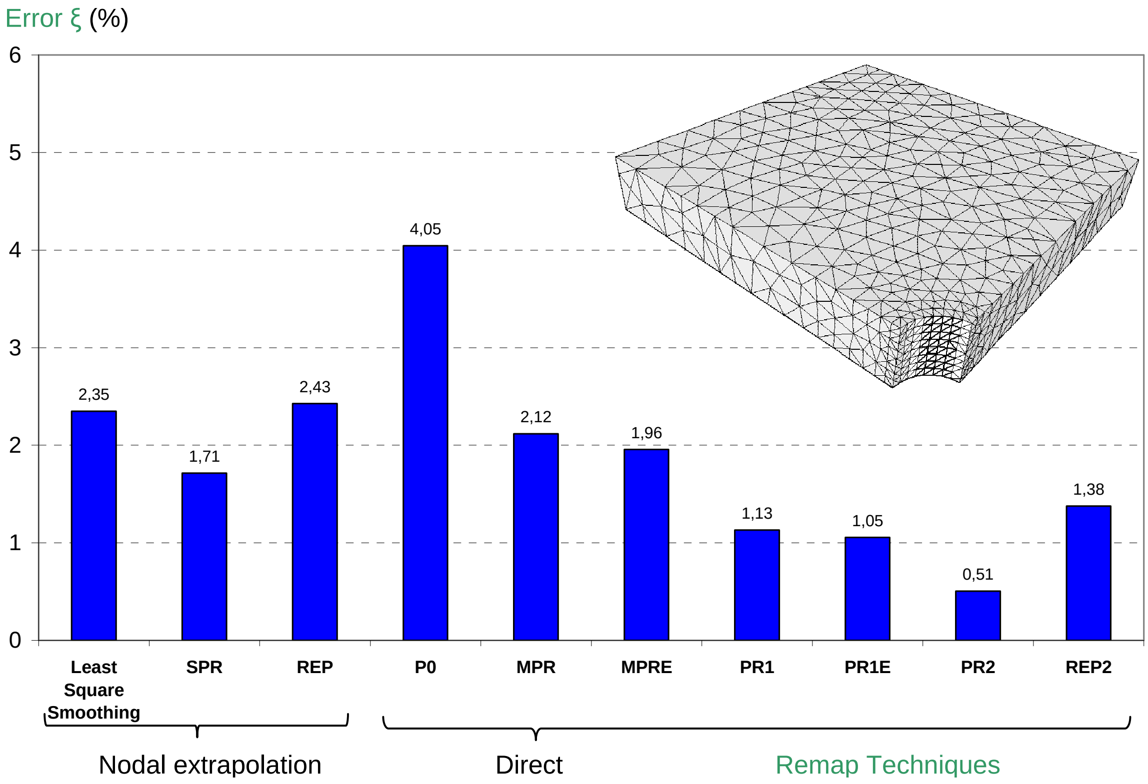


Figure 84: Evaluated error for the different implemented remapping techniques on the medium mesh

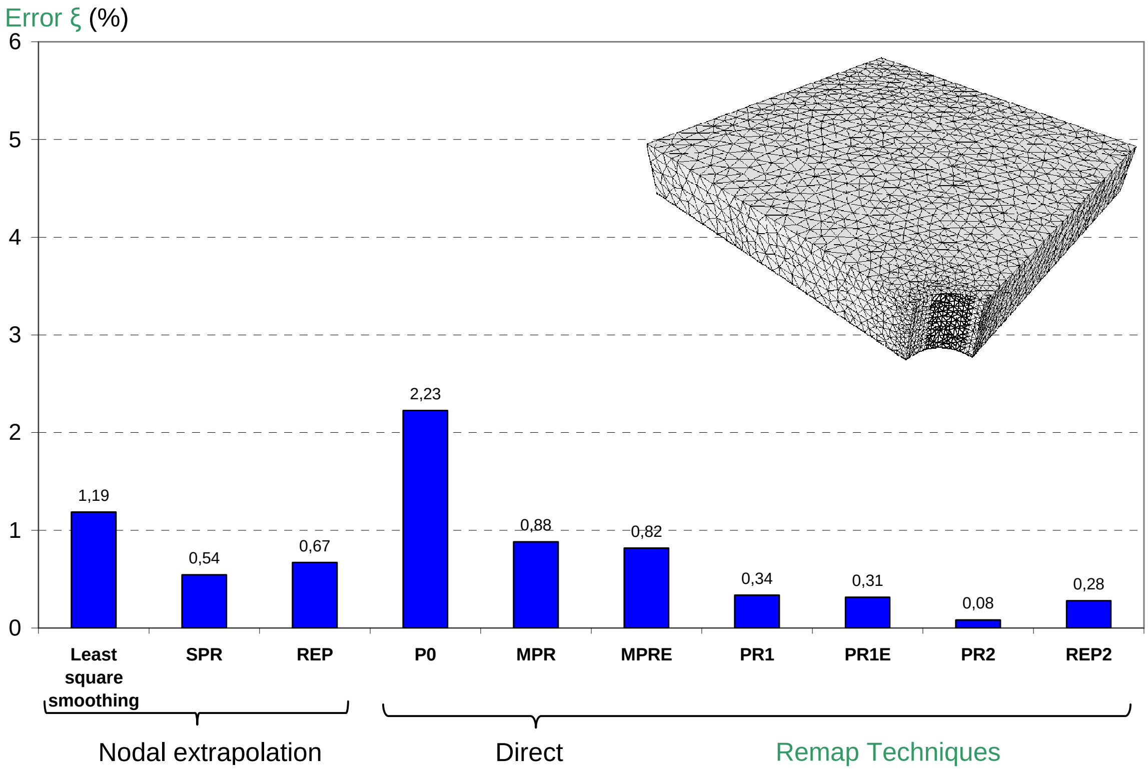


Figure 85: Evaluated error for the different implemented remapping techniques on the more refined mesh

4 *Industrial Application*

4.1 Orthogonal cutting

Adiabatic Shear Band (ASB) is a well known material alteration which takes place at very high speeds with materials having poor thermal conductivity. It results from the competition between plastic hardening and strain softening. The test studied in [28] is issued from Bäker's work [51]. Dimensions are summarized in the following table.

Length (a)	0.28125 mm
Width (b)	0.04 mm
Height (c)	0.15 mm
Cutting depth (d)	40 µm
Cutting angle (e)	10°
Roundness of the tool (f)	3 mm
Cutting speed (g)	50 m/s

Figure 86: Test dimension.

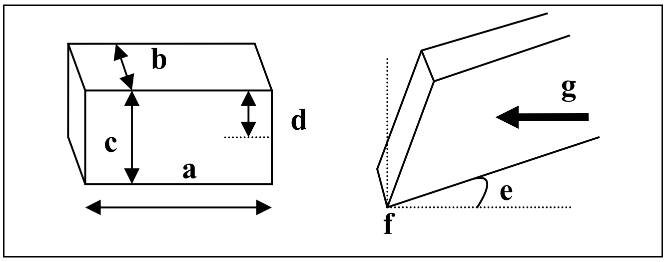


Figure 87: Scheme of the studied process: orthogonal cutting at high speed

The ALE formulation handles problems of mesh distortions at the contact or complex free surfaces (see Figure 88) while the adaptive formulation automatically refines the mesh allowing the formation of adiabatic shearing bands without using any additional damage criterion (see Figure 89).

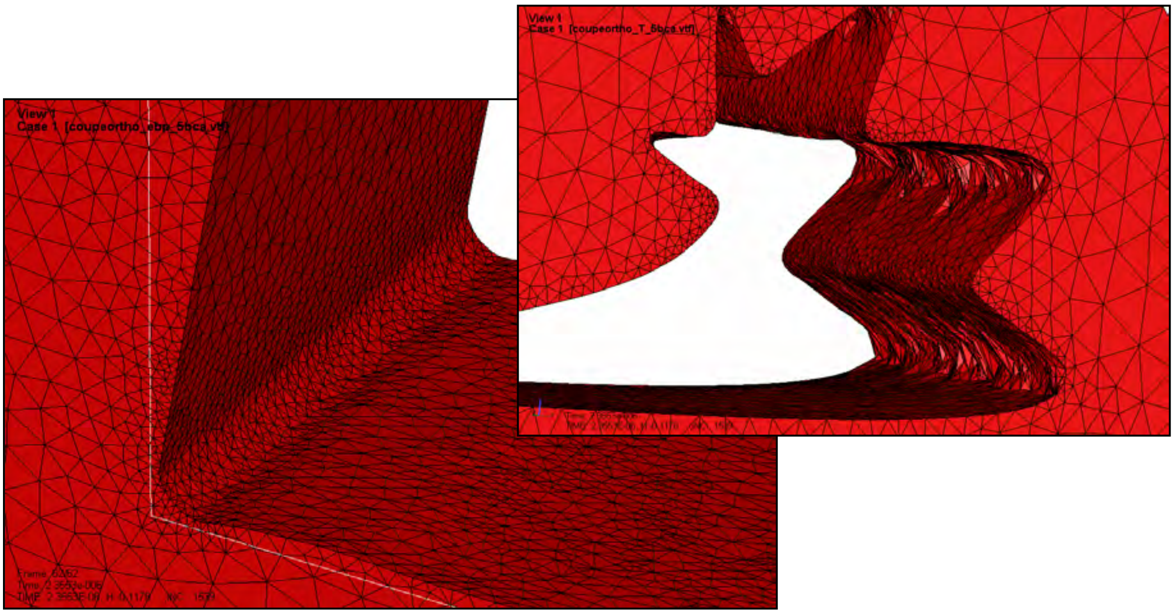


Figure 88: Coupled with smoothing tool procedure, the adaptive ALE formulation handles problems of mesh distortions in the contact area while complex free-surfaces of the segmented chip.

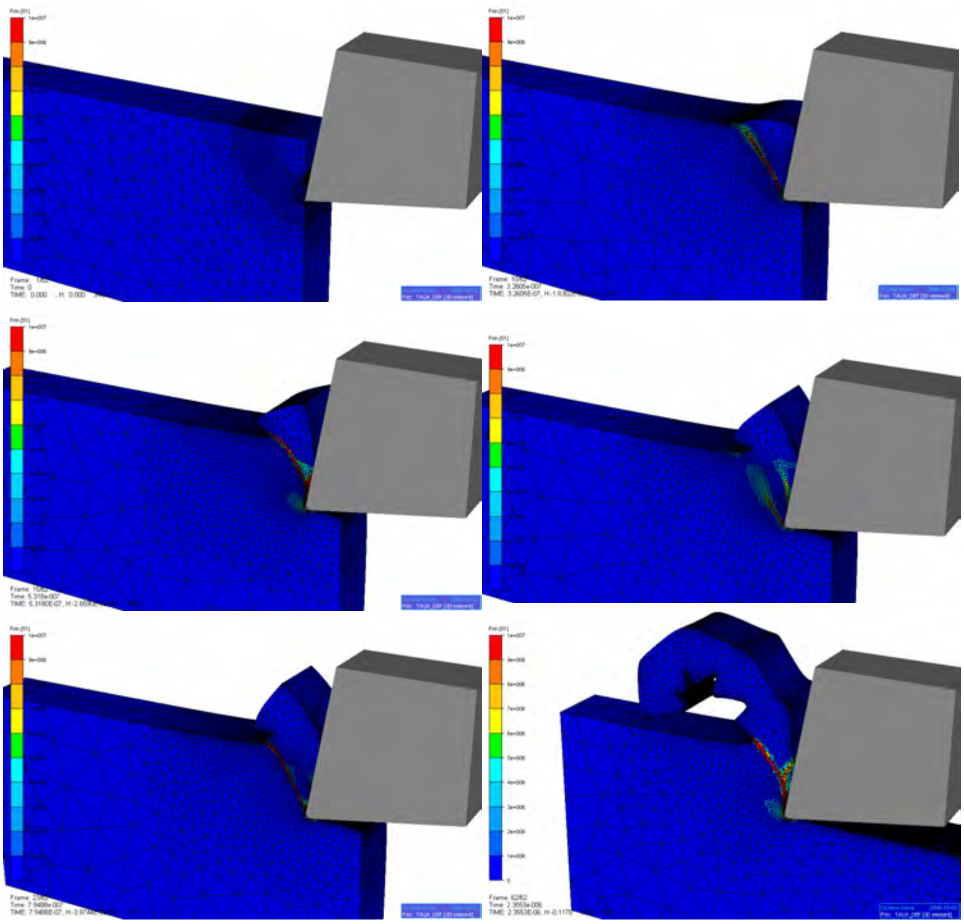


Figure 89: Strain rate in orthogonal cutting simulation using adaptive ALE formulation: the adaptation of the mesh allows the formation of adiabatic shearing bands.

The P1 nodal and P0 element variables are remapped using the developed PR2 technique. Figure 90 shows the strain and temperature field resulting at the end of the simulation, proving the efficiency of the remap techniques.

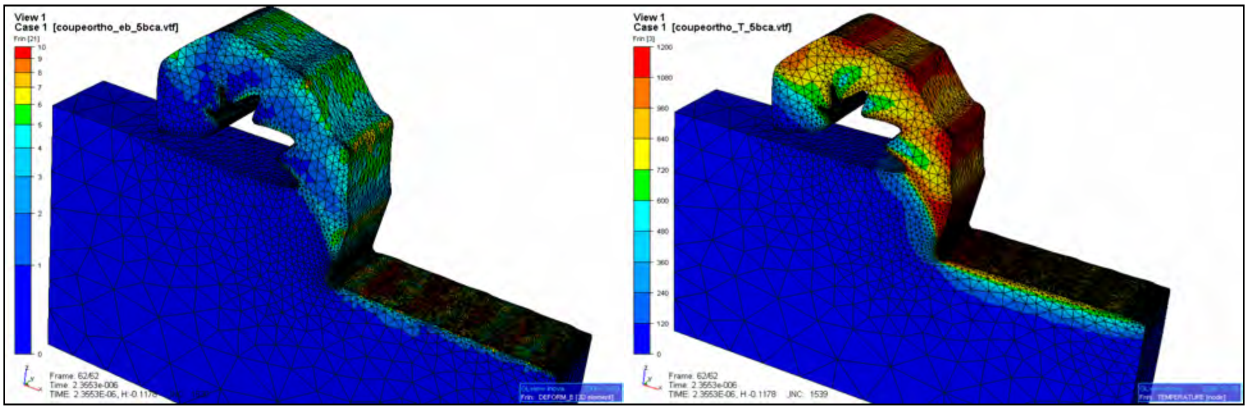


Figure 90: Strain (on left) and temperature field at the end of the orthogonal cutting simulation using adaptive ALE formulation.

4.2 Flat rolling

Flat rolling is another application which has been simulated using the ALE formulation. The difficulty lies in the ability of the formulation to simulate small free surface movements while conserving the edges geometry of the plate even if high rolling speeds are involved.

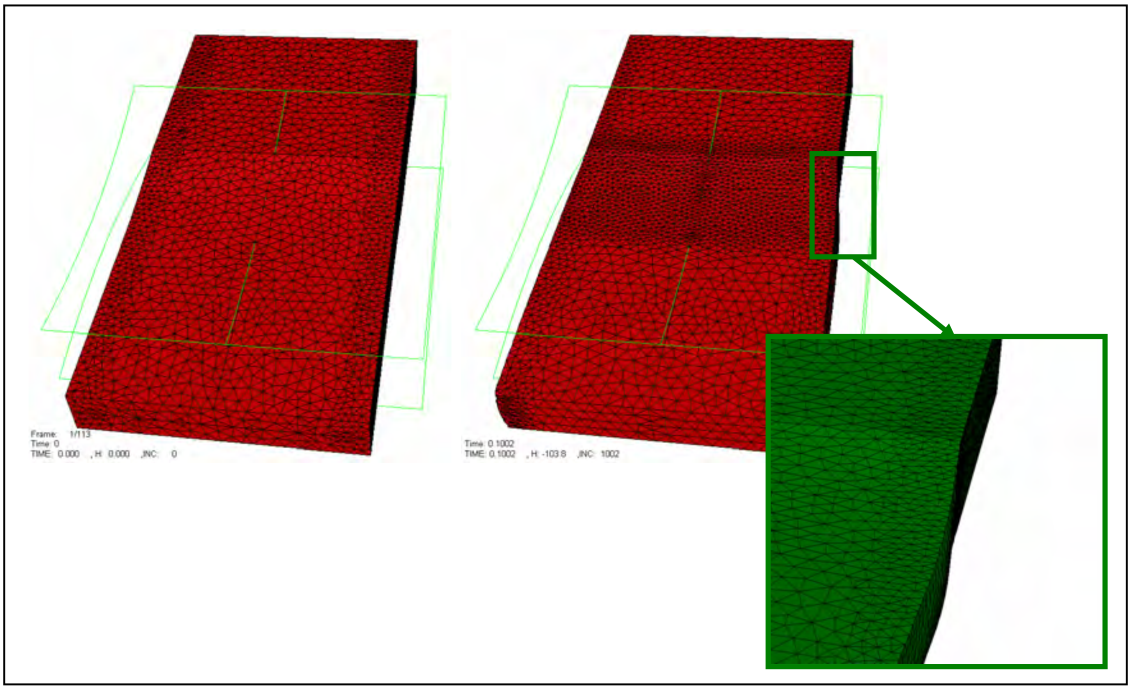


Figure 91: Evolution of the mesh refinement during flat rolling using adaptive ALE formulation: initial mesh on the left side, final mesh on the right side with a zoom on the enlargement of the plate observed in the lateral direction.

Once again, the remapping technique has provided satisfactory results (see Figure 92).

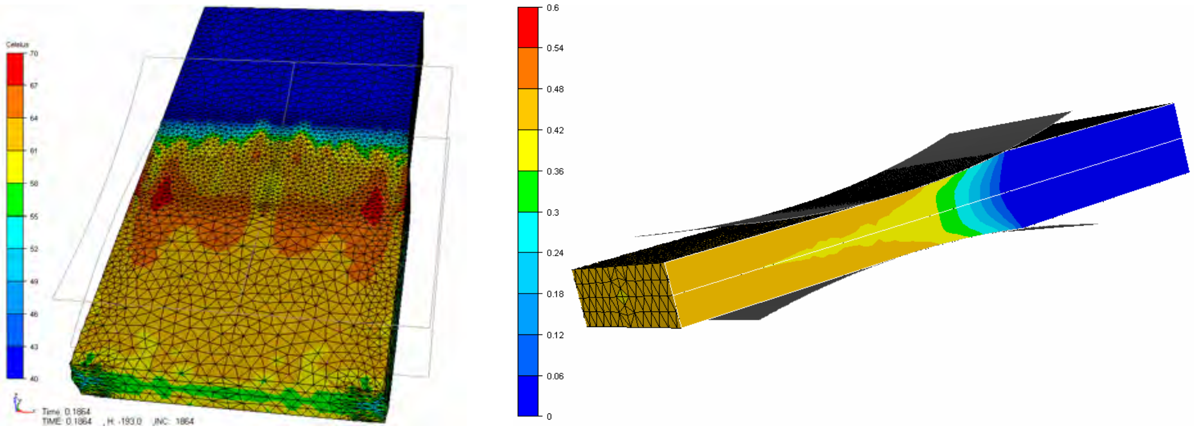


Figure 92: Strain field observed at the end of the simulation (in the central cutting plane on right).

Chapter V : Numerical results and experimental comparisons

The objective of this study is to develop a robust numerical tool, which makes it possible simulating all stages and possible configurations of Friction Sir Processing. Consequently, all presented developments (ALE formulation, contact smoothing…) have been implemented in the Forge3® software and have shown their efficiency and reliability. These enhancements are useful and often indispensable to simulate the different steps of Friction Stir Welding. Along with preliminary investigations to identify some model parameters, this chapter shows the robustness of the code and its ability to simulate different steps and configurations of the process.

After feasibility and robustness, the identification of unknown model parameters is the second step of a code validation process, while the complete validation consists in comparing numerical and experimental results. Infact, the geometry and process parameters, combined to accurate material data and boundary conditions, have to be the only required data to correctly predict the material state variables and flow, at each step of the process. Thus, temperatures, strains, stresses, and void formations should compare to experimental results.

The comparisons presented in this document are part of a first attempt to identify some unknown model parameters, such as the material constitutive and friction laws. More work will still need to be done in order to accurately identify them. In other words, these comparisons are not regarded as fully validating the model; and their preliminary goal is to show the feasibility and robustness of the code.

Most of them rest on experiments carried out in the Brigham Young University (BYU); others are based on observations found in literature. The first part of this chapter is derived from Covington [52, 53] thesis. It describes the experimental equipment utilized at BYU and consequently the source of experimental results. However few of them have been carried out in the specific aim of comparisons with simulation results.

The second part shows the possibility and interest of simulating the steady welding phase with a pure Eulerian formulation. This approach allows calibrating the friction coefficients within a viscoplastic friction law.

Simulations of the transient phases, such as plunging, are presented and discussed in a third part. The software so provides a large range of possibilities to simulate and study FSP. This last part shows that it enables visualizing whether the deposition is successful or not behind the probe (without or with void formation). The material flow can be observed along stream lines and through numerical markers once the simulation has reached a steady state, which can be helpful for process parameter adaptation and tooling design.

1 Experimental equipment

1.1 FSW Machine

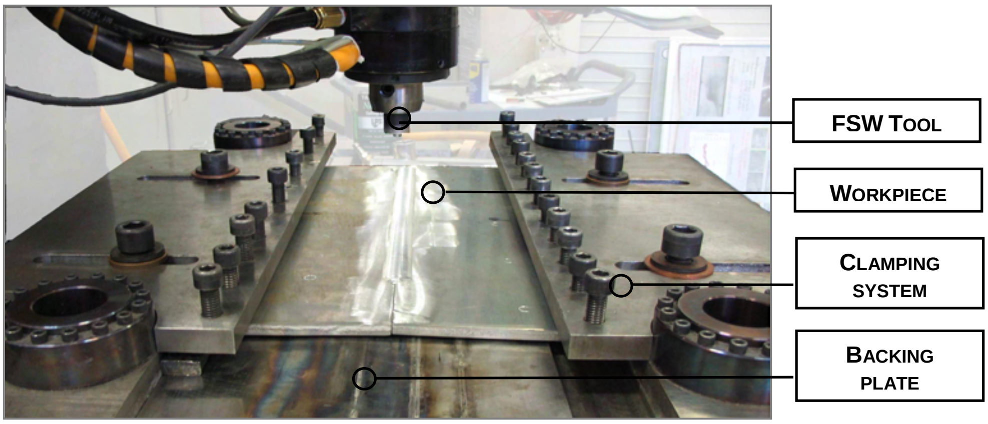


Figure 93: A view of the FSW machine after a typical line welding.

For both plunging and welding experiments, plates were friction stir processed (bead on plate) on a retrofitted Kearney & Trecker knee mill with PLC/PC control and data acquisition system (see Figure 93). The machine is capable of performing welds over 1000 mm (42 in) in length and has a maximum travel speed of approximately 790 mm/min. (31 in/min.). Each axis (x, y, and z (see Figure 94)) is servo-driven and the position and velocity of each axis was monitored and recorded at a frequency of 2 Hz during each weld.

The power required by the 22.4 kW (30 hp) spindle motor as well as all other measured parameters discussed hereafter were also recorded at 2 Hz. The spindle has a maximum speed of 1500 rpm. Z-load and Z-depth control are available and use feedback control to adjust the tool depth during welding such that a constant Z-force or tool depth is maintained. The latter was used in the current experiments.

1.2 Dynamometer

Mounted to the bed of the mill is a 1219 mm (48 in) long dynamometer capable of sensing forces up to 45 kN (10,000 lbf) in both the X- and Y-directions and 90kN (20,000 lbf) in the Z-direction with a resolution of 0.004 kN (1 lbf). The maximum possible workpiece width is approximately 305 mm (12 in). Fixtures for clamping the workpiece are mounted to the upper surface of the dynamometer (see Figure 94).

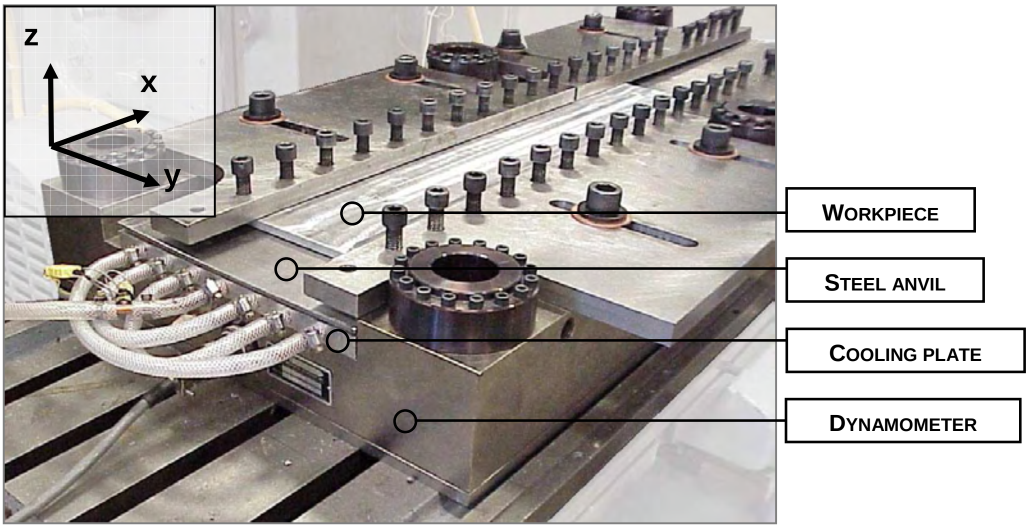


Figure 94: A view of the experimental clamping system, with the cooling plate.

1.3 Anvils

Conditions of the experiment carried out at BYU required the efficient running of multiple welds. As seen in Figure 94, a 15.9 mm (0.625 in) thick liquid-cooled aluminium cooling plate was fabricated to remove thermal energy from the workpiece, anvils, and dynamometer. A mixture of ethylene glycol and distilled water was pumped through the plate from a chiller and entered the plate at approximately 10°C. A 4.76 mm (0.1875 in) thick steel anvil was placed on top of the cooling plate for protection and to give a solid backing surface for the workpiece. As previous work showed that cooling by means of this cooling plate had no significant effect on steady-state results, coolant was allowed to flow continuously through the plate while welding [54].

1.4 Tool Holder/RF Telemetry System

A liquid-cooled tool holder was used to minimize heat flow into the machine head (see Figure 95). The coolant flow rate was approximately 1.9 L/min. (0.5 gal/min.) and was such that while welding there was typically less than 1°C rise in coolant temperature from the inlet of the tool holder to the outlet. Access holes near the top of the tool holder allowed tool thermocouples to be inserted through the back of the tool. A transmitting collar assembly was clamped to the rotating portion of the tool holder and housed RF transmitters which broadcasted the thermocouple readings as FM signals. The signals were captured by the receiver through a stationary loop antenna and transferred to the data acquisition system.

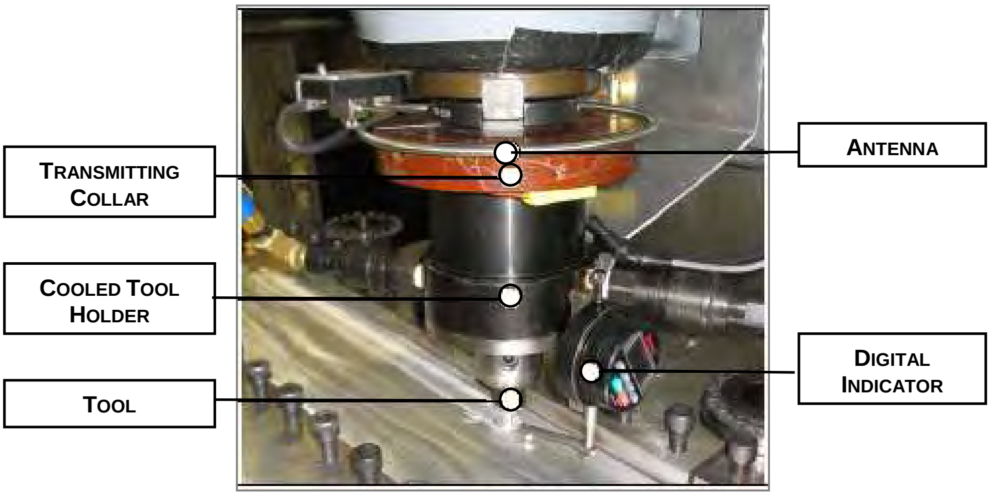


Figure 95: Cooled tool holder and electronic indicator used to measure shoulder depth.

1.5 Electronic Depth Measurement and Control

An electronic digital indicator was mounted to the tool holder for tool depth measurement and to provide an input for the Z-depth control. It has a range of 25.4 mm (1.0 in) and a resolution of 0.002 mm. An extension adapter was connected to the indicator so that weld depth was measured as close to the tool as possible to account for any local changes in tool depth. Readout error associated with attaching such an adapter is estimated to be 0.025 mm or less. The indicator readings are transferred to the data acquisition system throughout the weld. This digital indicator was used to measure the actual plunge and weld depth as seen in Figure 95. Due to machine deflection, the programmed plunge depth will not actually be achieved. Thus, it is important to measure the actual tool depth throughout the weld. The indicator was zeroed when the tip of the pin was in contact with the top of the workpiece.

1.6 Temperature measurement

The tool was modified for internal temperature measurement at three different locations near the tool/workpiece interface. An EDM drill was used to cut long, straight, square-bottomed holes to accommodate 1.6 mm (0.063 in) diameter 304 stainless steel sheathed thermocouples at the locations defined in Figure 5. The thermocouple locations are noted here as Pin Center, Root, and Shoulder. The distance between the end of the thermocouple and tool/workpiece interface at each location was 1.3 mm (0.05 in) or less. Temperature measurement locations were verified by checking hole position, diameter, depth, and shape. Since hole shape can be quite difficult to inspect on a long, deep hole, initial trials were visually inspected by destructively sectioning the tools. Thermocouples slid freely to the desired locations and only a limited amount of oxide, if any, was present in the holes. Although steps were taken to ensure solid contact at each location between the end of the thermocouple and the bottom of the hole, there still exists the possibility of either minimal contact or no contact (small air gap). Despite these possible conditions, the character of the hole as a blackbody cavity ensures measurement of an accurate steady-state tool temperature if these conditions arise.

To ensure that the thermocouple readings accurately reflected the true physics of the process, the FSW tool was modified as little as possible. Calculations showed that the cross-sectional area of the tool body removed to accommodate the thermocouples was less than 2% and the cross-sectional area of the pin removed was approximately 6%.

The thermocouples were manufactured by Omega Engineering, Inc., and were of the ungrounded junction type. The 30 AWG (0.010 in diameter) thermocouple wires run inside the metal sheath and are isolated from each other and the sheath by a magnesium oxide (MgO) powder and are joined separate from the sheath near the probe end. Although this configuration increases the response time of the thermocouple, it electrically isolates the thermocouple junction from the sheath, a characteristic required when using multiple thermocouples in the RF telemetry system. The time constant for the thermocouples when measuring the temperature of steels proved to be longer than that quoted by the manufacturer in water (0.25 sec.), suggesting that measurements taken during the transient phase of FSW may have some temporal lag. However, the measured time constant of 4 sec. proved that for steady-state regions the temperature measurements would be accurate.

1.7 Weld Process Data

Plate is affixed at a predetermined location on the anvil and clamped into place. The tool is then positioned directly over the plunge location and the pin is brought into contact with the top surface of the workpiece. The tool position is then zeroed. Contact between the pin and workpiece is known by a registered force of 44 N (10 lbf) or less on the dynamometer. Weld parameters are then adjusted to the desired values.

Although the FSW equipment automatically records various values as weld outputs, parameters of interest in the current studies were X-force, Z-force, Pin Center Temperature, Root Temperature, Shoulder Temperature, Shoulder Depth (to insure that control was sufficiently accurate), and Motor Power. Motor Power refers to the amount of power required by the welder to turn the spindle under load, which includes the power required to overcome frictional losses.

Figure 96 shows an example of typical registered data from a FSW run.

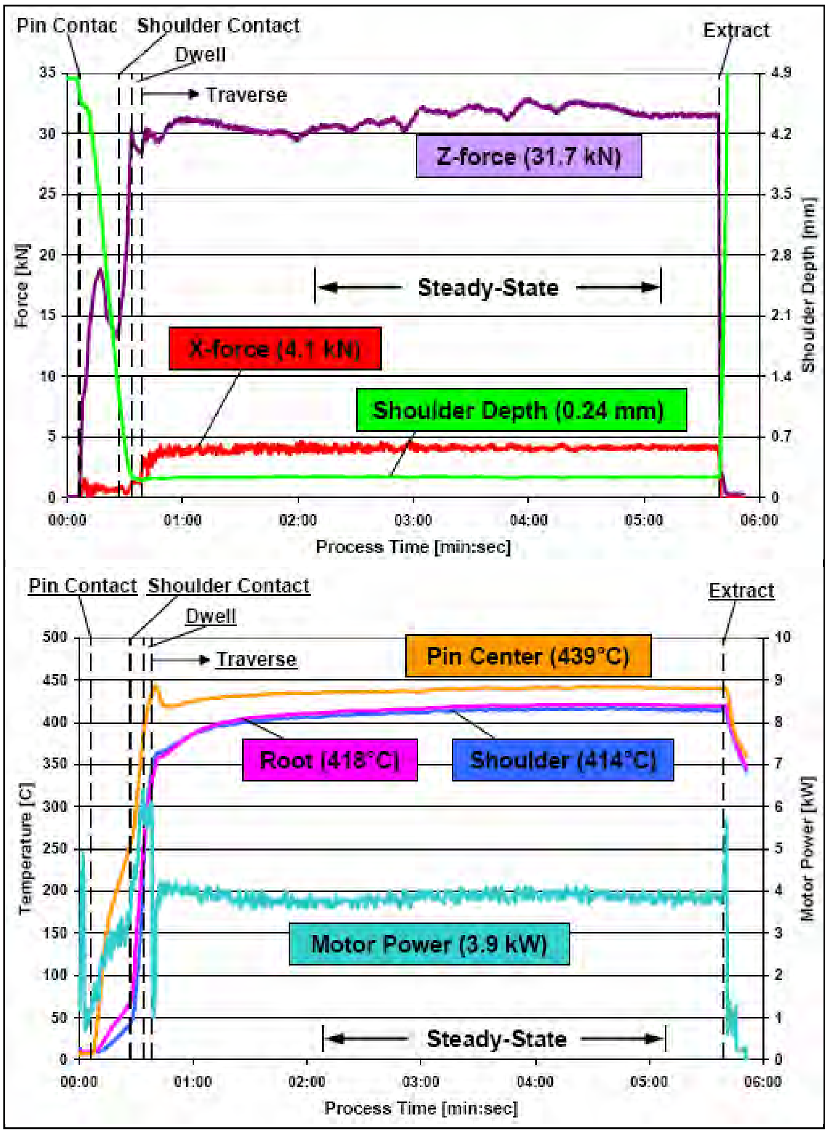


Figure 96: Typical data provided by the experimental equipment during one welding run.

2 Welding phase

2.1 Experiment description

In this first study, we consider experimental processing tests. They have been conducted at the Brigham Young University by Sean Darby. The original goal of the experiment, which is described thereafter, was the study of pin length influence (mainly on tool temperature). It has provided experimental welding data that have been utilized to carry out some comparisons with numerical simulations during the stationary welding phase.

Figure 97: Photos of some experimental welding runs, and some tool utilized to study the pin length influence.

The experimental design consisted in four runs per combination of one pin geometry and one welding speed. 14 different lengths of pin (threaded and unthreaded), from 1.8mm (0.071in) to 11.43mm (0.43in), have been tested at 3 welding speed: 1.69mm/s (4ipm), 3.39mm/s (8ipm), and 5.08mm/s (12ipm). The medium values of the parameters has been used for the first comparisons with numerical simulation: the transverse speed is 3.39mm/s (8ipm) and the pin is 6.35mm(0,25in) length and unthreaded.

The four welds (per combination) were runs on the same plate of AL 6061 material. With a thickness of 9.53 mm (0.375 in), the plates were sheared to nominal dimensions of 279 mm (11 in) x 457 mm (18 in). For each welding run, the programmed plunge depth was predetermined so that the tool depth near the end of the plunge was approximately equal to that needed during the weld. The tool was then allowed to dwell for 1 sec. The spindle speed during each dwell and welding run remained at 650 rpm. The depth of the tool into the plate is manually adjusted during the welding phase in order to maintain a quasi-constant axial force on the tool.

The length of each weld was about 406 mm (16 in). Since steady-state average values were desired, it was necessary that the steady-state region for each weld be sufficient in length. In order to minimize thermal or mechanical interaction with the clamping system, each weld was spaced of a minimum of 37 mm from the others and from the clamps. The oxide layer was removed with a portable disc sander and the surface was cleaned with methanol prior to processing. The thickness of the plate was predetermined so that the penetration of the pin into the plate was only partial, eliminating any possible interaction between the tool and the anvil. Thermal and process data were recorded at 50 milliseconds intervals.

The tool used for this study was manufactured from heat-treated H13 tool steel, its dimensions consisted of a shoulder diameter of 25.4 mm (1.0 in), body length (from the top of the tool to the shoulder) of 83.8 mm (3.3 in), shoulder concavity angle of 8 degrees (see Figure 98). Three thermocouples are located in tool as in the next plunge study (see Figure 117). The tool is unthreaded and was used at a tilt angle of 2.5 degrees.

2.2 Modelling

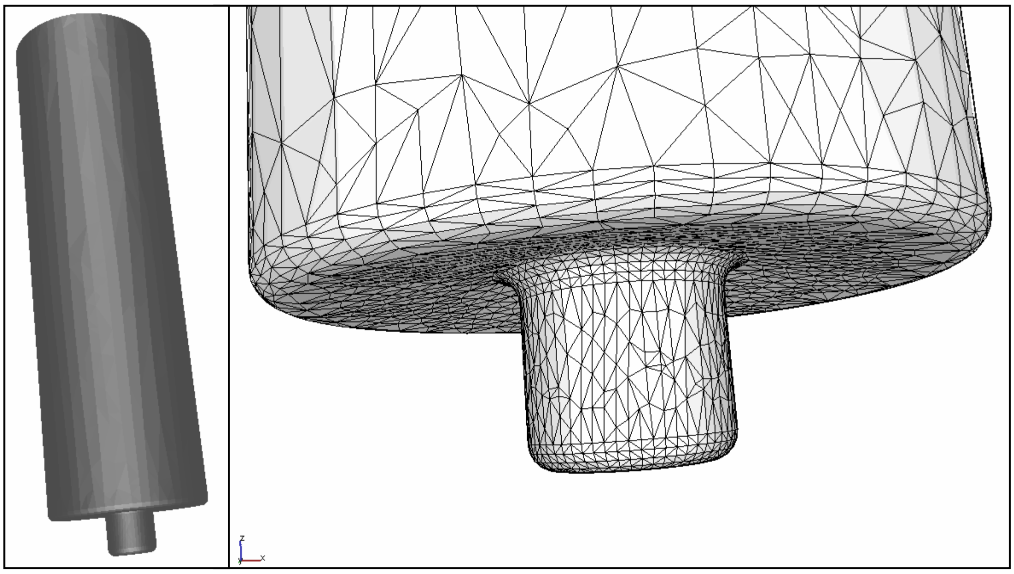


Figure 98: Global view of the tool and mesh visualisation.

The tool has been modelled according to its actual geometry and is tilted of 2.5 degrees. Figure 98 shows the discretized tool.

As the experimental plate is very large, it is modelled by a smaller plate, which is 150 mm wide, 300 mm long, and 9.53 mm thick. This latter dimension is the only one which corresponds to the experimental value. It is assumed that the distance from the tool to the edges is long enough to not modify the thermal field when the steady state is reached.

The backing plate is also modelled. It has the same dimensions as the workpiece but a thickness of 25 mm which corresponds to the actual thickness of the cooling plate.

The initially prescribed plunge depth is about 1.5 mm. It corresponds to the average of the shoulder penetrations for the four experimental welds. A preliminary simulation of the plunging phase would result in a better initial temperature map (see section 3.1), and would provide a good approximation of the free surface shapes at the beginning of the welding phase. However these informations are not necessary to compute the steady state of the welding phase. In fact, a numerical preliminary study has shown that the initial temperature field does not influence the final computed steady state.

The process parameters are taken from the experiment. The transverse velocity of 3.39 mm/s is prescribed as an inflow velocity at the nodes of the upwind side of the workpiece (see

Figure 99). The tool and the backing plate have a rigid body movement (so neglecting their elastic deflections and plastic wears). They are discretized by a volume mesh and a coupled thermal computation is possible within. An Eulerian formulation is used to speed up the calculation, so the specific procedure previously detailed and illustrated in Figure 18 is used to transport the computed temperature. About 32,000 elements (7,500 nodes) have been utilized to discretize the FSW tool and 25,000 (6,000 nodes) for the backing plate. Eulerian sensors have been introduced at the positions of the thermocouples in the experimental tool (see Figure 117).

A 3D thermomechanical computation is carried out in the workpiece only, while the temperature is calculated in the three considered bodies in a coupled manner. The initial number of elements (resulting from a preliminary plunge simulation) is about 50,000 (10,000 nodes) with minimal sizes of about 0.5mm.

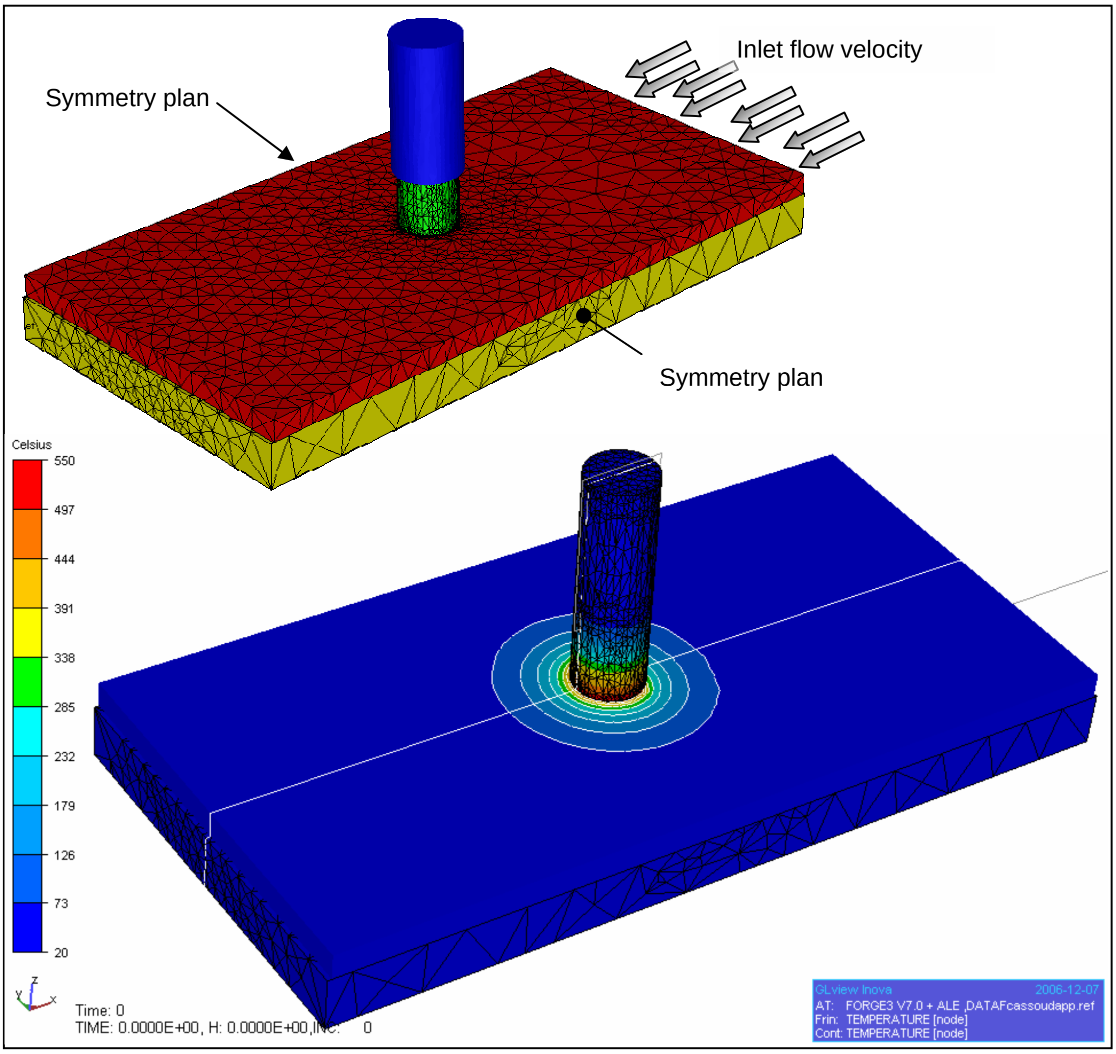


Figure 99: Global view of the body modelisation and initial temperature field (resulting from a preliminary plunge simulation).

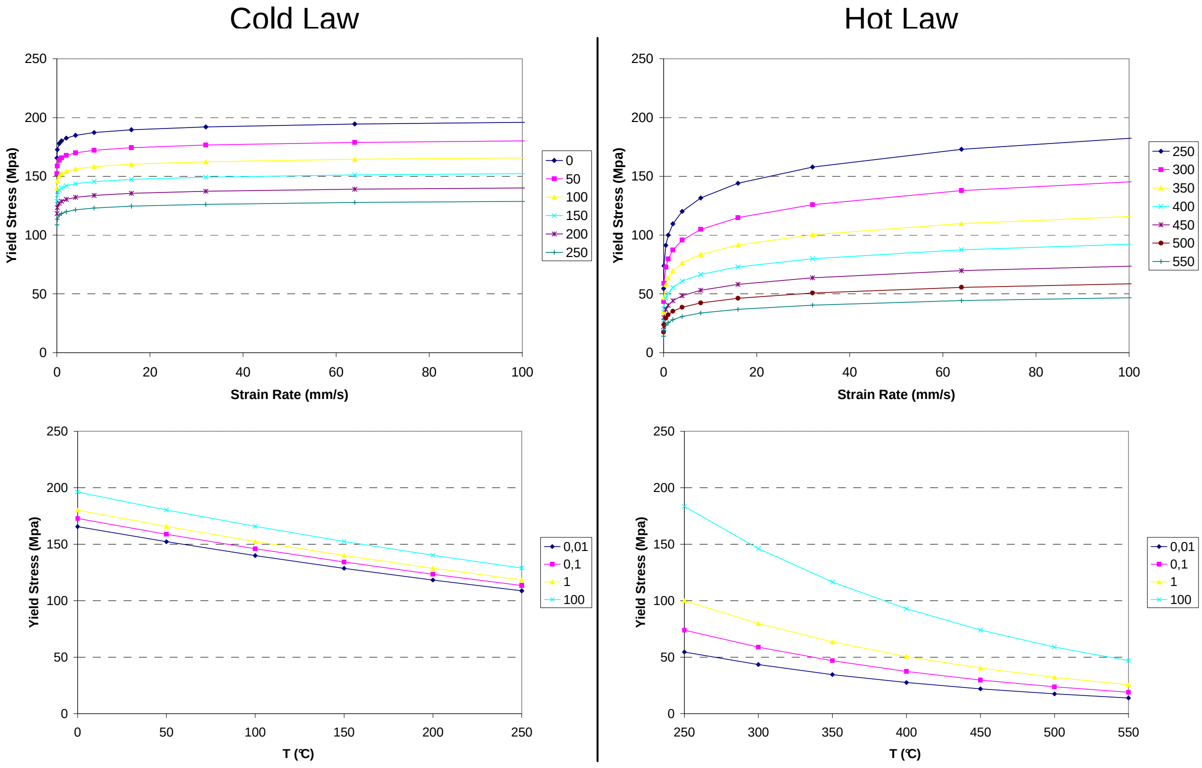
The Forge3® material data-base provides two different Hansel-Spittel laws to define the constitutive behaviour of the Al 6061 material. The first one is an Elasto-viscoplastic law, which is adapted to a temperature range of 20°C to 250°C, and the second one is a pure viscoplastic law, which is adapted to a temperature range of 250°C to 550°C. Figure 100 shows their coefficients, and the evolution of the resulting equivalent stress for a constant strain of 0.1%. These curves also show that the model does not perfectly join.

The solidus of Al 6061 is around 582°C and its liquidus around 652°C. During the process, the temperature of the material can locally reach the melting point, which exceeds the validity limit of the hot Hansel-Spitel law. On the other hand, the constitutive behaviour of the material can change rapidly in such thermally affected zones, which is difficult to model with such a simple analytical model. The Young Modulus is taken equal to 73 GPa and the Poisson’s coefficient equal to 0.3. In order to obtain a single constitutive model for the whole range of temperatures, the material consistency and the strain rate sensitivity have been tabulated as a function of temperature. The resulting constitutive law is presented in Figure 101, and provides a first approximation of the constitutive model of Al6061, which already allows obtaining interesting results.

Hansel-Spittel Coefficients for AlMgSi0,6Cr(DIN)--**6061**(US)

$$\sigma_f = A_1 \times e^{m_1 \times T} \times \bar{\varepsilon}^{m_2} \times \dot{\bar{\varepsilon}}^{m_3} \times e^{\frac{m_4}{\bar{\varepsilon}}}$$

Temperature range	**20 - 250**	**250 - 550**
Strain range	**0.04 - 3**	**0.04 – 1.5**
Strain rate range	**0 - 500**	**0 - 300**
A_1	260.49451	352.3876047
m_1	-0.00168	-0.00454
m_2	0.16992	0.06604
m_3	0.0184	0.13165
m_4	-0.00073	0.00241



Constant equivalent strain : $\bar{\varepsilon} = 0.1\%$

Figure 100: Table of coefficients of the Hansel-Spittel constitutive laws of Al 6061;
equivalent stress versus temperature and strain rate for hot and cold constitutive laws (at different strain rates / temperatures).

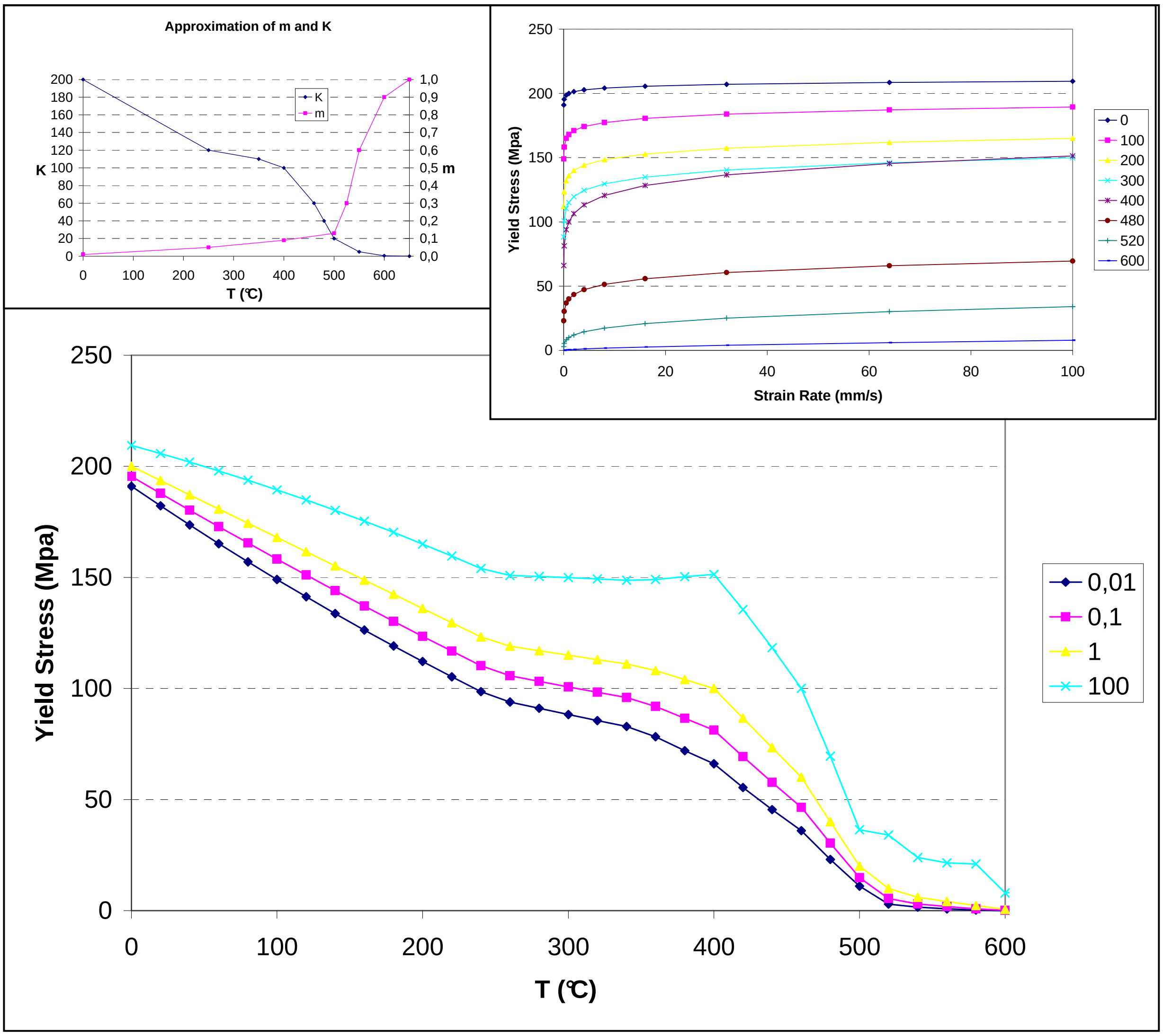


Figure 101: Yield stress visualization for the tabulated values of the material consistency K and its strain rate sensitivity m.

The thermal characteristics of the Al 6061 workpiece, H13 tool, and backing plate are assumed constant. They are summarized in table 102.

	Conductivity λ (W / m.℃)	Thermal capacity c (J / kg.℃)	Density ρ (kg/m3)	Effusivity $E = \sqrt{\lambda\rho c}$ (J/kg.m^3)(m^{-2}/s)	Emissivity
Workpiece Al 6061	180	896	2700	20868	0.05
FSW Tool H13	24,3	460	7850	9367	0.88
Backing plate	250	1230	2800	29343	0.05

Figure 102: Thermal characteristics of the modelled materials

2.2.1 Boundary conditions

2.2.1.1 Thermal

As already explained, the tool quickly reaches a steady-state temperature during the welding phase. This temperature is close to the plate temperature, which results in a poor thermal exchange between the tool and the workpiece.

The different utilized thermal exchange coefficients and the prescribed temperatures for bodies are assumed constant. They are summarized in the following table 103.

Thermal Exchange Coefficients h_c between ↓ and →	FSW Tool H13	Anvil Backing plate	Tool Holder 15℃	Rigid Cooling Plate 15 ℃	Ambiant Air 20 ℃
Workpiece Al 7075	50000	2000	---	---	30
FSW Tool H13	---	---	20000	---	20
Anvil Backing plate	---	---	---	2000	30

Figure 103: Thermal boundary conditions of the model

The temperature at the nodes where the inlet flow is prescribed is considered as constant and equal to 20°C.

2.2.1.2 Mechanical

In order to model the clamping system, two symmetry planes are applied on lateral sides of the workpiece. As already mentioned, the transverse velocity of 3.39mm/s is prescribed as an inflow velocity at the nodes of the upwind side of the workpiece (see Figure 99).The tool and the tool holder have a rigid body movement. The contact features between the tool and the workpiece actually depend on the process conditions, so they are a result of the model: the friction law determines the shear stress at the contact interface between the tool and the workpiece, so any point of the workpiece can get in or out of the contact surface depending on the solution of the thermo-mechanical computations.

However, in a first approximation, a pure Eulerian description is used to identify the friction coefficients α_f and p during the steady state welding. So assuming that the contact surface does not depend on the values of the friction coefficients (which is quite acceptable according to small tested variations of α_f and p).

2.3 Experimental results

Figure 104 shows reference recorded data during one of the four welding run, which have been realized with the process parameters further used in the simulation.

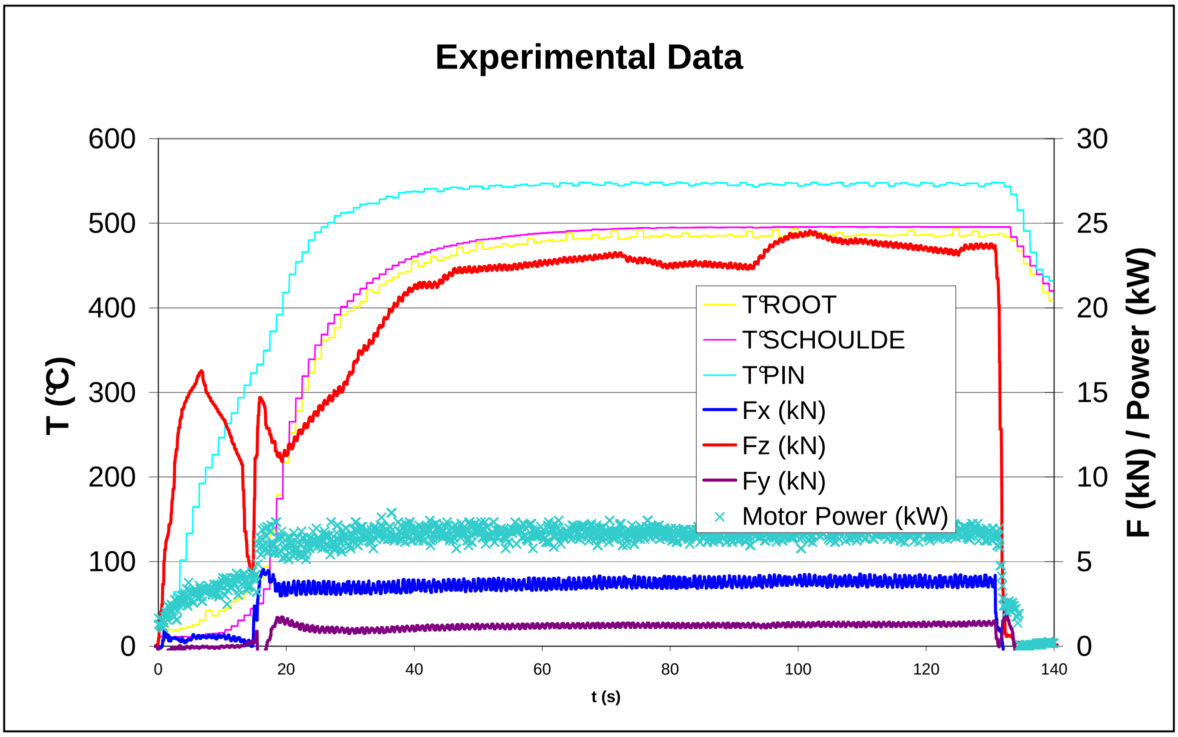


Figure 104: Experimental data recorded for one experimental weld test.

In this figure, the different phases (plunging, dwelling, welding) can be differentiated by the typical evolution of the recorded data. A steady state seems to be reached (in this run) for t = 60 s. The little variations of the registered data that are observed around 90 s are due to the manual adjustment of the plunge depth during the weld. This adjustment provides some possible changes between two runs. Figure 105 shows that the differences can be significant. Therefore the values provided by the four experimental runs are averaged, which provides results shown in the following table:

Fx	Fz	Motor Power	T°Pin	T°root	T°shoulder
3,88 kN	24,8 kN	6,88 kW	547°C	487°C	496°C

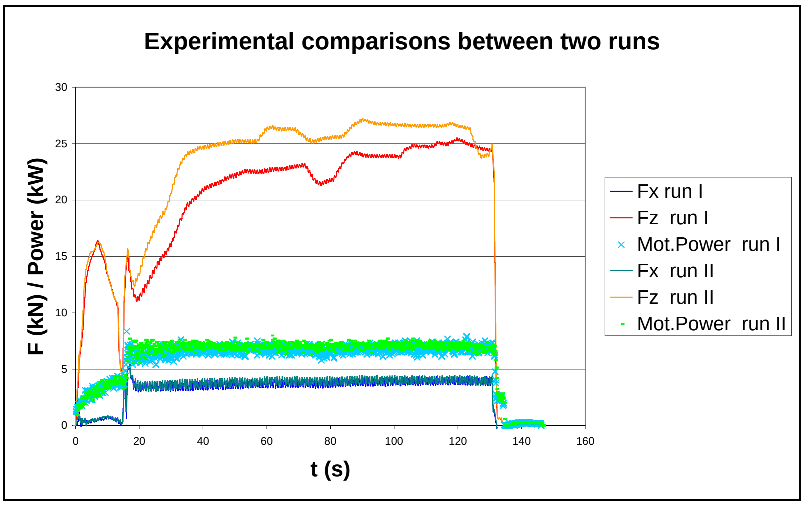


Figure 105: Comparison of forces and torque recorded for two experimental runs processed with the “same” parameters (experimental dispersion).

2.4 Stationary State and Eulerian Simulation

The easiest and fastest way to simulate the stationary welding phase (or steady welding state) is to use an Eulerian description. On the one hand, this formulation may provide less accurate results than the ALE formulation, because of the approximation done on mesh surface, but on the other hand, it is a robust formulation to approximate the steady state (illustrated in Figure 106) and identify some model parameters. As already mentioned above, the fre surface shape is calculated by an ALE simulation of the plunging step, which so significantly reduces the approximation made.

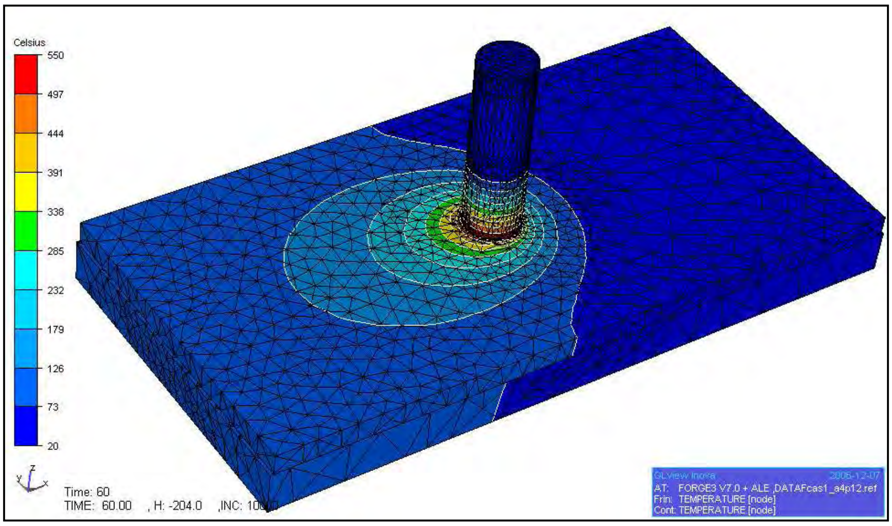


Figure 106: Illustration of the global temperature map obtained at steady state (after 60s of welding).

It should be emphasized that the modelling of friction is always a very tough problem in any metal/material forming process. In particular here in FSW, according to the very high tangential velocities and temperature rises, it is almost impossible to derive a simple representative friction test. Consequently, the process itself is used for the friction parameters identification by calibration.

2.4.1 First approach for friction coefficient identification

Three simulations have been carried out with different values of the friction coefficients. With the utilized viscoplastic friction law, two coefficients (the friction coefficient α_f and the sensitivity to sliding velocity p) are involved in the shear stress calculations (see equation (II-52) recalled in Figure 108). Other numerical calculations and comparisons with experimental results (see section 3 of this chapter in particular) have shown that the Norton (or viscoplastic) friction model provides the most realistic temperature fields with respect to the other simple available laws (among which the Coulomb one). Two values of α_f and two values of p have been

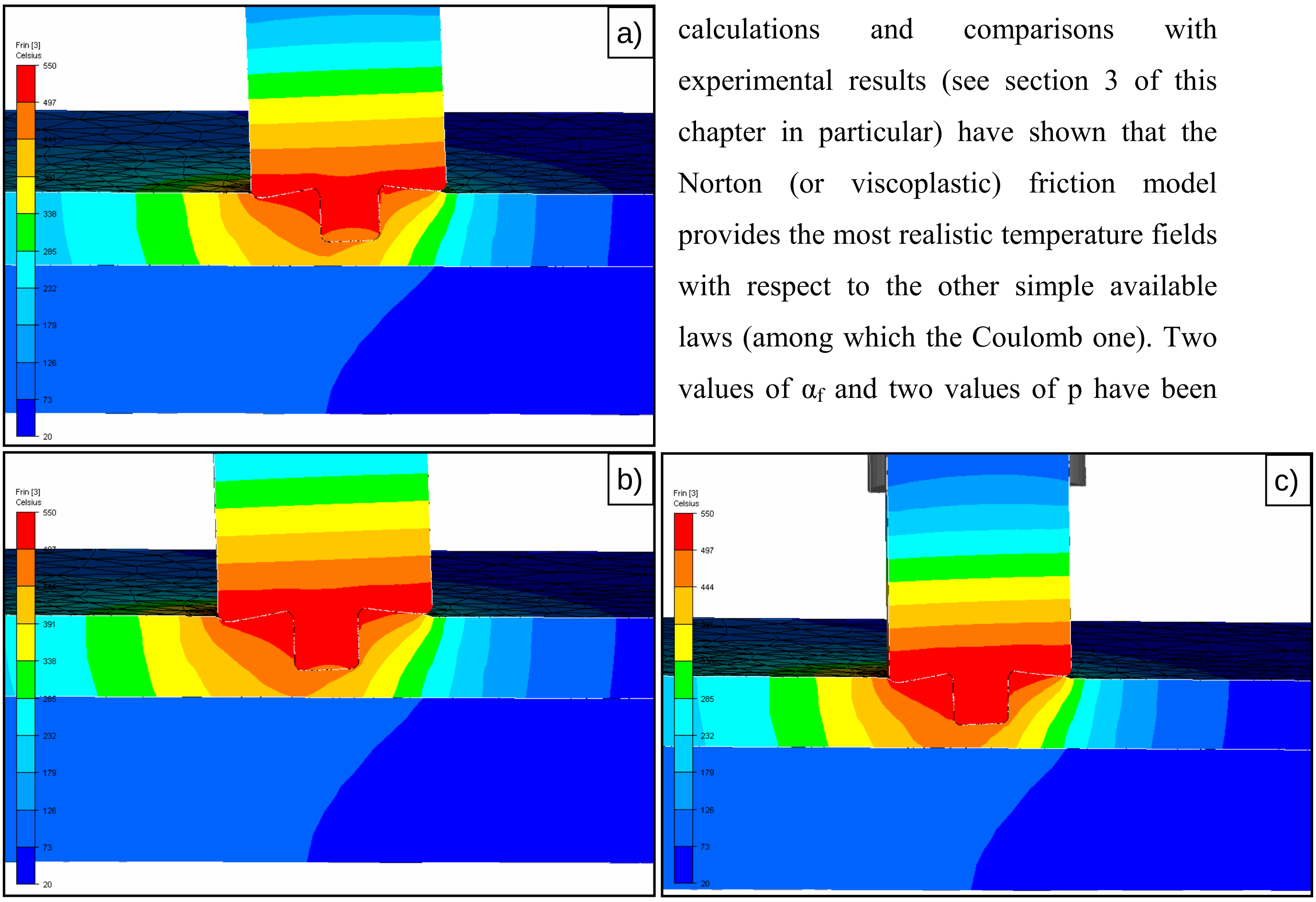


Figure 107: Cutting joint plane view of temperature maps of the 6061 aluminum plate after 60 s of welding (steady state conditions), the viscoplastic friction law is used with various coefficients:
a) α_f=0.3 and p=0.125
b) α_f=0.4 and p=0.100
c) α_f=0.4 and p=0.125

tested, providing the set of the three following combinations:

- case 1: α_f=0.3 and p=0.125
- case 2: α_f=0.4 and p=0.100
- case 3: α_f=0.4 and p=0.125

Changing the friction coefficients, while holding all other parameters constant, shows the relative contribution of friction to the heating of material around the tool. Figure 107 shows the temperature field obtained in the cross joint plane after 60 s of welding for the three different couples of friction parameters. Most of the material located under the shoulder and around the pin is between 500°C and 550°C. The effect of increasing the level of friction is significant in the generated heat.

In a first approach, it is preferred to calibrate these coefficients according to the measured forces. Figure 108 first shows the sensitivity of both the Fz (in the axial direction) and Fx (in the transverse direction) forces to friction. The obtained values are very different with the different parameters combinations. It so allows carrying out a satisfactory parameter calibration. Figure 108 then shows that a very good agreement is obtained between experimental measurements (in green) and simulation for case 3 (blue and red curves), with α_f=0.4 and p=0.125. Already shown in Figure 104, only the steady part (between 60 and 120s) of the experimental data is compared.

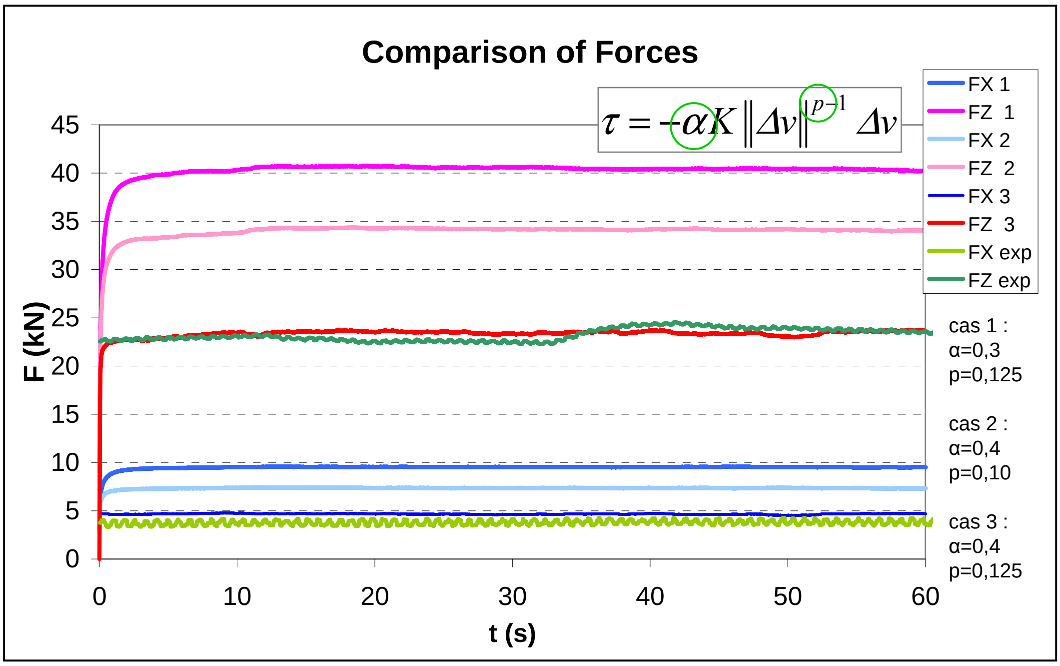


Figure 108: Comparison of simulated forces, for three different couples of friction parameters, with experimental data.

In the simulation a steady state is reached more quickly than in the experiment. This is mainly due to the fact that a preliminary simulation of 60 s had been run in order to obtain a common approximated quasi-steady temperature field as initial state for each of the three simulation cases. In the preliminary simulation, the friction (α_f =0.4 ; p=0.15) was too strong for this first preliminary state.

It may seem also quite natural to identify the friction parameter by comparing torques. However, in this experiment only the Motor Power was recorded. This is the amount of power required by the welder to turn the spindle under the load, which includes the power required to overcome frictional losses.

In a separate experiment, the free-wheel power required by the spindle motor at various spindle speeds was measured (see Table 2) to approximate the motor losses.

Spindle Speed [rpm]	Losses [kW]	Spindle Speed [rpm]	Losses [kW]
200	0.25	550	1.22
250	0.34	600	1.37
300	0.43	650	1.52
350	0.53	700	1.58
400	0.69	705	1.62
450	0.88	750	1.65
500	1.04		

Figure 109: Approximate losses of the FSW machinery at various spindle speed levels.

The value of 1,52kW was then subtracted from the steady-state averaged Motor Power values of the welding experiments. The resulting value is compared in Figure 110 to the simulated values for the three frictional cases. It shows that the equivalent power, which is the product of the axial torque with the spindle speed, is much less sensitive to the variations of friction than the forces, so that it is not a proper value for parameter identification. Moreover, the values of the equivalent motor power are very noisy, much more than the measured forces or temperatures.

In spite of the introduced correction, the measured values are much larger than the calculated ones. It is likely that the losses which occur while welding are larger than during a free-wheel state because of the increased strain on the gears and bearings. It is recognized that the free-wheel power measurements are a lower limit of these losses. Therefore the simulated values provide the good order of magnitude and the difference between the values has the right sign.

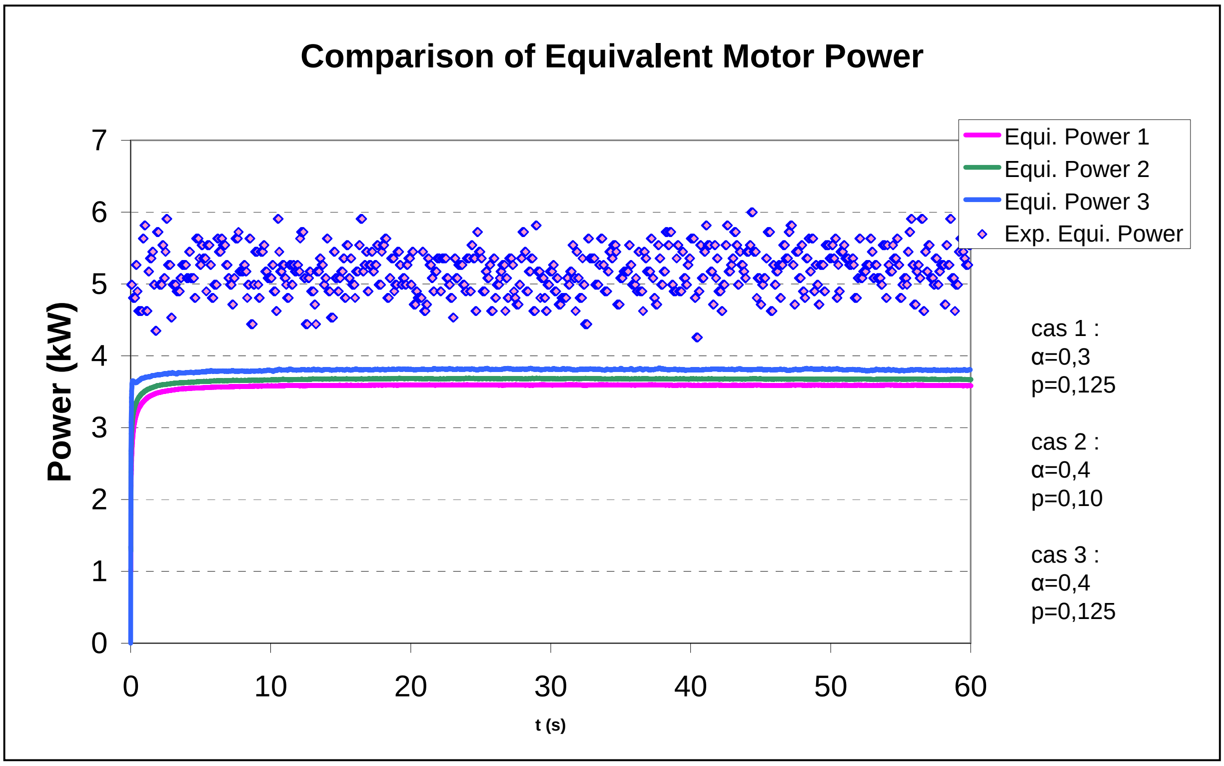


Figure 110: Comparison of simulated torques for three different couples of friction parameters, with experimental data

On the other hand, the simulated temperatures seem to be a little over-estimated. In fact, a large amount of material, located under the shoulder and in the pin area, reaches temperatures which go up to 550°C, which does not sound correct. Further more, the calculated temperature of the FSW tool are not very consistent with experimental measurements. Figure 111 actually shows that the maximum temperature is measured with the thermocouple located at the tip of the pin, whereas the simulation provides the highest temperature prediction at the shoulder sensor.

However, the simulated values belong to an acceptable range of temperatures. The difference may be mainly due to inadequate boundary conditions applied to the tool. In fact, the FSW tool is experimentally cooled down by liquid circulation. This boundary condition has been approximated by the contact with an external body at constant temperature. However the thermal exchange coefficients, like conductivity, of the H13 material are regarded as constant, which may be a too rough approximation.

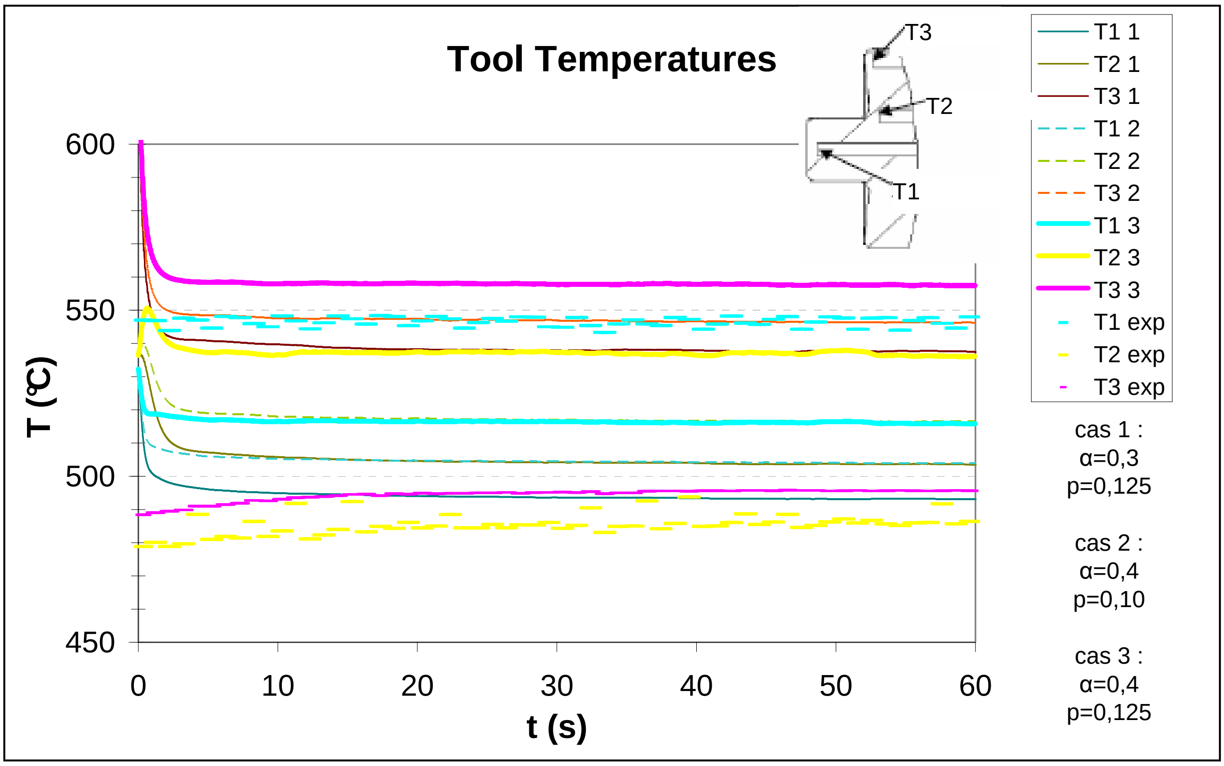


Figure 111: Comparison of simulated temperatures with experimental measurements, for the three couples of friction parameters.

In order to validate the identified friction parameters, ALE simulations have been performed. They have shown and confirmed that the deposition process occurs correctly with these process parameters and friction parameters. A little variation of the plunge force is nevertheless observed, which is due to the modification of the contact area with the shoulder. This area is also strongly dependant on the plunge depth of the shoulder into the welded material. Section 3.1.8 discusses the difficulties it can generate for the simulation.

Other validating simulations have to be run in order to assess or improve the accuracy of the model. Comparisons could be done with other process parameters, such as the welding speeds. Figure 112 compares the calculated forces for 2 different welding speeds. It is noticed that in the lower speed case, the calculations over predict a little the forces. So, further investigations are to be done to understand it or carry out more advanced calibrations (and probably using a more complex friction model). The change of tool geometry (pin length for example) could also be a manner to identify the model parameters.

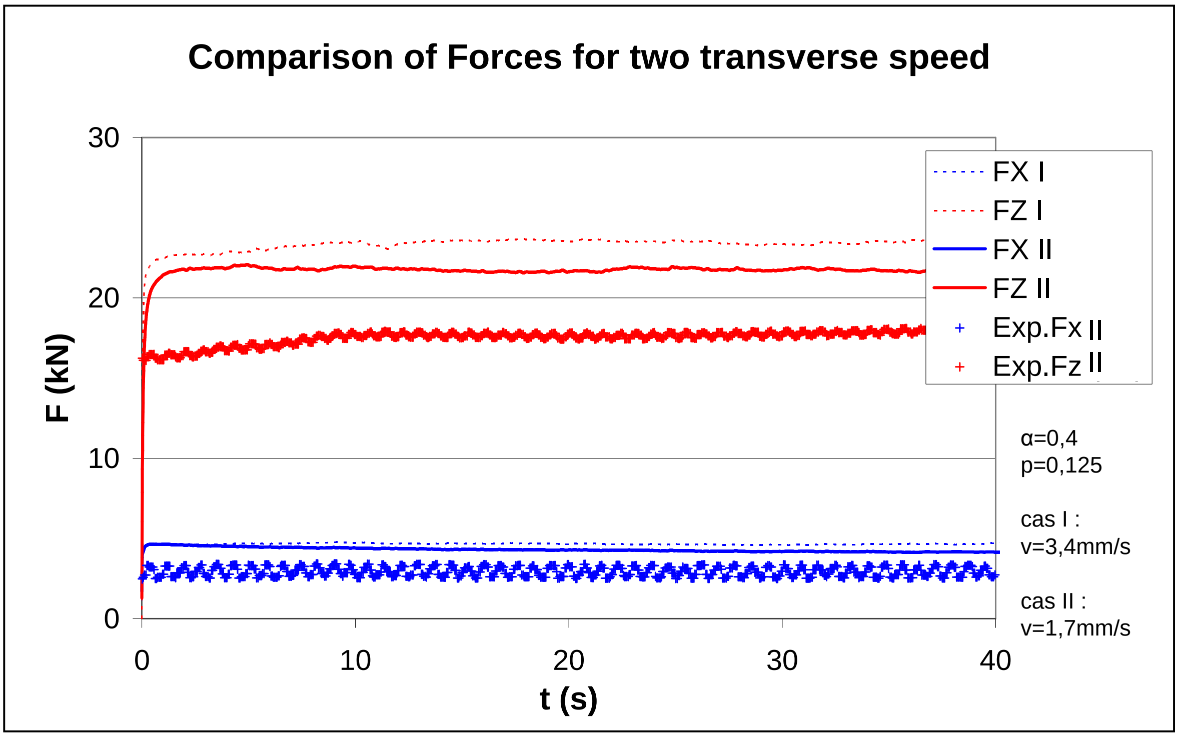


Figure 112: Comparison of simulated forces for two different welding speeds and with experimental data

2.4.2 Stream lines and material flow

It is widely accepted that material flow within the weld during FSW is very complex and still poorly understood. It depends on the tool geometry, process parameters, and welded material. For optimal tool design and to obtain high structural efficiency welds, it is of practical importance to fully understand the material flow characteristics. A number of approaches, such as tracer technique by markers, welding of dissimilar alloys/metals, have been used to visualize material flows in FSW.

As soon as a steady welding state is reached, the stream lines are assumed constant. They can be visualized by following the movement of Lagrangian sensors (as also shown in the plunge phase study in section 3), which are similar to embedded markers.

Figure 114 shows the initial and the final positions of 79 Lagrangian sensors which have gone under the tool during steady welding. The aim of this simulation was to reproduce an experimental weld test (conducted by Dongfang Huang and Professor L. Brent Adams in BYU) for which statistical texture observations had been done. The simulation had to provide the thermomechanical history of the particle along the stream lines of the 79 studied points in order to allow the computations of the final textures of the weld and the comparison with experimental results. The positions of the studied points and some data concerning the experimental parameters are shown in Figure113.

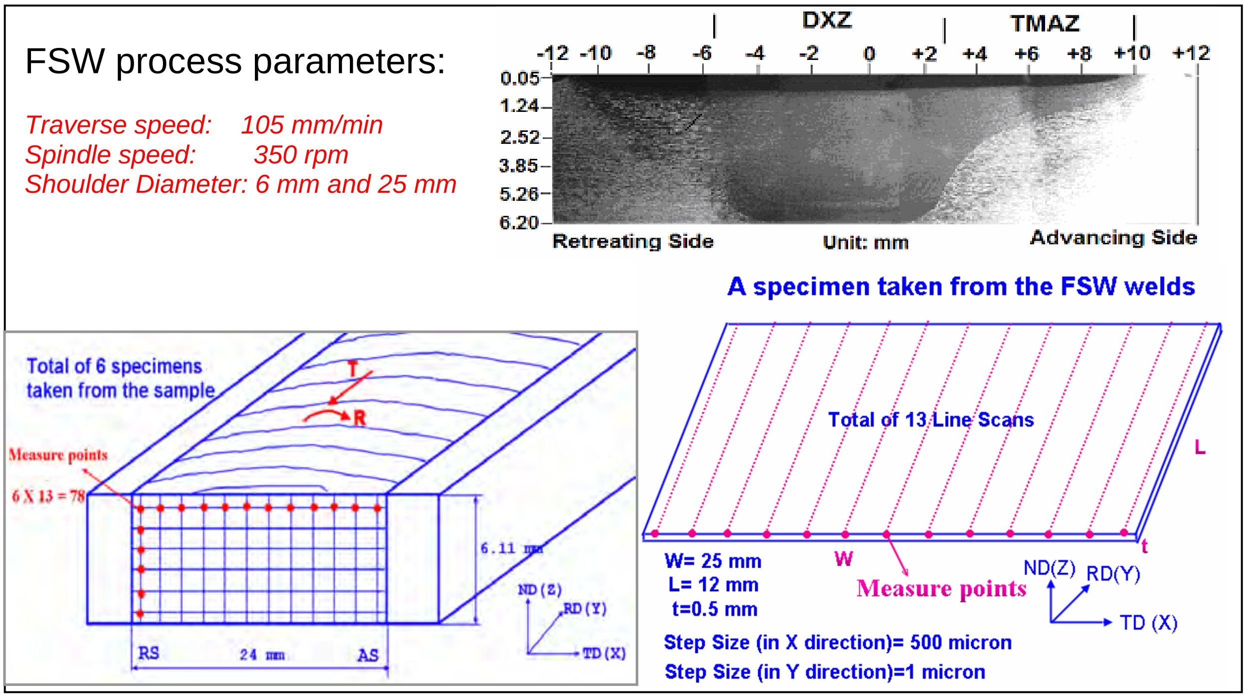


Figure 113: Comparison of simulated forces for two different welding speeds and with experimental data

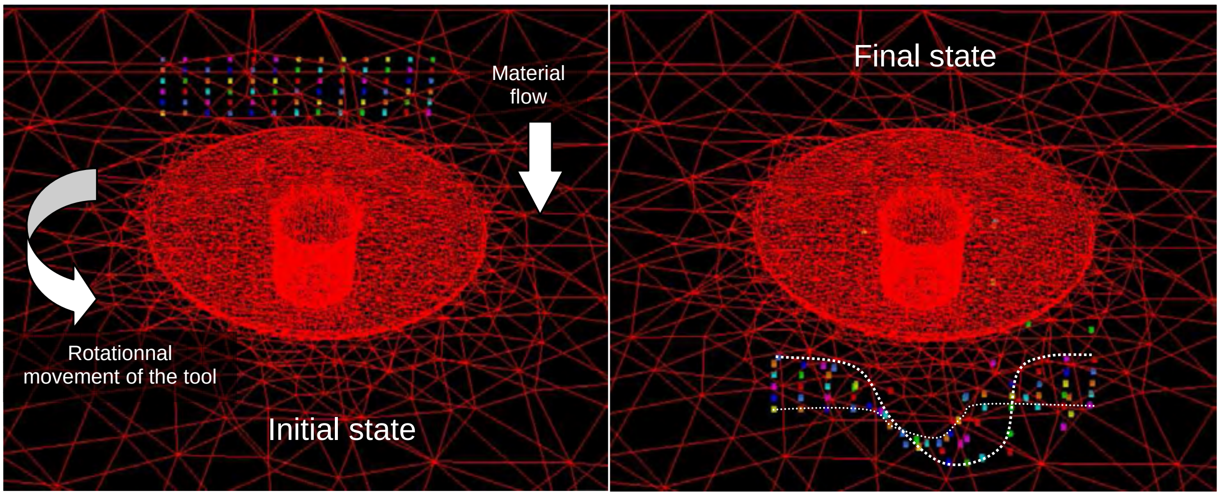


Figure 114: Movement of Lagrangian particles when passing through the welding tool at steady state.

These Lagrangian sensors can only follow the stream lines in the sense of the material flow. However, the experimentally studied points are distributed according to the desired final position of the sensors. Therefore, an enhanced procedure has been implemented to allow travelling forward and backward (and stored all desired data) along the streamlines passing at one given point at one given moment. A post treatment with Matlab permits to visualize the streamlines that are shown in Figure 115.

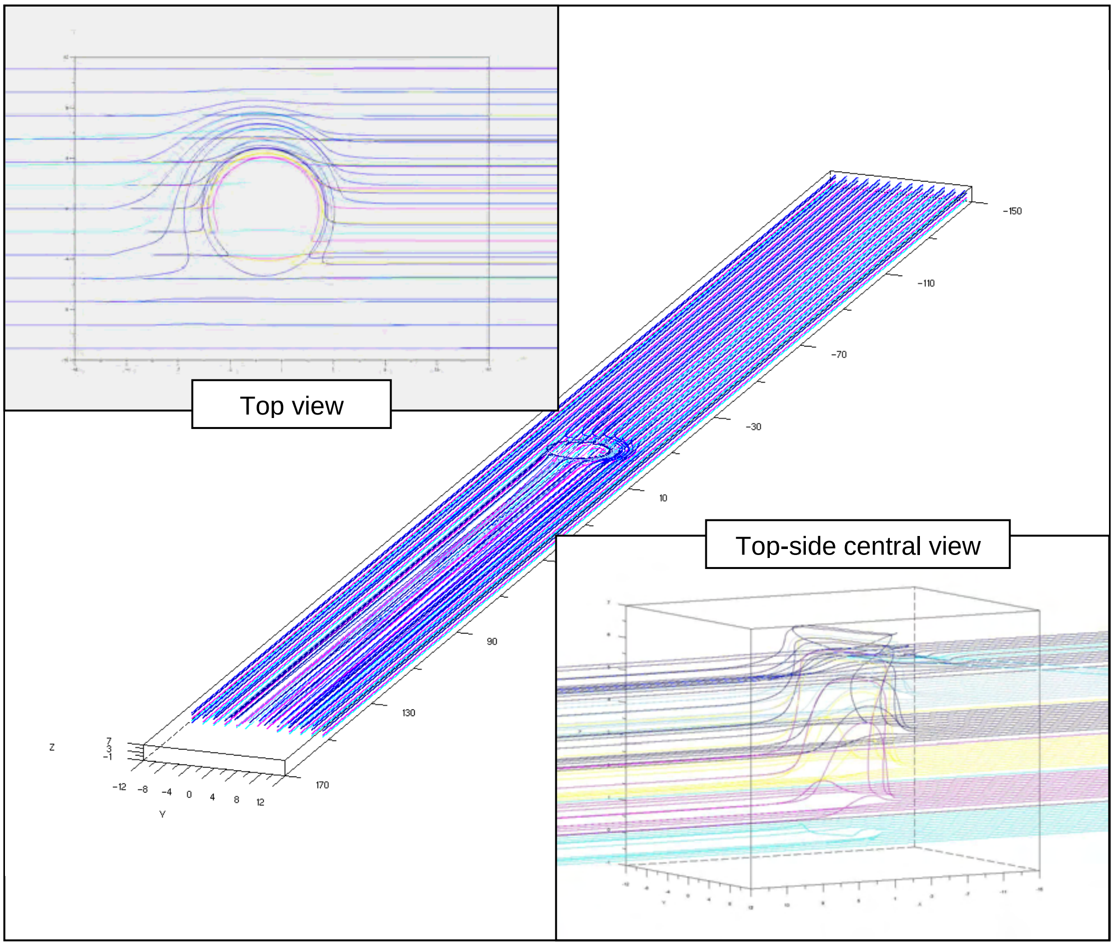


Figure 115: Movement of Lagrangian particles when passing through the welding tool at steady state.

The simulated weld provides some common flow patterns which are described in literature. It has been suggested by some researchers that FSW can be generally described as an in situ extrusion process and the stirring and mixing of material occurred only at the surface layer of the weld adjacent to the rotating shoulder.

The flow is not symmetric with respect to the welding centreline. Only a small amount of material of the advancing side is moved to a final position in front of its original position. The backward movement of material is limited to one pin diameter behind its original position. The material is pushed downward on the trailing side and moved toward the top at the front side within the pin diameter. At the top of the weld the material flow is directly influenced by the tool shoulder that moves material from the retreating side, around the pin, to the advancing side.

Further observations are made in section 3.2.2 for a new tooling design.

3 *Transient States*

Often ignored in literature, the simulation of the transient states of the FSW process is enabled with the developed ALE formulation. Since the contact condition are not prescribed, partial contact is allowed at the tool/matrix interface, e.g. at the periphery of the shoulder and the pin, where a solution dependent ‘foot print’ is established.

As already explained and detailed in previous chapters, the grid velocity is computed through a specific adaptive procedure, and an adaptive full remeshing procedure is also used to change the mesh topology. The formulation allows the description of movements of free surfaces, so, the plunging phase but also the dwelling and transient welding states can be simulated.

3.1 Plunging phase

3.1.1 Experiment description

In this second study, we consider a six seconds, beginning to end, experimental plunge test, which has been conducted at BYU by Mike Miles and Alma Oliphant for experimental validation of the 3D thermomechanical plunge simulation.

Figure 116: A view of the experimental plunge, just prior to the test.
Notice the thermocouple wires extending out from the bottom of the block.

The plunge rate was 1.19 mm/s, with a rotational speed of 600 RPM. The welded material was instrumented with six, 1.6 mm diameter type K, stainless steel sheathed thermocouples located at 4.8 millimeters below the welded surface, and at varying radial positions from the center of the plunge (see Figure 117). These thermocouples were threaded into the holes ensuring correct placement. Channels were milled into the plate block as shown in Figure 117 to route the thermocouple wires to the data acquisition system. A small center hole, as shown in Figure 117, was drilled in the center of the plunge block to ensure that the FSW tool was centrally located with respect to the six thermocouples. Data were recorded at 50 milliseconds intervals during the experimental test.

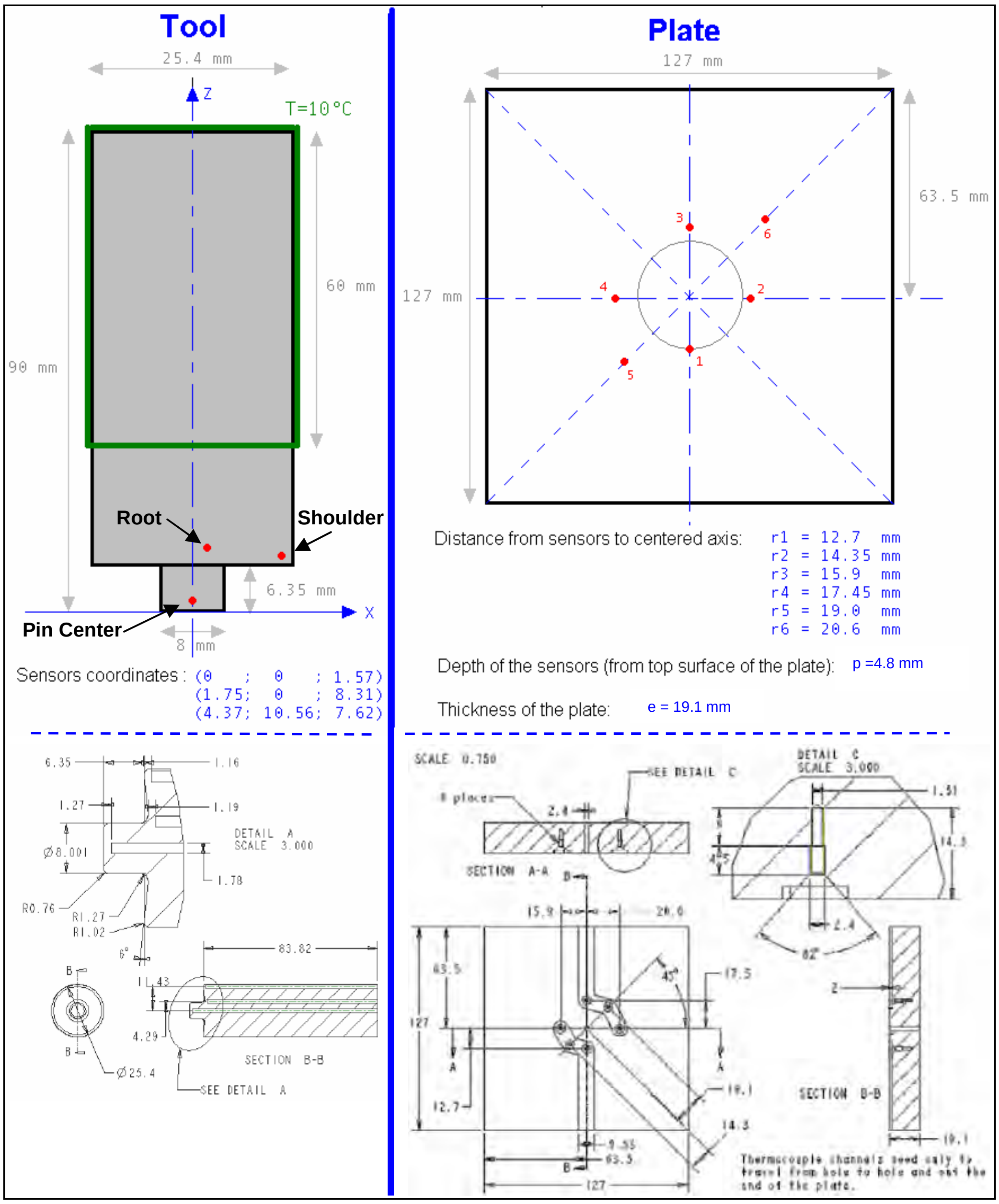


Figure 117: A drawing of the FSW tool and plate geometry used in the experimental plunge test, all dimensions in millimeters. Notice the holes placed in the material to allow for the insertion of thermocouples.

The tool used for this study was nearly the same as the tool used in the previous experimental welding test (section 2.1 of this chapter). The only difference is that the unthreaded pin was 4.8 mm (0.1875 in) length (not including the portion that extended into the concave shoulder cavity) and untilted. As previously described, separate thermocouples were embedded in the tool material (shown in Figure 117) at three locations, the tip of the pin, the root of the pin, and the edge of the shoulder, to allow for the measurement of the tool temperatures at those locations. These three thermocouples were not threaded into place.

3.1.2 Modelling

The experiment has been modelled using the actual geometry and process parameters: the 3D thermomechanical computation is carried out in the plate only, but the temperature is calculated in all three bodies (plate, tool, backing plate) in a coupled manner.

The initial mesh is constructed with a refined zone at the location where the tool is going to contact the workpiece. The initial number of elements is about 20,000 for 4,000 nodes. During the computations, the mesh is automatically adapted by the error-estimation procedure, providing a good trade-off between accuracy and model size. The ALE formulation is utilized and coupled to the automatic adaptive remeshing procedure when the elements are too distorded and when the mesh topology has to be changed. The number of elements is limited to 70,000 with a minimal size of about 0.5mm.

About 52,000 elements (10,500 nodes) have been utilized to discretized the FSW tool and 30,000 (6,000 nodes) for the backing plate. Eulerian sensors have been introduced at the same positions as the experimental thermocouples, both in the tool and in the workpiece (as presented in Figure 117 and 118).

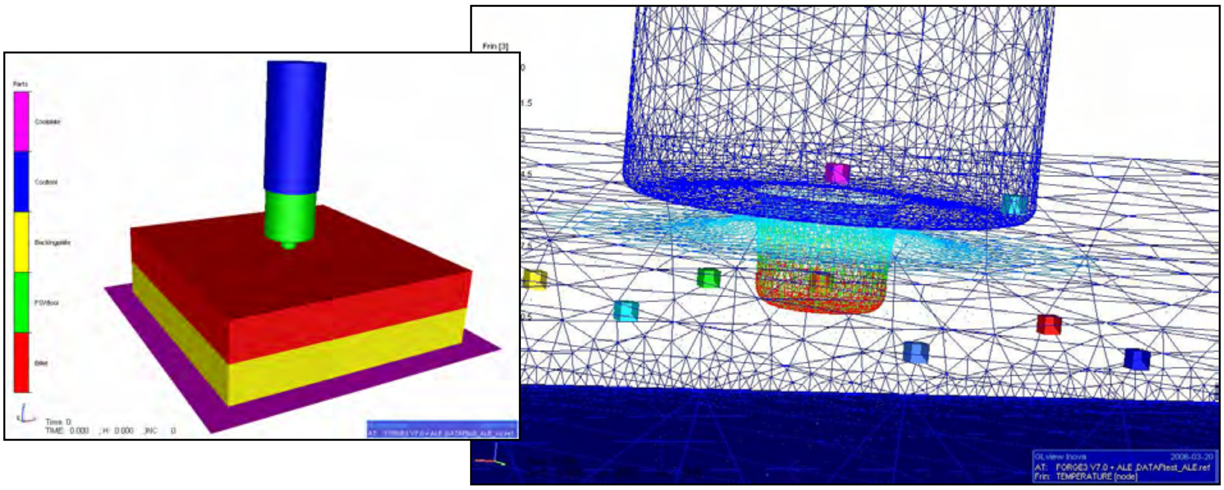


Figure 118: Global view of the model with numerical sensors located as in the experimental plunging test

3.1.2.1 Material constitutive law

The material used for this study was the AL 7075-T7351 with a thickness of 19.1 mm. The oxide layer was removed with a portable disc sander and the surface was cleaned with methanol prior to processing. The thickness of the plate was predetermined so that no interaction could exist between the tool and the anvil.

As already discussed in the first study, the temperature of the material may reach locally the melting point during the process, and so the validity limits of the Hansel-Spittel constitutive laws provided by the Forge3® material data-base (shown in Figure 119) can be over passed (the solidus of Al 7075 is around 480°C and its liquidus around 632°C). So, in the absence of more precise data, the constitutive law has been approximated by the one already used for Al 6061 and detailed in the previous section (see Figure 100). The Young Modulus and Poisson's coefficients are assumed constant and respectively equal to 73GPa and 0.3.

Hansel-Spittel Coefficients for AlZn5,6Mg2,5Cu1,5(DIN)--**7075**(US)

$$\sigma_f = A_1 \times e^{m_1 \times T} \times \bar{\varepsilon}^{m_2} \times \dot{\bar{\varepsilon}}^{m_3} \times e^{\frac{m_4}{\bar{\varepsilon}}}$$

Temperature range	**20 - 250**	**300 - 500**
Strain range	**0.04 - 3**	**0.01 - 1**
Strain rate range	**0 - 500**	**0 - 200**
A_1	351.78634	731.55463
m_1	-0.0017	-0.00479
m_2	0.13386	-0.01383
m_3	0.01816	0.09964
m_4	-0.00811	-0.0011

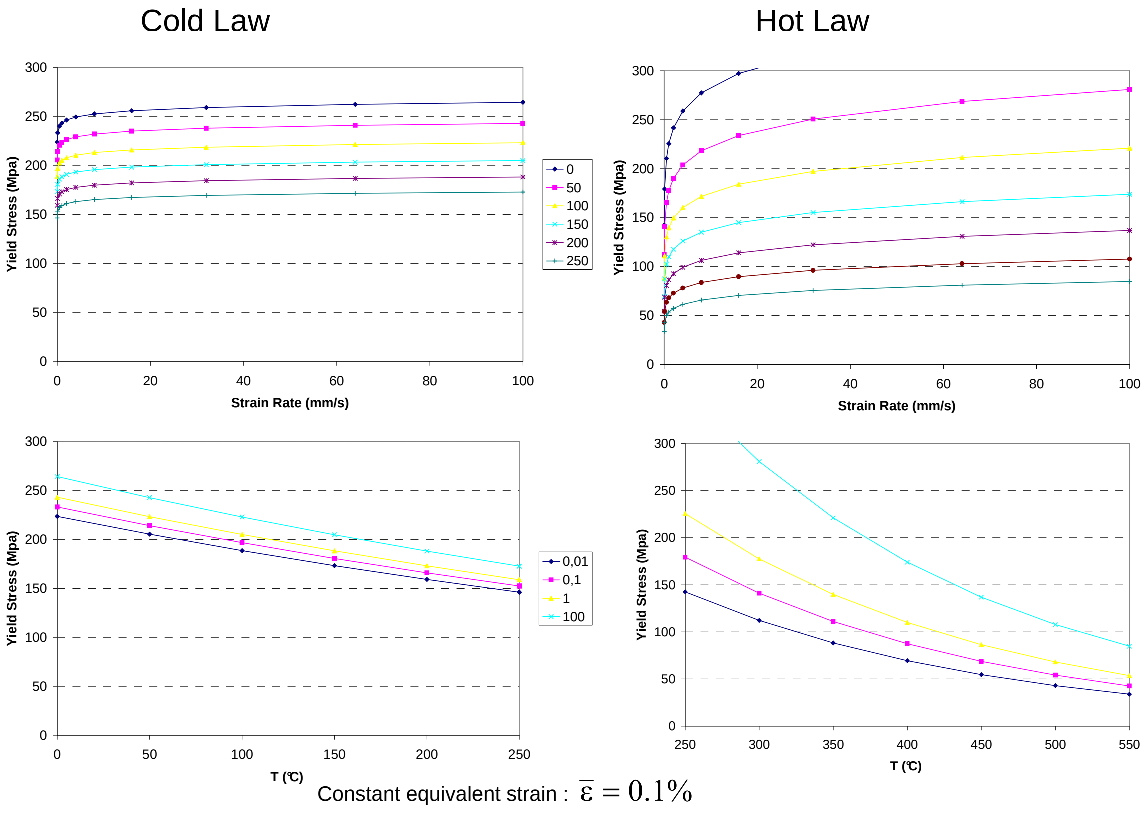


Figure 119: Yield Stress visualization for hot and cold constitutive law for Al 7075

3.1.2.2 Thermal characteristics

With respect to Figure 102 in section 2.2.1.1, only the conductivity of the welded material is changed to 167 W/m.°C (corresponding to Al 7075).

3.1.2.3 Boundary conditions

3.1.2.3.1 Thermal

The thermal boundary conditions are assumed the same as in the first study. They are described in section 2.2.1.1.

3.1.2.3.2 Mechanical

The mechanical boundary conditions are also almost similar. The differences are:

- there is no inlet velocity, so there is no relative transverse movement between the tool and the weld material.
- the friction coefficients α_f and p are taken constant and respectively equal to the values identified in the previous welding test: 0.4 and 0.125.

3.1.3 Study of numerical results

3.1.3.1 Temperature

Since the temperature distribution within and around the stirred zone directly influences the microstructure of the welds, such as the grain size, the coarsening and dissolution of precipitates, and the resultant mechanical properties of the welds, it is again of major importance to accurately compute the temperature distribution.

The temperature field at the end of plunging provides the initial state of the welding phase and makes it possible to correctly predict the transient conditions of the initial deposition process at the back of the pin. Furthermore, simulating the plunging phase can be helpful for studying the Friction Spot Welding Process, which derives from the Friction Stir Welding in the sense that the two stacked parts are welded together by several plunging and dwelling phases at different locations.

The beginning of plunging results in intense friction between the tip of the pin and the workpiece. Heating increases as the friction area enlarges on the lateral face of the pin. Before the shoulder comes in contact, the plastic deformation is mainly due to the indentation of the tool. Then the intense friction and plastic deformations around the rotating tool are the two factors which contribute to the temperature increase within and around the stirred zone. The evolution of the temperature field in the cutting joint during the 7 sec. of simulation is shown in Figure 120.

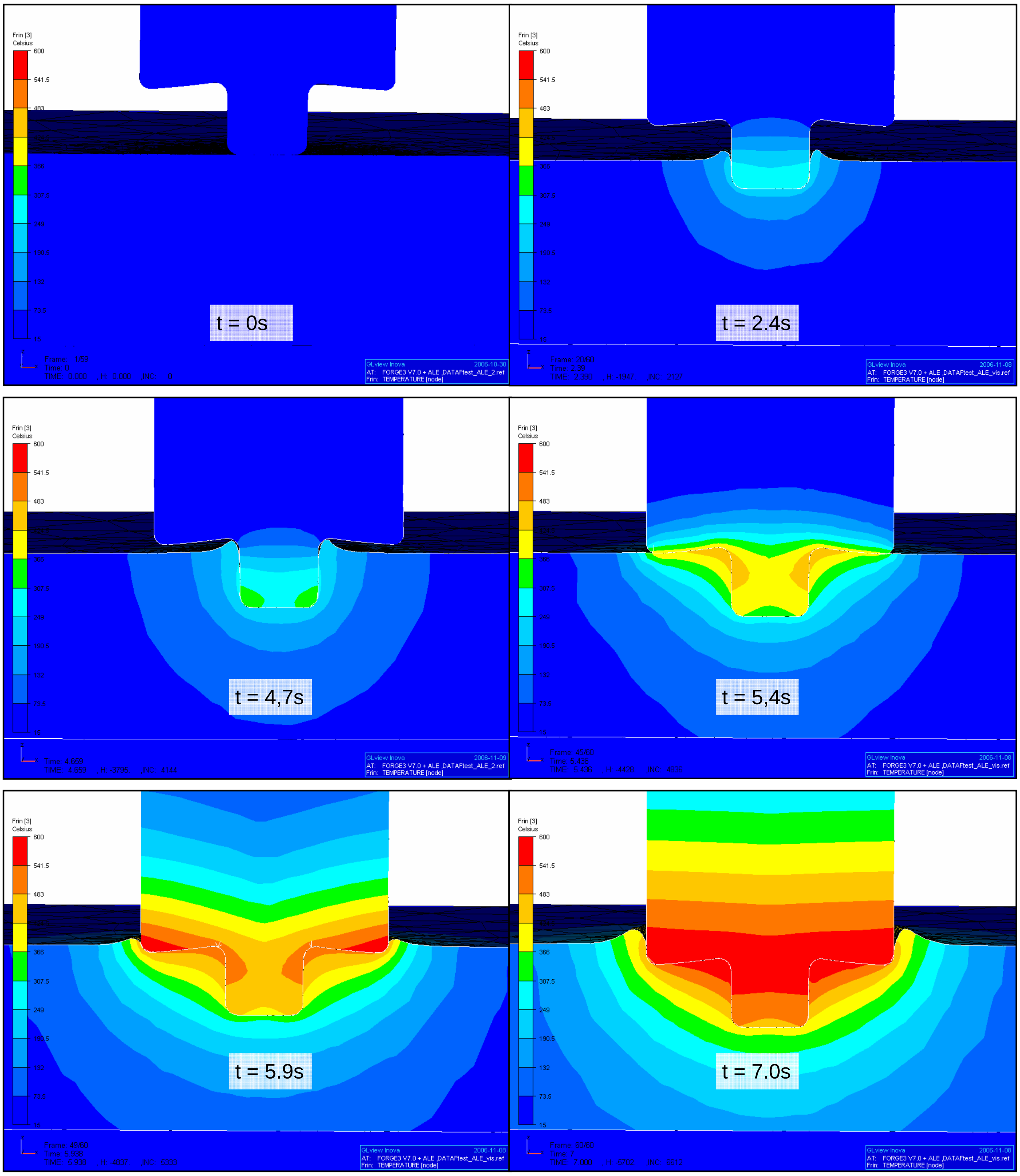


Figure 120: Temperature map in the cutting joint plane during the plunging phase.
* viscoplastic friction is use with α_f=0.4 and p=0.125
* the tool is untilted and unthreaded

Figure 121 shows the experimental temperature histories recorded at the three locations in the tool. Notice that the value at the tip of the pin increases with a nearly linear rate as the tool is plunged into the welded material. Later, the temperature increases at the root of the pin, and a little while latter the shoulder temperature raises as the shoulder contacts the workpiece. The generated heat is transferred into the pin first, traveling up the pin as its tip plunges into the welded material. This heat then moves up and reaches the root of the pin, more or less at the same time as the root is in contact with the plate. Eventually, it is conducted out to the edge of the shoulder. The additional heat generation resulting from the shoulder increases its temperature and makes it match that of the root temperature.

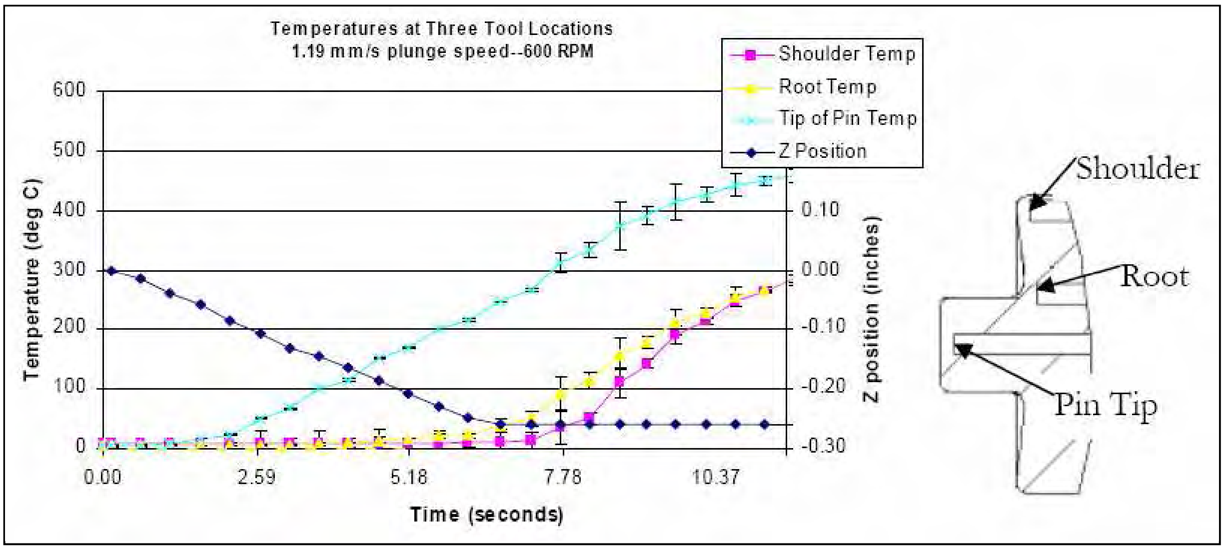


Figure 121: Averaged temperature experimentally measured into tool during the three plunging experiment.

The calculated temperatures at the same locations are shown in Figure 122. The above description of the experimental temperatures evolution can also be repeated for the computed ones, although some singular differences exist. Firstly, instead of increasing at a nearly linear rate as the tool is plunged into the plate, the temperature at the tip of the pin increases very quickly at the beginning and then more softly until the shoulder comes into contact. This difference may come from the fact that in the experiment, the centring hole provides an outlet for the material trapped under the pin. A part of the material being extruded into the central hole, the pressure is lower under the pin (this is confirmed by the axial force shown in Figure 128) and there is a more progressive increase of the temperature.

Secondly, the calculated temperature of the tool shoulder increases very sharply when the shoulder contacts the workpiece, and finally looks over predicted. In the simulation, the shoulder is the hottest part of the tool at the end of the plunge and the maximum computed temperature around the 580°C.

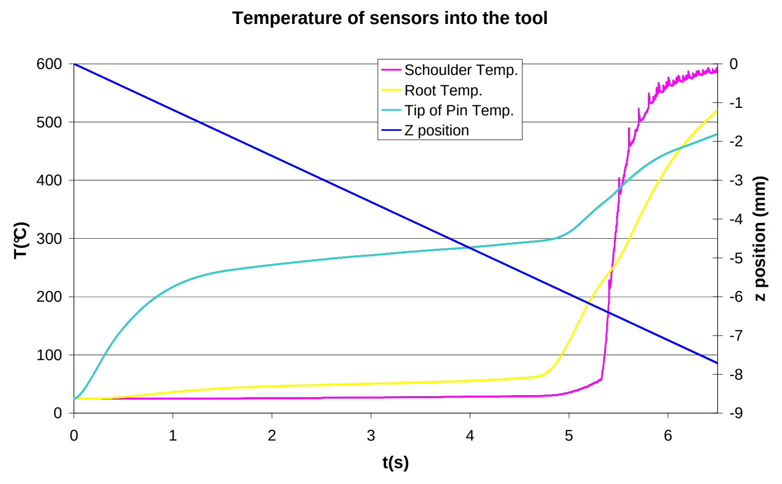


Figure 122: Temperature calculated into tool during plunging phase

Many factors of the model influence the temperature evolution. Those which are supposed to be the main causes of the observed differences between simulated and measured results are discussed thereafter in § 3.1.3.3.

Regarding the plate, the temperature plots shown in Figure 123 are an average of three experimental plunging tests. It shows the measured values by the six thermocouples located in the plate. Error bars are included in the plot for one standard deviation above and below the data point.

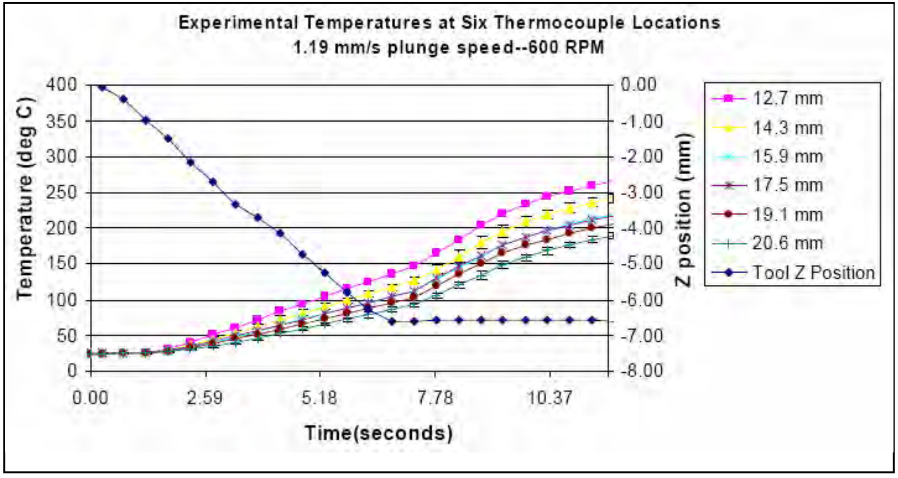


Figure 123: Plots of temperature histories in the plate during the experimental 6 seconds plunge at various radii from the pin center.

The temperature of each of the thermocouple evolves in parallel according to their distance to the center of the weld, linearly increasing as the tool plunges into the material (see Figure 123). Also, notice in Figure 123 that the highest temperature value was recorded at the thermocouple located nearest to the pin. This is intuitively sound, as one would expect the temperatures to decrease along with the distance from the pin center. Each of the temperatures recorded in Figure 123 still increases slightly after eight seconds, which results from the additional heat generated by the shoulder (at approximately 8 seconds) to the already existing heat generated by the pin (between 0 and 7 seconds) and to possible time lag from measurements as discussed later (see section 3.2.1.2). The temperatures rise at an increased rate under these combined heat phenomena.

The temperature calculated at the six sensors locations is shown in Figure 124.

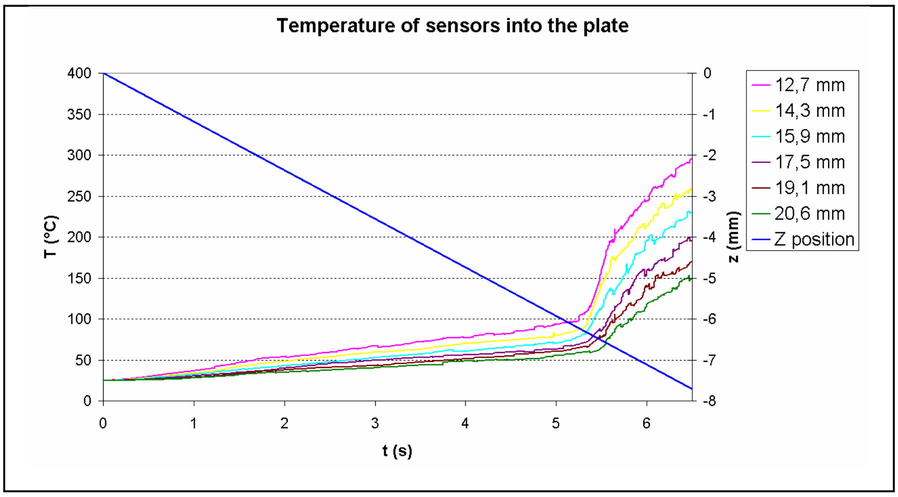


Figure 124: Temperature at Eulerian sensors located in the workpiece during the plunge simulation.

The parallel behaviour of the curves is in good agreement with the same parallel behaviour of the measured data (Figure 123). Also, as the shoulder contacts the plate at 5.3 seconds, the temperatures jump rapidly under the effects of the shoulder heating, which is intuitively sound. However, this rise is much sharper than the experimental one.

Temperature measurements within the stirred zone are very difficult to carry out due to the intense plastic deformation produced by the rotation and translation of the tool. Therefore, these temperatures are rather estimated from the microstructure of the resulting weld. Figure 125 shows an image of a sectioned, polished, and etched experimental sample after plunging. The right image has been modified to highlight the heat-affected and mechanically deformed areas of the welded material. Also shown are the threaded thermocouple holes at 20.6 mm on the left and 15.9 mm on the right.

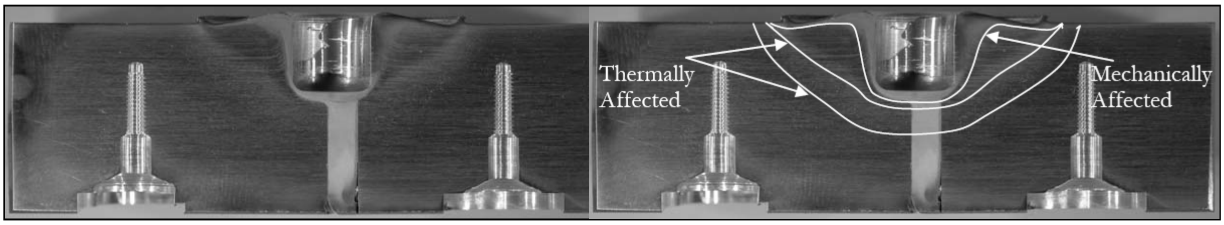


Figure 125: Cross section of a 7075 sample after plunging, showing mechanically and thermally areas and thermocouple locations.

Notice that the profile of the thermally affected area in Figure 125 extends from the edge of the shoulder, down underneath the pin tip, and back to the edge of the shoulder on the opposite side of the pin. These profiles provide a physical indication of the isotherms taking place during the plunging sequence, and a comparison with numerical results is done in Figure 126. The temperature provided by the simulation between the two red lines ranges from 250°C to 370 °C. The material located upper this second red line has reached the annealing temperature, which is about 413°C for the Al 7075. The material located in the mechanically affected zone (MAZ) has reached the solidus temperature of 477°C.

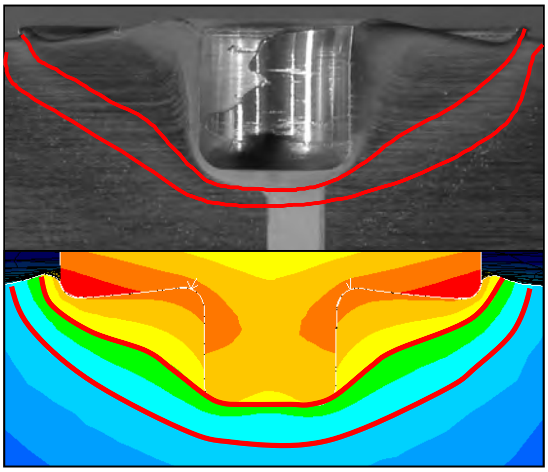


Figure 126: Comparison between the simulated temperature field and microstructure observation in the cross section

The equivalent strain field obtained through the simulation is shown in Figure 127 at 5.4 s. It is noticed that the equivalent strain ranges between 2% and 5% in the experimentally observed MAZ. This value quickly increases when the shoulder is maintained in contact, and a value of about 10% is reached in the MAZ at t = 6 s.

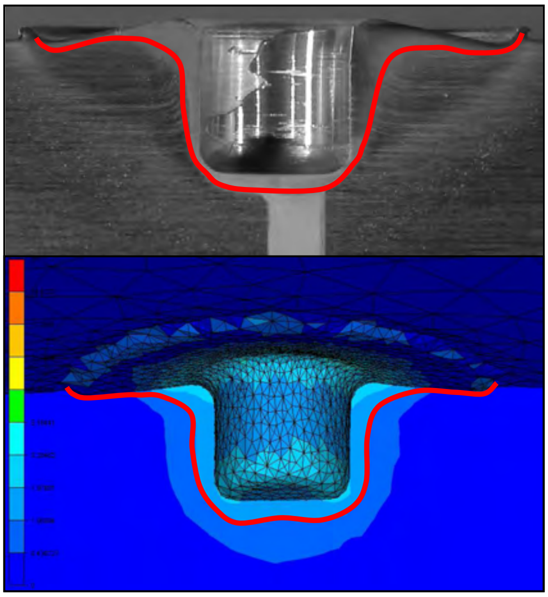


Figure 127: Comparison between the experimentally observed MAZ and simulated equivalent strain field in the cross section when the shoulder is just getting in contact with the plate (t=5.4 s)

3.1.3.2 Forces and Torques

Figure 128 shows a plot of the experimental axial force and the motor power versus time. The three events of the plunge consist of the initial pin penetration, the full pin travel, and the shoulder contact with the plate.

Notice in Figure 128 that the force jumps rapidly to a steady state value of 10 KN when the initial pin penetration occurs. This first step corresponds to the force necessary to make the material trapped under the pin flow into the centring hole which has been preliminary drilled (see cross section in Figure 125). As the tool travels into the plate, the force increases to about 23kN until about 6.5 seconds when the shoulder contacts the plate and the force jumps rapidly to about 45 KN. The motor power jumps rapidly to the value of 2 kW and then increases slowly to 4kW until the shoulder contacts the plate. Then it raises its maximum of 9.6 kW and decreases gently to 7 kW as the plunge rate is nullified.

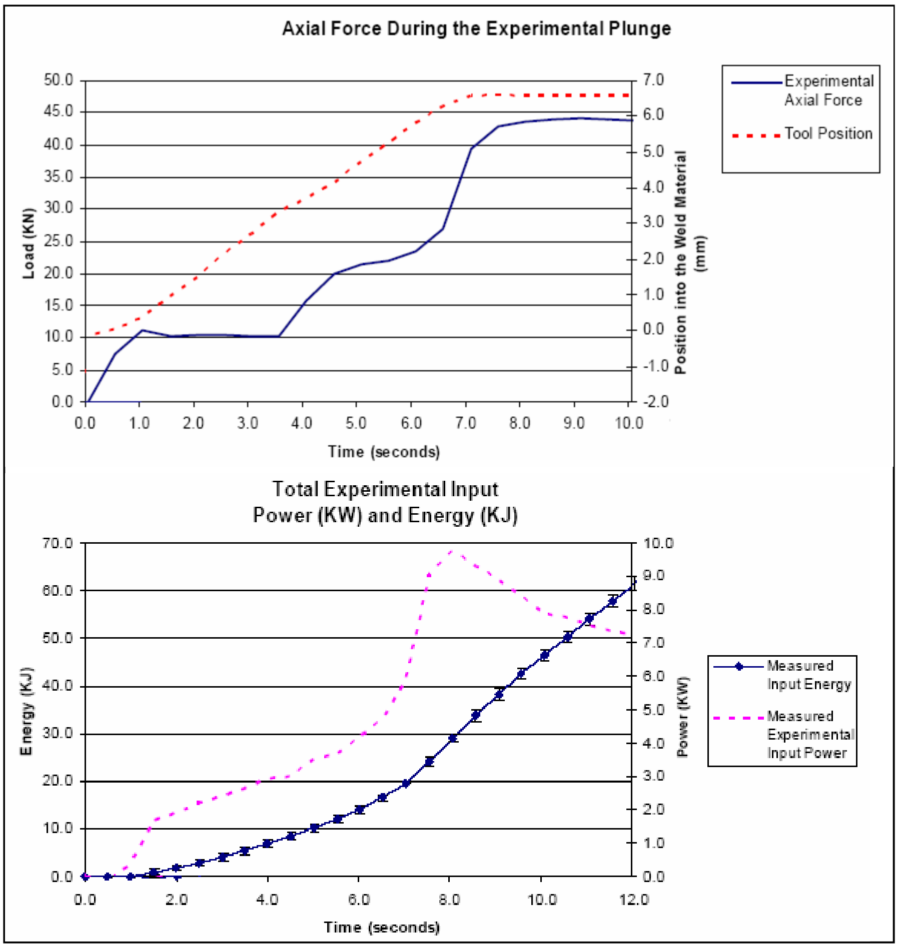


Figure 128: Axial force (top graph.) and Motor Power (bottom graph.) registered during the experimental plunging test.

These two graphics can be compared to the numerical results of Figure 129, which shows the calculated axial force and the power developed by the simulated torque.

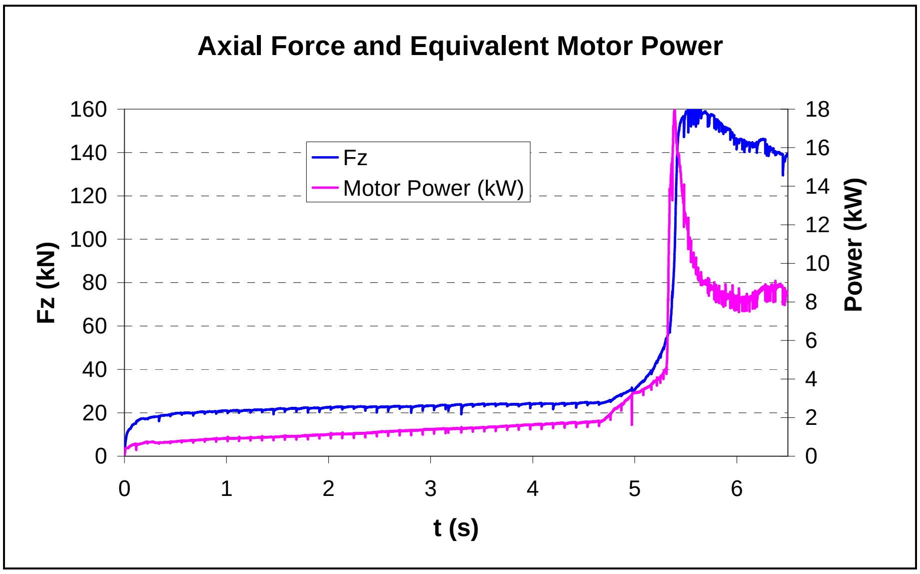


Figure 129: z- force and equivalent motor power simulated during the plunging phase

The axial force increases rapidly to about 19kN when the initial pin penetration occurs. As the tool travels into the weld material, the force increases to about 23kN and then raise up to the over-estimated value of 160 kN when the shoulder contacts the material. Even if the plunge velocity is maintained, it then decreases as the temperature increases and mesh is auto-adapted to the new contact conditions.

The power is calculated from the z-torque using the equation (V-1) and can be abusively compared to the motor power experimentally measured (frictional losses are not taken into account). The simulated power jumps to 1kW at the pin contact and increases slightly to 2 kW as the pin travels into the weld material. Then the values jump to 18kW before drop and stabilize around the 8kW. Except the pick value reached when the shoulder contacts the weld, it compares well to the experimental measurement.

$$P = T.\omega \qquad \text{(V-1)}$$

P is the power (W) abusively compared to the Motor Power, T is the axial-torque (N.m) and ω is the rotational velocity of the tool (rad/s).

3.1.3.3 Discussion

Some numerical factors may be at the origin of the observed differences between experimental and simulated results. They are listed as follows:

- Even though, a preliminary sensitivity analysis confirmed that there is no significant variation of temperature in the workpiece as the interface heat exchange coefficients change, different coefficients could generate significant temperature change at some tool locations.
- When the tool shoulder contacts the workpiece, there is a quick and huge change in the mechanical boundary conditions which are imposed by contact. The mesh is not immediately well adapted and needs to be quickly and permanently discretized until the whole shoulder surface properly contacts the workpiece. The mechanical and thermal errors generated during the mesh adaptation can be significant with respect to the time required by the shoulder to contact the workpiece.
- The friction model could be the major source of errors on temperature. More than 80% of the heat is generated by friction. According to the simulation results, it looks that not enough heat is generated by friction between the pin and the workpiece during the time the pin travels into the plate, whereas too much heat is generated by friction with the shoulder. The same problem was observed during the welding experiment of previous section.

Other reasons, which are not numerical, also complicate the comparisons and create dissimilarity between measured and simulated values:

- In the experiment, the tool plunges into the workpiece within 6 seconds, after what the tool is kept rotating at a final constant Z position. In the simulation, the tool plunges into the workpiece within 7 seconds. Therefore, the results can be compared only during the first 6 seconds, after what the simulation logically results into higher loads and higher temperatures.
- Moreover, the final experimental Z position, which is about 6.6 mm inside the workpiece, does not correspond to the position of the simulation after 6s because of the deflection of the experimental machine which has not an infinite stiffness. Figure 130 shows the non linear variation of the actual tool position due to the machine deflection. Thus, the time scale is

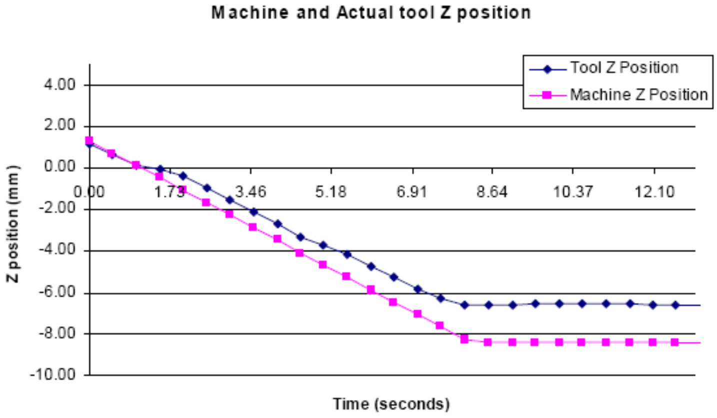


Figure 130: A plot of the machine input Z position compared to the actual tool Z position.

fundamentally different in the experimental test. The acceptable time range to compare simulated and experimental results is about 5.6 s on the graphics showing the simulated results (time at which the tool has penetrated 6.6 mm into the workpiece). Easier comparisons might be possible by representing the temperature values versus the plunge depth position.

- In addition, the thermocouples were of the ungrounded junction type (see Figure 131). This configuration increases the response time of the thermocouples, so measurements during the transient phase of FSP may have some time lag. The temperature measurements are proved to be accurate in steady-state regions for a measured time constant of 4 sec. This time lag is also visible for temperatures measured into the plate, which six values are shown in Figure 123.

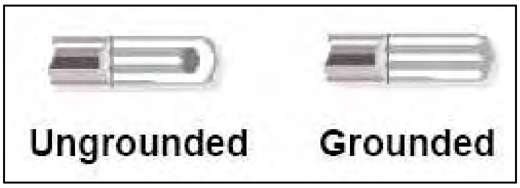


Figure 131: Thermocouple junction types.

In a preliminary study, carried out at BYU by Alma Oliphant [55], the heat transfer condition between the FSW tool and the weld material was assumed to be adiabatic. This condition was chosen to simplify the computations, with the assumption that the temperature measurements of the resulting analysis would be higher than the experimental results. The resulting calculated temperatures are shown in Figure 132, from the initial Lagrangian formulation of Forge3®.

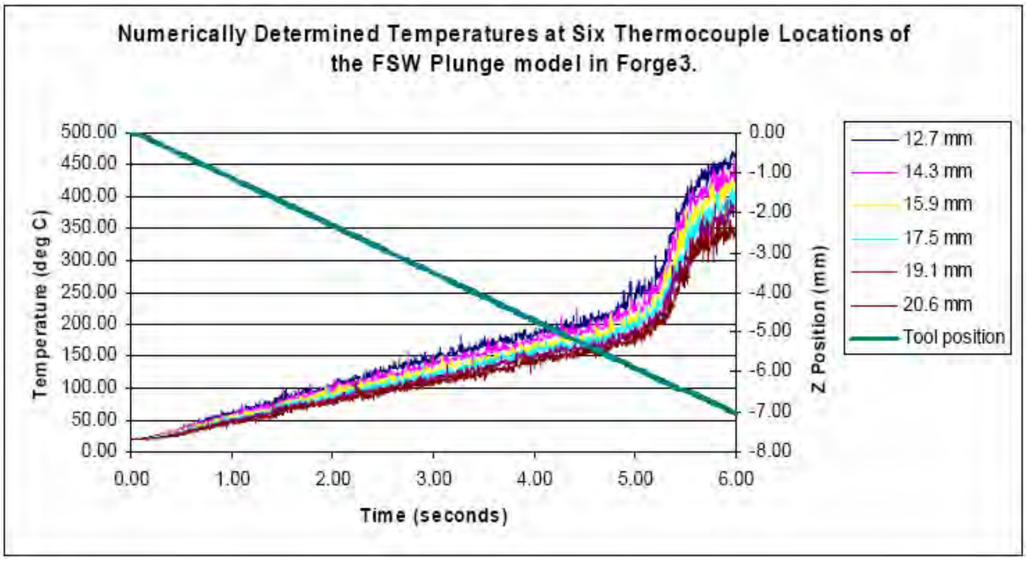


Figure 132: Temperature histories calculated with Forge3 at the six thermocouple locations.

As expected the resulting temperatures are over-estimated. The same curve profiles similar to the ones of Figure 124 are observed. However it is noticed that the simulated results compare better to experimental ones when a thermal computation is performed inside the tool. Another remarkable observation is that the adaptive ALE formulation provides less diffusive results both in terms of temperatures and forces. Figure 133, which shows the resulting axial force obtained with the Lagrangian formulation, can be compared to Figure 129, which shows the force obtained with the improved ALE formulation. This diffusion is mainly caused by remeshing, which number is considerably reduced by the ALE formulation.

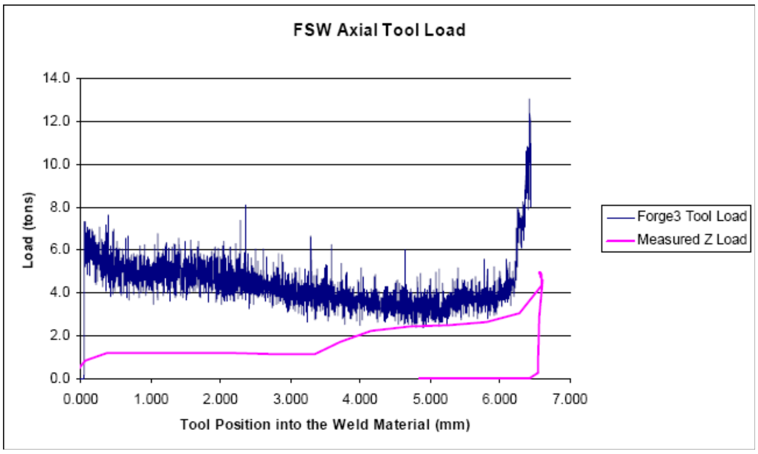


Figure 133: Comparison of the experimental z-force with the force preliminary simulated using the initial Lagrangian formulation of forge3

3.1.4 Material Flow

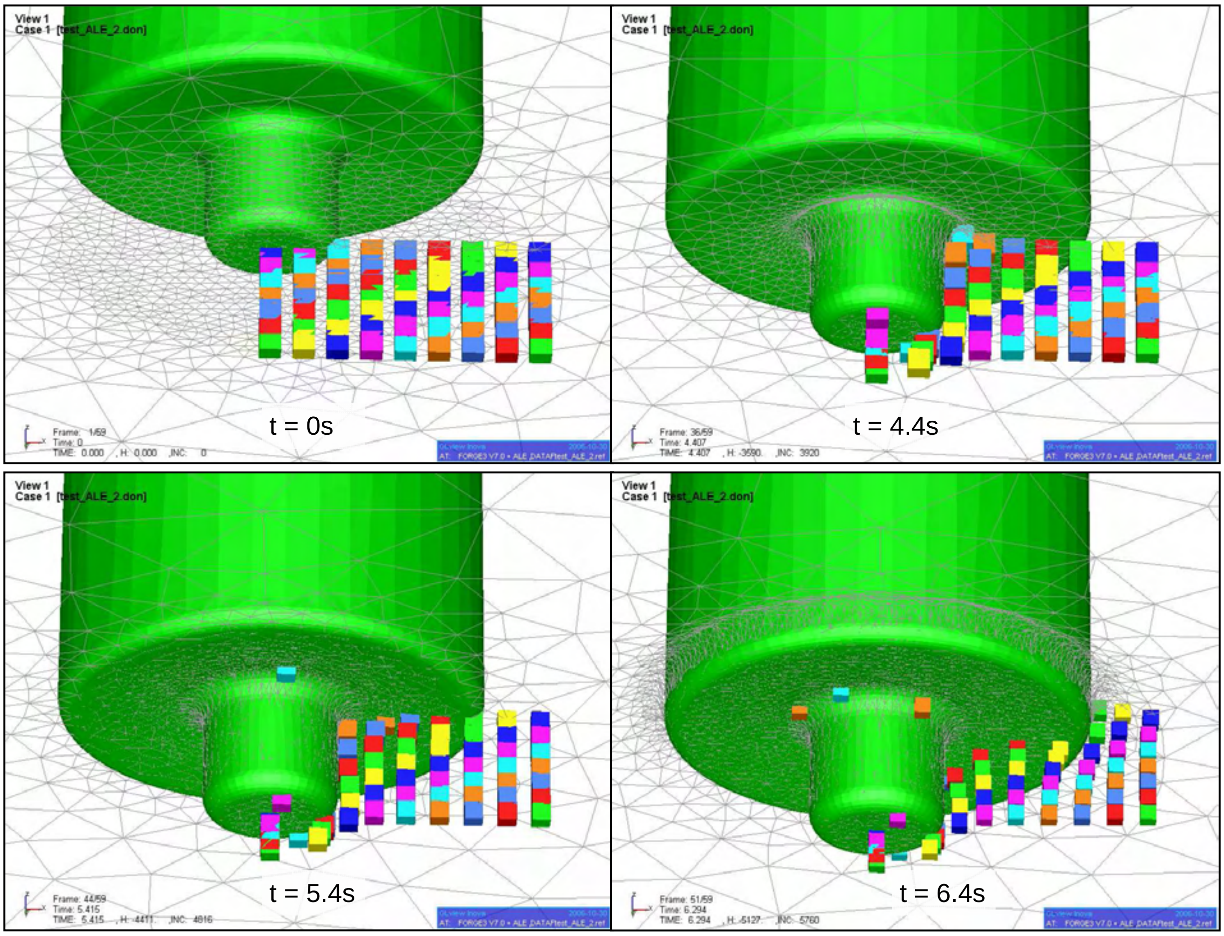


Figure 134: Visualization of the material flow using Lagrangian sensors during the plunge simulation.

In order to visualize the material flow during the transient plunging phase, 63 Lagrangian sensors, corresponding to material particles, have been introduced into the plate. They are initially disposed in a same plane, in 7 rows (from 0 to 7 mm under the weld surface) and 9 columns (0 to 18 mm from the tool axis). Figure 134 shows their positions at four different moments. While the pin is traveling into the plate, the two columns of sensors initially located under the pin are "compressed" (a vertical displacement is observed for all of their sensors; the surface ones have a weak movement of rotation) and the sensors of the third row tends to move away from the pin. Huge rotational movements are observed for the sensors located in the mechanically affected zone after the shoulder contacts the workpiece (see Figure 127).

3.1.5 Dwelling simulation

The dwelling phase is a transition between plunging and welding that can be used to increase the heat produced by friction before translating the FSW tool along the joint line: the tool is maintained horizontally fixed by rotating at the constant final plunge depth. The tooling kinematic change requires running a new simulation. However, the stress and loading state resulting from the plunge phase is to be preserved as initial state of the dwelling phase.

An unloaded initial state leads to bad contact conditions: some nodes are rapidly removed from contact and a rotating wave of material is observed under the shoulder. The friction area between the tool and the workpiece decreases and so the temperature.

Assuming that it could be a numerical bias, a low numerical plunge velocity has also been applied to the tool in order to insure a proper loading state and good contact conditions. However, this modified loading state does not perfectly correspond to the experimental load.

Figures 135 and 136 show the differences respectively observed on the temperature field (top surface of the workpiece and cross section views) and on the forces and torques during two 2.5 s dwelling simulations. The initial state (resulting from the plunge phase simulation) is the same for the two simulations but the first one is performed with an additional constant plunge velocity of 0.05 mm/s.

However it is noticed that the simulation with no plunge velocity provides consistent results: the plunge force and torque rapidly decrease in the beginning, because of the plunge rate cancellation, then stabilize and finally slowly decrease as the temperature of the workpiece globally increases. The “modified” simulation shows that the additional low plunge velocity logically increases the load and therefore the heat produced by friction.

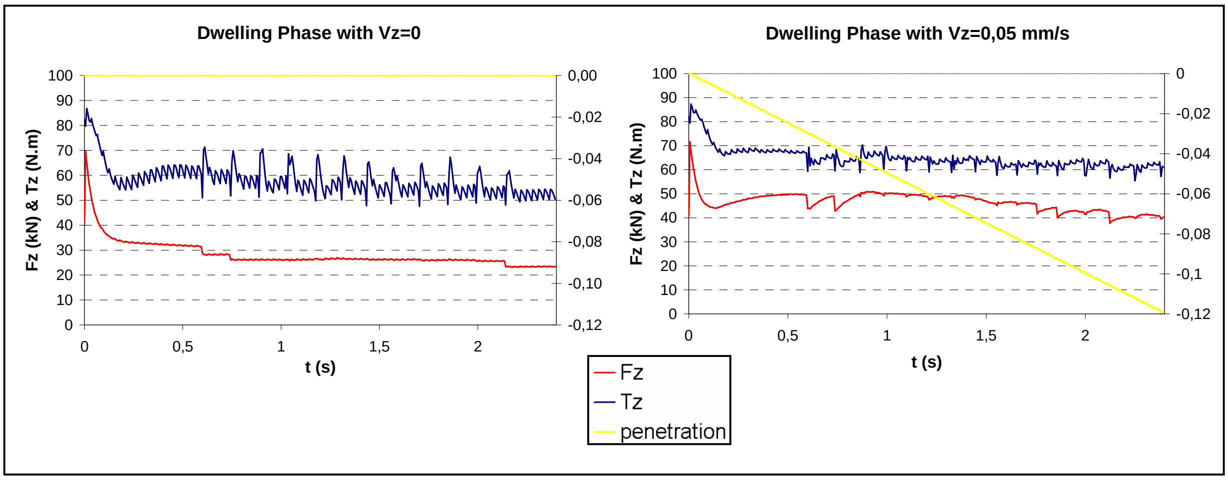


Figure 135: Comparison of vertical forces and torques observed for the dwelling phase simulation with (on right) and without (on left) low plunge velocity.

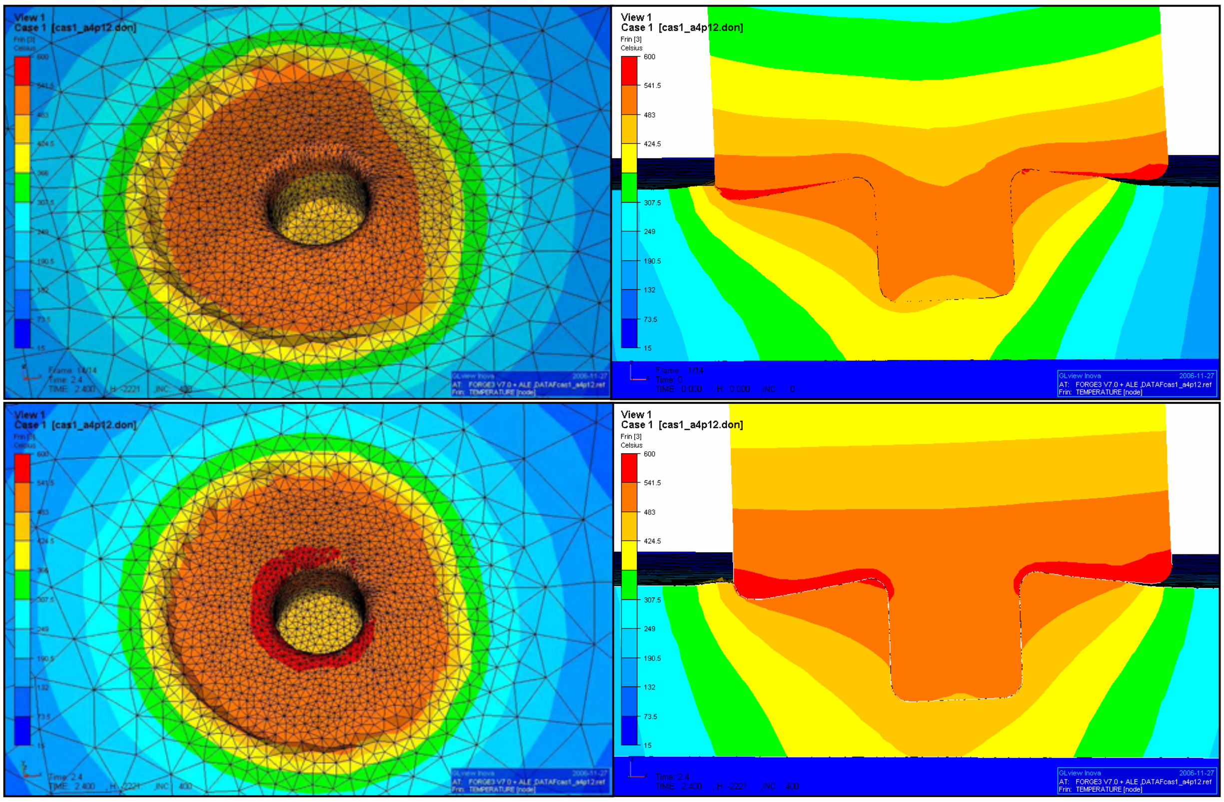


Figure 136: Comparison of the temperature field observed after 2.5 s of dwelling phase simulation with (bottom) and without (on top) low plunge velocity.

3.1.6 Defect simulation

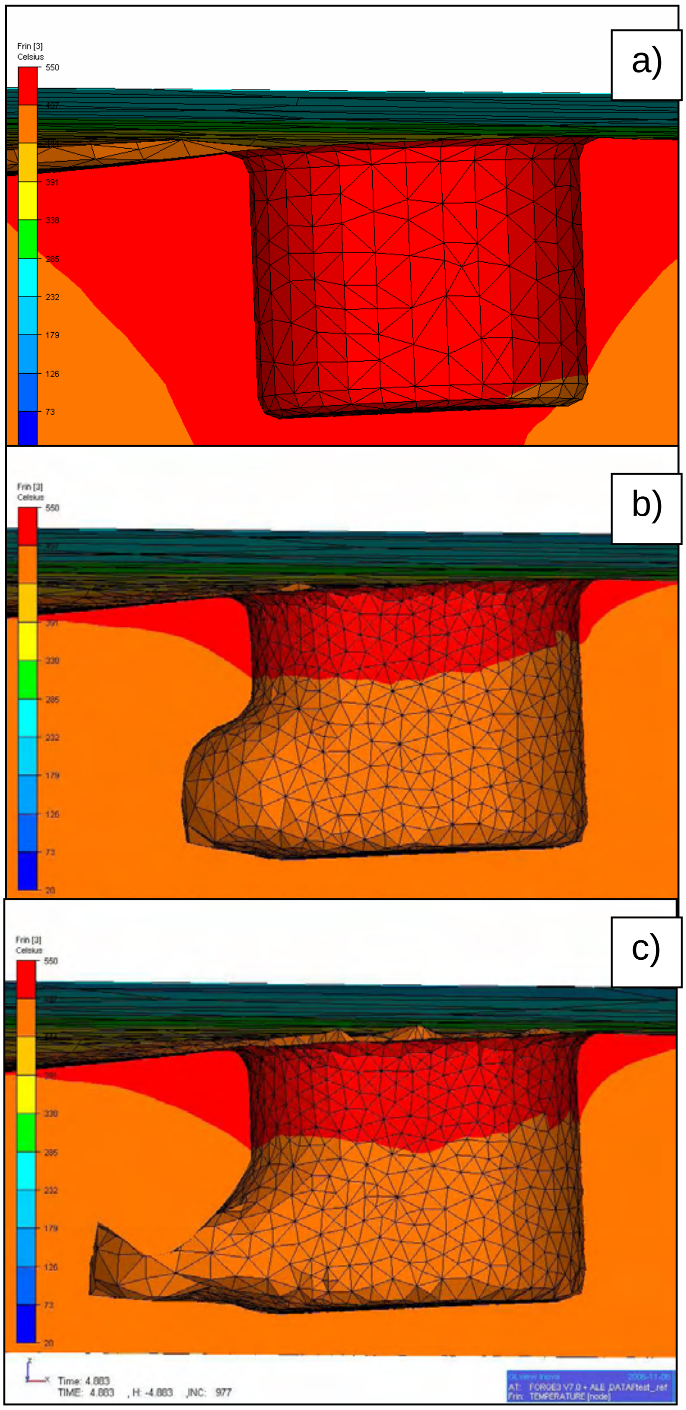


Figure 137: Void formation at the back of the pin (cross section view) a) at t=0s, b) t=2.5s, c) t=5s .

One of the main advantages of the proposed model is the possibility of predicting void formation. The ALE formulation actually allows the separation between the workpiece and the tool.

Figure 137 shows an example of a non-successful deposition in which case a void is formed at the lower advancing trailing side of the probe/matrix interface. It is to be noticed that the voids actually form on the advancing side in the actual process, for large range of alloys. Some voids may form periodically while other may result into so-called “worm holes”. This cross section view also shows that the void tends to be partially filled along with the process. This capacity of simulating voids is actually a very strong point of the proposed approach, because there is not other actually satisfactory method to model this very important defect. Other approaches (within an Eulerian formulation) based on a damage criterion or other analysis of the stress components do not allow stating whether a void is formed or about to be formed and do not make it possible to model non-steady voids.

This model capacity is helpful to:

- Understand under which specific thermomechanical conditions the deposition process properly unfolds.
- Adapt process parameters to avoid void formation, which is suggested as a preliminary criterion for evaluating the success of a weld formation. When the simulation of the material flow leads to a filling of this cavity while reaching a steady state the deposition process is regarded as successful, and is thus contrary to the case where the cavity prevails, which leads to an unsuccessful weld.
- Design new tools (further discussed in part 3.2).

3.1.7 Complete Process Simulation

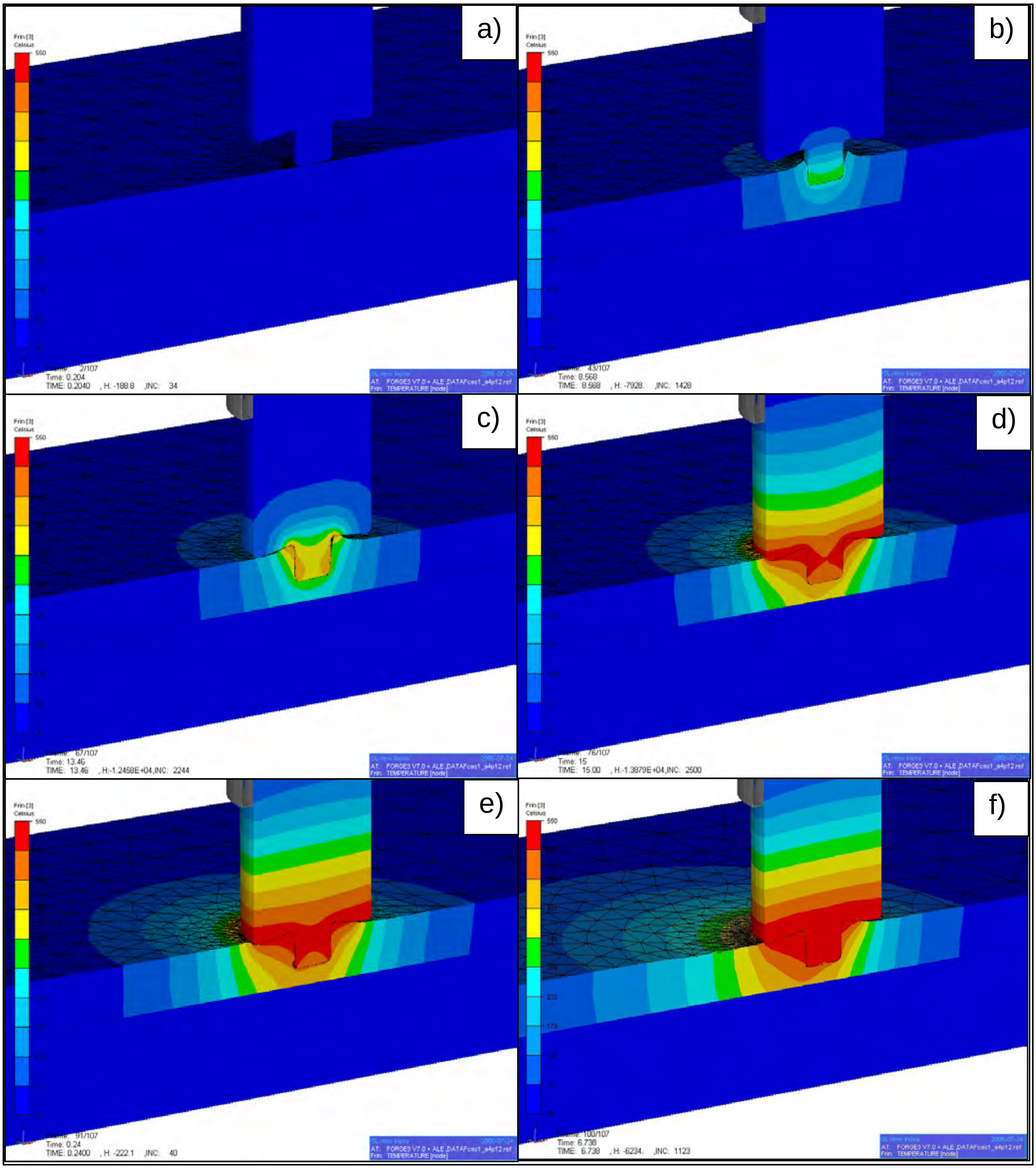


Figure 138: Temperature field during transient states of the whole process (cross section view): a) initial state, b-c) plunging phase, d) end of the plunging phase, e) after dwelling phase, f) transient welding phase.

Figure 138 shows the capacity of the software to simulate the entire process: the temperature field inside the tool, the workpiece and the backing plate is shown in the cross joint section for the plunging phase, after the dwelling phase and at transient welding time.

This simulation is similar to the Eulerian ones which are detailed in section 2.4. The previously identified viscoplastic friction parameters (α_f =0.4 ; p=0.125) are used. The use of an ALE formulation shows the actual feasibility of the weld.

The results are consistent with the experiment; however some improvements and differences with the Eulerian simulations are noticeable. Contrary to them, free surface movements are allowed, therefore the friction area is no longer constant during transient welding, and is strongly dependent on the plunge depth of the tool.

Figure 139 shows the temperature map in the cross joint section after 13 s of welding and the corresponding friction interface. It shows that the shoulder is not fully in contact with the workpiece. Therefore the heat generated by friction is less than in the Eulerian simulation, the material is less softened and the resulting forces are a little bit higher. Forces and torques simulated in the Eulerian and ALE frame are compared in Figure 140.

It is also noticed that some nodes located at the bottom trailing side of the probe appear in blue (not in contact) in Figure 139. It reveals that either the friction coefficient is close to be under evaluated, or that the welding configuration is close to the limits of a good deposition process: although these nodes are not always in contact with the probe, the created little gap is automatically filled while the probe moves along the joint line.

As for void formation, this feature of the proposed approach is very important with respect to alternative (mainly Eulerian formulations) found in literature, where the portion of the surface actually in contact has to be estimated in some way.

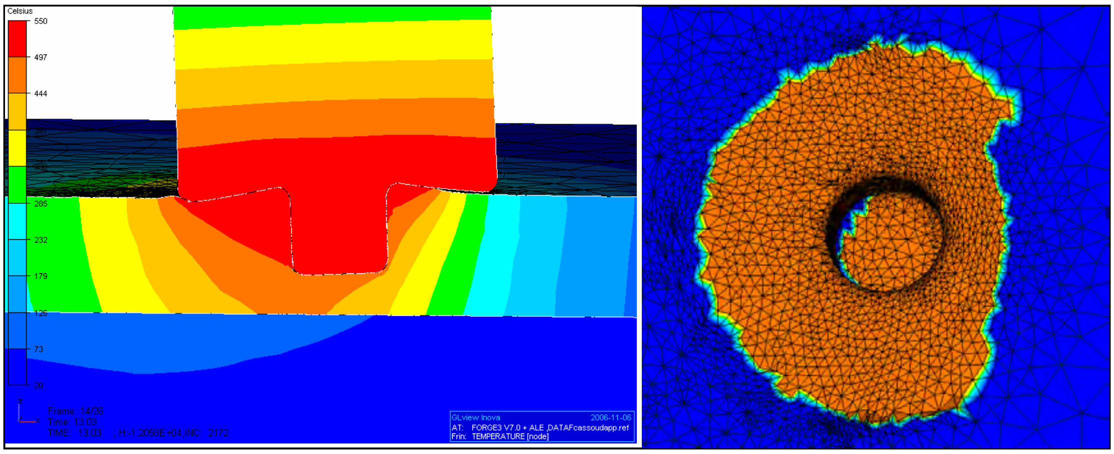


Figure 139: Visualization of the temperature field obtained after 13s of welding with ALE formulation in the cutting joint plan (on left side) and top view of the friction area at the same time (on right side): blue nodes are not in contact.

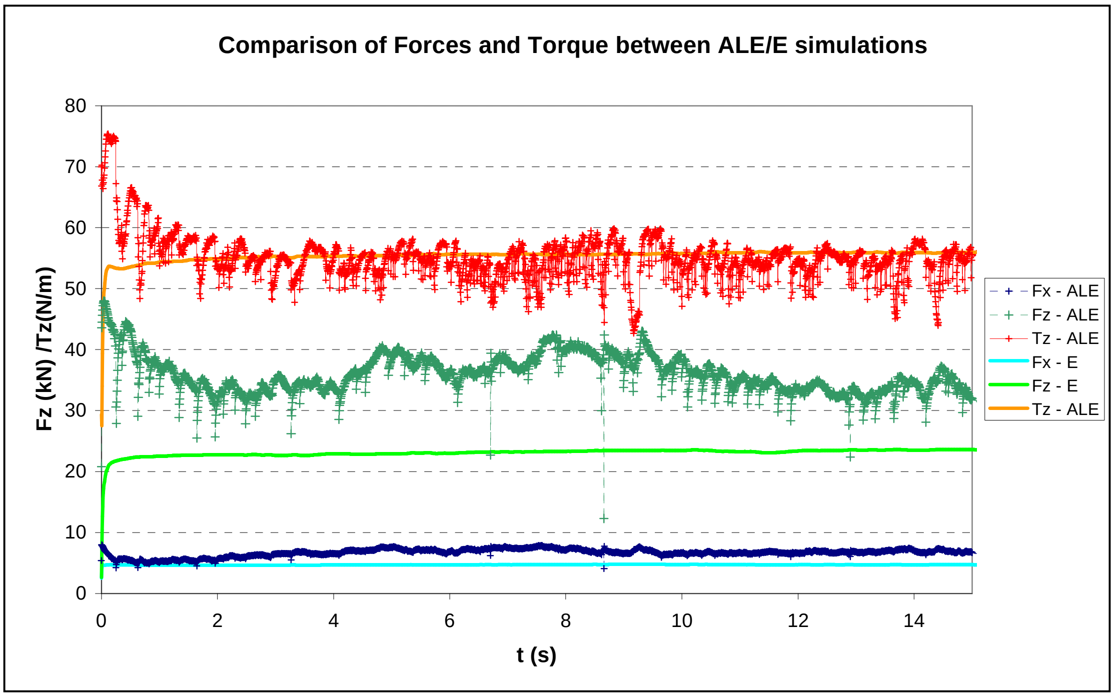


Figure 140: Comparison of resulting forces between the Eulerian and ALE simulations.

It has been noticed that the contact surface change at the beginning of the dwelling and then of the welding phases. In the actual process, the plunge depth is often manually adjusted to handle the resulting variations, but it has not been simulated here, so the comparisons with experimental data should consider this fact.

3.1.8 Key points and encountered difficulties

On the one hand, the ALE model is expected to take into account any movements of the free surface. However, a significant amount of flash of the workpiece material is generated during the plunging or welding experimental phases. It curls up from beneath the tool when the tool penetration into the workpiece is deep enough (see Figure 142). Figure 143 shows the ability of the formulation to simulate very fine details of the process. However, these details are not significant for studying the process and the complete simulation… The simulation of such flashes would over pass the present computational capacity in terms of mesh size that would be necessary to render all these details.

On the other hand, the plunge depth of the tool into the workpiece cannot be artificially modified because it is determinant in FSW. It actually influences the friction area and so loads and temperature of the process.

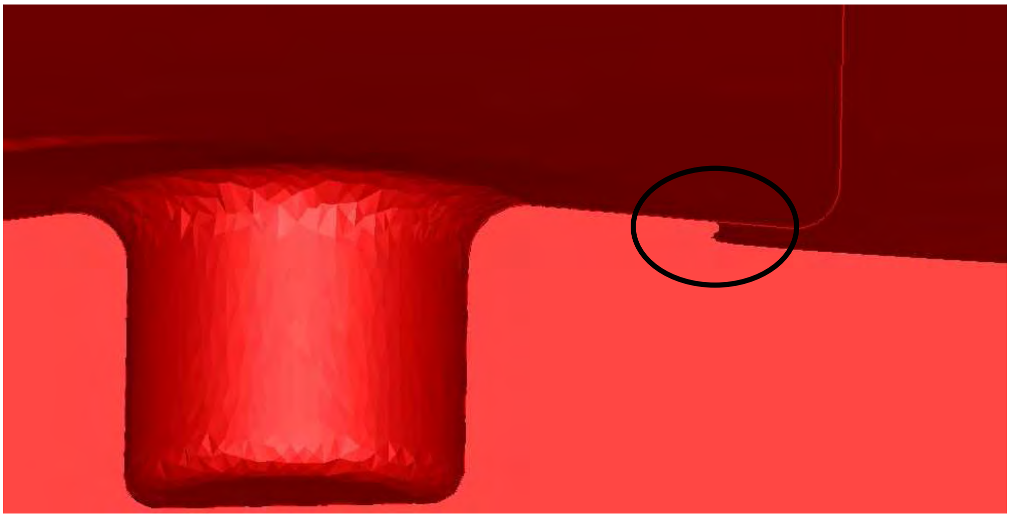


Figure 141: Fold appearance at the end of the plunging phase simulation with a tilted tool (cross section view)

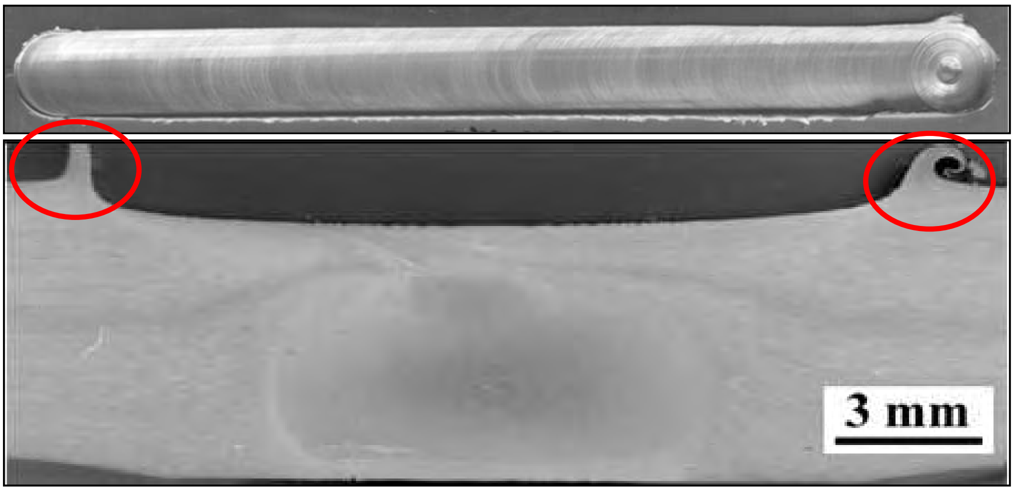


Figure 142: Experimental flash formation during welding (top view and transverse cross section).

Therefore, the numerical model has to propose a good compromise between the accuracy with which the free surfaces movements are described and the revelantness of this phenomenon on the deposition. A judicious choice of the time step (it has to be small enough: ranging between 8.10^{-4}s and 2.10^{-3}s), combined with the tool smoothing procedure and the adaptive ALE mesh formulation, has made it possible to come through these difficulties: the model is accurate enough to take into account possible void formations at the back of the pin, but not to simulate minor phenomena like flash formations at the shoulder periphery.

The problem of flash simulation remains for larger plunge depth of the shoulder, in other words when the actual process exhibits a large volume of flash. Figure 143 shows an example of flash appearance in a welding simulation. A local Eulerian description can then be used to tackle with this problem, in other words, the formulation can be locally ALE or purely Eulerian (or purely Lagrangian) in order to avoid taking unnecessary details of the material flow.

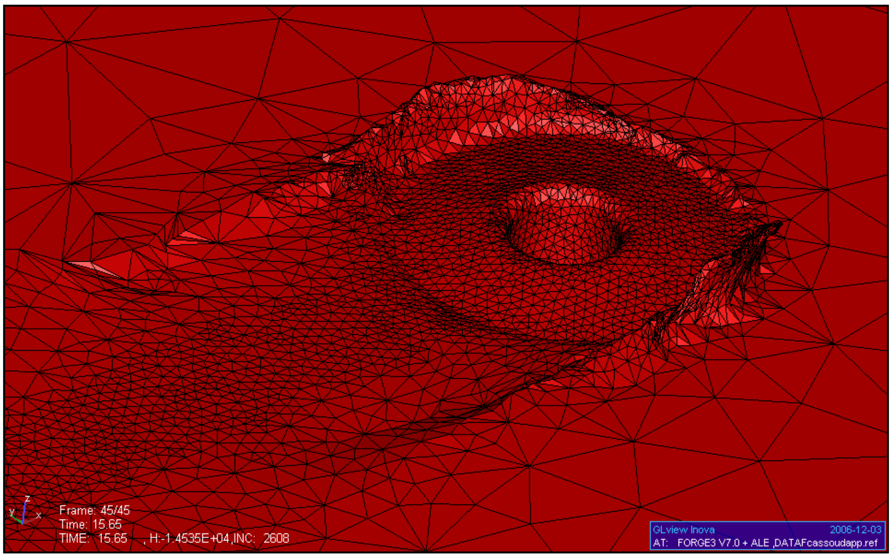


Figure 143: Example of flash occurring during an ALE simulation of transient welding phase.

The other key point to simulate the transient welding phase is the initial state: the initial temperature, but also the initial loading is crucial for establishing a stable deposition process in the model. If one of these two initial conditions is badly configured, the deposition process will not occur during the first time increments. The resulting loss of contact at the trailing edge of the probe leads to lower friction heat creation and void occurrence. Then, even if a stable deposition process is to occur, the calculation will not succeed because of the difficulty to merge two surfaces in a continuous approach, as shown in Figure 144. In fact the ALE mesh velocity calculation allows handling the formation of folds and so of small voids behind the pin (by eliminating the fold surface), but larger voids (as it is the case in Figure 144) are properly modelled and cannot be removed in this continuous frame.

Figure 144: Initial void formation due to bad initial state (cross section view)
The void tends to be field but leads to material interpenetration which is not handled by the code.

3.2 Further investigations and Tooling Design

The tool design plays a very important role in the production of successful joints by FSW. The tool shape and size dictate, to some degree, the material flow and heat generation in the weld zone, which in turn, affects the weld final properties. The simulation provides some informations which can be helpful for tooling design. This section presents some preliminary studies which show the flexibility, robustness and interest of the developed numerical tool.

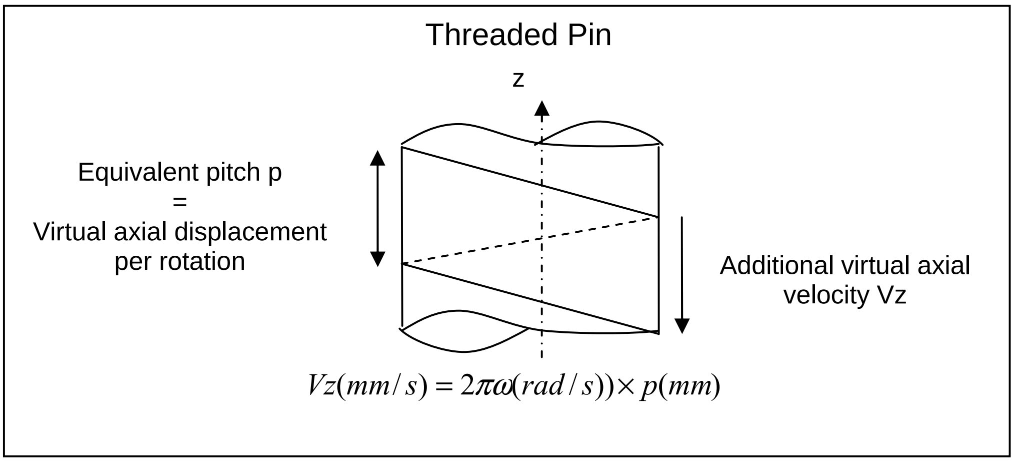
3.2.1 Plunging phase

3.2.1.1 Friction law and threads influence

Two variations of the previous plunging simulation have been studied: one in order to study the influence of the friction law, and the other in order to study the influence of threads.

The simulation has been run keeping the same thermal and mechanical parameters and conditions except that:

- In the first case a Coulomb friction law (see equation (II-55)) has been used with a friction coefficient μ equal to 0.5 (which is representative of a strong friction in hot forging process).
- In the second case, the influence of threads has been modelled. The probe is kept perfectly cylindrical (without visible shape change) since the actual addition of the thread shapes would lead to excessive mesh distortions and size. However their action is modelled by imposing an additional virtual axial velocity to the tool during the friction analysis with the lateral faces of the pin. This additional axial velocity can be correlated to an "equivalent pitch of screw", which is illustrated in Figure 145 and the included equation. This equivalent screw pitch is more representative of the actual material flow.



$$Vz(mm/s) = 2\pi\omega(rad/s)) \times p(mm)$$

Figure 145: Illustration of the additional virtual axial velocity for modelling the influence of the threads.

The observations done in the two cases are discussed together thereafter.

Figure 146 shows the temperature maps in the cross section for the three simulation cases. Notice that the Coulomb friction law provides an increase of temperature at the tip of the pin, while the Norton friction results in a more homogeneous temperature increase along the pin, and the 2 mm equivalent pitch screw provides an increase of the temperature at the contacting lateral sides of the pin.

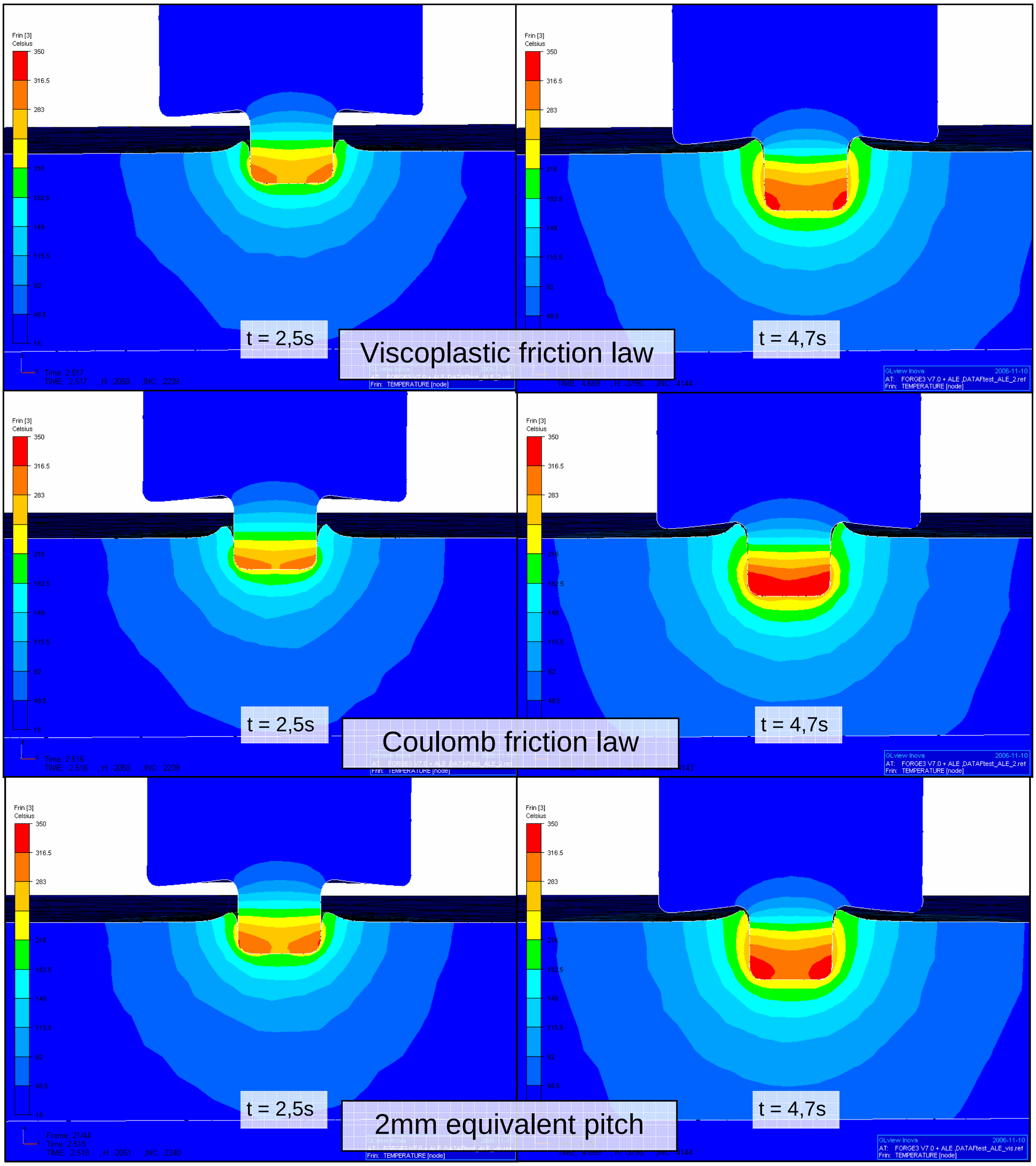


Figure 146: Comparison of temperature maps in the cross section during the plunging phase for viscoplastic friction at top (α_f=0.4 and p=0.125) and Coulomb limited Tresca in the middle (μ=0.5) and screwed pin (2mm equivalent pitch) at the bottom.

This tendency is confirmed in Figures 147 and 148, which show the temperatures at the three tool sensors locations (tip of pin, root and shoulder). In Figure 147, the results of a simulation with a 10 mm equivalent pitch screw are also shown. This new case has been modelled in order to evaluate the influence of threads: threads clearly increase the temperature in the pin.

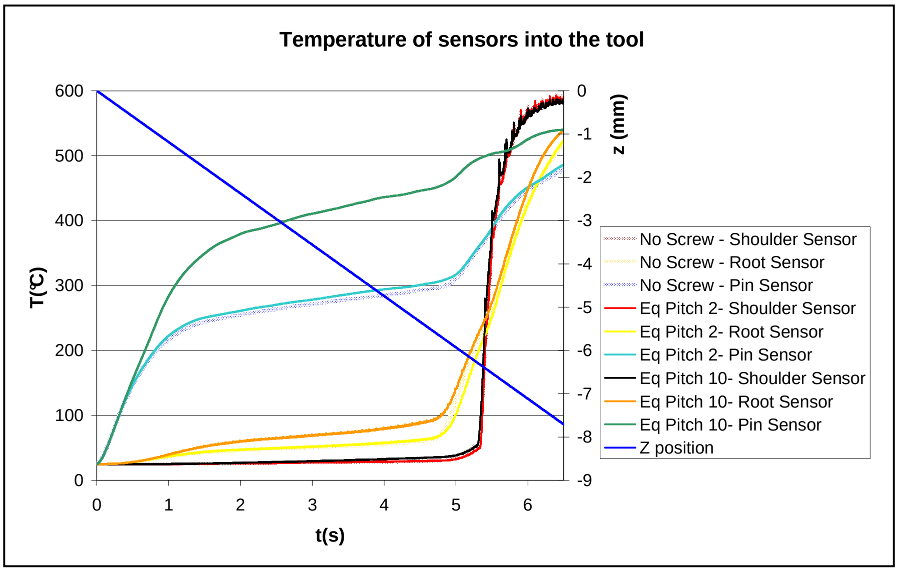


Figure 147:Comparison of temperatures recorded at three locations in tool during a 7s plunge in cases of unthreaded tool, 2mm and 10mm equivalent screw pitch (Norton friction is used with α_f=0.4 and p=0.125).

Figure 148 shows that with the Coulomb law, the heating rate of the shoulder is less than with Norton friction law, but that the final heating is higher. Notice also that the heating of the pin seems greater but more progressive, which compares better with the experimental results shown in Figure 121. Which is fortunate because the threads influence cannot be simply taken into account with the Coulomb law, as it is with the Norton one.

The temperatures registered into the plate at the nearest and the most distant sensors from the tool axis are shown in Figure 149, and agree well with previous observations.

Torques and forces computed in the three configurations are also compared in Figure 150. Note that contrary to the torque, and even though the heating of the material is higher with threaded pin (10 mm equivalent screw pitch) which should result into a lower welding force, the axial load is nearly the same as in the unthreaded configuration because of the vertical reaction force that is generated by the modelled threads. The plunge velocity is maintained constant after the shoulder has contacted the plate, so the pressure under the tool is kept high and therefore the Coulomb friction law results in such high torque and temperature values.

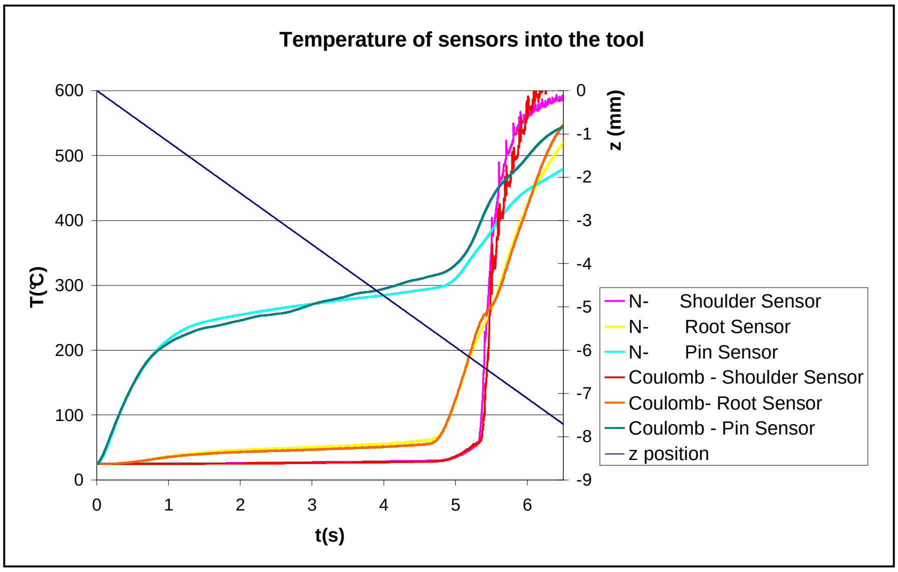


Figure 148: Comparison of temperatures recorded at three locations in the tool during a 7s plunge in cases of Norton friction law (with α_f=0.4 and p=0.125) and Coulomb friction law (μ=0.5) (pin is unthreaded).

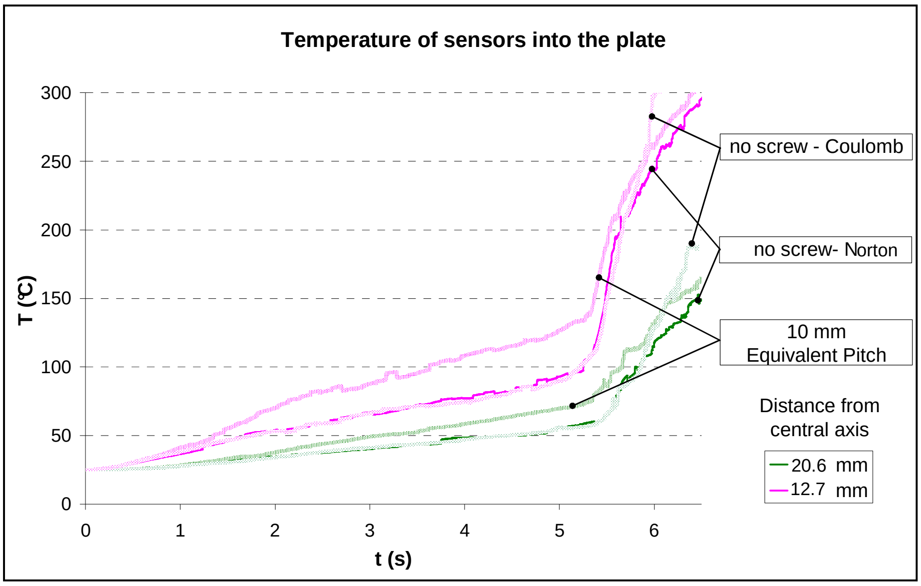


Figure 149: Comparison of temperature values during the plunge phase simulation at sensors located at 20.6 and 12.7 mm from the tool axis in the three different studied cases.

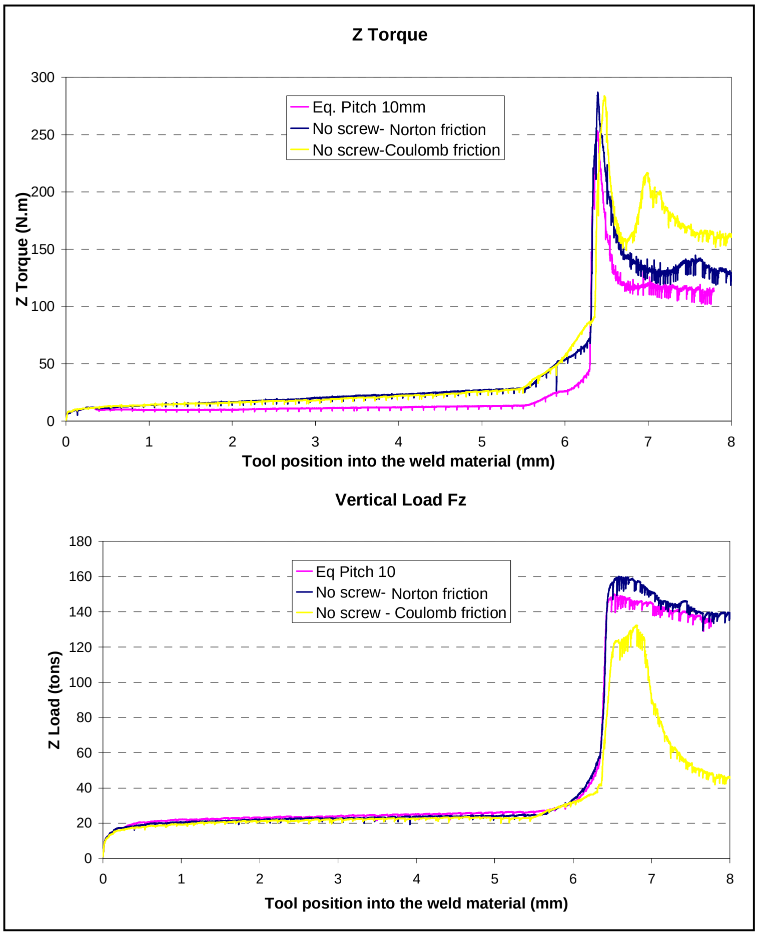


Figure 150: Comparison of torques and axial forces during the plunge phase for the three studied cases: viscoplastic friction law (α_f=0.4 and p=0.125), Coulomb friction law (μ=0.5) and threaded pin (10mm equivalent pitch).

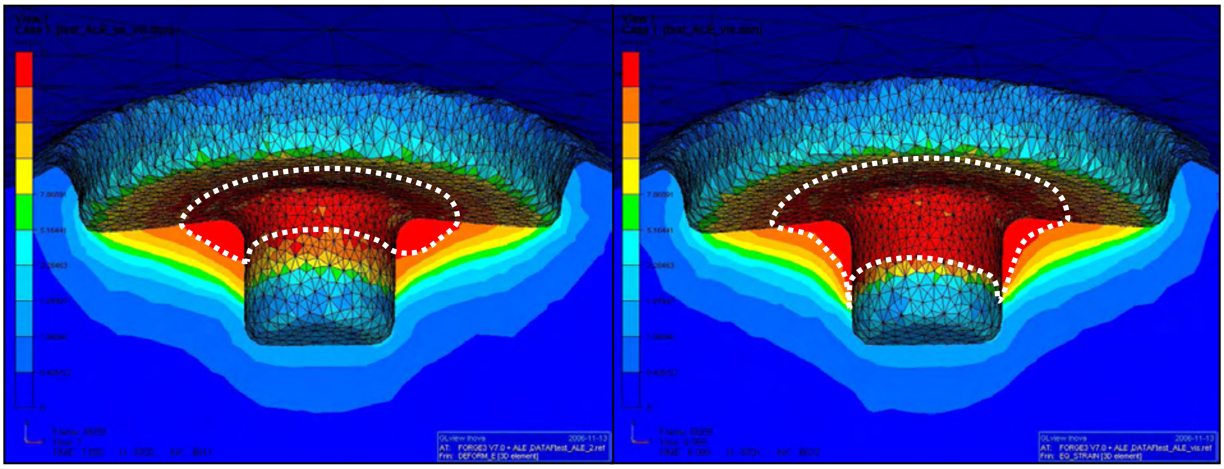


Figure 151: Comparison of the equivalent strain maps in the cross section after a 7 s plunge for unthreaded pin on right and threaded pin on left (10 mm equivalent pitch).

Figure 151 shows that the threads have also an effect on the equivalent strain map: the strain is higher and less concentrated at the root of the pin when this latter is threaded.

3.2.1.2 Discussion

The Coulomb friction law seems to be better adapted to simulate the beginning of the plunge phase: the temperature of the pin increases more progressively and more heat is generated at the base of the pin. However, the comparison to the experimental results indicates that a coefficient of 0.5 is over estimated. Moreover, when the pressure is not important enough, which is the case on the lateral faces of the pin and which could be the case during the simulation of the dwelling phase, the heat generated by friction is under predicted and the effect of threads (modelled as previously proposed) will be under estimated.

Therefore, it is believed that the simulation of the plunging phase would be more predictive with a mixed friction law. The Coulomb limited by Tresca law could be limited up by a viscoplastic Norton law, when the material reaches a predetermined temperature. This could be implemented as follows:

$$
\begin{array}{l}
\bullet\ \text{Coulomb law estimation}\ \begin{cases} \text{if} \quad \left|\mu\sigma_n\right| < \overline{m}\dfrac{\sigma_0}{\sqrt{3}} \quad \text{then} \quad \boldsymbol{\tau}_{1f} = -\mu\sigma_n \dfrac{\Delta\mathbf{v}_s}{\left\|\Delta\mathbf{v}_s\right\|} \\ \text{else if} \quad \left|\mu\sigma_n\right| \geq \overline{m}\dfrac{\sigma_0}{\sqrt{3}} \quad \text{then} \quad \boldsymbol{\tau}_{1f} = -\overline{m}\dfrac{\sigma_y}{\sqrt{3}}\dfrac{\Delta\mathbf{v}_s}{\left\|\Delta\mathbf{v}_s\right\|} \end{cases} \\
\bullet\ \boldsymbol{\tau}_f = \boldsymbol{\tau}_{1f} \\
\bullet\ \text{If } T > T_{\lim} \text{ then} \\
\qquad \bullet\ \text{Norton law estimation : } \boldsymbol{\tau}_{2f} = -\alpha_f K(T,\overline{\varepsilon})\left\|\Delta\mathbf{v}_s\right\|^{q-1}\Delta\mathbf{v}_s \\
\qquad \bullet\ \text{If } \left\|\boldsymbol{\tau}_{2f}\right\| > \left\|\boldsymbol{\tau}_{1f}\right\| \text{ then } \boldsymbol{\tau}_f = \boldsymbol{\tau}_{2f}
\end{array}
\tag{V-2}
$$

However, the stability of such friction model is not guarantied. Another suggestion would be to introduce an evolution of the friction coefficients with temperature, which would also increase the number of parameters to be identified, whereas this identification work is already quite difficult.

3.2.2 Threads influence during welding phase

Practically, threads are introduced to push the material down along the probe in order to avoid the formation of voids.

Two ALE simulations, from transient to steady welding state, have been run with the same model parameters (previously detailed in part 3). For one of them a 10 mm equivalent screw pitch is added. Figure 152 shows the difference observed on the temperature field in the cross joint section after 21 s of welding and in the equivalent strain observed in a transverse cross section just behind the probe.

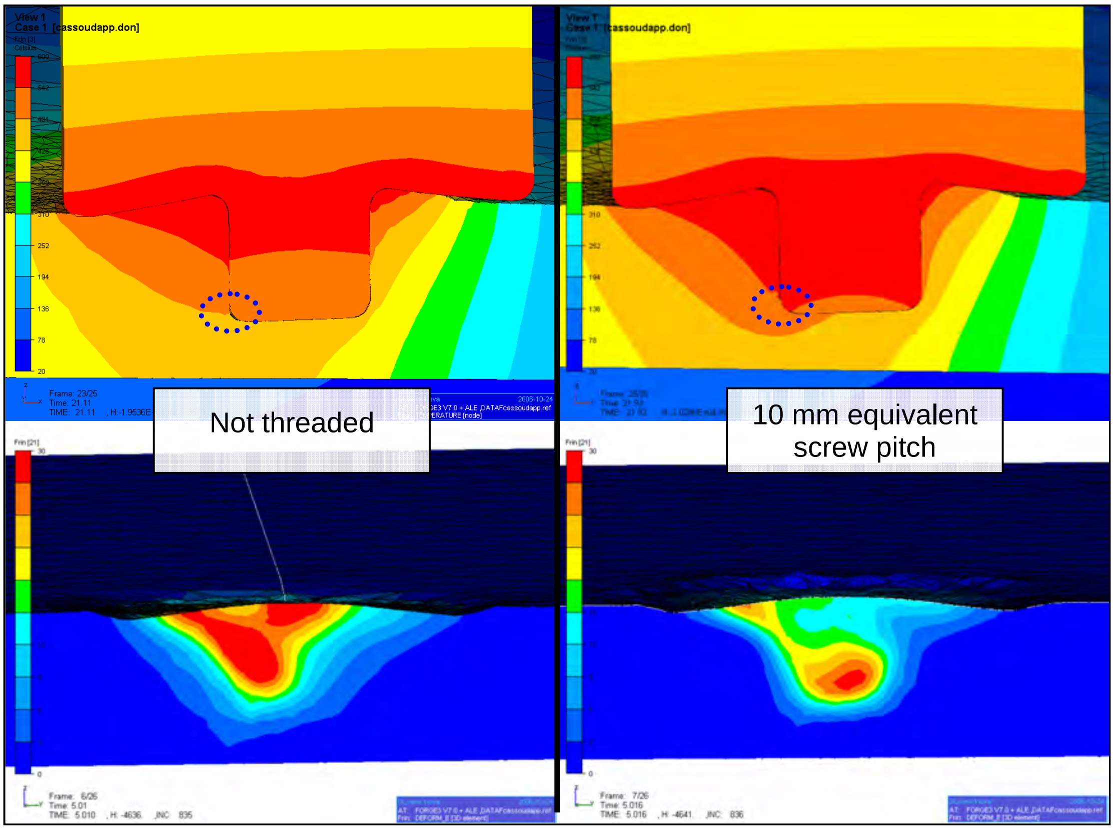


Figure 152: Comparison of the ALE welding phase with threaded and unthreaded pin tool.
At top, temperature after 21s of welding (cross joint section view)
At bottom, equivalent strain at the back of the pin.
* Viscoplastic friction is use with α_f=0.4 and p=0.125

The modelled threads clearly increase the temperature along the pin. But the simulations show that threads also have an effect on the material flow. The resulting higher equivalent strain looks more concentrated at the bottom of the probe, so the so-called nugget zone is clearly visualised. Moreover, the losses of contact, which appear at the bottom trailing edge of the pin (see Figure139) in the case of unthreaded pin, are not observed with the so modelled threads. This is a very important observation because the threads are mainly introduced with the objective of avoiding the void formations, which the simulation confirms.

Figure 153 compares the stream lines for 78 material points (Figure113 shows the locations of the final particles) in the two different simulation cases. It is very likely that the material flow within the nugget consists of several independent deformation processes. In the case of a threaded tool the material particles are quickly dragged from the top surface to the bottom surface; the particles close to the top surface are less trapped under the shoulder than in the unthreaded case. It correlates well with the fact that the equivalent strain close to the top surface is lower in case of threaded tool (see Figure 152).

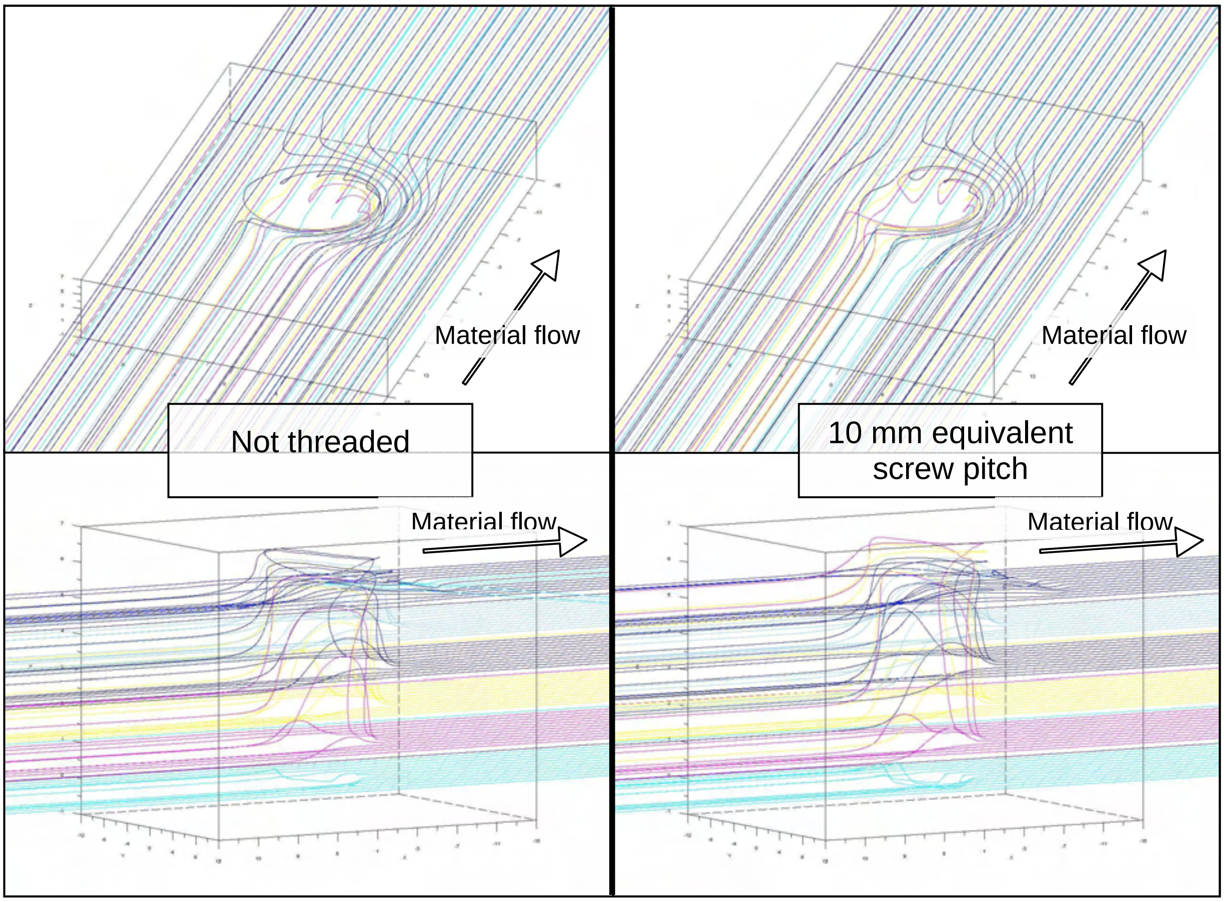


Figure 153: Comparison of stream lines simulated with and without threads.

Threads also have an effect on the forces involved in the process. Table 154 compares experimental and numerical averaged results obtained with threaded and unthreaded tool.

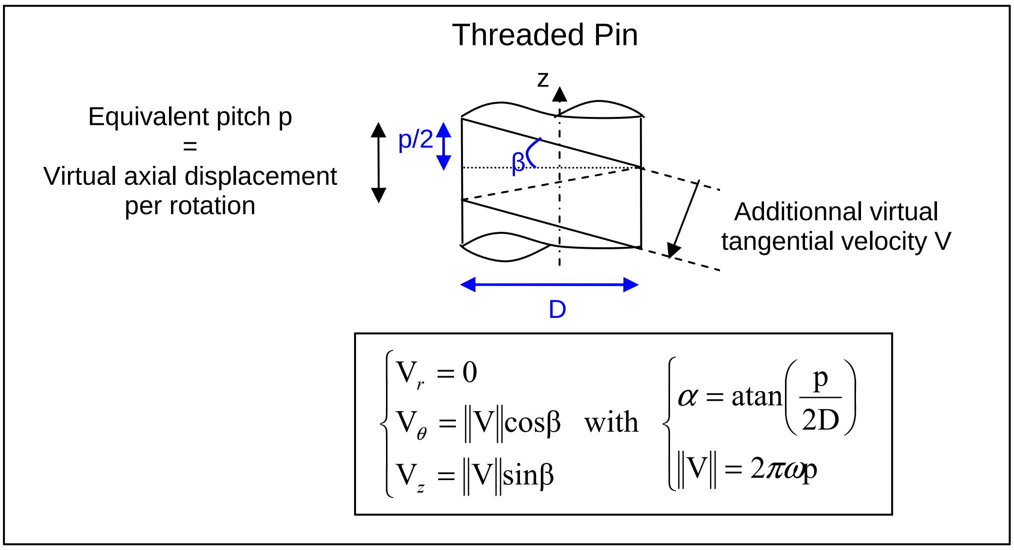
The traverse force is lower in the case of threaded pin. This result is observed in the experiment and in the simulation. This decrease is not observed on the vertical plunging force. But conclusions have to be carefully taken for the evolution of the vertical force because of the scatterness of the experimental results: such an increase of the plunging force due to the threads is not observed for other experimental configurations with different transverse speed or pin length.

	Traverse Force Fx (kN)		Vertical Force Fz (kN)		(Equivalent) Motor Power (kW)	
	Experimental	Numerical	Experimental	Numerical	Experimental	Numerical
Unthreaded tool	3.9	6.8	24.8	33.9	5.4	3.7
Threaded tool	2.7	6.2	30.3	30.8	5.3	2.9

Figure 154: Comparison of Forces and equivalent Motor Power for threaded and unthreaded tool resulting from experimental and numerical simulation

Note that the model results in a lower equivalent motor power in the case of threaded pin compared to the unthreaded case, contrary to the experiment in which it stays quasi-constant. The numerical results are highly sensitive to the variations of the contact area (these are ALE simulations) but it is nearly the same for the two simulations. So this difference may come from the fact that the effect of the threads on the motor power is not properly taken into account in the utilised model (i.e. some corrections in the formula are required).

Experiments show that threads also have an influence on the torque: they also generate a frictional component in the θ direction (within a cylindrical frame). Thus the additional virtual velocity modelling the threads should not be exactly axial but modelled as orthogonal to threads in planes which are tangent to the lateral pin surface. Figure 155 illustrates this proposed modification. The resulting angle β could also be adjusted independently from p, but once again, this would be one more coefficient to identify…



$$\begin{cases} V_r = 0 \\ V_\theta = \|V\|\cos\beta \\ V_z = \|V\|\sin\beta \end{cases} \quad \text{with} \quad \begin{cases} \alpha = \operatorname{atan}\left(\frac{p}{2D}\right) \\ \|V\| = 2\pi\omega p \end{cases}$$

Figure 155: Illustration of the modified additional virtual velocity for modelling the influence of the threads.

Threads clearly facilitate the deposition process on the trailing bottom side of the probe. Their action can be necessary in case of long pin. An ALE simulation has been run with a 10.16mm (0.4in) pin length with a 10mm equivalent screw pitch. For such a long pin only threaded pin have provided successful welds. Figure 156 shows the feasibility of the simulation: action of threads lead to a good deposition process even if some of the nodes lost the contact with the trailing bottom side of the pin. Without the thread modelisation, a void appears.

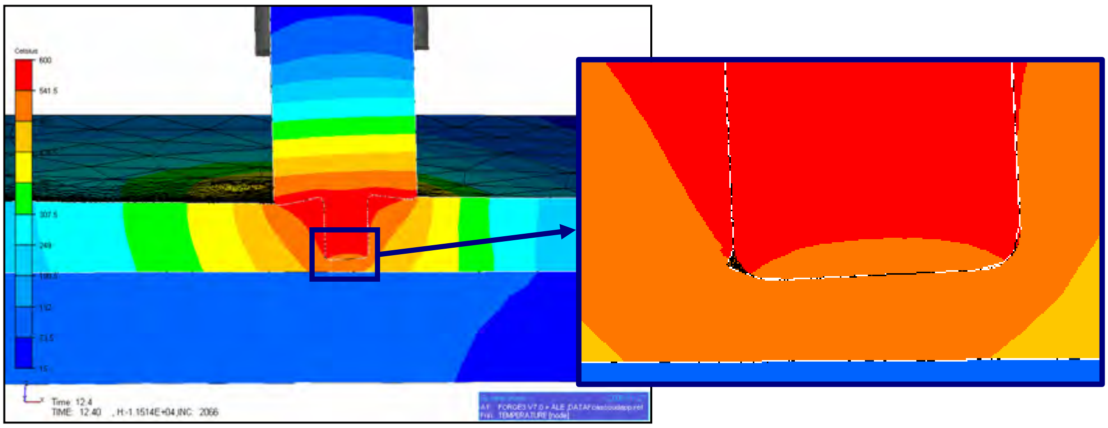


Figure 156: Temperature map (cross section view) after 12 s of an ALE welding simulation. The contact is lost for some of the nodes at the bottom trailing side of the pin while the action of threads is simulated through a 10 mm equivalent screw pitch.

Note also that for such a long pin a double nugget is experimentally observed. Figure 157 compares the equivalent strain in a transverse cross section just behind the pin to an experimental transverse cut. The numerical double nugget observation would necessitate a very fine mesh; however an increase of the nugget size is observed in the current simulation.

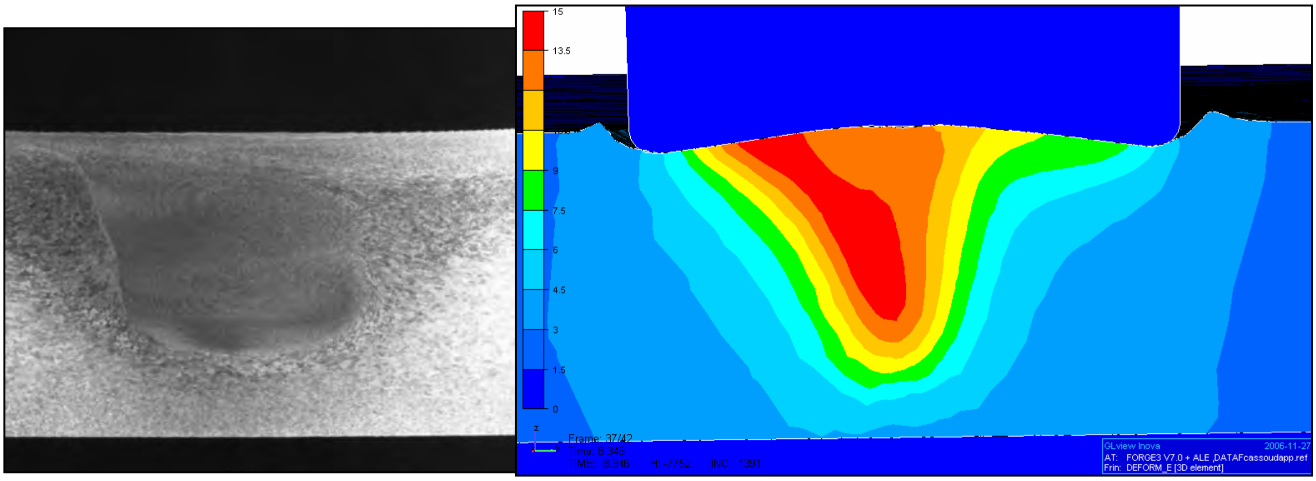


Figure 157: Comparison of experimental transverse cut and simulated equivalent strain in a transverse cutting plane at the back of the pin.

3.2.3 New tool geometry

The shoulder of the tool, which is regarded to be the main source of heat generation, is typically concave in shape. This was thought to aid in weld consolidation by forcing the softened material to remain in the welding zone as the tool moves along the joint line. But tool design is still in progress and evolves quickly. Figure 158 shows a new popular tool design used for steel welding.

In order to show the feasibility and flexibility of the model, this new tool has been approximatively modelled. It has a convex threaded shoulder and a conical threaded pin. Contrary to previous ones, this new tool is not tilted for welding (tilt and concavity are not necessary according to the threads on shoulder).

Two ALE simulations have been run with this new tooling design: one taking into account the threads and the other one considering the tool with a smooth surface (no threads modelisation). The same welding parameters as in the previous simulations are used: the tool rotates at 650 rpm and the material inlet flow rate is 3.4 mm/s (8 ipm).

Threads are modelled as before by an additional virtual velocity, which is tangent to the tool surface and oriented perpendicular to the experimental threads. This virtual velocity is taken into account by nodes which are in contact not only with the pin but also with the shoulder. It is illustrated in Figure 158.

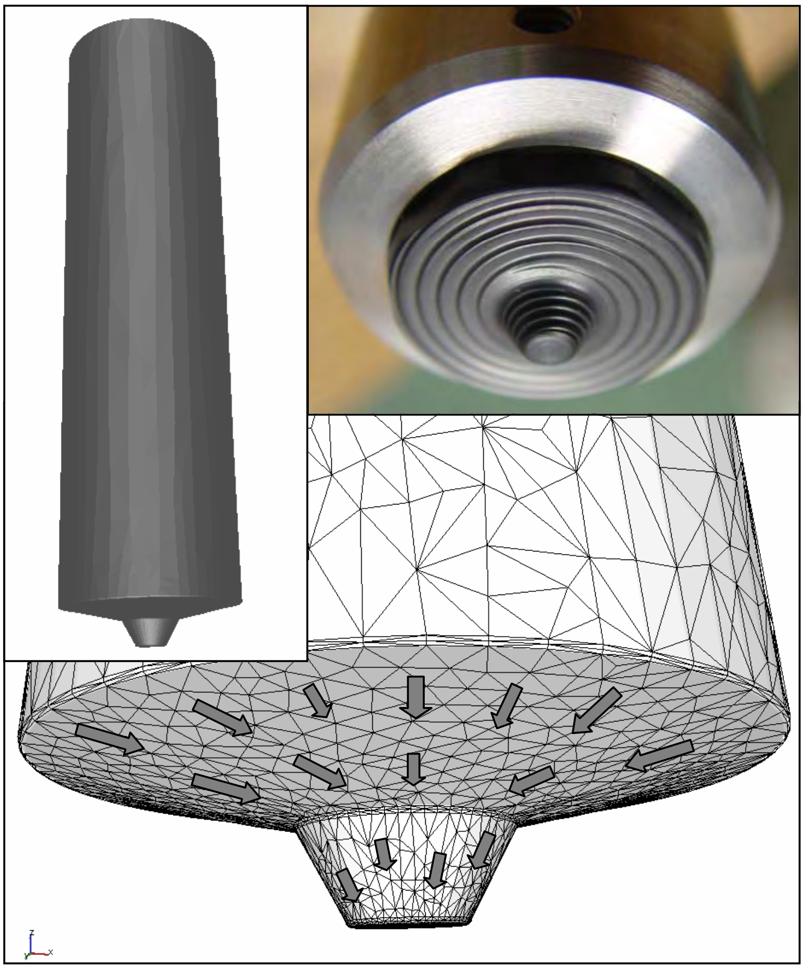


Figure 158: New tool shape: experimental tool (on top right) and modelled one (The arrows illustrate the added virtual velocities for threads modelling).

The simulation with the unthreaded tool rapidly stops: a void appears at the trailing advancing side of the probe. Figure 159 shows the initial contact area for the two simulations, the loss of contact which immediately occurs with the unthreaded tool and the “foot print” of the tool after 17 s of welding with the treaded tool (no void is generated).

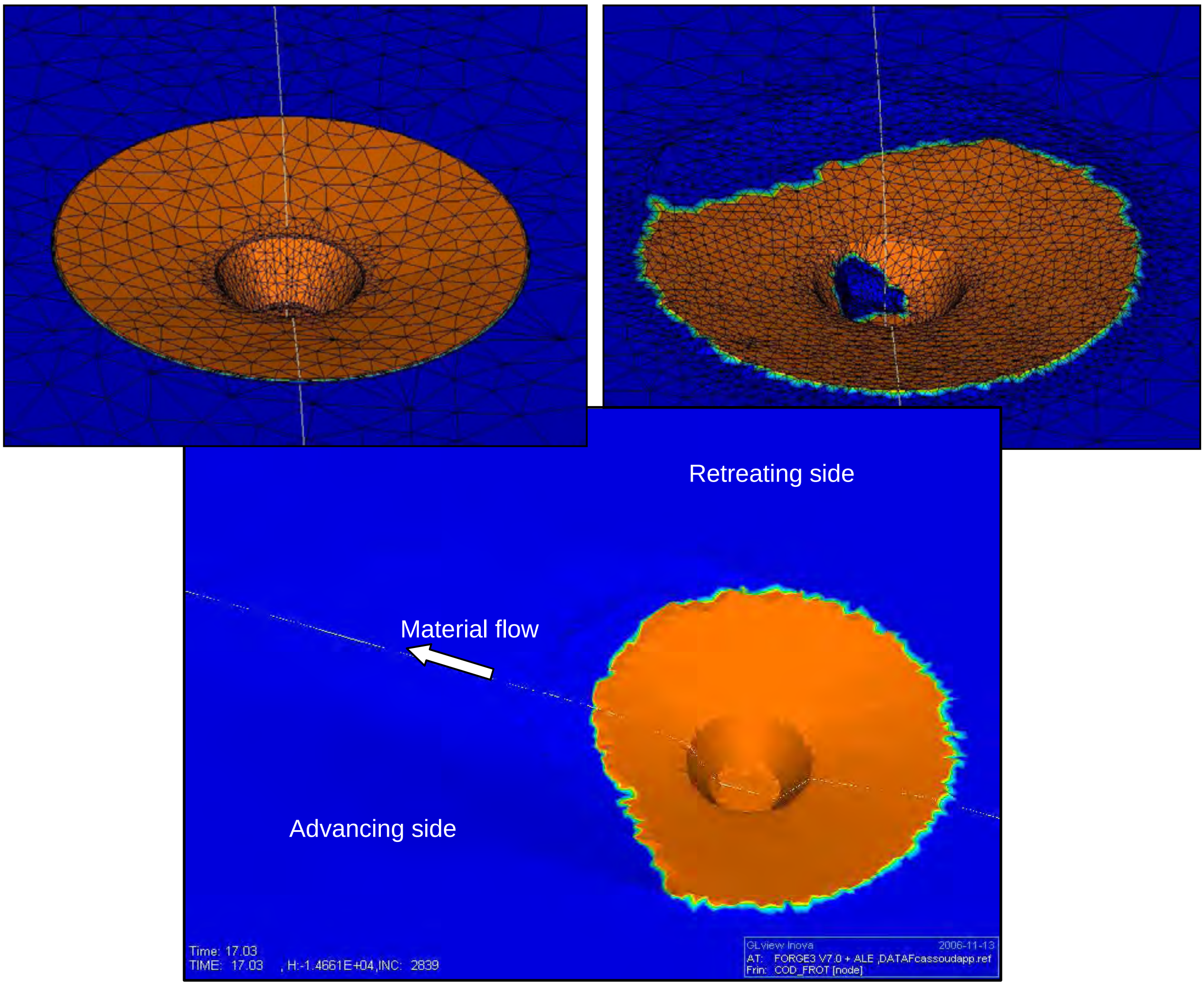


Figure 159: "foot print" (contact area) comparison. At t=0 at the top left, after 2 s with the unthreaded tool at the top right, and after 17 s of welding simulation with the threaded tool at the bottom (blue nodes are not in contact with the tool).

Figure 160 shows the stream lines and the equivalent strain observed in a cutting transverse plane located just behind the pin. The temperature field resulting from the simulation is shown in Figure 161.

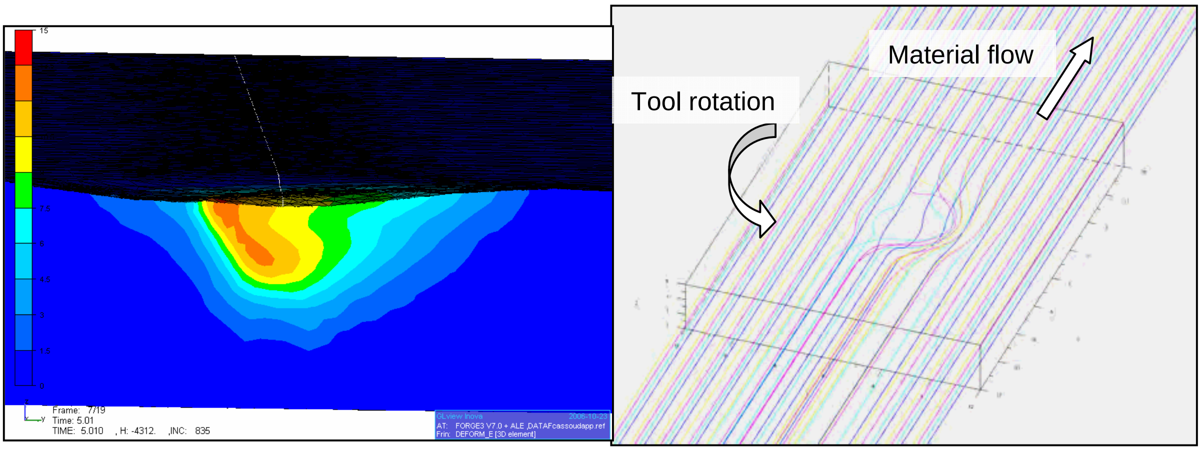


Figure 160: Interesting results with new tool design:
Equivalent strain in transverse cross section at the back of the probe on the left and material stream lines computed after 17 s of welding simulation on right

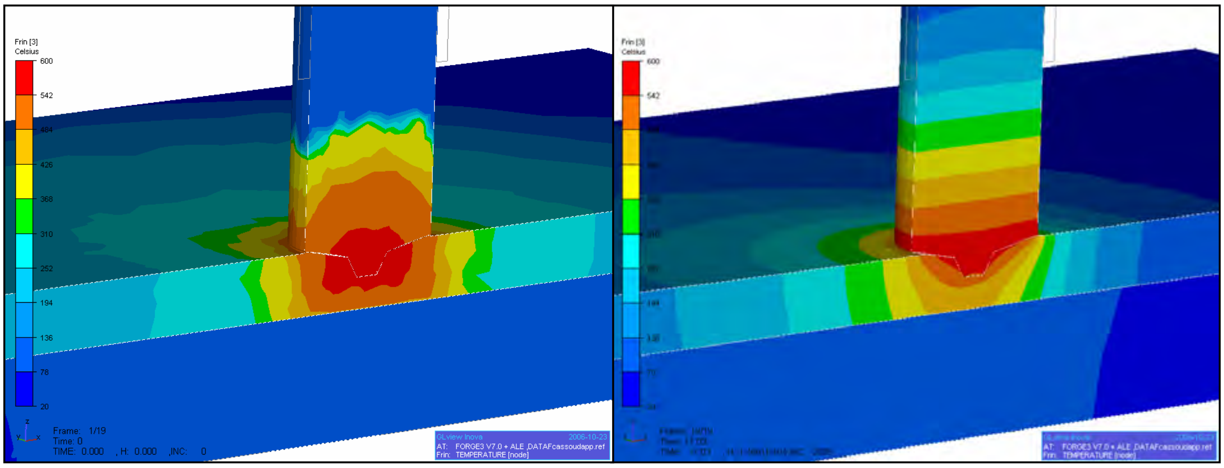


Figure 161: Temperature map in the cross joint section: initial state (a-priori built) and after 17s of welding simulation (on right).

All these results are not subjected to comparison with experimental ones. But they already show the feasibility of the simulation with a very different kind of tool design. They also show the kind of information that the simulation could provide to help in tooling design.

Conclusions and prospects

This work has contributed to a numerical tool derived from the Forge3® software for the simulation of the different phases of FSW. It is based on the development of an adaptive ALE formulation using a splitting method, in which several enhancements are proposed in particular for the variable transfer, the mesh velocity calculation, and the contact algorithm.

The efficiency of the different implemented transfer techniques, with nodal or element patches, with or without equilibrium constraint, has been evaluated. The comparison with respect to more classical methods has shown a significant increase of accuracy. The second order expansion method using the patch recovery technique has provided the best results, for both nodal and element variables transfer.

The calculation of the mesh velocity lies on the combination of two criteria: the first one is based on geometrical considerations, while the second is adaptive and derived from discretization error estimation. The method allows the mesh to be continuously adapted and of high quality at each time increment, thus providing better computational accuracy.

Complex developments have been also necessary to decrease the volume diffusion due to numerical smoothing of boundary singularities, which is one of the main encountered problems when applying an ALE formulation to industrial process with arbitrary and complex shapes. Singular geometrical angles are so automatically tracked and preserved inside the mesh velocity calculation procedure.

In the same manner, a new contact smoothing method has been developed, and provides a relatively simple manner to smooth complex shapes that are discretized by facets. It allows for the existing contact algorithms to be used without many changes into the available finite element code. It only requires transforming the contact surface in two shifted ones, and then two contact analyses (one for each transformed surface). The obtained results have shown the quality of the smoothing, as well as its possible applications in metal forming.

The ALE formulation, combined with full adaptive remeshing and tool smoothing procedures, has proved to be quite robust and efficient: several different processes have been successfully simulated, such as high speed machining and rolling. The first simulation of friction stir welding using the ALE formulation have underlined the fact that the combination of good material data, initial conditions (temperature and pressure) and friction modelling are crucial to successfully simulate the welding phase. The model has allowed for calibrating a first couple of friction coefficients, within a viscoplastic friction model, so providing a quite good agreement with respect to observed experimental forces at steady welding state. The computed temperature fields also compare well with the experiments.

One interest of this ALE model is that the simulation of both the stationary welding and the transient phases are possible. The model has shown that the process parameters can lead to successful welding simulation or void formation at the beginning of the process.

The simulation of the transient phases is important to understand how the material flows. The model can provide this understanding by computing the material stream lines and the particles thermo-mechanical history. The model flexibility has finally proved to be suitable for different tool design studies.

Further work has to be done in order to make the model fully predictive. On the one hand, material data, constitutive coefficients, and friction parameters need to be better identified: simulations results should accurately compare experimental ones in terms of forces, temperature and process singularities, for different process parameters, conditions and materials. On the other hand, several enhancements (continuous transfer method, improved treatment for computation of the domain boundary velocity, reduction of the computational time) could be brought to the actual numerical model, in order to increase its efficiency. These issues should make the model able to confirm or optimize the welding feasibility for a given combination of process parameter. It should finally be used for new tooling design and wear study.

Bibliography

1. McClure, J.C., et al., *A thermal model of friction stir welding.* 5th International Conference: Trends in Welding Research, Pine Mountain,GA,USA, 1-5 June 1998, 1999. **590-595**.
2. Russell, M.J., H.R. Shercliff, and P.L. Threadgill, *Development and application of an analytical process model for friction stir welding.* TMS 2001 Annual Meeting Aluminum Automotive and Joining Symposia, New Orleans,LA,USA, 12-14 Feb. 2001, 2001. **225-234**.
3. Feng, Z., J.E. Gould, and T.J. Lienert, *A heat flow model for friction stir welding of aluminum alloys.* Second Symposium on Hot Deformation of Aluminum Alloys II, Rosemont,IL,USA, 11-15 Oct. 1998, 1998. **149-158**.
4. Gould, J.E. and Z.L. Feng, *Heat flow model for friction stir welding of aluminum alloys.* Journal of Materials Processing & Manufacturing Science, 1999. **7**(2): p. 185-194.
5. Schmidt, H., J. Hattel, and J. Wert, *An analytical model for the heat generation in friction stir welding.* Modelling Simul. Mater. Sci. Eng., 2004. **12**: p. 143-157.
6. Khandkar, M.Z.H., J.A. Khan, and A.P. Reynolds, *Input torque based thermal model of friction stir welding of Al-6061.* 6th International Conference: Trends in Welding Research, Pine Mountain,GA,USA, 15-19 Apr. 2002, 2003. **218-223**.
7. Chen, C.M. and R. Kovacevic, *Finite element modeling of friction stir welding-thermal and thermomechanical analysis.* International Journal of Machine Tools & Manufacture (UK), 2003. **43**(13): p. 1319-1326.
8. Lawrjaniec, D., et al., *Numerical simulation of friction stir welding.* THERMEC 2003: International Conference on Processing & Manufacturing of Advanced Materials, Madrid,Spain, 7-11 Jul 2003, 2003. **426-432**(Part 4): p. 2993-2998.
9. Seidel, T.U. and A.P. Reynolds, *Two-dimensional friction stir welding process model based on fluid mechanics.* Science and Technology of Welding and Joining (UK), 2003. **8**(3): p. 175-183.
10. Chao, Y.J., X. Qi, and W. Tang, *Heat transfer in friction stir welding: Experimental and numerical studies.* Journal of manufacturing science and engineering, 2003. **125 (1)**: p. 138-145.
11. Ulysse, P., *Three-dimensional modeling of the friction stir-welding process.* International Journal of Machine Tools and Manufacture, 2002. **42**(14): p. 1549-1557.
12. Long, T., et al., *A friction stir welding model using computational fluid dynamics.* Third Symposium on Hot Deformation of Aluminum Alloys III as held at the 2003 TMS Annual Meeting, San Diego,CA,USA, 2-6 Mar. 2003, 2003. **299-312**.
13. Colegrove, P.A. and H.R. Shercliff, *Experimental and numerical analysis of aluminium alloy 7075-T7351 friction stir welds.* Science and Technology of Welding and Joining, 2003. **8**(5): p. 360-368.

14. Colegrove, P.A., H.R. Shercliff, and P.L. Threadgill. *Modelling and development of the Trivex Friction Stir Welding Tool (4th International Symposium on Friction Stir Welding)*. 2003.

15. Xu, S., et al., *Finite element simulation of material flow in friction stir welding.* Science and Technology of Welding and Joining, 2001. **6**(3): p. 191-193.

16. Schmidt, H. and J. Hattel, *A local model for the thermomechanical conditions in friction stir welding.* Modelling and Simulation in Materials Science and Engineering, 2005. **13**(1): p. 77-93.

17. Coupez, T., N. Soyris, and J.-L. Chenot, *3-D finite element modelling of the forging process with automatic remeshing.* Journal of Materials Processing Technology, 1991. **27**(1-3): p. 119-133.

18. Stoker, H.C., *Developments of the Arbitrary Lagrangian-Eulerian Method in non-linear Solid Mechanics*, ed. Anonymous. 1999: Enschede, The Netherlands. 152.

19. Perchat, E., *MINI-élément et factorisation incomplètes pour la parallelisation d'un solveur de Stokes 2D: application au forgeage*, 2000, ENSMP: Centre de Mise en Forme des Matriaux (Sophia-Antipolis).

20. Zlà¡mal, M., *Finite element method in physical and technical applications.* Computer Physics Communications, 1980. **20**(1): p. 37-42.

21. Boussetta, R., T. Coupez, and L. Fourment, *Adaptive remeshing based on a posteriori error estimation for forging simulation.* Computational Metal Forming, 2006. **195**(48-49): p. 6626-6645.

22. Gruau, C. and T. Coupez, *3D tetrahedral, unstructured and anisotropic mesh generation with adaptation to natural and multidomain metric.* Unstructured Mesh Generation, 2005. **194**(48-49): p. 4951-4976.

23. Chenot, J.L., L. Fourment, and K. Mocellin, *Numerical treatment of contact and friction in FE simulation of forming processes.* Journal of Materials Processing Technology, 2002. **125-126**: p. 45-52.

24. Chamoret, D., et al., *New smoothing procedures in contact mechanics.* Selected Papers from the Second International Conference on Advanced Computational Methods in Engineering (ACOMEN 2002), 2004. **168**(1-2): p. 107-116.

25. Lee, M.C. and M.S. Joun, *Adaptive triangular element generation and optimization-based smoothing, Part 1: On the plane.* Advances in Engineering Software, 2008. **39**(1): p. 25-34.

26. Borouchaki, H., et al., *Surface mesh enhancement with geometric singularities identification.* Unstructured Mesh Generation, 2005. **194**(48-49): p. 4885-4894.

27. Mocellin, K., *Contribution à la simulation numérique tridimensionnelle du forgeage à chaud: étude du contact et calcul multigrille*. 1999, ENSMP: CEMEF Centre de Mise en Forme des Matriaux.

28. Delalondre, F., S. Guerdoux, and L. Fourment, *3D Simulation of Adiabatic Shear Bands in High Speed Machining.* AIP Conference Proceedings, 2007. **908**(1): p. 1137-1142.

29. Donea, J., S. Giuliani, and J.P. Halleux, *An arbitrary lagrangian-eulerian finite element method for transient dynamic fluid-structure interactions.* Computer Methods in Applied Mechanics and Engineering, 1982. **33**(1-3): p. 689-723.

30. Magnin, B., *Modelisation du remplissage des moules d'injection pour polymères thermoplastiques par une méthode Eulérienne-Lagrangienne Arbitraire*. 1994, Ecole Nationnale Supérieure des Mines de Paris (Sophia-Antipolis).

31. Wang, H., et al., *A Family of Eulerian-Lagrangian Localized Adjoint Methods for Multi-dimensional Advection-Reaction Equations.* Journal of Computational Physics, 1999. **152**(1): p. 120-163.

32. Bigot, E., *Simulation tridimensionnelle du remplissage de corps minces par injection*. 2001, ENSMP: CEMEF Centre de Mise en Forme des Matriaux.

33. Traoré, K., et al., *Three-dimensional finite element simulation of ring rolling*. 2001, NUMIFORM'2001, 7th Int. Conf. on Numerical Methods in Industrial Forming Processes: Toyohashi, Japon. p. K. Mori (Ed.), Balkema, Lisse,pp 595-600.

34. Bellet, M. and V.D. Fachinotti, *ALE Method for solidification modelling.* Computer Methods in Applied Mechanics and Engineering 2004. **193**(39-41): p. 4355-4381.

35. Maury, B., *Characteristics ALE Method for the Unsteady 3D Navier-Stokes Equations with a Free Surface.* Comp. Fluid Dyn., 1996. **6**: p. 175-188.

36. Boussetta, R., *Estimateurs d'erreur et ramaillage adaptatif: application à la simulation 3D des procédés de mise en forme des matériaux*. 2005, ENSMP: CEMEF Centre de Mise en Forme des Matriaux.

37. Huétink, J. *Analysis of metal forming processes based on a combined Eulerian-Lagrangian finite element formulation*. in *International conference on Numerical methods in industrial Forming Processes*. 1982: Pineridge Press.

38. Stocker, H.C., et al., *Investigation of an improved ALE formulation using cases with analytical solutions.* 1997.

39. Traoré, K., *Simulation thermomécanique du laminage circulaire. Developpement d'une formulation quasi-Eulérienne tridimensionnelle sur une architecture parallèle*. 2001, ENSMP: Sophia-Antipolis.

40. Srikanth, A. and N. Zabaras, *Shape optimization and preform design in metal forming processes.* Computer Methods in Applied Mechanics and Engineering, 2000. **190**(13-14): p. 1859-1901.

41. Krok, J., *An extended approach to error control in experimental and numerical data smoothing and evaluation using the meshless FDM*, in *Meshfree Computational Mechanics*, Anonymous, Editor. 2002. p. 913-945.

42. Breitkopf, P., A. Rassineux, and P. Villon, *An intruduction to Moving Least Squares Meshfree Methods*, in *Meshfree Computational Mechanics*, Anonymous, Editor. 2002. p. 825-867.

43. Liszka, T. and J. Orkisz, *The finite differences method at arbitrary irregular grids and its application in applied machanics*, in *Computers & Structures*. 1980. p. 88-95.

44. Liszka, T., *An interpolation method for an irregular net of nodes.* International Journal for Numerical Methods in Engineering, 1984. **20**: p. 1599-1612.

45. Zienkiewicz, O.C. and J.Z. Zhu, *Superconvergence and the superconvergent patch recovery.* Robert J. Melosh Medal Competition, 1995. **19**(1-2): p. 11-23.

46. Zienkiewicz, O.C. and J.Z. Zhu, *The superconvergent patch recovery (SPR) and adaptive finite element refinement.* Computer Methods in Applied Mechanics and Engineering, 1992. **101**(1-3): p. 207-224.

47. Brancherie, D. and P. Villon, *Diffuse approximation for field transfer in non linear mechanics*, in *Revue Européenne de Mécanique Numérique*. 2006. p. 571-587.

48. Zienkiewicz, O.C., B. Boroomand, and J.Z. Zhu, *Recovery procedures in error estimation and adaptivity Part I: Adaptivity in linear problems.* Computer Methods in Applied Mechanics and Engineering, 1999. **176**(1-4): p. 111-125.

49. Boroomand, B. and O.C. Zienkiewicz, *Recovery procedures in error estimation and adaptivity. Part II: Adaptivity in nonlinear problems of elasto-plasticity behaviour.* Computer Methods in Applied Mechanics and Engineering, 1999. **176**(1-4): p. 127-146.

50. Boroomand, B., O.C. Zienkiewicz, and P.L.z.a.J.T. Oden, *Recovery procedures in error estimation and adaptivity: Adaptivity in non-linear problems of elasto-plasticity behaviour*, in *Studies in Applied Mechanics*, Anonymous, Editor. 1998, Elsevier. p. 383-410.

51. Bäker, M., *Finite element simulation of high-speed cutting forces.* Journal of Materials Processing Technology, 2006. **176**(1-3): p. 117-126.

52. Covington, J.L., E*xperimental and numerical investigation of tool heating during friction stir welding*, in *Department of Mechanical Engineering*. 2005, faculty of Brigham Young University.

53. Covington, J.L., et al. *Fundamental Characterization of Friction Stir Welding*. 2004.

54. Record, J.H., S*tatistical investigation of friction stir processing parameter relationships*, in *Department of Mechanical Engineering*. 2005, faculty of Brigham Young University. p. 87.

55. Oliphant, A.H., N*umerical modeling of friction stir welding: a comparison of alegra and forge3*, in *Department of Mechanical Engineering*. 2004, faculty of Brigham Young University. p. 133.

ABSTRACT

Despite considerable interests in the Friction Stir Welding (FSW) technology in past decades, the basic physical understanding of the process is still insufficient. Clearly, the complete understanding of the material flow around the rotating tool is crucial to the optimization of FSW parameters (including tool rotation rate, traverse speed, spindle tilt angle, and target depth) and design of tool geometry. Numerical simulation, conducted on these aspects of the process, can so contribute to the increase in weld quality and productivity.

This work presents the development of a numerical tool. An Arbitrary Lagrangian Eulerian (ALE) formulation is implemented in the 3D FORGE3® F.E. software to simulate the different stages of the Friction Stir Welding (FSW) process. A splitting method is utilized: a) the material velocity/pressure and temperature fields are calculated, b) the mesh velocity is derived from the domain boundary evolution and an adaptive refinement criterion provided by error estimation, c) nodal and P0 variables are remapped. Different velocity computations and remap techniques are investigated, providing significant advantages with respect to more standard approaches. Improvement is also brought to the contact algorithm through a tool smoothing procedure. These proposed enhancements have been tested and applied on industrial cases. They allow for the entire FSW process simulation.

Steady state welding, but also transient welding phases are simulated, exhibiting good robustness and accuracy of the developed ALE formulation. On the first hand, friction parameters are identified using Eulerian steady welding state simulations by comparison with experimental results. On the second hand, one major interest of the ALE model being the possibility to simulate void formation at the tool/workpiece interface, the transient plunge and welding phases are modeled. Their simulations can thus help to better understand the mechanisms of the deposition process that occurs at the trailing edge of the probe in order to obtain sound and defect-free welds. Finally, the flexibility and robustness of the model allows the investigation of new tooling designs influence in the deposition process.

RESUME

Bien que le soudage par frottement malaxage ait suscité un intérêt croissant ces dix dernières années, les phénomènes physiques qui sont à la base de ce procédé sont encore mal connus. Clairement, la compréhension du flux de matière autour du pion de l’outil en rotation est crucial pour l’optimisation des paramètres du procédé (comprenant la vitesse de rotation, la vitesse d’avance, l’angle d’inclinaison de l’outil et sa profondeur de pénétration) et la conception de nouvelles géométries d’outils. La simulation numérique, portant sur ces aspects du procédé, peut ainsi contribuer à l’amélioration de la qualité des soudures et de leur productivité.

Ce travail présente le développement d’un outil numérique. Une formulation arbitrairement lagrangienne-eulérienne (ALE) est implémentée dans le logiciel 3D éléments finis FORGE3® pour simuler les différentes étapes du procédé de soudage par frottement-malaxage (FSW). Une méthode découplée est utilisée : a) les champs de vitesses, pressions et températures du matériau sont calculés, b) la vitesse de maillage est calculée à partir de l’évolution des frontières du domaine et d’un critère de raffinement adaptatif procuré via une estimation d’erreur, c) les variables nodales et P0 sont transportées. Différentes techniques de calcul de la vitesse de maillage et de transport des variables sont étudiées, apportant des avantages significatifs par rapport à des approches plus standard. L’algorithme de contact a également été enrichi par une procédure de lissage d’outil. Ces améliorations ont été testées et appliquées sur des cas industriels. Elles permettent la simulation du procédé complet de soudage FSW.

L’état stationnaire de soudage, tout comme les phases transitoires, sont simulés, montrant une bonne robustesse et une bonne précision de la formulation ALE développée. Dans un premier temps, la simulation de la phase de soudage stationnaire permet d’identifier, par comparaison avec des résultats expérimentaux, les paramètres de frottement. Dans un second temps, un des intérêts majeurs du modèle ALE étant la possibilité de simuler la formation de vide à l’interface outil/matière, la phase de plongée et des phases transitoires sont modélisées. Leurs simulations peuvent ainsi aider à mieux appréhender les mécanismes du phénomène complexe de déposition de matière qui doit avoir lieu à l’arrière du pion de façon à obtenir un joint de soudure correct, sans défaut. La flexibilité et la robustesse du modèle permettent enfin d’étudier l’influence de nouvelles formes d’outil sur ce phénomène.